DAY 4 - Intro to CLI

Purinat33

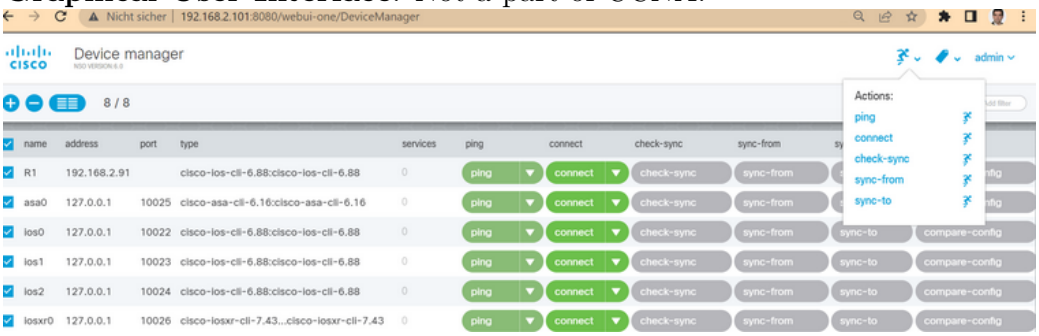
Cisco IOS = Operating System for Cisco devices

Interacting with Cisco IOS:

1. **Command Line Interface**: More common and is in CCNA

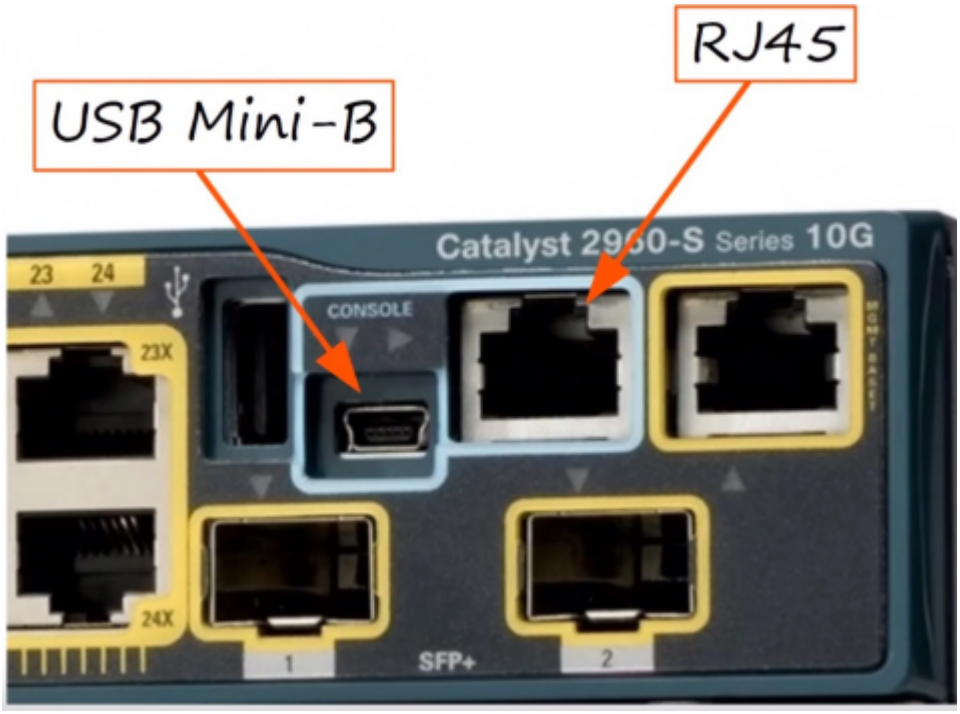
```
Router>enable
Router#configure terminal
Enter configuration commands, one per line.  End with CNTL/Z.
Router(config)#interface g0/0/0
Router(config-if)#ip address 192.168.1.1 255.255.255.0
Router(config-if)#no shutdown

Router(config-if)#
%LINK-5-CHANGED: Interface GigabitEthernet0/0/0, changed state to up

Router(config-if)#exit
Router(config)#do copy running-config startup-config
Destination filename [startup-config]?
Building configuration...
[OK]
Router(config)#
```

2. **Graphical User Interface**: Not a part of CCNA.

