Basic Router CLI Configuration

This chapter contains the following sections:

IR1101 Interface Naming

The supported hardware interfaces and their naming conventions are in the following table:

Hardware Interface	Naming Convention
Gigabit Ethernet combo port	gigabitethernet 0/0/0
Fast Ethernet ports	fastethernet0/0/1-0/0/4
Cellular Interface	cellular 0/1/0 and cellular 0/1/1
Asynchronous Serial Interface	async 0/2/0
USB	usbflash0:
mSATA	msata
IR1101 Base Unit Alarm input	alarm contact 0

Interface names for the different expansion modules are found in the following chapters:

- IRM-1100-4A2T Expansion Module

- IRM-1101 Expansion Module

Basic Configuration

The basic configuration is a result of the entries you made during the initial configuration dialog. This means the router has at least one interface set with an IP address to be reachable, either through WebUI or to allow the PnP process to work. Use the **show running-config** command to view the initial configuration, as shown in the following example:

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

Current configuration : 8079 bytes
!
! Last configuration change at 17:33:19 GMT Tue Jun 25 2019
!
version 16.12
service timestamps debug datetime msec localtime show-timezone
service timestamps log datetime msec localtime show-timezone
service internal
service call-home
platform qfp utilization monitor load 80
no platform punt-keepalive disable-kernel-core
!
hostname IR1101
!
boot-start-marker
boot-end-marker
!
!
no aaa new-model
clock timezone GMT 0 0
call-home
 ! If contact email address in call-home is configured as sch-smart-licensing@cisco.com
 ! the email address configured in Cisco Smart License Portal will be used as contact email
 address to send SCH notifications.
 contact-email-addr sch-smart-licensing@cisco.com
 profile "CiscoTAC-1"
  active
  destination transport-method http
  no destination transport-method email
!
!
ip name-server 171.70.168.183 198.224.173.135 8.8.8.8
no ip domain lookup
ip domain name cisco.com
!
login on-success log
ipv6 unicast-routing
!
chat-script lte "" "AT!CALL" TIMEOUT 20 "OK"
chat-script hspa-R7 "" "AT!SCACT=1,1" TIMEOUT 60 "OK"
!
crypto pki trustpoint SLA-TrustPoint
 enrollment pkcs12
 revocation-check crl
!
crypto pki trustpoint TP-self-signed-756885843
 enrollment selfsigned
 subject-name cn=IOS-Self-Signed-Certificate-756885843
```

