Design and Simulation of an IPv6 Network Using Two Transition Mechanisms

Lefty Valle-Rosado[1], Lizzie Narváez-Díaz[1], Cinhtia González-Segura[1], Victor Chi-Pech[1]

[1] Universidad Autónoma de Yucatán, Facultad de Matemáticas, Unidad Multidisciplinaria Tizimín
Tizimín, Yucatán 97700, México

Abstract
This research is focused on the most important theoretical concepts of the IPv6 protocol, such as addressing, address allocation, routing with the RIPng protocol and two IPv4 to IPv6 transition mechanisms. It describes the design and simulation of connectivity between devices of a network configured with IPv6 protocol, employing dual stack and tunneling as transition mechanisms, all that through the use of two network simulators, Packet Tracer and GNS3.

Keywords: *IPv6-IPv4, double stack, tunneling, RIP-RIPng, simulators.*

1. Introduction

The Internet is becoming omnipresent and has been exponentially growing in size. The number of users, sub-networks and domains connected to the Internet seem to be exploding [1]. Nowadays exists many devices that connect through the Internet using the IPv4 protocol; the amount of this devices has exponentially increased in the last years; since not only personal computers and laptops connect to the network, but also devices like cellular phones, automobiles with GPS, PDAs, video game consoles, domestic appliances, measuring devices, among others [2].

For the above is that the IPv4 protocol is in trouble, since it has no more the capability to support so many devices, among other problems inherent to the protocol; to overcome this situation methods were developed to extend the life of IPv4, such as: Network Address Translation (NAT), Classless Inter-Domain Routing (CIDR), Variable Length Subnet Mask (VLSM), private network addressing, among others [3]; however this has not been enough, network technology has matured and, new applications, protocols and devices had emerged, making IPv4 unable to support anymore the technological trend.

On February 3, 2011 the Latin American and Caribbean Internet Addresses Registry (LACNIC) issued a statement saying that the global IPv4 central address pool managed by the Internet Assigned Numbers Authority (IANA) was finally exhausted, according to the global policies agreed by the Internet communities of all regions. On that date the last available IPv4 address blocks were allocated, corresponding one for each of the five Regional Internet Registries (RIR) in the whole world [4].

June 6, 2012 was the selected date by the Internet Society (ISOC) and other organizations in the field as the worldwide launch of IPv6. On that date numerous companies and organizations from around the world enabled the operations of their portals and other forms of presence on the Internet with the IPv6 protocol in a definitive way [5].

In Mexico, the Autonomous National University of Mexico (UNAM) started investigations in the subject since December 1998, date in which its IPv6 Project constitutes. The early tests emphasizes connecting to the 6Bone, which was an experimental global network used to test the concepts and operation of IPv6 implementation. At the end into 6Bone 47 countries participated around the world, including Mexico, where UNAM was the first node in the country, as it was recorded in June 1999 [6].

2. IPv6

The IPv6 protocol is an upgrade of the IPv4 protocol, belonging to the TCP/IP (Transmission Control Protocol / Internet Protocol) suite's protocol stack, used to identify, by means of an IP address, each computer interface or device that connects to Internet or to an Intranet [7]. Migrating from IPv4 to IPv6 in an instant is impossible because of the huge size of the Internet and of the great number of IPv4 users. [8]

IPv6 is basic to the operation of the network and the first specifications of this protocol were developed by Internet Engineering Task Force (IETF) at the 90's. An important factor for the adoption of the new protocol is the expansion in use of new technologies based on the concept 'always on', such as xDSL, cable, Ethernet, optical fiber, and Power Line Communication; however, but the main motivation for the transition to the new protocol is the expansion of available public addresses for Internet, which will allow the connection to the network for multiple devices such as PDAs and mobile phones, among others

[8]. The size of an IPv6 address is 128 bits, 4 times bigger than an IPv4 address; an 32 bits address space allows up to 4.294.967.296 combinations, while the 128 bits of an IPv6 address allows up to 340.282.266.920.938.463.463.374.607.431.768.211.465 (or 3,4 x 1038), therefore it is obvious the increase in available addresses [10].

