Useful DOS Commands

Analyzing and troubleshooting computers and network can be difficult when problem arise. One can look at the Windows event viewer for warning and error event messages. Enough times these event messages can be informative. But it may not show the cause for errors or performance issues without further investigation. The network could be bogged down between computers and the server. The computer system could be problematic beside possible hard drive warnings and CLSID/APPID issues.

The Windows operating system contain built-in tools as networking command line utilities that users and administrators could use to help troubleshoot. There are 11 built-in commands but the NbtStat has been depreciated since 2000 and NetDiag has been depreciated since Windows XP. Both will not be talked about here.

- Ping
- NetStat
- NbtStat - depreciated
- ARP
- Hostname
- Tracert
- Ipconfig
- NSLookup
- Route
- PathPing
- NetDiag - depreciated

The information below sufficiently describe these most commonly used commands. For more in-depth use, additional search may be required.

Ping (Packet Internet Groper)

Ping command is the most familiar and commonly used utility. It helps in determining the possible issues with the network both local and Internet. Ping is used to test the ability of one network host to communicate with another basically at the TCP/IP level. To use, open the command prompt then enter the Ping command, followed by the domain name or the IP address of the destination host.

TCP/IP (**Transmission Control Protocol/Internet Protocol)** is a set of rules (protocols to be exact) that governs communication among all computers on the network and Internet. TCP/IP dictates how information should be packaged into bundles of information called packets, then use to send, and receive network locally and Internet wise.

In assuming that there are no network problems or firewalls preventing the ping from completing, the remote host on the LAN or WAN will respond to the ping with standard

four packets and status. Receiving these packets confirms that a valid and functional network path exists between the two hosts.

Nbtstat syntax

ping [-t] [-a] [-n count] [-l size] [-f] [-i TTL] [-v TOS] [-r count] [-s count] [[-j host-list] | [-k host-list]] [-w timeout] [-R] [-S srcaddr] [-c compartment] [-p] [-4] [-6] target_name

Switches	Description	IPV4	IPV6	Depreciated
-t	Ping specified host 4 times. To see statistics and continue - type Control-Break; To stop - type Control-C.	X	X	
-a	Resolve addresses to hostnames	X	X	
-n count	Number of echo requests to send	X	X	
-l size	Send buffer size	X	X	
-i TTL	Time To Live	X	X	
-v TOS	Type Of Service	X		X
-r count	Record route for count	X		
-s count	Timestamp for count hops	X		
-j host-list	Loose source route along host-list	X		
-k host-list	Strict source route along host-list	X		
-w timeout	Timeout in milliseconds to wait for each reply	X	X	
-R	Use routing header to test reverse route also. Per RFC 5095 use of routing header has been deprecated. Some systems may drop echo requests if this header is used.		X	X
-S srcaddr	Source address to use	X	X	
-c	compartment Routing compartment identifier	X	X	
-p	Ping a Hyper-V Network Virtualization provider address	X	X	
-4	Force using IPv4	X		
-6	Force using IPv6.		X	

Ping example

Standard ping use
C:\Users\Randy>ping www.cnn.com

Pinging turner-tls.map.fastly.net [151.101.1.67] with 32 bytes of data:
Reply from 151.101.1.67: bytes=32 time=38ms TTL=47
Reply from 151.101.1.67: bytes=32 time=41ms TTL=47
Reply from 151.101.1.67: bytes=32 time=37ms TTL=47

Reply from 151.101.1.67: bytes=32 time=46ms TTL=47

Ping statistics for 151.101.1.67:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 37ms, Maximum = 46ms, Average = 40ms

C:\Users\Randy>

Ping use with large buffer size with "-l" option
C:\WINDOWS\system32>ping -l 50000 www.cnn.com

Pinging turner-tls.map.fastly.net [151.101.1.67] with 50000 bytes of data:
Reply from 151.101.1.67: bytes=50000 time=114ms TTL=47
Reply from 151.101.1.67: bytes=50000 time=111ms TTL=47
Reply from 151.101.1.67: bytes=50000 time=115ms TTL=47
Reply from 151.101.1.67: bytes=50000 time=111ms TTL=47

