



基于策略的路由

本章介绍如何配置思科 ASA 以支持基于策略的路由 (PBR)。以下部分介绍基于策略的路由、PBR 准则和 PBR 配置。

- [关于基于策略的路由，第 1 页](#)
- [基于策略的路由指南，第 3 页](#)
- [配置基于策略的路由，第 4 页](#)
- [基于策略的路由示例，第 7 页](#)
- [基于策略的路由的历史记录，第 14 页](#)

关于基于策略的路由

传统路由是以目标为基础的，这意味着数据包基于目标 IP 地址进行路由。但是，在基于目标的路由系统中更改特定流量的路由是较为困难的。使用基于策略的路由 (PBR)，您可以基于非目标网络的条件定义路由 - 通过 PBR，可以基于源地址、源端口、目标地址、目标端口、协议或所有这些的组合来路由流量。

基于策略的路由：

- 用于为差分流量提供服务质量 (QoS)。
- 用于跨低带宽、低成本的永久路径以及高带宽、高成本的交换路径分发交互式 and 批处理流量。
- 允许互联网运营商及其他组织通过明确定义的网络连接来路由源自各组用户的流量。

基于策略的路由通过在网络边缘对流量进行分类和标记，然后在整个网络中使用 PBR 沿着特定路径路由标记的流量，来实施 QoS。这样，可以将源自不同源的数据包路由至不同网络，甚至在目标不同时亦可以；并且在将多个私有网络互连时，这一点可能很有用。

为什么使用基于策略的路由？

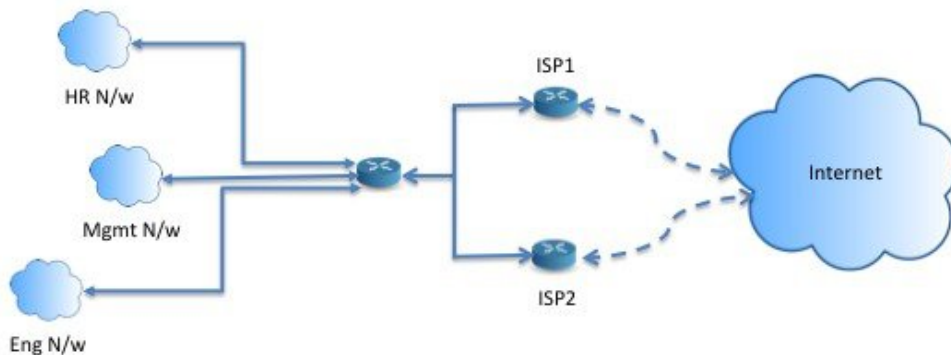
假设一家公司在不同位置之间有两条链路：一条是高带宽、低延迟、较为昂贵的链路，而另一条是低带宽、高延迟、不太昂贵的链路。使用传统路由协议时，高带宽链路将基于通过该链路的带宽和/或延迟（使用 EIGRP 或 OSPF）特性所实现的指标节约而获得大部分（如果不是全部）跨该链路发

送的流量。通过PBR，您可以通过高带宽/低延迟的链路来路由优先级较高的流量，而通过低带宽/高延迟链路发送其他所有流量。

基于策略的路由的部分应用为：

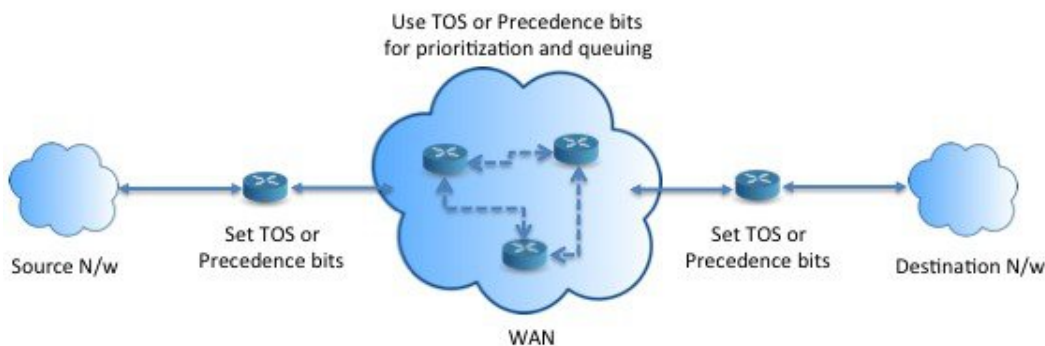
同等访问权限和源敏感路由

在此拓扑中，来自人力资源网络和管理网络的流量可配置为通过ISP-1，来自工程网络的流量可配置为通过ISP-2。因此，基于策略的路由支持网络管理员提供同等访问权限和源敏感路由，如下所示。