```
 revocation-check none
 rsakeypair TP-self-signed-756885843
!
!
crypto pki certificate chain SLA-TrustPoint
 certificate ca 01
  30820321 30820209 A0030201 02020101 300D0609 2A864886 F70D0101 0B050030
  32310E30 0C060355 040A1305 43697363 6F312030 1E060355 04031317 43697363
  6F204C69 63656E73 696E6720 526F6F74 20434130 1E170D31 33303533 30313934
  3834375A 170D3338 30353330 31393438 34375A30 32310E30 0C060355 040A1305
  43697363 6F312030 1E060355 04031317 43697363 6F204C69 63656E73 696E6720
  526F6F74 20434130 82012230 0D06092A 864886F7 0D010101 05000382 010F0030
  82010A02 82010100 A6BCBD96 131E05F7 145EA72C 2CD686E6 17222EA1 F1EFF64D
  CBB4C798 212AA147 C655D8D7 9471380D 8711441E 1AAF071A 9CAE6388 8A38E520
  1C394D78 462EF239 C659F715 B98C0A59 5BBB5CBD 0CFEBEA3 700A8BF7 D8F256EE
  4AA4E80D DB6FD1C9 60B1FD18 FFC69C96 6FA68957 A2617DE7 104FDC5F EA2956AC
  7390A3EB 2B5436AD C847A2C5 DAB553EB 69A9A535 58E9F3E3 C0BD23CF 58BD7188
  68E69491 20F320E7 948E71D7 AE3BCC84 F10684C7 4BC8E00F 539BA42B 42C68BB7
  C7479096 B4CB2D62 EA2F505D C7B062A4 6811D95B E8250FC4 5D5D5FB8 8F27D191
  C55F0D76 61F9A4CD 3D992327 A8BB03BD 4E6D7069 7CBADF8B DF5F4368 95135E44
  DFC7C6CF 04DD7FD1 02030100 01A34230 40300E06 03551D0F 0101FF04 04030201
  06300F06 03551D13 0101FF04 05300301 01FF301D 0603551D 0E041604 1449DC85
  4B3D31E5 1B3E6A17 606AF333 3D3B4C73 E8300D06 092A8648 86F70D01 010B0500
  03820101 00507F24 D3932A66 86025D9F E838AE5C 6D4DF6B0 49631C78 240DA905
  604EDCDE FF4FED2B 77FC460E CD636FDB DD44681E 3A5673AB 9093D3B1 6C9E3D8B
  D98987BF E40CBD9E 1AECA0C2 2189BB5C 8FA85686 CD98B646 5575B146 8DFC66A8
  467A3DF4 4D565700 6ADF0F0D CF835015 3C04FF7C 21E878AC 11BA9CD2 55A9232C
  7CA7B7E6 C1AF74F6 152E99B7 B1FCF9BB E973DE7F 5BDDEB86 C71E3B49 1765308B
  5FB0DA06 B92AFE7F 494E8A9E 07B85737 F3A58BE1 1A48A229 C37C1E69 39F08678
  80DDCD16 D6BACECA EEBC7CF9 8428787B 35202CDC 60E4616A B623CDBD 230E3AFB
  418616A9 4093E049 4D10AB75 27E86F73 932E35B5 8862FDAE 0275156F 719BB2F0
  D697DF7F 28
   quit
crypto pki certificate chain TP-self-signed-756885843
 certificate self-signed 01
  3082032E 30820216 A0030201 02020101 300D0609 2A864886 F70D0101 05050030
  30312E30 2C060355 04031325 494F532D 53656C66 2D536967 6E65642D 43657274
  69666963 6174652D 37353638 38353834 33301E17 0D313930 35333130 30303530
  385A170D 33303031 30313030 30303030 5A303031 2E302C06 03550403 1325494F
  532D5365 6C662D53 69676E65 642D4365 72746966 69636174 652D3735 36383835
  38343330 82012230 0D06092A 864886F7 0D010101 05000382 010F0030 82010A02
  82010100 D2F61742 3B651909 95856431 9BC2CCB7 D4B04861 DD6E0924 4C3E6A51
  8BF2ABD9 5C3A597D 2EE0112C ECA615AA D0297F9E 071B6B5D 9B831332 021E61F4
  2352EEC9 EE70742E 46EFBAFC A03744D8 A22E4DA3 AAF919CC 0A7929A7 3BDB3B17
  C04DA5B9 028DD3EC 992493A6 EA864ED6 354CB3F4 094D3EBF 5307CAA3 192B5759
  E458712D 841A43CD 709D4D9E 72A9DE3E F935A688 59B6F278 65B59EE0 6B72469E
  7B97582A 64E511A6 D81735FF 117CE399 4C2A2973 F5FD407D BCEB62A6 FD7C6B08
  882E0749 ACE5BD44 32634790 3607ADEA 9F319343 4CA76B0D B1DE6A1C AD144548
  E38119E2 8B34F7AC 090C0450 03166B42 8C7C9EA7 5132687F E1F7BF6E B065CD4E
  889F02BB 02030100 01A35330 51300F06 03551D13 0101FF04 05300301 01FF301F
  0603551D 23041830 16801405 77954127 36509205 7025CF4E 84B5D4A2 A3D53730
  1D060355 1D0E0416 04140577 95412736 50920570 25CF4E84 B5D4A2A3 D537300D
  06092A86 4886F70D 01010505 00038201 01004147 49C6A0A9 56F5BD4D 4892AEE0
  22955E06 AF192FA6 868D5556 959ACF05 398F3907 DFE3148B 0E2CFC12 20BEEA05
  DC23E8D7 A47DB4AE D6CB6665 BCAE7F39 24D010F0 DB8F0E70 5E7C3F73 25AB1783
  1346D540 47BB7E89 2BB1BE4D 16990318 A4612CC5 C7CC9376 7DF1A1F4 C09C0051
  4D950D99 3CC0C65B 0A98859A 3B81E324 BAB34EDF 64CA8C38 184DC796 47DDD9DD
  F71F8D5E D3B7A962 3D0FDE44 012AC034 D0E7F75A DB1BF12A CF23E2F5 6A4FDA14
  A588DCDA 8272CE33 36ABC57A BFF52980 5FFC7C34 4D4307BB AC0C0F18 AA783B9D
  27C61E89 0EC1C6AA 6AB3F73B EF8450FD 782DFC63 038F6A27 456CA32B D3FEDB97
  C8064523 EBB93FF5 8B98B546 44F853E9 0E04
   quit
!
license udi pid IR1101-K9 sn FCW222700KS
```

