CHAPTER 18 – INTERNET TECHNOLOGY AND NETWORKS

Introduction

This chapter briefly describes some of the most important issues in Internet technology and network management. It is concerned principally with how the Internet works, including how it differs from telecommunications networks, and with some of the technical issues that arise in discussions of internet services and governance. The structure of the Internet – *i.e.* the relationships between different actors in the Internet supply chain – and the services it offers to end-users are discussed in Chapter 19. Issues of Internet management and governance are discussed in Chapters 20 and 21.

Two things crucially distinguish the Internet from other communications media.

1. Firstly, it is a packet-based network, *i.e.* the way in which it transmits information between users is focused on the data that are distributed rather than on the connections between users. In particular, unlike traditional telephone connections, the links between Internet users do not require a dedicated channel between users to be set up before communication begins, or to be continuously open while communication continues.

○ Secondly, the packet-based nature of the Internet enables it to function as a network of more or less independent networks. The Internet is defined by the principles as well as the technology that holds these disparate networks together into a common global network.

Technical descriptions of the internet often focus on the specifics of technology, such as its multilayer stacked architecture, the interfaces between these layers, technical protocols, and the bits and bytes that define how the protocols work at a detailed level. Some of these issues are discussed in this chapter and/or elsewhere in this handbook.

While detailed technical discussion is useful in an introduction to network technology, it does not sufficiently explain the entities that hold the various networks together in a single Internet and which are crucial to understanding Internet policy. This chapter is therefore most concerned to describe the logical constructs that make the network work.

The underlying logical structure comes in two varieties: design principles and organisational constructs. The chapter describes the constructs briefly, and also gives a basic overview of the roles of code, protocols and standards. Firstly, however, it describes the Internet Protocol suite, commonly referred to as TCP/IP, and the fundamental layered architecture of the internet (although this architecture is often followed more in the breach then in actuality).

Basic Viewpoint

At a very high level, the mechanics of the Internet are quite simple. Computer systems and other networking entities (including telephones, PS3 systems and some household appliances, even refrigerators) can all be connected to the Internet. Each of these named entities can be found at an endpoint that sits at some location in the network. When they are connected, each must have an identity (name/number), which is globally unique. Specialised systems manage the movement of messages/data from one named entity to another by following routes that are usually discovered and selected by the network itself. In short, there are things with names that live at addresses and which send messages to each other along routes.

This works because the network is based on certain principles and uses code based on protocols that have been standardised. The fundamental protocols are included in a suite that is known as TCP/IP. Before describing them in more detail, it is useful to clarify the role of protocols, standards and codes within the Internet.

Protocols, standards and code

The rules, which govern the organisation of the Internet, are set out in protocols, standards and codes.

○ The term "standard" is used in a wide range of industries, to identify technical interfaces and specifications with which the designers of new products and services must comply. Standardisation has been particularly important in telecommunications networks, especially in enabling the interoperability of different networks, technologies and equipment. It gives formal or *de facto* authority to agreed approaches to technology development.

○ Within the Internet, the details of addressing, naming and routing are standardised in what are known as protocols. A protocol is the set of rules that determines the format and transmission of data. A protocol defines a generally loosely ordered set of instructions and defines the meaning and position of all of data within a message.

○ Code is the symbolic arrangement of data or instructions in a computer programme, or the set of such instructions that constitutes instantiation of a protocol. In short it is code that gives substance to a protocol and makes it a part of the Internet; and it is code that makes physical hardware interoperate.

There are many protocols used within the Internet. Two sets of protocols are most prominent: the TCP/IP suite of protocols which enable packet-forwarding and data delivery and are maintained by the Internet Engineering Task Force (IETF); and the HTTP, HTML and other protocols which underpin the World Wide Web and which are maintained by the World Wide Web Consortium (W3C).

There are many different ways in which protocols and standards can be created. While there is no rule that says that all Internet protocols and standards are created in exactly the same way, a common process has often been followed.[1]

In the process involved in the TCP/IP suite of protocol by the IETF, most often a need – technical-, service- or business-related – becomes apparent for which there is no existing protocol, or for which existing protocols are insufficient. Although this was not always so, a requirements, or framework, document is often written before a new protocol is developed to meet the need. Often a specification for a protocol is written and distributed through a set of public documents, called Internet Drafts, to any other person who is interested in a new protocol. If there is widespread interest in it, especially in a commercial environment, a decision may be taken to form a working group to work on the protocol and to move it in the direction of standardisation. Though a working group is not necessary, one is often set up.

