

TAREA #987

ALUMNO: JAFET NILMAR SANGUINO COB

MATERIA: Sistemas operativos

PROFESOR: JIMENEZ SANCHEZ ISMAEL

GRUPO: 27AV Ingeniería en software

7° Séptimo cuatrimestre



UNIVERSIDAD
POLITÉCNICA
DE QUINTANA ROO

CAPTURAS DE PANTALLA DE EVIDENCIA

Practica de laboratorio Comandos en MSDOS

A) Anotar los comandos necesarios para ejecutar las siguientes instrucciones desde la consola de Ms-DOS

- 1.- Obtener la ayuda del comando ping
- 2.- Enviar un ping a 127.0.0.1 aplicando cualquier parametro
- 3.- Verificar la conectividad del equipo utilizando el comando ping, anotar conclusiones
- 4.- Obtener la ayuda del comando nslookup
- 5.- Resolver la direccion ip de <https://upgroo.edu.mx/> usando nslookup
- 6.- Hacer ping a la ip obtenida en el paso anterior, anotar conclusiones
- 7.- Obtener la ayuda del comando netstat
- 8.- Mostrar todas las conexiones y puertos de escucha
- 9.- Ejecutar netstat sin resolver nombres de dominio o puertos
- 10.- Mostrar las conexiones TCP
- 11.- Mostrar las conexiones UDP
- 12.- Utilizar el comando tasklist
- 13.- Utilizar el comando taskkill
- 14.- Utilizar el comando tracert
- 15.- Utilizar el comando ARP

B) Contesta con tus propias palabras las siguientes preguntas:

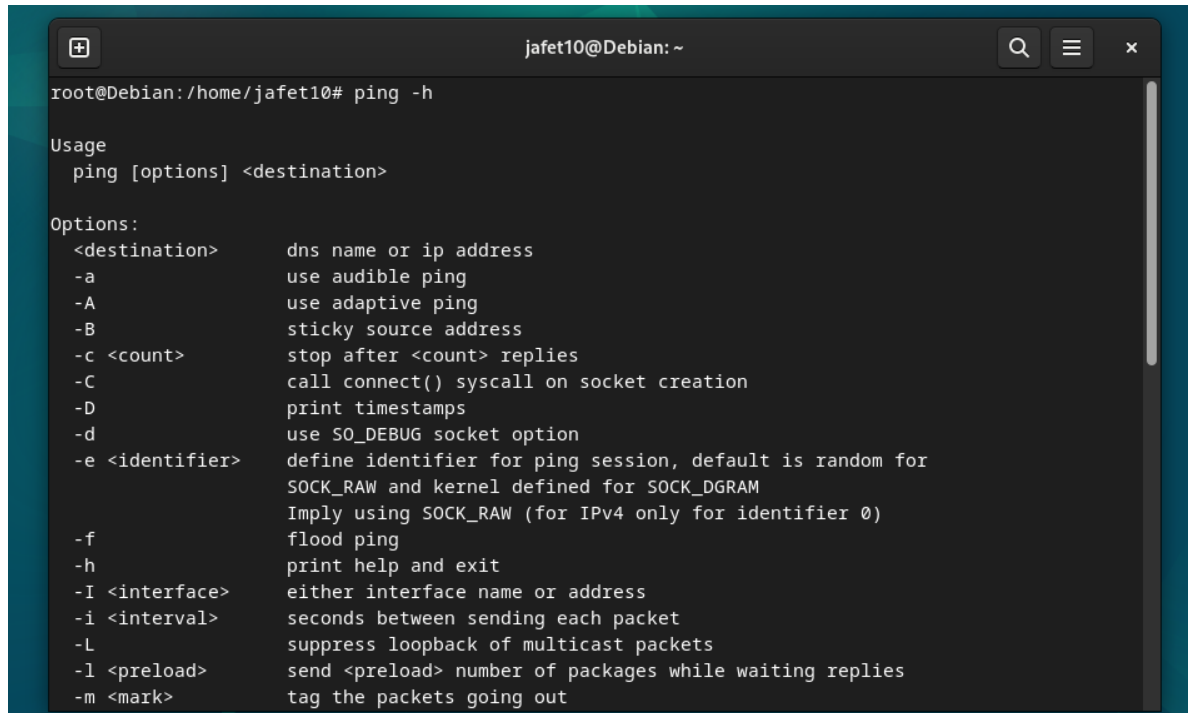
- 1.- ¿Para que sirve el comando ping?
- 2.- ¿Para que sirve el comando nslookup?
- 3.- ¿Para que sirve el comando netstat?
- 4.- ¿Para que sirve el comando tasklist?
- 5.- ¿Para que sirve el comando taskkill?
- 6.- ¿Para que sirve el comando tracert?
- 7.- ¿Como ayudan los primeros tres comandos para detectar problemas en la red?

