Global Information Assurance Certification Paper

Security Implementations in the Next Generation of the Internet Protocol

Jess Rutherford, GSEC v1.4b, Option 1

Abstract

The system is outdated. The fourth version of the Internet Protocol (today's standard) has been showing its age for some time now. The Internet Engineering Standards Group (IESG), the Internet Engineering Task Force (IETF), as well as the Internet community at large, have been developing a system to fix the problems inherent in IPv4.

Some of the reasons IPv4 is becoming antiquated are as follows: severely crippled available addressing space, address management and host configuration complexities and problems, and the lack of security features at the internet layer, to name a few.

Why IPv6?

IPv4, developed in the 70's, was designed for that era in computing. Nearly thirty years later, much has changed in the use of the Internet Protocol.

The number of devices has dramatically increased, still expanding from computers to wireless telephones and pocket computing devices. The addressing space has increased from 32-bit (IPv4) to 128-bit (IPv6).

With the 32-bit addressing scheme, 4.3 billion addresses are theoretically available. However, due to the class-system, far fewer are actually usable (For example, addresses 1.x.x.x through 127.x.x.x are all Class A addresses, and individual entities can control the entire address space after, for example, the "1". Approximately 16,800,000 hosts are available in each Class A address, of which roughly 126 different entities control, reducing the total available space by 2,116,800,000, or about half.). Further, there are restricted addresses (for local networks, etc), which cannot be publicly assigned, reducing the amount even more.

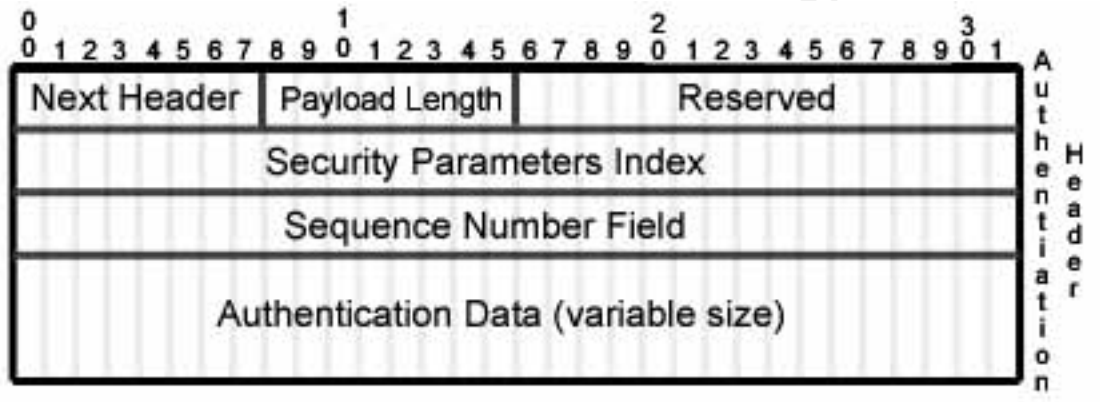
With the 128-bit scheme, you have 3.4 x 10^38, or 655,570,793,348,866,943,898,599 addresses available, in a classless system.

In addition, many new features exist, such as Neighbor Discovery, Autoconfiguration, multicast, as well as various Mobile Networking options, for cellular phones and other mobile devices, but the backgrounds and explanations behind these mechanisms are beyond the scope of this document.

Authentication Header

The Authentication Header is a service provided by IPv6 to provide integrity checks for traffic, and to provide an optional protection against replay attacks.

The Authentication Header's main purpose is to take the information from as many of the IP Header fields as possible (i.e. the ones that should not change in transit), and create a unique value which is then stored in the Authentication header and compared upon receipt of the packet. Some of the fields in the IP header are dynamic, nor can be predicted by the packet recipient, therefore cannot be protected by the Authentication Header.



The first field in the Authentication Header layout is the Next Header field, which is an 8-bit field which identifies the next payload in the datagram which follows the AH. The value is one of the numbers defined by the Internet Assigned Numbers Authority

The Payload Length Field follows the Next Header field, and simply specifies the length of the Authentication Header in 32-bit words.

The field following the Payload Length field has been set aside for future use. For the authentication calculation the field is set to "zero", but is otherwise ignored.

The Security Parameters Index, or SPI, is a 32-bit value, when combined with the security protocol and the destination IP address, identifies any given security association with a datagram.

