单接口网络地址转换

目录

简介

单接口网络地址转换 (NAT) 是什么意思？术语“单接口”通常意味着一个任务使用一个路由器的单个物理接口。正如我们可以使用同一个物理接口的子接口执行交换机间链路(ISL) 中继一样，我们也可以在路由器上使用单个物理接口，以完成NAT。

注意：由于环回接口，路由器必须处理每个数据包。这将降低路由器的性能。

先决条件

要求

本文档没有任何特定的要求。

使用的组件

此功能要求您使用支持NAT的Cisco IOS®软件版本。使用 Cisco Feature Navigator II（仅限注册用户）可确定哪些 IOS 版本支持此功能。

规则

有关文档规则的详细信息，请参阅 Cisco 技术提示规则。

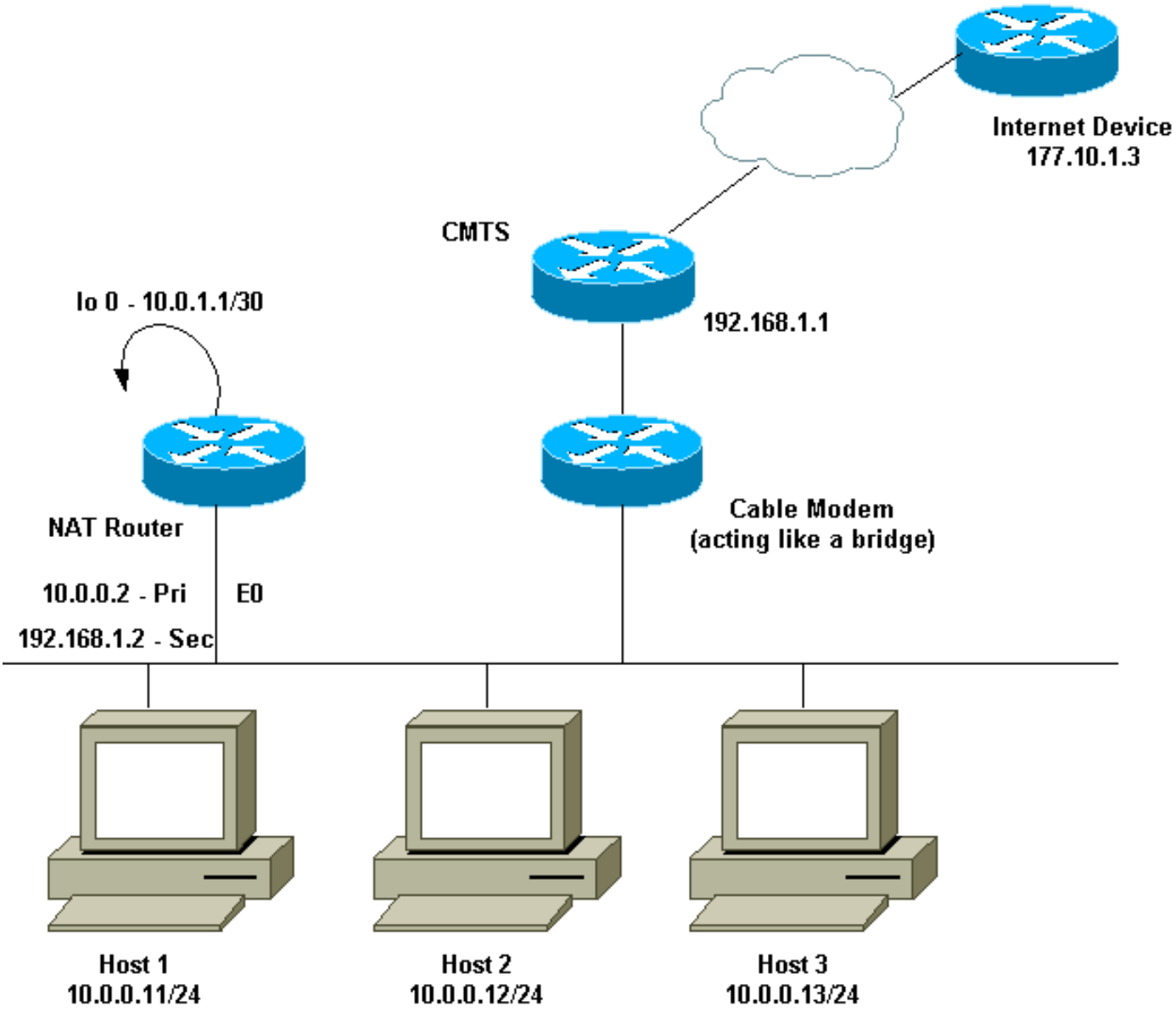
背景信息

为了让NAT发生作用，信息包必须从NAT“内部”定义接口切换到NAT"外部"定义接口，反之亦然。NAT 的要求并未更改，但本文档将演示您如何使用虚拟接口（也称为环回接口）和基于策略的路由使 NAT 在具有单个物理接口的路由器上工作。

对单接口 NAT 的需求非常少见。实际上，本文中的示例可能是需要此配置的唯一的情况。虽然用户在与NAT一起使用应用策略路由时会出现其他情况，但我们不把这种情况当作单接口NAT，因为这些实例使用一个以上的物理接口。

示例 1 网络图和配置

网络图



上面的网络图在电缆调制解调器设置中很常见。有线调制解调器终端系统（CMTS）是路由器，而有线调制解调器是类似网桥的设备。我们面临的问题是网络服务提供商(ISP)未给我们提供足够的有效地址，供需要到达互联网的主机数量使用。ISP为我们提供了地址192.168.1.2，这个地址将用于设备。在收到进一步的请求时，我们收到另外三个地址（192.168.2.1 到 192.168.2.3），这三个地址是 NAT 将 10.0.0.0/24 范围内的主机转换到的地址。

要求

我们的要求是：

- 网络上的所有主机都必须可以访问 Internet。
- 主机2一定能够带IP地址192.168.2.1的互联网上到达 。
- 由于我们的主机数量多于合法地址，我们使用 10.0.0.0/24 子网来进行内部编址。

出于本文目的，我们只显示NAT路由器配置。但是，我们确实会提到一些与主机有关的重要配置说明。

NAT 路由器配置

NAT 路由器配置

```
interface Loopback0
 ip address 10.0.1.1 255.255.255.252
 ip nat outside
!--- Creates a virtual interface called Loopback 0 and
assigns an !--- IP address of 10.0.1.1 to it. Defines
interface Loopback 0 as !--- NAT outside. ! ! interface
Ethernet0 ip address 192.168.1.2 255.255.255.0 secondary
ip address 10.0.0.2 255.255.255.0 ip Nat inside !---
Assigns a primary IP address of 10.0.0.2 and a secondary
IP !--- address of 192.168.1.2 to Ethernet 0. Defines
interface Ethernet 0 !--- as NAT inside. The 192.168.1.2
address will be used to communicate !--- through the CM
to the CMTS and the Internet. The 10.0.0.2 address !---
will be used to communicate with the local hosts. ip
policy route-map Nat-loop !--- Assigns route-map "Nat-
