IPSec（**I**nternet**P**rotocol**Sec**urity）

“Internet 协议安全性 (IPSec)”是一种开放标准的框架结构，通过使用加密的[安全服务](https://baike.baidu.com/item/%E5%AE%89%E5%85%A8%E6%9C%8D%E5%8A%A1" \t "_blank)以确保在 Internet 协议 (IP) 网络上进行保密而安全的通讯。Microsoft® Windows® 2000、Windows XP 和 Windows Server 2003 家族实施 IPSec 是基于“Internet 工程任务组 (IETF)”IPSec [工作组](https://baike.baidu.com/item/%E5%B7%A5%E4%BD%9C%E7%BB%84" \t "_blank)开发的标准。

IPSec（**I**nternet**P**rotocol**Sec**urity）是安全联网的长期方向。它通过[端对端](https://baike.baidu.com/item/%E7%AB%AF%E5%AF%B9%E7%AB%AF" \t "_blank)的安全性来提供主动的保护以防止[专用网络](https://baike.baidu.com/item/%E4%B8%93%E7%94%A8%E7%BD%91%E7%BB%9C" \t "_blank)与 Internet 的攻击。在通信中，只有发送方和接收方才是唯一必须了解 IPSec 保护的计算机。在 Windows 2000、Windows XP 和 Windows Server 2003 家族中，IPSec 提供了一种能力，以保护[工作组](https://baike.baidu.com/item/%E5%B7%A5%E4%BD%9C%E7%BB%84" \t "_blank)、局域网计算机、域[客户端](https://baike.baidu.com/item/%E5%AE%A2%E6%88%B7%E7%AB%AF" \t "_blank)和服务器、分支机构（物理上为远程机构）、Extranet 以及漫游客户端之间的通信。

IPSec是IETF（Internet Engineering Task Force，Internet工程任务组）的IPSec小组建立的一组IP安全协议集。IPSec定义了在网际层使用的安全服务，其功能包括数据加密、对网络单元的访问控制、数据源地址验证、数据完整性检查和防止重放攻击。

IPSec的安全服务要求支持共享密钥完成认证和/或保密，并且手工输入密钥的方式是必须要支持的，其目的是要保证IPSec协议的互操作性。当然，手工输入密钥方式的扩展能力很差，因此在IPSec协议中引入了一个密钥管理协议，称Internet密钥交换协议——IKE，该协议可以动态认证IPSec对等体，协商安全服务，并自动生成共享密钥。

**中文名**

Internet 协议安全性

**外文名**

ipsec

**本    质**

开放标准的框架结构

**作    用**

保障网络安全

目录

1. 1 [作用目标](https://baike.baidu.com/item/ipsec/2472311?fr=aladdin" \l "1)
2. 2 [安全结构](https://baike.baidu.com/item/ipsec/2472311?fr=aladdin" \l "2)
3. ▪ [安全协议](https://baike.baidu.com/item/ipsec/2472311?fr=aladdin" \l "2_1)
4. ▪ [安全联盟 SA](https://baike.baidu.com/item/ipsec/2472311?fr=aladdin" \l "2_2)
5. ▪ [密钥管理协议](https://baike.baidu.com/item/ipsec/2472311?fr=aladdin" \l "2_3)
6. 3 [封装模式](https://baike.baidu.com/item/ipsec/2472311?fr=aladdin" \l "3)
7. 4 [常见问题](https://baike.baidu.com/item/ipsec/2472311?fr=aladdin" \l "4)
8. ▪ [安全特性](https://baike.baidu.com/item/ipsec/2472311?fr=aladdin" \l "4_1)
9. ▪ [公钥认证](https://baike.baidu.com/item/ipsec/2472311?fr=aladdin" \l "4_2)
10. ▪ [预置共享密钥](https://baike.baidu.com/item/ipsec/2472311?fr=aladdin" \l "4_3)
11. ▪ [公钥加密](https://baike.baidu.com/item/ipsec/2472311?fr=aladdin" \l "4_4)
12. ▪ [Hash函数](https://baike.baidu.com/item/ipsec/2472311?fr=aladdin" \l "4_5)
13. ▪ [数据可靠性](https://baike.baidu.com/item/ipsec/2472311?fr=aladdin" \l "4_6)
14. ▪ [密钥管理](https://baike.baidu.com/item/ipsec/2472311?fr=aladdin" \l "4_7)
15. 5 [数据包](https://baike.baidu.com/item/ipsec/2472311?fr=aladdin" \l "5)
16. ▪ [协议结构](https://baike.baidu.com/item/ipsec/2472311?fr=aladdin" \l "5_1)
17. ▪ [ESP和AH](https://baike.baidu.com/item/ipsec/2472311?fr=aladdin" \l "5_2)
18. 6 [攻击防御](https://baike.baidu.com/item/ipsec/2472311?fr=aladdin" \l "6)
19. ▪ [攻击方式](https://baike.baidu.com/item/ipsec/2472311?fr=aladdin" \l "6_1)
20. ▪ [防御策略](https://baike.baidu.com/item/ipsec/2472311?fr=aladdin" \l "6_2)