C) Investigar los siguientes comandos y anotar ejemplos practicos:

atmadm, bitsadmin, cmstp, ftp, getmac, hostname, nbtstat, net, net use, netsh, pathping, rcp, rexec, route, rpcping, rsh, tcnsetup, telnet, tftp

A) Anotar los comandos necesarios para ejecutar las siguientes instrucciones desde la consola de Ms-DOS

1. Obtener la ayuda del comando ping

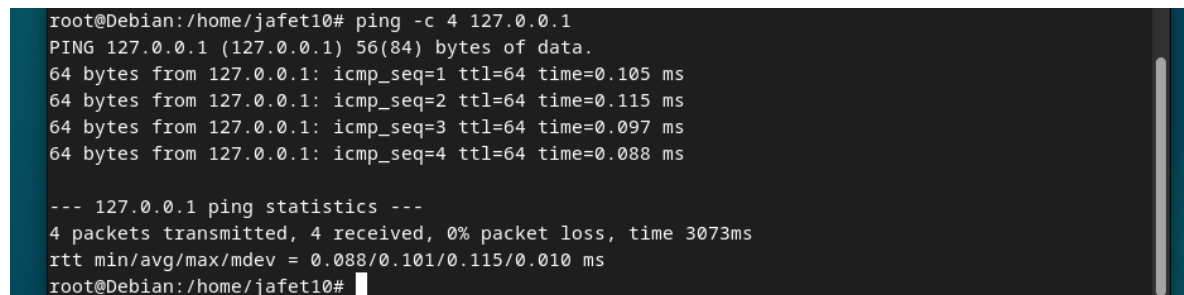
A terminal window titled 'jafet10@Debian: ~' with search, menu, and close icons in the top right. The command 'ping -h' has been executed, displaying the usage and options for the ping command.

```
root@Debian:/home/jafet10# ping -h

Usage
  ping [options] <destination>

Options:
  <destination>      dns name or ip address
  -a                  use audible ping
  -A                  use adaptive ping
  -B                  sticky source address
  -c <count>          stop after <count> replies
  -C                  call connect() syscall on socket creation
  -D                  print timestamps
  -d                  use SO_DEBUG socket option
  -e <identifier>     define identifier for ping session, default is random for
                      SOCK_RAW and kernel defined for SOCK_DGRAM
                      imply using SOCK_RAW (for IPv4 only for identifier 0)
  -f                  flood ping
  -h                  print help and exit
  -I <interface>      either interface name or address
  -i <interval>        seconds between sending each packet
  -L                  suppress loopback of multicast packets
  -l <preload>         send <preload> number of packages while waiting replies
  -m <mark>           tag the packets going out
```

2. Enviar un ping 127.0.0.1 aplicando cualquier parámetro

A terminal window showing the execution of the command 'ping -c 4 127.0.0.1'. It displays four successful ping responses with their respective sequence numbers, TTLs, and times. It also shows the ping statistics summary at the end.

```
root@Debian:/home/jafet10# ping -c 4 127.0.0.1
PING 127.0.0.1 (127.0.0.1) 56(84) bytes of data.
64 bytes from 127.0.0.1: icmp_seq=1 ttl=64 time=0.105 ms
64 bytes from 127.0.0.1: icmp_seq=2 ttl=64 time=0.115 ms
64 bytes from 127.0.0.1: icmp_seq=3 ttl=64 time=0.097 ms
64 bytes from 127.0.0.1: icmp_seq=4 ttl=64 time=0.088 ms

--- 127.0.0.1 ping statistics ---
4 packets transmitted, 4 received, 0% packet loss, time 3073ms
rtt min/avg/max/mdev = 0.088/0.101/0.115/0.010 ms
root@Debian:/home/jafet10#
```

3. Verificar la conectividad del equipo utilizando el comando del ping, anotar conclusiones.

```
root@Debian:/home/jafet10# ping www.google.com
PING www.google.com (142.250.189.132) 56(84) bytes of data.
64 bytes from mia09s26-in-f4.1e100.net (142.250.189.132): icmp_seq=1 ttl=117 time=29.0 ms
64 bytes from mia09s26-in-f4.1e100.net (142.250.189.132): icmp_seq=2 ttl=117 time=38.2 ms
64 bytes from mia09s26-in-f4.1e100.net (142.250.189.132): icmp_seq=3 ttl=117 time=54.8 ms
64 bytes from mia09s26-in-f4.1e100.net (142.250.189.132): icmp_seq=4 ttl=117 time=28.7 ms
64 bytes from mia09s26-in-f4.1e100.net (142.250.189.132): icmp_seq=5 ttl=117 time=25.7 ms
64 bytes from mia09s26-in-f4.1e100.net (142.250.189.132): icmp_seq=6 ttl=117 time=26.7 ms
64 bytes from mia09s26-in-f4.1e100.net (142.250.189.132): icmp_seq=7 ttl=117 time=28.9 ms
64 bytes from mia09s26-in-f4.1e100.net (142.250.189.132): icmp_seq=8 ttl=117 time=27.8 ms
64 bytes from mia09s26-in-f4.1e100.net (142.250.189.132): icmp_seq=9 ttl=117 time=35.9 ms
64 bytes from mia09s26-in-f4.1e100.net (142.250.189.132): icmp_seq=10 ttl=117 time=30.1 ms
64 bytes from mia09s26-in-f4.1e100.net (142.250.189.132): icmp_seq=11 ttl=117 time=34.2 ms
64 bytes from mia09s26-in-f4.1e100.net (142.250.189.132): icmp_seq=12 ttl=117 time=22.8 ms
64 bytes from mia09s26-in-f4.1e100.net (142.250.189.132): icmp_seq=13 ttl=117 time=22.2 ms
64 bytes from mia09s26-in-f4.1e100.net (142.250.189.132): icmp_seq=14 ttl=117 time=28.3 ms
64 bytes from mia09s26-in-f4.1e100.net (142.250.189.132): icmp_seq=15 ttl=117 time=34.7 ms
64 bytes from mia09s26-in-f4.1e100.net (142.250.189.132): icmp_seq=16 ttl=117 time=24.5 ms
64 bytes from mia09s26-in-f4.1e100.net (142.250.189.132): icmp_seq=17 ttl=117 time=24.7 ms
```

En conclusión: Demuestra que si existe la conectividad de mi equipo al servidor de Google.

