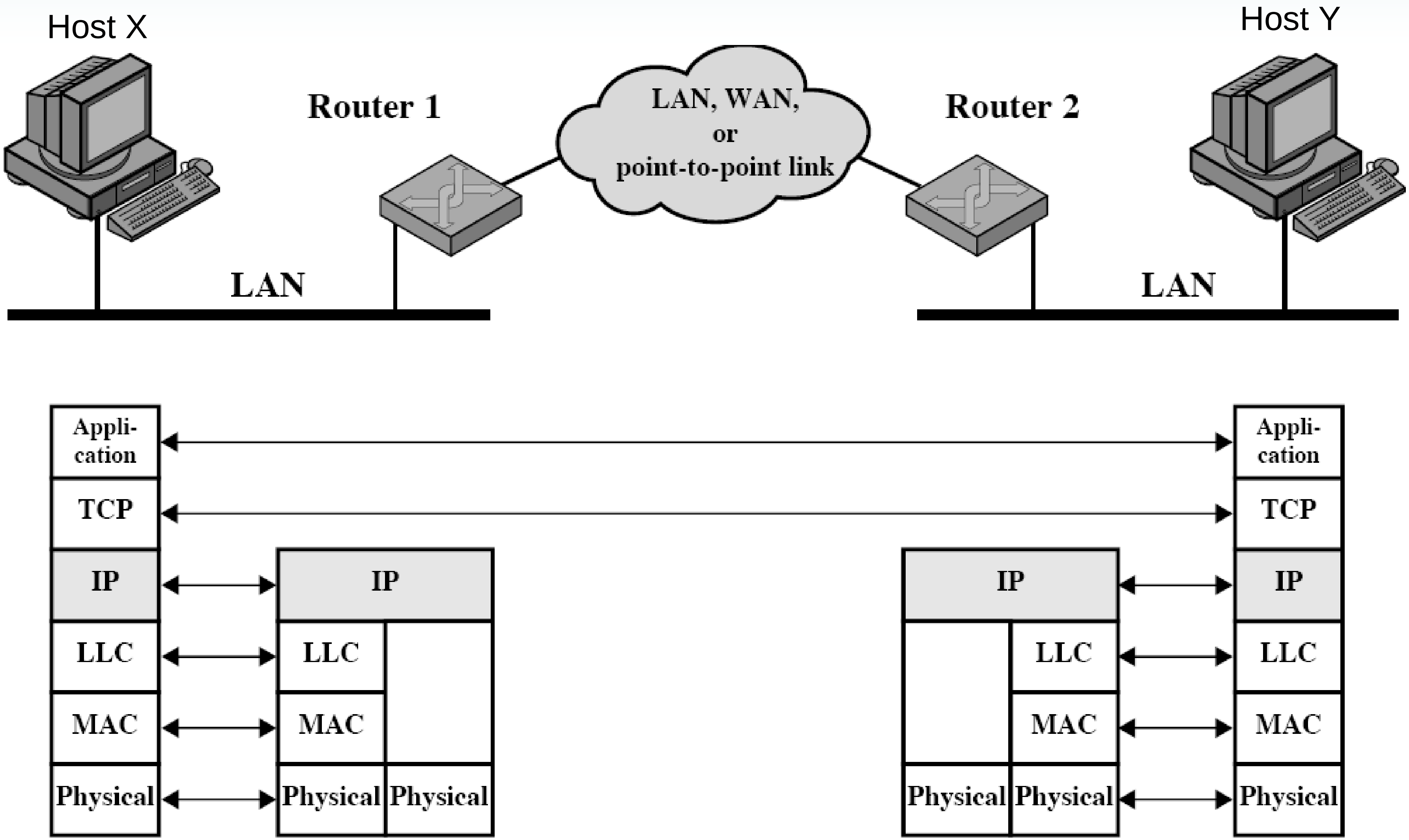
Internet Security

CSS 322 – Security and Cryptography

Contents

- Review of Internet Architecture
- Network Layer Security
  - IPsec
- Transport Layer Security
  - TLS
    - Used by HTTPS and others
- Email Security
  - PGP