loop" to Ethernet 0 for policy routing. ! ip Nat pool
external 192.168.2.2 192.168.2.3 prefix-length 29 ip Nat
inside source list 10 pool external overload ip Nat
inside source static 10.0.0.12 192.168.2.1 !--- NAT is
defined: packets that match access-list 10 will be !---
translated to an address from the pool called
"external". !--- A static NAT translation is defined for
10.0.0.12 to be !--- translated to 192.168.2.1 (this is
for host 2 which needs !--- to be accessed from the
Internet).


ip classless
!
!
ip route 0.0.0.0 0.0.0.0 192.168.1.1
ip route 192.168.2.0 255.255.255.0 Ethernet0
 !--- Static default route set as 192.168.1.1, also a
static !--- route for network 192.168.2.0/24 directly
attached to !--- Ethernet 0 ! ! access-list 10 permit
10.0.0.0 0.0.0.255 !--- Access-list 10 defined for use
by NAT statement above.


access-list 102 permit ip any 192.168.2.0 0.0.0.255
access-list 102 permit ip 10.0.0.0 0.0.0.255 any
!--- Access-list 102 defined and used by route-map "Nat-
loop" !--- which is used for policy routing.

!
Access-list 177 permit icmp any any
 !--- Access-list 177 used for debug.

!
route-map Nat-loop permit 10
 match ip address 102
 set ip next-hop 10.0.1.2
!--- Creates route-map "Nat-loop" used for policy
routing. !--- Route map states that any packets that
match access-list 102 will !--- have the next hop set to
10.0.1.2 and be routed "out" the !--- loopback
interface. All other packets will be routed normally. !-
-- We use 10.0.1.2 because this next-hop is seen as
located !--- on the loopback interface which would
result in policy routing to !--- loopback0.
Alternatively, we could have used "set interface !---
loopback0" which would have done the same thing. ! end
```

```
NAT-router#
```