作用目标

随着物联网设备不断融入人们的日常生活，物联网设备与互联网的连接就成为不可缺少的条件。那么物联网设备接入互联网时，又如何确保网络通讯的安全呢？而早已演进成熟的IPSec（Internet Protocol Security，Internet 协议安全性）就成为众多物联网设备确保网络通讯安全的选择。IPSec有如下两条。

1.保护 IP 数据包的内容。

2.通过[数据包](https://baike.baidu.com/item/%E6%95%B0%E6%8D%AE%E5%8C%85" \t "_blank)筛选及受信任通讯的实施来防御[网络攻击](https://baike.baidu.com/item/%E7%BD%91%E7%BB%9C%E6%94%BB%E5%87%BB" \t "_blank)。

这两个目标都是通过使用基于加密的保护服务、安全协议与动态[密钥管理](https://baike.baidu.com/item/%E5%AF%86%E9%92%A5%E7%AE%A1%E7%90%86" \t "_blank)来实现的。这个基础为专用[网络计算机](https://baike.baidu.com/item/%E7%BD%91%E7%BB%9C%E8%AE%A1%E7%AE%97%E6%9C%BA" \t "_blank)、域、站点、[远程站点](https://baike.baidu.com/item/%E8%BF%9C%E7%A8%8B%E7%AB%99%E7%82%B9" \t "_blank)、Extranet 和拨号用户之间的通信提供了既有力又灵活的保护。它甚至可以用来阻碍特定通讯类型的接收和发送。

其中以接收和发送最为重要。

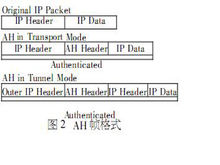
安全结构

IPsec协议工作在OSI 模型的第三层，使其在单独使用时适于保护基于[TCP](https://baike.baidu.com/item/TCP" \t "_blank)或[UDP](https://baike.baidu.com/item/UDP" \t "_blank)的协议（如 安全套接子层（[SSL](https://baike.baidu.com/item/SSL" \t "_blank)）就不能保护UDP层的通信流）。这就意味着，与传输层或更高层的协议相比，IPsec协议必须处理可靠性和分片的问题，这同时也增加了它的复杂性和处理开销。相对而言，[SSL](https://baike.baidu.com/item/SSL" \t "_blank)/[TLS](https://baike.baidu.com/item/TLS" \t "_blank)依靠更高层的[TCP](https://baike.baidu.com/item/TCP" \t "_blank)（OSI的第四层）来管理可靠性和分片。

安全协议

(1)AH(AuthenticationHeader) 协议。

它用来向 IP通信提供数据完整性和身份验证,同时可以提供抗重播服务。

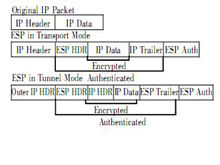
[](https://baike.baidu.com/pic/ipsec/2472311/0/79f0f736afc37931bc14f0d6ebc4b74542a911d4?fr=lemma%26ct=single)

AH帧格式

在 IPv6 中协议采用 AH 后, 因为在主机端设置了一个基于算法独立交换的秘密钥匙, 非法潜入的现象可得到有效防止, 秘密钥匙由客户和服务商共同设置。在传送每个数据包时,IPv6 认证根据这个秘密钥匙和数据包产生一个检验项。在数据接收端重新运行该检验项并进行比较,从而保证了对数据包来源的确认以及数据包不被非法修改。

(2)ESP(EncapsulatedSecurityPayload) 协议。

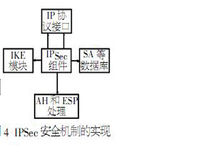
它提供 IP层加密保证和验证数据源以对付网络上的监听。因为 AH虽然可以保护通信免受篡改, 但并不对数据进行变形转换, 数据对于黑客而言仍然是清晰的。为了有效地保证数据传输安全, 在IPv6 中有另外一个报头 ESP,进一步提供数据保密性并防止篡改。

[](https://baike.baidu.com/pic/ipsec/2472311/0/279759ee3d6d55fb64081cd96d224f4a21a4dd07?fr=lemma%26ct=single)

ESP帧格式

安全联盟 SA

安全联盟 SA,记录每条 IP安全通路的策略和策略参数。安全联盟是 IPSec 的基础, 是通信双方建立的一种协定,决定了用来保护数据包的协议、转码方式、密钥以及密钥有效期等。AH和 ESP都要用到安全联盟,IKE的一个主要功能就是建立和维护安全联盟。

