Basic Networking Commands

This assignment explains basic networking commands such as ipconfig, ping, route, tracert, traceroute, arp, netstat, NetBIOS, nslookup, finger, nmap/port scan in detail with examples.

Ipconfig

Displays all current TCP/IP network configuration values and refreshes Dynamic Host Configuration Protocol (DHCP) and Domain Name System (DNS) settings. This command is most useful on computers that are configured to obtain an IP address automatically. This enables users to determine which TCP/IP configuration values have been configured by DHCP, Automatic Private IP Addressing (APIPA), or an alternate configuration.

- If the Adapter name contains any spaces, use quotation marks around the adapter name (that is, "Adapter Name").
- For adapter names, ipconfig supports the use of the asterisk (*) wildcard character to specify either adapters with names that begin with a specified string or adapters with names that contain a specified string.
- For example, **Local*** matches all adapters that start with the string Local and ***Con*** matches all adapters that contain the string Con.

Syntax

ipconfig [**/all**] [**/renew** [Adapter]] [**/release** [Adapter]] [**/flushdns**] [**/displaydns**] [**/registerdns**] [**/showclassid** Adapter] [**/setclassid** Adapter [ClassID]]

Parameters

Used without parameters, ipconfig displays the IP address, subnet mask, and default gateway for all adapters.

/all Displays the full TCP/IP configuration for all adapters. Without this parameter, ipconfig displays only the IP address, subnet mask, and default gateway values for each adapter. Adapters can represent physical interfaces, such as installed network adapters, or logical interfaces, such as dial-up connections.

/renew [Adapter] Renews DHCP configuration for all adapters (if an adapter is not specified) or for a specific adapter if the Adapter parameter is included. This parameter is available only on computers with adapters that are configured to obtain an IP address automatically. To specify an adapter name, type the adapter name that appears when you use ipconfig without parameters.

/release [Adapter] Sends a DHCPRELEASE message to the DHCP server to release the current DHCP configuration and discard the IP address configuration for either all adapters (if an adapter is not specified) or for a specific adapter if the Adapter parameter is included. This parameter disables TCP/IP for adapters configured to obtain an IP address automatically. To specify an adapter name, type the adapter name that appears when you use ipconfig without parameters.

/flushdns Flushes and resets the contents of the DNS client resolver cache. During DNS troubleshooting, you can use this procedure to discard negative cache entries from the cache, as well as any other entries that have been added dynamically.

/displaydns Displays the contents of the DNS client resolver cache, which includes both entries preloaded from the local Hosts file and any recently obtained resource records for name queries resolved by the computer. The DNS Client service uses this information to resolve frequently queried names quickly, before querying its configured DNS servers.

/registerdns Initiates manual dynamic registration for the DNS names and IP addresses that are configured at a computer. You can use this parameter to troubleshoot a failed DNS name registration or resolve a dynamic update problem between a client and the DNS server without rebooting the client computer. The DNS settings in the advanced properties of the TCP/IP protocol determine which names are registered in DNS.

/showclassid Adapter Displays the DHCP class ID for a specified adapter. To see the DHCP class ID for all adapters, use the asterisk (*) wildcard character in place of Adapter. This parameter is available only on computers with adapters that are configured to obtain an IP address automatically.

/setclassid Adapter [ClassID] Configures the DHCP class ID for a specified adapter. To set the DHCP class ID for all adapters, use the asterisk (*) wildcard character in place of Adapter. This parameter is available only on computers with adapters that are configured to obtain an IP address automatically. If a DHCP class ID is not specified, the current class ID is removed.

Examples:

To display the basic TCP/IP configuration for all adapters, type:

- `ipconfig`

To display the full TCP/IP configuration for all adapters, type:

- `ipconfig /all`

To renew a DHCP-assigned IP address configuration for only the Local Area Connection adapter, type:

- ipconfig /renew "Local Area Connection"

To flush the DNS resolver cache when troubleshooting DNS name resolution problems, type:

- ipconfig /flushdns

To display the DHCP class ID for all adapters with names that start with Local, type:

- ipconfig /showclassid Local

To set the DHCP class ID for the Local Area Connection adapter to TEST, type:

- ipconfig /setclassid "Local Area Connection" TEST

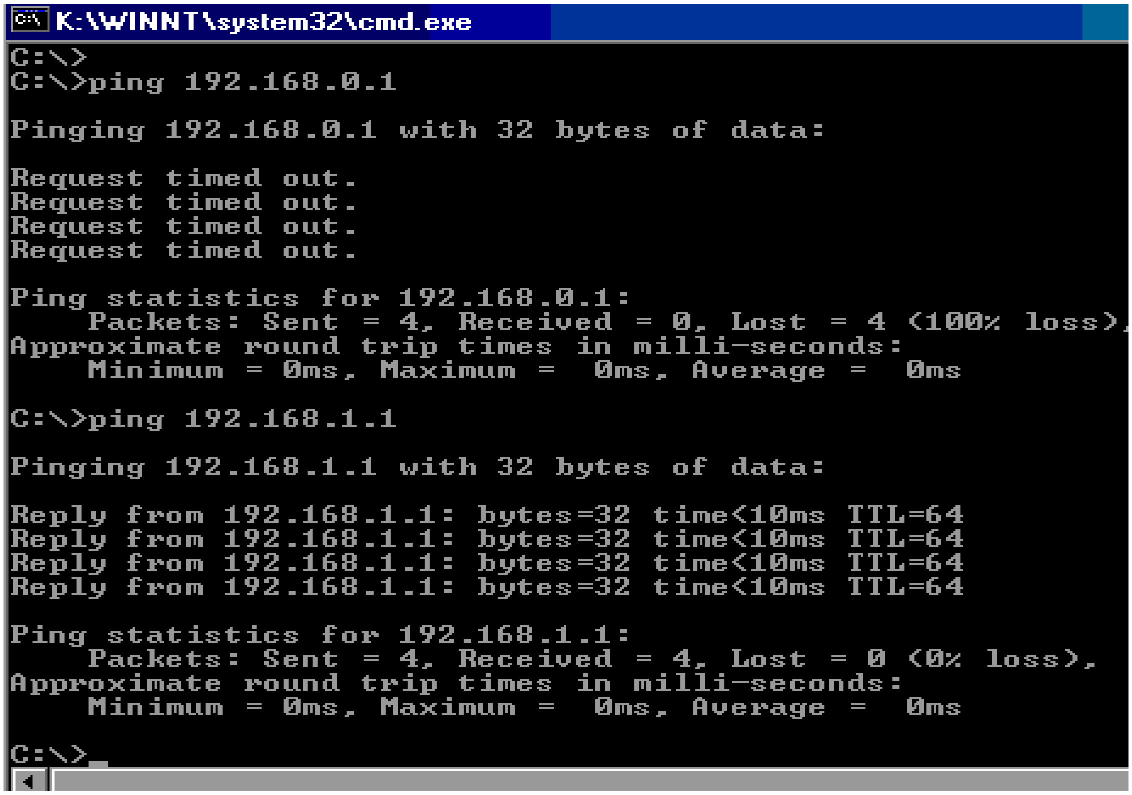
winipcfg

This utility allows users or adminstrators to see the current IP address and other useful information about your network configuration. You can reset one or more IP addresses. The Release or Renew buttons allow you to release or renew one IP address. If you want to release or renew all IP addresses click Release All or Renew All. When one of these buttons is clicked, a new IP address is obtained from either the DHCP service or from the computer assigning itself an automatic private IP address. **To use the winipcfg utility:**

- Click Start,and then click Run and type **winipcfg**
- Click More Info.
- To see the addresses of the DNS servers the computer is configured to use, click the ellipsis (...) button to the right of DNS Servers.
- To see address information for your network adapter(s), select an adapter from the list in Ethernet Adapter Information.

