# I Commands

## identification

**Description:** Specifies the assignment ID of an L2TP or L2F tunnel in Domain Map Tunnel Configuration mode. The **no** version removes the assignment ID from the tunnel.

**Syntax:** identification *serverId*

no identification

- *serverId* – L2TP or L2F tunnel assignment ID up to 32 characters

**Mode(s):** Domain Map Tunnel Configuration

## igmp disable

**Description:** Disables IGMP on a virtual router.The **no** version reenables IGMP on a virtual router.

**Syntax:** [ no ] igmp disable

**Mode(s):** Router Configuration

## igmp promiscuous

**Description:** Allows all IGMP interfaces on the router to accept IGMP reports from hosts on any subnet. The **no** version allows IGMP interfaces on the router to accept IGMP reports only from hosts on their associated subnets.

**Syntax:** [ no ] igmp promiscuous

**Mode(s):** Router Configuration

## ignore dcd

**Description:** Prevents the system from using the DTD signal when determining whether the interface is working, if an interface is configured as a V.35 DTE. The **no** version allows the system to use the DTD signal when determining whether the interface is working.

**Syntax:** [ no ] ignore dcd

**Mode(s):** Interface Configuration

## ignore link-state-signals

**Description:** Prevents the system from using any link state signal when determining whether a V.35/X.21 interface is working. The **no** version allows the system to use the link state signals when determining whether the interface is working.

**Syntax:** [ no ] ignore link-state-signals

**Mode(s):** Interface Configuration

## ignore-lsp-errors

**Description:** Allows the router to ignore IS-IS link state packets that are received with internal checksum errors rather than purging the link state packets. The **no** version disables this function.

**Syntax:** [ no ] ignore-lsp-errors

**Mode(s):** Router Configuration

## import map

**Description:** Associates a route map with a VRF to filter routes received by the VRF. The **no** version disables the application of the route map to imported routes.

**Syntax:** [ no ] import map *routeMap*

- *routeMap* – name of a route map

**Mode(s):** VRF Configuration

## index

**Description:** Sets a next hop at a particular index in the MPLS explicit path. The **no** version removes the next hop from the index.

**Syntax:** index *indexNumber* next-address *ipAddress* [ mask *ipMask* ] [ loose ]

no index *indexNumber*

- *index* – number of a node in an ordered set of abstract nodes, a value ranging from 1–255
- *ipAddress* – address of the next hop
- *ipMask* – [ not currently used] mask for the next adjacent address
- loose – node is not necessarily directly connected (adjacent) to the previous node in the path. If loose is not configured, the configuration defaults to strict. Strict indicates that the node is directly connected to the previous node.

**Mode(s):** Explicit Path Configuration

## interface

**Description:** Configures an interface. The **no** version removes the subinterface or the logical interface.

**Syntax:** [ no ] interface *interfaceType interfaceSpecifier* [ *extension* ]

- *interfaceType* – interface type; see *Interface Types and Specifiers* in *About This Guide*
- *interfaceSpecifier* – particular interface; format varies according to interface type; see *Interface Types and Specifiers* in *About This Guide.*
- *extension* – option that depend on the type of interface

***Note:*** *See the following entries for the syntax of each type of interface.*

**Mode(s):** Global Configuration

## interface atm

**Description:** Configures an ATM interface or subinterface type. The **no** version removes the interface or subinterface.

**Syntax:** [ no ] interface atm *interfaceSpecifier* [ multipoint | point-to-point ]

- *interfaceSpecifier* – particular interface; format varies according to interface type; see *Interface Types and Specifiers* in *About This Guide.*
- multipoint – use to specify an NBMA subinterface
- point-to-point – default; use to specify an ATM interface or subinterface

**Mode(s):** Global Configuration, Interface Configuration

## interface-event-disable

**Description:** Specifies that RIP does not purge the routing table on a RIP interface that has been brought down. The **no** version restores the default condition, wherein RIP does purge the routing table on an interface after a down event.

**Syntax:** [ no ] interface-event-disable

**Mode(s):** Router Configuration

## interface fastEthernet

**Description:** Specifies a Fast Ethernet interface or subinterface or creates a subinterface over a Fast Ethernet interface. The **no** version removes the interface or subinterface.

**Syntax:** [ no ] interface fastEthernet *interfaceSpecifier*

- *interfaceSpecifier* – location of the interface in the format *slot/port* [ .*subinterface1* [ .*subinterface2* ] ]
  - › *slot* – number of the chassis slot in the range 0–13 (ERX-1400 series) and 0–6 (ERX-700 series)
  - › *port* – number of the port; 0 or 1 for a dual-port Fast Ethernet module, 0–7 for an eight-port Fast Ethernet module, 0 for the Fast Ethernet port on the SRP module

  . For a list of interface types and their corresponding specifiers, see *Interface Types and Specifiers* in *About This Guide.*

  The meaning of the subinterface variables depends on the configuration context. You can configure Fast Ethernet interfaces with or without VLANS. If you do not use VLANS, you can use either of two different configuration methods.

  VLANs

  - › *subinterface1* – number of the VLAN subinterface in the range 1–4294967293; no more than 4096 VLAN subinterfaces per Fast Ethernet physical port
  - › *subinterface2* – when using PPPoE, the number of the PPPoE subinterface in the range 1–4294967293; no more than 4094 PPPoE subinterfaces per Fast Ethernet physical port

  No VLANs – New Configuration Method

  - › *subinterface1* – when using PPPoE, the number of the PPPoE subinterface in the range 1–4294967293; no more than 4094 PPPoE subinterfaces per Fast Ethernet physical port
  - › *subinterface2* – not used

  No VLANs – Alternative Configuration Method

  - › *subinterface1* – number of the Fast Ethernet subinterface in the range 1–4294967293; no more than 2 subinterfaces per Fast Ethernet physical port
  - › *subinterface2* – when using PPPoE, the number of the PPPoE subinterface in the range 1–4294967293; no more than 4094 PPPoE subinterfaces per Fast Ethernet physical port

***Note:*** *If you are configuring Fast Ethernet interfaces without VLANs and you are running software release 3.0.0 or higher, we highly recommend that you use the current configuration method. If you are configuring Fast Ethernet*

*interfaces without VLANs and you are running a software release earlier than 3.0.0 or if you are using scripts or macros created with such a release, you must use the alternative configuration method.*

**Mode(s):** Global Configuration

## interface gigabitEthernet

**Description:** Specifies or creates a Gigabit Ethernet interface or a subinterface over a Gigabit Ethernet interface. The **no** version removes the interface or subinterface.

**Syntax:** [ no ] interface gigabitEthernet *interfaceSpecifier*

- *interfaceSpecifier* – the location of the interface in the format *slot/port* [ *.subinterface1* [ *.subinterface2* ] ]

  For a list of interface types and their corresponding specifiers, see *Interface Types and Specifiers* in *About This Guide.*

  The meaning of the subinterface variables depends on the configuration context. You can configure Gigabit Ethernet interfaces with or without VLANS.

  VLANs

  - *subinterface1* – the number of the VLAN subinterface in the range 1–4294967293; no more than 4096 VLAN subinterfaces per Gigabit Ethernet physical port
  - *subinterface2* – when using PPPoE, the number of the PPPoE subinterface in the range 1–4294967293; no more than 4094 PPPoE subinterfaces per Gigabit Ethernet physical port

  No VLANs

  - *subinterface1* – when using PPPoE, the number of the PPPoE subinterface in the range 1–4294967293; no more than 4094 PPPoE subinterfaces per Gigabit Ethernet physical port
  - *subinterface2* – not used

***Note:*** *You can configure only the primary port, 0, on a GE I/O module. The system automatically uses the redundant port if the primary fails.*

**Mode(s):** Global Configuration

## interface hssi

**Description:** Selects a HSSI interface. The **no** version clears the configuration on the HSSI interface.

**Syntax:** [ no ] interface hssi *interfaceSpecifier*

- *interfaceSpecifier* – particular interface; format varies according to interface type; see *Interface Types and Specifiers* in *About This Guide.*

**Mode(s):** Global Configuration

## interface ip

**Description:** Defines a shared IP interface. The **no** version removes the IP interface.

**Syntax:** [ no ] interface ip *interfaceName*

- *interfaceName* – string of up to 15 characters

**Mode(s):** Global Configuration

## interface loopback

**Description:** Defines a loopback interface. The **no** version removes the loopback interface.

**Syntax:** [ no ] interface loopback *interfaceSpecifier*

- *interfaceSpecifier* – particular interface; format varies according to interface type; see *Interface Types and Specifiers* in *About This Guide.*

**Mode(s):** Global Configuration

## interface mlframe-relay

**Description:** Defines an MFR bundle or a subinterface in a bundle. The **no** version removes the bundle or subinterface.

**Syntax:** [ no ] interface mlframe-relay *interfaceSpecifier*

- *interfaceSpecifier* – particular interface; format varies according to interface type; see *Interface Types and Specifiers* in *About This Guide.*

**Mode(s):** Global Configuration

## interface mlppp

**Description:** Creates an MLPPP network interface, also known as the MLPPP bundle. The **no** version deletes the MLPPP bundle.

**Syntax:** [ no ] interface mlppp *interfaceSpecifier*

- *interfaceSpecifier* – particular interface; format varies according to interface type; see *Interface Types and Specifiers* in *About This Guide.*

**Mode(s):** Interface Configuration, Subinterface Configuration

## interface null

**Description:** Defines a null interface, which does not forward traffic. The **no** version removes the null interface.

**Syntax:** [ no ] interface null 0

**Mode(s):** Global Configuration

## interface pos

**Description:** Configures a Packet over SONET interface. The **no** version removes the interface.

**Syntax:** [ no ] interface pos *interfaceSpecifier*

- *interfaceSpecifier* – particular interface; format varies according to interface type; see *Interface Types and Specifiers* in *About This Guide.*

**Mode(s):** Global Configuration

## interface serial

**Description:** Specifies the location of the serial interface on CE1/CT1, CT3, E3/T3 FRAME, cOCx/STMx, and X.21/V.35 modules. The **no** version disables the interface.

**Syntax:** [ no ] interface serial *interfaceSpecifier*

- *interfaceSpecifier* – particular interface; format varies according to interface type; see *Interface Types and Specifiers* in *About This Guide.*

**Mode(s):** Global Configuration

## interface tunnel

**Description:** Creates a tunnel interface for use by DVMRP, GRE, MPLS, or IPSec. You can specify that the tunnel be established in the routing space of a virtual router other than the current VR. If you specify another VR, all tunnel commands apply to the tunnel in that VR. If you do not specify another VR, tunnel commands apply to the current VR. The **no** version removes the tunnel interface.

**Syntax:** [ no ] interface tunnel *interfaceSpecifier* [ transport-virtual-router *vrName* ]

- *interfaceSpecifier* – particular interface; format varies according to interface type; see *Interface Types and Specifiers* in *About This Guide.*
- *vrName* – name of virtual router in which the tunnel will be established

**Mode(s):** Global Configuration

## invert data

**Description:** Enables data stream inversion. Data stream inversion must be turned on by network personnel at the other end of the line. The **no** version disables data stream inversion.

**Syntax:** [ no ] invert data

**Mode(s):** Interface Configuration

## invert txclock

**Description:** Enables inversion of the transmit clock signal, to compensate for differences between the clock and data rate when data is travelling at fast speeds over long cables. Data stream inversion must be turned on by network personnel at the other end of the line. The **no** version disables inversion of the transmit clock signal.

**Syntax:** [ no ] invert txclock

**Mode(s):** Interface Configuration

## ip access-routes

**Description:** Enables the ability to create host access routes on a PPP interface, which is useful for the B-RAS application. It also enables an access route in a profile. The **no** version disables the feature.

