



12 DE OCTUBRE DE 2023

TAREA #987
SISTEMAS OPERATIVOS

YOSHUA R. MORENO ARREDODNO
JIMENEZ SANCHES ISMAEL



A)

Obtener ayuda del comando ping

```
(kali㉿kali)-[~]
$ 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
  -M <pmtud opt>      define mtu discovery, can be one of <do|dont|want>
  -n                 no dns name resolution
  -O                 report outstanding replies
  -p <pattern>        contents of padding byte
  -q                 quiet output
  -Q <tclass>         use quality of service <tclass> bits
  -s <size>           use <size> as number of data bytes to be sent
  -S <size>           use <size> as SO_SNDBUF socket option value
  -t <ttl>            define time to live
  -U                 print user-to-user latency
  -v                 verbose output
  -V                 print version and exit
  -w <deadline>       reply wait <deadline> in seconds
  -W <timeout>        time to wait for response
```

Enviar un ping a 127.0.0.1 aplicando cualquier parametro

```
(kali㉿kali)-[~]
$ ping -c 4 -i 0.2 127.0.0.1
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=0.024 ms
64 bytes from 127.0.0.1: icmp_seq=2 ttl=64 time=0.037 ms
64 bytes from 127.0.0.1: icmp_seq=3 ttl=64 time=0.049 ms
64 bytes from 127.0.0.1: icmp_seq=4 ttl=64 time=0.037 ms

— 127.0.0.1 ping statistics —
4 packets transmitted, 4 received, 0% packet loss, time 612ms
rtt min/avg/max/mdev = 0.024/0.036/0.049/0.008 ms
```

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

```
(kali㉿kali)-[~]
$ ping www.google.com
PING www.google.com (172.217.3.132) 56(84) bytes of data:
64 bytes from yyz08s13-in-f132.1e100.net (172.217.3.132): icmp_seq=1 ttl=115 time=67.5 ms
64 bytes from yyz08s13-in-f132.1e100.net (172.217.3.132): icmp_seq=2 ttl=115 time=79.6 ms
64 bytes from yyz08s13-in-f132.1e100.net (172.217.3.132): icmp_seq=3 ttl=115 time=69.4 ms
64 bytes from yyz08s13-in-f132.1e100.net (172.217.3.132): icmp_seq=4 ttl=115 time=77.3 ms
64 bytes from yyz08s13-in-f132.1e100.net (172.217.3.132): icmp_seq=5 ttl=115 time=68.9 ms
64 bytes from yyz08s13-in-f132.1e100.net (172.217.3.132): icmp_seq=6 ttl=115 time=70.1 ms
64 bytes from yyz08s13-in-f132.1e100.net (172.217.3.132): icmp_seq=7 ttl=115 time=70.4 ms
64 bytes from yyz08s13-in-f132.1e100.net (172.217.3.132): icmp_seq=8 ttl=115 time=71.3 ms
64 bytes from yyz08s13-in-f132.1e100.net (172.217.3.132): icmp_seq=9 ttl=115 time=74.4 ms
64 bytes from yyz08s13-in-f132.1e100.net (172.217.3.132): icmp_seq=10 ttl=115 time=68.0 ms
64 bytes from yyz08s13-in-f132.1e100.net (172.217.3.132): icmp_seq=11 ttl=115 time=67.2 ms
64 bytes from yyz08s13-in-f132.1e100.net (172.217.3.132): icmp_seq=12 ttl=115 time=77.9 ms
64 bytes from yyz08s13-in-f132.1e100.net (172.217.3.132): icmp_seq=13 ttl=115 time=67.9 ms
64 bytes from yyz08s13-in-f132.1e100.net (172.217.3.132): icmp_seq=14 ttl=115 time=67.6 ms
64 bytes from yyz08s13-in-f132.1e100.net (172.217.3.132): icmp_seq=15 ttl=115 time=67.7 ms
64 bytes from yyz08s13-in-f132.1e100.net (172.217.3.132): icmp_seq=16 ttl=115 time=67.6 ms
64 bytes from yyz08s13-in-f132.1e100.net (172.217.3.132): icmp_seq=17 ttl=115 time=71.0 ms
64 bytes from yyz08s13-in-f132.1e100.net (172.217.3.132): icmp_seq=18 ttl=115 time=70.2 ms
^C
— www.google.com ping statistics —
18 packets transmitted, 18 received, 0% packet loss, time 17020ms
rtt min/avg/max/mdev = 67.234/70.771/79.590/3.798 ms
```

