

# IOS Commands

## Line Editor

ctrl-A Beginning of Line	ctrl-B Back One Char	<esc> B Back One Word	ctrl-P Previous Command
ctrl-E End of Line	ctrl-F Forward One Char	<esc> F Forward One Word	ctrl-N Next Command
ctrl-K Delete Rest of Line	ctrl-Z Hang-up; Exit	ctrl-shift-6 (let go) [repeat] x	Pause a Telnet(s)

## Shell Conveniences

```
Device>?  
Device>e?  
<tab>  
Device#configure ?
```

Lists commands available in current mode  
Lists commands starting with 'e'  
Command completion, if unique.  
Lists options for the configure command

## Naming

```
Device>enable  
Device#configure terminal  
Device(config)#hostname D  
D(config)#banner { motd | login | exec } %blah%  
D(config)#exit  
D#exit
```

Switch to privileged executive mode—administrator # prompt  
Enter global configuration mode—(config) prompt  
From now on, when you see, a (config) prompt, assume above  
To enclose multiline banner, choose any character not within (%)  
“end” works also—jumps to priv exec, replacing multiple exits  
“Disable” works also; opposite of “enable.”

## Display & File System

```
D#show running-config [ line ]  
D#show run | [include|exclude] foo  
D#show run | begin foo  
D#show run | section Serial  
D#show run interface fa0/0  
D#show file systems  
D#{show flash | dir flash:}  
D#show terminal
```

Line adds line numbers  
Limits output to lines containing (or not) “foo” Default include  
Output from first occurrence of “foo” to end of file  
Case-sensitive limit to sections whose name contains “Serial”  
Just for interfaces  
Size, free space, etc for flash, nvram, etc.  
“Show flash” recursively shows directory contents

## Save Config

```
D#copy running-config startup-config  
D#copy startup-config tftp  
D#copy run usbflash0:[path/filename]
```

Saves config to non-volatile RAM to survive reboot  
Copies to a tftp server for rapid restore over net  
Copies running config to a memory stick in router's USB jack

```
D(config)#archive  
D(config-archive)#path ftp://ben@10.0.0.2/  
D(config-archive)#time-period 1440
```

Periodically copy running config to remote server

1440 minutes = every 24 hours. The filename gives timestamp.

## Restore Config

```
D#copy tftp running-config  
D#copy running-config startup-config  
D#copy usbflash0:[path/file] run
```

Doing this first avoids a reboot

Merges config from memory stick atop current running-config

## Erase & Reload

```
D#erase startup-config  
Switch#delete flash:vlan.dat  
D#reload  
D#setup
```

To Restore Defaults

“write erase” is equivalent; “erase nvram:” is new preferred.

Only needed on a switch, harmless error if vlan.dat missing

Or just cycle the power

Manually invoke the startup interview

## Boot Control

```
D(config)#config-register 0x2102  
rommon 1>confreg 0x2142  
D#copy ftp://user@address/file flash  
D#verify flash:<filename>  
D#boot system {flash filename | flash:filename | filename} 3 ways to name the same IOS .bin file  
D#boot system { tftp | ftp | rcp } <filename> 10.0.0.1
```

2102 default. Last digit 0=rommon

Used for password bypass (2142 = ignore startup-config)

Copy a new IOS image to flash

Check IOS img corruption—computed vs. embedded MD5 hash

3 ways to name the same IOS .bin file

```
D#show version
```

Tells which IOS is running and will boot next plus config-reg value

## IOS Licensing

```
R#license boot module c2900 technology-package securityk9 Add a “right-to-use” (eval) feature to a router  
R#license install <url>  
R#show license [ feature | udi ]
```

### For IOS Universal Images ≥ version 15.0

Add a “right-to-use” (eval) feature to a router

Installs a license key file into a router

Feature is a list of Y/N. UDI is unique identifier for equipment

## Security

```
D(config)#enable password cisco  
D(config)#enable secret class  
D(config)#enable algorithm-type {md5 | sha-256 | scrypt} secret class Option to choose crypto technique  
D(config)#service password-encryption Takes plaintext out of the running-config  
D(config)#username myName [0|5|15] {password|secret} myPass Number is privilege level (15 starts enabled)  
D(config)#security passwords min-length 5  
D(config)#login block-for 120 attempts 3 within 60 Blocks for 2 minutes after 3 fails within 1 minute  
D(config)#line console 0 Next two lines set console passwd  
D(config-line)#no password Removes existing Password; not necessary if replacing  
D(config-line)#password cisco123  
D(config-line)#login [ local ] Require the password at login; Local also requires username
```

Passwd to protect privileged exec mode

“Secret” password overrides the “password” passwd

Option to choose crypto technique

Takes plaintext out of the running-config

Number is privilege level (15 starts enabled)

Blocks for 2 minutes after 3 fails within 1 minute

Next two lines set console passwd

Removes existing Password; not necessary if replacing

Require the password at login; Local also requires username

## Convenience

```
D(config)#no ip domain-lookup  
D(config)#ip name-server <ip> [<ip>...]  
D#ip host <name> <ip>  
D(config)#[no] logging console  
D(config)#line { console 0 | vty 0 4 }  
D(config-line)#logging synchronous  
D(config-line)#exec-timeout 0 0  
D(config-line)#transport preferred none  
D(config-line)#history size 100  
D#terminal history size 100  
D#show history  
D#reload in 10  
D#reload cancel  
D(config)#do show ...
```

Don't send typos to DNS. Better to kill preferred transport, below

Like /etc/hosts

Whether to send logging messages to console.