[1] This is the model followed by the IETF which is responsible for most of the standards that make up the lower layers of the internet. A full explanation can be found in http://www.ietf.org/rfc/rfc2026.txt – The Internet Standards Process –Revision 3. . The W3C uses a different standardisation process.

Once a protocol has been developed it is tested before it can begin moving towards standardisation. In this case, testing means that several independent instances of the protocol must be created and tested against one another to demonstrate that they can interoperate. If they can do so, this is taken to mean that the description of the protocol is sufficiently clear for unambiguous implementation. If not, then further clarification is required before the protocol proceeds towards standardisation.

Since standards are meant to indicate that code implemented in accordance with a standard will work with other code implemented in accordance with that standard, this step – the writing and testing of code – has become one of the most important in the IETF process. As a standard and its protocol mature through public use it can progress from being a Proposed Standard to a Draft Standard and finally to Internet Standard status. These stages of standard reflect the degree of deployment and testing the protocol receives in the Internet.

TCP/IP

The term often used to refer to the protocol suite used in the Internet, TCP/IP protocols, is a historical reference as well as a reflection of current usage today. TCP – the Transmission Control Protocol (RFC 793, Std 007) – and IP – the Internet Protocol (RFC 791, Std 005) – were two of the first three protocols introduced as the new Internet developed in 1980/1981. The third original protocol was User Datagram Protocol (UDP, RFC 768, Std 006). IP, specifically IPv4 (IP version 4), and TCP still handle most of the network traffic; IPv4 effectively handles over 99.99% of the traffic at the Internet layer. While use of IPv6 (IP version 6) was still negligible in the Internet at the time of writing (mid-2009), it did figure into some research networks such as CERNET2 which is 100% IPv6. TCP handles somewhere between 90 and 95% of traffic in the transport layer, depending on where it is measured, with UDP handling somewhere between 6% and 9% of traffic. There are also other transport protocols, but these have little usage proportionally.

IP provides the central datagram functionality of the Internet. The basic principles involved are both simple and highly flexible. This is generally felt to have contributed substantially to the Internet's ability to absorb new technological opportunities and to innovate in the provision of services. IP basically encapsulates the datagram, or packet, with the source and destination addresses as well as information such as Type of Service, which gives an indication of how a packet is to be treated in terms of priority and queuing, total length of the datagram, "time to live" of the packet (*i.e.* how many hops it can take through the network before it should be discarded), a checksum for confirming that the information in the header has not been tampered with or accidentally changed, and a protocol identifier that tells the system the identity of the next encapsulation, most often the value 6 for TCP. There is also a flags field that gives indications of details such as whether a datagram can be fragmented into smaller packets if one of the networks transited requires it, and whether the packet has been fragmented.

The TCP header and protocol is much more complicated then IP or UPD and is still an active object of research study today. As indicated it is the most common transport encapsulation. While IP is responsible for the datagram, hop-by-hop nature of the Internet, TCP is responsible for establishing connections between two end-points. UDP, on the other hand only provides a minimal encapsulation for those upper layer protocols that do not require a connection between the end-points. TCP is also critical in helping to control congestion in the network by modulating the sending rate based on conditions picked up from the connection it establishes.

Both the TCP and UDP encapsulation headers include information about the source and destination ports. Ports are internal endpoints that identify the next level encapsulation of the packet, most often an application protocol. Each protocol has its own defined port, which is defined by IANA,[2] as are all protocol parameters. Additionally TCP contains information necessary for initiating a connection (sometimes called a data stream), SYN and ACK indicators, as well as the window size, an indicator of how much data the receiver is willing to have sent before the sender must wait for an acknowledgement that the receiver is willing to receive more data. This mechanism provides much of the congestion control mentioned above. The TCP header also includes sequence numbers so that the receiver can determine if it received all of the packets that belong to the stream. Packets can arrive in TCP out of order since the nature of the IP datagram layer is to send each packet on as best it can without any consideration of the other packets in a stream – IP has no indication of the stream or non-stream nature of the data it forwards. The TCP receiver, however, is responsible for ordering these packets on receipt before passing them on to the next layer.