4. Obtener la ayuda del comando nslookup

```
root@Debian:/home/jafet10# man nslookup
```

```
jafet10@Debian: ~
NSLOOKUP(1)                                BIND 9                                NSLOOKUP(1)

NAME
    nslookup - query Internet name servers interactively

SYNOPSIS
    nslookup [-option] [name | -] [server]

DESCRIPTION
    nslookup is a program to query Internet domain name servers. nslookup has two modes:
    interactive and non-interactive. Interactive mode allows the user to query name servers
    for information about various hosts and domains or to print a list of hosts in a do-
    main. Non-interactive mode prints just the name and requested information for a host
    or domain.

ARGUMENTS
    Interactive mode is entered in the following cases:

    a. when no arguments are given (the default name server is used);

    b. when the first argument is a hyphen (-) and the second argument is the host name or
       Internet address of a name server.

Manual page nslookup(1) line 1 (press h for help or q to quit)
```

5. Resolver la dirección ip <https://upqroo.edu.mx/> usando nslookup

```
root@Debian:/home/jafet10# nslookup upqroo.edu.mx
Server:      8.8.8.8
Address:     8.8.8.8#53

Non-authoritative answer:
Name:   upqroo.edu.mx
Address: 77.68.126.20

root@Debian:/home/jafet10#
```

6. Hacer ping a la ip obtenida en el paso anterior, anotar conclusiones.

```
root@Debian:/home/jafet10# ping -c 5 77.68.126.20
PING 77.68.126.20 (77.68.126.20) 56(84) bytes of data.
64 bytes from 77.68.126.20: icmp_seq=1 ttl=49 time=170 ms
64 bytes from 77.68.126.20: icmp_seq=2 ttl=49 time=127 ms
64 bytes from 77.68.126.20: icmp_seq=3 ttl=49 time=128 ms
64 bytes from 77.68.126.20: icmp_seq=4 ttl=49 time=246 ms
64 bytes from 77.68.126.20: icmp_seq=5 ttl=49 time=130 ms

--- 77.68.126.20 ping statistics ---
5 packets transmitted, 5 received, 0% packet loss, time 4195ms
rtt min/avg/max/mdev = 127.472/160.362/245.856/45.747 ms
root@Debian:/home/jafet10#
```

7. Obtener la ayuda del comando netstat.

```
root@Debian:/home/jafet10# netstat -h
usage: netstat [-vWeenNcCF] [<Af>] -r          netstat {-V|--version|-h|--help}
       netstat [-vWnNcaeol] [<Socket> ...]
       netstat { [-vWeenNac] -i | [-cnNe] -M | -s [-6tuw] }

-r, --route           display routing table
-i, --interfaces      display interface table
-g, --groups           display multicast group memberships
-s, --statistics       display networking statistics (like SNMP)
-M, --masquerade       display masqueraded connections

-v, --verbose         be verbose
-W, --wide            don't truncate IP addresses
-n, --numeric         don't resolve names
--numeric-hosts       don't resolve host names
--numeric-ports       don't resolve port names
--numeric-users       don't resolve user names
-N, --symbolic        resolve hardware names
-e, --extend          display other/more information
-p, --programs        display PID/Program name for sockets
-o, --timers          display timers
-c, --continuous     continuous listing
```

8. Mostrar todas las conexiones y puertos de escucha.

```
root@Debian:/home/jafet10# netstat -tuln
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
tcp        0      0 127.0.0.1:631           0.0.0.0:*               LISTEN
tcp6       0      0 :::1:631                :::*                   LISTEN
udp        0      0 0.0.0.0:631            0.0.0.0:*
udp        0      0 0.0.0.0:53984           0.0.0.0:*
udp        0      0 0.0.0.0:5353            0.0.0.0:*
udp6       0      0 :::41305                :::*
udp6       0      0 :::5353                 :::*
```

9. Ejecutar netstat sin resolver nombres de dominio o puertos.

```
root@Debian:/home/jafet10# ss -n
Netid State  Recv-Q Send-Q           Local Address:Port      Peer Address:Port
Process
u_str ESTAB  0      0                * 19973                  * 19974
u_str ESTAB  0      0                * 21032                  * 21621
u_str ESTAB  0      0                * 21505                  * 20865
u_str ESTAB  0      0                /run/user/1000/at-spi/bus 20820 * 20819
u_str ESTAB  0      0                * 19828                  * 19829
u_str ESTAB  0      0                /run/user/1000/wayland-0 20807 * 20737
u_str ESTAB  0      0                * 15036                  * 15457
u_str ESTAB  0      0                * 21606                  * 21013
u_str ESTAB  0      0                /run/systemd/journal/stdout 14990 * 14989
u_str ESTAB  0      0                * 20190                  * 20191
```

10. Mostrar las conexiones TCP

```
root@Debian:/home/jafet10# ss -tn
```

State	Recv-Q	Send-Q	Local Address:Port	Peer Address:Port	Process
-------	--------	--------	--------------------	-------------------	---------

11. Mostrar las conexiones UDP

```
root@Debian:/home/jafet10# ss -un
```

Recv-Q	Send-Q	Local Address:Port	Peer Address:Port	Process
0	0	10.0.2.15:enp0s3:68	10.0.2.2:67	

```
root@Debian:/home/jafet10#
```

