Security Considerations in Distributed Systems using IPv6 and ATM

David M. LaMacchia

April 11, 1995

1 Introduction

As the Internet continues to grow, great demand has been placed on the existing protocols that allow users to communicate. Expected development of the Internet questions the efficiency of these protocols; a desire for low-level secure transactions makes current protocols obsolete. Research has led to two suggestions for improvements to the existing substructure: Internet Protocol, version 6 (IPv6 or IPng) and Asynchronous Transfer Mode (ATM) networks.

While the original motive for IPv6 was to increase the address space for Internet hosts, encryption and authentication are now of great importance to the new protocol. This paper discusses the security issues relevant to their implementation in IPv6 and how this relates to IPv6 over ATM networks. The viability for these modifications in distributed systems at a user-level is discussed and shown to be a great improvement over current Internet protocol, although some security problems still need to be overcome.[11]

2 IPv6 Security Architecture

IPv6 offers substantial improvements over the IPv4, the current Internet Protocol version that is in wide use. IPv6 expands IPv4's addressing capabilities (address are now 128 bits instead of 32 bits), drops some of the IPv4 header fields to compensate for the longer addresses, and supports "flows" of traffic to better handle real-time applications. In addition, IPv6 has optional encryption and authentication of datagrams.[12]

The IPv6 security architecture is composed of two main layers: an unencrypted authentication header and an encrypted Encapsulated Security

Next Payload	Length	Reserved
Security Association Identifier (SAID)		
Authentication Data		
(more Authentication Data)		

Figure 1: The structure of the IPv6 Authentication Header
Next Payload = payload type after authentication payload
Length = length of authentication data in 64-bit double words
Authentication Data = data in 64-bit double words

Payload (ESP). These layers may be used independently or in conjunction with each other. There are some concepts that are fundamental to both the authentication header and ESP.

One of these is a "Security Association," the combination of a given Security Association ID (SAID) and a Destination Address (which may be a multicast group address).[3] The SAID is a pseudo-random value. The Security Association itself contains these values as well as optional ones. For the authentication header, these include authentication algorithm and mode as well as block size. For the ESP, encryption algorithm, keys, and presence/absence and size of a cryptographic synchronization vector field for the encryption algorithm are all in the SAID.[2] The Security Association represents one-way traffic; for two-way communication two Security Associations are used, one for each direction (although, as mentioned above, one Security Association may be used for a multicast group). The sender uses the originator and destination to select an appropriate Security Association and SAID. The receiving host is able to devise the Security Association using the originator's address and SAID value.

2.1 Authentication Header

The structure of the authentication header looks like:

The authentication header is the means IPv6 uses to add security through authentication to datagrams.[4] This header is individually computed for packets prior to sending. However, the header field itself can be easily ignored by existing IPv4 systems. In IPv6 systems, the authentication header field is produced using the non-transient fields in the header block of each datagram (those which do not change hop-to-hop). The calculation of this header and its processing time to marginally increase the latency of sending a datagram as opposed to IPv4. Of course, this latency (and the security of the authentication header itself) varies with respect to the authentication algorithm used, as does non-repudiation, the ability for a receiver to prove the sender of a packet actually sent it, although they may claim they did not. At the time of this writing, keyed MD5 is the default authentication algorithm for IPv6 and does not support non-repudiation, as it is a symmetric authentication algorithm.[6]

Authentication via the IPv6 authentication header is worthwhile for a number of reasons. It puts the burden of authentication in the hands of the sender and is as strong (or a weak) as the authentication algorithm used. Although MD5 with 128-bit keys is the least required authentication algorithm, other non-symmetric ones that allow for non-repudiation can be easily used.

2.2 Encapsulating Security Payload

The Encapsulating Security Payload (ESP) is a mechanism that provides encryption of individual packet information at the IPv6 level or for a transport-layer segment (TCP, UDP, etc).[5] While this is a means for confidentiality, ESP can also provide authenticity as well, depending on the encryption algorithm. Like the authentication header, the latency and degree of security are directly related to the algorithm chosen.

