

Cisco Routers for the Intelligent.

This is not a 'for dummies' guide.

Dummies should not be making decisions which can affect the global internet routing table.

(c) Andy Davidson (Devonshire IT Limited.)

Some rights reserved – email sales@devonshire.it with questions regarding redistribution rights.

Trademarks of Cisco Systems Incorporated: Catalyst, Cisco, Cisco IOS, Cisco Press, Cisco Systems, IOS, Linksys, PIX, SMARTnet, The Fastest way to increase your Internet Quotient.

Aims

- To introduce the router, the filesystem, making changes, upgrading IOS
- Interfaces
- Static Routing
- Security
- Debugging network problems

Course is designed as an introduction to running routers at end-sites.

Is designed to help systems-administrators make good choices, when they are told that they are now network administrators.

We don't really teach networking fundamentals, but these can be explained in-line with the Cisco specific slides, in the event that a networking point is not clear.

Introduction

- Runs own operating system - 'Internetwork Operating System' - IOS
- Use Flash Memory rather than disks for permanent/non-volatile storage.
- Startup-Config stored in NVRAM.

Learning Cisco IOS is really useful because the interface style is so widely copied.

NVRAM is often no larger than 256k, so don't fill it up !

Use config option 'service compress-config' to compress the router's configuration file before saving it to NVRAM. This is a global configuration option (this will make sense very soon).

IOS Versions

- Expressed as Major.Minor(Release)Train !
- e.g. 12.4(1)T (first 12.4 IOS release)
- Trains denote which features are available - e.g. 'E' train for 'enterprises', 'T' train is 'technology rich' and has newer features

Sometimes an IOS is quickly rebuilt and packaged up for distribution when a big bug is squashed by Cisco, shortly after the release of a new IOS version. A rebuild IOS version is highly recommended over the originally released version, and will be expressed as a number after the train letter, e.g. 12.3(10)T5 - 5th rebuild of 12.3(10)T.

Getting Help

- ? (online help)

```
bccliffe-gwy#sh ip acc ?
% Ambiguous command: "sh ip acc "
bccliffe-gwy#sh ip accou ?
  access-violations  show access violations in accounting database
  checkpoint          The checkpointed IP accounting database
  output-packets     show output packets in accounting database
  |                  Output modifiers
  <->
```

A modern IOS supports tab completion and context sensitive online help. If you need help completing a command, press the question mark key to see which options are available next.

No need to complete the command with a tab, if there is only one command which will complete if you press tab (i.e. providing that you are non-ambiguous), you can leave it shortened, e.g. (show / sh)

Looking at config

- Difference in context between running-config and startup-config.
- show running-config
- sh startup-config

```
bcliff-guy#sh startup-config
Using 2984 out of 131072 bytes
!
version 12.3
no service pad
service timestamps debug uptime
service timestamps log uptime
service password-encryption
!
hostname bcliff-guy
!
no logging buffered
!
username sysadm privilege 15 password ? 0312546
no aaa new-model
ip subnet-zero
ip dhcp excluded-address 10.1.1.254
ip dhcp excluded-address 10.1.1.1 10.1.1.99
ip dhcp excluded-address 10.1.1.201 10.1.1.254
!
ip dhcp pool CLIENT
import all
network 10.1.1.0 255.255.255.0
default-router 10.1.1.254
```

The configuration that describes how the router is configured NOW is known as the 'running config'. The current config might be different to how the router will be configured after a reboot.

You can inspect the config which will be used at next boot with the 'show running-config' command.

You can save the running config as the next startup config by typing 'copy running-config startup-config' – or a more shorthand 'write memory' (deprecated).

The "oldest trick in the book" is to reboot a router if you make a mistake configuring it, as it will be restored to startup-config. Always, therefore, test changes to routers.

Erasing Configuration

- startup-config stored in NVRAM - therefore just erase that bit of ram..
- erase nvram:
reload
[when prompted to save startup-config say no!]

If a router is changing jobs, then it's a good idea to erase the configuration so that the router starts with a known clean configuration.