12. Utilizar el comando tasklist

```
root@Debian:/home/jafet10# ps aux
```

USER	PID	%CPU	%MEM	VSZ	RSS	TTY	STAT	START	TIME	COMMAND
root	1	0.0	0.4	102396	12452	?	Ss	18:28	0:02	/sbin/init
root	2	0.0	0.0	0	0	?	S	18:28	0:00	[kthreadd]
root	3	0.0	0.0	0	0	?	I<	18:28	0:00	[rcu_gp]
root	4	0.0	0.0	0	0	?	I<	18:28	0:00	[rcu_par_gp]
root	5	0.0	0.0	0	0	?	I<	18:28	0:00	[slub_flushwq]
root	6	0.0	0.0	0	0	?	I<	18:28	0:00	[netns]
root	8	0.0	0.0	0	0	?	I<	18:28	0:00	[kworker/0:0H-events_highpri]
root	10	0.0	0.0	0	0	?	I<	18:28	0:00	[mm_percpu_wq]
root	11	0.0	0.0	0	0	?	I	18:28	0:00	[rcu_tasks_kthread]
root	12	0.0	0.0	0	0	?	I	18:28	0:00	[rcu_tasks_rude_kthread]
root	13	0.0	0.0	0	0	?	I	18:28	0:00	[rcu_tasks_trace_kthread]
root	14	0.0	0.0	0	0	?	S	18:28	0:00	[ksoftirqd/0]
root	15	0.0	0.0	0	0	?	R	18:28	0:01	[rcu_preempt]
root	16	0.0	0.0	0	0	?	S	18:28	0:00	[migration/0]
root	18	0.0	0.0	0	0	?	S	18:28	0:00	[cpuhp/0]
root	19	0.0	0.0	0	0	?	S	18:28	0:00	[cpuhp/1]
root	20	0.0	0.0	0	0	?	S	18:28	0:00	[migration/1]
root	21	0.0	0.0	0	0	?	S	18:28	0:00	[ksoftirqd/1]
root	22	0.0	0.0	0	0	?	I	18:28	0:01	[kworker/1:0-events]
root	23	0.0	0.0	0	0	?	I<	18:28	0:00	[kworker/1:0H-events_highpri]
root	26	0.0	0.0	0	0	?	S	18:28	0:00	[kdevtmpfs]
root	27	0.0	0.0	0	0	?	I<	18:28	0:00	[inet_frag_wq]

13. Utilizar el comando taskkill

```
jafet10 4407 5.5 1.3 391388 39952 ? S 23:48 0:00 /usr/bin/gr
jafet10 4408 4.8 1.6 565092 49808 ? Ssl 23:48 0:00 /usr/libexe
jafet10 4440 0.3 0.3 311592 10264 ? Sl 23:48 0:00 /usr/libexe
root 4461 1.1 0.2 16300 6776 ? Ss 23:48 0:00 /lib/system
jafet10 4486 3.9 0.8 341488 25860 ? Sl 23:48 0:00 /usr/bin/gr
jafet10 4571 0.3 0.1 8004 4828 pts/0 Ss 23:48 0:00 bash
_apt 4590 0.4 0.3 21820 9312 ? S 23:48 0:00 /usr/lib/ap
jafet10 4592 400 0.1 11216 4780 pts/0 R+ 23:48 0:00 ps aux
jafet10@Debian:~$ kill 4571
jafet10@Debian:~$
```

14. Utilizar el comando tracer

```
jafet10@Debian:~$ traceroute localhost
traceroute to localhost (127.0.0.1), 30 hops max, 60 byte packets
 1  localhost (127.0.0.1)  0.165 ms  0.015 ms  0.010 ms
jafet10@Debian:~$
```

15. Utilizar el comando ARP

```
jafet10@Debian:~$ arp
bash: arp: command not found
jafet10@Debian:~$ ip neigh
10.0.2.2 dev enp0s3 lladdr 52:54:00:12:35:02 REACHABLE
```

NOTA: aunque use apt update y luego apt install net-tools no me funciona el comando ARP

B) Contesta con tus propias palabras las siguientes preguntas:

1- ¿Para qué sirve el comando ping?

Sirve para enviar paquetes de datos a ese destino y esperar respuestas, lo que permite determinar si el destino está accesible y la calidad de la conexión.

2- ¿Para qué sirve el comando nslookup?

Sirve para obtener información sobre la resolución de nombres de dominio, lo que permite determinar la dirección IP asociada a un nombre de host o viceversa.

3- ¿Para qué sirve el comando netstat?

Sirve para obtener información relacionada con la red, como conexiones activas, tablas de enrutamiento, entre otras cosas.

4- ¿Para qué sirve el comando tasklist?

Sirve para mostrarnos información de los procesos que están corriendo actualmente en la máquina.

5- ¿Para qué sirve el comando taskkill?

Sirve para terminar un proceso en específico.

6- ¿Para qué sirve el comando tracer?

Sirve para rastrear la ruta que toman los paquetes de datos desde tu computadora hasta un destino específico en una red y mostrar los saltos intermedios.

7- ¿Cómo ayudan los primeros tres comandos para detectar problemas en la red?

Son esenciales para diagnosticar problemas de conectividad, problemas de DNS y problemas de red en general.

C) Investigue los siguientes comandos y anotar ejemplos prácticos

- **atmadm**

atmadm es una utilidad de línea de comandos en sistemas Windows que se utiliza para administrar y configurar servicios relacionados con la Asynchronous Transfer Mode (ATM), que es una tecnología de transmisión de datos. ATM se utiliza en redes de telecomunicaciones y proporciona una alta velocidad de transferencia de datos, especialmente en aplicaciones que requieren ancho de banda constante, como transmisión de video y voz.

