امنیت شبکه

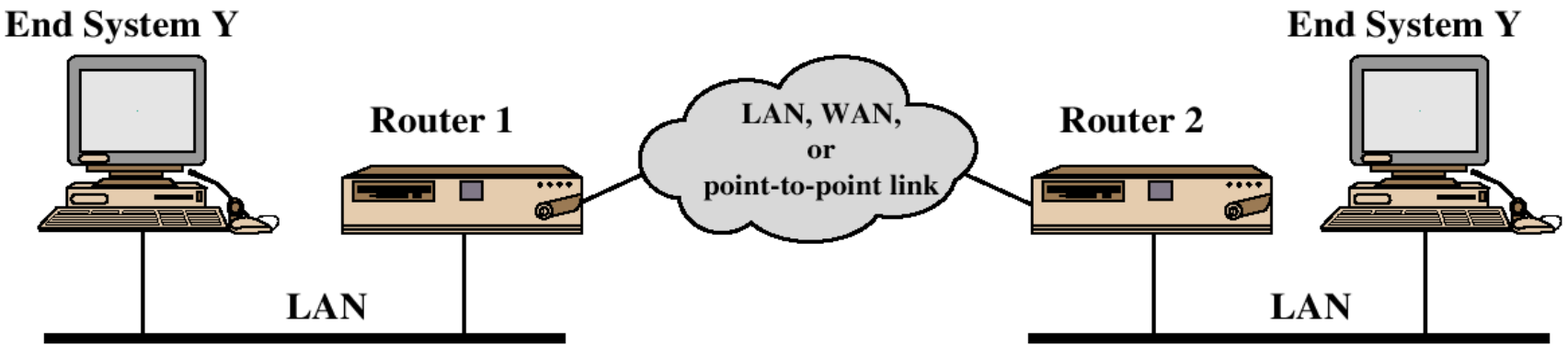
گردآوری: محمدعلی عظیمی

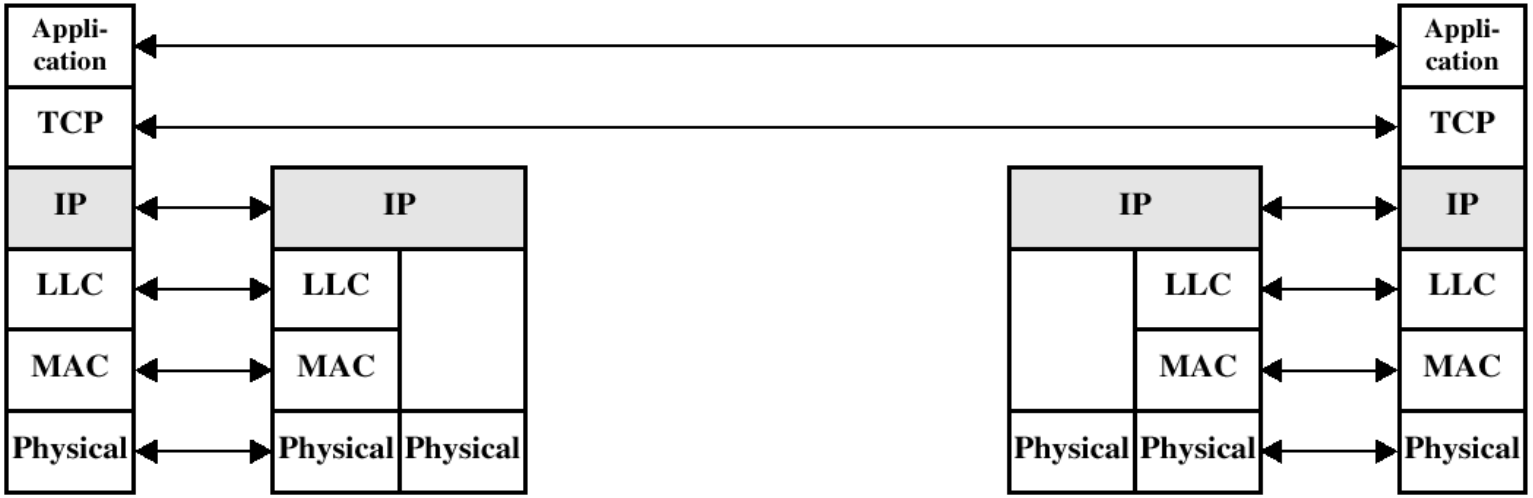
امنیت در لایه ها

گردآوری: محمدعلی عظیمی

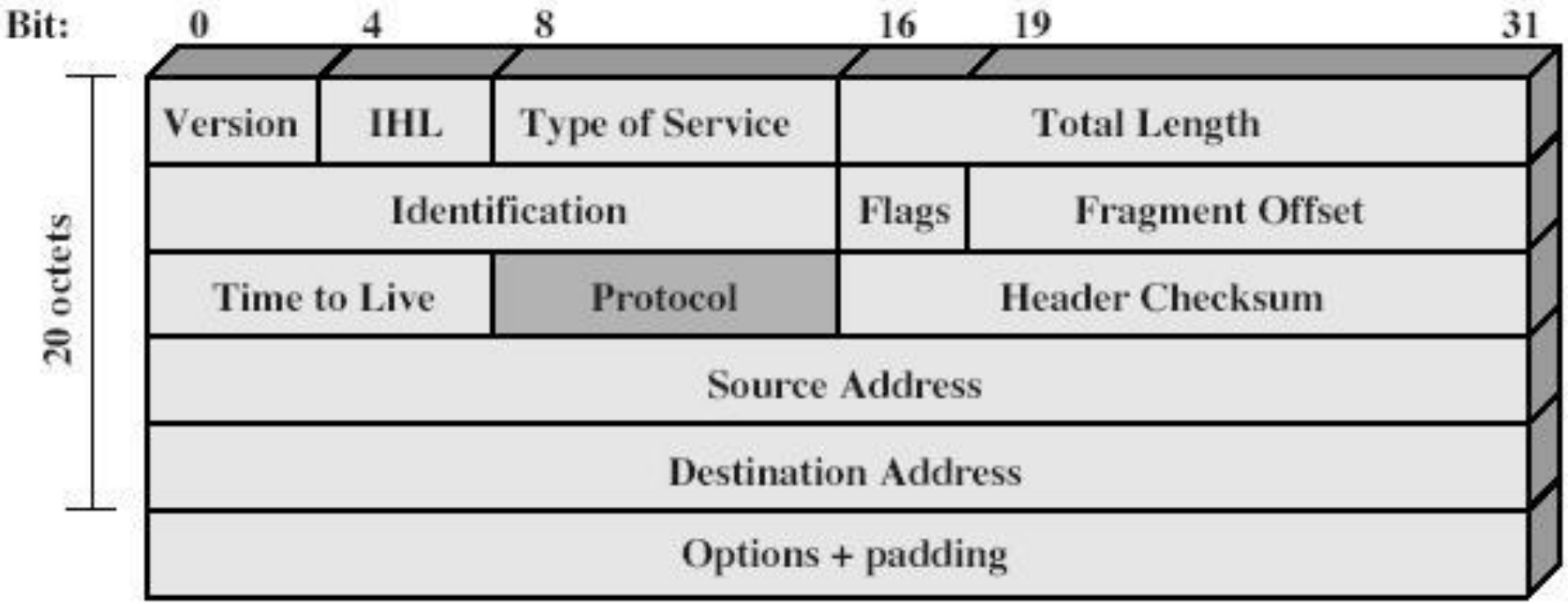
امنیت لایه شبکه

مقدمه – مثالی از TCP/IP





IPV4


Bit:
0
4
8
16
19
31
20 octets
Version
IHL
Type of Service
Total Length
Identification
Flags
Fragment Offset
Time to Live
Protocol
Header Checksum
Source Address
Destination Address
Options + padding

مقدمه

- راه حل های امنیتی وابسته به کاربرد(تاکنون)
  - S/MIME و PGP : امنیت پست الکترونیکی
  - Kerberos : امنیت بین کاربر-کارگزار(احراز هویت)
  - SSL : ایجاد یک کانال امن در وب
- نیاز به امنیت در سطح IP
  - محرمانگی محتوای بسته های IP
  - هویت شناسی فرستنده و گیرنده بسته ها

مقدمه

- IPSec یک پروتکل تنها نیست بلکه مجموعه ای از الگوریتمهای امنیتی و چارچوبی کلی فراهم می کند که به کمک آن ارتباط امنی برقرار کرد.

مقدمه

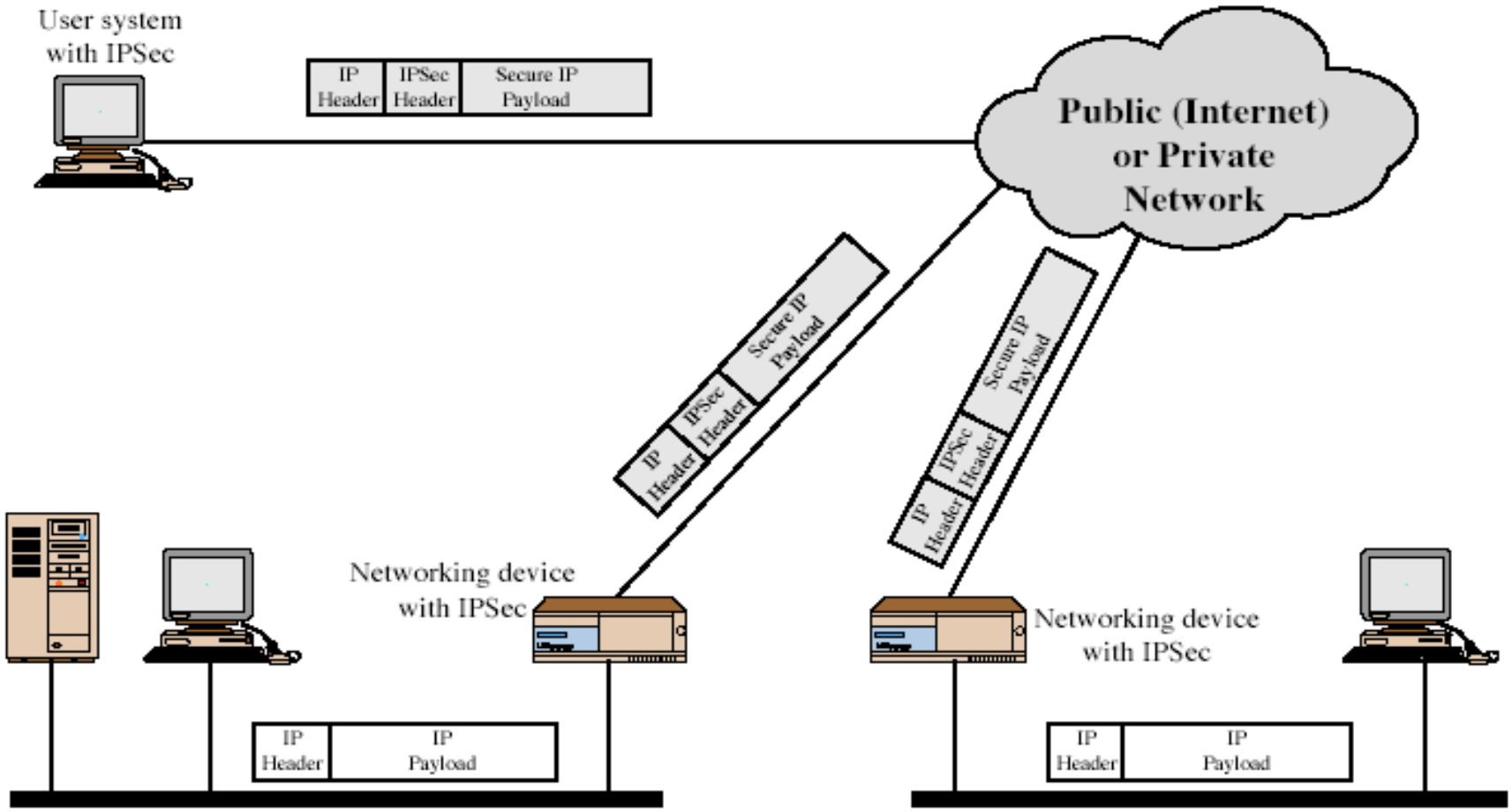
سرویس های امنیتی فراهم شده توسط IPsec (RFC 4301 :)

- Access control
- Connectionless integrity
- Data origin authentication
- Rejection of replayed packets
- Confidentiality (encryption)
- Limited traffic flow confidentiality

مقدمه

- نمونه کاربردهای IPSec
  - ایجاد VPN برای شعبه های مختلف یک سازمان از طریق اینترنت
  - دسترسی امن کارمندان شرکت به منابع شبکه از طریق اینترنت
  - امکان ارتباط امن بین چند سازمان
  - به وجود آوردن خدمات امنیتی برای کاربردهای دیگر(مثل تجارت الکترونیک)

IPSec


User system with IPSec
IP Header
IPSec Header
Secure IP Payload
Public (Internet) or Private Network
IP Header
IPSec Header
Secure IP Payload
IP Header
IPSec Header
Secure IP Payload
Networking device with IPSec
Networking device with IPSec
IP Header
IP Payload
IP Header
IP Payload

مقدمه

- مزایای استفاده از IPSec
  - تامین امنیت قوی بین داخل و خارج LAN در صورت بکارگیری در راهیابها و حفاظ ها (Firewallها)
    - عدم سربار رمزنگاری در نقاط انتهایی
  - شفافیت از نظر کاربران
  - شفافیت از دید برنامه های کاربردی لایه های بالاتر
  - ایجاد ارتباط امن بین کارکنان سازمان از خارج به داخل

تفاوتهای SSLو IPsec

- بر خلاف SSLکه با یک هدف مشخص(HTTPامن) طراحی شد، هدف IPsec عام بوده و هر پروتکلی را که روی IP اجرا می شود را در بر میگیرد
- SSLدر لایه کاربرد است.
- کتابخانه SSLباید به برنامه کاربردی اضافه شود.
- IPsecدر لایه شبکه است.
- IPsecدر سیستم عامل پیاده سازی می شود.
- به کارگیری IPsecمبنی بر سیاستهای تعریف شده در سیستم عامل

معماری IPSec: ویژگیها

- ویژگیها
  - دارای توصیف نسبتا مشکل
  - الزامی در IPv6 و اختیاری در IPv4
  - در برگرفتن موارد زیر:
  - پروتکل IPSec در سرآیند (Header)های توسعه یافته و بعد از سرآیند اصلی IP پیاده سازی می شود
  - مستندات IPSec بسیار حجیم بوده و به صورت زیر دسته بندی شده است:
    - Architecture
    - (ESP) Encapsulating Security Payload : رمزنگاری بسته ها (احراز هویت به صورت اختیاری)
    - (AH) Authentication Header : تشخیص هویت بسته ها
    - مدیریت کلید : تبادل امن کلیدها
    - الگوریتم های رمزنگاری و هویت شناسی

معماری IPSec: سرویس ها

- سرویس های ارائه شده:
  - کنترل دسترسی
  - تضمین صحت داده ها در ارتباط Connectionless
  - احراز هویت منبع داده ها (Data Origin)
  - تشخیص بسته های دوباره ارسال شده و رد آنها (Replay Attack)
  - محرمانگی بسته‌ها
  - محرمانگی جریان ترافیک

معماری IPSec: سرویس ها

	AH	ESP (encryption only)	ESP (encryption plus authentication)
Access control	✔	✔	✔
Connectionless integrity	✔		✔
Data origin authentication	✔		✔
Rejection of replayed packets	✔	✔	✔
Confidentiality		✔	✔
Limited traffic flow confidentiality		✔	✔

پروتکل IPSec

- IPSec مجموعه ای از استاندارد ها ست که ارتباطی امن با استفاده از رمزنگاری برقرار میکند.

- دو مد AH و ESP وجود دارد که به تنهایی یا در ترکیب با هم بکار رفته و **سرویس های** زیر را ارائه می دهند:
  - Anti replay
  - Confidentiality
  - Data origin authentication
  - Privacy
  - Authentication

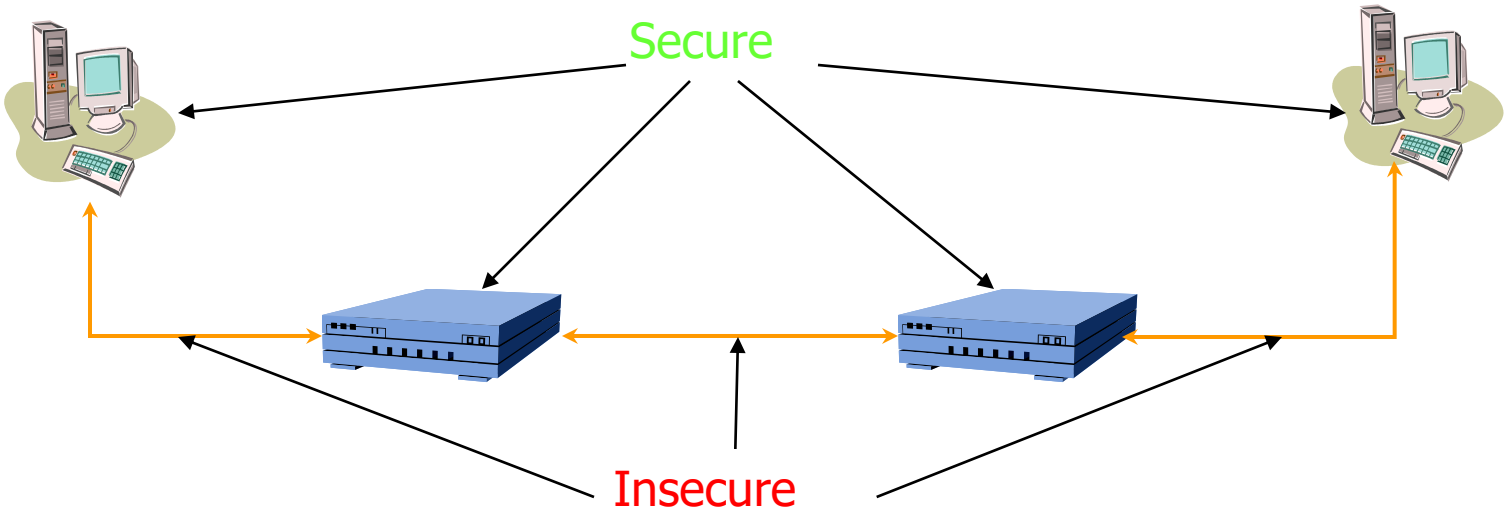
- پروتکل هادر ساختمان داده ای به نام Security Association(SA) پیکربندی می‌شوند.