注意：所有主机的默认网关都设置为10.0.0.2，即NAT路由器。ISP 和 CMTS 必须具有到 192.168.2.0/29（指向 NAT 路由器）的路由以使返回数据流可以正常工作，因为来自内部主机的数据流显示为来自此子网。在本示例中，CMTS 会将发往 192.168.2.0/29 的数据流路由到 NAT 路由器上配置的备用 IP 地址 192.168.1.2。

示例 1 show 及 debug 命令输出

本部分提供的信息可帮助您确认您的配置是否可正常运行。

为了演示上述配置可以发挥作用，我们在监控 NAT 路由器上的 **debug 输出的同时，运行了几个 ping 测试。**您能发现 **ping 命令是成功的，并且调试输出正确显示了正在发生的情况。**

注意：在使用debug命**令之**前，请参阅有关debug命令的重要信息。

测试一

在第一次测试中，我们从实验室定义的Internet中的设备**ping**主机2。请记住，其中一个要求是Internet中的设备必须能够与IP地址为192.168.2.1的主机2通信。以下是NAT路由器上的**debug**输出。在 NAT 路由器上运行的 **debug 命令包括使用定义的 access-list 177** 的 debug ip packet 177 详细信息，**debug ip Nat** 和为我们显示策略路由信息包的 **debug ip 策略。**

以下是在 NAT 路由器上执行的 **show ip Nat translation 命令的输出：**

```
NAT-router# show ip Nat translation
Pro Inside global      Inside local       Outside local      Outside global
--- 192.168.2.1        10.0.0.12          ---                ---
NAT-router#
```

我们成功地从 internet 上的设备（在本示例中为路由器）**ping 通 192.168.2.1，如下所示：**

```
Internet-device# ping 192.168.2.1

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 192.168.2.1, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 92/92/92 ms
Internet-device#
```