Internet Architecture



Host X

Host Y

Router 1

Router 2

LAN, WAN, or point-to-point link

LAN

LAN

Appli-cation

TCP

IP

LLC

MAC

Physical

IP

LLC

MAC

Physical

Physical

IP

LLC

MAC

Physical

Physical

Appli-cation

TCP

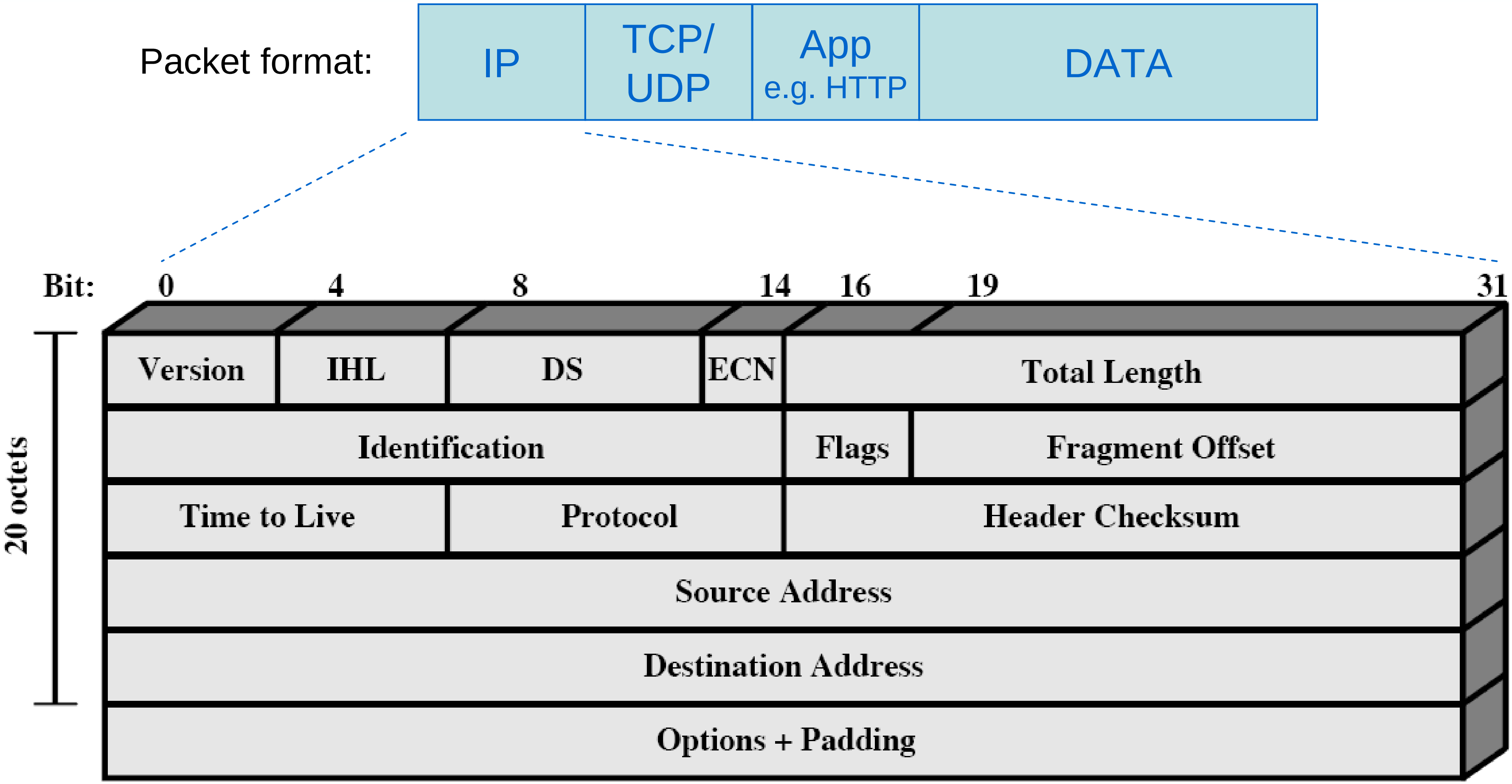
IP

LLC

MAC

Physical

IPv4 Datagram Header

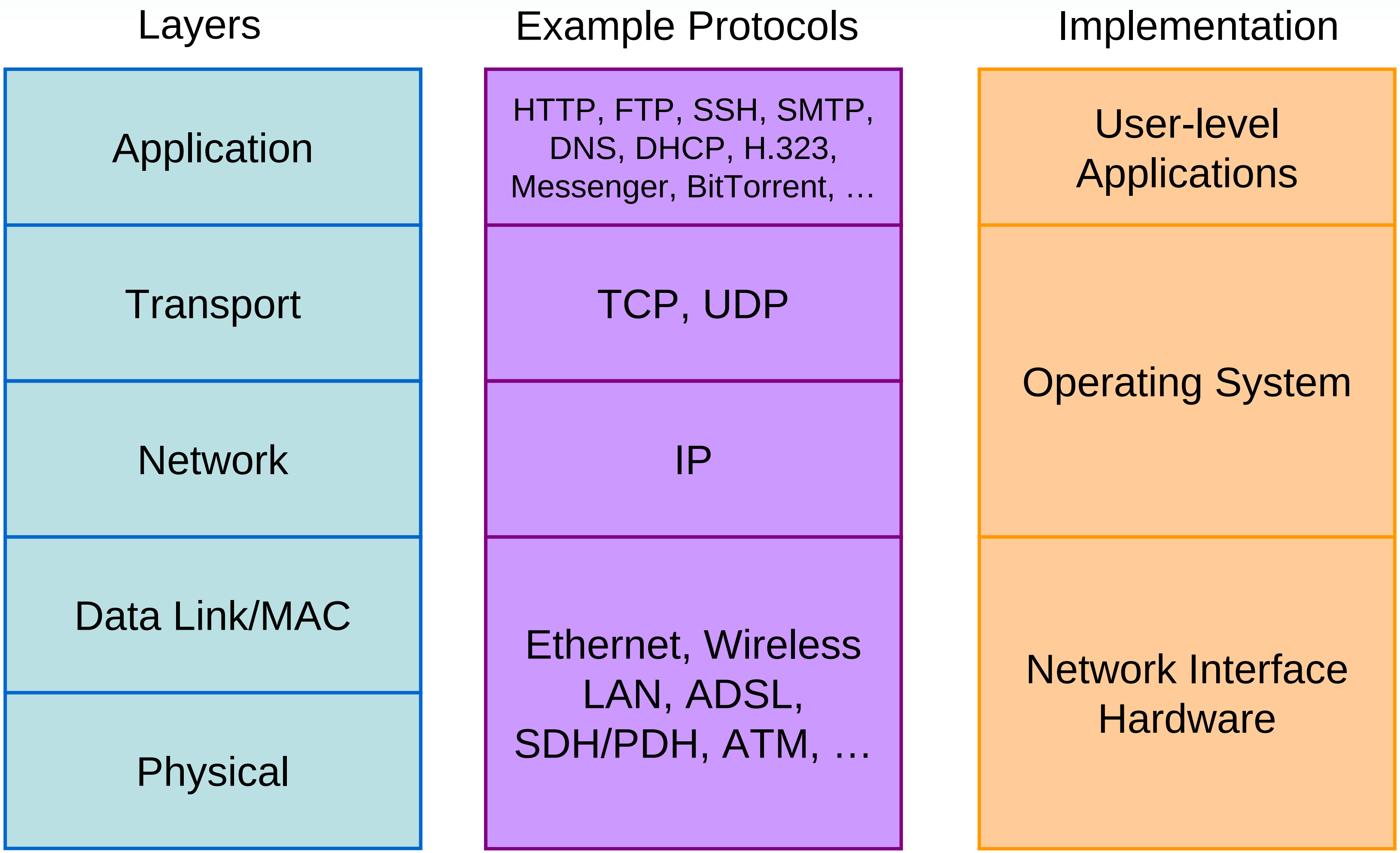


Packet format:

IP	TCP/ UDP	App e.g. HTTP	DATA

Version	IHL	DS	ECN	Total Length	
Identification				Flags	Fragment Offset
Time to Live		Protocol		Header Checksum	
Source Address					
Destination Address					
Options + Padding					

Internet Protocols

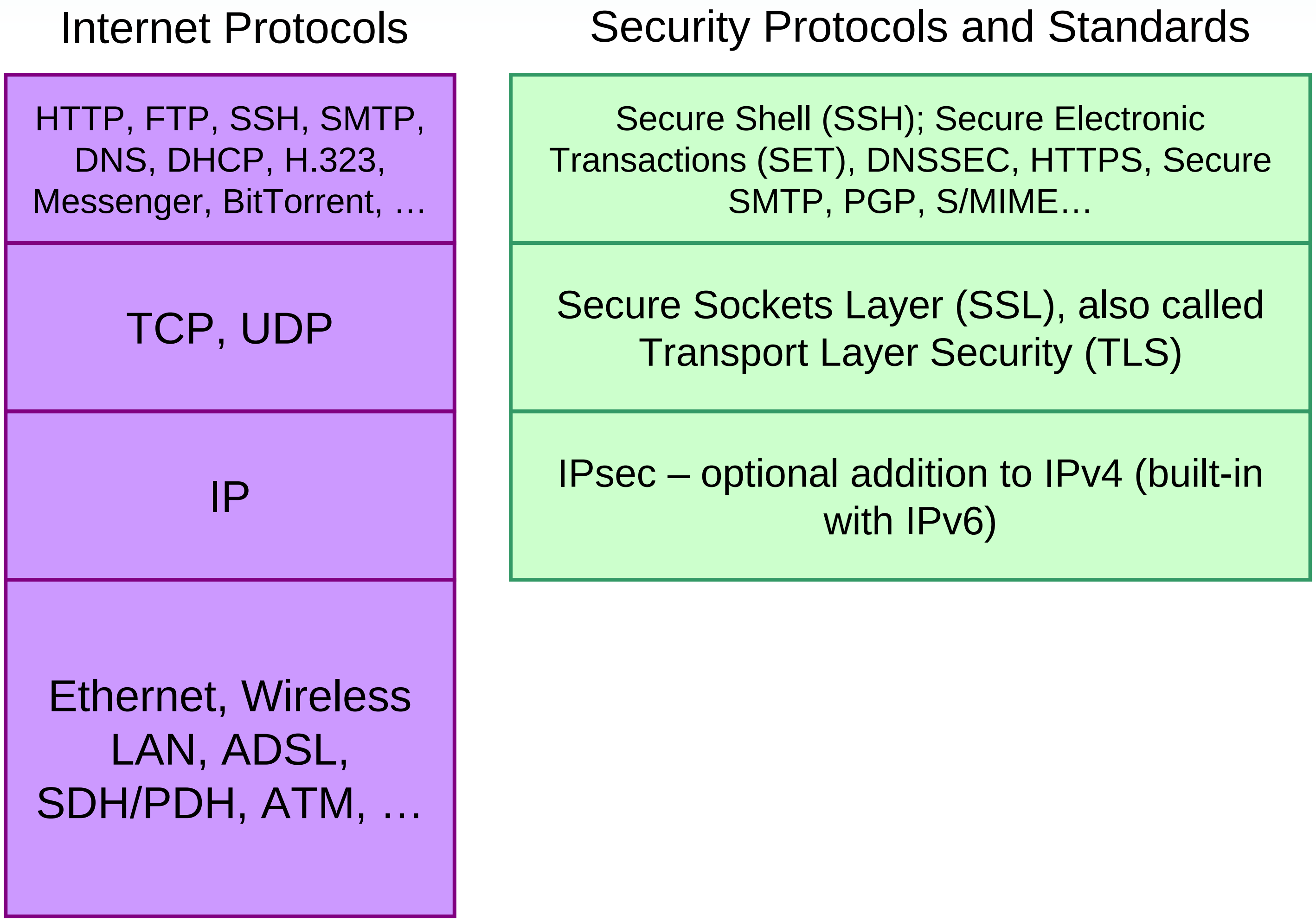


Layers	Example Protocols	Implementation
Application	HTTP, FTP, SSH, SMTP, DNS, DHCP, H.323, Messenger, BitTorrent, …	User-level Applications
Transport	TCP, UDP	Operating System
Network	IP	
Data Link/MAC	Ethernet, Wireless LAN, ADSL, SDH/PDH, ATM, …	Network Interface Hardware
Physical		

