Sicurezza delle reti[1]

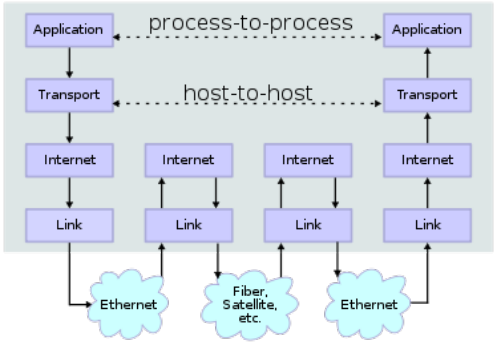
Mattia Monga

Dip. di Informatica e Comunicazione
Università degli Studi di Milano, Italia
mattia.monga@unimi.it

a.a. 2010/11

Lezione III: I protocolli di base

Il livello di trasporto



La comunicazione avviene fra processi attivi su nodi (distinti) della rete.

Porte

I numeri IP (rappresentativi di un nodo) non bastano per identificare una comunicazione. Uno scambio di dati fra due processi necessita di 4 numeri

$$\langle ip_1, n_1 : ip_2, n_2 \rangle$$

Port

I numeri n_1, n_2 (0–65536) che servono ad identificare la connessione si dicono porte, perché quelle del lato server devono essere note al client e rappresentano quindi il punto *d'accoglienza*.

Nota: il client è il nodo che inizia la connessione con il server.

Porte ben note

da http://www.iana.org/assignments/port-numbers

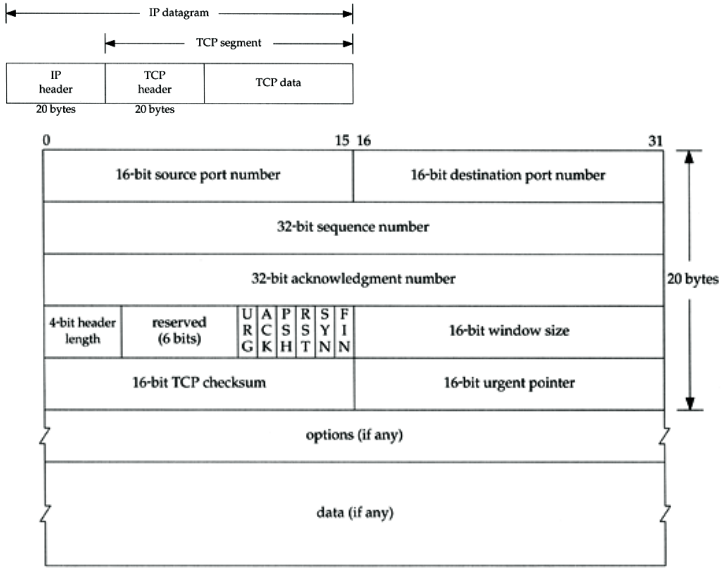
```
discard 9/tcp sink null
discard 9/udp sink null
ftp-data 20/tcp
ftp 21/tcp
ssh 22/tcp # SSH Remote Login Protocol
ssh 22/udp
telnet 23/tcp
smtp 25/tcp mail
domain 53/tcp # name-domain server
domain 53/udp
finger 79/tcp
www 80/tcp http # WorldWideWeb HTTP
pop3 110/tcp pop-3 # POP version 3
nntp 119/tcp readnews untp # USENET News Transfer Protocol
ntp 123/udp # Network Time Protocol
irc 194/tcp # Internet Relay Chat
https 443/tcp # http protocol over TLS/SSL
printer 515/tcp spooler # line printer spooler

# Non fissate da IANA
socks 1080/tcp # socks proxy server
openvpn 1194/tcp
openvpn 1194/udp
rmiregistry 1099/tcp # Java RMI Registry
# ...
```


Porte convenzionali

Ricordare sempre che le porte sono numeri convenzionali (concordate con IANA per i numeri ≤ 1024)

- in generale non identificano un servizio, ma la possibilità di stabilire una connessione.
- vietare che una connessione usi la porta destinazione 22 non significa vietare connessioni SSH, ma impedire che client e server possano accordarsi sull'uso della porta 22.
- il divieto può funzionare solo se l'amministratore controlla il server: se gestisce solo la rete il divieto è facilmente aggirabile.

TCP



Flag

SYN richiesta di connessione, sempre il primo pacchetto di una comunicazione

FIN : indica l'intenzione del mittente di terminare la sessione in maniera concordata

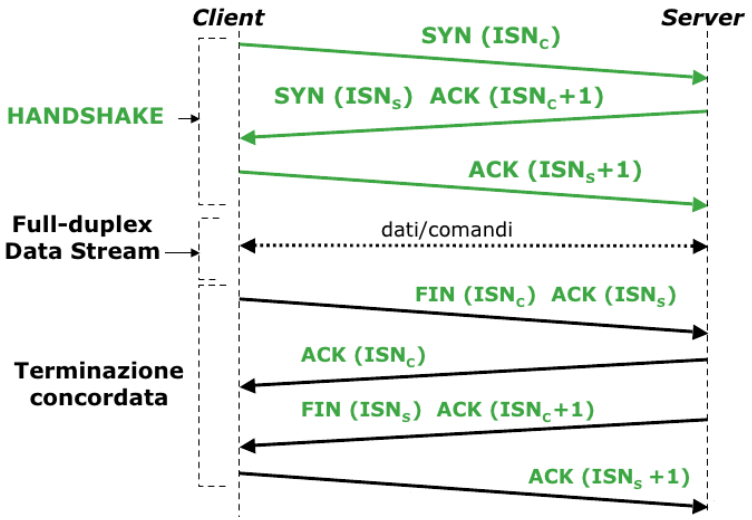
ACK conferma del pacchetto precedente, sia esso dati, SYN o FIN