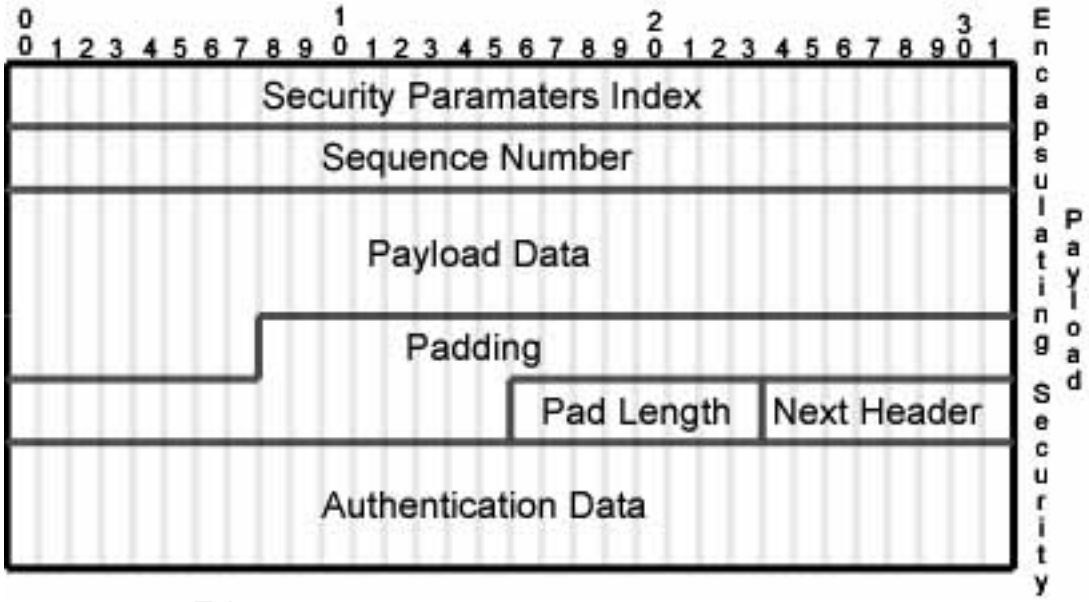
Following the SPI is the Sequence Number, which is an unsigned 32-bit counter value. It is required, even if the receiver is not using the anti-replay service of the Authentication Header. When a Security Association is to be made, both the sender and the receiver's counters are reset to zero. By this statement, the first packet using a Security Association should have a sequence number of

1. If using anti-replay, a counter must not cycle – it will have to be reset when it hits the end of the 32-bit number, creating a new Security Association.

The Authentication Data field is variable length, and contains the Integrity Check Value (ICV) for the datagram. The field length must be a multiple of 64-bits (in IPv6, 32-bits in IPv4), and the field may contain padding to offset the value to complete the 64 bit multiple.

Encapsulating Security Payload

The Encapsulating Security Payload (ESP) header is a service provided in IPv6 which offers a few different services, including: "confidentiality, data origin authentication, connectionless integrity, an anti-replay service (a form of partial-sequence integrity) , and limited traffic flow confidentiality." ([3], p. 2)



The ESP header is broken down into the fields named in the graphic above. The first is the Security Parameter index, which functions the same in the ESP header as it does in the Authentication Header; a 32-bit value which identifies any given security association for the datagram.

The next field is the Sequence Number, which also serves the same function as in the AH; a 32-bit field containing an increasing counter value.

The Payload Data field contains data which is identified by the Next header field (described below). The Payload Data field is where the data resides after it has been offered the various services of the ESP header, in addition to the cryptographic synchronization data, if the chosen cryptographic algorithm calls for one.

The Padding field is used for one of several functions:

- For alignment purposes, to make sure the cyphertext (data post-encryption) ends on a 4-byte boundary (see above graphic). “The Pad Length and Next Header fields must be right aligned within a 4-byte word… to ensure that the Authentication Data field (if present) is aligned on a 4-byte boundary.” ([3], p. 4).
- Algorithm requirements may need the plaintext (data pre-encryption) to be a multiple of some number of bytes. The padding field will be used in conjunction with the sizes of the Payload Data, Pad Length, and Next Header fields to meet the required block size.
- Partial Traffic Flow Confidentiality can be achieved by using excess padding to conceal the actual size of the payload (with the obvious expense of increased bandwidth).

Inclusion of the 0 to 255 bytes of padding is optional in any given ESP implementation, but the consumption of the padding is required in all.

The following header is the Next header, which functions similar to the one used in the Authentication Header; it identifies the data being sent in the Payload Data Field, and is a mandatory field.

The last field in the ESP header construction is the Authentication Data field, which functions the same way it does in the Authentication Header. This field is optional, and is only included if the service has been selected by the appropriate Security Association.

When it comes down to the actual encryption, there are several different algorithms which may be employed. ESP is designed to use symmetric encryption algorithms, and each packet must carry the data required to establish cryptographic synchronization on the receiving end to decrypt the packet, since packets may be received out of order. The AH uses different algorithms, depending on if the connection is uni- or multicast: Unicast (point-to-point) would optimally use Message Authentication Codes based on symmetric algorithms, and multicast transmissions could use one-way hash algorithms combined with asymmetric signature algorithm. Note that authentication is optional. The algorithms for both encryption and authentication are specified by the Security Association.

