IP Security

DR. VASUDHA ARORA

VASUDHA.ARORA@GDGU.ORG, VASUDHARORA6@GMAIL.COM

DEPARTMENT OF COMPUTER SCIENCE AND ENGINEERING

GD GOENKA UNIVERSITY, GURUGRAM

Outline

Introduction to TCP/IP Protocol

IP Security Overview

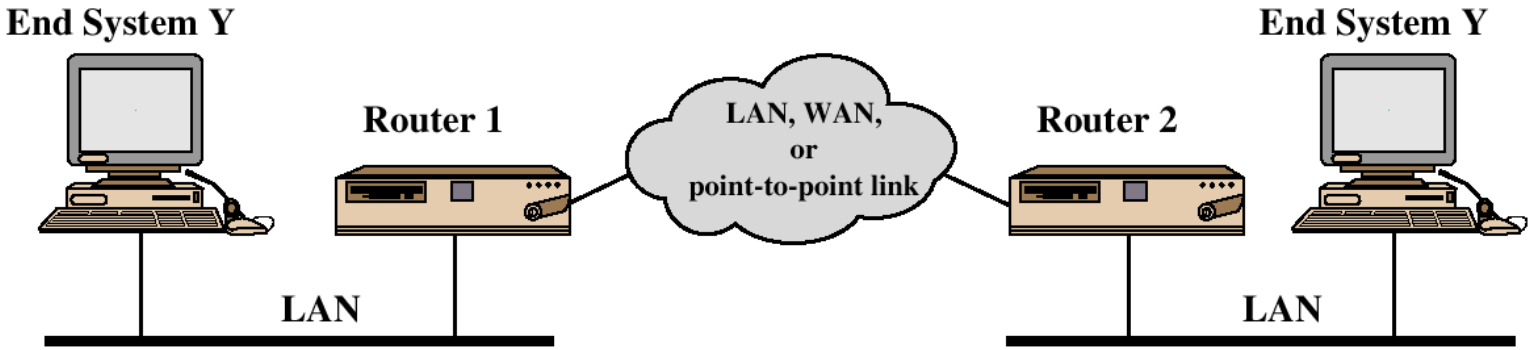
IP Security Architecture

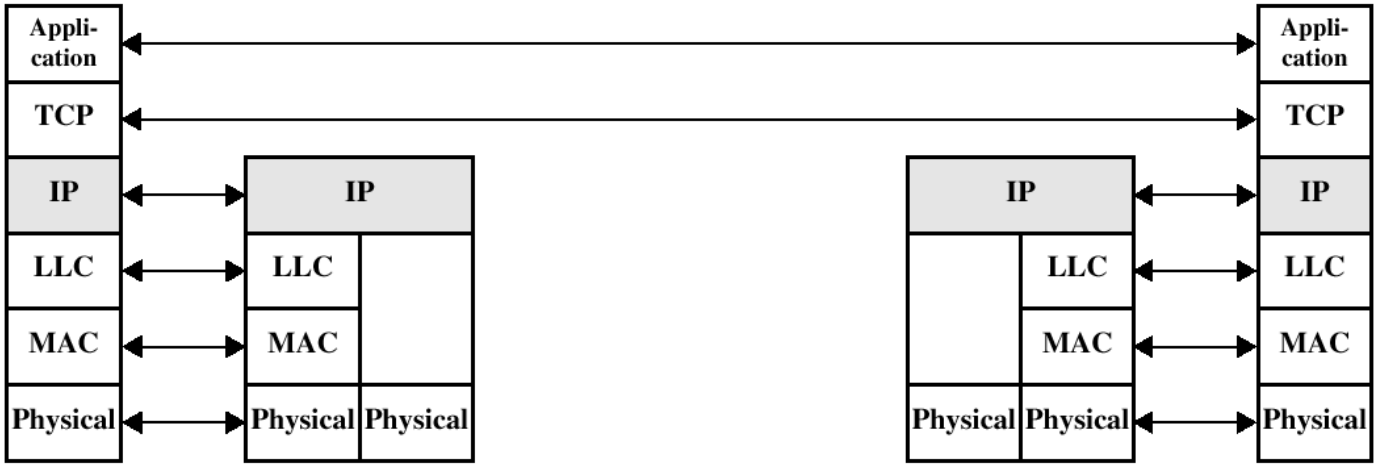
Authentication Header

Encapsulating Security Payload

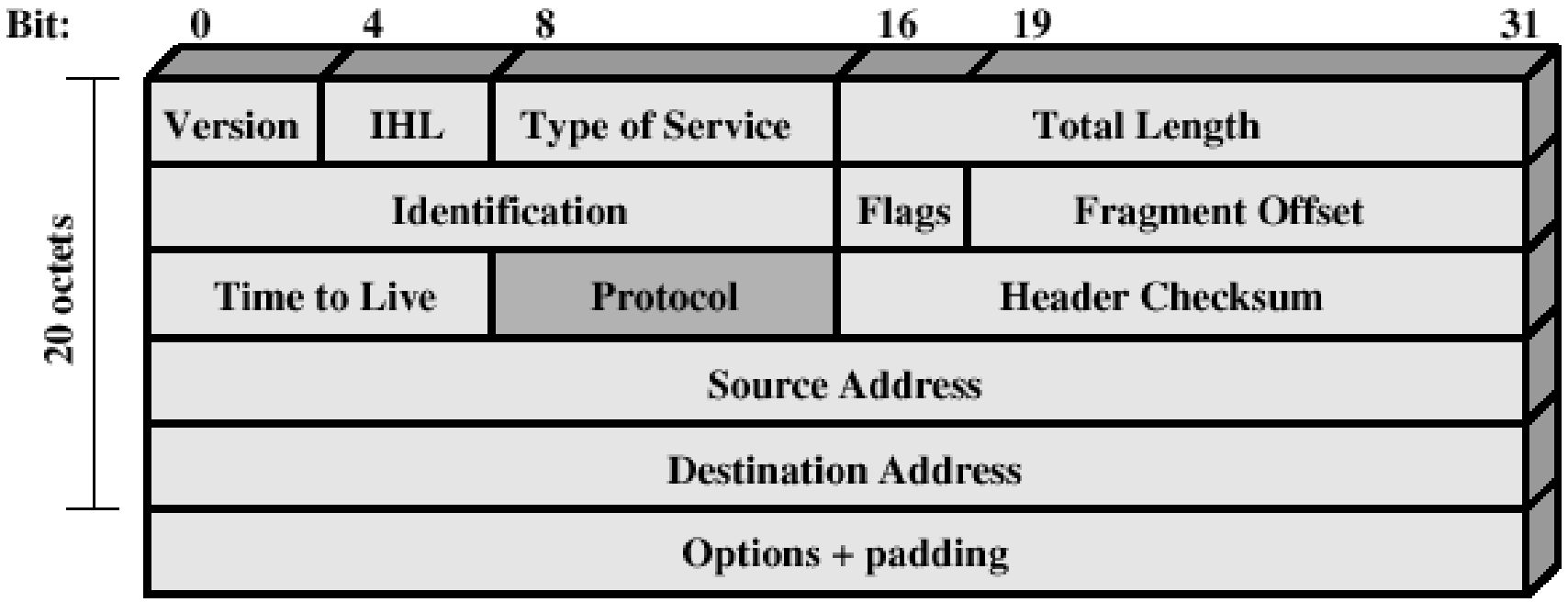
Key Management

TCP/IP Example

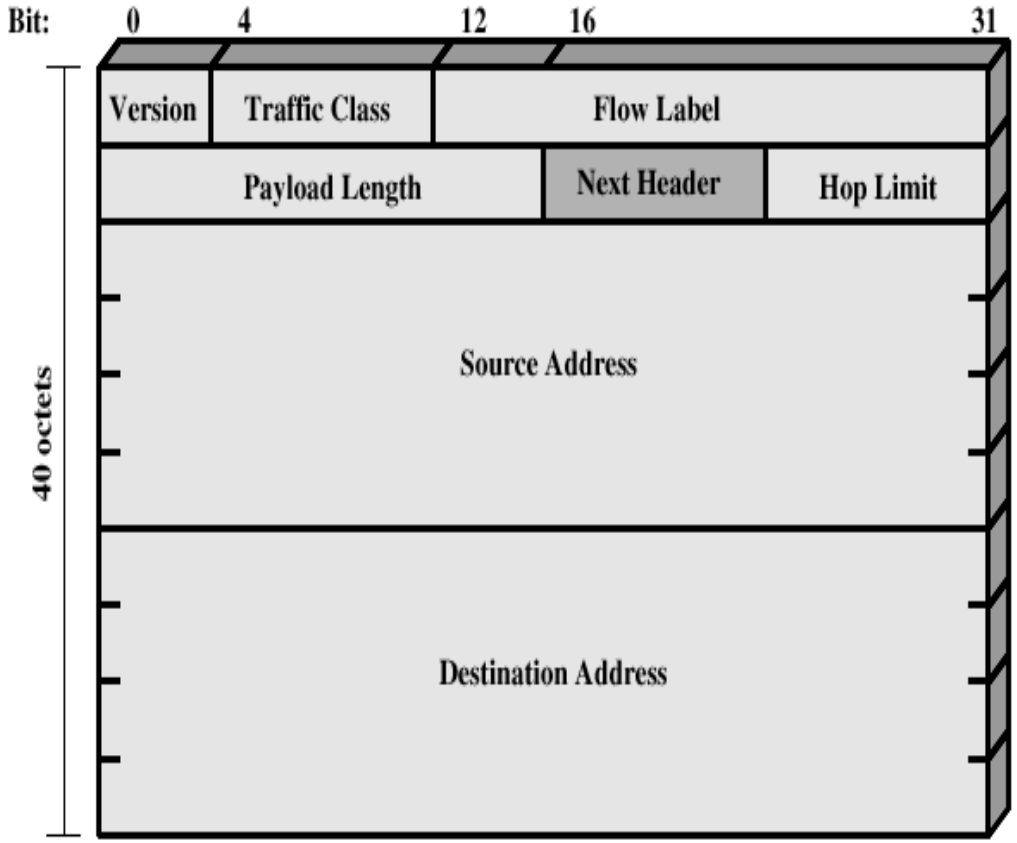