[](https://baike.baidu.com/pic/ipsec/2472311/0/09fa513d269759eee9940a34b2fb43166c22df0d?fr=lemma%26ct=single)

IPSec安全机制结构

密钥管理协议

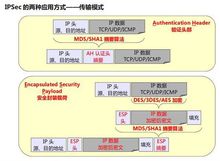
密钥管理协议 ISAKMP, 提供共享安全信息。Internet密钥管理协议被定义在应用层,IETF规定了 Internet安全协议和 ISAKMP(Internet Security Association and Key Management Protocol) 来实现 IPSec 的密钥管理,为身份认证的 SA 设置以及密钥交换技术

封装模式

隧道（tunnel）模式：用户的整个IP数据包被用来计算AH或ESP头，AH或ESP头以及ESP加密的用户数据被封装在一个新的IP数据包中。通常，隧道模式应用在两个安全网关之间的通讯。

[](https://baike.baidu.com/pic/ipsec/2472311/0/c75c10385343fbf21f5826ecb17eca8064388f8d?fr=lemma%26ct=single)

隧道模式封装结构

[](https://baike.baidu.com/pic/ipsec/2472311/0/9c16fdfaaf51f3de6ff939ab95eef01f3b2979b2?fr=lemma%26ct=single)

传输模式封装结构

传输（transport）模式：只是传输层数据被用来计算AH或ESP头，AH或ESP头以及ESP加密的用户数据被放置在原IP包头后面。通常，传输模式应用在两台主机之间的通讯，或一台主机和一个安全网关之间的通讯。定义了一个通用格式。

常见问题

IPSec 基于[端对端](https://baike.baidu.com/item/%E7%AB%AF%E5%AF%B9%E7%AB%AF" \t "_blank)的安全模式，在源 IP 和目标 IP 地址之间建立信任和安全性。考虑认为 IP 地址本身没有必要具有标识，但 IP 地址后面的系统必须有一个通过[身份验证](https://baike.baidu.com/item/%E8%BA%AB%E4%BB%BD%E9%AA%8C%E8%AF%81" \t "_blank)程序验证过的标识。只有发送和接收的计算机需要知道通讯是安全的。每台计算机都假定进行通讯的媒体不安全，因此在各自的[终端](https://baike.baidu.com/item/%E7%BB%88%E7%AB%AF" \t "_blank)上实施安全设置。除非两台计算机之间正在进行防火墙类型的数据包筛选或[网络地址转换](https://baike.baidu.com/item/%E7%BD%91%E7%BB%9C%E5%9C%B0%E5%9D%80%E8%BD%AC%E6%8D%A2" \t "_blank)，否则仅从源向目标[路由](https://baike.baidu.com/item/%E8%B7%AF%E7%94%B1" \t "_blank)数据的计算机不要求支持 IPSec。该模式允许为下列企业方案成功部署 IPSec：

局域网 (LAN)：[客户端](https://baike.baidu.com/item/%E5%AE%A2%E6%88%B7%E7%AB%AF" \t "_blank)/服务器和[对等网络](https://baike.baidu.com/item/%E5%AF%B9%E7%AD%89%E7%BD%91%E7%BB%9C" \t "_blank)

[广域网](https://baike.baidu.com/item/%E5%B9%BF%E5%9F%9F%E7%BD%91" \t "_blank)(WAN)：[路由器](https://baike.baidu.com/item/%E8%B7%AF%E7%94%B1%E5%99%A8" \t "_blank)到路由器和[网关](https://baike.baidu.com/item/%E7%BD%91%E5%85%B3" \t "_blank)到网关

[远程访问](https://baike.baidu.com/item/%E8%BF%9C%E7%A8%8B%E8%AE%BF%E9%97%AE" \t "_blank)：拨号客户机和从[专用网络](https://baike.baidu.com/item/%E4%B8%93%E7%94%A8%E7%BD%91%E7%BB%9C" \t "_blank)访问 Internet

通常，两端都需要 IPSec 配置（称为 IPSec 策略）来设置选项与安全设置，以允许两个系统对如何保护它们之间的通讯达成协议。Windows 2000、Windows XP 和 Windows Server 2003 家族实施 IPSec 是基于“Internet 工程任务组 (IETF)”IPSec[工作组](https://baike.baidu.com/item/%E5%B7%A5%E4%BD%9C%E7%BB%84" \t "_blank)开发的业界标准。IPSec 相关服务部分是由 Microsoft 与 Cisco Systems, Inc. 共同开发的。

