Design and Implementation of Stateless BD-SIIT

1Hanumanthappa J.
1Teacher Fellow, Dept of CS, Mangalagangotri, Mangalore, INDIA
hanums_j@yahoo.com.

2Dr.Manjaiah D.H.
2Associate Professor & Chairman, Mangalore University, Mangalagangotri, Mangalore, INDIA
ylm321@yahoo.co.in

Abstract
IPv6 is a new version of internetworking protocol designed to remove all the shortcomings of the very standard current Internet protocol IPv4. Unfortunately IPv4 and IPv6 are incompatible with each other. So application programming interfaces (API’s) and systems designed to one standard cannot communicate with those designed to the other. IPv4 systems are ubiquitous and are not about to go away “over night” as the IPv6 systems are rolled in. Consequently it is necessary to create smooth transition mechanisms that a transition mechanism is required during the time of migration from IPv4 to IPv6 networks. This paper also aims the design and development of BD-SIIT stateless translator which converts IPv4 to IPv6 and IPv6 to IPv4 etc. In this paper we are showing the design and implementation of a transparent transition mechanism that converts packet headers like IPv4 to IPv6, IPv6 to IPv4, ICMPv4 to ICMPv6, ICMPv6 to ICMPv4 when they cross between IPv4 and IPv6 network’s. Already so many transition mechanisms have been implemented and proposed to illustrates the transition scheme from IPv4 to IPv6 and IPv6 to IPv4.

KEYBOARD: BD-SIIT, IPv4, IPv6, Transition mechanisms.

1.INTRODUCTION
In Recent days transition of IPv6 is in great demand. This has given boost to lot of research in the field of IPv6 Networking. The current Internetworking protocol IPv4 eventually will be unable to adequately support additional nodes or the requirements of the new applications. IPv6 is an advanced version of IPv4 that support improved scalability and routing, security and ease-of-configuration, address space cardinality, high-density mobility, multimedia and real-time applications like Audio and Video etc. IPv6 is an improved version of an IPv4 i.e. designed to coexist with IPv4 and eventually provide better internetworking capabilities than IPv4. IPv6 offers the potential of achieving the scalability, reachability, end-to-end internetworking, Quality of service (QoS), and commercial-grade robustness for data as well as for VoIP, IP-based TV (IPTV), distribution and triple play networks.

2.PREVIOUS WORK.
The Transition from IPv4 to IPv6 plays a vital role in the Transition of IPv6. The IETF has already proposed many IPv6 research transition mechanisms works in an on demand fashion. SIIT is a type of IPv6 transition mechanism which has been implemented by IETF is a stateless transition scenario for the migration of IPv4 to IPv6 protocols. In this paper we proposed a new transition algorithm called SIIT how it works with UDP and TCP protocols. As we know that SIIT (Stateless Internet Protocol/Internet control messaging Protocol Translation SIIT) is an IPv6 transition mechanism that allows IPv6 only hosts to talk to IPv4 only hosts. SIIT is said to be a stateless IP/ICMP translation, which means that the translator is able to process each conversion individually without any reference to previously translated packets.

2.1.IP HEADER CONVERSION.
As we know that SIIT is a stateless IP/ICMP translation. The SIIT translator is able to process each conversion individually, without any reference to previously translated packets. Although most of the IP header field translations are relatively very simple to handle, however one of the issue related with SIIT translator is how to map the IP addresses between IPv4 and IPv6 packets. The SIIT translation process supports the other two additional types of IPv6 addresses.
1. IPv4-mapped address (0:FFFF:v4): This is an IPv6 address simply created by including the IPv4 address of the IPv4 host (v4) with the prefix shown. The SIIT mainly uses this type of address for the conversion of IPv4 host addresses to IPv6 addresses.