Ping statistics for 151.101.1.67:
    Packets: Sent = 4, Received = 4, Lost = 0 (0% loss),
Approximate round trip times in milli-seconds:
    Minimum = 111ms, Maximum = 115ms, Average = 112ms

C:\WINDOWS\system32>

NetStat (Network Statistics)

Netstat is a command-line network utility tool that displays network connections for the Transmission Control Protocol (both incoming and outgoing), routing tables, number of network interface (network interface controller or software-defined network interface), and network protocol statistics. It is used for finding problems in the network and to determine the amount of traffic on the network as a performance measurement.

This article will focus on its usage for Microsoft Windows since NetStat is mostly obsolete for Linux. For example for Linux:

- netstat (part of "net-tools") is superseded by `ss` as part of iproute2.
- Replacement for `netstat -r` is `ip route`.
- Replacement for `netstat -i` is `ip -s link`
- Replacement for `netstat -g` is `ip maddr`, all of which are recommended instead

When experiencing problems with network communications, network statistics may help identify the its root cause. This command has a number of different functions, but the

most useful of these is to display network summary information for the device. To see this type of summary information, just type NetStat -e.

Nbtstat syntax

NETSTAT [-a] [-b] [-e] [-f] [-n] [-o] [-p proto] [-r] [-s] [-x] [-t] [interval]

-a	Displays all connections and listening ports.
-b	Displays the executable involved in creating each connection or listening port. In some cases, well-known executables host multiple independent components, and in these cases the sequence of components involved in creating the connection or listening port is displayed. In this case the executable name is in [] at the bottom, on top is the component it called, and so forth until TCP/IP was reached. Note that this option can be time-consuming and will fail unless you have sufficient permissions.
-e	Displays Ethernet statistics. This may be combined with the -s option.
-f	Displays Fully Qualified Domain Names (FQDN) for foreign addresses.
-n	Displays addresses and port numbers in numerical form.
-o	Displays the owning process ID associated with each connection.
-p proto	Shows connections for the protocol specified by proto; proto may be any of: TCP, UDP, TCPv6, or UDPv6. If used with the -s option to display per-protocol statistics, proto may be any of: IP, IPv6, ICMP, ICMPv6, TCP, TCPv6, UDP, or UDPv6.
-r	Displays the routing table.
-s	Displays per-protocol statistics. By default, statistics are shown for IP, IPv6, ICMP, ICMPv6, TCP, TCPv6, UDP, and UDPv6; the -p option may be used to specify a subset of the default.
-t	Displays the current connection offload state.
-x	Displays NetworkDirect connections, listeners, and shared endpoints.
-y	Displays the TCP connection template for all connections. Cannot be combined with the other options.
interval	Redisplays selected statistics, pausing interval seconds between each display. Press Ctrl+C to stop redisplaying statistics. If omitted, netstat will print the current configuration information once.

Netstat examples

Tip: If you have network applications open, such as the Internet browser you are using to view a web page, additional items will be listed when you run "netstat" and the "netstat -a" command. So, you may see items from a computer in your list. If you want a true listing of background Internet connections, close all programs and run the netstat command. It is also normal to see one or more 0.0.0.0 and 127.0.0.1 addresses.

Entering "netstat" in the command prompt will displays all local network information. Below is an example of how this may look.

Protocol	Local Address	Foreign Address	State

TCP	hope:4409	www.computerhope.com:telnet	ESTABLISHED
TCP	hope:3708	multicity.com:80	CLOSE_WAIT
TCP	hope:4750	www.google.com:80	CLOSE_WAIT

Entering in the command prompt the following:

"netstat –an" -- displays all connections on the computers in numerical format, only displaying the local and foreign IP addresses.

"netstat 5" -- will run refresh the information after 5 seconds until Ctrl+C is entered

ARP (Address Resolution Protocol)

The ARP command allows user to display and modify the Address cache. An *ARP cache* is a simple mapping of IP addresses to MAC addresses. Each time a computer's TCP/IP stack uses ARP to determine the Media Access Control (MAC) address for an IP address, it records the mapping in the ARP cache so that future ARP lookups go faster.

