C

**NOMBRE:** CONDORI LANZA SILVIA EUGENIA **C.I.:** 6164141 LP

**LICENCIADO:** RAMIRO GALLARDO

**PARALELO:** A

**FECHA:** 2 DE JULIO DE 2020

**EJERCICOS DE AUXILIATURA**

**AUX: ALEJANDRO**

TAREA EN PYTHON

Sniffer

CONSTRUCCIÓN DE UN SNIFFER.

El código debe estar realizado en python, debe ser capaz de leer las cabeceras de.

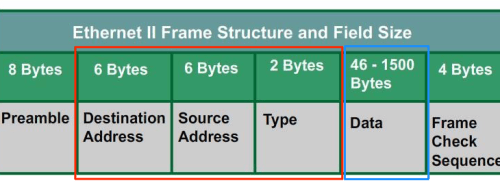
-Ethernet

-IP

-(TCP, UDP, ICMP) según corresponda Y deben tener el siguiente formato NO SE ACEPTARA OTRO FORMATO

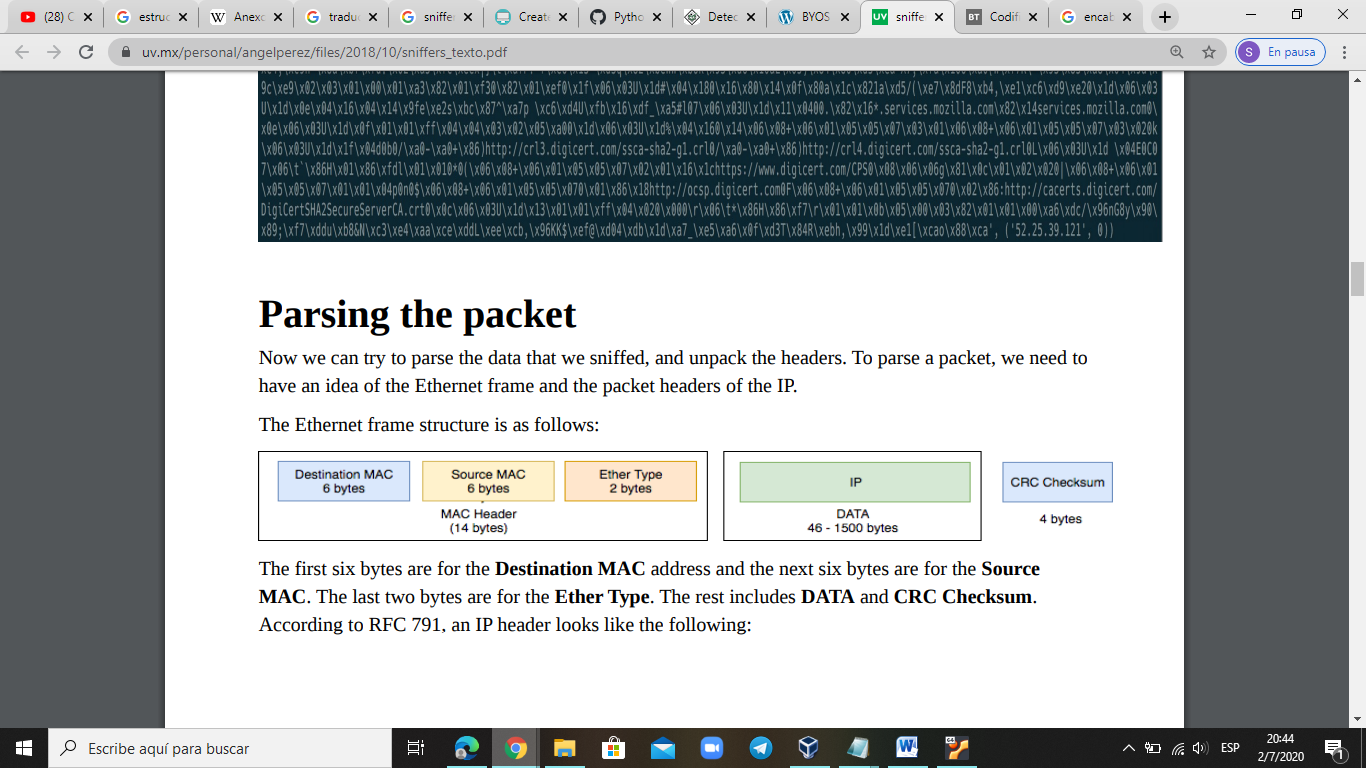


**ETHERNET header**



En particular, el segmento de datos viene después de los primeros 14 bytes de la trama Ethernet.

Los primeros seis bytes son para la dirección MAC de destino y los siguientes seis bytes son para la fuente MAC. Los dos últimos bytes son para el tipo Ether. El resto incluye DATA y CRC Checksum. Según RFC 791, un encabezado IP se parece a lo siguiente:



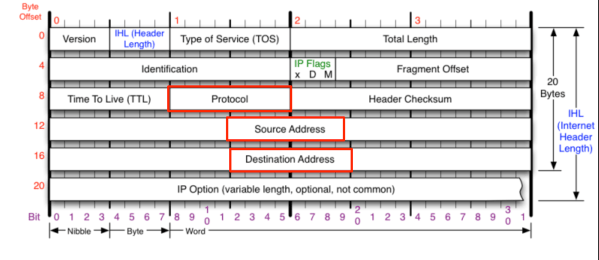
La siguiente decisión sobre cómo proceder primero proviene del hecho de que el paquete debe ser un paquete UDP ya que tanto LLMNR como mDNS envían su tráfico en UDP.

Los protocolos de paquetes IP vienen en el siguiente formato:

8 - ETHERNET

1 - ICMP  
6 - TCP  
17 - UDP

