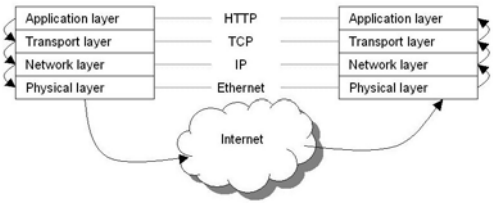
The fundamentals of TCP/IP networking

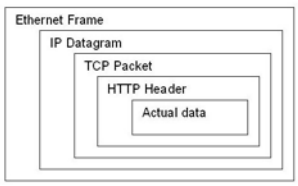
TCP/IP (Transmission Control Protocol / Internet Protocols) is a set of networking protocols that is used for communication on the Internet and on many other networks. TCP/IP is often referred to as a *protocol stack* because it consists of a series of protocols that intercommunicate in a layered model.

Within a layered/stacked approach, each layer has its own responsibilities and can work more or less separate from all the other layers. It helps to make networking independent of hardware, OS and applications, and it allows to see ongoing communication on different levels. When dealing with applications only, one can simply look at the web client / web server layer with no regard to the underlying networking and communication protocols. Similarly, when looking at lower levels, there is no need to know details of application data being pumped around in the network.



If an application wants to send a packet of data to a peer on another machine (for example a web browser sending a request to a web server), the data is passed down the protocol stack until the physical network is reached. Within this process the data sent by an upper layer will be encapsulated within a data packet of the layer underneath it.

- The browser creates a request in the Hyper Text Transfer Protocol (HTTP)
- This request is encapsulated in a Transmission Control Protocol (TCP) packet
- The TCP packet is encapsulated in an Internet Protocol (IP) packet
- The IP packet is encapsulated in an Ethernet frame and sent to the network.



On the other end of the connection, the reverse applies:
- The incoming Ethernet frame is found to be an IP one and passed to the IP layer;
- Here the packet is found to be a TCP one and passed on to the TCP layer
- The TCP layer finds the packet to be HTTP and passes it to the web server.

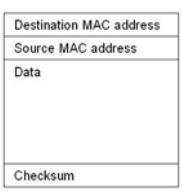
Physical Layer: Ethernet

An Ethernet is traditionally a network that all connected hosts can access using techniques for multiple access and collision detection and avoidance. In modern days this still applies to wireless networks, which offer a more literal Ethernet than ever.

All hosts in an Ethernet have a 48-bit address that is hard coded in the networking hardware. It is called the MAC address (Media Access Control) and usually written in 6 pairs of hexadecimal digits separated by colons or dashes like: **00:30:BD:12:34:56** or: **00-11-50-4B-2A-5C**

MAC addresses are assigned by the hardware manufacturer and therefore do not tell anything about location of the host in the network. Packets on an Ethernet are called frames. In a simplified model, they contain the source and destination MAC address of the packet, the actual data (which in itself can be a packet in another networking protocol such as IP and a checksum.

Traditionally, an Ethernet had a "bus structure" where all hosts were connected to one coaxial cable. Later the networks were built in a star structure with all the hosts individually wired to a hub or switch using Unshielded Twisted Pair (UTP) cabling. In the traditional Ethernet only one host could transmit at any one time and techniques of carrier sensing and collision detection were used to prevent and correct collisions. In a wireless Ethernet similar techniques are used to avoid collisions.



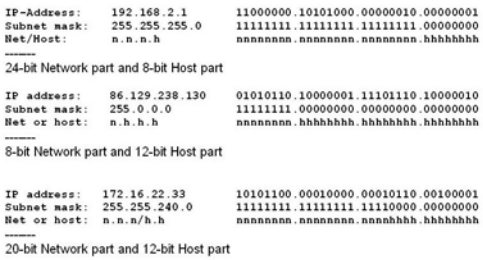
Ethernet Frame

Network Layer: Internet Protocol (IP)

The Internet Protocol is used for communication between hosts on a wide variety of networks. All hosts using the Internet Protocol have a 32-bit address, which is usually written as four decimal numbers in the so called “dotted decimal” notation: **192.168.2.1** or **86.129.238.130**

The IP address consists of two parts: the first part denotes the network, the last part the host. The size of both parts is given by the subnet mask, a 32-bit number assigned to any host. The subnet mask is a number that consists of one or more binary ones followed by 0 or more binary zeros. If a position in the subnet mask is a one, the corresponding bit in the IP address belongs to the net-part; if the bit in the subnet mask is a zero, the corresponding bit it the address belongs to the host-part.