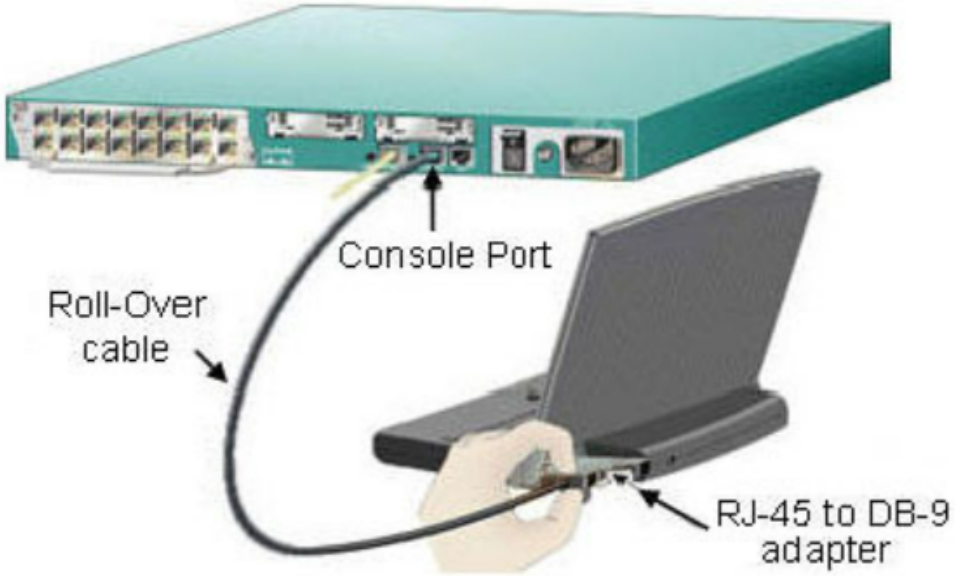


Connecting to a Cisco's device

Commonly connect using the device's **Console Port**.

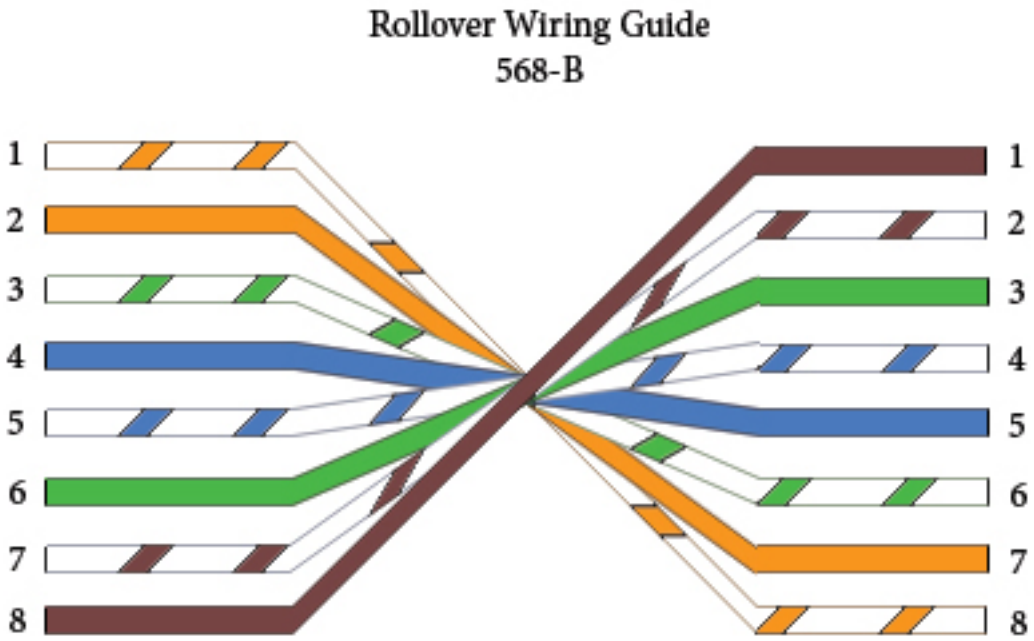


Using **Rollover Cable** with **DB-9** connector to the PC.