2.2. The IPv6 Mapped Address in SIIT Translator.
SIIT resides on an IPv6 host and converts outgoing IPv6 headers into IPv4 headers, and incoming IPv4 headers into IPv6. To perform this operation the IPv6 host must be assigned an IPv4 address either configured on the host or obtained via a network service left unspecified in RFC 2765. When the IPv6 host wants to communicate with an IPv4 host, based on DNS resolution to an IPv4 address, the SIIT algorithm recognizes IPv6 address as an IPv4 mapped-address. A typical IPv4-mapped IPv6 address is shown in Fig-1 which depicts its dynamic architecture.

<table>
<thead>
<tr>
<th>0</th>
<th>79</th>
<th>80</th>
<th>95</th>
<th>96</th>
<th>127</th>
</tr>
</thead>
<tbody>
<tr>
<td>80-Zero bits</td>
<td>FFFF(16 bits)</td>
<td>32 bits(IPv4 Address)</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Fig-1. A Typical IPv4-mapped-IPv6 address.

2. IPv4 translated addresses (0:FFFF:0:v4): According to IETF specification this address is created by IPv4 address temporarily assigned to the IPv6-only host and allows for the mapping of the IPv4-translated address of the IPv6 host to an IPv4 address.

3. IPv6 over IPv4 Dynamic/Automatic tunnel addresses. These addresses are designated as IPv4-Compatible IPv6 addresses and allows the sending of IPv6 traffic over IPv4 networks in a transparent manner. They are represented as: 156.55.23.5.

4. IPv6 over IPv6 Addresses Automatic representation. These addresses allow for IPv4-only nodes to still work in IPv6 networks. They are designated as IPv4-mapped IPv6 addresses and are represented as: FFFF. They are also represented as ::FFFF.156.55.43.3.

The Fig-2 shows a typical protocol stack of SIIT and Fig-3 signifies the network based SIIT translation process.

<table>
<thead>
<tr>
<th>IPv6 applications</th>
<th>IPv4 applications</th>
</tr>
</thead>
<tbody>
<tr>
<td>Sockets API</td>
<td>TCP/UDP v4</td>
</tr>
<tr>
<td>TCP/UDP v6</td>
<td>SIIT</td>
</tr>
<tr>
<td>IPv4</td>
<td>IPv6</td>
</tr>
<tr>
<td>Datalink</td>
<td>Physical</td>
</tr>
</tbody>
</table>

Fig 2. Typical protocol stack view of SIIT.

![Fig 3. Network Based BD-SIIT Translation Process.](image)

Translations are broadly divided into two types like Network Address and Protocol Translation. The address and protocol translation presented in this section enables both the communication between nodes in an IPv4 site with nodes in the IPv6 network and between nodes in an IPv6 site with nodes in an IPv4 nodes. Figure-4 and Figure-5 illustrate these scenarios and the following paragraphs describe them in more detail. Figure-1 illustrates a BD-SIIT translator for an IPv6 site communicating with nodes in an IPv4 network. The internal routing of the IPv6 site must be configured such that packets intended for IPv4 nodes route to the translator. Hosts in the IPv6 site send packets to nodes in the IPv4 network using IPv6 addresses that map to individual IPv4 hosts. For this scenario a design presented in [4] proposes that IPv6 nodes use an IPv4-compatible IPv6 address as their own address and an IPv4-mapped IPv6 address when communicating with IPv4-only nodes.