Layered architecture

In the basic explanation to TCP and IP above, reference is made several times to 'layers'. The basic notion of layers involves the idea that a particular sort of task is dealt with by one protocol in an ordered set of protocols called a protocol suite. In contrast to the OSI 7 layer model, the Internet is sometimes discussed as having four essential layers above the hardware:

- An application layer that includes protocols network control protocols such as DHCP, DNS, NNTP, NTP; internet telephony protocols such as SIP, or MGCP; web protocols such HTTP and SOAP; email protocols such as IMAP4, POP3, and SMTP; management protocols such as SNMP; security protocols such as SSH, SSL and TLS, middlebox control protocols such as STUN; and the routing protocols such as BGP and RIP.
- A transport layer that includes: TCP, UDP, DCCP and SCTP.
- A network protocol layer that includes IPv4, IPv6 and ICMP
- And link layer protocols that allow access to the underlying physical layer such as Ethernet, WiFi and DSL

The services provided by the Internet rely on these protocols and the mechanisms provided by the layered architecture for progressive encapsulation of data received from the higher layer protocols.

In addition to the layered structure, several recent developments have made the actual Internet less structured in practice. There are many occasions where a protocol like GMPLS, used to control optical networks using an IP based control mechanism, is overlaid by IP, which in turn is overlaid by MPLS (which is used to create Virtual Private Networks (VPN)), which is in turn overlaid by the rest of the TCP/IP stack. Such layer inversion and layer stacking become more prominent as the complexity of the interconnect increases. Protocols like MPLS and IPSec (IP security protocols) create tunnels through the Internet that make many of its traditional elements based on strict layers inoperable.

These inverted and tunnel structures have been necessitated by some of the services required by users. The services delivered through the Internet, and the role of Internet Service Providers, are discussed in Chapter 19.

Routing

[2] IANA, The Internet assigned number Authority, is responsible for all names and numbers used n the internet. While dealing with domain names, it is answerable to ICANN while in ems of protocol numbers it is answereable to the Internet Architecture Board (see Chapter 20).

Routing is a complicated and esoteric field of network engineering. It is also crucial to the function of the packet/datagram-oriented Internet. Without routing of some sort, packets could not travel from their source to their destination.

Using some rules, some preset knowledge and a variety of methods, devices known as routers transfer packets from one part of the Internet to another one hop at a time. They do this by building tables that identify the direction a packet should take in order to reach another network, computer or person, very much like the road signs found at crossroads. To describe it simply, every time a packet enters a router, the router's programming checks its destination address against the table and sends that packet onward on a route that will most effectively move it towards its destination. After a packet is despatched by one router, it is received by another. The process repeats until such time as one of the routers passes the packet to its final destination.

Routing has been affected by the use of GMLS and MPLS and is involved in creating the map needed for the use of these protocols. Internet Service Providers and carriers are responsible for deploying and maintaining the routing infrastructure.

Design principles

Having looked at some of the details of the Internet protocols, we can now return to the theoretical constructs that have allowed this complex network to come into existence.

Design principles are engineering constructs that are used to guide system designers – in the case of the Internet, network system architects and protocol designers – in their work.

Much of the work involved in engineering, of all kinds, requires specialists to consider several possible solutions to a problem and select that which best satisfies a set of aims while meeting relevant constraints. Many factors affect this choice, including cost, ease of deployment and political sensitivities as well as technical feasibility. In order to achieve coherence, it is critically important that the principles that guide decisions are consistent throughout a system, regardless of who designs particular components or when those components are designed. The technology that constitutes today's Internet has been in development since 1980 (although some of the earliest relevant work was done as early as the 1960s, in the Arpanet, or even earlier – see Chapter 20). The TPC/IP-based Internet itself has been undergoing continuous evolution and development since the 1980s and is still subject to very rapid change today.

Four design principles are particularly worth bearing in mind when thinking about how the Internet evolves:

- packet-based networking
- the end-to-end principle
- the "hourglass" model
- and what has been called the Postel robustness principle.

These are described in the following paragraphs.

Packet-based networking

The possibility of packet switching as a network technology was first discussed by Paul Baran and Leonard Kleinrock[3] in the 1960s, as part of the Arpanet project to build a network that could survive catastrophic destruction of environments. It differs fundamentally in concept and structure from traditional communications networks such as those in telephony and broadcasting.