If a compliant ESP system is desired, the following encryption algorithms must be implemented: DES in CBC mode, HMAC with MD5, and HMAC with SHA-1 (the workings of these encryption algorithms are beyond the scope of this document). There are two other systems which must be implemented, a NULL Encryption algorithm, and a NULL Authentication algorithm, both of which are used for consistency when Security Associations are negotiated, and there is a lack of an

encryption algorithm, or authentication functions. Either can be NULL, but they both cannot be NULL simultaneously.

Security Associations

A security association (SA) is a unidirectional “connection” between two points. The concept of a security association is required in any implementation and use of the Authentication Header and the Encapsulating Security Payload, making it a vital part of any IPv6 implementation. Each security association gives the security services of an AH or an ESP, but not both. if both are to be used, two (or more) separate security associations are to be made.

As mentioned previously, each security association can be used in only one direction, which means any attempt at a secure bidirectional communication will need a second set of security associations to complete the two-way link (between two hosts, two security gateways, or any combination thereof). When multiple security associations are used, sometimes this group is referred to as a “security association bundle” (or “SA Bundle”).

A security association is composed of three things: a security protocol identifier (either Authentication Header or an Encapsulation Security Payload), a destination IP address, and a Security Parameter Index (SPI, which is simply a unique identifier for any given SA).

Existing, are two methods of applying security associations to Internet traffic, tunnel mode and transport mode.

Transport mode is when the SA exists between two hosts. Routinely, the protection offered by ESP in a SA in transport mode protects the datagram’s payload, but not the IP information itself (destination address, etc). When an AH is used in transport mode, more protection can be offered to select sections of the IP header, sections of the extension header, and selected options. Traffic in a transport mode SA could be passed just as easily over the Internet if there was no SA present.

In tunnel mode, more protection is offered to the IP datagram itself, in addition to the payload. In a tunnel mode SA, there is both an “inner” and an “outer” IP header. The “outer” header specifies the destination for IPSec processing (i.e., a security gateway), and the “inner” header is the final destination of the packet, revealed only after the packet is opened by the security gateway.

“A host MUST support both transport and tunnel mode,” and “a security gateway is required to support only tunnel mode. If it supports transport mode,

that should be used only when the security gateway is acting as a host, e.g., for network management." ([2], p.9)

In the end, security associations are but the bricks of a security design, and the Security Policy Database is the mortar that holds them all together. The Security Policy Database (SBD) is a user- (network/security admin-) controlled database, which must be called upon during the processing of all network traffic (inbound and outbound), including any traffic destined to bypass IPSec.

For any passing datagram, three choices are possible: Apply IPSec, bypass IPSec, or discard the datagram.

If the datagram is to have IPSec applied, it is afforded protection, and it is up to the SPD to state the security services to be provided, the encryption algorithms to be employed, and the protocols to be used. If the datagram is allowed to bypass IPSec, the traffic continues without any additional security protection. If the packet is to be discarded, it is dropped, not reaching the intended host or destination application.

Ipv6: The Silver Bullet?

Although the generous offered security features in IPv6 will allow both security officers and the home user alike, greater assurance that their data is secure, this, like many other things in the security world, is by no means any sort of end-all solution. Many factors independent of any particular IPSec implementation exist, including personnel, physical, procedural, compromising emanations (emissions), computer security practices, defects in Operating System security, poor quality of random number sources, and sloppy system management protocols and practices, to name a few.

Encryption algorithms for the ESP header may also fall under the "flaw" category, due to current export laws on the strength of the algorithm. Global implementations of ESP would have to use a weak encryption scheme in order to conform to the export laws. This, combined with the "security cure-all" mentality of many users and corporations (remember when people thought firewalls were magic devices that stopped hackers?), and lull them into a false sense of security. Given enough time, any encryption scheme can be broken; the weaker the key, the less time it takes. With computing power rising and the prices falling, malicious users could crack streams of data within days or weeks.

Closing Topic-

IPv6 will be both a godsend and a nightmare for network administrators and security engineers. After deployment, like many other new or foreign technologies, the nightmare will pass. What will be left behind is a medium which is more flexible, and easier to work with then what we have had before, in addition to giving everyone much needed security technologies which will restrict the ways malicious users can abuse computer systems, and the internet.

References:

[1] Loshin, Pete. IPv6: Clearly Explained. San Francisco: Morgan Kaufmann Publishers, Inc.

[2] Kent & Atkinson. “Security Architecture for the Internet Protocol.” Nov 1998. http://www.ietf.org/rfc/rfc2401.txt (1/31/2003).

