

# Cisco Frequently Asked Questions (FAQ)

## 1. What does ``cisco'' stand for?

A) At one point in time, the first letter in cisco Systems was a lowercase ``c''. At present, various factions within the company have adopted a capital ``C'', while fierce traditionalists (as well as some others) continue to use the lowercase variant, as does the cisco Systems logo. This FAQ has chosen to use the lowercase variant throughout.

cisco is not C.I.S.C.O. but is short for San Francisco, so the story goes. Back in the early days when the founders Len Bosack and Sandy Lerner and appropriate legal entities were trying to come up with a name they did many searches for non similar names, and always came up with a name which was denied. Eventually someone suggested ``cisco'' and the name wasn't taken (although SYSCO may be confusingly similar sounding). There was an East Coast company which later was using the ``CISCO'' name (I think they sold in the IBM marketplace) they ended up having to not use the CISCO abbreviation. Today many people spell cisco with a capital ``C'', citing problems in getting the lowercase ``c'' right in publications, etc. This lead to at least one amusing article headlined ``Cisco grows up''. This winter we will celebrate our 10th year.

## 2. How do I save the configuration of a cisco?

A) If you have a tftp server available, you can create a file on the server for your router to write to, and then use the write network command. From a typical unix system:

```
mytftpserver$ touch /var/spool/tftpboot/myconfig
mytftpserver$ chmod a+w /var/spool/tftpboot/myconfig
```

```
myrouter#write net
Remote host [10.7.0.63]? 10.7.0.2
Name of configuration file to write [myrouter-config]? myconfig
Write file foobar on host 10.7.0.2? [confirm] y
```

Additionally, there's a Macintosh TFTP server available:

```
ftp://nic.switch.ch/software/mac/peterlewis/
```

Additionally, you can also use expect, available from:

```
ftp://ftp.uu.net/languages/tcl/expect/expect.tar.gz
ftp://ftp.cme.nist.gov/expect/
```

or, in shar form from ftpeng.cisco.com.

Expect allows you to write a script which telnets to the router and performs a ``write terminal'' command, or any other arbitrary set of

command(s), using a structured scripting language (Tcl).

### 3. How can I get my cisco to talk to a third party router over a serial link?

A) You need to tell your cisco to use the same link-level protocol as the other router; by default, ciscos use a rather bare variant of HDLC (High-level Data Link Control) all link-level protocols use at some level/layer or another. To make your cisco operate with most other routers, you need to change the encapsulation from HDLC to PPP on the relevant interfaces. For instance:

```
sewer-cgs#conf t
```

```
Enter configuration commands, one per line.
Edit with DELETE, CTRL/W, and CTRL/U; end with CTRL/Z
interface serial 1
encapsulation ppp
^Z
```

```
sewer-cgs#sh int s 1
```

```
Serial 1 is administratively down, line protocol is down
Hardware is MCI Serial
MTU 1500 bytes, BW 1544 Kbit, DLY 20000 usec, rely 255/255, load 1/255
Encapsulation PPP, loopback not set, keepalive set (10 sec)
^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^^
[...]
```

If you're still having trouble, you might wish to turn on serial interface debugging:

```
sewer-cgs#ter mon
sewer-cgs#debug serial-interface
```

### 4. How can I get my cisco to talk to a 3rd-party router over Frame Relay?

A) You should tell your cisco to use ``encapsulation frame-relay ietf'' (instead of ``encapsulation frame-relay'') on your serial interface that's running frame relay if your frame relay network contains a diverse set of manufacturers' routers. The keyword ``ietf'' specifies that your cisco will use [RFC1294](#)-compliant encapsulation, rather than the default, [RFC1490](#)-compliant encapsulation (other products, notably Novell MPR 2.11, use a practice sanctioned by 1294 but deemed verboten by 1490, namely padding of the nlpid). If only a few routers in your frame relay cloud require this, then you can use the default encapsulation on everything and specify the exceptions with the frame-relay map command:

```
frame-relay map ip 10.1.2.3 56 broadcast ietf
^^^^
```

(ietf stands for Internet Engineering Task Force, the body which evaluates Standards-track RFCs; this keyword is a misnomer as both

RFC1294 and [RFC1490](#) are ietf-approved, however 1490 is most recent and is a Draft Standard (DS), whereas 1294 is a Proposed Standard (one step beneath a DS), and is effectively obsolete).

## 5. How can I use debugging?

A) The ``terminal monitor'' command directs your cisco to send debugging output to the current session. It's necessary to turn this on each time you telnet to your router to view debugging information. After that, you must specify the specific types of debugging you wish to turn on; please note that these stay on or off until changed, or until the router reboots, so remember to turn them off when you're done.