**Syntax:** [ no ] ip access-routes

**Mode(s):** Interface Configuration, Subinterface Configuration, Profile Configuration

## ip address

**Description:** Sets a primary or secondary IP address for an interface or subinterface. The **no** version removes an IP address or disables IP processing. You must specify the layer 2 encapsulation before you can set the IP address.

**Syntax:** ip address *ipAddress ipMask* [ secondary ]

no ip address [ *ipAddress ipMask* [ secondary ] ]

- *ipAddress* – IP address in 32-bit dotted decimal format (for example, 192.56.32.2)
- *ipMask* – mask for associated IP subnet
- secondary – specifies that the configured address is a secondary IP address; if omitted, the configured address is the primary IP address

**Mode(s):** Interface Configuration, Subinterface Configuration, Profile Configuration

## ip address-pool

**Description:** Specifies to the system where to get an IP address for the remote user. The **no** version uses the default value, **local**.

**Syntax:** ip address pool { dhcp | local | none }

no ip address pool

- dhcp – enable the use of a DHCP server for address allocations
- local – enable the use of local address pool for address allocations
- none – do not enable an IP address pool

**Mode(s):** Global Configuration

## ip alwaysup

**Description:** Forces the interface to appear as if it is up, regardless of the state of the lower layers. Use this command to reduce route topology changes when the network attached to this link is single-homed. The **no** version makes the interface appear in its current state.

**Syntax:** [ no ] ip alwaysup

**Mode(s):** Interface Configuration, Subinterface Configuration

## ip as-path access-list

**Description:** Defines a BGP-related access list. You can specify an access list filter on both inbound and outbound BGP routes. Each filter is an access list based on regular expressions. If the regular expression matches the representation of the AS path of the route as an ASCII string, then the permit or deny condition applies. The AS path does not contain the local AS number. The **no** version removes a single access list entry if **permit** or **deny** and a *pathExpression* are specified. Otherwise, the entire access list is removed.

**Syntax:** ip as-path access-list *accessListName* { permit | deny } *pathExpression*

no ip as-path access-list *accessListName*

- *accessListName* – name of the access list; a string of up to 32 characters
- permit – permits access for matching conditions
- deny – denies access to matching conditions
- *pathExpression* – the regular expression describing the AS paths to be matched

  Use a sequence of one or more elements, each of which is either an AS number or one of the following punctuation characters:

  | | |
  |---|---|
  | ^ | start of the path |
  | $ | end of the path |
  | { | start of an AS_SET |
  | } | end of an AS_SET |
  | ( | start of an AS_CONFED_SET or AS_CONFED_SEQ |
  | ) | end of an AS_CONFED_SET or AS_CONFED_SEQ |

  Use the following regular expression metacharacters to match individual elements:

  | | |
  |---|---|
  | . | matches any single element |
  | * | matches zero or more occurrences of any element |
  | + | matches one or more occurrences of any element |
  | [ ] | matches any elements enclosed between brackets ([ ]) |
  | – | hyphen; used within brackets to specify a range of AS numbers |
  | ^ | matches any AS number except the ones specified when used as a first item within brackets |
  | _ | underscore; used in non-ERX implementations on either side of a path to specify a literal and disallow substring matching. Allowed but not required in our CLI. |

**Mode(s):** Global Configuration

## ip atm-vc

**Description:** Associates a protocol and address to a specific virtual circuit.

**Syntax:** ip *ipAddress* atm-vc *vcd* broadcast

no ip *ipAddress* atm-vc *vcd*

- *ipAddress* – ip address to be associated with the virtual circuit
- *vcd* – number in the range 1–4294967295; virtual circuit descriptor; an identifier for the VC in other commands
- broadcast – specifies that the circuit should participate in broadcast operations

**Mode(s):** Map List Configuration

## ip bgp-community new-format

**Description:** Specifies that communities must be displayed in *AA*:*NN* format, where *AA* is a number that identifies the autonomous system and *NN* is a number that identifies the community within the autonomous system. The **no** version restores the default display.

**Syntax:** [ no ] ip bgp-community new-format

**Mode(s):** Global Configuration

## ip bgp-confed-as-set new-format

**Description:** Specifies that AS-confed-sets must be displayed within square brackets, [ ], with the ASs delimited by commas. The **no** version restores the default, displaying AS-confed-sets within parentheses, ( ), with the ASs delimited by spaces.

**Syntax:** [ no ] ip bgp-confed-as-set new-format

**Mode(s):** Global Configuration

## ip broadcast-address

**Description:** Defines a broadcast address for an interface. The **no** version restores the default IP broadcast address.

**Syntax:** [ no ] ip broadcast-address [ *ipAddress* ]

- *ipAddress* – broadcast IP address

**Mode(s):** Interface Configuration, Subinterface Configuration

## ip community-list

**Description:** Creates a community list for BGP and controls access to it. The **no** version removes the community list, including all list entries.

**Syntax:** ip community-list *communityLisName* { permit | deny } { *communityNumber* | *asCommunityNumber* | no-export | no-advertise | local-as | internet } [ *communityNumber* | *asCommunityNumber* | no-export | no-advertise | local-as | internet ]*

ip community-list *communityLisName* { permit | deny } *communityExpression*

no ip community-list *communityLisName*

- *communityLisName* – name of a community list; a string of up to 32 characters; identifies one or more permit or deny groups of communities; used for standard community lists
- permit – permits access for a matching condition
- deny – denies access for a matching condition
- *communityNumber* – community number in the range 1–4294967295
- *asCommunityNumber* – community number in the format *AA*:*NN*, where *AA* is a number that identifies the autonomous system and *NN* is a number that identifies the community within the autonomous system.
- no-export – specifies that BGP does not advertise this route outside a BGP confederation boundary
- no-advertise – specifies that BGP does not advertise this route to any peer (internal or external)
- local-as – specifies that BGP does not advertise this route to external peers; sometimes known as the no-export-subconfed community
- internet – specifies the Internet community
- * – indicates that one or more parameters can be repeated multiple times in a list in the command line
- *communityExpression* – a regular expression that matches the community

**Mode(s):** Global Configuration

## ip debounce-time

**Description:** Defines the minimum time an IP interface must be in a given state—for example, up or down—before being reported. The **no** version removes the debounce time.

**Syntax:** ip debounce-time [ vrf *vrfName* ] *period*

no ip debounce-time [ vrf *vrfName* ]

- *vrfName* – name of the VRF; string of 1–32 alphanumeric characters
- *period* – interval in the range 0–60000 seconds

**Mode(s):** Global Configuration

## ip demux-type da-prefix

**Description:** Specifies that a subscriber interface will demultiplex traffic using destination addresses. The **no** version restores the default situation, in which the subscriber interface demultiplexes traffic using source addresses.

**Syntax:** [ no ] ip demux-type da-prefix

**Mode(s):** Interface Configuration

## ip destination-prefix

**Description:** Configures a subscriber interface to demultiplex traffic with the specified destination address. The **no** version removes the association between the subscriber interface and the specified destination address.

**Syntax:** [ no ] ip destination-prefix *ipAddress ipAddressMask*

- *ipAddress* – destination IP address that the system uses to identify packets for this subscriber interface
- *ipAddressMask* – network mask for associated IP subnet

**Mode(s):** Interface Configuration

## ip dhcp-local cable-modem

**Description:** Specifies the IP address of the external DHCP server to which the DHCP local server will relay DHCP messages from cable modems. The **no** version removes the cable modem configuration.

**Syntax:** [no] ip dhcp-local cable-modem dhcp-server *ipAddress*

- *ipAddress* – IP address of the cable modem DHCP server

**Mode(s):** Global Configuration

## ip dhcp-local excluded-address

**Description:** Specifies IP addresses that the DHCP local server should not supply from the default address pool because those addresses are already used by devices on the subnet. The **no** version allows the DHCP local server to supply the specified IP address.

**Syntax:** [ no ] ip dhcp-local excluded-address *ipAddressStart ipAddressStop*

- *ipAddressStart* – single IP address or start of the range of IP addresses that the DHCP local server should not supply
- *ipAddressStop* – end of the range of IP addresses that the DHCP local server should not supply

**Mode(s):** Global Configuration

## ip dhcp-local limit

**Description:** Specifies the maximum number of IP addresses that the DHCP local server can supply to each VPI, VCI, VLAN, or Ethernet subnet. The **no** version restores the default situation, in which there is no limit on the number of token IP addresses that the DHCP local server can supply to each VPI, VCI, VLAN, or Ethernet subnet.

**Syntax:** ip dhcp-local limit { atm | ethernet | vlan | } *leaseNumber*

no ip dhcp-local limit [ atm | ethernet | vlan ]

- atm – specifies the limit for VPIs and VCIs
- ethernet – specifies the limit for Ethernet subnets
- vlan – specifies the limit for VLANs
- *leaseNumber* – maximum number of leases in the range 0–32767

**Mode(s):** Global Configuration

## ip dhcp-local pool

**Description:** Accesses Pool Configuration mode. The **no** version prevents the DHCP local server from supplying IP addresses from the specified pool.

**Syntax:** [ no ] ip dhcp-local pool { *poolName* | default }

- *poolName* – name of the address pool
- default – specifies the default address pool

**Mode(s):** Global Configuration

## ip dhcp-server

**Description:** Adds the IP address of a single DHCP server to the list of DHCP servers from which the system can request addresses to allocate to remote users. A maximum of five DHCP servers can be specified. The **no** version removes the specified DHCP server or removes all DHCP servers from the list.

**Syntax:** ip dhcp-server *dhcpServerAddress* [ *adminStatus* ]

no ip dhcp-server [ *dhcpServerAddress* [ *adminStatus* ] ]

- *Ipaddress* – IP address of the DHCP server that will allocate addresses for remote users
- *adminStatus – one of the following options:*
  - disable – disable the DHCP server
  - drain – drain the DHCP server

**Mode(s):** Global Configuration

## ip directed-broadcast

**Description:** Enables translation of directed broadcast to physical broadcasts. The **no** version disables the function.

**Syntax:** [ no ] ip directed-broadcast

**Mode(s):** Interface Configuration, Subinterface Configuration, Profile Configuration

## ip disable-forwarding

**Description:** Disables forwarding of packets on the SRP Ethernet interface to maintain system performance. The **no** version enables forwarding of packets on the SRP Ethernet interface. You see an error message if you try to set this command for interfaces other than the SRP Ethernet interface.

**Syntax:** [ no ] ip disable-forwarding

**Mode(s):** Interface Configuration, Subinterface Configuration

## ip domain-lookup

**Description:** Without the **transit-virtual-router** option, enables the system to query the configured DNS name servers when it needs an IP hostname–to–IP address translation. With the **transit-virtual-router** option, configures a virtual router to use the name servers you configured for another virtual router. The **no** version without the **transit-virtual-router** option restores the default situation, in which the system does not query the DNS server. The **no** version with the **transit-virtual-router** option stops a virtual router from using the same name servers you configured for another virtual router.

**Syntax:** [ no ] ip domain-lookup [ transit-virtual-router *vrName ]*

*vrName* – name of the virtual router that has the DNS configuration you want to use for a second virtual router

**Mode(s):** Global Configuration

## ip domain-name

**Description:** Defines a default domain name for the clients that a name resolver serves. The **no** version deletes the domain name; that is, the domain name will no longer be appended to hostnames in the static host table.

**Syntax:** [ no ] ip domain-name *domainName*

- *domainName* – default domain name for your hosts

**Mode(s):** Global Configuration

## ip dvmrp

**Description:** Activates DVMRP on an interface.The **no** version removes DVMRP from an interface.

**Syntax:** [ no ] ip dvmrp

**Mode(s):** Interface Configuration

## ip dvmrp accept-filter

**Description:** Filters incoming DVMRP reports in accordance with a standard IP access list. The **no** version disables the filter.