The PSTN networks that have provided the basis for telecommunications networks in the past (and still provide that for most today) require a centralised service to create and track the connections that are made between subscribers/users. In a packet-based network, by contrast, no continuously-open physical connections are made between source and destination subscribers by a centralised switching system. Instead, the information that is being transmitted is broken up into discrete chunks called packets or datagrams and is routed across the network using the best paths that are available at that instant, by hopping from one network connection point to another ("hop-by-hop routing"). Selection of routes is not predetermined, but done as and when a packet is transmitted. Instead of continuously-open channels, the Internet therefore makes use of opportunistic routing. This makes it much more robust than the PSTN because it can continue to transmit information when any particular link goes down.[4]

Packet switching also allows for the network to be built up in various areas as an emerging network. There is no need to conceive of a whole network being completed before any part of it is used. Rather, each group that is interested in building a network can build one and then find ways of connecting to others who are also building a network. While it is sometimes hard to see this original characteristic in today's global and commercial Internet, it did start as a collection of independent networks that were interconnected with one another, and this principle remains essentially true today.

The end-to-end principle

The end-to-end principle was first described in 1980 and has, to a large extent, also remained central to the architecture of the Internet. It is frequently cited in political arguments about the future direction of the Internet. Many use the end-to-end principle to support their views, though sometimes with different interpretations that do not necessarily reflect the original principle or its meaning.

In its simplest form, the principle suggests that the only elements that belong in the deepest layers of the network are those that are useful to all other parts of the network.[5] This has often been interpreted to mean that the specific functionality an application needs should be as close to the user as possible, in other words "at the edge or end of the network" – provided, of course, that this function is not also needed by other applications.

Another way in which this is sometimes expressed is the proposition that, in the Internet, "intelligence" is or should be "at the edges of the network". However, some Internet commentators would say that

[3] There are competing claims as to who first conceived the notions that are the foundation of the internet. Generally though there is agreement that Baran's work on packet switching and Kleinrock's research on queuing theory were instrumental in the creation of the Arpanet which was a precursor to today's internet.

[4] It should be understood that packet-based networks can support the creation of connections at higher levels of the system. Also, connection-oriented packets can support packet-based services – in fact many segments of the internet run over connection-oriented telecommunications networks. Additionally thee are several technologies today, such as MPLS, that use the packet based network to create path-oriented networks that bear a remarkable resemblance to connection-oriented networks.

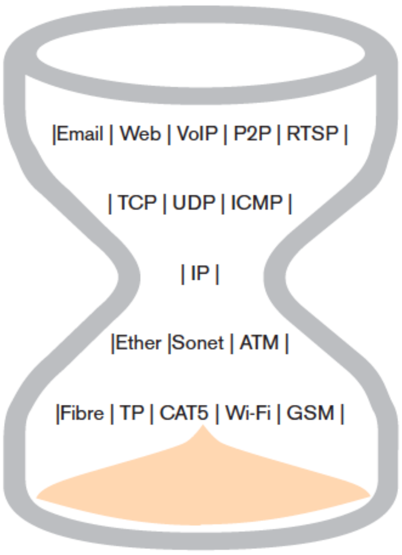
[5] The original article on the end to end design principle can be found at:
http://web.mit.edu/Saltzer/www/publications/endtoend/endtoend.txt.

this misunderstands the principle, which they say focuses on placing functionality at *the most appropriate place* in the network. If the function is most easily placed in the core and is useful to most or all of the network, then, they argue, it is not an infraction of the end-to-end principle to put it there rather than at the edge. For example, the intelligence needed to route messages from one network to another is placed in the core of the network without this being an infringement of the end-to-end principle.

The hourglass model

While rarely described as a principle, the "hourglass model" has been another central tenet in the design of Internet protocols (see above). Simply put, this is the design decision that places the Internet Protocol, IP, at the centre of an hourglass[i], as illustrated below.

Figure 18.1: The hourglass model


|Email | Web | VoIP | P2P | RTSP |
| TCP | UDP | ICMP |
| IP |
|Ether |Sonet | ATM |
|Fibre | TP | CAT5 | Wi-Fi | GSM |

According to this principle, all of the Internet's higher layer protocols converge into this one protocol, and all of the lower layer protocols fan out from it. The idea behind this is to have a common point in the protocol stack that allows for the addition of new connection technologies (such as WiFi and WiMax) and new applications (such as Voice over IP (VoIP) and IP television) without needing to change the basic network layer that guarantees the distributed connectivity of the internet.