Debugging messages are also logged to a host if you have trap logging enabled on your cisco. You can check this like so:

```
sl-panix-1>sh logging
Syslog logging: enabled (0 messages dropped, 0 flushes, 0 overruns)
Console logging: level debugging, 66 messages logged
Monitor logging: level debugging, 0 messages logged
Trap logging: level debugging, 69 message lines logged
Logging to 198.7.0.2, 69 message lines logged
sl-panix-1>
```

If you have syslog going to a host somewhere and you then set about a nice long debug session from a term your box is doing double work and sending every debug message to your syslog server. Additionally, if you turn on something that provides copious debugging output, be careful that you don't overflow your disk (``debug ip-rip'' is notorious for this).

One solution to this is to only log severity ``info'' and higher:

```
sl-panix-1#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
logging trap info
```

The other solution is to just be careful and remember to turn off debugging. This is easy enough with:

```
sl-panix-1#undebug all
```

If you have a heavily loaded box, you should be aware that debugging can load your router. The console has a higher priority than a vty so don't debug from the console; instead, disable console logging:

```
cix-west.cix.net#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
no logging console
```

Then always debug from a vty. If the box is busy and you are a little too vigorous with debugging and the box is starting to sink, quickly run, don't walk to your console and kill the session on the vty. If you are on the console your debugging has top priority and then the

only way out is the power switch. This of course makes remote debugging a real sweaty palms adventure especially on a crowded box. Caveat debugger!

Also, if you for some reason forget what the available debug commands are and don't have a manual handy, remember that's what on-line help is for. Under pre 9.21 versions, ``debug ?'' lists all commands. Under 9.21 and above, that gives you general categories, and you can check for more specific options by specifying the category: ``debug ip ?''.

As a warning, the ``logging buffered'' feature causes all debug streams to be redirected to an in-memory buffer, so be careful using that.

Lastly, if you're not sure what debugging criteria you need, you can try ``debug all''. BE CAREFUL! It is way useful, but only in a very controlled environment, where you can turn off absolutely everything you're not interested in. Saves a lot of thinking. Turning it on on a busy box can quickly cause meltdown.

## **6. How can I use NTP (Network Time Protocol) with my cisco?**

A) What level of software is required for NTP support in  
>a cisco router?

9.21 or above.

>Which cisco routers support NTP?

It is a software feature exclusively. Anything that supports 9.21 or 10 will run NTP (when running that s/w).

>How do I set it up?

The basic hook is:

```
ntp server <host> [version n]
or
ntp peer <host> [version n]
```

depending on whether you want a client/server or peer relationship. There's a bunch of other stuff available for MD5 authentication, broadcast, access control, etc. You can also use the context-sensitive help feature to puzzle it out; try ``ntp ?'' in config mode.

You'll also want to play with the SHOW NTP \* router commands. Here are two examples.

EXAMPLE 1:

```
router# show ntp assoc
```

```
address          ref clock      st  when  poll reach  delay  offset
disp
+~128.9.2.129    .WWVB.        1   109   512  377   97.8  -2.69
26.7
```

```
*~132.249.16.1 .GOES. 1 309 512 357 55.4 -1.34
27.5
* master (syncd), # master (unsyncd), + selected, - candidate, ~
configured
```

EXAMPLE 2:

```
router#show ntp stat
Clock is synchronized, stratum 2, reference is 132.249.16.1
nominal freq is 250.0000 Hz, actual freq is 249.9981 Hz, precision is
2**19 reference time is B1A8852D.B69201EE (12:36:13.713 PDT Tue Jun 14
1994) clock offset is -1.34 msec, root delay is 55.40 msec
root dispersion is 41.29 msec, peer dispersion is 28.96 msec
```

For particular cisco NTP questions, feel free to ask in  
comp.dcom.sys.cisco.

For broader NTP info, see <ftp://louie.udel.edu:pub/ntp/>. The file  
clock.txt in that directory has info about various public NTP servers.  
There is also information on radio time receivers that can be  
connected to an NTP server (this is handy on private networks, if you  
have an entire campus to get chiming, or if you become a hard core  
chimer).

The ``ntp clock-period'' command is added automagically to jump-start  
the NTP frequency compensation when the box is rebooted. This is  
essentially a representation of the frequency of the crystal used as  
the local timebase, and may take several days to calculate otherwise.  
(Do a ``write mem'' after a week or so to save a good value.)

Caveat of obsolescence: Note that the CS-500 will not be able to  
achieve quite the same level of accuracy as other platforms, since its  
hardware clock resolution is roughly 242Hz instead of the 1MHz  
available on other platforms. In practice this shouldn't matter for  
anyone other than true time geeks.

