Capítulo 1: Introducción - II

ELO322: Redes de Computadores

Agustín J. González

Este material está basado en:

- Material de apoyo al texto *Computer Networking: A Top Down Approach Featuring the Internet* Jim Kurose, Keith Ross

Introducción

1.1 ¿Qué es la Internet?

1.2 Red periférica

1.3 Red central (core)

1.4 Retardos, pérdidas, eficiencia (throughput) en redes.

1.5 Capas de protocolos, Modelo de servicio

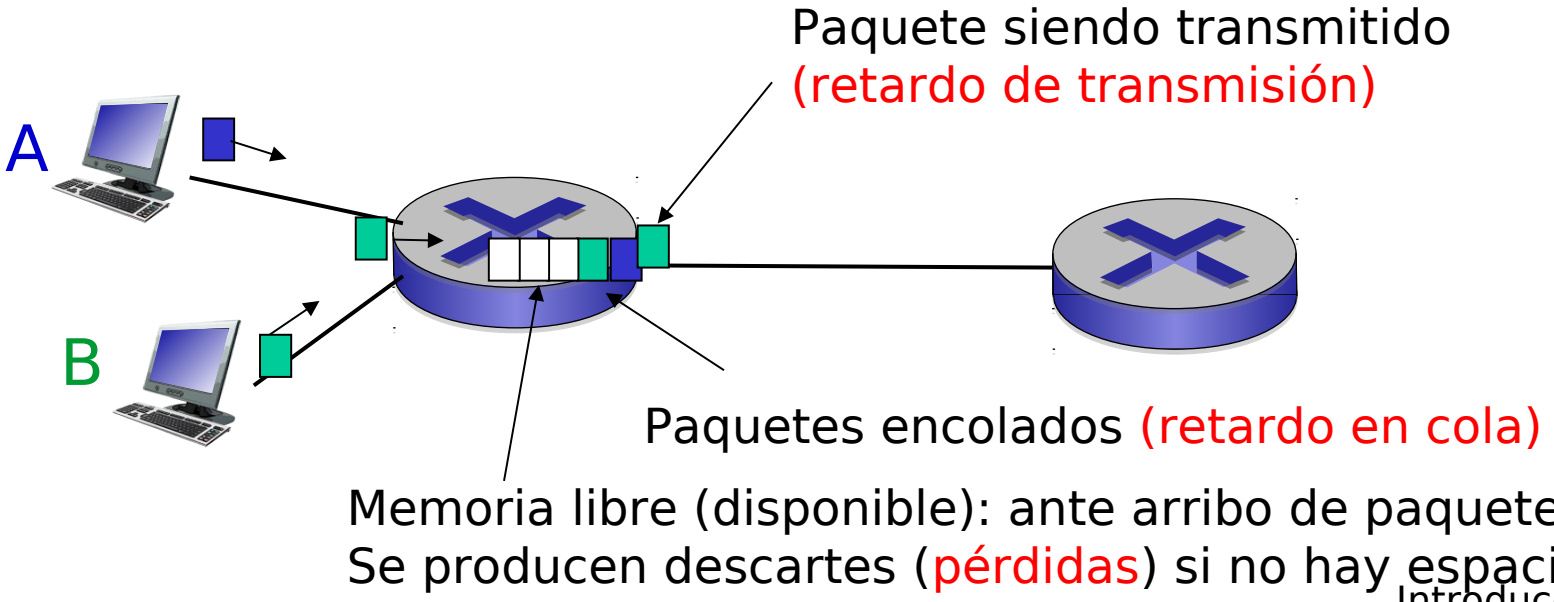
1.6 La red bajo ataque: seguridad

1.7 Historia (lectura personal)

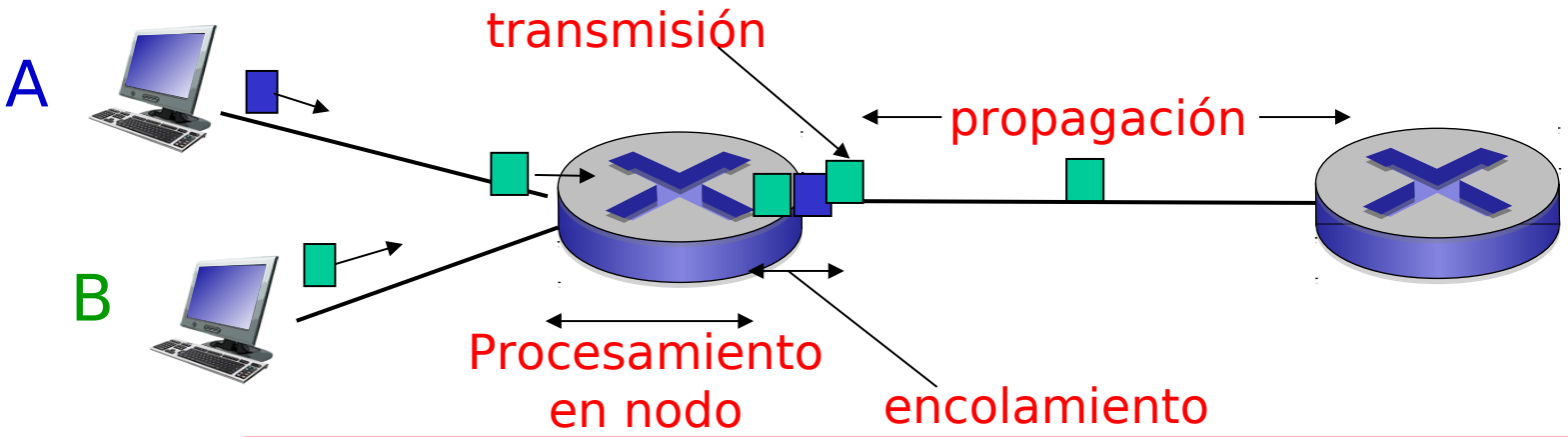
¿Cómo ocurren las pérdidas y retardos?

Los paquetes son *encolados* en la memoria (buffer) de cada router

- Tasa de arribo de paquetes puede exceder la capacidad de salida del enlace
- Los paquetes son encolados, y esperan por su turno