**IP header**



Vamos a desglosarlo porque está empezando a ser un poco complicado. El encabezado IP se encuentra dentro de la porción de DATOS de la trama de Ethernet y se ve como la imagen.

El encabezado IP incluye las siguientes secciones:

• **Versión del protocolo** (cuatro bits): los primeros cuatro bits. Esto representa el protocolo IP actual.

• **Longitud del encabezado** (cuatro bits): la longitud del encabezado IP se representa en palabras de 32 bits. Ya que este campo es de cuatro bits, la longitud máxima de encabezado permitida es de 60 bytes. Por lo general, el valor es 5, lo que significa cinco palabras de 32 bits: 5 \* 4 = 20 bytes.

• **Tipo de servicio** (ocho bits): los primeros tres bits son bits de precedencia, los siguientes cuatro bits representan el tipo de servicio, y el último bit se deja sin usar.

• **Longitud total** (16 bits): representa la longitud total del datagrama IP en bytes. Este es un campo de 16 bits. El tamaño máximo del datagrama IP es de 65.535 bytes.

• **Banderas** (tres bits): el segundo bit representa el bit No fragmentar. Cuando se establece este bit, el datagrama de IP nunca está fragmentado. El tercer bit representa el bit Más Fragmento. Si este bit es configurado, entonces representa un datagrama IP fragmentado que tiene más fragmentos después de él.