Sin embargo, el comando atmadm se ha vuelto menos común en sistemas modernos y no suele ser necesario en entornos de usuario final típicos.

atmadm.exe

- **bitsadmin**

El comando bitsadmin es una utilidad en sistemas Windows que se utiliza para administrar trabajos y tareas de transferencia de archivos en segundo plano utilizando el servicio Background Intelligent Transfer Service (BITS). BITS es un servicio de Windows que permite la transferencia de archivos de manera eficiente, especialmente en entornos de red donde la conexión puede ser interrumpida o lenta.

- 1. Crear un nuevo trabajo de transferencia:**

Para crear un nuevo trabajo de transferencia, puedes usar el siguiente comando:

bitsadmin /create myjob

Esto crea un nuevo trabajo llamado "myjob". Puedes reemplazar "myjob" con el nombre que desees para tu trabajo.

- 2. Añadir archivos a la tarea:**

Una vez que has creado el trabajo, puedes agregar archivos que desees transferir. Por ejemplo:

bitsadmin /addfile myjob https://example.com/archivo.zip C:\ruta\local\archivo.zip

Esto agrega un archivo para su descarga. Debes proporcionar la URL remota del archivo y la ubicación local donde se almacenará.

- 3. Iniciar la tarea:**

Luego, puedes iniciar el trabajo de transferencia de archivos:

bitsadmin /resume myjob

Esto inicia la transferencia de archivos en segundo plano. BITS se encargará de descargar el archivo y administrar la transferencia, incluso si la conexión se interrumpe.

4. Monitorear el trabajo:

Puedes verificar el progreso del trabajo utilizando:

```
bitsadmin /list /verbose
```

Esto mostrará información sobre los trabajos en ejecución.

5. Cancelar o completar el trabajo:

Cuando el trabajo se ha completado o si deseas cancelarlo, puedes usar:

```
bitsadmin /complete myjob
```

Esto marcará el trabajo como completado y detendrá la transferencia.

- **cmstp**

El comando **cmstp** es una utilidad en sistemas Windows que se utiliza para instalar o desinstalar perfiles de conexiones de red y configuraciones de red en sistemas Windows. "CMSTP" significa "Connection Manager Profile Installer", y se utiliza para trabajar con perfiles de conexión personalizados, como los utilizados en entornos corporativos o de redes móviles.

1. Preparar el archivo de perfil:

En primer lugar, debes tener el archivo de perfil de conexión preparado. Este archivo generalmente tiene una extensión .cms y contiene la configuración de la conexión, como la información del servidor, las credenciales, etc.

2. Ejecutar cmstp con el archivo de perfil:

Utiliza el comando cmstp en el símbolo del sistema junto con la ubicación del archivo de perfil de conexión para instalarlo.

```
cmstp /s nombre_del_archivo.cms
```

Reemplaza "nombre_del_archivo.cms" con el nombre real de tu archivo de perfil de conexión.

3. Sigue el asistente:

Cuando ejecutes el comando, se abrirá un asistente que te guiará a través del proceso de instalación. Deberás seguir los pasos y proporcionar la información requerida, como credenciales, servidor, etc.

4. Finaliza la instalación:

Una vez que hayas completado el asistente, el perfil de conexión se instalará en el sistema y estará disponible para su uso.

- **ftp**

El Protocolo de Transferencia de Archivos (FTP, por sus siglas en inglés) es un protocolo de red utilizado para transferir archivos entre un cliente y un servidor en una red. FTP es una forma común de transferir archivos en la web y se utiliza para cargar archivos a un servidor web, descargar archivos desde servidores, mantener sitios web y mucho más.

1. **Abrir una ventana de comandos:**

Abre una ventana de comandos en tu sistema. Puedes hacerlo buscando "cmd" en el menú Inicio y ejecutando el "Símbolo del sistema" en Windows.

2. **Iniciar sesión en un servidor FTP:**

Utiliza el comando ftp seguido de la dirección del servidor FTP al que deseas conectarte:

```
ftp nombre_del_servidor_ftp
```

Reemplaza "nombre_del_servidor_ftp" con la dirección del servidor FTP al que deseas conectarte.

3. **Ingresar credenciales:**

Una vez que te conectes al servidor FTP, el sistema te solicitará que ingreses un nombre de usuario y una contraseña. Proporciona las credenciales requeridas para acceder al servidor.

4. **Explorar y transferir archivos:**

Una vez que hayas iniciado sesión en el servidor FTP, puedes utilizar comandos como ls para listar archivos en el servidor remoto y cd para cambiar de directorio. Utiliza get para descargar archivos desde el servidor FTP al directorio local y put para cargar archivos desde tu sistema al servidor FTP. Por ejemplo:

- Para descargar un archivo desde el servidor FTP:

```
get nombre_del_archivo_remoto
```

- Para cargar un archivo al servidor FTP:

```
put nombre_del_archivo_local
```

5. **Cerrar la sesión y salir de FTP:**

Una vez que hayas completado tus operaciones, puedes salir de la sesión FTP utilizando el comando bye o simplemente cerrando la ventana de comandos.

- **getmac**

El comando getmac es una utilidad en sistemas Windows que se utiliza para obtener la dirección MAC (Media Access Control) de una interfaz de red específica en una computadora. La dirección MAC es un identificador único asignado a cada tarjeta de red y se utiliza para identificar dispositivos en una red local.