Ping

Verifies IP-level connectivity to another TCP/IP computer by sending Internet Control Message Protocol (ICMP) Echo Request messages. The receipt of corresponding Echo Reply messages are displayed, along with round-trip times. Ping is the primary TCP/IP command used to troubleshoot connectivity, reachability, and name resolution.



```
C:\>
C:\>ping 192.168.0.1

Pinging 192.168.0.1 with 32 bytes of data:

Request timed out.
Request timed out.
Request timed out.
Request timed out.

Ping statistics for 192.168.0.1:
    Packets: Sent = 4, Received = 0, Lost = 4 (100% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum =  0ms, Average =  0ms

C:\>ping 192.168.1.1

Pinging 192.168.1.1 with 32 bytes of data:

Reply from 192.168.1.1: bytes=32 time<10ms TTL=64
Reply from 192.168.1.1: bytes=32 time<10ms TTL=64
Reply from 192.168.1.1: bytes=32 time<10ms TTL=64
Reply from 192.168.1.1: bytes=32 time<10ms TTL=64

Ping statistics for 192.168.1.1:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 0ms, Maximum =  0ms, Average =  0ms

C:\>_
```

You can use ping to test both the computer name and the IP address of the computer. If pinging the IP address is successful, but pinging the computer name is not, you might have a name resolution problem. In this case, ensure that the computer name you are specifying can be resolved through the local Hosts file, by using Domain Name System (DNS) queries, or through NetBIOS name resolution techniques.

To test a TCP/IP configuration by using the ping command:

- To quickly obtain the TCP/IP configuration of a computer, open Command Prompt, and then type **ipconfig** . From the display of the ipconfig command, ensure that the network adapter for the TCP/IP configuration you are testing is not in a Media disconnected state.
- At the command prompt, ping the loopback address by typing **ping 127.0.0.1**
- Ping the IP address of the computer.
- Ping the IP address of the default gateway. If the ping command fails, verify that the default gateway IP address is correct and that the gateway (router) is operational.
- Ping the IP address of a remote host (a host that is on a different subnet). If the ping command fails, verify that the remote host IP address is correct, that the remote host is operational, and that all of the gateways (routers) between this computer and the remote host are operational.

- Ping the IP address of the DNS server. If the ping command fails, verify that the DNS server IP address is correct, that the DNS server is operational, and that all of the gateways (routers) between this computer and the DNS server are operational.

hostname

The **hostname** command shows or sets the system hostname.

hostname is used to display the system's DNS name, and to display or set its hostname or NIS (Network Information Services) domain name.

When called without any arguments, **hostname** will display the name of the system as returned by the gethostname function.

When called with one argument or with the **--file** option, **hostname** will set the system's host name using the sethostname function. Only the superuser can set the host name.

The host name is usually set once at system startup in the script **/etc/init.d/hostname.sh** normally by reading the contents of a file which contains the host name, e.g., **/etc/hostname**.

hostname syntax

```
hostname [-v] [-a|--alias] [-d|--domain] [-f|--fqdn|--long] [-A|--all-fqdns]
        [-i|--ip-address] [-I|--all-ip-addresses] [-s|--short] [-y|--yp|--nis]
hostname [-v] [-b|--boot] [-F|--file file name] [hostname]
hostname [-v] [-h|--help] [-V|--version]
```

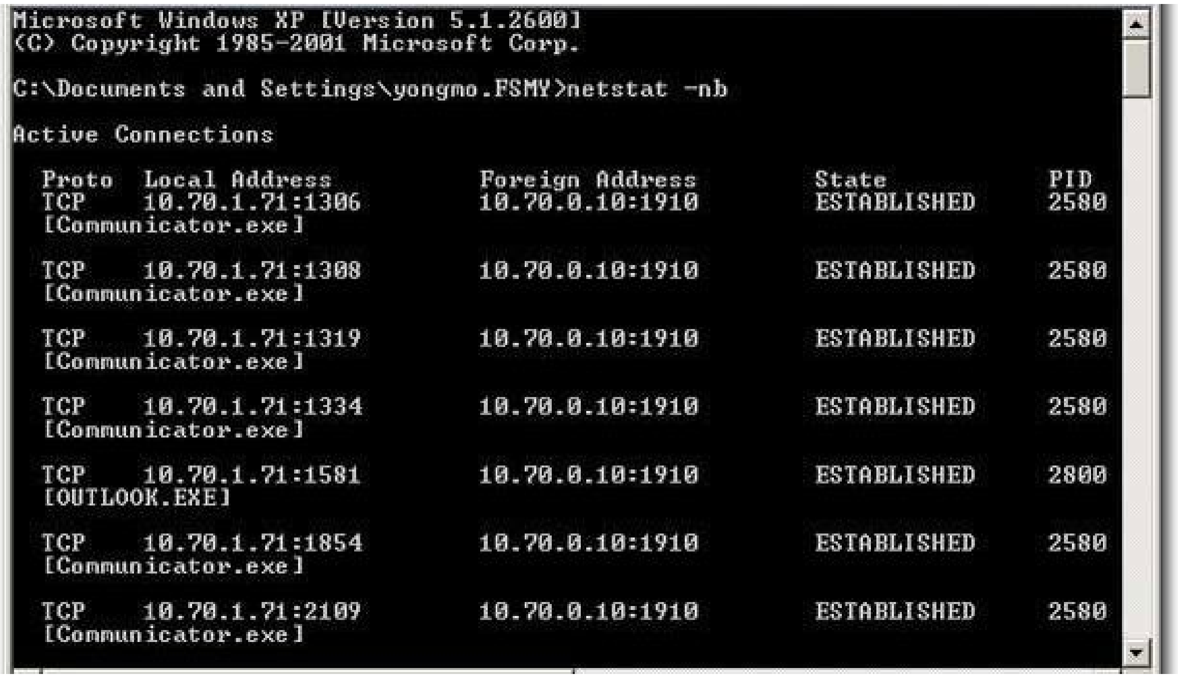
Options

-a, **--alias**	Display the alias name of the host (if used). This option is deprecated and should not be used anymore.
-A, **--all-fqdns**	Displays every FQDN of the machine. This option enumerates all configured network addresses on all configured network interfaces, and translates them to DNS domain names. Addresses that cannot be translated (i.e. because they do not have an appropriate reverse DNS entry) are skipped. Note that different addresses may resolve to the same name, therefore the output may contain duplicate entries. Do not make any assumptions about the order of the output.
-b, **--boot**	Always set a hostname; this allows the file specified by **-F** to be non-existant or empty, in which case the default hostname