## 7. Sample cisco NTP Configurations?

A) You will need to substitute your own NTP peers, timezones, and GMT  
offsets into the examples below, of course. Example 1 is in US Central  
Time Zone, while example 3 is in US Pacific Time Zone. Both account  
for normal US Daylight Savings Time practices.

EXAMPLE 1 (Charley Kline):

```
...
clock timezone CST -6
clock summer-time CDT recurring
ntp source eth 0
ntp peer <host1>
ntp peer <host2>
ntp peer <host3>
...
```

EXAMPLE 2 (Tony Li):

```
...
```

```
ntp source Ethernet0/0
ntp update-calendar
ntp peer <host1>
ntp peer <host2> prefer
...
```

EXAMPLE 3 (Dave Katz):

```
...
service timestamps debug datetime localtime
service timestamps log datetime localtime
clock timezone PST -8
clock summer-time PDT recurring
interface Ethernet0
ip address <mumble>
ntp broadcast
ntp clock-period 17180319
ntp source Ethernet0
ntp server <host1>
ntp server <host2>
ntp server <host3>
```

COMMENTS ON EXAMPLE 3:

The config file is commented with date and time (and user id, if TACACS is enabled) when the system thinks the clock is accurate. I've enabled timestamping of debug and syslog messages. I send NTP broadcast packets out onto the local ethernet. I'm in Pacific Standard Time, with U.S. standard daylight saving time rules. I use the IP address of the ethernet as the source for all NTP packets.

## 8. How do I avoid the annoying DNS lookup if I have misspelled a command?

A) By default, all lines are configured to automatically try a telnet connection if the first word in a input line is not recognized as a valid command. You can disable this by setting ``transport preferred none'' on every line (con, aux and vty). For instance:

```
sl-panix-1#conf t
Enter configuration commands, one per line. End with CNTL/Z.
line vty 0 10
transport preferred none
```

You can see the number of vty's currently configured with ``show lines''

Also, you can suspend connect attempts with ^^ followed by ``x'', ie shift-cntrl-6 x.[It has been suggested that ``no ip ipname-lookup'' to turn off IEN116 helps. I think this is the default -jhawk ]

## 9. Tracing bad routing information?

A) Here you could work with a default administrative distance. Administrative distance is the basis upon which the cisco prefers routing information of one protocol over another. In this example:

```

router rip
network 192.125.254.0
distance 255
distance 120 192.125.254.17      ! list all valid RIP suppliers
[...]
```

the value 255 has the implicit meaning of not putting this information into the routing table. Therefore, setting an administrative distance of 255 means that all RIP suppliers are by default accepted but their information is not put into the routing table. The administrative distance for the router 192.125.244.17 has been reset to the default (for RIP) of 120, causing its routes to be accepted into the routing table.

Then you can look them up with ``show ip protocols'' and restore the original administrative distance for the ones you want to fill in the routing table.

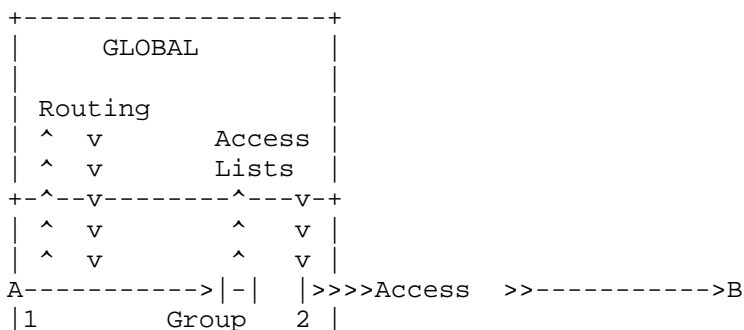
The same results can be achieved with an ip access-list, but with that, ``show ip protocols'' will only show the valid ones. But often it is more useful to see which systems were generating routing information at all.

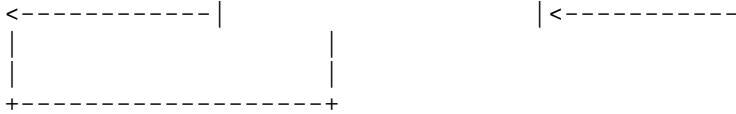
This trick works for other routing protocols as well, but please select the proper administrative distance (rather than 120) for the protocol you're using.

## 10. How to use access lists?

A) In general, Basic access lists are executed as filters on outgoing interfaces. Newer releases of the cisco code, such as 9.21 and 10, do have increased ability to filter on incoming ports. Certain special cases, such as broadcasts and bridged traffic, can be filtered on incoming interfaces in earlier releases. There are also special cases involving console access.

