VAASAN AMMATTIKORKEAKOULU
VASA YRKESHÖGSKOLA
UNIVERSITY OF APPLIED SCIENCES

Samwel Nyangala

IPV6 IMPLEMENTATION

Dual Stack

Technology and Communication

2013

VAASAN AMMATTIKORKEAKOULU

UNIVERSITY OF APPLIED SCIENCES
Degree Program in Information Technology

ABSTRACT

Author	Samwel Nyangala
Title	IPv6 Implementation
Year	2013
Language	English
Pages	66
Name of Supervisor	Antti Virtanen

IPv6 was proposed in the early 1990s to tackle the depletion of IPv4 addressing space. To ease the transition process from IPv4 to IPv6, a number of transition methods were also proposed. The IPv6 Dual Stack implementation is one of these proposed transition techniques.

This thesis was carried out with the goal of creating a teaching and test environment for the Dual Stack implementation of IPv6 and its associated services. An internet connected IPv6 network was intended but due to some difficulties, a Local Area Network became the fall-back plan. A Cisco 7200VXR NPE-G2 and a Linux machine running Ubuntu 12.10 distribution were used as core devices to provide three LANs and a DHCP server. Cisco Catalyst 2950 and 2960 switch were used to provide connection for multiple Windows 7 client nodes in the LANs.

At the completion of the thesis, a three LAN Dual Stack network that is capable of stateless and stateful auto-configurations was created and documented. An easy-to-read theoretical documentation of IPv6 services such as ICMPv6, DHCPv6 and Neighbour Discovery and IPv6 addressing and different IPv6 implementation mechanisms was created. Also documented are the well explained step-by-step implementations of stateless and stateful address auto-configurations.

Keywords: IPv6, Dual Stack, Stateful, Stateless, Autoconfiguration

CONTENTS

LIST OF FIGURES, TABLES AND FRAMES

ABBREVIATIONS

4G/LTE	Fourth Generation Long Term Evolution
AH	Authentication Header
ALG	Application Level Gateways
APIPA	Automatic Private IP Addresses
ARP	Address Resolution Protocol
BIS	Bump in the Stack
DAD	Duplicate Address Detection
DHCP	Dynamic Host Configuration Protocol
DHCPv6	Dynamic Host Configuration Protocol Versi0n 6
DNS	Domain Name System
DSTM	Dual Stack Transition Mechanism
DUID	DHCP Unique Identifier
EUI	Extended Unique Identifier
ESP	Encapsulating Security Payload
FICORA	Finnish Communication Regulatory Authority
IA	Identity Association
ICMP	Internet Control Messaging Protocol
ICMPv6	Internet Control Messaging Protocol Version 6
ID	Identity
IEEE	Institute of Electrical and Electronics Engineers
IKE	Internet Key Exchange
IN	Internet
IP	Internet Protocol
IPv4	Internet Protocol Version 4
IPv6	Internet Protocol Version 6
IPSec	Internet Protocol Security
ISP	Internet Service Provider
LAN	Local Area Connection
MAC	Media Access Control
MLD	Multicast Listener Discovery
MLDv2	Multicast Listener Discovery Version 2

MTU	Maximum Transmission Unit
NAT	Network Address Translation
NAT-PT	Network Address Translation-Protocol Translation
NAPT	Network Address Port Translation
NA	Neighbour Advertisement
ND	Neighbour Discovery
NIC	Network Interface Card
NS	Neighbour Solicitations
OSs	Operating Systems
QoS	Quality of Service
RA	Router Advertisement
RIP	Routing Information Protocol
RIR	Regional Internet Registries
RRs	Resource Records
RS	Router Solicitations
RFC	Request for Comments
RIPE NCC	Réseaux IP Européens Network Coordination Centre
SLAAC	Stateless Address Autoconfiguration
TCP/IP	Transmission Control Protocol/Internet Protocol
UDP	User Datagram Protocol
VAMK	Vaasan ammattikorkeakoulu
VoIP	Voice over Internet Protocol

1 INTRODUCTION

Due to the increasing demand for internet connectivity that is largely driven by the explosion of networked devices, the currently deployed Internet Protocol Version 4 (IPv4) needed to be replaced. The need for switching arises from the short comings in the IPv4 addressing capabilities, a 32 bit addressing space proficient of providing only 4,294,967,296 unique addresses.

IPv6, on the other hand, is a 128 bit addressing mechanism that allows for 2^128 (3.4*10^38) unique addresses. IPv6 was designed with a number of features such as:

- Simplified packet header format.
- Address autoconfiguration, stateless or stateful.
- Integrated and enhanced security, IPsec. /6, 2. /

To ease this transition process from IPv4 to IPv6, a number of IPv6 implementation methods have been introduced. The implementation of Dual Stack, a mechanism which enables the interoperability of both Internet Protocols on the same network, will be discussed in this document. This paper goes ahead to document a lengthy discussion and implementation of Stateless Address Autoconfiguration and Stateful Address Autoconfiguration mechanisms of IPv6. The project was carried out in LEC3, a telecommunications laboratory facility for Vaasan ammattikorkeakoulu (VAMK), at the Technobothnia laboratory. Upon completion, the project will be used as a teaching test environment for IPv6 addressing, services and concepts.

2 PROJECT BACKGROUND

This chapter pronounces the need for the project by giving a broader view of the topics discussed in the introduction and the relevancy of the project to VAMK. Reasons for choosing the project and the general project flow are also discussed.

2.1 Internet Protocol Version 4 (IPv4)

Internet Protocol (IP) is a set of rules and operations that enable the communication of nodes on a network. IPv4, introduced 1981, is the currently widely deployed version of IP. It is a 32 bit field addressing space expressed in a bit decimal notation such as 127.0.0.1, 255.255.255.0 etc.

There are a total of 35.078 /8 address blocks, out of the available addresses, that are 'reserved' leaving 220.922 /8 address blocks available for use in the public IPv4 Internet /5, pp.3–6/. Short term solutions such as Network Address Translation (NAT), which allows the reuse of private addresses within a company, and classless addressing, were introduced as a remedy for the depletion. RIPE NCC keeps a weekly updated interactive graph showing the available IPv4 addresses pool and more information at its website /19/ while Geoff Huston, an adjunct researcher at the Advanced Internet Architecture, keeps a daily updated report for the exhaustion of IPv4 addresses at his site /9/.

2.2 Internet Protocol Version 6 (IPv6)

IPv6 was developed as a long term solution to the depletion problem of IPv4. In comparison to IPv4, IPv6 uses a 128-bit addressing mechanism and adds a number of other features that give it an edge over IPv4. IPv6 addresses are represented in a colon hexadecimal format i.e. fe80::6120:9e85:4ddb:749f or FE80:0000:0000:0000:6120:9E85:4DDB:749F without abbreviations. IPv6 is the future of packet-switched internetworking and is at the moment being deployed alongside the existing IPv4 infrastructure.

The greatest of challenges though lies in the implementation of a total transition from IPv4 to IPv6. There are deployment approaches, such as tunnelling, transla-

tion and dual stack that are being used today. However, there has been support for IPv6 in personal computers and servers from major Operating Systems makers since the early 2000's.

2.3 Reasons for Choosing the Project

As demonstrated in the previous paragraphs, IPv6 is the future in packet-switched networking and it is imperative for future thinking companies like VAMK to embrace the technology. There are several ways to implement IPv6 but Dual Stack was chosen for the following reasons:

- Most practical approach: With dual Stack, VAMK will be able to communicate with both companies that have migrated to IPv6 and those that have not.
- The need for a teaching environment for IPv6 concepts at VAMK so as to equip its students with top class skills for the competitive labour market.
- With dual stack, it is easy to supress one protocol according to needs for example turning off IPv4 when a full migration to IPv6 happens. /20/

2.4 Project Flow

The project undertaking will be divided into different phases as mentioned below.

- Firstly there will be the research phase which is comprised of gathering of relevant information regarding the project and different technologies used therein.
- The second phase will involve the auditing of different network devices for compatibility with IPv6. Updates and installations of needed software and services will be done and a small scale network implemented.
- The Final phase, not chronologically, will be the documentation and presentation of findings, tests and results to students and teachers.

The technologies and devices listed below were used to carry out the project and are introduced in relevant details in the subsequent chapters.

- IPv6 features
- IPv6 address types
- Internet Control Messaging Protocol
- Dynamic Host Configuration Protocol
- Neighbour Discovery
- Domain Name System
- IPv6 Implementation schemes
- Cisco 7200VXR NPE-G2 Network Processing Engine
- Cisco Catalyst 2950 24 Switch
- Cisco Catalyst 2960 24 Switch
- Windows 7 and Linux (Ubuntu 12.4) hosts

3 TECHNOLOGY REVIEW

This section discusses the different technologies that were used in the implementation of the project. IP is not a standalone protocol and thus needs the embodiment of services, such as ICMP, DHCP and DNS. The details and lengths of the discussions of these technologies are limited to the relevance of their understanding and implementation herein. It is advised that one visits the different Request for Comments (RFC) documentations for a detailed discussion on these features.

3.1 Internet Protocol Version 6 Features

As already discussed briefly under section 2.2, the new protocol is a complete redesign that brings major changes with it. This section is dedicated to the introduction and brief discussion of some of these changes.

3.1.1 Address Autoconfiguration

IPv6 supports both stateless and stateful address configurations. The stateful address configuration is performed in the presence of a DHCP server while the stateless address configuration is in the absence thereof /7/. The stateless address configuration involves the automatic configuration of link-local addresses (on-link IPv6 addresses) and unicast addresses from prefixes in the local router advertisements /21, 3/.

3.1.2 IPv6 Header Format

The IPv6 header is designed to keep header overhead to a minimum by moving optional and non-essential fields to extension headers after it. This results into efficient processing of the IPv6 header at intermediate routers. The header format is as represented in Figure 1. /6, 4–5. /

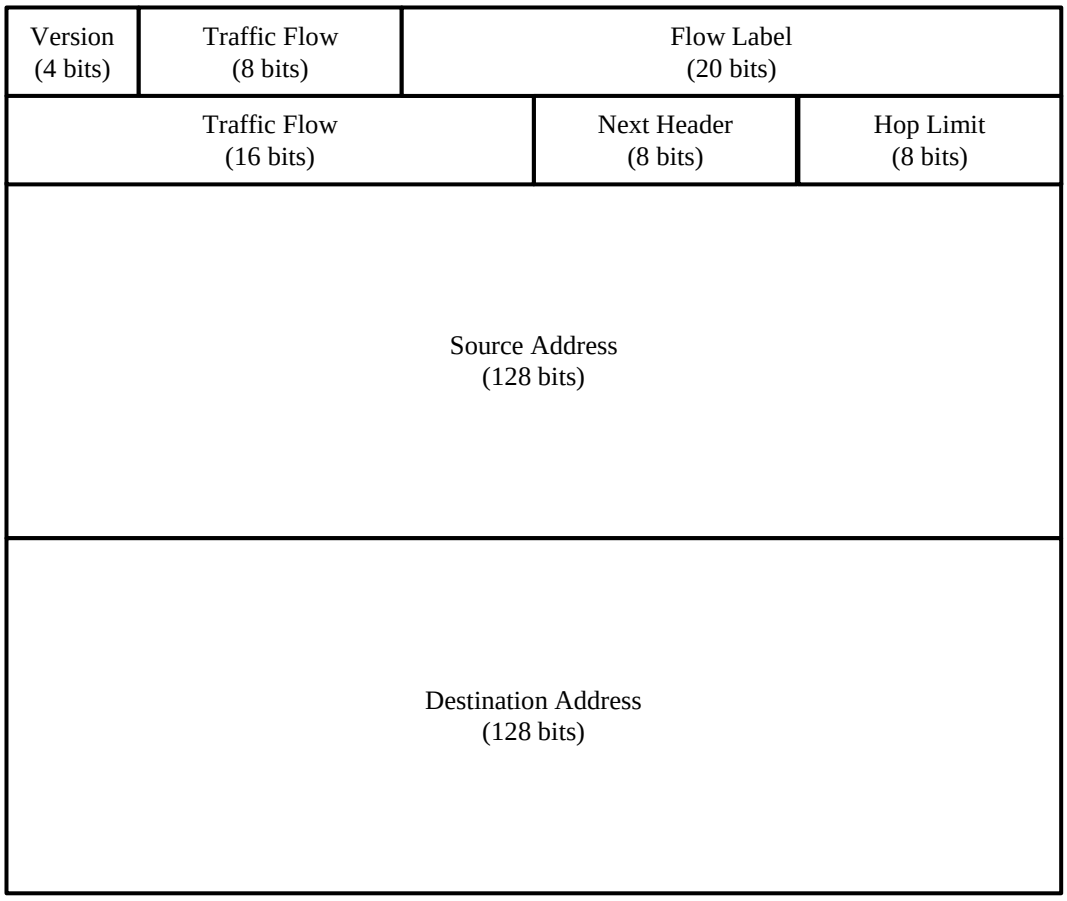

Figure 1. IPv6 Header format. /6/

3.1.3 Large Address Space and Better End-To-End Connectivity

The large 128-bit IPv6 addresses are designed for multi-level sub-netting from the internet backbone to an individual subnet in an organization. With IPv6, support for peer-to-peer applications, such as VoIP, file sharing, video streaming is improved since each end user has a unique IP address and a true end-to-end experienced is delivered. /11/

3.1.4 Better and Integrated Security

In IPv4 implementation, security was left in the charge of end devices with optional features, such as NAT, NAPT and IPSec. IPSec, however, is a major requirement of IPv6 to provide standardized security between different implementations.

IPSec ensures secure key exchange and data communications by using a set of cryptographic protocols which are:

- Internet Key Exchange (IKE) protocol which initiates and negotiates security parameters between two endpoints and tracks the information to ensure secure communications.
- Encapsulating Security Payload (ESP) protocol which ensures authentication, integrity and privacy of data.
- Authentication Header (AH) protocol for data authentication and integrity.

3.1.5 Better Quality of Service

IPv6 replaces the default Type of Service field with a 20-bit Flow Label field in which contains information of how particular packets are to be handled by the IPv6 routers. This results in a high degree of QoS by ensuring efficient delivery of information from one point to another without intermediary modifications. /6, 25/. /11/

3.1.6 Next Header and Extensibility

IPv6 introduces extensions which are indicated in the header by the Next Header field to carry optional internet-layer information. These extensions are placed between the IPv6 header and the header of an upper-layer protocol. The extension headers are not limited to a certain size i.e. they are an integer multiple of 8 octets long, but must not exceed the total size of the IPv6 packet.

