Translucent Implementation of IPv6 Addressing Scheme in Communication Networks

M. Raza Parwez[1], M. Ehsan ul Haq[1], M. Sajid Javaid[1], Kabeer Ahmed[2]
National University of Modern Languages (NUML) [1],
College of Telecommunication Engineering (NUST) [2], Pakistan
raza@numlit.edu.pk, muhammad.ehsan@gmail.com, sajid@numl.edu.pk,
kabeer-mcs@nust.edu.pk

Abstract

*This paper presents a translucent representation of currently implemented IPv6 address and proposes a more compact and end-user friendly format for IT professionals especially for naïve users. It has been evaluated that the next generation IPv6 address would not only facilitate network professionals but also be used by all other communities. IPv6 will also be employed on objects other than communication devices for tracking and remote administration viz. household electronic devices, mobile devices and even assign Human beings to track them. Considering the fact that it would be harder to remember 32 characters long IPv6 address separated by colons by humans like remembering telephone numbers, this paper presents an alpha-numeric IPv6 address Masking which contains 0-9, a-z, A-Z, . (dot) and – (Hyphen) using base64 number system. Total length of address reduces from 39 characters (32 + 7 colons) to the maximum of 22 characters that is approximately 56% (22/39*100). The proposed 22 characters address which is a user-friendly address could be further compressed by using "6 5 4 rule" which has also been proposed in this paper. Another approach that can be used as easy to remember IPv6 addresses is Domain Name Services (DNS). It translates IP addresses into human acceptable names and vice versa. However, it has been evaluated that the promulgation processing of modifying in DNS Server requires days in case of IPv4 and it multiplies in the case of IPv6. An authoritative response from a non-authoritative DNS server is also proposed for IP address resolution and tested, through which not only request time out has been reduced but also DNS response time decreased to 50% approximately in comparison with the existing DNS resolution process.*

1. Introduction

The next generation Internet Protocol version 6 (IPv6) is a networking protocol, which would be replacing IPv4 and allows users that include professional, naive users and users from all walks of life, to communicate with other users over the Internet. The major organizations in the fast growing market of Asia and Europe, as well as mobile service providers worldwide are under increasing pressure to migrate from the entrenched IPv4 standard to the emerging IPv6 one [1]. IPv6 is designed to meet requirements that did not exist in earlier IPv4. It provides capabilities that go beyond large address [2, 5, 8]. A unique identifier is capable of acting as the reference that enables communication and services. In the network world, a globally unique identifier that is based upon the Internet Protocol address [4].

Mr. Robert Elz proposed a base85 IPv6 addressing scheme, which identify a compact representation of IPv6 addresses that allow encoding in 20 bytes, although no implementation has been found on the extended version of IPv6 on his recommendations [6]. Due to large size of the available address space in IPv6, a hierarchical structure of the allocation space is necessary to permit the aggregation of routing information [7]. It interacts with all versions of naming services such as Domain Name System (DNS) and uses security techniques and technologies such as Internet Protocol Security (IPSec) because these services facilitate the successful and secure transfer of IP packets between two end points. At this point of time, pure IPv6 transmissions are attainable only through routers that support IPv6 and computers, which are running those Operating Systems that support IPv6.

On the other hand, DNS is the core in distributed database of hosts' name that provides mapping between hostname to an IP address, which is necessary predecessor of communication from source to destination [9]. For a DNS query one or more remote name-servers are involved in resolving the query. Resolution typically uses UDP exchanges and use timeout and retransmission, which adds a delay [10-12]. Textual names conform to certain structural conventions and the names are in the form of standard Resource Records (RRs). If client application needs to communicate with another party, but it does not have any information about logical addresses assigned; i.e. IP address (either version), except Fully Qualified Domain Name

(FQDN), it queries one of the nearest name servers and gets appropriate answer in response to the query.

Edith proposed two orthogonal approaches to reduce the delay in validating client request [13]. First, the renewal policy is effective for hostname where the typical time interval between requests is within some small factor of the TTL. Second, the simultaneous validation is useful when frequency of change of hostname to address mapping is lower than the frequency of request [13-15].

A resolution is the process that queries the root server which is authoritative for the domains i.e. .net, .com, .pk etc. The root server returns a reference to the name server authoritative for the name and sub-domains are delegated to other servers that are authoritative for their portion of the name space. If the returned name server lies outside the root-server's zone, then it is resolved separately [13].