If you interrupt my typing with logging, reprint what I typed so far

Don't hang up on me. 0 0 is minutes seconds.

Typos aren't telnets. Can leave domain-lookup on

Change own history size from within session

List of past commands (up arrow does one at a time)

Reboot from saved config in 10 minutes in case I lock self out

When don't need the above anymore

Avoids exiting back to privileged exec for one command

## Diagnostics

```
D#ping [ <ip> | <hostname> ] [source {fa0/0|<ip>}]  
D#traceroute [ <ip> | <hostname> ]  
D#show [ip] arp
```

Without parameters, interactive. Options like where ping FROM

Prints a line for each router “hop” on the path

## Telnet / SSH

D(config)#line vty 0 15	Now, password the vtys
D(config-line)#password cisco123	Needed for telnet to work if not using login local usernames
D(config-line)#login [local]	Local = username/password command used for authentication
D(config-line)#privilege level 15	Optional. Pointless if "login local" uses user privilege levels
D(config-line)#transport input {telnet ssh all none}+	Ordered list of preferences.
D#[telnet] { <ip>   <hostname> }	
D#terminal monitor	Allows a remote login to see syslog messages like console does
D#ctrl-^ x	(Shift-ctrl-6, let go, x) suspends a current session
D#show users   sessions	Users are incoming connections, sessions outgoing
D#{resume   disconnect} <session-number>	Resumes or drops a session (number from "show sessions")
D#exit   quit	Disconnects from the command line interface session

## SSH

D(config)#ip domain-name mydomain.com	Domain and host name (D) needed by "crypto key generate rsa"
D#(config)#crypto key zeroize rsa	Erase existing RSA keys (pointless if creating a new pair)
D#(config)#crypto key generate rsa	Needed if using ssh
D#(config)#ip ssh version 2	
D#ssh -l username 10.0.0.2	Outbound ssh connection
myPC\$ scp myFile myUser@myRouter:flash0:myFile	From a PC, copy an IOS image file to a router's flash. Requires "ip scp server enable"
D#show ssh	SSH connections
D#show ip ssh	SSH server info (version, timeout, retries, key)
D#show crypto key mypubkey rsa	Public (& Private ?) key

## Interface Basics

D(config-if)#speed { 10   100   1000   auto }	
D(config-if)#duplex { half   full   auto }	If set on one end, must set on other, otherwise auto fails to half
D(config-if)#description blah blah blah	
D(config)#int [ range ] fa0/0 [ - 4 ] [ . gi0/1 -2]...	Some models need a space before the dash
D(config-if)#[no] shutdown	"No Shutdown" is their way of saying "startup"
D#show interfaces [fa0/0 vlan1][status description]	Layers 1 & 2 information. "status" option useless on routers
D#show protocols [fa0/0]	Tells up/up + IP / CIDR, doesn't tell about CDP or IPv6 routing

## CDP (Cisco Discovery Protocol)

D>show cdp [interface fa0/0   traffic]	An ISO Level-2 Ciscoism
D>show cdp neighbors [fa0/0] [ detail ]	Timer and version information or packet counts
D>show cdp entry [device ID]	Shows directly connected neighbors incl. <u>their</u> port #
D>show cdp entry *	Details. "Outgoing port" is other side (i.e. message-perspective)
D(config)#[no] cdp run	Same as "show cdp neighbors detail"
D(config-if)#[no] cdp enable	Entire device—Default on
	Individual interface participation—default on

## LLDP (Link-Layer Discovery Protocol)

D#show lldp neighbors	Non-Proprietary alternative to CDP
D#show lldp entry [device ID]	Shows directly connected neighbors incl. <u>their</u> port #
	Details. "Port ID" is other side, "Local Intf" is you
D(config)#[no] lldp run	Entire device—Default off
D(config-if)#[no] lldp {transmit receive}	Individual interface—you'll probably want both (2 commands)
D# show lldp interface	Which interfaces configured on

## Time & NTP (Network Time Protocol)

D(config)#clock timezone PST -8	
D(config)#clock summer-time PDT recurring	
D(config)#ntp server <ip> [version {1...4}] [prefer]	Set clock from server. A switch might need a default-gateway
D(config)#ntp master 4	(Unnecessary) Be an NTP server, stratum 4 (radio stratum =1)
D#clock set 15:24:00 16 August 2016	Manual set—Get clock close enough for NTP to take over
D#show clock	
D#show ntp [ status   associations ]	

## Logging & Syslog

D(config)#[no] service timestamps	
D(config)#[no] service sequence-numbers	
D(config)#[no] logging {console   buffered   monitor} 4	4 is severity (shows 0-4) buffer=RAM, monitor=VTYs
D(config)#[no] logging 10.0.0.2	IP address is a syslog server
D(config)#[no] logging trap 4	Levels 0-4 are sent to syslog server (lower = more important)
D#[no] terminal monitor	"This" vty asks to see log messages available via "logging mon"
D#[no] debug <topic>	Extra event logging in realtime—CPU intensive; don't leave on
D#undebug all	Kill all debug logging
D#show logging	Displays log messages in RAM buffer

## SNMP

D(config)#access-list 99 permit host 192.168.2.100	Which host(s) are allowed to read/configure
D(config)#snmp-server community mySnmpPass RO 99	Access {RO, RW} 99 is the ACL (See "Access Control Lists")
D(config)#snmp-server location Red Bluff	Optional
D(config)#snmp-server contact Ben Steel	Optional

## Web Server and Small Services

D(config)#ip http { server   secure-server }	Useful for SDM or CCP (gone from curriculum)
D(config)#ip http authentication local	Uses same usernames as login local
D(config)#[no] service {tcp-small-servers   udp-small-servers}	Ping echos, etc.