IPSec 协议不是一个单独的协议，它给出了应用于IP层上网络[数据安全](https://baike.baidu.com/item/%E6%95%B0%E6%8D%AE%E5%AE%89%E5%85%A8" \t "_blank)的一整套体系结构，包括网络认证协议 Authentication Header（AH）、封装安全载荷协议Encapsulating Security Payload（ESP）、[密钥](https://baike.baidu.com/item/%E5%AF%86%E9%92%A5" \t "_blank)管理协议Internet Key Exchange （IKE）和用于网络认证及加密的一些算法等。这些协议用于提供数据认证、数据完整性和加密性三种保护形式。AH和ESP都可以提供认证服务，但AH提供的认证服务要强于ESP。而IKE主要是对密钥进行交换管理，对算法、协议和密钥3个方面进行协商。

IPSec 规定了如何在[对等层](https://baike.baidu.com/item/%E5%AF%B9%E7%AD%89%E5%B1%82" \t "_blank)之间选择安全协议、确定安全算法和[密钥](https://baike.baidu.com/item/%E5%AF%86%E9%92%A5" \t "_blank)交换，向上提供了[访问控制](https://baike.baidu.com/item/%E8%AE%BF%E9%97%AE%E6%8E%A7%E5%88%B6" \t "_blank)、数据源认证、[数据加密](https://baike.baidu.com/item/%E6%95%B0%E6%8D%AE%E5%8A%A0%E5%AF%86" \t "_blank)等[网络安全](https://baike.baidu.com/item/%E7%BD%91%E7%BB%9C%E5%AE%89%E5%85%A8" \t "_blank)服务。

安全特性

IPSec的安全特性主要有：

**·不可否认性**

"不可否认性"可以证实消息发送方是唯一可能的发送者，发送者不能否认发送过消息。"不可否认性"是采用公钥技术的一个特征，当使用公钥技术时，发送方用[私钥](https://baike.baidu.com/item/%E7%A7%81%E9%92%A5" \t "_blank)产生一个[数字签名](https://baike.baidu.com/item/%E6%95%B0%E5%AD%97%E7%AD%BE%E5%90%8D" \t "_blank)随消息一起发送，接收方用发送者的公钥来验证数字签名。由于在理论上只有发送者才唯一拥有[私钥](https://baike.baidu.com/item/%E7%A7%81%E9%92%A5" \t "_blank)，也只有发送者才可能产生该[数字签名](https://baike.baidu.com/item/%E6%95%B0%E5%AD%97%E7%AD%BE%E5%90%8D" \t "_blank)，所以只要数字签名通过验证，发送者就不能否认曾发送过该消息。但"不可否认性"不是基于认证的共享[密钥](https://baike.baidu.com/item/%E5%AF%86%E9%92%A5" \t "_blank)技术的特征，因为在基于认证的共享密钥技术中，发送方和接收方掌握相同的密钥。

**·反重播性**

"反重播"确保每个IP包的唯一性，保证信息万一被截取复制后，不能再被重新利用、重新传输回目的地址。该特性可以防止攻击者截取破译信息后，再用相同的[信息包](https://baike.baidu.com/item/%E4%BF%A1%E6%81%AF%E5%8C%85" \t "_blank)冒取非法访问权（即使这种冒取行为发生在数月之后）。

**·数据完整性**

防止传输过程中数据被篡改，确保发出数据和接收数据的一致性。IPSec利用Hash函数为每个数据包产生一个加密检查和，接收方在打开包前先计算检查和，若包遭篡改导致检查和不相符，数据包即被丢弃。

**·数据可靠性（加密）**

在传输前，对数据进行加密，可以保证在传输过程中，即使数据包遭截取，信息也无法被读。该特性在IPSec中为可选项，与IPSec策略的具体设置相关。

**·认证**

数据源发送信任状，由接收方验证信任状的合法性，只有通过认证的系统才可以建立通信连接。

公钥认证

一个架构良好的公钥体系，在信任状的传递中不造成任何信息外泄，能解决很多安全问题。IPSec与特定的公钥体系相结合，可以提供基于电子证书的认证。[公钥证书](https://baike.baidu.com/item/%E5%85%AC%E9%92%A5%E8%AF%81%E4%B9%A6" \t "_blank)认证在Windows 2000中，适用于对非Windows 2000[主机](https://baike.baidu.com/item/%E4%B8%BB%E6%9C%BA" \t "_blank)、独立主机，非信任域成员的客户机、或者不运行Kerberos v5认证协议的主机进行[身份认证](https://baike.baidu.com/item/%E8%BA%AB%E4%BB%BD%E8%AE%A4%E8%AF%81" \t "_blank)。

预置共享密钥

IPSec也可以使用预置共享[密钥](https://baike.baidu.com/item/%E5%AF%86%E9%92%A5" \t "_blank)进行认证。预共享意味着通信双方必须在IPSec策略设置中就共享的[密钥](https://baike.baidu.com/item/%E5%AF%86%E9%92%A5" \t "_blank)达成一致。之后在安全协商过程中，信息在传输前使用共享[密钥加密](https://baike.baidu.com/item/%E5%AF%86%E9%92%A5%E5%8A%A0%E5%AF%86" \t "_blank)，接收端使用同样的密钥解密，如果接收方能够解密，即被认为可以通过认证。但在Windows 2000 IPSec策略中，这种认证方式被认为不够安全而一般不推荐使用。

