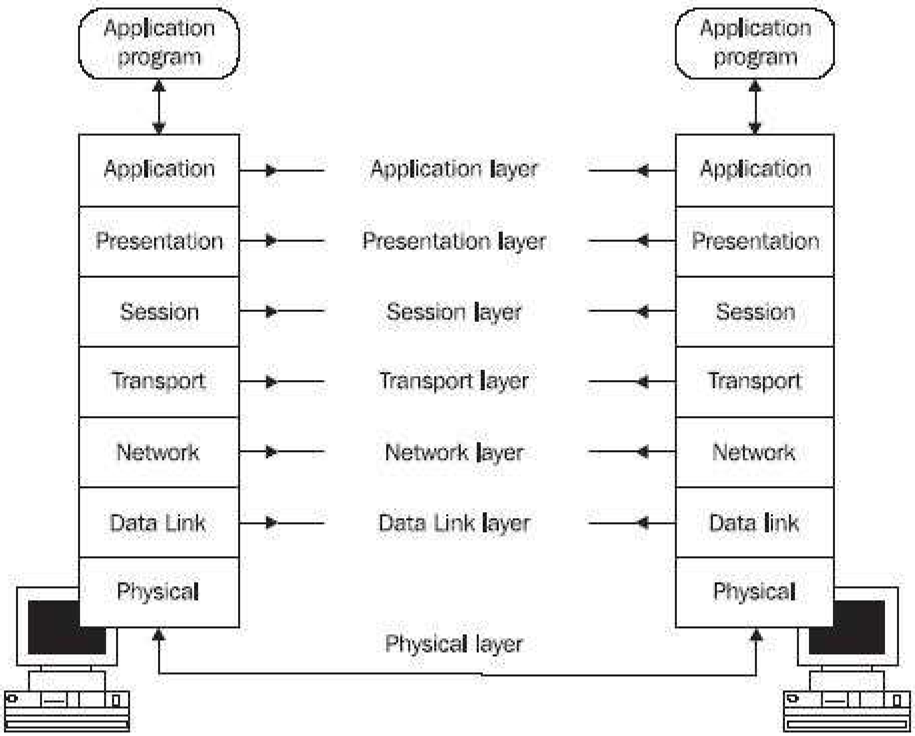
Information Security

Van K Nguyen - HUT

Network Security

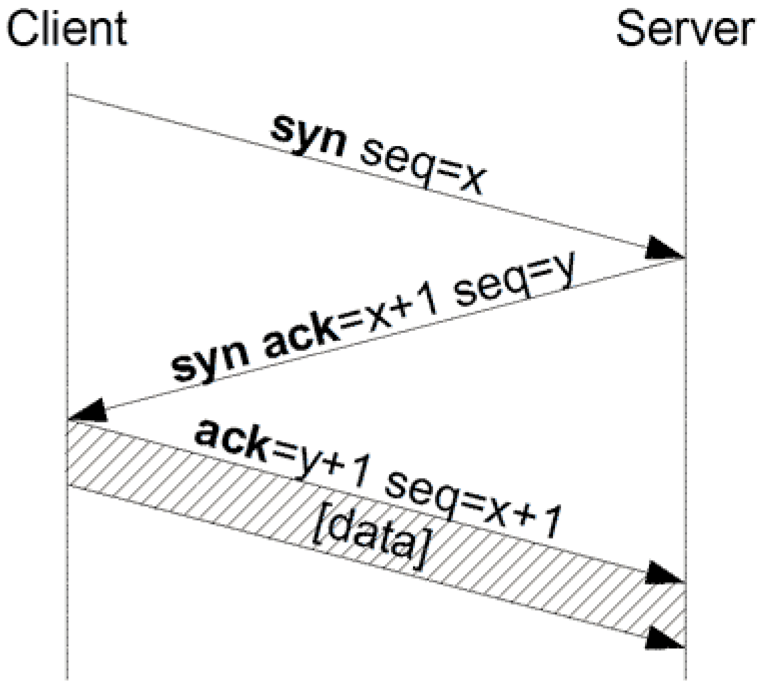

Application program
Application program
Application
Application layer
Application
Presentation
Presentation layer
Presentation
Session
Session layer
Session
Transport
Transport layer
Transport
Network
Network layer
Network
Data Link
Data Link layer
Data link
Physical
Physical
Physical layer

Attacks against TCP

Transmission Control Protocol - TCP

- Connection oriented protocol for a user process
  - **Reliable**, full-duplex channel: acknowledgements, retransmissions, timeouts
  - The packets are delivered in the same order
- Congestion control mechanisms

TCP 3-way handshake



- The sequence number x and y are random values that the other side need to **ack** by increment (x+1 or y+1)
- The connection only fully opened when server-side received client’s ack

SYN Attack

- An attacker sends flood of SYNs with source address spoofed packets to a target.
- If the limit is reached, target machine will refuse any incoming connections till the timeout expires
  - The server send the SYN-ACK to the falsified IP address, and thus never receive the ACK
    - Server wait for ACK for some time, as simple network congestion could also be the cause of the missing ACK.
- Spoofed address chosen to be a non-existent one
  - If the spoofed address belongs to a machine, then what ?

Why it works?

- There is no authentication of the source of the packets
- Addresses can be easily spoofed
- Server needs to allocate a lot or resources while client doesn't

Some measurements to the SYN attack

- Configuration Optimization
  - At the server
    - Reduce the timeout to 10 seconds
    - Increase the size of the queue
    - Disable non-essential services, reducing the number of ports to be attacked
  - At all routers in the Internet
    - Block packets to the outside that have source addresses from outside the internal network

