Migrating to IPv6

KINTU ZEPHERNIA

KTH Information and
Communication Technology

Degree project in
Communication Systems
First level, 15.0 HEC
Stockholm, Sweden

Migrating to IPv6

Kintu Zephernia

2012-06-02

Examiner: Prof. Gerald Q. Maguire Jr.

School of Information and Communications Technology
KTH Royal school of Technology
Stockholm, Sweden

Abstract

Today hundreds of millions of users are interconnected by communication channels allowing them to communicate and to share information. These users and the devices that interconnect them are what constitute the Internet. The Internet is a network of networks with a myriad of computer devices, including smartphones, game consoles (handheld/stationary), IP televisions, tablet computers, laptop computers, desktop computers, palmtop computers, etc.

This network of computers flourishes because of careful planning and maintenance by Internet Service Providers (ISPs), backbone network operators, and others. An additional factor that enables the Internet to operate is the four logical layers of abstraction in the TCP/IP protocol stack. One of these layers is the layer responsible for the transfer of datagrams/packets from one host to another. This layer is known as the Internet Protocol (IP) layer. However, as originally conceived a 32 bit address was thought to be more than enough. The space of IP addresses was distributed among different regions rather disproportionately, driven largely by the numbers of addresses that were requested (ordered in time). Today after a series of inventions in the field (such as the world wide web) and a rapid growth in the number of devices that wish to connect to the Internet the available unassigned address space has largely been depleted.

Regions with large populations, but with few assigned blocks of IP addresses have begun to exhaust all their assigned addresses, while other regions face the same fate in a few months. The need for a larger address space was predicted years ago and the next generation addressing scheme was devised as part of the development of Internet Protocol Version 6 (IPv6). Countries such as China and India had few IPv4 addresses and they have been forced to transition to IPv6 rather quickly. Today a significant number of the users in these countries are unable to communicate over IPv4 networks. The purpose of this thesis project is to discuss the transition to IPv6 and the transition to this new addressing scheme. IPv6 provides a much larger address space, along with a number of additional improvements in comparison to the previous version of IP (i.e., IPv4). Despite the advantages of adopting IPv6, the incentive to transition is low amongst well established businesses, especially those in regions that received a considerable number of IPv4 addresses initially. Instead different techniques have been employed in these places to mitigate the problem of IPv4 address exhaustion. It is also probable that this reluctance is a way to keep competing businesses out of the market for a while longer. This thesis aims to facilitate the transition from IPv4 to IPv6.

Abstrakt

Miljontals användare är idag sammankopplade genom kommunikationskanaler som tillåter utbyte av information. Datornätet Internet utgörs av dessa användare och de enheter som sammanbinder dem. Internet är ett nätverk av nätverk med en myriad av olika datorutrustning såsom; spelkonsoler, smartphones, bärbara datorer, stationära datorer, handdatorer, även IPTV, kylskåp, tvättmaskiner, osv.

Detta nätverk blomstrar på grund av noggrann planering och underhåll av internetleverantörer, nätoperatörer och andra. En ytterligare faktor som gör det möjligt för Internet att fungera är de fyra logiska skikt av abstaktion i TCP/IP-protokollstacken, en standard för datakommunikation. Ett av dessa skikt ansvarar för överföring av datapaket från en ändpunkt till en annan. Detta skikt är kallad *Internet Protocol(IP) layer.* Ursprungligen ansågs en 32-bitars adress vara mer än tillräcklig. Dessa IP-adresser delades ut till olika regioner rätt så oproportionerligt till stor del beroende på antalet adresser en region begärt. Idag efter en rad uppfinningar inom området(såsom webben/world wide web) och en snabb tillväxt i antal enheter som önskar ansluta sig till Internet är det tillgängliga adressutrymmet i stort sett slut.

Regioner med stor befolkning men med få tilldelade block av IP-addresser har börjat göra slut på sina tilldelade adresser medan andra regioner står inför samma öde inom några månader. Behovet av ett större adressrymd sågs flera år sedan och nästa generations addresseringsschema utformades som en del av utveckligen, Internet Protocol version 6(IPv6). Länder som Kina och Indien hade ett fåtal IPv4-adresser och de har varit tvungna att övergå till IPv6 ganska snabbt. Idag kan inte ett stort antal användare i dessa länder kommunicera över IPv4-nätverk. Syftet med detta examensarbete är att diskutera övergången till IPv6 samt övergången till detta nya adresseringsschema. IPv6 ger en mycket större adressrymd samt en rad ytterligare förbättringar i jämförelse med den tidigare versionen av IP(dvs IPv4). Trots fördelarna med att övergå till IPv6 är viljan låg bland väletablerade företag, särskilt i regioner som mottagit ett stort antal IPv4-adresser från början. Dessa regioner tillämpar istället olika tekniker för att bromsa utmattningen av IPv4-adresser. Det är också troligt att denna motvija är ett sätt att hålla konkurrerande företag från marknaden ett tag till. Detta examensarbete syftar till att underlätta övergången från IPv4 till IPv6.

Acknowledgements

The author wishes to express sincere appreciation to Professor Gerald Q. Maguire Jr. for his assistance in the preparation of this manuscript.

Table of Contents

List of Figures

List of Tables

List of Acronyms and Abbreviations

Acronym	Definition
AfriNIC	African Network Information Centre
AH	Authentication Header
APNIC	Asia-Pacific Network Information Centre(one of the five Regional Internet Registries)
ARPANET	Advanced Research Projects Agency Network(the first packet switching network)
BGP	Border Gateway Protocol
CIDR	Classless Inter-Domain Routing
CPE	Customer-premises equipment
ESP	Encapsulating Security Payload
HAN	Home Area Network
IANA	Internet Assigned Numbers Authority
IP	Internet Protocol
IPsec	Internet Protocol Security
IPv4	Internet Protocol version 4
IPv6	Internet Protocol Version 6
ISP	Internet Service Provider
LACNIC	Latin America and Caribbean Information Centre
LAN	Local Area Network
LIR	Local Internet Registry
NAT	Network Address Translation
NSAP	Network Service Access Point
OS	Operating System
OSPF	Open Shortest Path First (a routing protocol)
POP	Point Of Presence
RFC	Request For Comment, documents describing research/innovation of the internet
RIP	Routing Information Protocol
RIPE	European IP Network Coordination Centre
RIR	Regional Internet Registry
TCP	Transmission Control Protocol
VLSM	Variable Length Subnet Mask

1 Introduction

This chapter introduces the problem that was studied in this thesis project and identifies the project goals, purposes, and intended reader; as well as the limitations. The chapter concludes with an outline of the rest of the thesis.

1.1 Problem statement

The growth of interconnected devices has been exponential for the past decade. Today approximately 5 billion devices are connected to the Internet. Cisco predicts that by the year 2015 the number of connected devices will be twice the global population, approximately 14 billion devices. [1] The Internet Protocol version 4, with its 32 bit address space, is clearly not able to directly address this number of hosts (even assuming that each host has only a single interface). Already more than a decade ago these growth rates were the incentive for the creation of version 6 of the Internet Protocol, IPv6.

As of February 2011, the central pool of IPv4 addresses has been completely allocated by the Internet Assigned Numbers Authority (IANA) to the Regional Internet Registries (RIRs). The five RIRs are slowly following suit with Asia-Pacific Network Information Centre (APNIC) having already run out of addresses to assign. The other RIRs are likely to follow in at most a year or two. [2] This rapid exhaustion was bound to occur as more and more devices are connected to the Internet. Moreover, this exhaustion was accelerated by a careless allocation early in the development of this network. For example, many entities (such as Internet Service Providers (ISPs)) received more IP address blocks than they required. Additionally, many of those assigned large blocks of numbers do not wish to return any portion of the addresses that they have been assigned. Even if address blocks were to be returned to the IANA, it is unclear if this would really solve the longer term problem – since the demand for addresses is growing. Even more of a problem would be that large numbers of small blocks would pose a problem for the backbone routers as the routing tables would have to be much larger than they are today (~170k entries[1]).

1.2 Purpose/goals

A solution to the problems mentioned in the previous section is IPv6. Unfortunately, a general lack of adoption of IPv6 is hindering new users from interacting with others via IPv6. This thesis seeks to convince the reader of the advantages of making the transition from IPv4 to IPv6. Additionally, this thesis includes a hands-on guide on how to adopt IPv6. The goal is to persuade readers to migrate to IPv6 *now*, rather than waiting for later. A secondary goal is to increase the number of end users who are asking their ISPs for native IPv6 service.

1.3 The reader

The target group of this thesis is mainly the students studying Information and Communication technology, along with my supervisor and examiner. However, sufficient background is provided to accommodate even those with little prior knowledge on the subject. Although a certain degree of knowledge about computer networks is necessary, references to further reading can be found throughout the text. Because the thesis is after all aimed at raising awareness, the largest possible audience is desirable.

[1] 171 377 Prefixes after maximum aggregation, in the Internet Routing Table as seen from APNIC's router in Japan as of 24 March 2012, for details see http://seclists.org/nanog/2012/Mar/851.

1.4 What have others already done?

Several books and articles have been authored on migration to IPv6, much work has been done to stabilise and standardise the IPv6 infrastructure. However, despite these efforts, very few entities have native IPv6 connectivity. Furthermore, the majority of end users are oblivious about the impending doom of IPv4 address space. In the following chapters I will concisely summarise related work and bring some new insight to even the least conversant in computer networking.

1.5 Method

This thesis will be based on an extensive literature study. The relevant textbooks, articles etc. will be researched mainly in the KTH library and the library's access to online databases. These databases include access to many highly ranked journals, articles and textbooks from around the world. We will in addition study several Internet Engineering Task Force Requests for Comments (RFCs) – as these provide the primary documentation of IPv6 and other Internet Protocols.

1.6 Outline

Before discussing the methods to perform the actual transition to IPv6, it is necessary to discuss what we propose to change to. The text starts by presenting some of the background to the problem, starting from IPv4 and continuing up to the current tandem implementations of IPv4 and IPv6. Chapters 1 and 1 will provide the base for our subsequent discussion in the later chapters of this thesis. Chapter 1 starts off with a presentation of the transition proposed mechanisms, along with what has and can be done by both ISPs and end users. Chapter 1 concludes with a discussion about on-going research and development in the field. Finally chapter 1 discusses the future and reviews the goals that we set out to meet.

2 IPv4

The IP layer of abstraction is mainly charged with delivering Internet Protocol (IP) packets from source to destination. In order to perform this task, the source and destination IP addresses are identified by unique fixed length addresses. In IPv4, a 32 bit numeric identifier was deemed sufficient when the Internet was created. However, as the Internet growth has been exponential it is clear that there is a need for a revision of the IPv4 addressing scheme. This chapter presents the engineering decisions that led to the current state, as well as describing the need for a new Internet protocol, IPv6. We will not delve deeply into the techniques that have been employed to delay IPv4 address exhaustion; instead we show the progression of events in order to better understand the proposed solutions. Section 2.1 introduces classful network addressing architecture, the first classification of IP addresses. This scheme supported few individual networks and clearly could not support the growing Internet. Sections 2.2 through 2.5 present schemes used to limit the wastefulness of classful network addresses. Finally the chapter is summarised in section 2.6, we see that because the rate of growth was so rapid *all* of these schemes were merely short term solutions.

2.1 Class-based addressing

IPv4 utilises a 32 bit number as a source or destination IP address. This number is typically divided into two parts, one part identifying the network and the other the host[2]. At the Internet's infancy administrative groupings called classes were decided upon to enable different size networks [3]. These classes differ in the boundary between the network number and host number [3]. The main classes A, B, and C, have 8bit , 16bit, and 24bit network prefixes respectively written as /8, /16, and /24 [3]. There was no concern for growth or address conservation and so initially addresses were allocated carelessly. Two problems emerged from this; organizations with medium sized needs were offered /16 addresses (supporting up to 2^{16}- 2 = 65,634 interfaces) if their needs were just slightly above /24 (supporting up to 254 interfaces) resulting in wastefulness and depletion of blocks of /16 addresses. When the same organisations were offered several /24 blocks, the result was an increase in the Internet backbone router's routing tables. Several techniques to delay IP address space exhaustion have been employed and these are discussed below. However, the use of these techniques led to a number of problems. These concerns are the origin of the next generation addressing scheme IPv6, as it offers a long-term solution to these problems.

2.2 Subnetting

Subnetting is the division of the 32bit IP address into an additional part known as the subnet number [3]. This process is shown in Figure 2-1.

Subnetting reduces the number of IPv4 addresses assigned by hiding the internal structure of the network [3] [4]. This enables a network to be internally divided into several different subnets that share the same network prefix. This method mitigates against address exhaustion as the network can allocate larger and smaller subnets as needed without needing to get /16 or /24 address block assignments. Since one address prefix hides an arbitrary number of subnets, the number of routing table entries in the backbone routers is reduced as well.

[2] While many textbooks refer to the IP address as identifying a host, it is actually identifying an interface to the IP network. The idea that this was a host address is rooted in the early days of the network where it was unusual for a device (other than a router) to have multiple network interfaces because the cost of each interface was very expensive.

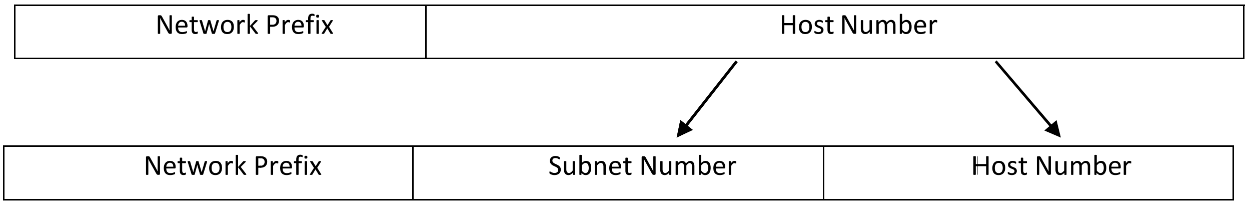


Figure 2-1: Dividing the host number into two parts: subnet number and host number

2.3 Variable Length Subnet Mask (VLSM)

Variable length subnet masking allows for the division of IP addresses into different sized subnets [4], unlike the same sized discussed in section 2.2, essentially subnetting a subnet. Because subnetting assumes that each subnet has the exact same needs of IP address space same sized subnetting becomes wasteful. To curb this waste of addresses more appropriate sized subnets were created by using variable length subnets.