3.2. Working methodology of BD-SIIT Translator.
Address translation is trivial when using IPv4-mapped and IPv4-compatible IPv6 addresses. For the IPv6-to-IPv4 direction the translator simply extracts the lower 32-bits of an IPv6 address to obtain an IPv4 address. For the opposite direction the translator sets the lower 32-bits of the IPv6 source/destination addresses to the IPv4 source/destination addresses and sets the upper 96-bits of the IPv4 source and destination addresses to the IPv4-mapped and IPv4-compatible prefix, respectively. However it is considered to be a very bad idea to use IPv4-mapped address as it has the drawback of requiring IPv6 routers to contain routes to IPv4-mapped addresses [4]. The alternative is to use IPv6-only addresses to refer to IPv4 nodes, which requires the translator to maintain an explicit mapping between IPv4 and IPv6 addresses. For clarity, we introduce an IPhNODEj notation to disambiguate among the types of addresses used in the translation process. Table-1 defines the four types of addresses in terms of this notation. The first two rows define the addresses that are native to the IPv4 and IPv6 nodes. The last two rows define address aliases which are assigned by the translator, used to translate between the IPv4 and IPv6 address domains. An example of using this IPhNODEj notation consider the following scenario: an arbitrary IPv4-only host wishes to communicate with our IPv4-only web server via the translator. For an IPv6 host to communicate with our IPv4 web server requires an IPv6 address that is an alias (IP6NODE4) address for the web server’s native IPv4 host (IP4NODE4) address. Similarly, for the web server to reply to the IPv6 host requires an IPv4 address that is an alias (IP4NODE6) address for the IPv6 host’s native (IP6NODE6) address. That is the translator maps the IP6NODE4 address to the IP4NODE4 address of the web server, and the IP4NODE6 address to the IP6NODE6 address of the IPv6 host. The translation of addresses has three phases: address binding, address lookup and translation and address unbinding which we describe in the following subsections.

Table 1: Four various types of nodes used in IPv6 Transition mechanism.

<table>
<thead>
<tr>
<th>IPhNODEj</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>1.IP4NODE4</td>
<td>v4 address of a 4 node.</td>
</tr>
<tr>
<td>2.IP6NODE6</td>
<td>v6 address of a 6 node.</td>
</tr>
<tr>
<td>3.IP6NODE4</td>
<td>v6 address referring to the v4 node.</td>
</tr>
<tr>
<td>4.IP4NODE6</td>
<td>v4 address referring to the v6 node.</td>
</tr>
</tbody>
</table>

Fig.4.BD-SIIT Translator in IPv6 zone.  Fig.5.BD-SIIT translator in IPv4 Network.
3.3 BD-SIIT Address Binding.
Address binding is the phase where an IPv4 address is associated with an IPv6 address and vice versa. The translator maintains key-to-value tuples listed in Table-2 to map between IPv4 and IPv6 addresses.

Table 2: Address mapping between IPv4 and IPv6 addresses used by translation process.

<table>
<thead>
<tr>
<th>Host-to-Value</th>
<th>Definition</th>
</tr>
</thead>
<tbody>
<tr>
<td>IP6NODE4-to-IP4NODE4</td>
<td>v6 addresses mapped to v4 node addresses.</td>
</tr>
<tr>
<td>IP4NODE6-to-IP6NODE6</td>
<td>v4 addresses mapped to v6 node addresses.</td>
</tr>
</tbody>
</table>

For addresses that are statically mapped, the binding happens when the translator is initialized. If the translator is configured to use IPv4 mapped/compatible IPv6 addresses then all the bindings are implicitly static as they are defined by these special IPv6 addresses. Other static mappings could be setup between arbitrary IPv4 and IPv6 addresses. Ex the binding of addresses for an IPv4 node to an IPv6 node could be done statically by a network manager when assigning IPv6 addresses to existing nodes in the IPv4 site. i.e IP6NODE4-to-IP4NODE4 are static mappings of IPv6 addresses assigned to IPv4 hosts. Otherwise, the binding between addresses needs to happen dynamically. IPv6 addresses are larger than IPv4 addresses and it is not possible to create a one-to-one IP4NODE6-to-IP6NODE6 binding. Consequently, it will be necessary to reuse IP4NODE6 addresses to bind them to other IP6NODE6 addresses.

