Trabajo Práctico 6

*Análisis de tráfico TCP y UDP*

Autores

Franco Di Lorenzo: [francodi65@gmail.com](mailto:francodi65@gmail.com)

Juan Del Boca: [juanmadelboca@gmail.com](mailto:juanmadelboca@gmail.com)

# Desarrollo

## Ejercicio 1: Análisis de tráfico TCP y UDP en GNU/Linux

## Tabla de asignación de direcciones IP

Cliente eth0 IPv6: 2001:bbbb:bbbb::10

Servidor eth0 IPv6: 2001:aaaa:aaaa::10

Router eth0 IPv6: 2001:aaaa:aaaa::1

eth1 IPv6: 2001:bbbb:bbbb::1

**1.- Servidor DNS**

## 1.1- Instalar y configurar el servicio DNS en Server

Para la configuración del servicio DNS utilizamos el programa bind9, el cual se instala:

Ubuntu@server:~$ sudo apt-get install bind9

A continuación modificamos el archivo named.conf.local, con el comando:

Ubuntu@server:~$ sudo nano /etc/bind/named.conf.local

Creamos la zona grupo13.redes.fcefyn.unc.edu.local:

zone "grupo13.redes.fcefyn.unc.edu.local" {

             type master;

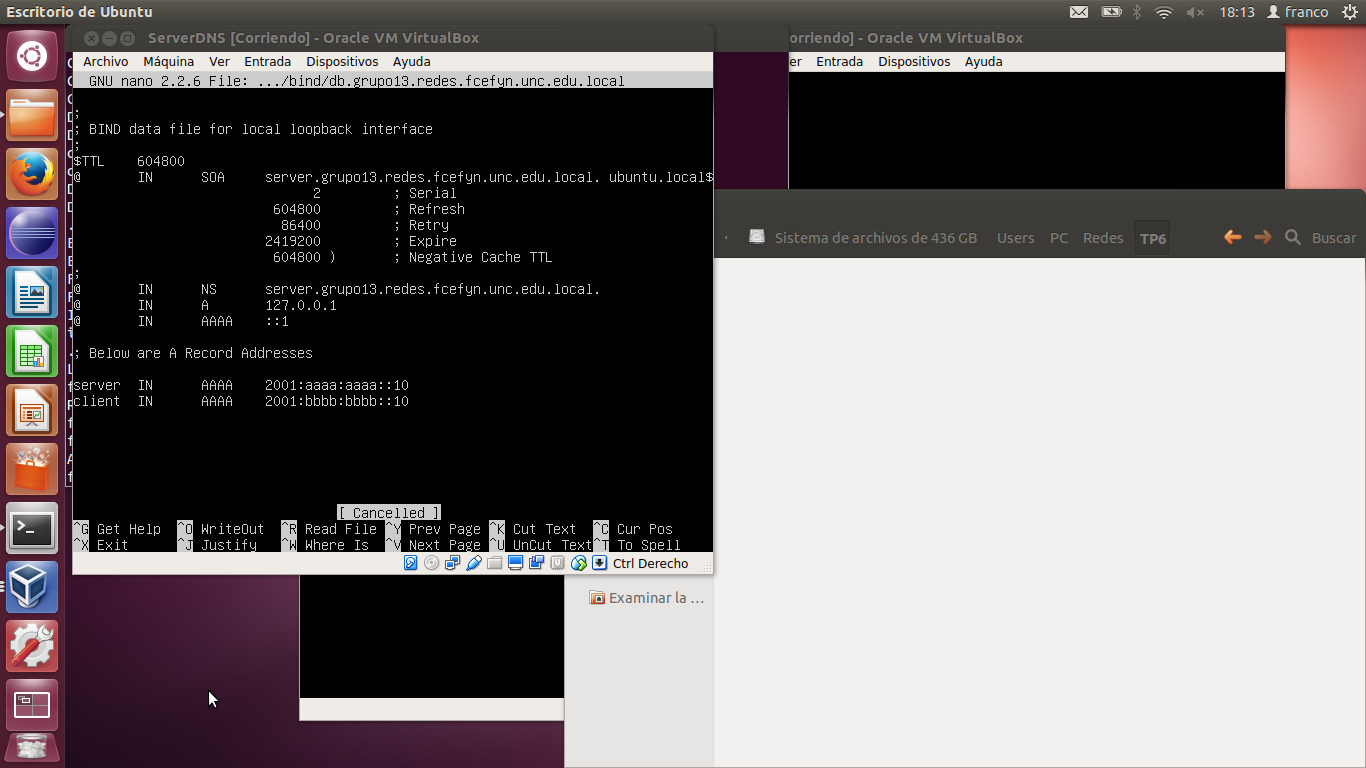
             file "/etc/bind/db. grupo13.redes.fcefyn.unc.edu.local ";