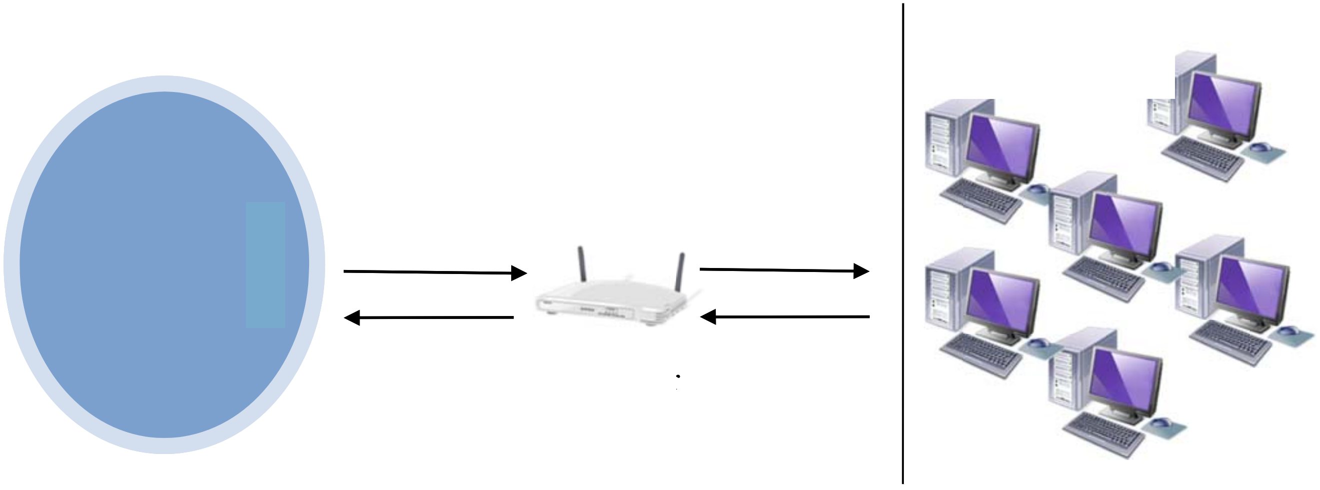
2.4 Classless Inter-Domain Routing (CIDR)

Classless Inter-Domain Routing (CIDR) gets rid of class-based network addressing allowing the network number to be of arbitrary length rather than the traditional 8bit, 16bit, and 24bit lengths [3]. This allowed for more efficient IPv4 allocation because organisations could be assigned addresses based on their actual need.

2.5 Network Address Translation (NAT)

Network Address Translation (NAT) enables multiple (internal) *private* IP addresses to be mapped to a single *external* IP address or a number of external IP addresses [5]. Typically a device such as a router which implements NAT represents the entire private network to the Internet as shown in Figure 2-2. The NAT device only requires one IP address to represent a number of interfaces outside the network they reside in.

There are many types of NATs, however their main feature is masquerading a subset of hosts using a single address or perhaps several addresses [5] [4] (the latter occurs when there are many simultaneous hosts within the private network that need a globally routable IP address). This is in effect a way to alleviate IPv4 address exhaustion. Unfortunately, NATs introduce a complexity to the network, especially in regard to a number of higher layer protocols [5] [4]. External hosts that wish to initiate communication with specific internal hosts cannot readily communicate to these internal hosts without these special tweaks to the firewall, such as adding static routing entries to the firewall. As a result, communication using several Internet protocols is hindered.



2.6 Chapter summary

Note that we have not dug deeply into these topics and only explain the gist of each one of them. From this overview some of the origins of the problem of IPv4 address space exhaustion should be fairly clear. Classful networking did not promote growth, because of its inherent inefficiency in regard to the allocation of address space. This has led to the different address space exhaustion mitigation techniques described in sections 2.2 through 2.5. These schemes come with variable efficiency, but each introduced new complexities. In hindsight, these efforts were clearly only short-term solutions as the rate of the Internet's growth continued, and thus each of these solutions is reaching the limits of its applicability. The major goal of these techniques was address conservation, such as hiding hosts inside private networks. Unfortunately, NATs not only delayed deployment of a long-term solution, but inevitably compromised the core principal of end-end connectivity within the Internet.

3 IPv6

The previous chapter described IP address space exhaustion mitigation techniques, each with their own drawbacks. These techniques were only short-term solutions to delay exhaustion, while more tangible solutions were sought. In this section we look at a long-term solution, the next generation addressing scheme, IPv6.

The steep growth of the Internet has determined the fate of the Internet Protocol. The Internet Protocol version 6 or IPv6 emerged amidst concerns about whether the Internet would adapt to increasing demands. IPv6 is now gaining momentum as the apocalyptic predictions concerning address exhaustion have been fulfilled. We start our study by identifying problematic areas in IPv4 and examining the solutions provided in IPv6.

3.1 Scalability

IPv6 utilises a 128 bit address. This increase in address length offers expanded capabilities because of the much larger address space. In fact theoretically 340,282,366,920,938,463,463,374,607,431,768,211,456 individual addresses are possible with a 128 bit address [6] [7].

An IPv6 address is denoted by eight groups of four hexadecimal digits with a colon as the delimiter. For example, the following 128 bits: 0010 0000 0000 1100 1101 1011 1010 0000 0000 0000 0000 … 0000 0000 0000 0000 0100 0011 0110 0111 1000 0101 0100 0001 are represented in hexadecimal as: 2001:cdba:0000:0000:0000:0000:4367:8541. For details of this hexadecimal encoding see Table 3-1.

A four digit group of zeros may be replaced by a single zero or altogether omitted and replaced with a double colon [6] [7]. Replacement with a double colon may only occur once so that consecutive groups of zeros can be replaced while avoiding ambiguity [6] [7]. Figure 3-1 below is an example of an IPv6 address illustrating these concepts. All representations are valid notations for the same address.

IPv6 addresses are classified into three groups, Unicast, Anycast and Multicast. A Unicast address identifies a single interface, a Multicast address identifies a group of interfaces and an Anycast address also identifies a group of interfaces [7]. Packets sent to a Unicast address are delivered to the interface that it identifies while packets sent to an Anycast are delivered to the nearest member of the group and packets sent to a multicast are delivered to all the interfaces that are identified by it. Broadcast addresses are not implemented in IPv6 as in IPv4 because the multicast addresses have this function [7].

The Unicast addresses are grouped into the following; Aggregatable global unicast addresses, Link-local addresses, site-local addresses, unspecified addresses, loopback addresses, compatibility addresses (for transitioning/coexistence purposes) and NSAP addresses [7].

The aggregatable global unicast addresses have a structure as shown in Figure 3-2 below. The routing prefix has three fields, the field Res is 8bits reserved for future use. The rest of the routing prefix field specifies routing and addressing hierarchy. The subnet ID identifies subnets and multiple levels of addressing hierarchy and lastly the interface ID specifies the interface [7]. These divisions allow for a very large amount of address space while only using a very small fraction of the total IPv6 address space.

The remaining unicast addresses have similar divisions and specific tasks to fulfil. They will in general be divided into a 64 bit routing prefix and a 64 bit interface identifier.

Addresses are classified into these groups for different routing procedures. The 128 bit address will have divisions depending on the group that the address belongs. These bits are then interpreted accordingly revealing their specific features.

Multicast addresses as mentioned above allow for packets to be forwarded to the entire network and are structured as in Figure 3-3 [7]. And finally Anycast addresses are essentially unicast addresses assigned to more than one interface.

Table 3-1: Number system conversion table

Binary	Hexadecimal	Decimal
0000	0	0
0001	1	1
0010	2	2
0011	3	3
0100	4	4
0101	5	5
0110	6	6
0111	7	7
1000	8	8
1001	9	9
1010	A	10
1011	B	11
1100	C	12
1101	D	13
1110	E	14
1111	F	15

Eight groups of four Hex. digits	leading zeros replaced with single digit	Zeros separated by colons replaced by double colon
2001:cdba:0000:0000:0000:0000:4367:8541	2001:cdba:0:0:0:0:4367:8541	2001:cdba::4367:8541

Figure 3-1: Illustration of IPv6 address notation

3bits	45 bits, routing prefix			16 bits	64 bits
001	TLA ID	Res	NLA ID	Subnet ID	Interface ID

Figure 3-2: Aggregatable global unicast address format

8bits	4bits	4 bits	112 bits
1111 1111	flags	Scope	group ID

Figure 3-3: IPv6 Multicast address format

3.2 Routing

Routing in IPv6 is accomplished using the same mechanisms as IPv4. The main protocols (RIP, OSPF and BGP) that are used to compute routing tables in IPv4 are used in IPv6 with modifications because of the 128 bit address field. Routers store information about routes to networks and other routers, therefore as more networks and hosts are added to the Internet these tables grow rapidly. By supernetting i.e. combining several networks with a common prefix, routing in IPv6 is handled more efficiently [4]. Routers will only store network aggregated addresses in their routing tables, thereby significantly reducing the number of routing table entries [4]. However, this depends upon spatial correlation in allocation of address prefixes- This spatial allocation is based upon the backbone topology and not simply based on geographic locality (in fact it might **not** be correlated with geographic locality).

3.3 Security

IP networks are susceptible to many kinds of attacks. For example, packet sniffing can be used to intercept sensitive information, such as passwords, credit card information, etc. [8]. Hence one of the required elements of IPv6 is IPsec. By using IPsec all information can be transported securely either end-to-end or from subnet to subnet. IP address spoofing, based upon forging a source IP address, can be used to gain access to otherwise denied information [9]. Similarly incorrect information could be sent that seems to come from a trustworthy communications partner. These and several other problems form the background for our discussion in this section. The importance of the security of a network cannot be stressed enough, especially given the ubiquity of monetary transactions over the internet.

In IPv6, IP security (IPsec) support is a requirement for *all* implementations of IPv6. IPsec is implemented using the Authentication Header (AH) and the Encapsulating Security Payload (ESP) [10] [6]. The Authentication Header (AH) provides data integrity and data authentication [10] [6], i.e. it protects the IP packets against undetected modification and packet spoofing. The ESP mechanism provides confidentiality through encryption of the IP packet [10] [6], ensuring that only the destination node can access the plaintext of the packet. Additional security features are provided by these extension headers and they can be used in whichever way suits the security needs. The large address space that IPv6 provides makes it very hard to randomly scan the network for vulnerable nodes.

3.4 Packet Header

An IPv6 data packet is made up of the payload and the header. A packet header stores the information necessary to route and deliver packets to their destination [11] [12]. The IPv6 header makes up 40bytes of the data packet and contains the fields shown in Figure 3-4 below. The IPv6 header has a revised design keeping only the useful features from its predecessor and moving features that are seldom used into optional extension headers. Although the larger IPv6 addresses add more data to the IPv6 data packet, they are simpler to route because the IPv6 header contains fewer fields than the IPv4 header. Below we examine the different IPv6 header fields.

<table>
<tr><td>Version
4 bits</td><td>Traffic class
8 bits</td><td colspan="3">Flow label
20 bits</td></tr>
<tr><td colspan="3">Payload length
16 bits</td><td>Next header
8 bits</td><td>Hop limit
8 bits</td></tr>
<tr><td colspan="5">Source address
128 bits</td></tr>
<tr><td colspan="5">Destination address
128 bits</td></tr>
</table>

Figure 3-4: IPv6 header format, figure adapted from IK1550 lecture notes [13]

3.4.1 Version

The version field is 4 bits like in IPv4 but will instead contain the number 6 in binary (0110) to indicate the version of the Internet Protocol [14]. Although this field is to allow both protocols to exist without conflicts, distinguishing the different packets is frequently done based on the protocol types in the data link layer.

3.4.2 Traffic Class

The 8 bit traffic class field replaces the Type of Service field in IPv4. The traffic class indicates the different types of IP packets and provides differentiated services [14]. A packet with a traffic class tag is distinguished at intermediate nodes for the application of the specified priority or class of traffic.

3.4.3 Flow Label

A flow is a series of packets that have some interconnection either by way of source and destination address, security, Quality of Service (QoS) or other parameters. The flow label field supports applications requiring per-flow handling [14]. Packets belonging to the same flow can be identified using this field and handled accordingly.

3.4.4 Payload Length

The payload length field is a 16 bit field indicating the length of the packet excluding the length of the IPv6 header [14]. The 16 bits allows for a maximum packet payload of 64 Kbytes however, an optional extension header allows a 32 bit length field which enables a payload of $2^{32} - 1 = 4294967295$ bytes known as a *Jumbo Payload*. If the jumbo payload extension header is desired, it is indicated by the value zero in the 16 bit payload length field [14].

3.4.5 Next header

The 8 bit next header field indicates the type of information that follows the IPv6 header [14]. For example determining the type of the transport layer packets, whether or not extension headers are present and what type of extension headers they are. The next header field is equivalent to the protocol field of the IPv4 header.

3.4.6 Hop Limit

The 8 bit Hop limit field is used to limit the number of intermediate hops [14]. The value is decremented at each node the packet goes through and upon reaching the value zero the packet is discarded. The 8 bits mean that the packet can go through a maximum of 254 nodes before it is discarded. Packets that are stuck in infinite loops because of incorrect routing table values can then be removed. The Hop limit field is similar to the TTL field in IPv4 except the router can decrement the field without recomputing the checksum. The checksum does not exist in IPv6 to protect the IP header, this task is instead transferred to higher layer protocols.

3.5 Extension Header

The IPv4 header has optional fields that are seldom used but nonetheless require processing by the routers. These optional fields are replaced by Extension headers in IPv6. The main IPv6 header (see Figure 3-4) has only the very essential fields that are required by a packet. Packets that require additional information can have this in an Extension header placed between the main IPv6 header and the upper layer header. The presence of an extension header is encoded in the Next header field of the main IPv6 header. In addition, the extension headers have a Next header field and so multiple headers can be chained together as illustrated in Figure 3-5. When more than one Extension header is used, it is recommended that they appear in the order specified by RFC 2460 [14].

