Network Working Group
Internet-Draft
Obsoletes: RFC4302 (if approved)
Updates: RFC8200 and RFC4303 (if approved)
Intended status: Standards Track
Expires: 6 July 2026

T. Herbert
XDPnet

2 January 2026

# Deprecate IP Authentication Header
# draft-herbert-deprecate-auth-header-01

## Abstract

This document deprecates the IP Authentication Header. The motivations are that authentication without confidentiality is not compelling, the Authentication Header is incompatible with some commonly deployed protocols, and there is likely no deployment of Authentication Header.

## Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at https://datatracker.ietf.org/drafts/current/.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on 6 July 2026.

## Copyright Notice

Table of Contents



## 1. Introduction

The IP Authentication Header (AH) [RFC4302] is used to provide connectionless integrity and data origin authentication for IP datagrams. AH provides authentication for as much of the IP header as possible, as well as for next level protocol data. However, some IP header fields may change in transit and the value of these fields, when the packet arrives at the receiver, may not be predictable by the sender. The values of such fields cannot be protected by AH. Thus, the protection provided to the IP header by AH is piecemeal.

This document deprecates the IP Authentication Header. There are three motivations: 1) There is no compelling reason to use authentication without encryption, 2) The Authentication Header is incompatible with a number of other protocols and breaks when those protocols are also used, and 3) There is likely zero deployment of the Authentication Header.

If approved, this document obsoletes [RFC4302] and it updates [RFC8200] and [RFC4303].

### 1.1. Requirements Language

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "NOT RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in BCP 14 [RFC2119] [RFC8174] when, and only when, they appear in all capitals, as shown here.

## 2. Related work

### 2.1. Draft to move AH to historical status

In 2011 [ah-to-hist] was posted for moving the Authentication Header to historical status. The primary motivations were that AH breaks Network Address Translation (NAT), and ESP-NULL can do everything useful that can be done with AH.

The draft did point at that it is argued that ESP in the tunnel mode is equivalent to AH transport mode, however ESP tunnel mode SA applied to an IPv6 flow results in at least 50 bytes of additional overhead per packet. The draft suggests that the issue may be alleviated by header compression schemes.

The draft mentions that firewalls in the enterprise environments often require visibility into packets. This is easier in AH than ESP since packets are transmitted in the clear. The draft asserts that this is solved by [RFC5840]

### 2.2. Draft recommending to avoid AH

[ah-to-hist] did not make progress in IETF so the author posted [ah-avoid]. This draft recommends retiring AH, and in particular it recommends that AH must not be used for new applications and protocols.

The motivations for retiring AH are the same as those given in [ah-to-hist]

### 2.3. RFC8221

[RFC8221] was published in 2017 and goes a long way towards deprecating the Authentication header. The RFC includes the following requirements:

* Using ESP with AH is NOT RECOMMENDED.

* ENCR_NULL is a MUST to enable the use of ESP with only authentication, which is preferred over AH due to NAT traversal

* It is NOT RECOMMENDED to use ESP with NULL authentication (with non-authenticated encryption) in conjunction with AH; some configurations of this combination of services have been shown to be insecure

## 3. Motivation

This section gives the motivation for deprecating the IP Authentication Header. The conclusion of this document is that the Authentication Header can be deprecated with little or no ill effects.

### 3.1. Authentication without encryption is not compelling

The Authentication Header only provides for authentication of packet data and not confidentiality, that it does not encrypt the data. In its nature, authenticating data but not encrypting it is weaker security then authenticating and encrypting the data.

The Encapsulating Security Payload header [RFC4303] (ESP) can provide both integrity and confidentiality. The primary difference between the integrity provided by ESP and AH is the extent of the coverage. Specifically, ESP does not protect any IP header fields unless those fields are encapsulated by ESP. The value in the additional coverage afforded by AH is marginal. If fields in the IP header itself are corrupted it’s likely that ESP with authentication would fail to be authenticated at the receiver, or at least the packet would be dropped due to a corrupted IP header. Since the lookup for the Security Association in ESP includes the packet’s source and destination addresses there is protection against spoofed or corrupted IP addresses.

AH also provides integrity for IPv4 options or for IPv6 extension headers that precede the Authentication Header whereas ESP does not. This is also of marginal value since the options or extension headers themselves are sent in plain text with no confidentiality thereby making end-to-end information visible to untrusted parties. Besides that, IP options have been effectively deprecated and [depeh] proposes to deprecate the use of plain text extension headers in untrusted networks which is where AH is most applicable.