پروتکل IPSec

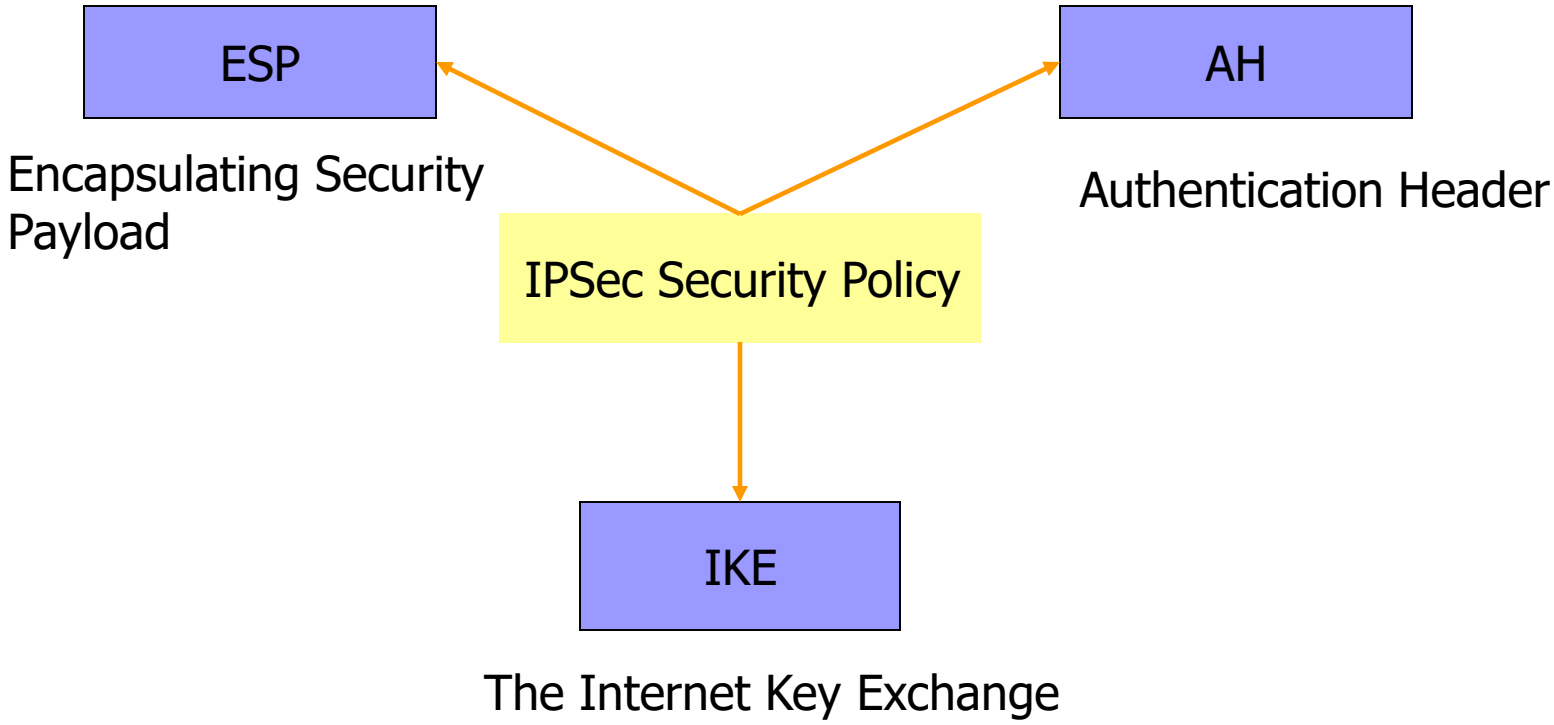
- **مولفه های اصلی** در IPSec :
  1. پروتکل امنیتی برای AH و ESP
  2. SA برای مدیریت سیاست و پردازش ترافیک
  3. مدیریت کلید دستی یا مدیریت کلید خودکار مانند IKE، Oakley، IKE و ISAKMP
  4. الگوریتم های رمزنکاری و تصدیق هویت

- IPSec به سیستم امکان انتخاب پروتکل امنیتی، الگوریتم های مورد استفاده و ورود کلیدهای رمزنگاری را میدهد.
- IPSec می تواند برای امن کردن ارتباط میان دو میزبان، میان دو دروازه (روتر یا فایروال) و یا میان میزبان و دروازه استفاده شود.

The IPSec Security Model



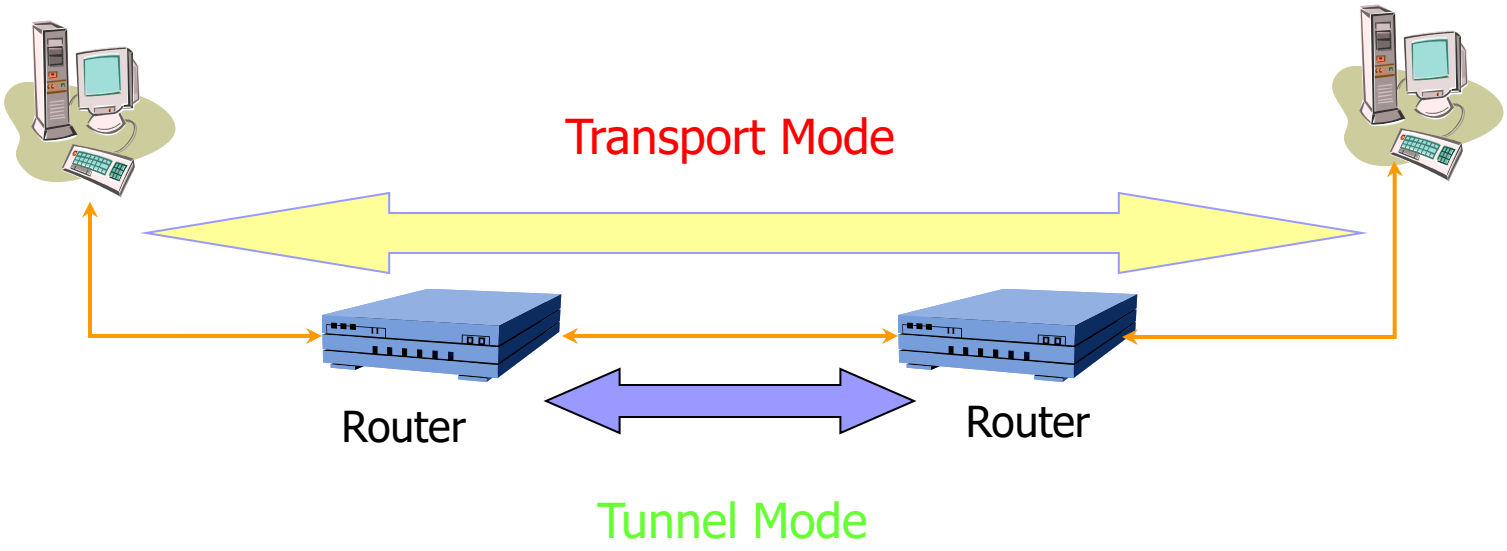
IPSec Architecture



IPSec Architecture

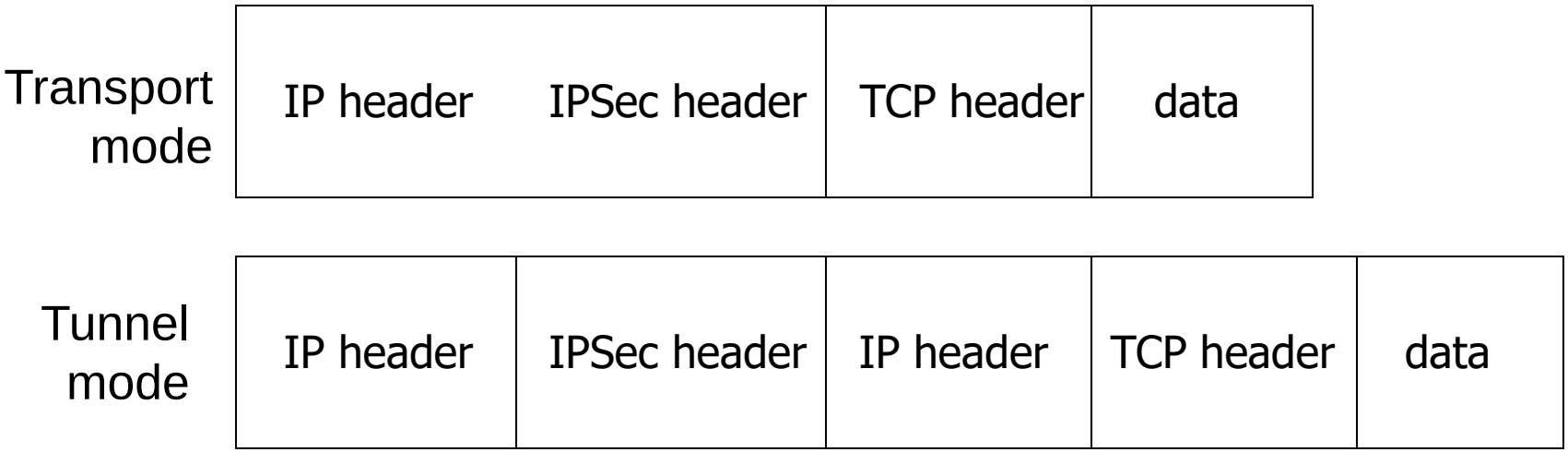
- IPSec provides security in three situations:
  - Host-to-host, host-to-gateway and gateway-to-gateway
- IPSec operates in two modes:
  - *Transport mode* (for end-to-end)
  - *Tunnel mode* (for VPN)

IPsec Architecture



Various Packets

Original IP header TCP header data



مستندات IPSec

پروتکل های استفاده شده در IPSec به هفت گروه دسته بندی می شود.

1. **معماری**: شامل مفاهیم عمومی، نیازمندی های امنیتی، مکانیزم های تکنولوژی IPSec.
2. **ESP**: شامل فرمت بسته ها و مفاهیم کلی ESP برای رمزنگاری و تصدیق هویت اختیاری
3. **AH**: شامل فرمت بسته و مفاهیم کلی AH برای تصدیق هویت
4. **الگوریتم رمزنگاری**: نحوه کاربرد الگوریتم متفاوت رمزنگاری در IPSec
   - سایز کلید و قدرت الگوریتم
   - ارزیابی کارایی الگوریتم

5. **الگوریتم تصدیق هویت**: نحوه کاربرد الگوریتم متفاوت تصدیق هویت در IPSec
   - پارامترهایی مانند تعداد دور و فرمت بلوک ورودی و خروجی
   - پارامترهای اختیاری
   - Paddig لازم
   - مقایسه الگوریتم تصدیق هویت
6. **مدیریت کلید:** نحوه مدیریت کلید مانند پروتکل های IKE، Oakley، IKE و ISAKMP
7. **DOI:** شامل مقادیر مورد نیاز سایر مستندات مانند شناسه الگوریتم های رمزنگاری و تصدیق هویت و یا پارامترهای طول عمر کلید

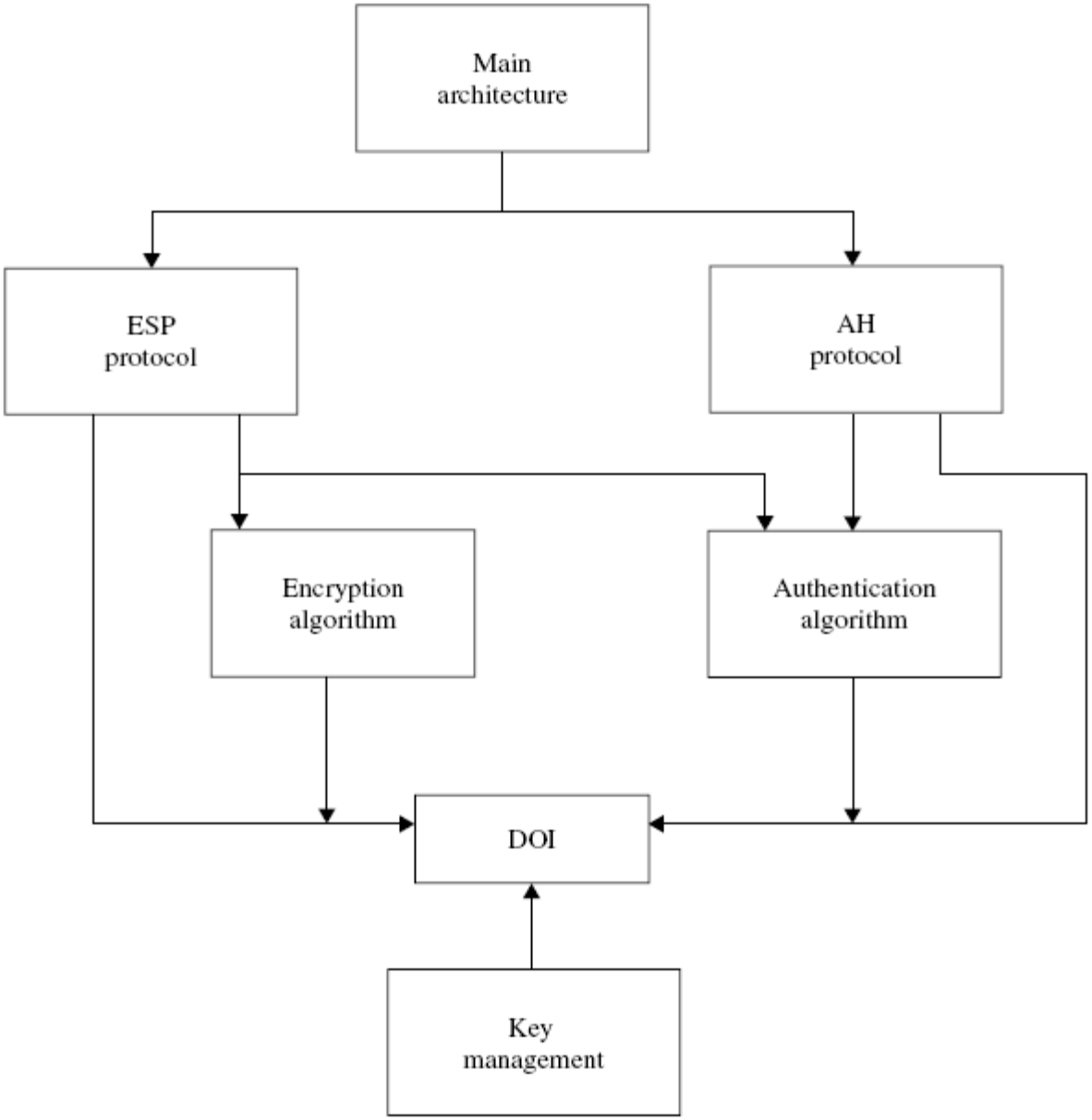


Figure 7.1 Document overview that defines IPsec.

SA

SA اتصالی یک طرفه میان فرستنده و گیرنده ای است که به ترافیک عیوری سرویس امنیتی می دهند. بازای هر پروتکل(AH و ESP) استفاده شده SA جداگانه ای زده می شود.

- هر SAبا سه پارامتر زیر شناخته می شود:

1. Security Parameters Index(**SPI**): شناسه ای که به هر SA تعلق می گیرد. فرستنده SPI متعلق به SA خاص را دانسته و هنگام ارسال بسته مقدار آن را در هدر پروتکل تنظیم کرده تا گیرنده SA مورد نظر برای پردازش بسته را انتخاب کند.
2. **آدرس IP مقصد**: برابر با آدرس IP مقصد SA که ممکن است آدرس میزبان یا روتر و فایروال میان راه باشد.
3. **شناسه پروتکل امنیتی**: مشخص می کند پروتکل استفاده شده AH یا ESP است.

Security policy database(SPD):

- تعیین سرویس های ارائه شده به بسته های IP و نحوه ارائه آن
- کنترل جریان های ترافیکی ورودی و خروجی با عبور دادن، دور انداختن یا پردازش بسته ها
- شامل لیستی از مدخل های سیاست است که هر مدخل با تعدادی انتخاب گر شناخته شده
- اعمال محدودیت بر مدخل ها با خصوصیات SA مانند پروتکل امنیتی، مد انتقال و الگوریتم

Security association database(SAD)

- شامل یک مدخل بازای هر SA
- برای پردازش ورودی، هر مدخل با SPI، آدرس IP مقصد و پروتکل IPSec مخص می شود.
- برای پردازش خروجی، مدخل های SPD با مدخل های SAD اشاره میکند.