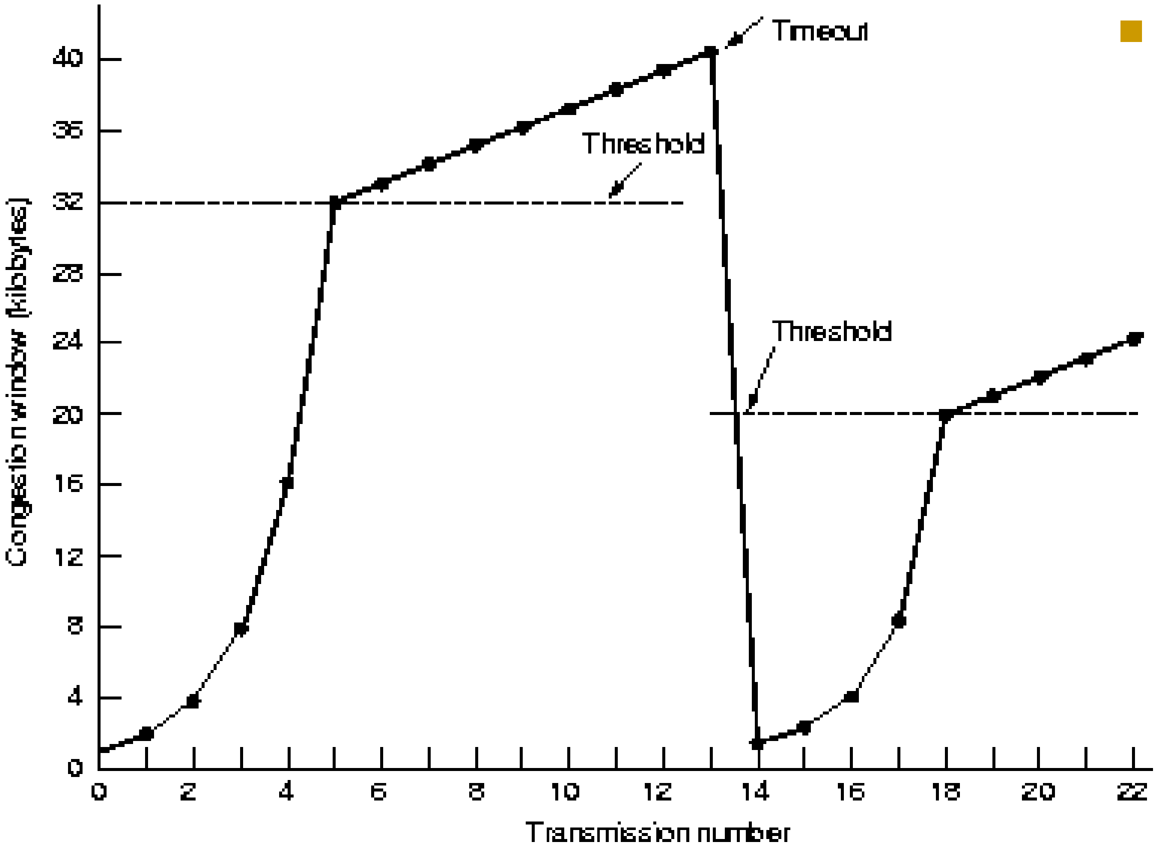
Some measurements to the SYN attack

- Using firewall as relay/gateway
  - Firewall acts in between, receive then forward the SYN packet to server
  - Firewall send “fake” ACK to server, then wait a little timeout then send RST to server if no real ACK coming.
- Active Monitoring
  - Monitor the TCP traffic within a local area network and figure out which ones are illegitimate connections.
  - Send RST for the illegitimate connections to close them

TCP Congestion Control

- Source determines how much bandwidth is available for it to send, it starts slow and increases the window of send packet based on ACKS.
- ACKS are also used to control the transmission of packets.
- Uses Additive Increase Multiplicative Decrease (AIMD)
- Uses Retransmission Timeout (RTO) to avoid congestion

TCP Congestion Control



- All the attacker needs to do is generate a TCP flow to force the targeted TCP connection to repeatedly enter a retransmission timeout state

IPsec:
secure communication for the IP layer

Intro

- **Internet Protocol Security** (**IPsec**) is a protocol suite for securing Internet Protocol (IP) communications by authenticating and encrypting each IP packet of a data stream.
  - Authentication/integrity
  - Confidentiality
  - Protection against replayed packets
- Transparent to applications
  - below transport layer (TCP, UDP)
- IETF IPSEC Working Group
  - Documented in RFCs and Internet drafts

Basics on IPSec

- Protocols
  - Internet key exchange (IKE): set up a ***security association*** (SA) with encryption and authentication keys to be used.
  - Authentication Header (AH): provides integrity and authentication without confidentiality
  - Encapsulating Security Payload (ESP): provides confidentiality and can also provide integrity and authentication
- Both AH/ESP can operate on two different modes
  - Transport-mode: encapsulates an upper-layer protocol (e.g. TCP or UDP) and prepends an IP header in clear
  - Tunnel-mode: encapsulates an entire IP datagram into new packet adding a new IP header

Transport mode

- ESP in Transport Mode
  - encrypts and optionally authenticates the IP payload (data), but not the IP header.
- AH in Transport Mode
  - authenticates the IP payload and selected portions of the IP header

Tunnel Mode

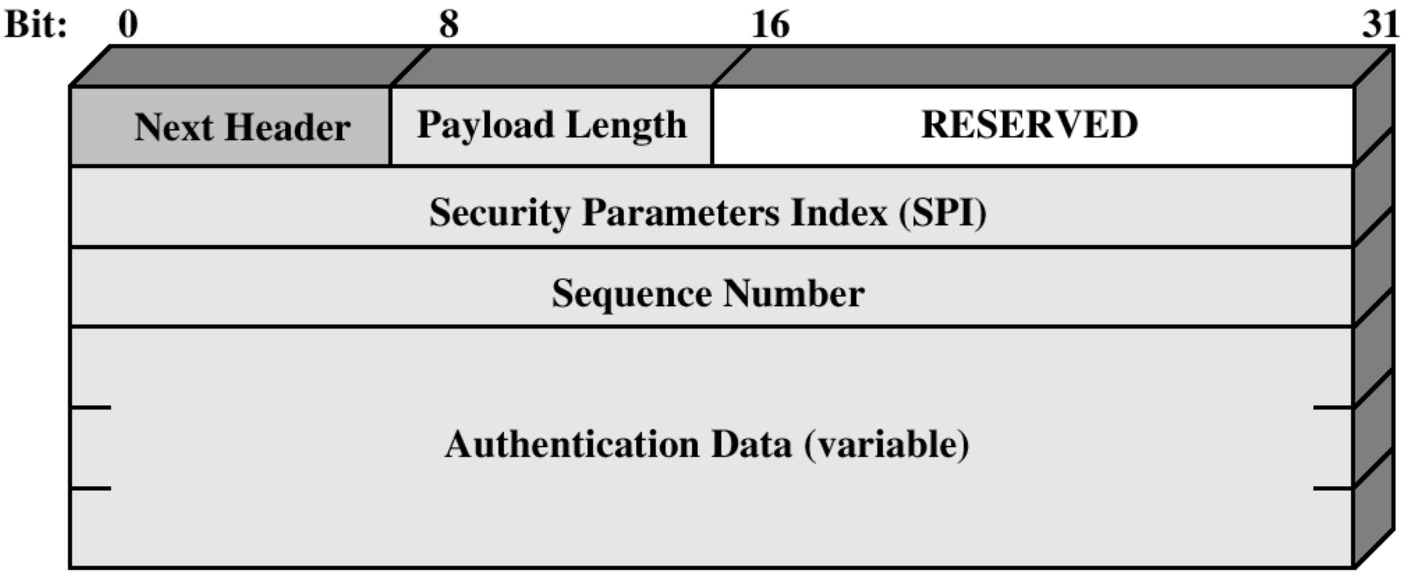
- ESP in Tunnel Mode
  - encrypts and optionally authenticates the entire inner IP packet, including the inner IP header.
- AH in Tunnel Mode
  - authenticates the entire inner IP packet and selected portions of the outer IP header.

