ip

Configure the following features.

- IP Address pools dynamic NAT and PPTP. See “pptp” on page 16-113.
- Static NAT
- Dual static NAT (based on RFC 2663)
- Dynamic NAT
- Static route
- Activate or deactivate IP routing

Use the **no** option to delete a route or an address pool. Figure 16-1 and Figure 16-2 describe networks for basic static NAT, dynamic NAT, and dual static NAT used in the examples.

Figure 16-1 Static and Dynamic NAT

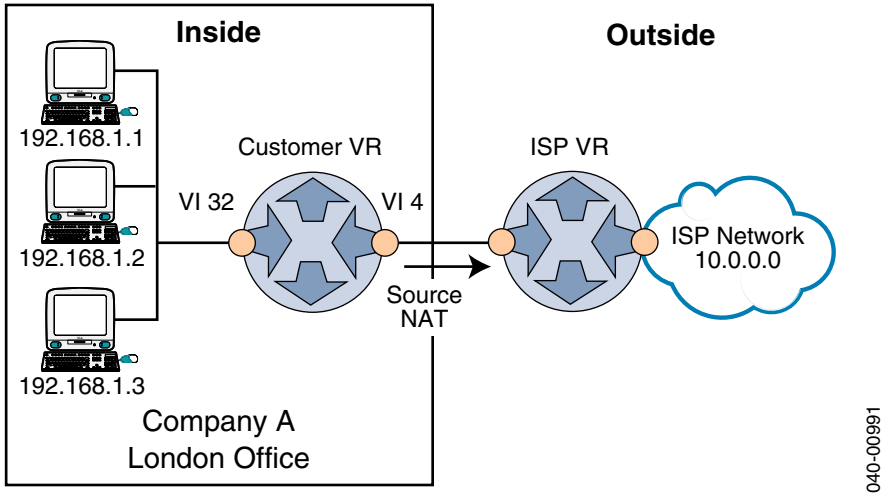

Figure 16-2 Dual Static NAT

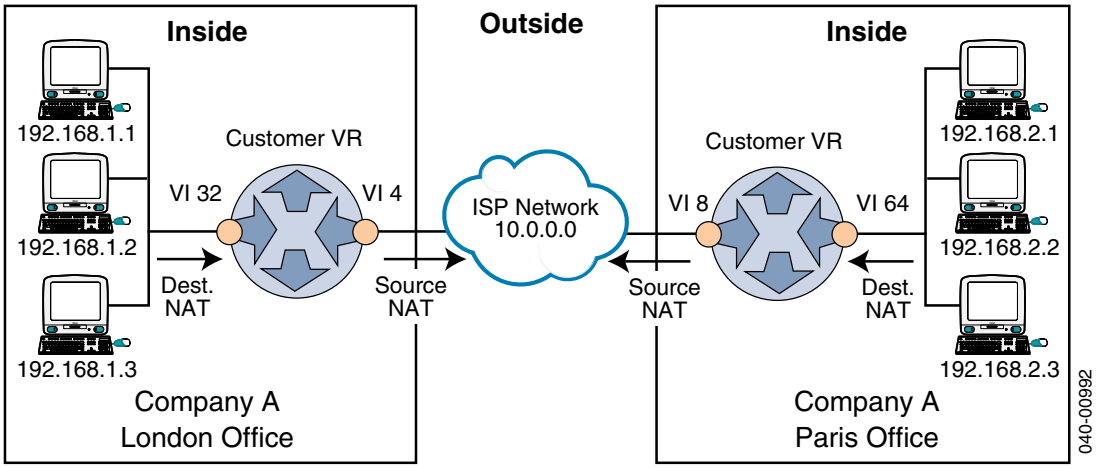

Table 16-10 describes the parameters available.