You do need to do this on a router which you don't need to access over the network, as the default state of all network interfaces is shutdown

If you accidentally remove the startup config (!) then copy to running config to startup config again.

Users and privileges

- IOS has 15 privilege levels. Users and commands are assigned to a priv. level.
- Needs a privilege level higher or equal to a command to run the command
- use 'enable' to get priv. 15 (highest)
Enable denoted by # prompt.

Users are defined in the running-configuration using the global configuration command 'username'

```
username <name> privilege <priv> password <encryption level> <hash>
```

e.g.

```
username manager privilege 15 password 7 1234567890goats
```

```
username support privilege 7 password 7 0987654321stoat
```

the user 'manager' logs straight in with the highest privs, but the user 'support' only has the ability to view things (e.g. routing table).

Commands & Privileges

- To give junior people the chance to run commands which are normally reserved for superusers:

```
privilege exec level 2 show startup-config
```

- Permits people with priv ≥ 2 to see file.

Sometimes necessary to allow people from outside your company the chance to log in and inspect the state of your router. Is typically a bad idea to start dropping the priv. levels of lots of commands if you have several users on a router, far better to give 'enable' access to the right people !

Additionally, you can increase the privilege required to run a command. Be very careful doing this as increasing the priv. levels for something under 'show ip' raises the privilege for 'show' and 'show ip'. To make the routing table secret to just superusers, would need to apply this config :

```
privilege exec level 15 show ip route
privilege exec level 1 show ip
privilege exec level 1 show
```

Router info

- sh version

hardware
ios version
conf reg
memory
uptime
image
ports

```
boliffe-guy@sh var
Cisco Internetwork Operating System Software
IOS (tm) C837 Software (C837-I903SY6-H), Version 12.3(2)XC2, EARLY DEPLOYMENT RELEASE SOFTWARE (fcl)
Synched to technology version 12.3(1.6)T
Technical Support: http://www.cisco.com/techsupport
Copyright (C) 1986-2004 by Cisco Systems, Inc.
Compiled Thu 04-Mar-04 01:03 by ealvon
Image text-base: 0x800131E0, data-base: 0x80CCB0D0

ROM: System Bootstrap, Version 12.2(0)YIN, RELEASE SOFTWARE (fcl)
ROM: C837 Software (C837-I903SY6-H), Version 12.3(2)XC2, EARLY DEPLOYMENT RELEASE SOFTWARE (fcl)

boliffe-guy uptime is 2 minutes
System returned to ROM by reload
System image file is "flash:c837-i903sy6-mz.123-2.XC2.bin"

This product contains cryptographic features and is subject to United
States and local country laws governing import, export, transfer and
use. Delivery of Cisco cryptographic products does not imply
third-party authority to import, export, distribute or use encryption.
Importers, exporters, distributors and users are responsible for
compliance with U.S. and local country laws. By using this product you
agree to comply with applicable laws and regulations. If you are unable
to comply with U.S. and local laws, return this product immediately.

A summary of U.S. laws governing Cisco cryptographic products may be found at:
http://www.cisco.com/wll/export/crypto/tool/starg.html

If you require further assistance please contact us by sending email to
export@cisco.com.

C1500 C837 (MPC8570SL) processor (revision 0x500) with 44237K/4915K bytes of memory.
Processor board ID A1E0631053E (979107125), with hardware revision 0000
CPU rev number 7
Brldging software.
4 Ethernet/IEEE 802.3 interface(s)
4 FastEthernet/IEEE 802.3 interface(s)
2 ATM network interface(s)
128K bytes of non-volatile configuration memory.
12288K bytes of processor board System flash (Read/Write)
2048K bytes of processor board Web flash (Read/Write)

Configuration register is 0x2102
```

show version indicates quite a lot of useful data, and should be inspected before certain tasks are performed

e.g. ios upgrade – demonstrates

- hardware platform
- whether enough memory to upgrade ios
- version of software running
- where the current image boots from

additionally which ports are physically available in this router, and the config register, which is a system that defines initial power-on behaviour (more soon)

Configuration

- Run config terminal (conf t)
- Start typing commands as they would appear in a show running-config
- Remove lines from config (or disable default features) by prepending 'no'.

```
bcliff-gw#conf t
Enter configuration commands, one per line. End with CNTL/Z.
bcliff-gw(config)#ip http-server
      ^
% Invalid input detected at '^' marker.
bcliff-gw(config)#ip http server
bcliff-gw(config)#no ip http server
```

Always use a fixed character length terminal as mistakes get highlighted with a ^ at the first point of mistake.