At the end of the seventies, when the IPv4 address space was designed, was unimaginable that it could be exhausted; however due to technological changes and assignation politics that did not foreseen the recent increase in the Internet hosts quantity, the IPv4 address space was depleted to such an extent that in 1992 was made evident the need for a replacement [10].

2.1 IPv6 Address Format

As mentioned, an IPv6 address has 128 bits or 16 bytes, this address is divided into eight hexadecimal blocks of 16 bits separated by colons ":"; for example: FE80:0000:0000:0000:0202:B3FF:FE1E:8329 [11].

In an IPv6 address, the zeros on the left on the block containing them can be omitted, and also contiguous blocks of zeros can be simplified using double colon "::". On the basis of the above, starting from the previous address can be obtained the following address, FE80::202:B3FF:FE1E:8329.

The network prefix in an IPv6 address is represented in the same way that IPv4, for example, take the IPv4 address 192.168.1.0/27, this means that the first 27 bits are network's and the remaining 5 are which identify a device, thus in IPv6 the following address ffe:b00:c18:1::1/64 indicates that the first 64 bits identifies the network (3ffe:b00:c18:1) and the remaining 64 bits identifies the device in that network (::1) [12].

2.2 Assignment of Addresses

IPv6 addresses can be statically assigned using an identifier (ID) of manual interface or an ID of EUI-64 interface, it also can be dynamically configured by using stateless address autoconfiguration or by DHCPv6.

Static configuration: Consists on manually enter the IPv6 address of a node in a configuration file or through the use of proper tools of the operative system. Information to be included is the IPv6 address and the network prefix size [12]. This configuration is divided into static configuration using the ID of manual interface, in which the entire IPv6 address is used, both the network section and the device identifier section [11]; and into static configuration using the ID of EUI-64 interface, in which in order to obtain the ID, the host takes the MAC address from the link layer device, however as the MAC address only has 48 bits, then the MAC address is split in half and in the middle is inserted the default hexadecimal value FFFE of 16 bits in order to complete an unique interface ID of 64 bits [11].

Dynamic configuration: Through this method the host automatically learns the necessary parameters to obtain an IP address that will be used in the communication process with end devices. It is divided into stateless autoconfiguration, in which each router broadcasts information of the network including the prefix assigned to each of its interfaces. With the obtained information in this broadcasting, the end systems create a unique address concatenating the prefix with the ID in EUI-64 interface format. The "stateless" name comes from that no device keeps track of the assigned IP addresses [14]. The other method is with DHCPv6, its operation is similar to the traditional DHCP, hosts obtain its interface address, information and configuration parameters from a server [15].

2.3 Advantages of using IPv6

Among the main advantages of using the IPv6 protocol are: improved IP address (a bigger address space offer several enhancements), the simplified header (better efficiency on routing to performance scalability and forwarding speed), enhanced mobility and security (ensure that abides standards functionality of mobile IP and IP security) [11].

3. Routing

Routing on IPv6 can be done through the use of static routes and dynamic routing protocols. Static routes are manually defined by the administrator so the router learns about a remote network, are usually used when the routing is from a network to a single connection network; a single connection network is a network accessible by only one route [15].

In regards to dynamic routing protocols, IPv6 uses updated versions of the same routing protocols available for IPv4; among the most important ones are: RIPng, EIGRP for IPv6, IS-IS for IPv6, MP-BGP4 (MultiProtocol BGP) and OSPFv3 [14].

This research will deepen a little more on RIPng since it is the routing protocol used to perform the connectivity tests with IPv6, RIPng (Routing Information Protocol next-generation) is the new generation of RIP for IPv6 and is based on RIPv2, is a distance-vector routing protocol that uses hop count as a routing metric, with 15 as maximum, its multicast updates are issued every 30 seconds, it uses split horizon and poison reverse updates to prevent routing loops [11, 14].