---

## Router Config

### Interfaces

R(config)#[no] ip subnet-zero	Allows (default) or prevents a zero subnet address on interface
R(config)#interface { S0/0/0   Fa0/0   lo1}	Serial might be 0/0 on simple routers
R(config-if)#ip address { 172.17.0.1 255.255.0.0   dhcp } [secondary]	
R#show interfaces [fa0/0] [description]	
R#show ip interface [brief fa0/0]	If no IP address, just says "Internet protocol processing disabled"

## SubInterfaces for VLANs

```
R(config)#interface fa0/0
R(config-if)#no ip address
R(config)#interface fa0/0.10
R(config-subif)#encapsulation [dot1q|isl] 10 [native]
R(config-subif)#ip address 192.168.20.1 255.255.255.0
```

R#show vlans

Could put native VLAN IP here with description or "native" below  
Just mentioning it is an automatic "no shutdown"

VLAN config & stats; tells which VLAN native on which interface

## Serial Lines

```
R(config-if)#clock rate 64000
R(config-if)#encapsulation { hdlc | ppp }
R(config-if)#bandwidth 64
R(config-if)#delay 1000
```

Only if you're on the DCE end of the line  
Optional—HDLC Default  
Thousands of bps—used by routing protocol path selection  
Tens of µsecs. Defaults 10µsec for gig, 100 µsec for 100 Mbps.

R#show controllers [ s0/0/0 | fa0/0 ]

Shows if you're DCE or DTE, even if clock rate set at other end

## Serial Lines—PPP Authentication—PAP / CHAP

```
R(config)#username R2 password cisco
R(config-if)#ppp authentication { pap | chap | pap chap | chap pap }
R(config-if)#ppp pap sent-username R password cisco
```

**Optional and only works with PPP encapsulation**  
For PAP or CHAP Authentication. R2=hostname of other router  
R is OWN login and passwd sent to other router

R#show ppp all

One line per interface. Tells if PAP or CHAP is configured

R#debug ppp { authentication | negotiation }

## Serial Lines—PPP Multilink

```
R(config)#interface multilink 42
R(config-if)#ppp multilink
R(config-if)#ppp multilink group 42
R(config-if)#ip address 10.0.0.1 255.0.0.0
```

Virtual interface for the combined links. # matches group #

Group number matches the component physical interfaces  
Now use the entire multilink like you would a single interface

```
R(config)#interface s0/0/0
R(config-if)#encapsulation ppp
R(config-if)#ppp multilink
R(config-if)#ppp multilink group 42
R(config-if)#bandwidth 64
```

Repeat entire block for each involved link

Group number local to the router—other end of link can differ  
Bandwidth on physical interface so multilink can dynamically sum

R#show ppp multilink

Lists member interfaces and if active

R#show interfaces multilink 42

Multilink "open" is good

## PPPoE

```
R(config)#interface Dialer 1
R(config-if)#mtu 1492
R(config-if)#ip address negotiated
R(config-if)#encapsulation ppp
R(config-if)#ppp chap hostname myISPaccount
R(config-if)#ppp chap password myPassword
R(config-if)#dialer pool 42
```

## Encapsulating PPP frames within Ethernet Frames

A Layer 3 interface "template" you'll use for IP routing, NAT, etc.  
Smaller max. IP packet size to make room for PPP headers  
Like receiving a DHCP address, except PPP server uses IPCP

(Optional) Authenticate yourself to the server

Ties this L3 virtual interface to the physical L2 Ethernet interface

```
R(config)#interface GigabitEthernet0/0
R(config-if)#pppoe-client dial-pool-number 42
R(config-if)#mac-address 0200.0000.0001
```

Used only as a Layer 2 Ethernet Interface—no IP address

Matches the number on the dialer interface

(Optional) in case you need to spoof ISP-supplied equipment

```
R#show ip interface brief [Dialer 1]
R#show interfaces Dialer 1
R#show interfaces Virtual-Access 1
R#show pppoe session interface Gi0/0
R#show run interface Virtual-Access 1
R#show interfaces Virtual-Access 1 configuration
```

Shows PPP-supplied IP address and if up/up

Shows PPP status and auto-created virtual-access interface

(Second half of "show interf Dialer 1" output) Shows PPP status

Shows related dialer, virtual-access, and Ethernet interfaces

Auto-created config. The # "1" shared with dialer is coincidence

Includes any CHAP account and password configs

## GRE Tunnel

```
R(config)#interface tunnel 0
R(config-if)#ip address 192.168.12.1
R(config-if)#tunnel source { fa0/0 | 10.12.0.1 }
R(config-if)#tunnel destination 10.12.0.2
```

## Routing can use as exit interface or far end as next hop ip

Our own "real" interface

Other end "real" address we can route to; Other end's "source"

