Scientific Journal Impact Factor (SJIF): 1.711
International Journal of Modern Trends in Engineering and Research
IJMTER
ISSN (ONLINE) : 2349 - 9745
ISSN (PRINT) : 2393 - 8161

IPV6 AND FUTURE WORLD WIDE WEB

K.CHOKKANATHAN[1],DR.P.SHANMUGARAJA[2]
Research Scholar, Veltech University
Asst.Prof IT,Sona College of Technology, Salem.

***Abstract*—**Each gadget on the Internet is doled out an IP address for recognizable proof and area definition. With the fast development of the Internet after commercialization in the 1990s, it got to be obvious that much a bigger number of address locations than the IPv4 location space has accessible were important to join new gadgets later on. IPv6 utilizes a 128-bit location, permitting 2128, or give or take 3.4×1038 locations, or more than 7.9×1028 times the same number of as IPv4, which utilizes 32-bit addresses and gives roughly 4.3 billion locations. The two conventions are not intended to be between operable, confusing the move to IPv6. On the other hand, a few IPv6 move instruments have been formulated to allow correspondence in the middle of IPv4 and IPv6 has. IPv6 gives other specialized profits notwithstanding a bigger tending to space. Specifically, it allows progressive location assignment routines that encourage course accumulation over the Internet, and in this way confine the development of directing tables. The utilization of multicast tending to is extended and rearranged, and gives extra improvement to the conveyance of administrations. Gadget portability, security, and arrangement angles have been considered in the configuration of the convention. Here i want to talk about this cutting edge convention and its supporting the development of the internet

Key words— bigger, progressive location, utilization, portability, security.

I. INTRODUCTION

Fundamental basis of the Internet is the TCP/IP protocol suite. Virtual Data communications among companies and individuals are based on these Internet protocols. Web browsers, electronic mail, file transfers, and remote logins are using this protocol suite. In the forthcoming years, the Internet will face an important transition that will enable it to grow beyond its current limitations. We need a newer, more reliable version of the Internet protocol (IP) — IPv6.

II. INTERNET PROTOCOLS AND GROWTH OF WORLD WIDE WEB

Today Internet usage is growing anything like that. Around 56 million hosts are identified in DNS. This is up from over 36 million a year earlier and from less than two million in July of 1993. So the annual growth of the Internet usage is more than 65 percent. This count includes only those systems that are publicly advertised by DNS, private nodes are not included in this count.

Due to this growth and its predictable future requirement we need a new version of the Internet protocol. With the Internet protocol, the total number of possible unique Internet addresses is in the four billion ranges. However, this is a theoretical limit and is actually constrained by several factors to a number in the range of a few hundred million addresses. Currently, there are estimated to be approximately 100 million computers connected to the Internet. Still it's not an absolute value, we don't know that when the Internet will run out of addresses, most of the experts' estimation is somewhere between the years 2002 and 2012.

Previously the PC is the only device which takes up an Internet address more cost also but now there are so many hand held devices like PDAs, ipads, notebooks, palmtops, cell phones, automobiles, and etc., are invented with inbuilt computers and will cost significantly less than a PC. As these devices start to network, many of them will require Internet connectivity. This will dramatically energies the need for more IP addresses.

The current internet version is IPv4, which was introduced in the year 1981. IPv4 addresses based on 32 bits. At that time approximately one thousand computers on the internet and they served universities and government offices with time share basis. 1981, there were four billion addresses available under IPv4 which seemed like a larger value. Later many of the addresses are not distributed properly, many of them wasted and could not be reallocated. In addition, some of the potential address space is used to make networks more maintainable and routable using blocks of addresses, which further consumes available addresses. To overcome the limitations of IPv4, the industry recognized the need to move from Version 4 (IPv4) to Version 6 (IPv6) of the Internet Protocol. IPv6 uses an address scheme that is based on a unique Internet address of 128 bits, as opposed to the 32 bits of IPv4.
One way to understand how many addresses 128 bits represents is to think of a book that describes every possible Internet node address with 100 node descriptions per page. If each page was .1mm thick and was printed double-sided, the book to describe all possible IPv4 addresses would be 2000 meters thick. This may sound big, but the book describing all possible IPv6 addresses would be $\mathbf{2*10^{16}}$ light-years thick. To date, the farthest sight seen by the Hubble telescope is less than $\mathbf{2*10^{13}}$ light years. Certainly IPv6 is considered quite adequate for the foreseeable future.