Windows devices maintain an ARP cache, which contains the results of recent ARP queries. You can see the contents of this cache by using the ARP -a command. If you are having problems communicating with one specific host, you can append the remote host's IP address to the ARP -a command. Displayed example below using the ARP -a.

arp syntax

ARP -a [inet_addr] [-N if_addr]

ARP -s inet_addr eth_adr [if_addr]

ARP -d inet_addr [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.
-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.

Hostname example

```
C:\Users\Randy>arp -a

Interface: 192.168.12.155 --- 0xc
  Internet Address      Physical Address      Type
  192.168.12.100        00-11-32-74-01-cd     dynamic
  192.168.12.122        00-0e-8e-6b-55-e8     dynamic
  192.168.12.150        90-f1-aa-ea-79-d0     dynamic
  192.168.12.154        24-77-03-b3-29-84     dynamic
  192.168.12.185        fc-15-b4-9e-15-d2     dynamic
  192.168.12.255        ff-ff-ff-ff-ff-ff     static
  224.0.0.22            01-00-5e-00-00-16     static
  224.0.0.251           01-00-5e-00-00-fb     static
  224.0.0.252           01-00-5e-00-00-fc     static
  239.255.255.250       01-00-5e-7f-ff-fa     static
  255.255.255.255       ff-ff-ff-ff-ff-ff     static

C:\Users\Randy>
```

Tracert (Trace route)

Tracert is a utility for examining the path to a remote host. Tracert works similarly to Ping. The major difference is that Tracert sends a series of ICMP (Internet Control Message Protocol) echo requests, and the request's TTL increased by 1 each time. This allows the utility to display the routers through which packets are passing to be identified. When possible, Windows displays the duration and IP address or fully qualified domain name of each hop.

ICMP (Internet Control Message Protocol) is an error-reporting protocol network device like routers use to generate error messages to the source IP address when network problems prevent delivery of IP packets. ICMP creates and sends messages to the source IP address indicating the gateway, series of Internet routers, service or host cannot be reached for packet delivery. Any IP network device has the capability to send, receive or process ICMP messages.

tracert syntax

tracert [-d] [-h maximum_hops] [-j host-list] [-w timeout] [-R] [-S srcaddr] [-4] [-6] target_name

Options:

-d	Do not resolve addresses to hostnames.
-h maximum_hops	Maximum number of hops to search for target.
-j host-list	Loose source route along host-list (IPv4-only).
-w timeout	Wait timeout milliseconds for each reply.
-R	Trace round-trip path (IPv6-only).
-S srcaddr	Source address to use (IPv6-only).
-4	Force using IPv4.
-6	Force using IPv6.

TRACERT is used to find out where a packet stopped on the network. In the following example, the default gateway has found that there is no valid path for the host on 22.110.0.1. Probably, either the router has a configuration problem, router failed, or the 22.110.0.0 network does not exist, reflecting a bad IP address. Below is the usage sample.

Tracert example

Failed trace
C:\>tracert 22.110.0.1

The output from the command:

```
Tracing route to 22.110.0.1 over a maximum of 30 hops
------------------------------------------------------
1    157.54.48.1  reports: Destination net unreachable.

Trace complete.
```

Successful trace
C:\>tracert 11.1.0.1

The output from the command:

```
Tracing route to 11.1.0.1 over a maximum of 30 hops
------------------------------------------------------
1     2 ms      3 ms      2 ms     157.54.48.1
2    75 ms     83 ms      88 ms      11.1.0.67
3    73 ms     79 ms      93 ms      11.1.0.1

Trace complete.
```

IPCONFIG (Internet Protocol Configuration)

Ipconfig is a very common command line tool used to get information of and manage the local computer network connections. This command can be utilized to verify a network connection as well as to verify your network settings. Entering just “ipconfig” at the Windows command prompt will present the IP address (IPV4 and IPV6), subnet mask, and default gateway for all network interface in the computer or server. There are three main commands: "all", "release", and "renew". The syntax table show much more as design to help resolve network connection issues.

