



UNIVERSIDAD TECNICA
FEDERICO SANTA MARIA



DEPARTAMENTO DE
ELECTRONICA

Proyecto final

ELO-322 Redes de Computadoras

SAMBA

Funcionamiento e implementación

Yeriel Paz – 201630042-2

Juan Pablo Sánchez – 201630002-3

Pascal Arriagada – 201630006-6

Profesor Agustín González

26 de junio 2017

Valparaíso, Chile

Resumen

¿Qué es *SMB*? ¿Qué es *Samba*? *SMB*, corresponde a un protocolo de red perteneciente a la capa de aplicación en el modelo *OSI*, desarrollado originalmente por la empresa IBM y continuado por Microsoft, que permite compartir archivos, información y/o equipos entre nodos de una red de computadores que usan principalmente un sistema operativo *Microsoft Windows*.

Mientras que *Samba*, es una implementación del protocolo *SMB*, junto con otros protocolos de Capa aplicación, orientado a sistemas operativos *Unix/GNU-Linux*, con la finalidad, al igual que el protocolo *SMB*, de compartir archivos e información con sistemas *Windows*.

De tal forma, el objetivo de este proyecto es implementar una conexión por *Samba* entre un sistema operativo *Unix* con *Windows* para compartir archivos entre ellos y mostrar cómo se puede realizar.

Introducción

Protocolo *SMB*, también llamado *Common Internet File System*, es un protocolo de la capa de aplicación que permite la conexión entre sistemas operativos, principalmente *Microsoft Windows*, para compartir archivos e información entre ellos. Por otro lado, *Samba*, es una implementación del protocolo *SMB*, destinada a conectar y compartir archivos entre sistemas operativos *Unix/GNU-Linux* y *Windows*, de preferencia dentro de una misma intranet.

De tal manera, según nuestro objetivo, lo que buscamos es realizar una conexión entre un sistema operativo *Debian* con *Windows*, compartiendo uno o varios archivos, además de realizar capturas con *Wireshark* para comprobar el éxito de la conexión mediante *SMB*. Además de explicar cómo se realiza la conexión en el caso de ambos sistemas.

¿Qué es SMB?

Bloque de Mensajes del Servidor (Server Message Block o SMB) es un protocolo de red, desarrollado por IBM en la década de los ochentas, que permite al usuario de una aplicación compartir archivos, discos, directorios, impresoras, puertos seriales y mail slots a través de una red, permitiendo de esta forma poder comunicarse con cualquier servidor, siempre y cuando este último se encuentre configurado para recibir una solicitud de un cliente *SMB*.

Actualmente, al protocolo *SMB* se le conoce como *Sistema Común de Archivos de Internet (Common Internet File System o CIFS)*, nombre que le asignó Microsoft en 1998 al registrar el protocolo. [Ref1]

¿Cómo funciona SMB?

SMB se basa en la estructura cliente-servidor, donde el cliente formula una solicitud y el servidor envía su respuesta. El servidor tiene su sistema de archivos y otros recursos, disponibles para los clientes sobre la red. Por su parte los clientes, independiente de su almacenamiento interno, pueden tener acceso al sistema de archivos e impresoras del servidor.

Los clientes se conectan al servidor usando *TCP/IP*, *NetBIOS* e *IPX/SPX* [ver anexo]. Una vez que la conexión está establecida, el cliente envía comandos (llamados *SMB's*) al servidor para trabajar con el sistema de archivos. [Ref2]

Estructura paquete SMB

Los paquetes *SMB* constan de una cabecera de 64 Bytes [Figura 2 en Anexo], un parámetro de tamaño variable y una sección para los datos. En el año 2006, Microsoft introdujo una nueva versión del protocolo, implementándola en *Windows Vista: SMB 2.0*, versión que es mayormente utilizada hoy en día en conexiones *SMB*, y versión del protocolo que se analizará.