[3] Kent & Atkinson. “IP Encapsulating Security Payload (ESP).” Nov 1998. http://ftp.isi.edu/in-notes/rfc2406.txt (2/6/2003).

[4] Kent & Atkinson. “IP Authentication Header.” Nov 1998. http://ftp.is.edu/in-notes/rfc2402.txt (2/6/2003).

[5] Hinden, Robert. “IP Version 6 (IPv6).” Jan 03, 2003. http://playground.sun.com/pub/ipng/html/ipng-main.html (1/17/2003).

[6] The Internet Architecture Board “The Case for IPv6” Dec 25, 1999. http://www.6bone.net/misc/case-for-ipv6.html (12/28/2002)

[7] C2Net Software, Inc. “Hackers Smash U.S. Government Encryption Standard.” Jun 18, 1997 http://www.itsc.ncs.com.sg/news.html (3/20/2003)

Last Updated: May 23rd, 2017

Upcoming Training

SANS Stockholm 2017	Stockholm, Sweden	May 29, 2017 - Jun 03, 2017	Live Event
Security Operations Center Summit & Training	Washington, DC	Jun 05, 2017 - Jun 12, 2017	Live Event
SANS Houston 2017	Houston, TX	Jun 05, 2017 - Jun 10, 2017	Live Event
Community SANS Ottawa SEC401	Ottawa, ON	Jun 05, 2017 - Jun 10, 2017	Community SANS
SANS San Francisco Summer 2017	San Francisco, CA	Jun 05, 2017 - Jun 10, 2017	Live Event
SANS Charlotte 2017	Charlotte, NC	Jun 12, 2017 - Jun 17, 2017	Live Event
SANS Rocky Mountain 2017 - SEC401: Security Essentials Bootcamp Style	Denver, CO	Jun 12, 2017 - Jun 17, 2017	vLive
SANS Secure Europe 2017	Amsterdam, Netherlands	Jun 12, 2017 - Jun 20, 2017	Live Event
Community SANS Portland SEC401	Portland, OR	Jun 12, 2017 - Jun 17, 2017	Community SANS
SANS Rocky Mountain 2017	Denver, CO	Jun 12, 2017 - Jun 17, 2017	Live Event
SANS Minneapolis 2017	Minneapolis, MN	Jun 19, 2017 - Jun 24, 2017	Live Event
SANS Columbia, MD 2017	Columbia, MD	Jun 26, 2017 - Jul 01, 2017	Live Event
SANS Cyber Defence Canberra 2017	Canberra, Australia	Jun 26, 2017 - Jul 08, 2017	Live Event
SANS Paris 2017	Paris, France	Jun 26, 2017 - Jul 01, 2017	Live Event
SANS London July 2017	London, United Kingdom	Jul 03, 2017 - Jul 08, 2017	Live Event
Cyber Defence Japan 2017	Tokyo, Japan	Jul 05, 2017 - Jul 15, 2017	Live Event
SANS Munich Summer 2017	Munich, Germany	Jul 10, 2017 - Jul 15, 2017	Live Event
Community SANS Phoenix SEC401	Phoenix, AZ	Jul 10, 2017 - Jul 15, 2017	Community SANS
SANS Cyber Defence Singapore 2017	Singapore, Singapore	Jul 10, 2017 - Jul 15, 2017	Live Event
Community SANS Minneapolis SEC401	Minneapolis, MN	Jul 10, 2017 - Jul 15, 2017	Community SANS
SANS Los Angeles - Long Beach 2017	Long Beach, CA	Jul 10, 2017 - Jul 15, 2017	Live Event
Mentor Session - SEC401	Macon, GA	Jul 12, 2017 - Aug 23, 2017	Mentor
Mentor Session - SEC401	Ventura, CA	Jul 12, 2017 - Sep 13, 2017	Mentor
Community SANS Atlanta SEC401	Atlanta, GA	Jul 17, 2017 - Jul 22, 2017	Community SANS
Community SANS Colorado Springs SEC401	Colorado Springs, CO	Jul 17, 2017 - Jul 22, 2017	Community SANS
SANSFIRE 2017	Washington, DC	Jul 22, 2017 - Jul 29, 2017	Live Event
Community SANS Charleston SEC401	Charleston, SC	Jul 24, 2017 - Jul 29, 2017	Community SANS
SANSFIRE 2017 - SEC401: Security Essentials Bootcamp Style	Washington, DC	Jul 24, 2017 - Jul 29, 2017	vLive
Community SANS Fort Lauderdale SEC401	Fort Lauderdale, FL	Jul 31, 2017 - Aug 05, 2017	Community SANS
SANS San Antonio 2017	San Antonio, TX	Aug 06, 2017 - Aug 11, 2017	Live Event
SANS Prague 2017	Prague, Czech Republic	Aug 07, 2017 - Aug 12, 2017	Live Event