

Claudio Acuña, Guillermo Rojas, José Acuña,
Cristian Garrido Leonardo Jofré.

15 de noviembre de 2013

Índice

1. Introducción	3
2. Marco Teórico:	3
2.1. Medios de transmisión.	3
2.1.1. Simplex.	3
2.1.2. Half-duplex.	4
2.1.3. Full-duplex.	5
3. Objetivos:	5
3.1. Objetivos específicos:	5
4. Desarrollo	5
4.1. Construcción del cable cruzado	5
4.2. Distribuciones de S.O	9
4.2.1. Instalación del software para transferencia de datos (udpcast).	9
4.2.2. Cambiar el puerto ethernet a modo local para transferencia de datos.	9
4.2.3. Pruebas Realizadas.	10
4.3. Transferencia de Linux a Linux	12
4.4. Transferencia de Linux a Windows	13
4.5. Transferencia de Windows a Windows	14
5. Conclusión	16

1. Introducción

Al transmitir grandes cantidades de datos un usuario se va a encontrar con problemas al hacer uso del internet básicamente porque funciona en gran parte con arquitectura cliente servidor cuyos protocolos no están diseñados para la transmisión de grandes volúmenes de datos durante tiempos largos. En este laboratorio presentamos formas más horizontales (p2p) como seguras de transmisión de datos entre computadores punto a punto.

PALABRAS CLAVE: TRANSFERENCIA PUNTO A PUNTO, ENVÍO DE DATOS, CABLE CRUZADO.

2. Marco Teórico:

En arquitectura de computadores, el bus (o canal) es un sistema digital que transfiere datos entre los componentes de una computadora o entre computadoras. Está formado por cables o pistas en un circuito impreso, dispositivos como resistores y condensadores además de circuitos integrados.

En los primeros computadores electrónicos, todos los buses eran de tipo paralelo, de manera que la comunicación entre las partes del computador se hacía por medio de cintas o muchas pistas en el circuito impreso, en los cuales cada conductor tiene una función fija y la conexión es sencilla requiriendo únicamente puertos de entrada y de salida para cada dispositivo.

La tendencia en los últimos años se hacía uso de buses seriales como el USB, Firewire para comunicaciones con periféricos reemplazando los buses paralelos, incluyendo el caso como el del microprocesador con el chipset en la placa base. Esto a pesar de que el bus serial posee una lógica compleja (requiriendo mayor poder de cómputo que el bus paralelo) a cambio de velocidades y eficacias mayores.

Existen diversas especificaciones de que un bus se define en un conjunto de características mecánicas como conectores, cables y tarjetas, además de protocolos eléctricos y de señales.

2.1. Medios de transmisión.

Dependiendo de la forma de conducir la señal a través del medio, los medios de transmisión se pueden clasificar en dos grandes grupos: medios de transmisión guiados y medios de transmisión no guiados. Según el sentido de la transmisión podemos encontrarnos con tres tipos diferentes: simplex, half-duplex y full-duplex. También los medios de transmisión se caracterizan por utilizarse en rangos de frecuencia de trabajo diferentes.

2.1.1. Simplex.

Este modo de transmisión permite que la información discorra en un solo sentido y de forma permanente. Con esta fórmula es difícil la corrección de errores causados por deficiencias de línea (por ejemplo, la señal de TV).



Figura 1: Transmisión simplex

2.1.2. Half-duplex.

En este modo la transmisión fluye en los dos sentidos, pero no simultáneamente, sólo una de las dos estaciones del enlace punto a punto puede transmitir. Este método también se denomina en dos sentidos alternos (p. ej., el walkie-talkie).