مد انتقال SA

- برقراری امنیت را در سطح لایه انتقال مانند بسته های TCP یا UDPیا ICMP(بر روی IP Header امنیت اعمال نمی شود)
- برقراری SA میان دو میزبان
- عدم تغییر در Payload بسته
- در صورت استفاده از AH :
  - تصدیق هویتPayload بسته و قسمت ثابت header
- درصورت استفاده از ESP:
  - انجام عمل رمزنگاری
  - انجام تصدیق هویت در صورت دلخواه تنها بر روی Payload بسته

مد تونل SA

- برقراری امنیت را در سطح لایه IP
- برقراری SA میان دو دروازه یا میزبان و دروازه
- اضافه کردن HEDEAR جدید به بسته
- هنگام انتقال بسته در راه روترهای میانی
- در صورت استفاده از AH :
  - تصدیق هویت کل بسته داخلی و قسمت ثابت header
- درصورت استفاده از ESP:
  - انجام عمل رمزنگاری و در صورت دلخواه تصدیق هویت بر روی کل بسته داخلی

HMAC

- MAC: مکانیزمی برای برقراری جامعیت داده بر اساس کلیدی محرمانه
- HMAC: مکانیزمی برای برقراری جامعیت داده و جامعیت منبع داده با استفاده از توابع درهم ساز(SHA-1، MD5) و کلیدی محرمانه

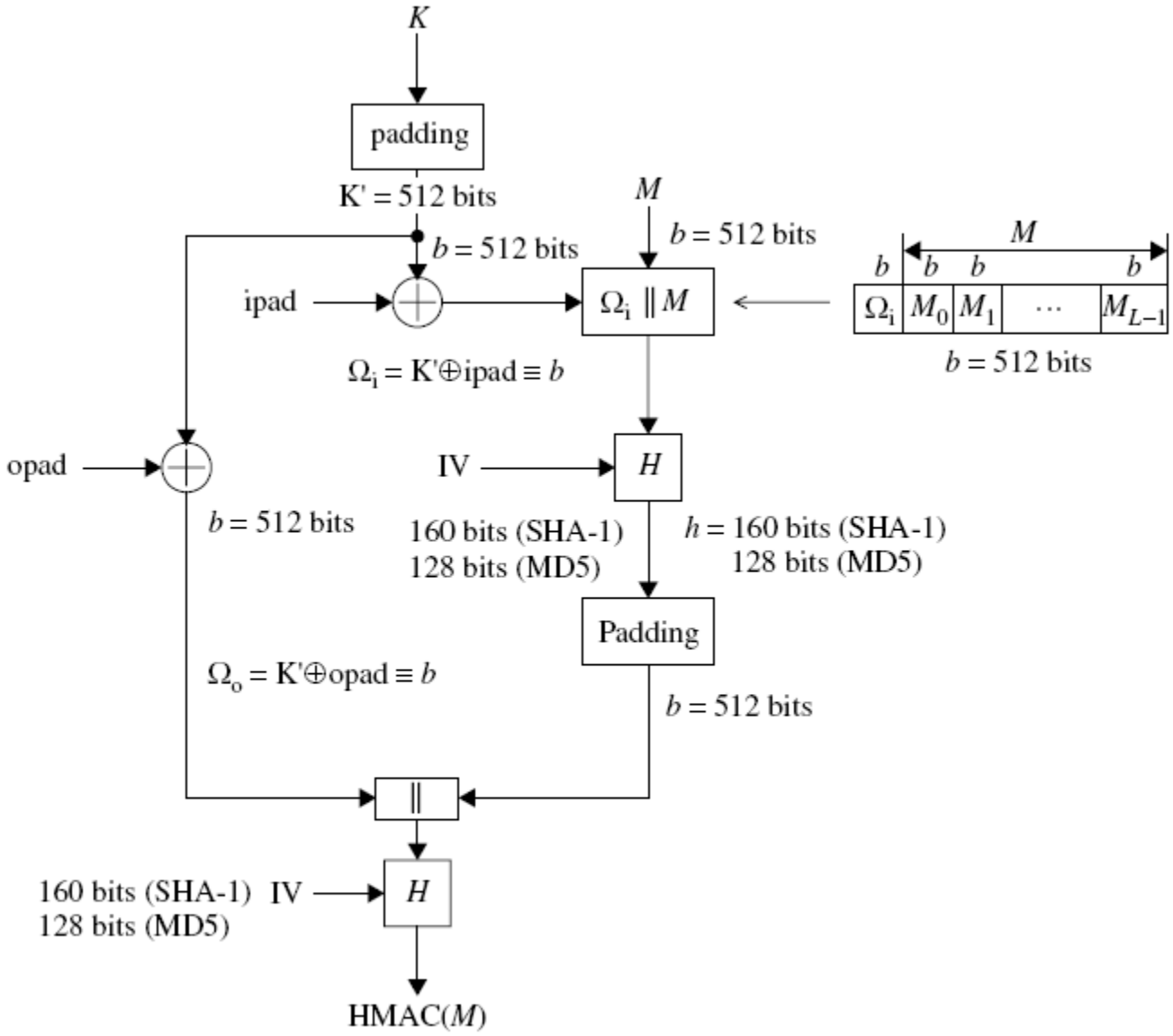


Figure 7.2 Overall operation of HMAC computation using either MD5 or SHA-1 (message length computation based on $\Omega_i \| M$).

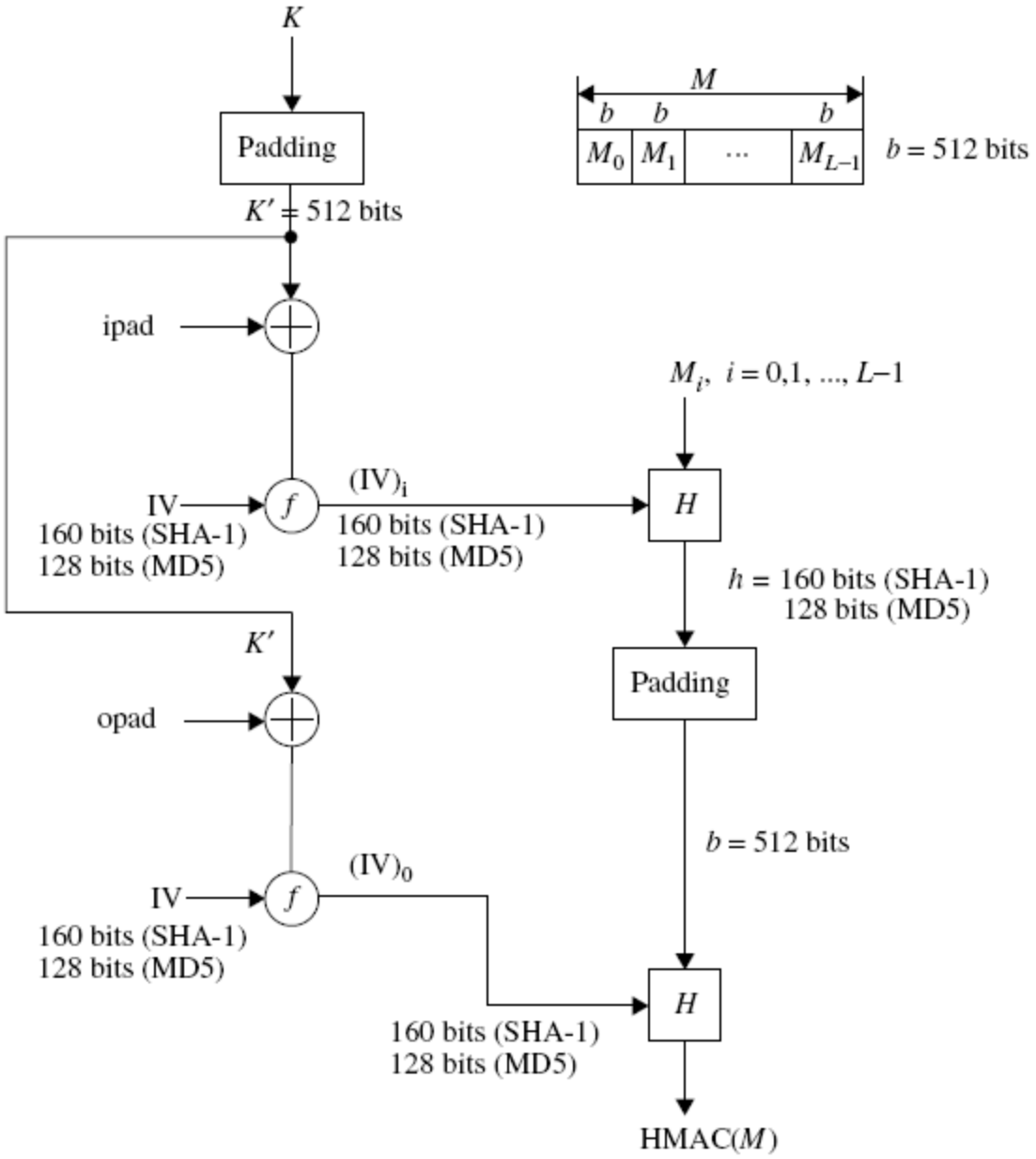


Figure 7.3 Alternative operation of HMAC computation using either MD5 or SHA-1 (message length computation based on M only).

فرمت بسته AH

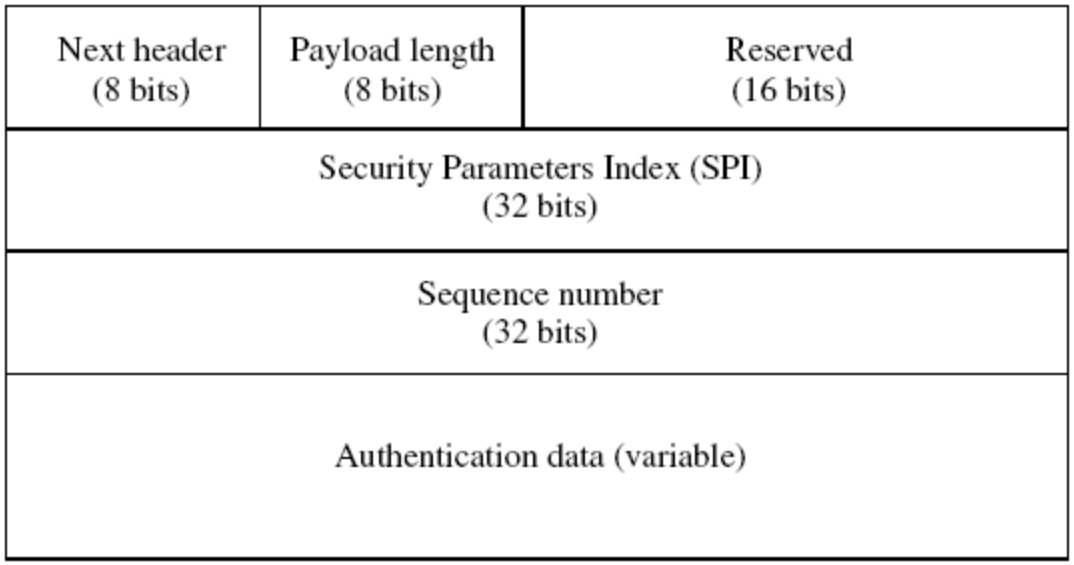
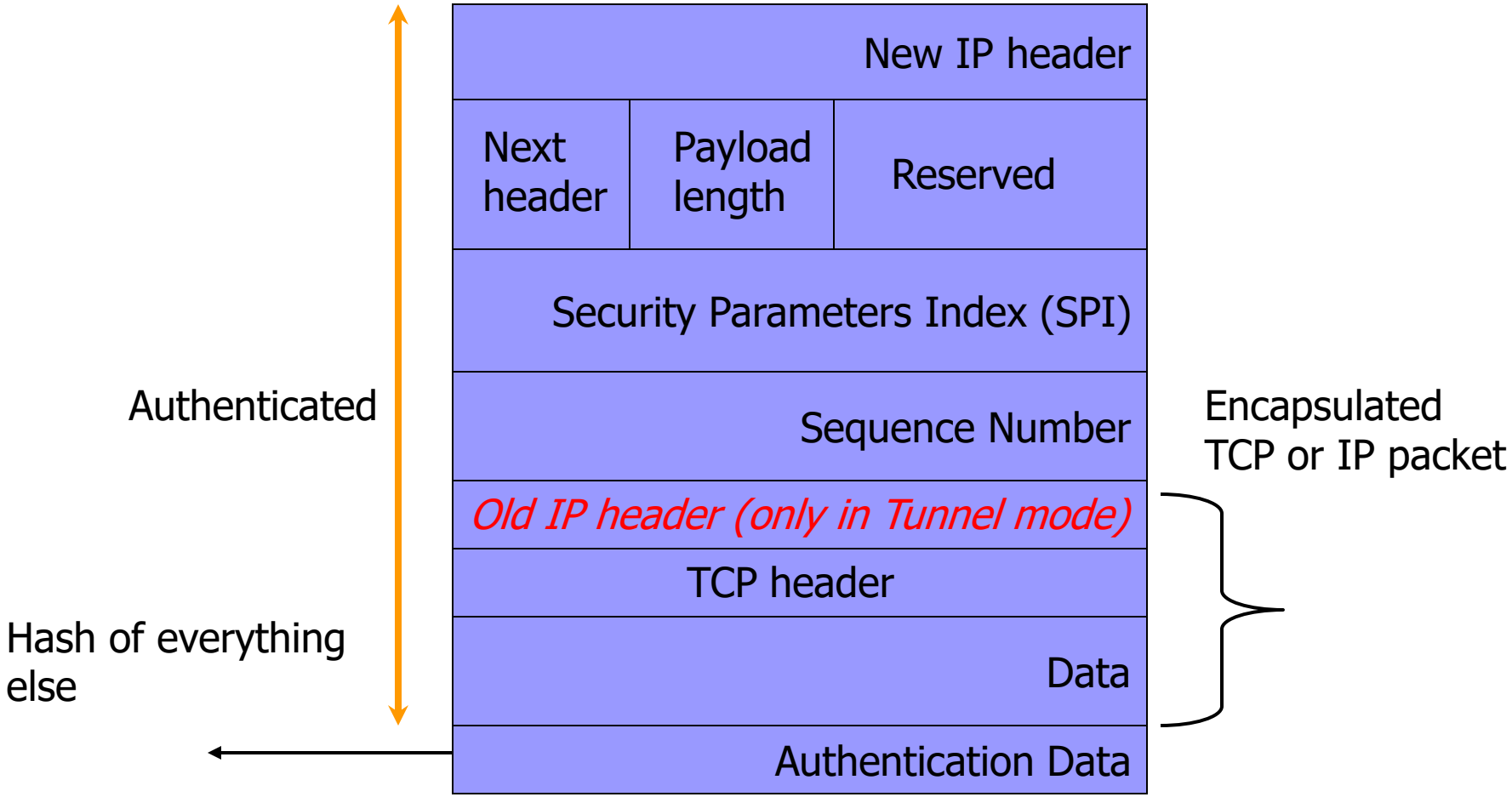


Figure 7.4 IPsec AH format.

AH Packet Details



- **Next header** (8 bits): بیان کننده نوع payload بعد از AH
- **Payload length** (8 bits): بیان کننده طول AH، مقدار پیش فرض آن ۹۶بیت است.
- **Reserved** (16 bits): رزو شده و برابر مقدار صفر
- **SPI** (32 bits): شناسه منحصر بفرد متعلق به SA . مقدار ۱-۲۵۵ رزرو شده است.
- **Sequence number** (32 bits): حاوی شمارنده افزایشی برای جلوگیری از حمله Replay.
  - شمارنده حتی در صورت فعال نبودن سرویس anti-replay باز هم ارسال میشود اما پردازش SA به صلاحدید گیرنده است. مقدار شمارنده ابتدا برابر صفر بوده، فرستنده شمارنده را افزایش داده و در صورت فعال بودن سرویس anti-replay، بررسی می کند که شمارنده چرخش نداشته باشد،که در این صورت SA جدیدی برقرار می شود.
- **Authentication data** (variable): حاوی ICV یا MAC بسته. برای رساندن طول بسته به مضرب ۳۲ یا ۶۴ ممکن است شامل Padding نیز باشد.