Rules, written as ACCESS-LIST statements, are global for the entire cisco box; they are activated on individual outgoing interfaces by ACCESS-GROUP subcommands of the INTERFACE major command. Filters are applied after traffic has entered on an incoming interface and gone through a routing process; traffic that originates in a router (e.g., telnets from the console port) is not subject to filtering.





Some types of ``filter,' ' using ``filter' ' as a broader class than ACCESS-LIST, can operate on incoming traffic. For example, the INPUT-SAP-FILTER used for Novell networks is applied to Service Advertisement Packets (SAP) seen at incoming interfaces. In general, incoming filtering can only be done for ``system' ' rather than user traffic.

Rules of thumb in defining access lists.

First, define what you want to do and in which directions. An informal drawing is a good first step. As opposed to the usual connectivity drawings among routers, it's often convenient to draw unidirectional links between routers. Second, informally write out your filtering rules. In general, it is best to go from most specific to least specific. Modify the order of writing things to minimize the number of rules needed. Third, determine which rules need to be on which routers. Explicitly consider the direction of flow, and the possible existence of additional paths that could inadvertently bypass a filter.

Can a cisco router be a ``true' ' firewall?

This depends on the definition of firewall. Some writers (e.g., Gene Spafford in Practical UNIX Security) define a firewall as a host on which an ``inside' ' and/or an ``outside' ' application process run, with application-level code linking the two. For example, a firewall might provide FTP access to the outside world, but it would not also provide direct FTP service to the inside world. To place a file on the FTP external server, a designated user would explicitly log onto the FTP server, transfer a file to the server, and log off. The firewall prevents direct FTP connectivity between the inside and outside networks; only indirect, application-level connectivity is allowed. Firewalls of this sort are complemented by chokes, which filter on network addresses and/or port numbers. Cisco routers cannot do application-level control with access control lists. Other authors do not distinguish between chokes and filters. Using the loose definition that a firewall is anything that selectively blocks access from the inside to the outside, routers can be firewalls.

#### IP Specific

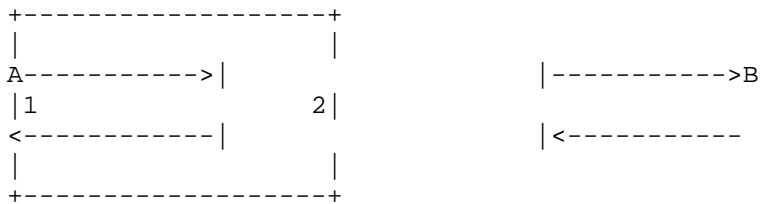
-----

Can the ``operand' ' field be used with a protocol keyword of IP to filter on protocol ID?

No. Operand filtering only works for TCP and UDP port numbers.

How can I prevent traffic for a certain Internet application to flow in one direction but not the other?

Remember that Internet applications flow from client port to server port. Denying traffic from port 23, for example, blocks flow from the client to the server.



If we deny traffic to Port 23 of address B by placing a filter at interface 2, we have blocked A's ability to telnet to B, but not B's ability to telnet to A. A second filter at interface A would be needed to block telnet in both directions.

Assume that we only have the filter at interface 2. Telnets to A from B will not be affected because the filter at 2 does not check incoming traffic.

-----

With the arrival of in-bound access lists in 9.21, it should be noted that both inbound and access lists are about equally efficient, in case any of you were wondering.

It's worth remembering that there are some kinds of problems that packet-filtering firewalls are not best suited for. There's reasonably good information in:

"Network (in)security through packet filtering"  
[ftp://ftp.greatcircle.com/pub/firewalls/pkt\\_filtering.ps.Z](ftp://ftp.greatcircle.com/pub/firewalls/pkt_filtering.ps.Z)

## 11. What really happens when a cisco router boots, from boot start to live interfaces?

A) First it boots the ROM os version. It reads the config. Now, it realizes that you want to netboot. It loads the netbooted copy in on top of itself. It then re-initializes the box and re-reads the config. Manly, yes, but we like it too....

[[ Ummm... in particular it loads the netbooted copy in as WELL as itself, decompresses it, if necessary, and THEN loads on top of itself. Note that this is important because it tells you what the memory requirements are for netbooting: RAM for ROM image (if it's a run from RAM image), plus dynamic data structures, plus RAM for netbooted image. ]]

The four ways to boot and what happens (sort of):

I (from bootstrap mode)

The ROM monitor is running. The I command causes the ROM monitor to walk all of the hardware in the bus and reset it with a brute force hammer. If the bits in the config register say to auto-boot, then goto B

B (from bootstrap mode)

Load the OS from ROM. If a name is given, tell that image to start silently and then load a new image. If the boot system command is given, then start silently and load a new image.