Table 16-10 IP Address Routing Commands

Routing Command	*Description*
IP local pool	Create a PPTP address pool. The command has the following parameters: • Name—The index number identifying the address pool. • Start-range—The first available IP address in the pool. • End-range—The final address in the pool (up to 256).
IP nat inside source static	Create a source static NAT to the VR or a VI. It translates a source IP address to a publicly known IP address. Add a subnet mask to retain the host part. The command has the following parameters: • Local IP address—The source IP address of the internal node. • Global IP address—The global IP address used for external communication. • Netmask—A network address mask. (Optional) • VI—Virtual Interface (Optional)
IP nat inside destination static	Create a destination static NAT to the VR or VI. . It translates a destination IP address to a publicly known IP address. Used with **ip nat inside source static**, the combination protects both the source and destination by using public IP addresses in their place. The command has the following parameters: • Global IP address—The destination IP address • Local IP address—The new public IP address. • Netmask—A network address mask. (Optional) • VI—Virtual Interface (Optional)

Table 16-10 IP Address Routing Commands (Continued)

Routing Command	*Description*
IP nat list	Attach a dynamic NAT pool to an access-list. See "access-list" on page 16-5. The command has the following parameters: • List ID—Access list • Pool ID—NAT address pool to which the access-list looks for the next available IP address for translation. • VI ID—VI to watch for inbound translatable packets. Must be the same VI in the access-list table.
IP nat pool	Create a dynamic NAT address pool. The IP address range is 0–255. The port ID range is 1–2048. The maximum number of IP addresses and ports for a single NAT pool is 100,000. The command has the following parameters: • Pool ID—index number identifying the address pool. • Start-address—The first available IP address in the pool. • End-address—The final address in the pool (the maximum size of an address pool is 255 IP addresses, or a class-C subnet). • Start IP port—The first available port for the NAT pool. • End IP port—The last available port for the NAT pool.
IP route	Set up a static route between a VI and a specific destination network. The command has the following parameters: • Prefix—The IP address of the destination network. • Mask—The IP subnet mask of the destination network. • Next-hop—The IP address of the next hop toward the destination. • VI <index>—The index number of the VI. • Metric—The metric value for the best path to a destination. Choose from -1 (wildcard) up to 2,147,483,648. • Tag—A marker to distinguish two static routes pointing to the same destination. Possible values are 1 to 256. *Note:* To set up a default route, set the IP address and the subnet mask to all zeroes.
IP routing	The **ip routing** command enables or disables IP routing on the VR.

Syntax	Example

Create local address pool for PPTP:
`[NO] IP LOCAL POOL <name> <start-range> <end-range>`
	`IPNOS: CONFIG VPN(50) VR(10.10.10.1)# ip local pool cosine 192.168.24.1 192.168.24.32`
	`VR(10.10.10.1): Object successfully set`
	`POOL cosine: Object successfully set`
	
	`IPNOS: CONFIG VPN(50) VR(10.10.10.1)# dis local`
	
	`LOCAL POOL(s):`
	`NAME | START-RANGE | END-RANGE`
	`cosine | 192.168.24.1 | 192.168.24.32`

Create source static NAT with a subnet to a VR:
`IP NAT INSIDE SOURCE STATIC <local-ip> <global-ip> [NETMASK <mask>] [VI <viid>]`
	`IPNOS: CONFIG VPN(50) VR(10.10.10.1)# ip nat inside source static 192.168.1.1 10.1.1.1`
	`VR(10.10.10.1): Object successfully set`
	`STATIC TRANSLATION (192.168.1.1 10.1.1.1 * *): Object successfully set`
	
	`IPNOS: CONFIG VPN(50) VR(10.10.10.1)# dis nat`
	
	`POOL(s):`
	`List is empty`
	
	`IP STATIC NAT(s):`
	`INSIDE LOCAL IP | INSIDE GLOBAL IP| NETMASK | VI | TYPE`
	`192.168.1.1 | 10.1.1.1 | * | * | SOURCE`
	