Security Associations

- SA- the basis for building security functions into IP.
- A security association is simply the bundle of algorithm selection and parameters (such as keys) that is being used to encrypt and authenticate a particular flow in one direction.
  - SPI (Security Parameter Index) + IP destination address uniquely identifies a particular Security Association.
- Therefore, in normal bi-directional traffic, the flows are secured by a pair of security associations.
  - SAs are unidirectional, sender supplies SPI to receiver.

Authentication Header

- Provides support for data integrity and authentication (MAC) of IP packets, using HMAC based on MD5 or SHA1.
- Defends against replay attacks (sequence number)



Bit: 0 8 16 31

Next Header | Payload Length | RESERVED

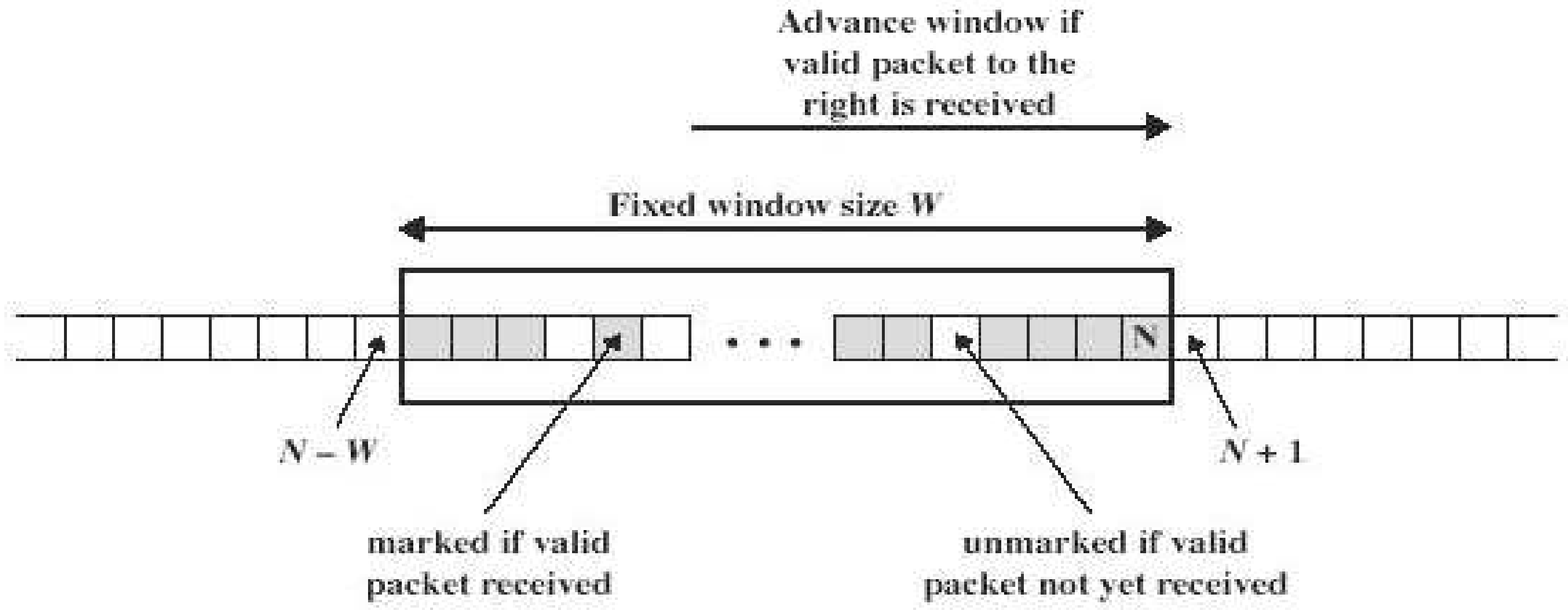
Security Parameters Index (SPI)

Sequence Number

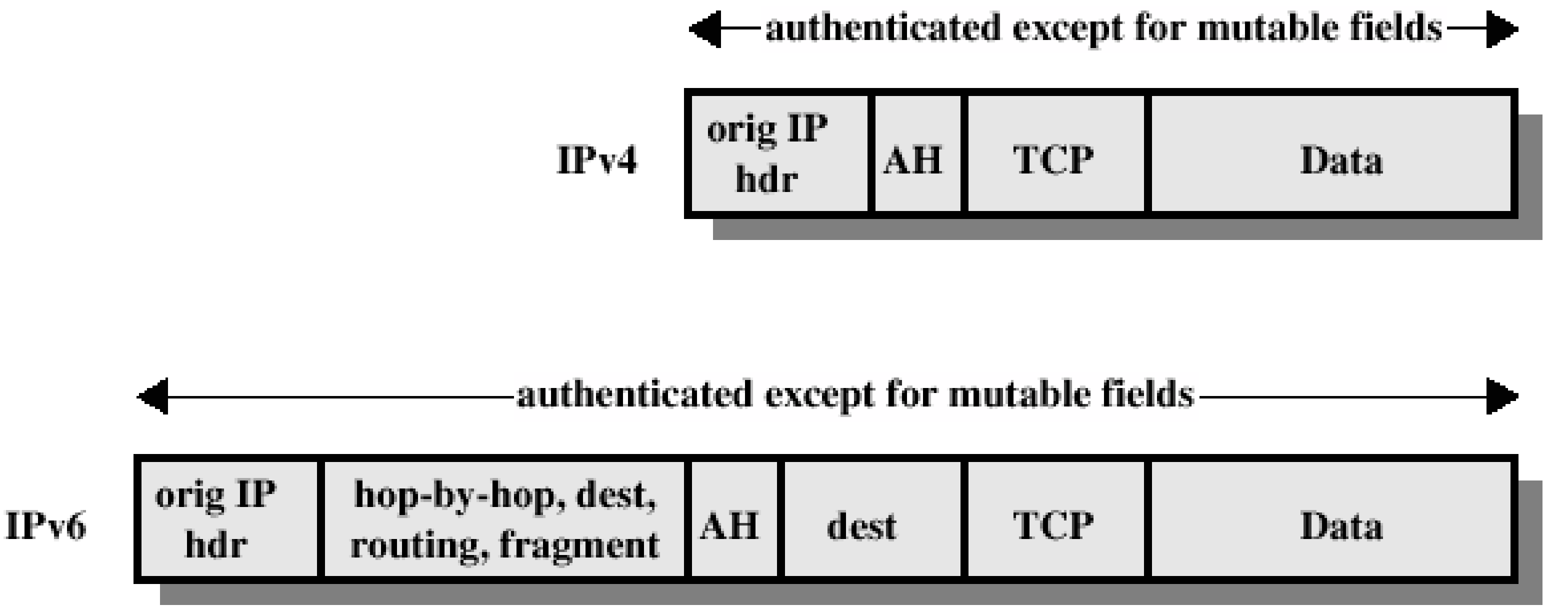
Authentication Data (variable)