III. IP-VERSION A UPGRADED TECHNOLOGY

When the Internet Engineering Task Force (IETF) began designing IPv6, it was recognized that for an extended time, both IPv4 and IPv6 nodes would co-exist on the network. To help make the transition, the IPv6 code interface was designed to communicate with both IPv4 and IPv6 nodes. This is achieved by supporting both protocols through a single programming interface. Specifically, the new socket interface API for IPv6 will support both IPv4 and IPv6 communication transparently.
Using this new interface, applications that have been upgraded can automatically use IPv4 to talk to existing IPv4 clients and servers, as well as to IPv6 systems that have been configured with only an address for the IPv4 protocol.
Once applications have been upgraded to the new IPv6 interface, they can start using the IPv6 protocol to connect to other IPv6 nodes. This will occur as soon as a company deploys an IPv6 network. Most importantly with this transition is that the application need not worry about which protocol is used. Once the new interfaces are used and the system is configured for IPv6 (given an IPv6 address and configuration), the IPv6 socket interface will automatically choose the right protocol to use to communicate with any given node.

IV. IPv6: THE NEXT GENERATION PROTOCOL

IPv6 has a significantly different header, making the two protocols not interoperable. IPv6 also specifies a new packet format, designed to minimize the computing overhead in processing the packet headers. The key features of IPv6 telling that the Ipv6 is the next generation protocol, like

4.1 Larger address space

The most important feature of IPv6 over IPv4 is that it provides a 128 bit address compared to a 32 bit one for IPv4, increasing the addressing capability by several trillions. An example of the addressing provided by IPv6 is shown below.

4.2 Stateless Address Auto-configuration (SAA)

SAA is a technical term describing a mechanism in which IPv6 hosts can automatically configure themselves using predefined handshake signals and messages defined by ICMPv6 – a protocol designed to augment IPv6. ICMPv6 provides for several messages which are used for diagnostic and error reporting activities.

4.3 Multicasting

Multicasting is the ability of a host to send out a single packet to multiple destinations. IPv4 included multicasting as an optional feature (although it was most commonly implemented in all hardware and associated software) whereas IPv6 specifies multicasting as a mandatory feature.

4.4 Included network layer security

IPv6 includes a protocol for encryption and authentication – IPSec. This feature too was optional in IPv4 (although it was usually implemented) but made mandatory in IPv6.

4.5 Simplified processing by routers

To make the process of packet forwarding easier, a number of simplifications, the details of which are beyond the scope of this blog, were made to the IPv4 packet header forming the base for the IPv6 header. These changes were targeted at making data forwarding by routers simpler and hence more efficient.

4.6 Mobility

Mobile IPv6 (MIPv6) dropped the concept of triangular routing, in which a packet is sent to a proxy before being sent to the intended destination. Hence, mobile IPv6 is as efficient as normal IPv6. Theoretically, this would also increase the efficiency of data flow.

4.6 Extensibility of Options

Options in IPv6 are implemented as additional extension headers after the IPv6 header, providing the size of an entire packet for implementing an option. This is unlike IPv4, which limits the options parameters to a fixed size of 40 octets.

V. COMPARATIVE STUDY ON IPv4 and IPv6

Most of the growth of the Internet has occurred using the IPv4 specification of the Internet protocol. While this version of the protocol has done an excellent job of handling the tremendous growth of the Internet, it is rapidly approaching the end of its physical limits. Given anticipated growth of the Internet in coming years, and the necessary upgrade of millions of network nodes and their users, a new protocol is required. However, before discussing the details of IPv6, it is important to understand the need for this transition.

IPv4 Background

The Internet Engineering Task Force published the IPv4 specification (RFC 791) in the fall of 198l. When the IPv4 specification was released, the Internet was a community of approximately one thousand systems. The IPv4 specification called for every IP address to be represented by a 32-bit number made up of four groups of eight-bit numbers. This provides a total of just over four billion addresses, although only a few hundred million are actually available due to hierarchic allocation schemes.