With the exception of the Hop-by-Hop option header, all extensions are only processed at the destination node as indicated by the Destination Address field and must be processed strictly in the order they appear in the packet. This fact greatly improves the processing time of the IPv6 packets along their paths and increases throughput. Figure 2 illustrates the placing of extension headers in an IPv6 packet. /6, 6–8/, /11/

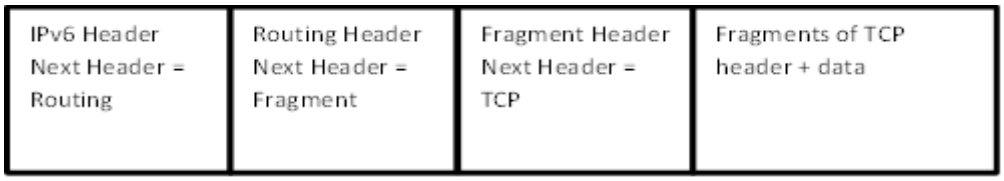

Figure 2. Illustration of IPv6 extensions. /6/

3.1.7 Administrative Ease

IPv6 includes an automatic network renumbering technique as opposed to the manual network renumbering in IPv4. The network renumbering is the replacement of an existing prefix throughout the network by a new one, preferably of the same size. It usually happens when a network changes its Internet Service Provider (ISP), merging of two networks due to acquisitions or when a network is expanding.

IPv6 also supports multi-homing, a condition in which simultaneous connections are established to two ISPs. This increases redundancy and thus provides reliable communications. /11/

3.2 IPv6 Addresses

There are three major types of IPv6 addresses i.e. unicast, multicast and anycast addresses which can be categorized by scope and type. For unicast and anycast addresses, the following scopes are true:

- Link-local. This scope extends to all nodes on the same subnet, technically called local link. Link-local addresses are used to communicate to on-link neighbours and for Neighbour Discovery processes.
- Global. This scope is equivalent to the public IPv4 address space covering the whole IPv6 portion of the internet.

There are also special IPv6 addresses, i.e. loopback and unspecified addresses, whose scope depend on the type of the address. An IPv6 host is identified by one of the unicast addresses assigned to one of its interfaces. /10, 8/

3.2.1 Unicast IPv6 Addresses

Unicast IPv6 addresses identify a single interface within its scope and enable the delivery of packets to that interface. These denote a one-to-one communication. Unicast addresses are divided into different types such as global, link-local, 6to4 unicast addresses etc.

3.2.1.1 Unicast Global Addresses.

IPv6 unicast global addresses, also known as aggregatable global unicast addresses, are globally reachable and routable. They take a three-level topology structure as illustrated in Figure 3.

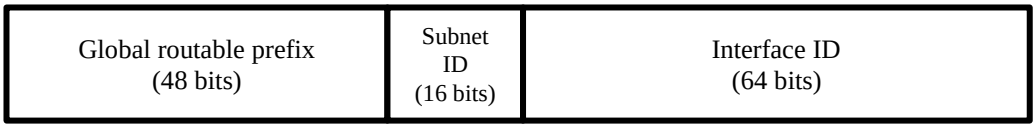

Figure 3. IPv6 Unicast global address structure. /10/

The global routable prefix is a globally unique value that is assigned to the site while the subnet ID is assigned within the site to links/subnets and the interface ID defines an interface of a node in the subnet. /10, 9–10/

3.2.1.2 Unicast Link-local Addresses

Link-local addresses, FE80::/64, are automatically configured and used for on link communications and the neighbour discovery process. They are similar to IPv4 APIPA (Automatic Private IP Addresses) addresses which are automatically configured where there are no other means of address configuration i.e. manual or DHCP. Their structure is as illustrated in Figure 4. /10, 11/

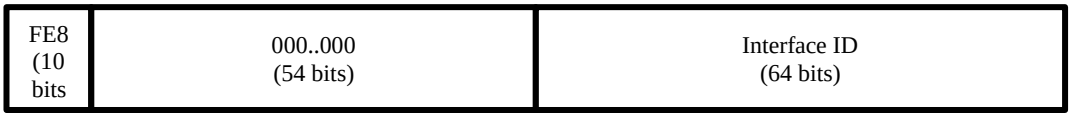

Figure 4. IPv6 Unicast link-local address structure. /10/

3.2.1.3 Unicast 6to4 Addresses

A 6to4 address concatenates the 2002::/16 prefix to the host's 32 bit IPv4 public address to make a 48-bit prefix 2002:WWXX:YYZZ::/48, with WWXX:YYZZ representing w.x.y.z in colon-hexadecimal notation. They are used for communications between two IPv6/IPv4 nodes over IPv4 internet. The address 193.166.97.143 translates to a 6to4 address prefix of 2002:C1A6:618F::/48. 6to4 is a tunnelling method as described in section 3.8.3.

The IPv6 unicast loopback address ::1 enables a node to send packets to itself and test TCP/IP functionality. The IPv6 unicast unspecified address, 0:0:0:0:0:0:0:0 or ::, is used to indicate the absence of an address. It is typically used as a source address for Duplicate Address Detection (DAD), when a host wants to prove the uniqueness of its tentative address. The unicast unspecified address is never used as a destination address or assigned to an interface. /10, 8/

3.2.2 Multicast IPv6 Addresses

Multicast IPv6 addresses, FF00::/8, identify many interfaces and enable the delivery of packets to these interfaces. IPv6 nodes are free to join or leave a multicast group and can listen to multiple multicast addresses at a time. Beyond the first eight bits of an IPv6 multicast addresses "FF" are structures to identify flags, scopes and multicast groups as illustrated in Figure 5. Multicast IPv6 addresses denote a one-to-many communication.

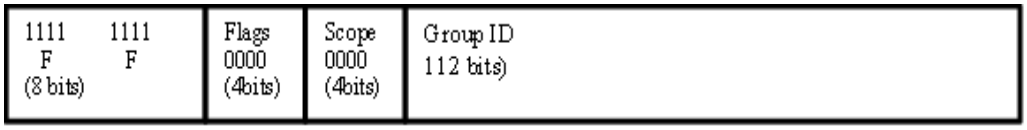

Figure 5. IPv6 Multicast addresses structure. /10/

The first 8-bits "FF" indicates that the address is an IPv6 multicast address. Flags on a multicast address are represented in the next 4-bits. The currently defined flags are Transient (T), Prefix (P) and Rendezvous (R) flags. The Transient flag indicates whether the multicast address is permanently assigned "0" or transient "1". The Prefix flag indicates if the multicast address is based on a unicast address

prefix. The Rendezvous point address flag indicates if the address includes a rendezvous point address.

The scope bits indicate the portion of the network for which the multicast traffic is intended i.e. 1 for interface-local, 2 for link-local and 5 for site-local. Routers use the multicast scope to make forwarding decision for multicast traffic. Most notable multicast groups are the link-local all-nodes multicast group, FF02::1, the link-local all-routers multicast group, FF02::2, the RIP routers multicast group FF02::9, the all MLDv2 routers address FF02::16 etc.

The IPv6 multicast solicited-node address is comprised of a FF02::1:FF00::/104 prefix and the last 24 bits of a unicast or anycast address. An IPv6 is required to join a solicited-node multicast address for all unicast and anycast addresses configured on its interfaces. For an IPv6 node with the unicast addresses FC00:22:22:0:B137:13B9:327B:A8AD will join the solicited-node addresses FF02::1:FF7B:A8AD. The solicited-node multicast address is used, instead of the full unicast address, for an efficient DAD process and for resolving IPv6 link-local addresses into link-layer addresses to maintain neighbour reachability. /10, 13–17/

3.2.3 Anycast IPv6 Address

IPv6 anycast addresses use the unicast address space but identify multiple interfaces and deliver packets to the nearest in terms of routing distances. The scope of an anycast address depends on the unicast address from which it is assigned. Anycast addresses are currently assigned to routers and are only used as destination addresses. They denote a one-to-one-of-many communication.

A Subnet-Router anycast address, derived from the subnet prefix for a given interface, is required and assigned to all interfaces of routers in the subnet. It is derived by keeping the bits in the subnet prefix at their values and setting all the other bits to 0. It is used to communicate to one of the routers in a remote subnet.

Different IPv6 nodes are assigned different addresses of different types for different uses. IPv6 hosts, unlike IPv4 hosts, are multi-homed i.e. they can have multi-

ple unicast addresses for an interface let alone each interface having its own set of multicast addresses. Possible addresses that can be assigned to a node's interface are shown in Table 1.

Table 1. Possible addresses that can be assigned to an IPv6 node. /10/

Node	**Address**
Host	Unicast addresses assigned to a host are: • A link-local address for each interface • Unicast (site-local or global) addresses for each interface • Loopback address for the loopback interface (::1) Hosts also listen for traffic on these multicast addresses: • Interface-local scope all nodes multicast address (FF01::1) • Link-local scope all nodes multicast address (FF02::1) • Solicited-node address for each unicast address on each interface • Multicast addresses of joined groups on each interface
Router	Routers have the following unicast addresses • A link-local address for each interface • Unicast (site-local or global) addresses for each interface • Subnet-Router anycast address • Loopback address for the loopback interface (::1) • Additional anycast addresses (optional) Routers also listen for traffic on these multicast addresses • Interface-local scope all nodes multicast address (FF01::1) • Interface-local scope all routers multicast address (FF01::2) • Link-local scope all nodes multicast address (FF02::1) • Link-local scope all routers multicast address (FF02::2) • Site-local scope all routers multicast address (FF05::2) • Solicited-node address for each unicast address on each interface • Multicast addresses of joined groups on each interface

3.3 Internet Control Message Protocol Version 6 (ICMPv6)

This section discusses the format of control messages used in ICMPv6. There are two message types in ICMPv6 i.e. error and informational messages. IPv6 nodes use ICMPv6 to report errors that are met while processing packets and other internet layer diagnostics functions. ICMPv6 also provides a platform for Multicast Listener Discovery (MLD) and Neighbour Discovery (ND) processes.

3.3.1 ICMPv6 Message Format

An ICMPv6 message comes after an IPv6 header and its extension headers. A Next Header value of 58 in an IPv6 extension header immediately precedes an ICMPv6 extension header. The general format of an ICMPv6 message is as indicated in Figure 6.

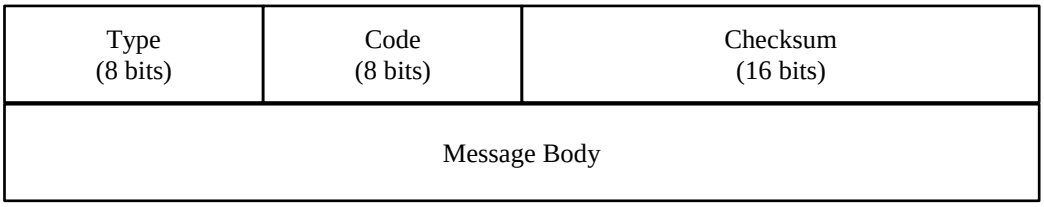

Figure 6. ICMPv6 message structure. /4/

The 8-bits Type field indicates the message type in which the high-order bit is set to 0 for error messages, 0 to 127, and 1 for informational messages, 128 to 225. The 8-bits Code field differentiates messages based on their types. If there is only one message for a given type, the code field is set to 0. The 16-bits Checksum field stores a checksum of the ICMPv6 message that is used to check for corruption in the message and parts of the IPv6 header. The Message Body field is of varying size and contains ICMPv6 message-specific data. The total size of the ICMPv6 message is however limited to the link's MTU. /4, 2–4/

3.3.2 ICMPv6 Error Messages

Table 2 shows different ICMPv6 error messages, their types, codes and description. Some of these error messages were received when carrying out deployment tests and are further discussed in those sections. /4, 7–12/

Table 2. ICMPv6 Error Messages. /4/

Message	Type	Code	Description
Destination Unreachable (sent by a router or a destination host when the packet cannot be forwarded to its destination)	1	0	No route to destination
		1	Administratively prohibited destination
		2	Not assigned
		3	Unreachable address
		4	Unreachable port
		5	Packet not allowed filtering policies
		6	Packet discarded because it marched a rejected route.
Packet Too Big (sent when the packet cannot be forwarded due to a smaller link MTU than of the IPv6 packet)	2	N/A	No code field is defined
Time Exceeded (typically sent by a router when the Hop Limit field is zero, either upon receipt or after decrementing its value)	3	0	Hop limit decreased to zero
		1	Fragment reassembly time exceeded
Parameter Problem (sent by a router or destination host when an error is encountered in either IPv6 header or an extension header)	4	0	Error in IPv6 or extension header
		1	Unrecognized Next Header field encountered
		2	Unrecognized IPv6 option encountered

3.3.3 ICMPv6 Informational Messages

Table 3 shows ICMPv6 informational messages, their types, codes and description. These messages are used for ND and MLD processes. These messages will be analysed further in chapter 4. /4, 12–14/

Table 3. ICMPv6 Informational Messages. /4/

Message	Type	Code	Description
Echo Request	128	0	Sent to a destination to solicit an immediate echo reply message
Echo Reply	129	0	Sent in response to an echo request message
Multicast Listener Query	130		Used in Multicast Listener Discovery and Multicast Group Management Protocol
Multicast Listener Report	131		
Multicast Listener Done	132		
Router Solicitation	133		Sent by nodes when joining networks to search for routers
Router Advertisement	134		Sent by router for periodic advertisements or in response to a solicitation message
Neighbour Solicitation	135		Sent by nodes for Neighbour Solicitation message advertisement and detecting Address Duplication
Neighbour Advertisement	136		Sent by nodes in response to Neighbour Solicitation messages
Redirect	137		Sent by routers to inform sending hosts of a better rout to destination

3.4 Neighbour Discovery Protocol (ND)

On-link IPv6 nodes use Neighbour Discovery for address autoconfiguration, duplicate address detection, discovering other nodes, discovering each other's link-layer addresses, discovering available routers and DNS servers and to maintain

reachability information about the paths to active neighbours. IPv6 Neighbour Discovery protocol relates to a number of the IPv4 protocols i.e. ARP, ICMP Router Discovery and ICMP Redirect. The Neighbour Discovery protocol employs five ICMPv6 packet types to perform these functions. /17/

Neighbour Discovery messages use ICMPv6 message structure and ICMPv6 types 133 through 137. They are made up of a Neighbour Discovery message header, which consists of an ICMPv6 header and ND message specific data, and zero or more ND options. The general structure of a Neighbour Discovery message is illustrated in Figure 7.

Figure 7. Neighbour Discovery message structure. /17/

3.4.1 Neighbour Solicitation Messages

Nodes learn of the link-layer address and reachability of a neighbour by sending out Neighbour Solicitation messages. Neighbour Solicitations are also used for Duplicate Address Detection. /17, 22–23/

3.4.2 Neighbour Advertisement Messages

Upon receiving a Neighbour Solicitation message, nodes send out Neighbour Advertisements in response. These responses carry answers to information request from Neighbour Solicitation messages. Unsolicited Neighbour Advertisements are sent to advertise new link-layer addresses. /17, 23–25/

3.4.3 Router Solicitation Messages

Newly connected host interfaces, if configured to, send out Router Solicitations to request for RAs straightaway rather than at their next scheduled time. This helps new hosts to speed up their configuration times. /17, 18–19/

3.4.4 Router Advertisement Messages

Routers use these messages to advertise their presence and other link and Internet parameters to on-link nodes. Some of the advertised parameters are prefixes that are used for on-link determination and address configuration, a suggested hop limit value, link MTU, etc. These messages are sent either periodically or in response to a Router Solicitation message. Address autoconfiguration is enabled by router advertisement messages. /15, 19–22/

3.5 Address Autoconfiguration

IPv6 hosts can automatically configure their addresses without the need of a DHCP server or manual configurations from an administrator through the address autoconfiguration process. To perform this, the host first generates an interface identifier and appends it to a reserved link-local prefix to create a link-local address. Other addresses are then generated by appending the interface identifier to advertised prefixes from Router Advertisement, from DHCPv6 servers or both. The use of router advertisements for address configuration is known as Stateless Address Autoconfiguration (SLAAC) while using a DHCPv6 server is Stateful Address Autoconfiguration.