**Syntax:** [ no ] ip dvmrp accept-filter *listName1* [ *distance* ] neighbor-list *listName2*

- *listName1* – name of the IP access list. If the name is 0, the interface accepts all destinations. You can specify a simple or extended access list; with an extended access list you can specify an address and a subnet mask.
- *distance* – distance associated with the DVMRP route when the router determines the RPF interface for the source of a multicast packet. The default is 0; the range is 0 to 255.
- *listName2* – name of an access list containing the neighbors from which the system will accept reports. If the name is 0, the interface accepts destinations from all its neighbors.

**Mode(s):** Interface Configuration

## ip dvmrp announce-filter

**Description:** Specifies a list of DVMRP routes that the system will advertise on an interface. The **no** version restores the default situation, in which the system advertises all known routes on the interface.

**Syntax:** ip dvmrp announce-filter *listName*

no ip dvmrp announce-filter

- *listName* – name of the IP access list that specifies the DVMRP routes that the system will advertise on the interface. You can specify a simple or extended access list; with an extended access list you can specify an address and a subnet mask.

**Mode(s):** Interface Configuration

## ip dvmrp auto-summary

**Description:** Summarizes routes automatically on an interface. By default, automatic summarization is enabled. The **no** version disables automatic summarization.

**Syntax:** [ no ] ip dvmrp auto-summary

**Mode(s):** Interface Configuration

## ip dvmrp disable

**Description:** Disables DVMRP on an interface without removing the DVMRP configuration. The **no** version reenables the DVMRP configuration on a disabled interface.

**Syntax:** [ no ] ip dvmrp disable

**Mode(s):** Interface Configuration

## ip dvmrp metric-offset

**Description:** Adjusts the number of hops associated with routes passing through an interface. This action indicates that the route is more efficient or less efficient than an alternative route. The **no** version restores the default values.

**Syntax:** [ no ] ip dvmrp metric-offset { in | out } [ *increment* ]

- in – increments the number of hops for a DVMRP route advertised in incoming DVMRP reports. If you do not specify a key word, this option is the default.
- out – increments the number of hops for a DVMRP route advertised in outgoing DVMRP reports
- *increment* – number of hops associated with this interface. The default is 1 for incoming reports and 0 for outgoing reports.

**Mode(s):** Interface Configuration

## ip dvmrp route-hog-notification

**Description:** Sets the number of DVMRP routes that the system can record before it generates a syslog warning message.The **no** version restores the default setting, 10,000 routes.

**Syntax:** [ no ] ip dvmrp route-hog-notification [ *limit* ]

**Mode(s):** Global Configuration

## ip dvmrp route-limit

**Description:** Limits the number of routes that the system can advertise on each interface. The default value is 7000. The **no** version removes the limit for the number of routes that the system can advertise on each interface.

**Syntax:** [ no ] ip dvmrp route-limit [ *limit* ]

- *limit* – number of routes that the system can advertise

**Mode(s):** Global Configuration

## ip dvmrp summary-address

**Description:** Advertises a DVMRP summary address on the interface. The **no** version stops the advertising of a summary address on an interface.

**Syntax:** [ no ] ip dvmrp summary-address *ipAddress mask* [ metric *cost* ]

- *ipAddress* – summary address
- *mask* – subnet mask
- *cost* – cost associated with this summary address

**Mode(s):** Interface Configuration

## ip dvmrp unicast-routing

**Description:** Enables the exchange of DVMRP unicast routes on an interface not owned by DVMRP. The **no** version disables the exchange of DVMRP unicast routes on an interface not owned by DVMRP.

**Syntax:** [ no ] ip dvmrp unicast-routing

**Mode(s):** Interface Configuration

## ip dynamic-interface-prefix

**Description:** Specifies the prefix for the names of dynamic shared IP interfaces created for overlapping BGP/MPLS VPNs.The **no** version restores the default prefix, **dyn**.

**Syntax:** ip dynamic-interface-prefix [ *vrfName* ] *prefix*

no ip dynamic-interface-prefix [ *vrfName* ]

- *vrfName* – name of the VRF in which the shared interface is created; a string of 1–32 alphanumeric characters
- *prefix* – string of 1–10 alphanumeric characters

**Mode(s):** Global Configuration

## ip explicit-path

**Description:** Defines an explicit path by name and also enables or disables explicit path routing in a non-ERX implementation. See the **mpls explicit-path** command for a complete description and syntax.

## ip extcommunity-list

**Description:** Defines an extended-community (extcommunity) list to be referenced in a route map. The **no** version deletes the extcommunity.

**Syntax:** ip extcommunity-list *listName* { permit | deny } *extendedCommunity* [ *extendedCommunity* ]*

no ip extcommunity-list *listName*

- *listName* – name of the extended-community list
- permit – permits membership in the extended community for matching conditions
- deny – denies membership in the extended community for matching conditions
- *extendedCommunity* – extended community specified in the format: {rt | soo } { *ASN*:*nn* | *ipAddress*:*nn* }
  - rt – specifies a route-target community; consists of one or more routers that can receive a set of routes advertised by BGP that carry the extended-community attribute
  - soo – specifies a Site-of-Origin community; consists of one or more routers that injects a set of routes into BGP that carry the extended-community attribute
  - *ASN*:*nn* – identifies the extended community by a 16-bit autonomous system number followed by a 32-bit integer
  - *ipAddress*:*nn* – identifies the extended community identified by an IP address followed by a 32-bit integer
- * – indicates that one or more parameters can be repeated multiple times in a list in the command line

**Mode(s):** Global Configuration

## ip filter-options all

**Description:** Enables filtering of packets with IP options on an interface. IP options filtering is disabled by default. The **no** version disables filtering of packets with IP options.

**Syntax:** [ no ] ip filter-options all

**Mode(s):** Interface Configuration

## ip ftp source-address

**Description:** Specifies an operational interface by IP address as the source interface in FTP packets sent via the system's FTP client. The **no** version restores the source address in the FTP packets to that on which the FTP connection is made.

**Syntax:** ip ftp source-address *sourceAddress*

no ip ftp source-address [ *sourceAddress* ]

- *sourceAddress* – source IP address

**Mode(s):** Global Configuration

## ip ftp source-interface

**Description:** Identifies an interface by type and location as the source interface in FTP packets sent via the system's FTP client. The **no** version restores the source address in the FTP packets to that on which the FTP connection is made.

**Syntax:** ip ftp source-interface *interfaceType interfaceSpecifier*

no ip ftp source-interface [ *interfaceType interfaceSpecifier* ]

- *interfaceType* – interface type; see *Interface Types and Specifiers* in *About This Guide*
- *interfaceSpecifier* – particular interface; format varies according to interface type; see *Interface Types and Specifiers* in *About This Guide*

**Mode(s):** Global Configuration

## ip-hint

**Description:** When enabled, the ERX will preallocate an IP address for the remote (B-RAS) user before calling authentication.The address is then passed as a hint in the authentication request to the RADIUS server. The **no** version disables the feature.

**Syntax:** ip-hint { enable | disable }

no ip-hint

**Mode(s):** Domain Map Configuration

## ip http access-class

**Description:** Allows only subscribers on a standard IP access list to connect to the HTTP local server. The **no** version removes the association between the access list and the HTTP local server.

**Syntax:** ip http access-class *listName*

no ip http access-class

- *listName* – name of the access list

**Mode(s):** Global Configuration

## ip http max-connection-time

**Description:** Specifies the maximum time that the HTTP local server maintains an inactive connection. The **no** version restores the default time, 30 seconds.

**Syntax:** ip http max-connection-time *seconds*

no ip http max-connection-time

- *seconds* – time that the HTTP local server maintains a connection; either 0 (forever) or in the range 3–7200 seconds

**Mode(s):** Global Configuration

## ip http not-found-url

**Description:** Specifies the Web page or message that appears if the subscriber requests a URL that is not available. The **no** version displays the standard HTTP message “404 not found.”

**Syntax:** ip http not-found-url { *url* | root-url }

no ip http not-found-url

- *url* – URL of the Web page that appears on the subscriber’s computer if the DHCP local server is not configured or enabled
- root-url – displays the root Web page

**Mode(s):** Global Configuration

## ip http port

**Description:** Specifies the port on which the HTTP local server receives connection attempts. The **no** version restores the default port number, 80.

**Syntax:** ip http port *portNumber*

no ip http port

- *portNumber* – number of the port on which connection attempts are received, in the range 0–65535

**Mode(s):** Global Configuration

## ip http realm

**Description:** Specifies the name the Web browser displays to subscribers when the HTTP client requests the user information. The **no** version restores the default value, the name of the virtual router.

**Syntax:** ip http realm *connectionName*

no ip http realm

- *connectionName* – connection name that subscribers see when the HTTP client requests user information

**Mode(s):** Global Configuration

## ip http root-refresh

**Description:** Specifies how often the browser updates the default internal starting (root) Web page. The **no** version restores the default time interval, 0 seconds.

**Syntax:** ip http root-refresh *seconds*

no ip http root-refresh

- *seconds* – time interval, in the range 0–65535 seconds, at which the default internal starting Web page is refreshed

**Mode(s):** Global Configuration

## ip http root-url

**Description:** Specifies a URL external to the system for the starting (root) Web page that appears on the subscriber's computer when the subscriber logs in. The **no** version restores the default internal URL as the starting Web page.

**Syntax:** ip http root-url *url*

no ip http root-url

- *url* – external URL of the Web page that appears on the subscriber's computer when the connection to the HTTP local server is established

**Mode(s):** Global Configuration

## ip http same-host-limit

**Description:** Specifies the maximum number of connections that can exist between one IP address and the HTTP local server. The **no** version restores the default number of allowed connections, 3.

**Syntax:** ip http same-host-limit *maxConnections*

no ip http same-host-limit

- *maxConnections* – maximum number of connections allowed between one IP address and the HTTP local server, in the range 0–1000

**Mode(s):** Global Configuration

## ip http server

**Description:** Creates or enables the HTTP local server. The **no** version deletes or disables the HTTP local server.

**Syntax:** [ no ] ip http [ server ]

- server – enables or disables the HTTP local server. If you do not specify the keyword **server**, the command creates the HTTP local server and the **no** version removes the server.

**Mode(s):** Global Configuration

## ip igmp

**Description:** Enables IGMPv2 on an interface. The **no** version disables IGMPv2 on an interface.

**Syntax:** [ no ] ip igmp

**Mode(s):** Interface Configuration, Profile Configuration

## ip igmp access-group

**Description:** Restricts hosts on this subnet to joining multicast groups on the specified IP access list. The **no** version removes the association with the specified access list and allows hosts on the subnet to join any multicast group.

**Syntax:** ip igmp access-group *accessListName*

no ip igmp access-group

- *accessListName* – name of the access list; a string of up to 32 characters

**Mode(s):** Interface Configuration, Profile Configuration

## ip igmp group limit

**Description:** Limits the number of IGMP groups that an interface can accept. The **no** version restores the default situation, in which there is no limit to the number of IGMP groups that the interface accepts.

**Syntax:** ip igmp group limit *groupLimit*

no ip igmp group limit

- *groupLimit* – maximum number of IGMP groups that an interface can accept in the range 0–64,000

**Mode(s):** Interface Configuration, Profile Configuration

## ip igmp immediate-leave

**Description:** Removes an interface immediately when the router receives an leave group membership message from the host associated with this interface. The **no** version restores the default situation, in which the router issues query messages to multicast groups and removes an interface if the associated host does not return a group membership report within a certain length of time.