1. Ipconfig – shows basic IP information
2. ipconfig /all – displays all current TCP/IP and DNS detail information
3. ipconfig /release - release all matching connections
4. ipconfig /renew – renewal all adapters IP settings. Refreshes Dynamic Host Configuration Protocol (DHCP) and Domain Name System (DNS)

ipconfig syntax

ipconfig [/? | /all | /renew [adapter] | /release [adapter] | /renew6 [adapter] | /release6 [adapter] | /flushdns | /displaydns | /registerdns | /showclassid adapter | /setclassid adapter [classid] | /showclassid6 adapter | /setclassid6 adapter [classid]]
where adapter Connection name (wildcard characters * and ?

Commands	Description	IPv4	IPv6
/?	Displays the help message of commands listed below	x	x
/all	Display full configuration information.	x	x
/release	Release the address for the specified adapter.	x	
/release6	Release the address for the specified adapter.		x
/renew	Renew the address for the specified adapter.	x	
/renew6	Renew the address for the specified adapter.		x
/flushdns	Purges the DNS Resolver cache.	x	x
/registerdns	Refreshes all DHCP leases and re-registers DNS names	x	x
/displaydns	Display the contents of the DNS Resolver Cache.	x	x
/showclassid	Displays all the DHCP class IDs allowed for adapter.	x	x
/setclassid	Modifies the DDHCP class id.		x
/showclassid6	Displays all DHCP class IDs allowed for adapter.		x
/setclassid6	Modifies the DHCP class id.		x

Ipconfig example

Calling ipconfig for simple
C:\Users\randy>ipconfig

```
Windows IP Configuration

Ethernet adapter Ethernet:

   Connection-specific DNS Suffix  . :
   Link-local IPv6 Address . . ….. . . : fe80::85bf:c422:e5c:a0d0%10
   IPv4 Address. . . . . . . . . . . . . . : 192.168.12.155
   Subnet Mask . . . . . . . . . .  . . . .  : 255.255.255.0
   Default Gateway . . . . . . .  . . . .  : 192.168.12.122

Tunnel adapter Local Area Connection* 10:

   Connection-specific DNS Suffix  . :
   IPv6 Address. . . . . . . . . . .. . . . : 2001:0:9d38:953c:c41:1722:f5f5:7f64
   Link-local IPv6 Address . . . . . . . . : fe80::c41:1722:f5f5:7f64%11
   Default Gateway . . . . . . . .  . . . . : ::

C:\Users\randy>C:\Users\randy>

C:\Users\randy>ipconfig /all

Windows IP Configuration

   Host Name . . . . . . . . . . . . : firefly
   Primary Dns Suffix  . . . . . . . :
   Node Type . . . . . . . . . . . . : Hybrid
   IP Routing Enabled. . . . . . . . : No
   WINS Proxy Enabled . . . . . . : No

Ethernet adapter Ethernet:

   Connection-specific DNS Suffix  . . .  :
   Description . . . . . . . . . . . . . . . . . . . : Intel(R) Ethernet Connection I217-LM
   Physical Address. . . . . . . . . . . . . . . . . : e5-b7-46-E2-66-49
   DHCP Enabled. . . . . . . . . . . . . . . . . . . : Yes
   Autoconfiguration Enabled . . . . . . . . . . . . : Yes
   Link-local IPv6 Address . . . . . . . . . . . . . : fe80::85bf:c422:e5c:a0d0%10(Preferred)
   IPv4 Address. . . . . . . . . . . . . . . . . . . : 192.1168.10.155(Preferred)
   Subnet Mask . . . . . . . . . . . . . . . . . . . .: 255.255.255.0
   Lease Obtained. . . . . . . . . . . . . . . . . . : Wednesday, May 9, 2018 7:23:16 AM
   Lease Expires . . . . . . . . . . . . . . . . . . : Thursday, May 10, 2018 7:23:24 AM
   Default Gateway . . . . . . . . . . . . . . . . . : 192.168.10.1
   DHCP Server . . . . . . . . . . . . . . . . . . . .: 192.168.10.1
   DHCPv6 IAID . . . . . . . . . . . . . . . . . . . .: 66629974
   DHCPv6 Client DUID. . . . . . . . . . . . . . . . : 00-01-00-01-1A-EB-04-1A-F8-B1-56-E2-7F-29
```

```
   DNS Servers . . . . . . . . . . . . . . . . . . . : 192.168.10.1
                                                       8.8.8.8
   NetBIOS over Tcpip. . . . . . . . . . . . . . . .: Enabled

Tunnel adapter Local Area Connection* 10:

   Connection-specific DNS Suffix  . . :
   Description . . . . . . . . . . . . . . . . . . . .: Microsoft Teredo Tunneling Adapter
   Physical Address. . . . . . . . . . . . . . . . . .: 00-00-00-00-00-00-00-E0
   DHCP Enabled. . . . . . . . . . . . . . . . . . . : No
   Autoconfiguration Enabled . . . . . . . . . . . .: Yes
   IPv6 Address. . . . . . . . . . . . . . . . . . . .: 2001:0:9d38:953c:c41:1722:f5f5:7f64(Preferred)
   Link-local IPv6 Address . . . . . . . . . . . . . : fe80::c41:1992:e4f5:7f64%11(Preferred)
   Default Gateway . . . . . . . . . . . . . . . . . .:
   DHCPv6 IAID . . . . . . . . . . . . . . . . . . . .: 184549376
   DHCPv6 Client DUID. . . . . . . . . . . . . . . . .: 00-01-00-01-1A-BB-04-11-F8-C3-56-E2-7F-22
   NetBIOS over Tcpip. . . . . . . . . . . . . . . . .: Disabled

C:\Users\randy>
```