```
diagnostic bootup level minimal
!
spanning-tree extend system-id
memory free low-watermark processor 50357
file prompt quiet
!
!
username cisco privilege 15 password 0 cisco
username lab password 0 lab123
!
redundancy
!
!
controller Cellular 0/1/0
 no lte firmware auto-sim
 lte modem link-recovery disable
!
controller Cellular 0/3/0
!
vlan internal allocation policy ascending
!
!
interface GigabitEthernet0/0/0
 no ip address
 shutdown
!
interface FastEthernet0/0/1
 switchport access vlan 192
 switchport mode access
!
interface FastEthernet0/0/2
 switchport access vlan 172
 switchport mode access
!
interface FastEthernet0/0/3
 switchport access vlan 172
!
interface FastEthernet0/0/4
 switchport mode access
!
interface GigabitEthernet0/0/5
!
interface Cellular0/1/0
 ip address negotiated
 load-interval 30
 dialer in-band
 dialer idle-timeout 0
 dialer watch-group 1
 ipv6 enable
 pulse-time 1
 ip virtual-reassembly
!
interface Cellular0/1/1
 no ip address
 shutdown
!
interface Cellular0/3/0
 ip address negotiated
 dialer in-band
 dialer idle-timeout 0
 dialer watch-group 2
 ipv6 enable
 pulse-time 1
 ip virtual-reassembly
```

```
!
interface Cellular0/3/1
 no ip address
 shutdown
!
interface Vlan1
 ip address 192.168.10.15 255.255.255.0
!
interface Vlan172
 ip address 172.27.167.121 255.255.255.128
!
interface Vlan175
 ip address 175.1.1.1 255.255.255.0
!
interface Async0/2/0
 no ip address
 encapsulation scada
!
ip default-gateway 172.27.167.1
ip forward-protocol nd
!
ip http server
ip http authentication local
ip http secure-server
ip route 0.0.0.0 0.0.0.0 172.27.167.1
ip route 0.0.0.0 0.0.0.0 Cellular0/1/0
ip route 0.0.0.0 0.0.0.0 Cellular0/3/0 253
ip route 8.8.4.0 255.255.255.0 Cellular0/3/0
ip route 171.70.0.0 255.255.0.0 172.27.167.1
ip route 192.1.1.0 255.255.255.0 Cellular0/1/0
ip route 192.168.193.0 255.255.255.0 192.168.10.1
!
!
ip access-list standard 1
 10 permit any
dialer watch-list 1 ip 5.6.7.8 255.255.255.255
dialer watch-list 1 delay route-check initial 60
dialer watch-list 1 delay connect 1
dialer watch-list 2 ip 5.6.7.8 255.255.255.255
dialer watch-list 2 delay route-check initial 60
dialer watch-list 2 delay connect 1
dialer-list 1 protocol ip permit
dialer-list 1 protocol ipv6 permit
ipv6 route ::/0 Cellular0/1/0
!
!
snmp-server community public RO
snmp-server community private RW
snmp-server host 171.70.127.43 version 2c public
snmp-server host 172.27.167.220 version 2c public
snmp-server manager
!
control-plane
!
line con 0
 exec-timeout 0 0
 stopbits 1
 speed 115200
line 0/0/0
line 0/2/0
line vty 0 4
 exec-timeout 0 0
 password cisco
 login
```