公钥加密

IPSec的公钥加密用于[身份认证](https://baike.baidu.com/item/%E8%BA%AB%E4%BB%BD%E8%AE%A4%E8%AF%81" \t "_blank)和[密钥](https://baike.baidu.com/item/%E5%AF%86%E9%92%A5" \t "_blank)交换。公钥加密，也被称为"不对称加密法"，即加解密过程需要两把不同的[密钥](https://baike.baidu.com/item/%E5%AF%86%E9%92%A5" \t "_blank)，一把用来产生[数字签名](https://baike.baidu.com/item/%E6%95%B0%E5%AD%97%E7%AD%BE%E5%90%8D" \t "_blank)和加密数据，另一把用来验证数字签名和对数据进行解密。

使用[公钥](https://baike.baidu.com/item/%E5%85%AC%E9%92%A5" \t "_blank)加密法，每个用户拥有一个[密钥](https://baike.baidu.com/item/%E5%AF%86%E9%92%A5" \t "_blank)对，其中[私钥](https://baike.baidu.com/item/%E7%A7%81%E9%92%A5" \t "_blank)仅为其个人所知，公钥则可分发给任意需要与之进行加密通信的人。例如：A想要发送加密信息给B，则A需要用B的公钥加密信息，之后只有B才能用他的[私钥](https://baike.baidu.com/item/%E7%A7%81%E9%92%A5" \t "_blank)对该加密信息进行解密。虽然[密钥](https://baike.baidu.com/item/%E5%AF%86%E9%92%A5" \t "_blank)对中两把钥匙彼此相关，但要想从其中一把来推导出另一把，以目前计算机的运算能力来看，这种做法几乎完全不现实。因此，在这种加密法中，[公钥](https://baike.baidu.com/item/%E5%85%AC%E9%92%A5" \t "_blank)可以广为分发，而[私钥](https://baike.baidu.com/item/%E7%A7%81%E9%92%A5" \t "_blank)则需要仔细地妥善保管。

Hash函数

Hash信息验证码HMAC（Hash message authentication codes）验证接收消息和发送消息的完全一致性（完整性）。这在数据交换中非常关键，尤其当传输媒介如公共网络中不提供安全保证时更显其重要性。

HMAC结合[hash算法](https://baike.baidu.com/item/hash%E7%AE%97%E6%B3%95" \t "_blank)和共享[密钥](https://baike.baidu.com/item/%E5%AF%86%E9%92%A5" \t "_blank)提供完整性。Hash散列通常也被当成是[数字签名](https://baike.baidu.com/item/%E6%95%B0%E5%AD%97%E7%AD%BE%E5%90%8D" \t "_blank)，但这种说法不够准确，两者的区别在于：Hash散列使用共享[密钥](https://baike.baidu.com/item/%E5%AF%86%E9%92%A5" \t "_blank)，而数字签名基于[公钥](https://baike.baidu.com/item/%E5%85%AC%E9%92%A5" \t "_blank)技术。[hash算法](https://baike.baidu.com/item/hash%E7%AE%97%E6%B3%95" \t "_blank)也称为[消息摘要](https://baike.baidu.com/item/%E6%B6%88%E6%81%AF%E6%91%98%E8%A6%81" \t "_blank)或单向转换。称它为单向转换是因为：

1）双方必须在通信的两个端头处各自执行Hash函数计算；

2）使用Hash函数很容易从消息计算出[消息摘要](https://baike.baidu.com/item/%E6%B6%88%E6%81%AF%E6%91%98%E8%A6%81" \t "_blank)，但其逆向反演过程以目前计算机的运算能力几乎不可实现。

Hash散列本身就是所谓加密检查和或消息完整性编码MIC（Message Integrity Code），通信双方必须各自执行函数计算来验证消息。举例来说，发送方首先使用HMAC算法和共享[密钥](https://baike.baidu.com/item/%E5%AF%86%E9%92%A5" \t "_blank)计算消息检查和，然后将计算结果A封装进数据包中一起发送；接收方再对所接收的消息执行HMAC计算得出结果B，并将B与A进行比较。如果消息在传输中遭篡改致使B与A不一致，接收方丢弃该数据包。

有两种最常用的[hash函数](https://baike.baidu.com/item/hash%E5%87%BD%E6%95%B0" \t "_blank)：

·HMAC-MD5 MD5（[消息摘要](https://baike.baidu.com/item/%E6%B6%88%E6%81%AF%E6%91%98%E8%A6%81" \t "_blank)5）基于RFC1321。MD5对MD4做了改进，计算速度比MD4稍慢，但安全性能得到了进一步改善。MD5在计算中使用了64个32位常数，最终生成一个128位的完整性检查和。