The ipconfig command can do much more than just display IP address configuration information. It also contains options that can help you to troubleshoot problems related to DNS and DHCP. For example, entering the ipconfig /flushdns command purges the contents of the computer's DNS resolver cache.

NSLookup (Name Server Lookup)

Nslookup is a useful tool for troubleshooting DNS name resolution problems, such as host name resolution. When starting nslookup, it shows the host name and IP address of the DNS server that is configured for the local system, and then display a nslookup command prompt for further queries. From there, you can type host names to see if the DNS server is able to resolve the specified host name.

nslookup does not use the operating system's local Domain Name System resolver library to perform its queries, and thus may behave differently from dig (used in Mac OS X and Linux). Additionally, vendor-provided versions can confuse matters by using or including output of other sources of name information (such as host files, Network Information Service).

Nslookup is a useful tool for troubleshooting DNS name resolution problems, such as host name resolution. nslookup operates in interactive or non-interactive mode.

1. Interactive mode - Invoke nslookup without arguments or with first argument is - (minus sign) directly followed by a hostname or Internet address of a name

server. This will present the nslookup prompt (>) followed by the default server with its IP address or the server with its IP address respectively. Example:

Minus sign only:

```
C:\WINDOWS\system32>nslookup -
Default Server:  rns01.charter.com
Address:  71.10.216.1

>
```

Minus sign with domain name:

```
C:\WINDOWS\system32>nslookup -google.com
*** Invalid option: google.com
Default Server:  rns01.charter.com
Address:  71.10.216.1

>
```

The prompt will all user to select options in the syntax list below.

2. Non-interactive mode – Invoke nslookup followed by a host name or Internet IP address. This will present its server with its IP address and the non-authoriitative answer showing its name and the addresses. Example:

```
C:\WINDOWS\system32>nslookup google.com
Server:  rns01.charter.com
Address:  71.10.216.1

Non-authoritative answer:
Name:    google.com
Addresses:  2607:f8b0:4002:807::200e
          172.217.0.78

C:\WINDOWS\system32>
```

nslookup syntax

```
Usage:
  nslookup [-opt ...]            # interactive mode using default server
  nslookup [-opt ...] - server    # interactive mode using 'server'
  nslookup [-opt ...] host        # just look up 'host' using default server
  nslookup [-opt ...] host server # just look up 'host' using 'server'
```