	`IP NAT TRANSLATION(s):`
	`List is empty`

Create source static NAT to a VI:
`IP NAT INSIDE SOURCE STATIC <local-ip> <global-ip> [NETMASK <mask>] [VI <viid>]`
	`IPNOS: CONFIG VPN(50) VR(10.10.10.1)# ip nat inside source static 192.168.1.2 10.1.1.2 vi 4`
	`VR(10.10.10.1): Object successfully set`
	`STATIC TRANSLATION (192.168.1.2 10.1.1.2 * 4): Object successfully set`
	
	`IPNOS: CONFIG VPN(50) VR(10.10.10.1)# dis nat`
	
	`POOL(s):`
	`List is empty`
	
	`IP STATIC NAT(s):`
	`INSIDE LOCAL IP | INSIDE GLOBAL IP| NETMASK | VI | TYPE`
	`192.168.1.1 | 10.1.1.1 | * | * | SOURCE`
	`192.168.1.2 | 10.1.1.2 | * | 4 | SOURCE`
	
	`IP NAT TRANSLATION(s):`
	`List is empty`

Syntax	Example
Create source static NAT with a subnet to a VI: IP NAT INSIDE SOURCE STATIC <local-ip> <global-ip> [NETMASK <mask>] [VI <viid>]	IPNOS: CONFIG VPN(50) VR(10.10.10.1)# **ip nat inside source static 192.168.1.0 10.1.1.0 netmask 255.255.255.0 vi 4** VR(10.10.10.1): Object successfully set STATIC TRANSLATION (192.168.1.0 10.1.1.0 255.255.255.0 4): Object successfully set IPNOS: CONFIG VPN(50) VR(10.10.10.1)# **dis nat** POOL(s): List is empty IP STATIC NAT(s): INSIDE LOCAL IP | INSIDE GLOBAL IP| NETMASK | VI | TYPE 192.168.1.0 | 10.1.1.0 | 255.255.255.0 | 4 | SOURCE IP NAT TRANSLATION(s): List is empty
Delete source static NAT: NO IP NAT INSIDE SOURCE STATIC <local-ip> <global-ip> [VI <viId>]	IPNOS: CONFIG VPN(50) VR(10.10.10.1)# **no ip nat inside source static 192.168.1.1 10.1.1.1** VR(10.10.10.1): Object successfully set STATIC TRANSLATION (192.168.1.1 10.1.1.1): Object successfully deleted IPNOS: CONFIG VPN(50) VR(10.10.10.1)# **dis nat** POOL(s): List is empty IP STATIC NAT(s): INSIDE LOCAL IP | INSIDE GLOBAL IP| NETMASK | VI | TYPE 192.168.1.2 | 10.1.1.2 | * | 4 | SOURCE IP NAT TRANSLATION(s): List is empty

Syntax	Example
Create destination static NAT with a subnet to a VI: IP NAT INSIDE DESTINATION STATIC <global-ip> <local-ip> [NETMASK <mask>] [VI <viid>]	IPNOS: CONFIG VPN(50) VR(10.10.10.1)# **ip nat inside destination static 192.168.2.0 10.1.2.0 netmask 255.255.255.0 vi 32** VR(10.10.10.1): Object successfully set STATIC TRANSLATION (192.168.2.0 10.1.2.0 255.255.255.0 32): Object successfully set IPNOS: CONFIG VPN(50) VR(10.10.10.1)# **dis nat** POOL(s): List is empty IP STATIC NAT(s): INSIDE LOCAL IP | INSIDE GLOBAL IP| NETMASK | VI | TYPE 192.168.1.0 | 10.1.1.0 | 255.255.255.0 | 4 | SOURCE 10.1.2.0 | 192.168.2.0 | 255.255.255.0 | 32 | DESTINATION IP NAT TRANSLATION(s): List is empty
Delete destination static NAT: NO IP NAT INSIDE DESTINATION STATIC <global-ip> <local-ip> [VI <viId>]	IPNOS: CONFIG VPN(50) VR(10.10.10.1)# **no ip nat inside destination static 192.168.2.0 10.1.2.0 vi 32** VR(10.10.10.1): Object successfully set STATIC TRANSLATION (192.168.2.0 10.1.2.0 32): Object successfully deleted IPNOS: CONFIG VPN(50) VR(10.10.10.1)# **dis nat** POOL(s): List is empty IP STATIC NAT(s): INSIDE LOCAL IP | INSIDE GLOBAL IP| NETMASK | VI | TYPE 192.168.1.0 | 10.1.1.0 | 255.255.255.0 | 4 | SOURCE IP NAT TRANSLATION(s): List is empty

