

## Sicurezza IP

## IP Security Overview

- 1994 – **RFC1636**, “*Security in the Internet Architecture*”
- Necessità di:
  - Un’infrastruttura di rete sicura da monitoraggio non autorizzato
  - Controllare traffic di rete
  - Comunicazioni end-to-end sicure, usando crittografia e autenticazione

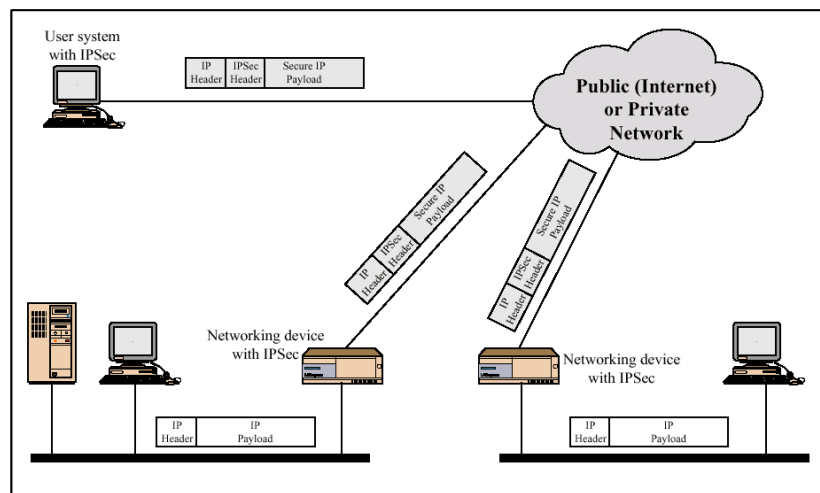
## IP Security Overview

- CERT – most serious attacks are IP spoofing and eavesdropping/packet sniffing
- Next generation IP includes authentication and encryption
- **IPv6**
- **IPSec**  $\subset$  IPv6
- Available with IPv4

## Applicazioni di IPSec

- Rendere sicura
  - la connessione Internet delle organizzazioni
  - La possibilità di accedere a servizi in remoto mediante Internet
- Stabilire connettività extranet e intranet con partner
- Favorire la sicurezza nell’e-commerce

## Applicazioni di IPSec



Sicurezza IP

5

## Benefici

- Rafforzare la sicurezza per il traffico che attraversa il perimetro, assumendo che sia implementato da un firewall o un router
- Trasparente
  - Alle applicazioni
  - Agli utenti

Sicurezza IP

6

## Routing & IPSec

- Requisiti:
  - Richieste di instradamento devono venire da router autorizzati
  - Richieste di redirectionamento devono venire dal router verso cui i pacchetti sono stati originariamente indirizzati
  - Prevenire distruzioni di pacchetti o invio di traffico verso direzioni non volute

Sicurezza IP

7

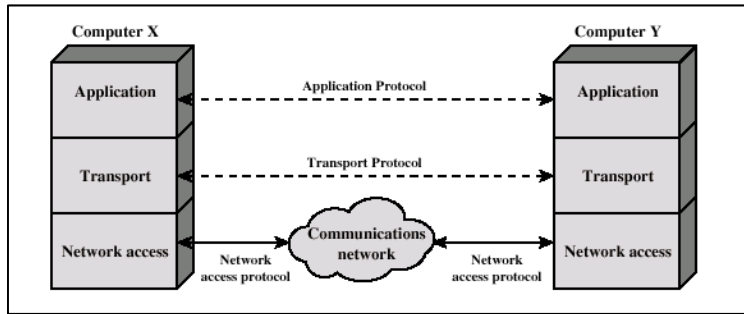
## Network Security

### Basic Networking – Part A

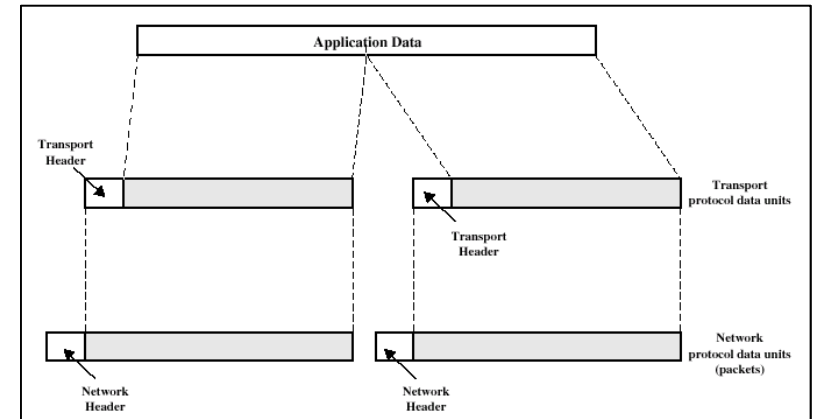
Sicurezza IP

8

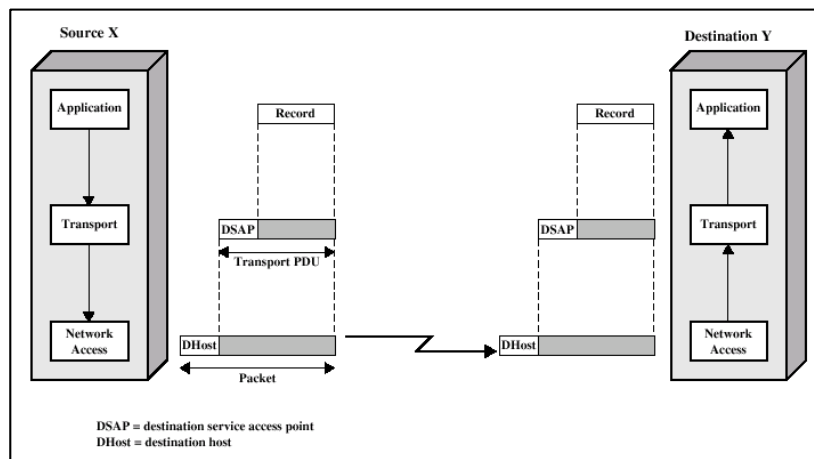
# Protocols in a Simplified Architecture