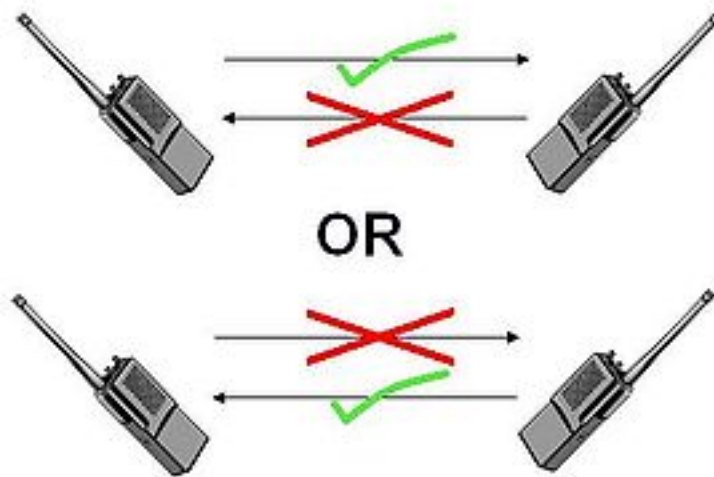


Figura 2: Transmisión half-duplex

2.1.3. Full-duplex.

Es el método de comunicación más aconsejable puesto que en todo momento la comunicación puede ser en dos sentidos posibles, es decir, que las dos estaciones simultáneamente pueden enviar y recibir datos y así pueden corregir los errores de manera instantánea y permanente.

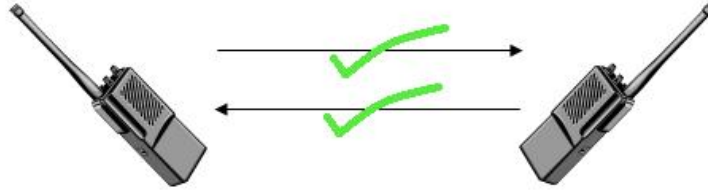


Figura 3: Transmisión full-duplex

3. Objetivos:

Al finalizar el trabajo el alumno estará capacitado para configurar y establecer una transmisión de datos punto a punto (P2P) empleando herramientas de software e interfaces de comunicación estándares en Linux.

3.1. Objetivos específicos:

- Conocer software de comunicación para realizar enlaces punto a punto simplex, half-duplex y duplex.
- Conocer interfaces de comunicaciones de la máquina como puertos seriales, paralelos, USB y NIC.

4. Desarrollo

4.1. Construcción del cable cruzado

Para poder realizar la comunicación de ambos equipos, implementamos un cable cruzado.

Para ello se utilizó la siguiente lista de materiales:

- Un cable utp categoría 5.
- Dos conectores rj45.
- Herramienta Crimpadora.

Según el estándar tanto el cable como el conector tiene los contactos numerados

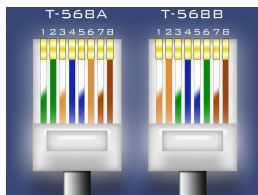


Figura 4: T-568A y T568B respectivamente

Luego procedemos a unir el cable con el conector siguiendo el siguiente esquema.

Extremo 1	Pin a pin	Extremo 2
1 Naranja y blanco	Pin 1 a Pin 3	1 Verde y Blanco
2 Naranja	Pin 2 a Pin 6	2 Verde
3 Verde y blanco	Pin 3 a Pin 1	3 Naranja y blanco
4 Azul	Pin 4 a Pin 4	4 Azul
5 Azul y blanco	Pin 5 a Pin 5	5 Azul Blanco
6 Verde	Pin 6 a Pin 2	6 Naranja
7 Marrón y blanco	Pin 7 a Pin 7	7 Marrón y blanco
8 Marrón	Pin 8 a Pin 8	8 Marrón

Figura 5: Extremos de Cable Cruzado

A través de la siguiente secuencia de fotos, se explicará como realizar las conexiones pertinentes.



Figura 6: Corte de la funda protectora a unos 4 cm



Figura 7: Se procede a su retiro.

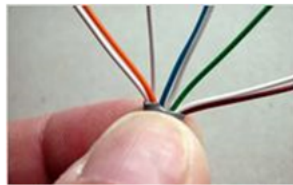


Figura 8: Desenrollo de los 4 pares de cable trenzados



Figura 9: Orden de los cables



Figura 10: Estiramiento de los cables, sin que estos cambien de posición

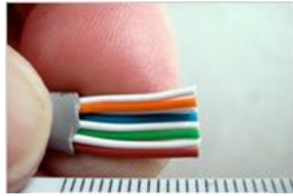


Figura 11: Posición Paralela



Figura 12: Corte de aprox. 14 y 15 mm

Luego se introducen dentro del conector, sin aflojar la presión sobre el extremo de la camisa, vigilando que cada uno entre por su carril. (Figura 8)



Figura 13: Se corta la funda protectora a unos 4 CM

Hasta que los cables llegan a tope al final de los carriles.