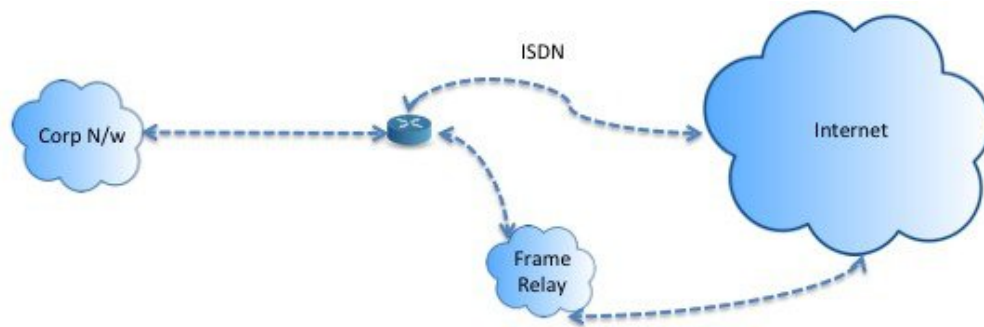
服务质量

通过标记使用基于策略的路由的数据包，网络管理员可以在网络边界对各种服务级别的网络流量进行分类，然后使用优先级、自定义或加权公平排队（如下图所示）在网络核心中实施这些服务级别。此设置无需在主干网络核心中的每个WAN接口对流量进行明确分类，从而能够提升网络性能。



成本节约

组织可以通过定义拓扑，将与特定活动关联的批处理流量定向为在短时间内使用较高带宽的高成本链路，并将较低带宽的低成本链路上的基本连接继续用于交互式流量，如下所示。



负载分担

除 ECMP 负载均衡提供的动态负载共享功能外，网络管理员现在还可以实施策略来根据流量特征在多个路径之间分发流量。

例如，在同等访问和基于源的路由场景所描绘的拓扑中，管理员可以配置基于策略的路由来对从人力资源网络至 ISP1 的流量和从工程网络至 ISP2 的流量进行负载共享。

实施 PBR

ASA 使用 ACL 来匹配流量，然后对流量执行路由操作。具体而言，配置指定用于进行匹配的 ACL 的路由映射，然后为该流量指定一个或多个操作。最后，将路由映射与接口相关联，在该接口上要所有传入流量应用 PBR。

基于策略的路由指南

防火墙模式

仅在路由防火墙模式下受支持。不支持透明防火墙模式。

每数据流路由

由于 ASA 基于每个数据流执行路由，所以会在第一个数据包上应用策略路由，并将生成的路由决策存储在为该数据包创建的数据流中。属于同一连接的所有后续包将简单地与此数据流匹配并正确进行路由。

未对输出路由查询应用的 PBR 策略

基于策略的路由是一种仅入口功能；也就是说，它仅会应用于新传入连接的第一个数据包，并在此时选择连接转发支路的出口接口。请注意，如果传入数据包属于现有连接或已应用 NAT，则不会触发 PBR。

集群

- 支持集群。

- 在集群情景下，没有静态或动态路由，已启用 `ip-verify-reverse` 路径，非对称流量可能会被丢弃。因此，建议禁用 `ip-verify-reverse` 路径。

IPv6 支持

支持 IPv6

其他准则

- 所有现有路由映射相关的配置限制和局限性都将继续适用。
- 请勿将包含匹配策略列表的路由映射用于基于策略的路由。`match policy-list` 仅用于 BGP。

配置基于策略的路由

路由映射由一个或多个路由映射语句组成。每个语句都有序列号以及 `permit` 或 `deny` 子句。每个 `route-map` 语句都包含 `match` 和 `set` 命令。`match` 命令表示要对数据包应用的匹配条件。`set` 命令表示要对数据包采取的操作。

- 在路由映射同时配置有 IPv4 和 IPv6 `match/set` 子句时或在使用了与 IPv4 和 IPv6 流量匹配的统一 ACL 时，将根据目标 IP 版本应用 `set` 操作。
- 当多个下一跳或接口被配置为 `set` 操作时，系统将逐个评估所有选项，直到找到有效的可用选项。在已配置的多个选项之间将不进行负载均衡。
- `Verify-availability` 选项不支持多情景模式。