Since the release of IPv4, the Internet population has grown to over 100 million computers, increasing far faster than anticipated. As the pool of available addresses decreases, it will become increasingly difficult to obtain IPv4 addresses. Furthermore, the pace of this growth is expected to continue for years to come. The bottom line is this: The Internet is running out of addresses. And by some hard estimates, this could happen as soon as 2002. Early IP assignments reserved addresses for some corporations and institutions in very large blocks. These “Class A” and “Class B” network assignments were issued in the early days when the current growth was not anticipated. While some early adopters may still have addresses available for internal usage, the pool of unissued addresses is becoming smaller every day

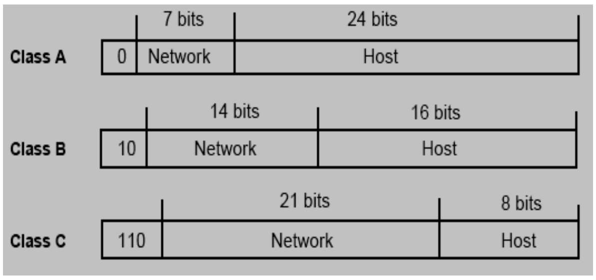

Figure 1: Mapping of network and host identifier in the 32 bit IPv4 protocol

VI. ALTERNATE MEASURE

In the short-term, Network Address Translation (NAT) is relieving the pressure for additional address space. By using some special addresses (such as 10.*.*.*, and 192.168.*.*) that are reserved for local usage, NAT allows network administrators to hide large communities of users behind firewalls and NAT boxes. The rest of the Internet sees all of the requests for users in the community as coming from one NAT box. When the response to the request comes back to the NAT box, it forwards the information to the appropriate local user. Since multiple, different corporate networks can each reuse the same local addresses, NAT reduces the need for new unique Internet addresses. Unfortunately, NAT is a stopgap measure, not a permanent solution. It addresses the needs of large communities of client systems, but it does not help Internet servers that each require a unique permanent address. Nor does it work for peer-to-peer communication for the same reason. NAT also breaks the end-to-end model of Internet access and undermines the IPsec model of Internet security.

IPv6 is the Long-Term Solution

The IPv6 protocol solves the pressing problems of IP addressing, while simultaneously making administration easier. It uses an address scheme made up of eight groups of 16 bits, defining a 128-bit network address. This addressing scheme provides roughly 6*1023 addresses per square meter over the entire face of the earth. As a result, it is expected that the IPv6 protocol will last many years into the future.

VII. ENDURING SOLUTION

The IPv6 protocol solves the pressing problems of IP addressing, while simultaneously making administration easier. It uses an address scheme made up of eight groups of 16 bits, defining a 128-bit network address. This addressing scheme provides roughly 6*1023 addresses per square meter over the entire face of the earth. As a result, it is expected that the IPv6 protocol will last many years into the future.

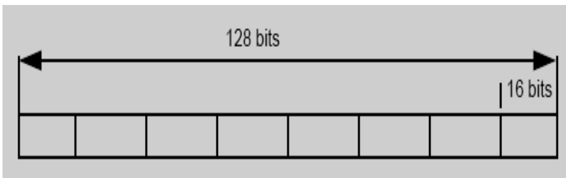

Figure 2: The 128 bit Ipv6 address format. Each 16 bit segment is identified by 4

hexa decimal digits with colons separating the segments. A single double colon can be used in the address to represent one or more segments of all zeros.

With a pool of addresses this large, it is possible to utilize some of the addresses to make network management and routing easier. The new address scheme allows automatic address configuration and reconfiguration (servers can re-number network addresses without accessing all clients, and mobile clients can move about the network). NAT servers are no longer required because there is no need to use private addresses. Each computing device can have a its own unique address without fear of running out of addresses.

VIII. FUTURE IMPLEMENTATION USING IPv6

The IPv6 specification has several possible APIs to enable IPv6 communications, and most are IP-version independent. By using these APIs, developers can write a single segment of code that will support both IPv4 and IPv6 communications. Based on the name of the system that is the target of communication and the configuration of the current node, the API will determine the target IP address and whether it's using IPv4 or IPv6 protocol. By using IP version independent APIs, developers can enable IPv6 communication transparently.