	localhost will be used if none is yet set.
-d, **--domain**	Display the name of the DNS domain. Don't use the command **domainname** to get the DNS domain name because it will show the NIS domain name and not the DNS domain name. Use **dnsdomainname** instead. See the warnings in section The FQDN, and avoid using this option if at all possible.
-f, **--fqdn**, **--long**	Display the FQDN (Fully Qualified Domain Name). A FQDN consists of a short host name and the DNS domain name. Unless you are using bind (Berkeley Internet Domain Name) or NIS for host lookups, you can change the FQDN and the DNS domain name (which is part of the FQDN) in the **/etc/hosts** file. See the warnings in section The FQDN, and avoid using this option if at all possible; use **hostname --all-fqdns** instead.
-F, **--file** *file name*	Read the host name from the specified file. Comments (lines starting with a \`**#**') are ignored.
-i, **--ip-address**	Display the network address(es) of the host name. Note that this works only if the host name can be resolved. Avoid using this option if at all possible; use **hostname --all-ip-addresses** instead.
-I, **--all-ip-addresses**	Display all network addresses of the host. This option enumerates all configured addresses on all network interfaces. The loopback interface and IPv6 link-local addresses are omitted. Contrary to option **-i**, this option does not depend on name resolution. Do not make any assumptions about the order of the output.
-s, **--short**	Display the short host name. This is the host name cut at the first dot.
-v, **--verbose**	Be verbose with all output.
-V, **--version**	Print version information on standard output and exit successfully.
-y, **--yp**, **--nis**	Display the NIS domain name. If a parameter is given (or **--file** name) then root (the superuser) can also set a new NIS domain.
-h, **--help**	Print a help message and exit.

Netstat

Displays active TCP connections, ports on which the computer is listening, Ethernet statistics, the IP routing table, IPv4 statistics (for the IP, ICMP, TCP, and UDP protocols), and IPv6 statistics (for the IPv6, ICMPv6, TCP over IPv6, and UDP over IPv6 protocols).



```
Microsoft Windows XP [Version 5.1.2600]
(C) Copyright 1985-2001 Microsoft Corp.

C:\Documents and Settings\yongmo.FSMY>netstat -nb

Active Connections

  Proto  Local Address          Foreign Address        State           PID
  TCP    10.70.1.71:1306        10.70.0.10:1910        ESTABLISHED     2580
  [Communicator.exe]

  TCP    10.70.1.71:1308        10.70.0.10:1910        ESTABLISHED     2580
  [Communicator.exe]

  TCP    10.70.1.71:1319        10.70.0.10:1910        ESTABLISHED     2580
  [Communicator.exe]

  TCP    10.70.1.71:1334        10.70.0.10:1910        ESTABLISHED     2580
  [Communicator.exe]

  TCP    10.70.1.71:1581        10.70.0.10:1910        ESTABLISHED     2800
  [OUTLOOK.EXE]

  TCP    10.70.1.71:1854        10.70.0.10:1910        ESTABLISHED     2580
  [Communicator.exe]

  TCP    10.70.1.71:2109        10.70.0.10:1910        ESTABLISHED     2580
  [Communicator.exe]
```

Netstat provides statistics for the following:

- Proto - The name of the protocol (TCP or UDP).
- Local Address - The IP address of the local computer and the port number being used. The name of the local computer that corresponds to the IP address and the name of the port is shown unless the -n parameter is specified. If the port is not yet established, the port number is shown as an asterisk (*).
- Foreign Address - The IP address and port number of the remote computer to which the socket is connected. The names that corresponds to the IP address and the port are shown unless the -n parameter is specified. If the port is not yet established, the port number is shown as an asterisk (*).

(state) Indicates the state of a TCP connection. The possible states are as follows:

- CLOSE_WAIT
- CLOSED
- ESTABLISHED
- FIN_WAIT_1
- FIN_WAIT_2
- LAST_ACK
- LISTEN
- SYN_RECEIVED
- SYN_SEND
- TIMED_WAIT

Syntax

```
netstat [-a] [-e] [-n] [-o] [-p Protocol] [-r] [-s] [Interval]
```

Parameters

Used without parameters, netstat displays active TCP connections.

-a Displays all active TCP connections and the TCP and UDP ports on which the computer is listening.

-e Displays Ethernet statistics, such as the number of bytes and packets sent and received. This parameter can be combined with -s.

-n Displays active TCP connections, however, addresses and port numbers are expressed numerically and no attempt is made to determine names.

-o Displays active TCP connections and includes the process ID (PID) for each connection. You can find the application based on the PID on the Processes tab in Windows Task Manager. This parameter can be combined with -a, -n, and -p.

-p Shows connections for the protocol specified by Protocol. In this case, the Protocol can be tcp, udp, tcpv6, or udpv6. If this parameter is used with -s to display statistics by protocol, Protocol can be tcp, udp, icmp, ip, tcpv6, udpv6, icmpv6, or ipv6.

-s Displays statistics by protocol. By default, statistics are shown for the TCP, UDP, ICMP, and IP protocols. If the IPv6 protocol for Windows XP is installed, statistics are shown for the TCP over IPv6, UDP over IPv6, ICMPv6, and IPv6 protocols. The -p parameter can be used to specify a set of protocols.

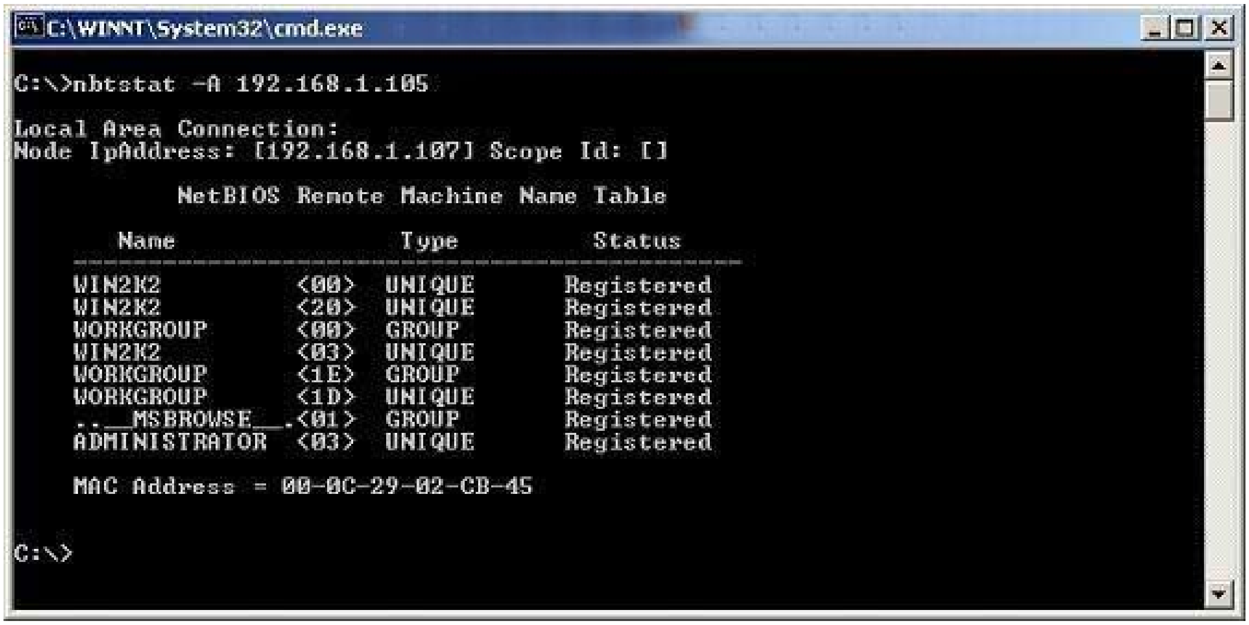
-r Displays the contents of the IP routing table. This is equivalent to the route print command.