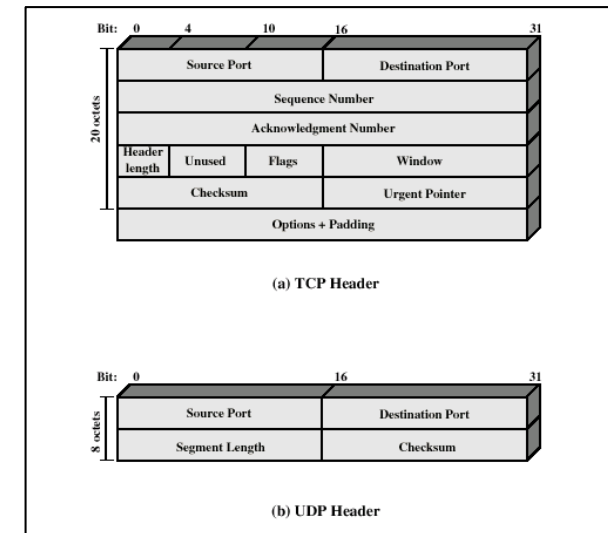
# Protocol Data Units



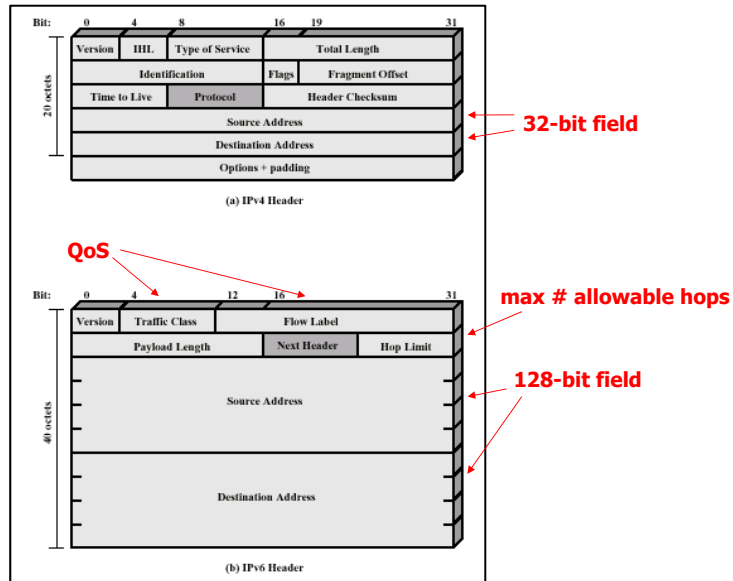
# Operation of a Protocol Architecture



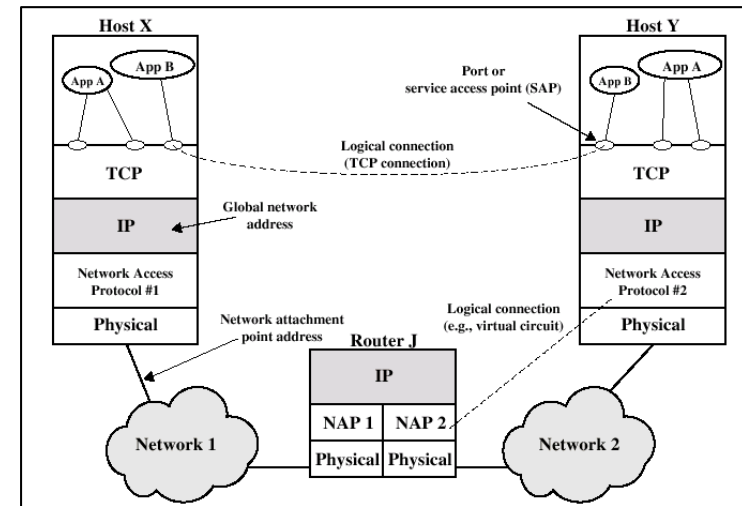
# TCP and UDP Headers



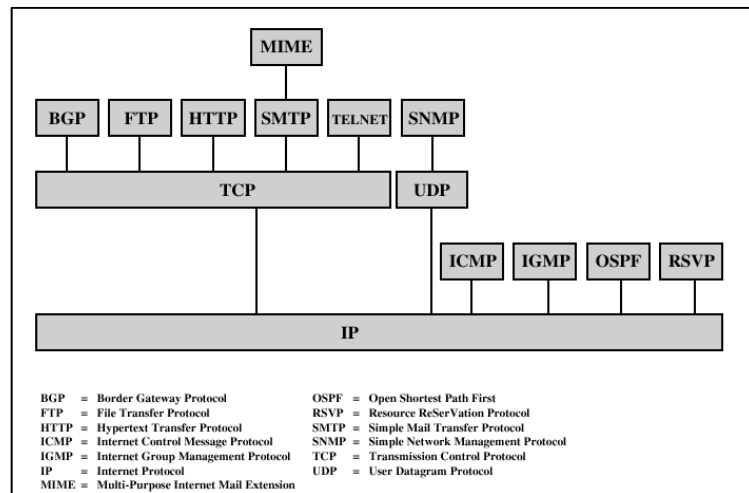
# IP Headers



# TP/IP Concepts

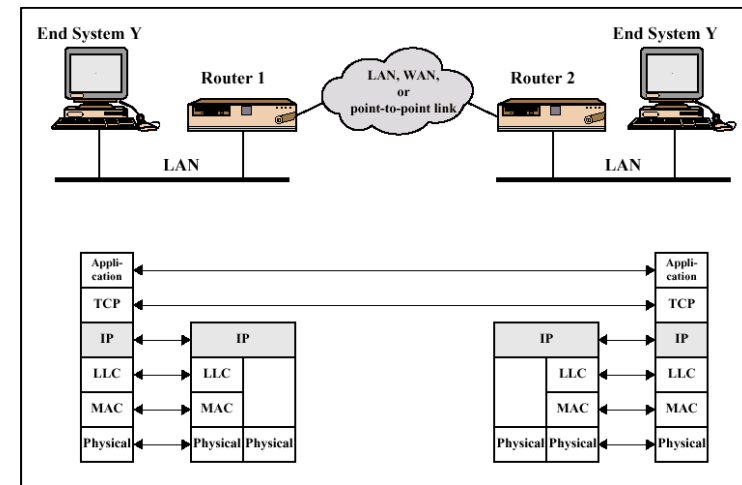


# Some TCP/IP Protocols

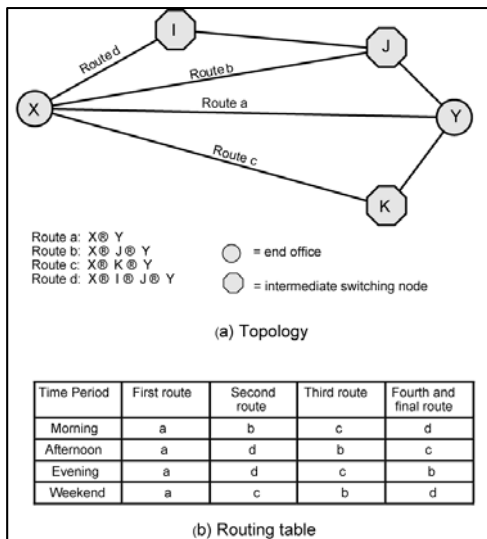


- BGP = Border Gateway Protocol
- FTP = File Transfer Protocol
- HTTP = Hypertext Transfer Protocol
- ICMP = Internet Control Message Protocol
- IGMP = Internet Group Management Protocol
- IP = Internet Protocol
- MIME = Multi-Purpose Internet Mail Extension
- OSPF = Open Shortest Path First
- RSVP = Resource ReSerVation Protocol
- SMTP = Simple Mail Transfer Protocol
- SNMP = Simple Network Management Protocol
- TCP = Transmission Control Protocol
- UDP = User Datagram Protocol