You must be in 'enable' mode to make changes

Interfaces and such

- in sh running config, some parts are indented, e.g. 'interface' blocks
- indentations are a change of the 'conf t' context - specific features/commands become available.

```
bcliffe-gwy#conf t
Enter configuration commands, one per line. End with CNTL/Z.
bcliffe-gwy(config)#int tunnel 100
bcliffe-gwy(config-if)#ip address 192.168.100.1 255.255.255.252
bcliffe-gwy(config-if)#exit
bcliffe-gwy(config)#ip route 192.168.200.0 255.255.255.0 tunnel 100
```

The 'ip address' command only makes any sense whatsoever when it is applied to an interface, therefore the address is not available in the global context. You must specify the interface that it applies to, and in doing so, the interface mode/context is activated, and the interface specific commands become available. Note how the config line changes to config-if To move back to the global context, just type exit

To back out these changes, use

```
no ip route 192.168.200.0 255.255.255.0 tunnel 100
```

```
no int tunnel 100
```

(killing the tunnel 100 interface kills the config which was applied in that interface too)

A word on interface defaults

- By default an interface is in 'shutdown' mode
no shut when you want to wake it up
- By default 'cisco discovery protocol' is enabled. Unless you want to use it on your own network do
no cdp enable

no shutdown is probably the sensible default for ports – but do remember you need to 'no shutdown' the ports before sending the router off to the data centre.

cdp allows you to see information about the connection between two cisco devices. It tends to be quite chatty, so if you use it on your own network, it is considered not very polite to leave it enabled on your connections to other networks.

Also need to put 'ip routing' into the default context in order to actually route packets..

Also good idea to ensure that 'ip classless' is in global context in order to honour CIDR addressing.

Config Register

- Defines boot time behaviour.
- If config-register last octet is not 2, will ignore the 'boot system' commands.
- 0x2102 is the factory-default configuration register value.
- 0x2142 boots from flash without using NVRAM contents good for password recovery.
- 0x2101 boots from boot prom image not flash, good for upgrading image on flash.
- 0x2141 boots from boot prom and ignores NVRAM contents.
- 0x141, which disables the Break key, ignores the NVRAM configuration, and boots the default system image from ROM.

Boot time options seem almost infinitely configurable. Whilst the router is in normal operation, it would be normally wise to leave it to 0x2102 – in fact you should only deviate from this setting if you have good reason to.

Changed by using the 'config-register' command in conf t

Unlike any other configuration command, the changes to the 'config-register' are saved immediately to nvram when you change the setting (i.e. no copy running-config startup-config required.) sh ver will show that config-register is about to change at next boot. Wrong config-registers mean you could setup a router, and then reboot, then have no working config at all (restored to factory). Your author has lost two hours work doing that. But only once.

Out of hours upgrades

- You can replace the 'reload' command with a scheduled reload request.
- reload at 02:00 Dec 30
Reload scheduled for 02:00GMT Sat Dec 30 2006 (in 35 hours and 12 minutes)
Proceed with Reload? [confirm]
- Can use reload to prevent yourself being locked out of routers during work (schedule a reboot, perform work, cancel)

If performing a change on a remote router, can configure IOS to reboot in 15–20 minutes, and then perform the work.

If the work is completed ok, cancel the reboot with 'reload cancel' (will say ---- SHUTDOWN ABORTED ----).

Interesting things to do

with a lonely Cisco router.

We've updated IOS, created user accounts for the team and become familiar with moving around the IOS filesystem and configuration.

We're not actually connected to any networks, though. Therefore we are not actually passing any packets. Let's look at the role of interfaces and the routing table.