Figura 14: Conector a herramienta crimpadora Rj45

Inmediatamente sujetando el cable muy cerca del conector, apretando la camisa gris sobre los cables interiores, para que no se desplacen, se introduce el conector en la herramienta de crimpar.

4.2. Distribuciones de S.O

Distribuciones Usadas:

- Ubuntu 12.04 Raring Ringtail.
- Linux Mint 15 Olivia.
- Windows 7.

4.2.1. Instalación del software para transferencia de datos (udpcast).

Lo primero que se realizó, una vez asignados los equipos, fue instalar Ubuntu 12.04 y Linux Mint 15. Para ello, se descargó una imagen ISO de la página oficial y se utilizó el programa “unetbootin” para crear un ejecutable en un pendrive.

Una vez instalado el sistema operativo Ubuntu y Mint, se actualizaron los repositorios y se cargaron dichas actualizaciones en el equipo. Esto fue realizado abriendo una terminal (ctrl+alt+t), y escribiendo lo siguiente:

```
SUDO APT-GET UPDATE
```

y luego,

```
SUDO APT-GET UPGRADE
```

Una vez actualizado el sistema operativo, se procedió a instalar udpcast, mediante terminal:

```
SUDO APT-GET INSTALL UDPCAST
```

Para usar udpcast en windows es necesario descargar dos archivos:

- UDP-RECEIVER.EXE
- UDP-SENDER.EXE

Estos se pueden descargar de :

<http://www.udpcast.linux.lu/exe.html>

4.2.2. Cambiar el puerto ethernet a modo local para transferencia de datos.

- Se procede a realizar “click derecho” sobre el equipo de la red que esta ubicado en la esquina inferior derecha.



Figura 15: Barra de tarea en escritorio Cinnamon

- A continuación el menú indicado se procede a seleccionar “preferencia de redes”—>”Cableada”—>”opciones ”—>”Ajustes de IPV4” y a continuación en este punto en la opción Método se elige la opción (Manual).

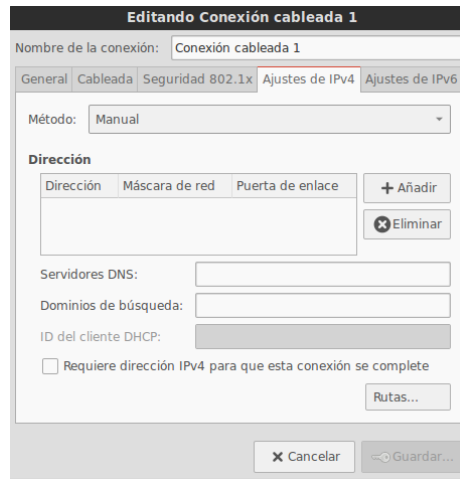


Figura 16: Menu de Cofiguración de redes Linux Mint 15

- Se selecciona Añadir y se completa en la sección de dirección con los siguientes datos.

1. Dirección: Se llena con la dirección ip del computador en uso
2. Máscara de red: se procede a llenar con la MAC correspondiente
3. Puerta de enlace: se coloca la dirección ip con cual se quiere hacer puente.
4. Enseguida se selecciona guardar, y con esto queda completamente enlazado los dos computadores a través de el cable cruzado para realizar las pruebas correspondientes con udpcast a través líneas de comandos.

4.2.3. Pruebas Realizadas.

- Primer lugar se realizó una prueba de testeo para visualizar la conexión de los computadores donde se procedió a verificar el ping de ambos.

PING 169.254.10.154 (PC QUE MANDA ARCHIVOS)

PING 169.254.10.154 (169.254.10.154) 56(84) BYTES OF DATA.