NAME	print info about the host/domain NAME using default server
NAME1 NAME2	as above, but use NAME2 as server
help or ?	print info on common commands
set OPTION	set an option **all** - print options, current server and host **[no]debug -** print debugging information **o]d2 -** print exhaustive debugging information **[no]defname** - append domain name to each query **[no]recurse** - ask for recursive answer to query **[no]search** - use domain search list **[no]vc** - always use a virtual circuit **domain=NAME -** set default domain name to NAME **srchlist=N1[/N2/.../N6] -** set domain to N1 & search list to N1,N2, etc. **root=NAME -** set root server to NAME **retry=X -** set number of retries to X **timeout=X -** set initial time-out interval to X seconds **Type=X** - set query type (ex. A,ANY,CNAME,MX,NS, PTR,SOA,SRV) **queryType=X -** same as type **class=X -** set query class (ex. IN (Internet), ANY) **[no]msxfr -** use MS fast zone transfer **ixfrver=X -** current version to use in IXFR transfer request
server NAME	set default server to NAME, using current default server
lserver NAME	set default server to NAME, using initial server
finger [USER]	finger the optional NAME at the current default host
root	set current default server to the root
ls [opt] DOMAIN [> FILE]	list addresses in DOMAIN (optional: output to FILE) **-a** list canonical names and aliases **-d** list all records **-t type** list records of the given type (e.g., A,CNAME,MX,NS,PTR
view FILE	sort an 'ls' output file and view it with pg
exit	exit the program

set an option
all - print options, current server and host
[no]debug - print debugging information

o]d2 -	print exhaustive debugging information
[no]defname -	append domain name to each query
[no]recurse -	ask for recursive answer to query
[no]search -	use domain search list
[no]vc -	always use a virtual circuit
domain=NAME -	set default domain name to NAME
srchlist=N1[/N2/.../N6]	set domain to N1 and search list to N1,N2, etc.
root=NAME -	set root server to NAME
retry=X -	set number of retries to X
timeout=X -	set initial time-out interval to X seconds
Type=X -	set query type (ex. A,ANY,CNAME,MX,NS, PTR,SOA,SRV)
queryType=X -	same as type
class=X -	set query class (ex. IN (Internet), ANY)
[no]msxfr -	use MS fast zone transfer
ixfrver=X -	current version to use in IXFR transfer request

Note: if you are having trouble getting true DNS information then may have to flush the DS in your system.

Run command as administrator then type the following in order.

ipconfig /flushdns
ipconfig /registerdns
ipconfig /release
ipconfig /renew
NETSH winsock reset catalog
NETSH int ipv4 reset reset.log
NETSH int ipv6 reset reset.log
Exit

Route

Route is a command used to view and manipulate the IP routing table in both for Microsoft Windows Unix-like. Manual manipulation of the routing table is characteristic of static routing. Static routing is a form of routing that occurs when a router uses manually-configured routing entries, rather than information from a dynamic routing traffic.

For Linux, the iproute2 is used to view and define a computer on the network. In the past `ipconfig` and `route` commands are used together to connect a computer to a

network, and to define routes between computer networks. The later Linux kernels have deprecated `ipconfig` and `route` commands.

A routing table from the Route commands is a data table stored in a router or a networked computer that lists the routes to network destinations. This data table shows the network destination, netmask, gateway interface, and in some cases, metrics (distances by hops). The routing table contains information about the topology of the immediate network.

