IPv6 Security Architecture

STATUS OF THIS MEMO
    This document is an Internet Draft. Internet Drafts are working
   documents of the Internet Engineering Task Force (IETF), its Areas, and
   its working groups. Note that other groups may also distribute working
   documents as Internet Drafts.

    Internet Drafts are draft documents valid for a maximum of 6 months.
   Internet Drafts may be updated, replaced, or obsoleted by other
   documents at any time. It is not appropriate to use Internet Drafts as
   reference material or to cite them other than as "work in progress".

    This particular Internet Draft is a product of the IETF's IPng
   working group. It is intended that a future version of this draft be
   submitted to the IESG for publication as a standards-track RFC.
   Discussion of this draft normally takes place on the IPng Working
   Group mailing list: ipng@sunroof.eng.sun.com
   To add/drop from that mailing list, send an email request to:
           ipng-request@sunroof.eng.sun.com

1. INTRODUCTION

    The Internet community is making a transition from version 4 of the
   Internet Protocol (IPv4) to version 6 of the Internet Protocol (IPv6).
   [Hi94] This memo describes the security mechanisms integrated into
   version 6 of the Internet Protocol (IPv6) and the services that they
   provide. Each security mechanism is specified in a separate document.
   It also describes key management for the IPv6 security mechanisms.

1.1 Definitions

   This section provides a few basic definitions that are applicable to
   this document. Other documents provide more definitions and background
   information. [VK83, HA94]

   Authentication

          The property of knowing that the data received is the same as
          the data that was sent and that the claimed sender is in fact
          the actual sender.

    Integrity
          The property of ensuring that data is transmitted from source
          to destination without undetected alteration.

    Confidentiality
          The property of keeping communications confidential so that
          intended participants can know what is being sent but
          unintended parties are unable to determine what is being sent.

    Encryption
          A mechanism commonly used to provide confidentiality.

    Non-repudiation
          The property of a receiver being able to prove that the sender
          of some data did in fact send the data even though the sender
          might later desire to deny ever having sent that data.

    SAID
          Acronym for "Security Association IDentifier"

    Security Association
          The set of security information relating to a given network
          connection or set of connections. This usually includes
          the cryptographic key, key lifetime, algorithm, algorithm mode,
          sensitivity level (e.g. Unclassified, Secret, Proprietary),
          what kind of security service is provided (authentication-only,
          Transport-Mode Encryption, IP-Mode Encryption, or some combination),
          and possibly other data.

    Traffic Analysis
          A kind of network attack where the adversary is able to make
          deductions about oneself just by analysing the network traffic
          patterns (such as frequency of transmission, who is talking with
          whom, size of packets, Flow Identifier used, etc).

2. DESIGN OBJECTIVES

   This section describes some of the design objectives of this
security architecture and its component mechanisms. The primary
objective of this work is to ensure that IPv6 will have solid security
mechanisms available to users who desire security. These mechanisms
are designed such that Internet users who do not employ these
mechanisms will not be adversely affected. These mechanisms are
intended to be algorithm-independent so that the cryptographic

algorithms can be altered without affecting the other parts of the
implementation. Standard default algorithms (i.e. keyed MD5, DES CBC)
are specified to ensure interoperability in the global Internet. The
selected algorithms are the same as the standard default algorithms
used in SNMPv2. The IPv6 Security mechanisms should be useful in
enforcing a variety of security policies.

3. IPv6 SECURITY MECHANISMS

There are two security mechanisms in IPv6. The first is the
Authentication Header which provides integrity and authentication
without confidentiality. [Atk95a] The second is the Encapsulating
Security Payload which, depending on algorithm and mode, might provide
integrity, authentication, and always provides confidentiality.
[Atk95b] The IPv6 mechanisms do not provide security against a number
of traffic analysis attacks. However, there are several techniques
outside the scope of this specification (e.g. bulk link encryption)
that might be used to provide protection against traffic analysis.
[VK83] The two IPv6 security mechanisms may be combined.