64 BYTES FROM 169.254.10.154: ICMP_REQ=1 TTL=64 TIME=0.878 MS

64 BYTES FROM 169.254.10.154: ICMP_REQ=2 TTL=64 TIME=0.486 MS

64 BYTES FROM 169.254.10.154: ICMP_REQ=3 TTL=64 TIME=0.473 MS

64 BYTES FROM 169.254.10.154: ICMP_REQ=4 TTL=64 TIME=0.521 MS

64 BYTES FROM 169.254.10.154: ICMP_REQ=5 TTL=64 TIME=0.486 MS

64 BYTES FROM 169.254.10.154: ICMP_REQ=6 TTL=64 TIME=0.492 MS
64 BYTES FROM 169.254.10.154: ICMP_REQ=7 TTL=64 TIME=0.496 MS
64 BYTES FROM 169.254.10.154: ICMP_REQ=8 TTL=64 TIME=0.526 MS

— 169.254.10.154 PING STATISTICS —

8 PACKETS TRANSMITTED, 8 RECEIVED, 0 % PACKET LOSS, TIME 6999MS

RTT MIN/AVG/MAX/MDEV = 0.473/0.544/0.878/0.130 MS

PING 169.254.7.171 (PC QUE RECIBE ARCHIVOS)

PING 169.254.7.171 (169.254.7.171) 56(84) BYTES OF DATA.

64 BYTES FROM 169.254.7.171: ICMP_REQ=1 TTL=64 TIME=0.291 MS
64 BYTES FROM 169.254.7.171: ICMP_REQ=2 TTL=64 TIME=0.355 MS
64 BYTES FROM 169.254.7.171: ICMP_REQ=3 TTL=64 TIME=0.368 MS
64 BYTES FROM 169.254.7.171: ICMP_REQ=4 TTL=64 TIME=0.491 MS
64 BYTES FROM 169.254.7.171: ICMP_REQ=5 TTL=64 TIME=0.514 MS
64 BYTES FROM 169.254.7.171: ICMP_REQ=6 TTL=64 TIME=0.420 MS
64 BYTES FROM 169.254.7.171: ICMP_REQ=7 TTL=64 TIME=0.498 MS
64 BYTES FROM 169.254.7.171: ICMP_REQ=8 TTL=64 TIME=0.475 MS
64 BYTES FROM 169.254.7.171: ICMP_REQ=9 TTL=64 TIME=0.345 MS
64 BYTES FROM 169.254.7.171: ICMP_REQ=10 TTL=64 TIME=0.330 MS
64 BYTES FROM 169.254.7.171: ICMP_REQ=11 TTL=64 TIME=0.483 MS
64 BYTES FROM 169.254.7.171: ICMP_REQ=12 TTL=64 TIME=0.469 MS
64 BYTES FROM 169.254.7.171: ICMP_REQ=13 TTL=64 TIME=0.469 MS

