



INSTITUTO POLITÉCNICO NACIONAL  
ESCUELA SUPERIOR DE CÓMPUTO

## Práctica 1 - Instalación de Netbeans

Unidad de aprendizaje: Redes de computadoras

Grupo: 2CM10

*Alumnos(a):*

Nicolás Sayago Abigail  
Ramos Díaz Enrique

*Profesor(a):*

Moreno Cervantes Axel

26 de Febrero 2017

# Índice

<b>1</b>	<b>Introducción</b>	<b>2</b>
<b>2</b>	<b>Marco teórico</b>	<b>2</b>
2.1	Dirección MAC	2
2.2	Formato básico de una trama Ethernet	3
<b>3</b>	<b>Parte I - Biblioteca en NetBeans</b>	<b>4</b>
<b>4</b>	<b>Parte II - MAC destino y MAC origen</b>	<b>11</b>
<b>5</b>	<b>Funcionamiento</b>	<b>12</b>
<b>6</b>	<b>Conclusiones</b>	<b>14</b>
	<b>Referencias</b>	<b>14</b>



## 2.2. Formato básico de una trama Ethernet

A continuación mostramos su estructura básica, la longitud mínima que puede tener es de 1518 Bytes, y la longitud máxima es de 64 Bytes.



Donde:

- **Preámbulo**

Secuencia de valores alternados 1 y 0 usados para la sincronización y para detectar la presencia de señal, indica el inicio de la trama.

- **Dirección de destino**

Este campo identifica la dirección MAC del dispositivo que debe recibir la trama. La dirección de destino puede especificar una dirección individual o una dirección multicast destinada a un grupo de estaciones. Una dirección destino con todos los bits en 1 se refiere a todos los dispositivos de la red denominada dirección de broadcast o difusión. Del byte 1 al byte 6.

- **Dirección de origen**

Este campo identifica la dirección MAC del dispositivo que debe enviar la trama. Del byte 7 al byte 12

- **Tipo**

Indica el tipo de protocolo de capa superior. Del byte 13 al byte 14.

- **Datos**

Este campo contiene los datos transferidos desde el origen hasta el destino. El tamaño máximo de este campo es de 1500 bytes. Si el tamaño de este campo es menor a 46 bytes, entonces es necesario el uso del campo siguiente (Pad) para añadir bytes hasta que el tamaño de la trama alcance el valor mínimo. Inicia en el byte 15 y se tomará hasta la longitud que se obtiene al multiplicar el último número a la derecha del byte 15 por 4. Generalmente son 20 posiciones, es decir, del byte 15 al byte 34.

- **FSC**

Campo de comprobación de la trama, este campo contiene un valor de chequeo de redundancia de 4 bytes (CRC) para verificación de errores. La estación origen efectúa un cálculo y lo transmite como parte de la trama. Cuando la trama es recibida

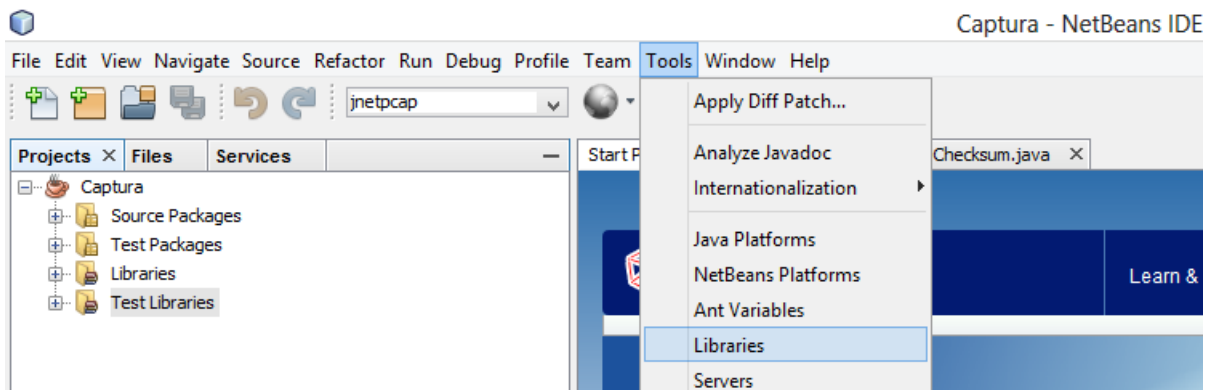
### 3. Parte I - Biblioteca en NetBeans

Para comenzar, es necesario tener instalado NetBeans y Winpcap. También será necesario que se tengan descargados los siguientes archivos, y que se encuentren en una carpeta conocida.