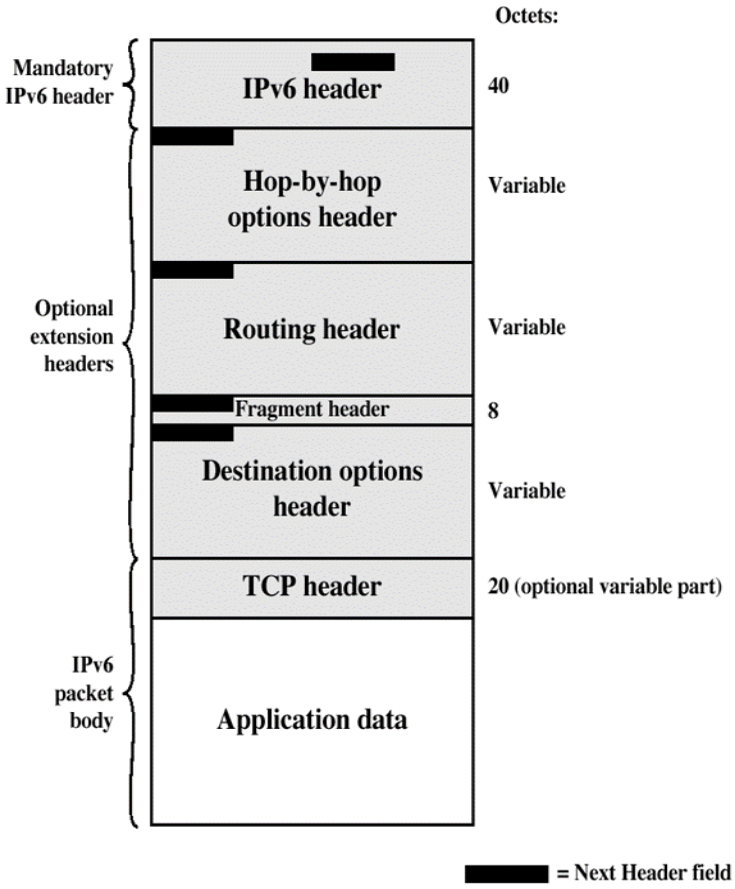


IPv4 Header


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

IPv6 Header


Bit:
0
4
12
16
31
Version
Traffic Class
Flow Label
Payload Length
Next Header
Hop Limit
Source Address
Destination Address
40 octets


Octets:
Mandatory IPv6 header
IPv6 header
40
Optional extension headers
Hop-by-hop options header
Variable
Routing header
Variable
Fragment header
8
Destination options header
Variable
TCP header
20 (optional variable part)
IPv6 packet body
Application data
= Next Header field