AH: Preventing Replay

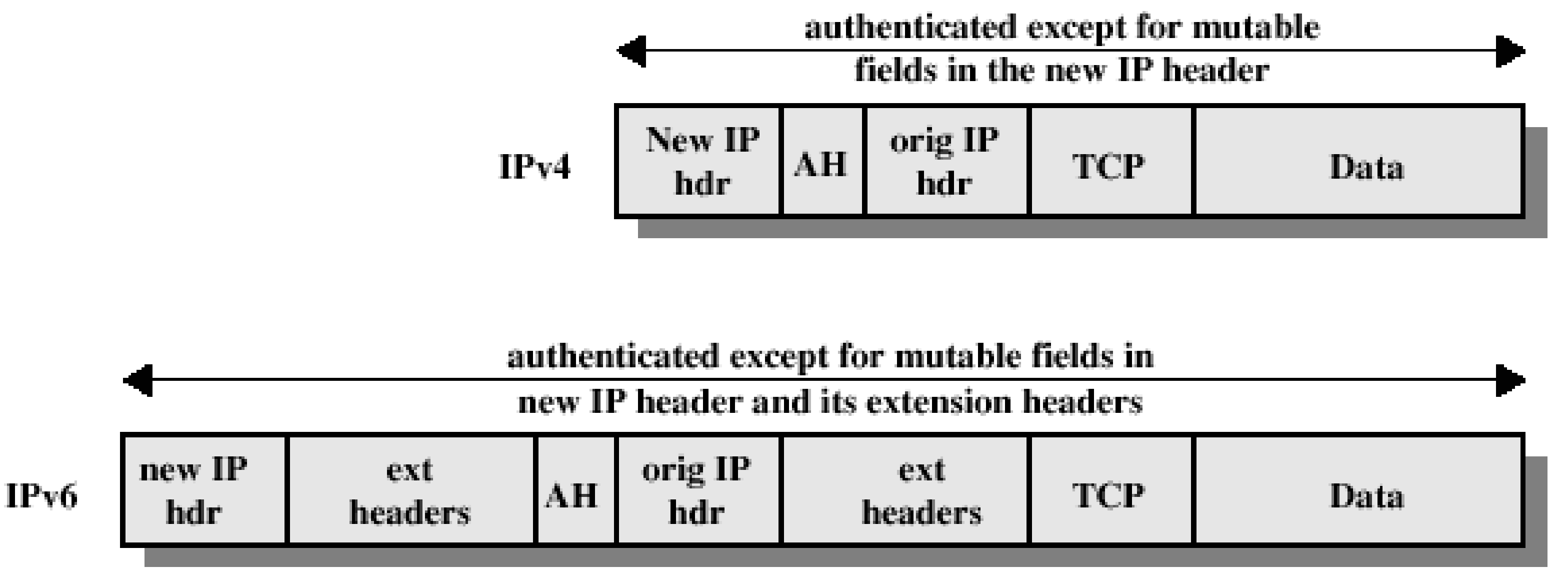
- When a SA is established, sender initializes sequence counter to 0.
- Every time a packet is sent the counter is incremented and is set in the sequence number in the AH header.
- When sequence number 2^{32} - 1 is reached, a new SA should be negotiated.


Advance window if valid packet to the right is received
Fixed window size W
N
N – W
N + 1
marked if valid packet received
unmarked if valid packet not yet received

AH Authentication: Transport Mode


authenticated except for mutable fields
IPv4
orig IP hdr
AH
TCP
Data
authenticated except for mutable fields
IPv6
orig IP hdr
hop-by-hop, dest, routing, fragment
AH
dest
TCP
Data