## Static Routes & Routing in General

```
R(config)#[no] ip routing
R(config)#ip route <destIp> <mask> { [next-hop-ip]
```

On by default. Turn off to simulate a host.

| [exit interface] } [adminDist]

For "Floating Static" backup, set adminDist > protocol in use

Default route

```
R(config)#ip route 0.0.0.0 0.0.0.0 s0/0/0
R(config)#ip route 0.0.0.0 0.0.0.0 172.16.24.1
R(config)#ip default-network 192.168.2.0
```

Goes through routing protocols, sets "gateway of last resort"

```
R#show ip route [ <ip-address> | connected | static | rip | ospf | eigrp | ... ] | section <subnet>
R#show ip protocols
```

Tell about any routing protocols that are learning & adding routes

## RIP

```
R(config)#router rip
R(config-router)#version 2
R(config-router)#network 192.168.1.0
R(config-router)#no auto-summary
R(config-router)#redistribute static
R(config-router)#default-information originate
R(config-router)#maximum-paths 4
```

Allows classless routes

Repeat for each network it connects to

Allows it to handle discontinuous subnets

Advertises even the default route

More selective—just default path, not all static routes

How many best routes to load-balance. Range 1-32. Default all

```
R(config-router)#[no] passive-interface fa0/0
R(config-router)#passive-interface default
```

5.2.4.2 Incorrectly says this is in interface mode

Use individual "no" entries for exceptions

```
R(config-if)#ip rip { send | receive } version { 1 | 2 | 1 2 }
```

Configures an interface different from system-wide setting to pair legacy with current

```
R(config-if)#ip summary-address rip 10.0.0.0 255.255.0.0
```

Summary of subnets; must be classful or smaller

R#show ip rip database

Shows known routes

R#debug ip rip

Shows updates sent & received in realtime

<b>EIGRP</b>	
R(config)#router eigrp 10	10 is the AS number. Neighbors must match. 1...65,535
R(config-router)#eigrp router-id 0.0.0.1	Default highest up/up loopback IPv4 addr, normal interf if none
R(config-router)#network <addr> [ 0.0.0.255 ]	Optional wildcard; classful if missing. ( 255...255 – netmask )
R(config-router)#passive interface {fa0/0 default}	Don't use interface for EIGRP updates, but still advertise subnet
R(config-router)#maximum-paths 6	How many (unequal cost routes in routing table. Default 4
R(config-router)#variance 2	Allow unequal load balancing up to 2x metric of best route
R(config-router)#no auto-summary	Prevents summary when advert cross classful net boundaries
R(config-router)#redistribute static	Advertise static routes; could just have a network statement for the non-local destination address mentioned in the static route
R(config)#interf s0/0/0	Explicitly summarize routes shared out this interface as the
R(config-if)#ip summary-address eigrp 10 172.17.0.0 255.255.252.0	following supernet
R(config-if)#ip hello-interval eigrp 10 60	60 seconds for ASN 10; Default 5 ethernet or 60 for T1 & slower
R(config-if)#ip hold-time eigrp 10 180	Default 3x hello-interval
R(config-if)#no ip split-horizon eigrp 10	
R(config-if)#bandwidth 64	1000s of bps (EIGRP can't see the "clock rate 64000" cmd.)
R(config-if)#delay 1000	10s of microseconds (µSecs). 1000 is 0.01 seconds
R#show ip eigrp { neighbors [fa0/0]   interfaces [detail] [fa0/0]   traffic }	
R#show ip eigrp topology [<subnet>   all-links]	Feasible successors are a 2nd "via." All-links shows beyond FSs
R#debug ip eigrp [10]	
R#debug eigrp fsm	Changes to successor and FS routes
<b>OSPF</b>	<b>Doesn't auto-summarize</b>
R(config)#router ospf 10	10 is the process number. Local scope, 1...65,535
R(config-router)#network <addr> 0.0.0.255 area 0	
R(config-router)#area 0 authentication message-digest	Enable Encryption for an area
R(config-router)#router-id 10.0.0.1	Manually set router ID fro DR election
R(config-router)#default-information originate [always]	Propagate 0.0.0.0 via self [even if not in routing table]
R(config-router)#[no] passive interface { fa0/0   default }	Don't use interface for OSPF traffic
R(config-router)#auto-cost reference-bandwidth <Mb/s>	Default is 100, making FastEthernet cost 1, but Gig same...
R(config)#interface s0/0/0	
R(config-if)#ip ospf 10 area 0	Alternative to "network" statements to involve interface in OSPF
R(config-if)#bandwidth 64	Tells OSPF metric calc actual serial line clock rate (64000 bps)
R(config-if)#ip ospf cost 1562	Same as "bandwidth 64" (1562 = 100,000,000 / 64,000 bps)
R(config-if)#ip ospf hello-interval <seconds>	Different order from EIGRP
R(config-if)#ip ospf priority {0-255}	For DR election on one interface (ethernet segment) Default 1
R(config-if)#ip ospf network point-to-point	Don't hold an election on ethernet crossover cables
	Also causes loopbacks to advertise actual mask, not /32
R(config-router)#area 1 range <addr> <mask>	Summary—Tell an ABR to advertise the specified net & mask
R#clear ip ospf [10] process	Reset adjacencies plus new DR election after change router-ID
R#show ip ospf [ neighbor [fa0/0   <ID>]   database ]	Verify adjacencies. (Full, or 2-way for DROthers)
R#show ip ospf interface [ fa0/0   brief ]	
<b>BGP</b>	
R(config)#router bgp 130092	130092 is your AS (Autonomous System) Number
R(config-router)#router-id 1.1.1.1	Optional. Normal order of preference applied if not specified
R(config-router)#neighbor 10.0.0.1 remote-as 64496	Neighbors explicitly defined
R(config-router)#neighbor 10.0.0.1 shutdown	
R(config-router)#network 203.0.113.0 mask 255.255.255.0	Explicit Injection—must already be in your routing table
R(config-router)#ip route 203.0.113.0 255.255.255.0 null0	Overcomes above if you want to advertise a summary
R#show ip bgp [summary]	Shows BGP table of nets. Summary=Router ID, AS, neighbors.
R#show tcp brief	BGP neighbor connections have port 179 in foreign address
<b>Access Control Lists (ACLs)</b>	
R(config)#access-list 1 remark blah blah blah	
R(config)#access-list 1 {permit   deny} {any   host <srcIP>   <srcIP> <wildcard> }	Standard ACL—#s 1-99 and 1300-1999. "host" keyword is useless since default wildcard is 0.0.0.0. ACL can have multiple lines with same #
R(config)#access-list 100 deny { ip   tcp   udp   icmp } <src_ip> 0.0.0.255 {any   host <dest_ip> } { eq   gt   lt } { telnet   23   range 20 21 }	Extended ACL. #s100-199, 2000-2699
R(config)#access-list 100 deny any any log	Final deny-all with "log" to see what's being implicitly blocked.
R(config)#ip access-list { standard   extended } MYACL	Named ACL.
R(config-ext-nacl)#{permit   deny } host <ip> <wld>	
R(config-ext-nacl)#permit any	Overcomes the implicit "deny all else"
R(config)#access-list 101 permit tcp any any established	Allow incoming response to outwent request
R(config)#access-list 101 permit icmp any any { echo-reply   unreachable }	Ping <u>responses</u> onl
R(config)#interface s0/0/0	
R(config-if)#ip access-group 100 in	Apply access-list 100 to the input of an interface
R(config)#no ip access-group s0/0/0	Remove from interface; keep ACL definition
R(config)#line vty 0 4	Apply ACL to a vty
R(config-line)#access-class 100 in	Use access-class instead of "ip access-group" for vty
R#show [ip] access-lists [ 40   MYACL ]	Show the ACL with line numbers for editing. "IP" narrows list.
R(config)#ip access-list {standard   extended } MYACL	"IP" is key to editing mode for named or numbered ACLs
R(config-ext-nacl)#no 20	Delete one line of the ACL. Line # is from "show access-lists"
R(config-ext-nacl)#30 permit...	Replace line 30. (31-39 could insert a line between 30 and 40)
R(config)#ip access-list resequence MYACL 100 10	Renumber lines by 10s starting at 100 (reload also renumbers)
R#show ip interface fa0/0	Shows inbound & outbound applied ACLs
<b>HSRP</b>	<b>Hot Standby Router Protocol—Cisco</b>
R(config-if)#standby 1 version [1   2]	1 is HSRP group number. Max 255 for v1, 4095 for v2
R(config-if)#standby 1 ip 172.16.10.1	Gateway virtual IP. No router has it directly configured
R(config-if)#standby 1 priority 150	Default 100; higher better; 0-255 allowed
R(config-if)#standby 1 name myGroup	Optional
R#show standby [brief]	

