

IPv4 vs. IPv6
Subject: Engelsk B
Theme: The ip protocol

by F. Sørensen, 2.X HTX Aalborg

March 23, 2003

0.1 Preface

This paper is a description of the ip protocol version 4, ip protocol version 6¹, and why we are to deploy IPv6 as a standard instead of IPv4. There is also a IPv5 protocol², it is called the streaming protocol, but it has not been deployed very widely. It doesn't provide any of the breaking news as IPv6. This is why we aren't going to discuss it here.

To get an idea of what the ip protocol is, I am first going to give a short description of what it is.

The ip protocol is the part of a network that makes it all work. The ip protocol is the basic part of the Internet. It is the part that transfers all of the data around the network.

0.2 Problem formulation

In this paper I am going to give a description of how the IPv4 and IPv6 protocols work. This description will mainly be about how the addresses work, and what the ip protocols are, and what they do.

After that I am going to give a description of the reasons why we should change, from the old IPv4 to the new IPv6 protocol. I will try to describe possible problems that users and firms might have with the change and which problems they might face if they don't make the change to IPv6.

¹From now referred to as IPv4 and IPv6

²RFC 1819

Contents

- 0.1 Preface i
- 0.2 Problem formulation i

- 1 Intro to the IP protocols 1**
- 1.1 Overview 1
- 1.2 History 2
- 1.3 The protocols 2
 - 1.3.1 Addresses 4
 - 1.3.2 IPv4 4
 - 1.3.3 IPv6 4
- 1.4 Summary 5

- 2 Why is it a good idea to implement IPv6? 6**
- 2.1 Consider the future 6
 - 2.1.1 The users 7
 - 2.1.2 The firms 8
- 2.2 When are we to displace IPv6 8
- 2.3 Conclusion 9

- A The numbering systems 11**

- B ISO/OSI Network Model 15**

Chapter 1

Intro to the IP protocols

1.1 Overview

The ip protocol is the part of the Internet that makes it all work, it is the part that is responsible that the data we sent arrive at the right places. Without a network protocol no computers could communicate with one another. The ip protocol is in the OSI model¹ called the network layer. We all know that the number of computers connected to the Internet is constantly growing, which means that we are running out of IPv4 addresses. This is why we are starting to deploy IPv6 as a standard. This new version has several improvements over the old one.

The most important ones are:

- More addresses.
- Simpler layout.
- Smaller, meaning faster communication.
- Builtin security.

¹This model is invented by ISO www.iso.com see Appendix B

Most of the servers that make the backbone of the Internet² have at this time been converted to use IPv6³.

The ip protocol is used when you want to see a web page, when you receive e-mail, if you print to a network printer, if you access files that is on another computer than the one used. These are just some of the things that utilize the ip protocol. In the light of this we can conclude that the ip protocol is essential to the Internet.

1.2 History

This part builds on the information provided by [1]⁴ and [2].

The TCP/IP suite of protocols was developed by the Department of Defense in the sixties. It was developed to connect the "ARPANET"⁵. This network consisted mainly of university computers and computers at other research places.

The protocol and its popularity grew until it in 1981 became a network standard.

See the timeline in appendix C. Page 11.

1.3 The protocols

This chapter builds on the facts presented in [3], [4], [5] and [6]. The ip protocol(all versions) is a package switching protocol, meaning that all data sent and received are in packages. The ip protocol is a connectionless protocol. This means that ip protocol only sees the data as individual packages. This fact is not entirely true because to some extent IPv6 does have knowledge about a connection⁶, this is still experimental though.

²Part of the Internet that most Internet traffic travels through

³[1] page 474

⁴Page 2 "Problems we face"

⁵ARPANET was the precedence to the Internet

⁶[5] section 6

The computers communicate by using addresses. One way to think about the ip protocol is like the postal service. Every computer has a unique address just like our houses. This way it is possible to sent data to any computer if one knows the address. If the address sent to is not known(i.e. spelled wrong, not there or other things) one would either get a message back, that the package was refused, or a timeout⁷. The way the addresses are constructed depends on the version of the protocol.