Ostensibly, an advantage of performing authentication without encryption is that the algorithms are simpler and lend themselves to practical and performant implementation. While this may have been true for legacy hardware twenty years ago, modern hardware has excellent support for computing AES at line rate so that the differences in algorithmic and implementation complexity between authentication and encryption is no longer pertinent.

### 3.2. Incompatibility of AH with other protocols

The Authentication Header does not provide integrity for data that might be modified in-flight. For this reason, when calculating the Integrity Check Value (IVC) for AH any fields that might be modified are removed from the calculation. For IPv4 this includes the TOS, Fragment Offset, TTL, and header checksum fields in the IPv4 header, as well as any mutable IP options; for IPv6 this includes the Hop Limit, Flow Label, and Traffic Class fields in the IPv6 header, as well as Hop-by-Hop and Destination options that are marked as modifiable. The same procedures must be followed by both the sender and the receiver, and the implementation for all this is rather complex. If fields are modified in-flight beyond those that AH allows to be modified then authentication will fail at the receiver. There are a number of protocols that make such modifications.

The Authentication Header is fundamentally incompatible with Network Address Translation (NAT) [RFC2663]. Network Address Translation modifies IP addresses and possibly port numbers of packets in-flight. If a packet containing an Authentication Header goes through a NAT device where IP addresses or port numbers are modified then the packet will fail to be authenticated at the receiver. There is no workaround for this. Even if the NAT detected the presence of an Authentication Header there is no means to incrementally update it like the TCP checksum can be updated by NAT.

There are other protocols that modify packets in-flight in ways that are incompatible with the Authentication Header. For Segment Routing [RFC8754], it is explicitly stated that use of the Segment Routing header with the Authentication Header is not supported. Similarly, In-flight removal of Hop-by-Hop Options header or the Routing header [inflrm] will not work if an Authentication Header is present in the packet.

Another issue is that the Authentication Header does not work with checksum offload. Checksum offload is a ubiquitous feature in Network Interface Cards (NICs) where a hardware device computes and sets the TCP or UDP checksum as packets are sent. From the point of view of AH the effect checksum offload is an unexpected in-flight modification to a packet, so packets with an Authentication Header that go through checksum offload will fail to be authenticated at the receiver.

### 3.3. No deployment of AH

It is a reasonable extrapolation that the Authentication Header is not used anywhere in deployment. There is no material advantage to using the Authentication Header instead of ESP even if the use case is just authentication. The prevalence of NAT and checksum offload that break the Authentication Header severely limit the environments in which AH could be productively deployed.

## 4. Updates to RFC8200

[RFC8200] is updated to remove references to the Authentication header.

The following text replaces the list and the note following the list of Section 4 of [RFC8200].

OLD (RFC8200)

| Hop-by-Hop Options
| Fragment
| Destination Options
| Routing
| Authentication
| Encapsulating Security Payload
|
| The first four are specified in this document; the last two are
| specified in [RFC4302] and [RFC4303], respectively. The
| current list of IPv6 extension headers can be found at [IANA-
| EH].

NEW:

| Hop-by-Hop Options
| Fragment
| Destination Options
| Routing
| Encapsulating Security Payload
|

```
|     The first four are specified in this document; the last one is
|     specified in [RFC4303].  The current list of IPv6 extension
|     headers can be found at [IANA-EH].
```

The following text replaces the list and the paragraph following the list of Section 4.1 of [RFC8200].

OLD (RFC8200)

```
|     IPv6 header
|     Hop-by-Hop Options header
|     Destination Options header (note 1)
|     Routing header
|     Fragment header
|     Authentication header (note 2)
|     Encapsulating Security Payload header (note 2)
|     Destination Options header (note 3)
|     Upper-Layer header
|
|     note 1:  for options to be processed by the first destination
|              that appears in the IPv6 Destination Address field
|              plus subsequent destinations listed in the Routing
|              header.
|
|     note 2:  additional recommendations regarding the relative
|              order of the Authentication and Encapsulating Security
|              Payload headers are given in [RFC4303].
|
|     note 3:  for options to be processed only by the final
|              destination of the packet.
```

NEW:

```
|     IPv6 header
|     Hop-by-Hop Options header
|     Destination Options header (note 1)
|     Routing header
|     Fragment header
|     Encapsulating Security Payload header
|     Destination Options header (note 2)
|     Upper-Layer header
|
|     note 1:  for options to be processed by the first destination
|              that appears in the IPv6 Destination Address field
|              plus subsequent destinations listed in the Routing
|              header.
|
|     note 2:  for options to be processed only by the final
```

| destination of the packet.

The following text replaces the fourth paragraph of Section 4.2 of [RFC8200].