IPv6 header Next Header = 43 (Routing)	Routing header Next Header = 44 (Fragment)	Fragment header Next Header = 6 (TCP)	fragment of TCP header + data

Figure 3-5: Chaining Extension headers (see RFC 2460 [14])

3.6 Autoconfiguration

Stateless Address Autoconfiguration (SLAAC), Stateful Autoconfiguration (DHCPv6) and static configuration are the methods employed to configure hosts on an IPv6 network. Configuring an IPv6 network statically means the network administrator has to manually configure the parameters of the related interfaces ensuring neighbouring nodes can reach one another. In section 4.9 we see an example of manually configuring a sample IPv6 network with Cisco routers and generic switches.

SLAAC is a new feature in IPv6 that enables hosts to be automatically configured, i.e. hosts assign a unique address to themselves and configure their IPv6 interfaces [15]. A host goes through a series of steps to accomplish this. First a link-local address is generated for the interface which must not already be in use therefore its uniqueness is tested. To test for the uniqueness a node will send requests known as neighbour solicitations asking whether an address is in use. The node receives neighbour advertisements containing the answers it needs. After uniqueness is verified the link-local address is assigned to the interface and IP connectivity is established with neighbouring nodes [15].

DHCPv6 is the Dynamic Host Configuration Protocol for IPv6. A DHCP configured node maintains a server with the necessary information a host may need to achieve IP connectivity this includes IP addresses from its stock of addresses and other configuration parameters [16]. The method of assigning addresses and other configuration parameters from a server that maintains a database is known as Stateful Autoconfiguration. DHCPv6 has certain improvements from its counterpart in IPV4. It does not dwell on backwards compatibility, the IPv6 link-local addresses can be used to send and receive instead of employing various system-dependent tweaks in order to obtain an address. Multiple interfaces may also be specified to a DHCPv6 server and all the interfaces are offered addresses simultaneously.

Although SLAAC offers a much simpler configuration process, it lacks the configuration of other parameters such as DNS domain, DNS server etc. DHCPv6 on the other hand offers a more comprehensive solution providing all the required parameters. However, the stateless approach can be used for its simplicity in address assignment and complemented with DHCPv6 where necessary. The Stateful approach could then be used when more control over address assignment is desired.

3.7 Mobility Features

IPv6 mobility is supported by Mobile IPv6, a communications protocol allowing mobile nodes to traverse networks while maintaining a fixed IP address. Each mobile node is assigned a permanent IP address known as the home address which remains unchanged along all the networks the node connects [17]. The mobile node will thus continue to be addressed by a home address belonging to a home subnet and have a so called care-of address representing its current location [17]. Mobile IPv6 uses the IPv6 Autoconfiguration features to obtain the care-of address and other routing information [17]. Furthermore, the IPv6 extension headers can be used to offer QoS technologies. Further details about Mobile IPv6 are addressed in RFC 6275 [17]. Mobile device users are increasing every day because of the convenient, always connected and very powerful devices available on the market. This increase is overwhelming the IPv4 internet which, though it offers mobility support does not have it inherent in its design. The IPv6 design however took into account mobility, it is not an add-on feature like in IPv4 and thus Mobile IPv6 implementations are more efficient than their IPv4 counterparts.

3.8 Quality of Service

Different applications in an IP network require certain specific amounts of network resources to function properly. We need to understand the different needs of the main categories of service in order to distribute the network resources appropriately so the different network applications can co-exist on the same network. For instance, voice communications over an IP network may generate a constant stream of traffic travelling through the network. Each voice user has relatively benign bandwidth consumption and their traffic is quite predictable. However, voice communications are very sensitive to delay and somewhat sensitive to lost packets. Video on the other hand is greedy and consumes a lot of bandwidth, but is not very drop/delay sensitive. Losing a few pixels is not very important while losing some voice packets could make a conversation completely unintelligible. Quality of service (QoS) is provided by various technologies that control how network resources are shared to adequately meet the needs of each service. QoS technologies recognise the type of data in packets and divides them into different priority traffic classes enabling critical flows to be served before others [18].

IPv6 is designed to be flexible enough to support QoS mechanisms through the IPv6 packet header and the extension headers. The IPv6 packet header has two fields that can be used for QoS namely, the Traffic class and the Flow label field. The traffic class field has 6 of its 8 bits representing differentiated services, by specifying a differentiated services code point (DSCP) -- see RFC 2474 [19]. Differentiated Services is a per-hop mechanism for prioritising bandwidth, i.e. every router would have certain routines to prioritise packets based on the value in the differentiated service field. The Flow label field can have its 20 bits used to request special handling by a router as specified in RFC 3697 [20]. Two extension headers can be used for QoS in IPv6 namely the Routing header and the Hop-by-Hop Options header specified in RFC 2460. The Routing header can be used to request a specific route and the Hop-by-Hop Options header can be used to send a router alert message to every router on a path indicating that the specified packet be processed.

3.9 Chapter Summary

From the discussions in the previous chapter and this chapter we have covered the main areas of concern. IPv6 supports many new features; in section 3.1 we looked at the address space offered by IPv6, the 128 bit address provides an extremely large number of addresses. The allocation of these addresses is done with much more consideration, because of the pitfalls experienced with IPv4. In section 3.2 we saw a solution to the routing table sizes that plagued the IPv4 network. In section 3.3 we saw that IPv6 supports network-layer security, by using the Authentication Header (AH) and the Encapsulating Security Payload to ensure authenticity, integrity, encapsulation and encryption of the IP packet. The security features provided by the extension headers can be tailored to suit the specific needs of the network. We also saw that IPv6 has a simpler header that reduces the header overhead, enables easy configuration of hosts, has built in mobility and offers better QoS support. These new features represent problem areas in IPv4. However, Ipv6 is not without shortcomings, these features are simply an important evolution of the Internet.

4 Implementation of Ipv6

Migrating to IPv6 will not require a sudden upgrade of all the hosts. This is in fact not feasible because such a transition in itself is a daunting task. Furthermore, unlike the case for the ARPANET, there is no single authority with the mandate to oversee such an operation. For many parties that are comfortably connected to the Internet such a move is both unwelcome and unprofitable. A more graceful approach has been employed where IPv6 is introduced to coexist with existing IPv4 implementations. This chapter presents this gradual process, along with the main actors who might be involved.

4.1 Dual Stack

The dual stack approach utilises two protocol stacks that operate in parallel, i.e. provides the hosts & routers with both IPv4 and IPv6 protocol stacks. The appropriate protocol stack is used together with encapsulation when necessary as the packets are forwarded from source to destination [21].

4.2 Translation

Translation is a mechanism that enables the forwarding of packets between hosts of different protocols. This allows the direct conversion of IPv6 to IPv4 where it is desired [21]. To facilitate this all IPv4 addresses have a corresponding IPv6 address, thus in some sense the IPv4 Internet can be seen as a subset of the IPv6 network.

4.3 Tunnelling

IPv6 tunnelling encapsulates IPv6 packets in IPv4 packets at the entrance node to the tunnel, and then sends them over an IPv4 infrastructure [10] [21]. The tunnel exit/destination node decapsulates the packet and forwards it as an IPv6 packet [10] [21]. The reverse is equally applicable, that is having IPv4 packets forwarded over an IPv6 network infrastructure. The routers/hosts will need to utilise both the IPv4 and IPv6 network protocols. Because of the dual stack the endpoints can at some point turn either of the protocols on or off.

Several of the main tunnelling schemes are described in more detail below:

Configured tunnelling An IPv6 packet is forwarded to a router/encapsulating node that must be configured [21] [10]. The packet is encapsulated to an IPv4 packet and tunnelled through the IPv4 infrastructure and finally delivered to the destination address [21] [10]. The reverse is equally applicable. We are generally creating encapsulated connectivity over one infrastructure to another.

Automatic tunnelling Dual stack routers and or hosts will communicate this way if the encapsulated packet is compatible with the infrastructure it is travelling over. If an IPv6 address is IPv4 compatible, then automatic tunnelling is used to deliver the packet [21] [10]. Configuration is not necessary since the appropriate address can be derived from the compatibility type.

6to4 Deliver IPv6 packets over IPv4 infrastructure without automatic or configured tunnels [21] [10]. Reserved IPv6 address space is used to provide IPv4 hosts with IPv6 connectivity. An IPv4 host is assigned an IPv6 address derived from the interface’s IPv4 address.

6over4 An isolated IPv6 host with no direct connection to an IPv6 router use an IPv4 multicast domain as the link layer [21].

4.4 Native IPv6

This requires acquisition of IPv6 addresses from an Internet Service Provider (ISP), just as one would or already is doing with IPv4. However, the gradual procedures discussed in the previous sections allow operators to develop a more robust IPv6 infrastructure and protect their IPv4 investment. This coexistence is the natural next step and will continue to exist until the devices with obsolete technology have been expensed & replaced and the ISP's IPv6 network is stable.

4.5 Internet Service Providers

In this section we discuss the different steps an ISP needs to take in order to enable users on the IPv4 internet to use IPv6 services. Below we look at the stages an ISP will need to consider as they work their way toward providing native IPv6 internet. A majority of the end user devices have for the past few years already been IPv6 compatible. Many networking devices (routers, switches etc.) already exhibit IPv6 readiness along with the main OSs (Windows, Linux etc.) that run on the majority of computer devices. The first steps for ISPs are discussed in subsections 4.5.1 through 4.5.3. These first steps are to ensure that their networks facilitate the transitional mechanisms. When this is accomplished native IPv6 can then be offered as we will see in subsection 4.5.4.

4.5.1 Apply for addresses

An ISP obtains a prefix allocation from the respective RIR after an application is submitted [22]. Several criteria need to be fulfilled, some of these requirements may be local to the RIR, but in general most fall under an international consensus [23]. A noteworthy difference from IPv4 address allocation is that the IPv6 address space is managed by IANA and only leased out to qualified entities.

4.5.2 Get connectivity

The Internet is a tiered hierarchy of networks, i.e. a network of networks. ISPs are at the top of this hierarchy and even these are divided into different tiers. Therefore, in order to route traffic globally ISPs will need to establish connectivity with upstream providers (i.e. ISPs on tier above) and peers (i.e. ISPs on same tier) [22]. This interconnectivity involves obtaining IPv6 transit, i.e. obtaining IPv6 routes and destinations from around the world.

4.5.3 Check for IPv6 compatibility

An ISP needs to make its backbone IPv6 capable through software & hardware upgrades [22]. The backbone network equipment has to be upgraded and IPv6 parameters (such as those within routing protocols) need to be configured [22]. ISPs have their own backbone which interconnects the ISP to other ISPs. These interconnection points are known as Points of Presence (POPs) [24]. Packets destined to the ISP's network are received at the POPs, thus the devices within the POP need to be upgraded to support IPv6 connectivity.

4.5.4 Native IPv6 service

The final step in the checklist for ISPs is to adjust their backbone to enable native IPv6 service. This means that their entire network infrastructure supports IPv6, unlike IPv6 capability which merely supports IPv6 through transition technologies. This is achieved when all the different parts of the network (i.e., computers, network devices (routers, servers, switches etc.), network services, routing infrastructure, etc.) completely support IPv6. Demand will generally dictate how fast an ISP transitions to this stage [22].

4.6 End Users

Migrating to IPv6 for end users will involve using one of the above mechanisms. Entities without assigned IPv6 addresses are defined as end users. If an end user needs many more addresses than can be provided by an ISP, then they will need to apply to their respective RIR for the number of addresses that they need.

4.6.1 Enterprise

In the middle of this transitional phase enterprises need to work out an IPv6 implementation plan. They need to look at which of the transition mechanisms best suits their needs. Subsections 4.6.1.1 through 4.6.1.6 below discuss the main stages when implementing IPv6 in an enterprise. Naturally, enterprises will draw their plans differently; however, the steps presented below will be contained in these various plans in some respect.

4.6.1.1 Apply for addresses/get addresses from ISP

The distribution of IPv6 address space follows a hierarchical structure. The IANA allocates address blocks to the RIRs (i.e., AfriNIC, APNIC, ARIN, LACNIC, and RIPE NCC). The RIRs distribute address blocks to Local Internet Registries (LIRs) which mainly consist of ISPs and enterprises [23]. The LIRs then distribute addresses to end users. IPv6 addresses will not be allocated carelessly as with IPv4, in fact IPv6 addresses are only leased by the user. Although RIRs may have local distribution policies IPv6 address space allocation is in general managed under strict international guidelines for the long-term interest of the Internet community [23].

4.6.1.2 Check connectivity from ISP

ISPs use a variety of ways to provide IPv6 service. This means that the enterprise must consult its ISP about which of the access methods they use, i.e. Dual stack, 6to4, 6over4, etc.

4.6.1.3 Train staff

Staff that will deploy and manage the IPv6 network in the enterprise must have suitable education. The staff in the enterprise who are only familiar with IPv4 address infrastructure, have over time developed habits, for example memorising certain special IPv4 addresses. IPv6 introduces a significant number of changes as we have discussed previously. The address notation, address length, and address divisions among other things are different in IPv6. Memorising IPv6 addresses and performing manual calculations of addresses may not be easy, thus specialist tools might be required.

4.6.1.4 Design addressing scheme/management system

What lengths of IPv6 prefix will be needed? Considering this question could be a chance to restructure the enterprise's whole network topology.

4.6.1.5 Check for software & hardware compatibility

Identify the pieces of equipment (such as routers, switches, modems etc.) that need to be upgraded or perhaps replaced.

4.6.1.6 Check internally developed applications for compatibility

Software developed in the enterprise that uses IP address information needs to be adjusted. Changes will need to be made in regard to the IP address length, routing protocols, security etc.

4.6.2 Home User

A contributor to the slight stagnation we are experiencing in IPv6 deployment is that home users (among other end users) are not demanding IPv6 from their ISPs. The Internet is becoming more pervasive in our daily lives and the average user needs to learn about the benefits of IPv6. In subsections 4.6.2.1 through 0 we discuss the requirements necessary for a home user to become IPv6 ready.