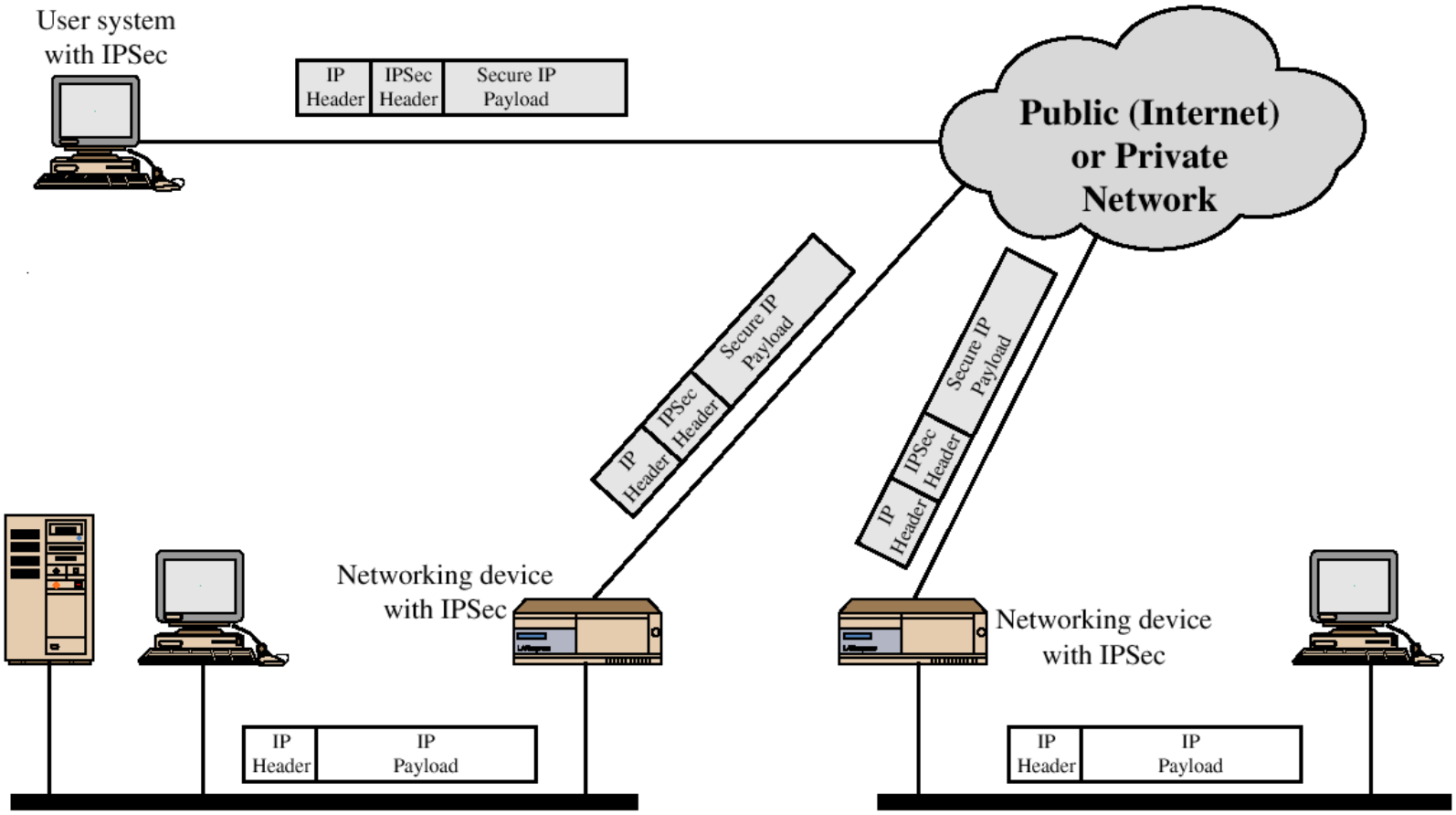
IP Security Overview

IPSec is not a single protocol.

Instead, IPSec provides a set of security algorithms plus a general framework that allows a pair of communicating entities to use whichever algorithms to provide security appropriate for the communication.

- Applications of IPSec
  - Secure branch office connectivity over the Internet
  - Secure remote access over the Internet
  - Establishing extranet and intranet connectivity with partners
  - Enhancing electronic commerce security