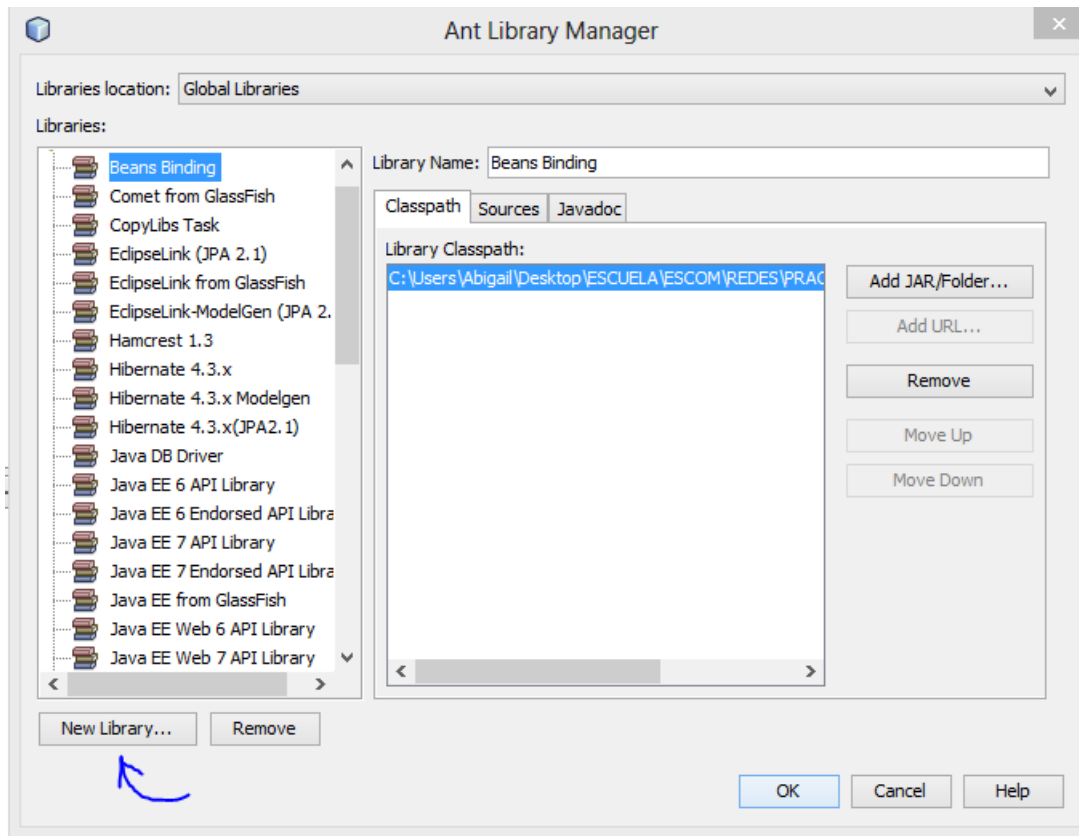
ESCUELA ► ESCOM ► REDES ► PRAC1				
	Nombre	Fecha de modifica...	Tipo	Tamaño
ar	Captura	04/03/2018 23:52	Carpeta de archivos	
	jnetpcap-1.3.0	01/04/2011 18:31	Carpeta de archivos	
	Reporte	10/03/2018 23:35	Carpeta de archivos	
	Captura	10/03/2018 21:37	Archivo JAVA	12 KB
	Checksum	04/03/2018 16:46	Archivo JAVA	3 KB
	instalacion_netbeans	21/02/2018 22:33	Adobe Acrobat D...	29 KB
	jnetpcap-1.3.0-1.win64	21/02/2018 23:29	Archivo WinRAR Z...	491 KB
	jnetpcap-javadoc-1.3.0-1	21/02/2018 22:33	Archivo WinRAR Z...	1,286 KB
	jnetpcap-src-1.3.0-1	21/02/2018 22:33	Archivo WinRAR Z...	1,670 KB

También es necesario tener un proyecto llamado **Captura**. (Omitimos ese paso de explicación).