Obtener ayuda del comando nslookup

```
(kali㉿kali)-[~]
$ nslookup -h
*** Invalid option: h
> --help
nslookup: '--help' is not a legal IDNA2008 name (string start/ends with forbidden hyphen), use +noid
nin
```

Resolver la direccion ip de https://upqroo.edu.mx/ usando nslookup

```
(kali㉿kali)-[~]
$ nslookup upqroo.edu.mx
Server:      192.168.1.254
Address:     192.168.1.254#53

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

Hacer ping a la ip obtenida anteriormente , anotar conclusiones

```
(kali㉿kali)-[~]
$ ping 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=55 time=158 ms
64 bytes from 77.68.126.20: icmp_seq=2 ttl=55 time=158 ms
64 bytes from 77.68.126.20: icmp_seq=3 ttl=55 time=158 ms
64 bytes from 77.68.126.20: icmp_seq=4 ttl=55 time=158 ms
64 bytes from 77.68.126.20: icmp_seq=5 ttl=55 time=158 ms
64 bytes from 77.68.126.20: icmp_seq=6 ttl=55 time=160 ms
64 bytes from 77.68.126.20: icmp_seq=7 ttl=55 time=159 ms
64 bytes from 77.68.126.20: icmp_seq=8 ttl=55 time=159 ms
64 bytes from 77.68.126.20: icmp_seq=9 ttl=55 time=158 ms
64 bytes from 77.68.126.20: icmp_seq=10 ttl=55 time=163 ms
64 bytes from 77.68.126.20: icmp_seq=11 ttl=55 time=165 ms
64 bytes from 77.68.126.20: icmp_seq=12 ttl=55 time=158 ms
64 bytes from 77.68.126.20: icmp_seq=13 ttl=55 time=158 ms
64 bytes from 77.68.126.20: icmp_seq=14 ttl=55 time=158 ms
64 bytes from 77.68.126.20: icmp_seq=15 ttl=55 time=161 ms
^C
 77.68.126.20 ping statistics ---
15 packets transmitted, 15 received, 0% packet loss, time 14013ms
rtt min/avg/max/mdev = 157.845/159.378/164.697/2.005 ms
```

Obtener ayuda del comando netstat

```
(kali㉿kali)-[~]
$ netstat --help
netstat: invalid option -- '='
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

    -l, --listening       display listening server sockets
    -a, --all             display all sockets (default: connected)
    -F, --fib             display Forwarding Information Base (default)
    -C, --cache           display routing cache instead of FIB
    -Z, --context         display SELinux security context for sockets

<Socket> { -t|-tcp } { -u|-udp } { -U|-udplite } { -S|-sctp } { -w|-raw }
          { -x|-unix } --ax25 --ipx --netrom
<AF> Use '-6|-4' or '-A <af>' or '-<af>'; default: inet
List of possible address families (which support routing):
inet (DARPA Internet) inet6 (IPv6) ax25 (AMPR AX.25)
netrom (AMPR NET/ROM) rose (AMPR ROSE) ipx (Novell IPX)
ddp (Appletalk DDP) x25 (CCITT X.25)
```


Mostrar todas las conexiones y puertos , anotar conclusiones

```
(kali㉿kali)-[~]
$ netstat -tuln
Active Internet connections (only servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
```

Ejecutar netstat sin resolver nombres de dominio o puertos

```
(kali㉿kali)-[~]
$ netstat -n
Active Internet connections (w/o servers)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
udp        0      0 10.0.2.15:68            10.0.2.2:67            ESTABLISHED