This paper provides a translucent addressing scheme and provides new masking technique for IPv6, which is very easily understandable for human and memorize it. Section 2 introduces IPv6 addressing, section 3 identifies the problem with base16 addressing, section 4 discusses the first approach for base64 addressing convention, section 5 introduces an authoritative response from non-authoritative server and section 6 provides a conclusion of whole study.

2. IPv6 addressing

In contrast to IPv4 addressing that is represented in dotted-decimal format and is 32 bit addresses. It is divided into 8-bit boundaries where as IPv6 is 128-bit address that is represented in colon-hexadecimal format and is divided along 16-bit boundaries. Each 16-bit block corresponds to a 4-digit hexadecimal number, separated by colons [3]. IPv6 provides an extremely large address space with 128-bit-long address. How it does not change the data plane function of a router [2]. Table 1 shows the IPv6 address in binary format i.e. (16 bits * 8 rows = 128 bits address)

Table 1: IPv6 Binary Format

128-bit address			
1100	0111	1000	1100
0010	0111	1111	1010
1000	0010	0101	1110
1111	1011	0011	0101
0011	1111	1101	0101
0111	0011	0110	0000
0010	1000	1011	1111
1111	0101	0000	0001

Now look at another type representation of IPv6 that is in hexa-decimal format and delimited with colons (:) i.e. (32 Hexa-characters * 4 bits = 32*4 = 128 bits) [3].

C78C : 2FFA : 835E : FB35 : 3FD5 : 736F : 28BF : F501

The longer address length and larger address space IPv6 pose a challenge to existing internet routers [7] and the total possible addresses in IPv6 are:

i. 340,282,366,920,938,463,463,374,607,431,768, 211,456
ii. (2)128 or 3.4028236692E+38
iii. (16)32 or 3.4028236692E+38

2.1 IPv6 Base16 Addressing Convention

The IPv6 address convention in base16, which represented in textual format. The IPv6 conventions described in detail, which is as follows [5]:

i. IPv6 is a Hexa-Decimal format addressing scheme
ii. Maximum 32 characters long addresses
iii. Each IPv6 address contains 8-blocks (x:x:x:x:x:x:x:x)
iv. Each Block contains 4 characters in Hexa-Decimal format separated by colons (:)
v. Each character represents 4-bits.
vi. IPv6 is a 128-bits addresses (8-blocks * 4-Characters * 4-Bits = 128 bits)
vii. Leading zeros are removed in each block
viii. All zeros in multiple blocks would be replaced with '::' e.g. 2008:0000:0000:0000:0000:0000:0F01:00C3/prefix equals to 2008::0F01:00C3/prefix or can also be written as 2008::F01:C3/prefix
ix. Loop-back address is represented as ::1
x. Unspecified address is represented as :: (Double Colons)
xi. IPv4-compatible dot-notated addresses are represented as ::w.x.y.z
xii. IPv6 Hexa-decimal format addresses are case insensitive and can be written in any case.

3. Problem in IPv6 Base16 Addressing

In contrast to IPv4 addressing which is a 32-bits (4 octet notation) long dot-notated address and are purely based on decimal format viz. 192.168.145.123/prefix, IPv6 addresses are 128-bits long hexa-decimal format contains 32 characters estranged by colon (:). An example of IPv6 would be like C78C:2FFA:835E:FB35:3FD5:736F:28BF:F501 /prefix.

While looking at both versions of IP addresses, it has been perceived that, it would be harder to remember IPv6 format address than memorizing

IPv4 format address. Though, we have compressed format of IPv6 available viz. 2007::25/prefix but the said technique only reduces multiple and leading zeros. Another thing which needs to be taken into consideration that IPv6 is in a very initial stage of deployment and we have surplus of IPv6 addresses available. When this version is deployed to its full capacity, a long addressing scheme of IPv6 will be experienced. At that time it would be quite harder to remember unfriendly meaningless hexadecimal format IPv6 multiple addresses.

It should also be taken into consideration that who would be the ultimate operator and consumer of this IPv6 addresses. IPv4 is mainly utilized for data communication purpose and be operated by IT professional, but the newer next generation version of IP would also be used for remote administration viz. electronic devices and tracking of objects which includes Human-beings in addition to its traditional use for data communication over Intranet, Extranet and Internet.