SMB2, como se conoce a la versión 2.0, principalmente utiliza *TCP*, en el puerto 445, para el manejo de archivos, o *NetBIOS*, que a su vez utiliza *UDP*, en puertos 137 y 138, o *TCP*, en puertos 137 y 139. [Ver *Paquete SMB* en Anexo][Ref3]

¿Qué es Samba?

Samba es una implementación del protocolo *SMB* creado por Microsoft para sistemas *Unix/GNU-Linux* con la finalidad de compartir recursos con sistemas Windows.

Funcionamiento

Samba implementa los protocolos *NetBIOS* y *SMB*. A su vez utiliza dos **demonios** (programas que se ejecutan en segundo plano): *smbd* y *nmbd*: [Ref4]

- **Smbd (Samba Daemon)**: Permite la compartición de archivos e impresoras sobre una red SMB, y proporciona autenticación y autorización de acceso para clientes SMB; ofrece los dos modos de compartición de recursos existentes en Windows:
 - Modo basado en usuarios o modo *user* (propio de los dominios Windows NT o 2000)
 - Modo basado en recursos o modo *share* (propio de Windows 3.11/95)
- **Nmbd (Network Management DataBase)**: Permite que el sistema *Unix* participe en los mecanismos de resolución de nombres propios de Windows (WINS), lo que incluye:
 - Anuncio en el grupo de trabajo.
 - Gestión de la lista de ordenadores del grupo de trabajo.
 - Contestación a peticiones de resolución de nombres.
 - Anuncio de los recursos compartidos.



Implementación

Para implementar *Samba* se deben configurar los directorios de *Unix/GNU-Linux* como recursos compartidos a través de una red, para así permitir que los clientes puedan acceder a la información.

Instalación y configuración de usuario de Samba

- *Ubuntu* [Ref5]

Para instalar *samba* desde la terminal, se ha de hacer lo siguiente:

```
>$apt-get install samba samba-common python-glade2 system-config-samba
>$useradd usuariosamba -G sambashare
>$smbpasswd -a usuariosamba
```

Tras lo cual se ha de hacer la siguiente configuración:

```
>$vi /etc/samba/samba.conf

#===== Global Settings =====#
[global]
workgroup = WORKGROUP
server string = Samba %v
wins support = no
load printers = no

#===== Share Definitions =====#
[nombre de la carpeta a compartir]
comment = Informacion Compartida (apartado opcional)
path = Ruta absoluta de la carpeta
available = yes
browsable = yes
writable = no (Si se quiere dar permisos de escritura, dejar en "yes")

>$/etc/init.d/smbd (start/restart/stop/status)
//Solo debe ir uno de los comandos listados entre paréntesis, donde start: iniciar, restart:
reiniciar, stop: detener, status: estado.
```

Es importante destacar que la carpeta a compartir **debe** tener permisos de escritura y lectura en el grupo, lo cual se hace de la siguiente manera:

```
>$chmod -R 755 ~/carpeta a compartir
```

- *Windows* [Ref6]

Para facilitar el acceso al computador con *Debian*, lo registraremos en la lista de *hosts* permitidos por el sistema operativo:

```
notepad C:\Windows\System32\drivers\etc\hosts
```

Y añadimos una nueva línea al final del archivo como la siguiente:

```
192.168.1.100      debian.example.com  debian
```

Donde la dirección IP ha de ser la que tenga el dispositivo, el dominio es opcional, pero de ponerse ha de ser reemplazado por el dominio escogido para el servidor, y el nombre es para identificar a la máquina.

Si la edición fue exitosa, y el servidor se encuentra activo, se podrá ingresar a la carpeta compartida de la siguiente manera desde la ventana de comandos:

```
\\debian
```

O bien:

```
\\192.168.1.100
```

Donde nuevamente la IP ha de ser la que tenga el dispositivo al que se quiere acceder. (Ver figura 1)

Como observación cabe destacar que *Samba* está pensado para acceder a dispositivos dentro de una misma *intranet*, por lo que de esta manera fue probado.