3.1 AUTHENTICATION HEADER

The IPv6 Authentication Header seeks to provide integrity and
authentication for IPv6 datagrams. It does this by computing a
cryptographic authentication function over the IPv6 datagram and using
a secret authentication key in the computation. [Atk95a] The sender
computes the authentication data just prior to sending the
authenticated IPv6 packet and the receiver verifies the correctness of
the authentication data upon reception. Certain fields which must
change in transit, such as the Hop Limit field decremented on each
hop, are omitted from the authentication calculation. However the
omission of the Hop Limit field does not adversely impact the security
provided. Non-repudiation might be provided by some authentication
algorithms (e.g. asymmetric algorithms when both sender and receiver
keys are used in the authentication calculation) used with the
Authentication Header, but it is not necessarily provided by all
authentication algorithms that might be used with the Authentication
Header. The default authentication algorithm is keyed MD5, which like
all symmetric algorithms cannot provide non-repudiation.
Confidentiality and traffic analysis protection are not provided by
the Authenticaton Header.

The IPv6 Authentication Header holds authentication information
for its IPv6 datagram. This authentication information is calculated
using all of the fields in the IPv6 datagram which do not change
during transit from the originator to the recipient. All IPv6
headers, payloads, and the user data are included in this calculation.
The only exception is that fields which need to change in transit (e.g.

   IPv6 Header's "Hop Count" or the IPv6 Routing Header's "Next Address")
   are omitted when the authentication data is calculated.

   Use of the Authentication Header will increase the IPv6 protocol
   processing costs in participating systems and will also increase the
   communications latency. The increased latency is primarily due to the
   calculation of the authentication data by the sender and the
   calculation and comparison of the authentication data by each receiver
   for each IPv6 datagram containing an Authentication Header (AH).

   The Authentication Header provides much stronger security than
   exists in most of the current Internet and should not affect
   exportability or significantly increase implementation cost. While
   the Authentication Header might be implemented by a security gateway
   on behalf of hosts on a trusted network behind that security gateway,
   this mode of operation is not encouraged. Instead, the Authentication
   Header should be used from origin to final destination.

   All IPv6-capable hosts MUST implement the IPv6 Authentication Header
   with at least the MD5 algorithm using a 128-bit key. Other
   authentication algorithms MAY be implemented in addition to keyed MD5.

3.2 ENCAPSULATING SECURITY PAYLOAD

   The IPv6 Encapsulating Security Payload (ESP) seeks to provide
   integrity, authentication, and confidentiality to IPv6
   datagrams. [Atk95b] It does this by encapsulating either an entire
   IPv6 datagram or only the upper-layer protocol data inside the ESP,
   encrypting most of the ESP contents, and then appending a new
   cleartext IPv6 header to the now encrypted Encapsulating Security
   Payload. This cleartext IPv6 header is used to carry the protected
   data through the internetwork. The recipient of the cleartext
   datagram removes and discards the cleartext IPv6 header and cleartext
   IPv6 options, decrypts the ESP, processes and then removes the ESP
   headers, and then processes the (now decrypted) original IPv6 datagram
   or upper-layer protocol data as per the normal IPv6 protocol
   specification.

3.2.1 Description of the ESP Modes

   There are two modes within ESP. The first mode, which is known as
   IP-mode, encapsulates and entire IP datagram within the ESP header.
   The second mode, which is known as Transport-mode, usually encapsulates
   a UDP or TCP frame inside IP.

3.2.2 **Usage of ESP**

   ESP works between hosts, between a host and a security gateway, or
between security gateways. This support for security gateways permits
trustworthy networks behind a security gateway to omit encryption and
thereby avoid the performance and monetary costs of encryption, while
still providing confidentiality for traffic transiting untrustworthy
network segments. When both hosts directly implement ESP and there is
no intervening security gateway, then they may use the Transport-mode
(where only the upper layer protocol data (e.g. TCP or UDP) is
encrypted and there is no encrypted IPv6 header). This mode reduces
both the bandwidth consumed and the protocol processing costs for
users that don't need to keep the entire IPv6 datagram confidential.
ESP works with both unicast and multicast traffic.