Security in Standard Internet Protocols

- The Internet was originally used by researchers, academics and government employees in 1970’s and 1980’s
  - No need for in-built security
    - Everyone was trustworthy; if you wanted security, simple to use proprietary application between users
  - As a result, most Internet protocols do not have any security mechanisms
    - IP, TCP, UDP, HTTP, FTP, SMTP, DNS, …
- Growth and commercialisation of Internet in 1990’s
  - Businesses, governments, consumers depend on Internet
  - Significant growth in malicious users
    - Financial, political, personal motivations
  - Security is now vital
- Most Internet security protocols are add-ons or enhancements to existing protocols
  - IP/IPsec; TCP/TLS; Telnet/SSH, FTP/SSH, …

Internet Security Protocols/Standards



Internet Protocols	Security Protocols and Standards
HTTP, FTP, SSH, SMTP, DNS, DHCP, H.323, Messenger, BitTorrent, …	Secure Shell (SSH); Secure Electronic Transactions (SET), DNSSEC, HTTPS, Secure SMTP, PGP, S/MIME…
TCP, UDP	Secure Sockets Layer (SSL), also called Transport Layer Security (TLS)
IP	IPsec – optional addition to IPv4 (built-in with IPv6)
Ethernet, Wireless LAN, ADSL, SDH/PDH, ATM, …	

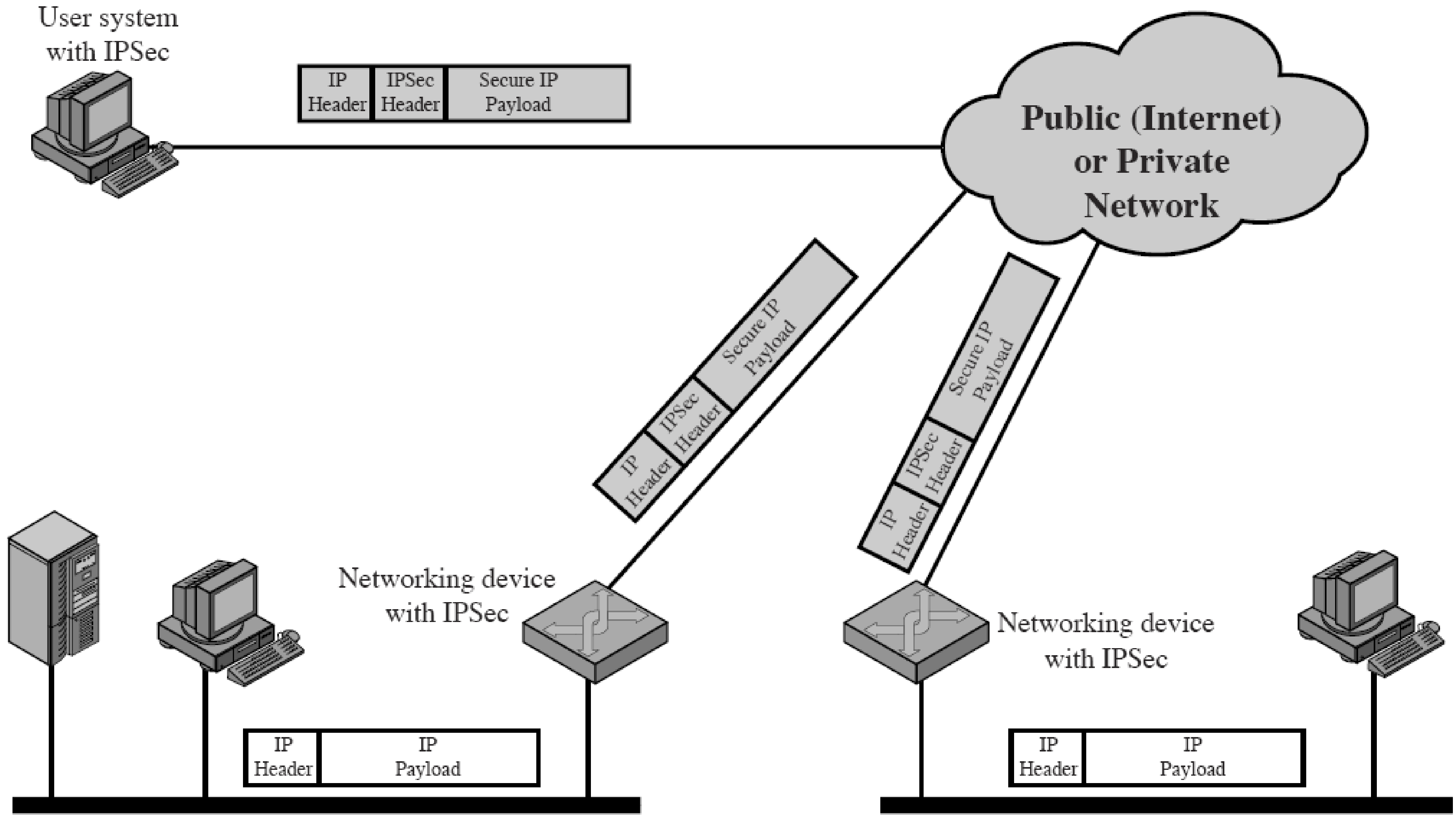
IPsec

Network Layer Security

IPsec

- Internet Engineering Task Force (IETF) defined RFC 2401 (Internet security architecture)
  - IPsec is optional for IPv4 and mandatory in IPv6
    - Mandatory: implementations must support it; but users do not have to use it
  - Implemented as extension headers for IP
- Functionality offered by IPsec:
  - Authentication: verify the sender of IP datagrams
  - Confidentiality: encrypt contents of IP datagrams
  - Data Integrity: guarantee integrity of IP datagrams
  - Key Management: secure exchange of keys
- Allows all traffic to be encrypted at IP (network layer) level
  - Can provide security for all Internet applications (web browsers, email, e-commerce, …)
  - No need to change application or transport protocol software
    - Must have IPsec support on selected PCs, routers, firewalls

Example IPsec Scenario



IPsec Components

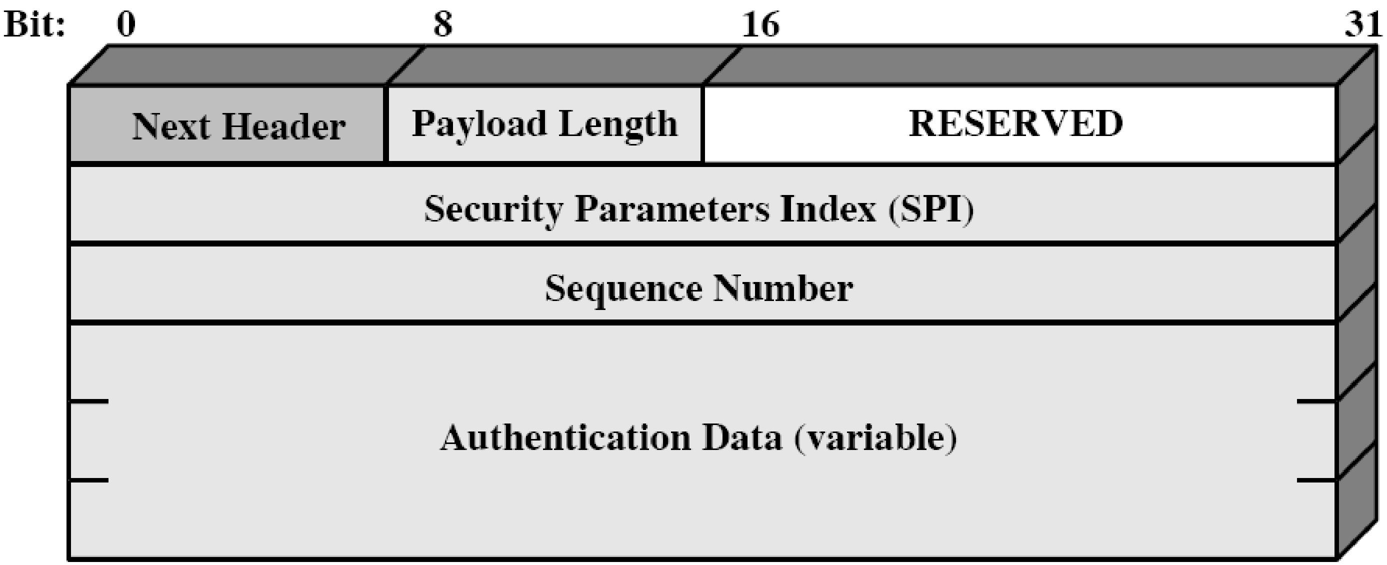
- Security Association (SA)
  - Sender and receiver must establish relationship, called Security Association
  - Traffic sent within that SA is given services agreed upon between sender and receiver
- Encapsulating Security Payload (ESP)
  - Allows for encryption of payload (e.g. TCP packet), as well encryption plus authentication of payload
- Authentication Header (AH)
  - Separate from ESP, allows for authentication-only of payload
- Key Management
  - Mechanisms for exchanging keys
  - Two automated protocols
    - Oakley: based on Diffie-Hellman secret key exchange
    - Internet Security Association and Key Management Protocol (ISAKMP): framework for using different algorithms for key exchange