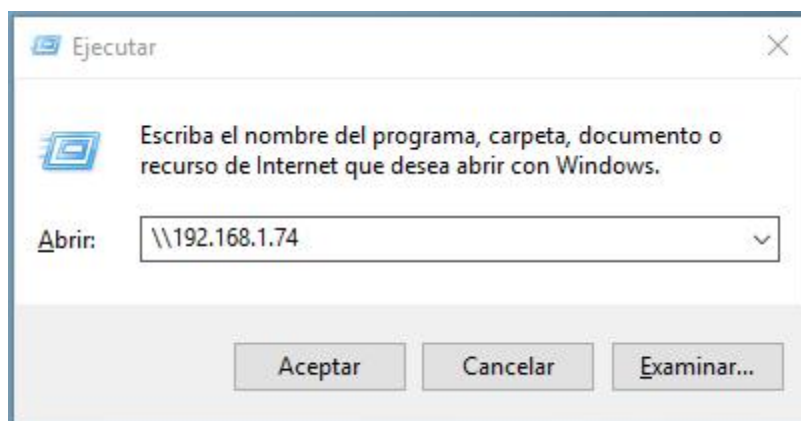


Figura 1

Conclusión

La aplicación *Samba* es el perfecto ejemplo de una implementación de protocolos y servicios buscando favorecer a los usuarios y permitir la interconexión de equipos de diferentes sistemas operativos, esto mediante una interpretación libre de un protocolo ya establecido, generando una solución para un problema existente en base a algo ya creado.

El desarrollo de soluciones para problemáticas como esta, es una rama de investigación común en las ciencias de la computación, donde lo que se hace es aprovechar el material ya existente y perfeccionarlo hacia el área que se encuentre en déficit o con conflictos, o perfeccionar la solución ya existente al corregir los posibles problemas que este mismo presente. Samba fue ideado como una solución de este tipo, solución que hasta el día de hoy es utilizada y sigue siendo actualizada, buscando así permitir un libre uso de las redes y del manejo de la información, independiente de las posibles limitaciones entre sistemas operativos.