        };

Luego creamos el archivo de configuración de la zona:

Ubuntu@server:~$ sudo cp /etc/bind/db.local /etc/bind/db.grupo13.redes.fcefyn.unc.edu.local

Y realizamos las siguientes modificaciones:



Finalmente, reseteamos bind9 y habilitamos el servidor DNS en los clientes

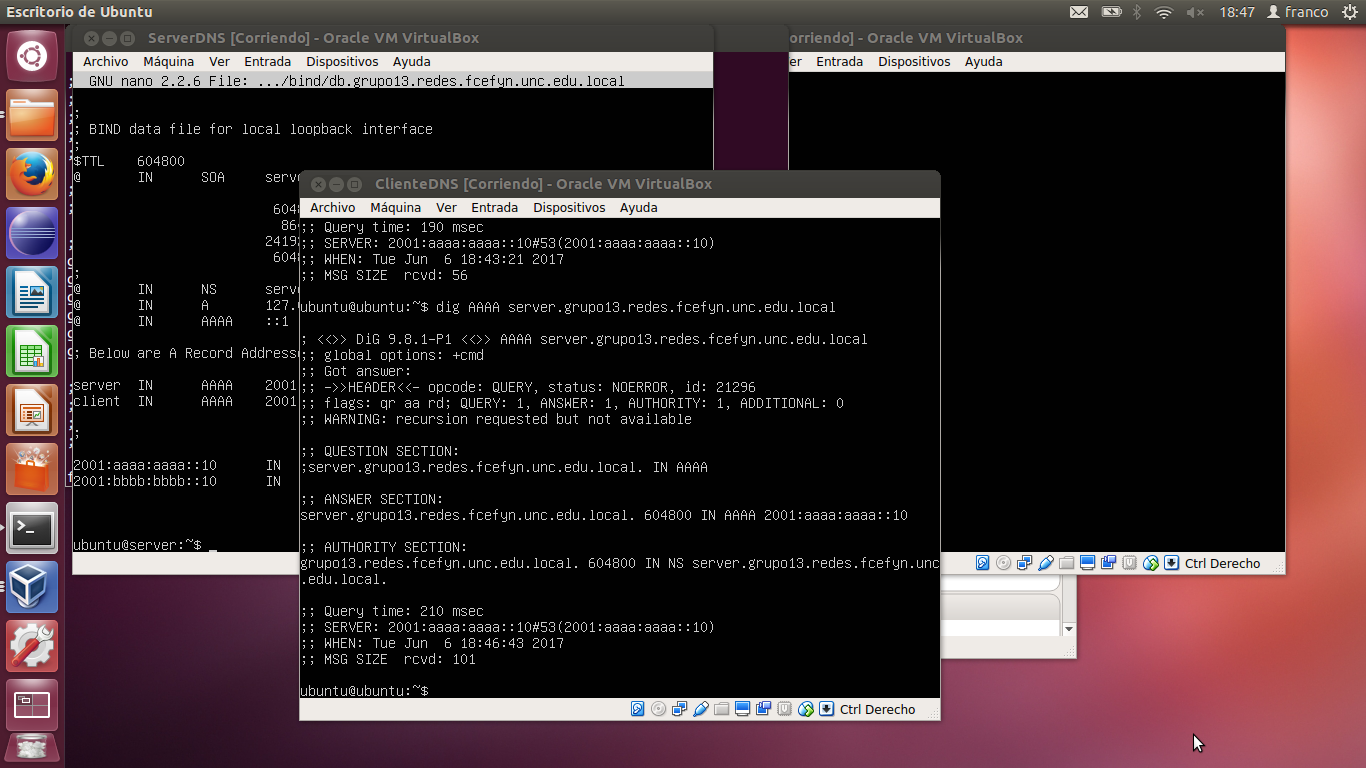
Ubuntu@server:~$ sudo /etc/init.d/bind9 restart

