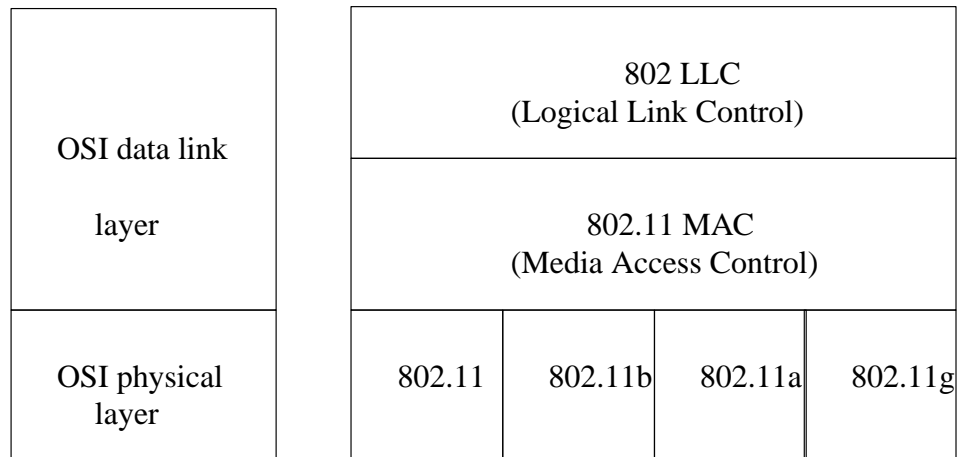


# Introduction to wireless security

- many topics to cover
- practicals will include use of Python to demonstrate theoretical concepts
- heavy use of secure shell in later weeks
- books
  - 802.11 Security, O'Reilly, by Bruce Potter and Bob Fleck
  - SSH The Secure Shell, The Definitive Guide, O'Reilly, by Daniel Barrett and Richard Silverman

# ISO OSI 7 Layer model and 802



## ISO OSI 7 Layer model and 802

■

Physical layer specifications			
802.11 PHY	Max Data Rate	Frequency	Modulation
802.11	2Mb/s	2.4GHz and IR	FHSS and DSSS
802.11b	11Mb/s	2.4GHz	DSSS
802.11a	22Mb/s	2.4GHz	OFDM
802.11g	54Mb/s	5GHz	OFDM
802.11n	74Mb/s	5GHz	unknown

- key
- MAC = media access control (CSMA/CD 802.3)  
(CSMA/CA 802.11)
  - LLC = logical link control (packet format)
  - FHSS = frequency hopping spread spectrum
  - DSSS = direct sequence spread spectrum
  - OFDM = orthogonal frequency division multiplexing