## 12. How are packets switched?

A) There are 3 basic types of switching (in order of increasing performance).

process switching  
fast switching  
autonomous switching

Process and fast switching support inbound and outbound, simple and extended, access lists. Of course, for fast switching, such lists only restrict traffic on the particular fast-switched interface.

Autonomous switching is done in the switch processor, a microcoded device that is capable of switching IP, IPX, and bridging packets in the 100kpps range. This is known as the "SP" card on the 7000 and the CBUS controller on the AGS+. Encapsulation support is rather limited (Ethernet, HDLC, HSSI...).

The cisco 7000 also supports:

silicon switching

Silicon switching is done in the silicon switching engine (creative, eh? ;-).

The silicon switch processor (SSP) is the board which combines both the switch processor and a silicon switching engine.

The SSP supports simple and extended outbound access lists in 10.3 and later.

The SSP supports simple and extended inbound access lists in 11.1 and later.

The cisco 75xx series supports:

"optimal" switching (cruddy name, eh?)  
"flow" switching  
"distributed" switching

\* "optimal" switching (cruddy name, eh?)

The 7500 platform does not have a separate SP or SSP card, rather the RISC processor on the "integrated route/switch processor card (IRSP)" handles switching directly, similar to the 4000 series routers. There are several hardware and software enhancements made though to increase the throughput to a level that is several times above what you would normally get from "fast" switching. Everything that "fast" switching supports is supported "optimal" switching.

\* "flow" switching

Basicly the "optimal" switching method, however things have been front-ended with an additional small "flow" cache. This flow cache contains information about source/destination addresses & ports which allow the router to make more informed queueing decisions and process access lists faster. This is a win in routers that would tend to carry a reasonably small number of flows at any one time, such as what you would expect in a corporate network or in a smaller internet service provider network. It's unclear if there are any advantages in a large internet backbone.

\* "distributed" switching

cisco has announced a new type of interface-processor card, called a "VIP" available in the 7500 platform that is intelligent enough to switch packets with no intervention on the part of the IRSP card. This once again separates switching from routing, as in the earlier CBUS/SP/SSP design.

The first packet of every session or connection is always Process Switched. The route table is consulted (this resides in DRAM on the CPU) and the "result" is cached in the system memory cache. If the protocol can only be process switched, then it will continue this way and interrupt the CPU for a route table lookup each time. [comment: Process Switching is brutally slow compared to other switching methods. Some features (usually new features do this for the first few software releases) force every packet to be process switched. If you can't avoid process-switching every packet, at least get a router with a fast CPU, such as the 75xx, 4500, and 4700. The 4700 is currently the fastest at process-switching packets, with the 4500 and 75xx tied for second. The 75xx can optimum-switch, however, so it's a lot faster than either of the 4x00s, if you can use it).

The second and subsequent packets of each session are capable of being Fast Switched (more session types are becoming fast-switchable), and will consult only the route-cache. This still involves a memory lookup on the board, but the packet can be transferred from the source card directly to the destination card without requiring full storage on the CSC [the CSC refers to the CPU card, basically].

There are some undocumented commands that are useful for obtaining per-interface statistics on what sort of switching was performed.

For instance:

```
frobozz-magic-robot>sh int atm4/0 switch
ATM4/0
Throttle count:          0
Protocol      Path      Pkts In   Chars In   Pkts Out   Chars Out
IP    Process  104851   7669968   116378    11180988
Cache misses      35826
Fast              0         0         0         0         0
Auton/SSE        0         0         0         0         0
frobozz-magic-robot>sh int atm4/0 stat
ATM4/0
```

Switching path	Pkts In	Chars In	Pkts Out	Chars Out
Processor	105024	7679155	116422	11184108
Route cache/FIB	0	0	0	0
Distributed cache	0	0	0	0
Total	105024	7679155	116422	11184108.

### 13. How should I restrict access to my router?

A) Many admins are concerned about unauthorized access to their routers from malicious people on the Internet; one way to prevent this is to restrict access to your router on the basis of IP address.

Many people do this, however it should be noted that a significant number of network service providers allow unrestricted access to their routers

to allow others to debug, examine routes, etc. If you're comfortable doing this, so much the better, and we thank you!

