

Tarea 4

Respuestas:

Caso 1:

R1:

El filtro que se necesita para lograr lo pedido seria: `udp && ip.dst == 255.255.255.255`

R2:

Existen tres paquetes que llegan. Cada *frame* tiene 72, 73 y 74 bytes. De estos, cada uno tiene 30, 31 y 32 bytes de UDP correspondientes en el mismo orden. Esta diferencia en cada uno es debido a que lo que generalmente se llama como *payload* es solo una parte del paquete completo. Estos tienen una diferencia entre UDP y el frame completo debido a que el mensaje viene complementado con otras secciones y flags que necesita el paquete para poder enviar de manera correcta el *payload*.

R3:

Los tres mensajes que llegan son “Mi numero de la suerte es: 7”, “Mi numero de la suerte es: 83” y “Mi numero de la suerte es: 364”. Cada mensaje respectivamente tiene un largo de bytes de 30, 31 y 32.

R4:

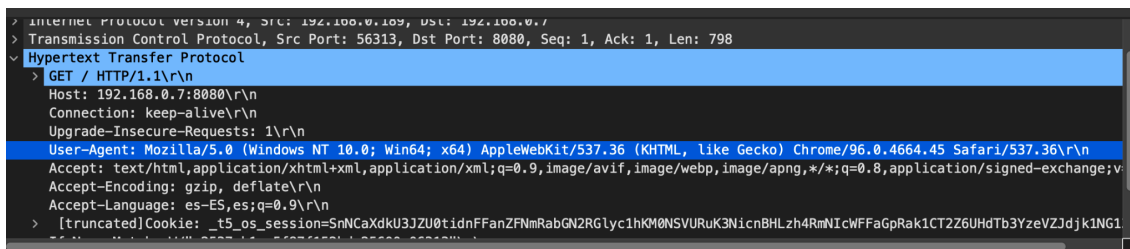
Para poder lograr esto lo que hay que hacer es enviar el mensaje a la dirección de *broadcast* de la subred. En este caso lo que podríamos hacer son dos cosas. Por un lado, sería mandar el mensaje a la dirección 255.255.255.255.

Caso 2:

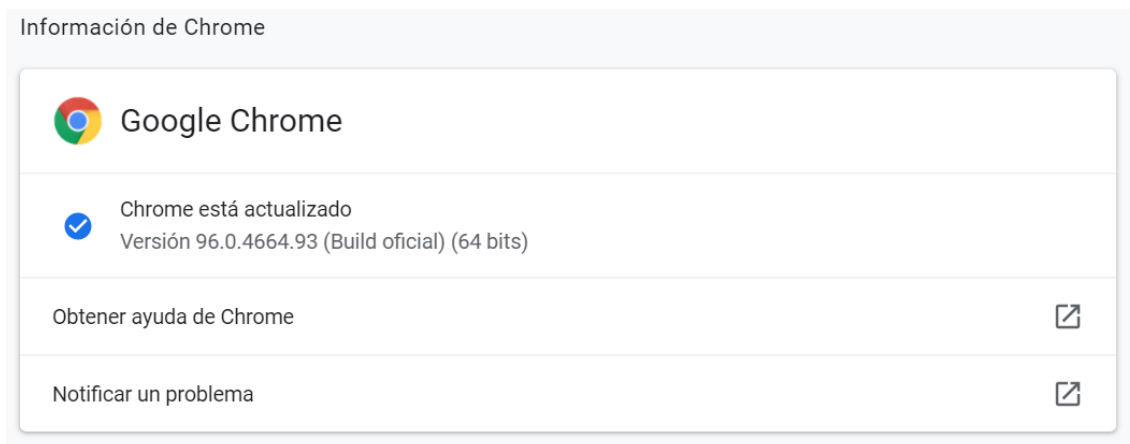
R1:

El *browser* ocupado es Google Chrome versión 96.0.4664.45.

A continuación, se adjunta una foto donde se puede ver el *browser* descrito:



```
> Internet Protocol version 4, Src: 192.168.0.109, Dst: 192.168.0.1
> Transmission Control Protocol, Src Port: 56313, Dst Port: 8080, Seq: 1, Ack: 1, Len: 798
> Hypertext Transfer Protocol
  > GET / HTTP/1.1\r\n
    Host: 192.168.0.7:8080\r\n
    Connection: keep-alive\r\n
    Upgrade-Insecure-Requests: 1\r\n
    User-Agent: Mozilla/5.0 (Windows NT 10.0; Win64; x64) AppleWebKit/537.36 (KHTML, like Gecko) Chrome/96.0.4664.45 Safari/537.36\r\n
    Accept: text/html,application/xhtml+xml,application/xml;q=0.9,image/avif,image/webp,image/apng,*/*;q=0.8,application/signed-exchange;v
    Accept-Encoding: gzip, deflate\r\n
    Accept-Language: es-ES,es;q=0.9\r\n
    > [truncated] Cookie: _t5_os_session=SnNCaXdkU3JZU0tidnFFanZFnmRabGN2RGlyc1hkM0NSVURuK3NicnBHLzh4RmNlcWFaGpRak1CT2Z6UHdTb3YzeVZjdjk1NG1
```