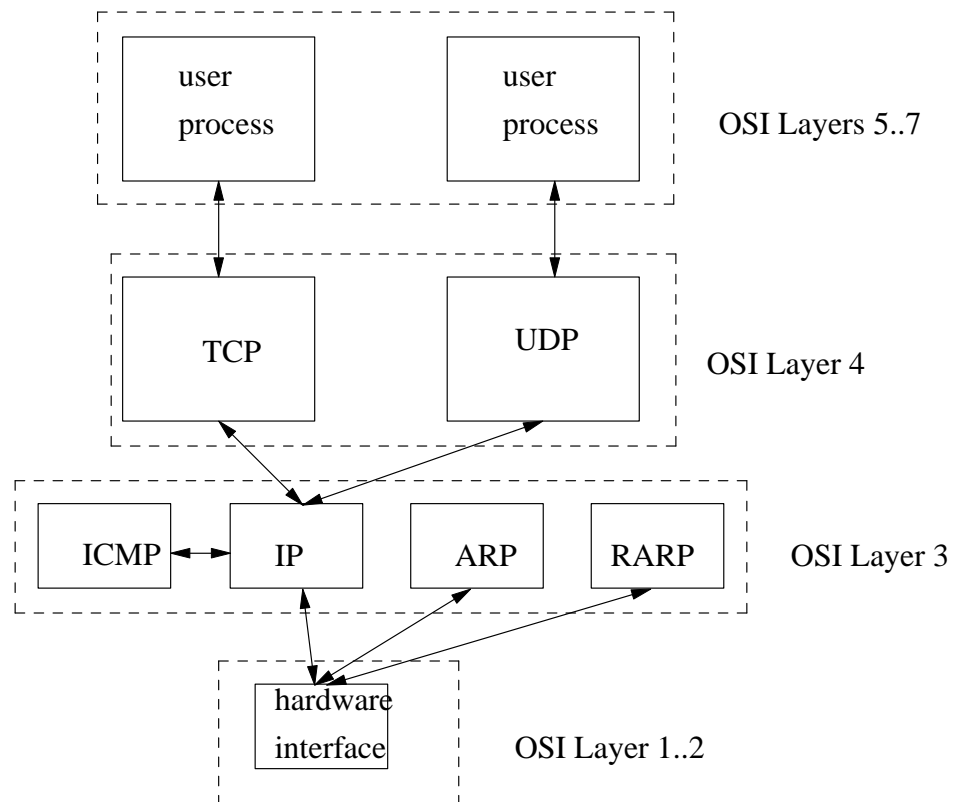


## ISO OSI 7 Layer model and 802

- 802.11n specification will be released 2009 (current throughput claims range from 72 Mb/s..300 Mb/s).