ESP works by first encrypting either the entire datagram or the transport-layer segment, and then prefixing a new header before the encrypted data but after the IPv6 header. It is interesting to note that IPv6 headers as well as data can be encrypted this way; the recipient discards the plain-text headers and only pays attention to the decrypted ones within the ESP data block.

Figure 2 displays a detailed ESP header. The opaque transform data is left unspecified, as the algorithm and algorithm mode used in encrypting the data block are included in the SAID. While the default IPv6 encryption algorithm is the Data Encryption Standard with Cipher Block Chaining (DES-CBC) transform, the option has been left for future use of other algorithms and algorithm modes.[3]

As mentioned, there are two distinct ESP modes: IP tunnel-mode, and transport-mode.[4] The main differences are in how each handles head-

Security Association Identifier (SAID), 32 bits		
Algorithm-dependent Initialization Vector (length variable)		
Next Header (8 bits)	Length (8 bits)	Reserved (16 bits)
Encrypted Data		Algorithm-dependent Trailer

Figure 2: A Typical ESP Header

ers; otherwise operation is remarkably similar. In IP tunnel-mode, each IP packet is first encapsulated in ESP at which time a Security Association is calculated based on destination and a key extracted via whatever key management protocol the host is using. For a host with a host-oriented keyring, the same Security Association should be used for all users for the same destination host (as one might expect). Once encapsulated, the resulting encrypted packet is prefixed with clear-text headers and sent to the destination; when received these clear-text headers are, unlike in transport mode, discarded. The combination of the destination address and SAID value are enough to determine the decryption algorithm and to obtain the original headers and data from the packet. Transport-mode differs in that instead of the IP packet the sender takes the original UDP, TCP, or ICMP frame and encapsulates it in ESP. When it arrives at the destination address, the clear-text headers are temporarily stored, and, after decryption, the clear-text IP header is used in conjunction with the now decrypted UDP or TCP header to determine the application to send the data to.[4][10]

The real problems with ESP currently are two-fold.[5] First of all, no clear key management protocol has been established: this is discussed below in the *Key Management* section. Another concern, one that has no clear solution, are the legal problems surrounding encryption on an international level. Certain countries (e.g. the United States) forbid export of certain forms of encryption, including DES-CBC, the proposed standard for IPv6. Other countries (e.g. France) forbid its citizens to use strong encryption. Regulating the use of encryption at the IP level may be difficult, if not impossible, to accomplish.

2.3 Key Management

Perhaps the most difficult element of the IPv6 protocol (and IPv4 with regards to ESP) that has yet to be determined is a method for encryption key management. One problem, of course, is tying the mechanism too closely to the IP headers; not only because of a possible desire to update the key management protocol in the future, but because of legal problems as well involving export or banning of encryption algorithms. The current plan is to place the key management mechanism in an upper-layer protocol such as UDP or TCP on a widely-known port number.[3].

Two main methods of key management exist, each with distinct advantages and disadvantages. Manual key distribution requires a host to be manually configured with its own key and with the keys of hosts it will be communicating with. While this is by far the simplest mechanism, it has problems scaling beyond a local environment. A router, however, could be configured with only a few keys so it can protect data flowing through it. Users behind firewalls might also be interested in a manual key distribution system.[1][3]

Automated key distribution is by far the more scalable and better option; unfortunately it is difficult to agree on which mechanism is appropriate. The two most viable options are host-to-host keying and user-to-user keying. The first approach has users of a particular machine all using the same key to contact the same destination host. The second is more fine-grain, as users each have their own key for unique sessions with a destination user. It would seem that user-to-user keying is the more appropriate choice for an automated distribution system, since users in a host-to-host system can acquire the global key and capture traffic.

One possible solution to the automated key management problem involves DNS key distribution; this idea is suggested by the Integrated Routing & Addressing model, and is discussed below in *IP over ATM*.