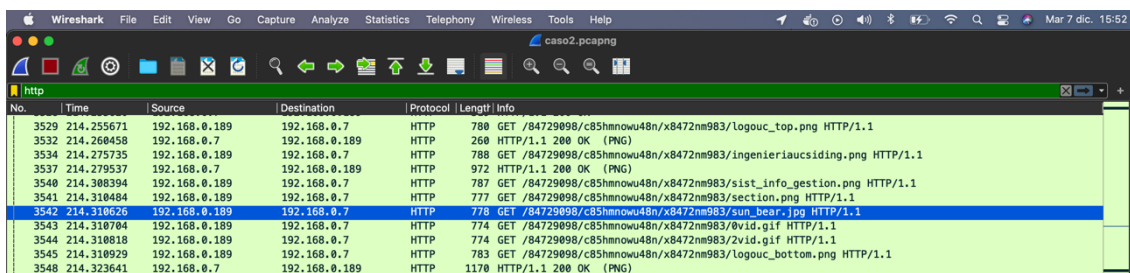
R2:

El código HTTP que se ve al consultar por /big.txt es 200. Esto significa que la solicitud y la respuesta se completaron de manera exitosa. Por otro lado, al consultar por /small.txt la respuesta es de 404. Esto significa que el archivo que se está buscando por el cliente en el servidor en aquella dirección no fue encontrado.

R3: La dirección que viene de la imagen de la primera vista es:

192.168.0.7/84729098/c85hmnou48n/x8472nm983/sun_bear.jpg

Una imagen que corrobora esto es:



No.	Time	Source	Destination	Protocol	Length	Info
3529	214.255671	192.168.0.189	192.168.0.7	HTTP	780	GET /84729098/c85hmnou48n/x8472nm983/logouc_top.png HTTP/1.1
3532	214.268458	192.168.0.7	192.168.0.189	HTTP	268	HTTP/1.1 200 OK (PNG)
3534	214.275735	192.168.0.189	192.168.0.7	HTTP	788	GET /84729098/c85hmnou48n/x8472nm983/ingenieriaucsd.png HTTP/1.1
3537	214.279537	192.168.0.7	192.168.0.189	HTTP	972	HTTP/1.1 200 OK (PNG)
3540	214.308394	192.168.0.189	192.168.0.7	HTTP	787	GET /84729098/c85hmnou48n/x8472nm983/sist_info_gestion.png HTTP/1.1
3541	214.310484	192.168.0.189	192.168.0.7	HTTP	777	GET /84729098/c85hmnou48n/x8472nm983/section.png HTTP/1.1
3542	214.310626	192.168.0.189	192.168.0.7	HTTP	778	GET /84729098/c85hmnou48n/x8472nm983/sun_bear.jpg HTTP/1.1
3543	214.310704	192.168.0.189	192.168.0.7	HTTP	774	GET /84729098/c85hmnou48n/x8472nm983/ovid.gif HTTP/1.1
3544	214.310818	192.168.0.189	192.168.0.7	HTTP	774	GET /84729098/c85hmnou48n/x8472nm983/2vid.gif HTTP/1.1
3545	214.310929	192.168.0.189	192.168.0.7	HTTP	783	GET /84729098/c85hmnou48n/x8472nm983/logouc_bottom.png HTTP/1.1
3548	214.323641	192.168.0.7	192.168.0.189	HTTP	1178	HTTP/1.1 200 OK (PNG)

R4: Cada vez que se quiere hacer un GET de la captura del *sun bear* hay una captura de 778 bytes.

R5: Cada vez que se ingresa a la pagina dada por 192.168.0.7/. se hacen alrededor de 29 GET. Esto es debido a que son múltiples cosas que se necesitan traer desde el servidor para que se pueda desplegar la página de manera íntegra. Primero se trae lo relacionado con el ícono del tab y ciertas imágenes png y gifs. Posteriormente, se traen funciones de js y assets necesarios. Luego viene el diseño por medio de archivos css. Finalmente, se trae el resto de los assets que faltaban.

Caso 3:

R1:

Se envían principalmente paquetes del protocolo ICMP. Este paquete es sobre una red de protocolo que es responsable de reportar errores a través de la generación y envío de mensajes a la dirección de IP de origen cuando hay problemas de red que son encontrados por el sistema. El contenido de este paquete es un tipo, un código y una suma de verificación. Finalmente, lo que trae en otras palabras son textos que le dicen a la IP de origen si es que se encuentran problemas en el sistema.

R2:

Los paquetes ARP permiten obtener la dirección MAC de otros dispositivos conectados a la red.

R3:

Al desconectarse y volver a conectarse a la red local, el primer paquete que se obtiene es el siguiente. Comprobando la conexión mediante DHCP.

1 0.000000	0.0.0.0	255.255.255.255	DHCP	364 DHCP Request - Transaction ID 0x818640d9
------------	---------	-----------------	------	----------------------------------------------