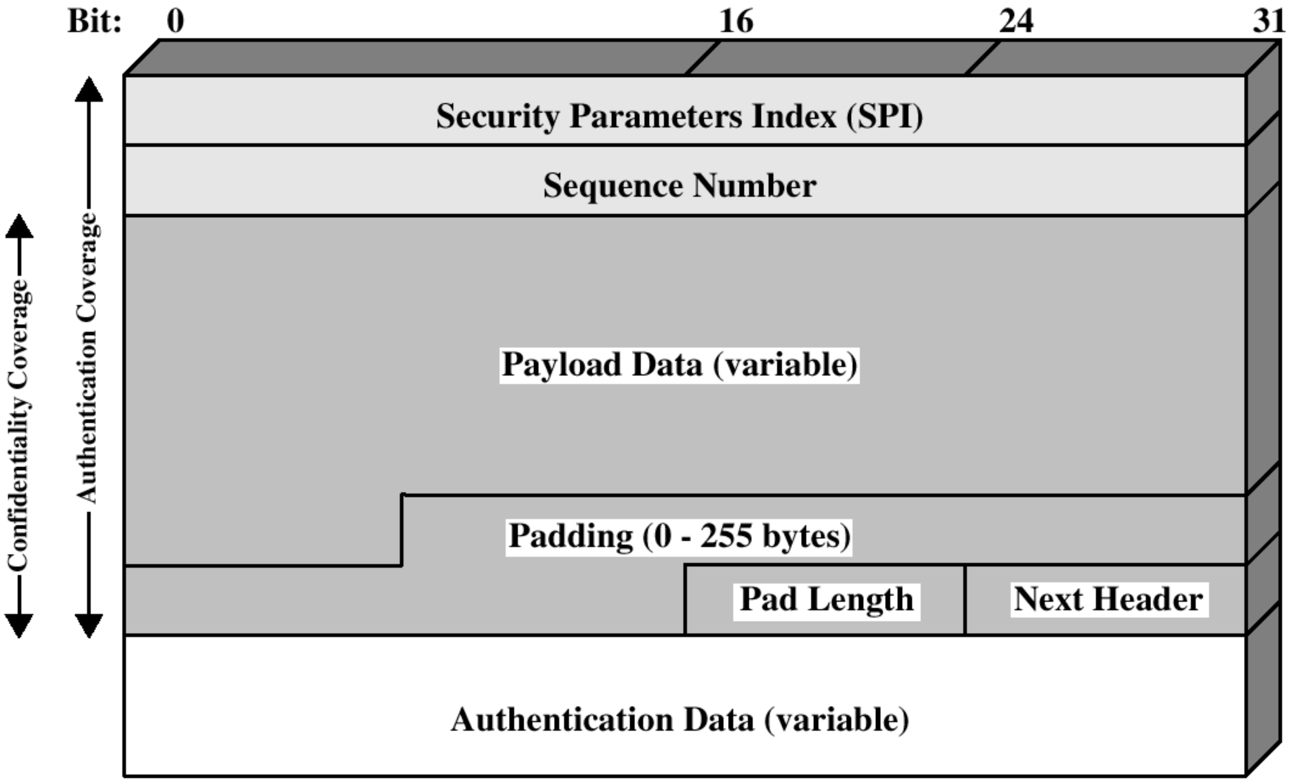
AH Authentication: Tunnel Mode


authenticated except for mutable fields in the new IP header
IPv4
New IP hdr
AH
orig IP hdr
TCP
Data
authenticated except for mutable fields in new IP header and its extension headers
IPv6
new IP hdr
ext headers
AH
orig IP hdr
ext headers
TCP
Data

- The new IP header contains different IP addresses than the ultimate destination and source

Encapsulating Security Payload

- ESP provides confidentiality services, optionally can provide the same services as AH
- Encryption: 3DES, Blowfish, CAST, IDEA, 3IDEA



Bit: 0 16 24 31

Security Parameters Index (SPI)

Sequence Number

Payload Data (variable)

Padding (0 - 255 bytes)

Pad Length

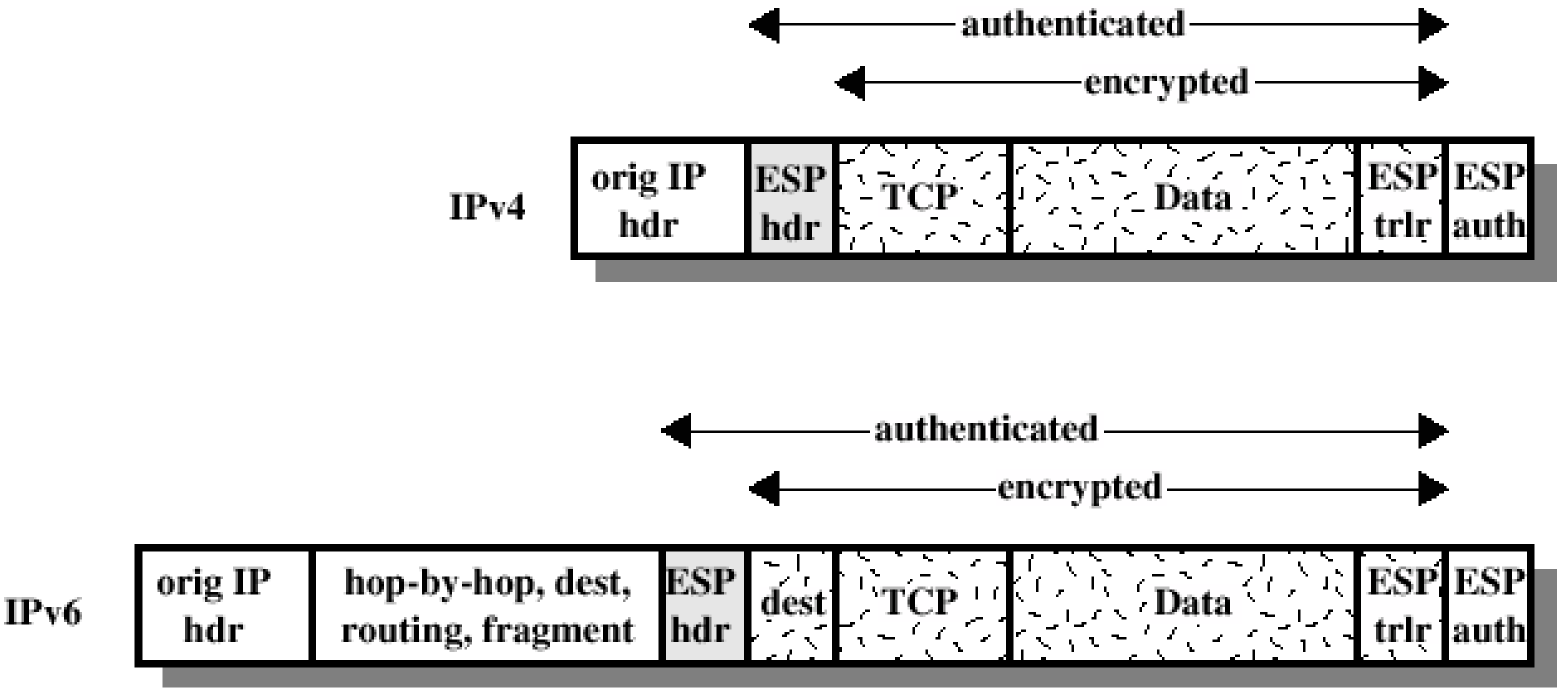
Next Header

Authentication Data (variable)