1. **Abrir una ventana de comandos:**

Abre una ventana de comandos en tu sistema. Puedes hacerlo buscando "cmd" en el menú Inicio y ejecutando el "Símbolo del sistema" en Windows.

2. **Ejecutar el comando getmac:**

Utiliza el comando getmac seguido de la opción /v para obtener información detallada de las direcciones MAC de todas las interfaces de red en la computadora:

```
getmac /v
```

Esto mostrará una lista de todas las interfaces de red en la computadora junto con sus direcciones MAC.

3. **Obtener la dirección MAC de una interfaz específica:**

Si deseas obtener la dirección MAC de una interfaz de red específica, proporciona el nombre de la interfaz como argumento después del comando getmac. Por ejemplo, para obtener la dirección MAC de la interfaz Ethernet, ejecuta:

```
getmac /v /fo csv | find "Ethernet"
```

Esto mostrará la dirección MAC de la interfaz Ethernet en el formato de salida CSV.

- **Hostname**

El comando hostname es una utilidad de línea de comandos que se utiliza para mostrar o configurar el nombre de host de una computadora en sistemas Unix y Linux. El nombre de host es una etiqueta que se asigna a una computadora en una red y se utiliza para identificarla en la red.

Para mostrar el nombre de host actual:

Ejecuta el siguiente comando en una ventana de terminal:

```
hostname
```

Esto mostrará el nombre de host actual de la computadora.

Para cambiar el nombre de host temporalmente:

Puedes utilizar el comando **hostname** para cambiar el nombre de host de forma temporal durante la sesión actual. Por ejemplo:

```
sudo hostname nuevo_nombre_de_host
```

- **nbtstat**

El comando nbtstat es una utilidad de línea de comandos en sistemas Windows que se utiliza para diagnosticar y mostrar información relacionada con el Protocolo NetBIOS (Network Basic Input/Output System). NetBIOS es un protocolo de red utilizado en sistemas Windows más antiguos para compartir recursos, como archivos e impresoras, en una red local.

Para mostrar las estadísticas del Protocolo NetBIOS:

1. **Abre una ventana de comandos en tu sistema. Puedes hacerlo buscando "cmd" en el menú Inicio y ejecutando el "Símbolo del sistema" en Windows.**
2. **Ejecuta el siguiente comando para mostrar las estadísticas del Protocolo NetBIOS:**

```
nbtstat -s
```

3. **Esto mostrará estadísticas relacionadas con NetBIOS, incluyendo las conexiones activas y las tablas de nombres NetBIOS.**

Para mostrar la tabla de nombres NetBIOS local:

Puedes utilizar el comando nbtstat para mostrar la tabla de nombres NetBIOS local, que se utiliza para resolver nombres de host en una red. Ejecuta el siguiente comando:

```
nbtstat -n
```

Esto mostrará la tabla de nombres NetBIOS local en tu sistema.

Para mostrar la tabla de nombres remotos NetBIOS:

Puedes utilizar el comando nbtstat para mostrar la tabla de nombres remotos NetBIOS, que contiene nombres NetBIOS de otros sistemas en la red con los que tu sistema ha interactuado. Ejecuta el siguiente comando:

```
nbtstat -r
```

Esto mostrará la tabla de nombres remotos NetBIOS en tu sistema.

- **Net**

El comando net es una utilidad de línea de comandos que se utiliza para administrar y realizar diversas operaciones relacionadas con redes y servicios en sistemas Windows. Ofrece una amplia variedad de subcomandos que permiten realizar tareas como administrar recursos compartidos, usuarios y grupos, servicios, impresoras, y más.

1. **Mostrar información sobre usuarios y grupos:**

Para ver una lista de usuarios en el sistema:

```
net user
```

Para ver una lista de grupos en el sistema:

```
net group
```

- **net use**

El comando net use es un comando específico de sistemas Windows que se utiliza para asignar o desconectar unidades de red, como unidades de red compartidas o impresoras en una red. Este comando es útil para conectar unidades de red desde la línea de comandos y administrar conexiones a recursos compartidos.

Para asignar una unidad de red:

Puedes utilizar el comando net use para asignar una unidad de red a una ubicación específica en tu sistema. La sintaxis general es la siguiente:

```
net use unidad_de_red: \\servidor\nombre_del_recurso
```

- unidad_de_red es la letra de unidad que deseas asignar a la ubicación del recurso compartido.
- \\servidor\nombre_del_recurso es la ubicación del recurso compartido en la red.

Por ejemplo, para asignar la letra de unidad Z: a un recurso compartido en un servidor llamado "Servidor1" con el nombre del recurso compartido "Compartido1", puedes ejecutar el siguiente comando:

```
net use Z: \\Servidor1\Compartido1
```

Para desconectar una unidad de red:

Puedes utilizar net use para desconectar una unidad de red previamente asignada. La sintaxis es la siguiente:

```
net use unidad_de_red: /delete
```

Por ejemplo, para desconectar la unidad de red Z:, puedes ejecutar el siguiente comando:

```
net use Z: /delete
```

- **Netsh**

El comando netsh es una utilidad de línea de comandos que se utiliza en sistemas Windows para administrar diversas configuraciones de red y funciones relacionadas con la red. Con netsh, puedes configurar, visualizar y modificar una amplia variedad de parámetros y configuraciones de red, incluyendo configuraciones de interfaz de red, firewall, enrutamiento, entre otros.