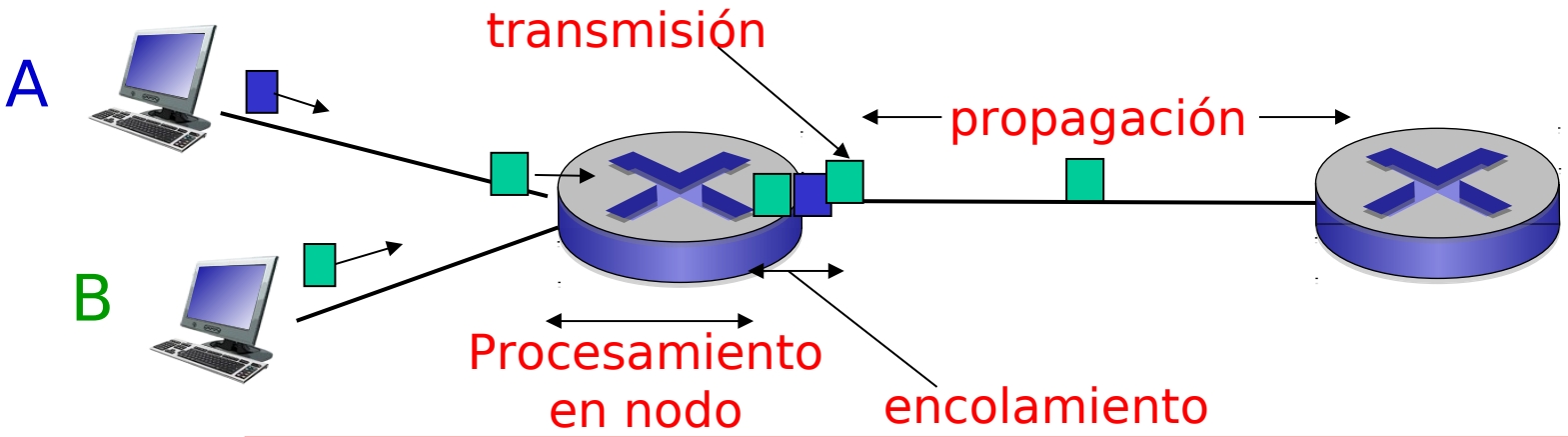
Cuatro fuentes de retardo de paquetes



$$d_{\text{nodo-nodo}} = d_{\text{proc}} + d_{\text{cola}} + d_{\text{trans}} + d_{\text{prop}}$$

- 1. d_{proc} Retardo de procesamiento en el nodo:
  - Chequeo de bits de error (como el dígito verificador del RUT)
  - Determinar el enlace de salida
  - Típicamente < 1 [ms]
- 2. d_{cola} Retardo en cola
  - Tiempo esperado en la cola para que los paquetes anteriores sean transmitidos
  - Depende del nivel de congestión del router

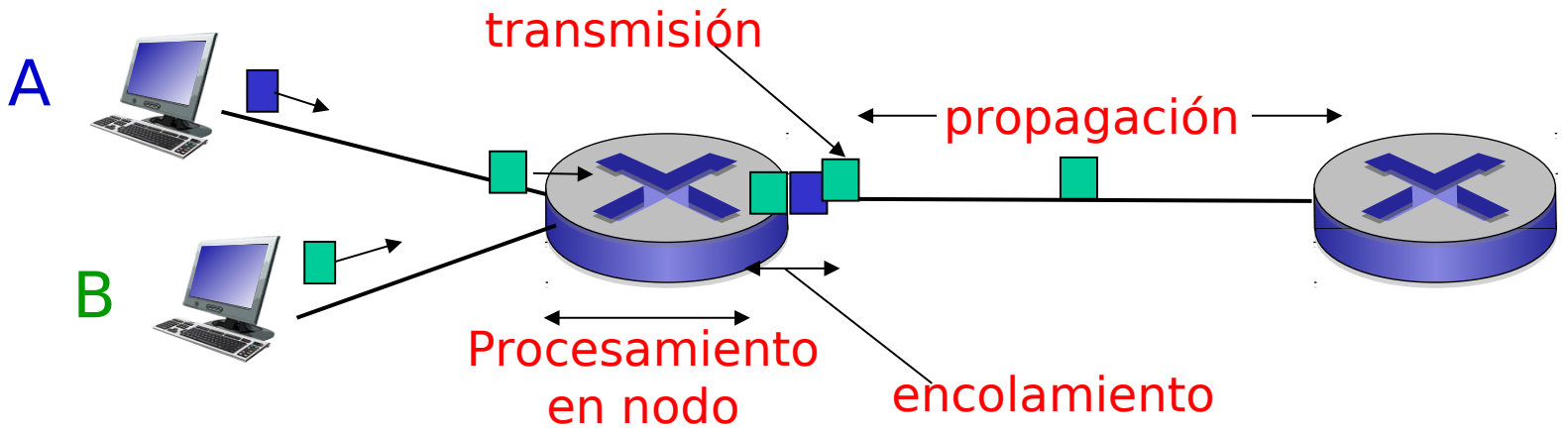
Cuatro fuentes de retardo de paquetes

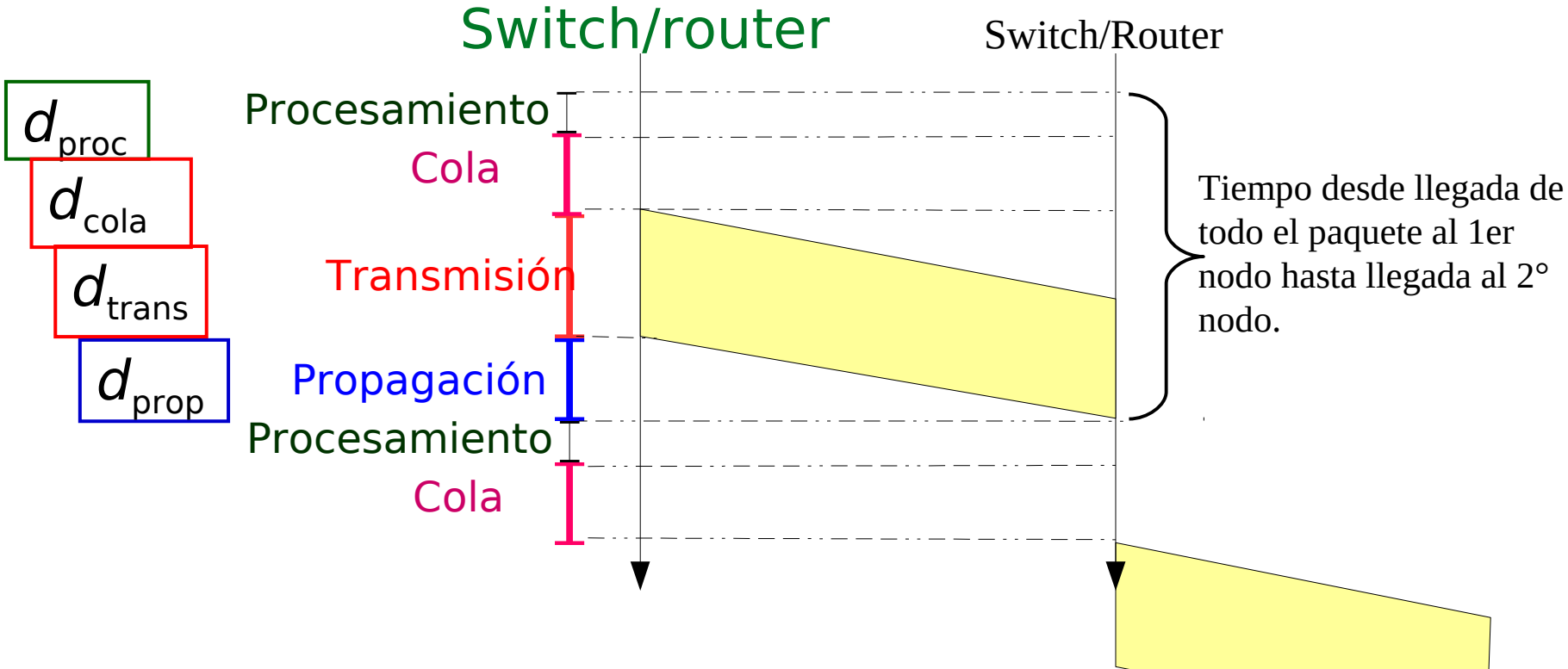


$$d_{\text{nodo-nodo}} = d_{\text{proc}} + d_{\text{cola}} + d_{\text{trans}} + d_{\text{prop}}$$