OLD (RFC8200)

| The third-highest-order bit of the Option Type specifies
| whether or not the Option Data of that option can change en
| route to the packet's final destination. When an
| Authentication header is present in the packet, for any option
| whose data may change en route, its entire Option Data field
| must be treated as zero-valued octets when computing or
| verifying the packet's authenticating value.

NEW:

| The third-highest-order bit of the Option Type specifies
| whether or not the Option Data of that option can change en
| route to the packet's final destination.

The following text replaces the third paragraph after the list and notes of Section 4.6 of [RFC8200].

OLD (RFC8200)

| Note that there are two possible ways to encode optional
| destination information in an IPv6 packet: either as an option
| in the Destination Options header or as a separate extension
| header. The Fragment header and the Authentication header are
| examples of the latter approach. Which approach can be used
| depends on what action is desired of a destination node that
| does not understand the optional information:

NEW:

| Note that there are two possible ways to encode optional
| destination information in an IPv6 packet: either as an option
| in the Destination Options header or as a separate extension
| header. The Fragment is an example of the latter approach.
| Which approach can be used depends on what action is desired of
| a destination node that does not understand the optional
| information:

The following text replaces the first paragraph of Section 8.4 of [RFC8200].

OLD (RFC8200)

| When an upper-layer protocol sends one or more packets in
| response to a received packet that included a Routing header,
| the response packet(s) must not include a Routing header that
| was automatically derived by "reversing" the received Routing
| header UNLESS the integrity and authenticity of the received
| Source Address and Routing header have been verified (e.g., via
| the use of an Authentication header in the received packet).
| In other words, only the following kinds of packets are
| permitted in response to a received packet bearing a Routing
| header:

NEW:

| When an upper-layer protocol sends one or more packets in
| response to a received packet that included a Routing header,
| the response packet(s) must not include a Routing header that
| was automatically derived by "reversing" the received Routing
| header UNLESS the integrity and authenticity of the received
| Source Address and Routing header have been verified. In other
| words, only the following kinds of packets are permitted in
| response to a received packet bearing a Routing header:

The following reference should be removed from Section 10.2 of [RFC8200].

REMOVE (from RFC8200):

| [RFC4302] Kent, S., "IP Authentication Header", RFC 4302, DOI
| 10.17487/RFC4302, December 2005, <http://www.rfc-
| editor.org/info/rfc4302\>.

## 5. Updates to RFC4303

[RFC4303] is updated to remove references to the Authentication header.

The following text replaces the first paragraph of Section 1 of [RFC4303].

OLD (RFC4303)

| This document assumes that the reader is familiar with the
| terms and concepts described in the "Security Architecture for
| the Internet Protocol" [Ken-Arch], hereafter referred to as the
| Security Architecture document. In particular, the reader
| should be familiar with the definitions of security services
| offered by the Encapsulating Security Payload (ESP) and the IP
| Authentication Header (AH), the concept of Security

| Associations, the ways in which ESP can be used in conjunction
| with AH, and the different key management options available for
| ESP and AH.

NEW

| This document assumes that the reader is familiar with the
| terms and concepts described in the "Security Architecture for
| the Internet Protocol" [Ken-Arch], hereafter referred to as the
| Security Architecture document. In particular, the reader
| should be familiar with the definitions of security services
| offered by the Encapsulating Security Payload (ESP), the
| concept of Security Associations, and the different key
| management options available for ESP.

The following text replaces the third paragraph of Section 1 of [RFC4303].

OLD (RFC4303)

| The Encapsulating Security Payload (ESP) header is designed to
| provide a mix of security services in IPv4 and IPv6 [DH98].
| ESP may be applied alone, in combination with AH [Ken-AH], or
| in a nested fashion (see the Security Architecture document
| [Ken-Arch]). Security services can be provided between a pair
| of communicating hosts, between a pair of communicating
| security gateways, or between a security gateway and a host.
| For more details on how to use ESP and AH in various network
| environments, see the Security Architecture document [Ken-
| Arch].

NEW

| The Encapsulating Security Payload (ESP) header is designed to
| provide a mix of security services in IPv4 and IPv6 [DH98].
| Security services can be provided between a pair of
| communicating hosts, between a pair of communicating security
| gateways, or between a security gateway and a host. For more
| details on how to use ESP in various network environments, see
| the Security Architecture document [Ken-Arch].

The following text replaces the sixth paragraph of Section 1 of [RFC4303].