RIPng uses IPv6 to transport, includes the IPv6 prefix and the next-hop IPv6 address, uses the multicast group FF02::9 as destination address for RIP updates and sends updates for the UDP port 521 [11].

The main steps that must be followed to configure RIPng in a Cisco router (these devices were used because were available): Activate IPv6 routing using the global command "ipv6 unicast-routing", activate the routing protocol using the global configuration command "ipv6 router rip NAME", configure an IPv6 unicast address in each interface, using the interface command "ipv6 address ADDRESS/PREFIX_LENGHT [EUI-64]" and activate the routing in the interface, by example through the interface subcommand "ipv6 rip NOMBRE enable" (where the NAME coincides whit the provided in the global configuration command "ipv6 router rip NAME") [16].

4. Transition Mechanisms

As seen in recent years transition from IPv4 to IPv6 has not been an immediate process but they had to coexist together for several years, thus mechanisms has been developed that have allowed the coexistence and migration from one protocol to another; there are several mechanisms but in this work there are only mentioned two, since those are the used to perform the tests.

Dual stack: This mechanism implements both protocols on each node in the network; IPv4 and IPv6, each node with dual stack in the network will have two addresses, one for IPv4 and other for IPv6. This procedure is easy to implement and is widely supported; however it has the disadvantage that the network topology requires two tables and two routing processes [17].

Tunneling: On using tunneling, the routers that execute IPv4 and IPv6 at the same time encapsulate IPv6 traffic inside IPv4 packets. The origin of the IPv4 is the own local router, and the destination will be the router at the end of the tunnel. When the destination router receives the IPv4 packet, decapsulates it and send forwards the IPv6 traffic that was encapsulated. This tunneling system is effective although increases the MTU since it consumes 20 bytes with each IPv4 header on the intermediate links and is also difficult the resolution and tracking of problems [14].

5. Statement of the Problem

As shown in recent information about the IPv4 protocol, this is exhausted and is already necessary to start with the transition to the new IPv6 protocol, is for this reason that it was decided to carry out this investigation to learn more thoroughly the basic concepts of this protocol, as it increasingly has more relevance in the field of networks.

The IPv4 protocol is currently used in the Tizimin Multidisciplinary Unit (UMT) of the Autonomous University of Yucatan (UADY), and one of the main reasons of this research is to provide knowledge and have people prepared in the area especially in the IPv6 protocol for its eventual implementation. The UMT computer center facilities are in full development and acquiring new computer equipment, many students already has a computer and portable devices, resulting in the need for more IP addresses, together with the above is the need to stay ahead, thus at some point in time will be necessary to make the transition to the IPv6 protocol in the UMT network.

In this work a test network is implemented using two simulators, Packet Tracer and Graphical Network Simulator (GNS3), the network will support IPv6 as addressing protocol, the routing protocol RIPng was implemented and finally the network was tested with two transition mechanisms, dual-stack and tunneling.

6. Development of the Problem

As mentioned in section 5, two simulators were used for the development of research, Packet Tracer (version 5.3.1) and GNS3 (0.8.2). Packet tracer is a network simulator of Cisco that allows users to create network topologies, configure devices, insert packages and simulate a network with multiple visual representations. GNS3 is a free software that allows users to perform the same jobs than Packet Tracer, the only difference is that this software allows to emulate the operative systems of routers and PCs. Below are described the details of the network configuration with Packet Tracer in which the dual stack transition method was implemented; were used three PCs, three switches 2960 and three routers 2811. Figure 1 shows the topology and the IPv4 and IPv6 addresses that were used in the test.