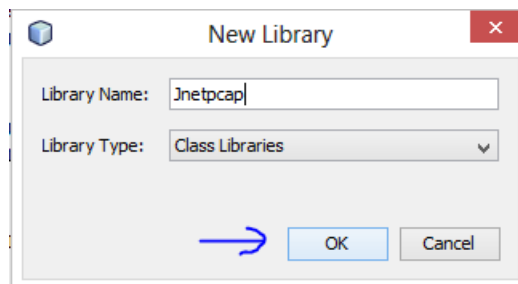
El siguiente paso es ir a la barra de herramientas y seleccionar **Tools / Libraries**.



Aparecera la siguiente ventana llamada **Ant Libray Manager**, donde podremos configurar nuestra biblioteca. Seleccionamos **New Library**.

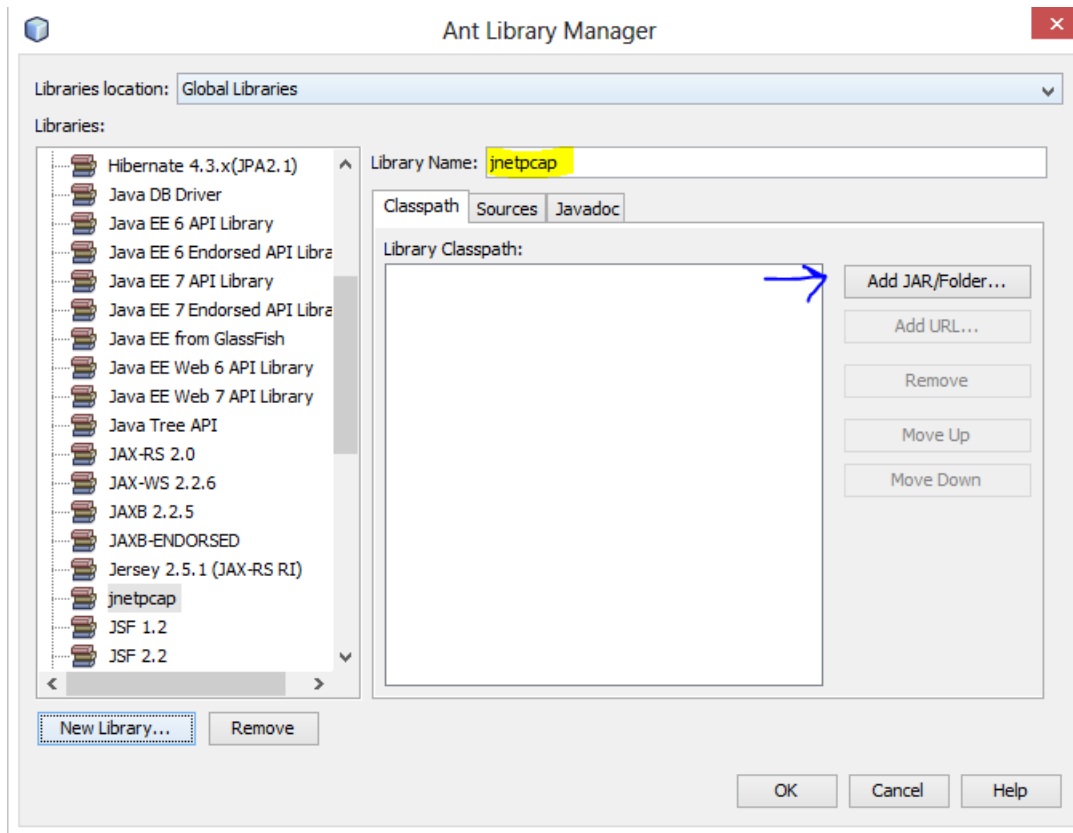


Podremos configurar el nombre de nuestra biblioteca a la que llamaremos **jnetpcap**.