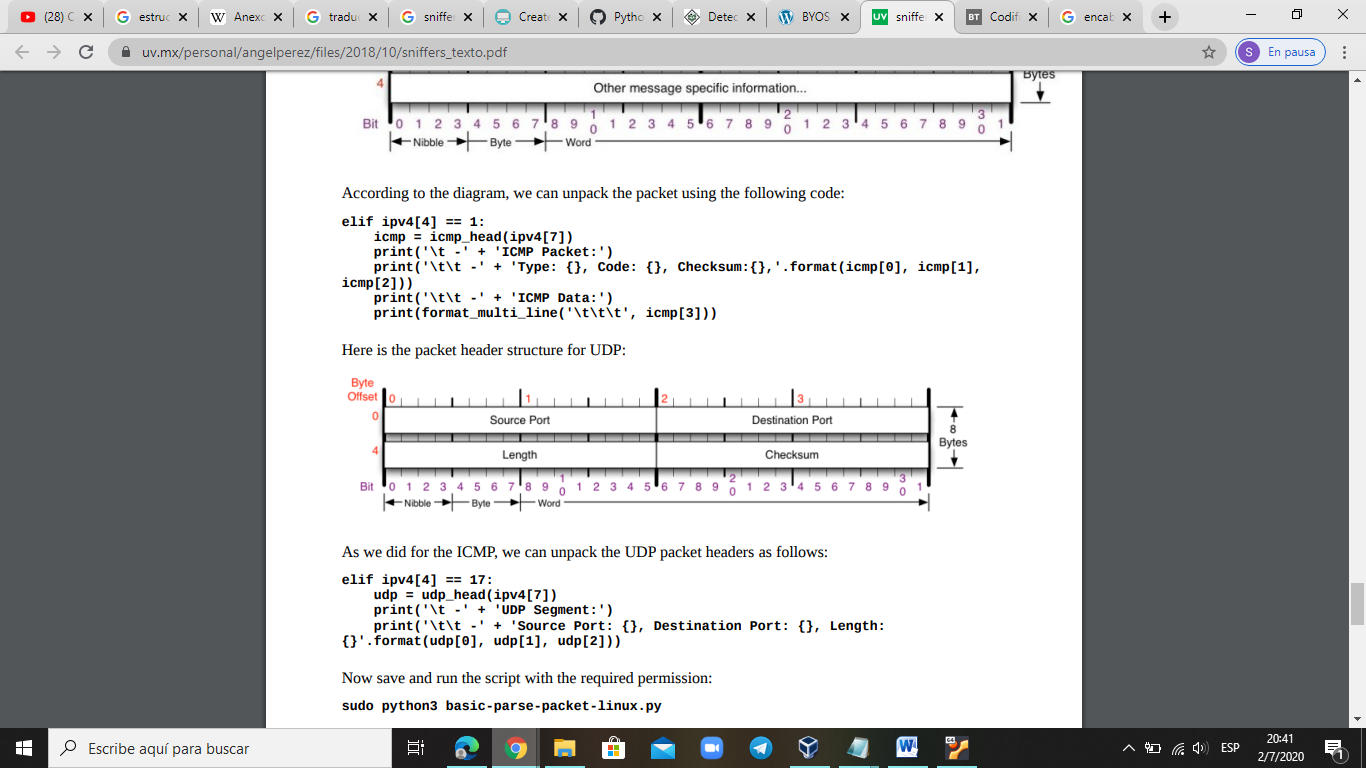
• **Tiempo de vida** (ocho bits): este valor representa el número de saltos que el datagrama IP pasar antes de ser descartado.

• **Protocolo** (ocho bits): representa el protocolo de la capa de transporte que entregó los datos a la IP capa.

• **Suma de verificación del encabezado** (16 bits): este campo ayuda a verificar la integridad de un datagrama IP.