Syntax	Example
Create destination static NAT to a VR: [NO] IP NAT INSIDE DESTINATION STATIC <global-ip> <local-ip> [VI <viid>]	IPNOS: CONFIG VPN(50) VR(10.10.10.1)# **ip nat inside destination static 192.168.2.1 10.1.2.1** VR(10.10.10.1): Object successfully set STATIC TRANSLATION (192.168.2.1 10.1.2.1 * *): Object successfully set IPNOS: CONFIG VPN(50) VR(10.10.10.1)# **dis nat** POOL(s): List is empty IP STATIC NAT(s): INSIDE LOCAL IP | INSIDE GLOBAL IP| NETMASK | VI | TYPE 10.1.2.1 | 192.168.2.1 | * | * | DESTINATION IP NAT TRANSLATION(s): List is empty
Create destination static NAT to a VI: [NO] IP NAT INSIDE DESTINATION STATIC <global-ip> <local-ip> [VI <viid>]	IPNOS: CONFIG VPN(50) VR(10.10.10.1)# **ip nat inside destination static 192.168.2.2 10.1.2.2 vi 32** VR(10.10.10.1): Object successfully set STATIC TRANSLATION (192.168.2.2 10.1.2.2 * 32): Object successfully set IPNOS: CONFIG VPN(50) VR(10.10.10.1)# **dis nat** POOL(s): List is empty IP STATIC NAT(s): INSIDE LOCAL IP | INSIDE GLOBAL IP| NETMASK | VI | TYPE 10.1.2.1 | 192.168.2.1 | * | * | DESTINATION 10.1.2.2 | 192.168.2.2 | * | 32 | DESTINATION IP NAT TRANSLATION(s): List is empty

Syntax	Example
For NAT access lists: [NO] IP NAT LIST <listId> POOL <poolId> <viId>	IPNOS : CONFIG VPN(50) VR(10.1.1.1)# **access-list 1 permit 10.1.1.2 10.1.100.100 1** VR(10.1.1.1): Object successfully set ACCESS-LIST 1: Object successfully set IPNOS : CONFIG VPN(50) VR(10.1.1.1)# **ip nat list 1 pool 1 1** VR(10.1.1.1): Object successfully set LIST 1: Object successfully set IPNOS : CONFIG VPN(50) VR(10.1.1.1)# **Note:** You need to create a NAT access list using the **access-list** command before creating a NAT list.
Create dynamic NAT pool: [NO] IP NAT POOL <poolId> <start-address> <end-address> <start-port> <end-port>	IPNOS: CONFIG VPN(50) VR(10.10.10.1)# **access-list 1 permit 192.168.1.1 192.168.1.3 32** VR(10.10.10.1): Object successfully set ACCESS-LIST 1: Object successfully set IPNOS: CONFIG VPN(50) VR(10.10.10.1)# **dis access** ACCESS-LIST(s): ACCESS-LIST | VI | START-ADDRESS | END-ADDRESS | POOL 1 | 32 | 192.168.1.1 | 192.168.1.3 | 0 IPNOS: CONFIG VPN(50) VR(10.10.10.1)# **ip nat pool 1 10.1.1.1 10.1.1.10 1000 1050** VR(10.10.10.1): Object successfully set POOL 1: Object successfully set IPNOS: CONFIG VPN(50) VR(10.10.10.1)# **dis nat** POOL(s): POOL | START-ADDRESS | END-ADDRESS | START-PORT | END-PORT 1 | 10.1.1.1 | 10.1.1.10 | 1000 | 1050 IP STATIC NAT(s): List is empty IP NAT TRANSLATION(s): List is empty