# Configuration of TCP/IP



# Alternate Routing Diagram



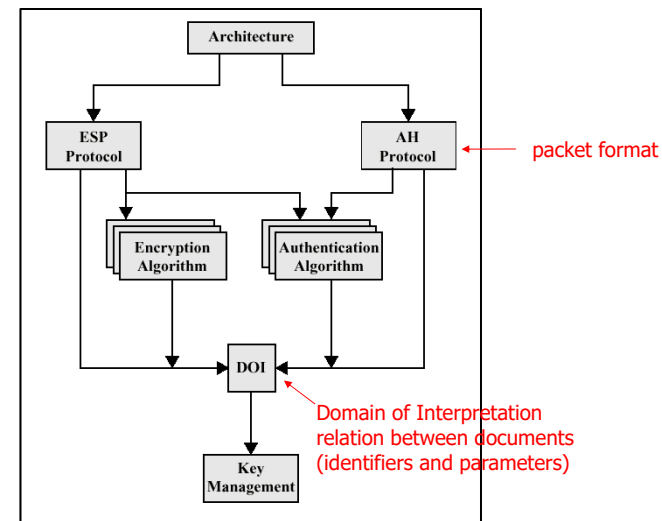
# Network Security

## IP Security – Part 1

# IPSec Documents

- November - 1998
  - RFC 2401 – Overview
  - RFC 2402 – Packet Authentication Extension
  - RFC 2406 – Packet Encryption Extension
  - RFC 2408 – Key Management Capabilities
- Implemented as extension headers that follow the main header:
  - Authentication Header (AH)
  - Encapsulating Security Payload Header (ESP)

# IPSec Documents



## IPSec: Servizi

- Fornisce servizi di sicurezza al livello di Internet protocol
- Permette al sistema di:
  - Selezionare i protocolli di sicurezza opportuni
  - Determinare gli algoritmi di sicurezza da usare
  - Definire le chiavi necessarie

## Servizi IPSec – 2 Protocolli

- **Authentication protocol** – designated by the **authentication header (AH)**
- **Encryption/Authentication protocol** – designated by the format of the packet, **Encapsulating Security Payload (ESP)**; it is a mechanism for providing *integrity* and *confidentiality* to IP datagrams
- **AH** and **ESP** are vehicles for **access control**

## 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		✓	✓

two cases

## Associations per la IPSec

*Concetto chiave:*

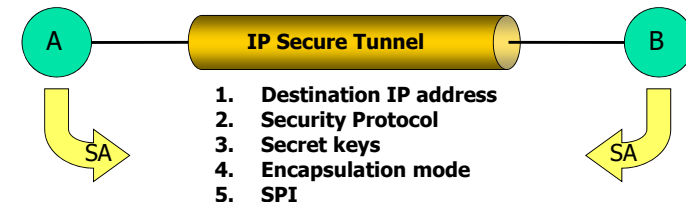
- **Security Association (SA)** – is a *one-way* relationship between a sender and a receiver that *defines the security services that are provided to a user*
- I requisiti sono specificati in 2 database: **security policy database (SPD)** e **security association database (SAD)**

## Security Associations

Uniquely identified by:

- **Destination IP address** – address of the destination endpoint of the SA (end user system or firewall/router)
- **Security protocol** – whether association is AH or ESP. Defines key size, lifetime and crypto algorithms (transforms)
- **Security parameter index (SPI)** – bit string that provides the receiving device with info on how to process the incoming traffic

## Security Associations



## Security Associations

- SA è **unidirezionale**
- Definisce le operazioni che occorrono solo in una direzione
- La bidirezionalità richiede una coppia di SA (e.g., secure tunnel)
- **Two SAs** use the same meta-characteristics but employ **different keys**

## Security Association Database

- Each IPsec implementation has a **Security Association Database (SAD)**
- SAD defines the **parameters association (SPI)** with each SA
- SAD **stores pairs of SA**, since SAs are unidirectional

## Security Association Database

- . Sequence number counter
- . Sequence counter overflow
- . Anti-replay window
- . AH information
- . ESP information
- . Lifetime of this SA
- . IPsec protocol mode – tunnel, transport, wildcard

Sicurezza IP Path MTU

29

## Security Policy Database

- Provides considerable flexibility in way IPsec services are applied to IP traffic
- Can discriminate between traffic that is afforded IPsec protection and traffic allowed to bypass IPsec
- The Security Policy Database (SPD) is the means by which IP traffic is related to specific SAs

Sicurezza IP

30

## Security Policy Database

- Each entry defines a subset of IP traffic and points to an SA for that traffic
- These selectors are used to filter outgoing traffic in order to map it into a particular SA

Sicurezza IP

31

## Security Policy Database

- . Destination IP address
- . Source IP address
- . User ID
- . Data sensitivity level – secret or unclassified
- . Transport layer protocol
- . IPsec protocol – AH or ESP or AH/ESP
- . Source and destination ports
- . IPv6 class
- . IPv6 flow label
- . IPv4 type of service (TOS)

Sicurezza IP

32



## Security Policy Database

**Outbound processing** of packet:

- 1) Compare fields in the packet to find a matching SPD entry
- 2) Determine the SA and its associated SPI
- 3) Do the required IPsec processing

## Transport and Tunnel Modes

- SA supports two modes:

**Transport** – protection for the upper layer protocols

**Tunnel** – protection for the entire IP packet

## Transport Mode

- Protection extends to the payload of an IP packet
- Primarily for upper layer protocols – TCP, UDP, ICMP
- Mostly used for end-to-end communication
- For AH or ESP the payload is the data following the IP header (IPv4) and IPv6 extensions
- Encrypts and/or authenticates the payload, but *not the IP header*

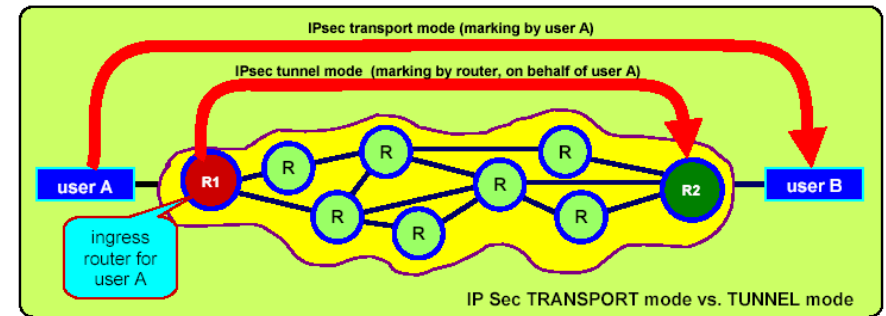
## Tunnel Mode

- Protection for the entire packet
- Add new *outer* IP packet with a new outer header
- AH or ESP fields are added to the IP packet and entire packet is treated as payload of the outer packet
- Packet travels through a *tunnel* from point to point in the network