Para cambiar la configuración de la interfaz de red:

Puedes utilizar netsh para administrar y configurar las interfaces de red en tu sistema. Por ejemplo, para cambiar la dirección IP de una interfaz de red, puedes ejecutar el siguiente comando:

```
netsh interface ip set address "Nombre de la interfaz" static DIRECCIÓN_IP  
MÁSCARA SUBRED PUERTA_DE_ENLACE
```

- "Nombre de la interfaz" es el nombre de la interfaz de red que deseas configurar.
- DIRECCIÓN_IP es la dirección IP que deseas asignar.
- MÁSCARA es la máscara de subred.
- SUBRED es la subred.
- PUERTA_DE_ENLACE es la puerta de enlace (gateway).

- **Pathping**

El comando pathping es una utilidad de línea de comandos en sistemas Windows que combina las funcionalidades de ping y tracert para proporcionar información detallada sobre la ruta que sigue un paquete de datos a través de la red hacia un destino específico. Esta herramienta es útil para diagnosticar problemas de red y determinar dónde se producen retrasos o pérdida de paquetes en una ruta de red.

```
pathping [opciones] [nombre_del_host]
```

- opciones: Puedes incluir varias opciones para ajustar el comportamiento de pathping. Algunas opciones comunes incluyen -n (no realiza la resolución de nombres de host) y -h (especifica el número máximo de saltos).
- nombre_del_host: Es el nombre de dominio o la dirección IP del destino al que deseas realizar el seguimiento.

Ejemplo de uso:

Supongamos que deseas realizar un seguimiento de la ruta hacia el servidor de Google en www.google.com y obtener información detallada de la ruta. Puedes ejecutar el siguiente comando:

```
pathping www.google.com
```

- **rcp**

El comando rcp (Remote Copy) es un comando de línea de comandos que solía utilizarse para copiar archivos entre sistemas Unix y Unix-like a través de una red. Sin embargo, es importante destacar que rcp ha quedado obsoleto en muchos sistemas y ha sido reemplazado por protocolos y comandos más seguros, como scp (Secure Copy) o rsync, que ofrecen cifrado y mejores prácticas de seguridad.

```
rcp [opciones] origen destino
```

- opciones: Pueden incluir opciones específicas, como -r para copiar directorios y sus contenidos recursivamente, o -p para conservar las propiedades de los archivos, como permisos y propietario.
- origen: Es la ruta al archivo o directorio que deseas copiar desde la máquina local o remota.
- destino: Es la ruta al archivo o directorio de destino en la máquina local o remota.

Un ejemplo de uso sería copiar un archivo local llamado "archivo.txt" a un sistema remoto con la dirección IP "192.168.1.100" en el directorio "/tmp" utilizando rcp:

```
rcp archivo.txt 192.168.1.100:/tmp
```

- **rexec**

El comando rexec (Remote Execution) es una utilidad de línea de comandos que solía utilizarse para ejecutar comandos en un sistema remoto en una red a través del protocolo REXEC. Sin embargo, rexec también es un protocolo no seguro y ha quedado obsoleto en muchos sistemas debido a problemas de seguridad. En su lugar, se recomienda el uso de protocolos y comandos más seguros, como SSH (Secure Shell) y ssh para la ejecución remota de comandos.

```
rexec [-l username] hostname comando
```

- -l username: Es una opción que permite especificar el nombre de usuario en el sistema remoto al que deseas conectarte.
- hostname: Es el nombre de host o la dirección IP del sistema remoto.
- comando: Es el comando que deseas ejecutar en el sistema remoto.

Un ejemplo de uso sería ejecutar el comando "ls" en un sistema remoto con la dirección IP "192.168.1.100" utilizando rexec:

```
rexec -l usuario 192.168.1.100 ls
```

- **route**

El comando route es una utilidad de línea de comandos que se utiliza en sistemas Unix y Unix-like, incluyendo Linux, para visualizar y gestionar la tabla de enrutamiento del sistema. La tabla de enrutamiento contiene información sobre cómo se deben encaminar los paquetes de datos en una red. Puedes usar el comando route para ver rutas de red, agregar rutas, eliminar rutas y realizar otras operaciones relacionadas con la administración de la tabla de enrutamiento.

1. Mostrar la tabla de enrutamiento:

Para ver la tabla de enrutamiento completa en tu sistema, puedes ejecutar el siguiente comando:

```
route -n
```

La opción -n se utiliza para mostrar las direcciones IP en formato numérico en lugar de intentar realizar resolución de nombres.

2. Agregar una ruta estática:

Puedes agregar una ruta estática a la tabla de enrutamiento utilizando el siguiente comando como ejemplo:

```
sudo route add -net 192.168.10.0 netmask 255.255.255.0 gw 192.168.1.1
```

- **rpcping**

El comando `rpcping` es una utilidad de línea de comandos que se utiliza para realizar pruebas y diagnósticos relacionados con el Protocolo de Comunicación Remota (RPC, por sus siglas en inglés) en sistemas Windows. RPC es un protocolo que permite la comunicación entre procesos en sistemas distribuidos y se utiliza ampliamente en entornos Windows para la comunicación entre aplicaciones y servicios.