	IPv4.	IPv6:
Address	192.168.100.100	3ffe:ffff:1000:f101:2100:a4ff:fee3:9566
Netmask	255.255.255.000	ffff:ffff:ffff:ffff:0000:0000:0000:0000
Net-id	192.168.100.	3ffe:ffff:1000:f101:
Node	.100	:2100:a4ff:fee3:9566

This table is explained later.

The IPv4 addresses consist of 32bits⁸ which means that we have 2^{32} ⁹ possible combinations. The IPv6 has 128bit addresses which means that we have 2^{128} ¹⁰ possible combinations. As we can see there are considerably more addresses in IPv6. This should give us a sufficient number of addresses so we don't run out in the near future.

When a computer wants to communicate with another computers it first finds out whether or not the other computer is on the same network. To do this it uses the netmask¹¹. If it is on the local network it transmits the data without delay. But if the computer is not on the local network it sends the data to the router¹² the router now is responsibil of making the data go to the designated place. If it can't accomplish this task it notifies the computer that sent the package.

The sole responsibility of the ip protocol is to sent and receive packages.

⁷If it take to long time to get a response ones computer will timeout

⁸A bit is a number written in the binary numbering system

⁹4,294,967,296

¹⁰340,282,366,920,938,463,463,374,607,431,768,211,456

¹¹How it does this is beyond the scope of this paper

¹²A device that transmit data between networks

1.3.1 Addresses

1.3.2 IPv4

The IPv4 addresses consist of an address and a netmask. The address consists of 4 numbers, each can range from 1 to 255. The numbers in the address consist of net-id and node. The net-id is the number that identifies the network. The normal way to write an IPv4 address is to use base 10 numbering system¹³. The net-id is the number that defines which network the node is on. The node is the number that identifies the unique computer. What the net-id and node are inside the address is specified by the netmask.

IPv4 address specs.	
Address	192.168.100.100
Netmask	255.255.000.000
Net-id	192.168.
Node	.100.100

If we change the netmask, the net-id and node will change accordingly¹⁴. The netmask is only used at the local network to decide if the address transmitted to is local or not. The sole purpose of the netmask is thus to identify if a computer is local or not. This means that any computer not on the local network will not know the netmask of the other part and doesn't need to. This is due to the fact that when a computer wants to send data to a non local computer the data is sent to a router and it takes it from there.

1.3.3 IPv6

The IPv6 addresses consist of an address and a netmask, just like the IPv4 addresses. The address consist of 8 numbers, each can range from 0001 to ffff in the base 16 numbering system¹⁵. The numbers in the address consists

¹³Normal decimal numbers

¹⁴See the first table

¹⁵See appendix A for details

of net-id and node just like in IPv4. The net-id is the number that identifies the network. The only difference from IPv4 are that the netmask and the address is notated in base 16 and that the addresses are longer.

IPv6 address specs:	
Address	1234:d5ef:764e:f619:ffff:5fff:5643:fe32
Netmask	ffff:ffff:ffff:ffff:0000:0000:0000:0000
Net-id	1234:d5ef:764e:f619:
Node	:ffff:5fff:5643:fe32

Like in IPv4, if we change the netmask the net-id and node will change accordingly. The netmask functions exactly the same way as in IPv4.

1.4 Summary

This chapter was a short description of how the ip protocols implement the addresses. How it transmits packages to other computers.

All of this happens transparent to the user. To make it even easier for us, we have a "name translation" service called DNS. This service has the assignment to translate the addresses we know i.e. www.aats.dk into ip addresses that the programs can use. Because of this, normal users will never have to operate with ip addresses directly.

Chapter 2

Why is it a good idea to implement IPv6?

2.1 Consider the future

First of all, we are running out of IPv4 addresses. This fact means that if we don't make the change, the cost of an IPv4 address will rise so much that all but big corporations can afford it. This fact will mean that the price one pays to the ISPs¹ will rise so high that many users would choose to not use the Internet. This will be a large step back for the global and free nature of the Internet.

Second, the ipv4 protocol was proposed as a standard with RFC791 in 1981. As we can see the standard is over 20 years old; this is extremely old when we talk about computer technology. This means that the "inventors" could not foresee that the size of the Internet would explode as it has, thus they could not know anything about many of the problems we face today.

One of these problems is that there is no security built in too the protocol itself² although this protocol have driven the whole Internet for many years!