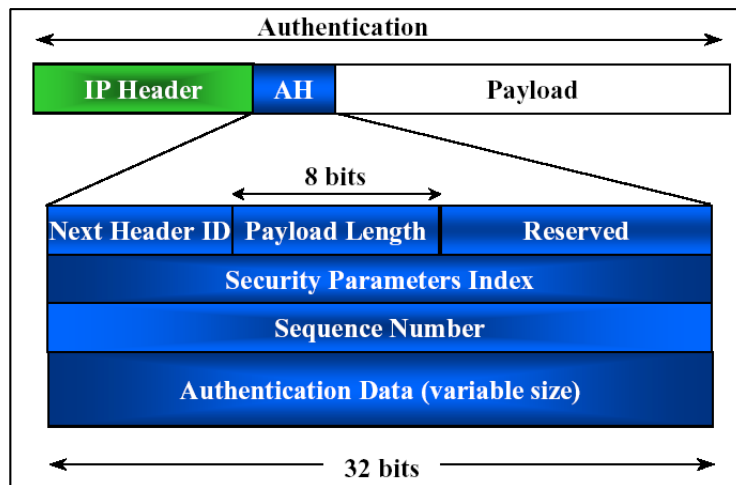
# Tunnel and Transport Mode

	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 (inner header plus IP payload) plus selected portions of outer IP header and outer IPv6 extension headers.
ESP	Encrypts IP payload and any IPv6 extension headers following the ESP header.	Encrypts inner IP packet.
ESP with Authentication	Encrypts IP payload and any IPv6 extension headers following the ESP header. Authenticates IP payload but not IP header.	Encrypts inner IP packet. Authenticates inner IP packet.

# Transport vs Tunnel Mode



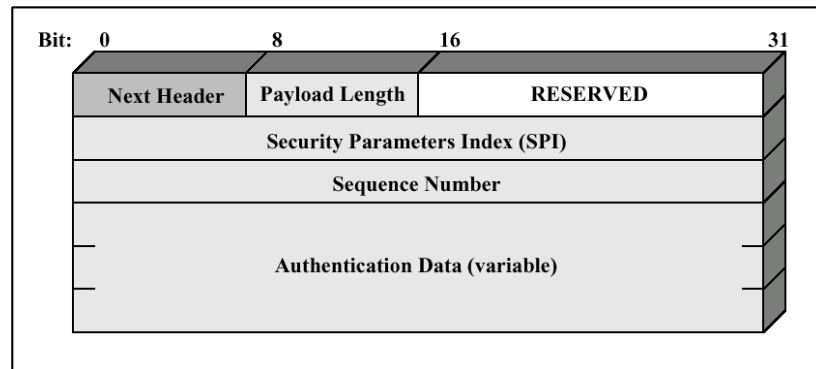
# Authentication Header



# Authentication Header

- Provides support for data integrity and authentication of IP packets
- Undetected modification in transit is impossible
- Authenticate the user or application and filters traffic accordingly
- Prevents address spoofing attacks
- Guards against replay attacks
- Based on the use of a message authentication code (MAC) so two parties must share a key

# IPSec Authentication Header



# Authentication Header

- **Next header** – type of header following
- **Payload length** – length of AH
- **Reserved** – future use
- **Security Parameters Index** – idents SA
- **Sequence Number** – 32bit counter
- **Authentication data** – variable field that contains the Integrity Check Value (ICV), or MAC

# Servizio Anti-Replay

- **Replay Attack**: Obtain a copy of authenticated packet and later transmit to the intended destination
- Mainly **disrupts** service
- **Sequence number** is designed to **prevent** this type of attack

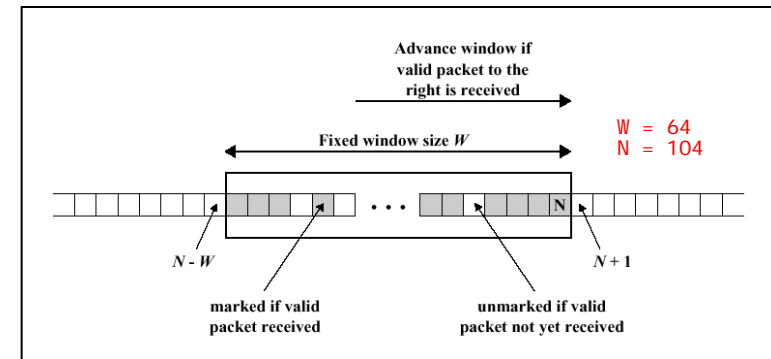
# Servizio Anti-Replay

- **Sender** inizializza un contatore (seq num) a 0 e lo incrementa per ogni pck inviato
- Seq num <  $2^{32}$ ; otherwise new SA
- IP is **connectionless**, unreliable service
- **Receiver implements window** of **W**
- **Right edge** of window is highest seq num, N, received so far

## Servizio Anti-Replay

- Per ogni nuovo pck ricevuto all'interno di una window si verifica il MAC, e se autenticato viene marcato
- I pck sul lato destroy della window svolgono il **check/mark & advance window** verso il nuovo seq num che è il nuovo **right edge**
- Packet to the **left**, or authentication fails, **discard packet**, & flag event

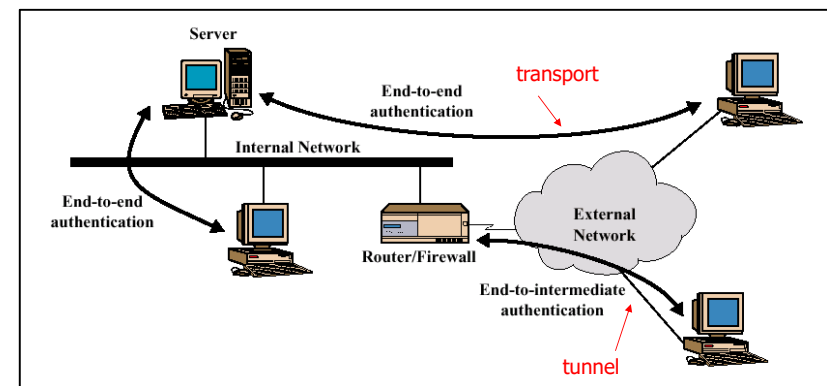
## Anti-Replay Mechanism



## Integrity Check Value

- Verificato grazie all'**Authentication Data field**
- **ICV** è un **Message Authentication Code (MAC)**
- Consiste di una versione troncata di un codice prodotto dall'algoritmo del MAC
- HMAC value è calcolato da solo i primi 96 bit
- MAC è calcolato a partire da un campo fisso, e.g., source address in IPv4

## End-to-end Authentication



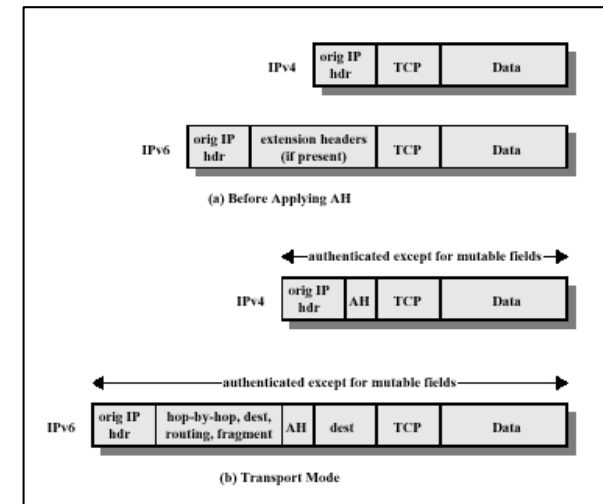
Two Ways To Use IPSec Authentication Service