- 3. d_{trans} Retardo de transmisión:
  - R=tasa de bits del enlace (bps)
  - L=largo del paquete (bits)
  - Tiempo de envío = L/R
- 4. d_{prop} Retardo de propagación:
  - d = largo del enlace físico
  - v = rapidez de propagación en medio (~2×10^8 m/s)
  - Retardo de propagación = d/v

Retardo nodo a nodo de una trama



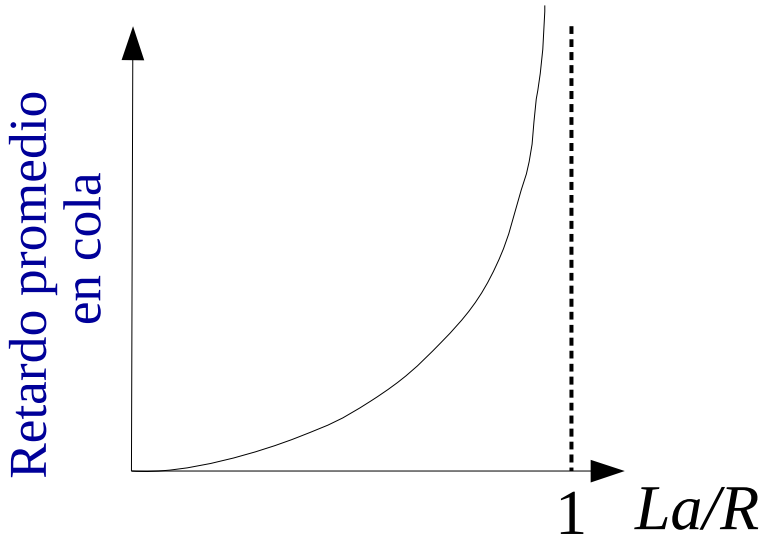


Retardo desde llegada a un nodo a llegada al otro

$$d_{\text{nodo\_nodos}} = d_{\text{proc}} + d_{\text{cola}} + d_{\text{trans}} + d_{\text{prop}}$$

- d_{proc} = retardo de procesamiento
  - Típicamente unos pocos microsegundos o menos
- d_{cola} = retardo de espera en cola(s)
  - Depende de la congestión (tráfico en nodo)
- d_{trans} = retardo de transmisión
  - = L/R, significativo en enlaces de baja tasa en bps
- d_{prop} = retardo de propagación
  - De pocos microsegundos a cientos de milisegundos

Retardo de encolamiento (revisitado)



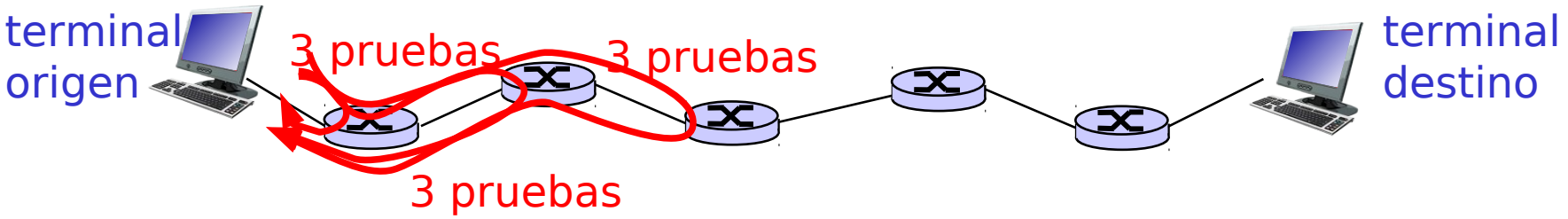
- R=”bandwidth” del enlace de salida [bit/s]
- L=largo del paquete [bit], supondremos cte.
- a=tasa promedio de **a**rribo de paquetes [paquetes/s]
- **L*a**=n° bits/s de entrada
- Pregunta: ¿Qué pasa con diferentes valores de L*a/R?

- L*a/R ~ 0: => pequeño retardo de encolamiento
- L*a/R tiende a 1: retardo se hace grande
- L*a/R > 1: llega más “trabajo” que el posible de servir, retardo promedio tiende a infinito!

- La carretera tiene “buffer” infinito, un router no.

Retardo “Real” en Internet y rutas

- ¿Cuáles son los retardos reales en Internet y las rutas de los paquetes?
- **Programa traceroute**: entrega medidas del retardo de ida y vuelta desde el terminal de origen hacia routers en la ruta al destino en Internet. (en windows tracert como en trace route)
- Para cada distancia de *i* enlaces:
  - manda tres paquetes que van a llegar al router *i* en la ruta hacia el destino
  - router *i* devuelve paquetes de información al terminal origen
  - terminal de origen mide el intervalo entre transmisión y respuesta.



Retardo “Real” en Internet y rutas

En windows usar > tracert www.google.cl

- Probar: traceroute www.google.cl

agustin@pcagv:~$ traceroute www.google.cl /* la salida varía con el tiempo*/

traceroute to www.google.cl (64.233.163.104), 30 hops max, 60 byte packets

1 elo-gw.elo.utfsm.cl (200.1.17.1) 0.479 ms 0.938 ms 1.123 ms

2 telmex-gw.usm.cl (200.1.20.131) 2.286 ms 2.355 ms 2.343 ms

3 border-gw.usm.cl (200.1.20.130) 2.302 ms 2.331 ms 2.319 ms

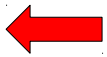
4 ge-1-1-0.452.ar1.SCL1.gblx.net (208.178.62.9) 5.300 ms 5.357 ms 5.476 ms

5 te4-3-10G.ar3.SCL1.gblx.net (67.16.130.78) 5.319 ms 7.266 ms 7.404 ms

6 72.14.216.105 (72.14.216.105) 7.308 ms 5.997 ms 5.942 ms

7 209.85.240.138 (209.85.240.138) 5.989 ms 5.120 ms 6.961 ms

8 72.14.238.48 (72.14.238.48) 53.155 ms 72.14.233.134 (72.14.233.134) 51.959 ms 51.948 ms



9 72.14.233.91 (72.14.233.91) 52.973 ms 72.14.233.95 (72.14.233.95) 51.146 ms 52.047 ms