Confidentiality Coverage

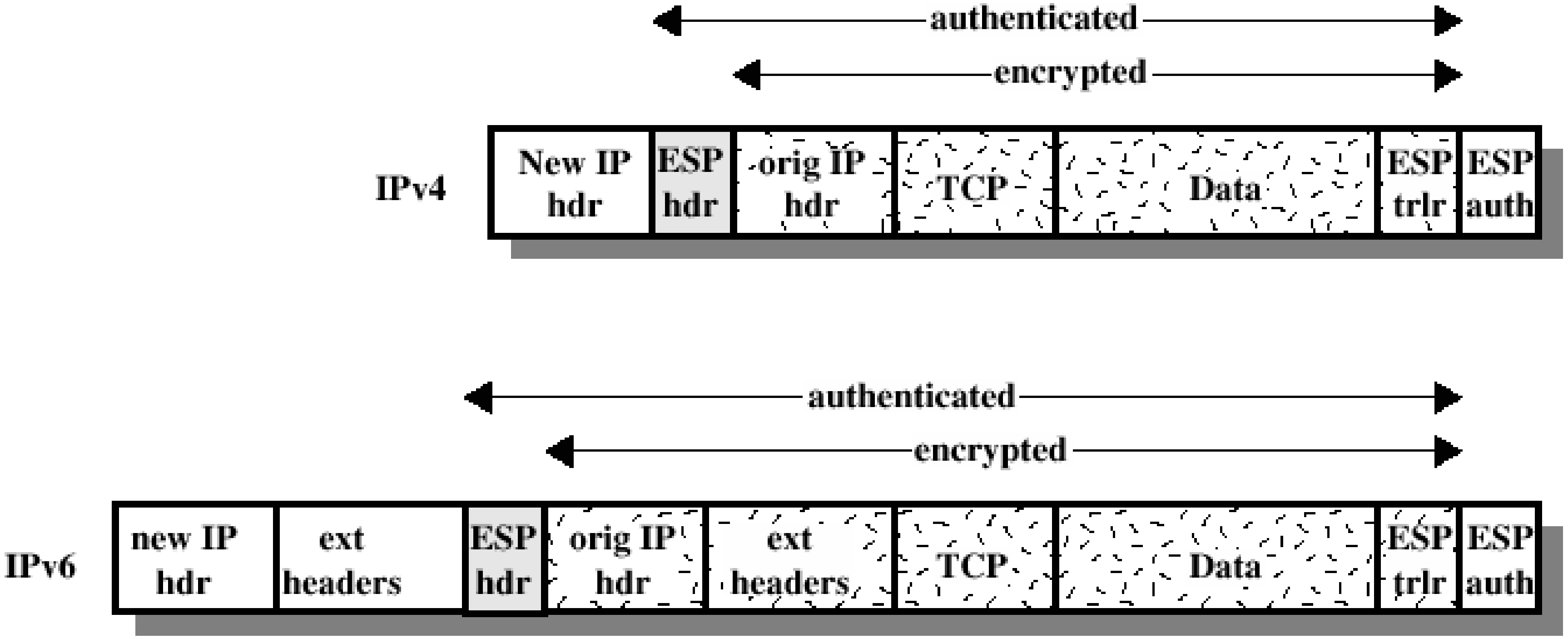
Authentication Coverage

ESP Encryption and Authentication: Transport Mode



(a) Transport Mode

ESP Encryption and Authentication: Tunnel Mode



(b) Tunnel Mode

TLS/SSL: SECURE END-TO-END COMMUNICATION

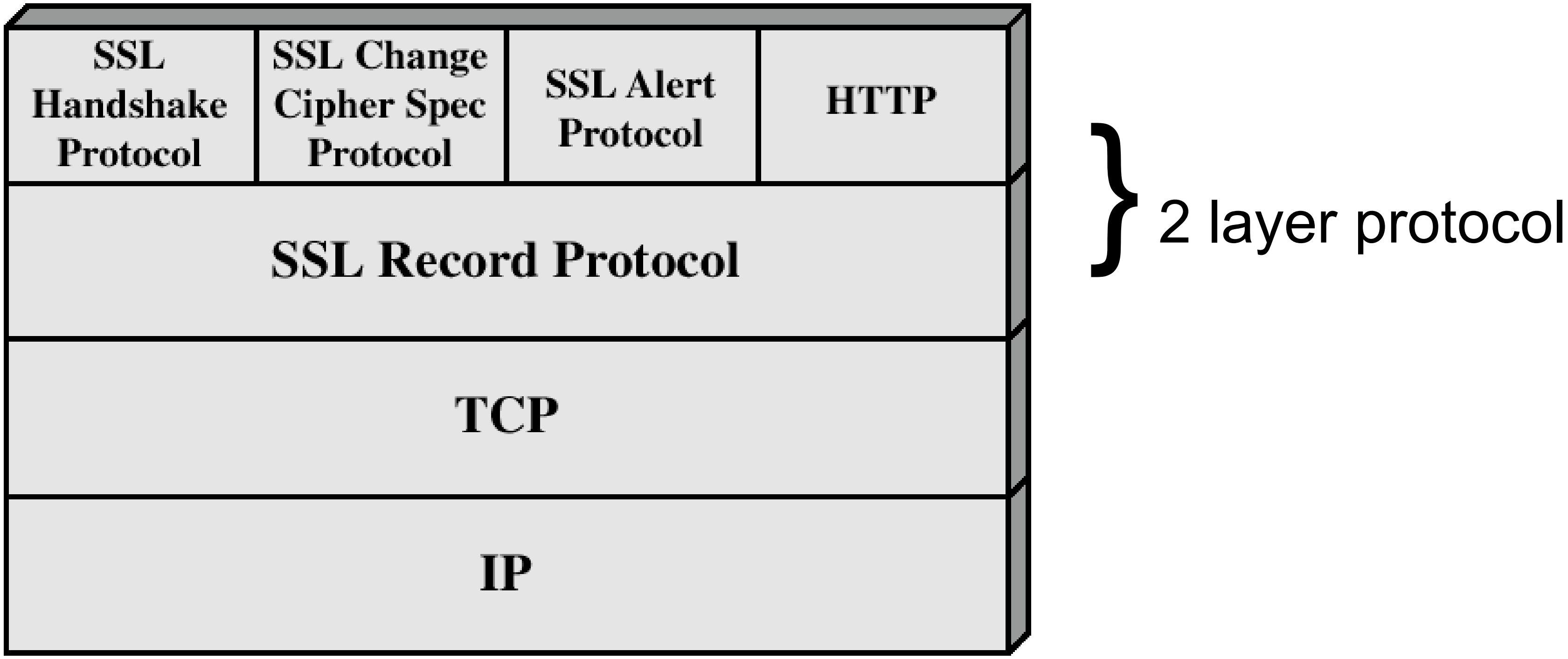
History

- Netscape Communications developed the first three versions of Secure Socket Layer (SSL) with significant assistance from the Web community.
  - Although SSL's development was open, and Netscape encouraged others in the industry to participate, the protocol technically belonged to Netscape.
- Beginning in May **1996, however,** SSL **development became the responsibility** of the Internet Engineering Task Force (IETF).
- The IETF renamed SSL to *Transport Layer Security (TLS).*
  - *The final version* of the first official TLS specification was released in January **1999.**
- Despite the change of names, TLS is nothing more than a new version of SSL.
  - In fact, there are far fewer differences between TLS **1.0** and SSL 3.0 than there are between SSL 3.0 and SSL 2.0.