## AH Tunnel and Transport Modes

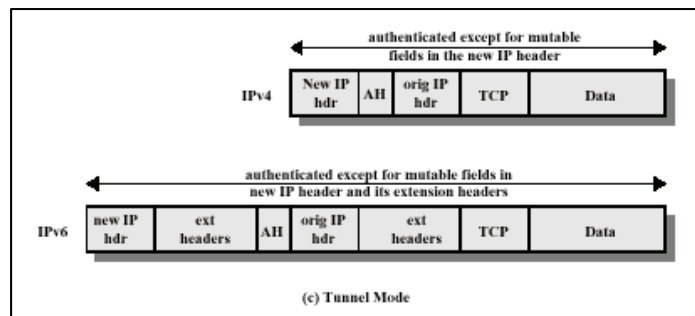
- Considerations are different for IPv4 and IPv6
- Authentication covers the entire packet
- **Mutable** fields are set to 0 for MAC calculation

What's a mutable field?

## Scope of AH Authentication



## Scope of AH Authentication



## Important URLs

- [www.rfc-editor.org](http://www.rfc-editor.org) - Search for RFC 1636, Security in the Internet Architecture, and other RFCs related to IPsec
- <http://en.wikipedia.org/wiki/IPV6> - Great info and links related to IPv6
- <http://www.ipv6tf.org/> - This portal has lots of news and info about IPv6

## Important URLs

- <http://www.ipv6.org/>  
Includes introductory material, news on recent IPv6 product developments, and related links.
- [www.redbooks.ibm.com/pubs/pdfs/redbooks/qg243376.pdf](http://www.redbooks.ibm.com/pubs/pdfs/redbooks/qg243376.pdf) Very good TCP/IP Tutorial from IBM Redbook Series with a good section (chap. 5) on security

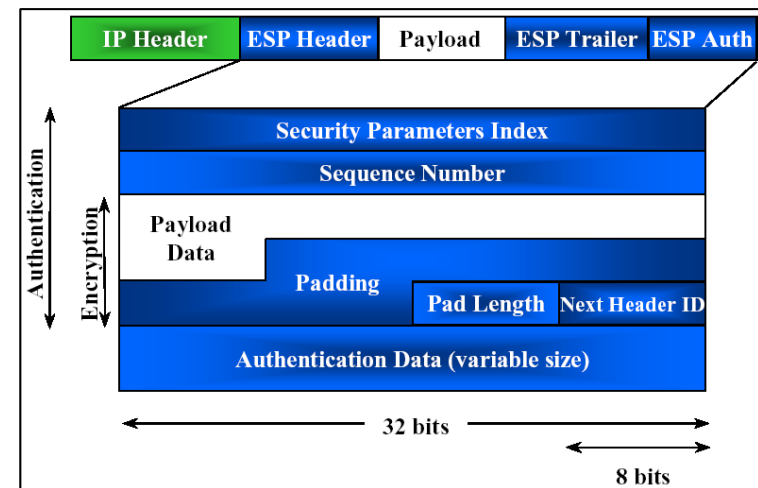
## Network Security

### IP Security – Part 2

## Encapsulating Security Payload

- Fornisce servizi di **confidentiality**
  - confidenzialità dei contenuti di un msg e limitata confidenzialità di del flusso del traffico
- ESP can also provide the same **authentication** services as AH

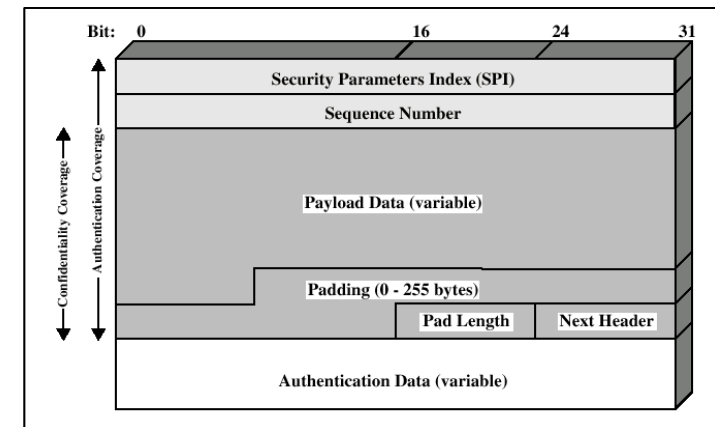
## Encapsulating Security Payload



## Encapsulating Security Payload

- **Security Parameters Index** – idents SA
- **Sequence Number** – 32bit counter
- **Payload Data** – variable field protected by encryption
- **Padding** – 0 to 255 bytes
- **Pad Length** – number of bytes in preceding
- **Next header** – type of header following
- **Authentication data** – variable field that contains the Integrity Check Value (ICV)

## IPSec ESP Format



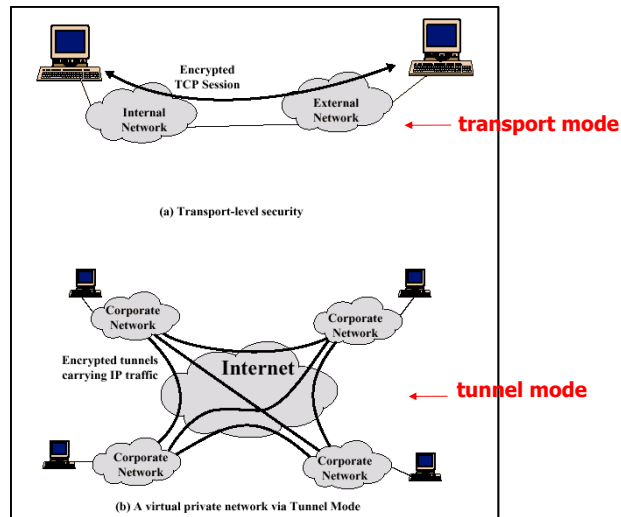
## ESP and AH Algorithms

- Le implementazioni devono supportare DES in cipher block chaining (CBC) mode
- Others:  
**3DES, PC5, IDA, 3IDEA, CAST, Blowfish**
- ESP support use of a **96bit MAC** similar to AH

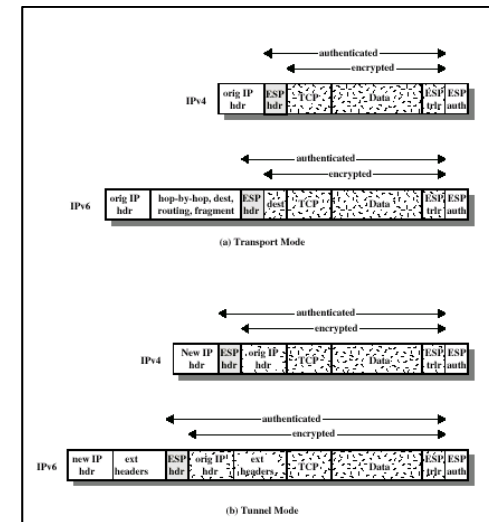
## ESP Padding

- Algorithm may require plaintext to be a **multiple** of some number of bytes
- Pad Length and Next Header must be **right aligned**
- Additional padding may be used to **conceal** actual length of the payload