RST : reset della sessione

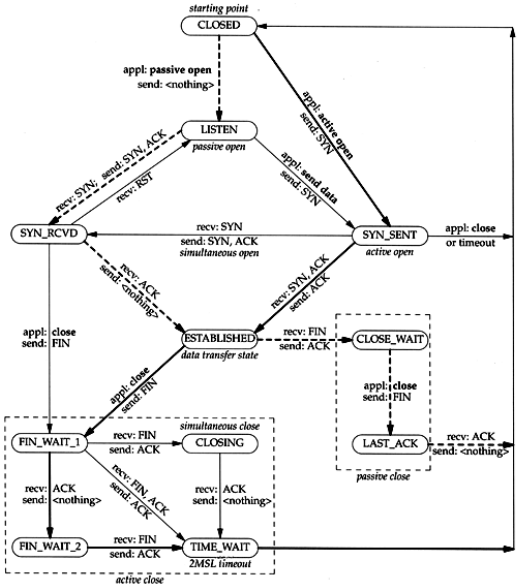
PSH : operazione di push, i dati vengono subito inviati al destinatario senza bufferizzarli

URG : dati urgenti (es. CTRL+C) vengono inviati con precedenza sugli altri

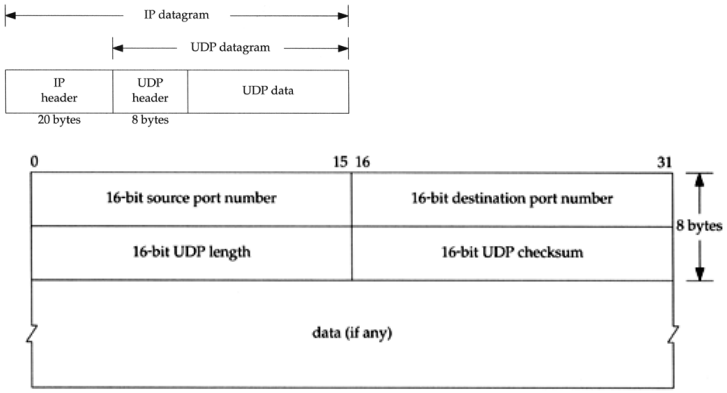
Sequence diagram



State diagram



UDP



Frammentazione

Nel protocollo IP (rete a pacchetto) è intrinseca la tecnica di frammentare un messaggio:

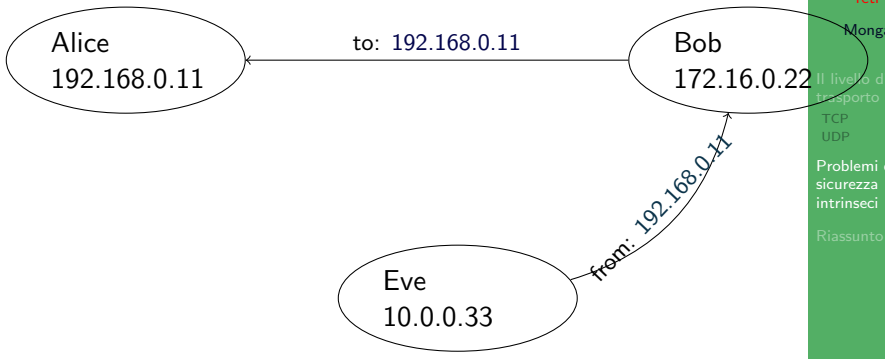
- Ogni "frammento" viaggia e *rischia* singolarmente
- Quali difese ha il "flusso" TCP?
  1. Il numero di sequenza TCP (ISN)
     - Deve essere imprevedibile
  2. Il checksum dei dati
     - Deve essere difficilmente falsificabile

"Spoofing"

Il numero IP nell'header può essere prodotto in maniera arbitraria, ma per entrare in una comunicazione serve l'ISN. I numeri di sequenza sono numeri non sempre ottenuti in maniera casuale:

- Negli anni '90 IRC kick-off wars: inserire un RST in una sessione IRC prodotta con uno stack
- Ora sono prodotti in maniera veramente casuale, ma resta la possibilità di esplorare lo spazio delle possibilità (32 bit $\mapsto$ 80GB di dati)

Spoofing



Questa versione è sufficiente per i Denial of Service, ma per partecipare ad una *conversazione* occorre saper predire i numeri di sequenza.

"Hijacking"

Nell'hijacking occorre ricostruire anche il checksum.

- Purtroppo è facile, perché è solo una somma modulo 16 bit
- Il checksum è pensato per proteggere da errori di trasmissione casuali, non da manomissioni volontarie

Fingerprinting

Dall'esame (non intrusivo) dei pacchetti di rete è possibile identificare molti dettagli utili negli attacchi...

- p.es. `p0f` è in grado di riconoscere molte implementazioni di stack TCP/IP
- è possibile studiare la topologia della rete esaminando il TTL
  - p.es. Windows TTL=128, Linux TTL=64
  - TTL==80 $\Rightarrow$ Windows, e il nodo è distante 48 hop
- gli header dei pacchetti rivelano molte informazioni ai potenziali attaccanti

Riassunto dei punti fondamentali

- Le porte identificano una connessione, non un servizio
- La frammentazione dei messaggi comporta la possibilità di
  - *spoofing*
  - *hijacking*