The figure on the right shows three examples of IP addresses and subnet masks. In the first one, 24 bits, the first three digits of the address, are used for the network and one, the last digit for the host. In the second example this is the other way round: just 8 bits or one digit is used for the network, and 24 bits or 3 digits for the host. The last example shows that de separation between the network and the host does not always occur on a digit level: the third digit is used in port for the network and in part for the host.



A third important part of IP addressing is the default gateway. The default gateway is a host in the local network that will handle traffic that goes outside it, for example a router. If a host wants to send an IP packet, it first checks the destination IP address against its own and the subnet mask. If the address is in the same network, i.e. if the network-part of the addresses are equal, the packet is sent direct to the host. If it is in another network, the packet is sent to the default gateway, in order for it to be passed on.

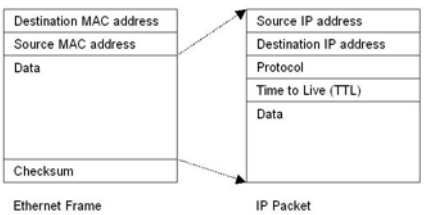
IP Packets

The figure on the right shows (a simplified version of) the layout of an IP packet. The packet contains the source and destination IP address, the actual data itself (which usually is a packet of a higher level protocol such as TCP or UDP) and some additional fields. The most important of these are the **protocol** field that determines which higher level protocol the packet belongs to and **Time To Live** that is used to prevent undeliverable packets from floating around on the network forever. More info on this will follow.

Using IP over Ethernet

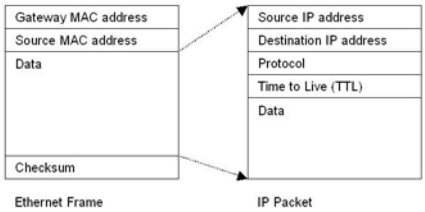
As said above, a packet to a host on the same network can be sent direct to that host. In order to do so over an Ethernet, we need to know the MAC address of that host. A MAC address can be obtained by sending an ARP (Address Resolution Protocol) over the network. An ARP request is basically a broadcast to the network asking all hosts “Which one of you has got this IP address”. The host with the requested address will answer to the MAC address of the sender. The sender now knows the MAC address of the destination and can send the packet. All hosts keep a cache of IP addresses and corresponding MAC addresses.

An IP packet sent to a host on the local network will contain Source and Destination IP address, the data and the other info. It will be encapsulated in an Ethernet frame containing Source and Destination MAC addresses and be sent on the wire.



When we need to send a packet outside of the local network, we will send it to the default gateway which will pass it on on our behalf. The sender therefore does an ARP find the MAC address of the default gateway rather than that of the destination.

The IP packet will be the same as in the previous diagram, the Ethernet frame will be different though: the destination MAC address is the Gateway's rather then the destination hosts.



Classes of IP addresses and special addresses

An IP address is a 32-bit address usually written in dotted decimal notation. The subnet mask determines which part of the address is the network and which part the host. The IP address space is divided in several classes of addresses with default subnet masks for them. The classes are:

Class	First addr.	Last address	Default subnet mask
A	1.0.0.0	126.255.255.255	255.0.0.0
B	128.0.0.0	191.255.255.255	255.255.0.0
C	192.0.0.0	223.255.255.255	255.255.255.0
D	224.0.0.0	239.255.255.255	Used for multicasting

Higher numbers are “reserved”. Within these ranges, there are some special address ranges:

The **loopback** network addresses, 127.0.0.0 – 127.255.255.255 always point to the local host. These addresses, usually only 127.0.0.1 is used are used to address services on the local machine.

The **automatic personal address** range, 169.254.0.0 – 169.254.255.255 is used by MS Windows and many other operating systems to assign hosts an IP address if there is no other way to do this.

IP addresses where the host-part contains all binary zeroes or all binary ones are reserved. The all-zero address is the **network address**, the all-ones address is the **broadcast address**. For example, in a network that uses the 192.168.2.x range with subnet mask 255.255.255.0, the network address is 192.168.2.0, and the broadcast address 192.168.2.255.