**2.- Análisis de Tráfico**

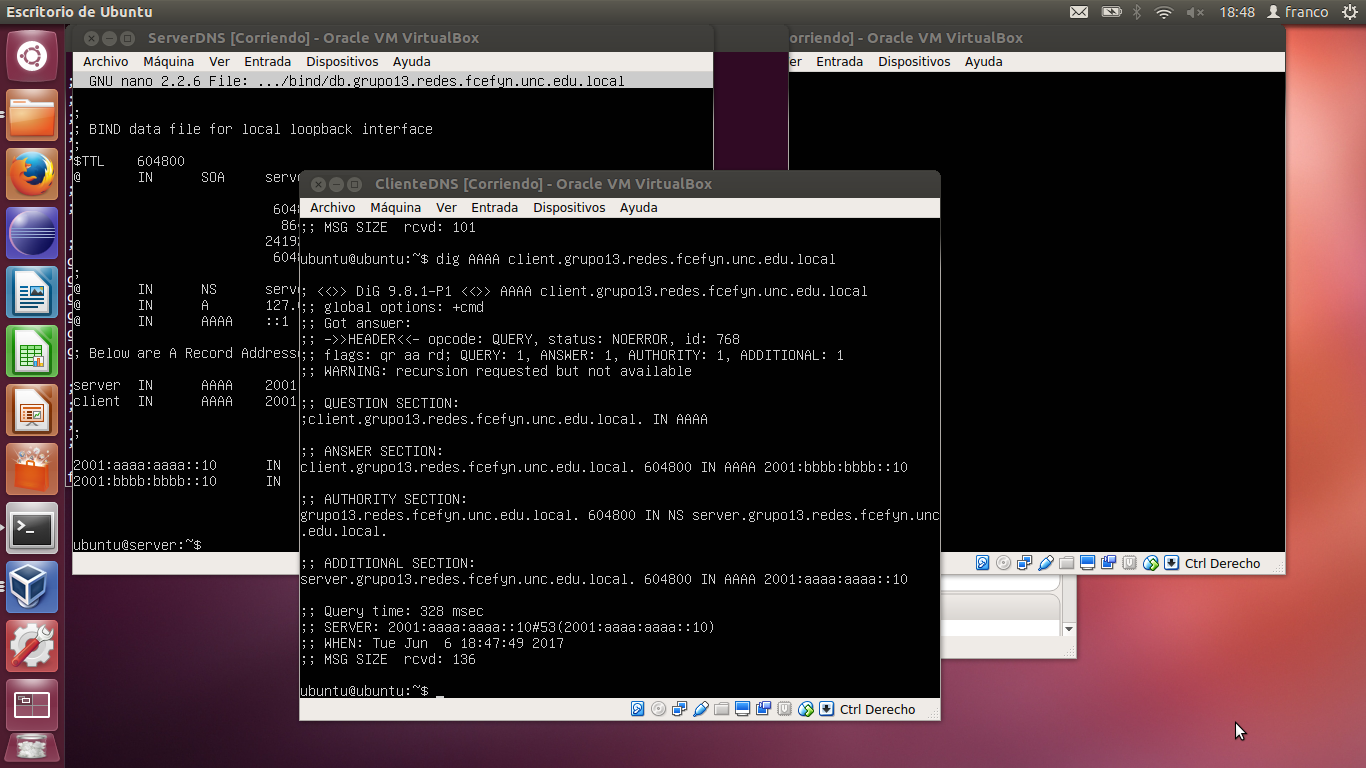
## 2.1- Realizar consultas DNS con el comando “dig”

Desde una pc cliente, ejecutamos la consulta dig(QUESTION SECTION), y observamos la respuesta recibida(ANSWER SECTION).