In case of remote administration of electronic devices and locking system of a house (an example), the end-user is the one, who will not be familiar with the technicalities of IPv6 and unable to remember multiple IPv6 addresses, which have been assigned to the household devices. For-example, a system will be designed for the residents of a house who would be able to check the lights of their houses whether they are on/off or the uncooked food which has been placed in a microwave oven is cooked or not, from a remote location or manage to lock and unlock their houses remotely, an IP enabled devices would have been the solution, but the problem would be to remember the long format 128-bits IP addresses which have been assigned to the devices to their houses.

Currently, the newer version of IP i.e. IPv6 is the only available solution which fulfills the intensive demand of IPs.

4. Approach-I: Base64 Addressing Convention

i. The character set to encode the base64 IPv6 address is: 0 to 9, a to z, A to Z, . (Dot) and – (Hyphen)
ii. Case sensitive IP scheming
iii. Each character represents 6-bits
iv. Last character has to be among 0 to 3
v. Maximum number of characters are 22 or more precisely 21.33 characters
vi. Total number of addresses: $(64)^{21.33}$ equals to 3.4028236692E+38
vii. Resultantly: $(2)^{128} = (16)^{32} = (64)^{21.33}$
viii. 22 characters multiply by 6-bits = 132 bits
ix. Compress 132 bits into 128 bits by removing 4-bits preceding to second last bit from 132 bits array
x. In contrast to base16 addressing scheme, base64 addressing scheme compresses zeros (0), dots (.) and hyphens (-)
xi. Multiple zeros, dots and hyphens would be replaced by ":" , "~" and "+" respectively
xii. Each ":" , "~" and "+" represents 5 consecutive zeros, dots and hyphens respectively
xiii. Maximum 4 compact characters are used. This rule defines as "6 5 4 rule". This means that each character represents 6-bits, 5 successive zeros, dots or hyphens and maximum 4 compact characters in an address
xiv. Loop-back address is represented as ::1
xv. Unspecified address is represented as :: (double colons)
xvi. # Sign will be used for defining the IPv4-compatible dot-notated addresses
xvii. # Sign must only be the 17th character, e.g. numlit.edu.pkisb#192.168.150.10

4.1 Proposed "6 5 4" Rule

This proposed rule defines the compression of base64 IPv6 addressing standard. Each zero, dot & hyphen represents 6-bits, 5 consecutive zeros, dots & hyphens will be replaced by ":" , "~" & "+" sign respectively, called "Compressed signs" and there will be a maximum of 4 compressed signs in a base64 IPv6 address. Table 2 shows compressed and uncompressed signs with respective number of bits.

Table 2: Compressed and Uncompressed Signs

Uncompressed Signs	Compressed Sign	No. of bits
….. (5 dots)	~	30 bits
----- (5 hyphens)	+	30 bits
00000 (5 zeros)	:	30 bits

4.2 "6 5 4 rule" examples

Examples of "6 5 4 rule" are demonstrated as under:

i. NUML……------RnD-01= NUML~.+-RnD-01
ii. 2008Y000000000000000a1 = 2008Y:::a1
iii. MicroSoft.com-----0Aa2= MicroSoft.com+0Aa2
iv. -----…..00000abcde-2 = +~:abcde-2

4.3. Proposed Solution IPv6 Base64 Masking

Traditional telephone numbering system shows that it is difficult for users to remember telephone / mobile numbers along with the users' name. On the other hand, it is convenient and easy to remember the standardized alpha-numeric numbering system Table 3 shows few examples of proposed masking solution.

Table 3: IPv6 Masking Examples

S.No	Number	Base64 masking
1	111-686-548	111-NUMLIT
2	0345-3628677	0345-DOCTORS
3	051-5299377	051-LAWYERS
4	042-2547292	042-ALIRAZA
5	0333-3444663	033EDHIHOME

It has been observed that it is easy to memorize / remember the alpha-numeric representation of the telephone / mobile numbers as compare to remember pure numeric numbering conventions. Same is the case with Hexa-decimal representation of IPv6 addressing. It is hard to remember 32 characters IPv6 address (Actually 39 characters; 32 + 7-colons which separate blocks). Therefore, a more user-friendly and compact alpha-numeric representation is proposed, which will be used in base64 numbering system and each character in an address will represent number of bit i.e. 6-bits rather than 4-bits used by base16 IPv6 format Table 4 shows the bit-wise conversion of the proposed representation.

