Taller de Capa de Enlace

Teoría de las Comunicaciones

Departamento de Computación
Facultad de Ciencias Exactas y Naturales
Universidad de Buenos Aires

05.09.2012

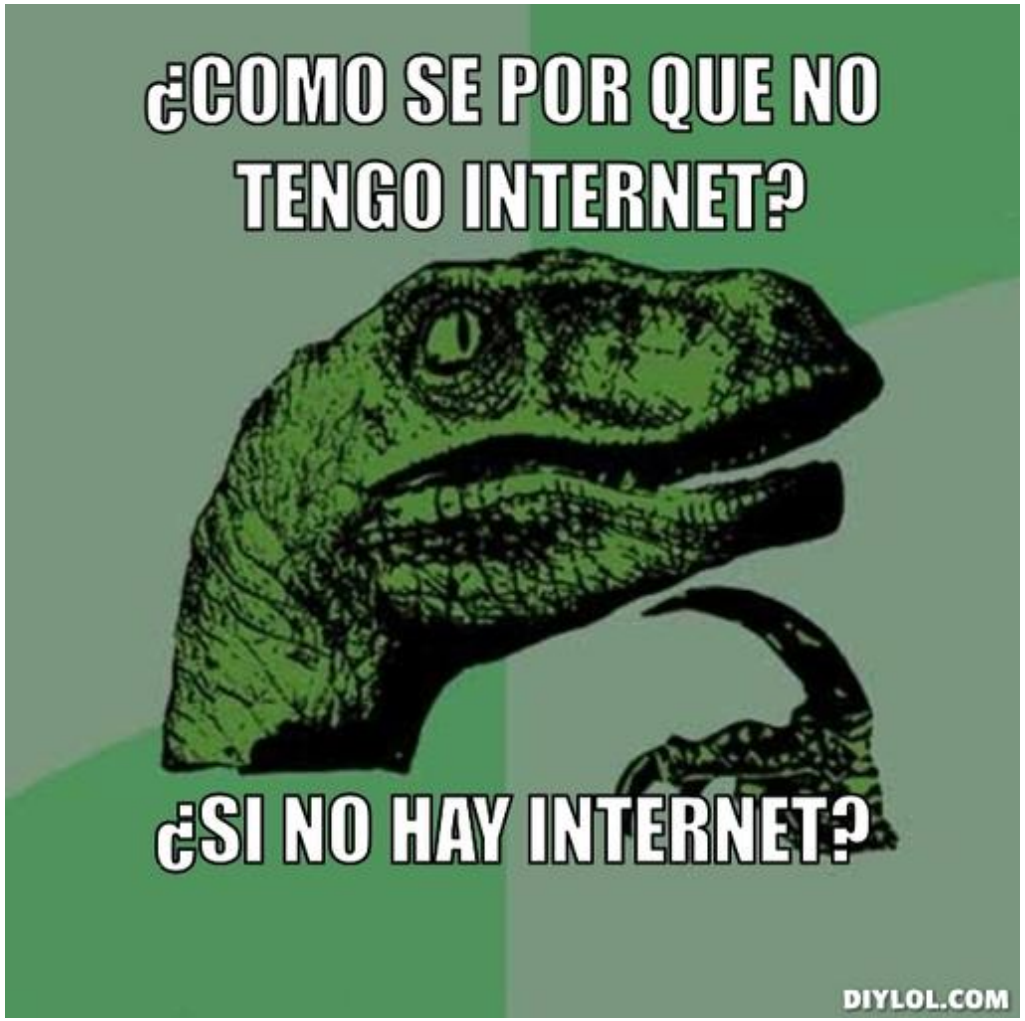
Agenda

- Anuncios parroquiales, ie: régimen de los talleres.
- Introducción a lo que veremos hoy, HWAddr? IP? Nivel OSI 2.5
- ARP Exposed, Qué? Quién? Cómo? Por qué?
- Tools varias para orientarse en la red.
- Presentación del taller.


CANAL 7

¿Qué son los talleres?

- Un paseo por las capas OSI.
- 3 entregables y uno presencial.
- Un acercamiento interactivo a las problemáticas de red.
- O, como diría el Philosoraptor...