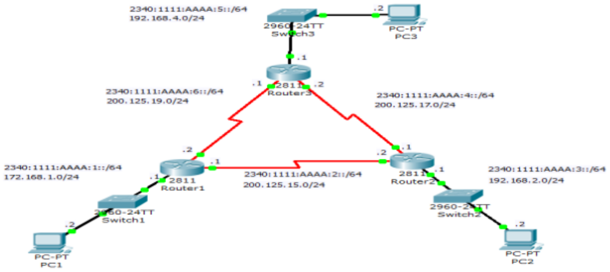
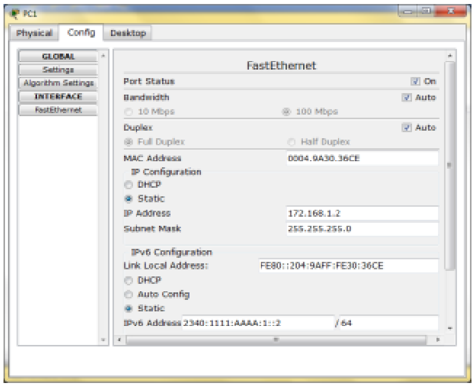


Fig. 1. Network Topology in the Packet Tracer simulator.

The process to configure PC1 is as follows: PC1 is selected and unfolds a visual interface that has three tabs, in tab Physical are the physical components that could be adapted to the host, whether cameras, USB interfaces, earphones and wireless cards, among others. The tab Desktop is the place where different utilities are located, such as a small search engine, a terminal for commands like the CMD of Windows or the Terminal of GNU/Linux, and a small text editor, among others. Finally the tab Config is where IP addresses are configured, both IPv4 and IPv6, in this case the host name is PC1 and the Gateway addresses was defined statically, and were for IPv4 172.168.1.1 and for IPv6 2340:1111:AAAA:1::1. In this same visual interface was selected the tab FastEthernet to statically configure the IPv4 and IPv6 addresses used to the hosts communications; the IPv4 address used was 172.168.1.2/24 and the IPv6 address was 2340:1111:AAAA:1::2 with a prefix /64, this process similarly repeats in PC2 and PC3 of the Figure 1 topology. See Figure 2.



FastEthernet

Port Status ☑ On

Bandwidth ☑ Auto

○ 10 Mbps ● 100 Mbps

Duplex ☑ Auto

● Full Duplex ○ Half Duplex

MAC Address 0004.9A30.36CE

IP Configuration

○ DHCP

● Static

IP Address 172.168.1.2

Subnet Mask 255.255.255.0

IPv6 Configuration

Link Local Address: FE80::204:9AFF:FE30:36CE

○ DHCP

○ Auto Config

● Static

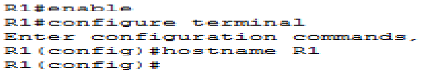
IPv6 Address 2340:1111:AAAA:1::2 / 64

Fig. 2. Configuration of IPv4 and IPv6 addresses.

The following describes the routers configuration; taking router 1 as example. Like in the hosts, the router is selected and unfolds a visual interface that has the three tabs, the Physical one has the same functions as mentioned in the host, the tab Config is used to perform the router basic configurations in graphical mode, and the tab CLI (Command Line Interface) is to perform configurations with commands.

Before configuring the router it is necessary to add serial interfaces since the router 2811 used does not have them, this was done by adding the WIC-2T module to the slot in the router, on the tab Physical. The router was configured by commands, in the tab CLI.

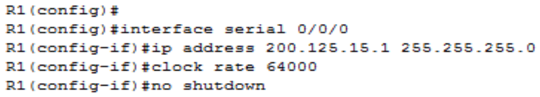
Figure 3 shows the change in the router's mode from user to global configuration and the assignment of the name "R1".



```
R1#enable
R1#configure terminal
Enter configuration commands,
R1(config)#hostname R1
R1(config)#
```

Fig. 3. Modification of the router's name.

Configuración de IPv4 y RIPv2 en el router R1. En la figura 4 se puede observar los comandos de la asignación de la dirección IPv4 a la interfaz serial, la asignación de la frecuencia de reloj a la interfaz y al final se levanta la interfaz.

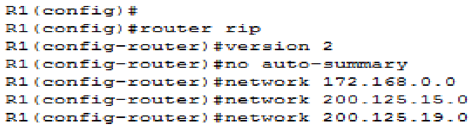


```
R1(config)#
R1(config)#interface serial 0/0/0
R1(config-if)#ip address 200.125.15.1 255.255.255.0
R1(config-if)#clock rate 64000
R1(config-if)#no shutdown
```

Fig. 4. Configuration of interface's IPv4.