3.5.1 Interface Identifier

IPv6 nodes can be identified by the interface identifiers from their unicast addresses. These interface identifiers are required to be unique within a subnet prefix i.e. the same interface identifier must not be assigned to multiple nodes in a link. The interface identifier for all unicast addresses that use the prefixes 001 through 111 must be 64 bits long. There are a number of ways by which a node generates a Modified EUI-64 format interface identifier, such as:

i. The IEEE EUI-64
ii. The IEEE 802 48-bit MAC

For an IEEE EUI-64 identifier, the global/local bit, “u”, needs to be inverted to create an interface identifier. The “u” bit (0 = globally administered and 1 = local)

distinguishes between an IEEE EUI-64 identifier and an IEEE EUI-64 derived interface identifier. Figure 8 illustrates the construction of a Modified EUI-64 interface identifier from an IEEE EUI-64 identifier. In Figure 8, “c” represents the assigned company_id bits, “0 or 1” is the value assigned to the “u” bit, “g” represents the group/individual bit and “e” represents manufacture selected extension identifier. /10, 7–9/

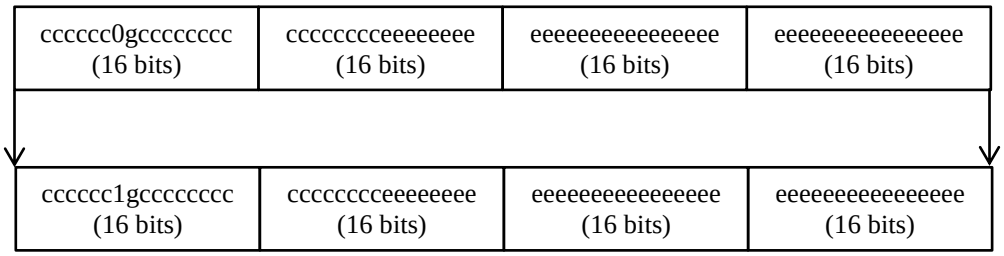

Figure 8. Formation of a Modified EUI-64 interface identifier. /13/

The IEEE 802 48-bit MAC address can be used to generate an IEEE EUI-64 identifier by inserting the hexadecimal value 0xFFFE after the first 24 bits i.e. between the company_id and vendor-supplied id. The “u” bit is then changed to indicate the scope of the interface identifier as described in the previous paragraph and effectively form a Modified EUI-64 interface identifier. Figure 9 illustrates the formation of a Modified EUI-64 using this approach. The Ubuntu 12.04 host, in which the DHCP service was run, uses this approach of interface identifier generation. /10, 7–9/

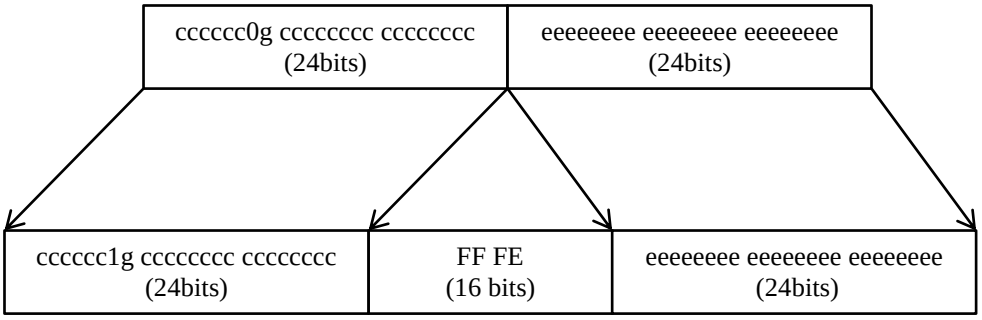

Figure 9. Modified EUI-64 interface identifier from an IEEE 802 48-bit MAC. /13/

The modified EUI-64 interface identifiers, generated from IEEE EUI-64 and IEEE 802 48-bit MAC identifiers, are permanent and thus poses a security risk i.e. if tracked they can be used to identify a node or user. RFC 4941 proposes the use of a pseudo-random sequence to generate interface identifiers to address the security issues presented by other techniques that maintain a permanent interface identifier. A temporary pseudo-random identifier adds security from potential eavesdroppers. Hosts running Windows Vista, Windows 7 and later distributions of Windows use this as the default approach for IPv6 address assignment. /16, 8–15/

Having generated the 64-bit interface identifier, the host continues the autoconfiguration process by:

a. Create a 128-bit link-local address by appending the generated 64-bit interface identifier to a 64-bit link local prefix, FE80::/64.
b. The host then tests the uniqueness of this address, tentative, by sending out Neighbour Solicitation messages and waits for Neighbour Advertisement messages. Receiving an NA message implies that the generated address is not unique and the host has to find other ways to obtain configurations i.e. DHCP.
c. If the address is unique, the host stores it for local communications and sends Router Solicitation messages to on-link routers to obtain a global address. The router responds with a Router Advertisement containing a global unicast and a subnet prefix which the host uses with its interface identifier to generate a global unicast address. The Router Advertisement

may also contain variables informing the host to obtain its prefix by other means i.e. DHCP. /6/

3.5.2 States of an Auto configured Address

The state of an auto configured address could be tentative, preferred, deprecated, valid or invalid. These states are time defined and Figure 10 illustrates how they are related.

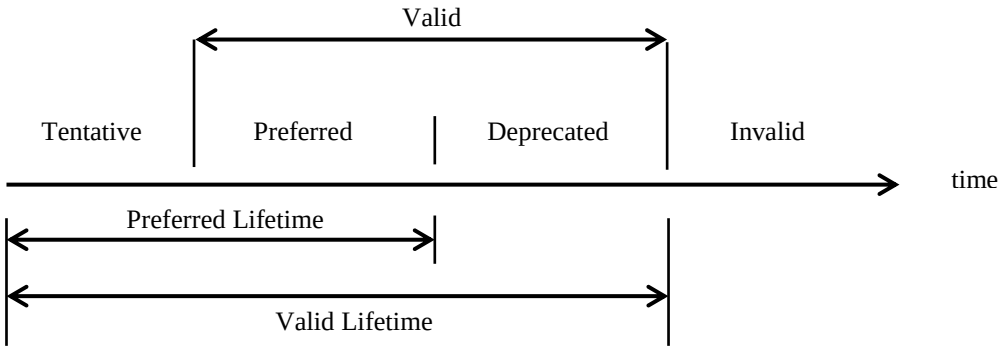

Figure 10. States of an auto configured address over time. /12/

The tentative address is in the process of being verified as unique. This verification is through the DAD process. Packets addressed to this address, except Neighbour Discovery related packets, are dropped.

A preferred address has been verified as unique and the node can send and receive unicast packets to and from it. The use of a preferred address by upper-layer protocols is unrestricted.

Use of a deprecated address, even though it is assigned to an interface, is discouraged for new communications but not forbidden. Existing communication sessions can still use it especially where address switching can cause hardship to upper-layer activities e.g. TCP connections. Both preferred and deprecated addresses are considered as valid addresses and may be used as source or destination addresses.

An invalid address cannot be used as a source or destination for unicast packets. An address becomes invalid after its valid lifetime has expired. A time for which

an address remains valid is the valid lifetime. This time must be greater than or equal to the preferred lifetime. The preferred lifetime is that in which an address is preferred i.e. before it becomes deprecated. These times are included in router advertisement messages. /12/, /21/

3.6 Dynamic Host Configuration Protocol Version 6 (DHCPv6)

Devices operating in IPv6 networks use the client/server protocol DHCPv6 for managed configurations. The IPv6 nodes can obtain addresses and other configuration options from a DHCPv6 server in a process called stateful autoconfiguration. DHCPv6 Clients use link-local addresses for DHCPv6 communications while servers use a reserved link-scoped multicast address to receive these communications. Clients use on-link DHCPv6 relay agents to reach off-link servers. DHCPv6 Clients use UDP port 546 while DHCPv6 servers and DHCPv6 relay agents use UDP port 547 /7, 13/.

DHCPv6 communications can involve two or four message exchanges. A four message exchange happens when a client wants to obtain address information from a DHCPv6 server. The clients send out a Solicit message to FF02::1:2 to locate available DHCPv6 servers. Servers that meet the client's requirements respond with an Advertise message. The clients choose a server to send Request messages asking for confirmed assignment of addresses and other configuration information. The server sends a Reply message with the requested information. /7, 7/

A two message exchange happens when a client, that has already obtained IPv6 addresses from a server wants other configuration information, such as a list of available DNS servers. The client sends an Information-Request message with indications that it is willing to commit to an immediate reply, to which a server replies with the requested configuration information. /7, 6–7/

3.6.1 DHCPv6 Multicast Addresses

DHCP relay agents use FF05::1:3, the site-scoped All_DHCP_Servers multicast address to send messages to all servers, when they do not know the unicast ad-

dress of the servers. This multicast group is joined by all servers within the network/organization. DHCP clients use FF02::1:2, the link-local multicast address to communicate to on-link relay agents and servers. All servers and relay agents are members of this multicast group too. /7, 13/

3.6.2 DHCPv6 Message types

DHCPv6 defines a set of message types for communications between servers and clients. These message types are Solicit, Advertise, Request, Confirm, Renew, Rebind, Reply, Decline, Reconfigure, Information-Request, Relay-Forw and Relay-Repl. Messages between clients and servers have an identical fixed format header and a varied format options area as illustrated in Figure 11. DHCPv6 uses status codes to inform clients and servers of the success or failure of requested information. /7, 13–14/

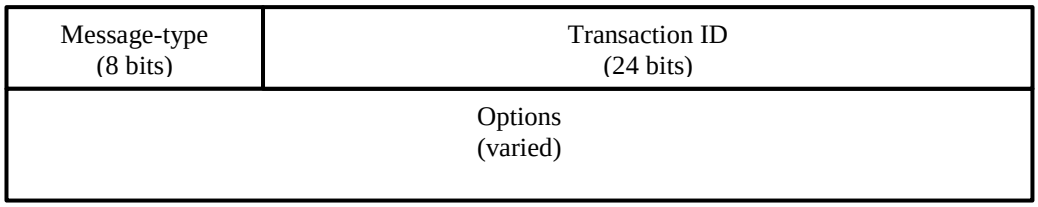

Figure 11. Clients/Servers DHCPv6 message types. /7/

The Message-type field identifies the type of the DHCPv6 message being transferred. Transaction Id field bears the transaction identifier for the message exchange and is used to synchronize the server responses to the client messages. The Options field contains options carried in the message. /7, 17/

The DHCPv6 clients and servers have opaque DHCP Unique Identifiers (DUIDs) that they use to identify each other when there is need. The DUID is carried in options field because it is of varied length and it must be unique for all DHCPv6 clients and servers and must remain the same overtime if possible. /7, 19/

For security reasons, DHCPv6 clients and servers should discard messages that contain options that are not allowed in the received message. A server must discard any Solicit, Confirm, Information-Request or Rebind message it receives

with a unicast destination address i.e. addressed to the servers unicast address. Clients must discard any received Solicit messages while the server must discard Solicit messages that include a Server ID option or do not include Client ID option. /7, 27–28/

3.7 Domain Naming System Version 6 (DNSv6)

DNS is used to resolve queries for resource names into IP addresses and vice versa for location purposes. DNS matches up the domain name hierarchy (host names) and IP addresses namespaces through the internet name servers upon user requests to resolve queries. A new address resource record type, AAAA or quad-A, was defined for IPv6 addresses. IPv6 supports DNS traffic for clients and servers configured using anycast or unicast DNS server IP addresses. The Pointer (PTR) resource record for IPv6, IP6.ARPA which replaces the IP6.INT, was added for reverse queries. /14/

3.7.1 DNS Operation

DNS uses UDP on port 53 for normal traffic and TCP for larger than 512 bytes response data and zone transfers. Query resolutions start at the top-level servers down the hierarchy until the fully qualified domain name (FQDN) such as *tb.technobothnia.puv.fi.* is resolved.

To resolve names, a host application queries a DNS Client service, a resolver, which may respond from a locally saved cache or query a name server in the domain which may answer in one of the following ways:

- respond with an answer from its local cache records
- query other servers until a resolved answer is obtained, utilizing the recursive querying technique (Figure 12)
- send a referral answer pointing to a server that is to be queried by the client, utilizing an iterative query resolution technique (Figure 12)

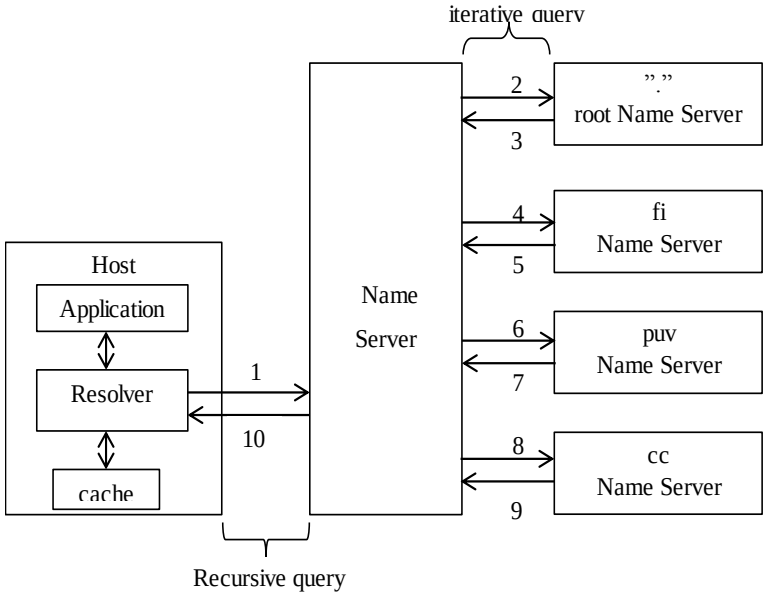

Figure 12. DNS operations showing recursive and iterative queries. /15/

DNS queries contain questions that servers have to answer. These questions are:

- a specified DNS fully qualified domain name
- a specified query type identifying a resource records (RRs) by type or a certain query operation
- a specified class for the DNS domain name, such as the Internet (IN) class for Windows DNS /15/

3.8 IPv6 Implementation Schemes

Many Regional Internet Registries (RIR) and Internet Service Providers (ISP) support IPv6. In Finland, FICORA (Finnish Communication Regulatory Authority) supports IPv6 addresses in its DNS and allows the registration of new domains with IPv6 addresses. Recent advancements in the cellular technology i.e. the introduction 4G/LTE networks that carry voice data as VoIP service are pushing forward the transition to IPv6. To ease the transition, a number of implementation/transition schemes have been proposed and widely accepted and used.

3.8.1 Native Implementation of IPv6

In this implementation, all hosts and routers are configured for operations in an IPv6 only environment. The network is limited to only IPv6 communications and translation is used for communications to IPv4 networks. Unless it is deployed for an intranet, these implementations have very limited practical applications since a very large part of the internet is still under IPv4 deployments.