3.4 Address mapping Lookup Table and Translation in BD-SIIT.
Once a binding is established it can be used for address lookup and translation. The example in Figure 7 illustrates the translation using the IPhNODEj notation defined earlier. When the IPv4 node sends a packet to the IPv6 node it is routed through the translator. The translator receives the packet, translates the 195.18.231.17 to 1C::DACF source address using the IP4NODE4-to-IP6NODE4 mapping and translates the 12.23.200.23 to ABC2::4321 destination address using the IP4NODE6-to-IP6NODE6 mapping. Likewise, IP packets on the return path go through a reverse address translation. Notice that this requires no changes to hosts or routers. As far as the IPv4 host is concerned, IP4NODE6=195.18.231.17 is the address used by the IPv6 hosts. Conversely, the IPv6 host believes that IP6NODE4=ABC2::4321 is the address used by the IPv4 hosts. The address translation is transparent to both hosts. In BD-SIIT we can use two different types of Address mapping techniques.

Case-1: Direct address mapping from IPv4 to IPv4 and IPv6 to IPv6

![Fig.7.BASIC ADDRESS MAPPING TRANSLATION OPERATION IN BD-SIIT.](image)

Case-2: Conversion of IPv4 header to IPv6 Header.
The IPv6 and IPv4 headers have some similarity, but there are a number of fields that are either missing or have different sizes or meaning. The translator either directly copies, translates, ignores, or sets fields in the IP header to a default value when translating from one version of IP to the other. Figures 3 illustrates which fields of IPv6/IPv4 headers are directly copied, requires translation, or ignored. Many of the fields require a simple adjustment. The IPv4 checksum field is computed when translating from IPv6-to-IPv4 and ignored when translating from IPv4-to-IPv6. The IPv4 total-length field includes the IPv4 header size whereas the IPv6 payload-length field does not. The translation needs to account for this difference. The hop-limit/time-to-live fields are copied and decreased by one. Finally, the protocol field can be directly copied from one version of IP to the other. With ICMPv4 and ICMPv6 protocol numbers being the only exception. The IPv4 tos and IPv6 traffic class are also...
discarded by the translator because there is no mapping exists between them. The conversion from IPv4 to IPv6 increases the packet size by another extra 20 bytes because of header length differences between the two IPv4 and IPv6 protocols.

Scenario-1: [Conversion from IPv4 to IPv6 Headers]
Case-1: Mapping the contents of DS (Differentiated Services) to Traffic Class (TC) or Priority.

![Fig.8. Mapping IPv4 Differentiated Services to Traffic Class (TC).]

Case-2: Flow chart of IPv4 to IPv6 Header Translation.

![Fig.9. Flowchart of IPv4 to IPv6 Header Translation.]

Scenario-2: Conversion of IPv6 to IPv4 Header.
Case-1: Mapping the values of TC from IPv6 to DS of IPv4.

![IPv6 TC to IPv4 DS Flowchart.]

Case-2: The IPv6 to IPv4 header conversion Flowchart.
4. BD-SIIT Address mapping algorithms

The address mapping algorithms are broadly divided into two types.
1. Forward Address mapping algorithm.
2. Feedback Address mapping algorithm.

Case-1: Forward address mapping algorithm
Address unbinding is the phase when the association between an IPv4 and IPv6 address is broken. We expect the number of bindings of the IP6NODE4-to-IP4NODE4 mapping to remain fairly constant during the day-by-day operation of the translator new bindings are only necessary when adding new hosts to the site. On the other hand, the number of bindings of the IP4NODE6-to-IP6NODE6 mapping are more dynamic and depend on the number of connections established to different hosts in the network. The number of reserved IP4NODE6 addresses used by the translator limits the number of bindings possible for the IP4NODE6-to-IP6NODE6 mappings. For the scenario where the translator is providing service for an IPv6 site (as illustrated in Fig.1) the IP4NODE6 addresses are a small number of unique IPv4 addresses. It is crucial for the translator to detect when an IP4NODE6 address can be reused in order to create new bindings otherwise, new sessions may be refused if there are no IP4NODE6 addresses available. For the scenario where a translator is providing service to an IPv4 site (as illustrated in Fig.4) the IP4NODE6 addresses may come from a relatively large pool of private network addresses (as mentioned earlier, there are roughly 17 million of such addresses available). Here the concern is to safely remove unused bindings to ensure that the mapping table does not require too much memory and that address lookup performance does not deteriorate. Removing a binding too early should never occur, as it would effectively terminate any ongoing communication that relied on the binding.