3.2.3 **Performance Impacts of ESP**

   The encapsulating security approach used by ESP can noticeably
impact network performance in participating systems, but should not
adversely impact routers or other intermediate systems that are not
participating in the particular ESP association. Protocol processing
in participating systems will be more complex when encapsulating
security is used, requiring both more time and more processing power.
Use of encryption will also increase the communications latency. The
increased latency is primarily due to the encryption and decryption
required for each IPv6 datagram containing an Encapsulating Security
Payload. The precise cost of ESP will vary with the specifics of the
implementation, including the encryption algorithm, key size, and
other factors. Hardware implementations of the encryption algorithm
are recommended when high throughput is desired. Because of the
performance impact, users not requiring confidentiality will probably
prefer to use the IPv6 Authentication Header instead of ESP. For
interoperability throughout the worldwide Internet, all conforming
implementations of IPv6 Encapsulting Security Payload MUST support the
use of the Data Encryption Standard (DES) in Cipher-Block Chaining
(CBC) Mode. Other confidentiality algorithms and modes may also be
implemented in addition to this mandatory algorithm and mode. Export
of encryption and use of encryption are regulated in some countries.
[OTA94]

3.3 **COMBINING SECURITY MECHANISMS**

   In some cases the IPv6 Authentication Header might be combined with
the IPv6 Encapsulating Security Protocol to obtain the desired
security properties. The Authentication Header always provides
integrity and authentication and can provide non-repudiation if used
with certain authentication algorithms (e.g. RSA) . The Encapsulating
Security Payload always provides integrity and confidentiality and can

also provide authentication if used with certain authenticating
encryption algorithms. Adding the Authentication Header to a IPv6
datagram prior to encapsulating that datagram using the Encapsulating
Security Protocol might be desirable for users wishing to have strong
integrity, authentication, confidentiality, and perhaps also
non-repudiation. When the two mechanisms are combined, the placement
of the IPv6 Authentication Header makes clear which part of the data
is being authenticated. Details on combining the two mechanisms are
provided in the IPv6 Encapsulating Security Payload
specification. [At94b]

3.4 OTHER SECURITY MECHANISMS
   Protection from traffic analysis is not provided by any of the
security mechanisms described above. It is unclear whether meaningful
protection from traffic analysis can be provided economically at the
Internet Layer and it appears that few Internet users are concerned
about traffic analysis. One traditional method for protection against
traffic analysis is the use of bulk link encryption. Another
technique is to send false traffic in order to increase the noise in
the data provided by traffic analysis. Reference [VK83] discusses
traffic analysis issues in more detail.

4. KEY MANAGEMENT

   The Key Management protocol that will be used with IPv6 is not
specified in this document. However, because the key management
protocol is coupled to the other security mechanisms only via the
Security Association Identifier (SAID), those other security
mechanisms have been defined in two companion documents. IPv6 is not
intended to support so-called "in-band" key management, where the key
management data is carried in a distinct IPv6 header. Instead it will
primarily use so-called "out-of-band" key management, where the key
management data will be carried by an upper layer protocol such as UDP
or TCP on some specific port number. This permits clear decoupling of
the key management mechanism from the other security mechanisms, and
thereby permits one to substitute new and improved key management
methods without having to modify the implementations of the other
security mechanisms. This is clearly wise given the long history of
subtle flaws in published key management protocols. [NS78, NS81] What
follows in this section is a brief discussion of a few alternative
approaches to key management.

4.1 Manual Key Distribution

   The simplest form of key management is manual key management, where
a person manually configures each system with its own key and also
with the keys of other communicating systems. This is quite practical
in small, static environments but does not scale. It is not a viable

medium-term or long-term approach, but might be appropriate and useful
in many environments in the near-term. For example, within a small
LAN it is entirely practical to manually configure keys for each
system. Within a single administrative domain it is practical to
configure keys for each router so that the routing data can be
protected and to reduce the risk of an intruder breaking into a
router. Another case is where an organisation has an encrypting
firewall between the internal network and the Internet at each of its
sites and it connects two or more sites via the Internet. In this
case, the encrypting firewall might selectively encrypt traffic for
other sites within the organisation using a manually configured key,
while not encrypting traffic with other destinations. It also might
be appropriate when only selected communications need to be secured.

4.2 Some Existing Key Management Techniques

   There are a number of key management algorithms that have been
described in the public literature. Needham & Schroeder have proposed
a key management algorithm which relies on a centralised key
distribution system. [NS78, NS81] This algorithm is used in the
Kerberos Authentication System developed at MIT under Project
Athena. [KB93] More recently, Diffie & Hellman have devised an
algorithm which does not require a centralised key distribution
system. [DH76] Unfortunately, the original Diffie-Hellman technique is
vulnerable to an active "man in the middle" attack. However, this
vulnerability can be mitigated by using signed keys to authentically
bootstrap into the Diffie-Hellman exchange.