## DHCP

```
R(config-if)#ip helper-address <ip> Allows a DHCP LAN broadcast to be forwarded to central DHCP
R(config)#ip dhcp excluded-address <startIP> [ <endIP> ] Don't give out addresses already used (e.g. self)
R(config)#ip dhcp pool myPool Create a pool of addresses
R(dhcp-config)#network <net> { <mask> | /24 } For the pool
R(dhcp-config)#domain-name example.com Things to provide the host along w/ IP #
R(dhcp-config)#dns-server <ip> [<ip>...] Can have >1 dns server
R(dhcp-config)#default-router <ip> Default Gateway, usually self
R(dhcp-config)#lease <days> <hours> <minutes> How long address is promised to client
R(dhcp-config)#next-server <ip> TFTP server for phones (like option 150)

R#show ip dhcp { binding | pool [<name>] | server statistics | conflict} Binding tells assignments
R#clear ip dhcp conflict Empties list after problem fixed
R#show dhcp lease Client command, not DHCP server
```

## IP SLA

```
R(config)#ip sla 42 42 is an arbitrary identifying "entry number" local to the router
R(config-ip-sla)#icmp-echo <ip> source-ip <ip> Choosing an inside source-ip can test routing tables also
R(config-ip-sla-echo)#frequency 5 Ping every 5 seconds (this is minimum allowed)
R(config-ip-sla-echo)#threshold 100 Round-trip time should be under 100 ms
R(config-ip-sla-echo)#history filter all Keep all data in history
R(config-ip-sla-echo)#history buckets-kept 6 Limit history to last 6 pings
R(config-ip-sla-echo)#history lives-kept 1
R(config)#ip sla schedule 42 life forever start-time now

R#show ip sla { configuration | history | statistics } [42]
R#show ip sla summary Not available on older IOS v15.1 (2800-series routers)
```