The most common API, particularly in the short term, is the Basic Socket API. Once an application has been ported to this interface, the socket code determines, based on the current configuration, whether to communicate using IPv4 or IPv6. The application automatically decides which version of the protocol to choose. This enables the use of IPv4 now and requires no new coding to utilize IPv6 later. Code once and get both protocols. Name servers manage IPv4 and IPv6 environments and their cooperation. Once a name server has been upgraded to support IPv6, it will return the appropriate address (IPv4 or IPv6) for the remote system based on the capabilities and configuration of the environment. Using the name server to manage information about IPv4 and IPv6 makes code development and management much easier for developers. More detail on how name servers work is available in Chapter 4, "Porting Client Code to IPv6".

Using Socket Scrubber, developers can quickly identify where changes need to be made, easing the process of upgrading code to use the new interface. For more detail on this process, see Chapter 6, "Tools to Help Upgrade Applications". All new development should use IP version independent APIs. Since these interfaces support both IPv4 and IPv6, their use assures a seamless upgrade path to the IPv6 standard. Continuing to use IPv4 interfaces at this point is to build in obsolescence. IPv6 is the future of the Internet. The question is not whether IPv6 will be the future standard protocol for the Internet, but rather when this transition will occur. Developers should prepare for the future now by being proactive with new development and effectively planning the conversion of current IPv4 applications.

8.1 Specification of IETF

The IETF has released several RFC specifications for the transition to the IPv6 environment. The most significant specifies the basic socket communication used in 90 to 95 percent of all IPv6 applications. In addition, the IETF is currently working on an RFC — an advanced IPv6 API — that will deliver additional socket capabilities and extended features to enable greater levels of control for programmers. It is expected that less that 10 percent of IPv6 applications will use this extension.

RFC 2553 is a primary IPv6 specification that covers basic socket communication and identifies IPv6 extensions to the existing socket interface. RFC 2553 is the basic API because applications that use it will be insulated from the specific underlying protocol. Since this specification supports both IPv4 and IPv6 communication, it can be thought of as an upward extension of the current IPv4 specification. In general, a one-to-one relationship exists between the IPv4 and IPv6 calls. As a result, it is fairly simple to upgrade existing IPv4 socket-based applications to use the IPv6 interface.

8.2 API Socket Fundamental

There is very little doubt that, for an extended period of time, the Internet will be made up of both IPv4 and IPv6 hosts. For that reason, the IPv6 basic socket API supports both IPv4 and IPv6. This approach is called a dual-stack interface. Once an application has been upgraded to the IPv6 socket interface, no more

Volume 02, Issue 06, [June– 2015] ISSN (Online):2349–9745 ; ISSN (Print):2393-8161

code is required to enable communication with both IPv4 and IPv6 systems. When a call is made to the new socket interface, it will look at the data structures and determine if it is possible to communicate with this node using IPv6. If not, the socket will automatically make the connection using an IPv4 protocol connection. Since all current Internet software use IPv4, a dual-stack IPv6 application can communicate using IPv4 to all current software without any additional coding of the IPv4 applications.

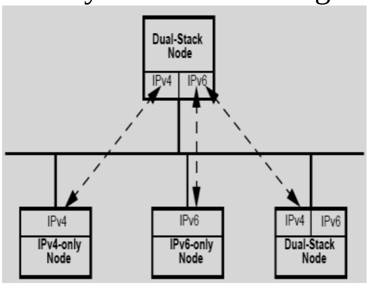

Figure 3: The dual-stack socket interface will default to using
the IPv6 protocol if it is available. If not, it will automaticallycommunicate using IPv4 protocol.

TABLE 1: Ipv4 and Ipv6 interoperability

Type of Application (Type of Node)	IPv6-Unaware Server (IPv4-Only Node)	IPv6-Unaware Server (IPv6-Enabled Node)	IPv6-Aware Server (IPv6-Only Node)	IPv6-Aware Server (IPv6-Enabled Node)
IPv6-unaware client (IPv4-only node)	IPv4	IPv4	X	IPv4
IPv6-unaware client (IPv6-enabled node)	IPv4	IPv4	X	IPv4
IPv6-aware client (IPv6-only node)	X	X	IPv6	IPv6
IPv6-aware client (IPv6-enabled node)	IPv4	(IPv4)	IPv6	IPv6

In the above table, “X” denotes that communication between the respective server and client is not possible at the IP layer. Other tools, like proxies, can be used for this communication. This table assumes that the dual-stack interface has both an IPv4 and IPv6 address in the respective name service(s). An IPv6-unaware node has the IPv4 stack only. An IPv6-enabled node has a dual stack and at least one interface IPv6 configured. An IPv6-only system implements only the IPv6 stack and only has one address in the name service database.