Para *server.grupo13.redes.fcefyn.unc.edu.local*

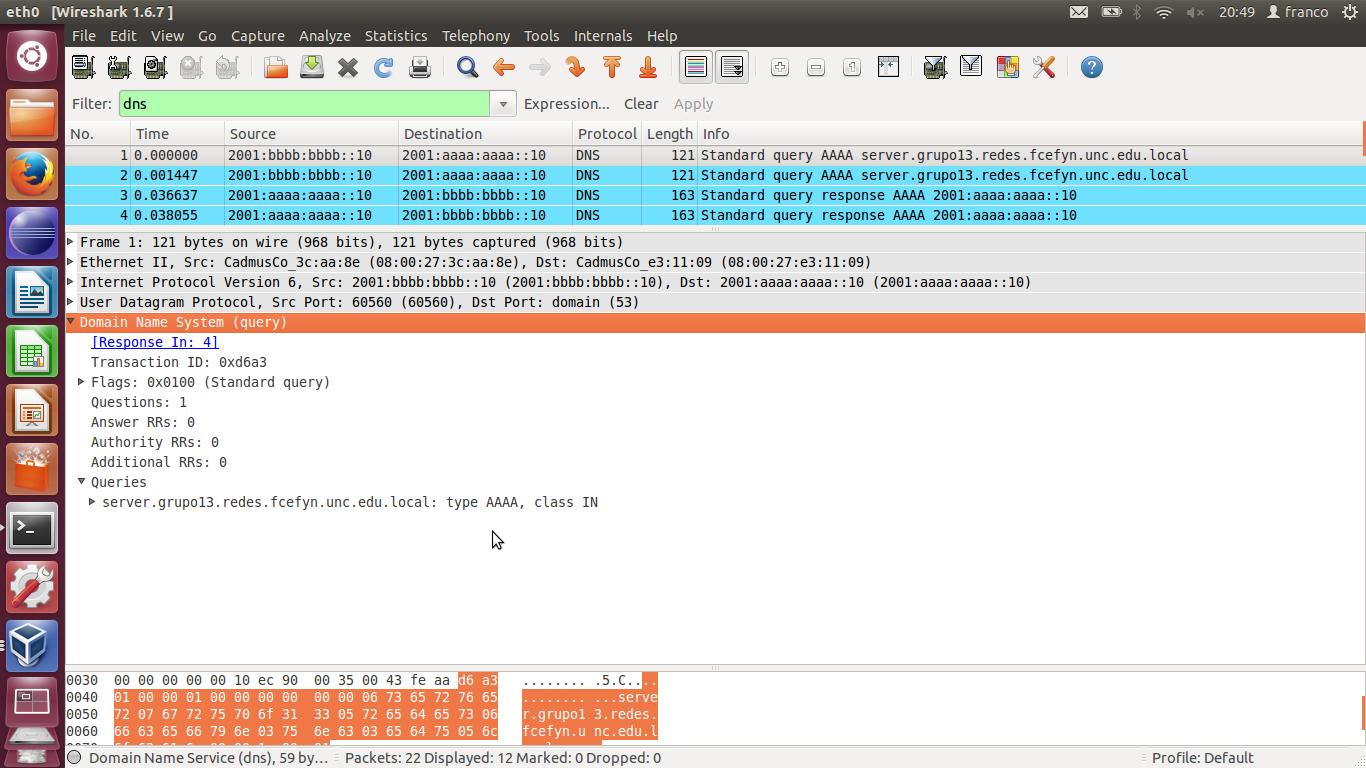


Luego ejecutamos la consulta para *client.grupo13.redes.fcefyn.unc.edu.local:*

**

## 2.2- Analizar el tráfico con Wireshark

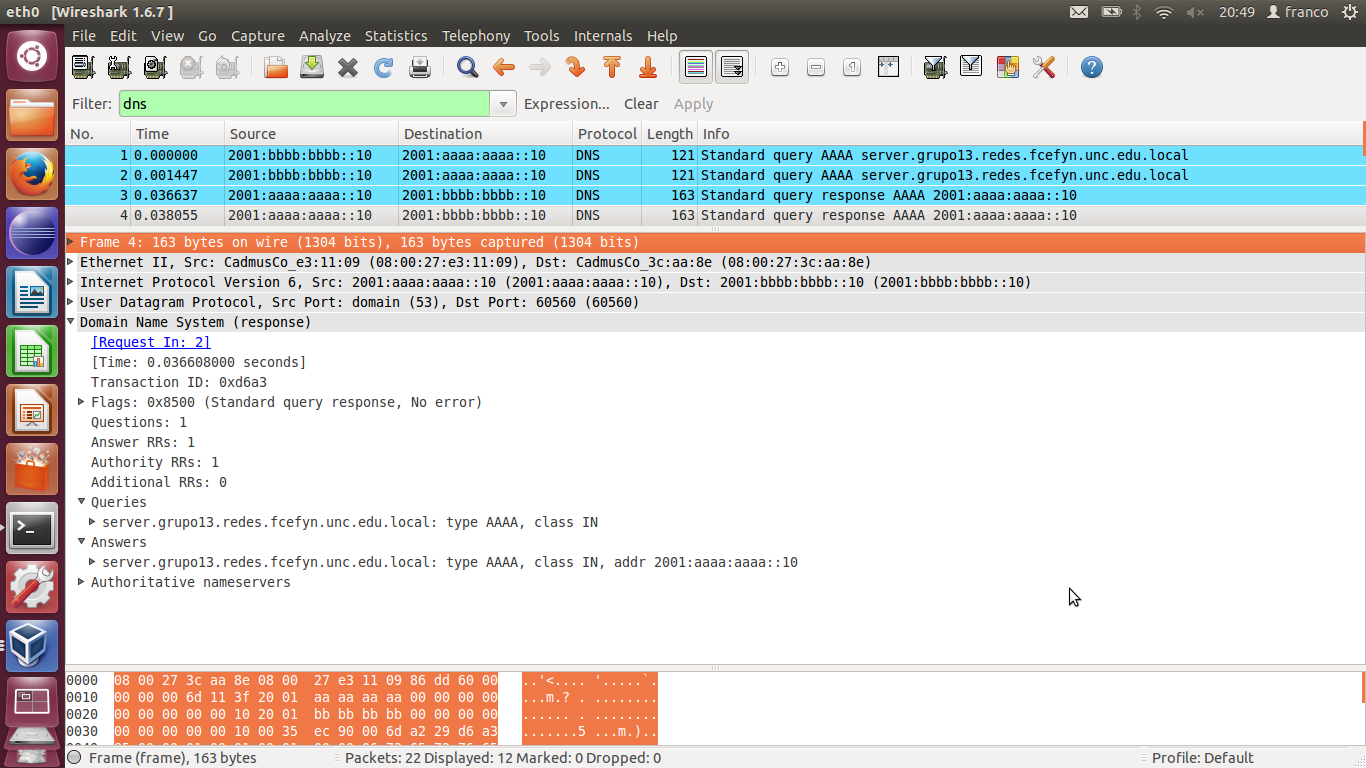
En la siguiente imagen, se observan los 4 paquetes con protocolo DNS, 2 de consulta(1 debido al reenvío por parte del router) y 2 de respuesta(también 1 debido al reenvío del router).



Analizando las capas del primer paquete DNS, se observa en la capa 2 la dirección mac de origen(pc cliente) y dirección mac destino(router). En el otro paquete, la dirección mac origen será la del router y la mac destino la del servidor DNS.  
En la capa 3, se observa el origen del datagrama IP(pc cliente) y destino(servidor DNS).  
Luego en la capa 4, se utiliza el protocolo UDP para el envío de mensajes de la capa de aplicación. Se utiliza el puerto 53 para los mensajes DNS.