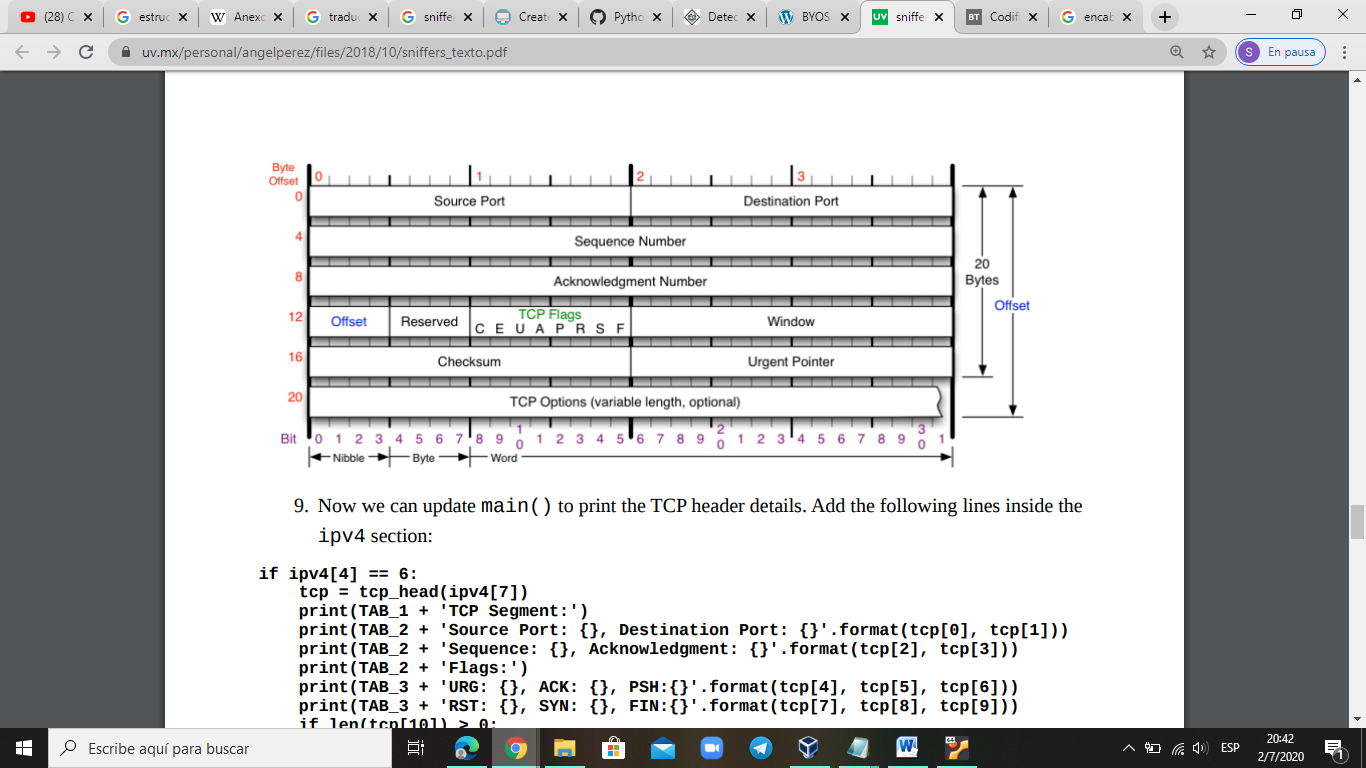
• **IP de origen y destino** (32 bits cada uno): estos campos almacenan la dirección de origen y de destino, respectivamente.