Unroutable addresses are special address ranges that are to be used in local networks only, they should not be forwarded to any public network. Because of this property the same addresses can be used over and over again in different networks helping tremendously to prevent an IP address shortage. Unroutable addresses are:

Class A: 10.x.x.x
Class B: 172.16.x.x – 172.31.x.x
Class C: 192.168.x.x

Glueing the network together: Hubs, Switches and Routers

A Hub is a simple device to connect a number of hosts. Hubs are **repeaters**: they echo traffic coming in on either of their ports to all the other ports without regard to the content of the packets.

A Switch is a hub with some intelligence built in. Switches monitor the MAC addresses of the packets they receive and use the source MAC addresses to determine which hosts are connected to which ports. A switch is a bridge: packets that are known not to be of any interest to a port given the MAC addresses of them, will not be echoed to that port. This can tremendously speed up networking.

A router is a device connected to two or more networks, that forwards IP packets between these based on the destination IP address in order for them to reach their destination. Typically, if a packet is sent from one host to another at the other end of the world, it will pass many routers. Each of these will forward the packet to the next one, called the next hop, in a best effort system to deliver the packet.

Because of this "best effort" approach there is no guarantee that a packet can actually be delivered. Furthermore, there could be occasions where, as a result of a configuration error, a packet ends up in a circular route. To prevent undeliverable packets from being forwarded between routers forever every IP packet has a field called **Time To Live (TTL)**. The sending host sets TTL to an initial value, for example 64. Every router that handles the packet will decrease TTL by one. Should the situation arise that TTL reaches zero, the packet is discarded as undeliverable and an error message is sent back to the sender.

Internet Control Message Protocol (ICMP)

ICMP is one of the simplest High Level Protocols on top of IP, its protocol number is 1. ICMP is used for sending error messages through the Internet, such as the error message in the paragraph above indicating that a packet is undeliverable. Another very common and popular use of ICMP is the ICMP Echo function that is used by the well-know **ping** program. Ping sends a packet to a host requesting to echo it back. This can be used to check if a host is "alive".

Another popular utility, **traceroute** uses a clever combination of IP's TTL field and the ICMP Echo and ICMP Error functions to determine the most likely way a packet will travel to it destination. What Traceroute does is principally this. First ping the destination with a TTL of 1. The packet will be discarded by the local router when it decrements TTL and it will send an ICMP error back, from which Traceroute knows the first hop's IP address. Now Traceroute will ping the destination again with a TTL of 2. The local router will pass it on to the next hop with a decremented TTL of 1. The next router will now decrement TTL to 0, discard the packet and send back an ICMP error message revealing its IP address as the next hop. Now, Traceroute will ping again with a TTL of 3, and a TTL of 4, and so on, until the destination is reached.

UDP and TCP

The most commonly used protocols on top of IP are the **Universal Datagram Protocol (UDP)** and the **Transmission Control Protocol (TCP)**. We'll have a look at them in this section. Beware that all the protocols discussed so far are datagram protocols. The network will do its best to deliver IP packets, but there is no guarantee that it will succeed.

UDP builds on the basic IP datagram service to provide a form of it that is usable for applications in the higher layers, TCP extends the basic datagram protocol to provide **reliable connections** with mechanisms for error correction and re-transmission of lost packets. A simplified version of the packet layout is shown below.

UDP and TCP introduce **port numbers**. These are used by applications. If a client sends a packet to a server, the Destination Port number tells the server which service is wanted or to which application the client wants to connect. All well known TCP/IP applications have their own well known port number. Think of a server as a large row of doors and of the port numbers as the numbers on the doors. The specific service you want, e.g. FTP, determines on which door you need to knock (in this case 21) to ask for the service.

Well known port numbers include:

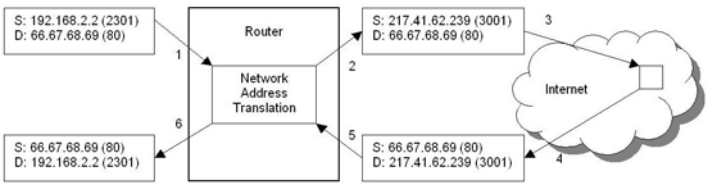
21	FTP	File Transfer Protocol
22	SSH	Secure Shell
23	Telnet	
25	SMTP	Simple Mail Transfer Protocol (for sending mail)
53	DNS	Domain Name Service
80	HTTP	Hyper Text Transfer Protocol (used for the world wide web)
110	POP3	Post Office Protocol (for receiving mail)

The source port number is assigned and used by the local host. If you request a web page with two images on it, your browser will create three connections to the web server: one for the page itself and two for the images. All have the destination port set to 80 (HTTP) and the source port to some arbitrary number. The replies from the server will have the port numbers reversed (source port: 80, destination port: the source port of the request). The protocol stack uses the destination ports of the incoming replies to determine which requests they belong to.