Finalmente en la capa 5, se observa la consulta generada por la pc cliente es decir, se consulta por la dirección IPv6(type AAAA) de la dirección *server.grupo13.redes.fcefyn.unc.edu.local.*

Luego, en la respuesta del mensaje, de forma similar en las capas inferiores, se devuelve una respuesta, la cual se observa en la siguiente imagen:



En la capa 5, se observa la consulta original en *Queries* y la respuesta con la dirección IPv6 en el campo *Answers*.

## Ejercicio 2: Análisis de Tráfico TCP sobre servidor HTTP

## 1.- Servidor Wordpress

**1.1.- Instalar y configurar el servicio Wordpress en el Server.**

Para comenzar instalamos LAMP (apache2, mysql, y php) una vez instalado eso. Procedemos a instalar y configurar Wordpress.

Primero nos logeamos en mysql y creamos una base de datos y le asignamos los permisos:

Ubuntu@WordPressServer:~$ mysql –u root –p

Mysql> CREATE DATABASE wordpress DEFAULT CHARACTER SET utf8 COLLATE utf8\_unicode\_ci;

Mysql> GRANT ALL ON wordpress.\* TO 'wordpressuser'@'localhost' IDENTIFIED BY 'password';

Mysql> FLUSH PRIVILEGES;

Mysql> EXIT;

Ubuntu@WordPressServer:~$ sudo apt-get install bind9

Luego instalamos unas extensiones de php necesarias para wordpress usando el siguiente comando:

Ubuntu@WordPressServer:~$ sudo apt-get update

Ubuntu@WordPressServer:~$ sudo apt-get install php-curl php-gd php-mbstring php-mcrypt php-xml php-xmlrpc

Y reiniciamos apache. Sobreescribimos la configuración de apache para que Wordpress tenga la configuración que necesita agregando en /etc/apache/2/apache2.conf las siguientes líneas:

<Directory /var/www/html/>

AllowOverride All

</Directory>

Descargamos wordpress desde una carpeta temporal (recomendado por la pagina oficial) y luego lo descomprimimos para que cree el directorio desde donde trabajara:

Ubuntu@WordPressServer:~$cd /tmp

Ubuntu@WordPressServer:~$ curl -O <https://wordpress.org/latest.tar.gz>

Ubuntu@WordPressServer:~$ tar xzvf latest.tar.gz

Ubuntu@WordPressServer:~$ touch /tmp/wordpress/.htaccess

Ubuntu@WordPressServer:~$ chmod 660 /tmp/wordpress/.htaccess

Luego copiamos copiamos a la carpeta a donde trabajara:

Ubuntu@WordPressServer:~$ sudo cp -a /tmp/wordpress/. /var/www/html

Ajustamos los permisos para que toda la carpeta de donde trabaja wordpress sea accesible para el usuario no root y luego pedimo al servidor de wordpress secret key y la copiamos en nuestro archivo de cofiguracion /var/www/html/wp-config.php:

Ubuntu@WordPressServer:~$ curl -s <https://api.wordpress.org/secret-key/1.1/salt/>

Modificamos lo siguiente:

define('DB\_NAME', 'wordpress');

/\*\* MySQL database username \*/

define('DB\_USER', 'wordpressuser');

/\*\* MySQL database password \*/

define('DB\_PASSWORD', 'password');

. . .