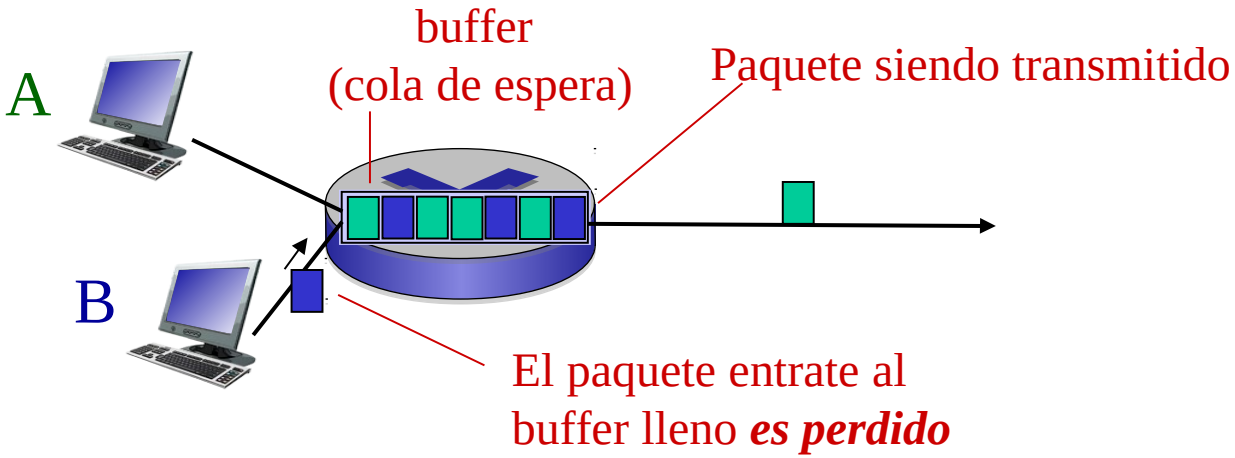
10 64.233.175.62 (64.233.175.62) 55.207 ms 55.211 ms 56.045 ms

11 bs-in-f104.1e100.net (64.233.163.104) 51.918 ms 51.869 ms 54.939 ms

* Corra traceroute desde paises exóticos vía www.traceroute.org

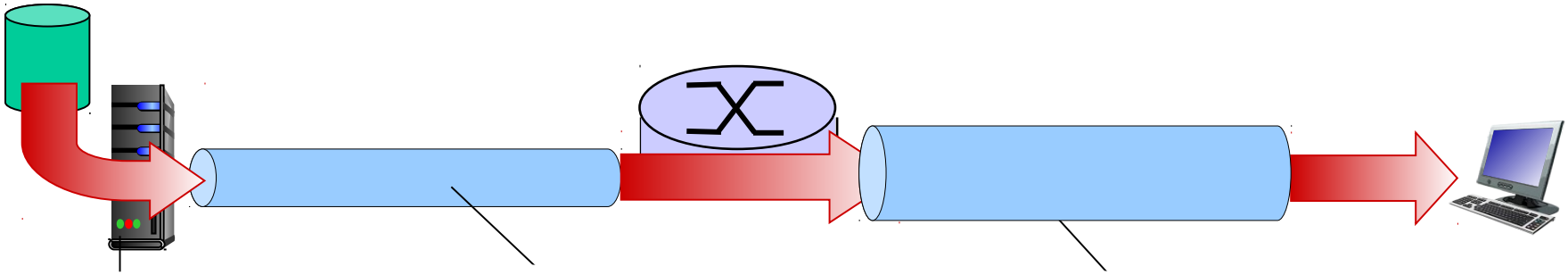
Pérdida de paquetes

- El buffer de la cola en conmutadores (routers, switches, etc) tiene capacidad finita
- Cuando un paquete llega a una cola llena, el paquete es descartado (pérdida)
- Paquetes perdidos pueden ser retransmitidos por nodo previo, por origen o ignorados.



Throughput (“rendimiento”)

- throughput: tasa (bits/tiempo) a la cual bits son transferidos desde capa transmisora a receptora
- instantáneo: tasa en un punto dado del tiempo
- promedio: tasa sobre largos periodos
- Cuello de botella: quien (enlace, equipo, etc) limita el throughput extremo a extremo



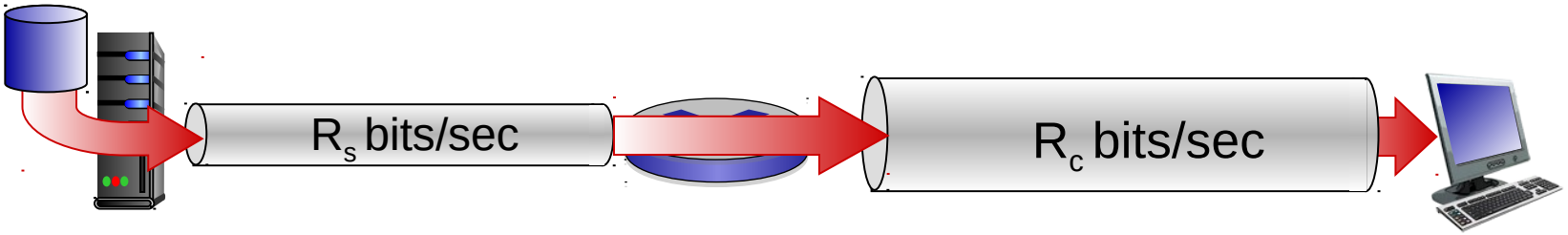
Tx envía bits por el enlace

Enlace que puede llevar bits a tasa R_s bits/sec

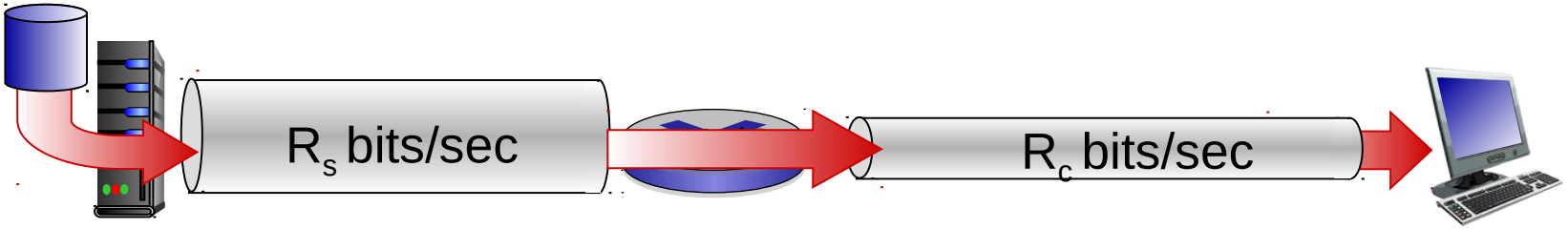
Enlace que puede llevar bits a tasa R_c bits/sec

Throughput (más)

- $R_s < R_c$ ¿Cuál es el rendimiento promedio?



- $R_s > R_c$ ¿Cuál es el rendimiento promedio?