محل AH

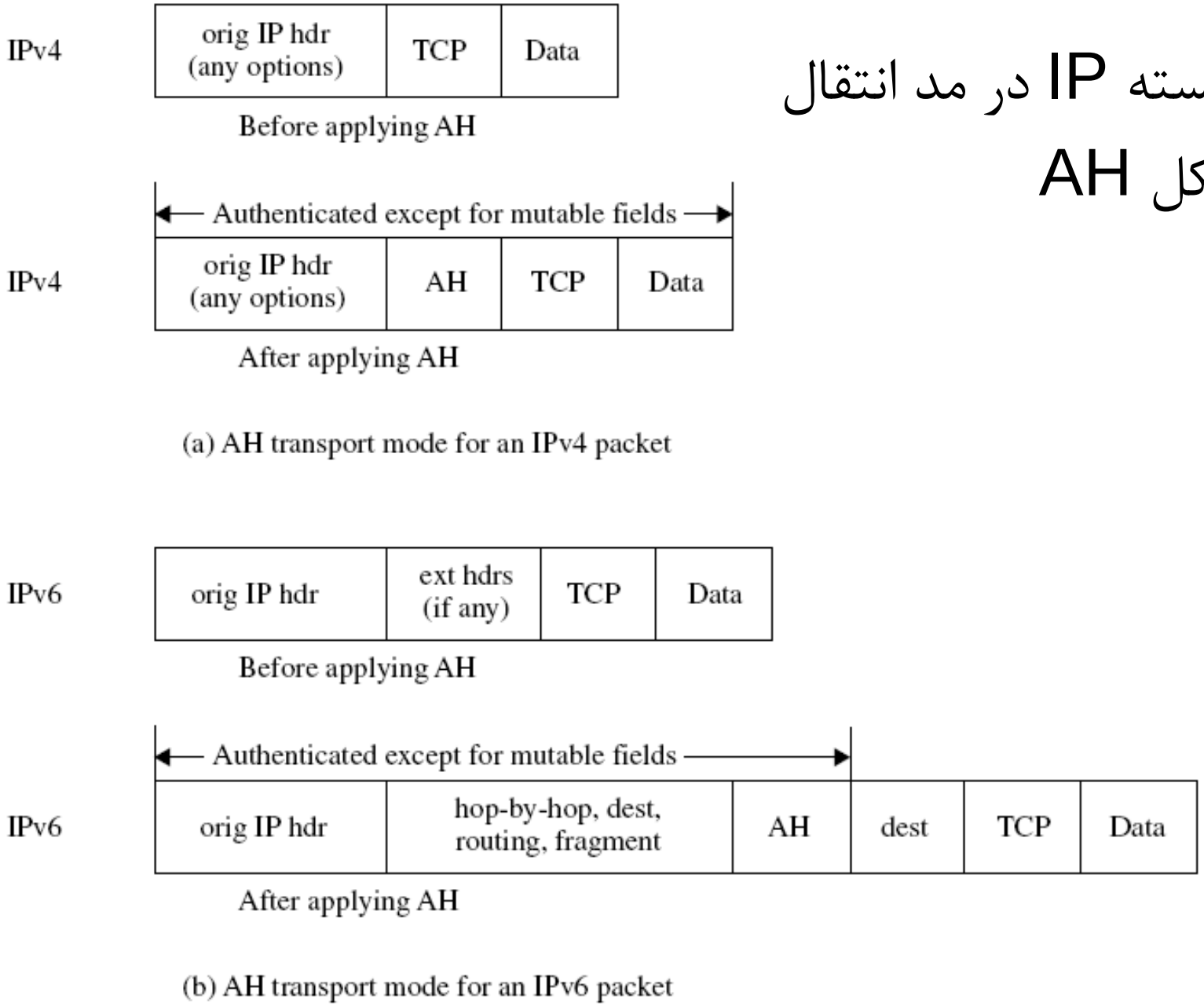
مد انتقال: میان دو میزبان

- IPv4:
  - AH بعد از IP header اصلی و قبل از پروتکل لایه بعدی (TCP,UDP) قرار می گیرد.
  - تصدیق هویت بر وری کل بسته بجز فیلدهای متغیر در header
- IPv6:
  - AH بعد از فیلد های hop-to-hop، routing ، fragmentation و extension headers قرار می گیرد
  - تصدیق هویت بر روی کل بسته به جز فیلدهای متغیر در header

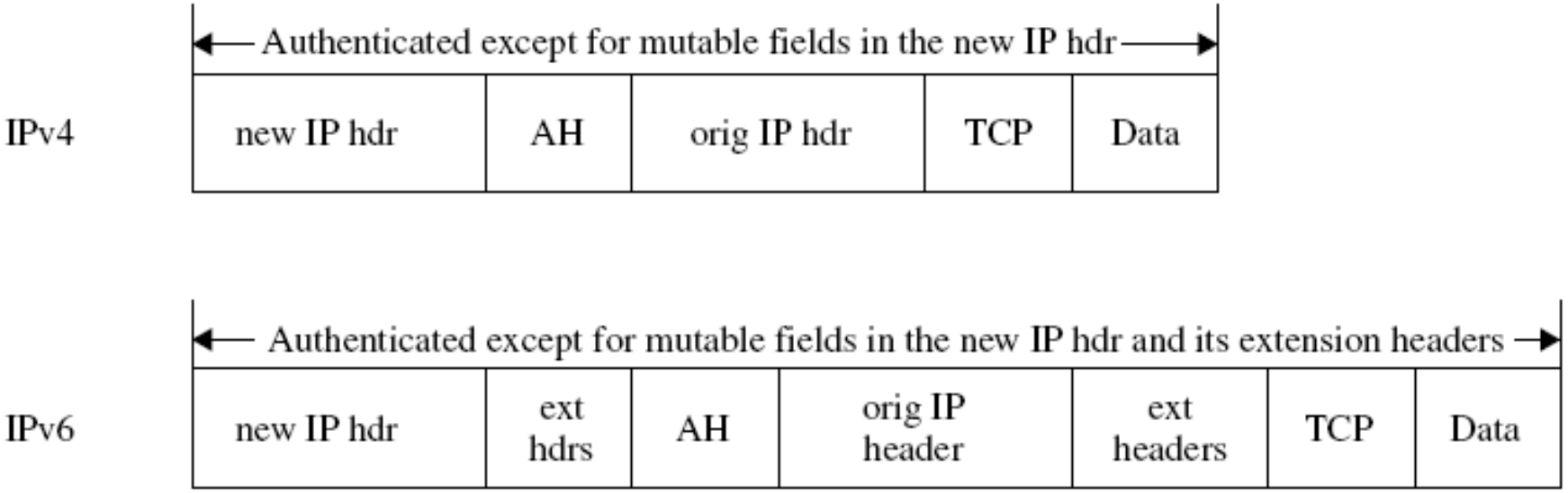
مد تونل: میان میزبان ها و دروازه ها

- IP header داخلی حاوی آدرس واقعی مبدا و مقصد و IP header حاوی آدرس های میانی (آدرس فایروال یا دروازه ها)
- حفاظت از کل بسته داخلی شامل IP header داخلی

فرمت بسته IP در مد انتقال با پروتکل AH



فرمت بسته IP در مد تونل با پروتکل AH



(c) AH tunnel mode for typical IPv4 and IPv6 packets

فرمت بسته ESP

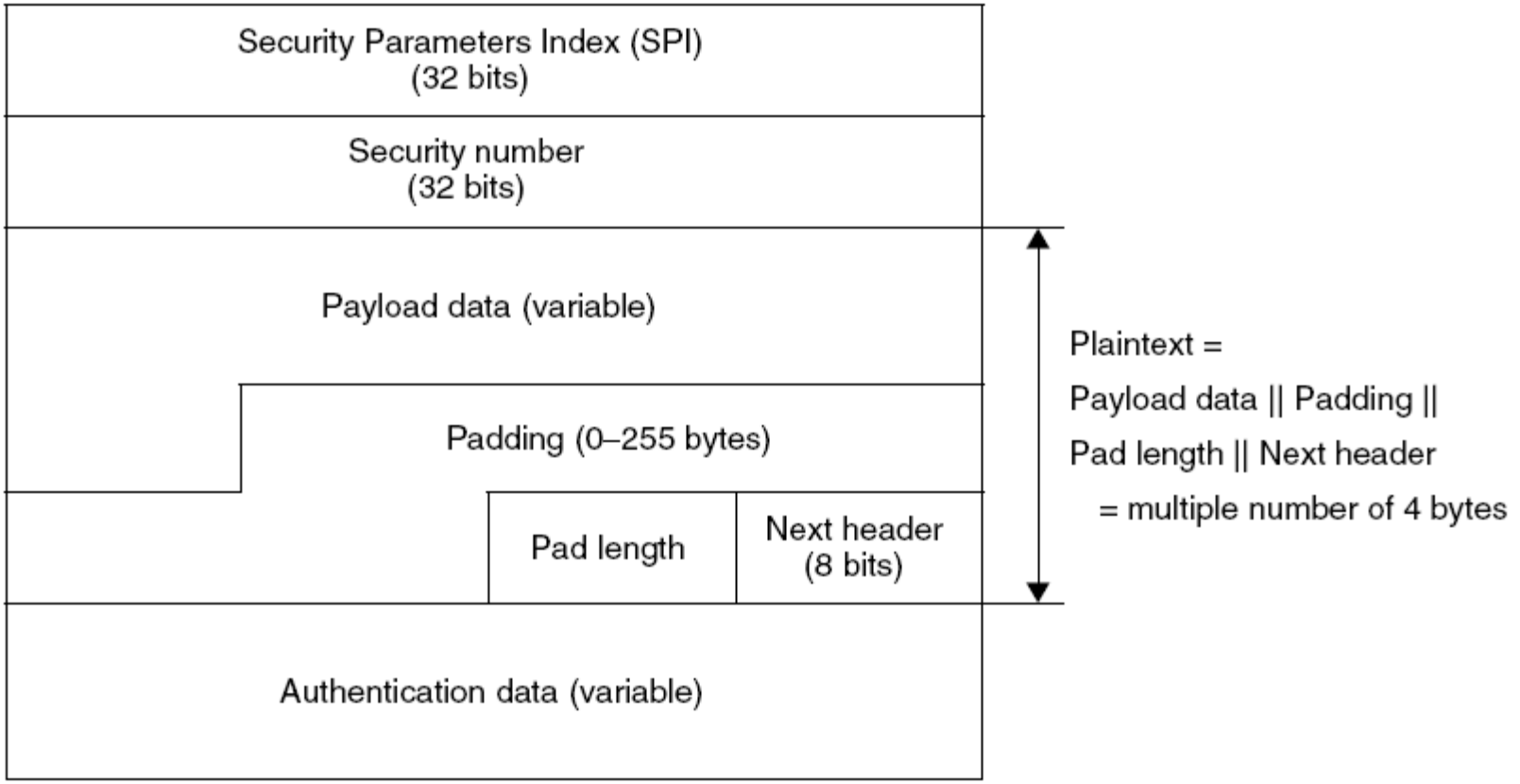
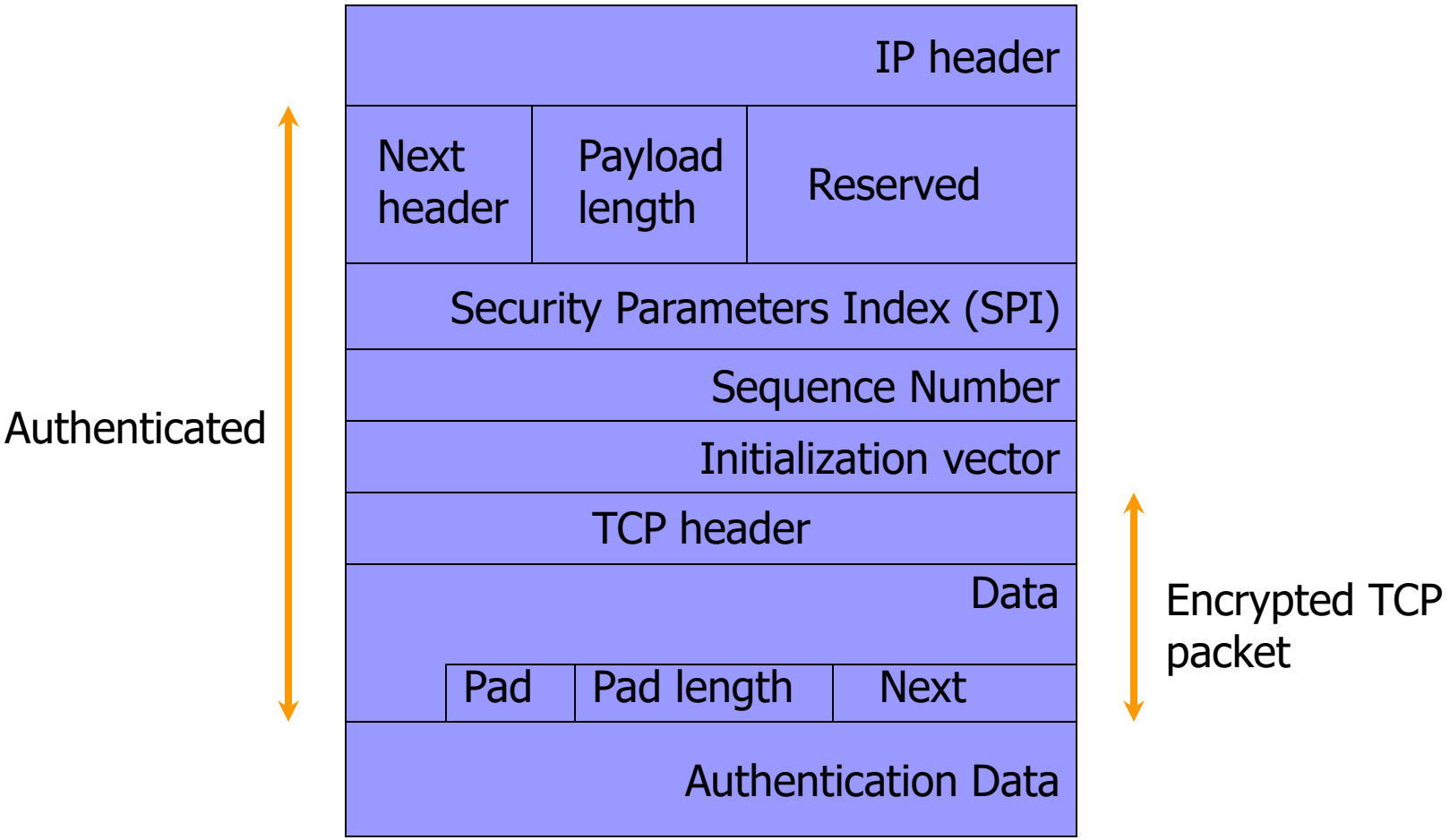


Figure 7.6 IPsec ESP format.

ESP Packet Details



- **SPI** (32 bits): شناسه منحصر بفرد ۳۲ بیتی برای SA.
- **Sequence number** (32 bits): حاوی شمارنده افزایشی برای جلوگیری از حمله Replay.
  - شمارنده حتی در صورت فعال نبودن سرویس anti-replay باز هم ارسال میشود اما پردازش SA به صلاحدید گیرنده است. مقدار شمارنده ابتدا برابر صفر بوده، فرستنده شمارنده را افزایش داده و در صورت فعال بودن سرویس anti-replay، بررسی می کند که شمارنده چرخش نداشته باشد،که در این صورت SA جدیدی برقرار می شود.
- **Padding**:
  - اگر لازم باشد پیام اصلی برای الگوریتم رمزنگاری مضربی از بایت ها باشد، از padding استفاده میشود. متن اصلی شامل payload، padding و header بعدی است.
  - پیام رمزشده باید به ۳۲ بیت ختم شود، بنابراین به padding استفاده می شود.

بایت های padding با یک سری از اعداد صحیح مقدار اولیه می گیرد. اگر الگوریتم رمزنگاری به padding داشته باشد، نوع الگوریتم و مد کاری آن مقدار padding را مشخص می‌کند .

- **Pad length**: بیان کننده تعداد بایت های pad. عددی بین ۰-۲۵۵
- **Next header** (8 bits): بیان کننده نوع payload مانند شناسه پروتکل لایه بالاتر
- **Authentication data** (variable): شامل ICV محاسبه شده بر روی بسته ESP بجز داده تصدیق هویت، تنها در صورت فعال بودن سرویس تصدیق هویت این فیلد پر می شود. طول این فیلد وابسته به نوع الگوریتم تصدیق هویت است.

ESP

- ویژگیها
  - پشتیبانی از محرمانگی داده و تا حدی محرمانگی ترافیک
  - امکان استفاده از هویت شناسی(مشابه AH)
  - استفاده از الگوریتم DES در مد CBC(امکان استفاده از 3-DES, RC5, IDEA, 3-IDEA, CAST و Blowfish نیز وجود دارد)

ESP

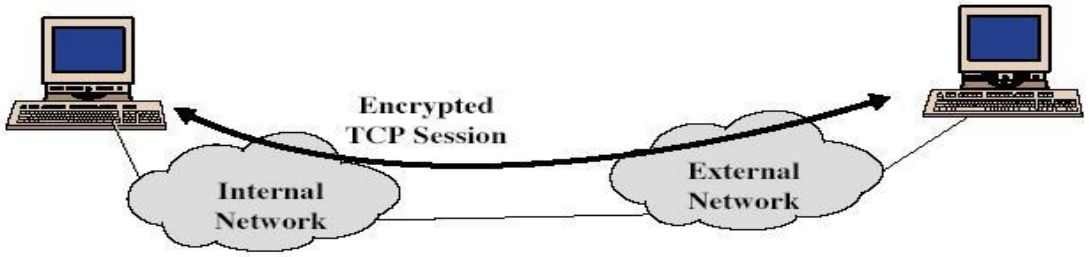
- فیلدهای ESP
  - SPI : شناسه SA
  - Sequence Number : شمارنده برای جلوگیری از حمله تکرار مشابه AH
  - Payload : محتوای بسته که رمز می شود
  - Padding : بیتهای اضافی
  - Pad Length : طول فیلد بالا
  - Next Header : نوع داده موجود در Payload Data
  - Authentication Data : مقدار MAC محاسبه شده (بدون در نظر گرفتن خود فیلد)

ESP

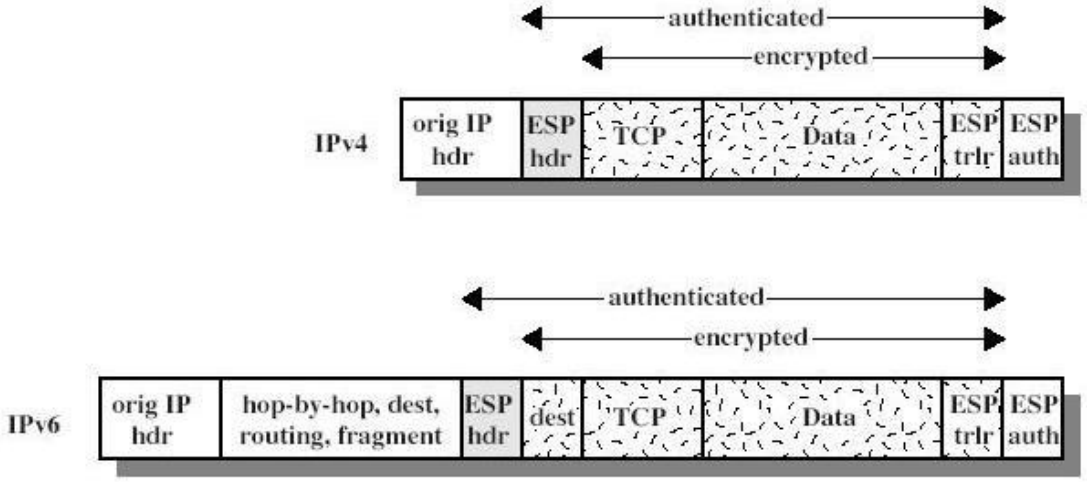
- حالت انتقال
  - تضمین محرمانگی بین host ها
  - رمزنگاری بسته داده، دنباله ESP و اضافه شدن MAC درصورت انتخاب هویت شناسی توسط مبداء
  - تعیین مسیر توسط Router های میانی با استفاده از سرآیندهای اصلی(که رمز نشده اند)
  - چک کردن سرآیند IP توسط مقصد و واگشایی رمز باقیمانده پیغام
  - امکان آنالیز ترافیک