TLS/SSL basics

- Protocol suite that allows to establish an end-to-end secure channel:
  - Confidentiality: by encryption using DES, 3DES, RC2, RC4, IDEA.
  - Integrity: by computing a MAC and send it with the message; MD5, SHA1.
  - Key exchange: by public key encryption
- Defines how the characteristics of the channel are negotiated
  - key establishment, encryption cipher, authentication mechanism
- Requires reliable end-to-end protocol, so it runs on top of TCP
- Typically, used by other session protocols (HTTPS …)
- Several implementations:
  - e.g. SSLeay, open source implementation (www.openssl.org)

TLS: Protocol Architecture



SSL Protocol Stack

Session and Connection

- **Session**
  - association between a client and a server
  - created by the Handshake Protocol
  - defines secure cryptographic parameters that can be shared by multiple connections.
- **Connection**
  - end-to-end reliable secure communication
  - every connection is associated with a session

Session

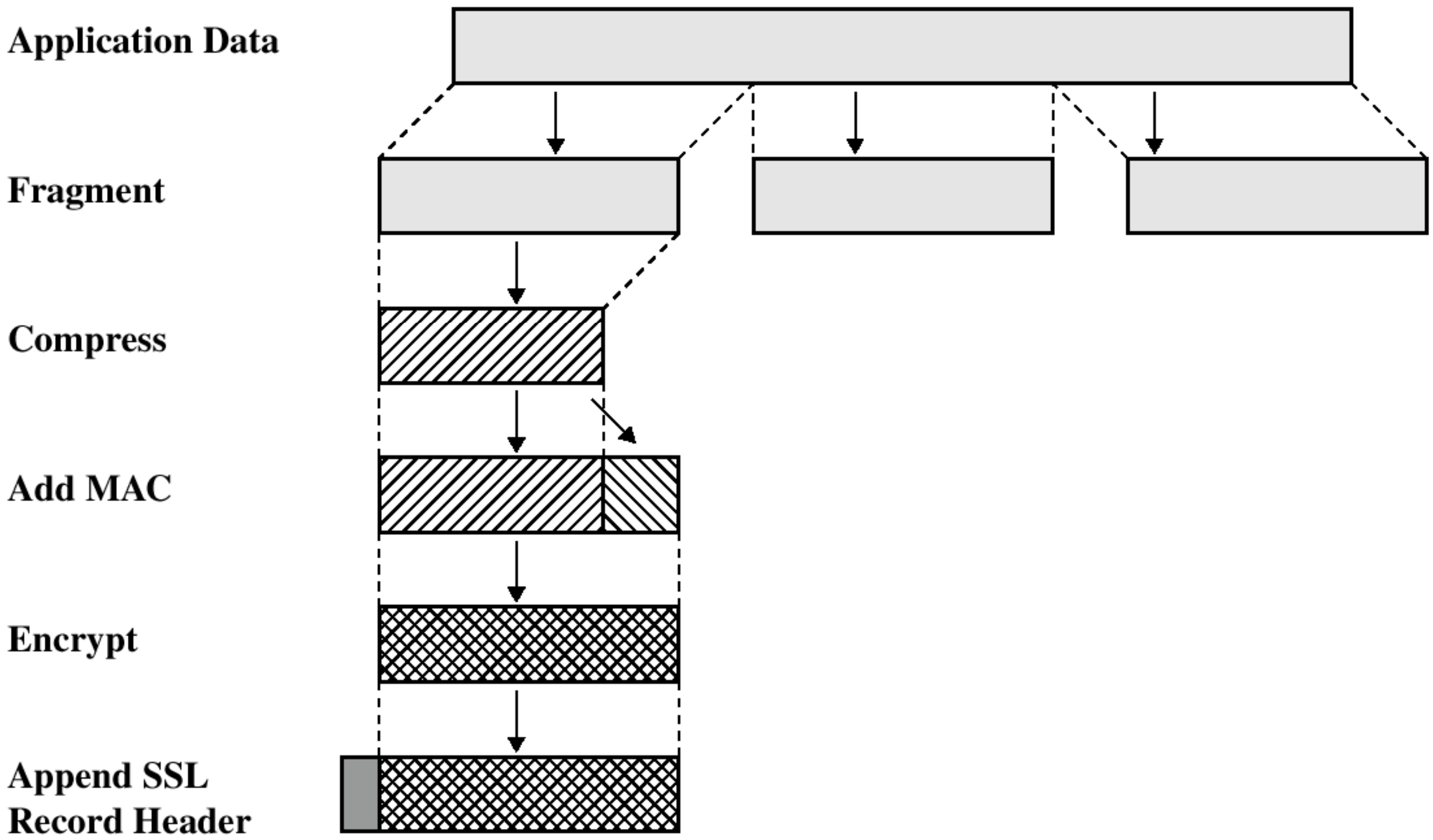
- Session identifier: generated by the server to identify an active or resumable session.
- Peer certificate: X 509v3 certificate.
- Compression method: algorithm used to compress the data before encryption.
- Cipher spec: encryption and hash algorithm, including hash size.
- Master secret: 48 byte secret shared between the client and server.
- Is resumable: indicates if the session can be used to initiate new connections.

Connection

- Server and client random: chosen for each connection.
- Server write MAC secret: shared key used to compute MAC on data sent by the server.
- Client write MAC secret: same as above for the client
- Server write key: shared key used by encryption when server sends data.
- Client write key: same as above for the client.
- Initialization vector: initialization vectors required by encryption.
- Sequence numbers: both server and client maintains such a counter to prevent replay, cycle is 2^{64} - 1.