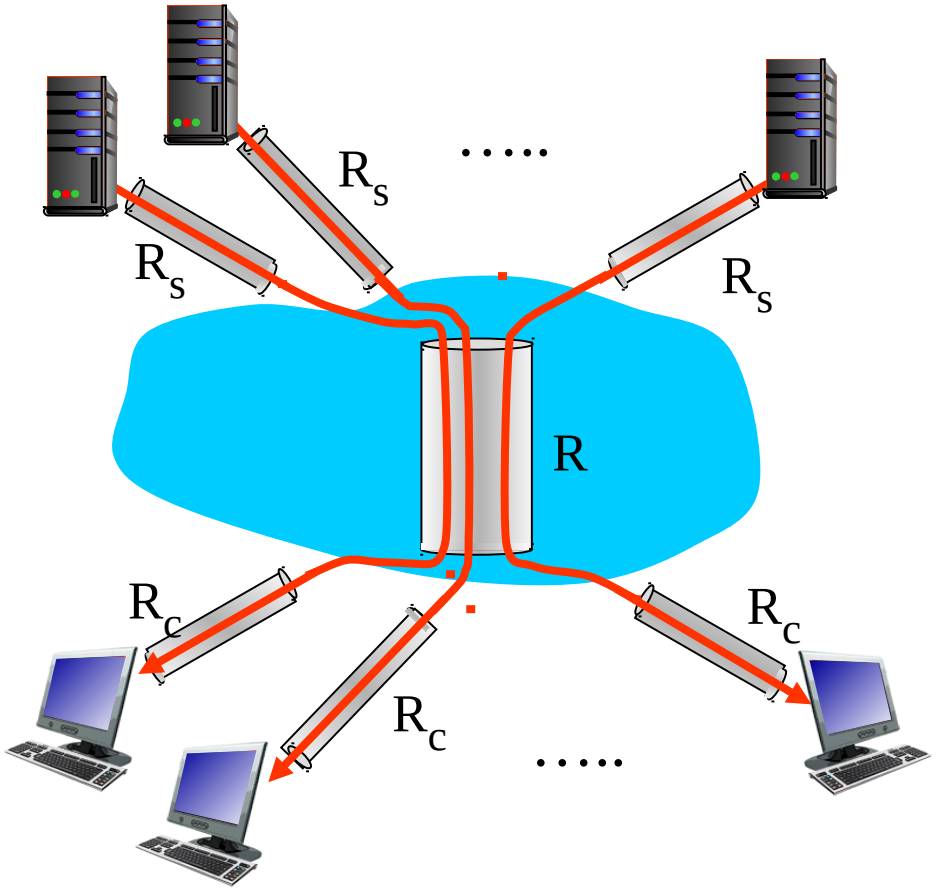


Enlace cuello de botella

Enlace en la ruta extremo a extremo que limita el rendimiento total.

Throughput: escenario Internet

- Throughput extremo a extremo por conexión: *min(*R_c*,*R_s*,R/10)*
- En la práctica: R_c o R_s es a menudo el cuello de botella



10 conexiones (equitativas) comparten enlace backbone de *R* bits/sec

* Ver ejercicios interactivos online en: http://gaia.cs.umass.edu/kurose_ross/interactive/

Introducción

1.1 ¿Qué es la Internet?

1.2 Red periférica

1.3 Red central (core)

1.4 Retardos, pérdidas, eficiencia (throughput) en redes.

1.5 Capas de protocolos, Modelos de servicio

1.6 La red bajo ataque: seguridad

1.7 Historia (lectura personal)

“Capas” de Protocolos

Las redes son complejas!

- Muchos “componentes”:
  - hosts
  - routers
  - enlaces de varios tipos (cable, RF, óptico)
  - aplicaciones
  - protocolos
  - hardware, software

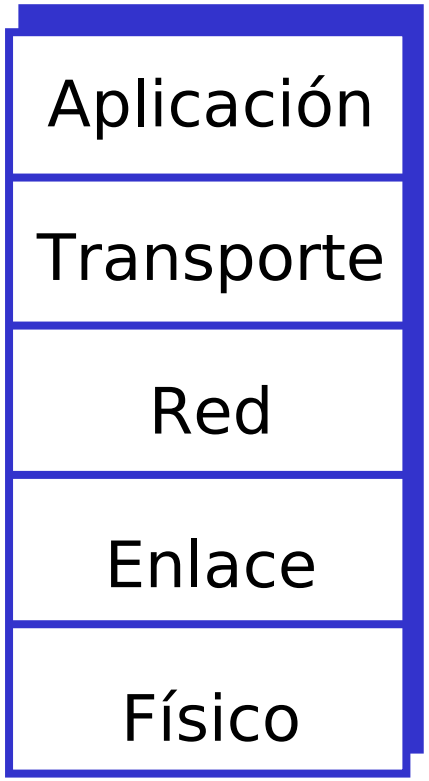
Pregunta:

Hay alguna esperanza de *organizar* la estructura de la red?

O al menos nuestra discusión de la red?

Pila de protocolos en Internet (protocol stack) – modelo TCP/IP

- aplicación: compuesto por las aplicaciones de red (todas las otras capas trabajan para ésta)
  - SSH, SMTP, HTTP, Skype, etc
- transporte: transferencia de datos host-host para una aplicación específica
  - **TCP, UDP**, SCTP (2000), DCCP (2006)
- red: ruteo de datagramas desde host fuente a destino
  - IP, protocolos de ruteo
- enlace: transferencia de datos entre elementos vecinos en la red
  - PPP, Ethernet, Wifi
- físico: transporte de bits “en el cable”
- El modelo OSI (Open System Interconnection) incluye capas de Presentación y Sesión adicionales no incluidos en el modelo TCP/IP



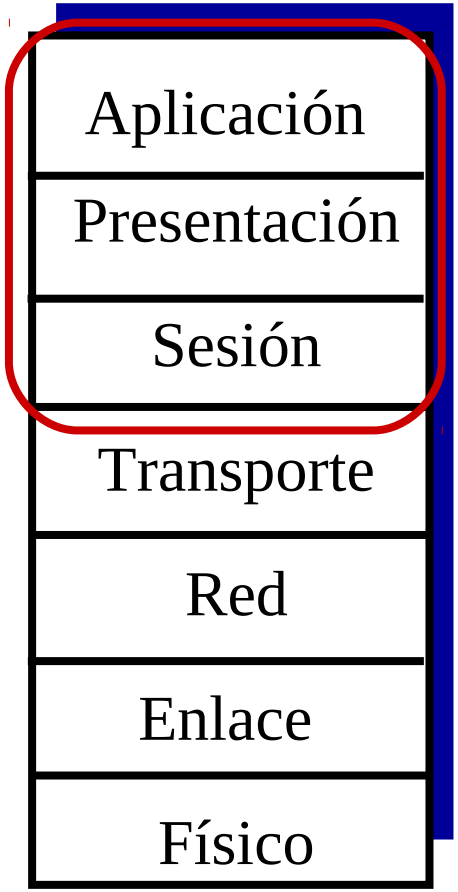
¿Por qué usar capas?

Nos enfrentamos a sistemas complejos:

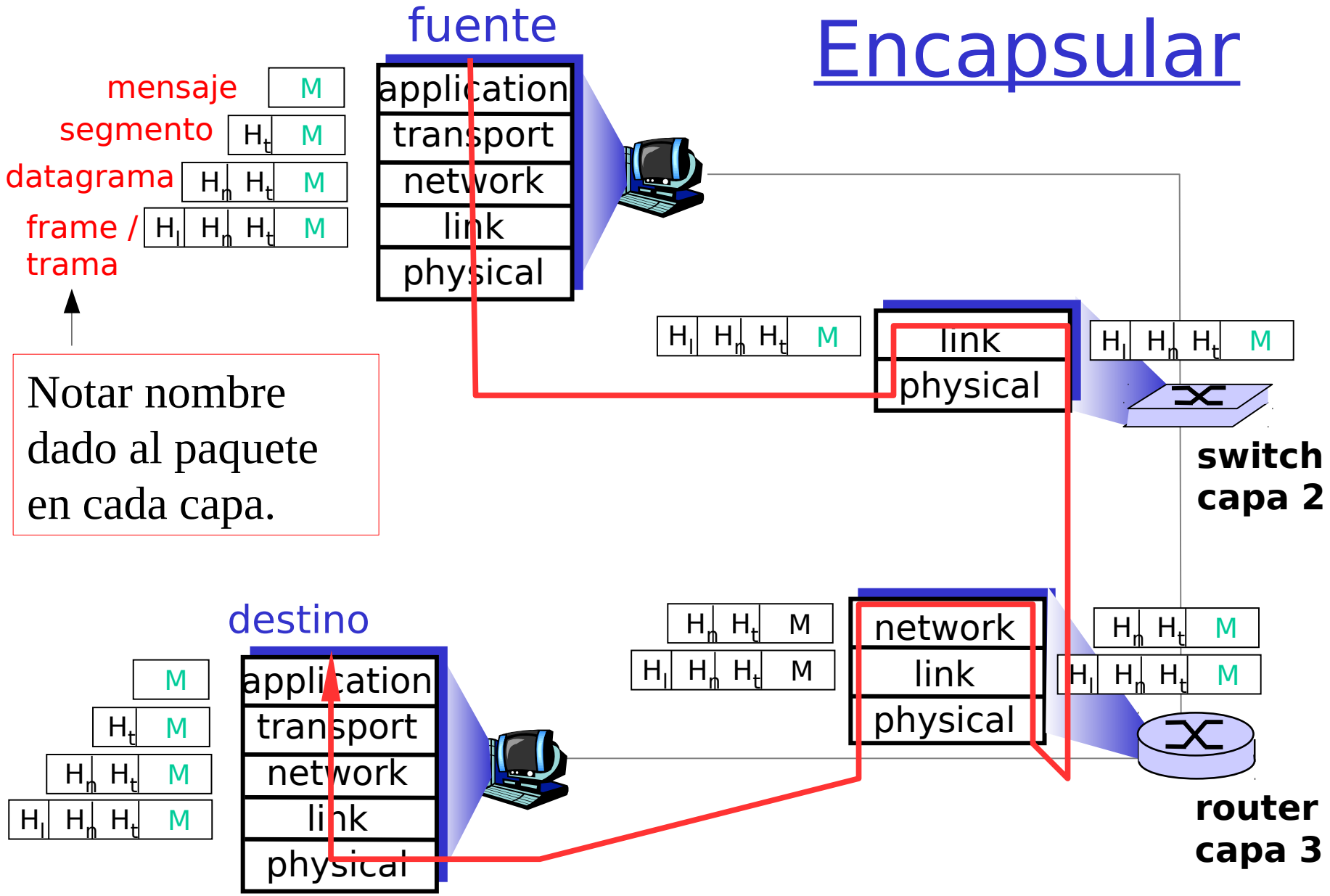
- Estructura explícita permite identificación y relación de la partes complejas del sistema
  - modelo de referencia de capas para análisis y discusión
- Modularización facilita mantención, actualización del sistema
  - Un cambio en la implementación de la capa de servicio es transparente al resto del sistema
  - e.g., cambio en capa de enlace no afecta al resto, las aplicaciones no notan aparición de 4G

Modelo de Referencia ISO/OSI

- *presentación*: Permite a las aplicaciones interpretar los datos, e.g., encriptación, compresión, convenciones específicas de una máquina (orden de bits)
- *sesión*: sincronización, puntos de chequeo, recuperación de caídas de conexión
- La pila de Internet "omite" estas capas!
  - Estos servicios, si son necesarios, deben ser implementados en la aplicación.



Encapsular


fuente
mensaje
M
segmento
H_t M
datagrama
H_n H_t M
frame / trama
H_l H_n H_t M
application
transport
network
link
physical
Notar nombre dado al paquete en cada capa.
H_l H_n H_t M
link
physical
H_l H_n H_t M
switch capa 2
destino
M
H_t M
H_n H_t M
H_l H_n H_t M
application
transport
network
link
physical
H_n H_t M
H_l H_n H_t M
network
link
physical
H_n H_t M
H_l H_n H_t M
router capa 3

Introducción

1.1 ¿Qué es la Internet?

1.2 Red periférica

1.3 Red central (core)

1.4 Retardos, pérdidas, eficiencia (throughput) en redes.

1.5 Capas de protocolos, Modelos de servicio

1.6 La red bajo ataque: seguridad

1.7 Historia (lectura personal)

Seguridad de la Red

- Campo de la seguridad:
  - Cómo gente mala pueden atacar la red
  - Cómo podemos defender la red de los ataques
  - Cómo diseñar arquitecturas que son inmunes a ataques
- La Internet no fue originalmente diseñada con (mucha) seguridad en mente
  - *Visión original:* “un grupo de usuarios mutuamente confiables conectados a una red transparente” ☺
  - Diseñadores de protocolos Internet juegan a “ponerse al día”
  - Se requiere consideraciones de seguridad en todos los niveles!

Gente mala: pone *malware* en hosts vía Internet

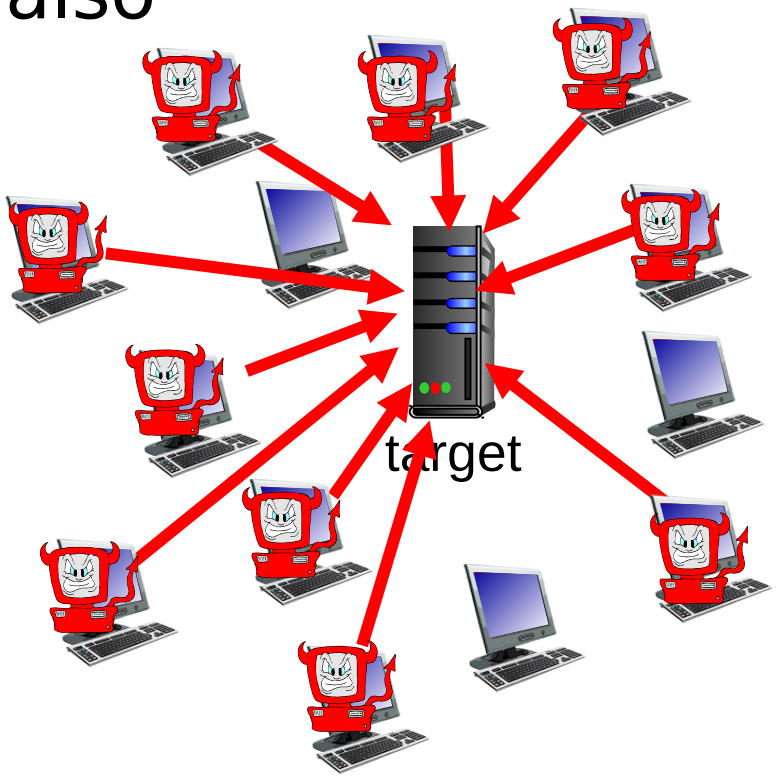
- Malware (soft**ware mal**icioso) llega en forma de:
  - *virus:* infección auto-replicable por medio de la recepción y ejecución de objetos (e.g. adjuntos a email). Requieren participación del usuario para infectar.
  - *worm* infección auto-replicable a través de la recepción pasiva (usuario no interviene) de objetos que se ejecutan a sí mismos. Aprovechan hoyos de seguridad de algunas aplicaciones.
- spyware malware puede grabar lo digitado, sitios webs visitados, y sube información a sitio recolector
- Host infectados pueden ser enrolados en botnet, usados por spam y ataques de denegación de servicios distribuidos.