Many commentators argue that the hourglass model has been a critical enabler of innovation in new applications and services for users through the Internet. One implication of the introduction of IPv6 (see below) is that it has widened the waist of the hourglass, such that now applications and link technologies need to have awareness of more then one network protocol, *i.e.* of both IPv4 and IPv6. This effect is compounded by the addition of multicast and quality of service functionality at the network layer.

Many writers have also suggested that the original hourglass principle is threatened by layer inversion such as layering MPLS over IP over GMPLS, and by the proliferation of tunnelling technologies in the core of the internet (see above).

The Postel robustness principle

This principle, originating with the internet standards pioneer Jon Postel, can be summarised as follows: "Be conservative in what you send and liberal in what you accept".[6] In the network sense it means that the utmost effort must be made to allow messages to continue their way across the system. By being as strict as possible in what a system sends, it attempts to be clear in its instructions and not give another system ambiguous information. On the other hand it also accepts that even when some other system is not as careful in the strictness of its messages, if there is any way to comply with the request within the security and stability constraints set by the system, the message should be processed.

While the robustness principle originated in the description of TCP, it has been applied to most of the protocols in the TCP/IP suite.

Organisational constructs

Having considered basic design principles, the following sections of the chapter look in turn at three fundamental organisational constructs of the Internet:

1. naming
2. addressing
3. and routing

Naming

Every system or network participating in the Internet has a name. These names are currently defined in a single distributed global naming framework called the domain name system (DNS).

The domain name system is a directory system that provides mapping between the name of a system or a service and the IP number by which and at which that named entity can be found. By referencing the DNS system with a name, the system gets back the number it needs to send datagrams of packets to the target system.

Management of the domain name system is the responsibility of the Internet Corporation for Assigned Names and Numbers (ICANN), together with regional and national Internet governance bodies. Governance mechanisms for the domain name system are described in Chapter 20. The following paragraphs describe a few technical issues associated with the DNS.

[6] The principle was first stated in RFC793, Transmission Control Protocol (the TCP of TCP/IP).

The DNS is a distributed address database available to all systems participating in the Internet. Its hierarchical structure is similar to that of the file hierarchy within a computer operating system such as Mac OS X, Linux, or Microsoft Windows.

Each level of a domain name defines another level in the hierarchy of a name. For example in the name www.apc.org (that of the Association for Progressive Communications):

1. .org is the top level domain name (TLD), designating the registry responsible for the root of this domain name;
2. .apc is a second level domain name, designating registered person or institution to whom this branch of the tree is assigned;
3. and .www is a third level name, identifying the location of the World Wide Web server in this network.

Specific pages within a website are located through additional strings of characters attached to this domain name. The unique web location address for a webpage or document is called its Unique Resource Locator (URL). For example, this handbook can be found on the APC website at the URL http://www.apc.org/en/pubs/books/apc-ict-policy-handbook-second-edition.

There are three varieties of TLDs:

- generic TLDs (gTLDs) such as .com, that are under the control of ICANN;
- country code TLDs (ccTLDs) such as .za (South Africa), that are mostly defined according to the ISO 3166 standard, which are independent of ICANN but may have a voluntary agreement with ICANN;
- and TLDs such as .mil, .gov and .edu, which are under US government, direct control.

At time of writing (mid-2009), there were sixteen generic TLDs governed by ICANN: .aero, .asia, .biz, .cat, .com, .coop, .info, .jobs, .mobi, .museum, .names, .net, .org, .pro, .tel and .travel. There were also 252 ccTLDs, of which over 90 participated in ICANN. Work was underway to open applications for the creation of more ICANN generic TLDs (see also Chapters 20 and 21).

The domain name system enables end-users of the Internet to access websites and other Internet resources using names (which are descriptive and easier to remember) rather than numbers (which are much more difficult for people to recall). In practice, however, protocols translate domain names into numbers in order to address resources on the Internet.