4.6.2.1 Check ISP for readiness

Ask ISP if they provide IPv6 connectivity, if they do not provide connectivity, consider consulting other ISPs. If an ISP provides IPv6 connectivity, then ask them which of the IPv6 access methods they offer (i.e. dual stack, 6to4, 6over 4, etc.). Consult the requirements that a customer has to fulfil (such as do they provide IPv6 compatible modems or other equipment necessary for connecting over IPv6).

4.6.2.2 Check network equipment for compatibility

All the necessary devices for Internet connection need to be IPv6 ready, these include routers, modems, etc. i.e., all of the customer-premises equipment (CPE).

4.6.2.3 Check software on computer devices for compatibility

Check that the version of the operating system (OS) in the computer device supports IPv6. The main OSs have had IPv6 readiness for quite some time, check the relevant manual for this information.

4.6.2.4 Configure LAN/HAN

Assign IP addresses to the computers in the Home Area Network (HAN) or Local Area Network (LAN). This can be done statically, i.e. manually or dynamically, i.e. automatically. The procedure for doing this address assignment depends on the OS that is running on the computers, the steps to set up the network interfaces are documented by the OS vendor.

4.7 IPv6 Experiment

In this chapter we are going to study IPv6 applications, connectivity and configuration for a home user. We will experiment with transition mechanisms in my own home network in order to achieve IPv6 connectivity.

4.7.1 Set-up

The set-up is as follows:

- A wireless network router (Netgear N300 model WNR3500L, firmware version V1.2.2.44_35.0.53, supports IPv6 and 6to4 tunnel)
- A Windows 7 PC connected via Ethernet
- A Triple Play Switch (model XG6745) and fiberoptic cable network provided by Telia.

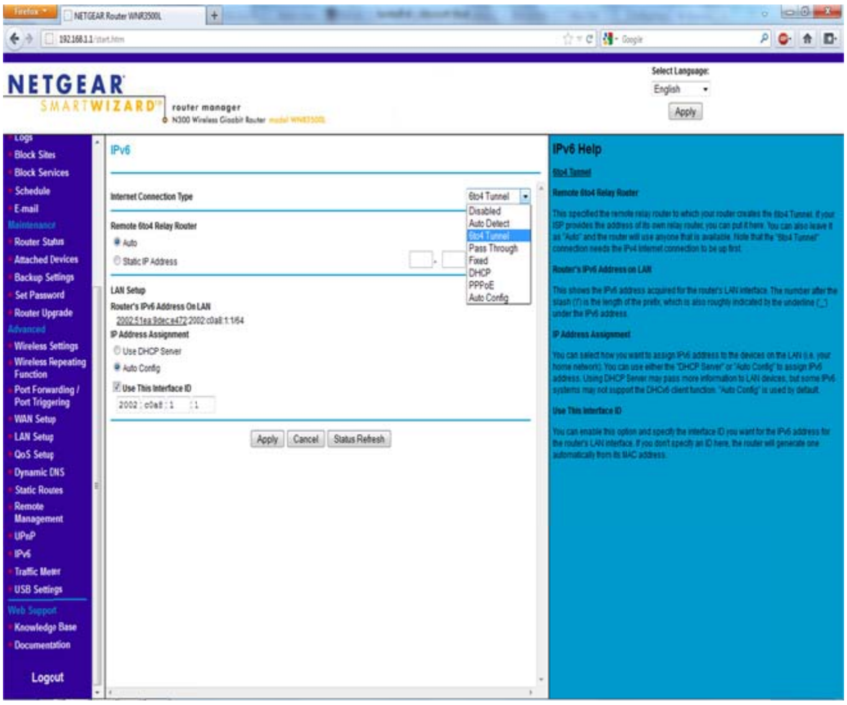
4.7.2 Method

Research is performed on the Internet to find the different methods that could be used to implement IPv6 on my home network. With the world IPv6 Launch right around the corner I expect many entities to be IPv6 ready. I expected that it should be relatively problem free to set up IPv6 on my home network by taking advantage of the current start of IPv6 readiness and making use of one of the transition mechanisms.

4.7.2.1 6to4 tunnelling

In the steps below we utilise 6to4 tunnelling to achieve IPv6 access to the Internet. 6to4 was as discussed in a previous section. It consists of encapsulation of IPv6 packets into IPv4 packet and delivering the resulting IPv4 packets over the IPv4 infrastructure. It is one of the non-native IPv6 methods that allow access to IPv6 services.

We use the Netgear smart wizard software utility to enable IPv6 in the home router. Logging on to the router is done by typing the Default Gateway into the web browser and providing the authentication information. Clicking on the IPv6 tab should produce a page that looks like Figure 4-1. From the drop down list we select a 6to4 tunnel, as this is a transition scheme that is supported by my ISP. We can assign IPv6 addresses to the devices in the home network manually or automatically from the Netgear software utility. After the selections we apply the chosen settings and can now test the IPv6 connectivity.



IPv6

Internet Connection Type: 6to4 Tunnel

Remote 6to4 Relay Router
● Auto
○ Static IP Address

LAN Setup

Router's IPv6 Address On LAN
2002:51ea:9dec:e472:2002:c0a8:1:1/64

IP Address Assignment
○ Use DHCP Server
● Auto Config

☑ Use This Interface ID
2002 : c0a8 : 1 : 1

IPv6 Help

6to4 Tunnel

Remote 6to4 Relay Router

This specified the remote relay router to which your router creates the 6to4 Tunnel. If your ISP provides the address of its own relay router, you can put it here. You can also leave it as "Auto" and the router will use anyone that is available. Note that the "6to4 Tunnel" connection needs the IPv4 internet connection to be up first.

Router's IPv6 Address on LAN

This shows the IPv6 address acquired for the router's LAN interface. The number after the slash (/) is the length of the prefix, which is also roughly indicated by the underline (_) under the IPv6 address.

IP Address Assignment

You can select how you want to assign IPv6 address to the devices on the LAN (i.e. your home network). You can use either the "DHCP Server" or "Auto Config" to assign IPv6 address. Using DHCP Server may pass more information to LAN devices, but some IPv6 systems may not support the DHCv6 client function. "Auto Config" is used by default.

Use This Interface ID

You can enable this option and specify the interface ID you want for the IPv6 address for the router's LAN interface. If you don't specify an ID here, the router will generate one automatically from its MAC address.

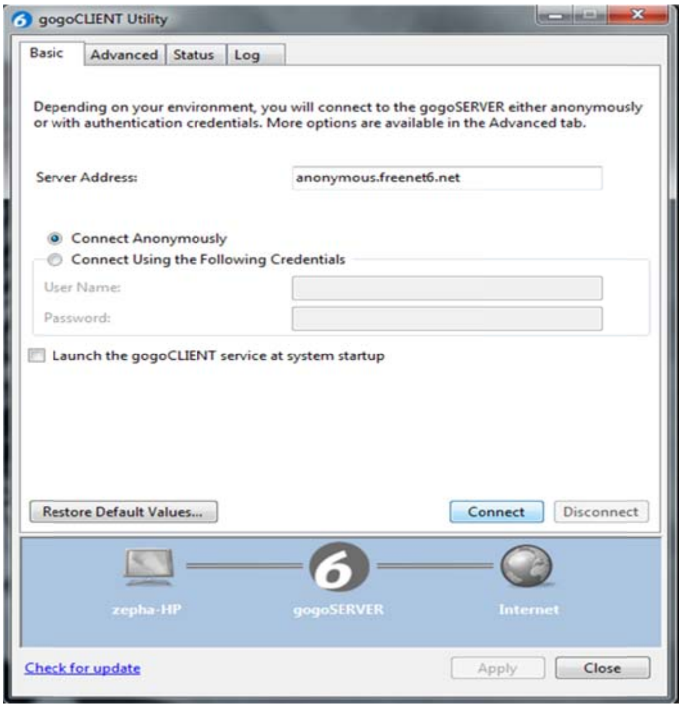
Figure 4-1: Netgear software utility interface used to configure router

4.7.2.2 Tunnel broker

As an alternative to using 6to4 we can utilize an IPv6 access service called Freenet6[3] that is based upon a client/server model. The client is software installed on a computer to automatically set up a tunnel. Below are the steps needed on my Microsoft Windows 7 desktop connected to an IPv6 compatible router with IPv4 internet access.

1. Sign up for an account on http://gogonet.gogo6.com/profile/gogoCLIENT
2. Download and install the gogoCLIENT
3. Launch the gogoCLIENT utility interface see Figure 4-2.
4. Go to *Advanced* tab to change the tunnel type if desired
5. Two connection methods are supported, *Anonymous* and *Authenticated*. We choose the Anonymous in this experiment.
6. Click connect to establish tunnel

[3] Freenet6 is a tunnel broker i.e. a service that provides a network tunnel. It is offered free of charge on a variety of platforms providing IPv6 connectivity including NAT traversal, for details see http://gogonet.gogo6.com/page/freenet6-tunnelbroker .



Depending on your environment, you will connect to the gogoSERVER either anonymously or with authentication credentials. More options are available in the Advanced tab.

Server Address: anonymous.freenet6.net

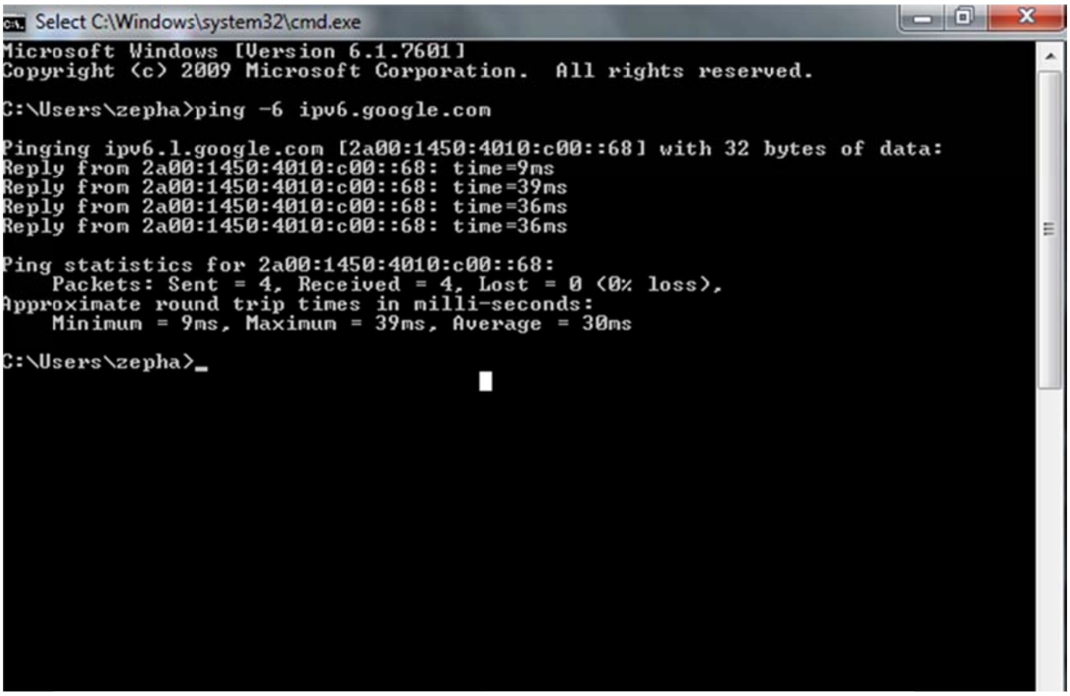
● Connect Anonymously
○ Connect Using the Following Credentials
User Name:
Password:

□ Launch the gogoCLIENT service at system startup

Figure 4-2: gogoCLIENT utility interface

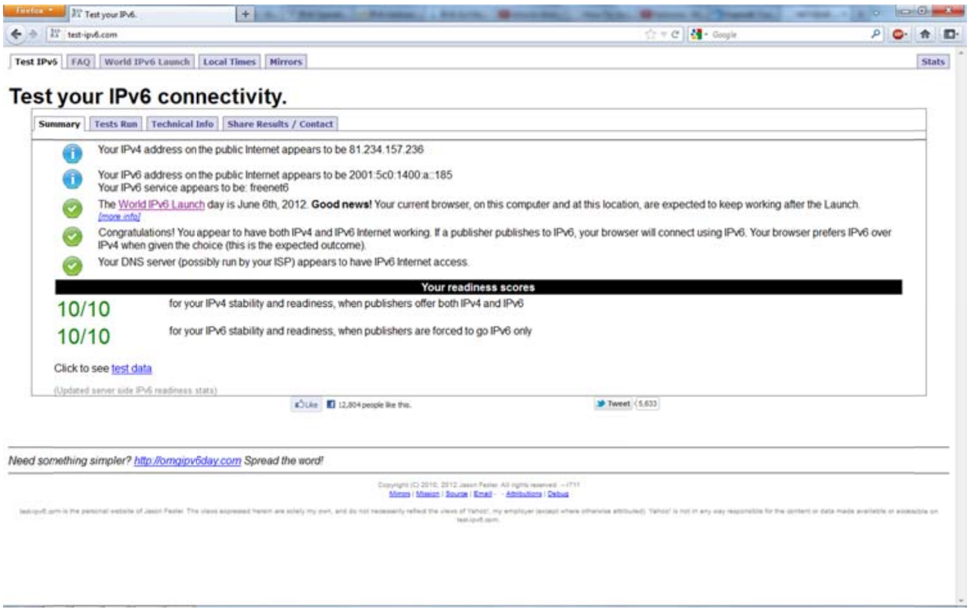
4.7.3 Results

The process of setting up tunnels (as described in sections 4.7.2.1 and 4.7.2.2) was straightforward. The Freenet6 access method has especially easy to follow instructions available at their website. Testing the IPv6 connectivity was done by visiting several IPv6 only sites (such as ipv6.google.com) see Figure 4-3. We also tested our connectivity at http://test-ipv6.com/ , a site dedicated to this sole purpose and the results as shown in Figure 4-4 below.



```
Microsoft Windows [Version 6.1.7601]
Copyright (c) 2009 Microsoft Corporation.  All rights reserved.

C:\Users\zepha>ping -6 ipv6.google.com

Pinging ipv6.l.google.com [2a00:1450:4010:c00::68] with 32 bytes of data:
Reply from 2a00:1450:4010:c00::68: time=9ms
Reply from 2a00:1450:4010:c00::68: time=39ms
Reply from 2a00:1450:4010:c00::68: time=36ms
Reply from 2a00:1450:4010:c00::68: time=36ms

Ping statistics for 2a00:1450:4010:c00::68:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 9ms, Maximum = 39ms, Average = 30ms

C:\Users\zepha>_
```

Figure 4-3: Pinging ipv6.google.com from the command prompt



Test your IPv6 connectivity.