***Caution:*** *Issue this command only on IGMPv2 interfaces to which one IGMP client is connected. Do not issue this command to interfaces to which more than one IGMP client is connected.*

**Syntax:** [ no ] ip igmp immediate-leave

**Mode(s):** Interface Configuration, Profile Configuration

## ip igmp last-member query-interval

**Description:** Specifies in tenths of a second how long the system waits after receiving an IGMP leave message before it sends another query. The **no** version restores the default value, 10 tenths of a second (1 second).

**Syntax:** ip igmp last-member-query-interval *tenthsOfaSecond*

no ip igmp last-member-query-interval

- *tenthsOfaSecond* – time interval between receipt of an IGMP leave message and sending out of a query in the range 1–254 tenths of a second. Using a lower value allows members to leave groups more quickly.

**Mode(s):** Interface Configuration, Profile Configuration

## ip igmp promiscuous

**Description:** Enables the specified interface to accept IGMP reports from hosts on any subnet. The **no** version specifies that an IGMP interface should use the Router Configuration mode setting (see the **igmp promiscuous** command) to determine from which subnets it can accept IGMP reports.

**Syntax:** ip igmp promiscuous { on | off }

no ip igmp promiscuous

- on – enables the interface to accept IGMP reports from hosts on any subnet
- off – allows the interface to accept IGMP reports only from hosts on subnets associated with this interface

**Mode(s):** Interface Configuration, Profile Configuration

## ip igmp-proxy

**Description:** Enables IGMP proxy on an interface. The **no** version disables IGMP proxy for an interface.

**Syntax:** [ no ] ip igmp-proxy

**Mode(s):** Interface Configuration

## ip igmp-proxy unsolicited-report-interval

**Description:** Specifies how often the upstream interface should transmit unsolicited reports. This command has no effect on interfaces other than the upstream value. The **no** version transmits unsolicited reports using the default value, 400 seconds.

***Note:*** *Issue this command only on the upstream interface. Otherwise, this command will have no effect.*

**Syntax:** ip igmp-proxy unsolicited-report-interval *seconds*

no ip igmp-proxy unsolicited-report-interval

- *seconds* – time interval at which the interface transmits unsolicited reports

**Mode(s):** Interface Configuration

## ip igmp-proxy V1-router-present-time

**Description:** Specifies how long the system assumes that there is an IGMPv1 querier router on the subnet after the system receives an IGMP V1 query on this interface. The **no** version restores the default value, 10 seconds.

**Syntax:** ip igmp-proxy V1-router-present-time *seconds*

no ip igmp-proxy V1-router-present-time

- *seconds* – time for which the system assumes that there is an IGMPv1 querier router on the subnet after the system receives an IGMP V1 query on this interface

**Mode(s):** Interface Configuration

## ip igmp querier

**Description:** Specifies that the interface will act as a querier when you configure IGMPv1 on an interface. The **no** version specifies that this interface will not issue query packets.

***Note:*** *This command is invalid for interfaces on which you configured IGMPv2.*

**Syntax:** [ no ] ip igmp querier

**Mode(s):** Interface Configuration, Profile Configuration

## ip igmp querier-timeout

**Description:** Sets the time that the interface waits before declaring itself as the querier. The **no** version restores the default value, twice the query interval.

**Syntax:** ip igmp querier-timeout *seconds*

no ip igmp querier-timeout

- *seconds* – time interval between the last query from the previous router and the first query from this interface

**Mode(s):** Interface Configuration, Profile Configuration

## ip igmp query-interval

**Description:** Sets how often the system sends IGMP host-query packets from this interface. The **no** version restores the default value, 125 seconds.

**Syntax:** ip igmp query-interval *seconds*

no ip igmp query-interval

- *seconds* – polling interval in the range 0–65535 seconds

**Mode(s):** Interface Configuration, Profile Configuration

## ip igmp query-max-response-time

**Description:** Specifies the period in tenths of a second during which the host is expected to respond to an IGMP query. IGMP version 2 includes this value in IGMP query messages sent out on the interface. You cannot set this value on interfaces running IGMP version 1. The **no** version restores the default value of 10 tenths of a second (1 second).

**Syntax:** ip igmp query-max-response-time *tenthsOfaSecond*

no ip igmp query-max-response-time

- *tenthsOfaSecond* – time interval between receipt of an IGMP query and the response; the range is 1–254 tenths of a second.

**Mode(s):** Interface Configuration, Profile Configuration

## ip igmp robustness

**Description:** Specifies the number of times that the system sends IGMP group-specific queries before declaring a group to no longer have any members on an interface. The **no** version restores the default value, 3.

**Syntax:** ip igmp robustness *numberOfMessages*

no ip igmp robustness

- *numberOfMessages* – number of times that the system sends IGMP group-specific queries in the range 1–4

**Mode(s):** Interface Configuration, Profile Configuration

## ip igmp static-group

**Description:** Assigns an interface to handle all multicast traffic for a group. The interface sets no timers for this group. The **no** version removes the group from the interface.

**Syntax:** [ no ] ip igmp static-group *groupAddress*

- *groupAddress* – address of the group

**Mode(s):** Interface Configuration, Profile Configuration

## ip igmp version

**Description:** Sets the IGMP version for the interface. The **no** version restores the default value, IGMPv2.

**Syntax:** ip igmp version { 2 | 1 }

no ip igmp version

2 – IGMP version 2

1 – IGMP version 1

**Mode(s):** Interface Configuration, Profile Configuration

## ip ignore-df-bit

**Description:** Specifies that the system ignores the don't-fragment bit if present in the IP header of packets crossing the configured interface; the system then fragments packets even if the bit is present. The **no** version restores the default behavior, which is to consider the DF bit before fragmenting.

**Syntax:** [ no ] ip ignore-df-bit

**Mode(s):** Interface Configuration

## ip interface

**Description:** This command has only a **no** version. See the **no ip interface**command for a complete description and syntax.

## ip irdp

**Description:** Enables ICMP Router Discovery Protocol processing on an interface. The **no** version disables IRDP routing.

**Syntax:** [ no ] ip irdp

**Mode(s):** Interface Configuration, Subinterface Configuration

## ip local pool

**Description:** Specifies the pool name, the starting address, the ending address, group name, the utilization threshold, and the SNMP trap flag. The **no** version deletes a local pool.

**Syntax:** [ no ] ip local pool *name* [ *startIpAddress* [ *endIpAddress* ]
[ warning *highUtilization abatedUtilization* ] [ snmpTrap ] ]

- *name* – text string in the range 1–16 characters that defines the name of the local address *pool*
- *startIpAddress* – starting IP address in the local address pool
- *endIpAddress* – ending IP address in the local address pool
- warning – specifies one of the following utilization warnings:
  - highUtilization – high utilization value; a number in the range 1–100; default is 85
  - abatedUtilization – abated utilization value; a number in the range 1–100; default is 75
- snmpTrap – enables SNMP pool utilization traps

**Mode(s):** Global Configuration

## ip local pool snmpTrap

**Description:** Enables SNMP pool utilization traps. The **no** version disables SNMP pool utilization traps.

**Syntax:** [ no ] ip local pool *name* snmpTrap

- *name* – text string in the range 1–16 characters that defines the name of the local address *pool*

**Mode(s):** Global Configuration

## ip local pool warning

**Description:** Use to identify the warning threshold values. The **no** version resets the thresholds to their default values.

**Syntax:** ip local pool *name* warning *highUtilization abatedUtilization* [ snmpTrap ]

no ip local pool *name* warning [ *highUtilization abatedUtilization* ]

- *name* – text string in the range 1–16 characters that defines the name of the local address *pool*
- *highUtilization* – high utilization value; a number in the range 1–100; default is 85
- *abatedUtilization* – abated utilization value; a number in the range 1–100; default is 75
- snmpTrap – enables snmp pool utilization traps

**Mode(s):** Global Configuration

## ip mac-validate

**Description:** Enables MAC address validation on a per interface basis. The **no** version disables the feature.

**Syntax:** ip mac-validate [ strict | loose ]

no ip mac-validate

- strict – prevents transmission of IP packets that do not reside in the validation table
- loose – allows IP packets to pass through even though the packets do not have entries in the validation table; only packets that have matching IP–MAC pair entries in the table are validated

**Mode(s):** Interface Configuration

## ip mask-reply

**Description:** Enables ICMP netmask reply. The **no** version disables the feature.

**Syntax:** [ no ] ip mask-reply

**Mode(s):** Interface Configuration, Subinterface Configuration

## ip mpls forwarding-mode label-switched

**Description:** Generates a label for each different FEC that a BGP route points to in a BGP/MPLS VPN. The **no** version restores the default, generating a single label for all BGP routes sent from a given VRF.

***Note:*** *For some types of routes, the system always generates a per-VRF label, regardless of the status of this command. See ERX Routing Protocols Configuration Guide, Vol. 2, Chapter 3, Configuring BGP/MPLS VPNs, for details.*

**Syntax:** [ no ] ip mpls forwarding-mode label-switched

**Mode(s):** VRF Configuration

## ip mpls vpn-interface per-label

**Description:** Creates a VPN interface for each received stacked label in a BGP/MPLS VPN, enabling collection of statistics on a per-label basis. The **no** version restores the default, creating a VPN interface for each next-hop PE.

***Note:*** *Operating in per-label mode limits the number of egress FECs supported to the order of thousands.*

**Syntax:** [ no ] ip mpls vpn-interface per-label

**Mode(s):** VRF Configuration

## ip mtu

**Description:** Sets the maximum transmission unit size of IP packets sent on an interface. The **no** version restores the default value.

**Syntax:** [ no ] ip mtu [ *mtuSize* ]

- *mtuSize* – maximum number of packet transmissions permitted on an interface. The range is 128–10240. The default is 0, which means that the system takes the value from a lower protocol layer.

**Mode(s):** Interface Configuration, Subinterface Configuration, Profile Configuration

## ip multicast-routing

**Description:** Enables IP multicast routing on the system. The **no** version disables IP multicast routing on the system.

**Syntax:** [ no ] ip multicast-routing

**Mode(s):** Global Configuration

## ip multipath round-robin

**Description:** Specifies round-robin as the mode for ECMP load sharing on an interface. The **no** version restores the default value, hashed.

**Syntax:** [ no ] ip multipath round-robin

**Mode(s):** Subinterface Configuration

## ip name-server

**Description:** Specifies a DNS name server that the system can query for hostname–to–IP address resolution. The **no** version deletes the name server.

**Syntax:** [ no ] ip name-server *serverIpAddress [ serverIpAddress ]**

- *serverIpAddress* – IP address of a DNS name server
- * – indicates that one or more parameters can be repeated multiple times in a list in the command line

**Mode(s):** Global Configuration

## ip nfs

**Description:** Specifies the ERX interface that the current virtual router uses to exchange NFS communications with an NFS server. The **no** version prevents this interface from sending or receiving NFS communications for the current virtual router.

**Syntax:** ip nfs { source-address *ipAddress* | source-interface *interfaceType interfaceSpecifier* }

no ip nfs { source-address | source-interface }

- *ipAddress* – IP address of an ERX interface that sends and receives NFS communications
- *interfaceType* – ERX interface type; see *Interface Types and Specifiers* in *About This Guide*
- *interfaceSpecifier* – particular ERX interface; format varies according to interface type; see *Interface Types and Specifiers* in *About This Guide.*

**Mode(s):** Global Configuration

## ip nfs host

**Description:** Configures a remote host as an NFS server for the current virtual router. The **no** version disassociates the NFS server from the virtual router.

**Syntax:** ip nfs host *hostName* [ user *userID* [ group *groupID* ] ]

no ip nfs host *hostName*

- *hostName* – name of the remote host
- *userID* – user identity in the range 0–4294967295 that a user must enter to connect to the remote host; default is 2001
- *groupID* – group identity in the range 0–4294967295 that the user must enter to connect to the remote host; default is 100

**Mode(s):** Global Configuration

## ip ospf authentication-key

**Description:** Assigns a password used by neighboring routers that are using OSPF simple password authentication. The **no** version deletes the password.