Anexo

- **Cabecera paquete SMB [Ref 3]**

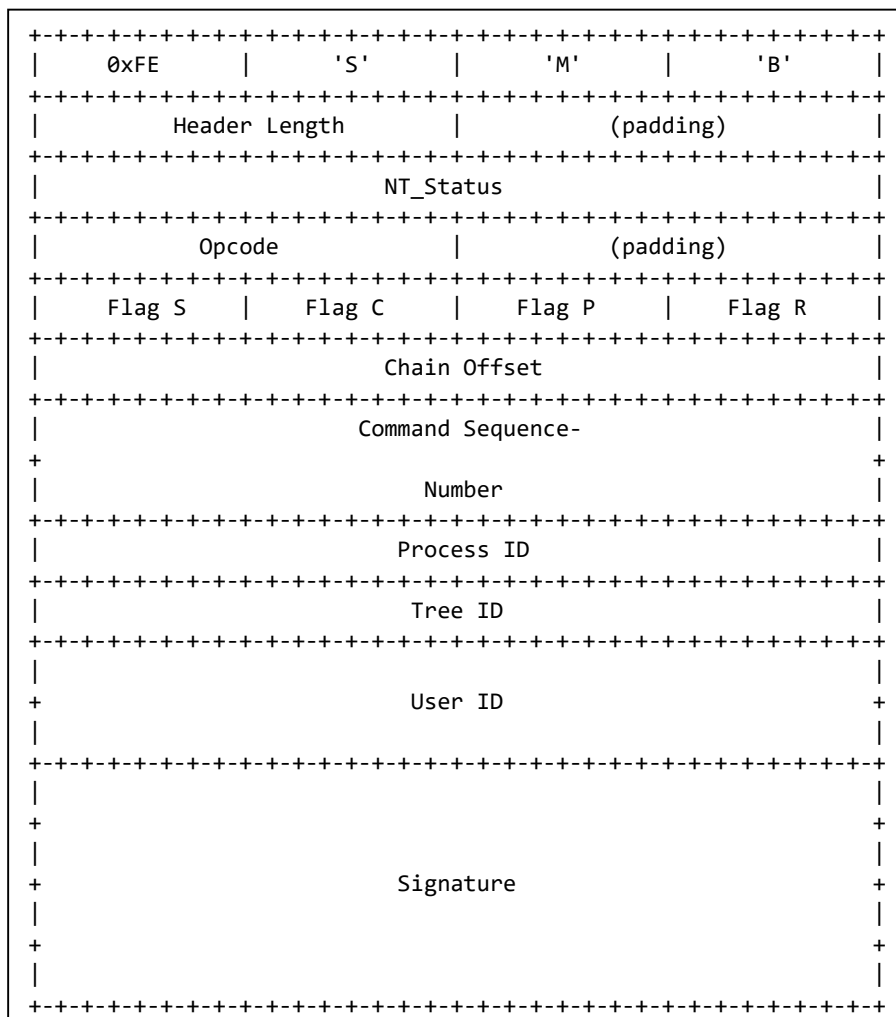


Figura 2

- **Header Length:** Largo total del header *SMB*, su tamaño mínimo es de 64 Bytes.
- **NT_Status:** Código de error (Para más detalle visitar: <https://msdn.microsoft.com/en-us/library/ee441884.aspx>)
- **Opcode:** Código de tipo del paquete *SMB* (Para más detalle visitar: <https://msdn.microsoft.com/en-us/library/ee441616.aspx>)

- **Flags:**
 - **R:** *Response Flag*, 1 para respuesta, 0 para petición.
 - **P:** *PID valid*, 1 si el *PID (Packet ID)* es válido, 0 si no lo es.
 - **C:** *End of Chain*, 1 si es el último *PDU (Protocol Data Unit)* en la cadena.
 - **S:** *Signature Present*, 1 si la firma está presente, 0 si no lo está.
- **Chain Offset:** Conexión al paquete siguiente.
- **Command Sequence Number:** Número a utilizar por *TCP* para enlazar las peticiones a las respuestas obtenidas. (Para más detalle visitar: <https://wiki.wireshark.org/SMB2/NegotiateProtocol>)
- **Process ID:** Valor que ha de ser iniciado en 0 por el cliente y que el servidor modifica.
- **Tree ID:** Valor de tipo entero que identifica el tipo de conexión *SMB* (ej. 0x03 para Impresora) (Para más detalle visitar: <https://wiki.wireshark.org/SMB2/TreeConnect>)
- **User ID:** Identifica la conexión establecida.
- **Signature:** Valor de comprobación del paquete.
- **TCP/IP:** *Transmission Control Protocol/Internet Protocol* o *Protocolo de Control de Transmisión/Protocolo de Internet*, Protocolo de comunicación en redes privadas, que se encarga de fragmentar el paquete en paquetes de menor tamaño para poder enviarlos, tal que el receptor ensambla nuevamente el paquete original. Se considera un modelo *cliente/servidor* por lo que utiliza una comunicación principalmente *punto-a-punto* [Ref 7].
- **NetBIOS:** *Network Basic Input/Output System*, es una especificación de interfaz para acceso a servicios de red, realizado para enlazar un sistema operativo de red con hardware específico. Principalmente es utilizado para permitir comunicación entre aplicaciones de diferentes computadoras dentro de una misma *LAN* [Ref 8].
- **IPX/SPX:** *Internetwork Packet Exchange/Sequenced Packet Exchange*, se compone de dos protocolos:

- *IPX*: Intercambio de Paquetes Entre Redes, implementa el protocolo *IDP (internet Datagram Protocol)*, siendo así un protocolo de datagramas rápido orientado a comunicaciones sin conexión, incluyendo en cada paquete la dirección de destino.
- *SPX: Intercambio de Paquetes en Secuencia*, implementa el protocolo *SPP (Sequenced Packet Protocol)*, siendo un protocolo basado en comunicaciones con conexión, encargándose de controlar la integridad de los paquetes y confirmar los paquetes recibidos. [Ref 9]