Your IPv4 address on the public Internet appears to be 81.234.157.236

Your IPv6 address on the public Internet appears to be 2001:5c0:1400:a::185
Your IPv6 service appears to be: freenet6

The World IPv6 Launch day is June 6th, 2012. **Good news!** Your current browser, on this computer and at this location, are expected to keep working after the Launch. [more info]

Congratulations! You appear to have both IPv4 and IPv6 Internet working. If a publisher publishes to IPv6, your browser will connect using IPv6. Your browser prefers IPv6 over IPv4 when given the choice (this is the expected outcome).

Your DNS server (possibly run by your ISP) appears to have IPv6 Internet access.

Your readiness scores

10/10	for your IPv4 stability and readiness, when publishers offer both IPv4 and IPv6
10/10	for your IPv6 stability and readiness, when publishers are forced to go IPv6 only

Click to see test data

(Updated server side IPv6 readiness stats)

Need something simpler? http://omgipv6day.com Spread the word!

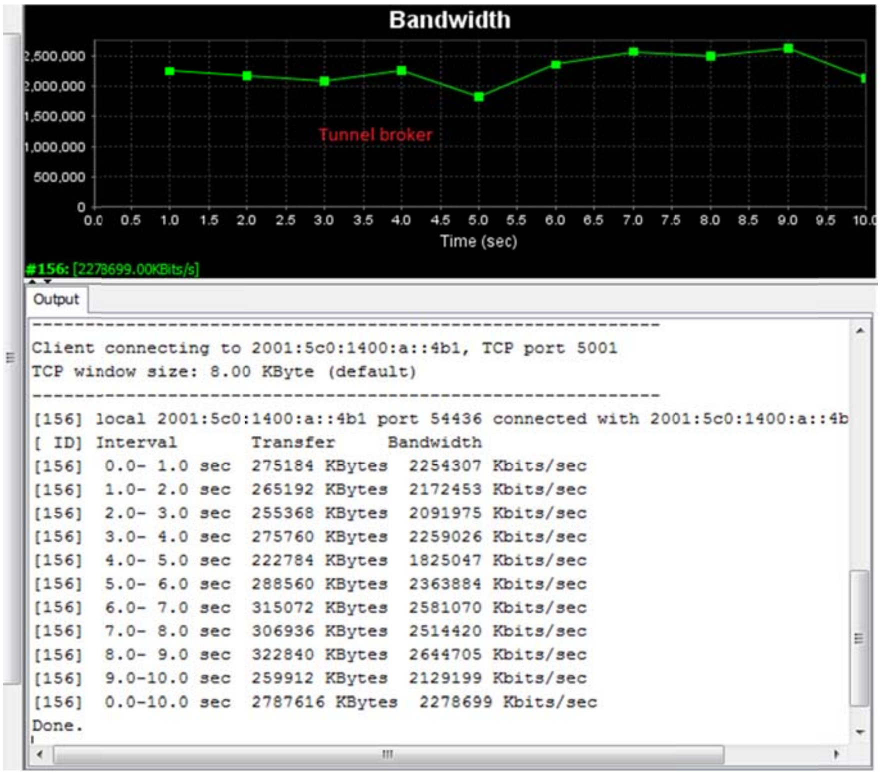
Figure 4-4: Results of testing Ipv6 connectivity

4.8 Performance analysis of transition mechanisms

In sections 4.1 through 4.3 we saw that IPv6 transition mechanisms are divided into three main categories, namely dual stack, translation, and tunnelling. In this section we present a performance evaluation of networks configured using the 6to4 tunnelling and tunnel broker methods shown in the previous section. The goal is to provide empirical measurements for reference when designing a network. The parameters we analyse are the throughput, latency, and packet loss.

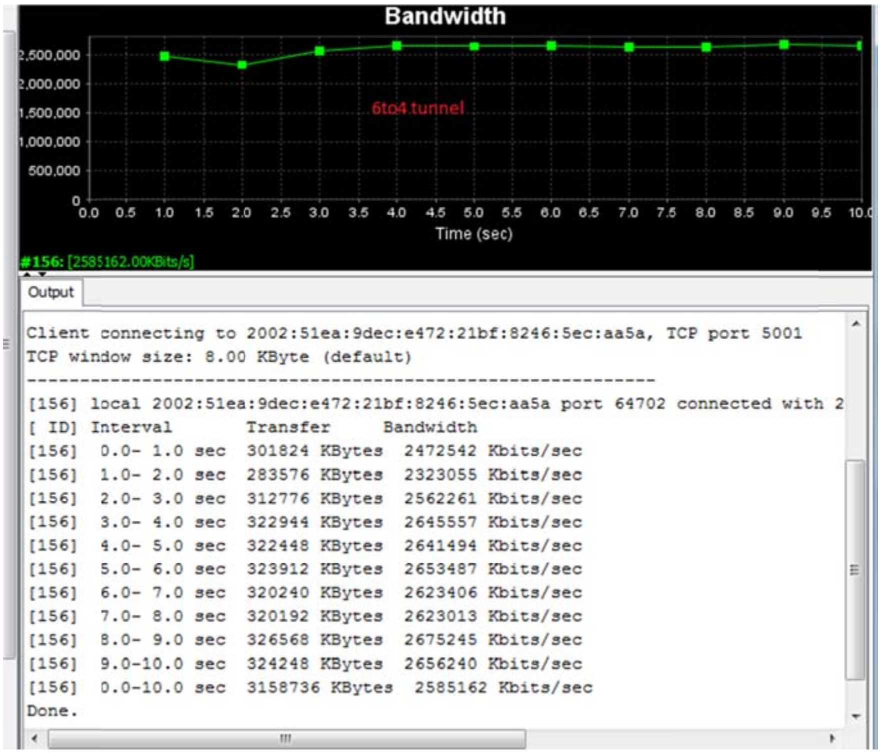
4.8.1 Throughput

Throughput is a performance measure in computer networks. It is defined as the rate at which a host receives data from a communications partner, i.e. the amount of data transferred over a communications link per unit time [24]. To calculate throughput we look at how long it takes to transfer a certain amount of bits. The throughput will then be the amount of data transferred divided by the time it takes to transmit the data yielding a throughput in bits/sec. Throughput is an important benchmark for the design of a network, especially if it is expected to support throughput critical applications such as file transfers and web browsing. In this experiment we use dedicated software to examine the throughput for the 6to4 tunnelling and tunnel broker transition strategies in a home network. We used Iperf, a network testing tool and the associated graphical frontend Jperf (available at http://openmaniak.com/iperf.php). The results of the bandwidth test for the different network configurations are shown in Figure 4-5, Figure 4-6, Figure 4-7, and Figure 4-8 below. From these figures we see that an increased number of transferred bytes leads to increased throughput. The 6to4 tunnel performs slightly better than the tunnel broker on the TCP test and the tunnel broker is slightly better on the UDP test.



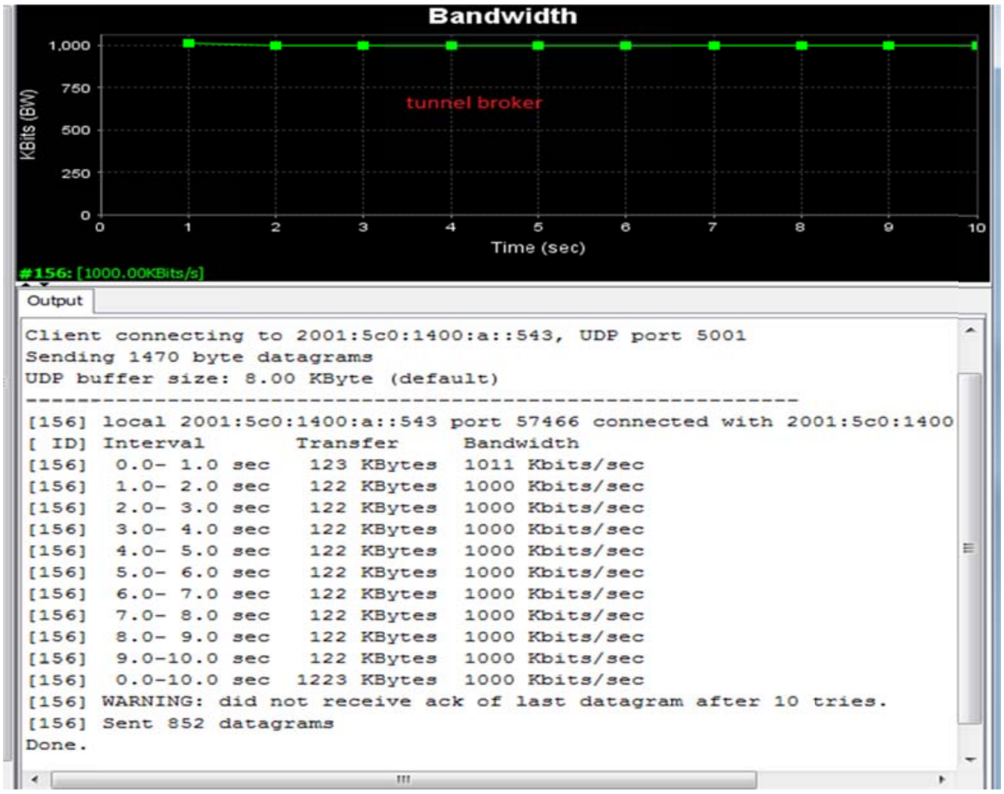
```
------------------------------------------------------------
Client connecting to 2001:5c0:1400:a::4b1, TCP port 5001
TCP window size: 8.00 KByte (default)
------------------------------------------------------------
[156] local 2001:5c0:1400:a::4b1 port 54436 connected with 2001:5c0:1400:a::4b
[ ID] Interval       Transfer     Bandwidth
[156]  0.0- 1.0 sec  275184 KBytes  2254307 Kbits/sec
[156]  1.0- 2.0 sec  265192 KBytes  2172453 Kbits/sec
[156]  2.0- 3.0 sec  255368 KBytes  2091975 Kbits/sec
[156]  3.0- 4.0 sec  275760 KBytes  2259026 Kbits/sec
[156]  4.0- 5.0 sec  222784 KBytes  1825047 Kbits/sec
[156]  5.0- 6.0 sec  288560 KBytes  2363884 Kbits/sec
[156]  6.0- 7.0 sec  315072 KBytes  2581070 Kbits/sec
[156]  7.0- 8.0 sec  306936 KBytes  2514420 Kbits/sec
[156]  8.0- 9.0 sec  322840 KBytes  2644705 Kbits/sec
[156]  9.0-10.0 sec  259912 KBytes  2129199 Kbits/sec
[156]  0.0-10.0 sec  2787616 KBytes  2278699 Kbits/sec
Done.
```

Figure 4-5: Bandwidth measured using Iperf through TCP tests for the tunnel broker



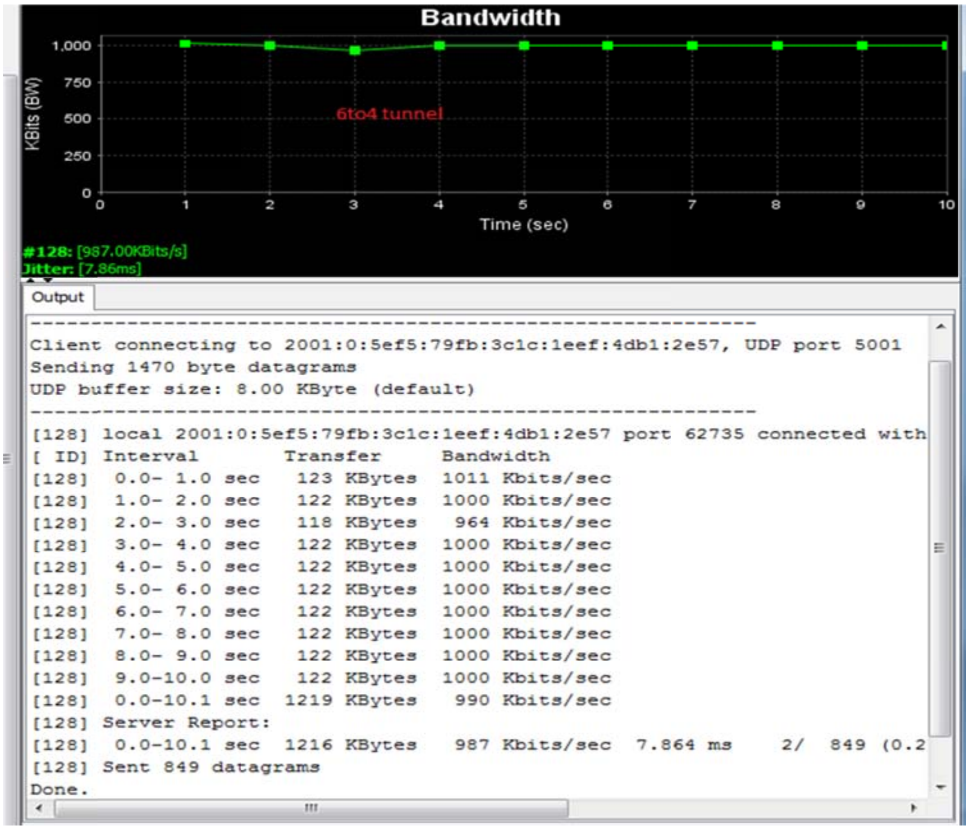
```
Client connecting to 2002:51ea:9dec:e472:21bf:8246:5ec:aa5a, TCP port 5001
TCP window size: 8.00 KByte (default)
------------------------------------------------------------
[156] local 2002:51ea:9dec:e472:21bf:8246:5ec:aa5a port 64702 connected with 2
[ ID] Interval       Transfer     Bandwidth
[156]  0.0- 1.0 sec  301824 KBytes  2472542 Kbits/sec
[156]  1.0- 2.0 sec  283576 KBytes  2323055 Kbits/sec
[156]  2.0- 3.0 sec  312776 KBytes  2562261 Kbits/sec
[156]  3.0- 4.0 sec  322944 KBytes  2645557 Kbits/sec
[156]  4.0- 5.0 sec  322448 KBytes  2641494 Kbits/sec
[156]  5.0- 6.0 sec  323912 KBytes  2653487 Kbits/sec
[156]  6.0- 7.0 sec  320240 KBytes  2623406 Kbits/sec
[156]  7.0- 8.0 sec  320192 KBytes  2623013 Kbits/sec
[156]  8.0- 9.0 sec  326568 KBytes  2675245 Kbits/sec
[156]  9.0-10.0 sec  324248 KBytes  2656240 Kbits/sec
[156]  0.0-10.0 sec  3158736 KBytes  2585162 Kbits/sec
Done.
```