Whenever someone accesses a domain such as www.apc.org, her/his computer uses the Internet to request a translation from that name to its associated numerical IP address. To do this - unless the name is already known and cached on the computer or close to it on a network - it submits a request to one of thirteen named "root servers".[7] The root servers act as directories for top level domains (such as .org) and point to other servers at other levels within the domain name hierarchy in order to help find the IP address required. In the case of the Luleå University of Technology, for example, whose World Wide Web domain name is www.ltu.se, the root server will first find out the address of the .se name server, that is the registry database that has definitive information and references on all the second level domain names registered under the domain .se (the country level top level domain for Sweden). Once this is obtained, the address for the definitive server for ltu.se is requested. Once the address of the name server for ltu.se is obtained then the numerical address for www.ltu.se is returned to the user's system, and allows connection to the university server to be made. This numerical address takes a form such as 130.240.42.55 in IPv4.

The DNS does not appear limited in the number of names that can be stored. It has been limited, however, in that it has been capable only of handling names stored in a subset of Latin characters called LDH. This comprises the Letters a to z in lowercase form, the Digits 0 to 9 and the simple Hypen (-). Moving towards a more international domain name structure, including more characters and more alphabets, has been an important issue in Internet governance, and a method has been developed for handling more names in other character sets. This is referred to as Internationalising Domain Names in Applications (IDNA).

IDNA[8] is defined in a series of standards and informational documents which set out how a character string typed in the script of a non-LDH based alphabet can be transformed into a unique LDH string called punycode. In order to distinguish these IDNs in the DNS the punycode contains a prefix - a tag beginning xn--. Using this, any system can identify and differentiate between conventional LDH domain names and IDNs. An example may help: the Hebrew word for "master", בעל, could be used as part of a domain name. In this case the DNS entry for that name would be xn--5dbwr.[9]

While IDNs were not yet generally available for Top Level Domain (TLDs) names at the time of writing (mid-2009), they had been in use for some time for second level domain names, and it was expected that ICANN would make IDN TLDs available in the near future.[10] Work was continuing on both the policy issues and the technology required to make more non-Latin scripts available for domain names.

Addressing and routing

Internet addresses come in three basic forms: IP version 4 (IPv4) addresses, IP version 6 (IPv6) addresses, and autonomous system numbers.[11] Based on the information contained in these numbers, as well as other information that may or may not be used, a message is sent from one system to another system along a route determined by rules set in the routing system of the Internet. Most debates in this policy area revolve around the two varieties of IP address, though occasionally, AS numbers will also be raised in non-technical discussions.

Depending on how you look at it, an IP address points either to a single object, a network or a multitude of networks. As described above, every system on the Internet has at least one IP address.

[7] There are thirteen named root servers serving the world. These thirteen root servers are replicated in order to distribute the load and bring it closer to the users of the internet. While the number of replicated servers is constantly increasing, there are currently 144 root servers worldwide. More information can be found at: http://www.root-servers.org/.

[8] http://www.ietf.org/rfc/rfc3490.txt: Internationalizing Domain Names in Applications (IDNA). The protocol actually consists of several documents. In addition to this RFC which defines the protocol, the set also includes RFC3454, "Preparations of Internationalized String: also called Stringprep ; RFC 3491 Nameprep: A Stringprep Profice for Internationalized Domain Names"; and RFC 3492, Punycode: A Bootstring encoding of Unicode for use with Internationalized Domain Names in Applications. The current IDNA is limited to strings that were encode in Unicode 3.1. Unicode has continued to added scripts for new alphabets since then is currently working on Unicode 5.2. The IETF is working on an update of IDNA, called IDNAbis, which will be able to support current and future version of Unicode.

[9] A tool for translating non Latin based words into punycode can be found at: http://www.nameisp.com/puny.asp

[10] In some language groups, various techniques have been used to give the users the appearance of IDN TLDs, but these are mostly based on an ability in the applications to provide aliasing.

[11] AS numbers are used by the core routers n the Interest to describe the paths between networks. These numbers are not discussed n this article and are listed here for the sake of completeness.

Normally the address for a particular system takes a four number form separated by stops, *i.e.* the form such as 223.68.100.1. This address, however, can also be expressed as 223.68.100.1/16.[12] The /16 at the end of the address means that the first 16 bits, in this case 223.68, designate the address of the network where the system can be found. This means that routers use only the 223.68 part of the numerical string when looking up this address until the message arrives at the network designated by 223.68, at which point it looks up 223.68.100.1 within that network.