要查看在 NAT 路由器中发生了什么情况，请参阅以下 **debug 输出和注释：**

```
IP: s=177.10.1.3 (Ethernet0), d=192.168.2.1, len 100, policy match
    ICMP type=8, code=0
IP: route map Nat-loop, item 10, permit
IP: s=177.10.1.3 (Ethernet0), d=192.168.2.1 (Loopback0), Len 100, policy routed
    ICMP type=8, code=0
!--- The above debug output shows the packet with source 177.10.1.3 destined !--- to
192.168.2.1. The packet matches the statements in the "Nat-loop" !--- policy route map and is
permitted and policy-routed. The Internet !--- Control Message Protocol (ICMP) type 8, code 0
indicates that this !--- packet is an ICMP echo request packet.


IP: Ethernet0 to Loopback0 10.0.1.2
```

```
IP: s=177.10.1.3 (Ethernet0), d=192.168.2.1 (Loopback0), g=10.0.1.2, Len 100,
forward
    ICMP type=8, code=0
 !--- The packet now is routed to the new next hop address of 10.0.1.2 !--- as shown above. IP:
NAT enab = 1 trans = 0 flags = 0 NAT: s=177.10.1.3, d=192.168.2.1->10.0.0.12 [52] IP:
s=177.10.1.3 (Loopback0), d=10.0.0.12 (Ethernet0), g=10.0.0.12, Len 100, forward ICMP type=8,
code=0 IP: NAT enab = 1 trans = 0 flags = 0 !--- Now that the routing decision has been made,
NAT takes place. We can !--- see above that the address 192.168.2.1 is translated to 10.0.0.12
and !--- this packet is forwarded out Ethernet 0 to the local host. !--- Note: When a packet is
going from inside to outside, it is routed and !--- then translated (NAT). In the opposite
direction (outside to inside), !--- NAT takes place first.


IP: s=10.0.0.12 (Ethernet0), d=177.10.1.3, Len 100, policy match
    ICMP type=0, code=0
IP: route map Nat-loop, item 10, permit
IP: s=10.0.0.12 (Ethernet0), d=177.10.1.3 (Loopback0), Len 100, policy routed
    ICMP type=0, code=0
IP: Ethernet0 to Loopback0 10.0.1.2
!--- Host 2 now sends an ICMP echo response, seen as ICMP type 0, code 0. !--- This packet also
matches the policy routing statements and is !--- permitted for policy routing. NAT:
s=10.0.0.12->192.168.2.1, d=177.10.1.3 [52] IP: s=192.168.2.1 (Ethernet0), d=177.10.1.3
(Loopback0), g=10.0.1.2, Len 100, forward ICMP type=0, code=0 IP: s=192.168.2.1 (Loopback0),
d=177.10.1.3 (Ethernet0), g=192.168.1.1, Len 100, forward ICMP type=0, code=0 IP: NAT enab = 1
trans = 0 flags = 0 !--- The above output shows the Host 2 IP address is translated to !---
192.168.2.1 and the packet that results packet is sent out loopback 0, !--- because of the
policy based routing, and finally forwarded !--- out Ethernet 0 to the Internet device. !--- The
remainder of the debug output shown is a repeat of the previous !--- for each of the additional
four ICMP packet exchanges (by default, !--- five ICMP packets are sent when pinging from Cisco
routers). We have !--- omitted most of the output since it is redundant.

IP: s=177.10.1.3 (Ethernet0), d=192.168.2.1, Len 100, policy match
    ICMP type=8, code=0
IP: route map Nat-loop, item 10, permit
IP: s=177.10.1.3 (Ethernet0), d=192.168.2.1 (Loopback0), Len 100, policy routed
    ICMP type=8, code=0
IP: Ethernet0 to Loopback0 10.0.1.2
IP: s=177.10.1.3 (Ethernet0), d=192.168.2.1 (Loopback0), g=10.0.1.2, Len 100,
forward
    ICMP type=8, code=0
IP: NAT enab = 1 trans = 0 flags = 0
NAT: s=177.10.1.3, d=192.168.2.1->10.0.0.12 [53]
IP: s=177.10.1.3 (Loopback0), d=10.0.0.12 (Ethernet0), g=10.0.0.12, Len 100,
forward
    ICMP type=8, code=0
IP: NAT enab = 1 trans = 0 flags = 0
IP: s=10.0.0.12 (Ethernet0), d=177.10.1.3, Len 100, policy match
    ICMP type=0, code=0
IP: route map Nat-loop, item 10, permit
IP: s=10.0.0.12 (Ethernet0), d=177.10.1.3 (Loopback0), Len 100, policy routed
    ICMP type=0, code=0
IP: Ethernet0 to Loopback0 10.0.1.2
NAT: s=10.0.0.12->192.168.2.1, d=177.10.1.3 [53]
IP: s=192.168.2.1 (Ethernet0), d=177.10.1.3 (Loopback0), g=10.0.1.2, Len 100,
forward
    ICMP type=0, code=0
IP: s=192.168.2.1 (Loopback0), d=177.10.1.3 (Ethernet0), g=192.168.1.1, Len 100,
forward
    ICMP type=0, code=0
IP: NAT enab = 1 trans = 0 flags = 0
```

测试二

另一个要求是允许主机能与互联网联络。在本测试中，我们从Host 1对Internet设备执行ping操作。结果为show 和debug 命令如下。

最初，NAT 路由器中的 NAT 转换表如下：

```
NAT-router# show ip Nat translation
Pro Inside global      Inside local       Outside local      Outside global
--- 192.168.2.1        10.0.0.12          ---                ---
NAT-router#
```