¿COMO SE POR QUE NO TENGO INTERNET?
¿SI NO HAY INTERNET?
DIYLOL.COM

¿Qué esperamos?

- Que reflexionen sobre lo que es una red.
- Que se vayan con herramientas prácticas para hacer diagnóstico.
- Que entiendan los conceptos teóricos de una manera aplicada.
- Ah, y que entreguen informes rigurosos sobre lo que ustedes descubrieron.
  - Y código legible.
  - Y que respeten las pautas de entrega.

Hay tabla si...

- No entregan a término.
- No cumplen las pautas.
- El código es inentendible.
- El informe es poco claro.

¿Cómo es la modalidad de trabajo?

- Tres trabajos prácticos entregables en las capas de enlace, red y transporte.
- Un taller final presencial donde estaremos todos metiendo mano a la red. Muchas sorpresas, no se lo pierdan.

Fechas importantes

- **Entrega del taller de hoy**: martes 25 de septiembre.
- **Segundo taller**: ídem anterior.
- **Entrega del segundo taller**: martes 30 de octubre.
- **Tercer taller**: ídem anterior.
- **Entrega del tercer taller**: miércoles 21 de noviembre.
- **Cuarto taller**: ídem anterior.
- **Última fecha de entrega de recuperatorios**: el día del recuperatorio del primer parcial.

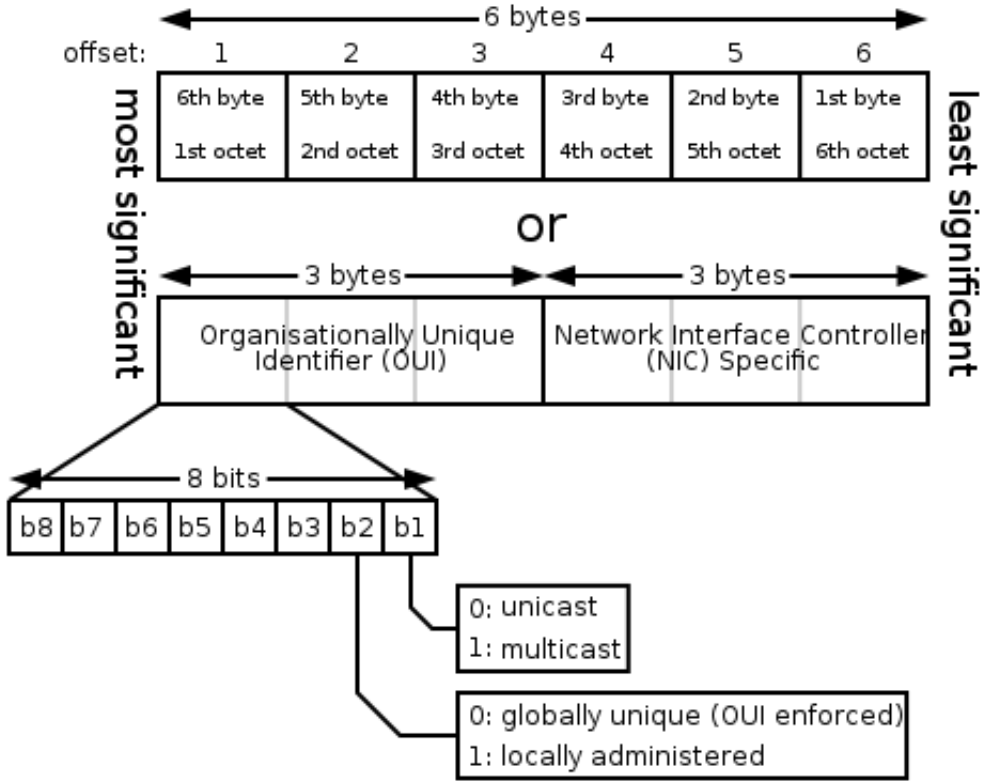
Algunas precauciones

- Todavía no vimos IP.
- No hablamos de routing.
- Nos estamos adelantando un poco.