Most of the applications listed above use TCP as their protocol, some use UDP and some can be used with both.

Network Address Translation (NAT)

Many local networks use unroutable addresses for two main reasons: hosts in the local network are not visible and not reachable from the outside world, offering a first level of protection, and the same unroutable IP addresses can be used on many different locations, preventing a shortage in public IP addresses.



In order to be able to use unroutable addresses internally, the router must provide a function called Network Address Translation, NAT; this is a major feature of many routers for home and small business use, so-called gateway routers.

When using a gateway router with NAT, hosts in the local network get assigned an unroutable address. In the case of Belkin routers, this will be in the range 192.168.2.x, the LAN side of router itself will be 192.168.2.1. The WAN side of the router gets assigned an IP address by the ISP, for example 217.41.62.239. When the hosts in the LAN create connections to hosts on the Internet, the router will perform Network Address Translation by substituting its own, routable, WAN IP address for the client's unroutable address as follows.

Client (192.168.2.2) sends HTTP packet to server (66.67.68.69). The source port is an arbitrary number assigned by the OS, let's say 2301, the destination port is 80 for HTTP. Because the packet's destination is not within the local network, it is sent to the default gateway, the router. In performing NAT. the router takes out the client's unroutable IP address and substitutes it with its own routable WAN IP address. The source port number will also be changed to something assigned by the router's OS. The router will store the original connection data from the client and its substitutions for it in the NAT connections table. The request will then be passed on to the web server.

When the reply from the web server comes in, the opposite route is taken. The router locates the now destination port number 3001 in its NAT table and translates the server's reply back to the client on the original port number.

The Domain Name System (DNS)

For ease of use, in most cases hosts are addressed by their names rather than by their IP addresses. For that purpose, a hierarchical name system is being used in which host have names like **www.belkin.com**. From left to right, the name consists of the host name, the domain name and the top level domain name. Because domains can have sub domains names can have more than three parts.

A Domain Name Server does the translation between domain names and IP addresses. Whenever a host is addressed by its name, e.g. www.google.com, a request is sent out to the DNS server to find the IP address that belongs to the name.

The Domain Name System uses UDP port 53. Most routers include a DNS forwarding service that will forward DNS requests from their clients to the ISP's DNS server. This way there is no need for the client to change DNS server address if the ISP's address information changes.

Dynamic Host Configuration Protocol (DHCP)

In most networks, hosts are configured to obtain IP addresses and further address information automatically. This is done via the Dynamic Host Configuration Protocol (DHCP). When a computer starts, it will send out a broadcast message asking for the availability of a DHCP server. If a server is available, it will reply to it. The client will then send a DHCP configuration request and receive its address information. As part of the request, the client can send its MAC address and/or host name. The configuration information most of the times consists of: an IP address, the subnet mask, the default gateway and one or more DNS server addresses. An IP address obtained via DHCP has a “lease time” associated with it, a time during which the address is valid. When the lease expires, the client needs to renew it in order to continue to use it. In most implementations of DHCP the lease will be renewed when its time is half up.

Using Ping as a diagnostic tool

As mentioned before, **ping** is a program using the ICMP Echo protocol. It sends an ICMP packet to a host, requesting for the same packet to be returned. This can be used as a valuable diagnostic tool. First of all it can be used to see if a host can be reached (provided it honours incoming ICMP Echo requests). If this is not the case, ping can be used to find some common problems. Last but not least, by measuring the time it takes for the response to arrive, ping can give us an idea of the network's performance.

When using the ping command as it is implemented in Windows, there are roughly four different replies we can get to a ping. Let's say we ping one of the root name servers with **ping 4.2.2.1**

Reply from 4.2.2.1: bytes=32 time=76ms TTL=243. This is the preferred reply that tells us the host is reachable and responding. Windows sends ping packets of 32 bytes by default, the turn around time was 76 ms in this case and the TTL of the reply packet was 243 (most likely the sending host has set it to 255 which is the maximum).