我们从主机 1 发出 ping 后，可以看到：

```
Host-1# ping 177.10.1.3

Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 177.10.1.3, timeout is 2 seconds:
!!!!!
Success rate is 100 percent (5/5), round-trip min/avg/max = 92/92/96 ms
Host-1#
```

我们看到上面的 ping 成功了。现在，NAT 路由器中的 NAT 表如下所示：

```
NAT-router# show ip Nat translation
Pro Inside global      Inside local       Outside local      Outside global
icmp 192.168.2.2:434   10.0.0.11:434      177.10.1.3:434     177.10.1.3:434
icmp 192.168.2.2:435   10.0.0.11:435      177.10.1.3:435     177.10.1.3:435
icmp 192.168.2.2:436   10.0.0.11:436      177.10.1.3:436     177.10.1.3:436
icmp 192.168.2.2:437   10.0.0.11:437      177.10.1.3:437     177.10.1.3:437
icmp 192.168.2.2:438   10.0.0.11:438      177.10.1.3:438     177.10.1.3:438
--- 192.168.2.1        10.0.0.12          ---                ---
NAT-router#
```

现在上面的NAT转换表显示了由动态NAT配置(与静态NAT配置相对)引起的其他转换。

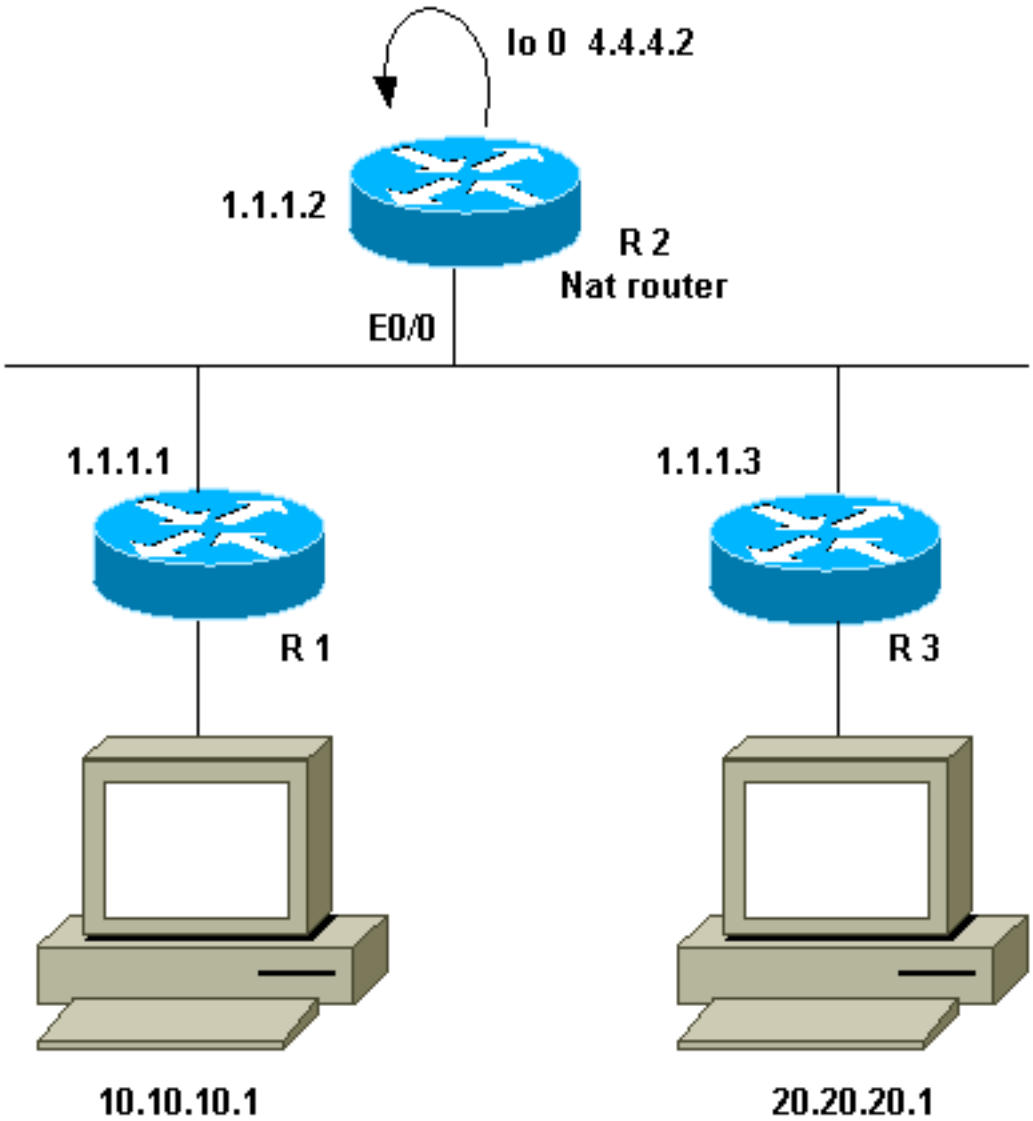
下面的 debug 输出显示 NAT 路由器上发生的情况。

```
IP: NAT enab = 1 trans = 0 flags = 0
IP: s=10.0.0.11 (Ethernet0), d=177.10.1.3, Len 100, policy match
    ICMP type=8, code=0
IP: route map Nat-loop, item 10, permit
IP: s=10.0.0.11 (Ethernet0), d=177.10.1.3 (Loopback0), Len 100, policy routed