If you wish to restrict access to your router, select a free IP access list (numbered from 1-100) -- enter ``sh access-list'' to see those numbers in use.

```
yourrouter#sh access-list
Standard IP access list 5
permit 192.94.207.0, wildcard bits 0.0.0.255
```

Next, enter the IP addresses you wish to allow access to your router from; remember that access lists contain an implicit "deny everything" at the end, so there is no need to include that. In this case, 30 is free:

```
yourrouter#conf t
Enter configuration commands, one per line. End with CNTL/Z.
yourrouter(config)#access-list 30 permit 172.30.0.0 0.0.255.255
yourrouter(config)#^Z
```

(This permits all IP addresses in the network 172.30.0.0, i.e. 172.30.\*.\*). Enter multiple lines for multiple addresses; be sure that you don't restrict the address you may be telnetting to the router from.

Next, examine the output of ``sh line'' for all the vty's (Virtual ttys) that you wish to apply the access list to. In this example, I want lines 2 through 12:

```
yourrouter#sh line
Tty Typ Tx/Rx A Modem Roty AccO AccI Uses Noise Overruns
0 CTY - - - - - 0 0 0/0
1 AUX 9600/9600 - - - - - 1 3287605 1/0
* 2 VTY 9600/9600 - - - - - 7 55 0 0/0
3 VTY 9600/9600 - - - - - 7 4 0 0/0
4 VTY 9600/9600 - - - - - 7 0 0 0/0
5 VTY 9600/9600 - - - - - 7 0 0 0/0
6 VTY 9600/9600 - - - - - 7 0 0 0/0
7 VTY 9600/9600 - - - - - 7 0 0 0/0
```

```

8 VTY  9600/9600  -   -   -   -   7   0   0   0/0
9 VTY  9600/9600  -   -   -   -   7   0   0   0/0
10 VTY 9600/9600  -   -   -   -   7   0   0   0/0
11 VTY 9600/9600  -   -   -   -   -   0   0   0/0
12 VTY 9600/9600  -   -   -   -   -   0   0   0/0

```

Apply the access list to the relevant lines:

```

yourrouter#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
yourrouter(config)#line 2 12
yourrouter(config-line)# access-class 30 in
yourrouter(config-line)# ^Z

```

(This apply access list 30 to lines 2 through 12. It's important to restrict access to the aux port (line 1) if you have a device (such as a CSU/DSU) plugged into [it.a](#))

Be sure to save your configuration with ``write mem''.

Please note that access lists for incoming telnet connections do NOT cause your router to perform significant CPU work, unlike access lists on interfaces.

## 14. What can I do about source routing?

A) What *is* source routing?

Source routing is an IP option which allows the originator of a packet to specify what path that packet will take, and what path return packets sent back to the originator will take. Source routing is useful when the default route that a connection will take fails or is suboptimal for some reason, or for network diagnostic purposes. For more information on source routing, see [RFC791](#).

Unfortunately, source routing is often abused by malicious users on the Internet (and elsewhere), and used to make a machine (A), think it is talking to a different machine (B), when it is really talking to a third machine (C). This means that C has control over B's ip address for some purposes.

The proper way to fix this is to configure machine A to ignore source-routed packets where appropriate. This can be done for most unix variants by installing a package such as Wietse Venema, [<wietse@wzv.win.tue.nl>](mailto:wietse@wzv.win.tue.nl), 's tcp\_wrapper:

```
ftp://cert.org:pub/tools/
```

For some operating systems, a kernel patch is required to make this work correctly (notably SunOS 4.1.3). Also, there is an unofficial kernel patch available for SunOS 4.1.3 which turns all source routing off; I'm not sure where this is available, but I believe it was posted to the firewalls list by Brad Powell sometime in mid-1994.

If disabling source routing on all your clients is not possible, a

last resort is to disable it at your router. This will make you unable to use ``traceroute -g'' or ``telnet @hostname1:hostname2'', both of which use LSRR (Loose Source Record Route, 2 IP options, the first of which is a type of source routing), but may be necessary for some. If so, you can do this with

```
foo-e-0#conf t
Enter configuration commands, one per line.  End with CNTL/Z.
foo-e-0(config)#no ip source-route
foo-e-0(config)#^Z
```

It is somewhat unfortunate that you cannot be selective about this; it disables all forwarding of source-routed packets through the router, for all interfaces, as well as source-routed packets to the router (the last is unfortunate for the purposes of ``traceroute -g'').

## 15. Is there a block of private IP addresses I can use?

A) Yes there is, however whether you wish to do so is an issue of some debate.