Common use for Route would be display the local routing table, deleting a specified route, and adding a specified route.

route syntax

ROUTE [-f] [-p] [-4|-6] command [destination] [MASK netmask] [gateway]

[METRIC metric] [IF interface]

-f	Clears the routing tables of all gateway entries. If this is used in conjunction with one of the commands, the tables are cleared prior to running the command.
-p	When used with the ADD command, makes a route persistent across boots of the system. By default, routes are not preserved when the system is restarted. When used with the PRINT command, displays the list of registered persistent routes. Ignored for all other commands, which always affect the appropriate persistent routes. This option is not supported Windows'95. command
-4	Force using IPv4.
-6	Force using IPv6.
command	One of these: PRINT Prints a route ADD Adds a route DELETE Deletes a route CHANGE Modifies an existing route destination
destination	Specifies the host.
MASK	Specifies that the next parameter is the 'netmask' value.
netmask	Specifies a subnet mask value for this route entry. If not specified, it defaults to 255.255.255.255.
gateway	Specifies gateway.
interface	the interface number for the specified route.
METRIC	Specifies the metric, ie. cost for the destination.

All symbolic names used for destination are looked up in the network database file NETWORKS. The symbolic names for gateway are looked up in the host name database file HOSTS.

If the command is PRINT or DELETE. Destination or gateway can be a wildcard, ('*'), or the gateway argument may be omitted.

If Dest contains a * or ?, it is treated as a shell pattern, and only matching destination routes are printed. The '*' matches any string, and '?' matches any one char. Examples: 157.*.1, 157.*, 127.*, *224*.

Pattern match is only allowed in PRINT command.
Diagnostic Notes:
Invalid MASK generates an error, that is when (DEST & MASK) != DEST.
Example> route ADD 157.0.0.0 MASK 155.0.0.0 157.55.80.1 IF 1

The route addition failed: The specified mask parameter is invalid.
(Destination & Mask) != Destination.

Route examples

To display the routing table

1. Run command prompt as administrator
2. Type route print then press Enter
3. Observe active routes by destinations, network mask, gateway, interface, and metric.

To delete a route

1. From the print result of the routing table, observe network destination 0.0.0.0 listed.
2. Type route print then press Enter
3. Observe active routes by destinations, network mask, gateway, interface, and metric.
4. When pinging 8.8.8.8 to test internet connectivity, it should be good.
5. Type ‘route delete 0.0.0.0’
6. Pinging 8.8.8.8 again and notice its failure

To add a route

1. From same elevated command prompt, type ‘route add 0.0.0.0 mask 0.0.0.0 <local network gateway IP>.

When ping 8.8.8.8 to test internet connectivity, it should be good.

PathPing

This network tool utility combines the functionality of ping with that of tracert. It provides information about network latency and loss at intermediate hops between a source and destination. Basically, pathping sends multiple echo request messages to get ping like statistics for each node (router) between the source and destination. The time period is dependent on how many nodes are between the start and end host. The advantages of *PathPing* over individually using ping and tracert are that each node is pinged as the result of a single command. The behavior of nodes is studied over an extended time

period, rather than the default ping sample of four messages or default *traceroute* single route trace. The disadvantages are:

1. It can take in the range of 300-500 seconds to get the statistics.
2. Many of the nodes will not ping back causing wait times to virtually report the loses.
3. The nodes showing ping loses offers no support to the true latency statistics.

pathping syntax

pathping [-g host-list] [-h maximum_hops] [-i address] [-n] [-p period] [-q num_queries] [-w timeout] [-P] [-R] [-T] [-4] [-6] target_name

Options:

-g host-list	Loose source route along host-list.
-h maximum_hops	Maximum number of hops to search for target.
-i address	Use the specified source address.
-n	Do not resolve addresses to hostnames.
-p period	Wait period milliseconds between pings.
-q num_queries	Number of queries per hop.
-w timeout	Wait timeout milliseconds for each reply.
-P	Test for RSVP PATH connectivity.
-R	Test if each hop is RSVP aware.
-T	Test connectivity to each hop with Layer-2 priority tags.
-4	Force using IPv4.
-6	Force using IPv6.