Another problem is that the header³ is not that clean and thus has a

¹Internet Service Provider. i.e. Teledanmark, Telia

²[1] page 473

³The part of an ip package that defines the logistic

lot of overhead⁴, this has serious impact on performance. That there is sent more information than is needed has the effect that the device that handles the info becomes more complicated than needed, this means that the price will become higher. Although the overhead we are talking about isn't more than a few bytes per package this has severe impact on performance when there are sent several billion packages.

Because of the many parts of society that need on the Internet it would affect many persons and firms that use the Internet.

If we implement the IPv6 protocol it would help on some of these problems. The security part is perhaps the part that will be used the most, mainly because it will be possible to secure all communication on the Internet, not only the communication that is potentially sensitive.

2.1.1 The users

The users have at this time nothing to be concerned about because IPv6 is back portable. This means that when all of the Internet converts to IPv6 it will still be possible to use IPv4 addresses for some time. The reason for this is that the migration to IPv6 is probably going to take a long time and because of this it is a good idea that IPv4 and IPv6 can work side by side at the same time.

One of the big problems is that when one converts to IPv6, none of the programs that utilize the network will function. This is due to the difference in address notation. Current programs are programmed to use 32bits addresses, these programs will need to be reprogrammed to use the new 128bits addresses. Although this is a rather small problem for commercial programs that are upgraded frequently. It will mean that many of the minor programs that are made by private persons or firms that don't exist any more will instantly be rendered unusable if a user converts to IPv6. This is due to the fact that the addresses are different. The problems with the programs will without doubt confuse many of the non technical computer users. This

⁴[1] fig. 24.1 and fig.24.2

will result in chaos for the tek support of the ISPs and the firms that produce the programs when they have hundreds of persons that call at the same time.

Those are some of the problems that the normal computer user faces when we convert to IPv6.

2.1.2 The firms

The problems with programs will no doubt be a big concern for the firms because they will need to convert all of their network programs and hardware to be able to use IPv6. This is especially a problem for firms that have custom made programs that depend on the network. This fact will with no doubt make the firms wait until the costs of using a IPv4 network are more expensive than the costs to convert to IPv6.

2.2 When are we to displace IPv6

The IPv6 protocol is not going to be displaced for many years, this is due to the fact that it includes security and has an enormous amount of addresses. We are entering a future where a lot of devices⁵ will be able to use the Internet to order things they need. This means that things like the refrigerator, the coffee machine and allot of other devices will be able to make it more comfortable for us. This will of course make other demands to the protocol it uses.

This scenario could make a demand of a protocol that would be simpler and have less features than IPv6. It will then be the question if we need a simpler protocol or we will have the advance that IPv6 can already communicate with the Internet. The IPv6 has another advance, it can encrypt the data that are sent. The reason for this is that it has become more and more necessary to secure our personal data, that be social security number, what we eat, or which medicine we use. All of those parts of information is probably information we don't want to share with the world.

⁵Routers, Switches, Computers, Toast machines, Refrigerator, etc.

Because of these facts IPv6 has a great potential to be the protocol of choice, also because the devices that implement it will be compatible with many other devices, this has an impact on the price of the devices. If the devices implement a protocol that is widely and goodly understood, it will have a positive effect on the price level.

2.3 Conclusion

As we can see it will be a good idea, to change from the old IPv4 to the new IPv6. The reason for this is mainly that IPv6 has builtin security and has better performance than IPv4. It will lead to big problems in the future if we don't change to IPv6. These problems will be that the prices will go up, and that nobody will be able to afford to use the Internet. It will also in the future be possible that things like coffee machines will be connected to the Internet. This will make other requirements to the protocols, which will mean that we will need simple protocols like IPv6.

Bibliography

- [1] John Ray: *Special Edition Using TCP/IP*. ISBN: 0-7897-1897-9.
- [2] *A Brief History of the Internet* from www.isoc.org, 20/12-2002 (cf. appendix C).
- [3] *RFC 791* from www.rfc-editor.org.
- [4] *RFC 1340* from www.rfc-editor.org.
- [5] *RFC 2460* from www.rfc-editor.org.
- [6] *RFC 2373* from www.rfc-editor.org.