Ethernet - MAC Address

- *Media Access Control Address*.
- Identificador de una interfaz de red.
- 6 octetos
- 3 de OUI (Organization Unique Identifier) `standards.ieee.org/develop/regauth/oui/public.html`
- 3 de NIC (Network Interface Controller)
- Intel Corporate: 00:1c:c0:fa:55:cc

Ethernet - MAC Address cont.



¿Dónde estamos parados?

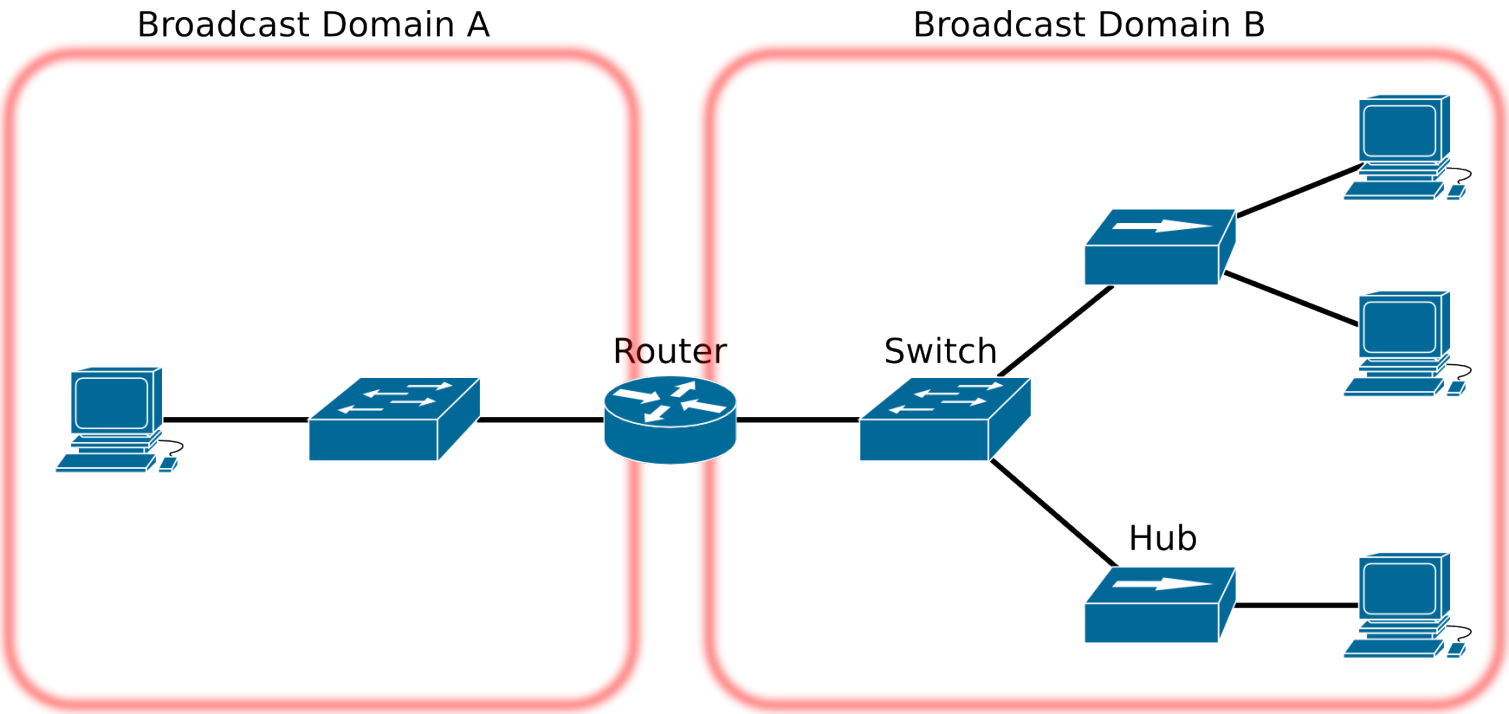
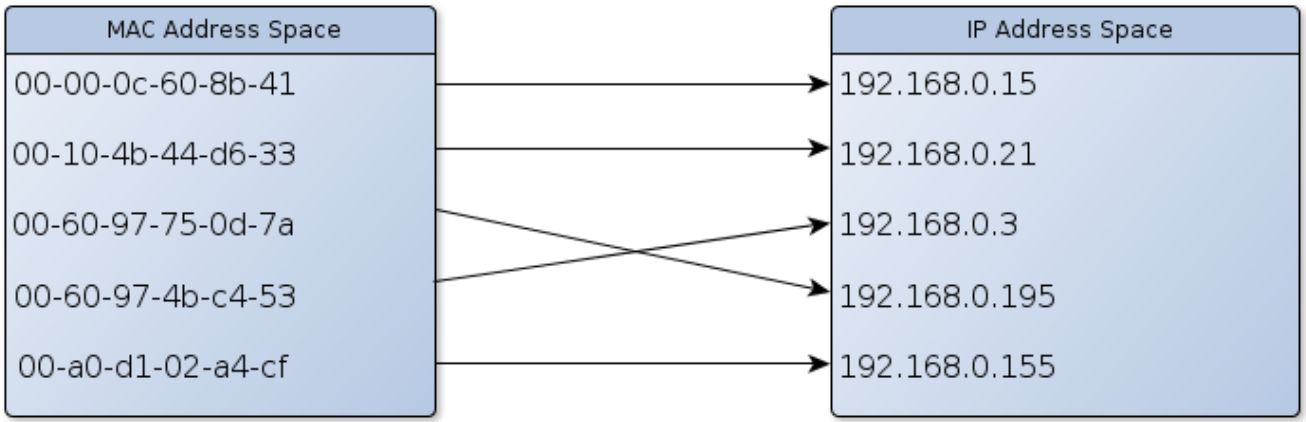