pathping example

```
C:\WINDOWS\system32>pathping -h 20 www.cnn.com

Tracing route to turner-tls.map.fastly.net [151.101.1.67]
over a maximum of 20 hops:
  0  FIREFLY [192.168.12.155]
  1  192.168.12.122
  2  68-118-50-233.static.stls.mo.charter.com [68.118.50.233]
  3  10.104.121.129
  4  dtr01astror-tge-0-1-0-1.astr.or.charter.com [96.34.104.238]
  5  acr02vancwa-gbe-9-1.vanc.wa.charter.com [96.34.106.29]
  6  96-34-111-216.static.unas.or.charter.com [96.34.111.216]
  7  96-34-108-162.static.unas.or.charter.com [96.34.108.162]
  8  96-34-108-175.static.unas.or.charter.com [96.34.108.175]
  9  96-34-108-160.static.unas.or.charter.com [96.34.108.160]
 10  bbr01snjsca-bue-1.snjs.ca.charter.com [96.34.2.252]
 11  bbr02snloca-bue-6.snlo.ca.charter.com [96.34.0.1]
 12  bbr01snloca-bue-4.snlo.ca.charter.com [96.34.0.28]
 13  bbr01mtpkca-bue-5.mtpk.ca.charter.com [96.34.0.26]
 14  bbr01ashbva-tge-0-1-0-1.ashb.va.charter.com [96.34.3.139]
 15  199.27.73.26
```

```
 16  151.101.1.67

Computing statistics for 400 seconds...
            Source to Here   This Node/Link
Hop  RTT    Lost/Sent = Pct  Lost/Sent = Pct  Address
 0                                  FIREFLY [192.168.12.155]
                          0/ 100 =  0%   |
 1    0ms     0/ 100 =  0%     0/ 100 =  0%  192.168.12.122
                          0/ 100 =  0%   |
 2    1ms     0/ 100 =  0%     0/ 100 =  0%  68-118-50-233.static.stls.mo.charter.com [68.118.50.233]
                          0/ 100 =  0%   |
 3  ---     100/ 100 =100%   100/ 100 =100%  10.104.121.129
                          0/ 100 =  0%   |
 4  ---     100/ 100 =100%   100/ 100 =100%  dtr01astror-tge-0-1-0-1.astr.or.charter.com [96.34.104.238]
                          0/ 100 =  0%   |
 5  ---     100/ 100 =100%   100/ 100 =100%  acr02vancwa-gbe-9-1.vanc.wa.charter.com [96.34.106.29]
                          0/ 100 =  0%   |
 6  ---     100/ 100 =100%   100/ 100 =100%  96-34-111-216.static.unas.or.charter.com [96.34.111.216]
                          0/ 100 =  0%   |
 7  ---     100/ 100 =100%   100/ 100 =100%  96-34-108-162.static.unas.or.charter.com [96.34.108.162]
                          0/ 100 =  0%   |
 8  ---     100/ 100 =100%   100/ 100 =100%  96-34-108-175.static.unas.or.charter.com [96.34.108.175]
                          0/ 100 =  0%   |
 9  ---     100/ 100 =100%   100/ 100 =100%  96-34-108-160.static.unas.or.charter.com [96.34.108.160]
                          0/ 100 =  0%   |
 10  ---     100/ 100 =100%   100/ 100 =100%  bbr01snjsca-bue-1.snjs.ca.charter.com [96.34.2.252]
                          0/ 100 =  0%   |
 11  ---     100/ 100 =100%   100/ 100 =100%  bbr02snloca-bue-6.snlo.ca.charter.com [96.34.0.1]
                          0/ 100 =  0%   |
 12  ---     100/ 100 =100%   100/ 100 =100%  bbr01snloca-bue-4.snlo.ca.charter.com [96.34.0.28]
                          0/ 100 =  0%   |
 13  ---     100/ 100 =100%   100/ 100 =100%  bbr01mtpkca-bue-5.mtpk.ca.charter.com [96.34.0.26]
                          0/ 100 =  0%   |
 14  ---     100/ 100 =100%   100/ 100 =100%  bbr01ashbva-tge-0-1-0-1.ashb.va.charter.com
[96.34.3.139]
                          0/ 100 =  0%   |
 15  ---     100/ 100 =100%   100/ 100 =100%  199.27.73.26
                          0/ 100 =  0%   |
 16   41ms     0/ 100 =  0%     0/ 100 =  0%  151.101.1.67

Trace complete.

C:\WINDOWS\system32>
```