`rpcping` se utiliza para verificar la conectividad y realizar pruebas de RPC entre sistemas. La sintaxis básica de `rpcping` es la siguiente:

```
rpcping [-t] [-s servidor] [-o] [-a algoritmo] [-u usuario] [-p contraseña]  
[-j dominio] [-e endpoint]
```

- `-t`: Realiza una prueba TCP en lugar de UDP (predeterminado).
- `-s servidor`: Especifica el servidor o el nombre de host al que deseas conectarte.
- `-o`: Realiza una prueba de RPC/HTTP (también conocida como RPC sobre HTTP).
- `-a algoritmo`: Especifica el algoritmo de autenticación a utilizar.
- `-u usuario`: Especifica el nombre de usuario para la autenticación.
- `-p contraseña`: Especifica la contraseña del usuario para la autenticación.
- `-j dominio`: Especifica el dominio del usuario para la autenticación.
- `-e endpoint`: Especifica el punto final de RPC al que deseas conectarte.

- **rsh**

El comando `rsh` (Remote Shell) es una utilidad de línea de comandos que solía utilizarse para ejecutar comandos en un sistema remoto en una red Unix o Unix-like a través del protocolo RSH (Remote Shell). Sin embargo, `rsh` también es un protocolo no seguro y ha quedado obsoleto en muchos sistemas debido a problemas de seguridad. En su lugar, se recomienda el uso de protocolos y comandos más seguros, como SSH (Secure Shell) y `ssh` para la ejecución remota de comandos.

```
rsh [hostname] [comando]
```

- `hostname`: Es el nombre de host o la dirección IP del sistema remoto al que deseas conectarte.
- `comando`: Es el comando que deseas ejecutar en el sistema remoto.

Un ejemplo de uso sería ejecutar el comando `"ls"` en un sistema remoto con la dirección IP `"192.168.1.100"` utilizando `rsh`:

```
rsh 192.168.1.100 ls
```

- **tcmsetup**

El comando tcmsetup es una utilidad de línea de comandos utilizada en sistemas Windows para configurar y administrar el servicio de Administración de Credenciales (Credential Manager). Credential Manager es una característica de Windows que permite a los usuarios almacenar y administrar credenciales, como contraseñas y nombres de usuario, de forma segura.

tcmsetup se utiliza para realizar diversas tareas relacionadas con Credential Manager, como agregar, listar o eliminar credenciales, así como configurar parámetros relacionados con el servicio. La sintaxis y las opciones específicas de tcmsetup pueden variar según la versión de Windows que estés utilizando.

```
tcmsetup /?
```

- **telnet**

El comando telnet es una utilidad de línea de comandos que se utiliza para establecer una conexión de red a un servidor remoto o un dispositivo a través del protocolo Telnet. Telnet es un protocolo que permite la comunicación con un sistema remoto en modo de texto y, por lo general, se utiliza para administrar dispositivos de red, servidores y sistemas Unix o Unix-like. Sin embargo, es importante destacar que Telnet no proporciona cifrado de datos, lo que significa que la información enviada a través de una conexión Telnet es transmitida en texto sin cifrar y puede ser potencialmente interceptada.

La sintaxis básica del comando telnet es la siguiente:

```
telnet [opciones] [host] [puerto]
```

- opciones: Pueden incluir diversas opciones para ajustar el comportamiento de la conexión, como la opción -l para especificar un nombre de usuario, la opción -a para especificar la dirección IP de origen y otras opciones de configuración.
- host: Es el nombre de host o la dirección IP del sistema remoto al que deseas conectarte.
- puerto: Es el número de puerto del servicio al que deseas conectarte en el sistema remoto. El puerto predeterminado para Telnet es el 23.

Un ejemplo de uso sería conectarse a un servidor remoto con la dirección IP "192.168.1.100" en el puerto 23:

```
telnet 192.168.1.100 23
```

- **tftp**

El comando tftp (Trivial File Transfer Protocol) es una utilidad de línea de comandos que se utiliza para transferir archivos entre sistemas en una red utilizando el protocolo TFTP. TFTP es un protocolo de transferencia de archivos simple y sin autenticación que se utiliza para cargar y descargar archivos en una red, principalmente en entornos de arranque remoto (como la carga de firmware en dispositivos de red) y en la administración de dispositivos de red.

La sintaxis básica del comando tftp es la siguiente:

```
tftp [opciones] host [puerto]
```

- opciones: Pueden incluir opciones específicas para configurar la transferencia, como -m para especificar el modo de transferencia (binario o ASCII) y otras opciones relacionadas con el puerto.
- host: Es el nombre de host o la dirección IP del sistema remoto TFTP al que deseas conectarte.
- puerto: Es el número de puerto del servicio TFTP en el sistema remoto. Por lo general, el puerto predeterminado para TFTP es el 69.

Algunos ejemplos de uso de tftp incluyen:

1. Descargar un archivo desde un servidor TFTP:

Para descargar un archivo desde un servidor TFTP, puedes utilizar el comando tftp de la siguiente manera:

```
tftp -g -r archivo_remoto -l archivo_local host
```

- -g: Modo de transferencia binario.
- -r archivo_remoto: Nombre del archivo en el servidor TFTP.
- -l archivo_local: Nombre del archivo local en el que se guardará el archivo descargado.
- host: Nombre de host o dirección IP del servidor TFTP.

2. Subir un archivo a un servidor TFTP:

Para cargar un archivo en un servidor TFTP, puedes utilizar el comando tftp de la siguiente manera:

```
tftp -p -l archivo_local -r archivo_remoto host
```

- -p: Modo de transferencia binario.
- -l archivo_local: Nombre del archivo local que se cargará en el servidor TFTP.
- -r archivo_remoto: Nombre del archivo remoto en el servidor TFTP.
- host: Nombre de host o dirección IP del servidor TFTP.