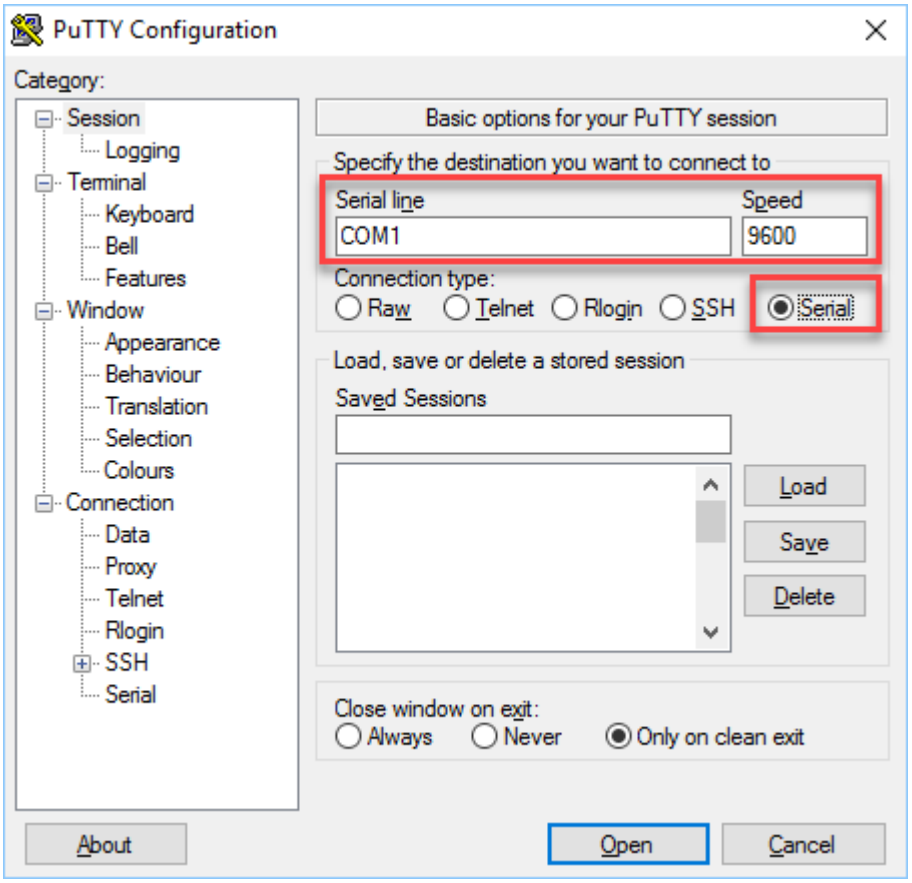
Rollover Cable with **DB-9** connector on the PC end:

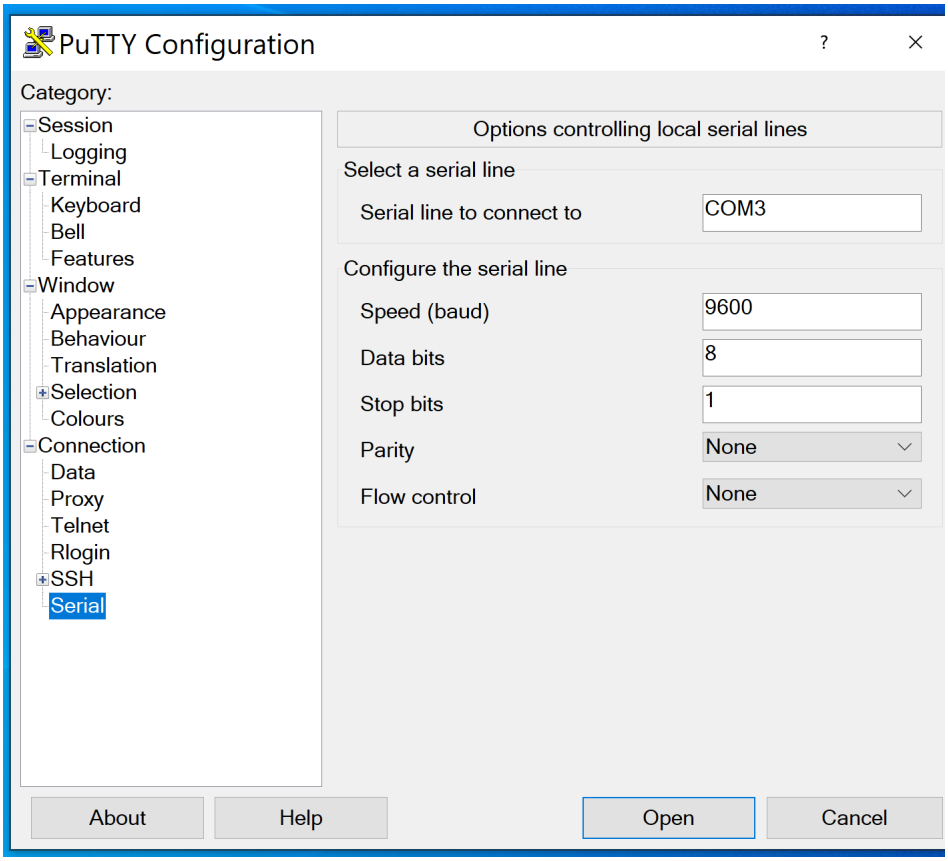
Rollover Cable pins:



Connecting to the IOS Terminal

Using a **Terminal Emulator** (eg. *PuTTy*)





PuTTy's default setting is the same as Cisco IOS so there's no need to change anything. The settings are beyond the scope of CCNA.

Inside the CLI

Booting up the device for the first time will ask if we want to proceed with the initial configuration dialog.

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

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

cisco ISR4331/K9 (1RU) processor with 1795999K/6147K bytes of memory.
Processor board ID FLM232010G0
3 Gigabit Ethernet interfaces
32768K bytes of non-volatile configuration memory.
4194304K bytes of physical memory.
3207167K bytes of flash memory at bootflash:.
0K bytes of WebUI ODM Files at webui:.



         --- System Configuration Dialog ---

Would you like to enter the initial configuration dialog? [yes/no]: no


Press RETURN to get started!
```

Mode of operations:

1. **User EXEC Mode**:
   (a) Default mode when entering the terminal.
   (b) Indicated by the `>` next to the Host Name.

   (c) Very Limited. Users can look at somethings but **cannot** make any changes to the configuration.

```
Router>?
Exec commands:
  <1-99>      Session number to resume
  connect     Open a terminal connection
  disable     Turn off privileged commands
  disconnect  Disconnect an existing network connection
  enable      Turn on privileged commands
  exit        Exit from the EXEC
  logout      Exit from the EXEC
  ping        Send echo messages
  resume      Resume an active network connection
  show        Show running system information
  ssh         Open a secure shell client connection
  telnet      Open a telnet connection
  terminal    Set terminal line parameters
  traceroute  Trace route to destination
```

2. **Privileged EXEC Mode**:
   (a) Seen from the # next to the host name.
   (b) Access via the `enable` command from **User EXEC Mode**

```
Router>enable
Router#
```

   (c) Provide complete access to view the device's configuration, restarting the device etc.
   (d) Cannot change the configuration, but can change the time, save the configuration file etc.
   (e) Comparing **User EXEC Mode** commands and **Privileged EXEC Mode** commands:

```
Router>?
Exec commands:
  <1-99>      Session number to resume
  connect     Open a terminal connection
  disable     Turn off privileged commands
  disconnect  Disconnect an existing network connection
  enable      Turn on privileged commands
  exit        Exit from the EXEC
  logout      Exit from the EXEC
  ping        Send echo messages
  resume      Resume an active network connection
  show        Show running system information
  ssh         Open a secure shell client connection
  telnet      Open a telnet connection
  terminal    Set terminal line parameters
  traceroute  Trace route to destination
Router>
```

```
Router#?
Exec commands:
  <1-99>      Session number to resume
  auto        Exec level Automation
  clear       Reset functions
  clock       Manage the system clock
  configure   Enter configuration mode
  connect     Open a terminal connection
  copy        Copy from one file to another
  debug       Debugging functions (see also 'undebug')
  delete      Delete a file
  dir         List files on a filesystem
  disable     Turn off privileged commands
  disconnect  Disconnect an existing network connection
  enable      Turn on privileged commands
  erase       Erase a filesystem
  exit        Exit from the EXEC
  logout      Exit from the EXEC
  mkdir       Create new directory
  more        Display the contents of a file
  no          Disable debugging informations
  ping        Send echo messages
  reload      Halt and perform a cold restart
 --More--
```

Trivia:

- Use the `?` command to view available commands.
- Appending `?` will show available commands containing the prepended strings.