# TCP/IP Support Protocols

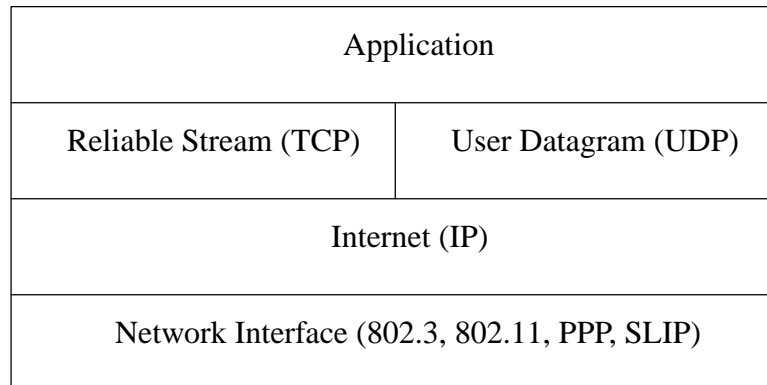
- are another reason TCP/IPs popularity



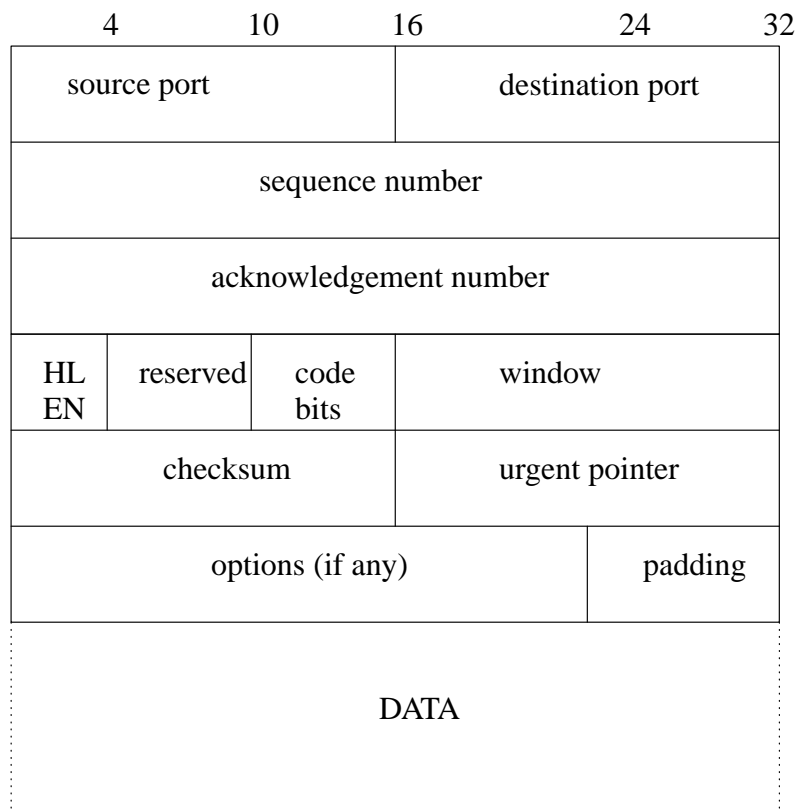
## Denial of service

- DOS attacks aim to prevent access to resources by traffic choking the network
  - may occur at any of the ISO OSI layers
- can also occur by physical removal of equipment ie disconnection
  - more commonly the term refers to traffic choking

# TCP/UDP model



# TCP Header format



## TCP Header format

- HLEN 4 bits
  - header length =  $n \times 32$

## TCP Header format



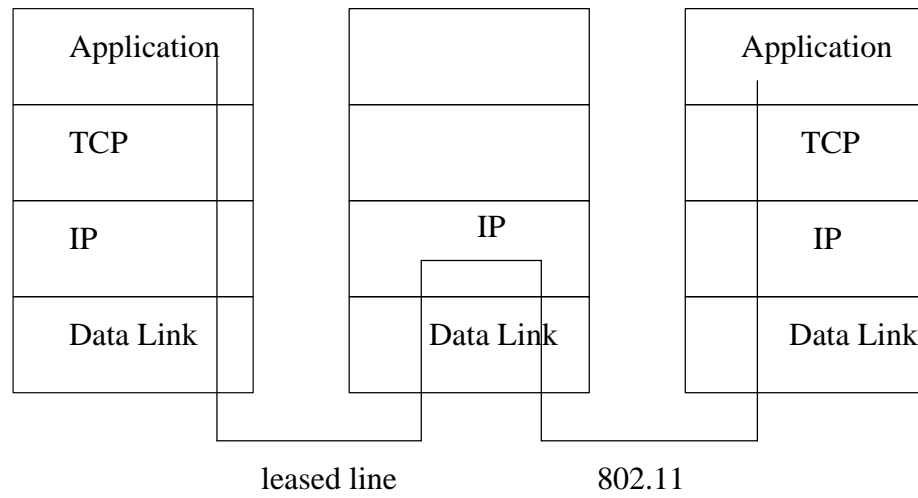
CODE field meaning	
URG	urgent pointer field valid
ACK	ack field valid
PSH	segment needs push
RST	reset connection
SYN	synchronize sequence no.s
FIN	sender reached end

# Services offered by TCP/IP



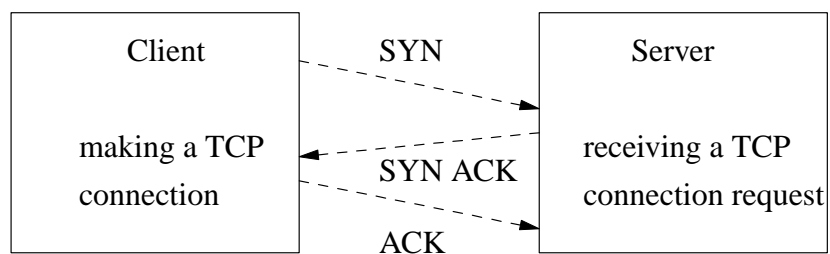
Layer	Protocols										
Process Application	NN TP	HT TP	PO P3	RC MDS	X rpc xdr	NFS N S	D N S	TF TP	SN NP	N T P	DH CP
Transport	TCP						UDP				
Internet	[ICMP, ARP, RARP]						IP [EGP, BGP, IGMP]				
Network	802.3		802.11			SLIP		PPP			