    ICMP type=8, code=0
IP: Ethernet0 to Loopback0 10.0.1.2
!--- The above output shows the ICMP echo request packet originated by !--- Host 1 which is
policy-routed out the loopback interface. NAT: s=10.0.0.11->192.168.2.2, d=177.10.1.3 [8] IP:
s=192.168.2.2 (Ethernet0), d=177.10.1.3 (Loopback0), g=10.0.1.2, Len 100, forward ICMP type=8,
code=0 IP: s=192.168.2.2 (Loopback0), d=177.10.1.3 (Ethernet0), g=192.168.1.1, Len 100, forward
ICMP type=8, code=0 IP: NAT enab = 1 trans = 0 flags = 0 !--- After the routing decision has
been made by the policy routing, !--- translation takes place, which translates the Host 1 IP
address of 10.0.0.11 !--- to an address from the "external" pool 192.168.2.2 as shown above. !--
- The packet is then forwarded out loopback 0 and finally out Ethernet 0 !--- to the Internet
device. IP: s=177.10.1.3 (Ethernet0), d=192.168.2.2, Len 100, policy match ICMP type=0, code=0
IP: route map Nat-loop, item 10, permit IP: s=177.10.1.3 (Ethernet0), d=192.168.2.2 (Loopback0),
Len 100, policy routed ICMP type=0, code=0 IP: Ethernet0 to Loopback0 10.0.1.2 IP: s=177.10.1.3
(Ethernet0), d=192.168.2.2 (Loopback0), g=10.0.1.2, Len 100, forward ICMP type=0, code=0 !---
```