You could consult:

1627 Network 10 Considered Harmful (Some Practices Shouldn't be Codified). E. Lear, E. Fair, D. Crocker & T. Kessler. June 1994. (Format: TXT=18823 bytes)

1918 Address Allocation for Private Internets. Y. Rekhter, B. Moskowitz, D. Karrenberg, G. J. de Groot & E. Lear. February 1996. (Format: TXT=22270 bytes) (Obsoletes [RFC1627](#), [RFC1597](#)) (Also BCP0005)

In any event, [RFC 1918](#) documents the allocation of the following addresses for use by ``private internets'':

10.0.0.0	-	10.255.255.255
172.16.0.0	-	172.31.255.255
192.168.0.0	-	192.168.255.255

Most importantly, it is vital that nothing using these addresses should ever connect to the global Internet, or have plans to do so. Please read the above RFCs before considering implementing such a policy.

As an additional note, some Internet providers provide network-management services, statistics gathering, etc. It is unlikely (if at all possible) that they would be willing to perform those services if you choose to utilize private address space.

With the increasing popularity and reliability of address translation gateways, this practice is becoming more widely accepted. Cisco has acquired Network Translation, who manufacture such a product. It is now available as the Cisco Private Internet Exchange. With it, you can use any addressing you want on your private internet, and the gateway will insure that the invalid addresses are converted before making out onto the global Internet. It also makes a good firewall. Information on this product is available at

## 16. Where can I get cisco documentation?

A) Cisco no longer distributes printed documentation with their routers; instead, they distribute a CDROM.

Paper documentation may be purchased, however if you purchase a support contract, documentation is free.

Cisco documentation is also available on the web -- if you have a fast Internet connecton this may be more useful than the CD. Try:

## 17. What IP routing protocol should I use?

A) This is a really complicated question, and a full answer is beyond the scope of this document. Here are the beginnings of an answer.

Note that Hello is no longer shipped with cisco routers, and that EGP has been declared Historical (and thus obsolete) by the IETF. Don't use them.

Protocol	RIP	HELLO	IGRP	OSPF	EIGRP	IS-IS	EGP	
BGP4								
-----								
Type	IGP	IGP	IGP	IGP	IGP	IGP	EGP	EGP
Algorithm	DV	DV	DV	SPF	DUAL	SPF	DV	PV
Metrics	Hopcnt	Delay	Speed	Arb.	Speed	Arb.	Policy	
Policy								
Convergence	Slow	Unstb	Mdt	Fast	Fast	Fast	Slow	
Fast								
Standard?	IETF	No	No	IETF	No	ISO	Hist.	
IETF								
Complexity	Simple	Simple	Simple	Complx	Complx	Complx	Simple	
Complx								
Multipath?	Yes	Yes	Yes	Yes	Yes	Yes	Yes	[*]
Var-netmask?	No	No	No	Yes	Yes	Yes	No	YES

### Notes

-----

IGP = interior gateway protocol, used to build routing tables within an AS.

EGP = exterior gateway protocol, used to communicate reachability information between AS's.

### Algorithms

-----

DUAL = DV with diffusing update algorithm (Garcia-Luna-Aceves et al)

DV = Distance Vector (Bellman-Ford)

PV = "Path Vector"

SPF = Shortest-path-first (Dijkstra)

#### Metrics

-----

A metric is how the protocol measures the network to determine the "best" path.

"Speed" refers typically to link speed, not available bandwidth.  
"Arb." indicates that the metrics are arbitrary and configurable.

HELLO tried to use available bandwidth by monitoring round-trip delay, but was not generally successful at this.

Metrics are not directly exchangable when redistributing routing information from one protocol to another. IGRP and EIGRP use compatible and automatically convertable metrics.

#### Convergence

-----

Qualitatively, convergence measures how fast routers using this protocol will adapt to changes in the topology of the network.

"Unstb" indicates a protocol which in general never decided on a stable configuration but continually oscillated between alternatives.

#### Complexity

-----

An observation of how complex the protocol is to implement.

#### Multipath

-----

Multipath indicates whether the protocol support and transport multiple equal- or different- cost pathways across between endpoints?

[\*] indicates that BGP4 supports multipath for IBGP (Internal BGP, a full mesh of all border routers within an AS), but not for EBGP (External BGP).

#### Variable netmask (Var-netmask)

-----

Indicates whether the protocol allows for and transports different masks for the subnets of a routed network.

### **18. How much memory is necessary to telnet to a cisco router?**

A) In order to login to a cisco router, it needs to have at least 64k of contiguous free memory.

### **19. When are static routes redistributed?**

A) In the simple case, any static route \*in the routing table\* is redistributed if the ``redistribute static'' command is used, and some

filter (set with either ``route-map'' or ``distribute-list out'') doesn't filter it out.

Whether the static route gets into routing table depends on:  
Whether the next hop address is reachable (if you use static route pointing to a next hop) OR Whether the interface is up (if you use static route pointing to an interface).