A typical Ethernet interface.

```
interface FastEthernet0/0
ip address 213.143.1.254 255.255.255.0
duplex auto
speed auto
no cdp enable
```

- We drop an ip address on the interface, no shut, and disable cdp. Everything else is default. Assume address is 213.143.1.254
- Router 'knows' it can reach 213.143.1.0/24 via this interface now.

```
C 213.143.15.0/24 is directly connected, FastEthernet0/0
```

To bring this interface up, we run
conf t
int fa 0/0
ip address x.x.x.x y.y.y.y
no cdp enable
no shutdown

When the interface is up (line protocol is up), then the router knows that all devices on that subnet are visible across that interface (directly connected). (sh ip route)

A typical end-site scenario

- interface Fa0/0 on 1.2.3.2/30
interface Fa0/1 on 2.3.4.254/24 (DMZ)
- interface Fa0/0 is on ISP network, so that the ISP know where to send your DMZ's data (2.3.4.anything) to.
- routing table shows
2.3.4.0/24 is directly connected, FastEthernet0/1
1.2.3.0/30 is directly connected, FastEthernet0/0

We are an end-site with one ISP. This is a very typical connection scenario. The ISP has assigned a block of 4 addresses to handle the router-to-router connection (4 addresses gives 2 usable addresses, one for the router at the isp, one for the end site router.) and a block of 24 addresses to host publicly reachable devices for this end-site.

The ISP routes all connections for 2.3.4.0/24 to the router on 1.2.3.2.

We are missing a default route. Until we set one, we can only see the two networks we are directly connected to.

The default route

- interface Fa0/0 on 1.2.3.2/30
interface Fa0/1 on 2.3.4.254/24
- ISP router is 1.2.3.1 - therefore dfl route is
ip route 0.0.0.0 0.0.0.0 1.2.3.1
- Routing table expresses this as 'gateway of
last resort'.

End Site routing tables.

- What's happening here? (sh ip route)

```
Gateway of last resort is 0.0.0.0 to network 0.0.0.0

  213.228.240.0/24 is variably subnetted, 2 subnets, 2 masks
C       213.228.240.161/32 is directly connected, Dialer1
S       213.228.240.160/28 is directly connected, Ethernet0
       10.0.0.0/24 is subnetted, 1 subnets
C       10.1.1.0 is directly connected, Ethernet0
       217.47.76.0/32 is subnetted, 1 subnets
C       217.47.76.73 is directly connected, Dialer1
S*    0.0.0.0/0 is directly connected, Dialer1
beliffe_guy@#
```

Mixture of Static Routes, and routes that we see by virtue of being connected to the network.

C signifies a route that we know about thanks to having an interface up on this network.

S signifies a route that we have explicitly

ip route 0.0.0.0 0.0.0.0 Dialer1 <--- note here that it is attached to an interface, not an ip address – useful in dynamic environments.

ip route 213.228.240.160 255.255.255.240 Ethernet0

If the line is down, then a static route that is created against this interface does not appear in the routing table

NAT

- Access Lists define hosts/services which should be affected by a policy on the router
- NAT requires you to define a network to be NATted, and configure nat overload
- e.g. internal network is 10.0.0.0/8 and external interface is FastEthernet 0/1

```
access-list 2 permit 10.0.0.0 0.0.0.255  
ip nat inside source list 2 interface FastEthernet0/1 overload
```

```
int Fa0/0  
  ip nat inside  
int Fa0/1  
  ip nat outside
```

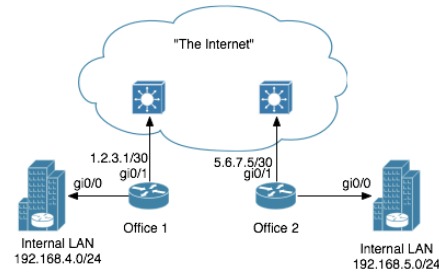
Access lists will be defined more thoroughly when we look at applying rules to packets as they pass through the router, but they need not be understood in detail to configure NAT.