— 169.254.7.171 PING STATISTICS —

13 PACKETS TRANSMITTED, 13 RECEIVED, 0 % PACKET LOSS, TIME 11997MS

RTT MIN/AVG/MAX/MDEV = 0.291/0.423/0.514/0.076

Terminada la prueba de ping se dispuso a realizar una análisis a las redes de una de las estaciones de trabajo para verificar el tipo de conexión realizada donde se muestra a continuación los resultados obtenidos a través del comando de linux:

```

$ ifconfig

eth0 LINK ENCAP:ETHERNET DIRECCIÓNHW E8:03:9A:05:28:F4
DIREC. INET:169.254.7.171 DIFUS.:169.254.255.255 MÃ|SC:255.255.0.0
DIRECCIÓN INET6: FE80::EA03:9AFF:FE05:28F4/64 ALCANCE:ENLACE
ACTIVO DIFUSIÓN FUNCIONANDO MULTICAST MTU:1500
MÉTRICA:1

PAQUETES RX:160 ERRORES:0 PERDIDOS:0 OVERRUNS:0 FRAME:0
PAQUETES TX:290 ERRORES:0 PERDIDOS:0 OVERRUNS:0 CARRIER:0

COLISIONES:0 LONG.COLATX:1000

BYTES RX:45871 (45.8 KB) TX BYTES:60061 (60.0 KB)

lo LINK ENCAP:BUCLE LOCAL
DIREC. INET:127.0.0.1 MÃ|SC:255.0.0.0
DIRECCIÓN INET6: ::1/128 ALCANCE:ANFITRIÓN
ACTIVO BUCLE FUNCIONANDO MTU:65536 MÉTRICA:1

PAQUETES RX:1988 ERRORES:0 PERDIDOS:0 OVERRUNS:0 FRAME:0
PAQUETES TX:1988 ERRORES:0 PERDIDOS:0 OVERRUNS:0 CARRIER:0

COLISIONES:0 LONG.COLATX:0

BYTES RX:229472 (229.4 KB) TX BYTES:229472 (229.4 KB)

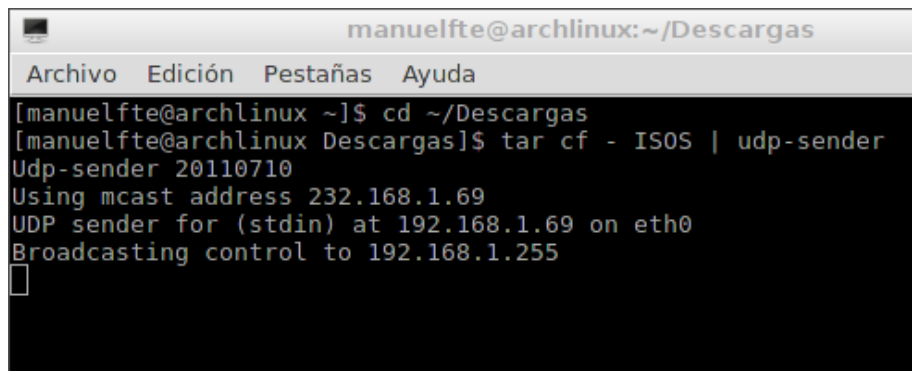
```

Finalmente se procedió a realizar una transferencia de directorios mediante `udpcast` utilizando los siguientes comandos:

4.3. Transferencia de Linux a Linux

- Antes de Realizar la transferencia se necesita tener conectado los computadores a través del cable cruzado y aplicar la configuración anteriormente descrita en la sección de “Cambios de ethernet a modo local”, donde a continuación se procede a escribir los siguientes comandos para la transferencia de archivos.

TAR CF - DIRECTORIO | UDP-SENDER



```
manuelfte@archlinux: ~/Descargas
Archivo Edición Pestañas Ayuda
[manuelfte@archlinux ~]$ cd ~/Descargas
[manuelfte@archlinux Descargas]$ tar cf - ISOS | udp-sender
Udp-sender 20110710
Using mcast address 232.168.1.69
UDP sender for (stdin) at 192.168.1.69 on eth0
Broadcasting control to 192.168.1.255
█
```

Figura 17: Comando escrito por el usuario que envía directorio

b. udp-receiver | tar xf -



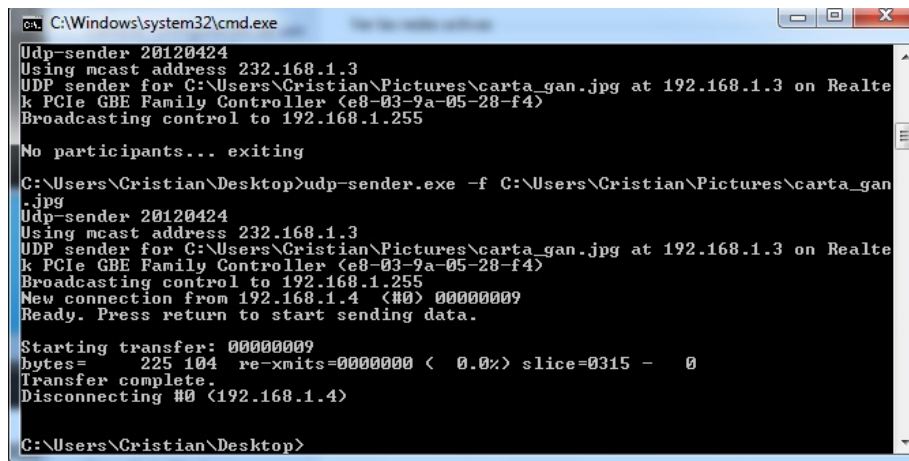
```
martha@debian: ~/Descargas
Archivo Edición Pestañas Ayuda
martha@debian:~$ cd ~/Descargas
martha@debian:~/Descargas$ udp-receiver | tar xf -
Udp-receiver 20100130
UDP receiver for (stdout) at 192.168.1.64 on eth0
received message, cap=00000009
Connected as #0 to 192.168.1.69
Listening to multicast on 232.168.1.69
Press any key to start receiving data!
Unexpected opcode 0700
Sending go signal
bytes= 37 259 040 ( 25.50 Mbps))
█
```

