CHAPTER 2

LITERATURE SURVEY

The current version of the Internet Protocol IPv4 was first developed in the 1970s (Tanenbaum 2002), and the main protocol standard RFC 791 that governs IPv4 functionality was published in 1981. The following statistic shows how quickly the address space has been getting consumed over the years after 1981, when IPv4 protocol was published. In 1985, 1/16 of total address space, in 1990, 1/8 of total address space, in 1995, 1/4 of total address space, in 2000 half of total address space, in 2002, 2/3 of total address space and in 2008 almost 90% of the total address space is utilized. (www.IPv6.com)

A brief recap of the major events in the development of the new protocol is given below:

- Basic socket API (RFC 2553) and DHCPv6 (RFC 3315) published in 2003.
- Mobile IPv6 (RFC 3775) published in 2004
- Flow label specifications (RFC 3697) added in 2004
- Address architecture (RFC 4291) stable, minor revision in 2006
- Node requirements (RFC 4294) published in 2006
As migration from IPv4 to IPv6 cannot take place all of a sudden (Forouzan 2003), we rely on address translation mechanisms which facilitate communication between IPv4/IPv6 networks. As defined in RFC 2765, all existing translation mechanisms address the following two types of translation:

- IPv4 to IPv6 translation
- IPv6 to IPv4 translation

When an IPv4-only node sends a packet to another node, whose destination is an IPv6 address in another network, the stateless IP/ICMP translation algorithm (SIIT) can be used (Nordmark 2000). The SIIT algorithm translates the IPv4 header of the packet to an IPv6 header, and discards the IPv4 header. Subsequently the packet is forwarded to the destination IPv6 address. Several translation mechanisms are based on the SIIT algorithm, (Stateless IP/ICMP Translation algorithm). The SIIT algorithm is used as a basis of the BIS (Bump in the Stack) (Tsuchiya et al. 2000).

An IPv4 Network Address Translator (NAT) was proposed by Srisuresh and Egevang (1994), which converts private internal addresses to globally unique addresses that are passed to the Internet backbone and vice versa. The IPv4 NAT has the main limitation that it is stateful in order to map between the globally unique and private internal addresses. A proposal for tunneling IPv6 with private IPv4 addresses through NAT devices has been given by Liu et al (2004).

Network Address Translation and Protocol Translation (NAT-PT) (Liu et al 2004) is another transition mechanism which provides transparent routing between IPv4 and IPv6 end nodes. (Tsirtsis and Srisuresh 2000).
NAT-PT should be avoided when translation occurs between IPv6 and the IPv4 part of a dual stack node. A tunneling mechanism should rather be chosen as an alternative. There are two existing ways to accomplish tunneling: The first one is automatic tunneling that uses IPv4-compatible IPv6 addresses to add a route to a special IPv6 prefix which points to a tunnel destination. Any packets destined for a v4-compatible address will be sent through the tunnel. Another way is configured tunneling, where the address of the tunnel exit point is configured on the tunnel entry point and similar encapsulation is used (Gilligan and Nordmark (Aug 2000)). A combination of automatic and configured tunneling also can be used to route IPv6 packets across a v4 network. Teredo (Huitema 2005), ISATAP (Templin et al 2004) 6 to 4 and IPv6 over IPv4 (Carpenter et al 2001) are other tunneling mechanisms. Tunnel Broker with the Tunnel Setup Protocol (TSP) has been discussed in (Blanchet and Parent 2005)


Owing to rapid growth of traffic in the Internet, backbone links of several Gigabit/sec are commonly deployed. To handle Gigabit/sec traffic rates, the backbone routers must be able to forward millions of packets per second on each of their ports. Recently, various algorithms for high performance IP address lookup have been proposed (Ergun et al 2001), (Gupta et al 2000), (Jean et al 2002), (Kim 2004).

The primary role of routers is to forward packets towards their final destination. Several fast routing lookup mechanisms have been proposed (Tzeng 1995), (Zukowski and Pe 1992), (McAulley and Francis 1993), (Degermark et al 1997), (Waldvogel et al 1997), (Lampson et al 1998), (Huang and Zhao 1999), (Gupta 1998), (Srinivasan and Varghese 1998), (S. Nilsson and Karlsson 1999), (Doeringer et al 1996) in the past few years,
including both software and hardware solutions. For example, Tzeng and Przygienda (1999) proposed a software-based lookup scheme, i.e., a multiresolution trie that can achieve 2 lookups. Huang and Zhao, (1999) proposed a novel IP routing lookup scheme and its hardware architecture. The lookup scheme introduces a small lookup table with about 450-470 KB. One to three memory accesses are needed in the scheme after using some compression techniques for the lookup table based on Huang and Zhao’s scheme. Wang et al (2001) presented an improved lookup scheme that reduces the memory access to a slightly larger memory size of 565. Jean et al (2002) also proposed an improved scheme with memory accesses. Although the memory sizes are relatively small in these schemes, the whole forwarding tables have to be rebuilt periodically and these schemes cannot be easily scaled to IPv6 with such compact tables.

IP address lookup has been an active area of research in the recent past. Several innovative solutions (Fledmann et al 2000), (Attia et al 2001), (Bruijn et al 2006), (Chandramenan et al 1996), (Sreedhar and Varghese, 1985), (Gupta et al 2000),(Dharmapurikar et al 2006) have been proposed in the literature for faster IP lookup algorithms. Most of these approaches (Pei et al 1991) organize the routing database of address prefixes in an efficient manner so as to reduce the required memory accesses for faster search times.

In the early days of the Internet, the following four issues were considered as invariants. An address received was the address sent, Addresses were stationary (non-mobile), Source and destination were reversible, and all hosts knew to which address they should send packets to reach the wanted host. (Nikander 2001)

According to Henderson, these assumptions cause fundamental problems in the network layer (Deering Hinden 1998). The first issue concerns addressing. As IP routing and addressing are hierarchically defined
for scalability, the mobile hosts usually have a topologically incorrect interface address when they attach to a new network. Secondly, when changing network, the mobile host may become unreachable to the rest of the network unless the new address is somehow mediated to other nodes (address translation). IP Mobility Support in v4 was proposed in (Charles Perkins, 1996) and MIPv6 was proposed in (Johnson, Perkins, 2004) and Mobile IPv6 security analysis was explained in (Aura 2002).

Route Optimization in Mobile IPv6 Security Design was proposed (Nikander et al 2003). When moving to a new subnet, the MN will discover the default router, perform (stateful or stateless) address auto-configuration, and use its new address as CoA (Thomas, Narten 1998), (Bound et al 2002). Moskowitz in 2001 proposed Host Identity Payload and Protocol, having a simple and fast solution to the key distribution system, which is one of the biggest issues with MIPv6.

An attempt has been made to study and compare the protocols IPv4, HIP and IPv6 in this work.