Build a list of hosts which are allowed to use NAT – 10.0.0.0 with a netmask of 255.0.0.0 is 10.0.0.0/8. The inverse of the mask is 0.0.0.255

Tell the router that this list of addresses is allowed to NAT through a particular interface

Place NAT rules on the interfaces

Configure 'Office 1'



- You have enough information here to configure two interfaces & routing table on the router called 'Office 1'

Hints – you don't know what model the router is (unless you have a handy router in the training scenario), so assume you need to turn off all bad defaults.

To keep this easy, the physical presentation of connectivity from your ISP is ethernet, and will fit into the gi0/1 slot on your router (GigabitEthernet 0/1 i full). Also we will only configure features on the router which have been mentioned in this document. **Beware NAT!**

Your internal office has one subnet, 192.168.4.0/24.

Answer

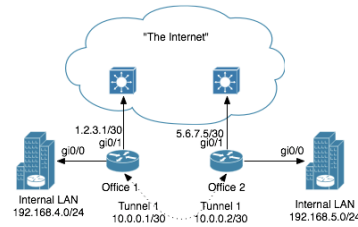
- hostname office1
- ip classless
- int Gi0/0
ip address 192.168.4.254 255.255.255.0
no cdp enable
ip nat inside
- int Gi0/1
ip address 1.2.3.1 255.255.255.252
no cdp enable
ip nat outside
- access-list 2 permit 192.168.4.0 0.255.255.255
ip nat inside source list 2 interface GigabitEthernet0/1 overload
- ip route 0.0.0.0 0.0.0.0 1.2.3.2
- ip routing

Should have basic routing now.

A point of order on NAT – what IP address will the outside world see this router as ?

1.2.3.1. Do you understand why? ip nat outside

Tunnel Interfaces



- The company wants the internal networks at both sites to be visible to all users
- A small, private /30 is used for ip conversations - ensure this subnet is unique

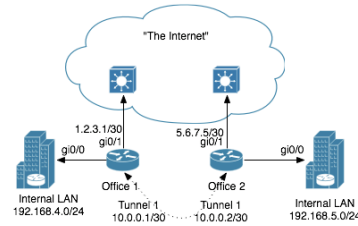
Tunnel interfaces have much in common with physical interfaces. An ip address should be assigned to the interface, and this interface should not overlap any other subnets in use. Is good practice to define a large network which can be subnetted into lots of /30s. User traffic wont really care about these addresses, they are just used for router-router conversation.

Don't assign and forget the addresses, they do show up in traceroutes, so will be important diagnostic aids.

A tunnel interface 'pretends' that there is a direct connection between one router and another. In reality, the tunnel is routed via existing ethernet/isdn/atm/other physical connections.

Tunnels are useful to expose two nat networks to each other, or to encrypt packets between two public networks, or allow point-to-point protocols to work over long distances.

Configuring Tunnels



- On office 1

```
int Tunnel1
ip address 10.0.0.1 255.255.255.252
tunnel source 1.2.3.1
tunnel destination 5.6.7.5
tunnel mode ipip
tunnel key 0987654321
```

The tunnel source and destinations need to be reachable (from the point of view of the other router in the pair at least !)

Tunnel mode can be aurp (Appletalk) cayman (for connecting to systems from Cayman Systems) dvmrp (Multicast) eon ipip "gre ip" or |nos

Commonly use ipip or ip gre.

Tunnel key is a quick way to stop man-in-the-middle style attacks.

Tunnels and the routing table

- Bringing the tunnel up adds a directly connected 'C' route only for the internal network used for the tunnel interfaces
- Also need to add static routes, e.g. office 1

```
ip route 192.168.5.0 255.255.255.0 Tunnel1
```

(and matching reverse route at the other end)
Shows up as 'S' route in 'sh ip route'

Quick DHCP Server

```
no ip dhcp use vrf connected
ip dhcp excluded-address 10.1.1.254
ip dhcp excluded-address 10.1.1.1 10.1.1.99
ip dhcp excluded-address 10.1.1.201 10.1.1.254
!
ip dhcp pool CLIENT
import all
network 10.1.1.0 255.255.255.0
default-router 10.1.1.254
domain-name nosignal.org
dns-server 213.232.80.2 195.92.195.92
lease 0 2
```

- 'network' specifies the range of addresses
- exclude the router, and ranges for statically address devices.