4.4 IPV6 Base64 examples

Examples of Valid IPV6 Base64 address are as under:

i. NUML.EDU.PK.ISB-#10.10.20.30
ii. encyclopedia.com.US-02
iii. IT------------------.1
iv. IT+++---.1
v. IEEE-AaBbCcDdEeFfGgHh3
vi. ::1 (Loopback Address)

4.5 Base64 IPv6 Address v/s Domain Name System (DNS)

This proposed address masking on IPv6 reduces the role of DNS. Considering the fact that, all IP addresses could not possibly have entries on DNS Server. IPv6 contains a very large number of IP addresses i.e. $(2)^{128}$ and requires high-end processing & caching servers to map all domain names to IPv6 for resolving without any considerable delay. It is further proposed that these masked IPv6 addresses could be used in replacement of Domain names in some extent, viz.

“numl.edu.pk-index.htm1” could be represented a web-server of NUML
“numl.edu.pk-webmail-11” could be represented a mail server of NUML
“Microsoft.com-websvr02” could be represented a web-server of Microsoft Inc.

Table 4: IPv6 bit-wise conversion

Base 10	Base 64	6-bit conversion	Base 10	Base 64	6-bit conversion
0	0	000000	32	w	100000
1	1	000001	33	x	100001
2	2	000010	34	y	100010
3	3	000011	35	z	100011
4	4	000100	36	A	100100
5	5	000101	37	B	100101
6	6	000110	38	C	100110
7	7	000111	39	D	100111
8	8	001000	40	E	101000
9	9	001001	41	F	101001
10	a	001010	42	G	101010
11	b	001011	43	H	101011
12	c	001100	44	I	101100
13	d	001101	45	J	101101
14	e	001110	46	K	101110
15	f	001111	47	L	101111
16	g	010000	48	M	110000
17	h	010001	49	N	110001
18	i	010010	50	O	110010
19	j	010011	51	P	110011
20	k	010100	52	Q	110100
21	l	010101	53	R	110101
22	m	010110	54	S	110110
23	n	010111	55	T	110111
24	o	011000	56	U	111000
25	p	011001	57	V	111001
26	q	011010	58	W	111010
27	r	011011	59	X	111011
28	s	011100	60	Y	111100
29	t	011101	61	Z	111101
30	u	011110	62	.	111110
31	v	011111	63	-	111111

5. Approach-II: Authoritative Response from Non-Authoritative DNS Server

It is further proposed that rather than additional mask a Hexa-decimal format address into Base64 number scheme; which is originally a 128-bit binary number convention (binary to Hexa-decimal to Base64 numbers), a more efficient DNS resolution process should be adopted for masking an IPv6 addresses. It is therefore proposed that instead of caching all DNS entries in a single bucket, it should be split into two different levels. Level-1 is used to maintain glue records (associated NS records) of each domain including IP address(es) of each name server for a particular domain and Level-2 is used to cache host addresses, mail exchangers, and pointers etc. along with their associated IP address(es) of a particular domain and type of cached entry. While resolving a host, non-authoritative NS first checks the caching entries from its Level-2 and replies associated IP address; if not found, it is referred to Level-1 of its database structure and it is forwarded to the associated domain's authoritative NS to resolve. If entry is found in Level-1 but the requesting host does not reach to the associated domain's authoritative server, non-authoritative NS will send a ICMP message to the requesting host and will also update its both the levels accordingly.

If both levels of non-authoritative Name Servers (NS) do not have any information about the requested domain / host; it is referred to the available TLD to find out the registering authority using 'WHOIS' records and forward it to the authority to obtain NS glue records maintained by them. After obtaining the NS records, request is then forwarded directly to one of the NS for the resolution of the request. Associated domain's authoritative NS then resolves and replies back to the requesting host. This process also updates and caches non-authoritative NS Level-1 and Level-2 for further fulfilling of requests of that domain.