It is a straightforward process to convert IPv4 socket applications to use the IPv6 socket extensions. In general, there is a direct conversion to update the IPv4 socket call to the IPv6 socket call. In fact, there are some places where two or three IPv4 calls can be replaced by a single IPv6 call, although users are not required to use these shortcuts. The first argument of the socket call will change and a few of the parameters will need to be updated, but generally this is a very simple process. The format of the calls and the data structures are similar, and tools are available to help identify where changes are required.

Advanced Socket Capabilities

As previously mentioned, the socket specification for IPv6 has been formalized with RFC 2553 and is expected to become the primary standard for Internet communication. This specification is considered complete and can handle virtually every need for IPv6. It has been fully defined, and most system vendors have publicly stated their commitment to support it. On the other hand, there has been a growing consensus that a few additional capabilities beyond the current specification would be desirable. RFC 2292 is designed to offer some extensions to the current specification, enabling developers to have more control of IPv6-only communications. It is estimated that these additional control requirements would only be used by approximately 5 to 10 percent of all socket applications. For example, advanced capabilities enable operations such as source routing, raw socket access, interface identification, and extension headers not defined in RFC 2553 are all being considered. These advanced capabilities are being evaluated by the IETF, and Sun is working with other vendors to help solidify all of these advanced capabilities. As more information is available, Sun will continue to update the developer community.

8.3 Applications of Remote Procedure calls

Another method for applications to communicate is through Remote Procedure Calls (RPC). Using this interface, application developers can use these calls to communicate transparently with IPv6 clients. RPC facility provides another way to communicate using the IPv6 protocol. By using standard calls, applications can use the existing RPC conduit to communicate to either IPv4 or IPv6 nodes. Since all of the modifications to allow the RPC to communicate using IPv6, there is no need for application developers to re-code their applications. By using standard dynamic linking to the newest libraries, applications will transparently use IPv6 when appropriate.

8.4 Porting Client Code to IPv6

One of the key technologies making the Internet possible is name services. Name server technologies, such as the Domain Name Service (DNS) and the Network Information Service (NIS and NIS+), are used to translate logical references (such as the names of computers) to physical references (addresses, names, etc.). IPv6 uses these name-to-address translation routines to help smooth the transition from IPv4-to IPv6-based protocols.

Name Services and the IPv4 to IPv6 Transition

When the IETF began planning for the transition from IPv4 to IPv6, they quickly recognized that this could be a complex undertaking. To help simplify the process, the IETF created a document to help companies make the transition more easily. The resulting specification is RFC 1933. There are several methods that the IETF recommends for updating from pure IPv4 to an IPv6-enabled environment. Depending on the needs of the organization, the transition can start by converting the clients in the environment, the servers in the environment, or the routers between the environments. Regardless of where the transition begins, IPv6-enabled name servers are an important early requirement. Name servers connect the two IP environments and determine which protocol stack is accessed. IPv6-enabled name servers store the name-to-address translation for both IPv4 and IPv6 addresses.

When a request is made to translate a network name to an address, the name server will reply with the appropriate address based on the requesting node, target node, and call parameters. If both the requesting and destination nodes are capable of IPv6 communication (and the calling parameters allow it), the name server will respond with the IPv6 address. Using this address, the socket interface code will communicate using the IPv6 protocol. As IPv6 is enabled through the environment, the name server will automatically convert the common communication protocol from IPv4 to IPv6.

IPv4 and IPv6 Interoperability Using tools provided by Sun makes converting from an IPv4 environment to an IPv6 environment straightforward. After installing several IPv6 capable name servers, the upgrade to the IPv6 environment may begin at any place on the network. For example, nodes are transitioned from an IPv4-only system to dual-stack systems by upgrading the IP socket calls from the IPv4 socket-only interface to the dual-stack IP socket interface. An example of this dual-stack arrangement is shown in the following figure. After an application has been upgraded to use the dual-stack IPv4/IPv6 interface, the next step is to configure the IPv6 information. To do this, a user must configure such parameters as the IPv6 address of the node and ensure that the node is connected to the IPv6 network.