Interval Redisplays the selected information every Interval seconds. Press CTRL+C to stop the redisplay. If this parameter is omitted, netstat prints the selected information only once.

/? - Displays help at the command prompt.

Nbtstat

Displays NetBIOS over TCP/IP (NetBT) protocol statistics,



```
C:\>nbtstat -A 192.168.1.105

Local Area Connection:
Node IpAddress: [192.168.1.107] Scope Id: []

           NetBIOS Remote Machine Name Table

       Name               Type         Status
    ---------------------------------------------
    WIN2K2         <00>  UNIQUE      Registered
    WIN2K2         <20>  UNIQUE      Registered
    WORKGROUP      <00>  GROUP       Registered
    WIN2K2         <03>  UNIQUE      Registered
    WORKGROUP      <1E>  GROUP       Registered
    WORKGROUP      <1D>  UNIQUE      Registered
    ..__MSBROWSE__.<01>  GROUP       Registered
    ADMINISTRATOR  <03>  UNIQUE      Registered

    MAC Address = 00-0C-29-02-CB-45


C:\>
```

NetBIOS

NetBIOS name tables for both the local computer and remote computers, and the NetBIOS name cache. Nbtstat allows a refresh of the NetBIOS name cache and the names registered with Windows Internet Name Service (WINS).

Nbtstat command-line parameters are case-sensitive.

Syntax

nbtstat [-a RemoteName] [-A IPAddress] [-c] [-n] [-r] [-R] [-RR] [-s] [-S] [Interval]

Parameters

Used without parameters, nbtstat displays help.

-a RemoteName Displays the NetBIOS name table of a remote computer, where RemoteName is the NetBIOS computer name of the remote computer. The NetBIOS name table is the list of NetBIOS names that corresponds to NetBIOS applications running on that computer.

-A IPAddress Displays the NetBIOS name table of a remote computer, specified by the IP address (in dotted decimal notation) of the remote computer.

-c Displays the contents of the NetBIOS name cache, the table of NetBIOS names and their resolved IP addresses.

-n Displays the NetBIOS name table of the local computer. The status of Registered indicates that the name is registered either by broadcast or with a WINS server.

-r Displays NetBIOS name resolution statistics. On a Windows XP computer that is configured to use WINS, this parameter returns the number of names that have been resolved and registered using broadcast and WINS.

-R Purges the contents of the NetBIOS name cache and then reloads the #PRE-tagged entries from the Lmhosts file.

-RR Releases and then refreshes NetBIOS names for the local computer that is registered with WINS servers.

-s Displays NetBIOS client and server sessions, attempting to convert the destination IP address to a name.

-S Displays NetBIOS client and server sessions, listing the remote computers by destination IP address only.

Interval Redisplays selected statistics, pausing the number of seconds specified in Interval between each display. Press CTRL+C to stop redisplaying statistics. If this parameter is omitted, nbtstat prints the current configuration information only once.

/? - Displays help at the command prompt.

Route

route - show / manipulate the IP routing table

Tag	Description
route	[-v] [-A family] add [-net|-host] target [netmask Nm] [gw Gw] [metric N] [mss M] [window W] [irtt I] [reject] [mod] [dyn] [reinstate] [[dev] If]
route	[-v] [-A family] del [-net|-host] target [gw Gw] [netmask Nm] [metric N] [[dev] If]
route	[-V] [--version] [-h] [--help]

Route manipulates the kernel’s IP routing tables. Its primary use is to set up static routes to specific hosts or networks via an interface after it has been configured with the ifconfig(8) program.

When the add or del options are used, route modifies the routing tables. Without these options, route displays the current contents of the routing tables.

OPTIONS

Tag	**Description**
-A family	
	use the specified address family (eg 'inet'; use 'route --help' for a full list).
-F	operate on the kernel's FIB (Forwarding Information Base) routing table. This is the default.
-C	operate on the kernel's routing cache.
-v	select verbose operation.
-n	show numerical addresses instead of trying to determine symbolic host names. This is useful if you are trying to determine why the route to your nameserver has vanished.
-e	use netstat(8)-format for displaying the routing table. -ee will generate a very long line with all parameters from the routing table.
del	delete a route.
add	add a new route.
target	the destination network or host. You can provide IP addresses in dotted decimal or host/network names.
-net	the target is a network.
-host	the target is a host.
netmask NM	
	when adding a network route, the netmask to be used.
gw GW	route packets via a gateway. NOTE: The specified gateway must be reachable first. This usually means that you have to set up a static route to the gateway beforehand. If you specify the address of one of your local interfaces, it will be used to decide about the interface to which the packets should be routed to. This is a BSDism compatibility hack.

metric M	
	set the metric field in the routing table (used by routing daemons) to M.
mss M	set the TCP Maximum Segment Size (MSS) for connections over this route to M bytes. The default is the device MTU minus headers, or a lower MTU when path mtu discovery occured. This setting can be used to force smaller TCP packets on the other end when path mtu discovery does not work (usually because of misconfigured firewalls that block ICMP Fragmentation Needed)
window W	
	set the TCP window size for connections over this route to W bytes. This is typically only used on AX.25 networks and with drivers unable to handle back to back frames.
irtt I	set the initial round trip time (irtt) for TCP connections over this route to I milliseconds (1-12000). This is typically only used on AX.25 networks. If omitted the RFC 1122 default of 300ms is used.
reject	install a blocking route, which will force a route lookup to fail. This is for example used to mask out networks before using the default route. This is NOT for firewalling.
mod, dyn, reinstate	
	install a dynamic or modified route. These flags are for diagnostic purposes, and are generally only set by routing daemons.
dev If	force the route to be associated with the specified device, as the kernel will otherwise try to determine the device on its own (by checking already existing routes and device specifications, and where the route is added to). In most normal networks you won’t need this. If dev If is the last option on the command line, the word dev may be omitted, as it’s the default. Otherwise the order of the route modifiers (metric - netmask - gw - dev) doesn’t matter.

EXAMPLES

Tag	Description
route add -net 127.0.0.0	

	adds the normal loopback entry, using netmask 255.0.0.0 (class A net, determined from the destination address) and associated with the "lo" device (assuming this device was prviously set up correctly with ifconfig(8)).
route add -net 192.56.76.0 netmask 255.255.255.0 dev eth0	
	adds a route to the network 192.56.76.x via "eth0". The Class C netmask modifier is not really necessary here because 192.* is a Class C IP address. The word "dev" can be omitted here.
route add default gw mango-gw	
	adds a default route (which will be used if no other route matches). All packets using this route will be gatewayed through "mango-gw". The device which will actually be used for that route depends on how we can reach "mango-gw" - the static route to "mango-gw" will have to be set up before.
route add ipx4 sl0	
	Adds the route to the "ipx4" host via the SLIP interface (assuming that "ipx4" is the SLIP host).
route add -net 192.57.66.0 netmask 255.255.255.0 gw ipx4	
	This command adds the net "192.57.66.x" to be gatewayed through the former route to the SLIP interface.
route add -net 224.0.0.0 netmask 240.0.0.0 dev eth0	
	This is an obscure one documented so people know how to do it. This sets all of the class D (multicast) IP routes to go via "eth0". This is the correct normal configuration line with a multicasting kernel.
route add -net 10.0.0.0 netmask 255.0.0.0 reject	
	This installs a rejecting route for the private network "10.x.x.x."