5.1 Non-Authoritative DNS Cache DB structure

Fields indicated in both levels of the proposed solution are as under:

i. Fields at Level-1 Caching:

<Domain Name> [FQDN]
<Primary DNS> <Type> [NS]
<Primary DNS IP-address>
<Secondary DNS> <Type> [NS]
<Secondary DNS IP-address>
<Tertiary DNS> <Type> [NS]
<Tertiary DNS IP-address>

ii. Fields at Level-2 Caching:

<Domain Name> [FQDN]
<Host Name> <Type> [A, AAA, MX, CName, PTR]
<Host IP Address>

5.1 Functioning of Proposed Solution

Proposed solution is divided into multiple stages starting from 0 to 7. These stages provide complete functionality of DNS response at different levels. Figure 1 shows the functionality of each stage and following are the descriptions:

- **Stage-0:**

  Host requests its assigned NS to resolve domain / sub-domain and receives a response from the NS. If request is resolved but resolved resource does not respond, the host sends back an ICMP message so that NS updates its level-2 caching database otherwise no ICMP message will be transmitted.

- **Stage-1:**

  Requested NS searches its level-2 cached database and response is sent back to the host. If record is not found in Level-2 database, NS refers it to its level-1 caching database structure.

- **Stage-2:**

  Level-1 database structure of NS is checked and replied back to the requesting host. If not found, the query is forwarded to the root server to get NS glue records of the domain. On the other hand, if authoritative NS record(s) are found in Level-1 cached database, the request is directly forwarded to the authoritative NS of the domain to get the request resolved and replied back to the requested host accordingly.

- **Stage-3:**

  Forwarding root server tracks record of the registering authorities. It checks its database and forwards to the assigning domain registering authority.

- **Stage-4:**

  Domain registering authority checks the status of the domain and sends back NS glue records of the domain to the requesting NS. Requesting NS then resolves the query by requesting Primary, Secondary and Additional authoritative NS one by one.

- **Stage-5:**

  Primary NS resolves the query and replies back to the non-authoritative requesting DNS at stage-2. Level-1 and Level-2 database is updated and host is replied accordingly.

- **Stage-6:**

  If Primary NS fails to resolve, Secondary NS tries to resolve the query and replies back accordingly and simultaneously level-1 and level-2 is updated at stage-2. Requesting host is also responded.

- **Stage-7:**

If both the NS (Primary and Secondary) are failed to response on the request then Additional NS, if available, checks its zone files and responds to the non-authoritative DNS at stage-2. Level-1 and Level-2 database is updated and host is responded back accordingly.

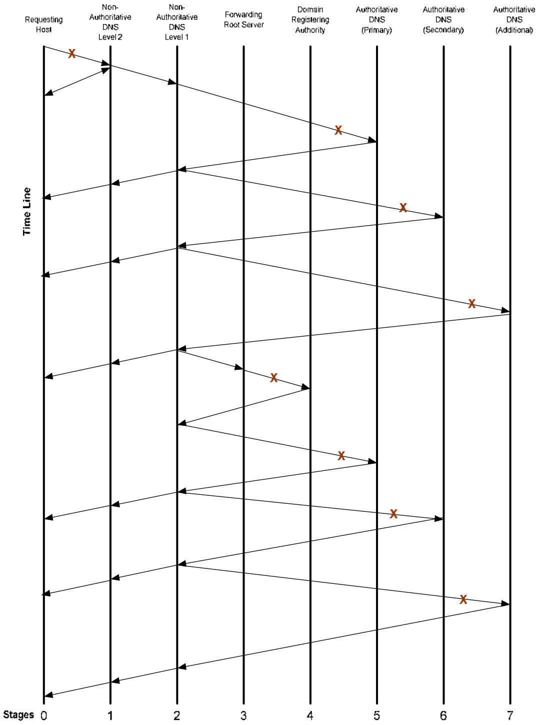
Requesting Host
Non-Authoritative DNS Level 2
Non-Authoritative DNS Level 1
Forwarding Root Server
Domain Registering Authority
Authoritative DNS (Primary)
Authoritative DNS (Secondary)
Authoritative DNS (Additional)
Time Line
Stages
0
1
2
3
4
5
6
7

Figure 1. Timeline Diagram of the Proposed Solution.

6. Conclusion

IPv6 is the next generation Internet protocol. Proposed solution has been designed to provide a mask to understand and remember address very easily. The proposed solution is provided with convention and followed by a rule to create a mask for any IPv6 address. There is a possibility that one day we will be having a 256bit, 512bit or even 1024bit IP address and so on. In that case it would be literally impossible to remember 64, 128 or 256 characters long meaningless IP addresses.