## Transport vs Tunnel Mode



## Scope of ESP Encryption



## Combining SAs

- SA can implement *either* AH or ESP protocol, *but not both*
- Traffic flow may require separate IPsec services between hosts
- **Security Association Bundle** refers to a sequence of SAs
- SAs in a bundle may terminate at different end points

## Combining SAs

SAs many combine into bundles in two ways:

- **Transport adjacency** – applying more than one security protocol to the same IP packet without invoking tunneling; only one level of combination, no nesting
- **Iterated tunneling** – application of multiple layers of security protocols effected through IP tunneling; multiple layers of nesting

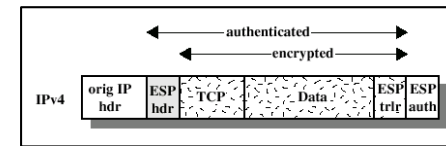


# Authentication + Encryption

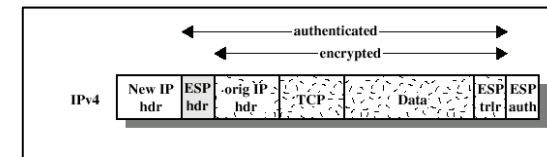
- Several approaches to combining authentication and confidentiality
- **ESP with Authentication Option**
  - First **apply ESP** then **append the authentication** data field
  - Authentication applies to ciphertext rather than plaintext

# Authentication + Encryption

- **ESP with Authentication Option**



Transport Mode



Tunnel Mode

# Authentication + Encryption

- **Transport Adjacency**
  - Use two bundled transport SAs
  - **Inner** being an ESP SA; **outer** being an AH SA
  - Authentication covers the ESP plus the original IP header
  - **Advantage:** authentication covers more fields, including source and destination IP addresses

# Authentication + Encryption

- **Transport-Tunnel Bundle**
  - First apply **authentication**, then encryption
  - Authenticated data is protected and easier to store and retrieve
  - Use a bundle consisting of an **inner AH** transport SA and an **outer ESP** tunnel SA
  - **Advantage:** entire authenticated inner packet is encrypted and a new outer IP header is added

## Basic Combinations

- IPsec architecture describe 4 esempi che devono essere supportati nelle implementazioni
- Sono rappresentate le connessioni logiche e fisiche
- Ogni SA può essere sia AH che ESP
- Host-to-host SAs are either transport or tunnel, otherwise it must be tunnel mode<sup>69</sup>

Sicurezza IP

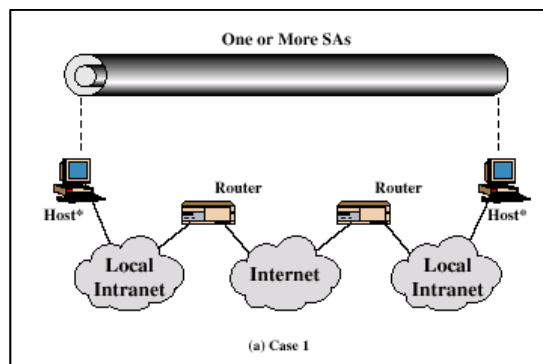
## Basic Combinations – Case 1

- All security is provided between end systems that implement IPsec
- Possible combinations
  - a. AH in transport mode
  - b. ESP in transport mode
  - c. AH followed by ESP in transport mode (an AH SA inside an ESP SA)
  - d. Any one of a, b, or c inside and AH or ESP in tunnel mode

Sicurezza IP

70

## Basic Combinations – Case 1



\* = implements IPsec

Sicurezza IP

71

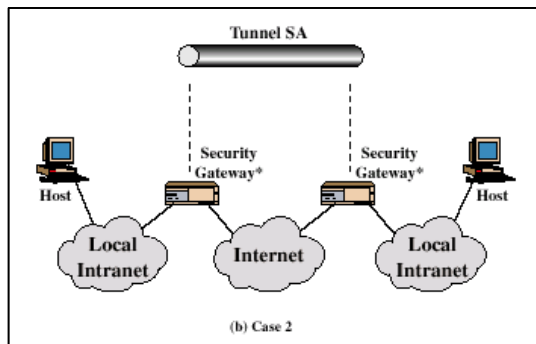
## Basic Combinations – Case 2

- Security is provided only between gateways and no hosts implement IPsec
- VPN – Virtual Private Network
- Only single tunnel needed (support AH, ESP or ESP w/auth)

Sicurezza IP

72

## Basic Combinations – Case 2

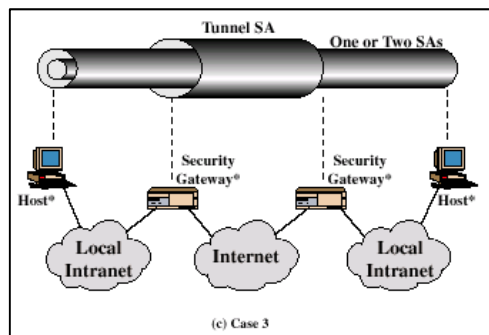


\* = Implements IPSec

## Basic Combinations – Case 3

- Builds on Case 2 by adding end-to-end security
- Gateway-to-gateway tunnel is ESP
- Individual hosts can implement additional IPSec services via end-to-end SAs

## Basic Combinations – Case 3

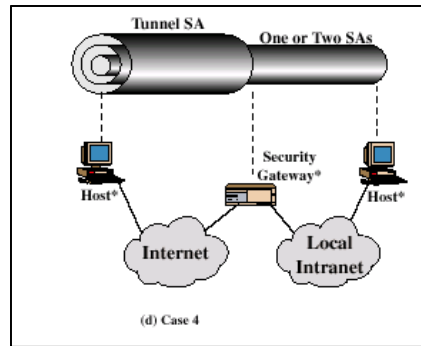


\* = Implements IPSec

## Basic Combinations – Case 4

- Provides support for a remote host using the Internet and reaching behind a firewall
- Only tunnel mode is required between the remote host and the firewall
- One or two SAs may be used between the remote host and the local host

## Basic Combinations – Case 4



\* = implements IPsec

## Key Management

- Determination and distribution of **secret keys**
- Four keys for communication between two applications:  
*transmit and receive pairs for both AH & ESP*
- Two modes: manual and automated
- Two protocols:
  - **Oakley Key Determination Protocol**
  - **Internet Security Association and Key Management Protocol (ISAKMP)**

## Oakley Key Determination Protocol

- **Refinement** of the **Diffie-Hellman** key exchange algorithm
- Two users A and B agree on two global parameters:  $q$ , a large prime number and  $\alpha$ , a primitive root of  $q$
- Secret keys created only when needed
- Exchange requires no preexisting infrastructure
- **Disadvantage**: Subject to **MITM** attack