The block inside the 'ip dhcp pool CLIENT' config specifies what is sent to the client, and not anything that configures the router, allowing the router to have a different set of dns servers than what get passed to clients.

lease option can take up to three arguments

lease days hours mins

lease 2 = lease for 2 days

lease 0 0 10 = lease for ten minutes.

Access Lists

A very basic overview.

Types

- Used for firewalling and other policies
- Access List number defines type of rule.
- 1-99 - Standard IP (basic lists of hosts)
- 100-199 - Extended IP (match on source or dest, and specific ports)
- Less common to use higher number than 199 (used for Appletalk/IPX...)

Access Lists are the elements of a router rule that defines which hosts or services will be affected by the rule. They work alongside a rule.

e.g. a shopping list defines the list of items you want to buy from a supermarket.

the supermarket is the framework with the actual products (packets), and checkouts (interfaces)
Without both of these things, the access list/rule lacks a real purpose.

Normally will use access lists numbered 1 to 99 to define a list of addresses only. If wanting to match a specific source/destination/service, need ip extended (101-199).

Standard IP access lists

- Numbered 1-99
- Just a list of IP addresses, nothing else.
- Though can be 1 address or a whole subnet
- `access-list 2 permit 10.22.0.0 0.0.255.255`
`access-list 2 permit 10.33.0.0 0.0.255.255`
`access-list 2 permit host 10.1.1.1`

Just add lots of access-l lines if you need to specify large numbers of subnets or addresses ..

Use the 'netmask' command to work out masks of addresses from CIDR notation.

```
factory:~ andy$ /usr/local/bin/netmask -i 10.22.0.0/16  
10.22.0.0 0.0.255.255
```

Extended IP access List

- Numbered 100-199
- Must have one 'permit' line or all traffic is dropped if assessed against that acl
- These two are equivalent :

```
access-list 101 permit ip 10.1.1.0 0.0.0.255 172.16.1.0 0.0.0.255
```

```
access-list 102 permit ip 10.1.1.0 0.0.0.255 172.16.1.0 0.0.0.255  
access-list 102 deny ip any any
```

102 is much easier to read though.

The first example of an extended access-list that we will look at defines a source and destination address

The rule says anything on the 10.1.1.0/24 network is allowed to reach anything on the 172.16.1.0/24 network, but no other routing is allowed.

Extended lists extended

- Can also make a rule match a given service.
- `access-list 102 permit tcp host 10.1.1.1 host 172.16.1.1 eq www`
- The bit after `eq` is actually the port, no protocol inspection is done. The port is translated into a well known service as defined by rfc 1700.
- Remember access-lists do not work on their own, they need something to kick them into action....

```
bcliffe-gwy(config)#access-l 101 permit tcp host 1.2.3.4 host 2.3.4.5 eq ?
```

```
<0-65535>  Port number
```

```
bgp      Border Gateway Protocol (179)  chargen  Character generator (19)  cmd      Remote commands (rcmd, 514)  daytime  Daytime (13)
```

```
discard  Discard (9)  domain   Domain Name Service (53)  echo     Echo (7)  exec     Exec (rsh, 512)
```

```
finger   Finger (79)  ftp      File Transfer Protocol (21)  ftp-data FTP data connections (20)  gopher   Gopher (70)
```

```
hostname NIC hostname server (101)  ident    Ident Protocol (113)
```

Turning lists on

- Simple 'firewall style' access lists (permit/deny) can be configured simply by attaching the access-l to an interface
- ```
int FastEthernet0/0
 ip access-group 101 in
```
- in = traffic that arrives at that interface
- out = traffic that has been through router

**Hardening**

# The really bad default

Results 1 - 10 of about 1,310,000 for [username cisco password cisco](#). (0.12 seconds)

- New routers tend to ship with one well published default :  
username: cisco  
password: cisco
- conf t  
username <me> privilege 15 password <x>  
don't forget to test
- conf t  
no username cisco

## Old fashioned (password only) access.

- Console port - enabled by default & the password recovery port.
- AUX port - provides out of band access via a modem.
- Virtual TTY (telnet, ssh)
- Use 'line' command in conf t to configure login capabilities.