**Syntax:** ip ospf authentication-key *authKey*

no ip ospf authentication-key

- *authKey* – password; continuous string of characters up to 8 characters in length

**Mode(s):** Interface Configuration, Subinterface Configuration

## ip ospf authentication message-digest

**Description:** Specifies that the authentication mode for the interface is MD5. The **no** version sets authentication for the interface to none, but leaves any configured MD5 key intact.

**Syntax:** [ no ] ip ospf authentication message-digest

**Mode(s):** Interface Configuration, Subinterface Configuration

## ip ospf authentication-none

**Description:** Specifies that no authentication is to be used for the interface. The **no** version has no effect.

**Syntax:** ip ospf authentication-none

**Mode(s):** Interface Configuration, Subinterface Configuration

## ip ospf cost

**Description:** Specifies a cost metric for an interface. Used in the calculation of the SPF routing table. The **no** version resets the path cost to the default.

**Syntax:** [ no ] ip ospf cost *intfCost*

- *intfCost* – link state metric cost; number in the range 0–65535; default value is 10

**Mode(s):** Interface Configuration, Subinterface Configuration

## ip ospf dead-interval

**Description:** Sets the time period during which the router's neighbors do not see hello packets before they declare the router to be down. The **no** version resets the dead interval to its default.

**Syntax:** [ no ] ip ospf dead-interval *deadInterval*

- *deadInterval* – number in the range 1–65535 seconds; default value is 40 seconds

**Mode(s):** Interface Configuration, Subinterface Configuration

## ip ospf hello-interval

**Description:** Specifies the interval between hello packets that the router sends on the interface. The **no** version resets the hello interval to its default.

**Syntax:** [ no ] ip ospf hello-interval *helloInterval*

- *helloInterval* – number in the range 1–65535 seconds; default value is 10 seconds

**Mode(s):** Interface Configuration, Subinterface Configuration

## ip ospf message-digest-key md5

**Description:** Enables OSPF MD5 authentication and configures the MD5 key. The **no** version deletes an MD5 key.

**Syntax:** ip ospf message-digest-key *keyID* md5 [ 0 | 8 ] *msgDigestKey*

no ip ospf message-digest-key *keyID*

- *keyID* – key identifier in the range 1–255
- md5 – specifies use of the MD5 algorithm
- 0 – indicates the *msgDigestKey* is entered in unencrypted form (plaintext); this is the default option
- 8 –indicates the *msgDigestKey* is entered in encrypted form (ciphertext)
- *msgDigestKey* – OSPF password; a continuous string of up to 16 alphanumeric characters

***Note:*** *If all the MD5 keys have been deleted, the authentication type is still MD5, but you must configure MD5 keys.*

***Note:*** *To disable MD5 authentication for the interface, use the* ***ip ospf authentication-none*** *command.*

***Note:*** *To display the password only in encrypted text, use the* ***service password-encryption*** *command.*

**Mode(s):** Interface Configuration, Subinterface Configuration

## ip ospf network

**Description:** Configures the OSPF network type to something other than the default for the network medium. The **no** version restores the default value for the medium.

**Syntax:** ip ospf network { broadcast | non-broadcast | point-to-point }

no ip ospf network

- broadcast – sets the network type to broadcast
- non-broadcast – sets the network type to NBMA
- point-to-point – sets the network type to point-to-point

**Mode(s):** Interface Configuration, Subinterface Configuration

## ip ospf priority

**Description:** Sets the router priority. Used in determining the designated router for the particular network. This designation applies only to multiaccess networks. Every broadcast and nonbroadcast multiaccess network has a designated router. The **no** version restores the default value.

**Syntax:** [ no ] ip ospf priority *intfPriority*

- *intfPriority* – priority value, an 8-bit number in the range 1–255; default value is 1

**Mode(s):** Interface Configuration, Subinterface Configuration

## ip ospf retransmit-interval

**Description:** Specifies the time between LSA retransmissions for the interface when an acknowledgment for the LSA is not received. The **no** version restores the default value.

**Syntax:** [ no ] ip ospf retransmit-interval *retransInterval*

- *retransInterval* – number in the range 1–65535 seconds; default value is 5 seconds

**Mode(s):** Interface Configuration, Subinterface Configuration

## ip ospf shutdown

**Description:** Disables OSPF on an interface. The **no** version enables OSPF on the interface.

**Syntax:** [ no ] ip ospf shutdown

**Mode(s):** Interface Configuration, Subinterface Configuration

## ip ospf transmit-delay

**Description:** Sets the estimated time it takes to transmit a link state update packet on the interface. The **no** version restores the default value.

**Syntax:** [ no ] ip ospf transmit-delay *transmDelay*

- *transmDelay* – link state transmit delay, a number in the range 1–65535 seconds; default value is 1 second

**Mode(s):** Interface Configuration, Subinterface Configuration

## ip pim

**Description:** Enables PIM on an interface. The **no** version disables PIM on an interface.

**Syntax:** [ no ] ip pim [ dense-mode | sparse-mode | sparse-dense-mode ]

- dense-mode – enables PIM in dense mode
- sparse-mode – enables PIM in sparse mode
- sparse-dense-mode – enables PIM in sparse-dense mode

**Mode(s):** Interface Configuration

## ip pim query-interval

**Description:** Specifies how often the system sends PIM router query messages from this interface. The **no** version specifies the default time interval, 30 seconds.

**Syntax:** ip pim query-interval *queryTime*

no ip pim query-interval

- *queryTime* – interval in the range 0–210 seconds at which the system sends PIM router query messages from this interface

**Mode(s):** Interface Configuration

## ip pim rp-address

**Description:** Specifies a static PIM group–to–RP mapping. The **no** version clears the mapping from this interface.

**Syntax:** [ no ] ip pim rp-address *ipAddress* [ *ipAccessList* ] [ override ]

- *ipAddress* – IP address of the system you want to designate as an RP router
- *ipAccessList* – name of the IP access list that specifies which multicast groups use this RP
- override – specifies that this static RP mapping has priority over group-to-RP mappings learned by auto-RP

**Mode(s):** Configuration Mode

## ip pim send-rp-announce

**Description:** Sends autoRP announcement messages from a system you configured as an RP. The **no** version clears the filter from this interface.

**Syntax:** ip pim send-rp-announce *interfaceType interfaceSpecifier* scope *ttl* [ group-list *ipAccessList*] [ interval *seconds* ]

no ip pim send-rp-announce *interfaceType interfaceSpecifier* [ group-list *ipAccessList* ]

- *interfaceType* – interface type; see *Interface Types and Specifiers* in *About This Guide*
- *interfaceSpecifier* – particular interface; format varies according to interface type; see *Interface Types and Specifiers* in *About This Guide.* The autoRP announcement messages will contain the IP address for this interface.
- *ttl* – time-to-live value; the number of hops for which the announcement is valid
- *ipAccessList* – name of the IP access list that specifies which multicast groups use this RP
- *seconds* – time interval at which the system sends the announcements; default interval is 60 seconds

**Mode(s):** Configuration Mode

## ip pim send-rp-discovery scope

**Description:** Configures the system as an RP mapping agent, which records RP-to-group mappings and notifies PIM DRs about the mappings. The **no** version stops the system from acting as an RP mapping agent.

**Syntax:** ip pim send-rp-discovery scope *ttl* [ *interfaceType interfaceSpecifier* ]

no ip pim send-rp-discovery

- *ttl* – time-to-live value; number of hops for which the RP discovery message is valid. Specify a value that covers the PIM domain.
- *interfaceType* – interface type; see *Interface Types and Specifiers* in *About This Guide*
- *interfaceSpecifier* – particular interface; format varies according to interface type; see *Interface Types and Specifiers* in *About This Guide.* If you specify an interface, the autoRP discovery messages will contain the IP address for this interface.

**Mode(s):** Configuration Mode

## ip pim spt-threshold

**Description:** Specifies the network configuration that PIM SM uses when a source starts sending multicast messages. The **no** version restores the default value, 0.

**Syntax:** [ no ] ip pim spt-threshold { 0 | *nonzero_integer* | infinity } [ group-list *ipAccessList* ]

- 0 – configures PIM SM to switch to an SPT when a source begins to send multicast messages
- *nonzero_integer* – integer in the range 1–4294967294; prevents PIM SM from switching from a shared tree to an SPT
- infinity – prevents PIM SM from switching from a shared tree to an SPT
- *ipAccessList* – name of the IP access list that specifies the groups to which the threshold applies

**Mode(s):** Global Configuration

## ip policy

**Description:** Assign a policy list to the ingress or egress of an interface. If you execute a **policy** command and the policy list does not exist, the system creates a policy list with no rules, the default. When no rules are found in a policy list, the system performs a routing table lookup and forwards packets on the interface based on the routing table information. You must specify the **input** or **output** keyword to assign the policy list to the ingress or egress of the interface. The **no** version removes the association between a policy list and an interface.

**Syntax:** [ no ] ip policy { input | local-input | output} *policyName*
[ statistics { enabled [ baseline { enabled | disabled } ] | disabled } ]

- input – apply policy to data arriving at this interface
- local-input – apply policy to data that arrives at this interface but is addressed to a local interface
- output – apply policy to data leaving this interface
- *policyName* – name of the policy; a maximum of 16 characters
- statistics – enable or disable collection of policy routing statistics
  - › enabled – enable collection of policy routing statistics
  - › baseline enabled – enables baselining of policy routing statistics
  - › baseline disabled – disables baselining of policy routing statistics
  - › disabled – disable collection of policy routing statistics

**Mode(s):** Interface Configuration

## ip prefix-list

**Description:** Creates a prefix list for route filtering; specifies a list entry—a permit or deny clause for a network address. The **no** version removes the specified prefix list or the specified list entry.

**Syntax:** ip prefix-list *listName* { description *desc* |
[ seq *sequence* ] { permit | deny } *ipPrefix* [ ge *geNumber* ] [ le *leNumber* ] }

no ip prefix-list *listName* [ description |
[ seq *sequence* ] [ { permit | deny } *ipPrefix* [ ge *geNumber* ] [ le *leNumber* ] ] ]

- *listName* – name of the prefix list; a string of up to 32 characters
- *desc* – description of the prefix list
- *sequence* – number in the range 0–65535 that indicates the position the prefix list entry is to have in the list of entries already configured for the prefix list. If given with the **no** version of this command, it specifies the position of the list entry to be deleted. If *sequence* is not specified, the value of the last sequence number + 5 is used.
- permit – if the prefix of the route being filtered matches the specified prefix and **permit** is specified, the route is redistributed as controlled by the set actions
- deny – if the prefix of the route being filtered matches the specified prefix and **deny** is specified, the route is not redistributed
- *ipPrefix* – network route to be filtered, in the format *network*/*length*, where
  - *network* – base address of the network route to be filtered; for example, 192.168.32.0 or 10.10.0.0
  - *length* – length of the network prefix; number of bits masking base address to produce address to be matched
- *geNumber* – route being filtered matches if its prefix is within the range specified: greater than or equal to *geNumber* and less than or equal to 32
- *leNumber* – route being filtered matches if its prefix is within the range specified: greater than or equal to *length* and less than or equal to *leNumber*

**Mode(s):** Global Configuration

## ip prefix-tree

**Description:** Creates a prefix tree for best-match route filtering; specifies a tree entry—a deny or permit clause for a network address. The **no** version removes the specified prefix tree or the specified tree entry.