Security Association

- SA is one-way: sender to receiver
  - That is, for traffic from A to B need one SA; for traffic from B to A need another SA
- SA is identified by:
  - *Security Parameters Index* (SPI): carried in AH and ESP headers
    - Allows A and B to identify the agreed upon algorithms/parameters used for this SA
    - For a SPI, A and B store the relevant parameter values in a local database
      - Parameters include: sequence numbers, encryption/authentication algorithms and keys, lifetime of SA, protocol mode (tunnel or transport) and others
  - *IP Destination Address*
  - *Security Protocol Identifier*: indicates either AH or ESP (depend on which one is used)

Authentication Header

- AH provides two services for IP datagrams
  - Integrity: ensure payload is not changed
  - Authentication: verify the identity of user
- AH uses a Message Authentication Code
  - Users must share a secret key



Bit: 0 8 16 31

Next Header | Payload Length | RESERVED

Security Parameters Index (SPI)

Sequence Number

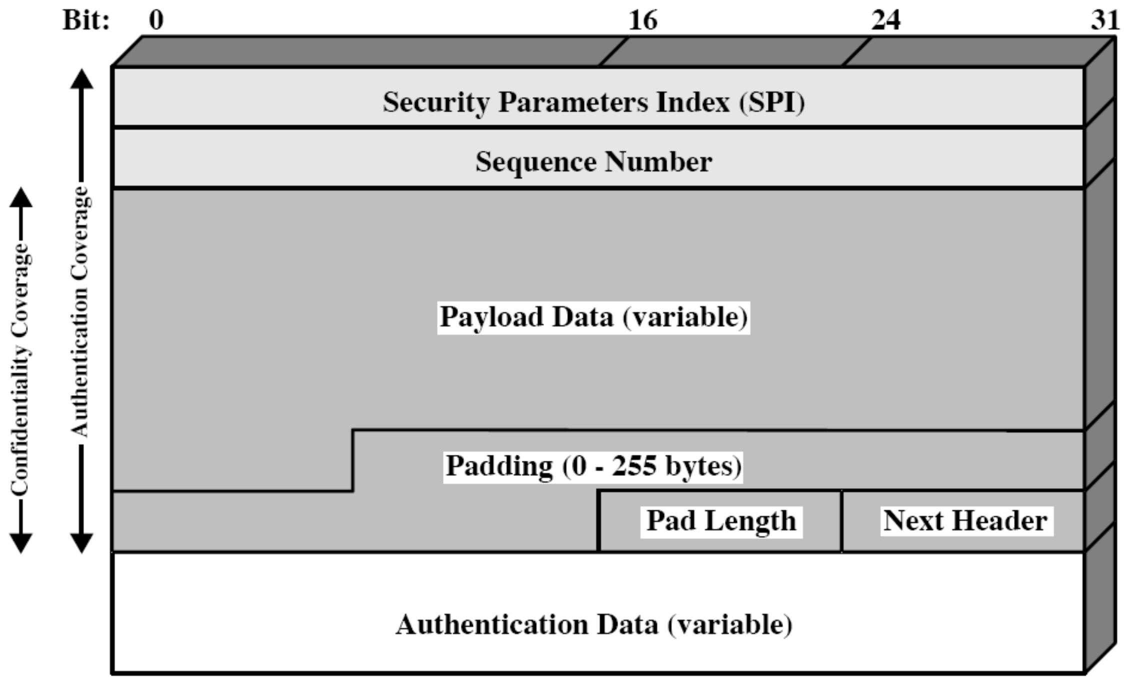
Authentication Data (variable)

AH Services and Algorithms

- AH provides authentication
  - MAC requires sharing of secret key; the sending having the correct secret key authenticates that user
- AH provides integrity check of packet
  - HMAC-MD5 and HMAC-SHA1 must be supported (others are optional)
  - Sender calculates MAC from:
    - IP header fields that do not change (immutable) or are predictable
    - AH fields (except Authentication Data field)
    - Entire upper-layer packet (e.g. TCP header, HTTP header and HTTP data)
  - Only first 96-bits of MAC are sent (in Authentication Data field)
  - Receiver checks the MAC upon receipt of packet
- AH can prevent replay attacks
  - Sequence number in header is used to identify replayed packets

Encapsulating Security Payload

- ESP provides several services for IP datagrams
  - Encryption (confidentiality) of IP datagram
  - Authentication of IP datagram (optional)
  - Integrity of IP datagram



ESP Algorithms

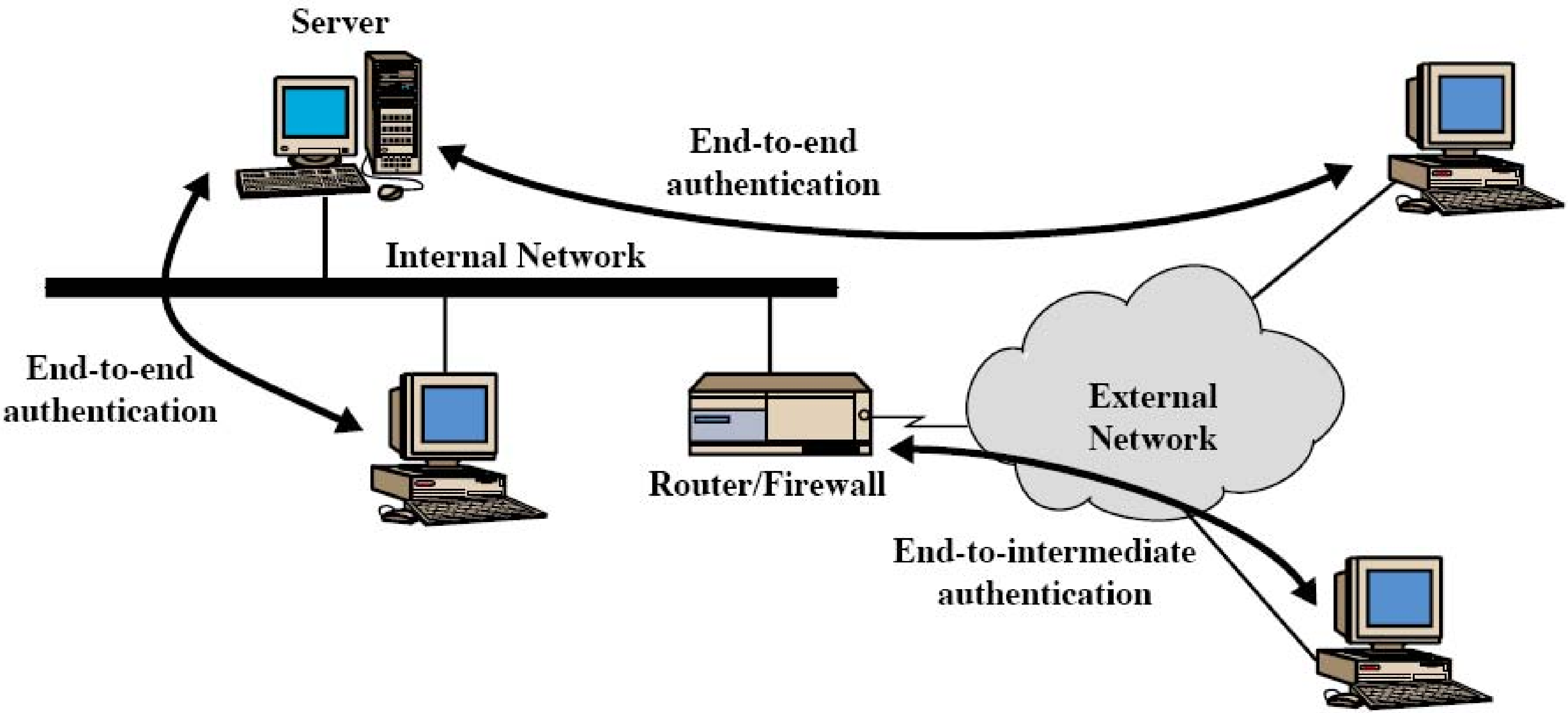
- Encryption
  - Mandatory to support DES; many algorithms are optional:
    - 3DES, RC5, IDEA, CAST, Blowfish, …
- Authentication
  - Same as AH: HMAC-MD5 and HMAC-SHA1 are mandatory