Some methods of access to the router are called 'out of band' – means no network is required to see the device (e.g. via Telephone line or serial console.)

"old fashioned" because this method does not use one of the authentication methods e.g. AAA which forces username and password. Need to know password only.

## Old fashioned access

- To configure access for console port

```
conf t
line console 0
login
password <console-password>
```

- 'login' means allow logins.

Never put a modem on the console port – remember this is the port which can be used for password recovery, so it's the same as handing out physical access to the router !

# Old fashioned vtys

- There are 5 vty ports, so to configure all five looks like

```
conf t
line vty 0 4
login
password <vty-password>
```

# Old fashioned enable

- enable password <pass>
- enable secret <pass>
- enable password is provided for backwards compatibility and not recommended
- Therefore always use enable secret.
- service password-encryption

```
conf t
enable secret enable-password
exit
write memory
```

service-password encryption stops plain text passwords appearing in a sh run ... but uses Vigenere password encryption which is reversible.

# Local usernames

- username bob password letmein

```
line vty 0 4
 login local
 exit
line con 0
 login local
 exit
```

'login local' says to use local authentication with usernames and passwords.

# no login

- If 'login' is the command to log in, what is 'no login'
- if 'password' sets the password, does 'no password' let people log in with no password ?
- NO. To turn off a port use  
no password  
(tells router not to let anyone log in)

no login lets anyone log in without a username or password ..... so never use it !

no password doesn't mean let anyone in, it turns off permissions to log in. login doesn't mean 'let someone log in', it means 'ask people questions before letting them in'. therefore 'login' and 'no password' cause error on connection attempt: 'password required, but not set'

Strongest rules to turn off port are:

```
line aux 0
login local no password transport input none no exec exec-timeout 0 1
```

# ssh

```
bcliffe-gwy#conf t
Enter configuration commands, one per line. End with CNTL/Z.
bcliffe-gwy(config)#hostname bcliffe-gwy
bcliffe-gwy(config)#ip domain-name nosignal.org
bcliffe-gwy(config)#crypto key generate rsa
The name for the keys will be: bcliffe-gwy.nosignal.org
Choose the size of the key modulus in the range of 360 to 2048 for your
 General Purpose Keys. Choosing a key modulus greater than 512 may take
 a few minutes.

How many bits in the modulus [512]: 1024
% Generating 1024 bit RSA keys, keys will be non-exportable...[OK]

bcliffe-gwy(config)#ip ssh
bcliffe-gwy(config)#ip ssh time-out 60
bcliffe-gwy(config)#ip ssh authentication-retries 2
bcliffe-gwy(config)#line vty 0 4
bcliffe-gwy(config-line)#transport input ssh
bcliffe-gwy(config-line)#^Z
bcliffe-gwy#write memory
```

To enable telnet you need a hostname and domain defined (but you may have these). If you need to set them use the example above.

As ssh isn't old fashioned, it cannot use line configuration (e.g. you must use local passwords or AAA).

## 'firewalling' a vty

- `access-list 23 permit 10.0.0.0 0.255.255.255`
- `line vty 0 4`  
`access-class 23 in`
- By default, vty ports allow any IP to connect and try to authenticate.