Figure 4-6: Bandwidth measured using Iperf through TCP tests for the 6to4 tunnel



```
Client connecting to 2001:5c0:1400:a::543, UDP port 5001
Sending 1470 byte datagrams
UDP buffer size: 8.00 KByte (default)
------------------------------------------------------------
[156] local 2001:5c0:1400:a::543 port 57466 connected with 2001:5c0:1400
[ ID] Interval       Transfer     Bandwidth
[156]  0.0- 1.0 sec   123 KBytes  1011 Kbits/sec
[156]  1.0- 2.0 sec   122 KBytes  1000 Kbits/sec
[156]  2.0- 3.0 sec   122 KBytes  1000 Kbits/sec
[156]  3.0- 4.0 sec   122 KBytes  1000 Kbits/sec
[156]  4.0- 5.0 sec   122 KBytes  1000 Kbits/sec
[156]  5.0- 6.0 sec   122 KBytes  1000 Kbits/sec
[156]  6.0- 7.0 sec   122 KBytes  1000 Kbits/sec
[156]  7.0- 8.0 sec   122 KBytes  1000 Kbits/sec
[156]  8.0- 9.0 sec   122 KBytes  1000 Kbits/sec
[156]  9.0-10.0 sec   122 KBytes  1000 Kbits/sec
[156]  0.0-10.0 sec  1223 KBytes  1000 Kbits/sec
[156] WARNING: did not receive ack of last datagram after 10 tries.
[156] Sent 852 datagrams
Done.
```

Figure 4-7: Bandwidth measured using Iperf sending UDP data streams for the tunnel broker



```
------------------------------------------------------------
Client connecting to 2001:0:5ef5:79fb:3c1c:1eef:4db1:2e57, UDP port 5001
Sending 1470 byte datagrams
UDP buffer size: 8.00 KByte (default)
------------------------------------------------------------
[128] local 2001:0:5ef5:79fb:3c1c:1eef:4db1:2e57 port 62735 connected with
[ ID] Interval       Transfer     Bandwidth
[128]  0.0- 1.0 sec   123 KBytes  1011 Kbits/sec
[128]  1.0- 2.0 sec   122 KBytes  1000 Kbits/sec
[128]  2.0- 3.0 sec   118 KBytes   964 Kbits/sec
[128]  3.0- 4.0 sec   122 KBytes  1000 Kbits/sec
[128]  4.0- 5.0 sec   122 KBytes  1000 Kbits/sec
[128]  5.0- 6.0 sec   122 KBytes  1000 Kbits/sec
[128]  6.0- 7.0 sec   122 KBytes  1000 Kbits/sec
[128]  7.0- 8.0 sec   122 KBytes  1000 Kbits/sec
[128]  8.0- 9.0 sec   122 KBytes  1000 Kbits/sec
[128]  9.0-10.0 sec   122 KBytes  1000 Kbits/sec
[128]  0.0-10.1 sec  1219 KBytes   990 Kbits/sec
[128] Server Report:
[128]  0.0-10.1 sec  1216 KBytes   987 Kbits/sec  7.864 ms    2/  849 (0.2
[128] Sent 849 datagrams
Done.
```

Figure 4-8: Bandwidth measured using Iperf sending UDP data streams for the 6to4 tunnel

4.8.2 Latency

In this test we will examine the round-trip latency, i.e. the time it takes for a data packet to be transmitted across a network and for an acknowledgement to be received by the sender. The test is performed by sending 32, 64, 128, 256, 512 and 1024 bytes size packets using the ping network utility from the Windows command prompt using the following command: *ping -6 –n 100 –l **size** ipv6.google.com*. In this command ***size*** is the size of the packet to be sent. We repeat the ping test 100 times for our different network configurations and packet sizes.

Table 4-1 below shows the results of a latency comparison between a 6to4 tunnel and a tunnel broker. The client in this test is a Windows 7 PC in a home network while the server is ipv6.google.com. The test results show that the 6to4 tunnel performs better since it has the lowest latency.

Table 4-1: Data from 100 pings with different packet sizes for 6to4 and tunnel broker

#of Bytes	Average RTT tunnel broker (ms)	Average RTT 6to4 (ms)	Loss rate tunnel broker (%)	Loss rate 6to4 (%)
32	68	37	0	50
64	68	44	0	50
128	68	36	0	75
256	68	37	0	75
512	69	36	0	75
1024	69	41	0	75

4.8.3 Packet loss

Queuing delays in the nodes of a network present interesting challenges because these queues have finite capacity. Because of the finite capacity of these queues packets will be dropped, i.e. lost. Packet loss increases as the traffic intensity increases [24]. Therefore to analyse packet loss we will increase the packet sizes sent across our test networks and then measure the loss rate. We analyse the loss rate from the results Table 4-1 which indicate an increasing loss rate for the 6to4 tunnel as the packet size increases. This is perhaps explained by the fact that the 6to4 tunnel in this experiment was created by a remote relay router in my home network. The whole set-up therefore lacks service level agreements and is thus unpredictable. Furthermore, the ping requests probably time out because of firewall restrictions at both client and server. Transmitting big IPv6 packets that have to be encapsulated in IPv4 packets is also a contributor to the loss of packets experienced by using an unmanaged tunnel. However, a UDP test using Iperf with increasing packet sizes does not experience high packet losses as with the ping utility that sends Internet Control Message Protocol (ICMP) messages. ICMP packets sent with ping time out because of imposed rate limits to prevent attacks, a router may not respond to a ping because of this and it will appear as if the packet were lost. Furthermore, ICMP packets have a low priority on a network and so if a node along the path has lots of things to process the ICMP packet is likely to be dropped resulting in a packet loss.

4.8.4 Results

In this section we evaluated the performance of 6to4 tunnelling and tunnel broker in a home network. Tests were performed using the ping network utility sending different size packets to a remote IPv6 server and the Iperf network tool creating TCP and UDP data streams. We examined two of the transition mechanisms; each has their advantages and disadvantages and may be used according to the needs of the network.

4.9 *Enabling IPv6 routing*

In this chapter we will experiment with assigning IPv6 addresses and enabling IPv6 traffic on Cisco routers. The procedures in this section show how to configure IPv6 addresses and implement IPv6 routing on a simple test topology with the aim of showing how an enterprise network (LAN/MAN/WAN) could be set up.

4.9.1 Set-up

The set-up is as follows:

- A windows 7 PC
- GNS3 version 0.8.2 all-in-one (available at http://www.gns3.net/download)
- Cisco 2691 IOS image (adventerprisek9-mz). Extracted from a router purchased on eBay.

4.9.2 Method

In this section we use Graphical Network Simulator (GNS3) a tool that simulates complex topologies based on Cisco and Juniper hardware. It is installed on the windows 7 PC where the Cisco IOS image is added to the environment. In GNS3 the simple test network in Figure 4-9 below is simulated in order to experiment with the features of IPv6 routing. The basis for this experiment is the Cisco software manual *Configuring IPv6 for Cisco IOS* [25]. The goal of the experiment is to enable reachability between subnets using the ping network utility. The steps needed to achieve this are:

- Assign IPv6 addresses to the routers in our test network
- Ping within subnets
- Implement IPv6 static routes /implement a routing protocol (RIPng, EIGRP,OSPF, BGP)
- Ping across subnets

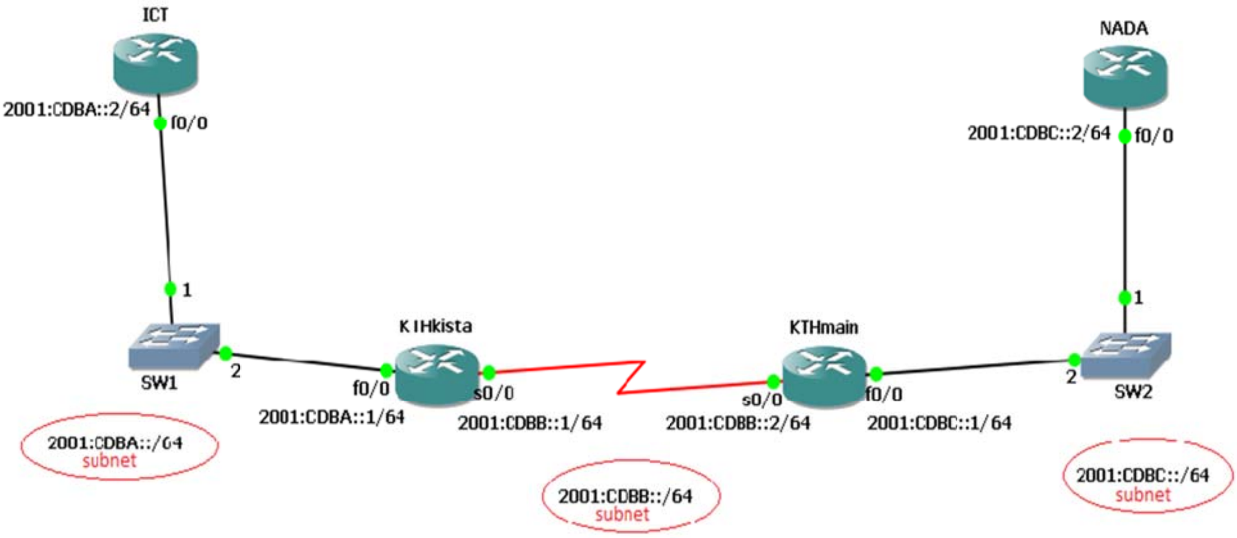


Figure 4-9: Simulation in GNS3 to experiment with IPv6 address assignment and routing

4.9.3 Configuring IPv6 addressing and implementing IPv6 routing

Below we go through the steps of assigning IPv6 addresses and enabling IPv6 routing in GNS3. In Figure 4-9 we have our simple test topology with four Cisco 2691 routers and two generic switches. The layer three topology for the test network results in three IPv6 subnets. We use the arbitrarily chosen IPv6 addresses 2001:cdba::/64 , 2001:cdbb::/64 and 2001:cdbc::/64 to denote the respective subnets. Routers ICT and KTHkista are on a shared

Ethernet network and the same goes for routers NADA and KTHmain; while KTHkista and KTHmain are connected via a serial link.

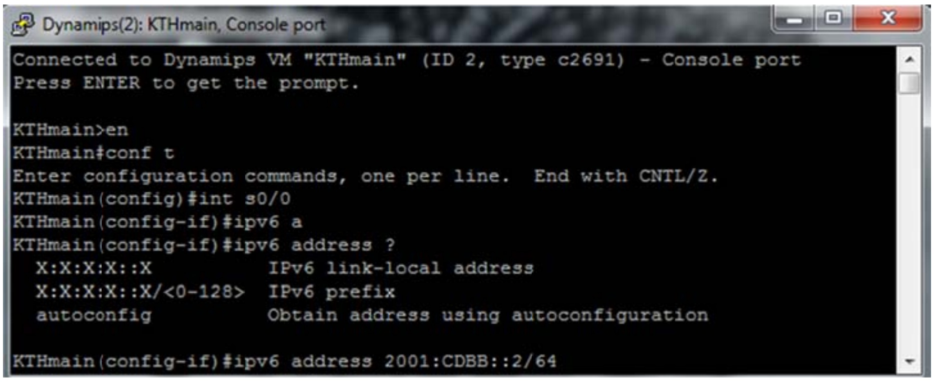
4.9.3.1 Assigning IPv6 addresses to our routers

We now assign IPv6 addresses to our routers according to our simple test topology with the Cisco commands below in the respective router's command prompt.