The second approach for DNS resolution process not only reduces the query response time from hours to seconds but also improves security parameters for the existing and newly registered domains. The system also reduces poisoning of DNS entries on zones rather it almost eliminates the effect of poisoning/pollution of DNS.

Domain Administrators require free-hand in changing zone records frequently without having fear of non-availability of network resources in order to maintain security parameters. This has also been facilitated in the proposed solution. In addition to the improvement in performance of the DNS resolution query time, the proposed system also provides measures to avoid attacks on the domain's shared resources.

7. Future Work

Apart from general reduction in the length of an IPv6 address, which is approximately 56% (22/39*100), it is easy to remember by all walks of life. Another benefit that will be taken into consideration is that whenever we need a pool of 4 addresses in each version of IP, subnetting requires which is itself quite a technical job and requires calculations. Under the proposed scheme the last character of base64 address must be among 0 to 3 and ultimately eliminates the need of subnetting for providing broadband connections to household customers.

8. Acknowledgement

My great gratitude and supplications to Almighty Allah, whose guidance and benevolence helped me in all matters of my life including this research paper. Special thanks to Dr. Muhammad Akbar for his technical support and suggestions which facilitated in finalizing the research paper. Thanks to Mr. Malik Faisal Rasool for textual review.

9. References

[1] Y. Law, M. Lai, W. Tan and W. Lau, "Empirical Performance of IPv6 vs. IPv4 under a Dual-Stack Environment", in Proceeding of IEEE International Conference of Communication, 2008, pp. 5924-5929.

[2] Z. Li and X. Deng, "Divide-and-Conquer: A Scheme for IPv6 Address Longest Prefix Matching", in Proceeding of IEEE Computer Society, 2006, pp. 37-42.

[3] R. Hinden and S. Deering, "Internet Protocol Version 6 (IPv6) Addressing Architecture", RFC-3513, April 2003.

[4] S. Cheekiralla and D. Engels, "An IPv6-Based Identification Scheme", in Proceeding of IEEE International Conference of Communication, 2006, pp. 281-286.

[5] S. Deering, R. Hinden, "Internet Protocol, Version 6 (IPv6) Specification", RFC-2460, Dec 1998.

[6] R. Elz, "A Compact Representation of IPv6 Address", RFC-1924, April 1996.

[7] M. Wang, S. Deering, T. Hain and L. Dunn, “Non-random Generator for IPv6 Tables”, in Proceeding of IEEE Computer Society, 2004, pp. 35-40.

[8] Y. Zhang and Z. Li, “IPv6 Conformance Testing: Theory and Practice”, in Proceeding of IEEE Computer Society, 2004, pp. 719-727.

[9] A. J. Gundacker. (2002, Nov. 02). Introduction to DNS (ver 2.34) [Online]. Available: http://www.linuxfocus.org/English/May1998/article45.html, Accessed on Dec. 5, 2009.

[10] J. Jung et al., “DNS Performance and the Effectiveness of Caching”, in IEEE/ACM Transactions on Networking, Oct 2002, vol. 10, pp. 589-603.

[11] Y. Musashi, R. Matsuba, and K. Sugitani, “Detection, Prevention, and Management of Security Incidents in a DNS Server,” in proceeding of the 4th International Conference on Emerging e-learning Technologies and Applications, Kosice, Slovakia, 2005, pp.207-211.

[12] N. Brownlee, K. C. Claffy, and E. Nemeth, “DNS measurements at a root server,” in IEEE Global Telecommunications Conference (GLOBECOM '01), San Antonio, TX, Nov. 25-29, 2001, pp. 1672-1676.

[13] E. Cohen and H. Kaplan “Proactive Caching of DNS Records: Addressing a Performance Bottleneck”, in IEEE proceedings on Applications and the Internet, Jan. 8-12, 2001, pp. 85-94.

[14] D. Romana, Y. Musashi, H. Nagatomi and K. Sugitani “Statistical Study of Unusual DNS Query Traffic”, in International Symposium on Communications and Information Technologies (ISCIT '07), Sydney,. NSW, Oct. 17-19, 2007, pp. 592 - 595.

[15] C. M. Kozierok. (2005, Sep. 20). The TCP/IP Guide (ver 3.0) [Online]. Available: http://www.tcpipguide.com, Accessed on Dec. 24, 2009.