The Internet device sends an ICMP echo response which matches our !--- policy, is policy-routed, and forward out the Loopback 0 interface. IP: NAT enab = 1 trans = 0 flags = 0 NAT: s=177.10.1.3, d=192.168.2.2->10.0.0.11 [8] IP: s=177.10.1.3 (Loopback0), d=10.0.0.11 (Ethernet0), g=10.0.0.11, Len 100, forward ICMP type=0, code=0 *!--- The packet is looped back into the loopback interface at which point !--- the destination portion of the address is translated from 192.168.2.2 !--- to 10.0.0.11 and forwarded out the Ethernet 0 interface to the local host. !--- The ICMP exchange is repeated for the rest of the ICMP packets, some of !--- which are shown below.* IP: NAT enab = 1 trans = 0 flags = 0 IP: s=10.0.0.11 (Ethernet0), d=177.10.1.3, Len 100, policy match ICMP type=8, code=0 IP: route map Nat-loop, item 10, permit IP: s=10.0.0.11 (Ethernet0), d=177.10.1.3 (Loopback0), Len 100, policy routed ICMP type=8, code=0 IP: Ethernet0 to Loopback0 10.0.1.2 NAT: s=10.0.0.11->192.168.2.2, d=177.10.1.3 [9] IP: s=192.168.2.2 (Ethernet0), d=177.10.1.3 (Loopback0), g=10.0.1.2, Len 100, forward ICMP type=8, code=0 IP: s=192.168.2.2 (Loopback0), d=177.10.1.3 (Ethernet0), g=192.168.1.1, Len 100, forward ICMP type=8, code=0 IP: NAT enab = 1 trans = 0 flags = 0 IP: s=177.10.1.3 (Ethernet0), d=192.168.2.2, Len 100, policy match ICMP type=0, code=0 IP: route map Nat-loop, item 10, permit IP: s=177.10.1.3 (Ethernet0), d=192.168.2.2 (Loopback0), Len 100, policy routed ICMP type=0, code=0 IP: Ethernet0 to Loopback0 10.0.1.2 IP: s=177.10.1.3 (Ethernet0), d=192.168.2.2 (Loopback0), g=10.0.1.2, Len 100, forward ICMP type=0, code=0 IP: NAT enab = 1 trans = 0 flags = 0 NAT: s=177.10.1.3, d=192.168.2.2->10.0.0.11 [9] IP: s=177.10.1.3 (Loopback0), d=10.0.0.11 (Ethernet0), g=10.0.0.11, Len 100, forward ICMP type=0, code=0

示例 2 网络图和配置

网络图



要求

我们希望两个站点（R1 和 R3）后的某些设备进行通信。这两个站点使用未注册的 IP 地址，因此当它们互相通信时，我们必须转换地址。在本例中，主机10.10.10.1被转换为200.200.200.1，主机20.20.20.1将被转换为100.100.1。因此，我们需要在两个方向上进行转换。出于记帐目的，这两个站点之间的流量必须通过R2。总之，我们的要求是：

- R1 后的主机 10.10.10.1 需要使用全局地址与 R3 后的主机 20.20.20.1 进行通信。
- 这些主机之间的数据流必须通过 R2 发送。
- 对于我们的示例，我们需要静态 NAT 转换，如下面的配置所示。

NAT 路由器配置

NAT 路由器配置

```
interface Loopback0
 ip address 4.4.4.2 255.255.255.0
 ip Nat inside
!--- Creates a virtual interface called "loopback 0" and
assigns IP address !--- 4.4.4.2 to it. Also defines for
it a NAT inside interface. ! Interface Ethernet0/0 ip
address 1.1.1.2 255.255.255.0 no ip redirects ip Nat
outside ip policy route-map Nat !--- Assigns IP address
1.1.1.1/24 to e0/0. Disables redirects so that packets
!--- which arrive from R1 destined toward R3 are not
redirected to R3 and !--- visa-versa. Defines the
interface as NAT outside interface. Assigns !--- route-
map "Nat" used for policy-based routing. ! ip Nat inside
source static 10.10.10.1 200.200.200.1 !--- Creates a
static translation so packets received on the inside
interface !--- with a source address of 10.10.10.1 will
have their source address !--- translated to
200.200.200.1. Note: This implies that the packets
received !--- on the outside interface with a
destination address of 200.200.200.1 !--- will have the
destination translated to 10.10.10.1.


ip Nat outside source static 20.20.20.1 100.100.100.1
!--- Creates a static translation so packets received on
the outside interface !--- with a source address of
20.20.20.1 will have their source address !---
translated to 100.100.100.1. Note: This implies that
packets received on !--- the inside interface with a
destination address of 100.100.100.1 will !--- have the
destination translated to 20.20.20.1.


ip route 10.10.10.0 255.255.255.0 1.1.1.1
ip route 20.20.20.0 255.255.255.0 1.1.1.3
ip route 100.100.100.0 255.255.255.0 1.1.1.3
!
access-list 101 permit ip host 10.10.10.1 host
100.100.100.1
route-map Nat permit 10
 match ip address 101
 set ip next-hop 4.4.4.2
```