# AAA

- Authentication
- Authorization
- Accounting
- We get the first two with local usernames and user privilege levels, but the third A needs aaa to be configured.

```
conf t
aaa new-model
aaa authentication login default local
```

```
then say what to log
aaa accounting exec default start-stop group local
aaa accounting system default stop-only group local
```

```
.
.
```

# Solving Problems

# arp

- Default arp cache timeout is 4 hours.
- clear arp-cache
- sh ip arp

| Protocol | Address         | Age (min) | Hardware Addr  | Type | Interface |
|----------|-----------------|-----------|----------------|------|-----------|
| Internet | 10.1.1.10       | 0         | 0014.515c.212d | ARPA | Ethernet0 |
| Internet | 213.228.240.163 | 0         | 00a0.c9dc.8b97 | ARPA | Ethernet0 |
| Internet | 213.228.240.162 | 0         | 00d0.b79a.8130 | ARPA | Ethernet0 |
| Internet | 10.1.1.102      | 0         | 0007.0eb3.d531 | ARPA | Ethernet0 |
| Internet | 213.228.240.175 | 0         | Incomplete     | ARPA |           |
| Internet | 10.1.1.100      | 0         | 0009.4552.0000 | ARPA | Ethernet0 |
| Internet | 10.1.1.116      | 0         | 000d.9388.034e | ARPA | Ethernet0 |
| Internet | 10.1.1.254      | -         | 0011.bbbd.bf12 | ARPA | Ethernet0 |

The ARP table is a mapping between a devices L2 or Hardware address, the interface that they could be seen through, and their L3/IP address. The table is built dynamically through a short conversation on the wire :

[from tcpdump]

14:58:50.218394 arp who-has 10.1.1.100 tell 10.1.1.116

14:58:50.227331 arp reply 10.1.1.100 is-at 00:09:45:52:00:00

Can also run show ip arp <ip address> or sh ip arp <xxxx.xxxx.xxxx>

Sometimes required to clear the arp cache. Reduce the 4hr cache timeout on a per-interface basis

```
int eth0
```

```
arp timeout 600
```

# Static Hosts

- Very much like `/etc/hosts`
- `ip host mail 10.1.1.2`  
`ip host ftp 10.1.1.3`
- Makes provision for a 'default port'  
`ip host mail 25 10.1.1.2`  
`telnet mail`  
*Trying mail (10.1.1.2, 25)... Open*

You might like to use static hosts so that your router is not dependent upon dns in order to resolve names. This may not work well in your organisation if you use DNS. You can use static hosts to override dns for testing.

# IP Accounting

- “League Tables”
- Attached to an interface :

```
bcliffe-gwy(config)#int ethernet0
bcliffe-gwy(config-if)#ip accounting ?
 access-violations Account for IP packets violating access lists on this
 interface
 mac-address Account for MAC addresses seen on this interface
 output-packets Account for IP packets output on this interface
 precedence Count packets by IP precedence on this interface
 <CR>

bcliffe-gwy(config-if)#ip accounting output-pa
bcliffe-gwy(config-if)#ip accounting output-packets
bcliffe-gwy(config-if)#end
```

In an emergency, (i.e. too much traffic saturating an interface) use 'output-packets' to start counting how much data is hitting an interface.

# accounting output

- 'bytes' shows where the most data is heading.

```
bciliffe-gwy#sh ip accounting output-packets
Source Destination Packets Bytes
207.46.26.131 10.1.1.116 6 918
213.232.80.2 213.228.240.162 4 652
213.232.80.2 10.1.1.116 3 883
217.47.220.121 213.228.240.163 1 56
24.202.33.192 213.228.240.162 2 122
195.92.195.92 213.228.240.162 4 824
89.107.41.3 10.1.1.116 684 119264
213.232.80.121 10.1.1.116 84 75897
195.225.218.240 10.1.1.116 91 57579
213.230.31.205 213.228.240.162 3 144
213.230.31.205 213.228.240.163 3 144
213.166.5.129 213.228.240.162 14 1065
72.14.253.93 10.1.1.116 2 80
192.246.69.186 213.228.240.162 12 834
216.73.95.11 10.1.1.116 7 465
207.46.109.37 10.1.1.102 8 384
195.171.63.140 10.1.1.116 32 2592
83.138.172.235 10.1.1.116 54 11724
207.46.109.20 10.1.1.116 28 4907
217.19.79.23 213.228.240.162 6 2660
Accounting data age is 5
```

Don't leave ip accounting switched on, it's a performance killer.