·HMAC-SHA 安全[Hash算法](https://baike.baidu.com/item/Hash%E7%AE%97%E6%B3%95" \t "_blank)定义在NIST FIPS 180-1，其算法以MD5为原型。 SHA在计算中使用了79个32位常数，最终产生一个160位完整性检查和。SHA检查和长度比MD5更长，因此安全性也更高。

数据可靠性

IPSec使用的[数据加密算法](https://baike.baidu.com/item/%E6%95%B0%E6%8D%AE%E5%8A%A0%E5%AF%86%E7%AE%97%E6%B3%95" \t "_blank)是DES--Data Encryption Standard（[数据加密标准](https://baike.baidu.com/item/%E6%95%B0%E6%8D%AE%E5%8A%A0%E5%AF%86%E6%A0%87%E5%87%86" \t "_blank)）。DES[密钥](https://baike.baidu.com/item/%E5%AF%86%E9%92%A5" \t "_blank)长度为56位，在形式上是一个64位数。DES以64位（8字节）为分组对[数据加密](https://baike.baidu.com/item/%E6%95%B0%E6%8D%AE%E5%8A%A0%E5%AF%86" \t "_blank)，每64位明文，经过16轮置换生成64位密文，其中每字节有1位用于[奇偶校验](https://baike.baidu.com/item/%E5%A5%87%E5%81%B6%E6%A0%A1%E9%AA%8C" \t "_blank)，所以实际有效[密钥](https://baike.baidu.com/item/%E5%AF%86%E9%92%A5" \t "_blank)长度是56位。 IPSec还支持3DES算法，3DES可提供更高的安全性，但相应地，计算速度更慢。

密钥管理

**·动态**[密钥](https://baike.baidu.com/item/%E5%AF%86%E9%92%A5" \t "_blank)更新

IPSec策略使用"动态密钥更新"法来决定在一次通信中，新密钥产生的频率。动态密钥指在通信过程中，数据流被划分成一个个"数据块"，每一个"数据块"都使用不同的[密钥加密](https://baike.baidu.com/item/%E5%AF%86%E9%92%A5%E5%8A%A0%E5%AF%86" \t "_blank)，这可以保证万一攻击者中途截取了部分通信数据流和相应的密钥后，也不会危及到所有其余的通信信息的安全。动态[密钥](https://baike.baidu.com/item/%E5%AF%86%E9%92%A5" \t "_blank)更新服务由Internet密钥交换IKE（Internet Key Exchange）提供，详见IKE介绍部分。

IPSec策略允许专家级用户自定义[密钥](https://baike.baidu.com/item/%E5%AF%86%E9%92%A5" \t "_blank)生命周期。如果该值没有设置，则按缺省时间间隔自动生成新[密钥](https://baike.baidu.com/item/%E5%AF%86%E9%92%A5" \t "_blank)。

**·**[密钥](https://baike.baidu.com/item/%E5%AF%86%E9%92%A5" \t "_blank)长度

密钥长度每增加一位，可能的密钥数就会增加一倍，相应地，破解密钥的难度也会随之成指数级加大。IPSec策略提供多种[加密算法](https://baike.baidu.com/item/%E5%8A%A0%E5%AF%86%E7%AE%97%E6%B3%95" \t "_blank)，可生成多种长度不等的[密钥](https://baike.baidu.com/item/%E5%AF%86%E9%92%A5" \t "_blank)，用户可根据不同的安全需求加以选择。

**·Diffie-Hellman算法**

要启动安全通讯，通信两端必须首先得到相同的共享[密钥](https://baike.baidu.com/item/%E5%AF%86%E9%92%A5" \t "_blank)（主密钥），但共享密钥不能通过网络相互发送，因为这种做法极易泄密。

[Diffie-Hellman](https://baike.baidu.com/item/Diffie-Hellman" \t "_blank)算法是用于[密钥](https://baike.baidu.com/item/%E5%AF%86%E9%92%A5" \t "_blank)交换的最早最安全的算法之一。DH算法的基本工作原理是：通信双方公开或半公开交换一些准备用来生成密钥的"材料数据"，在彼此交换过密钥生成"材料"后，两端可以各自生成出完全一样的共享密钥。在任何时候，双方都绝不交换真正的[密钥](https://baike.baidu.com/item/%E5%AF%86%E9%92%A5" \t "_blank)。

通信双方交换的密钥生成"材料"，长度不等，"材料"长度越长，所生成的密钥强度也就越高，密钥破译就越困难。 除进行[密钥](https://baike.baidu.com/item/%E5%AF%86%E9%92%A5" \t "_blank)交换外，IPSec还使用DH算法生成所有其他加密密钥。