Protocol Modes

- Transport Mode
  - Apply encryption or authentication end-to-end
    - E.g. from PC to PC
  - Original IP header is not protected
    - Only protected TCP/UDP and application layer data

- Tunnel Mode
  - Apply encryption or authentication from intermediate device
    - E.g. from router to router or from router to PC
  - Original IP header is protected
    - Protect IP plus TCP/UDP plus application layer data
  - Often used for creating Virtual Private Networks (VPNs)

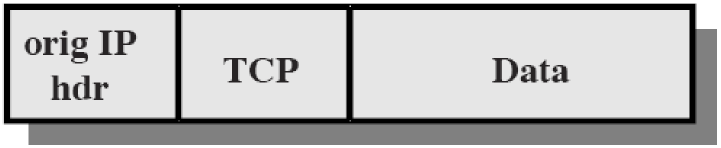
AH and Protocol Modes


Server
End-to-end authentication
Internal Network
End-to-end authentication
Router/Firewall
External Network
End-to-intermediate authentication

- Transport: end-to-end
- Tunnelling: end-to-intermediate, or intermediate-to-intermediate

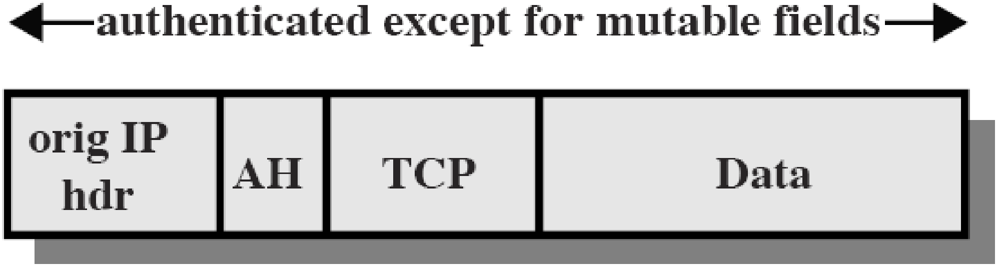
AH and Protocol Modes

- Original IP datagram (before IPsec)



orig IP hdr	TCP	Data

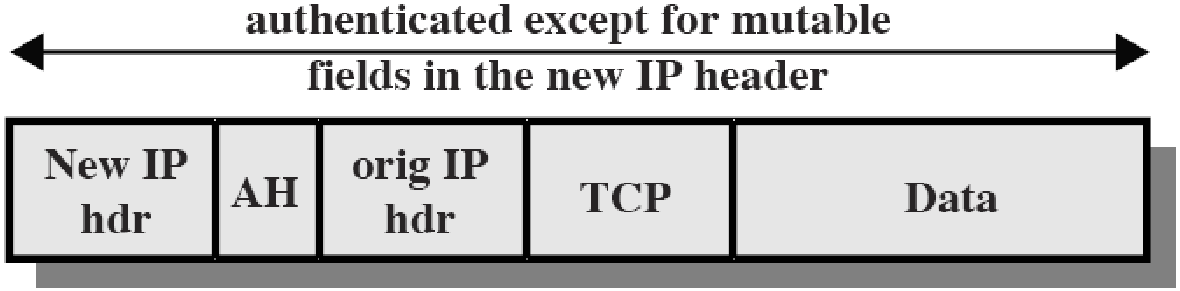
- AH with Transport Mode:



authenticated except for mutable fields

orig IP hdr	AH	TCP	Data

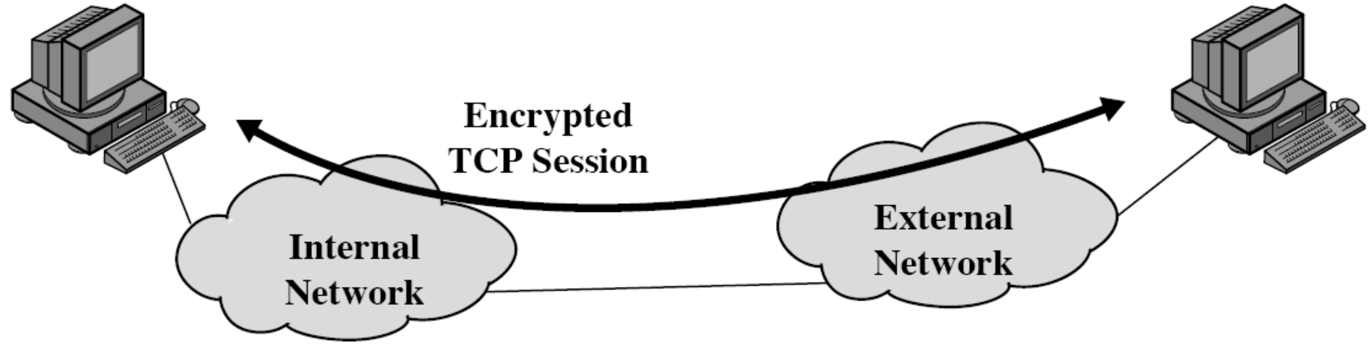
- AH with Tunnelling Mode:



authenticated except for mutable fields in the new IP header

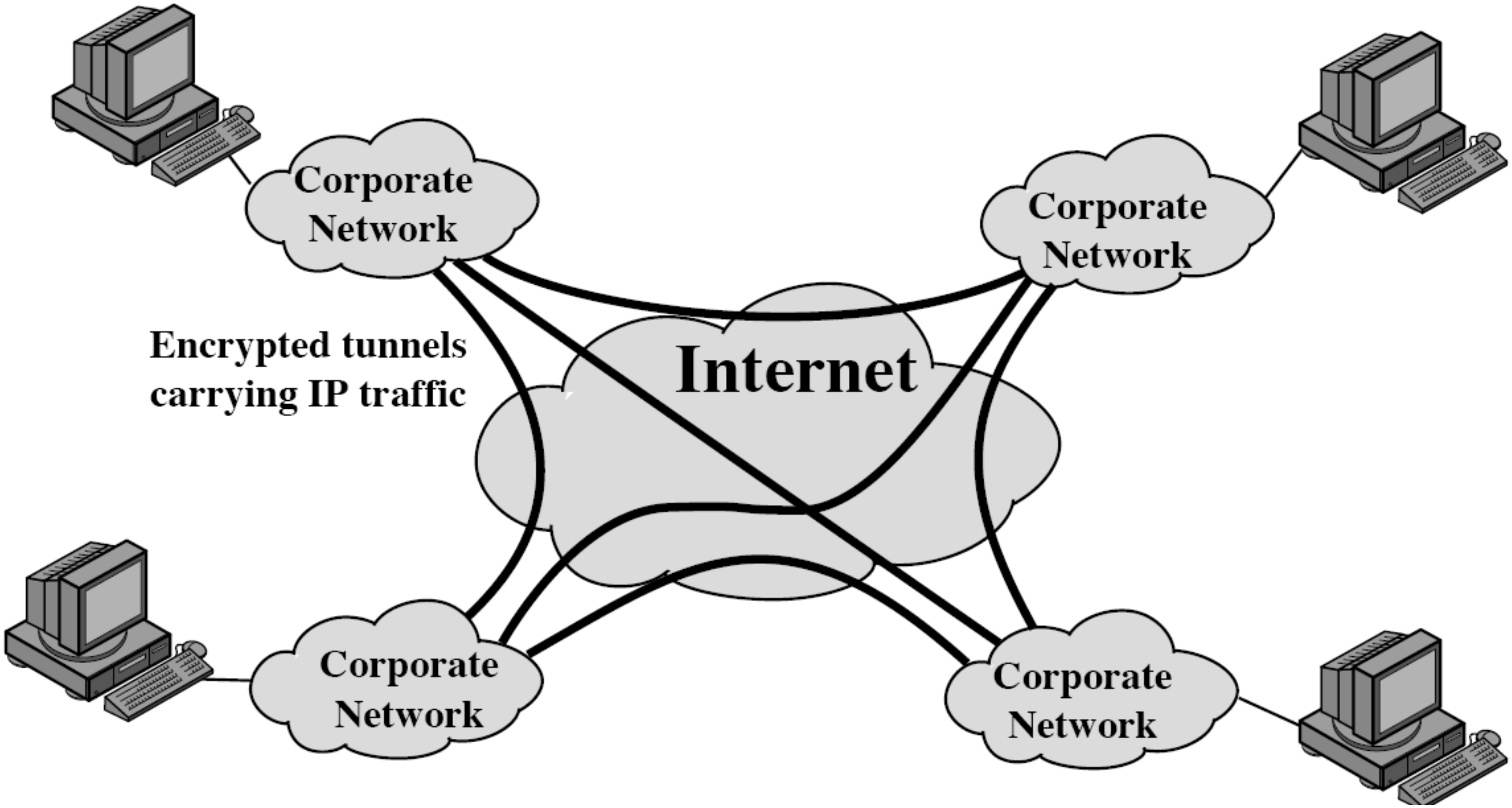
New IP hdr	AH	orig IP hdr	TCP	Data

ESP and Transport Mode



- PCs support IPsec
- Encrypt traffic end-to-end; PC-to-PC

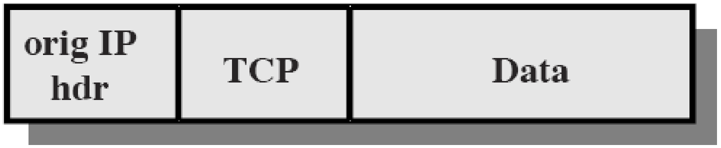
ESP and Tunnelling Mode



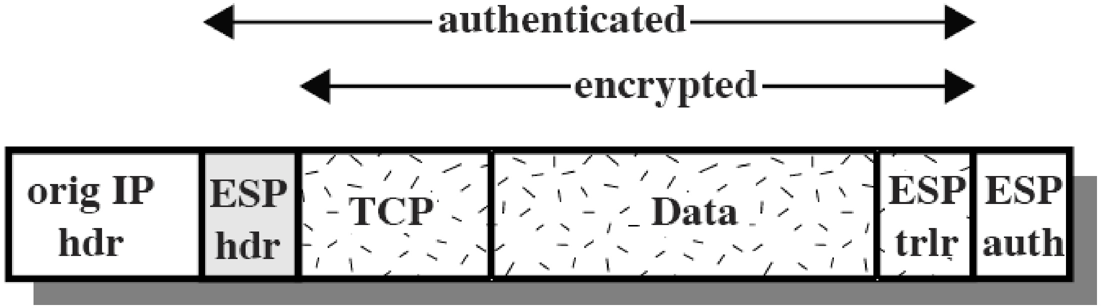
- Hosts/PCs send normal IP traffic (unencrypted)
- Routers at edge of local network creates an IPsec tunnel to other network

ESP and Protocol Models

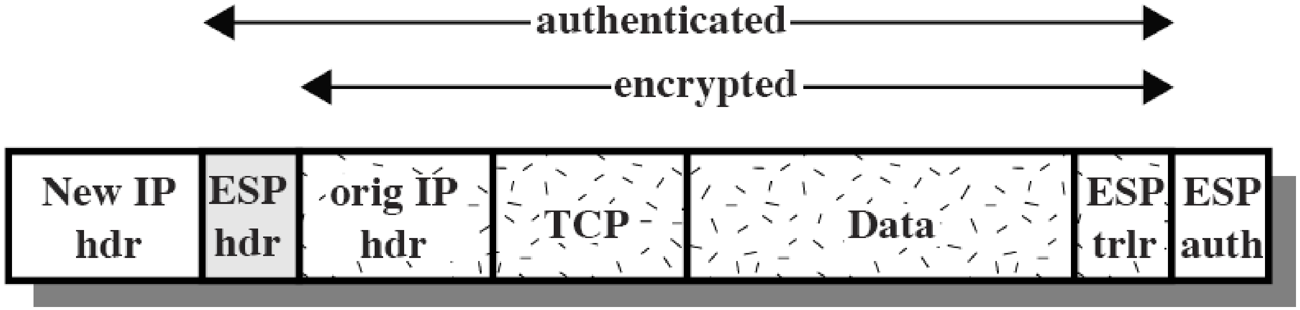
- Original IP datagram (before IPsec):


orig IP hdr
TCP
Data

- ESP and Transport Mode:


authenticated
encrypted
orig IP hdr
ESP hdr
TCP
Data
ESP trlr
ESP auth

- ESP and Tunnelling Mode


authenticated
encrypted
New IP hdr
ESP hdr
orig IP hdr
TCP
Data
ESP trlr
ESP auth

Summary of Protocol Modes

	Transport	**Tunnel**
AH	Authenticate IP payload and selected parts of IP header	Authenticates entire inner IP packet (payload plus header) and parts of outer header
ESP	Encrypts IP payload	Encrypts entire inner IP packet
ESP with Auth.	Encrypts IP payload; authenticates IP payload	Encrypts entire inner IP packet; authenticates inner IP packet

Summary of IPsec Services

	AH	ESP (encrypt only)	ESP (encrypt + auth.)
Access control	✓	✓	✓
Data integrity	✓		✓
Data origin authentication	✓		✓
Anti-replay	✓	✓	✓
Confidentiality		✓	✓
Limited traffic flow confidentiality		✓	✓

Applications of IPsec

- Connecting branches/offices securely over the Internet
  - Create a Virtual Private Network using IPsec from Office A to Office B
    - Use of Internet to connect offices is cheaper than dedicated lines (e.g. DSL, E1, ATM)
    - Use ESP in tunnelling mode
- Secure remote access over Internet
  - Employee connects from home/hotel via a ISP to office
    - VPN from user PC to office router
    - Use ESP in tunnelling mode
- Web sites and e-commerce applications
  - IPsec can be used as an alternative or complement to HTTPS and similar protocols
    - Use ESP in transport mode

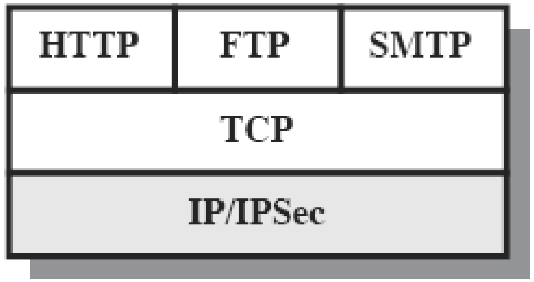
TLS

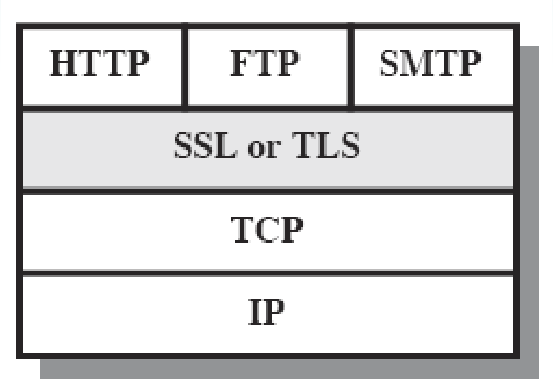
Transport Layer Security

Secure Socket Layer

- SSL originally developed by Netscape
  - SSL v3.1 is known as Transport Layer Security (TLS); develped by IETF
  - We will treat SSL and TLS as the same
- Provides security services between TCP and applications that use TCP
  - HTTP, FTP, SMTP, …
- SSL is commonly used for Web security
  - HTTPS is simply HTTP on top of SSL