Tracert / traceroute

Tracert: Determines the path taken to a destination by sending Internet Control Message Protocol (ICMP) Echo Request messages to the destination with incrementally increasing Time to Live (TTL) field values. The path displayed is the list of near-side router interfaces of the routers in the path between a source host and a destination. The near-side interface is the interface of the router that is closest to the sending host in the path. Used without parameters, tracert displays help.

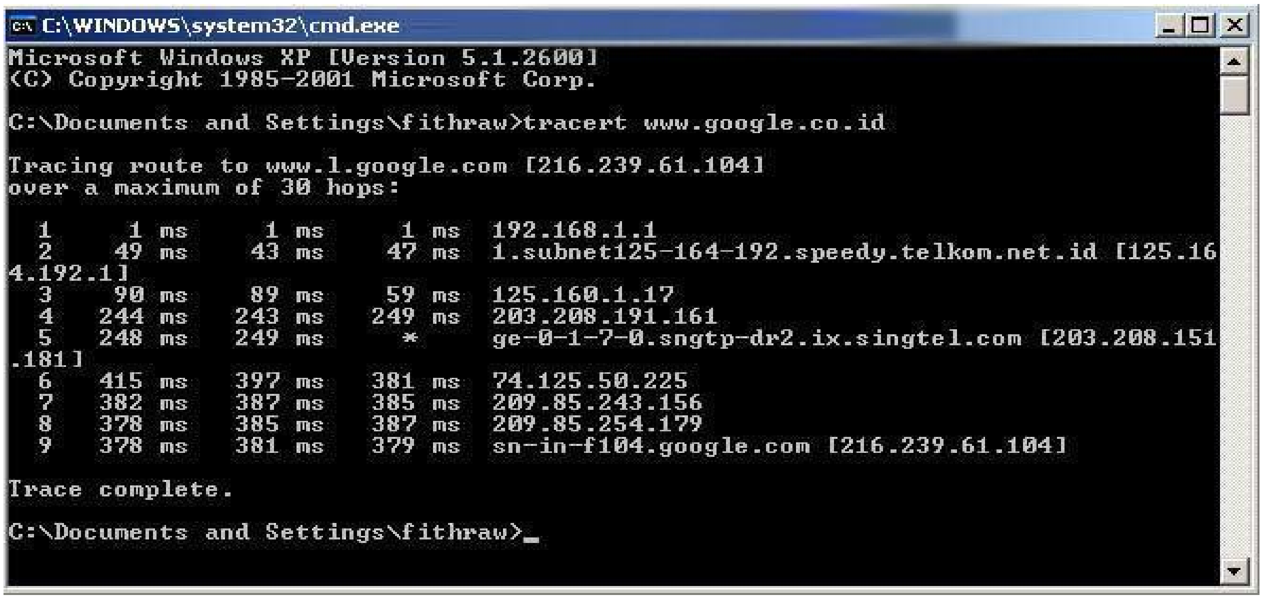
This diagnostic tool determines the path taken to a destination by sending ICMP Echo Request messages with varying Time to Live (TTL) values to the destination. Each router along the path is required to decrement the TTL in an IP packet by at least 1 before forwarding it.

Effectively, the TTL is a maximum link counter. When the TTL on a packet reaches 0, the router is expected to return an ICMP Time Exceeded message to the source computer. Tracert determines the path by sending the first Echo Request message with a TTL of 1 and incrementing the TTL by 1 on each subsequent transmission until the target responds or the maximum number of hops is reached. The maximum number of hops is 30 by default and can be specified using the -h parameter.

The path is determined by examining the ICMP Time Exceeded messages returned by intermediate routers and the Echo Reply message returned by the destination. However, some routers do not return Time Exceeded messages for packets with expired TTL values and are invisible to the tracert command. In this case, a row of asterisks (*) is displayed for that hop.

Examples:

To trace the path to the host named www.google.co.in type:
tracert www.google.co.in



```
Microsoft Windows XP [Version 5.1.2600]
(C) Copyright 1985-2001 Microsoft Corp.

C:\Documents and Settings\fithraw>tracert www.google.co.id

Tracing route to www.l.google.com [216.239.61.104]
over a maximum of 30 hops:

  1     1 ms     1 ms     1 ms  192.168.1.1
  2    49 ms    43 ms    47 ms  1.subnet125-164-192.speedy.telkom.net.id [125.16
4.192.1]
  3    90 ms    89 ms    59 ms  125.160.1.17
  4   244 ms   243 ms   249 ms  203.208.191.161
  5   248 ms   249 ms     *     ge-0-1-7-0.sngtp-dr2.ix.singtel.com [203.208.151
.181]
  6   415 ms   397 ms   381 ms  74.125.50.225
  7   382 ms   387 ms   385 ms  209.85.243.156
  8   378 ms   385 ms   387 ms  209.85.254.179
  9   378 ms   381 ms   379 ms  sn-in-f104.google.com [216.239.61.104]

Trace complete.

C:\Documents and Settings\fithraw>_
```

To trace the path to the host named www.google.com and prevent the resolution of each IP address to its name, type:
tracert -d www.google.com

To trace the path to the host named www.google.com and use the loose source route 10.12.0.1-10.29.3.1-10.1.44.1, type:
tracert -j 10.12.0.1 10.29.3.1 10.1.44.1 www.google.com

Syntax

tracert [-d] [-h MaximumHops] [-j HostList] [-w Timeout] [TargetName]

Parameters

-d Prevents tracert from attempting to resolve the IP addresses of intermediate routers to their names. This can speed up the display of tracert results.

-h MaximumHops Specifies the maximum number of hops in the path to search for the target (destination). The default is 30 hops.

-j HostList Specifies that Echo Request messages use the Loose Source Route option in the IP header with the set of intermediate destinations specified in HostList. With loose source routing, successive intermediate destinations can be separated by one or multiple routers. The maximum number of addresses or names in the host list is 9. The HostList is a series of IP addresses (in dotted decimal notation) separated by spaces.

-w Timeout Specifies the amount of time in milliseconds to wait for the ICMP Time Exceeded or Echo Reply message corresponding to a given Echo Request message to be received. If not received within the time-out, an asterisk (*) is displayed. The default time-out is 4000 (4 seconds).

Arp

Displays and modifies entries in the Address Resolution Protocol (ARP) cache, which contains one or more tables that are used to store IP addresses and their resolved Ethernet or Token Ring physical addresses. There is a separate table for each Ethernet or Token Ring network adapter installed on your computer.