3.8.2 IPv6 Only to IPv4 Only Translation

When IPv6 only networks need to communicate to IPv4 only networks, translation is needed. Application Level Gateways (ALG) is one of the methods used to accomplish this translation. The ALG uses a server that acts as a proxy to services that are on either IPv6 or IPv4 nodes. The server applications must be IPv6 aware and the server must be configured to support both protocols for ALG to work properly. The positioning of the ALG is determined by the targeted location. Other translation methods include TCP-UDP Relay, Dual Stack Transition Mechanism (DSTM), Bump in the Stack (BIS), NAT-PT and SOCKS-based IPV6/IPv4 gateway. Figure 13 illustrates how ALG might be implemented. /22/

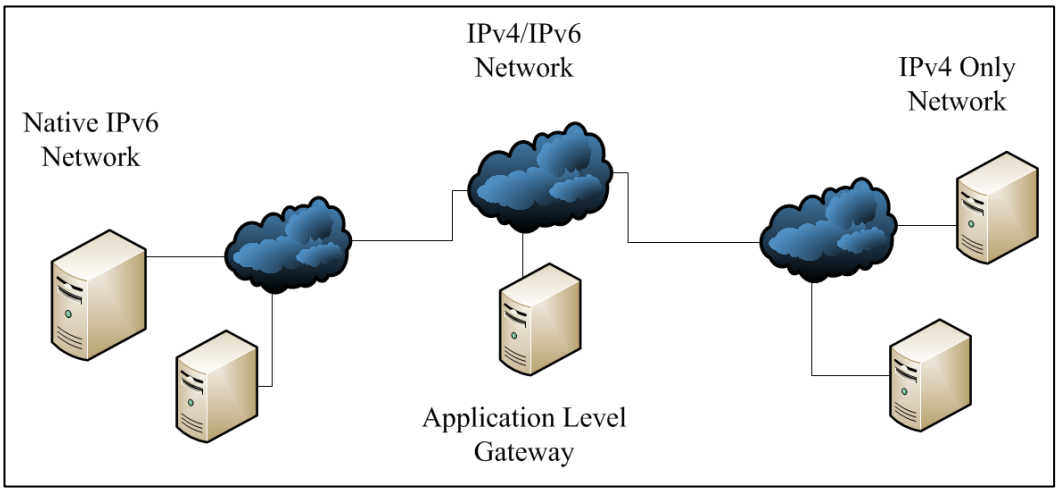

Figure 13. Application Level Gateway Translation. /22/

3.8.3 IPv6 Tunnelling

Tunnelling implementation of IPv6 creates a link between two or more native IPv6 implementations over an IPv4 only network. To enable communications between the IPv6 networks over IPv4 infrastructure, an IPv4 header is added in front of the IPv6 header at the sending router and stripped off at the receiving side of the network, without making changes to the IPv6 packet.

6to4, defined in RFC 3056: Connection of IPv6 Domains via IPv4 Clouds, is one of the most implemented tunnelling protocols. It supports a dynamic tunnelling of IPv6 addresses across IPv4 clouds and utilizes global unicast IPv6 prefixes for each IPv6 site. 6to4 must be configured on the edge routers to map addresses according to their global prefixes, and thus not needing IPv6 route propagation to other sites. Figure 14 displays the implementation of this tunnel and communication between two IPv6 native environments. /22/

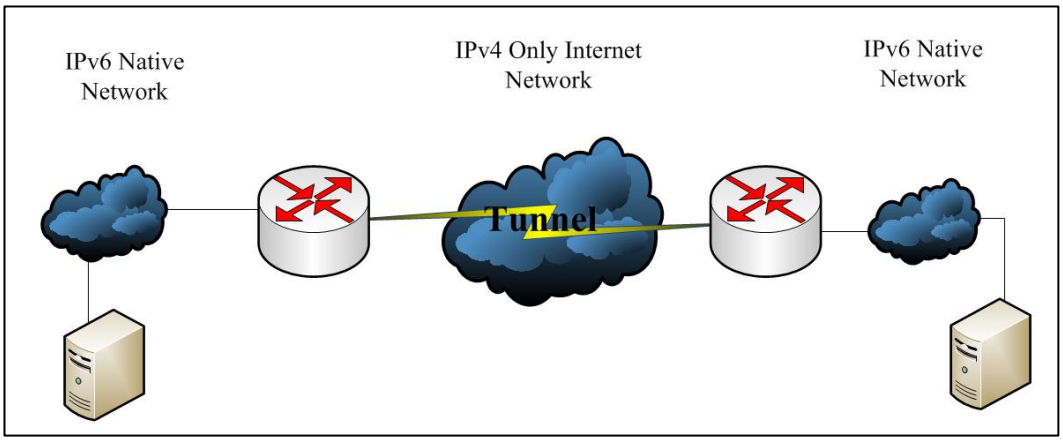

Figure 14. IPv6 Tunnelling over IPv4 only network. /22/

3.8.4 Dual Stack Implementation of IPv6

Dual stack implementation allows the existence of both IPv4 and IPv6 networks on the same physical and/or logical interface. It is the least challenging approach to implement on an already established/existing network environment. To implement Dual Stack, already existing software and hardware on the network must be evaluated to ensure that it meets the IPv6 requirements such as support for IPv6 by host node OSs.

When sending a packet to a destination, the source host queries the DNS to determine which version of IP to use. The host sends either an IPv4 or IPv6 packet depending on the answer it received from DNS /8/. Figure 15 and Figure 16 show the application data flows for an IPv4 only stack and a dual IPv4 and IPv6 stack. /22/

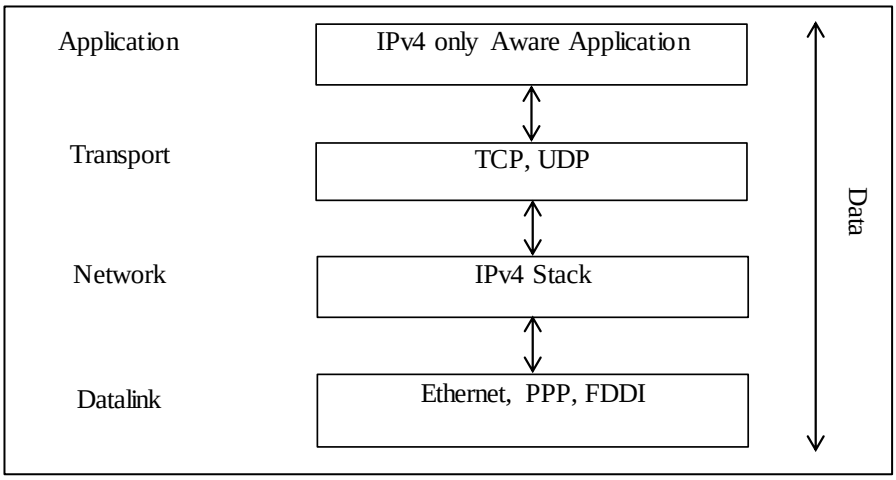

Figure 15. IPv4 only applications data flow. /22/

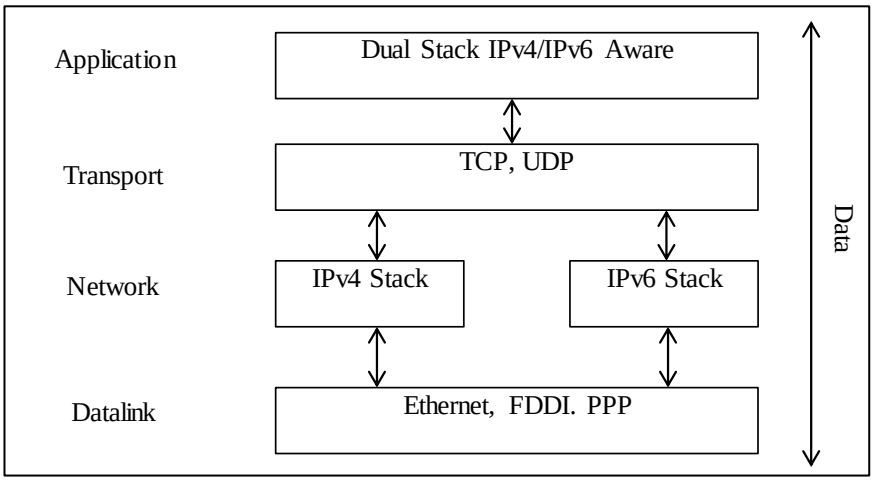

Figure 16. Dual stack aware application data flow. / 22/

4 TESTS AND RESULTS

This section documents the tests undertaken and the achieved results before, during and after the installation of the network. It comprises both software and hardware related examinations performed on both host and network nodes. The terms, as used in this section, DHCP refers to DHCP for IPv4 and DHCPv6 to DHCP for IPv6.

Two network scenarios were configured one with Stateless Address Auto Configuration (SLAAC) and another with Stateful Address Auto Configuration (DHCPv6). In both scenarios however, IPv4 addresses were configured through DHCP (see Appendix B). Figure 17 shows the network diagram for the implementation.

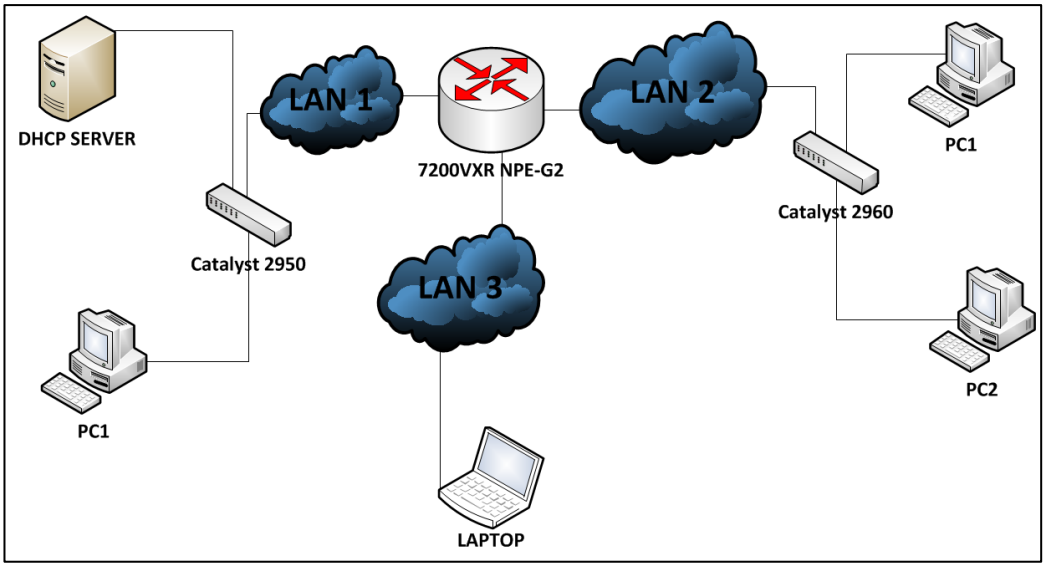

Figure 17. General Network Diagram showing nodes in their LANs.

4.1 Device Audit

A hardware and software audit was carried out to determine if the nodes in the network were suitable to operate in an IPv6 environment. The goals of a hardware audit test is to determine if the nodes have enough memory for route and switch forwarding tables and to handle IPv6 routes. A software audit is to ensure that the nodes support IPv6 configurations and routing protocols.

The majority of the host nodes in the environment in which the network was set up run the Windows 7 OS which supports IPv6 addresses. These nodes are by default configured to obtain their addresses via DHCP but the TCP/IPv6 option in Local Area Connection Properties (Control Panel\Network and Internet\Network Connections) has to be checked to enable IPv6 addressing.

There are a number of resources on the web that can be used to check the router status for IPv6 support, such as the Cisco Feature Navigator and the Cisco IOS IPv6 Feature Mapping. Cisco routers running IOS version 12.0S and greater have support for different IPv6 features. The `show version` and/or `show running configuration` commands will display the router's current IOS version. The router, 7200 VXR, used in this activity runs IOS 15.0 as indicated on the router running configuration file in the Appendix A. The router software version upgrade can be done to obtain a version that supports IPv6 features but one must pay attention to the hardware requirements needed to support IPv6. /2/, /1/

IPv6 packets are transparent to Layer 2 LAN switches because they do not examine Layer 3 packet information before forwarding IPv6 frames. IPv6 hosts can thus be directly attached to Layer 2 LAN switches and have connectivity established. A Dual Stack Switch Database Management Template can be used to manage switch resources better. The `sdm prefer dual-ipv4-and-ipv6` switch configuration command activates the dual IPv4 and IPv6 switch database management template. This feature is, however, not supported on Catalyst 2960 switches running LAN Lite image and its use is not recommended for IPv4 only environments. /3/

4.2 Stateless Autoconfiguration

After having performed a device audit, the router was configured for Stateless Address Auto Configuration (SLAAC) for a three LAN network. To achieve this, `ipv6 unicast routing` was enabled on the router in the global configuration mode and IPv6 addresses were assigned to the router interfaces. When assigning IPv6 addresses to the router, periodic Router Advertisements are automatically sent from that interface. Figure 18 shows the network diagram for the

SLAAC scenario with DHCP obtained IPv4 addresses and auto configured permanent and temporary IPv6 addresses. IPv6 permanent addresses use the link-local address interface identifiers.

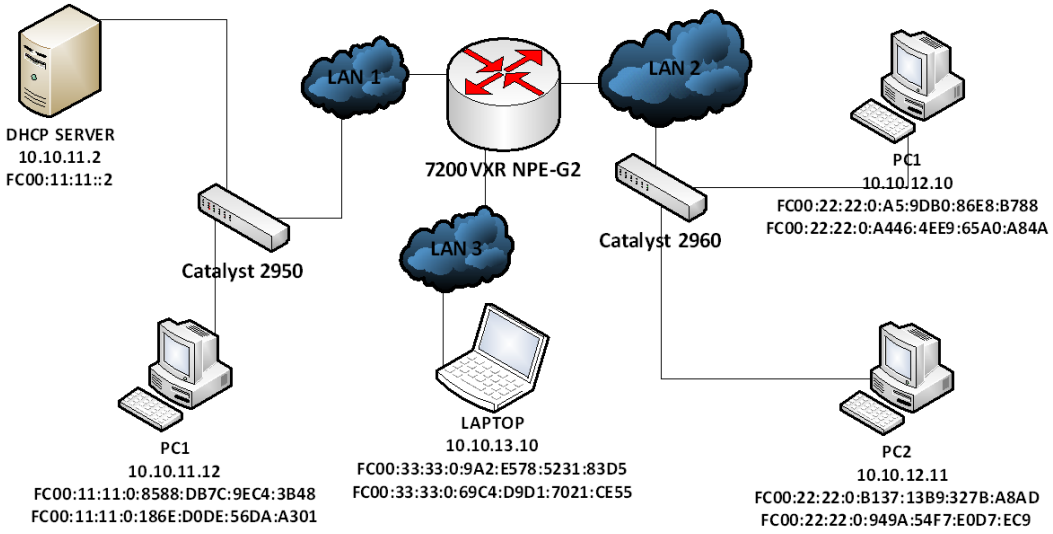

Figure 18. Network diagram with DHCP IPv4 addresses and SLAAC IPv6 addresses.