AH报头字段包括：

·Next Header（下一个报头）： 识别下一个使用IP协议号的报头，例如，Next Header值等于"6"，表示紧接其后的是TCP报头。

·Length（长度）： AH报头长度。

·Security Parameters Index (SPI，安全参数索引)： 这是一个为数据报识别安全关联的 32 位伪随机值。SPI 值 0 被保留来表明"没有安全关联存在"。

·Sequence Number（[序列号](https://baike.baidu.com/item/%E5%BA%8F%E5%88%97%E5%8F%B7" \t "_blank)）：从1开始的32位单增序列号，不允许重复，唯一地标识了每一个发送数据包，为安全关联提供反重播保护。接收端校验[序列号](https://baike.baidu.com/item/%E5%BA%8F%E5%88%97%E5%8F%B7" \t "_blank)为该字段值的数据包是否已经被接收过，若是，则拒收该数据包。

·Authentication Data（AD，认证数据）： 包含完整性检查和。接收端接收数据包后，首先执行hash计算，再与发送端所计算的该字段值比较，若两者相等，表示数据完整，若在传输过程中数据遭修改，两个计算结果不一致，则丢弃该数据包。

数据包

如图二所示，AH报头插在IP报头之后，TCP，UDP，或者ICMP等上层协议报头之前。一般AH为整个[数据包](https://baike.baidu.com/item/%E6%95%B0%E6%8D%AE%E5%8C%85" \t "_blank)提供完整性检查，但如果IP报头中包含"生存期（Time To Live）"或"服务类型（Type of Service）"等值可变字段，则在进行完整性检查时应将这些值可变字段去除。

图2 AH为整个数据包提供完整性检查

协议结构

ESP（**Encapsulating Security Payload**）为IP[数据包](https://baike.baidu.com/item/%E6%95%B0%E6%8D%AE%E5%8C%85" \t "_blank)提供完整性检查、认证和加密，可以看作是"超级 AH"， 因为它提供机密性并可防止篡改。ESP服务依据建立的安全关联（SA）是可选的。然而，也有一些限制：

·完整性检查和认证一起进行。

·仅当与完整性检查和认证一起时，"重播（Replay）"保护才是可选的。

·"重播"保护只能由接收方选择。

ESP的加密服务是可选的，但如果启用加密，则也就同时选择了完整性检查和认证。因为如果仅使用加密，入侵者就可能伪造包以发动密码分析攻击。

ESP可以单独使用，也可以和AH结合使用。一般ESP不对整个数据包加密，而是只加密IP包的有效载荷部分，不包括IP头。但在[端对端](https://baike.baidu.com/item/%E7%AB%AF%E5%AF%B9%E7%AB%AF" \t "_blank)的隧道通信中，ESP需要对整个数据包加密。

如图三所示，ESP报头插在IP报头之后，TCP或UDP等[传输层](https://baike.baidu.com/item/%E4%BC%A0%E8%BE%93%E5%B1%82" \t "_blank)协议报头之前。ESP由IP协议号"50"标识。

图3 ESP报头、报尾和认证报尾

ESP报头字段包括：

·Security Parameters Index (SPI，安全参数索引)：为数据包识别安全关联。

·Sequence Number（序列号）：从1开始的32位单增序列号，不允许重复，唯一地标识了每一个发送数据包，为安全关联提供反重播保护。接收端校验序列号为该字段值的数据包是否已经被接收过，若是，则拒收该数据包。 ESP报尾字段包括：

·Padding（扩展位）：0-255个字节。DH算法要求数据长度（以位为单位）模512为448，若应用数据长度不足，则用扩展位填充。

·Padding Length（扩展位长度）：接收端根据该字段长度去除数据中扩展位。

·Next Header（下一个报头）：识别下一个使用IP协议号的报头，如TCP或UDP。

ESP认证报尾字段：

·Authentication Data（AD，认证数据）： 包含完整性检查和。完整性检查部分包括ESP报头、有效载荷（[应用程序](https://baike.baidu.com/item/%E5%BA%94%E7%94%A8%E7%A8%8B%E5%BA%8F" \t "_blank)数据）和ESP报尾。见图四。

图4 ESP的加密部分和完整性检查部分

上图所示，ESP报头的位置在IP报头之后，TCP，UDP，或者ICMP等[传输层](https://baike.baidu.com/item/%E4%BC%A0%E8%BE%93%E5%B1%82" \t "_blank)协议报头之前。如果已经有其他IPSec协议使用，则ESP报头应插在其他任何IPSec协议报头之前。ESP认证报尾的完整性检查部分包括ESP报头、[传输层](https://baike.baidu.com/item/%E4%BC%A0%E8%BE%93%E5%B1%82" \t "_blank)协议报头，应用数据和ESP报尾，但不包括IP报头，因此ESP不能保证IP报头不被篡改。ESP加密部分包括上层[传输协议](https://baike.baidu.com/item/%E4%BC%A0%E8%BE%93%E5%8D%8F%E8%AE%AE" \t "_blank)信息、数据和ESP报尾。