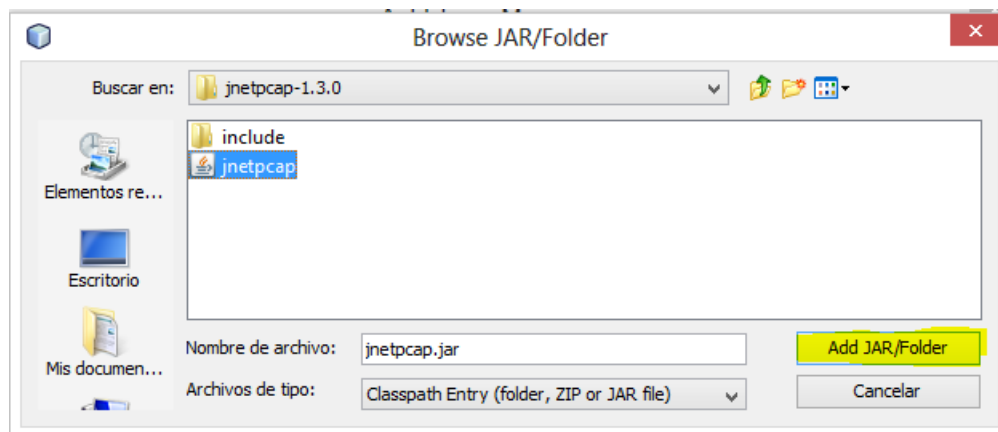


A continuación podremos configurar las tres direcciones que son : **Classpath**, **Sources** y **Javadoc**.

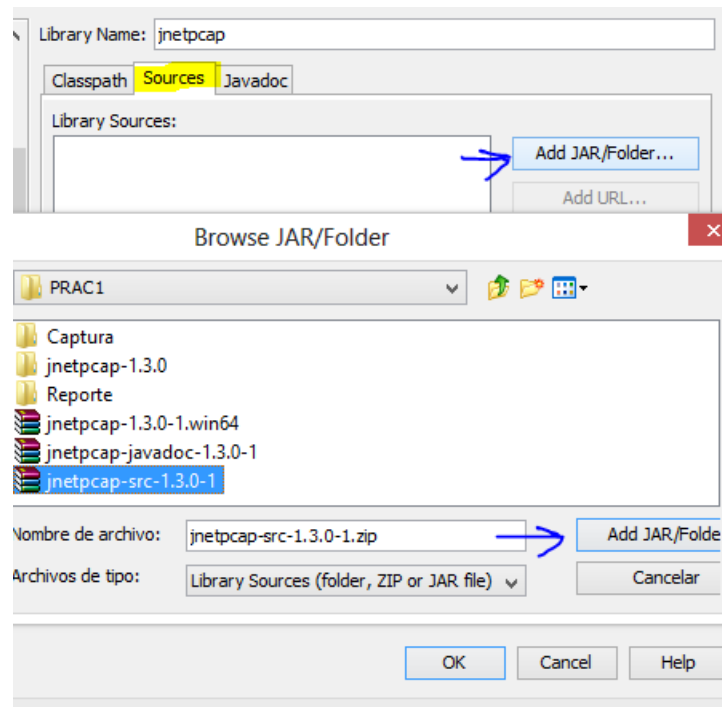
Empezamos con **Classpath**, vamos a colocar el archivo .jar, que anteriormente guardamos en una carpeta. Seleccionamos **Add JAR/Folder**.



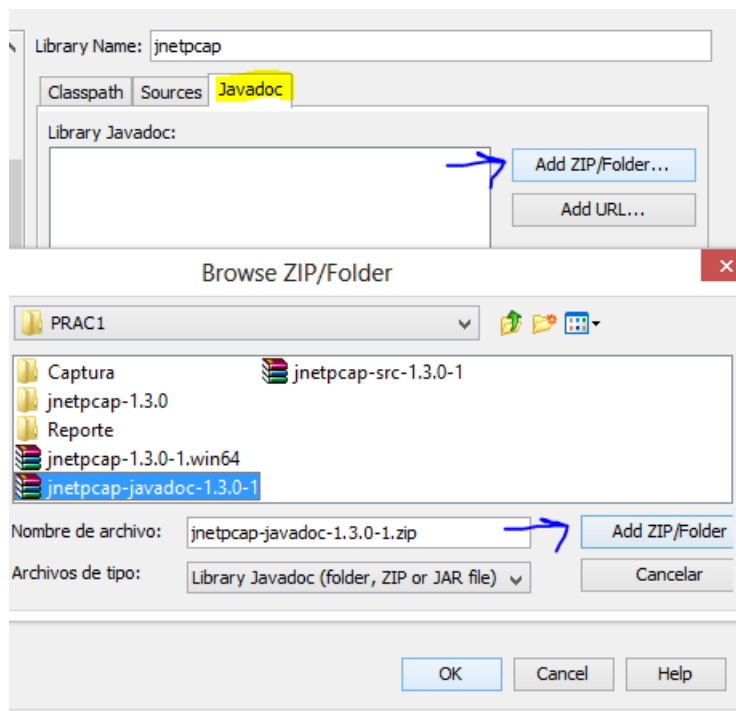
Podremos elegir nuestro archivo, así que nos dirigimos a la carpeta jnetpcap-1.3.0, y dentro de ella se encuentra el archivo **jnetpcap.jar**.