Figura: Mismo dominio de broadcast, mismo segmento de red

¿Perdón?


MAC Address Space
00-00-0c-60-8b-41
00-10-4b-44-d6-33
00-60-97-75-0d-7a
00-60-97-4b-c4-53
00-a0-d1-02-a4-cf
IP Address Space
192.168.0.15
192.168.0.21
192.168.0.3
192.168.0.195
192.168.0.155

¿Qué es ARP?

- La sigla: *Address Resoution Protocol*.
- Es un protocolo que, en esencia, permite mapear direcciones de nivel de red a direcciones físicas.
- Clave e indispensable en el funcionamiento de las redes modernas.
- Especificado en el RFC 826 (circa 1982).
- No está limitado a IP + Ethernet: la especificación es general.

¿Por qué lo queremos?

- Para hablar con una máquina remota (e.g., el servidor web de Google), mis datos van a tener que moverse primero por mi red local.
- Mi *router* será el encargado de recibirlos y mandarlos por donde corresponda.
- ¡Estos dispositivos de nivel de red necesitan también transmitir sobre la capa de enlace subyacente (e.g., Ethernet, 802.11, etc.)!

En definitiva, no es más que esto

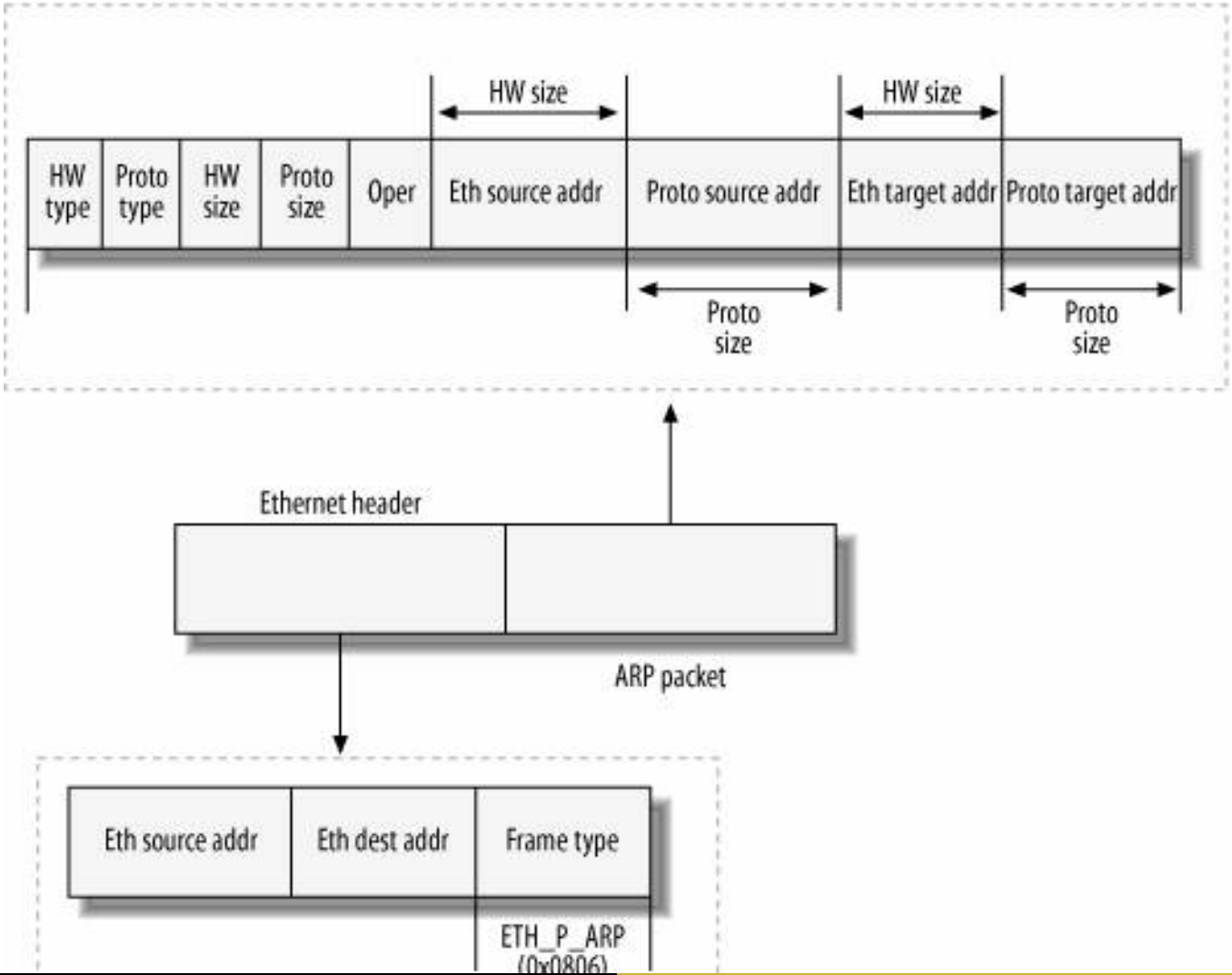
1. Para hablar con Google, sé que debo dirigirme, por ejemplo, a la dirección IP `173.194.42.19` (ya veremos cómo).
2. Mi máquina es astuta y sabe (ya veremos cómo) que, para ello, debe primero comunicarse con mi router, cuya IP es `192.168.1.1`.