IP Security Scenario

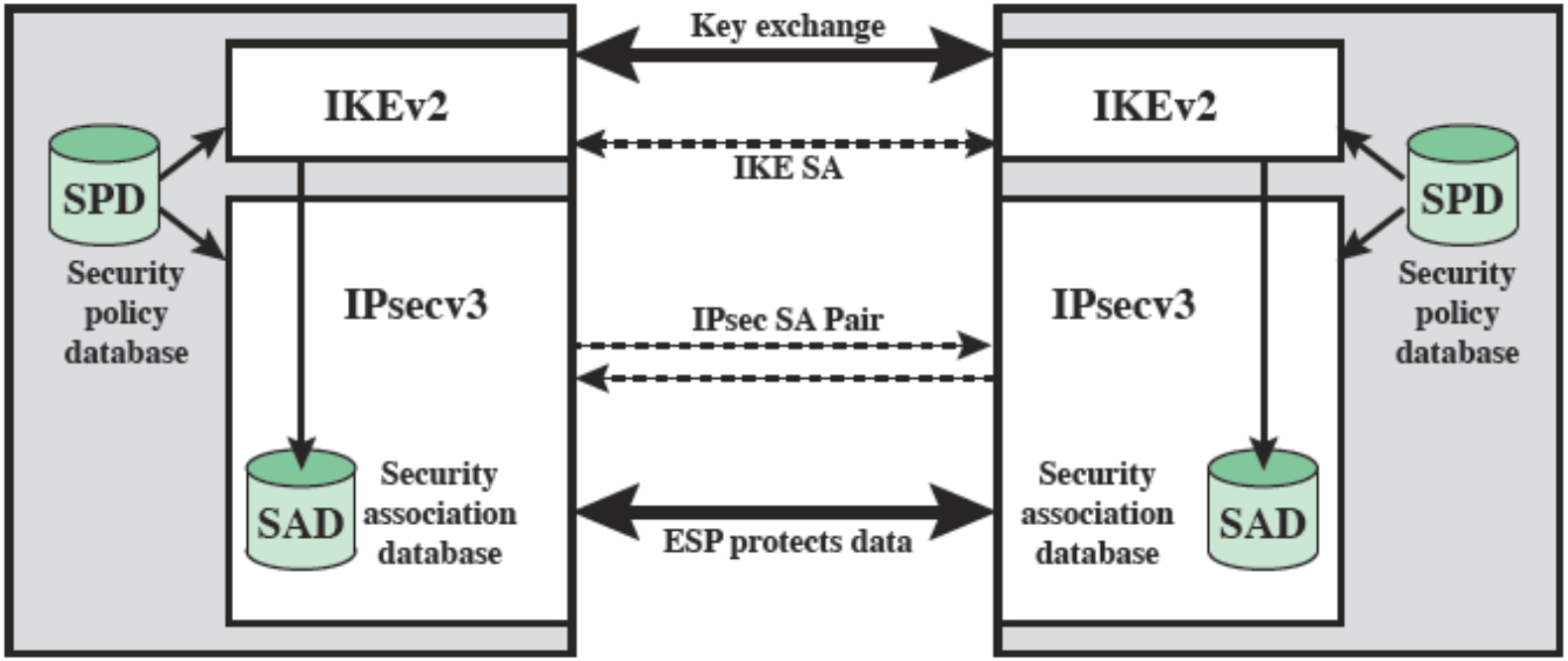


IP Security Overview

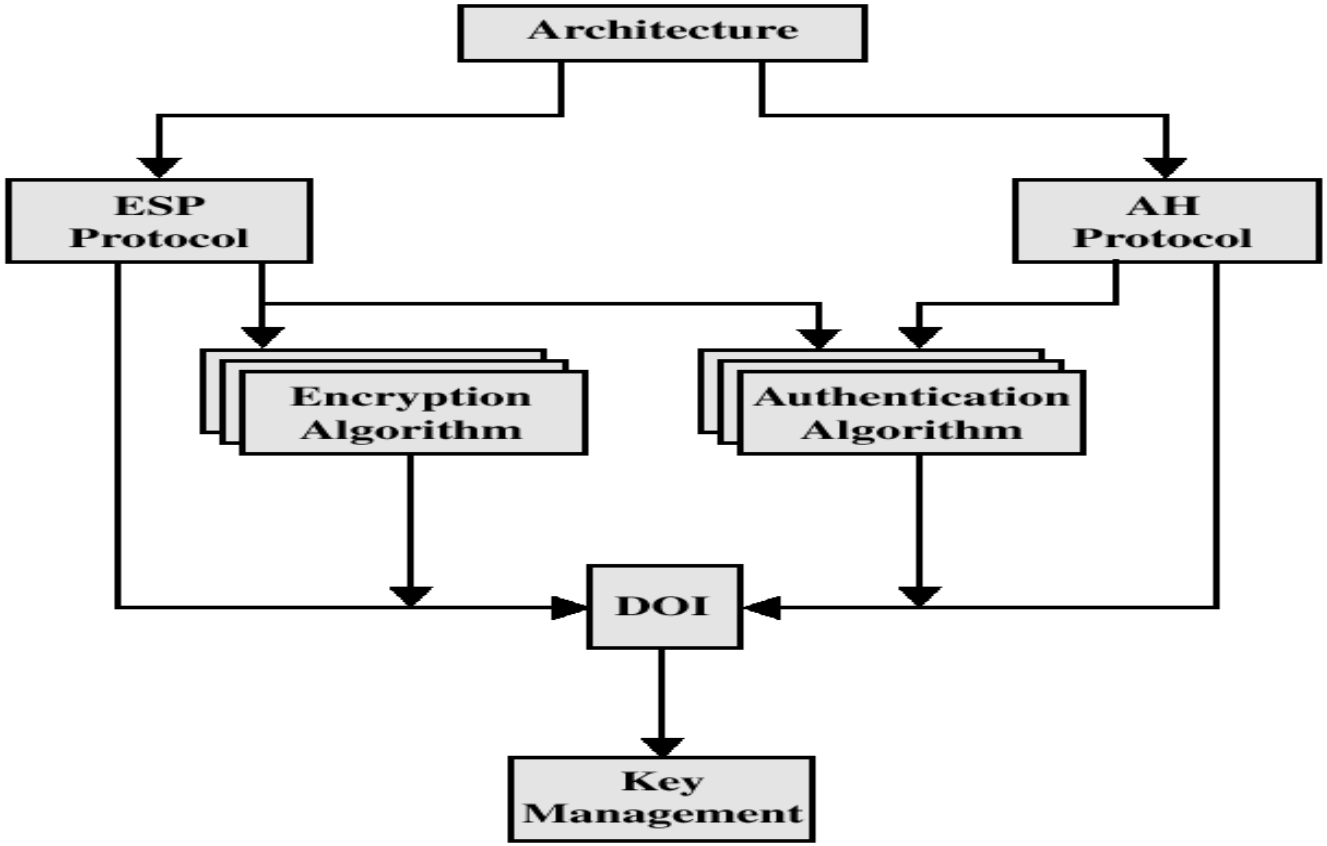
IPSec can assure that:

- A router or neighbor advertisement comes from an authorized router
- A redirect message comes from the router to which the initial packet was sent
- A routing update is not forged
- Provide security for individual users

IPSec General Architecture (Big Picture)



IPSec Document Overview



IPSec Services

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

Security Associations (SA)

A one way relationsship between a sender and a receiver.

Identified by three parameters:

- Security Parameter Index (SPI)
- IP Destination address
- Security Protocol Identifier

As SAs are one way. A minimum of two SAs are required for a single IPSec connection.

Security Associations

SAs contain parameters including:

- Authentication algorithm and algorithm mode
- Encryption algorithm and algorithm mode
- Key(s) used with the authentication/encryption algorithm(s)
- Lifetime of the key
- Lifetime of the SA
- Source Address(es) of the SA
- Sensitivity level (ie Secret or Unclassified)

	Transport Mode SA	Tunnel Mode SA
AH	**Authenticates** IP payload and selected portions of IP header and IPv6 extension headers	**Authenticates** entire inner IP packet plus selected portions of outer IP header
ESP	**Encrypts** IP payload and any IPv6 extesion header	**Encrypts** inner IP packet
ESP with authentication	**Encrypts** IP payload and any IPv6 extesion header. **Authenticates** IP payload but no IP header	**Encrypts** inner IP packet. **Authenticates** inner IP packet.