```
Router#c?
clear   clock   configure   connect   copy
```

3. **Global Configuration Mode**
   (a) Seen from the **(config)** next to the host name.
   (b) Access using `configure terminal` command from the **Privileged EXEC Mode**

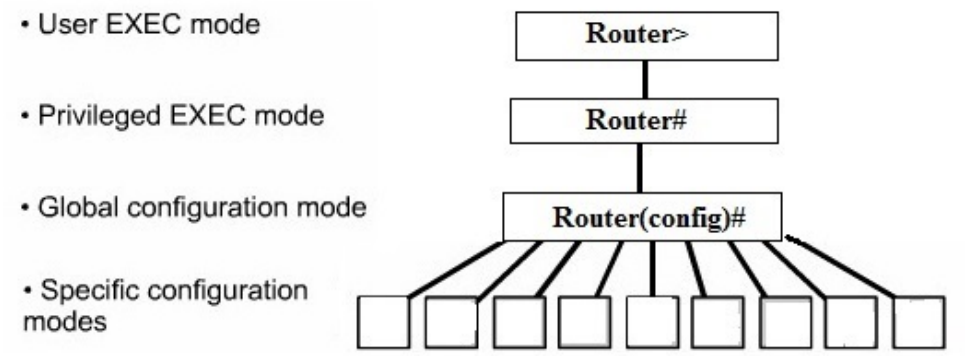
```
Router#configure terminal
Enter configuration commands, one pe
Router(config)#
```

   (c) Allows configuration on a *device* scale (not port/interface).

```
Router(config)#?
Configure commands:
  aaa                 Authentication, Authorization and Accounting.
  access-list         Add an access list entry
  banner              Define a login banner
  bba-group           Configure BBA Group
  boot                Modify system boot parameters
  cdp                 Global CDP configuration subcommands
  class-map           Configure Class Map
  clock               Configure time-of-day clock
  config-register     Define the configuration register
  crypto              Encryption module
  default             Set a command to its defaults
  do                  To run exec commands in config mode
  dot11               IEEE 802.11 config commands
  enable              Modify enable password parameters
  end                 Exit from configure mode
  exit                Exit from configure mode
  flow                Global Flow configuration subcommands
  hostname            Set system's network name
  interface           Select an interface to configure
  ip                  Global IP configuration subcommands
  ipv6                Global IPv6 configuration commands
```

   (d) Allows execution of **Privileged EXEC Mode** commands by adding a `do` command before adding any **Privileged EXEC Mode** commands afterwards.
      - eg. `Router(config)# do reload` is the same as `Router# reload`

4. **Sub Configuration Mode**
   (a) Allows for configuration on specific interfaces (eg. port)
   (b) Accessed from **Global Configuration Mode** and will depend on what we want to set. (Will be covered later. No need to remember these sub-modes right now).



Configuration Mode	Prompt
Interface	Router(config-if)#
Subinterface	Router (config-subif)#
Controller	Router(config-controller)#
Map-list	Router(config-map-list)#
Map-class	Router (config-map-class)#
Line	Router(config-line)#
Router	Router(config-router)#
IPX-router	Router(config-ipx-router)#
Route-map	Router(config-route-map)#

(Will be covered later. No need to remember these sub-modes right now!)

Device's Configuration Files:

Each device have **2 separate configuration files** being stored on the device at the same time.

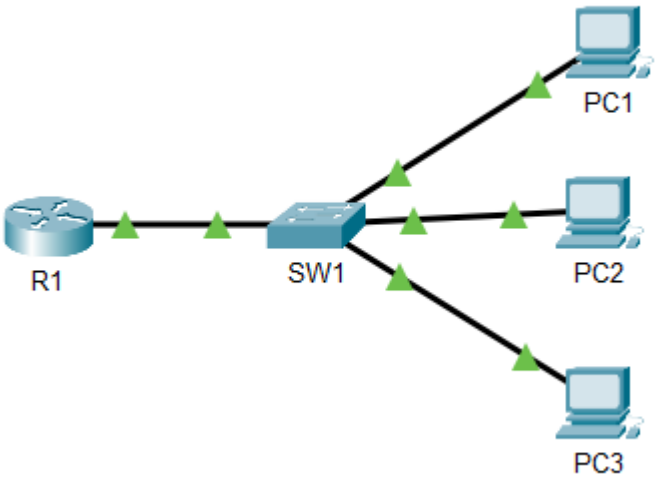
1. `running-config` : The current, active configuration file on the device. As you enter commands in the CLI, you edit the active configuration. Can be viewed using `show running-config` while in **Privileged EXEC Mode** (or `do show running-config` while in **Global Config Mode**). The file is stored in the **RAM** and is lost upon reloading/restarting.
2. `startup-config` : Stored in the **NVRAM**, it will be loaded upon starting the device. Viewed using `show startup-config`

To save the running configuration so that it persists, we can either use (while in **Privileged EXEC Mode**):

- `write`
- `write memory`
- `copy running-config startup-config`

Packet Tracer Lab

Configuring a router's security setting.



Tasks:

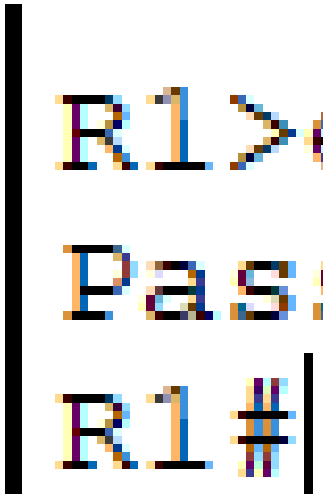
1. Change the hostnames of the router and switch to the appropriate names (R1, SW1)