Transport Mode ESP

- used for communication between hosts



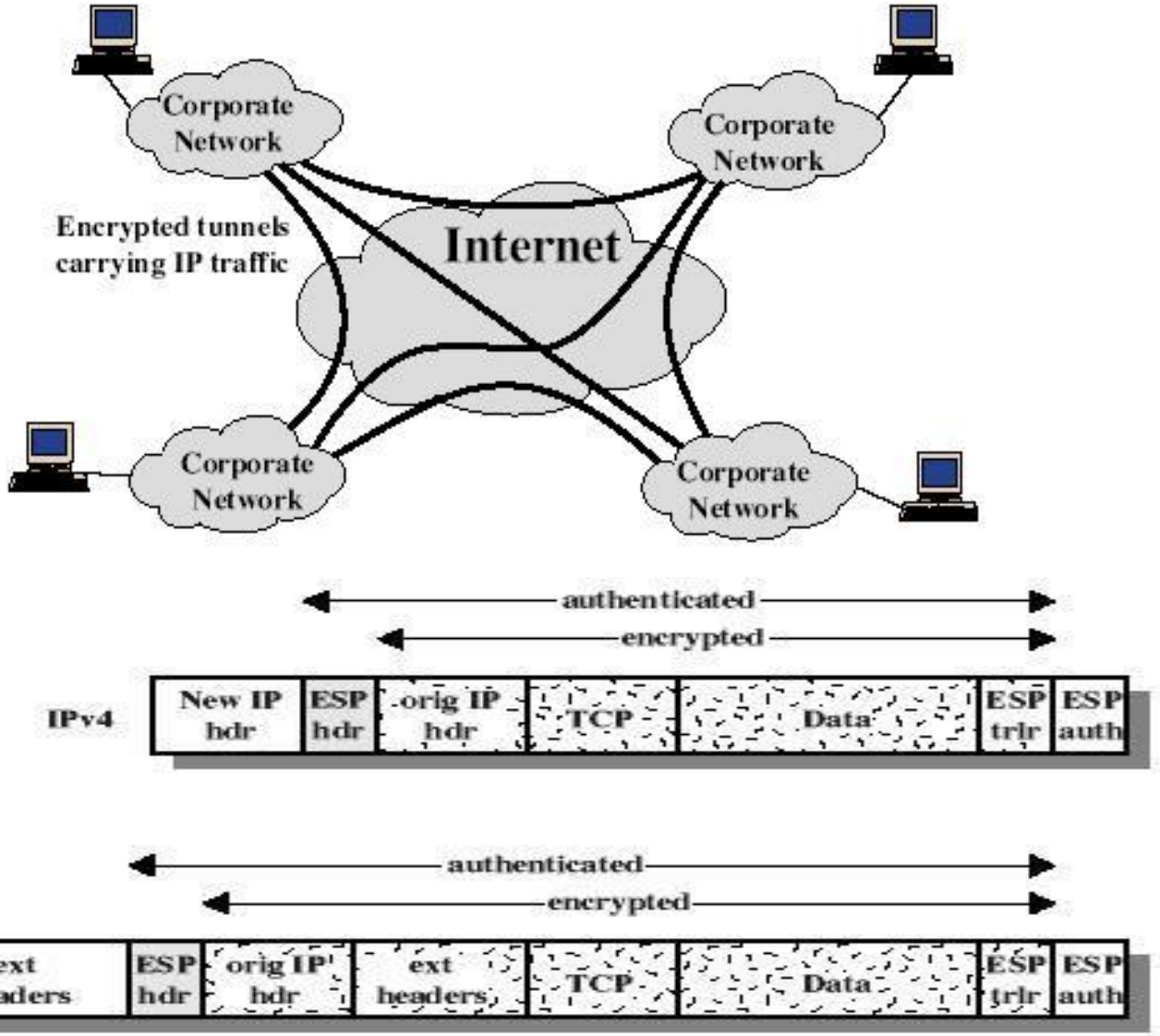
- scope



ESP

- حالت تونل
  - اضافه شدن آدرس مبداء و مقصد دروازه های خروجی فرستنده و گیرنده، سرآیند ESP و دنباله ESP و قسمت مربوط به MAC در صورت نیاز(برای هویت‌شناسی)
  - انجام مسیریابی در Routerهای میانی از روی آدرس‌های جدید
  - رسیدن بسته به فایروال شبکه مقصد و مسیریابی از روی آدرس IP قبلی تا گره نهایی
  - حالت تونل IPSec یکی از روشهای ایجاد VPNها است

Tunnel Mode ESP



محل ESP

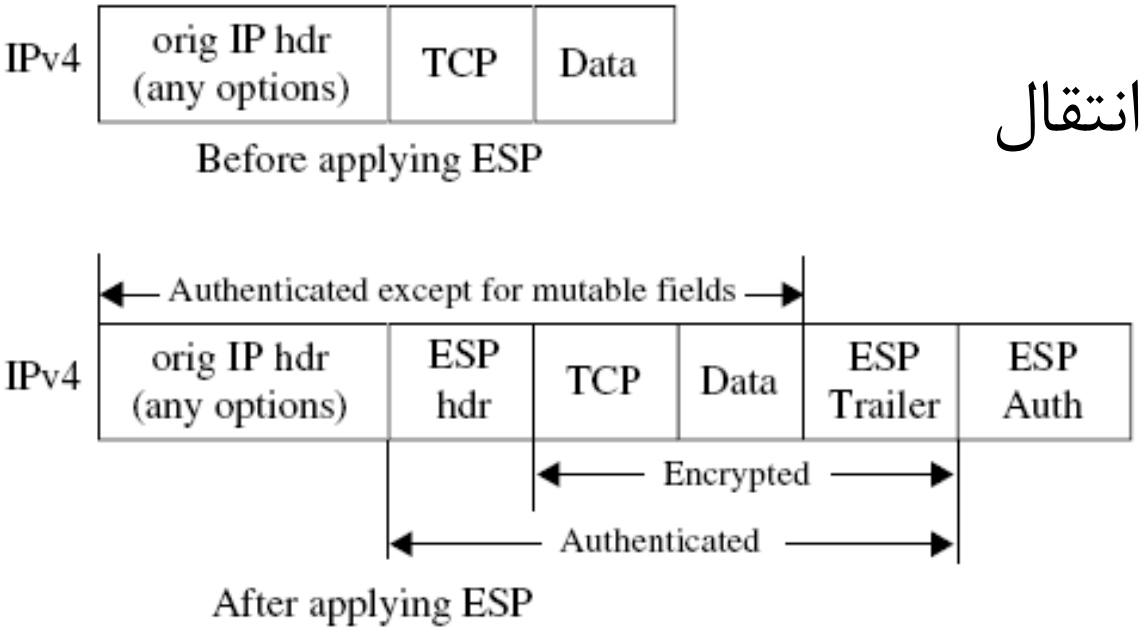
مد انتقال: میان میزبان ها

- IPv4:
  - ESP بعد از IP header اصلی و قبل از پروتکل لایه بعدی (TCP,UDP) قرار می گیرد.
  - ESP trailer شامل padding، pad length و فیلد next header
- IPv6:
  - ESP بعد از فیلد های hop-to-hop، routing ، fragmentation و extension headers قرار می گیرد

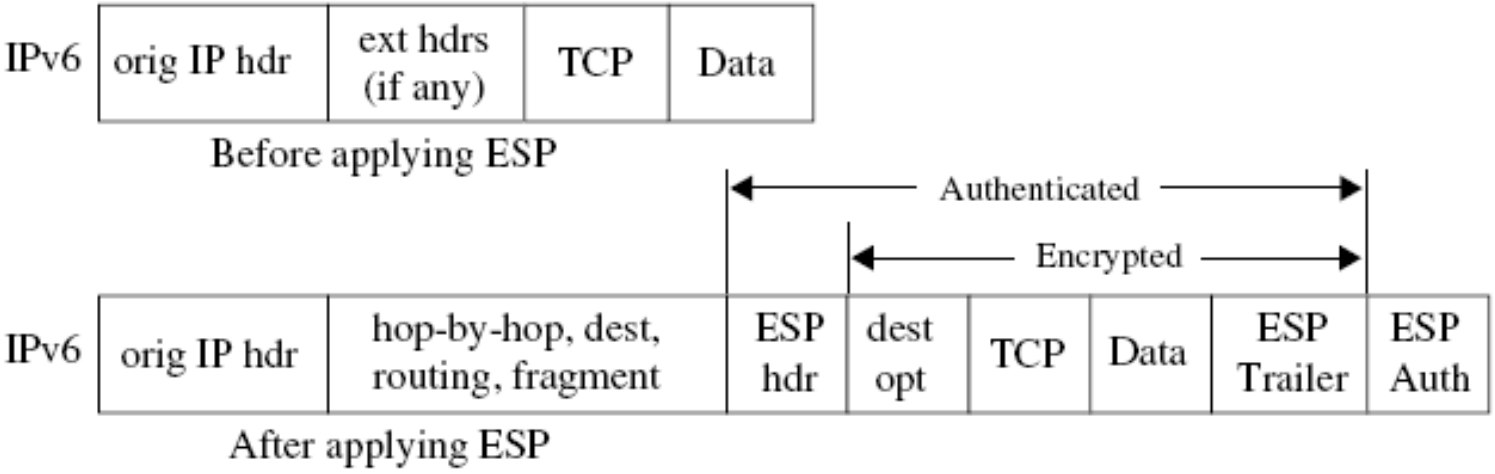
مد تونل: میان میزبان ها و دروازه ها

- IP header داخلی حاوی آدرس واقعی مبدا و مقصد و IP header حاوی آدرس های میانی (آدرس فایروال یا دروازه ها)
- حفاظت از کل بسته داخلی شامل IP header داخلی

فرمت بسته IP در مد انتقال با پروتکل ESP

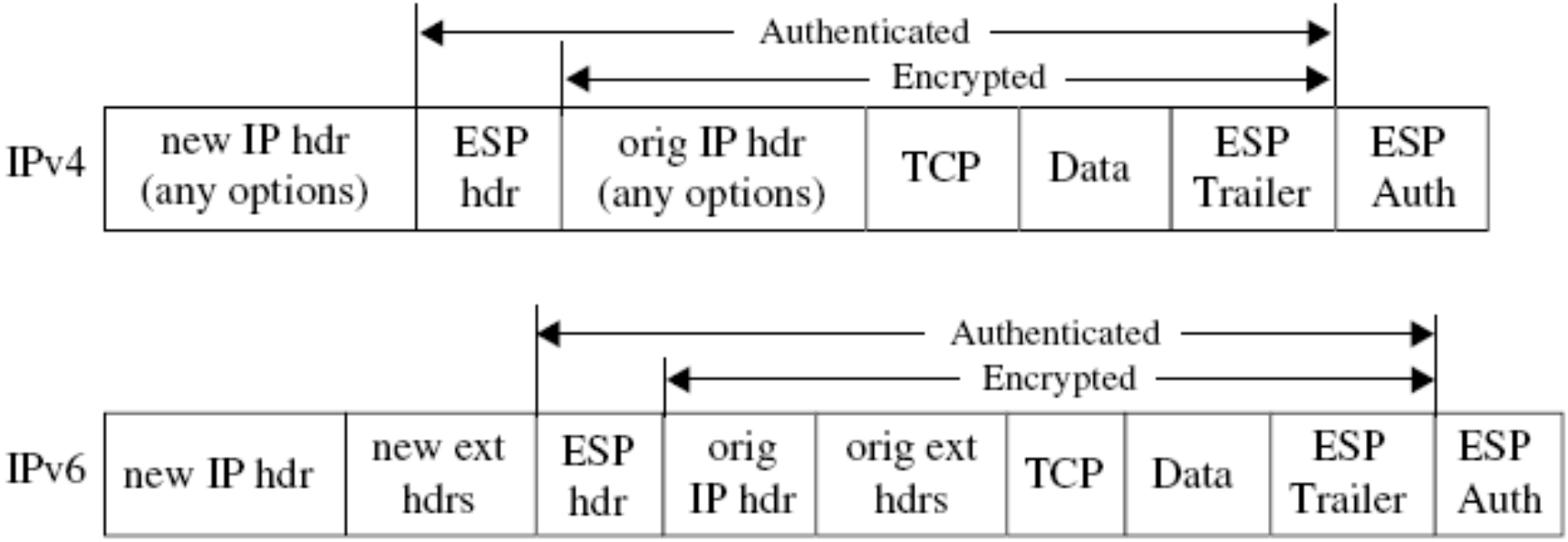


(a) ESP transport mode for an IPv4 packet



(b) ESP transport mode for an IPv6 packet

فرمت بسته IP در مد تونل با پروتکل ESP



(c) ESP tunnel mode for typical IPv4 and IPv6 packets

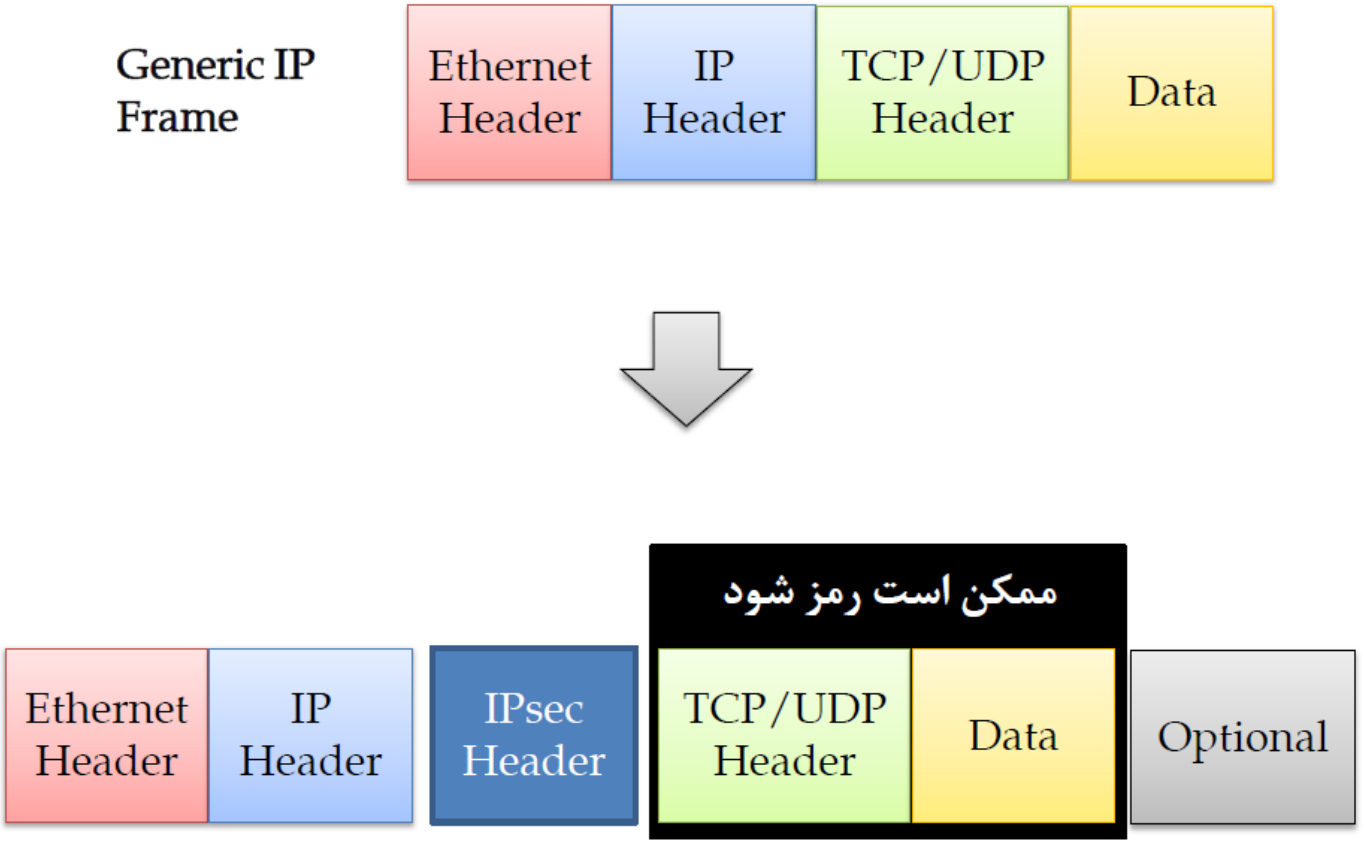
Figure 7.7 Transport mode and tunnel mode for ESP authentication.