If one of these is true, an attempt is made to add the route to the routing table; whether that succeeds depends on the administrative distance of the route -- a lower administrative distance (the route is "closer") than a preexisting route will cause the preexisting route to be overwritten.

## **20. When is the next hop of a route considered ``reachable''?**

A) When a static route is added, or during an important event (eg: interface up/down transition), the next hop for a route is looked up from the routing table (i.e. recursive routing).

As a consequence, if a route which is depended upon for evaluation of the next hop of a static route goes away, a mechanism is required to remove that (now-invalid) static route.

Scanning all static routes each time the routing table changes is too expensive, so instead, a period timer is used. One a minute, static routes are added and removed from the routing table based on the routes they depend upon.

It should be noted that a particular static route will be reevaluated when its interface transitions up or down.

## **21. How do name and phone number of ``dialer map'' interfere?**

A) We use the telephone number first actually. If the caller id matches the telephone number to call, then you don't need the 'name' parameter with a phone number.

I realized that the above is ambiguous, so let's do this. You have:  
dialer map ip x.x.x.x name <param1> <phone-num> <param1> is used for incoming authentication. It can be either the hostname, for PAP and CHAP, or it can be a number as returned by caller id. If this is not there, and it is an incoming call, and there is caller id, we will compare against <phone-num> to see if that matches.  
Not sure I've been clear here.

## **22. What is VLSM?**

A) A Variable Length Subnet Mask (VLSM) is a means of allocating IP addressing resources to subnets according to their individual need rather than some general network-wide rule. Of the IP routing protocols supported by Cisco, OSPF, Dual IS-IS, BGP-4, and EIGRP support "classless" or VLSM routes.

Historically, EGP depended on the IP address class definitions, and

actually exchanged network numbers (8, 16, or 24 bit fields) rather than IP addresses (32 bit numbers); RIP and IGRP exchanged network and subnet numbers in 32 bit fields, the distinction between network number, subnet number, and host number being a matter of convention and not exchanged in the routing protocols. More recent protocols (see VLSM) carry either a prefix length (number of contiguous bits in the address) or subnet mask with each address, indicating what portion of the 32 bit field is the address being routed on.

A simple example of a network using variable length subnet masks is found in Cisco engineering. There are several switches in the engineering buildings, configured with FDDI and Ethernet interfaces and numbered in order to support 62 hosts on each switched subnet; in actuality, perhaps 15-30 hosts (printers, workstations, disk servers) are physically attached to each. However, many engineers also have ISDN or Frame Relay links to home, and a small subnet there. These home offices typically have a router or two and an X terminal or workstation; they may have a PC or Macintosh as well. As such, they are usually configured to support 6 hosts, and a few are configured for 14. The point to point links are generally unnumbered.

Using "one size fits all" addressing schemes, such as are found in RIP or IGRP, the home offices would have to be configured to support 62 hosts each; using numbers on the point to point links would further compound the address bloat.

One configures the router for Variable Length Subnet Masking by configuring the router to use a protocol (such as OSPF or EIGRP) that supports this, and configuring the subnet masks of the various interfaces in the 'ip address' interface sub-command. To use supernets, one must further configure the use of 'ip classless' routes.

### **23. What are some methods for conserving IP addresses for serial lines?**

A) VLSM and unnumbered point to point interfaces are the obvious ways.

The 'ip unnumbered' subcommand indicates another interface or sub-interface whose address is used as the IP source address on messages that the router originates on the unnumbered interface, such as telnet or routing messages. By doing this, the router is reachable for management purposes (via the address of the one numbered interface) but consumes no IP addresses at all for its unnumbered links.

When a serial ip interface connects several sites, as an SMDS link might, then the use of an appropriate subnet mask (and a routing protocol that can make good use of the information) will minimize address consumption.

### **24. Why do some ip addresses get rejected?**

A) How come my cisco router doesn't accept an address like:  
"ip address 192.111.107.1 255.255.255.240"  
or "ip address 171.69.0.1 255.255.0.0"

When "subnetting" of IP networks was first sanctioned by the IETF, the first and last subnets (the all zeros subnet and all ones subnet) were reserved for rather obscure uses and because of the confusion that would be caused with routing protocols that don't carry net mask information. It was technically illegal to place hosts or routers on those two subnets.

Several hosts and most other vendor's router products have problems operating with the reserved subnets, so their use is discouraged. However, in 1995, the IETF removed the restrictions on the use of these reserved subnets as part of the classless routing effort.

If you would like to use the reserved subnets, simply add the line "ip subnet-zero" to your cisco configuration.

You might consider adding "ip subnet-zero" to all your configurations as a matter of course, to avoid being bitten by this in the future.