The MAC addresses of the router interfaces were hard coded with custom values for easy identification purposes. The RIP routing protocol was also configured to enable routing of packets between LANs. RIP in IPv6 has to be enabled at each router interface that is intended to forward packets, unlike in IPv4 where it was done at the global configuration level. The text frame (Frame 1), an extract from routers running configurations, shows the router's GigabitEthernet 0/1 interface configurations. The configurations for interfaces GigabitEthernet 0/2 and 0/3 can be found in the complete router running configurations file (see Appendix A).

```
R7200#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
R7200(config)#int gigabitethernet0/1
R7200(config-if)#description LAN1 Interface
R7200(config-if)#mac-address 0000.1111.1111
R7200(config-if)#ip address 10.10.11.1 255.255.255.0
R7200(config-if)#ipv6 address FC00:11:11::/64 eui-64
R7200(config-if)#ipv6 rip IPV6RIP enable
R7200(config-if)#ipv6 enable
R7200(config-if)#end
```

Frame 1. Interface configuration settings for GigabitEthernet 0/1. (LAN 1)

The `mac-address 0000.1111.1111` command was used to hard code the MAC address of the router interface. The `ipv6 address FC00:11:11::/64 eui-64` command assigns a /64 prefix to the interface to be issued to hosts through Router Advertisements. The command also generates an interface address for the router by concatenating the MAC address to the prefix using the EUI-64 technique. The MAC addresses and IPv6 prefixes pairs for interfaces GigabitEthernet 0/2 and 0/3 are 0000.2222.2222, FC00:22:22::/64 and 0000.3333.3333, FC00:33:33::/64 respectively. The `ipv6 enable` command kick-starts IPv6 processing on the interface and automatically assigns a link-local IPv6 address to that interface without having to configure an explicit IPv6 address. The `ipv6 rip IPV6RIP enable` command enables RIP on the interface. The RIP process for IPv6 requires a name to operate, IPV6RIP for this exercise. The complete IPv6 interface configurations for router interfaces are displayed in frames Frame **2**, Frame **3** and Frame **4**.

```
R7200#sh ipv6 int giga0/1
GigabitEthernet0/1 is up, line protocol is up
  IPv6 is enabled, link-local address is FE80::200:11FF:FE11:1111
  No Virtual link-local address(es):
  Description: LAN1 Interface
  Global unicast address(es):
    FC00:11:11:0:200:11FF:FE11:1111, subnet is FC00:11:11::/64 [EUI]
  Joined group address(es):
    FF02::1
    FF02::2
    FF02::9
    FF02::1:FF11:1111
  MTU is 1500 bytes
  ICMP error messages limited to one every 100 milliseconds
  ICMP redirects are enabled
  ICMP unreachables are sent
  ND DAD is enabled, number of DAD attempts: 1
  ND reachable time is 30000 milliseconds (using 30000)
  ND advertised reachable time is 0 (unspecified)
  ND advertised retransmit interval is 0 (unspecified)
  ND router advertisements are sent every 200 seconds
  ND router advertisements live for 1800 seconds
  ND advertised default router preference is Medium
  Hosts use stateless autoconfig for addresses.
```

Frame 2. IPv6 configurations for GigabitEthernet 0/1 interface.

```
R7200#sh ipv6 int giga0/2
GigabitEthernet0/2 is up, line protocol is up
  IPv6 is enabled, link-local address is FE80::200:22FF:FE22:2222
  No Virtual link-local address(es):
  Description: LAN2 Interface
  Global unicast address(es):
    FC00:22:22:0:200:22FF:FE22:2222, subnet is FC00:22:22::/64 [EUI]
  Joined group address(es):
    FF02::1
    FF02::2
    FF02::9
    FF02::1:FF22:2222
<Output omitted>
```

Frame 3. IPv6 configurations for GigabitEthernet 0/2 interface.

```
R7200#sh ipv6 int giga0/3
GigabitEthernet0/3 is up, line protocol is up
  IPv6 is enabled, link-local address is FE80::200:33FF:FE33:3333
  No Virtual link-local address(es):
  Description: LAN2 Interface
  Global unicast address(es):
    FC00:33:33:0:200:33FF:FE33:3333, subnet is FC00:33:33::/64 [EUI]
  Joined group address(es):
    FF02::1
    FF02::2
    FF02::9
    FF02::1:FF33:3333
<Output omitted>
```

Frame 4. IPv6 configurations for GigabitEthernet 0/3 interface.

The link-local, global and joined group addresses of the router interfaces are displayed in the above text frames. The subnets IDs are the advertised prefixes by which hosts configure their IPv6 addresses. The joined group addresses are: the link-local all-nodes multicast group, FF02::1, the link-local all-routers multicast group, FF02::2, the link-local RIP routers multicast group FF02::9 and the solicited-node multicast address, FF02::1:FF33:3333 for GigabitEthernet 0/3. Figure 19 shows a router advertisement in LAN 2 as captured by Wireshark.

Information such as address prefix, prefix length, address lifetimes, MTU values and router lifetime are all carried in the RA. The on-link flag (L) indicates that the advertised prefix should be added to the host's prefix list. The autonomous addresses-configuration flag (A) informs the host to generate tentative addresses using the advertised prefix. The Router address flag (R) indicates that the prefix filed contains a complete IP address assigned to the sending router /18, 65–66/.

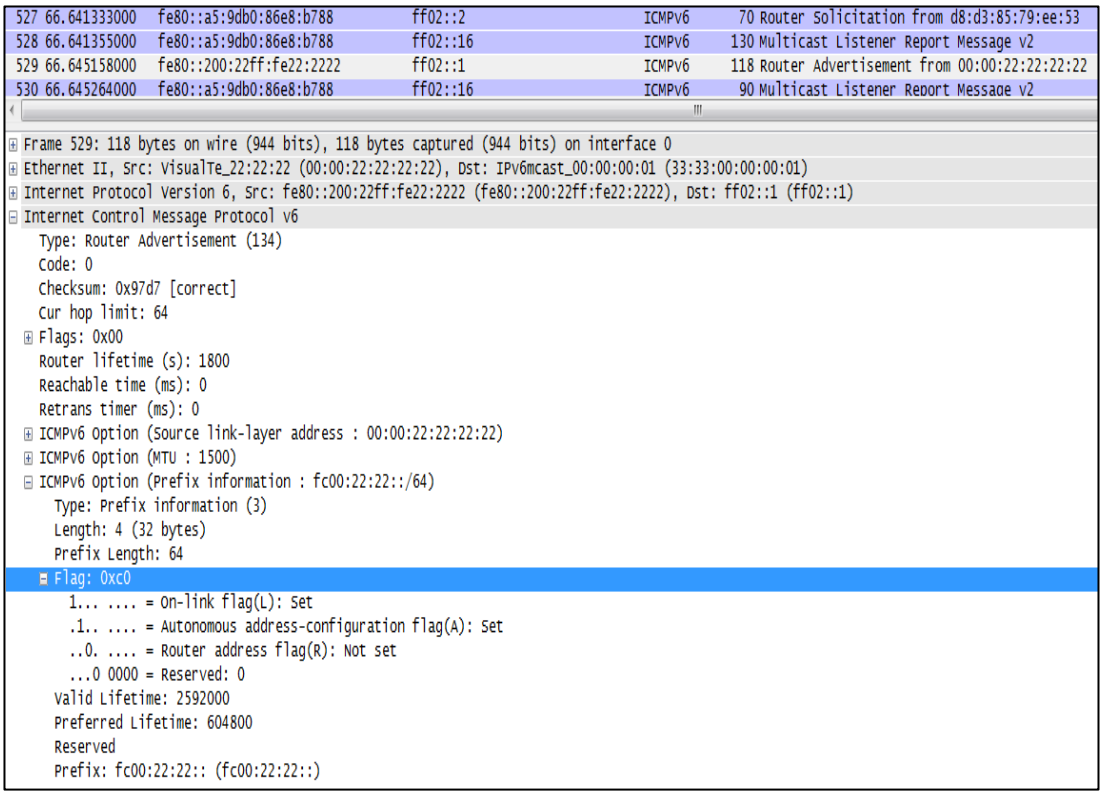

Figure 19. Wireshark capture of a Router Advertisement in LAN 2.

Under this scenario, the IP hosts configure their interfaces with IPv6 addresses as defined in the Router Advertisements and IPv4 addresses from a DHCP server. Three subnets for the three router interfaces/LANs are defined in the IPv4 DHCP configurations file (see Appendix B). The `ip helper-address 10.10.11.2` command was configured to enable DHCP relay agents on the router interfaces without a DHCP server. The corresponding IP configurations for host nodes in LAN1, LAN2 and LAN3 are displayed in frames Frame **5**, Frame **6** and Frame **7**. The text frame (Frame 8) is an extract from the dhcpd.lease file showing the lease information for PC-1 in LAN 2. The complete IPv4 DHCP lease file is in Appendix C. It is important to note that frames Frame **7** and Frame **8** show lease information for the same address but from different times.

```
Ethernet adapter Local Area Connection:

 Connection-specific DNS Suffix  . : example.org
 IPv6 Address. . . . . . . . . . . : fc00:33:33:0:9a2:e578:5231:83d5
 Temporary IPv6 Address. . . . . . : fc00:33:33:0:69c4:d9d1:7021:ce55
 Link-local IPv6 Address . . . . . : fe80::9a2:e578:5231:83d5%10
 IPv4 Address. . . . . . . . . . . : 10.10.13.10
 Subnet Mask . . . . . . . . . . . : 255.255.255.0
 Default Gateway . . . . . . . . . : fe80::200:33ff:fe33:3333%10
                                     10.10.13.1
```

Frame 5. Stateless IP configurations for Laptop in LAN 3.

```
Ethernet adapter Local Area Connection:

 Connection-specific DNS Suffix  . : example.org
 IPv6 Address. . . . . . . . . . . : fc00:11:11:0:8588:db7c:9ec4:3b48
 Temporary IPv6 Address. . . . . . : fc00:11:11:0:186e:d0de:56da:a301
 Link-local IPv6 Address . . . . . : fe80::8588:db7c:9ec4:3b48%10
 IPv4 Address. . . . . . . . . . . : 10.10.11.12
 Subnet Mask . . . . . . . . . . . : 255.255.255.0
 Default Gateway . . . . . . . . . : fe80::200:11ff:fe11:1111%10
                                     10.10.11.1
```

Frame 6. Stateless IP configurations for PC-1 in LAN 1.

```
Ethernet adapter Local Area Connection:

 Connection-specific DNS Suffix  . : example.org
 Description . . . . . . . . . . . : Intel(R) 82567LM-3 Gigabit Net-
work Connection
 Physical Address. . . . . . . . . : D8-D3-85-79-EE-53
 DHCP Enabled. . . . . . . . . . . : Yes
 Autoconfiguration Enabled . . . . : Yes
 IPv6 Address. . . . . . . . . . . : fc00:22:22:0:a5:9db0:86e8:b788
 Temporary IPv6 Address. . . . . . : fc00:22:22:0:a446:4ee9:65a0:a84a
 Link-local IPv6 Address . . . . . : fe80::a5:9db0:86e8:b788%10
 IPv4 Address. . . . . . . . . . . : 10.10.12.10(Preferred)
 Subnet Mask . . . . . . . . . . . : 255.255.255.0
 Lease Obtained. . . . . . . . . . : 18. huhtikuuta 2013 18:05:49
 Lease Expires . . . . . . . . . . : 18. huhtikuuta 2013 18:15:48
 Default Gateway . . . . . . . . . : fe80::200:22ff:fe22:2222%10
                                     10.10.12.1
 DHCP Server . . . . . . . . . . . : 10.10.11.2
 DNS Servers . . . . . . . . . . . : 193.166.140.100
                                     193.166.140.200
                                     8.8.8.8
 NetBIOS over Tcpip. . . . . . . . : Enabled
```

Frame 7. Stateless IP configurations for PC-1 in LAN 2.

```
lease 10.10.12.10 {
  starts 4 2013/04/18 16:36:27;
  ends 4 2013/04/18 16:38:27;
  tstp 4 2013/04/18 16:38:27;
  cltt 4 2013/04/18 16:36:27;
  binding state free;
  hardware ethernet d8:d3:85:79:ee:53;
  uid "\001\330\323\205y\356S";
}
```

Frame 8. DHCP lease information for PC-1 in LAN 2.

Host IP configuration frames reveal that the hosts did obtain their IPv6 addresses from advertised prefixes in their LANs and IPv4 addresses through DHCP. In Frame 7 information that was set in the DHCP server, such as server address, lease times, DNS server addresses is displayed and more host information like the physical address and a description of its NIC card.

From Frame 7 it can be noticed that the interface IDs of the generated IPv6 addresses, `a5:9db0:86e8:b788 and a446:4ee9:65a0:a84a`, have no relation to the node's physical address, `D8-D3-85-79-EE-53`. Furthermore, temporary addresses use a different interface identifier from that used by permanent addresses. This is due to the fact that the Windows OS distributions from Windows Vista and later use pseudo-random technique to generate interface identifiers. The temporary address is used to shield/hide client initiated communications and thus tackle a security issue of permanent addresses or reuse of addresses for an extended time period.

A successful connectivity test was carried out by sending IPv6 and IPv4 ping packets to hosts within a LAN and to inter-LAN hosts the results of which are displayed in the text frame (Frame 9) and Figure 20 and Figure 21. Link-local addresses are not routable and thus cannot be used for inter-LAN connectivity but find use in the Neighbour Solicitation process and for on-link communications.

```
C:\Users\Admin>ping fc00:11:11:0:8588:db7c:9EC4:3B48

Pinging fc00:11:11:0:8588:db7c:9ec4:3b48 with 32 bytes of data:
Reply from fc00:11:11:0:8588:db7c:9ec4:3b48: time=12ms
Reply from fc00:11:11:0:8588:db7c:9ec4:3b48: time<1ms
Reply from fc00:11:11:0:8588:db7c:9ec4:3b48: time<1ms
Reply from fc00:11:11:0:8588:db7c:9ec4:3b48: time<1ms

Ping statistics for fc00:11:11:0:8588:db7c:9ec4:3b48:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum = 12ms, Average = 3ms
```

Frame 9. IPv6 ping from PC-1 in LAN 2 to PC-1 in LAN 1.