Commands while in executive mode:

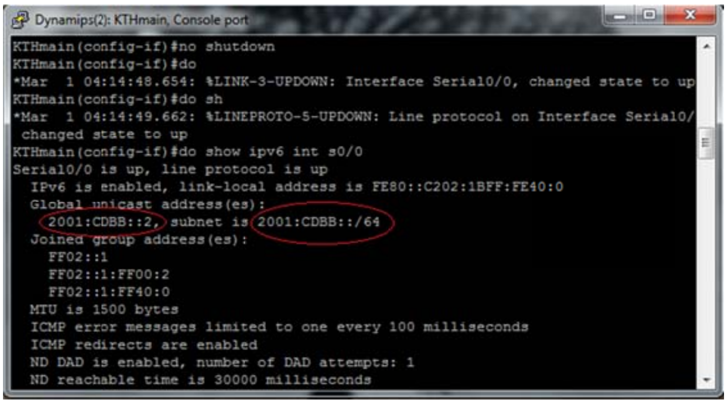
- enable
- configure terminal
- interface *type number*
- ipv6 address *IPv6-Prefix/Prefix-length*
- no shutdown

For example Figure 4-10 and Figure 4-11 below show how an IPv6 address was assigned to router KTHmain.



```
Connected to Dynamips VM "KTHmain" (ID 2, type c2691) - Console port
Press ENTER to get the prompt.

KTHmain>en
KTHmain#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
KTHmain(config)#int s0/0
KTHmain(config-if)#ipv6 a
KTHmain(config-if)#ipv6 address ?
  X:X:X:X::X          IPv6 link-local address
  X:X:X:X::X/<0-128>  IPv6 prefix
  autoconfig          Obtain address using autoconfiguration

KTHmain(config-if)#ipv6 address 2001:CDBB::2/64
```

Figure 4-10: Assigning IPv6 address to router KTHmain's serial interface

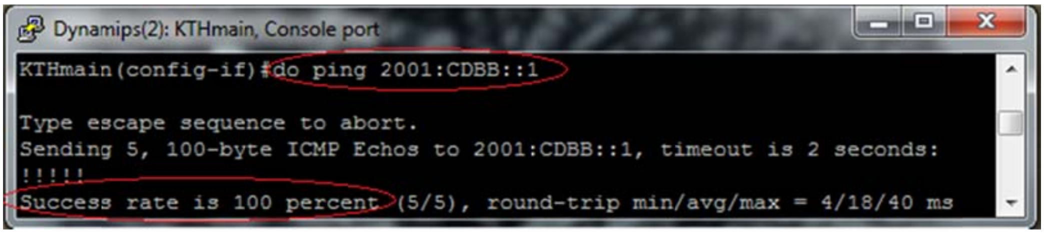


```
KTHmain(config-if)#no shutdown
KTHmain(config-if)#do
*Mar  1 04:14:48.654: %LINK-3-UPDOWN: Interface Serial0/0, changed state to up
KTHmain(config-if)#do sh
*Mar  1 04:14:49.662: %LINEPROTO-5-UPDOWN: Line protocol on Interface Serial0/
 changed state to up
KTHmain(config-if)#do show ipv6 int s0/0
Serial0/0 is up, line protocol is up
  IPv6 is enabled, link-local address is FE80::C202:1BFF:FE40:0
  Global unicast address(es):
    2001:CDBB::2, subnet is 2001:CDBB::/64
  Joined group address(es):
    FF02::1
    FF02::1:FF00:2
    FF02::1:FF40:0
  MTU is 1500 bytes
  ICMP error messages limited to one every 100 milliseconds
  ICMP redirects are enabled
  ND DAD is enabled, number of DAD attempts: 1
  ND reachable time is 30000 milliseconds
```

Figure 4-11: Turning the interface up and showing its status with the show command

4.9.3.2 Ping within subnets

After assigning the addresses as shown in our test topology in Figure 4-9 using the procedure in section 4.9.3.1 we should be able to ping within the subnets. Figure 4-12 below shows a successful ping from KTHmain to KTHkista via the serial interfaces that connect these two routers. Similar results were achieved for pings within the other subnets as well.

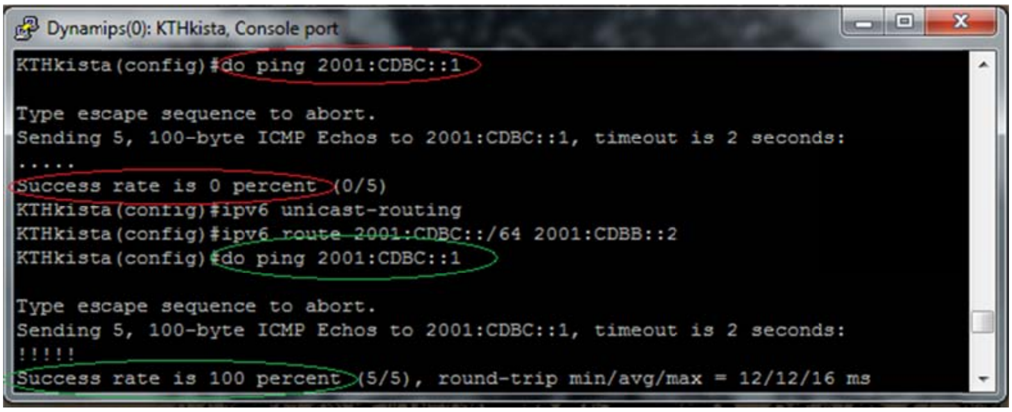


```
KTHmain(config-if)#do ping 2001:CDBB::1

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 2001:CDBB::1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 4/18/40 ms
```

Figure 4-12: Successful ping within subnet 2001:CDBB::/64

4.9.3.3 Implementing IPv6 static routes

To implement IPv6 static routes we start by enabling the forwarding of IPv6 unicast datagrams using the command: ipv6 unicast-routing. The routes are then implemented by typing in the network we are trying to reach and the next hop IP address or exit interface in the following command: ipv6 route *network/mask next hop*. For example Figure 4-13 below shows how we initially cannot ping the IP address 2001:CDBC::1/64. We implement static routing in each router for reachability to each subnet in the topology. After the set-up of the static routes in each router we are successfully able to ping 2001:CDBC::1/64.

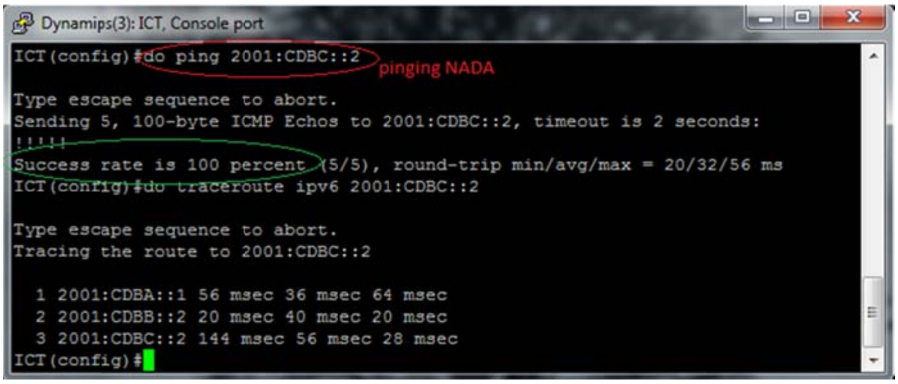


```
KTHkista(config)#do ping 2001:CDBC::1

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 2001:CDBC::1, timeout is 2 seconds:
.....
Success rate is 0 percent (0/5)
KTHkista(config)#ipv6 unicast-routing
KTHkista(config)#ipv6 route 2001:CDBC::/64 2001:CDBB::2
KTHkista(config)#do ping 2001:CDBC::1

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 2001:CDBC::1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 12/12/16 ms
```

Figure 4-13: Implementing static IPv6 for KTHkista to the subnet 2001:CDBC::/64

4.9.3.4 Ping across subnets

After the procedures in the previous steps we should be able to ping successfully across all the subnets. To test this we will ping from the router NADA to the router ICT and vice versa, Figure 4-14 below shows a successful ping from the router ICT to the router NADA.



```
ICT(config)#do ping 2001:CDBC::2

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 2001:CDBC::2, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 20/32/56 ms
ICT(config)#do traceroute ipv6 2001:CDBC::2

Type escape sequence to abort.
Tracing the route to 2001:CDBC::2

  1 2001:CDBA::1 56 msec 36 msec 64 msec
  2 2001:CDBB::2 20 msec 40 msec 20 msec
  3 2001:CDBC::2 144 msec 56 msec 28 msec
ICT(config)#
```

Figure 4-14: Pinging NADA from ICT and showing every hop with traceroute

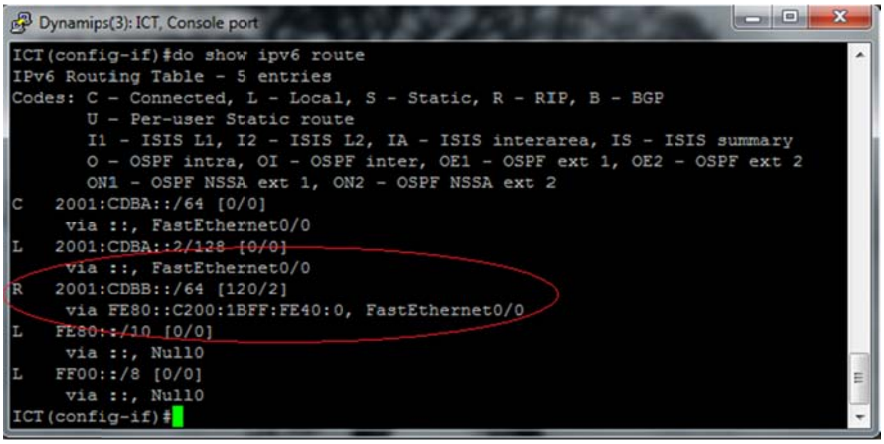
4.9.3.5 Implementing RIPng

In normal production networks static routing is not practical or scalable. This is where routing protocols (such as RIPng, EIGRP, OSPF, BGP) come into play. For the purpose of demonstration we will use RIPng in this experiment. The commands below show how to enable RIPng, advertising the interface and sending advertisements of the networks associated with the interface.

Cisco commands while in executive mode:

- enable
- configure terminal
- ipv6 unicast-routing
- interface *type number*
- ipv6 rip *name* enable

Figure 4-15 below shows that we eliminated all the static routes and use RIPng to learn the routes to the different networks in our test topology. After enabling rip for the different interfaces on the router in our test topology the IPv6 routes are updated. We see that RIPng uses link-local addressing for its communication.



Dynamips(3): ICT, Console port

```
ICT(config-if)#do show ipv6 route
IPv6 Routing Table - 5 entries
Codes: C - Connected, L - Local, S - Static, R - RIP, B - BGP
       U - Per-user Static route
       I1 - ISIS L1, I2 - ISIS L2, IA - ISIS interarea, IS - ISIS summary
       O - OSPF intra, OI - OSPF inter, OE1 - OSPF ext 1, OE2 - OSPF ext 2
       ON1 - OSPF NSSA ext 1, ON2 - OSPF NSSA ext 2
C   2001:CDBA::/64 [0/0]
     via ::, FastEthernet0/0
L   2001:CDBA::2/128 [0/0]
     via ::, FastEthernet0/0
R   2001:CDBB::/64 [120/2]
     via FE80::C200:1BFF:FE40:0, FastEthernet0/0
L   FE80::/10 [0/0]
     via ::, Null0
L   FF00::/8 [0/0]
     via ::, Null0
ICT(config-if)#
```

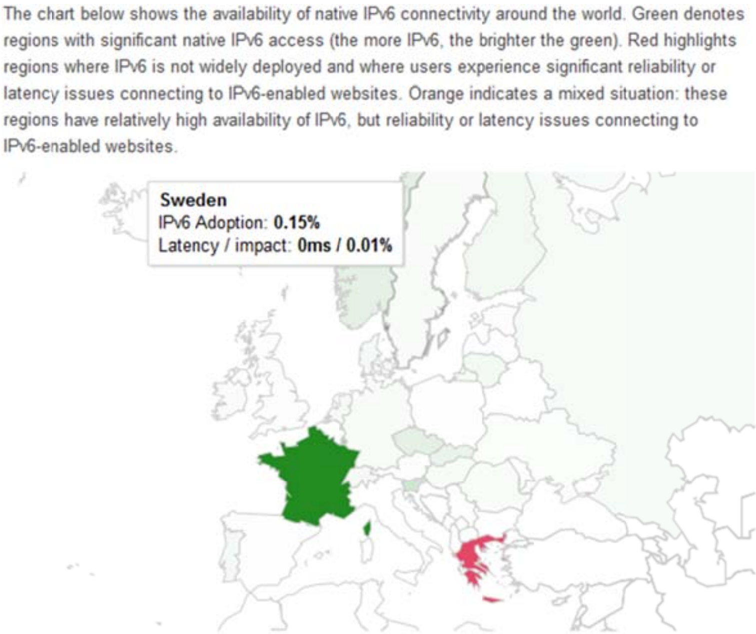
Figure 4-15: RIPng received as an advertisement with link-local address as next hop

4.9.4 Results

Reachability was achieved between the different subnets in our test topology. We experimented with IPv6 address assignment, implementing IPv6 static routes, and implementing RIPng. Implementing static routing as we observed gets quite messy when dealing with complex networks. Routing protocols such as RIPng which learn routes to different networks are therefore used and work very neatly.

4.10 IPv6 statistics

Statistics on IPv6 adoption and performance are continually collected in order to evaluate IPv6 connectivity. Below are a couple of graphs provided by Google and RIPE NCC that measure the country adoption, region adoption, and the general global adoption of IPv6.



The chart below shows the availability of native IPv6 connectivity around the world. Green denotes regions with significant native IPv6 access (the more IPv6, the brighter the green). Red highlights regions where IPv6 is not widely deployed and where users experience significant reliability or latency issues connecting to IPv6-enabled websites. Orange indicates a mixed situation: these regions have relatively high availability of IPv6, but reliability or latency issues connecting to IPv6-enabled websites.