Hacemos algo exactamente igual que en el paso anterior, solo que ahora con **Sources**, y agregamos el archivo **jnetpcap-src-1.3.0-1.zip**.

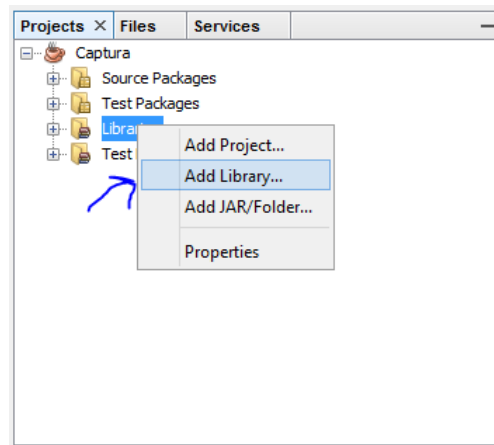


Y ahora con **Javadoc**, y agregamos el archivo **jnetpcap-javadoc-1.3.0-1.zip**.

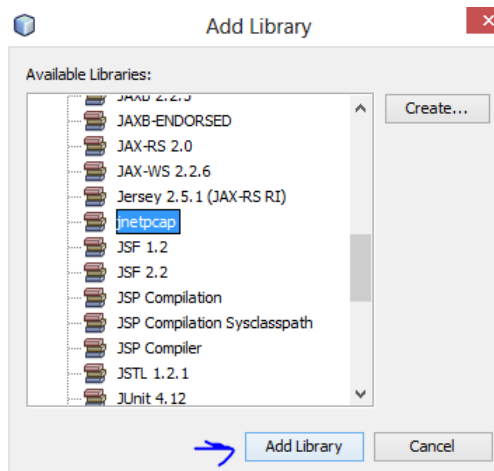




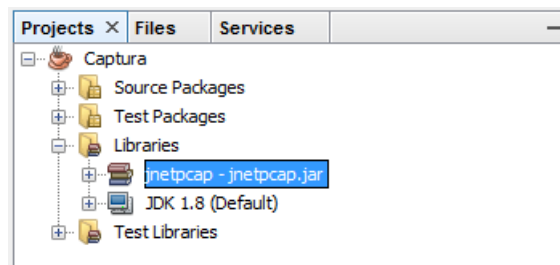
En el proyecto que hemos creado llamado **Captura**, iremos a **Libraries** y seleccionaremos **Add Library**.



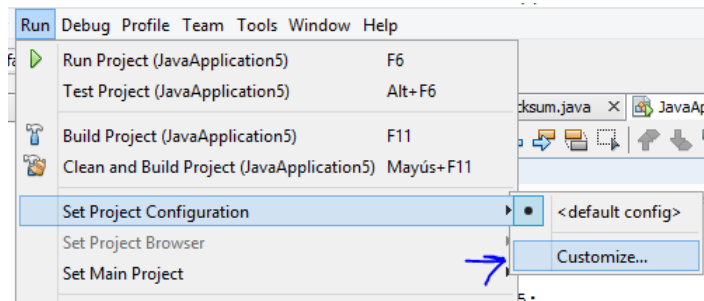
Nos desplegara una lista de bibliotecas, ahí buscaremos la que nosotros hemos creado. Entonces la agregamos **Add Libray**.