- **Captura Paquete *SMB2* con *Wireshark*:**

```

38 8.713825 192.168.1.74 192.168.1.82 SMB2 182 Close Response
Frame 38: 182 bytes on wire (1456 bits), 182 bytes captured (1456 bits) on
interface 0
Ethernet II, Src: IntelCor_0a:50:46 (78:0c:b8:0a:50:46), Dst:
HonHaiPr_9a:8b:cb (d8:5d:e2:9a:8b:cb)
Internet Protocol Version 4, Src: 192.168.1.74, Dst: 192.168.1.82
Transmission Control Protocol, Src Port: 445, Dst Port: 52491, Seq: 1593,
Ack: 1859, Len: 128
NetBIOS Session Service
SMB2 (Server Message Block Protocol version 2)
  SMB2 Header
    Server Component: SMB2
    Header Length: 64
    Credit Charge: 1
    NT Status: STATUS_SUCCESS (0x00000000)
    Command: Close (6)
    Credits granted: 1
    Flags: 0x00000031, Response, Priority
    Chain Offset: 0x00000000
    Message ID: 624
    Process Id: 0x0000feff
    Tree Id: 0x504ba4c1
    Session Id: 0x00000000c680a5e5
    Signature: 00000000000000000000000000000000
    [Response to: 37]
    [Time from request: 0.004104000 seconds]
  Close Response (0x06)
0000 d8 5d e2 9a 8b cb 78 0c b8 0a 50 46 08 00 45 00 .]....x...PF..E.
0010 00 a8 f5 c2 40 00 40 06 c0 a0 c0 a8 01 4a c0 a8 ....@.@.....J..
0020 01 52 01 bd cd 0b 7c c9 c0 5e 82 06 fb c5 50 18 .R....|..^....P.
0030 05 a6 b7 86 00 00 00 00 7c fe 53 4d 42 40 00 .....|.SMB@.
0040 01 00 00 00 00 00 06 00 01 00 31 00 00 00 00 .....1.....
0050 00 00 70 02 00 00 00 00 00 00 ff fe 00 00 c1 a4 ..p.....
0060 4b 50 e5 a5 80 c6 00 00 00 00 00 00 00 00 00 KP.....
0070 00 00 00 00 00 00 00 00 00 00 3c 00 00 00 00 .....

```

Referencias y Bibliografía

[Ref1] “*Just what is SMB?*”, Richard Sharpe, Recuperado el 16 de Junio 2016, de <https://www.samba.org/cifs/docs/what-is-smb.html>

[Ref2] “*Server Message Block (SMB) Version 2.0 Protocol Specification*”, Recuperado el 16 de Junio 2016, de <https://msdn.microsoft.com/en-us/library/cc212614.aspx>

[Ref3] “*Server Message Block versión 2 and 3*”, Recuperado el 16 de Junio 2016, de <https://wiki.wireshark.org/SMB2>

[Ref4] “*Samba: An Introduction*”, Chris Hertel, Recuperado el 16 de Junio 2016, de <https://www.samba.org/samba/docs/SambaIntro.html>

[Ref5] “*Cómo instalar y configurar Samba en Ubuntu 14.10*”, Recuperado el 18 de Junio 2016, de <https://ubunlog.com/como-instalar-y-configurar-samba-en/>

[Ref6] “*Samba Server Installation on Debian*”, Recuperado el 18 de Junio 2016, de <https://www.howtoforge.com/tutorial/debian-samba-server/>

[Ref 7] “*TCP/IP (Transmission Control Protocol/Internet Protocol)*”, Recuperado el 23 de Junio 2016, de <http://searchnetworking.techtarget.com/definition/TCP-IP>

[Ref 8] “*NetBIOS (Network Basic Input/Output System)*”, Recuperado el 23 de Junio 2016, de <http://searchnetworking.techtarget.com/definition/NetBIOS>

[Ref 9] “*IPX/SPX (Internetwork Packet Exchange/Sequenced Packet Exchange)*”, Recuperado el 23 de Junio 2016, de <http://www.linktionary.com/i/ipx.html>