```
 transport input none
!
!
end

IR1101#
```

Configuring Global Parameters

To configure global parameters for your router, follow these steps.

Procedure

	Command or Action	Purpose

Step 1	**configure terminal**
	Example:
	`Router> enable`
	`Router# configure terminal`
	`Router(config)#`
		Enters global configuration mode when using the console port.
		Use the following to connect to the router with a remote terminal:
		`telnet router-name or address`
		`Login: login-id`
		`Password: *********`
		`Router> enable`

Step 2	**hostname** *name* **Example:** `Router(config)# hostname Router`	Specifies the name for the router.
Step 3	**enable password** *password* **Example:** `Router(config)# enable password cr1ny5ho`	Specifies a password to prevent unauthorized access to the router. **Note** In this form of the command, password is not encrypted. To encrypt the password use enable secret password as noted in the previously mentioned Device Hardening Guide.

Configuring the Gigabit Ethernet Interface

The default configuration for the Gigabit Ethernet Interface (GI0/0/0) on the IR1101 is Layer 3 (L3). It is possible to configure the interface as a Layer 2 (L2) interface. The Gigabit Ethernet Interface on the IR1101 is a combo port, which means it is a RJ45+SFP connector.

The IRM-1100-SPMI Expansion Module also has an SFP port. The Gigabit Ethernet Interface (GI0/0/5) on the IRM-1100-SPMI is Layer 2 (L2) only. This means you can assign this port to any vlan (switchport acc vlan #) and use the SVI interface. You cannot assign an ip address directly under this port.

The correct connector must be selected, refer to the Cisco Catalyst IR1101 Rugged Series Router Hardware Installation Guide.

To manually define the Gigabit Ethernet interface, follow these steps, beginning from global configuration mode.

Procedure

	Command or Action	Purpose
Step 1	**interface GigabitEthernet** *slot/bay/port* **Example:** `Router(config)# interface GigabitEthernet 0/0/0`	Enters the configuration mode for an interface on the router.
Step 2	**ip address** *ip-address mask* **Example:** `Router(config-if)# ip address 192.168.12.2 255.255.255.0`	Sets the IP address and subnet mask for the specified interface. Use this Step if you are configuring an IPv4 address.
Step 3	**ipv6 address** *ipv6-address/prefix* **Example:** `Router(config-if)# ipv6 address 2001.db8::ffff:1/128`	Sets the IPv6 address and prefix for the specified interface. Use this step instead of Step 2, if you are configuring an IPv6 address. IPv6 unicast-routing needs to be set-up as well, see further information in the **IPv6 Addressing and Basic Connectivity Configuration Guide** located here: https://www.cisco.com/c/en/us/td/docs/ios-xml/ios/ipv6_basic/configuration/xe-16-10/ip6b-xe-16-10-book/read-me-first.html
Step 4	**ipv6 unicast-routing** **Example:** `Router (config)# ipv6 unicast-routing`	Enables forwarding of IPv6 unicast data packets.
Step 5	**no shutdown** **Example:** `Router(config-if)# no shutdown`	Enables the interface and changes its state from administratively down to administratively up.
Step 6	**exit** **Example:** `Router(config-if)# exit`	Exits the configuration mode of interface and returns to the global configuration mode.

Support for sub-interface on GigabitEthernet0/0/0

Cisco IOS-XE release 16.11.1 and beyond supports sub-interfaces and dot1q configuration on the g0/0/0 interface. For example:

```
Router(config)#interface g0/0/0 ?
  <1-4294967295>  GigabitEthernet interface number
Router(config-subif)#encapsulation ?
  dot1Q           IEEE 802.1Q Virtual LAN
```

Configuring a Loopback Interface

Before you begin

The loopback interface acts as a placeholder for the static IP address and provides default routing information.

To configure a loopback interface, follow these steps.

Procedure

	Command or Action	Purpose
Step 1	**interface** *type number* **Example:** `Router(config)# interface Loopback 0`	Enters configuration mode on the loopback interface.
Step 2	(Option 1) **ip address** *ip-address mask* **Example:** `Router(config-if)# ip address 10.108.1.1 255.255.255.0`	Sets the IP address and subnet mask on the loopback interface. (If you are configuring an IPv6 address, use the **ipv6 address** *ipv6-address/prefix* command described below.
Step 3	(Option 2) **ipv6 address** *ipv6-address/prefix* **Example:** `Router(config-if)# ipv6 address 2001:db8::ffff:1/128`	Sets the IPv6 address and prefix on the loopback interface.
Step 4	**exit** **Example:** `Router(config-if)# exit`	Exits configuration mode for the loopback interface and returns to global configuration mode.

Example

Verifying Loopback Interface Configuration

Enter the **show interface loopback** command. You should see an output similar to the following example:

```
Router# show interface loopback 0
Loopback0 is up, line protocol is up
  Hardware is Loopback
  Internet address is 192.0.2.0/16
  MTU 1514 bytes, BW 8000000 Kbit, DLY 5000 usec,
     reliability 255/255, txload 1/255, rxload 1/255
  Encapsulation LOOPBACK, loopback not set
  Last input never, output never, output hang never
  Last clearing of "show interface" counters never
  Queueing strategy: fifo
  Output queue 0/0, 0 drops; input queue 0/75, 0 drops
  5 minute input rate 0 bits/sec, 0 packets/sec
  5 minute output rate 0 bits/sec, 0 packets/sec
     0 packets input, 0 bytes, 0 no buffer
     Received 0 broadcasts, 0 runts, 0 giants, 0 throttles
     0 input errors, 0 CRC, 0 frame, 0 overrun, 0 ignored, 0 abort
     0 packets output, 0 bytes, 0 underruns
     0 output errors, 0 collisions, 0 interface resets
     0 output buffer failures, 0 output buffers swapped out
```

Alternatively, use the **ping** command to verify the loopback interface, as shown in the following example:

```
Router# ping 192.0.2.0
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.0.2.0, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 1/2/4 ms
```

Enabling Cisco Discovery Protocol

Cisco Discovery Protocol (CDP) is enabled by default on the router. It may be disabled if needed for security purposes.

For more information on using CDP, see Cisco Discovery Protocol Configuration Guide, Cisco IOS XE Release 3S.

Configuring Command-Line Access

To configure parameters to control access to the router, follow these steps.

Note Transport input must be set as explained in the previous Telnet and SSH sections of the guide.

Procedure

	Command or Action	Purpose
Step 1	**line** [**aux** | **console** | **tty** | **vty**] *line-number* **Example:** `Router(config)# line console 0`	Enters line configuration mode, and specifies the type of line. The example provided here specifies a console terminal for access.
Step 2	**password** *password* **Example:** `Router(config-line)# password 5dr4Hepw3`	Specifies a unique password for the console terminal line.
Step 3	**login** **Example:** `Router(config-line)# login`	Enables password checking at terminal session login.

Step 4	**exec-timeout** *minutes* [*seconds*]
	Example:
	`Router(config-line)# exec-timeout 5 30`
	`Router(config-line)#`
		Sets the interval during which the EXEC command interpreter waits until user input is detected. The default is 10 minutes. Optionally, adds seconds to the interval value.
		The example provided here shows a timeout of 5 minutes and 30 seconds. Entering a timeout of **0 0** specifies never to time out.

Step 5	**exit** **Example:** `Router(config-line)# exit`	Exits line configuration mode to re-enter global configuration mode.

Step 6	**line** [**aux** | **console** | **tty** | **vty**] *line-number*
	Example:
	`Router(config)# line vty 0 4`
	`Router(config-line)#`
		Specifies a virtual terminal for remote console access.

Step 7	**password** *password* **Example:** `Router(config-line)# password aldf2ad1`	Specifies a unique password for the virtual terminal line.
Step 8	**login** **Example:** `Router(config-line)# login`	Enables password checking at the virtual terminal session login.

	Command or Action	Purpose
Step 9	**end** **Example:** `Router(config-line)# end`	Exits line configuration mode, and returns to privileged EXEC mode.

Example

The following configuration shows the command-line access commands. Note that transport input none is the default, but if SSH is enabled this must be set to ssh.

You do not have to input the commands marked **default**. These commands appear automatically in the configuration file that is generated when you use the **show running-config** command.