```
Router(config)#hostname R1
R1(config)#
Switch(config)#hostname SW1
SW1(config)#
```

2. Configure an unencrypted enable password of 'CCNA' on both devices.

```
R1(config)#enable password CCNA
R1(config)#
```

3. Exit back to user EXEC mode and test the password.



Note: Password typed aren't visible (but it's there).

4. View the password in the running configuration.

```
R1#show running-config
Building configuration...

Current configuration : 711 bytes
!
version 15.1
no service timestamps log datetime msec
no service timestamps debug datetime msec
no service password-encryption
!
hostname R1
!
!
!
enable password CCNA
!
!
!
!
!
!
ip cef
no ipv6 cef
 --More--
```

5. Ensure that the current password, and all future passwords, are encrypted.

```
R1(config)#service password-encryption
```

6. View the password in the running configuration.

```
R1#show running-config
Building configuration...

Current configuration : 716 bytes
!
version 15.1
no service timestamps log datetime msec
no service timestamps debug datetime msec
service password-encryption
!
hostname R1
!
!
!
enable password 7 08026F6028
```

Note: `7` is Cisco's encryption algorithm (but weak)

- Removing/undoing encryption with `no` command (Will apply on next password being set):

```
R1(config)#no service password-encryption
```

7. Configure a more secure, encrypted enable password of 'Cisco' on both devices

```
R1(config)#enable secret Cisco
```

8. Exit back to user EXEC mode and then return to privileged EXEC mode.
   (a) Which password do you have to use?
   - Password is `Cisco`

```
R1#show running-config
Building configuration...