## Features of Oakley

- Employs **cookies** to thwart clogging attacks
- Two parties can negotiate a group (modular exponentiation or elliptic curves)
- Uses **nonces** to ensure against replay attacks
- Enables the exchange of Diffie-Hellman public key values
- Authenticates the Diffie-Hellman exchange to thwart MITM attacks

# Aggressive Oakley Key Exchange

```

I → R: CKYI, OK_KEYX, GRP, gx, EHAO, NIDP, IDI, IDR, NI, SkI[IDI || IDR || NI || GRP || gx || EHAO]
R → I: CKYR, CKYI, OK_KEYX, GRP, gy, EHAS, NIDP, IDR, IDI, NR, NI, SkR[IDR || IDI || NR || NI || GRP || gy || gx || EHAS]
I → R: CKYI, CKYR, OK_KEYX, GRP, gx, EHAS, NIDP, IDI, IDR, NI, NR, SkI[IDI || IDR || NI || NR || GRP || gx || gy || EHAS]
    
```

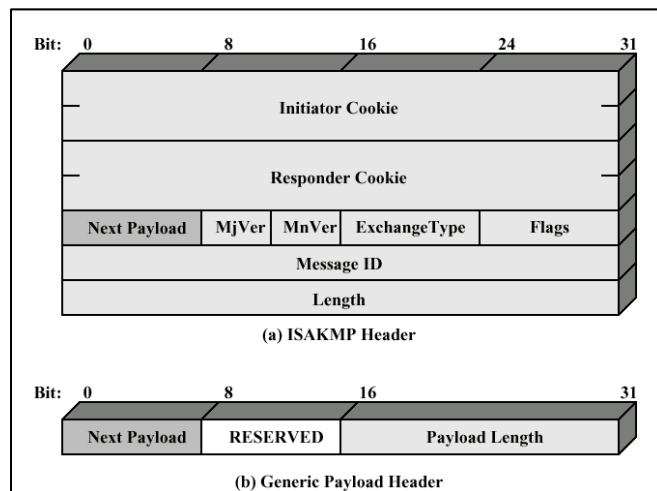
Notation:

- I = Initiator
- R = Responder
- CKY<sub>I</sub>, CKY<sub>R</sub> = Initiator, responder cookies
- OK\_KEYX = Key exchange message type
- GRP = Name of Diffie-Hellman group for this exchange
- g<sup>x</sup>, g<sup>y</sup> = Public key of initiator, responder; g<sup>xy</sup> = session key from this exchange
- EHAO, EHAS = Encryption, hash, authentication functions, offered and selected
- NIDP = Indicates encryption is not used for remainder of this message
- ID<sub>I</sub>, ID<sub>R</sub> = Identifier for initiator, responder
- N<sub>I</sub>, N<sub>R</sub> = Random nonce supplied by initiator, responder for this exchange
- Sk<sub>I</sub>[X], Sk<sub>R</sub>[X] = Indicates the signature over X using the private key (signing key) of initiator, responder

# ISAKMP

- Defines **procedures** and packet formats to establish, negotiate, modify and delete **SAs**
- Defines **payloads** for exchanging key generation and authentication data
- Now called **IKE**

# ISAKMP Formats



# ISAKMP Payload Types

Type	Parameters	Description
Security Association (SA)	Domain of Interpretation, Situation	Used to negotiate security attributes and indicate the DOI and Situation under which negotiation is taking place.
Proposal (P)	Proposal #, Protocol-ID, SPI Size, # of Transforms, SPI	Used during SA negotiation; indicates protocol to be used and number of transforms.
Transform (T)	Transform #, Transform-ID, SA Attributes	Used during SA negotiation; indicates transform and related SA attributes.
Key Exchange (KE)	Key Exchange Data	Supports a variety of key exchange techniques.
Identification (ID)	ID Type, ID Data	Used to exchange identification information.
Certificate (CERT)	Cert Encoding, Certificate Data	Used to transport certificates and other certificate-related information.
Certificate Request (CR)	# Cert Types, Certificate Types, # Cert Auths, Certificate Authorities	Used to request certificates; indicates the types of certificates requested and the acceptable certificate authorities.
Hash (HASH)	Hash Data	Contains data generated by a hash function.
Signature (SIG)	Signature Data	Contains data generated by a digital signature function.
Nonce (NONCE)	Nonce Data	Contains a nonce.
Notification (N)	DOI, Protocol-ID, SPI Size, Notify Message Type, SPI, Notification Data	Used to transmit notification data, such as an error condition.
Delete (D)	DOI, Protocol-ID, SPI Size, # of SPIs, SPI (one or more)	Indicates an SA that is no longer valid.

## ISAKMP Exchanges

- Provides a **framework** for message exchange
- **Payload type** serves as the building blocks
- Five default **exchange types** specified
- SA refers to an SA payload with associated Protocol and Transform payloads

## ISAKMP Exchange Types

Exchange	Note
<b>(a) Base Exchange</b>	
(1) <b>I → R</b> : SA; NONCE	Begin ISAKMP-SA negotiation
(2) <b>R → I</b> : SA; NONCE	Basic SA agreed upon
(3) <b>I → R</b> : KE; ID <sub>i</sub> ; AUTH	Key generated; Initiator identity verified by responder
(4) <b>R → I</b> : KE; ID <sub>R</sub> ; AUTH	Responder identity verified by initiator; Key generated; SA established
<b>(b) Identity Protection Exchange</b>	
(1) <b>I → R</b> : SA	Begin ISAKMP-SA negotiation
(2) <b>R → I</b> : SA	Basic SA agreed upon
(3) <b>I → R</b> : KE; NONCE	Key generated
(4) <b>R → I</b> : KE; NONCE	Key generated
(5) * <b>I → R</b> : ID <sub>i</sub> ; AUTH	Initiator identity verified by responder
(6) * <b>R → I</b> : ID <sub>R</sub> ; AUTH	Responder identity verified by initiator; SA established
<b>(c) Authentication Only Exchange</b>	
(1) <b>I → R</b> : SA; NONCE	Begin ISAKMP-SA negotiation
(2) <b>R → I</b> : SA; NONCE; ID <sub>R</sub> ; AUTH	Basic SA agreed upon; Responder identity verified by initiator
(3) <b>I → R</b> : ID <sub>i</sub> ; AUTH	Initiator identity verified by responder; SA established
<b>(d) Aggressive Exchange</b>	
(1) <b>I → R</b> : SA; KE; NONCE; ID <sub>i</sub>	Begin ISAKMP-SA negotiation and key exchange
(2) <b>R → I</b> : SA; KE; NONCE; ID <sub>R</sub> ; AUTH	Initiator identity verified by responder; Key generated; Basic SA agreed upon
(3) * <b>I → R</b> : AUTH	Responder identity verified by initiator; SA established
<b>(e) Informational Exchange</b>	
(1) * <b>I → R</b> : N/D	Error or status notification, or deletion

Notation:  
 I = initiator  
 R = responder  
 \* = signifies payload encryption after the ISAKMP header

## Internet Key Exchange

- **IKE** is now at Ver 2 – defined in RFC4306, 12/05
- It works within ISAKMP framework
- Uses **Oakley** and **Skeme** protocols for authenticating keys and rapid key refreshment