5. CONCLUSIONS.

In this paper an attempt has been made to design, implement IPv6/IPv4 Network and Protocol BD-SIIT translator. Our work subsumes both the stateless BD-SIIT translator design and stateful NAT-PT Translator design and implementation. In this paper we have designed and implemented the BD-SIIT stateless translator and briefly compared pros and cons of stateless with stateful translation.

ACKNOWLEDGEMENT
The authors would like to thank UGC for providing funding for this project with an entitled “Design Tool of IPv6 Mobility for 4G-Networks”, under eleventh plan of Major Research Project scheme (Ref.No F.No 36 - 167/2008(SR) dated 26.03.2009). Thanks also goes to the dedicated research group in the area of Computer Networking at the Dept of Computer Science, Mangalore University, Mangalore, India for many stimulating discussions for further improvement and future aspects of the project. Lastly but not least the author would like to thank every one, including the anonymous reviewers.

AUTHOR BIOGRAPHIES.

Hanumanthappa J. received his Bachelor of Engineering degree in Computer Science and Engineering from University B.D.T.College of Engineering, Davanagere Karnataka(S), India(C), from the Kuvempu University, at Shankarghatta, Shimoga in 1998 and his Master of Technology in CS & Engineering from the NITK Surathkal, Karnataka(S), India(C) in 2003 and Currently he is pursuing his PhD in Computer Science and Engineering from Mangalore University under the supervision of Dr. Manjaiah D. H on entitled “Investigations into the Design, Performance and Evaluation of a Novel IPv4/IPv6 Transition Scenarios in 4G advanced wireless networks”. Presently he is working as an Asst. Professor at DoS in Computer Science, Manasagangotri at Mysore. He has authored more than 40 research papers in international conferences and National Conferences and reputed Journals.

Dr. MANJAIAH D. H. is currently Professor and Chairman of the Dept. of Computer Science, Mangalore University, and Mangalore. He is also the BoE and BoS Member of all Universities of Karnataka and other reputed universities in India. He received PhD degree from University of Mangalore, M. Tech. from NITK, Surathkal and B.E. from Mysore University. Dr. Manjaiah D. H has more than 15-years extensive academic, Industry and Research experience. He has worked at many technical bodies like CSI [AM IND 00002429], ISTE [LM-24985], ACS, IAENG, WASET, IACSIT and ISOC.

He has authored more than -50 research papers in International / National reputed journals and conferences. He is the recipient of the several talks for his area of interest in many public occasions. He had written Kannada text book, with an entitled, “COMPUTER PARICHAYA”, for the benefits of all teaching and Students Community of Karnataka. Dr. Manjaiah D.H ‘s areas interest are Advanced Computer Networking, Mobile/Wireless Communication, Wireless Sensor Networks, Operations Research, E-commerce, Internet Technology and Web Programming.

He is the expert committee member of various technical bodies like AICTE, various technical Institutions and Universities in INDIA. He had been actively involving in chairing technical sessions of various International & National Conference and reviewer of the Journals.

He is working with Major Research project on “Design Tool for IPv6 Mobility for 4G – Networks ”, around Rs. 12 lakhs worth funded by UGC, New Delhi from year 2009 -2012.

He is recognized as a Ph.D. guide in Computer Science at Mangalore University, Mangalore and currently five students are doing their Ph.D., under the guidance of him. He is recognized as advisory editorial board member of the International Journal of Advanced Computing [IJAC], International Journal of Computer Science and Application [IJCSA], and Journal of Intelligent System Research and Journal of Computing. He visited most of the countries in the World.