Syntax

arp [-a [InetAddr] [-N IfaceAddr]] [-g [InetAddr] [-N IfaceAddr]] [-d InetAddr [IfaceAddr]] [-s InetAddr EtherAddr [IfaceAddr]]

Parameters

Used without parameters, ping displays help

-a [InetAddr] [**-N** IfaceAddr] Displays current ARP cache tables for all interfaces. To display the ARP cache entry for a specific IP address, use arp -a with the InetAddr parameter, where InetAddr is an IP address. To display the ARP cache table for a specific interface, use the -N IfaceAddr parameter where IfaceAddr is the IP address assigned to the interface. The -N parameter is case-sensitive.

-g [InetAddr] [**-N** IfaceAddr] Identical to -a.

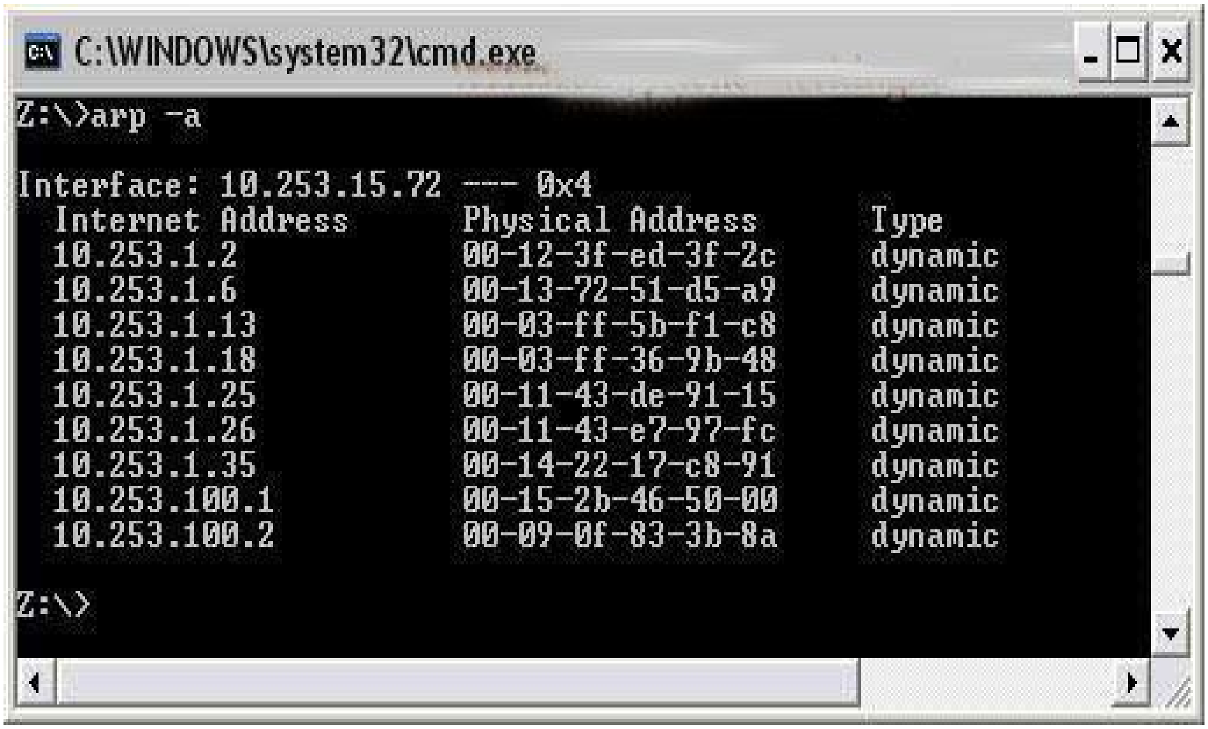
-d InetAddr [IfaceAddr] Deletes an entry with a specific IP address, where InetAddr is the IP address. To delete an entry in a table for a specific interface, use the IfaceAddr parameter where IfaceAddr is the IP address assigned to the interface. To delete all entries, use the asterisk (*) wildcard character in place of InetAddr.

-s InetAddr EtherAddr [IfaceAddr] Adds a static entry to the ARP cache that resolves the IP address InetAddr to the physical address EtherAddr. To add a static ARP cache entry to the table for a specific interface, use the IfaceAddr parameter where IfaceAddr is an IP address assigned to the interface.

Examples:

To display the ARP cache tables for all interfaces, type:
arp -a

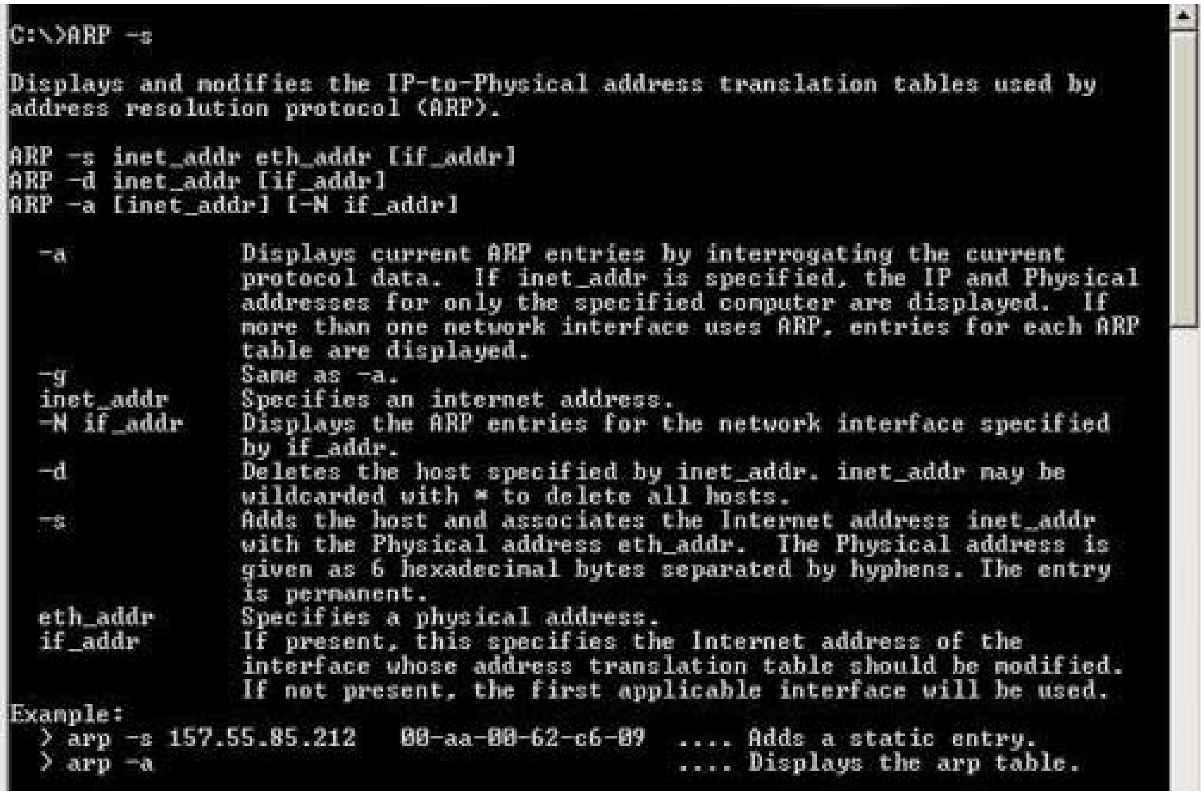
To display the ARP cache table for the interface that is assigned the IP address 10.0.0.99, type:



```
Z:\>arp -a

Interface: 10.253.15.72 --- 0x4
  Internet Address      Physical Address      Type
  10.253.1.2            00-12-3f-ed-3f-2c     dynamic
  10.253.1.6            00-13-72-51-d5-a9     dynamic
  10.253.1.13           00-03-ff-5b-f1-c8     dynamic
  10.253.1.18           00-03-ff-36-9b-48     dynamic
  10.253.1.25           00-11-43-de-91-15     dynamic
  10.253.1.26           00-11-43-e7-97-fc     dynamic
  10.253.1.35           00-14-22-17-c8-91     dynamic
  10.253.100.1          00-15-2b-46-50-00     dynamic
  10.253.100.2          00-09-0f-83-3b-8a     dynamic

Z:\>
```