Active UNIX domain sockets (w/o servers)
Proto RefCnt Flags               Type           State         I-Node  Path
unix   3      [ ]               STREAM         CONNECTED     20387    /run/user/1000/bus
unix   3      [ ]               STREAM         CONNECTED     20245    /run/user/1000/bus
unix   3      [ ]               STREAM         CONNECTED     20026    /run/user/1000/bus
unix   3      [ ]               STREAM         CONNECTED     19796    /run/dbus/system_bus_socket
unix   3      [ ]               DGRAM          CONNECTED     15207    /run/user/1000/bus
unix   3      [ ]               STREAM         CONNECTED     20250    /run/user/1000/bus
unix   3      [ ]               STREAM         CONNECTED     19144    /run/user/1000/bus
unix   3      [ ]               STREAM         CONNECTED     20520    /run/user/1000/bus
unix   3      [ ]               STREAM         CONNECTED     20435    /run/user/1000/bus
unix   3      [ ]               STREAM         CONNECTED     20146    /run/user/1000/bus
unix   3      [ ]               STREAM         CONNECTED     20187    /run/user/1000/bus
unix   3      [ ]               STREAM         CONNECTED     19041    /run/user/1000/pipewire-0
unix   3      [ ]               STREAM         CONNECTED     19192    @/tmp/.X11-unix/X0
unix   3      [ ]               STREAM         CONNECTED     19116    /run/dbus/system_bus_socket
unix   2      [ ]               DGRAM          CONNECTED     16520    /run/dbus/system_bus_socket
unix   3      [ ]               STREAM         CONNECTED     19286    /run/dbus/system_bus_socket
unix   3      [ ]               STREAM         CONNECTED     20652    /run/dbus/system_bus_socket
unix   3      [ ]               STREAM         CONNECTED     19261    /run/systemd/journal/stdout
unix   3      [ ]               STREAM         CONNECTED     19870    /run/user/1000/bus
unix   3      [ ]               STREAM         CONNECTED     19903    /run/dbus/system_bus_socket
unix   3      [ ]               STREAM         CONNECTED     20503    @/tmp/.X11-unix/X0
unix   3      [ ]               STREAM         CONNECTED     20384    /run/dbus/system_bus_socket
unix   3      [ ]               STREAM         CONNECTED     20281    /run/dbus/system_bus_socket
unix   3      [ ]               STREAM         CONNECTED     15889    /run/dbus/system_bus_socket
unix   3      [ ]               STREAM         CONNECTED     20240    /run/user/1000/at-spi/bus_0
unix   3      [ ]               STREAM         CONNECTED     19220    /run/user/1000/at-spi/bus_0
unix   3      [ ]               STREAM         CONNECTED     19143    /run/user/1000/at-spi/bus_0
unix   3      [ ]               STREAM         CONNECTED     20010    /run/user/1000/at-spi/bus_0
unix   3      [ ]               DGRAM          CONNECTED     15208    /run/user/1000/at-spi/bus_0
unix   3      [ ]               STREAM         CONNECTED     19375    /run/user/1000/at-spi/bus_0
unix   3      [ ]               STREAM         CONNECTED     20189    /run/user/1000/at-spi/bus_0
unix   2      [ ]               DGRAM          CONNECTED     14319    /run/user/1000/at-spi/bus_0
unix   3      [ ]               STREAM         CONNECTED     20515    /run/user/1000/at-spi/bus_0
unix   3      [ ]               STREAM         CONNECTED     18840    /run/user/1000/at-spi/bus_0
unix   3      [ ]               STREAM         CONNECTED     19175    /run/user/1000/at-spi/bus_0
```

Mostrar conexiones TCP

```
(kali㉿kali)-[~]
$ netstat -atn
Active Internet connections (servers and established)
Proto Recv-Q Send-Q Local Address           Foreign Address         State
```

Mostrar conexiones UDP

```
(kali@kali)-[~]  
$ netstat -aun  
Active Internet connections (servers and established)  
Proto Recv-Q Send-Q Local Address           Foreign Address         State  
udp        0      0 10.0.2.15:68           10.0.2.2:67            ESTABLISHED
```

Utilizar el comando tasklist