示例 2 show 及 debug 命令输出

注意：输出解释程序工具支持某些show命令，它允许您查看对show命令输出的分析。使用 debug 命令之前，请参阅**有关 Debug 命令的重要信息。**

测试一

如上配置所示，我们有 2 个静态 NAT 转换，可以在 R2 上使用 show ip nat translation 命令看到。

以下是在 NAT 路由器上执行的 **show ip Nat translation 命令的输出：**

```
NAT-router# show ip Nat translation
Pro Inside global      Inside local       Outside local      Outside global
--- ---                ---                100.100.100.1      20.20.20.1
--- 200.200.200.1      10.10.10.1         ---                ---
R2#
```

在本测试中，我们从R1后面的设备(10.10.10.1)发出**ping**，该设备发往R3后面的设备(100.100.100.1)的全局地址。在R2上运行**debug ip Nat**和**debug ip packet**，导致以下输出：

```
IP: NAT enab = 1 trans = 0 flags = 0
IP: s=10.10.10.1 (Ethernet0/0), d=100.100.100.1, Len 100, policy match
    ICMP type=8, code=0
IP: route map Nat, item 10, permit
IP: s=10.10.10.1 (Ethernet0/0), d=100.100.100.1 (Loopback0), Len 100, policy
routed
    ICMP type=8, code=0
IP: Ethernet0/0 to Loopback0 4.4.4.2
!--- The above output shows the packet source from 10.10.10.1 destined !--- for 100.100.100.1
arrives on E0/0, which is defined as a NAT !--- outside interface. There is not any NAT that
needs to take place at !--- this point, however the router also has policy routing enabled for
!--- E0/0. The output shows that the packet matches the policy that is !--- defined in the
policy routing statements. IP: s=10.10.10.1 (Ethernet0/0), d=100.100.100.1 (Loopback0),
g=4.4.4.2, Len 100, forward ICMP type=8, code=0 IP: NAT enab = 1 trans = 0 flags = 0 !--- The
above now shows the packet is policy-routed out the loopback0 !--- interface. Remember the
loopback is defined as a NAT inside interface. NAT: s=10.10.10.1->200.200.200.1, d=100.100.100.1
[26] NAT: s=200.200.200.1, d=100.100.100.1->20.20.20.1 [26] !--- For the above output, the
packet is now arriving on the loopback0 !--- interface. Since this is a NAT inside interface, it
is important to !--- note that before the translation shown above takes place, the router !---
will look for a route in the routing table to the destination, which !--- before the translation
is still 100.100.100.1. Once this route look up !--- is complete, the router will continue with
translation, as shown above. !--- The route lookup is not shown in the debug output.

IP: s=200.200.200.1 (Loopback0), d=20.20.20.1 (Ethernet0/0), g=1.1.1.3, Len 100,
forward
    ICMP type=8, code=0
IP: NAT enab = 1 trans = 0 flags = 0
!--- The above output shows the resulting translated packet that results is !--- forwarded out
E0/0.
```

以下是作为从路由器 3 后的设备发往路由器 1 后的设备的响应数据包的结果输出：

```
NAT: s=20.20.20.1->100.100.100.1, d=200.200.200.1 [26]
NAT: s=100.100.100.1, d=200.200.200.1->10.10.10.1 [26]
 !--- The return packet arrives into the e0/0 interface which is a NAT !--- outside interface.
In this direction (outside to inside), translation !--- occurs before routing. The above output
```

```
shows the translation takes place. IP: s=100.100.100.1 (Ethernet0/0), d=10.10.10.1
(Ethernet0/0), Len 100, policy rejected -- normal forwarding ICMP type=0, code=0 IP:
s=100.100.100.1 (Ethernet0/0), d=10.10.10.1 (Ethernet0/0), g=1.1.1.1, Len 100, forward ICMP
type=0, code=0 !--- The E0/0 interface still has policy routing enabled, so the packet is !---
check against the policy, as shown above. The packet does not match the !--- policy and is
forwarded normally.
```

摘要

本文展示了如何使用NAT和基于策略的路由创建“单接口NAT”情景。重要的是记住此配置在运行NAT的路由器上会降低性能，因为信息包可能通过路由器进行进程交换。

相关信息

- NAT 支持页
- 技术支持 - Cisco Systems