OLD (RFC4303)

| Using encryption-only for confidentiality is allowed by ESP.
| However, it should be noted that in general, this will provide
| defense only against passive attackers. Using encryption
| without a strong integrity mechanism on top of it (either in
| ESP or separately via AH) may render the confidentiality
| service insecure against some forms of active attacks [Bel96,
| Kra01]. Moreover, an underlying integrity service, such as AH,
| applied before encryption does not necessarily protect the
| encryption-only confidentiality against active attackers
| [Kra01]. ESP allows encryption-only SAs because this may offer
| considerably better performance and still provide adequate
| security, e.g., when higher-layer authentication/integrity
| protection is offered independently. However, this standard
| does not require ESP implementations to offer an encryption-
| only service.

NEW

| Using encryption-only for confidentiality is allowed by ESP.
| However, it should be noted that in general, this will provide
| defense only against passive attackers. Using encryption
| without a strong integrity mechanism on top of it (either in
| ESP or separately via another protocol) may render the
| confidentiality service insecure against some forms of active
| attacks [Bel96, Kra01]. Moreover, an underlying integrity
| service applied before encryption does not necessarily protect
| the encryption-only confidentiality against active attackers
| [Kra01]. ESP allows encryption-only SAs because this may offer
| considerably better performance and still provide adequate
| security, e.g., when higher-layer authentication/integrity
| protection is offered independently. However, this standard
| does not require ESP implementations to offer an encryption-
| only service.

The following text replaces the sixth paragraph of Section 1 of [RFC4303].

OLD (RFC4303)

| Data origin authentication and connectionless integrity are
| joint services, hereafter referred to jointly as "integrity".
| (This term is employed because, on a per-packet basis, the
| computation being performed provides connectionless integrity
| directly; data origin authentication is provided indirectly as
| a result of binding the key used to verify the integrity to the
| identity of the IPsec peer. Typically, this binding is
| effected through the use of a shared, symmetric key.)
| Integrity-only ESP MUST be offered as a service selection

| option, e.g., it must be negotiable in SA management protocols
| and MUST be configurable via management interfaces. Integrity-
| only ESP is an attractive alternative to AH in many contexts,
| e.g., because it is faster to process and more amenable to
| pipelining in many implementations.

NEW

| Data origin authentication and connectionless integrity are
| joint services, hereafter referred to jointly as "integrity".
| (This term is employed because, on a per-packet basis, the
| computation being performed provides connectionless integrity
| directly; data origin authentication is provided indirectly as
| a result of binding the key used to verify the integrity to the
| identity of the IPsec peer. Typically, this binding is
| effected through the use of a shared, symmetric key.)
| Integrity-only ESP MUST be offered as a service selection
| option, e.g., it must be negotiable in SA management protocols
| and MUST be configurable via management interfaces.

The following text replaces the third list item of Section 2.1 of [RFC4303].

OLD (RFC4303)

| 3: Search the SAD for a match on only {SPI} if the receiver
| has chosen to maintain a single SPI space for AH and ESP,
| or on {SPI, protocol} otherwise. If an SAD entry matches,
| then process the inbound ESP packet with that matching SAD
| entry. Otherwise, discard the packet and log an auditable
| event.

NEW

| 3: Search the SAD for a match on {SPI, protocol}. If an SAD
| entry matches, then process the inbound ESP packet with
| that matching SAD entry. Otherwise, discard the packet and
| log an auditable event.

The following text replaces the first paragraph of Section 3.1.1 of [RFC4303].

OLD (RFC4303)

| In transport mode, ESP is inserted after the IP header and
| before a next layer protocol, e.g., TCP, UDP, ICMP, etc. In
| the context of IPv4, this translates to placing ESP after the
| IP header (and any options that it contains), but before the

| next layer protocol. (If AH is also applied to a packet, it is
| applied to the ESP header, Payload, ESP trailer, and ICV, if
| present.) (Note that the term "transport" mode should not be
| misconstrued as restricting its use to TCP and UDP.) The
| following diagram illustrates ESP transport mode positioning
| for a typical IPv4 packet, on a "before and after" basis.
| (This and subsequent diagrams in this section show the ICV
| field, the presence of which is a function of the security
| services and the algorithm/mode selected.)

NEW

| In transport mode, ESP is inserted after the IP header and
| before a next layer protocol, e.g., TCP, UDP, ICMP, etc. In
| the context of IPv4, this translates to placing ESP after the
| IP header (and any options that it contains), but before the
| next layer protocol. (Note that the term "transport" mode
| should not be misconstrued as restricting its use to TCP and
| UDP.) The following diagram illustrates ESP transport mode
| positioning for a typical IPv4 packet, on a "before and after"
| basis. (This and subsequent diagrams in this section show the
| ICV field, the presence of which is a function of the security
| services and the algorithm/mode selected.)