## NAT/PAT—Network Address Translation

```
R(config)#interface fa 0/0 Configure the interface that will be on the "inside" side
R(config-if)#ip nat inside Declare it to be the inside side of the translating router
R(config)#interface se 0/0 Configure the interface that will be on the "outside" side
R(config-if)#ip nat outside Declare it to be the outside side of the translating router

R(config)#ip nat inside source static [insideIP] [outsideIP] Static—translation for one server
R(config)#ip nat inside source static tcp <IP> <port> <outsideIP> <port> Static for one port on server

R(config)#access-list 12 permit 172.31.232.0 0.0.0.255 Dynamic—Match participating internal hosts
R(config)#ip nat pool myGlobals <startIP> <endIP> { netmask 255... | prefix-length 24} Dynamic—Pool of external addresses for NAT (not PAT)
R(config)#ip nat inside source list 12 pool myGlobals NAT version of static above
R(config)#ip nat inside source list 12 interface s0/0/0 overload PAT version, where the external IP is that of the interface

R#show ip nat { statistics | translations } Translations shows the lookup table
R#clear ip nat translation { * | inside <globalIP> <localIP> | outside <local> <global> Clears
dynamic translations before their 24 hours are up
R#clear ip nat translation {TCP | UDP} inside <globalIP> <port> <localIP> <port> [ outside
<localIP> <globalIP>]

R#debug ip nat
```

---

## Switch Config

```
S#show interfaces {status | decription | counters} Status (e.g. notconnect), duplex, speed, and type (e.g. 10/100)
```

## Mac Address Table

```
S(config)#mac address-table static [XXXX.XXXX.XXXX] interface fastethernet 0/18 vlan 1 Add a static
S#show mac address-table [dynamic|static] [int fa0/1 | vlan 20 | address XXXX.XXXX.XXXX]
S#clear mac address-table [ dynamic ] [ fa0/1 ] Dynamic clears all "learned" MACs
```

## Port Security

```
S(config-if)#switchport port-security Must be in access mode (not trunk)
S(config-if)#switchport port-security maximum 2 Default 1 (command disappears from running-config)
S(config-if)#switchport port-security mac-address [HHHH.HHHH.HHHH |sticky] Sticky learns & puts in run-cfg
S(config-if)#switchport port-security violation [ protect | restrict | shutdown } Restrict adds logging.
Default = shutdown

S#show mac-adress-table secure [ interface fa0/18 ] Lists MACs associated with ports using port security
S#show port-security [ interface fa0/18 ] Enabled or not, Status, #MACs allowed,MACs present
S#show port-security address Table
S#clear port-security sticky interface fa0/18 Clears the sticky address entry
S(config-if)#shutdown To return interface from "error disable" to admin!ly up
S(config-if)#no shutdown
```

## VLANs

```
S(config)#vlan 2 Choose VLAN to configure
S(config-vlan)#name fred Name it
S(config)#no vlan 2 Kills VLAN and deactivates associated ports
S(config)#shutdown vlan 2 Can also say shutdown within the vlan config
S(config-if)#switchport mode access
S(config-if)#switchport access vlan 2 To put the port on a VLAN (default VLAN 1)
S(config-if)#switchport voice vlan 3 To put the port on a VLAN (default VLAN 1)
```

```
S#show vlan [brief | id 20 | name myVlan | summary] Port VLAN memberships (missing ports are trunked)
```

## Trunking

```
S(config-if)#switchport nonegotiate Trunk ports uplink all VLANs
S(config-if)#switchport mode { trunk | dynamic {auto | desirable} } Dynamic invokes DTP, with two options
S(config-if)#switchport trunk native vlan 12 Native VLAN untagged across link
S(config-if)#switchport trunk allowed vlan {all | [add | except | remove] <vlan-list> }
S(config-if)#switchport trunk encapsulation { dot1q | isl | negotiate } Unneeded on 2960s—always dot1q
S(config-if)#switchport trunk pruning vlan 5-1001 Don't flood broadcasts w/in VLAN to switches it's not in use for
hosts or as an administrative VLAN

S#show interfaces [ fa0/2 ] [ switchport | trunk ]
```

## VTP

```
S(config)#vtp domain DomainName
S(config)#vtp mode { client | server | transparent | off }
S(config)#vtp password MyPassword
S(config)#vtp pruning Propogates "on" status; ports on switches tell which VLANs
```

```
S#show vtp { password | status | counters }
```

## Spanning-Tree

```
S(config)#spanning-tree mode [rapid-]pvst
S(config)#spanning-tree vlan 20 priority <n*4096>
S(config)#spanning-tree vlan 20 root [ primary | secondary ]

S(config-if)#spanning-tree [vlan 1,3-5,7] port-priority 16
S(config-if)#spanning-tree [vlan <range>] cost 10

S(config-if)#spanning-tree portfast [disable]
S(config-if)#spanning-tree bpduguard {enable | disable}
S(config)#spanning-tree portfast [bpduguard] default

S#show spanning-tree [vlan 20] [ detail | summary ]
S#show spanning-tree [vlan 20] { bridge | root | interface fa0/1 } [ detail ]
S#show spanning-tree [vlan 20] blockedports
S#debug spanning-tree events
```

## On by default

pv=per vlan  
Lowest priority becomes root. Default = 32768 (midpoint)  
Adjusts the bridge priority once to make root