```
(kali@kali)-[~]  
$ netstat -asn  
Ip:  
  Forwarding: 2  
  66 total packets received  
  1 with invalid addresses  
  0 forwarded  
  0 incoming packets discarded  
  65 incoming packets delivered  
  68 requests sent out  
Icmp:  
  41 ICMP messages received  
  0 input ICMP message failed  
  ICMP input histogram:  
    echo requests: 4  
    echo replies: 37  
  41 ICMP messages sent  
  0 ICMP messages failed  
  ICMP output histogram:  
    echo requests: 37  
    echo replies: 4  
IcmpMsg:  
  InType0: 37  
  InType8: 4  
  OutType0: 4  
  OutType8: 37  
Tcp:  
  2 active connection openings  
  0 passive connection openings  
  2 failed connection attempts  
  0 connection resets received  
  0 connections established  
  4 segments received  
  4 segments sent out  
  0 segments retransmitted  
  0 bad segments received  
  2 resets sent
```

Utilizar el comando tracer

```
(kali@kali)-[~]
$ traceroute google.com
traceroute to google.com (142.251.34.46), 30 hops max, 60 byte packets
 1  10.0.2.2 (10.0.2.2)  0.252 ms  0.200 ms  0.165 ms
 2  * * *
 3  * * *
 4  * * *
 5  * * *
 6  * * *
 7  * * *
 8  * * *
 9  * * *
10  * * *
11  * * *
12  * * *
13  * * *
14  * * *
15  * * *
16  * * *
17  * * * ^C
```

Utilizar el comando ARP

```
(kali@kali)-[~]
$ arp
Address                  HWtype  HWaddress           Flags Mask            Iface
10.0.2.2                  ether    52:54:00:12:35:02    C                     eth0
```

B)

1. ¿Para qué sirve el comando ping? el comando ping se utiliza para probar la conexión de red entre su computadora y un host remoto. ICMP envía paquetes de solicitud de eco al host de destino y espera respuestas para determinar si se puede acceder al host. Además, se utiliza para medir la latencia de paquetes o el tiempo de ida y vuelta entre dos máquinas.
2. ¿Para qué sirve el comando nslookup? el comando nslookup se utiliza para realizar una búsqueda de DNS (Sistema de nombres de dominio). Le permite consultar y recuperar información sobre nombres de dominio, direcciones IP y otros registros DNS. Esto es útil para resolver nombres de host en direcciones IP (o viceversa)
3. ¿Para qué sirve el comando netstat? el comando netstat se utiliza para mostrar información sobre la configuración de la red, conexiones de red activas y estadísticas. Puede ver puertos de escucha, conexiones establecidas, tablas de enrutamiento y estadísticas de interfaz de red.
4. ¿Para qué sirve el comando tasklist? el comando lista de tareas se utiliza en sistemas Windows para enumerar todos los procesos que se ejecutan en el sistema. Muestra una lista de procesos en ejecución, junto con información como el nombre del proceso, el identificador del proceso (PID) y la cantidad de uso de CPU y memoria.
5. ¿Para qué sirve el comando tracert? el comando tracert se utiliza para rastrear la ruta de los paquetes desde una computadora a través de la red hasta el host de destino. Le permite identificar problemas en la red y determinar el retraso en cada salto mostrando el orden de los saltos (enrutadores) por los que pasan los paquetes para llegar a su destino.

c)

Investigar los siguientes comandos y anotar ejemplos practicos:

atmadm: El comando atmadm es específico de sistemas operativos Windows y se utiliza para administrar configuraciones y adaptadores de redes de banda ancha y redes ATM (Modo de Transferencia Asíncrona).

```
batch
atmadm -l
```

bitsadmin: Se utiliza para administrar trabajos de transferencia en segundo plano, conocidos como BITS (Background Intelligent Transfer Service). Puede ser útil para administrar descargas y transferencias de archivos en Windows.

```
bitsadmin /create myjob
bitsadmin /addfile myjob https://www.ejemplo.com/archivo.zip C:\carpeta\archivo.zip
bitsadmin /resume myjob
```

cmstp: El comando cmstp es una utilidad de línea de comandos que se utiliza en sistemas Windows para instalar o desinstalar perfiles de conexiones de red. Estos perfiles pueden incluir configuraciones de VPN, conexiones de red inalámbrica, etc.