TLS: SSL Record Protocol

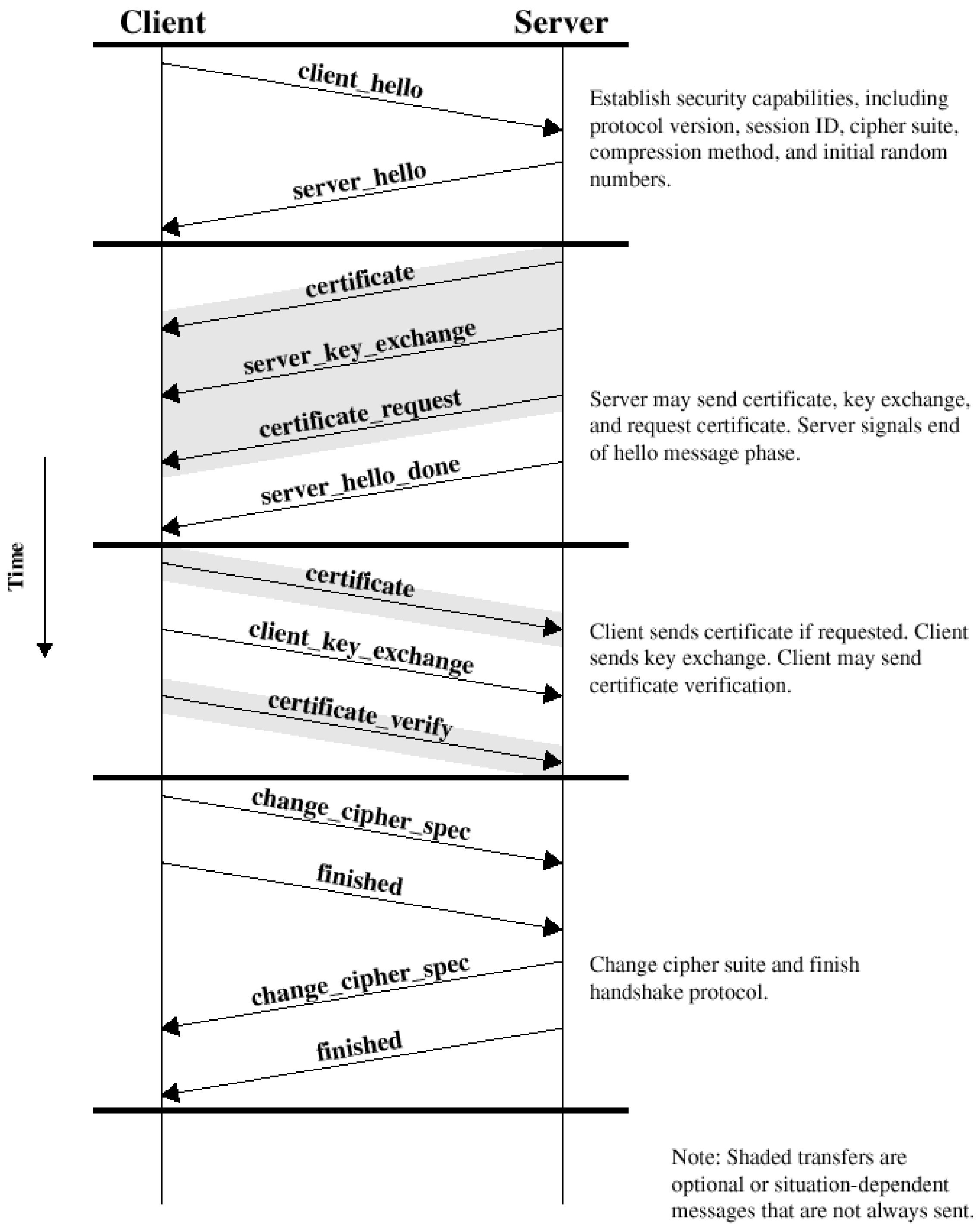
- Provides confidentiality and message integrity using shared keys established by the Handshake Protocol



TLS: Handshake Protocol

- Negotiate Cipher-Suite Algorithms to use
  - Symmetric cipher
  - Key exchange method
  - Message digest function
- Establish the shared master secret
- Optionally authenticate server and/or client

Handshake: At a glance



Establish security capabilities, including protocol version, session ID, cipher suite, compression method, and initial random numbers.

Server may send certificate, key exchange, and request certificate. Server signals end of hello message phase.

Client sends certificate if requested. Client sends key exchange. Client may send certificate verification.

Change cipher suite and finish handshake protocol.

Note: Shaded transfers are optional or situation-dependent messages that are not always sent.

Handshake: Hellos messages

- Client_hello_message has parameters:
  - Version
  - Random: timestamp + 28-bytes random
  - Session ID
  - CipherSuite: cipher algorithms supported by the client, first is key exchange
  - Compression method
- Server responds with the same
- Client may request use of cached session
  - Server chooses whether to accept or not

Handshake: Key Exchange

- Supported key exchange methods:
  - RSA: shared key encrypted with RSA public key
  - Fixed Diffie-Hellman
    - public parameters provided in a certificate
  - Ephemeral Diffie-Hellman
    - Diffie-Hellman with temporary secret key, messages signed using RSA or DSS
  - Anonymous Diffie-Hellman
    - vulnerable to man-in-the-middle

TLS: Authentication

- Verify identities of participants
  - Client authentication is optional
  - Certificate is used to associate identity with public key and other attributes

TLS: Change Cipher Spec/Finished

- Change Cipher Spec completes the setup of the connections.
- Announce switch to negotiated algorithms and values
- The client sends a message under the new algorithms, allows verification of that the handshake was successful

TLS vs. IPSEC

- Security goals are similar
- IPSec more flexible in services it provides, decouples authentication from encryption
- Different granularity: IPSec operates between hosts, TLS between processes

1. IC có quy luật liên hệ gì với văn bản mà từ đó nó được tính ra. Hãy thử so sánh giữa IC của bản mã sinh ra từ cùng một bản tin qua các hệ mã sau đây (có giải thích):

- Additive cipher
- One-time-pad cipher
- Vigenere cipher with key "loveisblue"
- Vigenere cipher with key "blue"
- DES

2. Giả sử ta dùng mật mã khối DES với khóa K và vec-tơ khởi đầu IV để bảo mật các tệp dữ liệu nhị phân trong máy tính. Như vậy một tệp dữ liệu gồm 2 khối 64 bit (X, Y) sẽ mã hóa thành (DES_K (X), DES_K(Y)) trong chế độ ECB.

Tệp dữ liệu A có biểu diễn nhị phân (X,Y,Z, X) trong đó X,Y,Z là các khối 64 bits. Cho biết dạng biểu diễn của A khi ta sử dụng chế độ mã hóa:

a) ECB b) CBC c)CTR

3. Một sinh viên có "cải tiến" sửa đổi TT trao chuyển khoá Needham-Schroeder như sau:

1) A → S: A, B, R_A
2) S → A: $E_{K_{AS}}$ (R_A, B, K, $E_{K_{BS}}$ ((K,B))
3) A → B: $E_{K_{BS}}$ (K,A)
4) B → A: E_K (R_B)
5) A → B: E_K (2*R_B)

Cho biết những chỗ sai/không hợp lý

- 1. Tại sao nói độ an toàn của một hệ chữ ký điện tử phụ thuộc vào việc lựa chọn hàm băm?
- Giả sử hàm băm H với đầu ra 64bit được Alice chọn xây dựng chữ ký điện tử. Eve là một kẻ gian muốn lừa dối Alice bằng cách tạo ra hai văn bản X và Y có nội dung đối nghịch nhau nhưng chữ ký của Alice lên cả 2 sẽ giống nhau. Eve có khả năng làm điều này hay không nếu có thời gian chuẩn bị khoảng 1 ngày (từ lúc biết đặc tả của H đến lúc gặp Alice), trong đó Eve có khả năng tính toán mỗi phép băm H mất 10^{-6}s ?

2. Phát biểu định lý Euler và ý nghĩa ứng dụng của nó với việc xây dựng hệ khóa công khai. Hãy cho biết giá trị và nêu cách tính giá trị của: φ(131) b) φ(133) c) φ(30)

3. Một sinh viên có “cải tiến” sửa đổi TT trao chuyển khoá Needham-Schroeder như sau:

1) A → S: A, B, R_A
2) S → A: $E_{K_{AS}}$ (R_A, B, K, $E_{K_{BS}}$ ((K,B))
3) A → B: $E_{K_{BS}}$ (K,A)
4) B → A: E_K (R_B)
5) A → B: E_K (2*R_B)

Cho biết những chỗ sai/không hợp lý