4.3 Automated Key Distribution

   Widespread deployment and use of IPv6 security will require an
Internet-standard scalable key management protocol. Ideally such a
protocol would support a number of protocols in the Internet protocol
suite, not just IPv6 security. There is work underway within the IETF
to add signed host keys to the Domain Name System [EK94] The DNS keys
enable the originating party would to authenticate key management
messages with the other key management party using an asymmetric
algorithm. The two parties would then have an authenticatible
communications channel that could be used to create a shared session
key using Diffie-Hellman or other means. [DH76]

   There are two keying approaches for IPv6. The first approach,
called host-to-host keying, has all users on host 1 share the same key
for use on traffic destined for all users on host 2. The second
approach, called user-to-user keying, lets user A on host 1 have a
unique session key with user B on host 2 that is not shared with other
users on host1. In many cases, a single computer system will have at
least two mutually suspicious users A and B that do not trust each

other. When host-to-host keying is used and mutually suspicious users
exist, it is possible for user A to determine the host-to-host key via
well known methods, such as a Chosen Plaintext attack. Once user A
has improperly obtained the key in use, user A can then either read
user B's encrypted traffic or forge traffic from user B. When
user-to-user keying is used, this kind of attack from one user onto
another user's traffic is not possible. Hence, support for
user-to-user keying must be present in all IPv6 implementations, as is
described in the "IPv6 Key Management Requirements" section below.

4.4 Multicast Key Distribution

   Multicast key distribution is an active research area in the
published literature as of this writing. For multicast groups having
relatively few members, manual key distribution or multiple use of
existing unicast key distribution algorithms such as modified
Diffie-Hellman appears feasible. For very large groups, new scalable
techniques will be needed. The use of Core-Based Trees (CBT) to
provide session key management as well as multicast routing might be
an approach used in the future. [BFC93]

4.5 IPv6 Key Management Requirements
   This section defines key management requirements for all IPv6
implementations. It applies equally to the IPv6 Authentication Header
and the IPv6 Encapsulating Security Payload.

   All IPv6 implementations MUST support manual key management. All
IPv6 implementations SHOULD support an Internet standard key
management protocol once the latter is defined. All IPv6
implementations MUST permit the configuration and use of user-to-user
keying for traffic originating at that system and MAY additionally
permit the configuration of host-to-host keying for traffic
originating at that system as an added feature to make manual key
distribution easier and give the system administrator more
flexibility.

   A device that encrypts or authenticates IPv6 packets originated on
other systems, for example a dedicated IP encryptor or an encrypting
gateway, cannot generally provide user-to-user keying for traffic
originating on other systems. Hence, such systems MUST implement
support for host-to-host keying for traffic originating on other
systems and MAY implement support for user-to-user keying for traffic
originating on other systems.

   The method by which keys are configured on a particular system is
implementation-defined. A flat file containing security association
identifiers and the security parameters, including the key(s), is an
example of one possible method for manual key distribution. An IPv6

   system MUST take reasonable steps to protect the keys and other security
   association information from unauthorised examination or modification
   because all of the security lies in the keys.

5. **USAGE**
   This section describes the possible use of the security mechanisms
   provided by IPv6 in several different environments and applications
   in order to give the implementer and user a better idea of how these
   mechanisms can be used to reduce security risks.

5.1 **USE WITH FIREWALLS**
   Firewalls are not uncommon in the current Internet. [CB94] While
   many dislike their presence because they restrict connectivity, they
   are unlikely to disappear in the near future. Both of the IPv6
   mechanisms can be used to increase the security provided by firewalls.

   Firewalls used with IPv6 will need to be able to parse the header
   daisy-chain to determine the transport protocol (e.g. UDP or TCP) in
   use and the port number for that protocol. Firewall performance
   should not be significantly affected by use of IPv6 because the header
   format rules in IPv6 make parsing easy and fast.