**Syntax:** ip prefix-tree *treeName* { description *desc* | { permit | deny } *ipPrefix* }

no ip prefix-tree *treeName* [ description | { permit | deny } *ipPrefix* ]

- *treeName* – name of the prefix list; a string of up to 32 characters
- *desc* – description of the prefix list
- deny – if the prefix of the route being filtered matches the specified prefix and **deny** is specified, the route is not redistributed
- permit – if the prefix of the route being filtered matches the specified prefix and **permit** is specified, the route is redistributed as controlled by the set actions
- *ipPrefix* – network route to be filtered, in the format *network*/*length*, where
  - *network* – base address of the network route to be filtered; for example, 192.168.32.0 or 10.10.0.0
  - *length* – length of the network prefix; number of bits masking base address to produce address to be matched

**Mode(s):** Router Configuration

## ip proxy-arp

**Description:** Enables proxy ARP on an Ethernet or bridge1483 interface. Proxy ARP is enabled by default. The **no** version disables proxy ARP on an Ethernet or bridge1483 interface.

**Syntax:** ip proxy-arp [ restricted | unrestricted ]

no ip proxy-arp

- restricted – restricts proxy-arp to hosts on the local interface
- unrestricted – enables proxy-arp for all reachable hosts

**Mode(s):** Interface Configuration, Subinterface Configuration

## ip redirects

**Description:** Enables the sending of redirect messages if the software is forced to resend a packet through the same interface on which it was received. The **no** version disables the sending of redirect messages.

**Syntax:** [ no ] ip redirects

**Mode(s):** Interface Configuration, Subinterface Configuration, Profile Configuration

## ip refresh-route

**Description:** Reinstalls routes removed from the IP routing table by the **clear ip route** command. There is no **no** version.

**Syntax:** ip refresh-route [ vrf *vrfName* ]

- *vrfName* – name of the VRF; string of 1–32 alphanumeric characters

**Mode(s):** Privileged Exec

## ip rip

**Description:** Configures RIP to run on the network specified by the **network** command. Uses the default values: send version is RIP version 1, receive version is RIP version 1 and version 2, authentication is not enabled. The **no** version deletes the RIP interface.

**Syntax:** [ no ] ip rip

**Mode(s):** Interface Configuration, Subinterface Configuration

## ip rip authentication key

**Description:** Specifies the password for text authentication and the key for MD5 authentication. The **no** version clears the key for the interface. Supported only in RIP version 2. Authentication is disabled by default.

**Syntax:** ip rip authentication key [ 0 | 8 ] *authkey*

no ip rip authentication key

- 0 – the *authKey* is entered in unencrypted form (plaintext); this is the default option
- 8 – the *authKey* is entered in encrypted form (ciphertext)
- *authkey* – password sent with RIP messages or the key used to encrypt/decrypt RIP messages, depending on the authentication mode set for this interface

**Mode(s):** Interface Configuration, Subinterface Configuration

## ip rip authentication mode

**Description:** Specifies the type of authentication used on this interface. The **no** version removes authentication from the interface. Supported only in RIP version 2. Authentication is disabled by default.

**Syntax:** ip rip authentication mode { text | md5 *keyID* }

no ip rip authentication mode

- text – a simple text password is sent with each RIP message. If the password is not possessed by neighbors, the message is rejected.
- md5 – MD5 message-digest algorithms are used to encrypt and compress the RIP message.
- *keyID* – number identifying the MD5 key. Neighbors must share the MD5 key to decrypt the message and encrypt the response.

**Mode(s):** Interface Configuration, Subinterface Configuration

## ip rip receive version

**Description:** Restricts the RIP version that the system can receive on an interface. The **no** version sets the interface back to the default value, receiving both RIP version 1 and version 2.

**Syntax:** ip rip receive version { 1 | 2 | 1 2 | 2 1 | off }

no ip rip receive version

- 1 – specifies RIP version 1 only
- 2 – specifies RIP version 2 only
- 1 2 – specifies RIP version 1 and version 2
- 2 1 – specifies RIP version 2 and version 1
- off – turns reception off

**Mode(s):** Interface Configuration, Subinterface Configuration

## ip rip send version

**Description:** Restricts the RIP version that the system can send on an interface. The **no** version sets the interface back to the default value, sending only RIP version 1.

**Syntax:** ip rip send version { 1 | 2 | 1 2 | 2 1 | off }

no ip rip send version

- 1 – specifies RIP version 1 only
- 2 – specifies RIP version 2 only
- 1 2 – specifies RIP version 1 and version 2
- 2 1 – specifies RIP version 2 and version 1
- off – turns reception off

**Mode(s):** Interface Configuration, Subinterface Configuration

## ip route

**Description:** Establishes static routes. The **no** version removes static routes.

**Syntax:** ip route [ vrf *vrfName* ] *ipAddress ipMask*
{ *ipNextHop* [ *InterfaceType interfaceSpecifier* ] |
*interfaceType interfaceSpecifier* } [ *distance* ] [ tag *tagVal* ] [ permanent ]

no ip route [ vrf *vrfName* ] *ipAddress ipMask* [ *ipNextHop* | *interfaceType interfaceSpecifier* ] [ *distance* ]

- *vrfName* – name of the VRF if the static route is being established within a VRF context
- *ipAddress* – destination IP address
- *ipMask* – IP mask for the destination
- *ipNextHop* – IP address of the next hop that can be used to reach the destination network
- *interfaceType* – interface type; see *Interface Types and Specifiers* in *About This Guide*
- *interfaceSpecifier* – particular interface; format varies according to interface type; see *Interface Types and Specifiers* in *About This Guide.*
- *distance* – administrative distance metric for this route in the range 0–254
- *tagVal* – number in the range 0–255 that identifies the tag for this route
- permanent – specifies that the route will not be removed, even if the interface shuts down

**Mode(s):** Global Configuration

## ip router-id

**Description:** Establishes the IP address of a router. The **no** version removes the IP address assignment.

**Syntax:** [ no ] ip router-id [ *vrfName* ] *ipAddress*

- *vrfName* – name of the VRF; string of 1–32 alphanumeric characters
- *ipAddress* – IP address of the router

**Mode(s):** Global Configuration

## ip router isis

**Description:** Configures an IS-IS routing process for IP on an interface. The **no** version disables IS-IS for IP on the interface.

**Syntax:** [ no ] ip router isis [ *tag* ]

- *tag* – meaningful name for a routing process. If not specified, a null tag is assumed. The name must be unique among all IP router processes for a given router. Use the same text for the argument tag as specified in the **router isis** command.

**Mode(s):** Interface Configuration, Subinterface Configuration

## ip route-type

**Description:** Specifies whether BGP, IS-IS, OSPF, or RIP routes are available only for unicast forwarding, only for multicast reverse path forwarding checks, or for both. The **no** version restores the default value, **unicast** for BGP or **both** for IS-IS, OSPF, and RIP.

**Syntax:** For BGP:

ip route-type [ unicast | | both ]

no ip route-type

For IS-IS, OSPF, and RIP:

ip route-type [ unicast | multicast | both ]

no ip route-type

- unicast – specifies that routes for the protocol are available only for unicast forwarding
- multicast – specifies that routes for the protocol are available only for multicast route path forwarding checks; this option is not available for BGP
- both – specifies that routes for the protocol are available for both unicast forwarding and multicast route path forwarding checks

**Mode(s):** Router Configuration

## ip rpf-route

**Description:** Customizes static routes that the system can use to verify source addresses in multicast packets. The **no** version removes the static route.

**Syntax:** ip rpf-route *ipAddress addressMask* { *nextHopIpAddress* | *nextHopInterfaceType nextHopInterfaceSpecifier* } [ *distanceValue* ] [ tag *tagValue* ]

[ no ] rpf-route *ipAddress address-mask*

- *ipAddress* – IP address of the destination network
- *addressMask* – subnet mask for the destination network
- *nextHopIpAddress* – IP address of the next hop
- *nextHopInterfaceType* – interface type; see *Interface Types and Specifiers* in *About This Guide*
- *nextHopInterfaceSpecifier* – particular interface; format varies according to interface type; see *Interface Types and Specifiers* in *About This Guide*
- *distanceValue* – number in the range 0–255 that indicates the preference for this route
- *tagValue* – number in the range 0–4294967295 that identifies the route in the routing table

**Mode(s):** Global Configuration

## ip rsvp bandwidth

**Description:** Specifies the total bandwidth *reservable* on the interface in a non-ERX implementation. See the **mpls bandwidth** command for a complete description and syntax.

## ip sa-validate

**Description:** Enables source address validation on an interface. This feature verifies that a packet has been sent from a valid source address. When a packet arrives on an interface, the system performs a route-table lookup using the source address. The result from the route-table lookup is an interface to which packets destined for that address are routed. This interface must match the interface that the packet arrived on. If it does not match, the system drops the packet.The **no** version disables source address validation.

**Syntax:** **[** no **]** ip sa-validate

**Mode(s):** Interface Configuration

## ip share-interface

**Description:** Specifies the layer 2 interface used by a shared IP interface in the current virtual router. The **no** version removes the association between the layer 2 interface and the shared IP interface.

**Syntax:** ip share-interface *interfaceType interfaceSpecifier*

no ip share-interface

- *interfaceType* – interface type; see *Interface Types and Specifiers* in *About This Guide*
- *interfaceSpecifier* – particular interface; format varies according to interface type; see *Interface Types and Specifiers* in *About This Guide.*

**Mode(s):** Interface Configuration

## ip share-nexthop

**Description:** Specifies that the shared IP interface dynamically tracks a next hop for the specified destination. The **no** version halts tracking of the next hop.

**Syntax:** ip share-nexthop *ipAddress* [ virtual-router *vrName* ]

no ip share-nexthop

- *ipAddress* – IP address of the destination for which the next hop is tracked
- *vrName* – name of the virtual router for the next hop

**Mode(s):** Interface Configuration

## ip shutdown

**Description:** Shuts down an IP interface. The **no** version restarts the interface.

**Syntax:** [ no ] ip shutdown

**Mode(s):** Interface Configuration, Subinterface Configuration

## ip source-prefix

**Description:** Configures a subscriber interface to demultiplex traffic with the specified IP address and mask. The **no** version removes the association between the subscriber interface and the specified IP address and mask.

**Syntax:** [ no ] ip source-prefix *ipAddress ipAddressMask*

- *ipAddress* – IP address of the physical interface that receives messages for this subscriber
- *ipAddressMask* – network mask for associated IP subnet

**Mode(s):** Interface Configuration

## ip source-route

**Description:** Enables the forwarding of source-routed packets. The **no** version disables forwarding. Forwarding is enabled by default.

**Syntax:** [ no ] ip source-route [ *vrfName* ]

- *vrfName* – name of the VRF; string of 1–32 alphanumeric characters

**Mode(s):** Interface Configuration

## ip speed

**Description:** Sets the speed of an IP interface in bits per second. The **no** version restores the default value, 0 bps.

**Syntax:** [ no ] ip speed *adminSpeed*

- *adminSpeed* – speed of the interface in bps in the range 1–4294967295

**Mode(s):** Interface Configuration, Subinterface Configuration

## ip split-horizon

**Description:** Enables split horizon, preventing the RIP router from advertising routes from the interface originating the route, reducing the possibility of routing loops; this is the default condition. The **no** version disables split horizon.

**Syntax:** [ no ] ip split-horizon

**Mode(s):** Interface Configuration, Subinterface Configuration

## ip ssh authentication-retries

**Description:** Sets the number of times that a user can retry a failed authentication (such as trying to correct a wrong password) before the server terminates the connection. The **no** version restores the default value of 20 retries.