The following reference should be removed from Section 10.2 of [RFC4303].

REMOVE

| [Ken-AH] Kent, S., "IP Authentication Header", RFC 4302,
| December 2005.

## 6. IANA Considerations

IANA is requested to mark "Authentication Header" in the "Protocol Numbers registry [IANA-PROTO-NUMS] as "Deprecated", and to mark "Authentication Header" in the "Internet Protocol Version 6 (IPv6) Parameters" registry [IANA-IPV6-PARAMS] as "Deprecated".

## 7. Security Considerations

This document deprecates the IP Authentication Header which is in itself a security protocol. The Authentication Header offers weaker security than alternative protocols and is not known to be deployed. Overall deprecation of the Authentication Header does not weaken security of Internet protocols and does not create any new security concerns.

# 8. References

## 8.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, DOI 10.17487/RFC2119, March 1997, <https://www.rfc-editor.org/info/rfc2119>.

[RFC2663] Srisuresh, P. and M. Holdrege, "IP Network Address Translator (NAT) Terminology and Considerations", RFC 2663, DOI 10.17487/RFC2663, August 1999, <https://www.rfc-editor.org/info/rfc2663>.

[RFC4302] Kent, S., "IP Authentication Header", RFC 4302, DOI 10.17487/RFC4302, December 2005, <https://www.rfc-editor.org/info/rfc4302>.

[RFC4303] Kent, S., "IP Encapsulating Security Payload (ESP)", RFC 4303, DOI 10.17487/RFC4303, December 2005, <https://www.rfc-editor.org/info/rfc4303>.

[RFC5840] Grewal, K., Montenegro, G., and M. Bhatia, "Wrapped Encapsulating Security Payload (ESP) for Traffic Visibility", RFC 5840, DOI 10.17487/RFC5840, April 2010, <https://www.rfc-editor.org/info/rfc5840>.

[RFC8174] Leiba, B., "Ambiguity of Uppercase vs Lowercase in RFC 2119 Key Words", BCP 14, RFC 8174, DOI 10.17487/RFC8174, May 2017, <https://www.rfc-editor.org/info/rfc8174>.

[RFC8200] Deering, S. and R. Hinden, "Internet Protocol, Version 6 (IPv6) Specification", STD 86, RFC 8200, DOI 10.17487/RFC8200, July 2017, <https://www.rfc-editor.org/info/rfc8200>.

[RFC8221] Wouters, P., Migault, D., Mattsson, J., Nir, Y., and T. Kivinen, "Cryptographic Algorithm Implementation Requirements and Usage Guidance for Encapsulating Security Payload (ESP) and Authentication Header (AH)", RFC 8221, DOI 10.17487/RFC8221, October 2017, <https://www.rfc-editor.org/info/rfc8221>.

## 8.2. Informative References

[ah-avoid] Bhatia, M., "Avoiding Authentication Header (AH)", December 2011, <https://www.ietf.org/archive/id/draft-bhatia-ipsecme-avoiding-ah-00.txt>.

[ah-to-hist]
Bhatia, M., "Moving Authentication Header (AH) to Historic", December 2011, <https://https://www.ietf.org/archive/id/draft-bhatia-moving-ah-to-historic-00.txt>.

[depeh] Herbert, T., "Infight Removal of IPv6 Hop-by-Hop and Routing Headers", February 2024, <https://www.ietf.org/archive/id/draft-herbert-deprecate-eh-00.txt>.

[IANA-IPV6-PARAMS]
"Internet Protocol Version 6 (IPv6) Parameters: IPv6 Extension Header Types"", <https://www.iana.org/assignments/ipv6-parameters/ipv6-parameters.xhtml>.

[IANA-PROTO-NUMS]
"Protocol Numbers", <https://www.iana.org/assignments/protocol-numbers/protocol-numbers.xhtml>.

[inflrm] Herbert, T., "Infight Removal of IPv6 Hop-by-Hop and Routing Headers", February 2024, <https://www.ietf.org/archive/id/draft-herbert-eh-inflight-removal-04.txt>.

[RFC8754] Filsfils, C., Ed., Dukes, D., Ed., Previdi, S., Leddy, J., Matsushima, S., and D. Voyer, "IPv6 Segment Routing Header (SRH)", RFC 8754, DOI 10.17487/RFC8754, March 2020, <https://www.rfc-editor.org/info/rfc8754>.

## Author’s Address

Tom Herbert
XDPnet
Los Gatos, CA,
United States of America
Email: tom@herbertland.com