Default 128. Range 0-240 increments of 16. Low wins  
Port cost changes choices too, but cascades root cost changes

Turns off STP on this interface (port used only for hosts)  
Disable a portfast port if a switch is connected to it.  
Then disable on ports that are the exception

Topology changes

## EtherChannel

```
S(config-if-range)#channel-group 1 mode { on | {desirable|auto} | {active|passive} }
S#show etherchannel [1] [summary | brief | detail | port | port-channel]
```

Speed, duplex, access/trunk, vlan must match

PaGP vs LACP  
Just use summary

## SPAN

```
S(config)#monitor session 1 source interface gi0/1
S(config)#monitor session 2 vlan 22 [- 44] both
S(config)#monitor session 1 destination interface gi0/2
```

Copy frames on a switchport or VLAN to an extra port or addr

{rx|tx|both} Can have more than one source per session  
One session can't mix port and VLAN sources.

```
S#show monitor [session {# | all | local | remote | range #-#}] [detail]
```

## SVI—Allow Administration via net and Pinging

```
S(config)#ip default-gateway [addr]
S(config)#interface vlan 1
S(config-if)#ip address <addr> <mask>
S(config-if)#no shutdown
```

Start up IP process; switches are normally layer 2 only

"Gateway," not "Network" because switch is layer 2, not a router

Give it an IP number

Associated VLAN doesn't pass traffic if its SVI is shutdown

```
S#show ip default-gateway
```

## L3 Routing on a Switch

```
S(config)#sdm prefer lanbase-routing
S(config)#[no] ip routing
```

An SVI is the default gateway for each VLAN

(Then reboot) Used for the 2960; not needed on a real L3 switch  
The "no" can turn a router into a host (needs a default-gateway)

```
S(config-if)#no switchport
S(config-if)#ip address <addr> <mask>
```

Create a L3 (router-like) port on a L3 switch (not in any VLAN)  
Now you can use it as an exit interface for routes

## IPv6

### Fundamentals

```
R(config)#ipv6 unicast-routing
R(config)#ipv6 cef
R(config-if)#ipv6 enable
R(config-if)#ipv6 address FEC0::2:1/112
R(config-if)#ipv6 address FE80::1 link-local
R(config-if)#ipv6 address FEC0:23::/64 eui-64
R(config-if)#ipv6 address FD23::1/64
R(config-if)#ipv6 address autoconfig [default]
R(config-if)#ipv6 address dhcp
```

Not needed for directly-connected networks

Optimization—Learn MACs for each next hop. Requires ipv4 cef  
Optional. Creates link-local if you want no manual address

No space before the /CIDR

Link-local address ( FE80::/10 = FE8 through FEB )

2nd half auto-filled with FFFE in the middle of MAC w/ 7th bit 1

Unique Local (Anything with FD::/8) unroutable private address

SLAAC. "Default" creates default route to rtr that answe'd the RS

Stateful DHCP

```
R#show ipv6 interface [ brief | s0/0/0 ]
R#(ping | traceroute) [ipv6] FEC0::13:1
```

The ipv6 keyword is unnecessary; appears entirely pointless.

### NDP

```
R#show ipv6 { neighbors | routers }
R#clear ipv6 { neighbors | routers }
R#debug ipv6 nd
```

### Neighbor Discovery Protocol

From NDP (NS, NA, RS, RA)—Like IPv4's ARP table

Force table to repopulate without recently stale entries

### DHCP

```
R(config-if)#ipv6 dhcp relay destination <ip6>
```

Allows a DHCP LAN broadcast to be forwarded to cent

### IPv6 ACLs

```
R(config)#ipv6 access-list myACL
R(config-ipv6-acl)#remark Same as IPv4
R(config-ipv6-acl){permit|deny} {ipv6|tcp|udp|icmp} any any [log]
R(config-ipv6-acl)#permit tcp host 2001:db8::1 [eq 80] any
R(config-ipv6-acl)#deny udp 2001:db8::/64
```

There's no such thing as numbered or extended vs. standard

The /64 replaces the wildcard mask of IPv4

```
R#show ipv6 access-list [myACL]
R#show ipv6 interface | include line|list
```

Show all ACLs applied to interfaces ("line" gives interface)

### Routing

```
R(config)#ipv6 route 2001:db8:1111:2/64 s0/0 FE80::FF:FE00:2
R(config)#ipv6 route ::/0 s0/1 [<next hop>]
```

Link-local next hop requires exit interface too

Static default

```
R#show ipv6 protocols
R#show ipv6 route [ ospf | eigrp | static | connected | local | summary ]
R#show ipv6 route 2001:db8:1111:2/64
R#show ipv6 route | section 2001:db8:1111:2/64
```

All sources of routes

Summary just counts type & prefix

### OSPFv3

```
R(config)#ipv6 router ospf 1
R(config-rtr)#router-id 0.0.0.1
R(config-rtr)#[no] passive-interface {fa0/0 | default }
R(config-rtr)#auto-cost reference-bandwidth <Mb/s>
R(config-rtr)#maximum-paths 2
```

Process ID local-scope; Defaults = no sub-commands.