Syntax	Example
Delete dynamic NAT pool: [NO] IP NAT POOL <poolId> <start-address> <end-address> <start-port> <end-port>	IPNOS: CONFIG VPN(50) VR(10.10.10.1)# **no ip nat pool 1 10.1.1.1 10.1.1.10 1000 1050** VR(10.10.10.1): Object successfully set POOL 1: Object successfully deleted IPNOS: CONFIG VPN(50) VR(10.10.10.1)# **dis nat** POOL(s): List is empty IP STATIC NAT(s): List is empty IP NAT TRANSLATION(s): List is empty
Attach NAT pool to access list: [NO] IP NAT LIST <listId> POOL <poolId> <viId>	IPNOS: CONFIG VPN(50) VR(10.10.10.1)# **dis access** ACCESS-LIST(s): ACCESS-LIST | VI | START-ADDRESS | END-ADDRESS | POOL 1 | 32 | 192.168.1.1 | 192.168.1.3 | 0 IPNOS: CONFIG VPN(50) VR(10.10.10.1)# **dis nat** POOL(s): POOL | START-ADDRESS | END-ADDRESS | START-PORT | END-PORT 1 | 10.1.1.1 | 10.1.1.10 | 1000 | 1050 IP STATIC NAT(s): List is empty IP NAT TRANSLATION(s): List is empty IPNOS: CONFIG VPN(50) VR(10.10.10.1)# i**p nat list 1 pool 1 32** VR(10.10.10.1): Object successfully set LIST 1: Object successfully set IPNOS: CONFIG VPN(50) VR(10.10.10.1)# **dis access** ACCESS-LIST(s): ACCESS-LIST | VI | START-ADDRESS | END-ADDRESS | POOL 1 | 32 | 192.168.1.1 | 192.168.1.3 | 1

Syntax	Example
For static routes: [NO] IP ROUTE <prefix> <mask> <next-hop> [VI <interface>] [<metric> [<tag>]]	IPNOS : CONFIG VPN(50) VR(10.1.1.1)# **ip route 1.1.1.1 255.255.255.0 1 5** VR(10.1.1.1): Object successfully set 1.1.1.1|255.255.255.0|0.0.0.0: Object successfully set IPNOS : CONFIG VPN(50) VR(10.1.1.1)# IPNOS : CONFIG VPN(50) VR(10.1.1.1)# **ip route 1.1.1.0 255.255.255.0 2.2.2.2** VR(10.1.1.1): Object successfully set 1.1.1.0|255.255.255.0|2.2.2.2: Object successfully set IPNOS : CONFIG VPN(50) VR(10.1.1.1)#
For two static routes pointing to the same destination: [NO] IP ROUTE <prefix> <mask> <next-hop> [VI <interface>] [<metric> [<tag>]]	IPNOS : CONFIG VPN(50) VR(10.1.1.1)# **ip route 1.1.1.1 255.255.255.0 1 5 10** VR(10.1.1.1): Object successfully set 1.1.1.1|255.255.255.0|0.0.0.0: Object successfully set IPNOS : CONFIG VPN(50) VR(10.1.1.1)# **ip route 1.1.1.1 255.255.255.0 2 20 12** VR(10.1.1.1): Object successfully set 1.1.1.1|255.255.255.0|0.0.0.0: Object successfully set
For enabling/disabling IP routing on the VR: [NO] IP ROUTING	IPNOS : CONFIG VPN(50) VR(10.1.1.1)# **ip routing** IPNOS : CONFIG VPN(50) VR(10.1.1.1)# **display routing** ROUTING : IP ROUTING

Default Value

IP routing is enabled.

Command Level

VR

Related Commands

access-list
pptp