Sweden
IPv6 Adoption: **0.15%**
Latency / impact: **0ms / 0.01%**

Figure 4-16: Native IPv6 adoption in Europe and Sweden specifically statistics provided by Google on http://www.google.com/intl/en/ipv6/statistics/

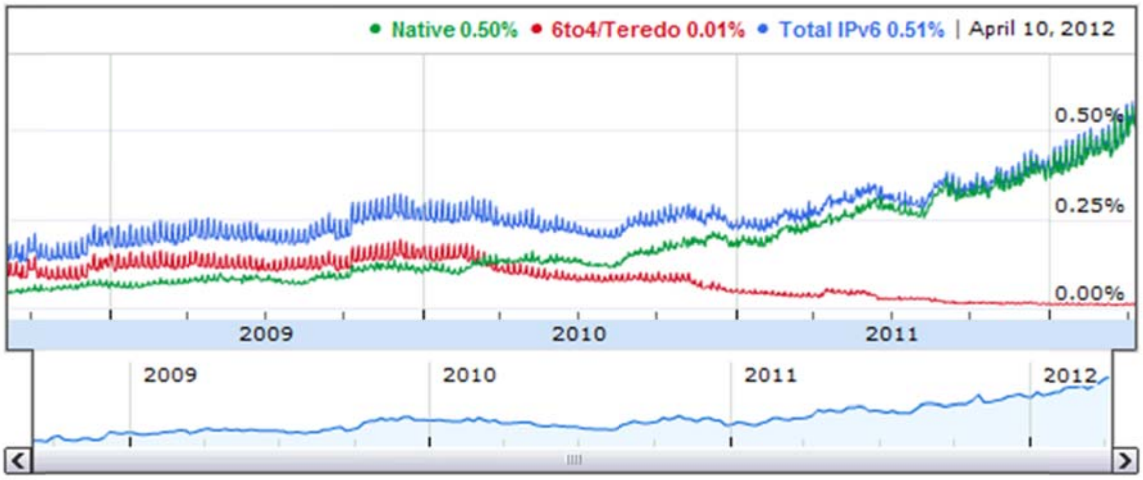


Figure 4-17: Percentage of users that would access the Google website over IPv6, graph provided by http://www.google.com/intl/en/ipv6/statistics/

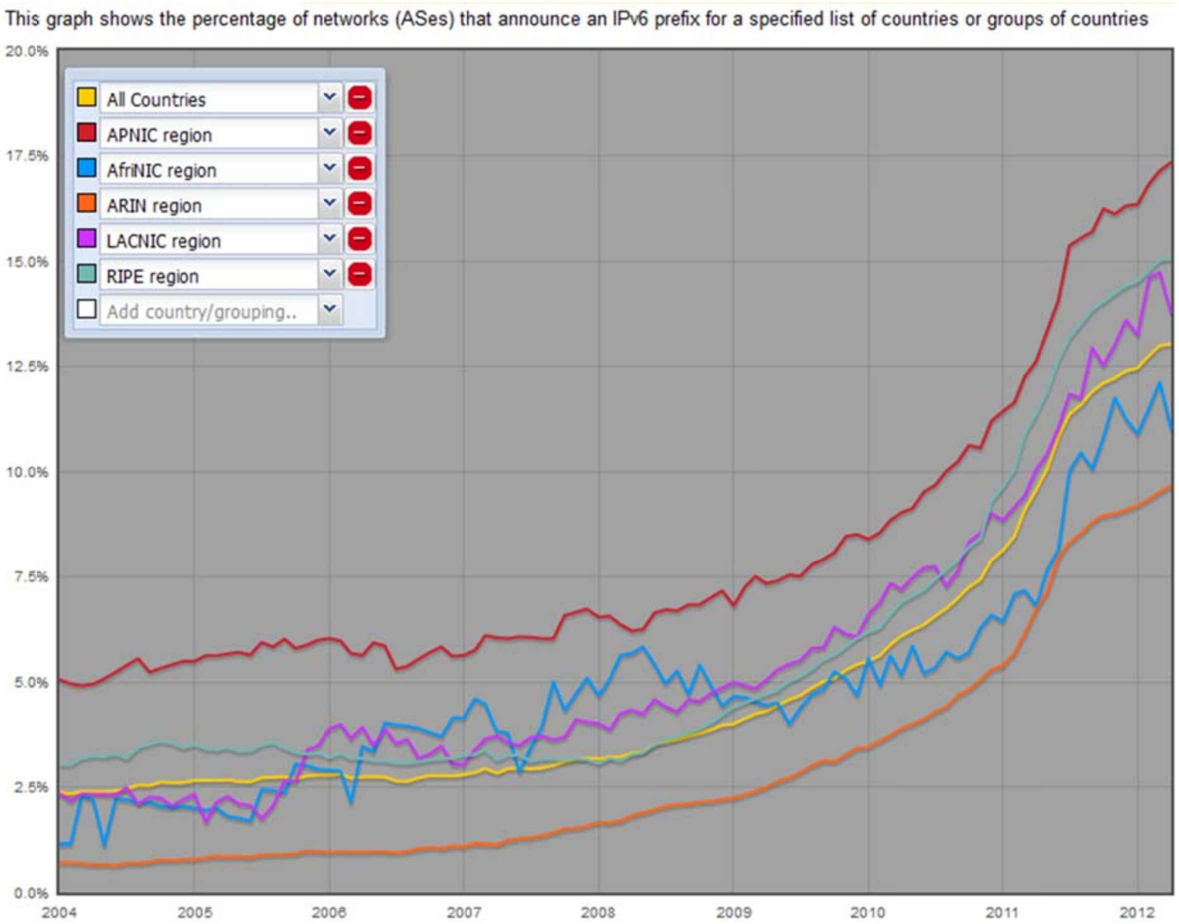


Figure 4-18: IPv6 adoption in all countries and Regional Internet Registry, graph provided by http://www.ipv6actnow.org/statistics/

4.11 Research and Development

On June 8th 2011 was World IPv6 day, an initiative to run IPv6 for 24 hours as a test to see if it really is the solution and also to uncover problems. This trial was meant to motivate all the main actors across the industry to get their products and services IPv6 ready. Hundreds participated in this test and the Internet traffic was measured in order to analyse potential problems [26]. No significant issues were reported by the parties involved, a mild victory in the evolution of the Internet [27].

Following the success of the test last year it is planned on June 6th 2012 to permanently deploy IPv6 for the services of the participants. Major ISPs participating include; AT&T, Comcast, FreeTelecom, Internode and many others. Prominent network equipment vendors including Cisco and D-Link will also participate. Content providers such as Google, Yahoo, Microsoft Bing and Yahoo will deploy their services and will undoubtedly be instrumental in encouraging several other actors to follow [28].

5 Conclusions and Future work

The growth of the Internet has come with many challenges for the different layers that constitute its infrastructure. The Internet has gone from its humble beginnings as a research project, and evolved to a research and education infrastructure. This was subsequently transformed into a global arena for the exchange of information. The think at the time about the future of the Internet shaped various critical aspects of the technology negatively from a developmental point of view. It was initially not expected that the Internet would expand much beyond the borders of universities. Nevertheless, it did and soon encountered a much more hostile environment. The network is now faced with rapid growth, but increasingly security issues are receiving more attention as more and more sensitive services (such as banking, shopping and health-care) are introduced to the network. The IP layer is frequently the target of various attacks. IPv4 address space exhaustion is looming dead ahead, and as a consequence the IPv4 routing table sizes are growing.

From the preceding chapters and sections we have discussed the major problems that plague IPv4 addressing. We have studied key aspects of the new version of the Internet Protocol (i.e. IPv6), specifically including deployment guidelines for end users. What is noteworthy of these discussions is that no definitive answers exist. We are constantly refining the methods we employ because of unforeseen events. Despite the availability of IPv6, deployment remains low and many factors contribute to this;

- Economic factors, like any major engineering undertaking it is no exception that migrating all hosts from IPv4 to IPv6 will consume a lot of resources. These resources include the expense of training network staff, upgrading equipment and applications, designing network infrastructure etc.
- Immediate profitability, a driving force for the major players in the field to take the initiative. Because of the lack of immediate profitability many well established organisations offer and utilise very few services on IPv6 only networks. The IPv6 Internet lacks a lucrative service for end users and has thus received poor media coverage. End users cannot use IPv6 because it is not provided to them by their ISPs. The ISPs claim that they are at mercy of hardware and software vendors to provide IPv6 support. All of these impediments demotivate /hinder websites and others from deploying/providing IPv6 services.
- IPv4 investments, many organisations are heavily invested in hardware and software that unfortunately may not support any of the transitioning mechanisms. The most economic and perhaps even environmental approach is to wait until these devices are replaced, but this will take time.
- IPv4 mitigation techniques, as we have seen these techniques push the exhaustion date just a bit further ahead, while at the same time working devastatingly against adoption of IPv6 and impeding the use of many application protocols.

These factors are the major ones that may significantly impede progress. Several others exist and some people are even of the belief that a transition is *not* necessary. This is in my opinion a ludicrous thought, especially given the advent of the Internet of things and the increased growth of the numbers and utilisation of mobile devices. All of these concerns will in my opinion soon be dwarfed by the future growth that IPv6 provides because of its address space, as opposed to the limited IPv4 network.

We discussed the main transitional mechanisms namely: dual stack, translation, and tunnelling. The intent of these schemes is to ease in the transition, avoid waste in terms of existing investments in IPv4, to give network operators time to develop a robust IPv6 infrastructure; while most importantly allowing the network to continue to grow. However, there is a possibility that the transition mechanisms will overshadow the original goal, which was to facilitate adoption of native IPv6. ISPs and others may start putting too much effort into optimizing what was meant to be a transitional phase.

IPv6 enables a broad range of innovative application areas. Our discussions open up for the exploration of new application areas such as IPv6 mobility facilitated by SLAAC. Mobility features can specifically aide in fields like emergency rescue, where vital information can be obtained and help prevent a disaster. Also, various monitoring sensors can be installed in homes/buildings and connected to the internet facilitated by the large IPv6 address space. This can also be extended to make transportation easy by monitoring/tracking vehicles. These are just some of many application areas that can be researched.

Spreading awareness about IPv4 address depletion and the need for IPv6 adoption is still of importance as we progress toward the future. Promoting deployment can be done by joining initiatives like World IPv6 Day and the upcoming World IPv6 Launch as we have seen in the previous chapter. If major content providers make their websites available through the IPv6 network (for example, Google still has its IPv6 site up since World IPv6 Day 2011) it will encourage all the other entities to follow.

Bibliography

[1] Cisco Systems Inc., "Cisco Visual Networking Index: Forecast and Methodology, 2010-2015," Cisco Systems Inc., 2011.

[2] IANA, "Number Resources: internet protocol v4 address space," IANA, [Online]. Available: http://www.iana.org. [Accessed 22 march 2012].

[3] C. Semeria, "Understanding IP Addressing: Everything you Ever Wanted to Know," NSD Marketing, 3Com Corporation, 1996.

[4] J. Wegner and R. Rockell, IP Addressing and Subnetting Including IPv6, Syngress Media Inc. 800 Hingham street Rockland, MA 02370, 2000.

[5] D. Wing, "Network Address Translation: Extending the Internet Address Space," *IEEE Internet Computing,* vol. 14, no. 4, pp. 66-70, 2010.

[6] I. v. Beijnum, Running IPv6, Berkeley, CA 94710: Apress Inc. 2560 Ninth street, Suite 219., 2006.

[7] J. Davies, Understanding IPv6, Redmond, Washington 98052-6399: Microsoft Press A Division of Microsoft Corporation One Microsoft Way, 2003.

[8] S. Ansari, S. Rajeev and H. Chandrashekar, "Packet Sniffing: a brief introduction," *IEEE potentials,* vol. 21, no. 5, pp. 17-19, 2002.

[9] P. R. Babu, B. L. Bhaskari and C. Satyanarayana, "A Comprehensive Analysis of Spoofing," *International Journal of Advanced Computer Science and Applications,* vol. 1, no. 6, 2010.

[10] P. Loshin, IPv6: Theory, Protocol and Practice, Second Edition, 500 Sansome Street, Suite 400, San Francisco, CA 94111: Morgan Kaufmann Publishers, 2004.

[11] Y. Mun and H. K.Lee, Understanding IPv6, Springer Science+Business Media, Inc., 233 Spring Street, New York, NY 10013, USA, 2005.

[12] W. Goralski, The Illustrated Network: How TCP/IP Works in a Modern Network, Morgan Kaufmann is an imprint of Elsevier. 30 Corporate Drive, Suite 400 Burlington, MA 01803, 2009.

[13] G. Q. Maguire jr., "Lecture notes for IK1550 Spring 2012," 11 May 2012. [Online]. Available: http://www.ict.kth.se/courses/IK1550/Coursepage-Spring-2012.html#lectureplan. [Accessed 11 May 2012].

[14] S. Deering and R. Hinden, "Internet Protocol, Version 6 (IPv6) Specification," The Internet Society, RFC Editor, RFC 2460, 1998.

[15] S. Thomson, T. Narten and T. Jinmei, "IPv6 Stateless Address Autoconfiguration," The Internet Society, RFC Editor, RFC 4862, 2007.

[16] R. Droms, J. Bound, B. Volz, T. Lemon, C. Perkins and M. Carney, "Dynamic Host Configuration Protocol for IPv6 (DHCPv6)," The Internet Society, RFC Editor, RFC 3315, 2003.

[17] D. Johnson, C. Perkins and J. Arkko, "Mobility Support in IPv6," The Internet Society, RFC Editor, RFC 6275, 2011.

[18] N. Olifer and V. Olifer, Computer Networks: Principles, Technologies and Protocols for Network Design, John Wiley & Sons Limited, 2005.

[19] K. Nichols, S. Blake, F. Baker and D. Black, "Definition of the Differentiated Services Field (DS Field) in the IPv4 and IPv6 Headers," The Internet Society, RFC Editor, RFC 2474, 1998.

[20] J. Rajahalme, A. Conta, B. Carpenter and S. Deering, "IPv6 Flow Label Specification," The Internet Society, RFC Editor, RFC 3697, 2004.

[21] C. Bouras, P. Ganos and A. Karaliotas, "The Deployment of IPv6 in an IPv4 world and transition strategies," *Internet Research: Electronic Networking Applications and Policy,* vol. 13, no. 2, pp. 86-93, 2003.

[22] M. Lind, V. Ksinant, S. Park, A. Baudot and P. Savola, "Scenarios and Analysis for Introducing IPv6 into ISP Networks," The Internet Society, RFC Editor, RFC 4029, 2005.

[23] APNIC, ARIN and RIPE communities, "http://www.ripe.net," [Online]. Available: http://www.ripe.net/ripe/docs/ripe-545#why_joint_policy. [Accessed 9 April 2012].

[24] J. F. Kurose and K. W. Ross, Computer Networking: A Top-Down Approach, Fifth Edition,International Edition, Boston, MA 02116: Pearson Education, Inc., 2009.

[25] S. Brown, B. Browne, N. Chen, P. J.Fong, R. Harell, E. Knipp, B. Saylors, R. Webber and E. J. Parenti, Configuring IPv6 for Cisco IOS, Syngress Publishing, 2002.

[26] ISOC Monthly Newsletter June 2011, "www.internetsociety.org," [Online]. Available: http://www.internetsociety.org/articles/world-ipv6-day-participation-continues-grow. [Accessed 27 march 2012].

[27] ISOC Monthly Newletter June 2011, "www.internetsociety.org," [Online]. Available: http://www.internetsociety.org/articles/successful-world-ipv6-day-demonstrates-global-readiness-ipv6. [Accessed 27 march 2012].

[28] D. York, "www.internetsociety.org," [Online]. Available: http://www.internetsociety.org/deploy360/blog/2012/01/world-ipv6-launch-on-june-6-2012-to-bring-permanent-ipv6-deployment/. [Accessed 27 march 2012].

TRITA-ICT-EX-2012:71

www.kth.se