**Syntax:** ip ssh authentication-retries *retryLimit*

no ip ssh authentication-retries

- *retryLimit* – number of times authentication can be retried after the initial failure within a given connection attempt

**Mode(s):** Global Configuration

## ip ssh crypto

**Description:** Adds an encryption algorithm to the specified list of supported algorithms. The **no** version removes or excludes an algorithm from the specified list. The **default** version restores the default algorithms for the specified list.

**Syntax:** ip ssh crypto [ client-to-server | server-to-client ] [ no | default ] *cipherAlgorithm*

- client-to-server – adds the specified algorithm to the SSH server's list of supported inbound algorithms
- server-to-client – adds the specified algorithm to the SSH server's list of supported outbound algorithms
- no – removes or excludes the specified algorithm from the list
- default – restores the specified list to the factory defaults, which includes 3des-cbc, twofish-cbc, and blowfish-cbc
- *cipherAlgorithm* – algorithm to add to the list

**Mode(s):** Global Configuration

## ip ssh disable-user-authentication

**Description:** Disables RADIUS password authentication, resulting in the acceptance of all SSH clients that pass protocol negotiation.

**Syntax:** [ no ] ip ssh disable-user-authentication

**Mode(s):** Global Configuration

## ip ssh mac

**Description:** Adds a MAC algorithm to the specified list of supported algorithms. The **no** version removes or excludes an algorithm from the specified list. The **default** version restores the default algorithms for the specified list.

**Syntax:** ip ssh mac [ client-to-server | server-to-client ] [ no | default ] *macAlgorithm*

- client-to-server – adds the specified algorithm to the SSH server's list of supported inbound algorithms
- server-to-client – adds the specified algorithm to the SSH server's list of supported outbound algorithms
- no – removes or excludes the specified algorithm from the list
- default – restores the specified list to the factory defaults, which includes hmac-md5, hmac-sha1, and hmac-sha1-96
- *macAlgorithm* – algorithm to add to the list

**Mode(s):** Global Configuration

## ip ssh sleep

**Description:** Sets a sleep period in seconds for users that have exceeded the authentication retry limit. Connection attempts from the user at the same host are denied until this period expires. The **no** version restores the default value of 600 seconds.

**Syntax:** ip ssh sleep *sleepPeriod*

no ip ssh sleep

- *sleepPeriod* – period in seconds

**Mode(s):** Global Configuration

## ip ssh timeout

**Description:** Sets a timeout period in seconds. The SSH server terminates the connection if protocol negotiation—including user authentication—is not complete within this timeout. The **no** version restores the default value of 600 seconds.

**Syntax:** ip ssh timeout *timeout*

no ip ssh timeout

- *timeout* – period in the range 10–600 seconds

**Mode(s):** Global Configuration

## ip summary-address

**Description:** Summarizes specified addresses for RIP. The **no** version removes the summarization.

**Syntax:** ip summary-address [ rip ] *ipAddress ipAddressMask* [ metric ]

no ip summary-address [ rip ] *ipAddress ipAddressMask*

- rip – optional keyword for compatibility with non-ERX implementations
- *ipAddress* – IP address identifying the route to be summarized
- *ipAddressMask* – network mask identifying the route to be summarized
- metric – specifies a metric for the summary address; the default is 1

**Mode(s):** Router Configuration

## ip ttl

**Description:** Sets the hop count specified by the TTL field in the IP header used by IP for all operations unless overridden by another command. The **no** version restores the default value, 127.

**Syntax:** ip ttl [ *vrfName* ] *ttlValue*

no ip ttl [ *vrfName* ]

- *vrfName* – name of the VRF; string of 1–32 alphanumeric characters
- *ttlValue* – number in the range 1–255

**Mode(s):** Global Configuration

## ip tunnel reassembly

**Description:** Enables the reassembly of fragmented IP tunnel packets that are received on the current virtual router. The **no** version restores the default of disabled.

**Syntax:** [ no ] ip tunnel reassembly

**Mode(s):** Global Configuration

## ip unnumbered

**Description:** Enables IP processing on an interface without assigning an explicit IP address to the interface. The **no** version disables IP processing on the interface.

**Syntax:** [ no ] ip unnumbered *interfaceType interfaceSpecifier*

- *interfaceType* – interface type; see *Interface Types and Specifiers* in *About This Guide*
- *interfaceSpecifier* – particular interface; format varies according to interface type; see *Interface Types and Specifiers* in *About This Guide.*

**Mode(s):** Interface Configuration, Subinterface Configuration, Profile Configuration

## ip unreachables

**Description:** Enables the generation of an ICMP unreachable message when a packet is received that cannot be delivered by the router. The **no** version disables this function.

**Syntax:** [ no ] ip unreachables

**Mode(s):** Interface Configuration, Subinterface Configuration

## ip virtual-router

**Description:** Specifies a virtual router in an IP profile. Dynamic interfaces created with the profile are assigned to this VR. The **no** version removes the VR from the profile; if a VR is not specified via RADIUS, then any subsequent creation process for dynamic interfaces using the profile fails.

**Syntax:** [ no ] ip virtual-router *vrName*

- *vrName* – name of the virtual router; a string of 1–15 alphanumeric characters

**Mode(s):** Profile Configuration

## ip vpn-id

**Description:** Associates a VPN ID with the virtual router. The **no** version removes the VPN ID from the virtual router.

**Syntax:** ip vpn-id [ *vrfName* ] oui *ouiNumber* index *ipAddress*

no ip vpn-id [ *vrfName* ]

- *vrfName* – name of the VRF; string of 1–32 alphanumeric characters
- *ouiNumber* – identifies the OUI portion of the VPN ID, ranges from 0–16777215
- *ipAddress* – IP address that identifies the index portion of the VPN ID

**Mode(s):** Global Configuration

## ip vrf

**Description:** Creates a VRF or accesses VRF Configuration mode to configure a VRF. The **no** version deletes the VRF.

***Note:*** *After creating the VRF, you must configure a route distinguisher for it via the* ***rd*** *command; otherwise, the VRF will not operate.*

**Syntax:** [ no ] ip vrf *vrfName*

- *vrfName* – name of the VRF; a string of 1–32 alphanumeric characters

**Mode(s):** Global Configuration

## ip vrf forwarding

**Description:** Assigns a VRF to an interface or subinterface. The **no** version removes the assignment.

**Syntax:** [ no ] ip vrf forwarding *vrfName*

- *vrfName* – name of the VRF; a string of 1–32 alphanumeric characters

**Mode(s):** Global Configuration

## ip vrrp

**Description:** Creates a VRRP instance ID. The **no** version removes a VRID. The default is disabled.

**Syntax:** [ no ] ip vrrp *vrid*

- *vrid* – VRID identifier; a number in the range 1–255

**Mode(s):** Interface Configuration

## ip vrrp advertise-interval

**Description:** Configures the VRRP advertisement interval time. You must use seconds to comply with RFC 2338. Use milliseconds only if all VRRP instances peering for the given VRID are composed of ERX systems. The **no** version restores the default value of 1 second.

**Syntax:** ip vrrp *vrid* advertise-interval *advertiseInterval* [ seconds | milliseconds ]

no ip vrrp *vrid* advertise-interval

- *vrid* – VRID identifier; a number in the range 1–255
- *advertiseInterval* – the advertisement period in seconds or milliseconds; 1–255 seconds; 100–255000 milliseconds
- seconds – specify interval in seconds
- milliseconds – specify interval in milliseconds

**Mode(s):** Interface Configuration

## ip vrrp authentication-key

**Description:** Specify the authentication key. This command is only valid if the **text** keyword was selected in the **ip vrrp authentication-type** command. The **no** version negates the command or restores the default.

**Syntax:** ip vrrp *vrid* authentication-key *key*

no ip vrrp *vrid* authentication-key

- *vrid* – VRID identifier; a number in the range 1–255
- *key* – string of 1–8 characters

**Mode(s):** Interface Configuration

## ip vrrp authentication-type

**Description:** Specifies the VRRP authentication type. The **no** version restores the default value, none.

**Syntax:** ip vrrp *vrid* authentication-type { none | text }

no ip vrrp *vrid* authentication-type

- *vrid* – VRID identifier; a number in the range 1–255
- none – authentication disabled
- text – simple text password

**Mode(s):** Interface Configuration

## ip vrrp enable

**Description:** Enables a VRID. The **no** version disables a VRID. The default is disabled.

**Syntax:** [ no ] ip vrrp *vrid* [ enable ]

- *vrid* – VRID identifier; a number in the range 1–255

**Mode(s):** Interface Configuration

## ip vrrp preempt

**Description:** Enables VRRP preemption. The **no** version disables VRRP preemption. The default is enabled.

**Syntax:** [ no ] ip vrrp *vrid* preempt

- *vrid* – VRID identifier; a number in the range 1–255

**Mode(s):** Interface Configuration

## ip vrrp priority

**Description:** Configures the priority of VRRP routers. The **no** version restores the default value, 100.

**Syntax:** ip vrrp *vrid* priority *priorityValue*

no ip vrrp *vrid* priority

- *vrid* – VRID identifier; a number in the range 1–255
- *priorityValue* – priority value of the VRRP router; a number in the range 1–255; default is 100

**Mode(s):** Interface Configuration

## ip vrrp virtual-address

**Description:** Associates an IP address to a VRID. The **no** version removes a list of IP addresses associated with a VRID. The **no** version clears the auto flag, if auto addresses are being used. There is no default.

**Syntax:** ip vrrp *vrid* virtual-address
{ auto | ipAddress *ipAddress* [ ipAddress *ipAddress* ]* }

no ip vrrp *vrid* virtual-address [ ipAddress *ipAddress* ]*

- *vrid* – VRID identifier; a number in the range 1–255
- *ipAddress* – the IP address that associates to the VRID

**Mode(s):** Interface Configuration

## ipsec clear sa

**Description:** Refreshes ISAKMP/IKE or IPSec SAs. There is no **no** version.

**Syntax:** ipsec clear sa { all [ state *tunnelState* ] | tunnel *tunnelName* } [ phase {1 | 2 } ]

- all – reinitializes all SAs
- state – reinitializes SAs on tunnels that are in a specific state
- *tunnelState* – state of tunnel, up, down, not-present
- tunnel – specifies that an SA on a specific tunnel is to be reinitialized
- *tunnelName* – name of tunnel
- phase – specifies one of the following types of tunnel to be reinitialized:
  - 1 – ISAKMP/IKE tunnels
  - 2 – IPSec tunnels

**Mode(s):** Global Configuration

## ipsec isakmp-policy-rule

**Description:** Defines and prioritizes an ISAKMP/IKE policy. ISAKMP/IKE policies define parameters to be used during ISAKMP/IKE negotiation. You can have up to 10 ISAKMP/IKE policies per system. The **no** version removes a policy. If you do not include a priority number with the **no** version, the software removes all ISAKMP/IKE policies.

**Syntax:** ipsec isakmp-policy-rule *priority*

no ipsec isakmp-policy-rule [ *priority* ]

- *priority* – identifies and prioritizes the ISAKMP/IKE policy; the range is 1–10000, with 1 having the highest priority

**Mode(s):** Global Configuration

## ipsec key manual

**Description:** Specifies that a peer use a manual key for authentication and displays the prompt from which you can enter manual keys. Manually configured keys are used during the tunnel establishment phase when the ISAKMP/IKE policy specifies either preshared key authentication or encrypted nonce authentication. The **no** version deletes a manually configured key.

***Note:*** *You must enter this command in the virtual router context where the IP address of the peer is defined.*

**Syntax:** [ no ] ipsec key manual { pre-share } *ipAddress*

- pre-share – specifies preshared manual keys as the authentication method
- *ipAddress* – address of the peer for which the key can be used

**Mode(s):** Global Configuration

## ipsec lifetime

**Description:** Specifies the default lifetime in volume of traffic and/or seconds. The default lifetime applies to secure tunnels that do not have a tunnel lifetime defined. When either the volume of traffic or number of seconds limit is reached, IPSec renegotiates the SA. The **no** version restores the default values.