Authentication Header (AH)

- Provides support for data integrity and authentication of IP packets
  - malicious modifications are detected
  - address spoofing is prevented
  - replays are detected via sequence numbers

Authentication is based on use of a MAC

- parties must share a secret key in SA

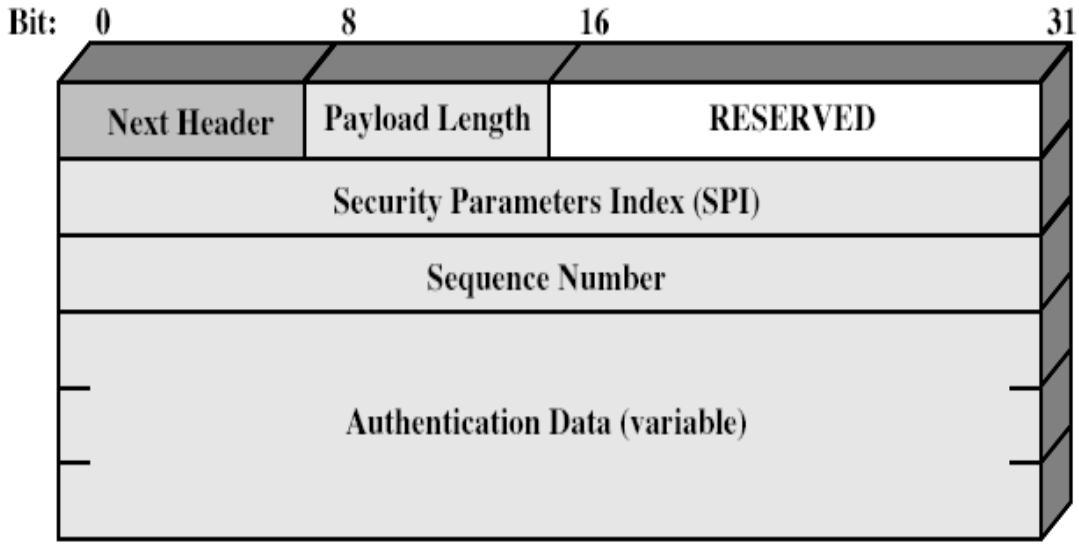
Authentication Header

Next Header: specifies next header or upper layer protocol

Payload length: to specify header length

SPI: to identify SA

Sequence number: used for replay control



Authentication data: MAC value (variable length)

AH – Anti-replay Service

Detection of duplicate packets

Sequence numbers are 32 bit values

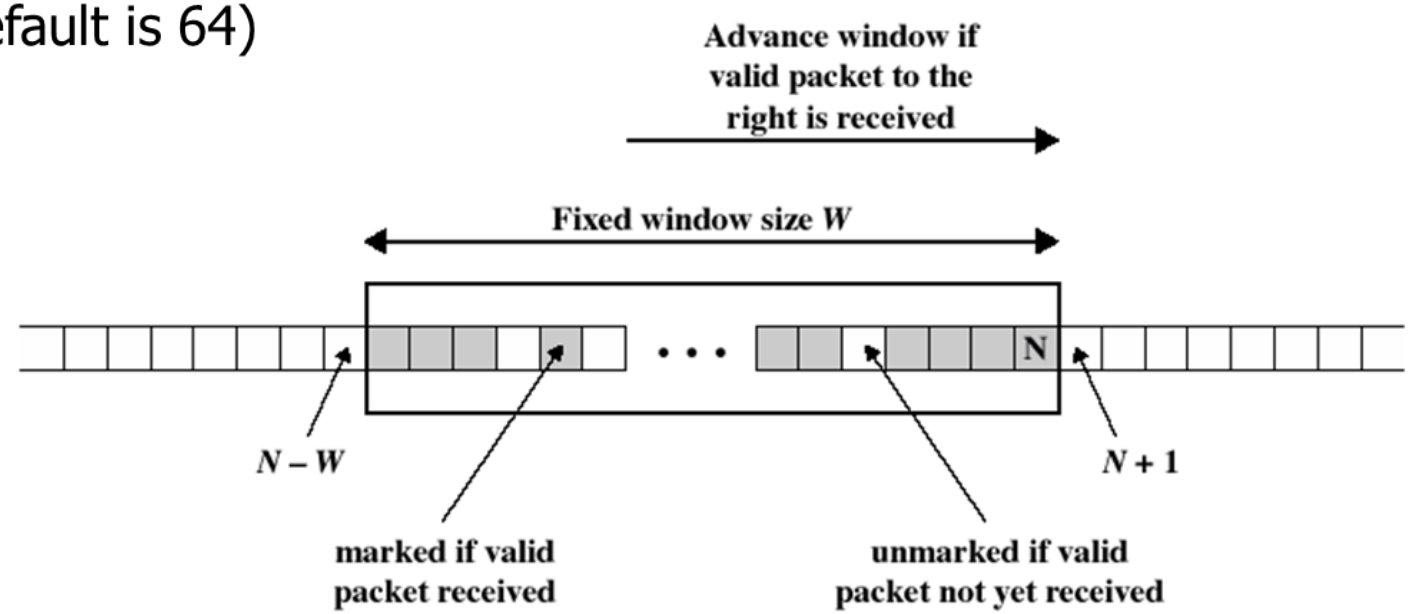
- associated with SAs
- when an SA is created, initialized to 0
  - when it reaches 2^{32}-1, SA must be terminated
  - not to allow overflows
- sender increments the replay counter and puts into each AH (sequence number field)

Problem: IP is unreliable, so the receiver may receive IP packets out of order

- Solution is window-based mechanism
  - Implemented at receiver side

AH – Anti-replay Service

window size W
(default is 64)



AH – Anti-replay Service

- If a received packet falls in the window
  - if authenticated and unmarked, mark it
  - if marked, then replay!
- If a received packet is > N
  - if authenticated, advance the window so that this packet is at the rightmost edge and mark it
- If a received packet is <= N-W
  - packet is discarded

AH - Integrity Check Value (ICV)