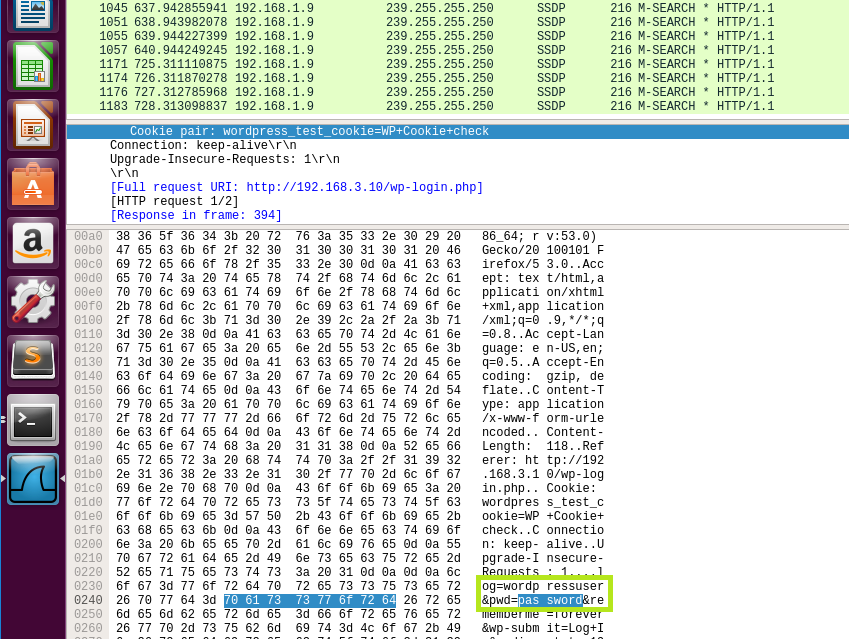
Y agregamos esta línea:

define('FS\_METHOD', 'direct');

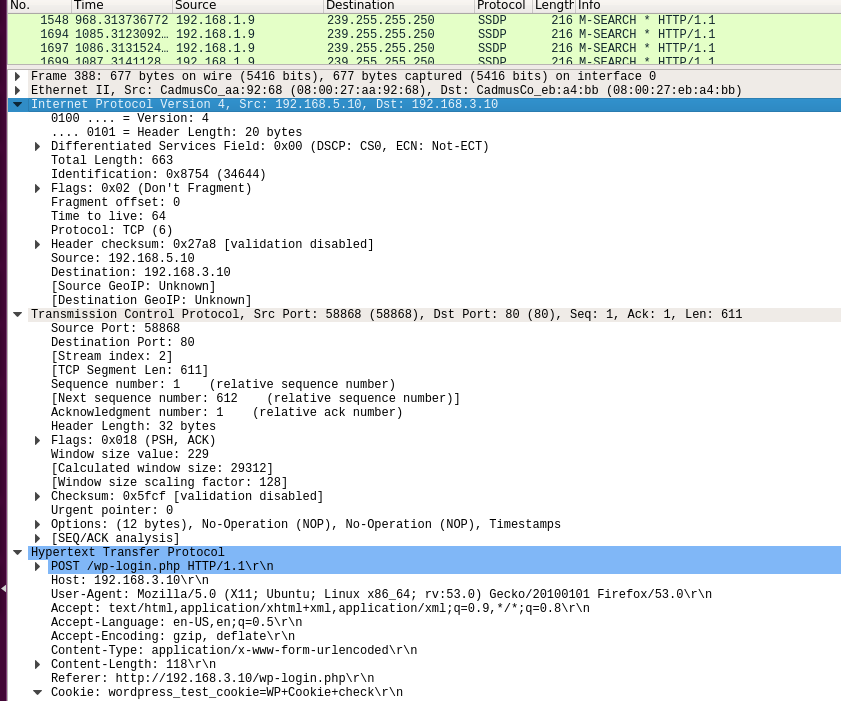
## 2.- Análisis de Tráfico

**2.1.- Desde el cliente utilizar el navegador web para poder autenticarse en el portal de Wordpress.**

Al logearse se puede ver que la contraseña y usuario se pueden ver a simple vista con wireshark ya que esta misma no se encuentra encriptada.



**2.2.- Analizar el tráfico HTTP con Wireshark. Identificar el usuario y password. Analizar capas 3, 4 y 5.**

En capa 3 tenemos la dirección de origen y destino del paquete, luego en capa 4 vemos el potocolo de transpote utilizado (TCP) y el puerto utilizado, y por ultimo en capa 5 tiene información para la aplicación, como el content type, longitud de la web UTF8 y otros datos utilizados para mostrar la pagina web al usuario.

# Bibliografía

Configuración de DNS

<http://mixeduperic.com/ubuntu/seven-easy-steps-to-setting-up-an-interal-dns-server-on-ubuntu.html>