arp -a -N 10.0.0.99

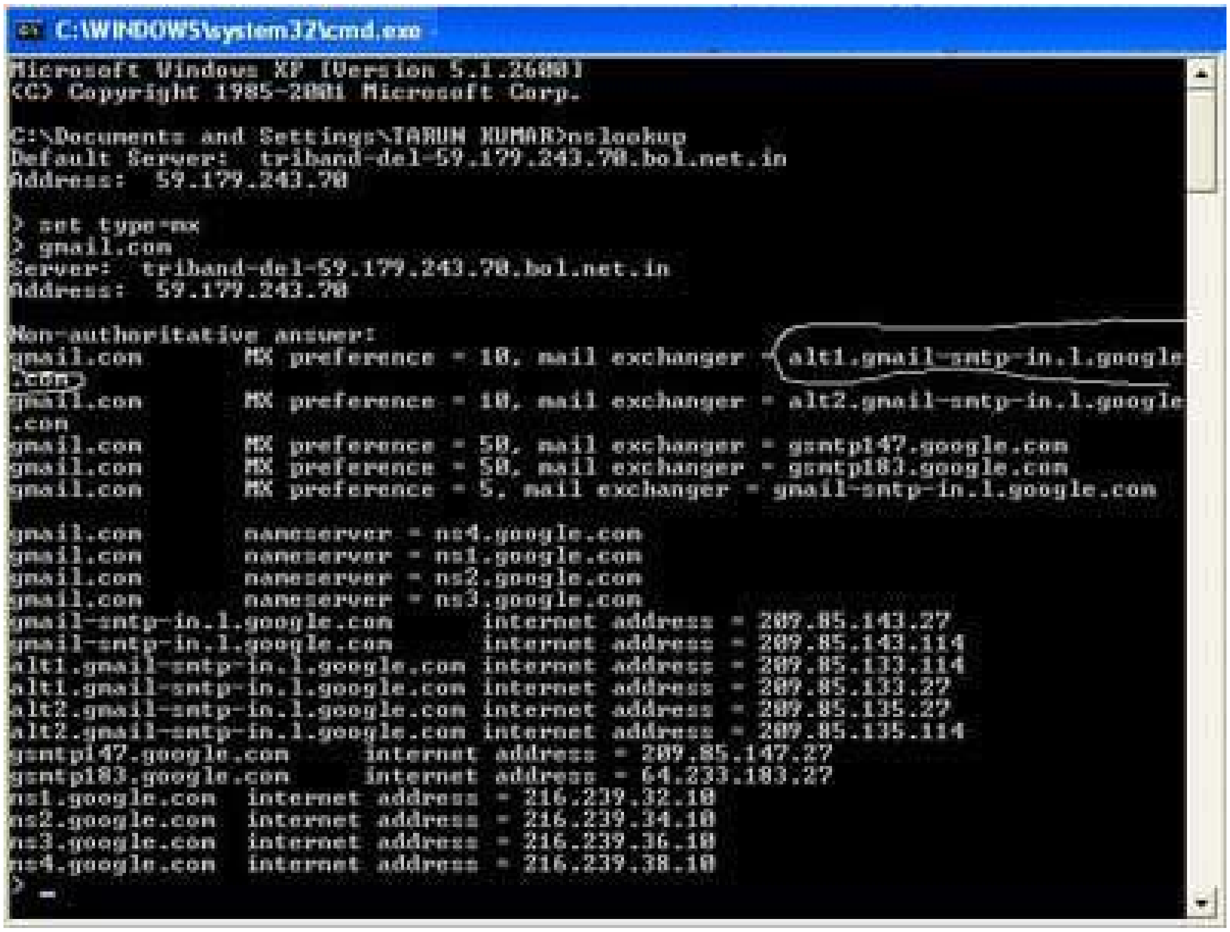
To add a static ARP cache entry that resolves the IP address 10.0.0.80 to the physical address 00-AA-00-4F-2A-9C, type:
arp -s 10.0.0.80 00-AA-00-4F-2A-9C



```
C:\>ARP -s

Displays and modifies the IP-to-Physical address translation tables used by
address resolution protocol (ARP).

ARP -s inet_addr eth_addr [if_addr]
ARP -d inet_addr [if_addr]
ARP -a [inet_addr] [-N if_addr]

  -a            Displays current ARP entries by interrogating the current
                protocol data.  If inet_addr is specified, the IP and Physical
                addresses for only the specified computer are displayed.  If
                more than one network interface uses ARP, entries for each ARP
                table are displayed.
  -g            Same as -a.
  inet_addr     Specifies an internet address.
  -N if_addr    Displays the ARP entries for the network interface specified
                by if_addr.
  -d            Deletes the host specified by inet_addr. inet_addr may be
                wildcarded with * to delete all hosts.
  -s            Adds the host and associates the Internet address inet_addr
                with the Physical address eth_addr.  The Physical address is
                given as 6 hexadecimal bytes separated by hyphens. The entry
                is permanent.
  eth_addr      Specifies a physical address.
  if_addr       If present, this specifies the Internet address of the
                interface whose address translation table should be modified.
                If not present, the first applicable interface will be used.
Example:
  > arp -s 157.55.85.212   00-aa-00-62-c6-09  .... Adds a static entry.
  > arp -a                                    .... Displays the arp table.
```

nslookup

Nslookup (Name Server lookup) is a UNIX shell command to query Internet domain name servers.



```
Microsoft Windows XP [Version 5.1.2600]
(C) Copyright 1985-2001 Microsoft Corp.

C:\Documents and Settings\TARUN KUMAR>nslookup
Default Server:  triband-del-59.179.243.70.bol.net.in
Address:  59.179.243.70

> set type=mx
> gmail.com
Server:  triband-del-59.179.243.70.bol.net.in
Address:  59.179.243.70

Non-authoritative answer:
gmail.com       MX preference = 10, mail exchanger = alt1.gmail-smtp-in.l.google
.com
gmail.com       MX preference = 10, mail exchanger = alt2.gmail-smtp-in.l.google
.com
gmail.com       MX preference = 50, mail exchanger = gsmtp147.google.com
gmail.com       MX preference = 50, mail exchanger = gsmtp183.google.com
gmail.com       MX preference = 5, mail exchanger = gmail-smtp-in.l.google.com

gmail.com       nameserver = ns4.google.com
gmail.com       nameserver = ns1.google.com
gmail.com       nameserver = ns2.google.com
gmail.com       nameserver = ns3.google.com
gmail-smtp-in.l.google.com      internet address = 209.85.143.27
gmail-smtp-in.l.google.com      internet address = 209.85.143.114
alt1.gmail-smtp-in.l.google.com internet address = 209.85.133.114
alt1.gmail-smtp-in.l.google.com internet address = 209.85.133.27
alt2.gmail-smtp-in.l.google.com internet address = 209.85.135.27
alt2.gmail-smtp-in.l.google.com internet address = 209.85.135.114
gsmtp147.google.com     internet address = 209.85.147.27
gsmtp183.google.com     internet address = 64.233.183.27
ns1.google.com  internet address = 216.239.32.10
ns2.google.com  internet address = 216.239.34.10
ns3.google.com  internet address = 216.239.36.10
ns4.google.com  internet address = 216.239.38.10
> _
```