The configuration of the remaining interfaces on the router R1 is similar to the above with the only difference that in the serial interface 0/0/1 as well as in the fastEthernet one 0/0, the clock frequency is not set.

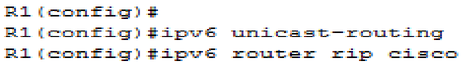
Figure 5 shows the procedure for configuring the routing protocol RIPv2 and the three networks that the router R1 will publish.



```
R1(config)#
R1(config)#router rip
R1(config-router)#version 2
R1(config-router)#no auto-summary
R1(config-router)#network 172.168.0.0
R1(config-router)#network 200.125.15.0
R1(config-router)#network 200.125.19.0
```

Fig. 5. Configuration of RIP version 2.

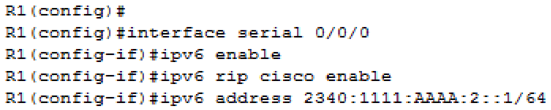
Configuration of IPv6 and RIPng on the router R1. Figure 6 shows the steps to enable the IPv6 traffic forwarding and the RIPng protocol.



```
R1(config)#
R1(config)#ipv6 unicast-routing
R1(config)#ipv6 router rip cisco
```

Fig. 6. Enabling IPv6 and RIPng.

In Figure 7 are the steps required to enable the IPv6 protocol in the serial interface 0/0/0, the routing protocol RIPng matching the process identifier with the previously assigned and finally is shown the assignment of the IPv6 address 2340:1111:AAAA:2::1/64. This process is done in a similar manner in the other interfaces and routers of the topology of Figure 1.



```
R1(config)#
R1(config)#interface serial 0/0/0
R1(config-if)#ipv6 enable
R1(config-if)#ipv6 rip cisco enable
R1(config-if)#ipv6 address 2340:1111:AAAA:2::1/64
```

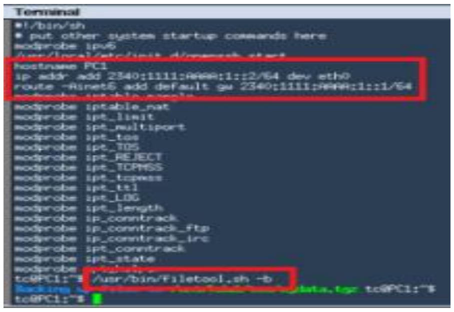
Fig. 7. Configuration of IPv6 and RIPng on the serial interface 0/0/0.

With all the previous performed configurations the network represented in the topology of Figure 1 is ready to carry out communications using the dual-stack transition method.

Since Packet Tracer does not support the tunneling transition method, it was necessary to choose another simulator, in this case the GNS3 simulator was used; its environment is very different from Packet Tracer because with this software is emulated the operative system of Cisco routers and also use some little GNU/Linux distributions; in this case the GNU/Linux Tinycore 3.8.2 was used. Since the object of study is the IPv6 protocol, will not be covered details of software configuration, will instead go directly to the configuration details of routers and PCs.

The following describes the network configuration using GNS3 simulator, in which the tunneling transition method was implemented using 3 PCs, 3 Ethernet switches and three routers 3725. Figure 1 shows the topology and addresses (IPv4 and IPv6) used. To configure the PCs with IPv6, the PC1 will be used as example. For which first the icon is selected and then Start.