Figura 18: Comando escrito por el usuario que recibe directorio

4.4. Transferencia de Linux a Windows

Antes de transferir los archivos se deben editar las direcciones ip de ambos equipos, luego se abren las terminales de los equipos para transferir los archivos mediante linea de comandos.

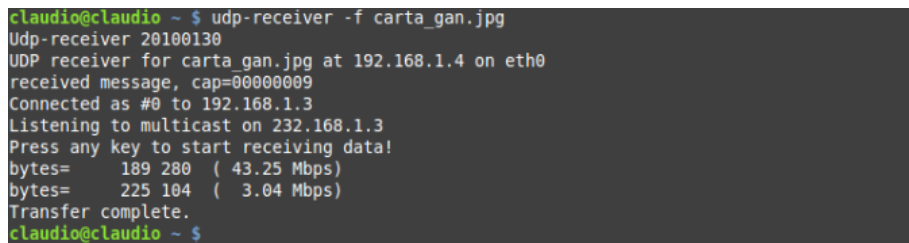
udp-sender.exe -f ruta y nombre del archivo



```
C:\Windows\system32\cmd.exe
Udp-sender 20120424
Using mcast address 232.168.1.3
UDP sender for C:\Users\Cristian\Pictures\carta_gan.jpg at 192.168.1.3 on Realtek PCIe GBE Family Controller {e8-03-9a-05-28-f4}
Broadcasting control to 192.168.1.255
No participants... exiting
C:\Users\Cristian\Desktop>udp-sender.exe -f C:\Users\Cristian\Pictures\carta_gan.jpg
Udp-sender 20120424
Using mcast address 232.168.1.3
UDP sender for C:\Users\Cristian\Pictures\carta_gan.jpg at 192.168.1.3 on Realtek PCIe GBE Family Controller {e8-03-9a-05-28-f4}
Broadcasting control to 192.168.1.255
New connection from 192.168.1.4 <#0> 00000009
Ready. Press return to start sending data.
Starting transfer: 00000009
bytes= 225 104 re-xmits=00000000 < 0.0%> slice=0315 - 0
Transfer complete.
Disconnecting #0 <192.168.1.4>
C:\Users\Cristian\Desktop>
```

Figura 19: Comando escrito por el usuario que envía directorio

udp-receiver -f nombre del archivo



```
claudio@claudio ~ $ udp-receiver -f carta_gan.jpg
Udp-receiver 20100130
UDP receiver for carta_gan.jpg at 192.168.1.4 on eth0
received message, cap=00000009
Connected as #0 to 192.168.1.3
Listening to multicast on 232.168.1.3
Press any key to start receiving data!
bytes= 189 280 ( 43.25 Mbps)
bytes= 225 104 ( 3.04 Mbps)
Transfer complete.
claudio@claudio ~ $
```

Figura 20: Comando escrito por el usuario que recibe directorio

4.5. Transferencia de Windows a Windows

- Configuración de la Ip por parte del receptor, previamente antes de recibir archivos para lograr la conexión a través de cable cruzado.