جمع بندی

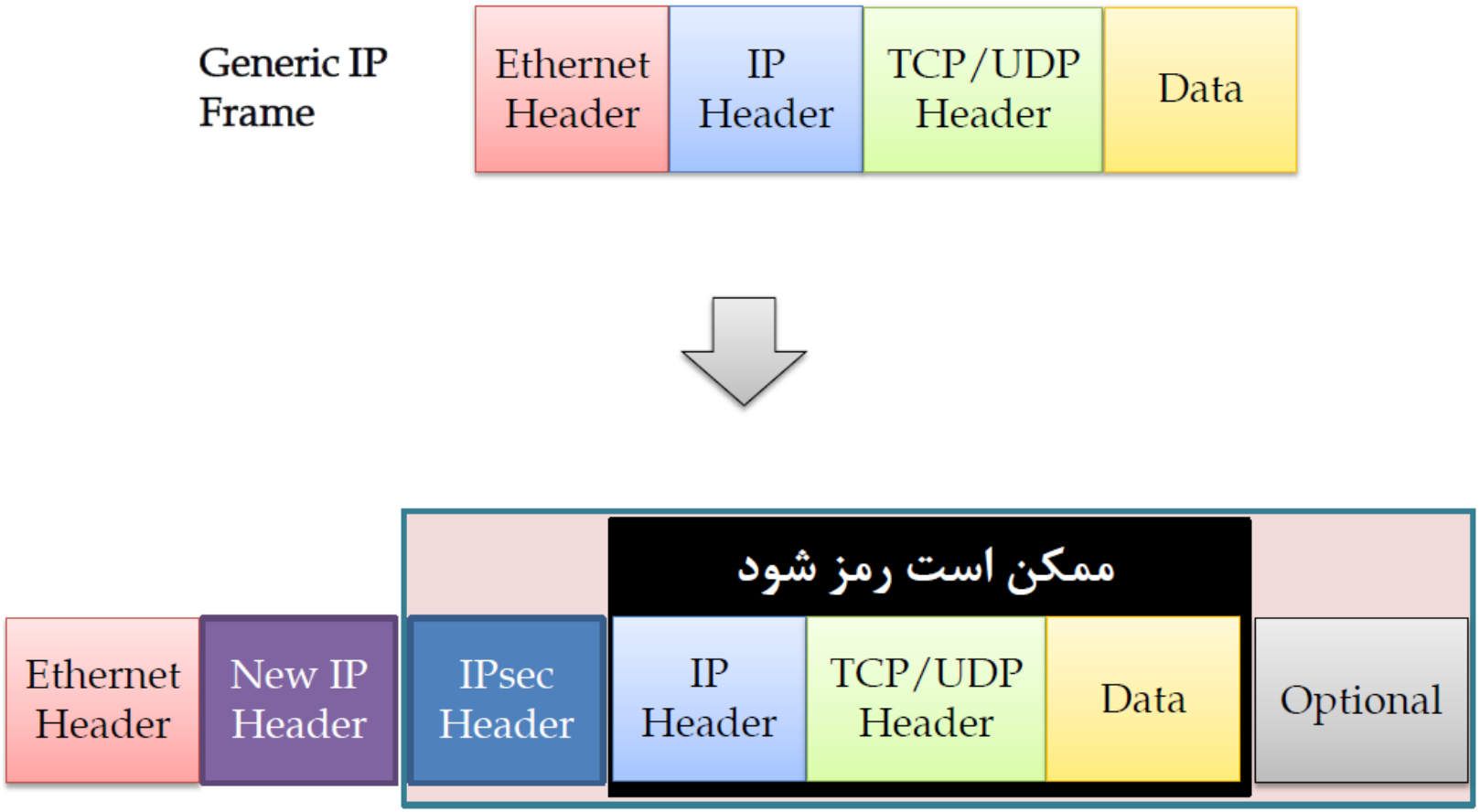
IPsec به دو سبک بسته ها را ارسال میکند: انتقال و تونل.

- سبک انتقال Transport Mode
  - سرآیند IPبسته اصلی تغییر نمی کند.
  - سرآیند IPsecبعد از سرآیند IPاضافه میشود.
  - در صورت لزوم، رمزنگاری به داده بسته IP اعمال میشود.
- سبک تونل Tunnel Mode
  - کل بسته IP شامل سرآیند و داده، به عنوان Payload یک بسته IPجدید قرار میگیرد (در صورت لزوم، با اعمال رمزنگاری)
  - سرآیند IPsec بعد از سرآیند IP اضافه میشود.

Transport Mode



Tunnel Mode


Generic IP Frame
Ethernet Header
IP Header
TCP/UDP Header
Data
ممکن است رمز شود
Ethernet Header
New IP Header
IPsec Header
IP Header
TCP/UDP Header
Data
Optional

مقایسه Tunnel Mode و Transport Mode

- سبک انتقال، سربار کمتری دارد.
  - نیازی به ایجاد سرآیند جدید و ارسال آن روی شبکه نیست.
  - مناسب برای کاربردهای میزبان به میزبان H2H
- سبک تونل برای کاربردهای N2N و H2Nمناسب است.
  - از دیدگاه درگاه شبکه، بسته برای وی ارسال شده است.
  - درگاه بسته را باز و بسته IPرا استخراج کرده و برای میزبان نهایی می فرستد.
  - در صورت رمزنگاری بسته، مقصد نهایی (و در N2N، مبدا اصلی) از دید شبکه عمومی مخفی خواهد بود.

پروتکلهای مورد استفاده در IPsec

IKE: Internet Key Exchange

- تبادل کلید و پارامترها
- دو نسخه: ۱ و ۲
- UDP با پورت ۵۰۰

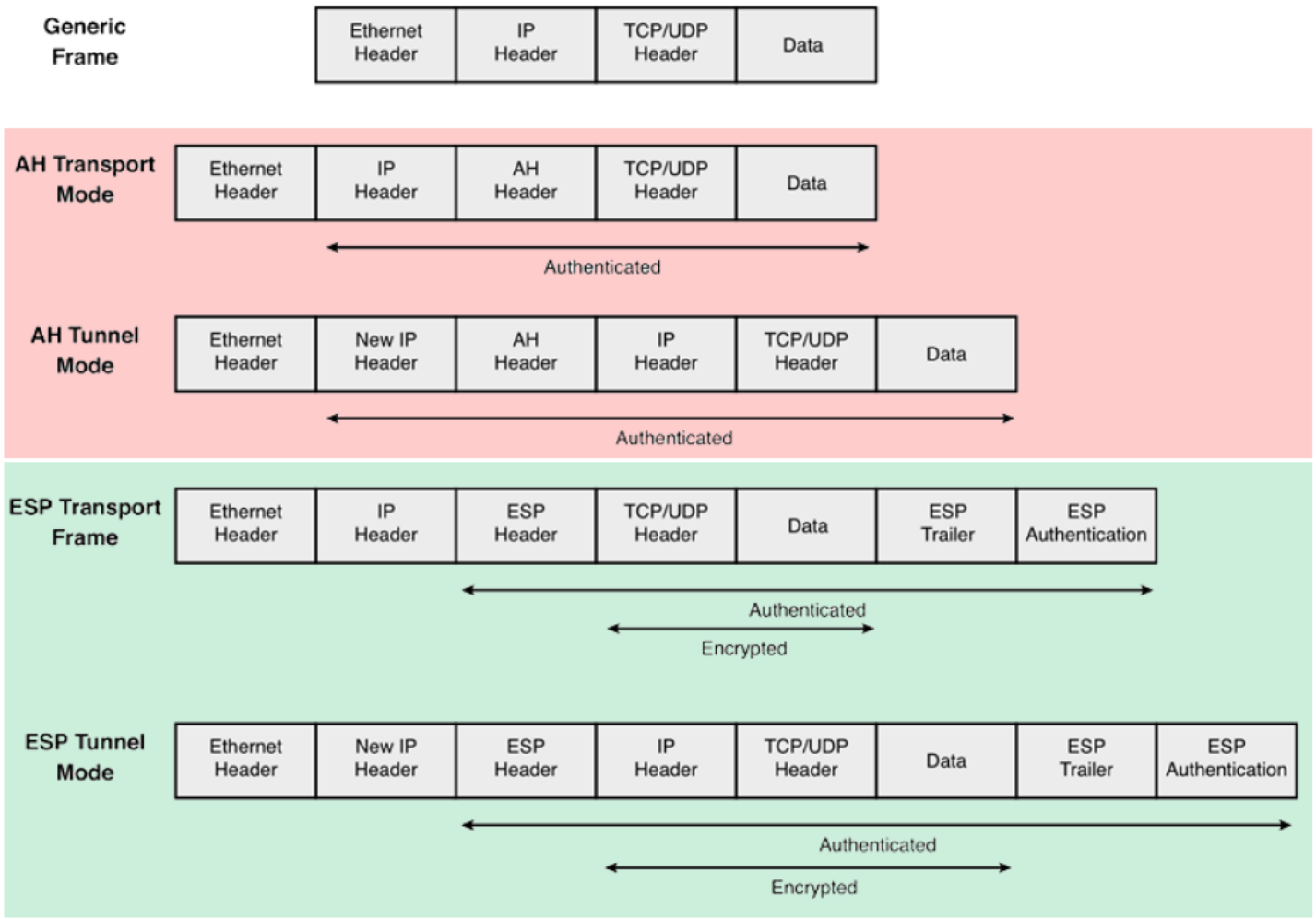
AH: Authentication Header

- صحت محتوا و سرآیند بسته
- عدم محرمانگی

ESP: Encapsulating Security Payload

- محرمانگی و صحت محتوای بسته
- عدم تضمین صحت سرآیند

مدهای تونل و انتقال در ترکیب با AHو ESP



رمزنگاری

- ESP از الگوریتم رمزنگاری متقارن مانند DES، IDEA، CAST یا Blowfish استفاده می کند.
- فرستنده فیلدهای Payload، Padding و طول pad و header بعدی را رمزنگاری می کند.
- اگر الگوریتم رمزنگاری به IV نیاز داشته باشد، مقدار IV صریحاً در payload ذکر می شود.
- عمل رمزنگاری قبل از تصدیق هویت انجام می شود،
  - در نتیجه تشخیص و رفع بسته های جعلی و replay سریع تر و قبل از رمزگشایی انجام شده و باعث کاهش تاثیر حملات می شود.
  - در گیرنده نیز رمزگشایی و تصدیق هویت به صورت موازی انجام می شوند.
- داده های تصدیق هویت با رمزنکاری محافظت نمی شود، بنابراین از الگوریتم تصدیق هویت با کلید برای محاسبه ICV استفاده می شود.

رمزگشایی

- گیرنده حداقل Payload، Padding و طول pad و header بعدی را با استفاده از الگوریتم رمزنگاری و مد الگوریتم و IV (از payload) بدست می آورد.

- در مد انتقال:
  - گیرنده بسته IP اصلی را از ip Header اصلی و اطلاعات پروتکل لایه بالایی در فیلد ESP payload بدست می آورد.
- در مد تونل:
  - header بسته IP به همراه کل بسته IP در فیلد ESP payload را بدست می آورد.

در صورت پردازش سریال بررسی اعتبار و رمز گشایی، ابتدا بررسی اعتبار ICV و MAC و سپس رمزگشایی انجام می شود.در صورت پردازش موازی، بررسی اعتبار باید قبل از پردازش های بعدی روی بسته رمزگشایی شده انجام شود.

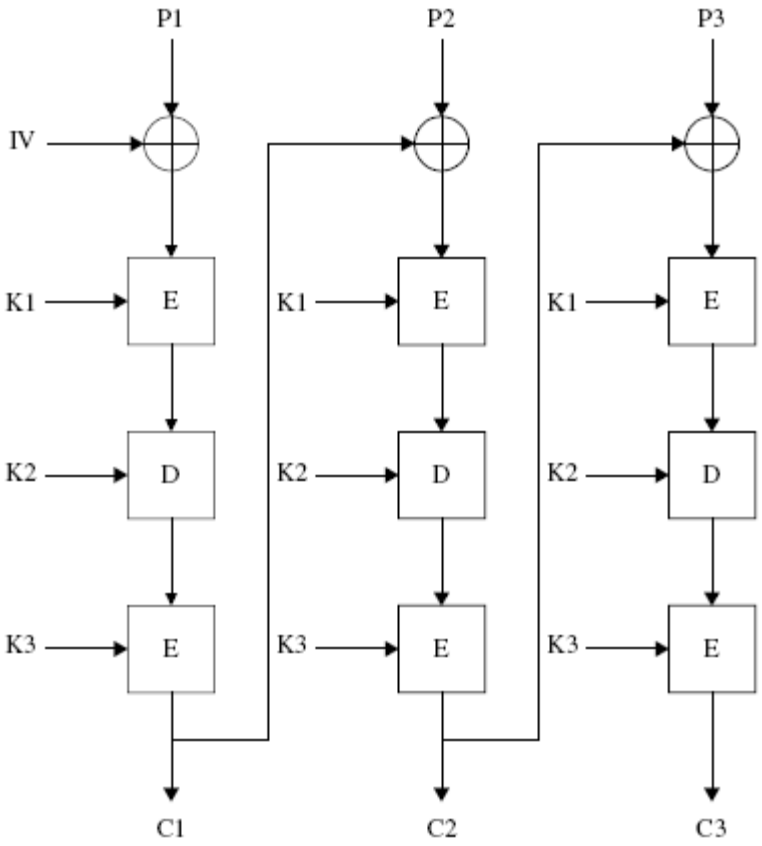


Figure 7.8 DES–EDE3–CBC algorithm.

تصدیق هویت

- SA الگوریتم محاسبه ICV را مشخص می کند.
- در ارتباط نقطه به نقطه:
  - الگوریتم مناسب شامل MAC بر اساس الگوریتم رمزنگاری متقارن(DES) یا تابع درهمساز یکطرفه(SHA-1، MD5) است
- در ارتباط یک نقطه به چند نقطه:
  - الگوریتم مناسب شامل الگوریتم امضای نامتقارن با تابع درهمساز است.

- فرستنده ICV را بر روی چهار فیلد رمزشده در ESP و Sequence number محاسبه می کند.

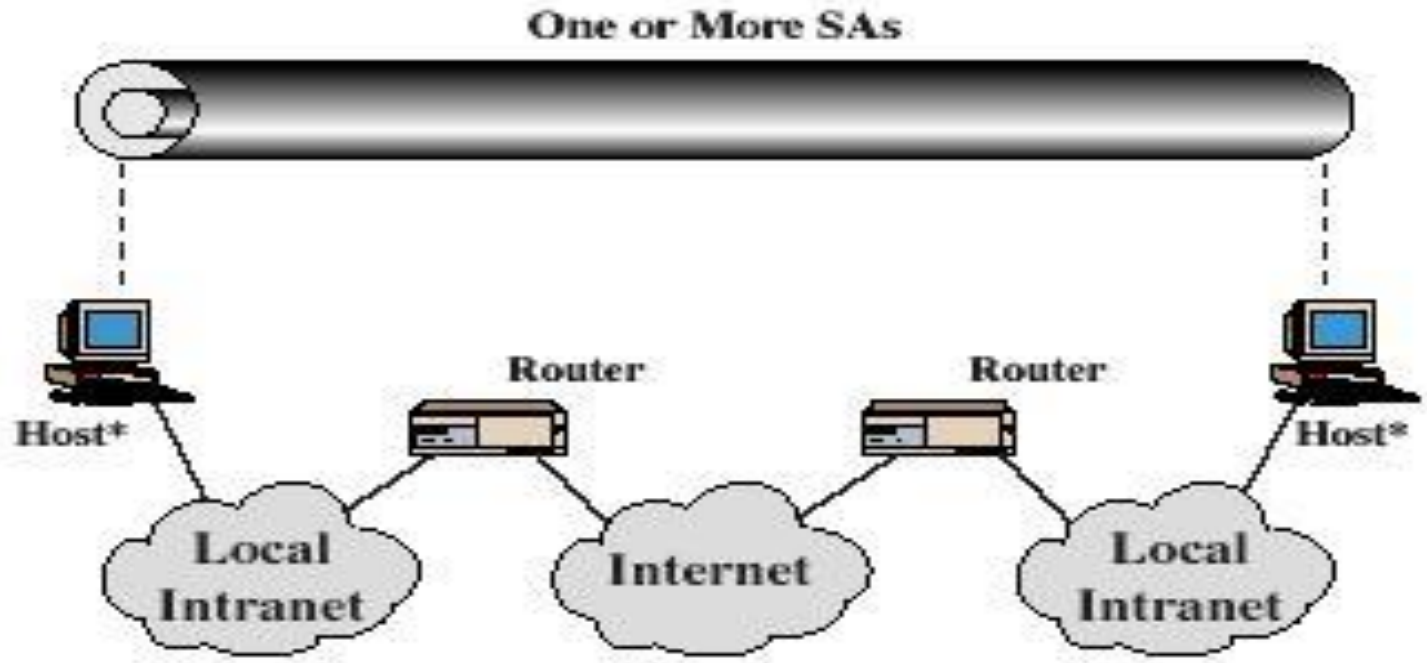
پروتکل مدیریت کلید در IPSec

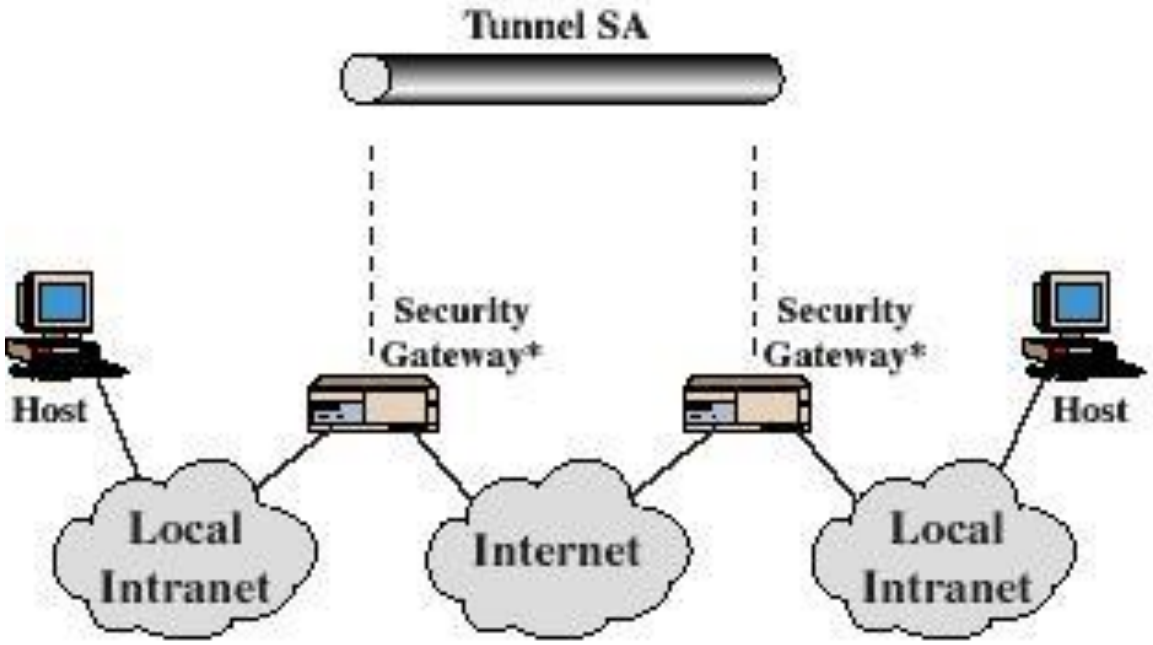
- مدیریت کلید شامل تعیین و توزیع کلید است.
- IKE پروتکلی است برای تبادل پارامترهای SA و برقراری SA در اینترنت.
- IKE شامل ISAKMP و پروتکل تبادل کلید Oakley