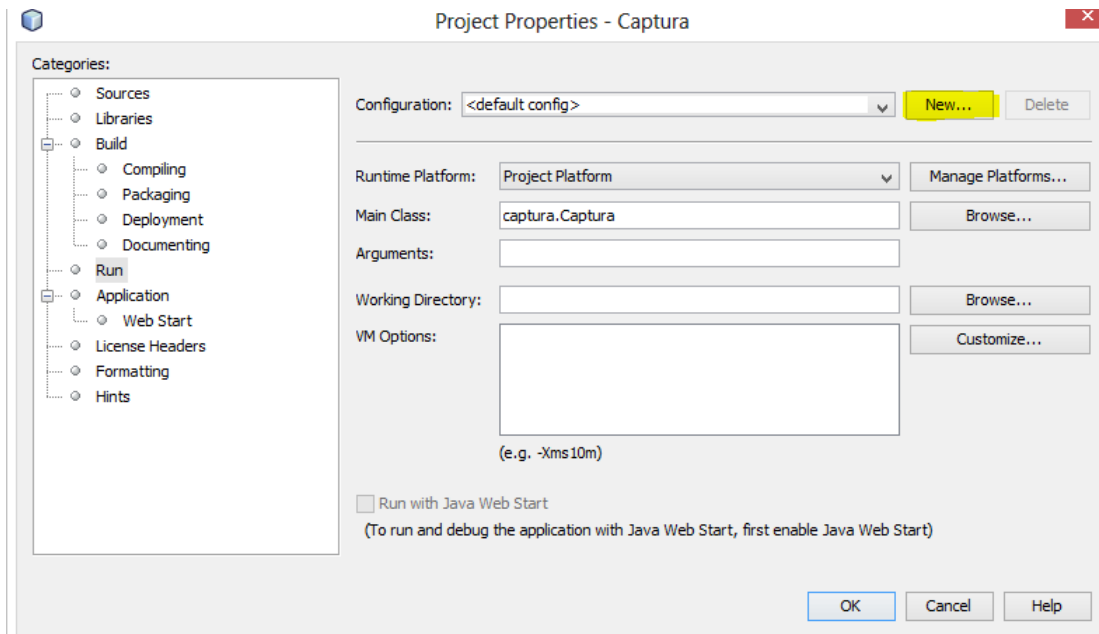
Ahora ya debe aparecer dentro de la carpeta de Library, nuestra biblioteca Jnetpcap.



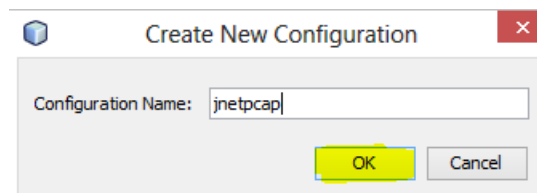
Para la siguiente configuración vamos a la barra de herramientas, en **Run / Set Project Configuration / Customize**



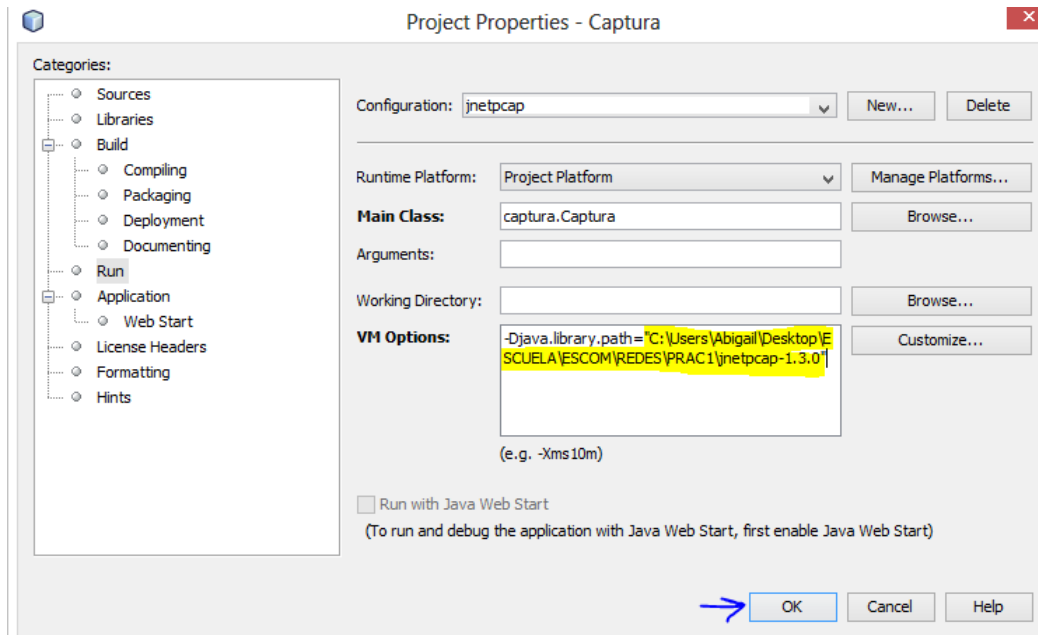
Se verá la siguiente ventana, en la cual seleccionaremos **New**.