3. Acá entra en juego ARP: mi máquina **pregunta** en mi red Ethernet local quién tiene registrada la IP `192.168.1.1`.
4. Y mi router **responde** diligentemente con su dirección física, `c0:c1:c0:0b:8f:2a`.

$\Rightarrow$ Mi máquina envía una trama Ethernet a esta MAC que encapsula un datagrama IP cuyo destino es el servidor de Google.

Tecnicismos varios

- La pregunta ARP consiste en un mensaje **broadcast** sobre la red local.
  - Recordar que no se propaga más allá de la red local!
- La respuesta, en cambio, es **unicast**.
- Optimización: se implementa una caché para guardar las direcciones resueltas (o conocidas).
  - Las entradas se agregan al resolver o bien al observar un pedido de otra máquina.
  - Cada entrada tiene un tiempo de expiración para evitar problemas.

Pormenores del paquete



Pormenores del paquete (cont.)

- El campo **Oper** puede tomar los valores 1 (`who-has`) o 2 (`reply`).
- Observar que la cantidad de bits asingada a las direcciones depende del valor que tomen los campos **HW size** y **Proto size**.
- Dichos campos tienen un largo de 8 bits (i.e., direcciones con un máximo de $2^8 - 1 = 255$ bits).
- **HW type** y **Proto type** indican los protocolos de nivel de enlace y de nivel de red respectivamente involucrados en la comunicación.