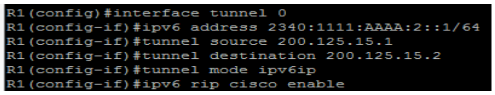
Once loaded the system desktop, a terminal was opened, then the next command was executed "vi /opt/bootlocal.sh" in order to edit the startup file so that in the next boot it was not necessary to reconfigure the PC. In figure 8 are shown the three necessary commands that had to be added to give to the PC a name and to assign the IPv6 addresses to both the interface and the gateway, the escape key close the editor and the command ":wq!" save the file, and finally a security copy can be done with the command "/usr/bin/filetool.sh -b". It is worth mentioning that all configurations were made in Linux super-user mode.



```
#!/bin/sh
# put other system startup commands here
modprobe ipv6
/usr/local/etc/init.d/openssh start
hostname PC1
ip addr add 2340:1111:AAAA:1::2/64 dev eth0
route -Ainet6 add default gw 2340:1111:AAAA:1::1/64
modprobe iptable_mangle
modprobe iptable_nat
modprobe ipt_limit
modprobe ipt_multiport
modprobe ipt_tos
modprobe ipt_TOS
modprobe ipt_REJECT
modprobe ipt_TCPMSS
modprobe ipt_tcpmss
modprobe ipt_ttl
modprobe ipt_LOG
modprobe ipt_length
modprobe ip_conntrack
modprobe ip_conntrack_ftp
modprobe ip_conntrack_irc
modprobe ipt_conntrack
modprobe ipt_state
modprobe [illegible]
tc@PC1:~$ /usr/bin/filetool.sh -b
Backing up [illegible]mydata.tgz tc@PC1:~$
tc@PC1:~$
```

Fig. 8. Configuration of PC1.

To make the settings of routers these were initiated in the same manner as the PCs; also were configured in the same way the IPv6 and IPv4 addresses in the necessary interfaces, as well as the same routing protocols used with the dual stack method; after that the tunnels in the routers were configured, taking router 1 as example, in the global configuration mode of the router was introduced the command to enable the interface: "interface tunnel", followed by an identifier, in this case, 0; an address was assigned with the command "ipv6 address" followed by the address and the prefix; then it was specified the tunnel source and destination with the commands "tunnel source" and "tunnel destination" followed by their respective IPv4 addresses; it was introduced the command "tunnel mode ipv6ip" to specify that the tunnel was manual and that IPv6 is the passenger protocol, being IPv4 in charge of encapsulate and transport IPv6; and finally RIPng was enabled with the command "ipv6 rip cisco enable" (Figure 9); this was done in a similar way in other routers so that the network would be ready to communicate using tunnels.



```
R1(config)#interface tunnel 0
R1(config-if)#ipv6 address 2340:1111:AAAA:2::1/64
R1(config-if)#tunnel source 200.125.15.1
R1(config-if)#tunnel destination 200.125.15.2
R1(config-if)#tunnel mode ipv6ip
R1(config-if)#ipv6 rip cisco enable
```

Fig. 9. Configuration of the tunnel on router 1.

With all the previous described several networks was configured and tested using IPv4 and IPv6 as addressing protocols, RIP and RIPng as routing protocols and two widely used transition mechanisms were implemented, such as tunneling and dual stack; the above through the use of the GNS3 and Packet Tracer simulators.

7. Conclusion

When performing the connectivity tests between devices was observed that effectively there was communication between them. In the case of dual stack transition mechanism, communication was for the IPv4 protocol as well as for the IPv6 protocol; for the tunneling transition

mechanism, the encapsulation of IPv6 packets within IPv4 was successfully, and therefore there was communication between the IPv6 local area networks.
During the time in which deployments were carried out, it was concluded that both mechanisms are good according to the scope of the network, since the dual-stack mechanism is easier to implement but devices must support both addressing protocols (IPv4 and IPv6), which makes routing tables to increase considerably and this creates processes and longer times. On the other hand the tunneling transition mechanism is a good choice for networks that have devices that do not yet support IPv6, since this can travel encapsulated, the disadvantage of this mechanism is that the MTU increases in 20 bytes the header of each IPv4 packet, besides troubleshooting becomes more complicated.

Regarding simulators, both are good. Packet Tracer is easy to use because the majority of functions are integrated in the software, but the version used during the investigation did not support certain configurations such as the tunneling transition mechanism, therefore GNS3 was also used, since it supported that feature, however, it becomes more slow to use since the software does not include all required for a network configuration and requires that the user configure several additional things, its advantage is that it is free and very powerful.