2.4 Concerns

There are many concerns about the security measures for the proposed IPv6 protocol. Besides questions regarding use or disuse by the Internet community at large, of primary importance is determining how to interface these security protocols with the existing Internet topology. As mentioned, any encryption mechanism must be implemented at the user-level not only because of the obvious legal implications but because encryption may unnecessarily increase latency of packet transmissions. Furthermore, it should be noted that encryption and authentication are still not enough to protect against traffic analysis attacks; malicious users can still determine packet routing. Firewalls are another concern; the inclusion of "dummy" IPv6 headers for encrypted packets may prove to be problematic for some hosts with regard to network-level packet translators.[7] In addition, these security provisions cannot prevent loss due to denial of service attacks. Since handling each packet has a definite cost (which is increased when more complicated algorithms are used for encryption), another means for a user to attack a system might be to overload the recipient with cost-intensive packets.

3 IP over ATM

Although networks can implement IPv6 on its own, research has already begun to examine how IPv6 can be implemented on top of ATM subnets as both become more viable. There are some security issues to take into consideration on both the network and application levels.[14]

As the future Internet will most likely be a mix of conventional and advanced link technologies, it is only natural to examine the relation of IPv6 to these and examine how its security measures such as authentication and encryption are effected.[9] The serviced-based ATM technology, allowing virtual channels to provide a means for transmission of interleaved data over a single wire, has been perhaps one of the most evident of these technologies.[16]

Several parameters much be established to map IPv6 services to the lower level of ATM services.[8] ATM signaling is handled by addressing parameter, quality-of-service parameters, and connection management information. In addition, a connection identifier is able to map IPv6 packets to ATM virtual circuits.

Because several packets share the same virtual channel, there is a great possibility for a violation of privacy; it is possible, for example, for a single wire to compromise the security of many users at once (by capturing and reconstructing these packets). IPv6 built on top of and ATM network could hamper this activity through use of the ESP and authentication data. This does not, however, impede traffic analysis attacks.

One new model for use of IP over ATM networks is the Integrated Routing & Addressing (IRA) model; this model takes into account some security considerations similar to IPv6 and lends a solution to the key management problem. The basic principle of IRA is that there are problems with a simple IP overlay on and ATM network: routing two virtual channels requires routing translation through an intermediate node, doubling network traffic due to redundant operations. To solve this IRA borrows from other developing protocols: it integrates IP and ATM addressing as well as routing. IRA can use a modified version of the above IPv6 protocol. In this case, it is possible increase the security of IRA by using SAIDs for the ATM layer as well as the IPv6 layer. DNS can provide the the required keys for both layers.[13] DNS may, actually, hold the answer to automated key management in IPv6, solving the problems of IPv6 and IRA at once.[15] Signed host keys in DNS could enable an authentic communications channel by allowing the sending host to authenticate key management messages via DNS keys and an asymmetric algorithm.[3]

4 Usage in Distributed Systems

The security mechanisms discussed so far have many uses in distributed systems; it is these uses that make their implementation worthwhile. Use of these mechanisms in different applications underlines this fact as well as giving users more information about decreasing security risks in the future.

One such use, as was briefly discussed in relation to IPv6, was the existence of firewalls. Since firewalls likely to still be used in the near future[3], both IPv6 and ATM networks should be able to exist in their presence, and not hinder performance across them. One suggested idea is to have firewalls use the IPv6 authentication header for access control (or, in the case of ATM networks, and authentication header provided by higher-level IP). Similarly, it has previously been difficult to have two sites protected by firewalls to communicate easily. Selective encryption of Internet traffic using ESP between these sites could not only protect both hosts but allow each to communicate freely with untrusted sites without fear of data trapping.

In networks where data classified with multiple levels of security are transmitted the security measures provided by ESP and the authentication proposed for IPv6 are ideal. These multi-level secure (MLS) systems require strict access control lists which could easily be incorporated into the authentication header of IPv6 packets. Furthermore, the entire packet can easily be deemed authentic. In old IPv4 systems, the clearance level information was passed in clear-text over the network; IPv6 systems could encrypt this header and patch the packet with plain-text routing headers. This is a great improvement over the old version of IP. Furthermore, data of different security levels could be encapsulated in ESP and combined with these authentication measures: only recipients of the appropriate security level would be able to obtain the correct decryption key.