Otro uso interesante

- Cuando una máquina bootea o se levanta una de sus interfaces, muchos SOs envían automáticamente un pedido ARP *gratuito*.
- En él, **Proto source addr** == **Proto target addr**.
- Objetivos:
  - Detectar IPs duplicadas en la red local: esto ocurre si se recibe una respuesta.
  - Actualizar la caché ARP de los otros hosts.

...y otro uso más: ARP spoofing

- De lo anterior se desprende que ARP es un protocolo **sin estado** y **sin seguridad**.
- La técnica de ARP spoofing se apoya precisamente en estas características.
- Idea: una máquina envía de la nada una respuesta ARP mapeando una IP objetivo con su propia MAC.
- $\Rightarrow$ todo el tráfico destinado a dicha IP va a ser recibido por ella.

¿Qué es Wireshark?

- Una herramienta de análisis de protocolos.
- Una herramienta de captura de paquetes (aka: sniffer).
- Una herramienta de diagnostico de networking.
- El mejor amigo del administrador de red, analista de seguridad, programador, etc.

Captura de paquetes, ¿cómo?

- ¿NIC? Network Interface Controller (wlan0, eth0, lo, prueben haciendo ifconfig).
- Modo promiscuo, lo que significa que los paquetes con mac destino ajena no se descartan.
- Igual veramos mensajes broadcast, multicast y unicast.

Opcional, leer: http://www.tcpdump.org/faq.html

Escenarios

Local

- loopback
- eth, wlan, etc

Red local

- Atrás de un hub. Todos los mensajes se floodean.
- Atrás de un switch. No podemos ver mensajes ajenos. (Salvo que...)

Filtros

- Es demasiada información, necesitamos poder manejarla.
- Sintaxis Berkeley Packet Filter (BPF)

Ejemplos

- **broadcast ethernet**: eth.dst == FF:FF:FF:FF:FF:FF
- **ethernet type**: eth.type == 0xFFFF (2 bytes)
- **ether src ehost**: eth.src == 90:4c:e5:bb:e0:d6
- etc. Ver secciones Expression y Filter en la barra de filtro.

Recomendado: http://biot.com/capstats/bpf.html

Intro a Scapy

- Scapy es un framework de manipulación de paquetes.
- Permite crear paquetes, capturar paquetes, enviar paquetes, analizar paquetes, etc.
- Orientado a capas. pkt = Ether() / IP() / TCP() nos genera un paquete TCP valido.

Un poco de código

Creando un paquete con un payload

```
>>> e= Ether() / "un payload para mostrar como se apilan las capas"
>>> e.display()
###[ Ethernet ]###
WARNING: Mac address to reach destination not
found.
Using broadcast.
  dst= ff:ff:ff:ff:ff:ff
  src= 00:00:00:00:00:00
  type= 0x0
###[ Raw ]###
     load= 'un payload para mostrar como se stackean los paquetes'
```

Mandando un paquete por una interfaz

```
>>> e= Ether() / "un payload para mostrar como se apilan las capas"
>>> sendp(e, iface = "eth1")
WARNING: Mac address to reach destination not found. Using broadcast.
.
Sent 1 packets.
```

Primera consigna

(a) Implementar un cliente ARP sencillo: definir una función en Scapy (u otro lenguaje a elección con soporte para networking) que, dada una dirección IP, realice un pedido por la dirección física asociada y reciba y muestre la respuesta.

(b) Analizar qué ocurre al suministrarle distintas direcciones de la red local:

- Una dirección inexistente,
- La misma dirección de la máquina origen,
- etc.

Segunda consigna

(a) Implementar una función para escuchar pasivamente en la red local por un lapso de tiempo dado y capturar cada mensaje ARP encontrado.

(b) A partir de lo anterior, obtener las direcciones físicas involucradas y exhibir los respectivos OUI (i.e., vendors). Ver `http://standards.ieee.org/develop/regauth/oui/oui.txt`.

Tercera consigna

- Utilizando lo hecho en la consigna previa, graficar los datos encontrados y realizar un análisis de lo observado.
- Algunas sugerencias: histogramas de IPs solicitadas y/o grafos dirigidos de IPs (request $\rightarrow$ reply).
- ¡Pensar!