Bot: programa que hace tareas automáticamente (viene de robot)

Gente mala: ataca servidores, infraestructura de red

Denegación de Servicio (DoS): atacantes dejan recursos (servidor, bandwidth) no disponible a tráfico legítimo inundando el recurso con tráfico falso

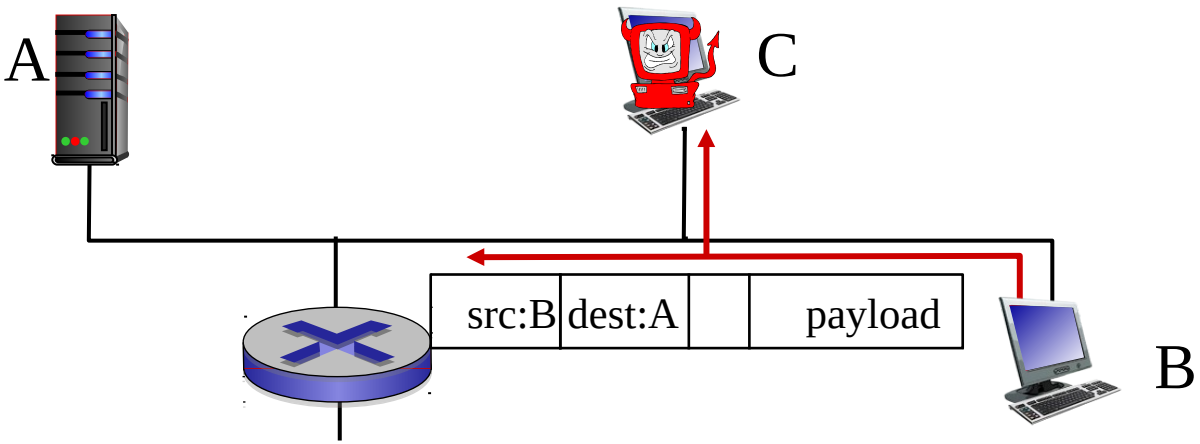
1. selecciona objetivo
2. infecta hosts en la red (ver botnet)
3. envía paquetes al objetivo desde los hosts infectados



Gente mala puede husmear paquetes

"sniffing" ("olfateo") de paquetes:

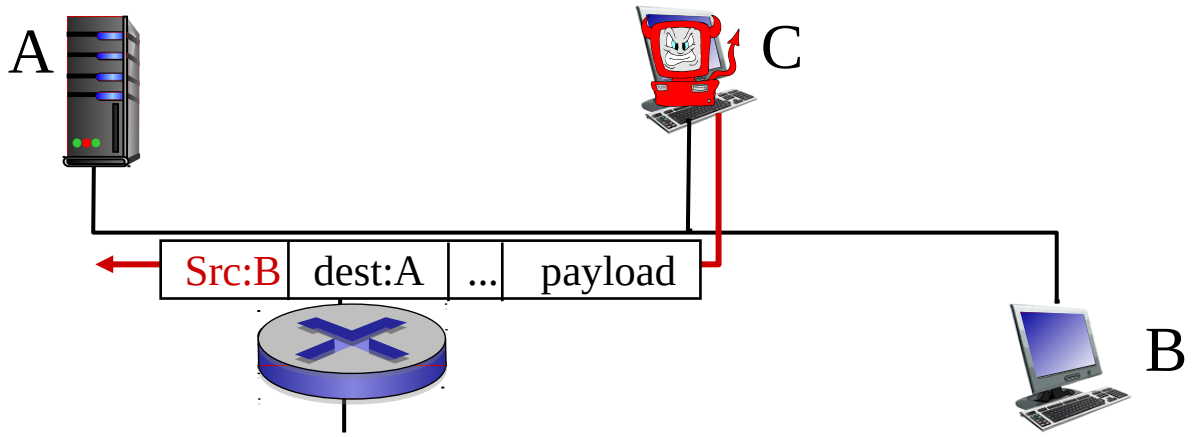
- Es fácil si el medio es de difusión, es decir compartido (en Ethernet compartida, wireless)
- Interfaces de red promiscuas leen y graban todos los paquetes (e.g., incluyendo passwords!)



- Software wireshark, a utilizar en tareas, es un packet-sniffer gratuito

Gente mala puede usar direcciones falsa

IP spoofing (suplantación de IP): envío de paquetes con dirección fuente falsa



... hay mucho más en seguridad (no todo en este curso)

Introducción

1.1 ¿Qué es la Internet?

1.2 Red periférica

1.3 Red central (core)

1.4 Retardos, pérdidas, eficiencia (throughput) en redes.

1.5 Capas de protocolos, Modelos de servicio

1.6 La red bajo ataque: seguridad

1.7 Historia (lectura personal) 

Historia de Internet:

ver también:
http://www.zakon.org/robert/internet/timeline/
http://www.computerhistory.org/internet_history/

1961-1972: Principios sobre packet-switching

- 1961: Leonard Kleinrock – Teoría de colas muestra efectividad de packet-switching
- 1964: Baran - packet-switching en redes militares
- 1967: ARPAnet concebida por Advanced Research Projects Agency
- 1969: primer nodo ARPAnet operacional usando IMP (Internet Message Processor)

- 1972:
  - ARPAnet demostrado públicamente
  - NCP (Network Control Protocol) primer protocolo host-host => TCP
  - 1° programa e-mail (Tomlinson)
  - ARPAnet tiene 15 nodos