   Firewalls can use the Authentication Header to gain assurance that
   the data (e.g. source, destination, transport protocol, port number)
   being used for access control decisions is correct and authentic.
   IPv4 firewalls are unable to authenticate the data being used for
   access control decisions and necessarily trust data that is not
   trustworthy. Authentication might be performed not only within an
   organisation or campus but also end to end with remote systems across
   the Internet. This use of the Authentication Header with IPv6
   provides much more assurance of security than IPv4 provides.

   Organisations with two or more sites that are interconnected using
   commercial IP service might wish to use a selectively encrypting
   firewall. If an encrypting firewall were placed between each site of
   the Foo Company and the commercial IP service provider, the firewall
   could provide an encrypted IP tunnel among all of the Foo Company's
   sites. It could also encrypt traffic between the Foo Company and its
   suppliers, customers, and other affiliates. Traffic with the NIC,
   with public Internet archive, or some other organisations might not be
   encrypted because of the unavailability of a standard key management
   protocol or as a deliberate choice to facilitate better
   communications, improved network performance, and increased
   connectivity. Such a practice could easily protect the organisation's
   sensitive traffic from eavesdropping and modification.

   Some organisations (e.g. governments) might wish to use a fully
   encrypting firewall to provide a protected virtual network over

   commercial IP service. The difference between that and a bulk IPv6
   encryption device is that a fully encrypting firewall would provide
   filtering of the decrypted traffic as well as providing encryption of
   IP packets.

5.3 **USE WITH IPv6 MULTICAST**
   In the past several years, the Multicast Backbone (MBONE) has grown
   rapidly. IETF meetings and other conferences are now regularly
   multicast with real-time audio, video, and whiteboards. Many people
   are now using teleconferencing applications based on IP Multicast in
   the Internet or in private internal networks. Hence it is important
   that the security mechanisms in IPv6 be suitable for use in an
   environment where multicast is the general case.

   The Security Association Identifiers (SAIDs) used in the IPv6
   security mechanisms are receiver-oriented, making them well suited for
   use in IP multicast. [Atk95a, Atk95b] Unfortunately, most currently
   published multicast key distribution protocols do not scale well.
   However, there is active research in this area. As an interim step, a
   multicast group could repeatedly use a secure unicast key distribution
   protocol to distribute the key to all members or the group could
   pre-arrange keys using manual key distribution.

5.4 **USE TO PROVIDE QOS PROTECTION**
   The recent IAB Security Workshop identified Quality of Service
   protection as an area of significant interest. [BCCH] The two IPv6
   security mechanisms are intended to provide good support for real-time
   services as well as multicasting. This section describes one possible
   approach to providing such protection.

   The Authentication Header can be used, with appropriate key
   management, to provide authentication of packets. This authentication
   is potentially important in packet classification within routers. The
   IPv6 Flow Identifier can act as a Low-Level Identifier (LLID). Used
   together, packet classification within routers becomes
   straightforward if the router is provided with the appropriate key
   material. For performance reasons the routers might authenticate only
   every Nth packet rather than every packet, but this is still a
   significant improvement over capabilities in the current Internet.
   Quality of service provisioning is likely to also use the Flow ID in
   conjunction with a resource reservation protocol, such as RSVP. Thus,
   the authenticated packet classification can be used to help ensure
   that each packet receives appropriate handling inside routers.

5.5 **USE IN COMPARTMENTED OR MULTI-LEVEL NETWORKS**
   A multi-level secure (MLS) network is one where a single network is
   used to communicate data at different sensitivity levels (e.g.
   Unclassified and Secret). Many governments have significant interest

in MLS networking. [DIA] The IPv6 security mechanisms have been
designed to support MLS networking. MLS networking requires the use
of strong Mandatory Access Controls (MAC) which ordinary users are
incapable of controlling or violating. Mandatory Access Controls
differ from Discretionary Access Controls in this respect.

   The Authentication Header can be used to provide strong
authentication among hosts in a single-level network. The
Authentication Header can also be used to provide strong assurance for
both mandatory access control decisions in multi-level networks and
discretionary access control decisions in all kinds of networks. If
IP sensitivity labels are used and confidentiality is not considered
necessary within the particular operational environment, the
Authentication Header is used to provide authentication for the entire
packet, including cryptographic binding of the sensitivity level to
the IPv6 header and user data. This is a significant improvement over
labelled IPv4 networks where the label is trusted even though it is
not trustworthy because there is no authentication or cryptographic
binding of the label to the IP header and user data.