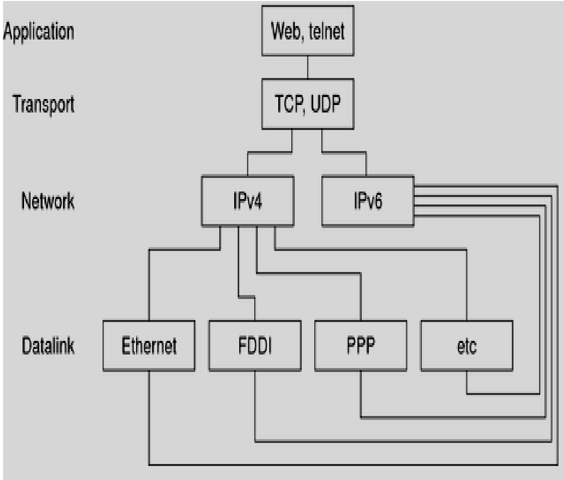

Figure 4: Dual-stack protocols

Once a node has been registered with the name server, that name server returns the IPv6 address for the destination, and packets are routed using IPv6 protocols. Any dual-stack node will have both an IPv4 and an IPv6 address in the name server database. The name server will automatically return the IPv4 address if it is the only address stored. If an IPv6 address, or if both IPv4 and IPv6 names are stored, the name server will return the IPv6 address. Based on the address returned, the application automatically selects the appropriate protocol and connects to the node.

IX. CONCLUSION

There's no doubt that IPv4 is going to be replaced by IPv6. The only question is when. With the rapid growth of the Internet, the pool of available addresses is quickly shrinking. And while tools like NAT perform a valuable stopgap role, the real solution is a new IP address standard that will increase the number of available addresses and allow easier configuration. Software developers need to begin focusing on protocol-independent, IPv6 development to ensure that their applications are viable for the future. All new applications should be developed using the new interfaces to support IPv6 — even if current networks are still IPv4 based. Using these new interfaces will also provide IPv4 communication support in the interim. The key advantage is that once customers begin to fully implement IPv6 networks, resulting applications will automatically switch over. Existing applications may be upgraded to use the new IPv6 interface as time and upgrade requirements warrant. Application developers can identify potential areas that need upgrading. This conversion from IPv4 to IPv6 is made easier because there is a direct relationship from IPv4 calls to corresponding IPv6 calls. There are such tools and the expertise to help customers prepare and transition to IPv6. With over 15 years of network experience, and an equally successful history of innovation, some Organizations provide the developer community with the tools and technologies to make the conversion to IPv6 easy and straightforward.

REFERENCES

[1] T. Narten, R. Draves, and S. Krishnan, "Privacy Extensions for Stateless Address Autoconfiguration in IPv6," RFC 4941, Internet Eng. Task Force, Sept. 2007.
[2] RFC 1886 — DNS Extensions to Support IP Version 6, S. Thomson, Bellcore,
[3] R. Hinden and S. Deering, "IP Version 6 Addressing Architecture," RFC 4291, Internet Eng. Task Force, Feb. 2006.
[4] T. Narten et al., "Neighbor Discovery for IP version 6 (IPv6)," RFC 4861, Internet Eng. Task Force, Sept. 2007.
[5] S. Thomson, T. Narten, and T. Jinmei, "IPv6 Stateless Address Autoconfiguration," RFC 4862, Internet Eng. Task Force, Sept. 2007.

[6] S. Deering and R. Hinden, "Internet Protocol, Version 6 (IPv6) Specification," RFC 2460, Internet Eng. Task Force, Dec. 1998.
[7] T.A. Almeida and A. Yamakami, "Content-Based Spam Filtering," Proc. 23rd IEEE Int'l Joint Conf. Neural Networks, IEEE Press, 2010.
[8] J. Wu and T. Deng, "Research in Anti-Spam Method Based on Bayesian Filtering," Proc. Pacific-Asia Workshop on Computational Intelligence and Industrial Application, IEEE CS Press, 2008.
[9] A. Ramachandran, N. Feamster, and S. Vempala, "Filtering Spam with Behavioral Blacklisting," Proc. 14thCM Conf. Computer and Communications Security, ACM, 2007