Appendix A

The numbering systems

This appendix is taken from:

<http://mccammon.org/articles/numbering.php>

Common Numbering System

Keith W. McCammon

A lot of folks struggle with these (myself included), so I figured I'd throw together a quick tutorial on the most common numbering systems encountered by in network-related fields. These are by no means thorough examinations of these numbering systems, but they should be sufficient to give you a working understanding of each, and to allow you to better understand other texts that make casual reference to these and others. Enjoy!

Base 10

Base 10, or decimal, doesn't seem as though it requires much explanation, as most of us use this on a day-to-day basis. However, it's helpful to touch on this, as the general principles of these numbering systems are the same—only the base changes, which you'll see as you read on.

Decimal systems consist of ten digits: 0, 1, 2, 3, 4, 5, 6, 7, 8, and 9.

When counting in decimal, we begin with a single place containing a 0, and increment upwards to 1, 2, 3, etc. Once the original place reaches a value of 9, we rotate that place back to a value of 0, and add another place to the left with a value of 1:

1 2 3 4 5 6 7 8 9 10

The value of the original place equals the value of the digit multiplied by ten to a power of zero, or $D \times 10^0$. The value of the next place (added to the left) equals $D \times 10^1$, and so on.

As an example, we'll look at decimal 1018:

10^3 10^2 10^1 10^0

1 0 1 8

The first place has a value of 8×10^0 (8). The second, 1×10^1 (10). The third, 0×10^2 (0). The fourth, 1×10^3 (1000). If you add up the place values ($1000+10+8$), you end up with 1018.

Base 2

Base 2, or binary, is one of the most common numbering systems that you'll encounter in networking, particularly if you're trying to make sense of subnet-related problems such as figuring host and/or network bits, bit swiping, etc.

Binary systems consist of two digits: 0 and 1. When counting in binary, we begin with a single place containing a value of 0, and increment it to a value of 1. Once the original place reaches a value of 1, we rotate that place back to a value of 0, and add another place to the left with a value of 1. As we continue, note that once all places have a value of 1, those places are all reset to 0, and a new place is added to the left with a value of 1 (decimal 7 in this example):

Binary 0 1 10 11 100 101 110 111 1000 1001 1010 1011 1100

Decimal 0 1 2 3 4 5 6 7 8 9 10 11 12

The value of the original place equals the value of the digit multiplied by

two to a power of zero, or $D \times 2^0$. The value of the next place (added to the left) equals $D \times 2^1$, and so on.

As an example, we'll look at decimal 224:

$$2^7 \ 2^6 \ 2^5 \ 2^4 \ 2^3 \ 2^2 \ 2^1 \ 2^0$$

$$1 \ 1 \ 1 \ 0 \ 0 \ 0 \ 0 \ 0$$

The first five places are multiples of zero, all of which obviously have values of 0. The sixth place has a value of 1×2^5 (32). The seventh, 1×2^6 (64). The eighth, 1×2^7 (128). If you add up the place values (128+64+32), you end up with 224.

Base 16

Base 16, or hexadecimal ("hex" for short), is another numbering system commonly encountered by networking types. Hex is used for all sorts of things, most notably network protocol header and data fields.

Note: When displaying or documenting numbers in hex, a common convention is to prefix the hex digits with "0x" for the sake of clarity. It is also common to display hex numbers in pairs, which is why you might see 0x01, instead of 0x1, even though they both represent the same decimal value.

Hex numbering systems consist of (you guessed it) 16 digits: 0, 1, 2, 3, 4, 5, 6, 7, 8, 9, A, B, C, D, E, F. It's a little odd to see A-F in a numbering system, huh? Not really, these letters actually represent decimal 10-15, respectively. When counting in hex, we begin with a single place containing a value of 0, and increment that place to a value of F. Once the original place reaches a value of F, we rotate that place back to a value of 0, and add another place to the left with a value of 1.

Hex 0 1 2 3 . . . F 10 11 12

Decimal 0 1 2 3 . . . 15 16 17 18

The value of the original place equals the value of the digit multiplied by sixteen to a power of zero, or $D \times 16^0$. The value of the next place (added to the left) equals $D \times 16^1$, and so on.

As an example, we'll look at decimal numbers 24, 212, and 11654.

$16^1 \ 16^0$

1 8

The first place has a value of 8×16^0 (8). The second has a value of 1×16^1 (16). Add these and you have 24.