ATM networks with IPv6 implemented on top of them would also find great benefits. One of the advantages of ATM is that it easily supports high-speed data as well as low-speed. A system that requires high-speed data transfer, such as the Multicast Backbone (MBONE) would benefit from an ATM network. Because the SAIDs used in IPv6 are receiver-oriented, they are optimal for use on the MBONE.[3] Thus the MBONE could benefit from not only a high-speed network, required for the large amount of multicast video and audio data, but authentication and encryption security measures could be provided as well in a system that combined IPv6 and ATM networks.

In addition, these security measures would be excellent for user-level communication applications. At the present time, many applications pass large amounts of data over the network in plain-text through untrusted systems to untrusted users. Some applications try to supersede this problem by using application-level authentication tools and encryption packages. The security mechanisms in IPv6 could easily provide a means for programs sending compromised data to protect that information. Of course, these mechanisms are only as strong as the algorithm used; the user may wish to add additional strong encryption (such as PGP) to the transparent encryption provided in ESP.

5 Conclusion

Both IPv6 and ATM are protocols that will define the future of the Internet. Unlike ATM, however, IPv6 is a *necessary* upgrade, for eventually the Internet will run out of IP addresses under the current scheme. The security mechanisms present in IPv6 (and IPv6 overlaid on ATM) are superior to the IPv4 ones: encryption and authentication become easily available to end-users yet remain transparent. A correct implementation would prevent such security holes as IP address spoofing and data snooping.

While these security provisions are a vast improvement over existing ones, there are still some problems. There is no mechanism to prevent traffic-analysis attacks, and some user-level applications should probably still use separate encryption schemes instead of ESP's default algorithm. Denail of service attacks would still be possible. Furthermore, the legal questions surrounding the international transfer of encrypted data need to be answered.

Even with these cautions in mind, the security mechanisms in IPv6 and IPv6-ATM networks solve many problems currently experienced by the Internet.

References

[1] Aziz, Ashar, "Simple Key-Management For Internet Protocols-Plus (SKIPP)." Internet Draft, March 23, 1995.

[2] Atkinson, R., "Security Architecture for the Internet Protocol." Internet Draft, March 23, 1995.

[3] Atkinson, R., "IPv6 Security Architecture." Internet Draft, February 16, 1995.

[4] Atkinson, R., "IPv6 Encapsulating Security Payload (ESP)." Internet Draft, February 16, 1995.

[5] Atkinson, R., "IP Encapsulating Security Payload (ESP)." Internet Draft, March 23, 1995.

[6] Atkinson, R., "IPv6 Authentication Header." Internet Draft, February 16, 1995.

[7] Bellovin, S., "Security Concerns for IPng." Request for Comments: 1675, August 1994.

[8] Brazdziunas, C., "IPng Support for ATM Services." Request for Comments: 1680, August 1994.

[9] Cole, R., and Shur, D., "IP over ATM: A Framework Document." Internet Draft, February 17, 1995.

[10] Conta, A., and Deering, S., "ICMP for the Internet Protocol Version 6 (IPv6)." Internet Draft, February 1995.

[11] Curran, J., "Market Viability as a IPng Criteria." Request for Comments: 1669, August 1994.

[12] Deering, S., and Hinden, R., "Internet Protocol, Version 6 (IPv6) Specification." Internet Draft, March 17, 1995.

[13] Fink, B., "IP/ATM Integrated Routing & Addressing (IRA) Model." Internet Draft, March 1995.

[14] Kim, B., and Wang, P., "ATM Network: Goals and Challenges." From Communications of the ACM, Vol. 38, No. 2, February 1995.

[15] Thomson, S., and Huitema, C., "DNS Extensions to support IP version 6." Internet Draft, March 24, 1995.

[16] Vetter, R., "ATM Concepts, Architectures, and Protocols." From Communications of the ACM, Vol. 38, No. 2, February 1995.