Actually it is a MAC

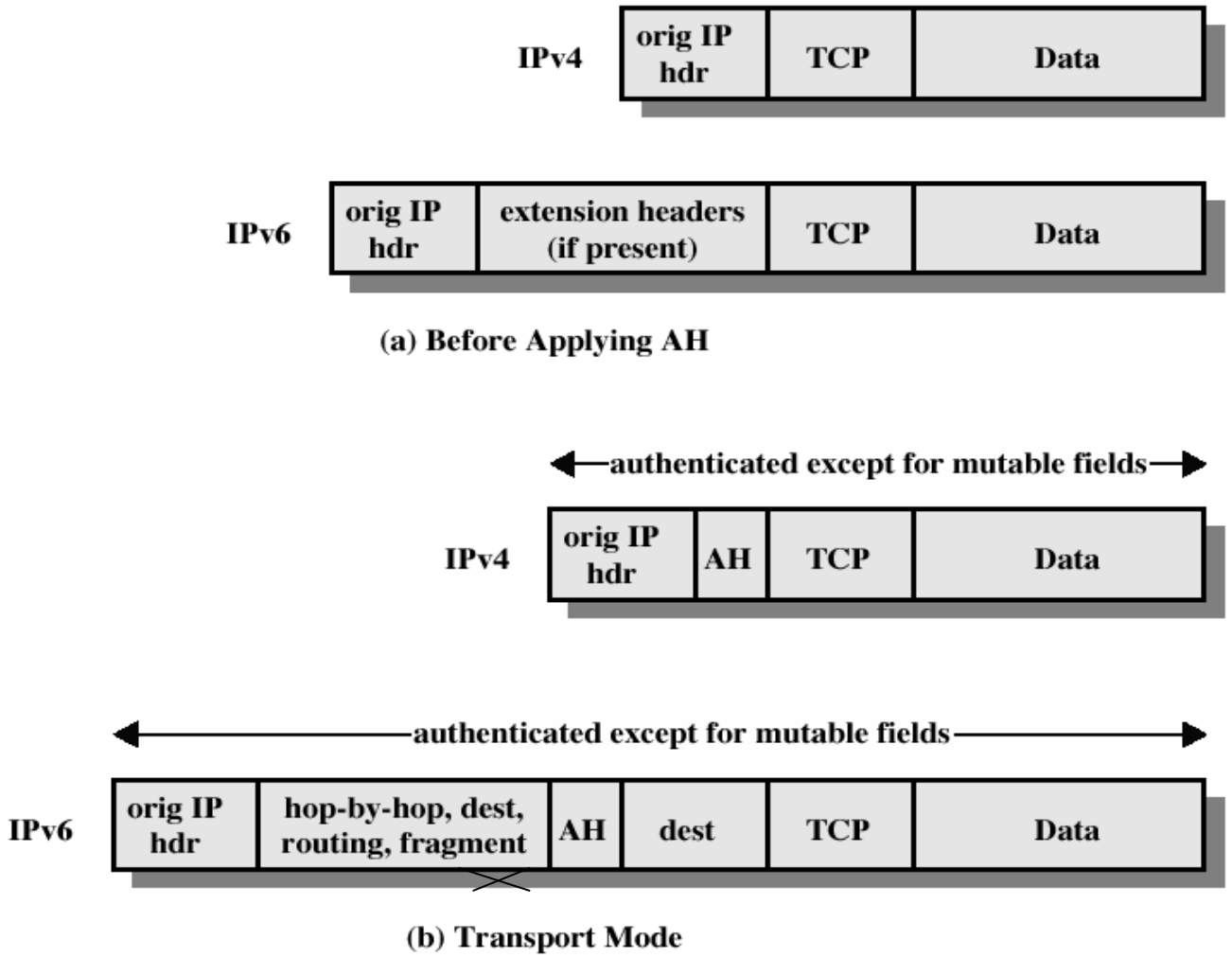
HMAC is used

- with a secure hash algorithm
- default length of authentication data field is 96 so HMAC output is truncated

MAC is calculated over

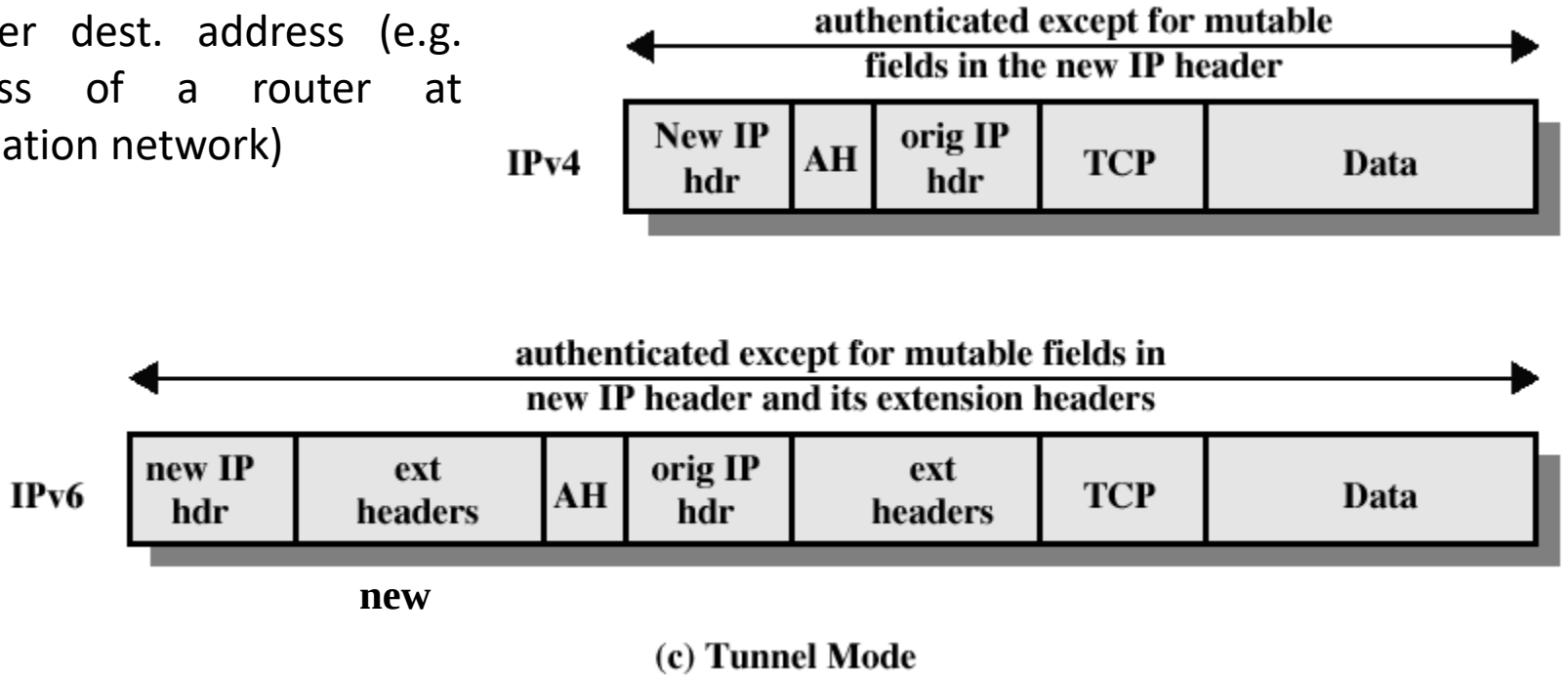
- IP payload (upper layer protocol data)
- IP Headers that are “immutable” or “mutable but predictable” at destination
  - e.g. source address (immutable), destination address (mutable but predictable)
  - Time to live field is mutable. Such mutable fields are zeroed for MAC calculation
- AH header (except authentication data of course, since authentication data is the MAC itself)