ESP和AH

以上介绍的是传输模式下的AH协议和ESP协议，ESP隧道模式和AH隧道模式与传输模式略有不同。

在[隧道模式](https://baike.baidu.com/item/%E9%9A%A7%E9%81%93%E6%A8%A1%E5%BC%8F" \t "_blank)下，整个原数据包被当作有效载荷[封装](https://baike.baidu.com/item/%E5%B0%81%E8%A3%85" \t "_blank)了起来，外面附上新的IP报头。其中"内部"IP报头（原IP报头）指定最终的信源和信宿地址，而"外部"IP报头（新IP报头）中包含的常常是做中间处理的安全[网关](https://baike.baidu.com/item/%E7%BD%91%E5%85%B3" \t "_blank)地址。

与传输模式不同，在[隧道模式](https://baike.baidu.com/item/%E9%9A%A7%E9%81%93%E6%A8%A1%E5%BC%8F" \t "_blank)中，原IP地址被当作有效载荷的一部分受到IPSec的安全保护，另外，通过对[数据加密](https://baike.baidu.com/item/%E6%95%B0%E6%8D%AE%E5%8A%A0%E5%AF%86" \t "_blank)，还可以将[数据包](https://baike.baidu.com/item/%E6%95%B0%E6%8D%AE%E5%8C%85" \t "_blank)目的地址隐藏起来，这样更有助于保护端对端隧道通信中数据的安全性。

ESP[隧道模式](https://baike.baidu.com/item/%E9%9A%A7%E9%81%93%E6%A8%A1%E5%BC%8F" \t "_blank)中签名部分（完整性检查和认证部分）和加密部分分别如图所示。ESP的签名不包括新IP头。

图5 ESP[隧道模式](https://baike.baidu.com/item/%E9%9A%A7%E9%81%93%E6%A8%A1%E5%BC%8F" \t "_blank)

下图标示出了AH[隧道模式](https://baike.baidu.com/item/%E9%9A%A7%E9%81%93%E6%A8%A1%E5%BC%8F" \t "_blank)中的签名部分。AH隧道模式为整个[数据包](https://baike.baidu.com/item/%E6%95%B0%E6%8D%AE%E5%8C%85" \t "_blank)提供完整性检查和认证，认证功能优于ESP。但在[隧道技术](https://baike.baidu.com/item/%E9%9A%A7%E9%81%93%E6%8A%80%E6%9C%AF" \t "_blank)中，AH协议很少单独实现，通常与ESP协议组合使用。

图6 AH隧道模式

攻击防御

IPv6提供了更广阔的[地址空间](https://baike.baidu.com/item/%E5%9C%B0%E5%9D%80%E7%A9%BA%E9%97%B4" \t "_blank)，但就安全性而言，IPv6并不比IPv4安全，甚至是某些攻击手段的天然土壤，比如DDos。

攻击方式

IPsec是IPv6引以为傲的[安全策略](https://baike.baidu.com/item/%E5%AE%89%E5%85%A8%E7%AD%96%E7%95%A5" \t "_blank)，下面我们说下这种策略带来的问题。最直接的，IPsec是端到端的加密，中间传输过程是透明的，如果窃听者没有对应[密钥](https://baike.baidu.com/item/%E5%AF%86%E9%92%A5" \t "_blank)，那么就无法解开数据内容；所以像路由[防火墙](https://baike.baidu.com/item/%E9%98%B2%E7%81%AB%E5%A2%99" \t "_blank)等基于[内容过滤](https://baike.baidu.com/item/%E5%86%85%E5%AE%B9%E8%BF%87%E6%BB%A4" \t "_blank)的[安全策略](https://baike.baidu.com/item/%E5%AE%89%E5%85%A8%E7%AD%96%E7%95%A5" \t "_blank)将无法存在；而如果通过某种方式，使得路由和防火墙能够解开数据包，那么IPsec的安全性也就不存在了。

更广阔的[地址空间](https://baike.baidu.com/item/%E5%9C%B0%E5%9D%80%E7%A9%BA%E9%97%B4" \t "_blank)导致更容易伪造源地址，也就更难以追踪。

利用IPv6的自定义扩展头，自定义路由路径，就可以定义复杂的路由路径，比如环路径等，进一步增加追踪难度并消耗更多路由资源。自定义扩展头，也导致更复杂的安全过滤。还可以使用极大的扩展头进行称为畸形炸弹的DDos攻击。

防御策略

目前主要防御思路有基于统计特征的防御、基于Netfilter的防御等。