**UDP header**

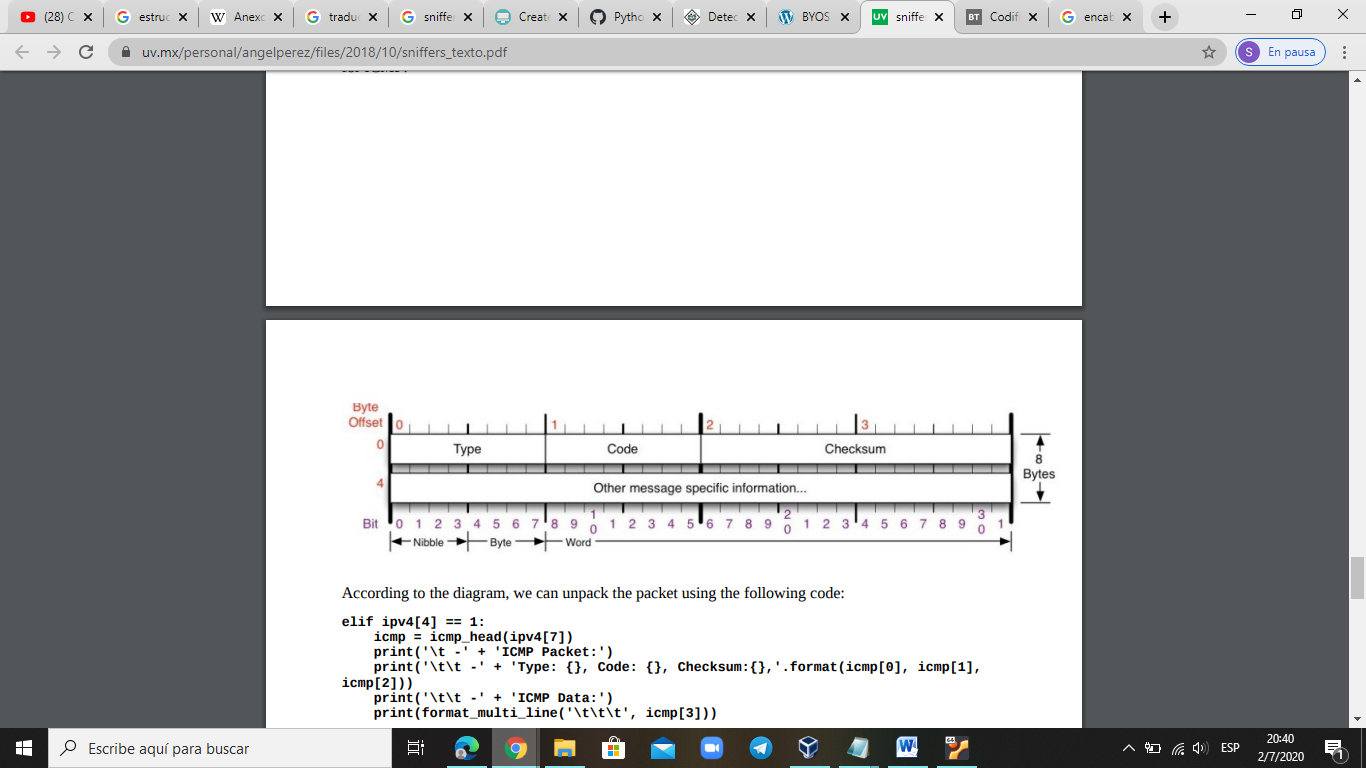


Entonces, sabemos que toda la parte completa del paquete IP, excluyendo las opciones y la porción de datos, es de 20 bytes, por lo tanto, toda la información que necesitamos está en esos 20 bytes. También necesitamos lo que viene después de los 20 bytes del encabezado, que son los datos del paquete UDP.

**TCP header**



**ICMP header**



1. El sniffer anterior recoge solo los paquetes TCP, debido a la declaración:

s = socket.socket (socket.AF\_INET, socket.SOCK\_RAW, socket.IPPROTO\_TCP)

Para UDP e ICMP, la declaración debe ser:

s = socket.socket (socket.AF\_INET, socket.SOCK\_RAW, socket.IPPROTO\_UDP)  
s = socket.socket (socket.AF\_INET, socket.SOCK\_RAW, socket.IPPROTO\_ICMP)

Ahora veamos cómo podemos superar los inconvenientes mencionados anteriormente. La solución es:

Esta línea :

s = socket.socket (socket.AF\_INET, socket.SOCK\_RAW, socket.IPPROTO\_TCP)

necesita ser cambiado a:

s = socket.socket (socket.AF\_PACKET, socket.SOCK\_RAW, socket.ntohs(0x0003))

Ahora recibirá el mismo socket:

1. Todo el tráfico entrante y saliente.
2. Todas las tramas de Ethernet, lo que significa todo tipo de paquetes IP (TCP, UDP, ICMP) e incluso otros tipos de paquetes (como ARP) si los hay.
3. También proporcionará el encabezado de ethernet como parte del paquete recibido.