$16^1 \ 16^0$

13 4

The first place here has a value of 4×16^0 (4). The second has a value of 13×16^1 (208). Add these and you have 212.

$16^3 \ 16^2 \ 16^1 \ 16^0$

2 D 8 6

The first place here has a value of 2×16^3 (8192). The second, 13×16^2 (3328). The third, 8×16^1 (128). The fourth, 6×16^0 (6). Add these and you have 11654.

©2002 Keith W. McCammon 12.30.2002

Appendix B

ISO/OSI Network Model

This appendix is taken from:

http://www.uwsg.iu.edu/usail/network/nfs/network_layers.html

The standard model for networking protocols and distributed applications is the International Standard Organization's Open System Interconnect (ISO/OSI) model. It defines seven network layers.

Layer 1 - Physical

Physical layer defines the cable or physical medium itself, e.g., thinnet, thicknet, unshielded twisted pairs (UTP). All media are functionally equivalent. The main difference is in convenience and cost of installation and maintenance. Converters from one media to another operate at this level.

Layer 2 - Data Link

Data Link layer defines the format of data on the network. A network data frame, aka packet, includes checksum, source and destination address, and data. The largest packet that can be sent through a data link layer defines the Maximum Transmission Unit (MTU). The data link layer handles the

physical and logical connections to the packet's destination, using a network interface. A host connected to an Ethernet would have an Ethernet interface to handle connections to the outside world, and a loopback interface to send packets to itself. Ethernet addresses a host using a unique, 48-bit address called its Ethernet address or Media Access Control (MAC) address. MAC addresses are usually represented as six colon-separated pairs of hex digits, e.g., 8:0:20:11:ac:85. This number is unique and is associated with a particular Ethernet device. Hosts with multiple network interfaces should use the same MAC address on each. The data link layer's protocol-specific header specifies the MAC address of the packet's source and destination. When a packet is sent to all hosts (broadcast), a special MAC address (ff:ff:ff:ff:ff:ff) is used.

Layer 3 - Network

NFS uses Internetwork Protocol (IP) as its network layer interface. IP is responsible for routing, directing datagrams from one network to another. The network layer may have to break large datagrams, larger than MTU, into smaller packets and host receiving the packet will have to reassemble the fragmented datagram. The Internetwork Protocol identifies each host with a 32-bit IP address. IP addresses are written as four dot-separated decimal numbers between 0 and 255, e.g., 129.79.16.40. The leading 1-3 bytes of the IP identify the network and the remaining bytes identifies the host on that network. The network portion of the IP is assigned by InterNIC Registration Services, under the contract to the National Science Foundation, and the host portion of the IP is assigned by the local network administrators, locally by noc@indiana.edu. For large sites, usually subnetted like ours, the first two bytes represents the network portion of the IP, and the third and fourth bytes identify the subnet and host respectively. Even though IP packets are addressed using IP addresses, hardware addresses must be used to actually transport data from one host to another. The Address Res-

olution Protocol (ARP) is used to map the IP address to its hardware address.

Layer 4 - Transport

Transport layer subdivides user-buffer into network-buffer sized datagrams and enforces desired transmission control. Two transport protocols, Transmission Control Protocol (TCP) and User Datagram Protocol (UDP), sit at the transport layer. Reliability and speed are the primary difference between these two protocols. TCP establishes connections between two hosts on the network through 'sockets' which are determined by the IP address and port number. TCP keeps track of the packet delivery order and the packets that must be resent. Maintaining this information for each connection makes TCP a stateful protocol. UDP on the other hand provides a low overhead transmission service, but with less error checking. NFS is built on top of UDP because of its speed and statelessness. Statelessness simplifies the crash recovery.

Layer 5 - Session

The session protocol defines the format of the data sent over the connections. The NFS uses the Remote Procedure Call (RPC) for its session protocol. RPC may be built on either TCP or UDP. Login sessions use TCP whereas NFS and broadcast use UDP.

Layer 6 - Presentation

External Data Representation (XDR) sits at the presentation level. It converts local representation of data to its canonical form and vice versa. The canonical uses a standard byte ordering and structure packing convention, independent of the host.

Layer 7 - Application

Provides network services to the end-users. Mail, ftp, telnet, DNS, NIS, NFS are examples of network applications.