SSL versus IPsec

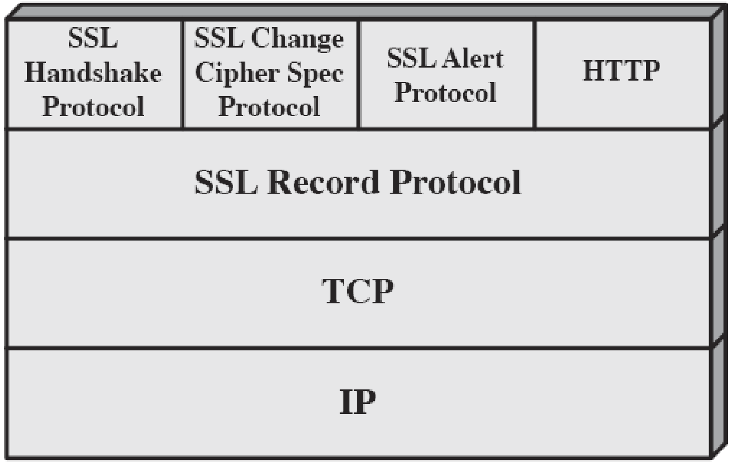




- IPsec is implemented in operating system at IP layer
  - Can be used by any Internet application
- SSL can only be used by applications that use TCP
  - Many real-time and messaging applications use UDP, not TCP
  - Implementation of SSL:
    - Option 1: Implement in operating system or common library so any application can use (e.g. OpenSSL library)
    - Option 2: Applications implement their own instance of SSL
      - E.g. web browsers like IE and Firefox, and web servers like Apache implement SSL

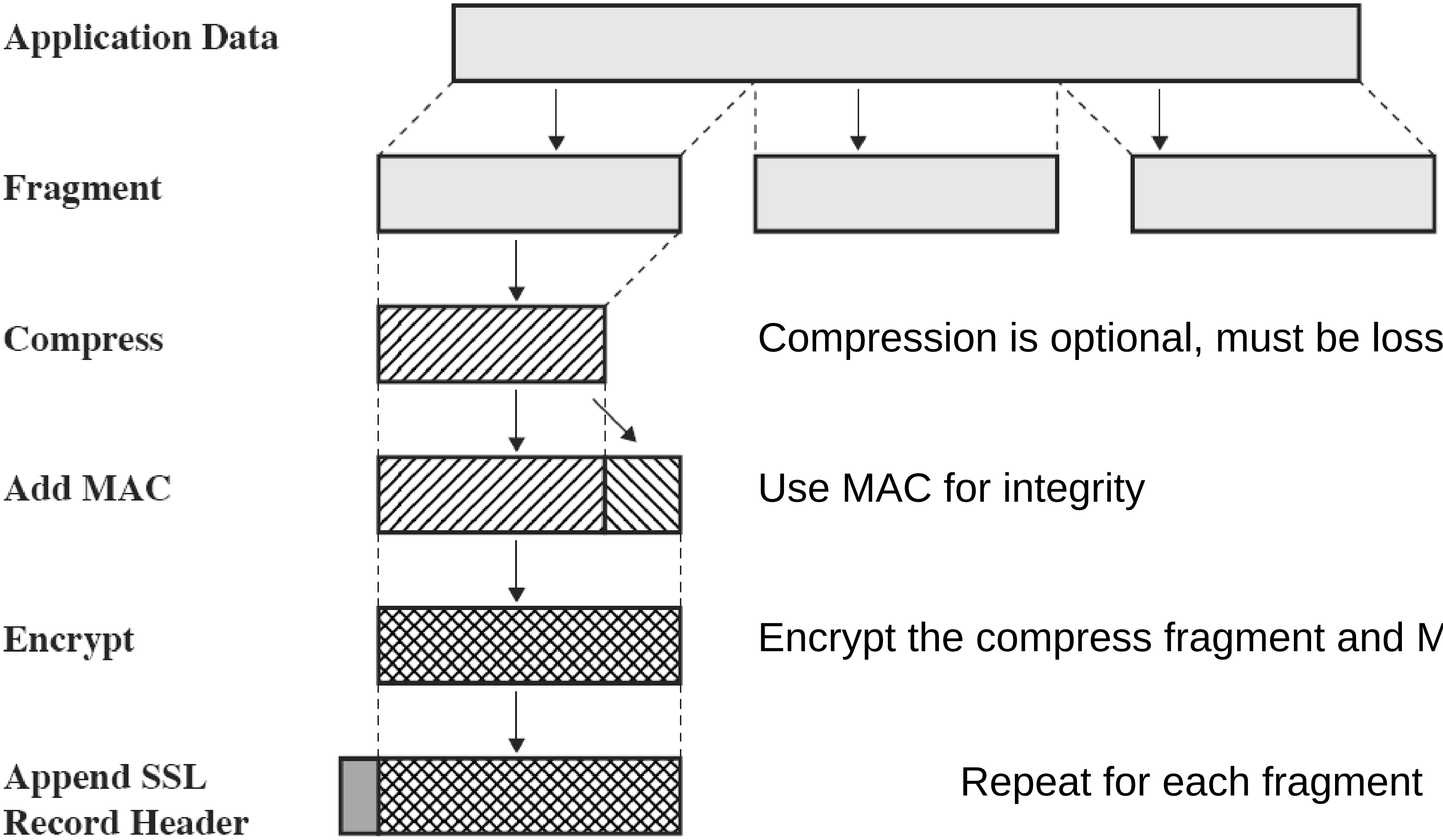
SSL Architecture

- SSL includes two layers:
  - Record Protocol: provides confidentiality and integrity of messages
  - Management layer:
    - Handshake protocol to agree upon parameters upon start
    - Change cipher protocol to change to the next cipher in the session
    - Alert protocol to alert the other side that something has happened
      - E.g. unexpected message, incorrect MAC, handshake failure, illegal parameter, bad certificate, certificate expired, …



SSL Record Protocol

- Confidentiality provided using symmetric key encryption
- Integrity provided using Message Authentication Code