Reply from 145.145.16.106: Destination net unreachable. A reply like this is in fact an ICMP error message from the host indicated. This is a router somewhere on the way to 4.2.2.1 telling us that it cannot reach the next network. This can point to a (temporary) problem somewhere along the route, it can also mean that the routing of ICMP Echo requests is not allowed.

Destination host unreachable. This is a message generated by the local operating system. It does not know how to reach a host. Most likely the default gateway has not been setup.

Request timed out. This can mean several things. When pinging a host on the local network it can mean that it does not reply to the ping or is switched off. When pinging a host outside of the local network something along the route is not replying. Best option in that case is start pinging the local default gateway, next ping the router's gateway, working your way further and further into the outside world.

The Mysterious MTU

Every network defines a maximum size of the packets that it can transport. In most of the cases this maximum, the **Maximum Transfer Unit (MTU)** is determined by the hardware used or chosen in a way to optimise the efficiency of the network. Sending larger packets will mean that more data can be sent with relatively less overhead, but it will also mean a larger re-transmission in case a packet gets lost. Common values for the MTU are 1500 for Ethernet, around 4500 for fibre optic connections and around 500 for dial-up connections.

The maximum packet size of IP is 64 kB, a packet of this size would be too large to handle for almost every physical medium. That is no problem in itself, because IP allows packets to be fragmented into multiple ones. This fragmentation takes place when the IP packets are encapsulated in the physical ones, in the step from the network layer to the physical layer. If, for example, on an Ethernet I use the command **ping 4.2.2.1 –l 1600** to send 1600 byte size packets to host 4.2.2.1, these packets will need to be fragmented as the MTU for Ethernet is 1500. Once a packet is fragmented it will not be re-assembled before it reaches its destination. For that reason fragmentation is to be avoided if possible.

When trying to reach a remote host, such as 4.2.2.1 not only the local MTU is involved, but also the MTU's of all network connections on the way to the destination. In fact, the maximum packet size for any connection is equal to the smallest MTU along the way. This value is called the **Path MTU** of the connection.

To determine the Path MTU of a connection, a host can send IP packets over it that have a special flag set, the DF (Don't Fragment) flag. As the name indicates, this flag prevents a packet from being fragmented. Should the packet arrive at a router that needs to forward it over a connection with a MTU for which the packet is too large, the router will discard the packet and return an ICMP error message to the sender indicating that the packet is too large for the next hop's MTU and mentioning what that MTU is. Basically the error message tells the sender something as "You sent me a packet with the DF flag of 1400 bytes, but my next hop will accept only 1000". Based on this reply, the sending host can retry with a smaller packet and determine the Path MTU, the smallest MTU along the way.

In this process things can go wrong. ICMP error messages are not always routed back to the sender correctly and may be filtered by firewalls along the way or at the sender's end itself. And even if the ICMP error arrives at the sending host, it may not be passed back up the stack to the point where the packet size was determined and the DF flag set. This, combined with the fact that the DF flag is often overused, can lead to problems with Path MTU negotiation and hosts sending DF-flagged packets that are too large for some point in the connection.

Typical symptoms of MTU problems are:

- All downstream traffic works fine: can download, receive e-mail, etc. etc without a problem
- Problems with upstream traffic sending (large) e-mail, uploading files with FTP etc
- Even more specific: uploading small files (200 bytes or so) works fine as does sending one-word e-mails
- Problems accessing some specific websites, most of the times sites where login is needed (things need to be sent); the same websites may be pingable without a problem
- VPN connections do not work properly

Most of the times, MTU problems can be solved by lowering the MTU on the (modem)-router and/or manually setting the MTU on the PC (using a program as Dr.TCP). To manually find out the Path MTU of a connection with a host that can be pinged use the following procedure:

ping host.domain.com –f –l 1600

Where the dash-f option sets the DF flag in the IP packet and the figure behind the dash-l option is the packet size. Experiment with the packet size until you get a reply to the ping. Then add 28 to it (the header overhead of ICMP) and set the resulting value as the MTU.

Please beware when using DrTCP to manually set the MTU on a Windows machine that you will need to restart the machine afterwards and that it is advisable (for LAN performance) to set the MTU the same on all computers in the network, even the ones that do not suffer from the problems.