Current configuration : 763 bytes
!
version 15.1
no service timestamps log datetime msec
no service timestamps debug datetime msec
service password-encryption
!
hostname R1
!
!
!
enable secret 5 $1$mERr$YlCkLMcTYWwkF1Ccndtll.
enable password 7 08026F6028
```

9. View the passwords in the running configuration.
   (a) What encryption type number is used for the encrypted `enable password` ? `7` (weak)
   (b) What encryption type number is used for the encrypted `enable secret` ? `5` (Stronger)
10. Save the running configuration to the startup configuration

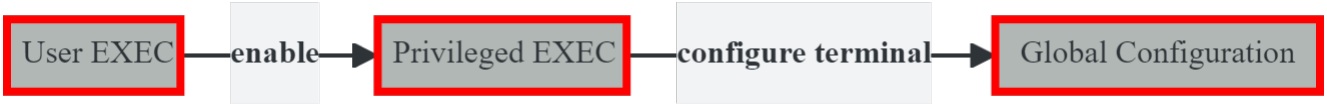
```
R1#copy running-config startup-config
Destination filename [startup-config]?
Building configuration...
[OK]
```

Summary:

Connecting to the device:

1. Connect to a Cisco's device using the device's console port
2. **Rollover Cable** with a **DB-9** Connector

Mode of Operations:



Commands:

1. **User EXEC Mode**:
   (a) `enable` : Enter **Privileged EXEC Mode**
2. **Privileged EXEC Mode**:
   (a) `configure terminal` : Enter **Global Configuration Mode**
   (b) `show ...` (eg. `show running-config` , `show startup-config`): View device's configuration detail.
   (c) `write` , `write memory` , `copy running-config startup-config` : Save the current configuration to startup configuration when device is booted.
3. **Global Configuration Mode**:
   (a) `do ...` (eg. `do show running-config`): Execute **Privileged EXEC Mode** commands while in this mode.
   (b) `hostname ...` (eg. `hostname R1`): Set device's hostname.
   (c) `enable password ...` (eg. `enable password CCNA`):
      i. Set password for entering **Privileged EXEC Mode**

ii. Password will be in plaintext in config files.

(d) `service password-encryption` : Type- `7` encrypt the once-plaintext password shown in config files.

(e) `no ...` (eg. `no service password-encryption`): undo/cancel commands. Will not decrypt currently encrypted password.

(f) `enable secret ...` (eg. `enable secret Cisco`)

i. Set password for entering **Privileged EXEC Mode**

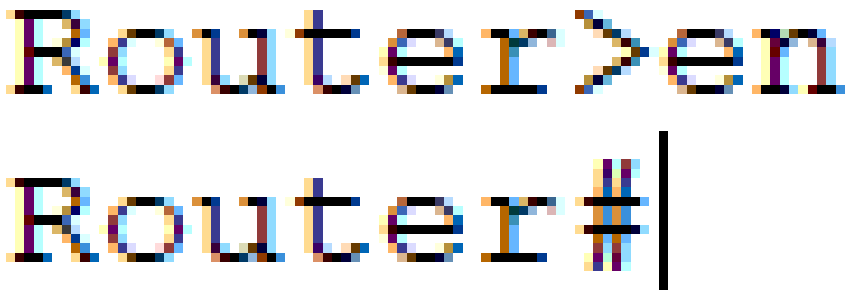
ii. Password automatically encrypted (Type- `5`) when viewing config files.

iii. Make `enable password ...` unusable.

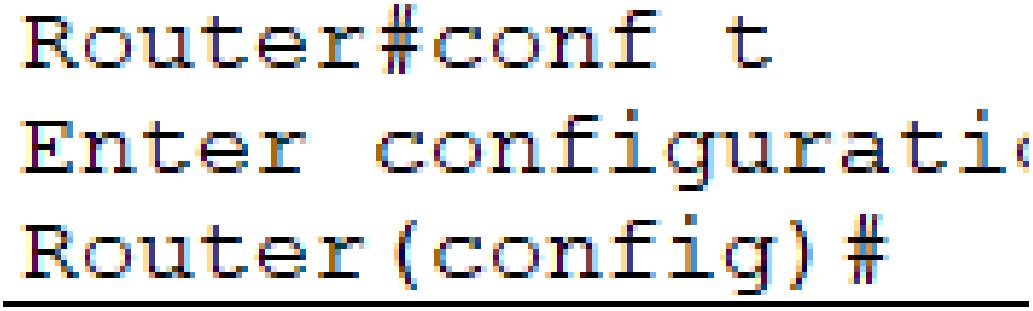
Shortcuts:

Appending `?` to partially written commands show possible autocomplete (eg. `en?` displays `enable` in **User EXEC**). Thus, we can abbreviate commands like:

- `enable` :



- `configure terminal` :



- `show running-config` :