Historia de Internet

1972-1980: Redes de comp., nuevas y propietarias

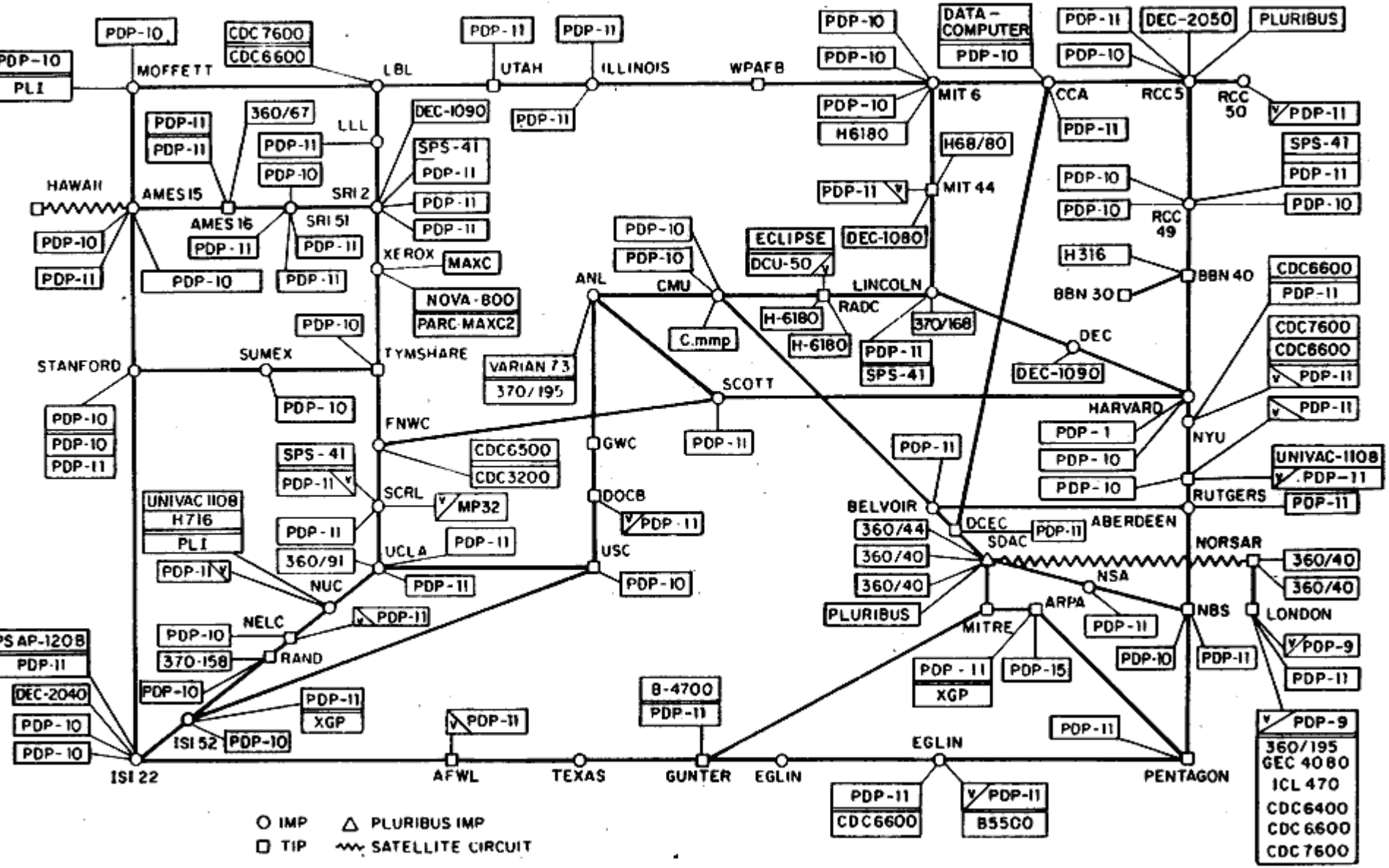
- 1970: ALOHAnet red satelital en Hawaii
- 1974: Cerf and Kahn – Arquitectura para interconectar redes
- late70's: arquitecturas propietarias: DECnet, SNA, XNA
- late 70's: Conmutación de paquetes de largo fijo (ATM precursor)
- 1979: ARPAnet tiene 200 nodos

Principios de redes de Cerf y Kahn :

- minimalismo, autonomía - no requiere cambios internos para interconectar redes
- Modelo de servicio de mejor esfuerzo (best effort service)
- Routers sin estado
- Control descentralizado

define la arquitectura actual de Internet: "KISS"

ARPANET LOGICAL MAP, MARCH 1977



(PLEASE NOTE THAT WHILE THIS MAP SHOWS THE HOST POPULATION OF THE NETWORK ACCORDING TO THE BEST INFORMATION OBTAINABLE, NO CLAIM CAN BE MADE FOR ITS ACCURACY)

NAMES SHOWN ARE IMP NAMES, NOT (NECESSARILY) HOST NAMES

Historia de Internet

1980-1990: nuevos protocolos, proliferación de redes

- 1983: establecimiento de TCP/IP
- 1982: smtp, se define el protocolo de correo (e-mail)
- 1983: Se define DNS para traducir nombres a direcciones IP
- 1985: Se define el protocolo ftp
- 1988: se define el mecanismo de control de congestión en TCP

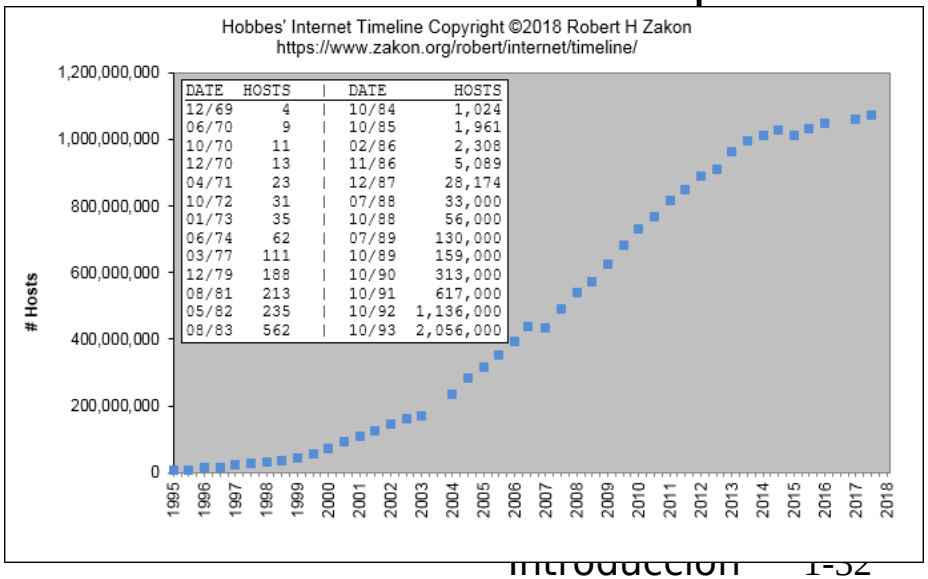
Historia de Internet

1990, 2000's: comercialización, la Web, nuevas apps

- Inicios 1990's: ARPAnet dejó de operar
- 1991: NSF levantó restricciones para uso comercial del NSFnet (ésta cesó, 1995)
- Inicios 1990s: Web
  - hypertext [Bush 1945, Nelson 1960's]
  - HTML, HTTP: Berners-Lee
  - 1994: Mosaic, luego Netscape
  - Finales de 1990's: comercialización de la Web

Finales 1990's – 2000's:

- Más killer apps: mensajería instantánea, P2P compartición de archivos
- Seguridad en redes
- 50 millones de hosts, 100 millones+ usuarios
- Backbone corre a Gbps



DATE	HOSTS	DATE	HOSTS
12/69	4	10/84	1,024
06/70	9	10/85	1,961
10/70	11	02/86	2,308
12/70	13	11/86	5,089
04/71	23	12/87	28,174
10/72	31	07/88	33,000
01/73	35	10/88	56,000
06/74	62	07/89	130,000
03/77	111	10/89	159,000
12/79	188	10/90	313,000
08/81	213	10/91	617,000
05/82	235	10/92	1,136,000
08/83	562	10/93	2,056,000

Historia de Internet

2005 en adelante:

- ~5 billones (5.000 millones) de hosts
  - Smartphones y tables
- Aumenta ubicuidad de acceso de alta velocidad
- Redes sociales: Facebook ~ 1 billón usuarios
- Google y Microsoft crean sus propias redes
- Comercio electrónico, universidades, empresas corren servicios en la “nube” (e.g. Amazon, E-comerce, YouTube, gaming, Twitter, Redes sociales (linkedin, Facebook)
- wireless, movilidad

Introducción: Resumen

- Vista global de Internet
- ¿Qué es un protocolo?
- Periferia de la red, su núcleo, y redes de acceso
  - Conmutación de paquetes versus conmutación de circuitos
  - Estructura de Internet
- Desempeño: pérdidas, retardo, throughput
- Modelo de servicio de capas
- Seguridad
- Historia

Ahora ustedes tienen:

- Contexto, visión general de la red
- Más detalles en profundidad *por venir!*