```
batch
cmstp /s mi_perfil.inf
```

fp: El comando fp es un comando que se utiliza para realizar operaciones relacionadas con la administración de paquetes en sistemas operativos basados en Linux. Sin embargo, el comando fp en sí no es un comando estándar de Linux, y su funcionalidad no está ampliamente reconocida en sistemas operativos basados en Linux.

getmac: El comando getmac es una utilidad de línea de comandos que se utiliza en sistemas Windows para recuperar y mostrar la dirección MAC (Media Access Control) de una o más interfaces de red en una computadora.

```
batch
getmac
```

hostname: El comando hostname se utiliza para mostrar o cambiar el nombre del host de una computadora. El nombre del host es una etiqueta que se asigna a una máquina en una red, y se utiliza para identificarla de manera única en esa red.


```
bash
hostname -F nuevo_nombre
```

nbtstat: El comando nbtstat se utiliza en sistemas Windows para diagnosticar y mostrar información relacionada con NetBIOS (Sistema de Nombres de Browsing por Internet, por sus siglas en inglés). NetBIOS es un conjunto de protocolos que permiten la comunicación en redes locales.

```
batch
nbtstat -c
```

net use: El comando net use se utiliza en sistemas Windows para conectar o desconectar unidades de red, como recursos compartidos en servidores o sistemas de almacenamiento en red. Puedes utilizarlo para asignar letras de unidad a recursos compartidos de red y gestionar las conexiones de red.

```
batch
net use Z: \\servidor\compartir /user:usuario contraseña
```

netsh: El comando netsh es una utilidad de línea de comandos que se utiliza en sistemas Windows para configurar y administrar una variedad de componentes de red, incluyendo la configuración de interfaces de red, firewall, enrutamiento y otros servicios de red.

```
batch
netsh firewall show config
```

pathping: El comando pathping se utiliza en sistemas Windows para realizar un seguimiento de la ruta de un paquete a un host de destino y proporcionar información detallada sobre cada salto en la ruta, incluyendo latencia y pérdida de paquetes en cada nodo intermedio. Combina la funcionalidad de traceroute y ping.

```
batch
pathping www.ejemplo.com
```

rexec: El comando rexec se utiliza para ejecutar comandos en una máquina remota en sistemas Windows. Permite a un usuario ejecutar comandos en una computadora remota autenticándose con un nombre de usuario y contraseña.

```
batch
rexec -l usuario -p contraseña servidor ipconfig
```

route: El comando route se utiliza para mostrar y configurar la tabla de enrutamiento IP en sistemas Windows. Permite agregar, modificar o eliminar rutas en el sistema, lo que afecta cómo se dirigen los paquetes en la red.

```
batch
```

```
route add 192.168.1.100 mask 255.255.255.255 192.168.1.1
```

rpcping: El comando rpcping se utiliza para probar la conectividad RPC (Remote Procedure Call) en sistemas Windows. RPC es un mecanismo utilizado para la comunicación entre procesos en una red. rpcping ayuda a diagnosticar problemas de conectividad RPC entre sistemas.

```
batch
```

```
rpcping -s servidor -e 135
```

rsh: El comando rsh permite ejecutar comandos en una máquina remota a través del protocolo RSH (Remote Shell). Es importante mencionar que el uso de rsh es inseguro y desaconsejado debido a problemas de seguridad, y su uso no es recomendable.

```
bash
```

```
rsh servidor ls
```

tcmsetup: El comando tcmsetup se utiliza en sistemas Windows para configurar servicios de administración de tarjetas inteligentes (smart cards). Puede ser útil en entornos donde se utilizan tarjetas inteligentes para la autenticación y la seguridad.

telnet: El comando telnet permite iniciar una sesión de terminal remota con un servidor a través del protocolo Telnet. Se utiliza para administrar servidores o dispositivos de red a través de la línea de comandos.

```
batch
```

```
telnet servidor
```

tftp: El comando tftp se utiliza para transferir archivos a través del protocolo TFTP (Trivial File Transfer Protocol). TFTP es una versión simplificada de FTP y se utiliza para transferir archivos en redes locales

```
batch
```

```
tftp -i -v -t get archivo.txt
```