**Syntax:** [ no ] ipsec lifetime { kilobytes *kilobytes* | seconds *seconds* }

- *kilobytes* – volume of traffic in kilobytes that can pass between IPSec peers before the SA expires; the range is 102400–4294967295; the default is 4294967295 kilobytes; a setting of zero turns off the kilobyte lifetime
- *seconds* – number of seconds an SA lives before expiring; the range is 7200–4294967295; the default is 28800 seconds (8 hours)

**Mode(s):** Global Configuration

## ipsec local-endpoint

**Description:** Defines a default local endpoint used for ISAKMP/IKE negotiations and all IPSec tunnels for a transport virtual router. The **no** version restores the default settings of the local endpoint.

**Syntax:** [ no ] ipsec local-endpoint *ipAddress* transport-virtual-router *transportVRName*

- *ipAddress* – IP address to use as the local endpoint
- *transportVRName* – name of transport virtual router in which the IP address is defined

**Mode(s):** Global Configuration

## ipsec transform-set

**Description:** Creates a transform set. Transform sets used for manually configured tunnels can have only one transform. Transform sets used for signaled tunnels can have up to six transforms. Transforms are numbered in a priority sequence in the order in which you enter them. The **no** version deletes the transform set.

Each transform provides a different combination of data authentication and confidentiality. Use the online Help to view available transforms.

**Syntax:** ipsec transform-set *transformSetName transform0* [ *transform1* [ t*ransform2* [ *transform3* [ *transform4* [ *transform5* ] ] ] ] ]

no ipsec transform-set *transformSetName*

- *transformSetName* – name of the transform set
- *transform0* through *transform5* – AH or ESP transform

**Mode(s):** Global Configuration

## isis authentication-key

**Description:** Assigns a password for IS-IS level 1 and level 2 hellos used by neighboring routers that are using IS-IS password authentication. The **no** version deletes the password.

**Syntax:** isis authentication-key [ level-1 | level-2 ] *authKey*

no isis authentication-key [ level-1 | level-2 ]

- level-1 – inserts the password into level 1 hello packets
- level-2 – inserts the password into level 2 hello packets
- *authKey* – a password, a continuous string of characters up to 8 characters in length

**Mode(s):** Interface Configuration, Subinterface Configuration

## isis circuit-type

**Description:** Use to configure the type of adjacency desired for the specified interface. The **no** version resets the circuit type to level 1 and level 2.

**Syntax:** isis circuit-type [ level-1 | level-1-2 | level-2-only ]

no isis circuit-type

- level-1 – establishes a level 1 adjacency if there is at least one area address in common between this system and its neighbors
- level-1-2 – (default) establishes a level 1 and 2 adjacency if the neighbor is also configured as a level 1-2 router and there is at least one area in common. If there is no area in common, a level 2 adjacency is established.
- level-2-only – establishes a level 2 adjacency on the circuit. If the neighboring router is a level 1 only router, no adjacency will be established.

**Mode(s):** Interface Configuration, Subinterface Configuration

## isis csnp-interval

**Description:** Configures the IS-IS CSNP interval for the specified interface. The **no** version restores the default value.

**Syntax:** isis csnp-interval *seconds* [ level-1 | level-2 ]

no isis csnp-interval [ *seconds* ] [ level-1 | level-2 ]

- *seconds* – number in the range 0–65535; the interval of time in seconds between the transmission of CSNPs on multiaccess networks for the designated router; default is 10 seconds, except for WAN interfaces, where the default is 0
- level-1 – sets the interval of time between transmission of CSNPs for level 1 independently
- level-2 – sets the interval of time between transmission of CSNPs for level 2 independently

**Mode(s):** Interface Configuration, Subinterface Configuration

## isis hello-interval

**Description:** Specifies the length of time in seconds between hello packets that the router sends on the specified interface. The **no** version restores the default value.

**Syntax:** isis hello-interval *seconds* [ level-1 | level-2 ]

no isis hello-interval [ *seconds* ] [ level-1 | level-2 ]

- *seconds* – number in the range 0–65535; a value equal to the *hello multiplier* times the *hello interval seconds* is advertised as the *holdtime* in the hello packets transmitted; the default is 10 seconds. The value must be the same for all routers attached to a common network. With smaller hello intervals, topological changes are detected faster, but there is more routing traffic.
- level-1 – sets the *hello-interval* for level 1 independently
- level-2 – sets the *hello-interval* for level 2 independently

**Mode(s):** Interface Configuration, Subinterface Configuration

## isis hello-multiplier

**Description:** Specifies the number of IS-IS hello packets a neighbor must miss before the router should declare the adjacency to be down. The **no** version restores the *multiplier* default value of 3.

**Syntax:** isis hello-multiplier *multiplier* [ level-1 | level-2 ]

no isis hello-multiplier [ *multiplier* | level-1 | level-2 ]

- *multiplier* – number in the range 3–1000; the default is 3. The advertised hold time in IS-IS hellos will be set to the *hello-multiplier* times the *hello-interval*. Neighbors will declare an adjacency to this router to be down after not having received any IS-IS hellos during the advertised hold time. The hold time (and thus the *hello-multiplier* and the *hello-interval*) can be set on a per interface basis, and can be different between different routers in one area. Using a smaller *hello-multiplier* will give fast convergence, but can result in more routing instability. Increment the *hello-multiplier* to a larger value to help network stability when needed. Never configure a *hello-multiplier* lower than the default.
- level-1 – sets the hello-multiplier independently for level 1 adjacencies
- level-2 – sets the hello-multiplier independently for level 2 adjacencies

**Mode(s):** Interface Configuration, Subinterface Configuration

## isis lsp-interval

**Description:** Configures the time delay between successive IS-IS link state packet transmissions. The **no** version restores the default value of 33 milliseconds.

**Syntax:** isis lsp-interval *milliseconds*

no isis lsp-interval

- *milliseconds* – number of milliseconds in the range 1–4294967295; an interval between successive link state packets

**Mode(s):** Interface Configuration, Subinterface Configuration

## isis mesh-group

**Description:** Configures an interface in the same mesh group to act as a virtual multiaccess network. The **no** version disables the feature.

**Syntax:** isis mesh-group { blocked | *number* }

no isis mesh-group

- blocked – blocks reserved LSPs from being flooded out on this defined configured interface
- *number* – mesh group number in the range 1–4294967295

**Mode(s):** Interface Configuration, Subinterface Configuration

## isis message-digest-key

**Description:** Specifies an HMAC MD5 key that the system uses to create a secure, encrypted message digest of IS-IS level 1 or level 2 hello packets on the interface. Level 1 packets are the default. The digest is inserted into the packet from which it is created. Using this algorithm protects against intrusion by preventing unauthorized routers from forming adjacencies with your system.

You can specify when the system will start (default is the current time) and stop (default is never) accepting packets that include a digest made with this key. You can specify when the system will start (default is the current time plus 2 minutes) and stop (default is never) generating packets that include a digest made with this key. The **no** version deletes the key specified by the key-id.

**Syntax:** isis message-digest-key *keyId* hmac-md5 *key*
[ start-accept *startAcceptTime* [ { *startAcceptMonth startAcceptDay* | *startAcceptDay startAcceptMonth* } *startAcceptYear* ] ]
[ start-generate *startGenTime* [ { *startGenMonth startGenDay* | *startGenDay startGenMonth* } *startGenYear* ] ]
[stop-accept { never | *stopAcceptTime* [ { *stopAcceptMonth stopAcceptDay* | *stopAcceptDay stopAcceptMonth* } *stopAcceptYear* ] } ]

[ stop-generate { never | *stopGenTime* [ { *stopGenMonth stopGenDay* | *stopGenDay stopGenMonth* } *stopGenYear* ] } ]
[ level-1 | level-2 ]

no isis message-digest-key *keyId* [ level 1 | level 2 ]

- *keyId* – integer from 1 to 255 that is a unique identifier for the secret key, sent with the message digest in the packet.
- *key* – string of up to 20 alphanumeric characters; secret key used by the HMAC MD5 algorithm to generate the message digest.
- *startAcceptTime, startAcceptMonth, startAcceptDay, startAcceptYear* – time, month, day, year that the system will start accepting packets created with this password. Use military time format *HH*:*MM*[ :*SS* ].
- *startGenTime, startGenMonth, startGenDay, startGenYear* – time, month, day, year that the system will start inserting this password into packets. Use military time format *HH*:*MM*[ :*SS* ].
- never – the system never stops accepting or generating packets; overrides previously specified stop times.
- *stopAcceptTime, stopAcceptMonth, stopAcceptDay, stopAcceptYear* – time, month, day, year that the system will stop accepting packets created with this password. Use military time format *HH*:*MM*[ :*SS* ].
- *stopGenTime, stopGenMonth, stopGenDay, stopGenYear* – time, month, day, year that the system will stop inserting this password into packets. Use military time format *HH*:*MM*[ :*SS* ].
- level1 – Inserts the password into level 1 hello packets
- level2 – Inserts the password into level 2 hello packets

**Mode(s):** Interface Configuration, Subinterface Configuration

## isis metric

**Description:** Configures the metric (cost) for the specified interface. The **no** version restores the default metric value.

**Syntax:** isis metric *defaultMetric* [ level-1 | level-2 ]

no isis metric [ *defaultMetric* | level-1 | level-2 ]

- *defaultMetric* – metric used for the redistributed route; a number in the range 0–63 if the system is configured with the **metric-style narrow** command; a number in the range 0–16777215 if the system is configured with the **metric-style transition** or **metric-style wide** command; the default value is 10
- level-1 – apply metric to level 1 links
- level-2 – apply metric to level 2 links

**Mode(s):** Interface Configuration, Subinterface Configuration

## isis priority

**Description:** Configures the priority of this system for designated router election. The **no** version resets priority to the default value, 64.

**Syntax:** isis priority *value* [ level-1 | level-2 ]

no isis priority [ *value* | level-1 | level-2 ]

- *value* – number in the range 0–127; the priority of a router; the default value is 64
- level-1 – sets the priority of a router for level 1 independently
- level-2 – sets priority of a router for level 2 independently

**Mode(s):** Interface Configuration, Subinterface Configuration

## isis retransmit-interval

**Description:** Configures the number of seconds between retransmission of LSPs with the same lsp-id for point-to-point links. The **no** version restores the default value.

**Syntax:** isis retransmit-interval *seconds*

no isis retransmit-interval

- *seconds* – number of seconds in the range 1–65535; the default value is 5. The number should be greater than the expected round-trip delay between any two routers on the attached network. The setting of this parameter should be conservative, or needless retransmission will result. The value should be larger for serial lines.

**Mode(s):** Interface Configuration, Subinterface Configuration

## isis retransmit-throttle-interval

**Description:** Configures the amount of time between retransmissions of any IS-IS LSPs on a point-to-point interface. The **no** version restores the default value, 33 milliseconds.

**Syntax:** isis retransmit-throttle-interval *milliseconds*

no isis retransmit-throttle-interval

- *milliseconds* – the number of milliseconds in the range 0–65535; the minimum delay between LSP retransmissions on the interface

**Mode(s):** Interface Configuration, Subinterface Configuration

## is-type

**Description:** Configures the IS-IS level at which the router is to operate. The **no** version resets the parameter to the default level-1-2.

**Syntax:** is-type { level-1 | level-1-2 | level-2-only }

no is-type

- level-1 – causes the router to act as a station router
- level-1-2 – causes the router to act as both a station router and an area router; the default setting
- level-2-only – causes the router to act as an area router

**Mode(s):** Router Configuration