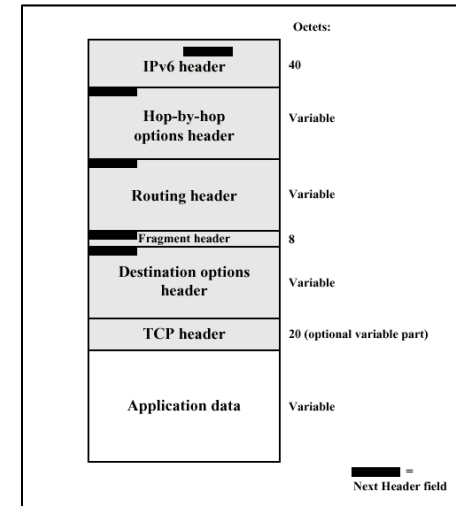
## Network Security

### Basic Networking – Part B

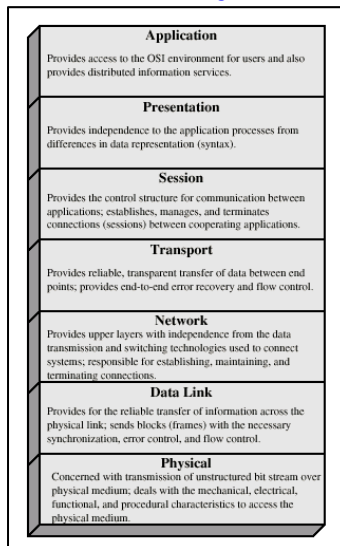
# IPv6

- 1995 – RFC 1752 IPng
- 1998 – RFC 2460 IPv6
- Functional enhancements for a mix of data streams (graphic and video)
- Driving force was address depletion  
128-bit addresses
- Started in Solaris 2.8, Windows 2000

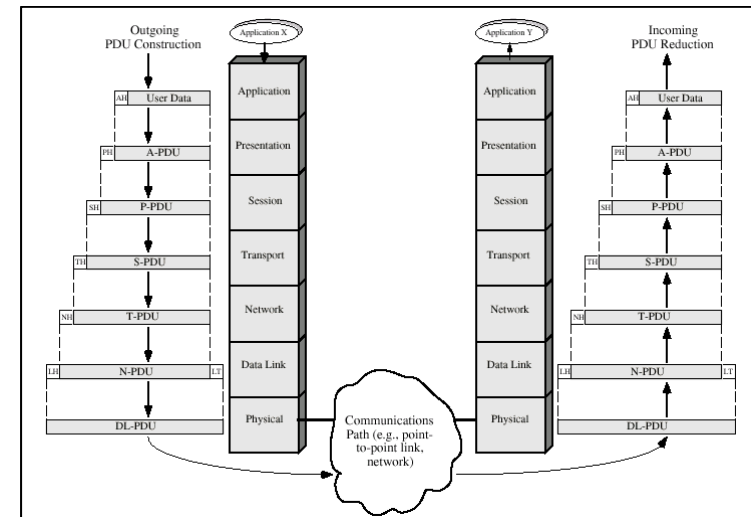
# IPv6 Packet w/Extension Headers



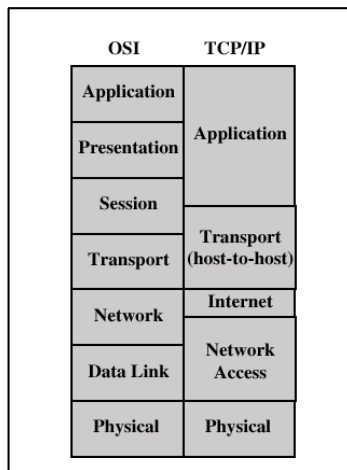
# OSI Layers



# OSI Environment



# OSI-TCP/IP Comparison



## Network Security

## IP Security – Part 2

# Ethereal

- Ethereal is a free network protocol analyzer for Unix and Windows
- Packet Sniffer - data can be captured "off the wire" from a live network connection
- [www.ethereal.com](http://www.ethereal.com) - Everything you ever wanted to know about ethereal
- [wiki.ethereal.com](http://wiki.ethereal.com) - This is the "User's Manual;" also has a nice "References" section

The screenshot shows the Ethereal interface with a packet capture for business.nytimes.com. The packet list shows an HTTP POST request. The packet details for frame 55 (829 bytes) show an arrival time of Mar 14, 2001 01:38:22.1334. The hex data shows a cookie being captured: "Cookie: RMID=18bdc883a0d1f6; NYT-5=1012032; fhezvwo/dpms4t5; .trivn91 CDGyWwnY5p0kk92 UB/Iw9Rvww.LoF5g xosq/zv4tzUN9Rk7 c2B0Nkpx90; RDB=c802002D2D00005 5530105649528310 10100000000002; weathercity=LEA; ASPSES=SIOWIDQG GGRYXcNHNKNGDA FIJPFLLHCEDKADCO. ...mode=news&act ion=quot e&ticker=sunw". The ASCII data shows the beginning of the cookie: "he..Cook ie: RMID=18bdc883a0d1f6; NYT-5=1012032; fhezvwo/dpms4t5; .trivn91 CDGyWwnY5p0kk92 UB/Iw9Rvww.LoF5g xosq/zv4tzUN9Rk7 c2B0Nkpx90; RDB=c802002D2D00005 5530105649528310 10100000000002; weathercity=LEA; ASPSES=SIOWIDQG GGRYXcNHNKNGDA FIJPFLLHCEDKADCO. ...mode=news&act ion=quot e&ticker=sunw".



## Ethereal Etiquette

- Be careful when and where you use this tool
- It makes people nervous
- Use prudence with the information you collect
- **When in doubt, seek permission!**

## Other Sniffing Tools

- [Ettercap](#) is an open source software tool for computer network protocol analysis and security cracking. It can be used to intercept traffic on a network segment, capture passwords, and conduct man-in-the-middle attacks against a number of common protocols.
- [dSniff](#) is a packet sniffer and set of traffic analysis tools. Unlike tcpdump and other low-level packet sniffers, dSniff also includes tools that decode information (passwords, most infamously) sent across the network, rather than simply capturing and printing the raw data, as do generic sniffers like Ethereal and tcpdump.
- [AiroPeek](#) was the first Wi-Fi (IEEE 802.11) packet analyzer, or packet sniffer, that provides network engineers with a view of the data traversing a Wireless LAN network. AiroPeek was created in 2001 and its interface was based closely on [EtherPeek](#), another product from [WildPackets](#), Inc. They also have some “free” utilities.

## Important URLs

- [www.insecure.org/tools.html](http://www.insecure.org/tools.html)  
Site has the top 50 security tools
- [Nmap](#) is a free software port scanner. It is used to evaluate the security of computers, and to discover services or servers on a computer network.
- [EtherApe](#) is a graphical network monitor for Unix. Featuring link layer, ip and TCP modes, it displays network activity graphically. Hosts and links change in size with traffic. Color coded protocols display.
- **Be judicious in the use of these tools!**