# TCP operation



# TCP Connection

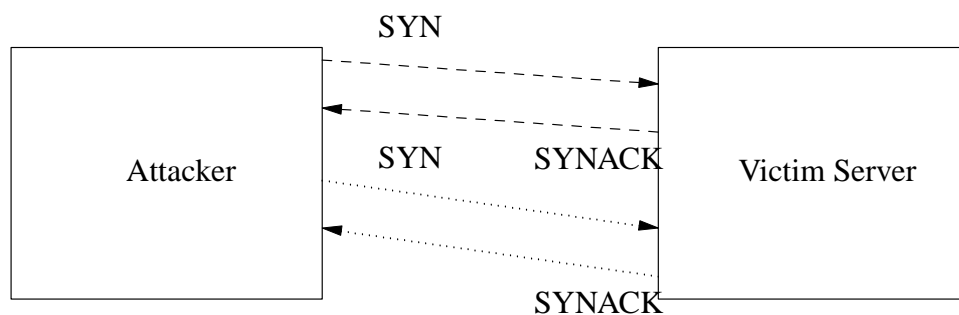
- TCP connection is via SYN (request connection)
  - server replies with SYN ACK
- client sends ACK and connection is established
- 



## TCP Denial of service: SYN Flooding

- a TCP implementation typically only allows 6..20 simultaneous connection establishments
  - not to be confused with simultaneous connections, which are normally measured in 1000's
  
- SYN flooding is where an attacker sends multiple SYNs to a victim, flooding the TCP establishment buffer
  - normally TCP implementations reset half open connections after several minutes
  - provides a valuable window of opportunity for attacker

## TCP Denial of service: SYN Flooding



- note the absence of an ACK from attacker
- victim does not know whether being attacked or message delay

## TCP Denial of service: Land attack

- attacker creates false TCP SYN packet
  - src address = victim.co.uk
  - src port number = dest port number
  - dest address = victim.co.uk
  
- attacker sends packet
  - and watches victim lock up or crash!
  
- solution
  - firewall must be set up disallow any packet with same src/dest addresses
  - inner router should be configured to only allow outgoing packets having source address of internal network

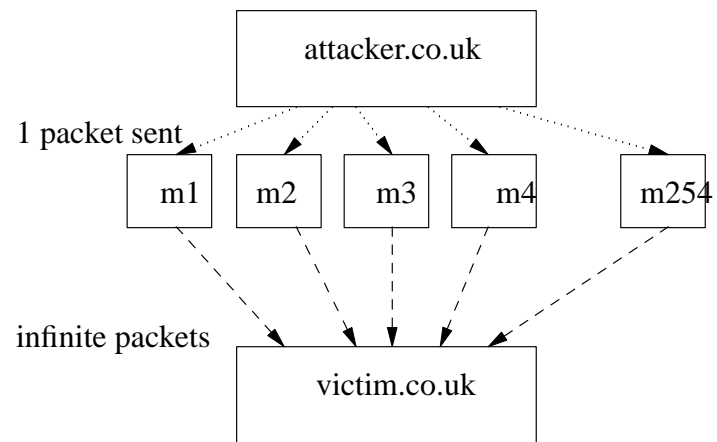
## SSL

- secure sockets layer is a security protocol that provides communications privacy across the Internet
  - independent of application protocol
  
- when connection establishment occurs
  - client and server exchange certificates
  
- both sides use certificates to encrypt and sign all information sent
  
- application protocols remain identical
  - however transport layer is encrypting and signing
  
- SSL is being superceded by TLS (Transport Layer Security)

## IP Denial of service: Smurf Attack

- attacker.co.uk sets
  - IP dest address as broadcast
  - IP src address as victim.co.uk
  - tcp port as chargen
    - chargen generates a continual ASCII alphabet
  
- solution disable chargen

# IP Denial of service: Smurf Attack



## Other variants of Smurf

- can you think of a variant?

## Other variants of Smurf

- obtain the sources to ping, and alter so that it continually sends ICMP packets, without any delay between transmission
- `dest = broadcast IP address`
- `src = victim.co.uk`

## Other variants of Smurf

- attacker creates false redirect ICMP packets with
  - dest = nowhere.man
  - src = broadcast.victim.co.uk