   The Encapsulating Security Payload can be combined with appropriate
key policies to provide full multi-level secure networking. In this
case each key must be used only at a single sensitivity level and
compartment. For example, Key "A" might be used only for sensitive
Unclassified packets, while Key "B" is used only for
Secret/No-compartments traffic, and Key "C" is used only for
Secret/No-Foreign traffic.

   In sensitive environments, appropriate organisational policies will
dictate the actual key management policy and also the set of
algorithms that are appropriate for use. In such environments, the
ability to communicate between the Internet and the hosts handling
sensitive data is probably undesirable. Hence, systems only handling
sensitive information might not implement the Internet standard
algorithms and instead only have algorithms approved by appropriate
policies for such use. Such systems would not be fully conforming to
the IPv6 Encapsulating Security Payload specification with regard to
implementation of the mandatory Internet algorithm, but those users
might not care or might consider that to be desirable.

   Encryption is very useful and desirable even when all of the hosts
are within a protected environment. The Internet-standard encryption
algorithm could be used, in conjuction with appropriate key
management, to provide strong Discretionary Access Controls (DAC) in
conjunction with either implicit or explicit sensitivity
labels. [Ken91] Some environments might consider the Internet-standard
encryption algorithm sufficiently strong to provide Mandatory Access
Controls (MAC). Full encryption SHOULD be used for all communications

between multi-level computers or compartmented mode workstations even
when the computing environment is considered to be protected.

[6](). **SECURITY CONSIDERATIONS**

   This entire draft discusses the IPv6 Security Architecture.

   Users need to understand that the quality of the security provided
by the mechanisms provided by IPv6 depends completely on the strength
of the implemented cryptographic algorithms, the strength of the key
being used, the correct implementation of the cryptographic
algorithms, the security of the key management protocol, and the
correct implementation of IPv6 and the several security mechanisms in
all of the participating systems. The security of the implementation
is in part related to the security of the operating system which
embodies the security implementations. For example, if the operating
system does not keep the private cryptologic keys confidential, then
traffic using those keys will not be secure. If any of these is
incorrect or insufficiently secure, little or no real security will be
provided to the user. Because different users on the same system might
not trust each other, each user or each session should usually be
keyed separately. This will also tend to increase the work required
to cryptanalyse the traffic since not all traffic will use the same key.

   Certain security properties (e.g. traffic analysis protection) are
not provided by any of these mechanisms. One possible approach to
traffic analysis protection is appropriate use of link
encryption. [VK83] Users must carefully consider which security
properties they require and take active steps to ensure that their
needs are met by these or other mechanisms.

   Certain applications (e.g. electronic mail) probably need to have
application-specific security mechanisms. Application-specific
security mechanisms are out of the scope of the IPv6 Security
Architecture. Users interested in electronic mail security should
consult the RFCs describing the Internet's Privacy-Enhanced Mail
system. Users concerned about other application-specific mechanisms
should consult the online RFCs to see if suitable Internet Standard
mechanisms exist.

ACKNOWLEDGEMENTS

   Many of the concepts here are derived from or were influenced by the
US Government's SDNS security protocol specifications, the ISO/IEC's
NLSP specification, or from the proposed swIPe security
protocol. [SDNS, ISO, IB93, IBK93] The work done for SNMP Security
and SNMPv2 Security influenced the choice of default cryptological
algorithms and modes. [GM93] Steve Bellovin, Steve Deering, Richard

   Hale, George Kamis, Phil Karn, Frank Kastenholz, and Dave Mihelcic
   provided critiques of early versions of this draft.

REFERENCES
   [Atk95a] Randall Atkinson, IPv6 Authentication Header, Internet Draft,
            draft-atkinson-ipng-auth-01.txt, 16 February 1995.

   [Atk95b] Randall Atkinson, IPv6 Encapsulating Security Payload, Internet
            Draft, draft-atkinson-ipng-esp-01.txt, 16 February 1995

   [BCCH94] R. Braden, D. Clark, S. Crocker, & C. Huitema, "Report of IAB
            Workshop on Security in the Internet Architecture", RFC-1636,
            DDN Network Information Center, June 1994.