```
!
line console 0
exec-timeout 10 0
password 4youreyesonly
login
transport input none (default)
stopbits 1 (default)
line vty 0 4
password secret
login
!
```

Configuring Static Routes

Static routes provide fixed routing paths through the network. They are manually configured on the router. If the network topology changes, the static route must be updated with a new route. Static routes are private routes unless they are redistributed by a routing protocol.

To configure static routes, follow these steps.

Procedure

	Command or Action	Purpose
Step 1	(Option 1) **ip route** *prefix mask* {*ip-address* | *interface-type interface-number* [*ip-address*]} **Example:** `Router(config)# ip route 192.10.2.3 255.255.0.0 10.10.10.2`	Specifies a static route for the IP packets. (If you are configuring an IPv6 address, use the **ipv6 route** command described below.)
Step 2	(Option 2) **ipv6 route** *prefix/mask* {*ipv6-address* | *interface-type interface-number* [*ipv6-address*]} **Example:**	Specifies a static route for the IP packets. See additional information for IPv6 here: https://www.cisco.com/c/en/us/td/docs/ios-xml/ios/ipv6_basic/configuration/xe-16-10/ip6b-xe-16-10-book/read-me-first.html

	Command or Action	Purpose
	`Router(config)# ipv6 route 2001:db8:2::/64 2001:db8:3::0`	
Step 3	**end** **Example:** `Router(config)# end`	Exits global configuration mode and enters privileged EXEC mode.

In the following configuration example, the static route sends out all IP packets with a destination IP address of 192.168.1.0 and a subnet mask of 255.255.255.0 on the Gigabit Ethernet interface to another device with an IP address of 10.10.10.2. Specifically, the packets are sent to the configured PVC.

You do not have to enter the command marked **default**. This command appears automatically in the configuration file generated when you use the **running-config** command.

```
!
ip classless (default)
ip route 2001:db8:2::/64 2001:db8:3::0
```

Verifying Configuration

To verify that you have configured static routing correctly, enter the **show ip route** command (or **show ipv6 route** command) and look for static routes marked with the letter S.

When you use an IPv4 address, you should see verification output similar to the following:

```
Router# show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

     10.0.0.0/24 is subnetted, 1 subnets
C       10.108.1.0 is directly connected, Loopback0
S*   0.0.0.0/0 is directly connected, FastEthernet0
```

When you use an IPv6 address, you should see verification output similar to the following:

```
Router# show ipv6 route
IPv6 Routing Table - default - 5 entries
Codes: C - Connected, L - Local, S - Static, U - Per-user Static route
       B - BGP, R - RIP, H - NHRP, I1 - ISIS L1
       I2 - ISIS L2, IA - ISIS interarea, IS - ISIS summary, D - EIGRP
       EX - EIGRP external, ND - ND Default, NDp - ND Prefix, DCE -
Destination
       NDr - Redirect, O - OSPF Intra, OI - OSPF Inter, OE1 - OSPF ext 1
       OE2 - OSPF ext 2, ON1 - OSPF NSSA ext 1, ON2 - OSPF NSSA ext 2
       ls - LISP site, ld - LISP dyn-EID, a - Application
```

```
C   2001:DB8:3::/64 [0/0]
      via GigabitEthernet0/0/2, directly connected
S   2001:DB8:2::/64 [1/0]
      via 2001:DB8:3::1
```

Configuring Dynamic Routes

In dynamic routing, the network protocol adjusts the path automatically, based on network traffic or topology. Changes in dynamic routes are shared with other routers in the network.

All of the Cisco IOS-XE configuration guides can be found here: https://www.cisco.com/c/en/us/support/ios-nx-os-software/ios-xe-17/series.html#Configuration.

Configuring Routing Information Protocol

To configure the RIP on a router, follow these steps.

Procedure

	Command or Action	Purpose
Step 1	**router rip** **Example:** `Router(config)# router rip`	Enters router configuration mode, and enables RIP on the router.
Step 2	**version** {**1** | **2**} **Example:** `Router(config-router)# version 2`	Specifies use of RIP version 1 or 2.

Step 3	**network** *ip-address*
	Example:
	`Router(config-router)# network 192.168.1.1`
	`Router(config-router)# network 10.10.7.1`
		Specifies a list of networks on which RIP is to be applied, using the address of the network of each directly connected network.

Step 4	**no auto-summary** **Example:** `Router(config-router)# no auto-summary`	Disables automatic summarization of subnet routes into network-level routes. This allows subprefix routing information to pass across classful network boundaries.
Step 5	**end** **Example:** `Router(config-router)# end`	Exits router configuration mode, and enters privileged EXEC mode.

Example

Verifying Configuration

To verify that you have configured RIP correctly, enter the **show ip route** command and look for RIP routes marked with the letter R. You should see an output similar to the one shown in the following example:

```
Router# show ip route
Codes: C - connected, S - static, R - RIP, M - mobile, B - BGP
       D - EIGRP, EX - EIGRP external, O - OSPF, IA - OSPF inter area
       N1 - OSPF NSSA external type 1, N2 - OSPF NSSA external type 2
       E1 - OSPF external type 1, E2 - OSPF external type 2
       i - IS-IS, su - IS-IS summary, L1 - IS-IS level-1, L2 - IS-IS level-2
       ia - IS-IS inter area, * - candidate default, U - per-user static route
       o - ODR, P - periodic downloaded static route

Gateway of last resort is not set

     10.0.0.0/24 is subnetted, 1 subnets
C       10.108.1.0 is directly connected, Loopback0
R    3.0.0.0/8 [120/1] via 2.2.2.1, 00:00:02, Ethernet0/0/0
```

Configuring Enhanced Interior Gateway Routing Protocol

The Enhanced Interior Gateway Routing Protocol (EIGRP) is an enhanced version of the Interior Gateway Routing Protocol (IGRP) developed by Cisco. The convergence properties and the operating efficiency of EIGRP have improved substantially over IGRP, and IGRP is now obsolete.

The convergence technology of EIGRP is based on an algorithm called the Diffusing Update Algorithm (DUAL). The algorithm guarantees loop-free operation at every instant throughout a route computation and allows all devices involved in a topology change to synchronize. Devices that are not affected by topology changes are not involved in recomputations

Details on configuring Enhanced Interior Gateway Routing Protocol (EIGRP), are found in the following guide: https://www.cisco.com/c/en/us/td/docs/routers/ios/config/17-x/ip-routing/b-ip-routing.html

Configuring the Serial Interface

This section describes configuring serial interface management.

The IR1101 supports asynchronous serial interface protocols used for SCADA, Raw Socket, or reverse Telnet. It has a single serial interface, designated async 0/2/0. The serial interface is DTE only.

Specifying an Asynchronous Serial Interface

To specify an asynchronous serial interface and enter interface configuration mode, use one of the following commands in global configuration mode.

Command or Action	Purpose
Router(config)# interface async 0/2/0	Enters interface configuration mode.

Specifying Asynchronous Serial Encapsulation

The asynchronous serial interfaces support the following serial encapsulation methods:

- Raw-TCP
- Raw-UDP
- SCADA
- Encapsulation Relay

Command or Action	Purpose
Router(config-if)# encapsulation {raw-tcp | raw-udp | scada}	Configures asynchronous serial encapsulation.

Encapsulation methods are set according to the type of protocol or application you configure in the Cisco IOS software.

The remaining encapsulation methods are defined in their respective books and chapters describing the protocols or applications.

Configuring the Serial Port

To configure the serial port perform the steps in the following example:

```
IR1101#sh run int async 0/2/0
Building configuration...
Current configuration : 62 bytes
!
interface Async0/2/0
no ip address
encapsulation raw-tcp
end
IR1101#show line
   Tty Line Typ     Tx/Rx    A Modem  Roty AccO AccI  Uses  Noise Overruns  Int
*     0    0 CTY              -    -      -    -    -     0      0    0/0      -
  0/2/0   50 TTY   9600/9600  -    -      -    -    -     0      0    0/0      -
     74   74 VTY              -    -      -    -    -     3      0    0/0      -
     75   75 VTY              -    -      -    -    -     0      0    0/0      -
     76   76 VTY              -    -      -    -    -     0      0    0/0      -
     77   77 VTY              -    -      -    -    -     0      0    0/0      -
     78   78 VTY              -    -      -    -    -     0      0    0/0      -
```

Line(s) not in async mode -or- with no hardware support:

1-49, 51-73, 79-726