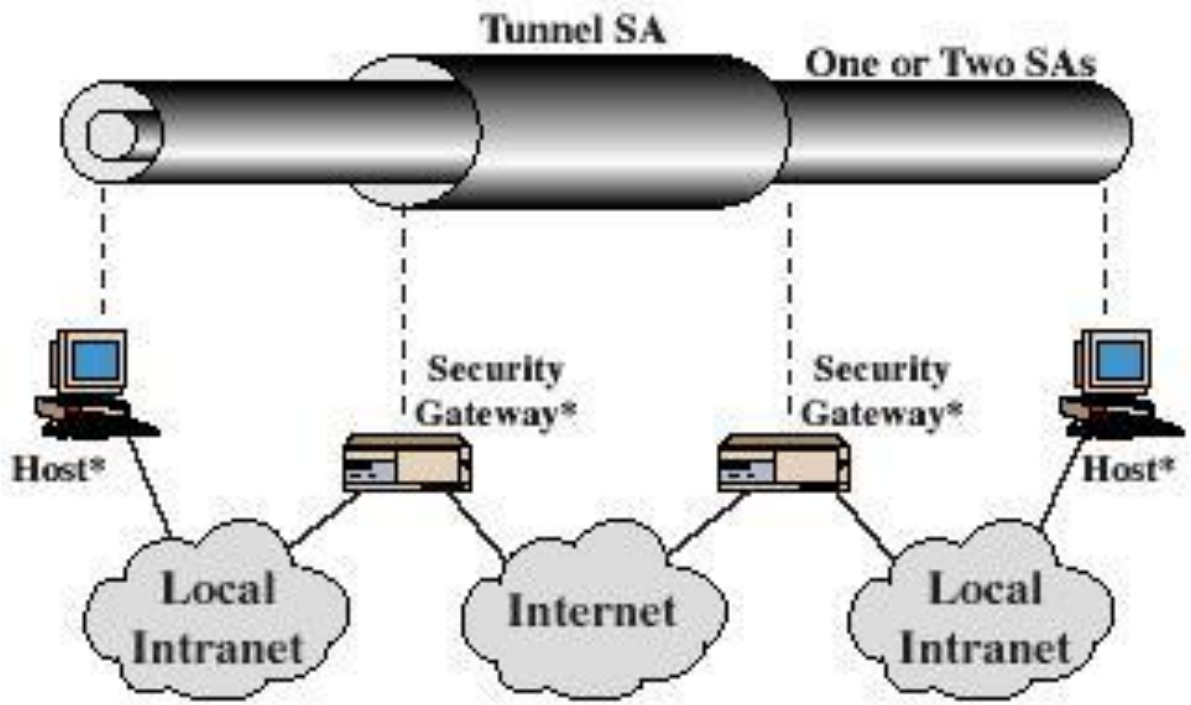
ترکیب SAها

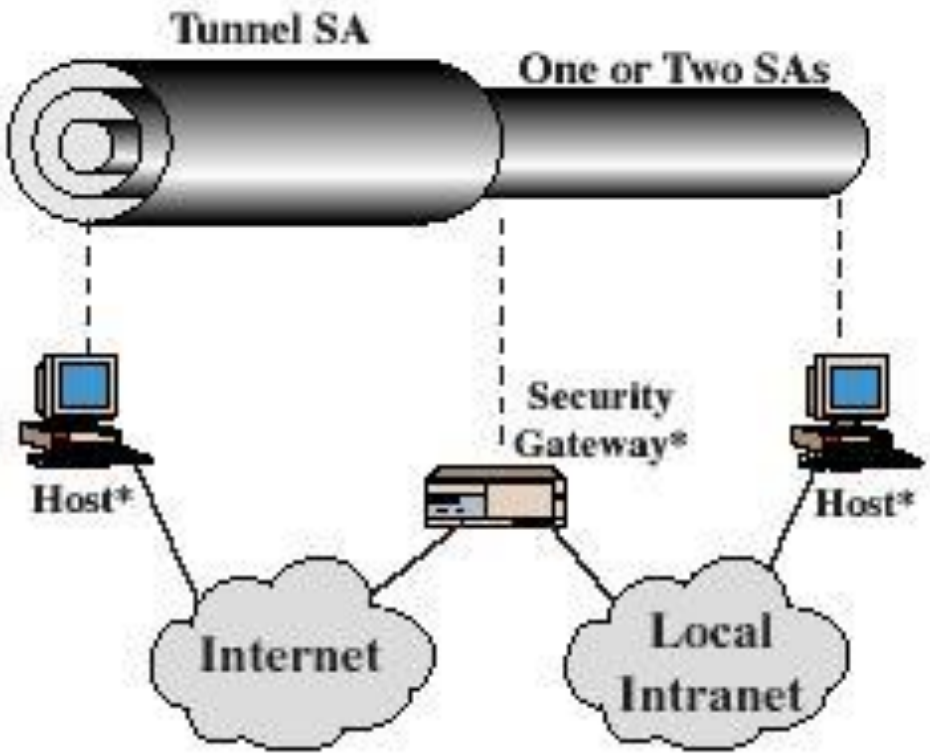
- با توجه به اینکه هر SA تنها یکی از سرویسهای AH یا ESP را پیاده سازی کرده است، برای استفاده از هر دو سرویس باید آنها را باهم ترکیب کرد

- ترکیبهای مختلف
  - پیاده سازی IPSec توسط host های متناظر
  - پیاده سازی IPSec توسط gateway ها
  - ترکیب دو حالت بالا


One or More SAs
Router
Router
Host*
Host*
Local Intranet
Internet
Local Intranet


Tunnel SA
Security Gateway*
Security Gateway*
Host
Host
Local Intranet
Internet
Local Intranet


Tunnel SA
One or Two SAs
Security Gateway*
Security Gateway*
Host*
Host*
Local Intranet
Internet
Local Intranet


Tunnel SA
One or Two SAs
Security
Gateway*
Host*
Host*
Internet
Local
Intranet

مدیریت کلید

- عموما به ٤ کلید سری، دو تا برای AH و دو تا برای ESP (در دو جهت) نیازمندیم. برای تولید و توزیع این کلیدها به یک مکانیزم مدیریت کلید نیازمندیم.

مدیریت کلید

- مدیریت کلید دستی : تنها در سیستم های ایستا و کوچک قابل استفاده است

- مدیریت خودکار :
  - پروتکل اتوماتیک و پیش فرض مدیریت و توزیع کلید IPSec اصطلاحا ISAKMP/Oakley نامیده می شود.

Internet Security Association
and Key Management Protocol

مدیریت کلید

- مدیریت کلید خودکار به نام ISAKMP/Oakley معروف است و شامل دو فاز است
  - پروتکل تعیین کلید Oakley : فرم توسعه یافته پروتکل Diffie-Hellman که ضعفهای آن را برطرف کرده است
    - Clogging Attack: منابع قربانی تلف می شود.
      - با استفاده از تعریف مفهومی تحت عنوان Cookie مشکل این حمله را برطرف می کند
    - Man-In-The-Middle-Attack
    - Replay Attack
      - با استفاده از Nonce با حمله های تکرار مقابله می کند.
  - پروتکل مدیریت کلید و SA در اینترنت(ISAKMP)
    - تعریف رویه ها و قالب بسته ها برای برقراری، مذاکره، تغییر یا حذف SA

Oakley

پروتکلی است که از Diffie-Hellman استفاده کرده و **تبادل کلید با تصدیق هویت** را انجام می دهد

- استفاده از **cookie**:
  - به منظور نامگذاری کلید و مقابله در برابر حمله clogging (DoS)
  - Cookie برابر است با خروجی تابعی درهم ساز بر روی مقداری محرمانه، آدرس مبدا و مقصد، پورت مبدا و مقصد
- استفاده از عددی تصادفی به نام **nonce** برای تازه نگه داشتن جلسات تبادل کلید و جلوگیری از حمله replay
- استفاده از چهارچوب**ISAKMP** برای تبادل کلید و برقراریSA

ISAKMP

- ISAKMP چهارچوبی برای مدیریت SA و تعیین کلید رمزنگاری در اینترنت فراهم می کند.
- ISAKMP پروتکلی برای ارائه سرویس های امنیتی در اینترنت است که امکان ایجاد SA برای چند پروتکل امنیتی را میدهد
- چهارچوب کاری شامل فرمت بسته، تبادلات، Payload و پروسه های تعریف شده است برای برقراری، مذاکره، تغییر و حذف SA، همچنین برای انتقال کلید و داده تصدیق هویت و تعیین پروتکل امنیتی

ISAKMP Header

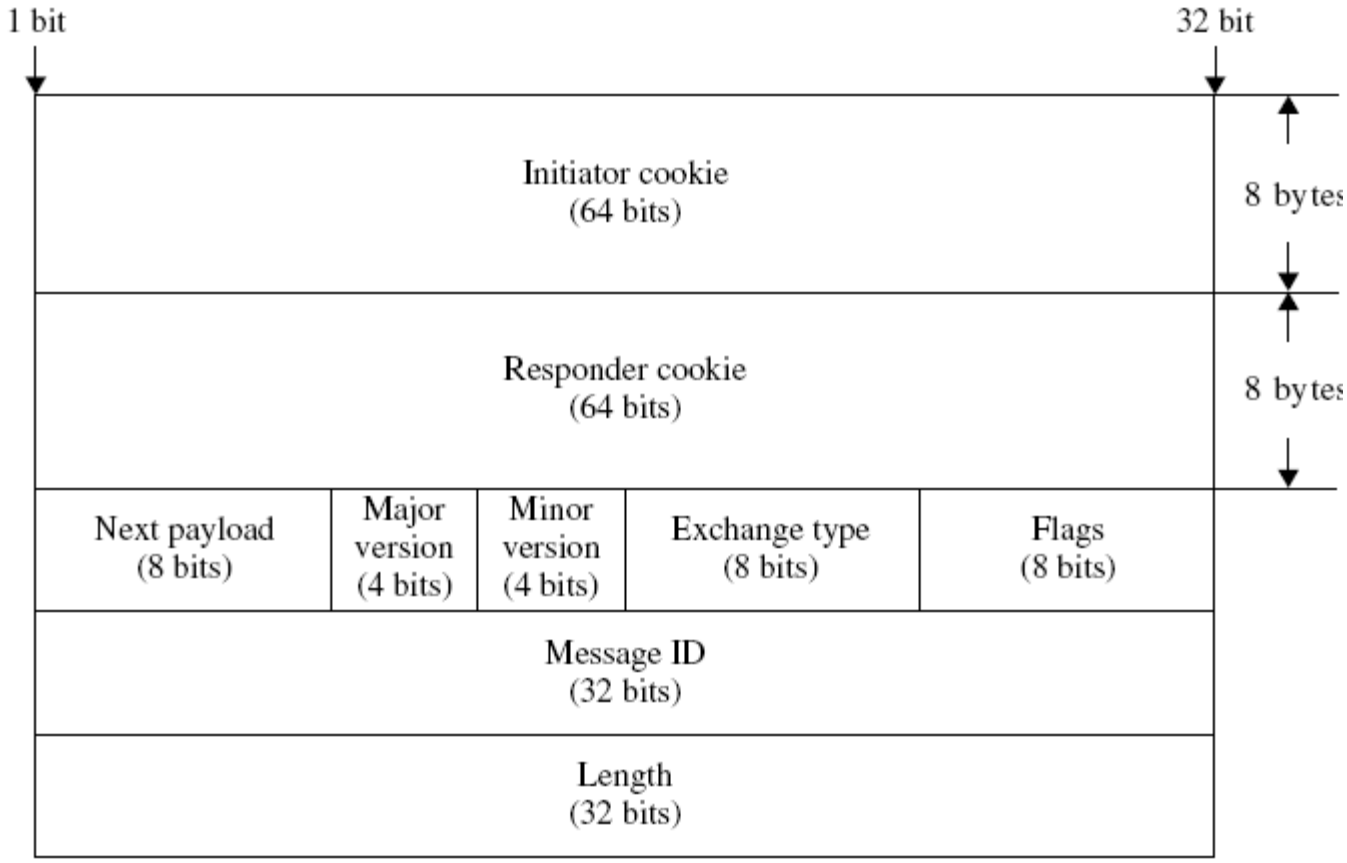


Figure 7.9 ISAKMP header format.

ISAKMP Header

1. *Initiator Cookie* (64 bits): cookie برای طرف شروع کننده برقراری، اعلام یا حذف SA
2. *Responder Cookie* (64 bits): cookie برای طرف مسئول درخواست برقراری اعلام یا حذف SA
3. *Next Payload* (8 bits): تعیین نوع اولین payload
4. *Major Version* (4 bits): تعیین بالاترین نسخه ISAKMP برابر 1
5. *Minor Version* (4 bits): تعیین پایین ترین نسخه ISAKMP برابر 0

5. *Exchange Type* (8 bits): تعیین نوع exchange استفاده شده
6. *Flags* (8 bits):
   - رمزنگاری در بیت 0: همه paylaod ها پس از تبادل کلید با الگوریتم تعریف شده رمزنگاری می شوند
   - commit در بیت 1: برای سنکرون کردن تبادل کلید و جلوگیری از گم شدن بسته ها و چندین بار ارسال مجدد
   - فقط تصدیق هویت در بیت 2 : بررسی جامعیت اطلاعات منتقل شده اما بدون رمزنگاری
   - سایر بیت ها برابر 0 قرار میگیرند.
7. *Message ID* (32 bits): برای تعیین حالت پروتکل در فاز ۲ که در فاز ۱ مقدار آن 0 است.
8. *Length* (32 bits) :برابر طول کل پیام(header || payload)

Payload Header عمومی

هر ISAKMP Payload با header عمومی آغاز می شود:

1. *Next Payload* (8 bits) :تعیین نوع payload بعدی، اگر آخرین Payload باشد مقدار صفر دارد
2. *Reserved* (8 bits): غیر استفاده و برابر مقدار 0
3. *Payload Length* (16 bits) : بیان کننده طول(شامل payload عمومی) در واحد بایت

ISAKMP paylaod انواع

1. Security Association Payload
2. Proposal Payload
3. Transform Payload
4. Key Exchange Payload
5. Identification Payload
6. Certificate Payload
7. Certificate Request Payload
8. Hash Payload
9. Signature Payload
10. Nonce Payload
11. Notification Payload
12. Delete Payload
13. Vendor ID Payload

Security Association Payload

این payload برای مذاکره پارامترهای امنیتی و تعیین DOI بکار می رود.

- مقدار 0 برای DOI در فاز ۱مشخص کننده ISAKMP عمومی است و برای هر پروتکل در فاز ۲ می تواند اجرا شود
- مقدار 1 برای DOI مختص IPSec است.

1. Next Payload field (8 bits): تعیین نوع payload بعدی
2. Reserved field (8 bits): بدون استفاده
3. Payload Length field (16 bits): بیان کننده طول Payload ها
4. Situation field (variable length): بیان کننده شرایط و سیاست هایی که مذاکره تحت آن انجام می شود

Proposal Payload

این payload(نوع ۲) برای ساخت پیام های ISAKMP برای مذاکره و برقراری SA بکار میرود و شامل اطلاعاتی است که در طول مذاکره برای امن کردن کانال استفاده می شود.

1. (8 bits) ***Next Payload field***: تعیین نوع payload بعدی. 0 اگر آخرین Proposal payload باشد و ۲ اگر Proposal payload بعدی هم وجود داشته باشد
2. (8 bits) ***Reserved field***: بدون استفاده
3. (16 bits) ***Payload Length field***: بیان کننده طول این payload شامل generic payload header و Proposal Payload و payloads Transform مرتبط با این Proposal
4. (8 bits) ***Proposal # field***: شماره proposal
5. (8 bits) ***Protocol-id field***: شناسه پروتکل استفاده شده در این مذاکره مثل IPsec ESP، IPsec AH، OSPF ، TLS وغیره
6. (8 bits) ***SPI Size***: طول SPI از ۰ تا ۱۶
7. (8 bits) ***# of Transform***: تعداد Transform ها برای Proposal که در Transform proposal هستند.
8. (variable) ***SPI field***: مقدار SPI فرستنده

Transform Payload

- این payload(نوع ۳) شامل اطلاعات استفاده شده در طول مذاکره Security Association مانند مکانیزم هایی برای امن کردن کانال است همچنین اطلاعات خصوصیات امنیتی را برای یک transform خاص نگهداری می کند.