Definitions

- **Nameserver:** These are the servers that the internet uses to find out more about the domain. Usually they are an ISP's computer.
- **Mailserver:** Where email is sent to.
- **Webserver:** The domains website.
- **FTPserver:** FTP is file transfer protocol, this server is where files may be stored.
- **Hostname:** The name of the host as given by the domain.
- **Real Hostname:** This is hostname that you get by reverse resolving the IP address, may be different to the given hostname.
- **IP Address:** Unique four numbered identifier that is obtained by resolving the hostname.

finger

finger looks up and displays information about system users.

finger syntax

```
finger [-lmsp] [user ...] [user@host ...]
```

Options

-s	Displays the user's login name, real name, terminal name and write status (as a "*****" after the terminal name if write permission is denied), idle time, login time, office location and office phone number. Login time is displayed as month, day, hours and minutes, unless more than six months ago, in which case the year is displayed rather than the hours and minutes. Unknown devices as well as nonexistent idle and login times are displayed as single asterisks.
-l	Produces a multi-line format displaying all of the information described for the **-s** option as well as the user's home directory, home phone number, login shell, mail status, and the contents of the files ".plan", ".project", ".pgpkey" and ".forward" from the user's home directory. Phone numbers specified as eleven digits are printed as "**+N-NNN-NNN-NNNN**". Numbers specified as ten or seven digits are printed as the appropriate subset of that string. Numbers specified as five digits are printed as "**xN-NNNN**". Numbers specified as four digits are printed as "**xNNNN**". If write permission is denied to the device, the phrase "**(messages off)**" is appended to the line containing the device name. One entry per user is displayed with the **-l** option; if a user is logged on multiple times, terminal information is repeated once per login. Mail status is shown as "**No Mail.**" if there is no mail at all, "**Mail last read DDD MMM ## HH:MM YYYY (TZ)**" if the person has looked at their mailbox since new mail arriving, or "New mail received ...", " Unread since ..." if they have new mail.

-p	Prevents the **-l** option of **finger** from displaying the contents of the "**.plan**", "**.project**" and "**.pgpkey**" files.
-m	Prevent matching of usernames. The *user* is usually a login name; however, matching will also be done on the users' real names, unless the **-m** option is supplied. All name matching performed by finger is case insensitive.

If no options are specified, **finger** defaults to the **-l** style output if operands are provided, otherwise to the **-s** style. Note that some fields may be missing, in either format, if information is not available for them.

If no arguments are specified, **finger** will print an entry for each user currently logged into the system.

Finger may be used to look up users on a remote machine. The format is to specify a user as "**user@host**", or "**@host**", where the default output format for the former is the **-l** style, and the default output format for the latter is the **-s** style. The **-l** option is the only option that may be passed to a remote machine.

If standard output is a socket, **finger** will emit a carriage return (**^M**) before every linefeed (**^J**). This format is for processing remote finger requests when invoked by **fingerd**, the finger daemon.

Files

~/.nofinger	If **finger** finds this file in a user's home directory, it will, for **finger** requests originating outside the local host, firmly deny the existence of that user. For this to work, the **finger** program, as started by **fingerd**, must be able to see the **.nofinger** file. This generally means that the home directory containing the file must have the **other-users-execute** bit set (**o+x**). (See chmod). If you use this feature for privacy, please test it with "**finger @localhost**" before relying on it, just in case.
~/.plan **~/.project** **~/.pgpkey**	These files are printed as part of a long-format request. The **.plan** file may be of any length.

finger Examples

```
finger -p ch
```

Display information about the user **ch**. Output will appear similar to the following:

```
Login name: admin
In real life: Computer Hope
On since Feb 11 23:37:16 on pts/7 from domain.computerhope.com
28 seconds Idle Time
Unread mail since Mon Feb 12 00:22:52 2001
```

Nmap / Port Scan

The **Nmap** aka Network Mapper is an open source and a very versatile tool for **Linux** system/network administrators. **Nmap** is used for exploring networks, perform security scans, network audit and finding open ports on remote machine

Example

```
$ nmap 207.218.248.50
```

Sample outputs:

Output

```
Starting Nmap 5.00 ( http://nmap.org ) at 2012-11-18 14:41 IST

Interesting ports on 207.218.248.50:

Not shown: 997 closed ports

PORT   STATE SERVICE

23/tcp open telnet

53/tcp open domain

80/tcp open http

MAC Address: 55:87:06:25:65:FC (Unknown)

Nmap done: 1 IP address (1 host up) scanned in 1.28 seconds
```

Other Examples

3) Scan an IP

```
nmap 207.218.248.50
```

4) Scan a range of IP address

```
nmap 207.218.248.5-45
```

5) Scan entire subnet

```
nmap 192.168.2.0/24
```

6) Ping only scan

```
nmap -sP 207.218.248.50
```

7) Scan and do traceroute

```
nmap –traceroute IP-ADDRESS

nmap –traceroute DOMAIN-NAME-HERE
```

8) TCP SYN Scan

```
nmap -sS 207.218.248.50
```

9) UDP Scan

```
nmap -sU 207.218.248.50
```

10) IP protocol scan

```
nmap -sO 207.218.248.50
```

11) Scan port 80, 25, 443

```
nmap -p 80 207.218.248.50

nmap -p http 207.218.248.50

nmap -p 25 207.218.248.50

nmap -p smtp 207.218.248.50

nmap -p 443 207.218.248.50

nmap -p 80,24,443 207.218.248.50
```

12) Scan port ranges

```
nmap -p 512-1024 207.218.248.50
```

13) Scan for Operating System Detection

```
nmap -O 207.218.248.50
```

```
nmap -O –osscan-guess 207.218.248.50
```

14) Scan for application server version

```
nmap -sV 207.218.248.50
```

15) Scan a host name

```
nmap google.com
```

16) Scan a host name with more info

```
nmap -v google.com
```

17) Scan a host when protected by the firewall

```
nmap -PN 207.218.248.50
```

```
nmap -PN google.com
```

18) Perform a fast scan

```
nmap -F 207.218.248.50
```

19) Show host interfaces and routes

```
nmap –iflist
```

20) Scan for IP protocol

This type of scan allows you to determine which IP protocols (TCP, ICMP, IGMP, etc.) are supported by target machines:

```
nmap -sO 207.218.248.50
```