   [BFC93] A. Ballardie, P. Francis, & J. Crocroft, "Core Based Trees:
            An Architecture for Scalable Inter-Domain Multicast Routing",
            Proceedings of ACM SIGCOMM 93, ACM Computer Communications Review,
            Volume. 23, Number 4, October 1993, pp. 85-95.

   [CB94]   William R. Cheswick & Steven M. Bellovin, Firewalls & Internet
            Security, Addiwon-Wesley, Reading, MA, 1994.

   [DIA]   US Defense Intelligence Agency, "Compartmented Mode Workstation
            Specification", Technical Report DDS-2600-6243-87.

   [DH76]   W. Diffie & M. Hellman, "New Directions in Cryptography", IEEE
            Transactions on Information Theory, Vol. IT-22, No. 6, November
            1976, pp. 644-654.

   [EK94]   D. Eastlake III & C. Kaufman, "Domain Name System Protocol
            Security Extensions", Internet Draft, March 1994.

   [GM93]   J. Galvin & K. McCloghrie, Security Protocols for version 2
            of the Simple Network Management Protocol (SNMPv2), RFC-1446,
            DDN Network Information Center, April 1993.

   [HA94]   N. Haller & R. Atkinson, "On Internet Authentication", RFC-1704,
            DDN Network Information Center, October 1994.

   [Hin94] Bob Hinden (Editor), Internet Protocol version 6 (IPv6)
Specification,
            draft-hinden-ipv6-spec-00.txt, October 1994.

   [ISO]   ISO/IEC JTC1/SC6, Network Layer Security Protocol, ISO-IEC
            DIS 11577, International Standards Organisation, Geneva,
            Switzerland, 29 November 1992.

   [IB93]   John Ioannidis and Matt Blaze, "Architecture and Implementation of
            Network-layer Security Under Unix", Proceedings of USENIX Security

           Symposium, Santa Clara, CA, October 1993.

    [IBK93] John Ioannidis, Matt Blaze, & Phil Karn, "swIPe: Network-Layer
            Security for IP", presentation at the Spring 1993 IETF Meeting,
            Columbus, Ohio.

    [Ken91] Steve Kent, US DoD Security Options for the Internet Protocol,
            RFC-1108, DDN Network Information Center, November 1991.

    [Ken93] Steve Kent, Privacy Enhancement for Internet Electronic Mail:
            Part II: Certificate-Based Key Management, RFC-1422, DDN Network
            Information Center, 10 February 1993.

    [KB93] J. Kohl & B. Neuman, The Kerberos Network Authentication Service
(V5),
            RFC-1510, DDN Network Information Center, 10 September 1993.

    [NS78] R.M. Needham & M.D. Schroeder, "Using Encryption for Authentication
            in Large Networks of Computers", Communications of the ACM,
            Vol. 21, No. 12, December 1978, pp. 993-999.

    [NS81] R.M. Needham & M.D. Schroeder, "Authentication Revisted",
            ACM Operating Systems Review, Vol. 21, No. 1., 1981.

    [OTA94] US Congress, Office of Technology Assessment, "Information Security
            & Privacy in Network Environments", OTA-TCT-606, Government Printing
            Office, Washington, DC, September 1994.

    [SDNS] SDNS Secure Data Network System, Security Protocol 3, SP3,
            Document SDN.301, Revision 1.5, 15 May 1989, published
            in NIST Publication NIST-IR-90-4250, February 1990.

    [VK83] V.L. Voydock & S.T. Kent, "Security Mechanisms in High-level
            Networks", ACM Computing Surveys, Vol. 15, No. 2, June 1983.

DISCLAIMER

    The views expressed in this note are those of the author and are not
    necessarily those of his employer. The Naval Research Laboratory has
    not passed judgement on the merits, if any, of this work. The author
    and his employer specifically disclaim responsibility for any problems
    arising from correct or incorrect implementation or use of this
    design.

AUTHOR INFORATION

    Randall Atkinson <atkinson@itd.nrl.navy.mil>
    Information Technology Division
    Naval Research Laboratory

     Washington, DC 20375-5320
     USA

     Voice: (DSN) 354-8590
     Fax: (DSN) 354-7942