De igual forma podemos configurar el nombre, en este caso jnetpcap.



Lo unico que haremos es poner exactamente lo mismo que esta marcado a continuación, lo que cambiara es la ruta donde tenemos el archivo "jnetpcap.jar"



Ese sería todo el proceso para agregar la libería y nuestros programas funcionen al usar la biblioteca jnetpcap.

## 4. Parte II - MAC destino y MAC origen

De acuerdo con el formato básico de una trama Ethernet que fue explicado en el marco teórico, la solución esta dada por simplemente recorrer el arreglo de bytes que estamos recibiendo.

```
1
2 System.out.println("\nTrama:");
3 for(int i=0;i<packet.size();i++)
4 {
5     System.out.printf("%02X ",packet.getUByte(i));
6     if(i%16==15)
7         System.out.println("");
8 }
9 System.out.println("\n\nEncabezado: \n"+ packet.toHexdump());
10
11 //Mostramos la MAC destino, la cual ocupa 6 Bytes
12 System.out.println("MAC destino:");
13 for(int i=0;i<6;i++)
14 {
15     System.out.printf("%02X ",packet.getUByte(i));
16 }
17 System.out.println("");
18
19 //Mostramos la MAC origen, la cual ocupa los siguientes 6 bytes
20 System.out.println("MAC origen:");
21 for(int i=6;i<12;i++)
22 {
23     System.out.printf("%02X ",packet.getUByte(i));
24 }
25 System.out.println("");
26
27 //Mostramos el tipo de protocolo
28 System.out.println("Tipo (hexadecimal):");
29 for(int i=12;i<14;i++)
30 {
31     System.out.printf("%02X ",packet.getUByte(i));
32 }
33 System.out.println("");
34 int tipo = (packet.getUByte(12)==0)?packet.getUByte(13):
35     (packet.getUByte(12)*256)+packet.getUByte(13);
36
37 System.out.printf("Tipo (decimal) = %d",tipo);
38 System.out.println("");
```

## 5. Funcionamiento

El programa muestra la trama, el encabezado, MAC destino, MAC origen y Tipo.

```

Output - Captura (run)

run:
Network devices found:
#0: \Device\NPF_{BE085CBE-F192-4DC6-A030-95D392F2E3F1} [Microsoft] MAC:[20:16:D8:EB:60:CE]
#1: \Device\NPF_{B2D03124-5E5F-486C-9B60-FE614920D5FD} [Microsoft] MAC:[20:16:D8:EB:60:CE]

Choosing 'Microsoft' on your behalf:

Paquete recibido el Sun Mar 11 17:18:18 CST 2018 caplen=42 len=42

Trama:
FF FF FF FF FF FF 0C 96 BF 99 BC 93 08 06 00 01
08 00 06 04 00 01 0C 96 BF 99 BC 93 C0 A8 01 FE
00 00 00 00 00 00 C0 A8 01 4B

Encabezado:
0000:*ff ff ff ff ff ff 0c 96 bf 99 bc 93 08 06*00 01 .....
0010: 08 00 06 04 00 01 0c 96 bf 99 bc 93 c0 a8 01 fe .....
0020: 00 00 00 00 00 00 c0 a8 01 4b* .....K

MAC destino:
FF FF FF FF FF FF
MAC origen:
0C 96 BF 99 BC 93
Tipo (hexadecimal):
08 06
Tipo (decimal) = 2054