AH – Transport Mode



AH – Tunnel Mode

Inner IP packet carries the ultimate destination address
Outer IP packet may carry another dest. address (e.g. address of a router at destination network)



(c) Tunnel Mode

Encapsulating Security Payload (ESP)

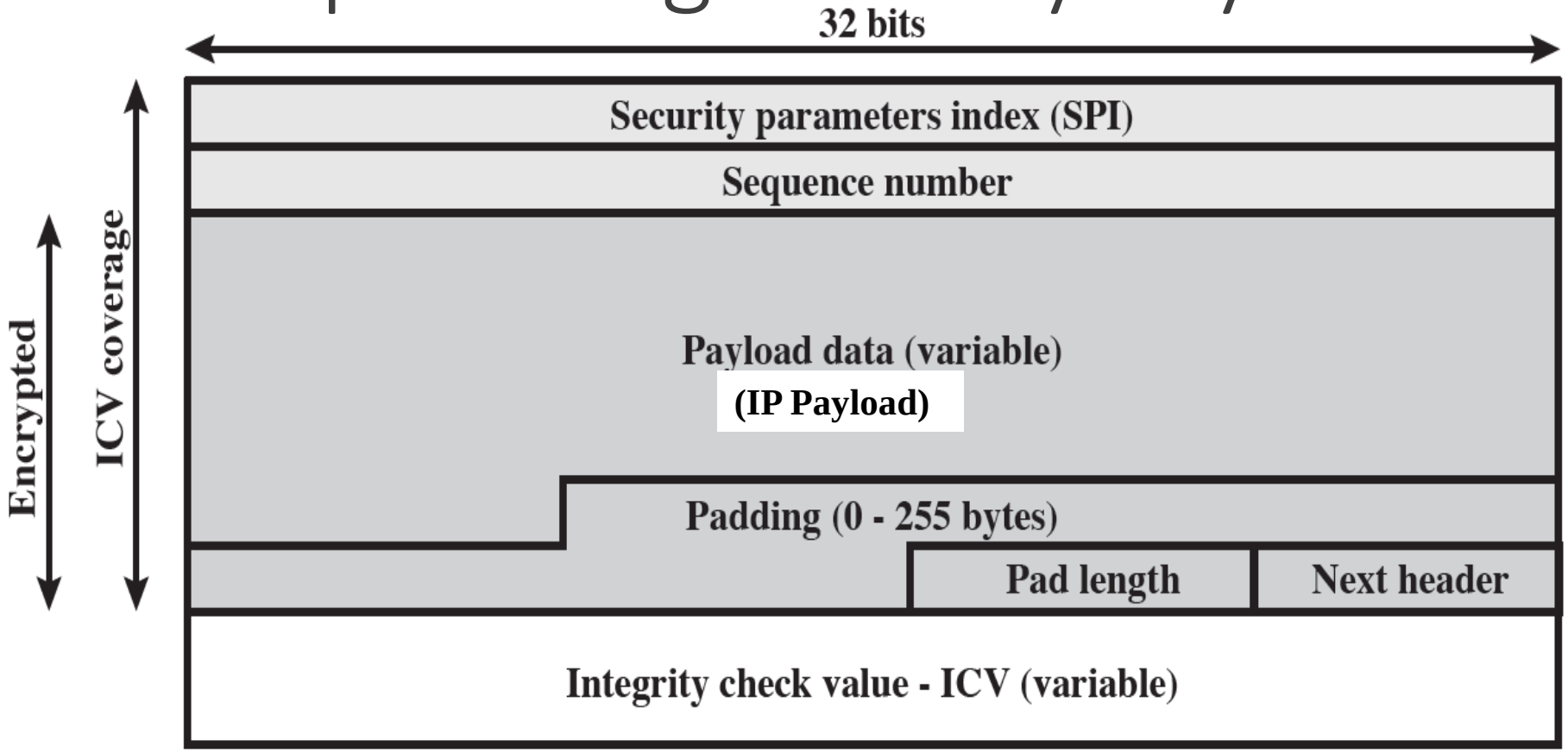
provides

- message content confidentiality via encryption
- limited traffic flow confidentiality and measures for traffic analysis
  - by padding (may arbitrarily increase the data)
  - by encrypting the source and destination addresses in tunnel mode
- optionally authentication services as in AH
  - via MAC (HMAC), sequence numbers