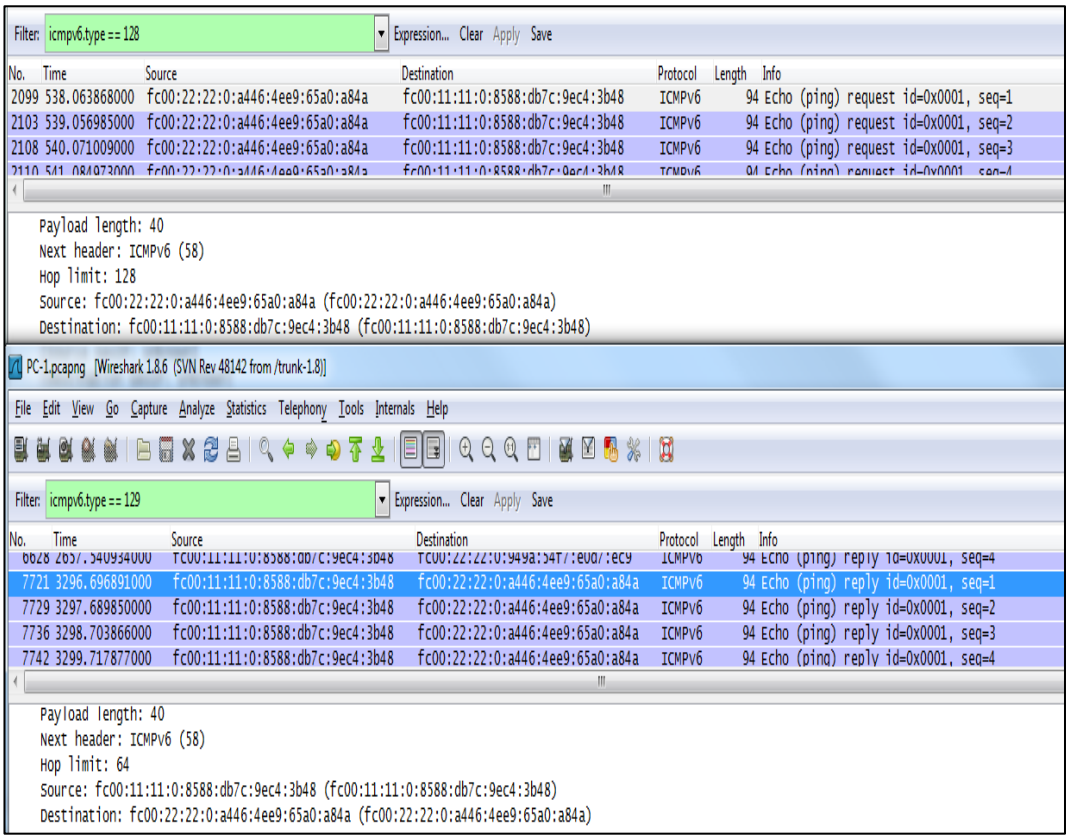

Figure 20. IPv6 ping exchange between LAN 2_PC-1 and LAN 1_PC-1.

In Figure 20 an IPv6 ping request and reply that was initiated by PC-1 in LAN 2 to PC-1 in LAN 1 is shown. From Frame 7, LAN1_PC-1 has a permanent and temporary IPv6 addresses of "fc00:22:22:0:a5:9db0:86e8:b788" and "fc00:22:22:0:a446:4ee9:65a0:a84a" respectively. Figure 20 shows that the

LAN2_PC-1 initiated ping request used the temporary address instead of the permanent address for the security reasons addressed earlier in section 3.5.1.

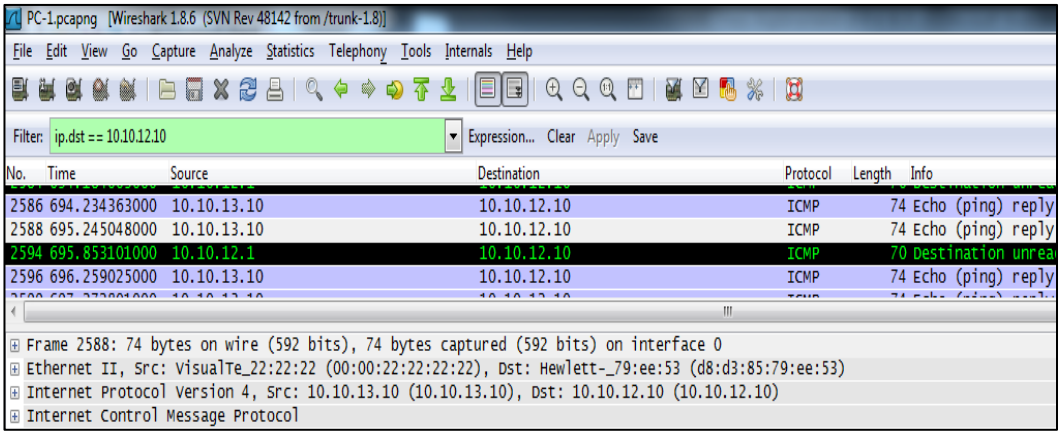

Figure 21. IPv4 ping reply from LAN 3 to LAN 2.

4.3 Stateful Autoconfiguration (DHCPv6)

The second scenario of the exercise used the Stateful Autoconfiguration technique to assign addresses to IPv6 hosts i.e. utilizing DHCPv6 server. The isc-dhcp-server, which supports DHCPv6, was used for dynamic address configurations. It comes with two start-up files i.e. */etc/init.d/isc-dhcp-server* for IPv4 and */etc/init.d/isc-dhcp6-server* for IPv6 but a single configurations file etc/dhcp/dhcpd.conf . The configurations file "*/etc/dhcp/dhcpd6.conf*" was created for IPv6 address pools and other configurations parameters. DHCPv6 configurations commands are explained in Frame 10. It is important to mention that the DHCP server node was manually configured with both IPv4 and IPv6 static addresses so as to be able to listen to DHCP clients request for each protocol. Frame 11 shows the interface configurations for the DHCP server node. Figure 22 shows the network diagram with DHCP obtained addresses as it was by the completion of the project.

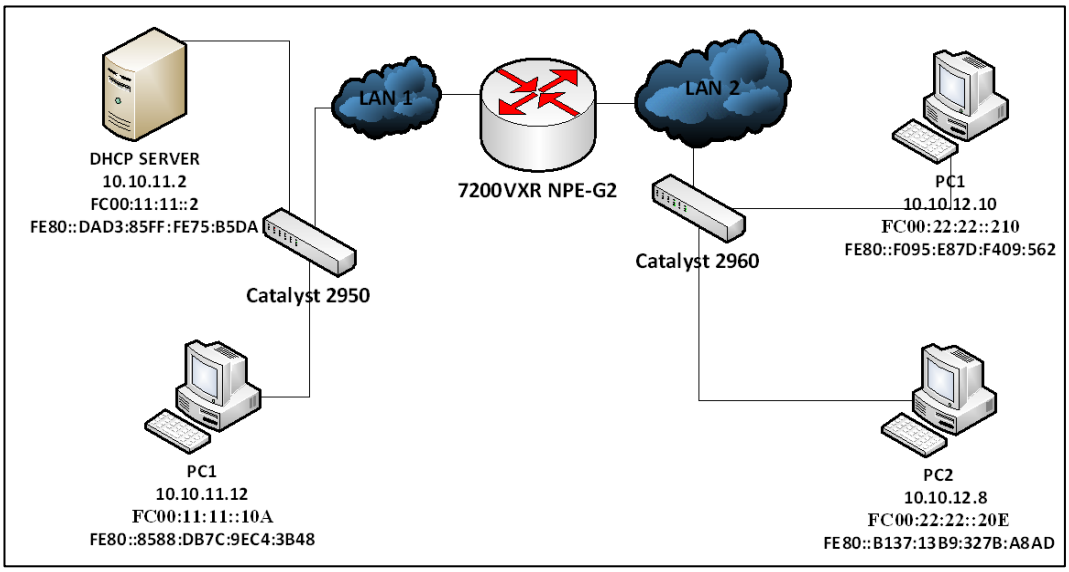

Figure 22. Stateful network diagram.

```
telecomm@tele:~$ cat /etc/dhcp/dhcpd6.conf
#optional parameter are declared here
option domain-name "testDomain.org";

#address lease lifetimes are declared below
default-lease-time 600;
max-lease-time 7200;

#enables the server to log DHCP transactions
log-facility local7;

#three subnets and their range of available addresses are declared as
#below
subnet6 fc00:11:11::/64 {
    range6 fc00:11:11::100 fc00:11:11::110;}

subnet6 fc00:22:22::/64 {
    range6 fc00:22:22::200 fc00:22:22::210;}

subnet6 fc00:33:33::/64 {
    range6 fc00:33:33::300 fc00:33:33::310;}
```

Frame 10. DHCPv6 configurations file.

```
telecomm@tele:~$ cat /etc/network/interfaces
auto lo
iface lo inet loopback

auto eth0
iface eth0 inet dhcp

auto eth1
iface eth1 inet static
        address 10.10.11.2
        netmask 255.255.255.0
        gateway 10.10.11.1
        broadcast 10.10.11.255
iface eth1 inet6 static
#The following line enables early loading of ipv6 configuration at
boot
        pre-up modprobe ipv6
        address fc00:11:11::2
        netmask 64
        gateway fe80::200:11ff:fe11:1111
```

Frame 11. Interface configurations for DHCP server node.

Router stateful autoconfiguration settings for GigabitEthernet 0/2 are shown in Frame 12. To enable links without a DHCPv6 server to obtain addresses from the server, the `ipv6 dhcp relay destination fc00:11:11::2` command was configured on the router interfaces of these links. The managed-configuration flag, `ipv6 nd managed-config-flag`, and other-configuration flags, `ipv6 nd other-config-flag`, were set to inform hosts to use DHCPv6 to obtain routable addresses and other configurations respectively. These flags are set on links in which nodes are desired to perform Stateful address autoconfiguration. These settings, however, do not stop the router from sending address prefixes in Router Advertisements. At this point the nodes obtain routable addresses from DHCPv6 and advertised prefixes i.e. both Stateful and stateless address autoconfiguration are performed. For situations where only Stateful address configuration is desired, the router can be stopped from sending address prefixes by using the `ipv6 nd prefix [`*`ipv6 prefix`*`] no-advertise` command. These were the conditions under which scenario two of the exercise was configured.

```
R7200(config)#int giga0/2
R7200(config-if)#ipv6 dhcp relay destination fc00:11:11::2
R7200(config-if)#ipv6 nd managed-config-flag
R7200(config-if)#ipv6 nd other-config-flag
R7200(config-if)#ipv6 nd prefix fc00:22:22::/64 no-advertise
R7200(config-if)#ipv6 nd ra lifetime 2400
R7200(config-if)#exit
```

Frame 12. Stateful address configurations on router interface GigabitEthernet 0/2.

Configurations similar to the above (Frame 12) were configured on GigabitEthernet 0/1 and 3 with the proper prefixes save for relay agent settings on GigabitEthernet 0/1. The `#ipv6 nd ra lifetime 2400` command deprecates the formerly received router advertisement lifetime from the default 1800 to 240 seconds. All interfaces configured as relay agents join the All DHCP Servers and Relay Agent multicast group, FF02::1:2. IPv6 configurations for GigabitEthernet 0/2 after enabling Stateful address auto-configurations are shown in Frame 13, highlighting the major changes.

```
R7200#sh ipv6 int giga0/2
GigabitEthernet0/2 is up, line protocol is up
  IPv6 is enabled, link-local address is FE80::200:22FF:FE22:2222
  No Virtual link-local address(es):
  Description: LAN2 Interface
  Global unicast address(es):
    FC00:22:22:0:200:22FF:FE22:2222, subnet is FC00:22:22::/64
[EUI]
  Joined group address(es):
    FF02::1
    FF02::2
    FF02::9
    FF02::1:2
    FF02::1:FF22:2222
  MTU is 1500 bytes
  ICMP error messages limited to one every 100 milliseconds
  ICMP redirects are enabled
  ICMP unreachables are sent
  ND DAD is enabled, number of DAD attempts: 1
  ND reachable time is 30000 milliseconds (using 30000)
  ND advertised reachable time is 0 (unspecified)
  ND advertised retransmit interval is 0 (unspecified)
  ND router advertisements are sent every 200 seconds
  ND router advertisements live for 240 seconds
  ND advertised default router preference is Medium
  Hosts use DHCP to obtain routable addresses.
  Hosts use DHCP to obtain other configuration.
```

Frame 13. IPv6 configurations for GigabitEthernet 0/2.

Figure 23 shows the received Router Advertisement with the above Stateful autoconfiguration settings. It can be noticed from the flags section that the managed, other configurations and the new router lifetime are set. Also noticeable is the missing prefix information at the bottom of the advertisement (see Figure 19). Stateful autoconfiguration settings do not affect the node's decision to use DHCP to obtain IPv4 addresses.

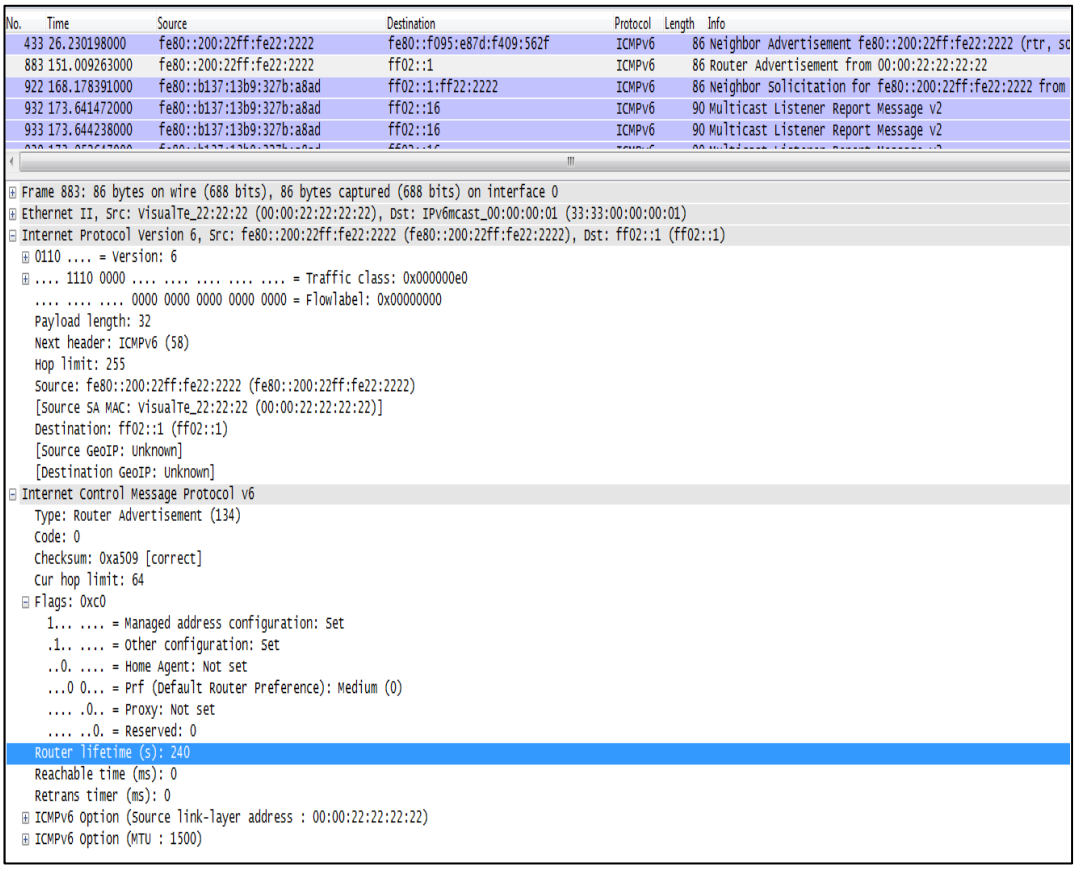

Figure 23. IPv6 Stateful Router Advertisement.