Compression is optional, must be lossless

Use MAC for integrity

Encrypt the compress fragment and MAC

Repeat for each fragment

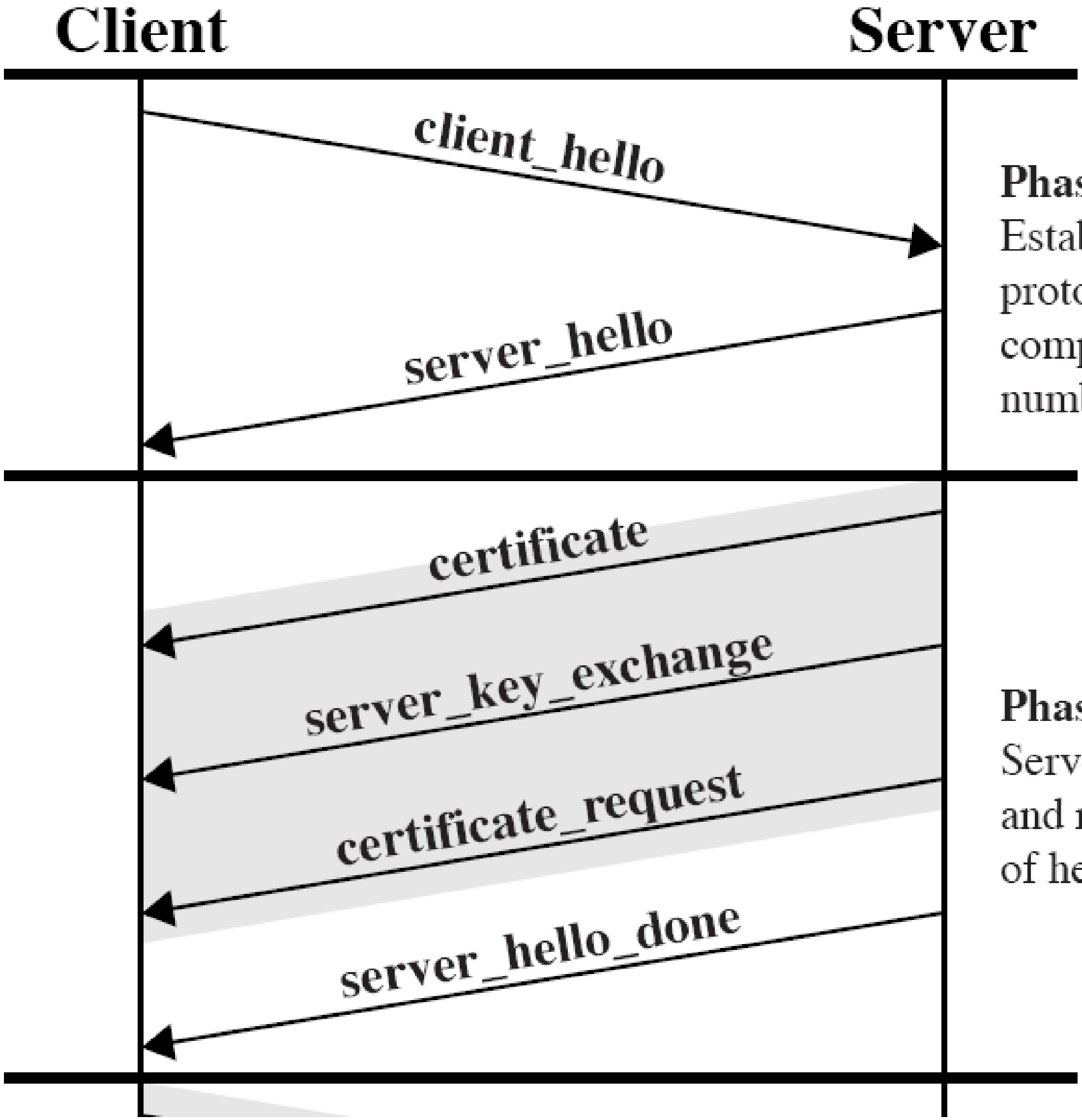
SSL Encrypt and MAC Algorithms

- Currently supported symmetric key algorithms for encryption:
  - Block ciphers
    - AES, IDEA, RC2, DES, 3DES, Fortezza
  - Stream ciphers
    - RC4
- MAC
  - MD5 or SHA1
  - TLS uses both MD5 and SHA1 and XORs the output together
    - Useful if one algorithm is considered insecure

SSL Handshake Protocol

- Performed before any data is sent. Four phases:
  - Create connection and agree upon security capabilities
    - Send HELLO messages between client and server
    - HELLO messages can contain list of encryption, MAC and compression algorithms implemented
  - Server Authentication and Key Exchange
    - X.509 certificates are exchanged
    - A server key may be exchanged depending on exchange protocol
    - Key exchange methods supported:
      - RSA, Diffie-Hellman, Fortezza
  - Client Authentication and Key Exchange
  - Finish
    - Use Change Cipher Protocol to change from key exchange cipher to encryption cipher
    - Send final finish message to check that encryption cipher is correct

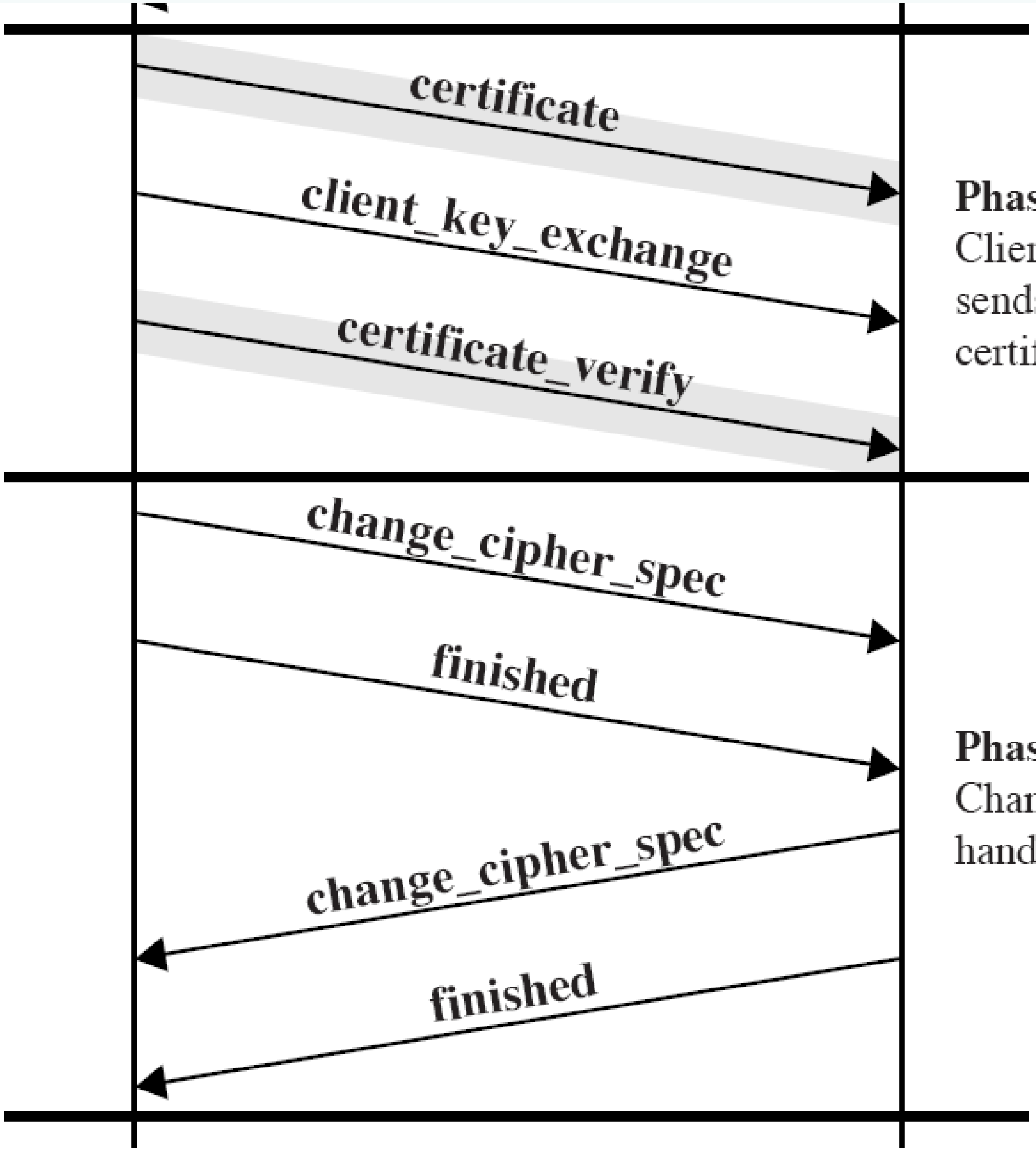
SSL Handshake Protocol



Phase 1
Establish security capabilities, including protocol version, session ID, cipher suite, compression method, and initial random numbers.

Phase 2
Server may send certificate, key exchange, and request certificate. Server signals end of hello message phase.

SSL Handshake Protocol



Phase 3
Client sends certificate if requested. Client sends key exchange. Client may send certificate verification.

Phase 4
Change cipher suite and finish handshake protocol.

SSL and HTTPS

- Secure web transfer is a key application of SSL
  - HTTPS is simple HTTP using SSL
- HTTP servers normally accept connections on port 80
- HTTPS servers accept connections on port 443
  - The Web Server that supports SSL (e.g. Apache) must be configured with a public key certificate
    - Certificate should be signed by a CA for public use
    - Web browser can the verify the certificate from its in-built list of CAs