The next step in this research it that the performed tests can be made in a laboratory environment to verify in a physical way the functioning of protocols and transition mechanisms, so the Tizimin Multidisciplinary Unit is better prepared for the final transition from IPv4 to IPv6.

References

[1]. H. Houassi, A. Bilami, IP address lookup for Internet routers using cache routing table, International Journal of Computer Science Issues, Vol. 7, No 8, 2010.
[2]. B. Luis, Crece la potencia de la Red, El Informador, 2004, México.
[3]. M. Francisco, Planificación y Administración de Redes, Ra-Ma, 2010.
[4]. Nueva era en Internet: Se terminó el stock central de direcciones IPv4 de Internet, 2011, http://lacnic.net/sp/, última consulta 6 Junio de 2012.
[5]. Lanzamiento Mundial de IPv6 2012, http://www.isocmex.org.mx/ipv6_2012.html, última consulta 7 Junio de 2012.
[6]. IPv6 México, http://www.ipv6.unam.mx/, última consulta 7 Junio de 2012.
[7]. F. Azael, IPv6 ¡toda una realidad! 2005, http://www.enterate.unam.mx/Articulos/2005/enero/ipvseis.htm, última consulta 7 agosto de 2012.
[8]. A. Abu, Comparison study between IPV4 & IPV6, International Journal of Computer Science Issues, Vol. 9, No 1, 2012.
[9]. IPv6 la transición necesaria, Computerworld, 2004, http://www.idg.es/computerworld/articulo.asp?id=154237, última consulta 28 Julio de 2012.
[10]. D. Yezid, et al, Prueba de conectividad y tiempo de respuesta del protocolo IPv6 en redes LAN, Redalyc, No. 011, 2002, pp. 55 – 68.
[11]. V. Bob, et al, Acceso a la Wan, Guía de Estudio de CCNA Exploratión, Cisco Press, 2009.
[12]. H. Silvia, IPv6 Essentials, O'Reilly, 2006.
[13]. J. Felipe, Estudio e Implementación de una Red IPv6 en la UTFSM, Título De Ingeniero Civil Telemático, Universidad Técnica Federico Santa María, Valparaíso Chile, 2009.
[14]. A. Ernesto, B. Enrique, Redes Cisco CCNP a fondo, Guía de estudio para profesionales, Alfaomega, 2010.
[15]. C. Mariano, et al, El protocolo IPv6, Departamento de electrónica Facultad de Ciencias Exactas, Ingeniería y Agrimensura, Universidad Nacional del Rosario, 2006.
[16]. O. Wendell, CCNA ICND2. Guía Oficial para el Examen de Certificación, Cisco Press, 2008.
[17]. A. Oscar, Migración del protocolo IPv4 a IPv6, ContactoS 79, 2011, pp. 55 - 60.

Lefty Valle-Rosado. He studied Computer Science from the Autonomous University of Yucatan – Multidisciplinary Unit Tizimín. (UADY) in 2012. Currently he is developing systems and working in networks.

Lizzie Edmea Narváez-Díaz. Received a degree in Computer Science from the the Autonomous University of Yucatán (UADY) in 1997. She received a Master of Computer Science degree from Monterrey Technological Institute (ITESM), Campus Cuernavaca, in 2007. She has been a full time teacher at the Autonomous University of Yucatán since 2000. She has participated in software engineering development projects. Currently she is giving courses on networks in the professional programs in the UADY.

Cinhtia Maribel González-Segura. Master in Computer Science for the Institute Technology of Monterrey in México, is professor of the Autonomous University of Yucatán, Actually responsible of several projects. His researcher lines: Optimization, Artificial Intelligence, Mobile Robots.

Victor Manuel Chi-Pech. Obtained his degree in Computer Science from the Autonomous University of Yucatan (UADY) in 1996 and his M. Sc. degree in Wireless Network from Monterrey Technological Institute (ITESM), Campus Cuernavaca, in 2007. Victor Chi has worked since 2000 at the Autonomous University of Yucatan, as a full time professor. He has participated in software engineering development projects.