Figure 24 shows the IP configuration of PC-1 in LAN 1 after it cleared the addresses it derived from router advertised prefix. There is only a single unicast local address, FC00:11:11::10A received from DHCPv6 server, in this scenario unlike in stateless autoconfiguration. This increases administrative control over the nodes but trades the security/anonymity feature that comes with temporary addresses. The choice of what addressing approach to take comes down to the network administrator's decision and the company's policies.

```
Ethernet adapter Local Area Connection:

Connection-specific DNS Suffix  . : example.org
Description . . . . . . . . . . . : Intel(R) 82567LM-3 Gigabit Network Connec-
tion
Physical Address. . . . . . . . . : D8-D3-85-7D-C8-1A
DHCP Enabled. . . . . . . . . . . : Yes
Autoconfiguration Enabled . . . . : Yes
IPv6 Address. . . . . . . . . . . : fc00:11:11::10a(Preferred)
Lease Obtained. . . . . . . . . . : 19. huhtikuuta 2013 17:56:43
Lease Expires . . . . . . . . . . : 19. huhtikuuta 2013 18:13:17
Link-local IPv6 Address . . . . . : fe80::8588:db7c:9ec4:3b48%10(Preferred)
IPv4 Address. . . . . . . . . . . : 10.10.11.12(Preferred)
Subnet Mask . . . . . . . . . . . : 255.255.255.0
Lease Obtained. . . . . . . . . . : 19. huhtikuuta 2013 17:56:34
Lease Expires . . . . . . . . . . : 19. huhtikuuta 2013 18:11:34
Default Gateway . . . . . . . . . : 10.10.11.1
DHCP Server . . . . . . . . . . . : 10.10.11.2
DHCPv6 IAID . . . . . . . . . . . : 249090949
DHCPv6 Client DUID. . . . . . . . . . . . . . . . . . . . :
                                    00-01-00-01-18-3E-2B-67-D8-D3-85-7D-C8-1A
DNS Servers . . . . . . . . . . . : 193.166.140.100
                                    193.166.140.200
                                    8.8.8.8
```

Figure 24. Stateful IP configurations for PC-1 in LAN 1.

DHCPv6 client nodes use DHCP Unique Identifier (DUID) to identify a DHCPv6 server for message exchanges that require server identification while DHCPv6 servers use DUID to identify clients for the selection of configuration parameters and association of Identity-associations (IA). The clients and server use IA to manage, group and identify related IPv6 addresses. Figure 25 shows a DHCPv6 request from PC-2 in LAN 2 and some DHCPv6 variables and confirms the UDP ports used by the clients and servers, ports 546 and 547 respectively. The server and client DUIDs are opaque as expected, the requested address and its lifetime options can be seen at the bottom of the figure. All client initiated communications to the DHCPv6 server were sent to FF02::1:2 as expected. The servers use the clients' unicast addresses to reply to on-link client messages and FF02::1:2 for requests received from relay-forward addresses. /7, 19/

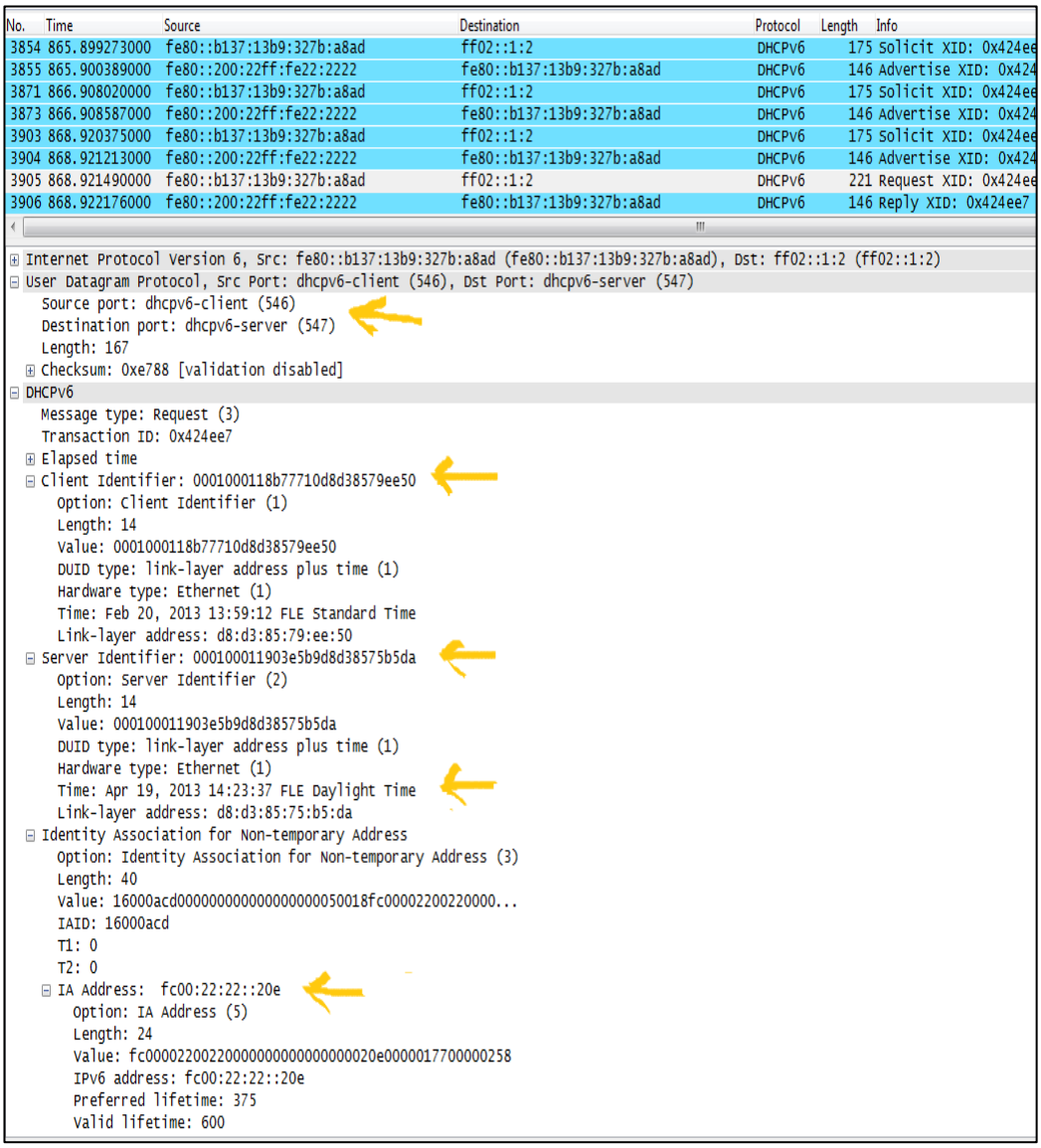

Figure 25. DHCPv6 address request in LAN 2.

An IPv6 ping from LAN 2 gateway to PC-1 in LAN 1 is shown in figure Figure 26. Frame 14 is an extract from DHCPv6 lease file showing active binding lease information LAN 1_PC-1.

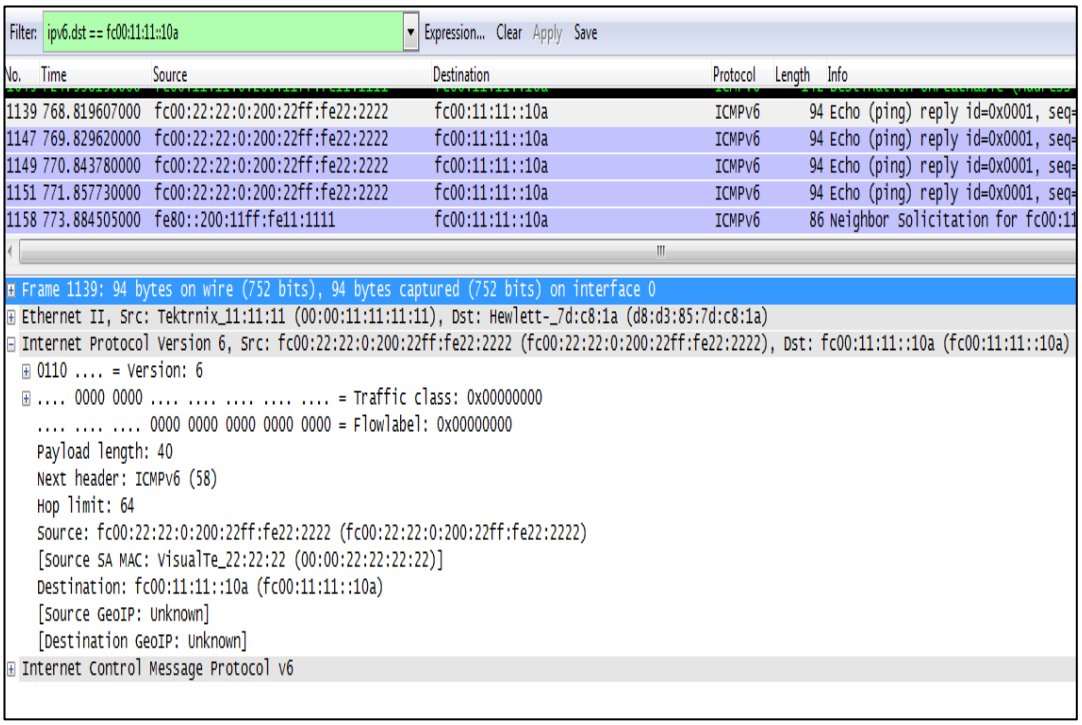

Figure 26. IPv6 ping reply from LAN 2 gateway to LAN 1 PC-1.

```
ia-na
"o\254\270\016\000\001\000\001\030\022\360\034\270\254o\016\034D" {
  cltt 5 2013/05/10 07:40:58;
  iaaddr fc00:11:11::10a {
    binding state active;
    preferred-life 375;
    max-life 600;
    ends 5 2013/05/10 07:50:58;
  }
}
```

Frame 14. DHCPv6 lease file extract showing LAN 1_PC-1 lease information.

5 OUTCOME OF THE PROJECT

The aim of this chapter is to give a clear outcome of the project by highlighting the results that are documented in the previous chapter. The section follows the same outline as was used in Chapter 4, i.e. it will address the results of scenario one followed by those of scenario two.

5.1 Stateless Address Autoconfiguration

Under this scenario, the IP host nodes were configured and obtained their IPv4 addresses from a DHCP server and their IPv6 addresses from router advertised prefixes. A "*/etc/dhcp/dhcpd.conf*" file was created containing three subnet declarations and their necessary information from each LAN (see Appendix B). The DHCP server was located in LAN 1 at address 10.10.11.2 or FC00:11:11::2. DHCP relay agents were configured on router interfaces in LANs that did not have a DHCP server, LAN 2 and LAN 3.

The router was manually configured with the global `IPv6 unicast routing` command to enable IPv6 processing. The router interfaces were also configured with IPv4 addresses, custom MAC addresses and IPv6 prefixes. These commands allowed the router interfaces to send out periodic advertisements to their links which were used by the host nodes for stateless autoconfiguration. Frame 15 shows an extract of IPv6 interface configuration for router interface GigabitEthernet 0/2, highlighting the stateless autoconfiguration declaration.

```
R7200#sh ipv6 int giga0/2
GigabitEthernet0/2 is up, line protocol is up
  IPv6 is enabled, link-local address is FE80::200:22FF:FE22:2222
!  <output ommited>
  Hosts use stateless autoconfig for addresses.
```

Frame 15. Extract of interface GigabitEthernet 0/2 IPv6 configurations.

With the above settings, the IP nodes in all LANs were able to auto configure their IPv6 address from the received router advertisements and IPv4 from DHCP. With IPv6 addressing, the interface can be assigned with more than one address

unlike IPv4 addressing, i.e. the IP nodes obtained a permanent, temporary and link-local IPv6 address compared to one IPv4 address.

A successful set of ping tests were performed to test the connectivity between the LANs, the results of which are displayed in Frame 9 and figures (Figure **20** and Figure **21**). With the successful results of ping tests for both IPv4 and IPv6 protocol, it was guaranteed that the objectives of setting up a SLAAC Dual Stack network were achieved.

5.2 Stateful Autoconfiguration

In the stateful scenario, the IP host nodes obtained both IPv6 and IPv4 addresses from DHCP and DHCPv6 servers. A configuration file "*/etc/dhcp/dhcpd6.conf*" (see Frame 10), in which IPv6 subnets and their address ranges and other lease parameter and optional information were declared, was created.

The router interface configurations were modified to allow for stateful autoconfiguration and stop stateless autoconfiguration by not advertising their address prefixes (see Frame 12). This was done so as to achieve a complete stateful Dual Stack environment. The host node IP configurations were analysed and verified as to have originated from the DHCP servers. DHCP lease files were used to reinforce this analysis.

Considering that the DHCP server was situated in LAN 1 and was able to lease IPv4 and IPv6 addresses to nodes in LAN 2 and LAN 3, the Stateful Dual Stack network was considered operational. The successful ping tests were a further proof of the operational state of the network.

6 CONCLUSIONS

At the beginning of the exercise I set out to research, document and implement IPv6. Documented in this paper are the needs for IPv6, features of IPv6 and IPv6 address types, IPv6 implementation mechanisms, how IPv6 addresses and interface identifiers are generated and assigned and the Neighbour Discovery and DHCPv6 services. The Dual Stack implementation of IPv6 and IPv4 on the same network was particularly covered and implemented for SLAAC and DHCPv6, covering mainly configurations on routers and DHCP server, on a multi-LAN environment. There was little to no challenge in getting the nodes to use IPv6 addressed since the protocol has been supported by major OSs manufactures since the early 2000s. There were however a few issues on a node over which I did not have administrative rights. Though it was not possible to obtain a global prefix, the multi-LAN environment provided enough room to test varied network scenarios.

There are issues that have been brought up, by people familiar to the technology, having to do with running two DHCP servers. As an expansion to the project, the DHCPv6 server can be modified to offer IPv4 information too. The current implementation of DHCPv6 also poses a security flaw by tending strongly to bind nodes and their addresses. A modification can be made that allows the nodes to acquire new addresses overtime. Though it is not possible at the moment to connect to the public IPv6 internet, the LAN can be expanded to cover more nodes such as networked printers etc.

Through this project I have learned the pains of hard work and perseverance and the joys of accomplish a task. Mine is the confidence that, given enough time and resources, I can accomplish even a greater tasks.