supports range of ciphers, modes

- DES, Triple-DES, RC5, IDEA, Blowfish, etc.
- CBC is the most common mode

Encapsulating Security Payload



Padding in ESP

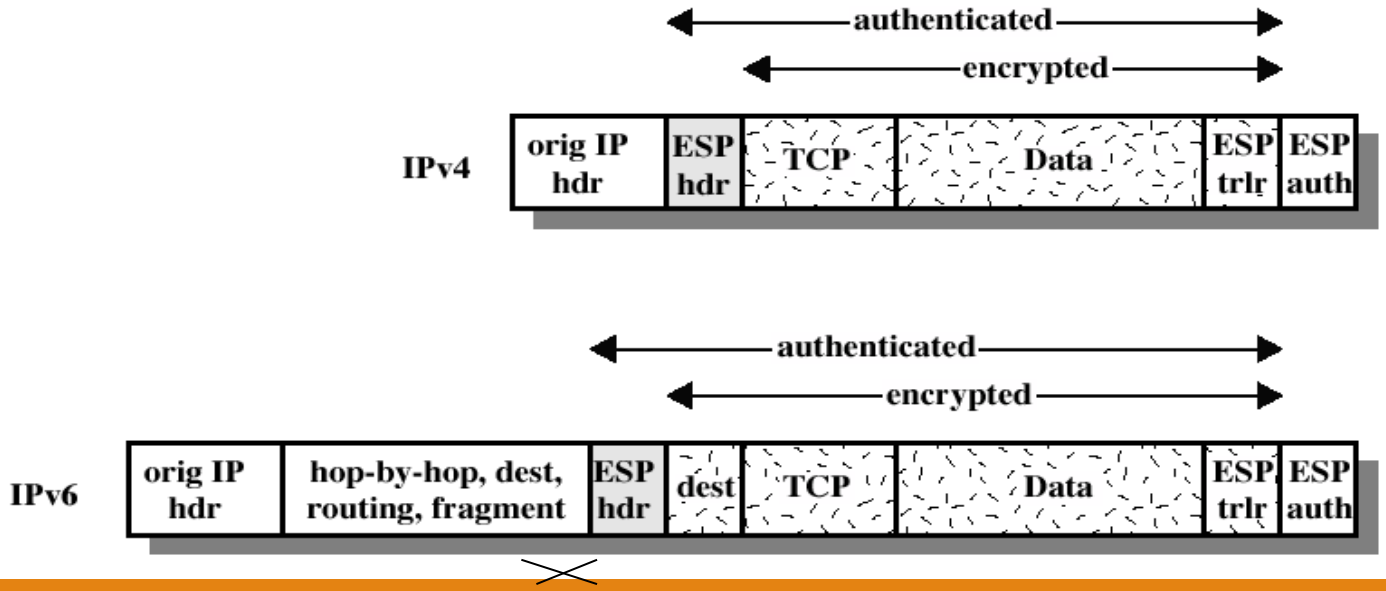
several purposes and reasons

- encryption algorithm may require the plaintext to be multiple of some integer *n*
- ESP format requires 32-bit words
- additional padding may help to provide partial traffic flow confidentiality by concealing the actual length of data
  - Other than the existing padding field, extra padding can be added to the end of the payload to improve traffic flow confidentiality

Transport Mode ESP

transport mode is used to encrypt & optionally authenticate IP payload (e.g. TCP segment)

- data protected but IP header left in clear
- so source and destination addresses are not encrypted
- Mostly for host to host (end-to-end) traffic

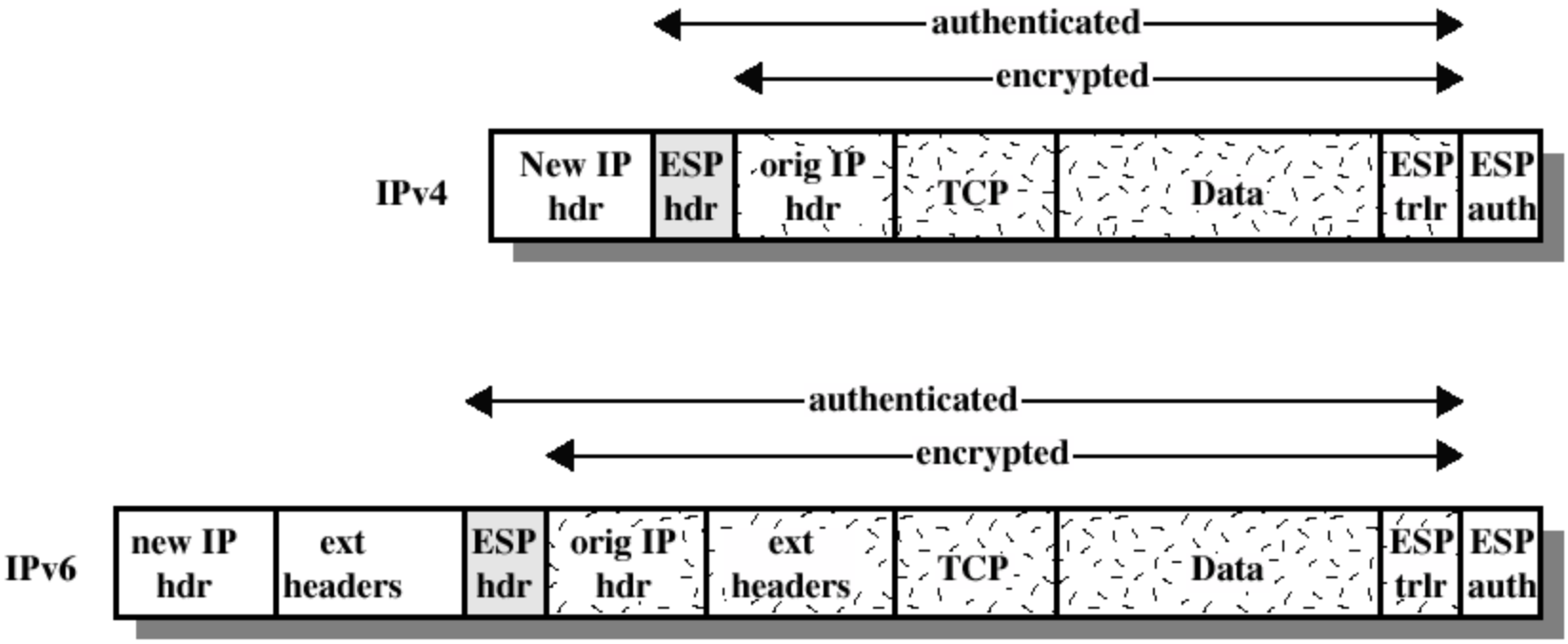


Tunnel Mode ESP

Encrypts and optionally authenticates the entire IP packet

- add new (outer) IP header for processing at intermediate routers
  - may not be the same as the inner (original) IP header, so traffic analysis can somehow be prevented
- good for VPNs, gateway to gateway (router to router) security
  - hosts in internal network do not get bothered with security related processing
  - number of keys reduced
  - thwarts traffic analysis based on ultimate destination

Tunnel Mode ESP



(b) Tunnel Mode

Key Management in IPSec

Ultimate aim

- generate and manage SAs for AH and ESP
- asymmetric
  - receiver and initiator have different SAs

can be manual or automated

- manual key management
  - sysadmin manually configures every system
- automated key management
  - on demand creation of keys for SA's in large systems

Key Management in IPSec

Complex system

- not a single protocol (theoretically)
- different protocols with different roles
  - intersection is IPSec
  - but may be used for other purposes as well

Several protocols are offered by IPSec WG of IETF

- Oakley, SKEME, SKIP, Photuris
- ISAKMP, IKE

IKE seems to be the IPSec key management protocol but it is actually a combination of Oakley, SKEME and uses ISAKMP structure

IKEv2 does not even use the terms Oakley and ISAKMP, but the basic functionality is the same