```

```

Output - Captura (run)

Paquete recibido el Sun Mar 11 17:18:19 CST 2018 caplen=42 len=42

Trama:
FF FF FF FF FF FF 0C 96 BF 99 BC 93 08 06 00 01
08 00 06 04 00 01 0C 96 BF 99 BC 93 C0 A8 01 FE
00 00 00 00 00 00 C0 A8 01 4B

Encabezado:
0000:*ff ff ff ff ff ff 0c 96 bf 99 bc 93 08 06*00 01 .....
0010: 08 00 06 04 00 01 0c 96 bf 99 bc 93 c0 a8 01 fe .....
0020: 00 00 00 00 00 00 c0 a8 01 4b* .....K

MAC destino:
FF FF FF FF FF FF
MAC origen:
0C 96 BF 99 BC 93
Tipo (hexadecimal):
08 06
Tipo (decimal) = 2054

```

```

Output - Captura (run)

Paquete recibido el Sun Mar 11 17:18:20 CST 2018 caplen=54  len=54

Trama:
0C 96 BF 99 BC 93 20 16 D8 EB 60 CE 08 00 45 00
00 28 09 43 40 00 80 06 CA 8C C0 A8 01 4A 1F 0D
46 01 08 17 01 BB E6 51 C0 6A B2 0E 4F 59 50 11
00 40 D6 9C 00 00

Encabezado:
0000:*0c 96 bf 99  bc 93 20 16  d8 eb 60 ce  08 00*45 00  .....`...E.
0010: 00 28 09 43  40 00 80 06  ca 8c c0 a8  01 4a 1f 0d  -(.C@.....J.
0020: 46 01*08 17  01 bb e6 51  c0 6a b2 0e  4f 59 50 11  F.....Q.j..OYP.
0030: 00 40 d6 9c  00 00*      .@....

MAC destino:
0C 96 BF 99 BC 93
MAC origen:
20 16 D8 EB 60 CE
Tipo (hexadecimal):
08 00
Tipo (decimal) = 2048

```

```

Output - Captura (run)

Paquete recibido el Sun Mar 11 17:18:20 CST 2018 caplen=54  len=54

Trama:
20 16 D8 EB 60 CE 0C 96 BF 99 BC 93 08 00 45 70
00 28 17 21 40 00 5A 06 E2 3E 1F 0D 46 01 C0 A8
01 4A 01 BB 08 17 B2 0E 4F 59 E6 51 C0 6B 50 11
00 6F D6 6C 00 00

Encabezado:
0000:*20 16 d8 eb  60 ce 0c 96  bf 99 bc 93  08 00*45 70  ...`.....Ep
0010: 00 28 17 21  40 00 5a 06  e2 3e 1f 0d  46 01 c0 a8  -(.!@.Z...> .F...
0020: 01 4a*01 bb  08 17 b2 0e  4f 59 e6 51  c0 6b 50 11  .J.....OY.Q.kP.
0030: 00 6f d6 6c  00 00*      .o.l..

MAC destino:
20 16 D8 EB 60 CE
MAC origen:
0C 96 BF 99 BC 93
Tipo (hexadecimal):
08 00
Tipo (decimal) = 2048

```

## 6. Conclusiones

### ■ Nicolás Sayago Abigail

Al finalizar esta práctica pude entender mejor como están estructuradas las direcciones MAC, así como también comprender la importancia de éstas al ser utilizadas como identificadores. Otro aspecto interesante es el agregar las bibliotecas en el IDE Netbeans puesto que será de gran utilidad a lo largo del curso de esta unidad de aprendizaje.

De igual forma fue fácil comprender la estructura de las tramas que están siendo recibidas o enviadas.

### ■ Ramos Días Enrique

Ésta práctica es posible realizarla en dos lenguajes de programación: Java y C. Optamos por trabajar en Java debido al paradigma orientado a objetos, que nos ayudará a usar métodos ya implementados en nuestras clases. Utilizamos el IDE Netbeans como apoyo debido a su flexibilidad.

Primero fue necesario instalar los paquetes Jpcap y Winpcap en Windows. Luego de esto se creo una biblioteca en donde se agregaron todos los JARs necesarios para la ejecución de “Captura.java”, y finalmente se editó la path de Java para poder compilar correctamente.

Las MAC destino, MAC origen y Tipo fueron obtenidos gracias al análisis de la trama Ethernet.

## Referencias

- [1] E. Ariganello, *Redes Cisco. Guía de estudio para la certificación CCNA 640-802, 2da Edición*, 2011.