REFERENCES

/1/ Cisco. 2013. Cisco IOS IPv6 Feature Mapping. Accessed 14.04.2013. http://docwiki.cisco.com/wiki/Cisco_IOS_IPv6_Feature_Mapping#IPv6_Data_Link-Layer_Features

/2/ Cisco. Cisco Feature Navigator. Accessed 14.04.2013. http://tools.cisco.com/ITDIT/CFN/jsp/index.jsp

/3/ Cisco. Configuring SMD Templates. Accessed 20.04.2013 http://www.cisco.com/en/US/docs/switches/lan/catalyst2960/software/release/15.0_1_se/configuration/guide/swsdm.html

/4/ Conta, A., Deering, S. & Gupta, M. Ed.2006, Internet Control Message Protocol (ICMPv6) for the Internet Protocol Version 6 (IPv6) Specification. Internet Engineering Task Force (IETF). RFC 4443. Accessed 2.10.2012. http://tools.ietf.org/html/rfc4443

/5/ Cotton, M. & Vegoda, L. 2010. Special Use IPv4 Addresses. Internet Engineering Task Force (IETF). RFC 5735. Accessed 10.9.2012. http://tools.ietf.org/html/rfc5735

/6/ Deering, S. & Hinden, R. 1998. Internet Protocol Version 6 (IPv6) Specification. Internet Engineering Task Force (IETF). RFC 2460. Accessed 10.9.2012. http://tools.ietf.org/html/rfc2460

/7/ Droms Ed, R., Bound, J., Volz, B., Lemon, T. & Carney, M. 2003. Dynamic Host Configuration Protocol for IPv6 (DHCPv6). Internet Engineering Task Force (IETF). RFC3315. Accessed 20.9.2012. http://www.ietf.org/rfc/rfc3315.txt

/8/ Forouzan, B. A. 2010. TCP/IP Protocol Suite. 4th Ed. India. McGraw-Hill Education Pvt Limited.

/9/ Geoff Huston. IPv4 Address Report. IANA Unallocated Address Pool Exhaustion. Accessed 10.9.2012. http://www.potaroo.net/tools/ipv4/index.html

/10/ Hinden, R. & Deering, S. 2006. IP Version 6 Addressing Architecture. Internet Engineering Task Force (IETF). RFC 4291. Accessed 05.05.2013 https://tools.ietf.org/html/rfc4291

/11/ Kaushik Das. Top 10 features that make IP6 greater that Pv4. Accessed 20.9.2012. http://www.ipv6.com/articles/general/Top-10-Features-that-make-IPv6-greater-than-IPv4.htm

/12/ Microsoft TechNet. 2005. Ipv6 Address Autoconfiguration. Windows Server. Accessed 15.11.2012. http://technet.microsoft.com/en-us/library/cc778502%28v=ws.10%29.aspx

/13/ Microsoft TechNet. 2005. Ipv6 Interface Identifiers. Windpws Server. Accessed 05.05.2013 . http://technet.microsoft.com/en-us/library/cc736439(v=ws.10).aspx

/14/ Microsoft TechNet. 2012. DNS Overview. Accessed 21.11.2012. http://technet.microsoft.com/library/hh831667

/15/ Microsoft TechNet. How DNS query works Accessed 21.11.2012. http://technet.microsoft.com/en-us/library/cc775637%28v=ws.10%29.aspx

/16/ Narten, T., Draves, R. and Krishnan, S. 2007. Privacy Extension for Stateless Address Autoconfiguration in IPv6. Internet Engineering Task Force (IETF). RFC 4941. Accessed 05.05.2013. http://tools.ietf.org/html/rfc4941

/17/ Narten, T., Nordmark, E., Simpson, W. and Soliman, H. 2007. Neighbour Discovery for IP version 6 (IPv6). Internet Engineering Task Force (IETF). RFC 4861. Accessed 26.11.2012. http://tools.ietf.org/html/rfc4861

/18/ Perkins, C. Ed., Johnson, D. & Arkko, J. 2011. Mobility Support in IPv6. Internet Engineering Task Force (IETF). RFC 6275. Accessed 09.05.2013. http://tools.ietf.org/html/rfc6275#section-7.2

/19/ RIPE NCC IPv4 Available Pool-Graph. Accessed 10.9.2012. http://www.ripe.net/internet-coordination/ipv4-exhaustion/ipv4-available-pool-graph

/20/ Scott Hogg. 2007. IPv6: Dual stack where you can; tunnel where you must. Network World. Accessed 13.9.2012. http://www.networkworld.com/news/tech/2007/090507-tech-uodate.html

/21/ Thomson, S., Narten, T. & Jinmei, T. 2007. IPv6 Stateless Address Autoconfiguration. Internet Engineering Task Force (IETF). RFC 4862. Accessed 20.3.2013. http://tools.ietf.org/pdf/rfc4862.pdf

/22/ Wilson, T. 2010. How to Implement IPv6 and Configure Cisco Router to use IPv6. Trainsignal. Accessed 02.12.2012. http://www.trainsignal.com/blog/ipv6-implementation

APPENDICES

Appendix A

Router Running Configurations

```
R7200#sh run
Building configuration...

Current configuration : 1680 bytes
!
! Last configuration change at 06:22:26 UTC Mon Jan 23
2006
!
upgrade fpd auto
version 15.0
service timestamps debug uptime
service timestamps log datetime msec
no service password-encryption
!
hostname R7200
!
boot-start-marker
warm-reboot
boot-end-marker
!
no aaa new-model
!
ipv6 unicast-routing
ipv6 cef
ip source-route
ip cef
!
no ip dhcp conflict logging
!
!multilink bundle-name authenticated
!
redundancy
!
interface Loopback0
 no ip address
!
interface GigabitEthernet0/1
 description LAN1 Interface
 mac-address 0000.1111.1111
 ip address 10.10.11.1 255.255.255.0
 duplex auto
 speed auto
 media-type rj45
 negotiation auto
 ipv6 address FC00:11:11::/64 eui-64
 ipv6 enable
 ipv6 rip IPV6RIP enable
 !
<Output omitted>
```

Router Running Configurations (continued)

```
interface FastEthernet0/2
 no ip address
 shutdown
 duplex auto
 speed auto
 !
!
interface GigabitEthernet0/2
 description LAN2 Interface
 mac-address 0000.2222.2222
 ip address 10.10.12.1 255.255.255.0
 ip helper-address 10.10.11.2
 duplex auto
 speed auto
 media-type rj45
 negotiation auto
 ipv6 address FC00:22:22::/64 eui-64
 ipv6 enable
 ipv6 rip IPV6RIP enable
 !
!
interface GigabitEthernet0/3
 description LAN2 Interface
 mac-address 0000.3333.3333
 ip address 10.10.13.1 255.255.255.0
 ip helper-address 10.10.11.2
 duplex auto
 speed auto
 media-type rj45
 negotiation auto
 ipv6 address FC00:33:33::/64 eui-64
 ipv6 enable
 ipv6 rip IPV6RIP enable
 !
!
router rip
 version 2
 network 10.0.0.0
!
ip forward-protocol nd
no ip http server
no ip http secure-server
!
!
ipv6 router rip IPV6RIP
!
!
control-plane
 !
!
line con 0
 stopbits 1
line aux 0
 stopbits 1
line vty 0 4
 login
!
end
```

Appendix B

IPv4 DHCP Configurations File (/etc/dhcp/dhcpd.conf)

```
#option definitions common to all supported networks...
option domain-name "example.org";
option domain-name-servers 193.166.140.100, 193.166.140.200, 8.8.8.8;

default-lease-time 600;
max-lease-time 7200;

# If this DHCP server is the official DHCP server for the local
# network, the authoritative directive should be uncommented.
authoritative;

# Use this to send dhcp log messages to a different log file (you also
# have to hack syslog.conf to complete the redirection).
log-facility local7;

subnet 10.10.11.0 netmask 255.255.255.0 {
       range 10.10.11.10 10.10.11.100;
       option routers 10.10.11.1;
       option subnet-mask 255.255.255.0;
       option broadcast-address 10.10.11.255;
}

subnet 10.10.12.0 netmask 255.255.255.0 {
        range 10.10.12.10 10.10.12.100;
        option routers 10.10.12.1;
        option subnet-mask 255.255.255.0;
        option broadcast-address 10.10.12.255;
}

subnet 10.10.13.0 netmask 255.255.255.0 {
        range 10.10.13.10 10.10.13.100;
        option routers 10.10.13.1;
        option subnet-mask 255.255.255.0;
        option broadcast-address 10.10.13.255;
}
```

Appendix C

IPv4 DHCP Lease File (/var/lib/dhcp/dhcpd.leases)

```
# The format of this file is documented in the
dhcpd.leases(5) manual page.
# This lease file was written by isc-dhcp-4.1-ESV-R4

lease 10.10.13.10 {
  starts 5 2013/04/19 15:29:50;
  ends 5 2013/04/19 15:39:50;
  tstp 5 2013/04/19 15:39:50;
  cltt 5 2013/04/19 15:29:50;
  binding state free;
  hardware ethernet 68:b5:99:eb:1c:66;
  uid "\001h\265\231\353\034f";
}
lease 10.10.12.10 {
  starts 4 2013/04/18 16:36:27;
  ends 4 2013/04/18 16:38:27;
  tstp 4 2013/04/18 16:38:27;
  cltt 4 2013/04/18 16:36:27;
  binding state free;
  hardware ethernet d8:d3:85:79:ee:53;
  uid "\001\330\323\205y\356S";
}
lease 10.10.12.11 {
  starts 5 2013/04/19 15:21:28;
  ends 5 2013/04/19 15:31:28;
  tstp 5 2013/04/19 15:31:28;
  cltt 5 2013/04/19 15:21:28;
  binding state free;
  hardware ethernet 00:0a:cd:0a:78:58;
  uid "\001\000\012\315\012xX";
}
lease 10.10.11.11 {
  starts 4 2013/04/18 12:52:11;
  ends 4 2013/04/18 13:02:11;
  tstp 4 2013/04/18 13:02:11;
  cltt 4 2013/04/18 12:52:11;
  binding state free;
  hardware ethernet d8:d3:85:7d:c8:1a;
  uid "\001\330\323\205}\310\032";
}
```

Appendix D

IPv6 DHCP Lease File (/var/lib/dhcp/dhcpd6.leases)

```
# The format of this file is documented in the dhcpd.leases(5) manu-
al page.
# This lease file was written by isc-dhcp-4.1-ESV-R4

server-duid "\000\001\000\001\031\003\345\271\330\323\205u\265\332";


ia-na
"\315\012\000\026\000\001\000\001\030\267w\020\330\323\205y\356P" {
  cltt 5 2013/04/19 15:24:16;
  iaaddr fc00:22:22::20e {
    binding state expired;
    preferred-life 375;
    max-life 600;
    ends 5 2013/04/19 15:34:16;
  }
}

ia-na "r\023\000\016\000\001\000\001\030\333P{\000\023r\030\310K" {
  cltt 5 2013/04/19 08:34:41;
  iaaddr fc00:11:11::10e {
    binding state expired;
    preferred-life 375;
    max-life 600;
    ends 5 2013/04/19 08:44:41;
  }
}

ia-na
"\272\231\264\016\000\001\000\001\025\224\274E\264\231\272\345/\277"
{
  cltt 5 2013/04/19 08:34:05;
  iaaddr fc00:11:11::10b {
    binding state expired;
    preferred-life 375;
    max-life 600;
    ends 5 2013/04/19 08:44:05;
  }
}

ia-na "\002\000\000\000\000\002\000\000\000\013x\254\300\217lD" {
  cltt 5 2013/04/19 15:33:32;
  iaaddr fc00:11:11::10d {
    binding state active;
    preferred-life 375;
    max-life 600;
    ends 5 2013/04/19 15:43:32;
  }
}

ia-na
"F\014\000\026\000\001\000\001\023\376\300?\330\323\205y\354\253" {
  cltt 5 2013/04/19 15:30:14;
  iaaddr fc00:22:22::210 {
    binding state active;
    preferred-life 375;
    max-life 600;
    ends 5 2013/04/19 15:40:14;
```

```
ia-na
"F\014\000\026\000\001\000\001\023\376\300?\330\323\205y\354\253" {
  cltt 5 2013/04/19 15:30:14;
  iaaddr fc00:22:22::210 {
    binding state active;
    preferred-life 375;
    max-life 600;
    ends 5 2013/04/19 15:40:14;
  }
}

ia-na
"\205\323\330\016\000\001\000\001\023\376\300?\330\323\205y\354\253"
{
  cltt 5 2013/04/19 08:33:09;
  iaaddr fc00:11:11::100 {
    binding state expired;
    preferred-life 375;
    max-life 600;
    ends 5 2013/04/19 08:43:09;
  }
}

ia-na "\205\323\330\016\000\001\000\001\030>*M\330\323\205u\265\256"
{
  cltt 5 2013/04/19 15:33:26;
  iaaddr fc00:11:11::109 {
    binding state active;
    preferred-life 375;
    max-life 600;
    ends 5 2013/04/19 15:43:26;
  }
}

ia-na "_.\010\016\000\001\000\001\027\303\267\236\010._%4\255" {
  cltt 5 2013/04/19 08:32:55;
  iaaddr fc00:11:11::10d {
    binding state expired;
    preferred-life 375;
    max-life 600;
    ends 5 2013/04/19 08:42:55;
  }
}

ia-na "\205\323\330\016\000\001\000\001\030>+g\330\323\205}\310\032"
{
  cltt 5 2013/04/19 15:30:15;
  iaaddr fc00:11:11::10a {
    binding state active;
    preferred-life 375;
    max-life 600;
    ends 5 2013/04/19 15:40:15;
  }
}

ia-na "\244\267!\000\000\001\000\001\000\000\000\"hu Feb" {
  cltt 5 2013/04/19 15:33:25;
  iaaddr fc00:11:11::106 {
    binding state active;
    preferred-life 375;
    max-life 600;
    ends 5 2013/04/19 15:43:25;
  }
}
```

```
ia-na "p!\000\017\000\001\000\001\0306g\374\000!p1\217\303" {
  cltt 5 2013/04/19 15:33:09;
  iaaddr fc00:11:11::110 {
    binding state active;
    preferred-life 375;
    max-life 600;
    ends 5 2013/04/19 15:43:09;
  }
}

ia-na
"\026\037\000\016\000\001\000\001\0261\252k\000\037\026\247\362\327"
{
  cltt 5 2013/04/19 08:34:29;
  iaaddr fc00:11:11::10f {
    binding state expired;
    preferred-life 375;
    max-life 600;
    ends 5 2013/04/19 08:44:29;
  }
}

ia-na "\031\"\000\020\000\001\000\001\027\303\267\236\010._%4\255" {
  cltt 5 2013/04/19 08:32:51;
  iaaddr fc00:11:11::103 {
    binding state expired;
    preferred-life 375;
    max-life 600;
    ends 5 2013/04/19 08:42:51;
  }
}

ia-na "\002\000\000\000\000\002\000\000\000\013<J\222\265E\036" {
  cltt 5 2013/04/19 15:34:18;
  iaaddr fc00:11:11::108 {
    binding state active;
    preferred-life 375;
    max-life 600;
    ends 5 2013/04/19 15:44:18;
  }
}

ia-na "r\023\000\016\000\001\000\001\030\2665\200\000\023r\032<\246"
{
  cltt 5 2013/04/19 08:33:00;
  iaaddr fc00:11:11::102 {
    binding state expired;
    preferred-life 375;
    max-life 600;
    ends 5 2013/04/19 08:43:00;
  }
}

ia-na "\205\323\330\016\000\001\000\001\030>+g\330\323\205}\310\032"
{
  cltt 5 2013/04/19 15:30:15;
  iaaddr fc00:11:11::10a {
    binding state expired;
    preferred-life 375;
    max-life 600;
    ends 5 2013/04/19 15:40:15;
  }
}
```