1. (8 bits) ***Next Payload field***: تعیین نوع payload بعدی. 0 اگرآخرین Transform payload باشد و ۳ اگر Transform payload بعدی هم وجود داشته باشد
2. (8 bits) ***Reserved field***: بدون استفاده حاوی مقدار 0
3. (8 bits) ***Transform # field***: شماره proposal هر Transform شماره های منحصر بفرد دارد
4. (8 bits) ***Transform-id field***: شناسه Transform برای پروتکل ذکر شده در Proposal

Key Exchange Payload

- این payload(نوع ۴) از چندین تکنیک تبادل کلید مانند Diffie-Hellman ،Oakley و تکنیک های بر اساس RSA مانند PGP حمایت می کند

1. (8 bits) ***Next Payload field***: تعیین نوع payload بعدی. 0 اگر آخرین payload باشد
2. (8 bits) ***Reserved field***: بدون استفاده حاوی مقدار 0
3. (16 bits) ***Payload Length field***: بیان کننده طول این payload شامل generic payload header
4. (variable length) ***Key Exchange Data field***: داده ای است که برای تولید کلید جلسته استفاده می شود که توسط DOI و الگوریتم تبادل کلید تفسیر می شود

Identification Payload

- این payload(نوع ۵) شامل داده های DOI برای تبادل اطلاعات هویتی است. این اطلاعات برای تعیین هویت طرف های ارتباط و صحت اطلاعات بکارمی رود.

1. (8 bits) ***Next Payload field***: تعیین نوع payload بعدی. 0 اگر آخرین payload باشد
2. (8 bits) ***Reserved field***: بدون استفاده حاوی مقدار 0
3. (8 bits) ***ID type field***: بیان کننده نوع identification استفاده شده.
4. (24 bits) ***DOI specific ID Data field***: حاوی داده های هویتی مربوط به DOI، در صورت عدم استفاده مقدار 0 دارد
5. (8 bits) ***Transform-id field***: شناسه Transform برای پروتکل ذکر شده در Proposal
6. (variable length) ***Identification Data field***: حاوی اطلاعات هویتی

Certificate Payload

Certificate Type	Value
NONE	0
PKCS #7 wrapped X.509 certificate	1
PGP Certificate	2
DNS Signed Key	3
X.509 Certificate-Signature	4
X.509 Certificate-Key Exchange	5
Kerberos Tokens	6
Certificate Revocation List (CRL)	7
Authority Revocation List (ARL)	8
SPKI Certificate	9
X.509 Certificate-Attribute	10
Reserved	11–255

- این payload(نوع ۶) برای انتقال گواهی از طریق ISAKMP است. هرگاه گواهی های از طریق دایرکتوری و به صورت توزیع شده قابل دسترس نباشند از آن استفاده می شود.

1. ***Next Payload field*** (8 bits): تعیین نوع payload بعدی. 0 اگر آخرین payload باشد
2. ***Reserved field*** (8 bits): بدون استفاده حاوی مقدار 0
3. ***Payload Length field*** (16 bits): بیان کننده طول این payload شامل generic payload header
4. ***Certificate Encoding field*** (8 bits): بیان کننده نوع گواهی نامه و اطلاعات مرتبط با گواهی نامه
5. ***Certificate Data field*** (variable length): بیان کننده encoding گواهی نامه

Certificate Request Payload

- این payload(نوع ۷) برای درخواست گواهی از طریق ISAKMP است. هرگاه گواهی های از طریق دایرکتوری و به صورت توزیع شده قابل دسترس نباشند از آن استفاده می شود. پاسخ دهنده در پاسخ باید گواهی خود را ارسال کند. در صورت نیاز به چند گواهی باید چند payload ارسال شود.

1. (8 bits) ***Next Payload field***: تعیین نوع payload بعدی. 0 اگر آخرین payload باشد
2. (8 bits) ***Reserved field***: بدون استفاده حاوی مقدار 0
3. (16 bits) ***Payload Length field***: بیان کننده طول این payload شامل generic payload header
4. (8 bits) ***Certificate Type field***: شامل encoding نوع گواهی نامه درخواست کرده
5. (variable length) ***Certificate Authority field***:شامل encoding های CA مورد قبول

Hash Payload

- این payload(نوع ۸) شامل خروجی تابع درهم ساز بر روی قسمتی از پیام/ حالت ISAKMP است که برای بررسی جامعیت داده های پیام ISAKMP یا تصدیق هویت طرف های مذاکره استفاده می شود.

1. (8 bits) ***Next Payload field***: تعیین نوع payload بعدی. 0 اگر آخرین payload باشد
2. (8 bits) ***Reserved field***: بدون استفاده حاوی مقدار 0
3. (16 bits) ***Payload Length field***: بیان کننده طول این payload شامل generic payload header
4. (variable length) ***Hash Data field***: شامل خروجی تابع درهم ساز بر روی پیام ISAKMP

Signature Payload

- این payload(نوع ۹) برابر با خروجی تابع امضای دیجیتال بر روی قسمت هایی از پیام/ حالت ISAKMP است که برای بررسی جامعیت داده های پیام/ حالت ISAKMP یا سرویس عدم انکار بکارمی رود.

1. ***Next Payload field*** (8 bits): تعیین نوع payload بعدی. 0 اگر آخرین payload باشد
2. ***Reserved field*** (8 bits): بدون استفاده حاوی مقدار 0
3. ***Payload Length field*** (16 bits): بیان کننده طول این payload شامل generic payload header
4. ***Signature Data field*** (variable length): شامل خروجی تابع امضای دیجیتال بر روی پیام ISAKMP

Nonce Payload

- این payload(نوع ۱۰) شامل داده ای تصادفی است که برای تازه نگهداشتن در طول پروسه تبادل کلید و جلوگیری از حمله replay است. nonce به عنوان بخشی از داده تبادل کلید یا به عنوان بخشی از payload منتقل می شود.

1. ***Next Payload field*** (8 bits): تعیین نوع payload بعدی. 0 اگر آخرین payload باشد
2. ***Reserved field*** (8 bits): بدون استفاده حاوی مقدار 0
3. ***Payload Length field*** (16 bits): بیان کننده طول این payload شامل generic payload header
4. ***Nonce Data field*** (variable length): شامل داده تصادفی تولید شده توسط فرستنده

Notification Payload

- این payload(نوع ۱۱) شامل داده های ISAKMP و DOI است که برای انتقال داده اطلاعاتی مانند شرایط خطا بکار می رود. می توان چندین Notification Payloads در یک پیام ISAKMP قرار داد. این payload ها توسط شروع کننده و پاسخ دهنده cookie شناسایی می شوند.

1. (8 bits) ***Next Payload field***: تعیین نوع payload بعدی. 0 اگر آخرین payload باشد
2. (8 bits) ***Reserved field***: بدون استفاده حاوی مقدار 0
3. (16 bits) ***Payload Length field***: بیان کننده طول این payload شامل generic payload header
4. (32 bits) ***Domain of Interpretation field***: تعیین کننده DOI که Notification تحت آن اتفاق می افتد
5. (8 bits) ***Protocol-id field***: شناسه پروتکل برای Notification جاری مانند ISAKMP، IPSec ESP، IPSec Ahو غیره
6. (8 bits) ***SPI Size field***: طول SPI تعریف شده توسط protocol_id
7. (16 bits) ***Notify Message Type field***: بیان کننده نوع notification
8. (variable length) ***Security Parameter Index (SPI) field*** :
9. (variable length) ***Notification Data field***: حاو ی اطلاعات یا خطای انتقال یافته

Delete Payload

این payload (نوع ۱۲) شامل شناسه SA بوده که فرستنده از پایگاه داده خود حذف کرده، بنابراین فرستنده دیگر معتبر نیست. می توان چندین SPI اما با یک پروتکل را در این payload قرار داد.

1. (8 bits) ***Next Payload field***: تعیین نوع payload بعدی. 0 اگر آخرین payload باشد
2. (8 bits) ***Reserved field***: بدون استفاده حاوی مقدار 0
3. (16 bits) ***Payload Length field***: بیان کننده طول این payload شامل generic payload header
4. (32 bits) ***Domain of Interpretation field***:بیان کننده DOI که عمل حذف تحت آن انجام می شود
5. (8 bits) ***Protocol-id field*** : ISAKMP می تواند SA را برای چند پروتکل ایجاد کند، این فیلد بیان کننده پایگاه داده ای است که درخواست حذف روی ان اعمل می شود.
6. (8 bits) ***SPI Size field***: طول SPI تعریف شده توسط protocol_id
7. (16 bits) ***# of SPIs field***: تعداد SPI ها در Delete Payload
8. (variable length) ***Security Parameter Indexes field***: بیان کننده SA مورد نظر برای حذف

Vendor ID Payload

- این payload (نوع ۱۲) شامل ثابت تعریف شده توسط فروشنده است که فروشنده از آن برای شناسایی و تشخیص نمونه های پیاده سازی شده استفاده می کند.

1. (8 bits) ***Next Payload field***: تعیین نوع payload بعدی. 0 اگر آخرین payload باشد
2. (8 bits) ***Reserved field***: بدون استفاده حاوی مقدار 0
3. (16 bits) ***Payload Length field***: بیان کننده طول این payload شامل generic payload header
4. (variable length) ***Vendor ID field***: شامل فروشندگان vendor-id را از تابع درهم ساز بدون کلید بر روی نام محصول و نسخه محصول بدست می آورند.

تبادلات ISAKMP

تبادلات ISAKMP محتوا و ترتیب پیام ها را در طول ارتباط تعیین می کند. تفاوت اصلی میان تبادلات در ترتیب پیام ها و تریبت payload ها در هر پیام است.

1. **Base Exchange**: امکان ارسال اطلاعات تبادل کلید و داده تصدیق هویت را همزمان داده و در ازای عدم حفاظت از هویت دو طرف، باعث کاهش تعداد رفت وبرگشت ها می شود
2. **Identity Protection Exchange**: امکان جداسازی اطلاعات تبادل کلید از شناسه و اطلاعات تصدیق هویت بکار رفته و در ازای افزایش ۲ پیام، از هویت دو طرف ارتباط محافظت می کند.

3. **Authentication Only Exchange**:تنها امکان انتقال اطلاعات تصدیق هویت را داده که باعث کاهش سربار محاسبه کلید می شود. در فاز اول، این تبادل با ISAKMP SA رمز شده اما سایر اطلاعات رمزنگاری نمی شوند.
4. **Aggressive Exchange**: امکان انتقال payload های security association، key exchange و تصدیق هویت را داده و در ازای عدم حفاظت از هویت دو طرف، باعث کاهش تعداد رفت وبرگشت ها می شود.
4. **Informational Exchange**:انتقال یک طرفه برای مدیریت SA است. اگر در فاز ۱، قبل از تبادل کلید رخ دهد، از ان حفاظت نشده اما اگر تبادل کلید صورت گرفته باشد یا ISAKMP SA برقرار شده باشد، از این تبادل حفاظت خواهد شد.

پردازش ISAKMP Payload

1. ISAKMP Header Processing

مبدأ:

- ایجاد cookie متناسب
- تعیین خصوصیات امنیتی جلسه
- ساخت ISAKMP header

مقصد:

- بررسی cookie فرستنده و پاسخ دهنده
- بررسی فیلد Next Payload
- بررسی بالاترین و پایین ترین نسخه
- بررسی فیلد Exchange Type
- بررسی فیلد flags
- بررسی فیلد Message ID

2. Generic Payload Header Processing

مبدا:

- قرار دادن مقدار نوع payload بعدی در فیلد Next Payload
- قرار دادن مقدار 0 در فیلد Reserved
- قرار دادن طول در مقدار فیلد Payload Length

مقصد:

- بررسی فیلد Next Payload
- بررسی فیلد Reserved

3. Security Association Payload Processing

مبدأ:

- تعیین DOI برای مذاکره در حال انجام
- تعیین شرایط DOI
- تعریف poposal و transform ها
- ساخت Security Association payload

مقصد:

- بررسی کند که آیا DOI حمایت می شود
- بررسی کند که آیا شرایط مطرح شده قابل حمایت است یا خیر
- پردازش باقیمانده payload (Proposal، Transform)
- درصورت مورد قبول واقع نشدن Proposal، رویداد در فایل بازرسی ثبت می شود.
- کلیه Information Exchange با Notification payload با نوع پیام No-Proposal-Chosen به کلیه فرستنده های ارسال می شود.

4. Proposal Payload Processing

مبدأ:

- تعریف پروتکل برای proposal
- تعریف تعداد proposal و تعداد transform در هر proposal
- تولید عدد تصادفی منحصر بفرد SPI و ساخت Proposal payload

مقصد:

- بررسی اینکه آیا proposal حمایت می شود
- بررسی فیلد Protocol-ID
- بررسی اینکه آیا SPI معتبر است یا خیر
- بررسی اینکه آیا proposal بدرستی انجام شده یا خیر
- پردازش Proposal payload و trasnform payload بر اساس فیلد Next Payload

5. Transform Payload Processing

مبدأ:

- تعیین شماره Transform، تعداد Transform و ساخت Transform payload

مقصد:

- تعیین اینکه آیا Transform حمایت می شود.
- اگر مقدار فیلد Transform-ID شامل مقدار ناشناخته و حمایت نشده باشد، از transform payload چشم پوشی می شود.
- اطمینان از اینکه Transforms مطابق با جزئیات Transform Payload و برقراری SA است.
- پردازش Proposal payload و trasnform payload تعریف شده در فیلد Next Payload

6. Key Exchange Payload Processing

مبدأ:

- تعریف تبادل کلید مورد استفاده DOI
- تعریف استفاده فیلد Key Exchange Data field و ساخت Key Exchange payload

مقصد:

- بررسی اینکه آیا Key Exchange حمایت می شود یا نه. در صورت عدم حمایت، پیام دور انداخته می شود
- رویداد اطلاعات غیر معتبر ممکن است در سیستم ثبت شود.
- Informational Exchange با Notification payload با پیام Invalid-Key-Information به فرستنده ارسال می‌شود.

7. Identification Payload Processing

مبدأ:

- تعریف اطلاعات شناسایی همانطور که در DOI تعریف شده است.
- تعریف استفاده فیلد Identification Dataهمانطور که در DOI تعریف شده است.
- ساخت Identification payload

مقصد:

- بررسی اینکه آیا Identification payload حمایت می شود یا نه. در صورت عدم حمایت، پیام دور انداخته می شود
- Informational Exchange با Notification payload با پیام Invalid-ID-Information به فرستنده ارسال می‌شود.

8. Certificate Payload Processing

مبدأ:

- تعیین Certificate Encoding بیان شده در DOI
- اطمینان از وجود گواهی نامه بر اساس Certificate Encoding

مقصد:

- بررسی اینکه آیا Certificate Encodingحمایت می شود یا نه. در صورت عدم حمایت، payload پیام دور انداخته می شود
- پردازش فیلد Certificate Data که در صورت فرمت نامناسب آن payload پیام دور انداخته می شود

9. Certificate Request Payload Processing

مبدأ:

- تعیین Certificate Encoding
- تعیین نام CA مورد قبول
- ساخت Certificate Request payload

مقصد:

- بررسی اینکه آیا Certificate Encodingحمایت می شود یا نه. در صورت عدم حمایت، payload پیام دور انداخته می شود
- بررسی اینکه آیا CA برای Certificate Encoding خاص حمایت می شود یا نه. اگر CA فرمت درستی نداشته باشد، payload پیام دور انداخته می شود.
- پردازش فیلد Certificate Data که در صورت فرمت نامناسب آن payload پیام دور انداخته می شود
- پردازش Certificate Request که اگر از نوع Certificate Request با CA بیان شده در دسترس نباشد، payload پیام دور انداخته می شود

10. Hash Payload Processing

مبدأ: تعیین تابع درهم ساز و تعیین مورد استفاده آن بر اساس DOI و ساخت Hash payload

مقصد:

- بررسی اینکه آیا از تابع درهم ساز حمایت می شود یا خیر. در صورت عدم حمایت پیام دور انداخته می شود.
- اجرای تابع درهمساز بر اساس DOI یا پروتکل تبادل کلید. در صورت عدم موفقیت تابع درهم ساز، پیام دور انداخته می شود..

11. Signature Payload Processing

مبدأ:

- تعیین تابع امضا و تعیین مورد استفاده آن بر اساس DOI و ساخت Signature payload

مقصد:

- بررسی اینکه آیا از امضا حمایت می شود یا خیر. در صورت عدم حمایت پیام دور انداخته می شود.
- اجرای تابع امضا بر اساس DOI یا پروتکل تبادل کلید. در صورت عدم موفقیت تابع درهم ساز، پیام دور انداخته می شود..

12. **Nonce Payload Processing**

مبدأ: تولید عدد تصادفی به عنوان nonce و ساخت Nonce payload

مقصد: پروسیجر پردازش nonce توسط نوع تبادل و DOI و تبادل کلید مشخص می شود.

13. **Notification Payload Processing**: پیام اطلاعاتی با Notify Payload امکان اعلام خطاهای رخ داده را به طرف های ارتباط می دهد.

مبدأ:

- تعیین DOI برای این Notification و تعیین فیلد Protocol-ID
- تعیین نوع پیام بر اساس خطا یا حالت پیام و تعیین SPI
- تعیین اینکه آیا Notification Data دیگری نیز وجود دارد یا خیر
- ساخت Notification Payload

مقصد:

- تعیین اینکه آیا رمزنگاری بر روی Informational Exchange رمزنگاری انجام می شود یا نه(با بررسی بیت رمزنگاری و بیت فقط تصدیق هویت در header)
- بررسی اینکه آیا DOI حمایت می شود یا خیر. بررسی اعتبار SPI
- بررسی اعتبار نوع پیام Notify و پردازش Notification payload و انجام عمل مناسب بر اساس سیاست ها

14. **Delete Payload Processing**: در صورتی که یک ارتباط در خطر کشف باشد، SA را حذف کرده و SA جدیدی برقرار می کنیم.

مبدأ:

- تعیین DOI، تعیین Protocol-ID، تعیین سایز SPI بر ساس فیلد Protocol-id
- تعیین شماره SPI ای که می خواهیم حذف کنیم و تعیین SPI های مرتبط و ساخت Delete payload

مقصد:

- بررسی اینکه آیا از DOI و Protocol-ID حمایت می شود یا نه.
- بررسی اعتبار هر SPI ذکر شده در Delete payload
- پردازش Delete payload و انجام عمل متناسب با سیاست داخلی