When IPv4 addresses were first created, the engineers who designed the system believed that it would provide more then enough addresses to meet any future requirements. After thirty years, however, addresses were already in restricted supply. This was due to the very rapid expansion of the Internet's user base and to the very considerable increase in the number of devices which can be connected to the internet and may require a separate IP address (computers, telephones, even domestic appliances).[13] While there are still many individual addresses left, these are no longer available in large number blocks. Two technical solutions have been offered for increasing the availability of addresses. One technical solution, which is widespread, is Network Address Translation (NAT). The other solution is IPv6. Additionally, efforts are underway to recover lost IPv4 addresses and discussions are ongoing about methods of allowing a market to develop in IPv4 addresses.

Network Address Translation (NAT)

For many years, several ranges of private addresses have been used by corporate networks and home networks. These addresses can only be used in one sub-network and may not be routed beyond this. Many readers, for example, will be are familiar with an address like 192.68.100.1, which is the default address found in most of the home routers sold on the open market.

While a very successful technology for allowing the Internet to grow in the face of IP address distribution problems, NAT has raised several challenges of its own. One of the most frequent complaints against NAT networks is that they interfere with the end-to-end nature of the network, because the system at the edge of a private network is responsible for translating the private address into a public globally unique address. As a result, many protocols have embedded these IP addresses in their messages, in itself possibly a breach of the end-to-end principle. On the other hand NAT technology has allowed the Internet to grow and can be said to keep translation at an edge as close as possible to the user.

However, NAT alone cannot solve the need for large countries with rapidly expanding Internet customer bases – such as Brazil, China, India and Russia – to have access to very much larger blocks of IP address numbers. This has created impetus for the deployment of IPv6.

IPv4 and IPv6

IPv6 will increase the number of addresses available and allow greater flexibility in their use. IPv6 addresses are longer and have a slightly different internal structure from those in IPv4. Because its addresses are longer, the IPv6 addressing system can be used to facilitate a greater number of systems without needing the NAT local addressing techniques necessary in IPv4. There is concerted effort among the Internet policy and some parts of the technical communities to foster a transition to IPv6.

A final point on addressing.

The meaning of IP addresses has historically been complex. They signify both the identification of the system and its location, referred to as overloading. In the days of the fixed Internet this was not much of an issue as the IP identity of a machine could easily be associated with its location, though it did create some problems for the routing architecture in terms of multi-homed systems. With the advent of the mobile Internet, where systems/devices move location, this has become much more of a problem. When a system/device moves from one location in the network to another or even from one network to another, it should not have to change its IP identity simply because it has moved to a different location. Research is underway on how to achieve decoupling of identity and location to suit this new environment.

Routing

The rudimentary principles of routing data through the Internet were described earlier in this chapter. Routing can either be static or dynamic.

- In static routing, the identity and location of every other router is configured into the system, allowing the router to produce a map of the network overall.
- In dynamic routing, protocols are used by the systems to discover paths through the network.

While there are many types of dynamic routing protocol, two types currently predominate: Distance/Cost Vector protocols and Link State protocols. Distance Vector protocols are most often used to connect one independent network, know as an autonomous system (AS), with another. They involve each of a pair of neighbouring routers informing the other about all the interconnections in the network of which it is aware. Border Gateway protocol (BGP-4) is the variant of this type of protocol used on the Internet today. In Link State protocols, most often used to describe the internal map of an autonomous system, each router in the network or subnetwork informs every other system in that network or subnetwork what it knows about all its neighbours.

Conclusion

All of the explanations in this chapter have been simplified in order to keep the content brief. The Internet is a rich and dynamic system that is constantly growing and changing. Due to the design principles and the organisational constructs described above, many people with varied interests can work on the network and produce results that can be used by others. The technologies that tie the network together – naming, addressing and routing – are dynamic, but they also form the core of what has enabled a collection of independent networks to become the Internet we know today. It is the standards that define these technologies that have enabled the loose association that is the Internet to hold together and provide the rich diversity of services with which Internet users have become familiar.

The principal author for this chapter was Avri Doria.

[12] This form of addressing was first defined in http://www.ietf.org/rfc/rfc1518.txt: *An Architecture for IP Address Allocation with CIDR,* and is still the fundamental organising structure for IPv4 addresses.

[13] While it is possible to assign an address to every possible object, the wisdom of doing so is being questioned by many Internet technical specialists. For example with a home, is it important that every device be globally addressd? or is it preferable that the control module be globally addressable with the devices themselves hidden from the outside network?