Lineas de comando para transferir archivos en windows

- udp-sender.exe -f ruta y nombre del archivo

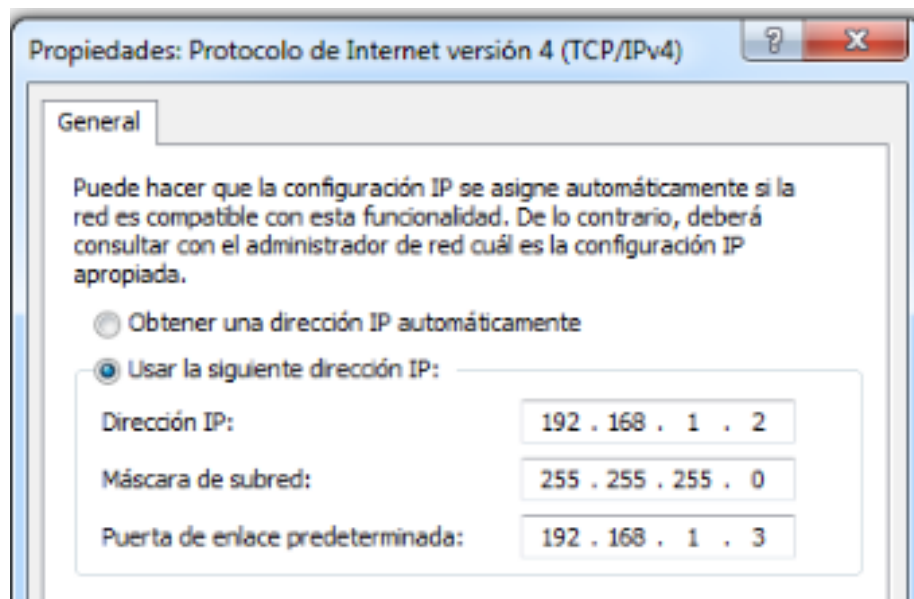


Figura 21: Pantalla de configuración

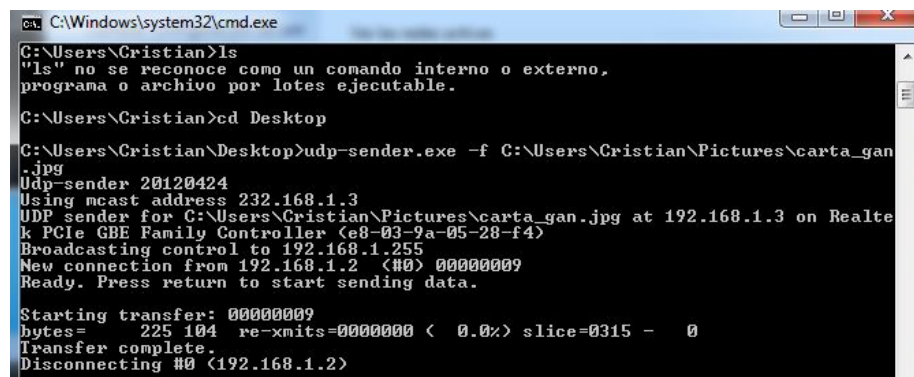


Figura 22: Usuario que envía archivo

Lineas de comando para recibir archivos en windows

- udp-receiver -f nombre del archivo

```

C:\Users\Jose\Documents\UTEM\Comunicacion>udp-receiver.exe -f carta_gan.jpg
Udp-receiver 20120424
UDP receiver for carta_gan.jpg at 192.168.1.2 on Realtek PCIe FE Family Controll
er <e0-db-55-98-26-e2>
received message, cap=00000009
Connected as #0 to 192.168.1.3
Listening to multicast on 232.168.1.3
Press return to start receiving data!
bytes=    225 104 <  2.84 Mbps>
Transfer complete.
C:\Users\Jose\Documents\UTEM\Comunicacion>

```

Figura 23: Usuario que recibe directorio

5. Conclusión

Al finalizar el trabajo, se ha aprendido resolver y configurar una transmisión de datos de punto a punto. Esto fue posible a la completa documentación de udpcast y la facilidad de realizarlo por terminal.

Además reconocer la importancia de la velocidad a la hora de transmisión de datos, debido que hoy en día, el mundo no espera.

Estudios más detallados serán hechos posteriormente con respecto a las comunicación entre máquinas en los puertos seriales y paralelos.

Referencias

- [1] Transmisión de Datos y la Red de Computadores. García Teodoro, Pedro; Díaz Verdejo, Jesús Esteban; López Soler, Juan Manuel; Ed. PEARSON, Pretice Hall. Edición: 2003.

[http://es.wikipedia.org/wiki/Telecomunicaciones\(2013\)](http://es.wikipedia.org/wiki/Telecomunicaciones(2013))