Required if there are no IPv4 addresses on router

Default = 4

```
R(config-if)#ipv6 ospf 1 area 0
R(config-if)#ipv6 ospf cost <interface-cost>
```

Invoke participation from interface, instead of "network" in OSPF

Manually set ospf cost for one interface

```
R#show ipv6 ospf
R#show ipv6 ospf interface [brief | fa0/0]
R#show ipv6 ospf neighbor [fa0/0 | <router-ID>]
R#show ipv6 ospf database
```

Link-State DataBase

## EIGRPv6

```
R(config)#ipv6 router eigrp 1
R(config-rtr)#[no] shutdown
R(config-rtr)#[eigrp] router-id 0.0.0.1
R(config-rtr)#[no] passive-interface {fa0/0 | default }
R(config-rtr)#auto-cost reference-bandwidth <Mb/s>
R(config-rtr)#maximum-paths 2
R(config-rtr)#variance 2

R(config-if)#ipv6 eigrp 1
R(config-if)#ipv6 summary-address eigrp 1 2001::/16
R(config-if)#ipv6 hello-interval eigrp 1 <seconds>
R(config-if)#ipv6 hold-time eigrp 1 <seconds>

R#show ipv6 eigrp interfaces [detail] [fa0/0]
R#show ipv6 protocols
R#show ipv6 eigrp neighbors [fa0/0]
R#show ipv6 eigrp topology [ 2001:db8:1111:2/64 ]
```

## Doesn't auto-summarize

AS Number must match on all routers  
Initial state default varies. 12.4 shutdown, 15.1 not shutdown.  
Req'd if no IPv4 interfaces. Anything OK except 0 & 255.255...

Invoke participation from interface, instead of "network" in OSPF  
Affects advertisements leaving this interface  
Independent of IPv4's EIGRP value

AS, enabled & passive interfaces, summarizations

Lots of detail on one route

## PC Commands

```
{ipconfig [/a] | ifconfig [eth0]}
nslookup <hostname>
netstat -rn
arp -a
{ping | ping6} { <ip> | <hostname> }
{traceroute | tracert | traceroute6 | pathping} { <ip> | <hostname> }
{netch interface ipv6 show neighbors | ndp -an | ip -6 neighbor show}
```

For extended ping, no address or host; it'll quiz you.

## Disavowed

### Frame Relay

```
R(config)#interface serial 0/0/1
R(config-if)#encapsulation frame-relay [ietf]
R(config-if)#frame-relay lmi-type { ansi | q933a | cisco }
R(config-if)#no frame-relay lmi-type
R(config-if)#keepalive <seconds>
R(config-if)#ip address 192.168.0.5 255.255.255.252
R(config-if)#[no] frame-relay inverse-arp
R(config-if)#ip ospf network point-to-multipoint
R(config-if)#frame-relay map ip 192.168.0.6 110 [broadcast] [cisco | ietf]
```

Give it a "no shutdown" also  
ietf specifies encapsulation (framing); default is Cisco  
Allow auto-negotiation (default)  
How often to send / expect LMI status inquiry messages  
Tell OSPF to find neighbors in (non-broadcast) multipoint  
Distant IP, local DLCI. Fixes InARP omissions in partial mesh multipoint where destination IP may be 2 hops away "Broadcast" allows OSPF multicast updates to cross. Cisco/ietf specifies encapsulation per-DLCI  
Shows keepalives, connection to telco status & traffic  
End-to-end PVC status. Per-DLCI within interface groupings  
DLCIs per (sub)interf, Dynam (InARP) or Static, Multipt has far IP

```
R#show frame-relay lmi
R#show frame-relay pvc [interface fa0/0] [ 100 ]
R#show frame-relay map
R#debug frame-relay lmi
R#clear frame-relay inarp
```

Clears the inverse-arp maps

### Frame Relay on Subinterfaces

```
R(config)#interface s0/0/0
R(config-if)#encapsulation frame-relay [ietf]
R(config)#interface s0/0/0.100 { point-to-point | multipoint }
R(config-subif)#frame-relay interface-dlci 100
R(config-subif)#ip address <IP> <Mask>
```

Multipoint allows >1 PVC  
Tells which of the LMI message PVCs are for THIS subinterface

### Netflow

```
R(config-if)#ip flow { ingress | egress }
R(config)#ip flow-export destination <IP> 99
R(config)#ip flow-export version 9
R(config)#ip flow-export source loopback 0

R#show ip cache flow
R#show ip flow { interface | export }
```

Could say both in two statements  
99 is a UDP port number on the collector—differentiates routers  
1, 5, 9 available on Cisco  
Force the source IP address of the netflow packets  
Tells that we're analyzing packets (top line "x total packets")  
Export tells where the server is that we're feeding

### GLBP

```
R(config-if)#glbp 10 ip 10.1.1.3
R(config-if)#glbp 10 priority 110
R(config-if)#glbp 10 name myGroup
R#show glbp [brief]
```

10 is GLBP group, ip is shared virtual gateway address  
For election. Default 100; higher better; 0-255 allowed  
Optional

### Gateway Load Balancing—Cisco

### Keychains

```
R(config)#key chain myChain
R(config-keychain)#key 1
R(config-keychain-key)# key-string myKey
R(config-if)#ip authentication mode eigrp 10 md5
R(config-if)#ip authentication key-chain eigrp 10 myChain
```

Used by RIP and EIGRP—Unusable in PacketTracer  
Set up a key for authentication/encryption—At Global Config  
There can be several keys in a chain (key 2...) Overlap expiration  
On the interface, e.g. s0/0/0