过程

步骤 1 定义标准或扩展访问列表：

```
access-list name standard {permit | deny} {any4 | host ip_address | ip_address mask}
```

```
access-list name extended {permit | deny} protocol source_and_destination_arguments
```

示例：

```
ciscoasa(config)# access-list testacl extended permit ip
10.1.1.0 255.255.255.0 10.2.2.0 255.255.255.0
```

如果使用标准 ACL，则仅基于目标地址进行匹配。如果使用扩展 ACL，可基于源、目标或两者进行匹配。

对于扩展 ACL，可以指定 IPv4、IPv6、身份防火墙或思科 TrustSec 参数。有关完整语法，请参阅 ASA 命令参考。

步骤 2 创建路由映射条目：

```
route-map name {permit | deny} [sequence_number]
```

示例:

```
ciscoasa(config)# route-map testmap permit 12
```

路由映射条目按顺序读取。可使用 *sequence_number* 参数标识顺序，否则 ASA 将使用添加路由映射条目的顺序。

此外，ACL 还包括自己的 *permit* 和 *deny* 语句。对于路由映射与 ACL 之间的 *Permit/Permit* 匹配，继续执行基于策略的路由处理。对于 *Permit/Deny* 匹配，对此路由映射的处理结束并检查其他路由映射。如果结果仍是 *Permit/Deny*，则使用普通路由表。对于 *Deny/Deny* 匹配，继续基于策略的路由处理。

注释 如果配置的路由映射不含 *permit* 或 *deny* 操作且不含序列号，则默认假定操作为 *permit*，序列号为 10。

步骤 3 使用访问列表定义要应用的匹配条件:

```
match ip address access-list_name [access-list_name...]
```

示例:

```
ciscoasa(config-route-map)# match ip address testacl
```

步骤 4 配置一个或多个 *set* 操作:

- 设置下一跳地址:

```
set {ip | ipv6} next-hop ipv4_or_ipv6_address
```

您可以配置多个下一跳 IP 地址，在这种情况下将按指定顺序对它们进行评估，直到找到有效的可路由下一跳 IP 地址。所配置的下一跳应为直连式，否则不会应用 *set* 操作。

- 设置默认下一跳地址:

```
set {ip | ipv6} default next-hop ipv4_or_ipv6_address
```

如果匹配流量的正常路由查询失败，则 ASA 会使用此指定的下一跳 IP 地址转发流量。

- 设置递归下一跳 IPv4 地址:

```
set ip next-hop recursive ip_address
```

set ip next-hop 和 **set ip default next-hop** 都要求可在直连式子网中找到下一跳。如果使用 **set ip next-hop recursive**，则下一跳地址不需要是直连式。匹配流量不会在下一跳地址上执行递归查询，而是根据路由器中使用的路由路径被转发到该路由条目使用的下一跳中。

- 验证路由映射的下一 IPv4 跳是否可用:

```
set ip next-hop verify-availability next-hop-address sequence_number track object
```

您可以配置 SLA 监控跟踪对象来验证下一跳的可访问性。要验证多个下一跳的可用性，可使用不同的序列号和不同的跟踪对象来配置多个 **set ip next-hop verify-availability** 命令。

- 设置数据包的输出接口:

set interface *interface_name*

或

set interface null0

此命令可配置通过其转发匹配流量的接口。您可以配置多个接口，在这种情况下将按指定顺序对它们进行评估，直到找到有效的接口。当指定 **null0** 时，匹配路由映射的所有流量将被丢弃。对于可通过指定接口（静态或动态）路由的目标，必须存在路由。

- 将默认接口设置为 null0:

set default interface null0

如果正常路由查询失败，ASA 将转发流量 null0，并且该流量将被丢弃。

- 在 IP 报头中设置不分段 (DF) 位值:

set ip df {0|1}

- 通过在数据包中设置差分服务代码点 (DSCP) 或 IP 优先值对 IP 流量进行分类:

set {ip | ipv6} dscp new_dscp

注释 当配置了多个 set 操作时，ASA 将按以下顺序评估它们：**set ip next-hop verify-availability;** **set ip next-hop;** **set ip next-hop recursive;** **set interface;** **set ip default next-hop;** **set default interface。**

步骤 5 配置接口并进入接口配置模式:

interface *interface_id*

示例:

```
ciscoasa(config)# interface GigabitEthernet0/0
```

步骤 6 为通过设备的流量配置基于策略的路由:

policy-route route-map *route-map_name*

示例:

```
ciscoasa(config-if)# policy-route route-map testmap
```

要删除现有的基于策略的路由映射，只需输入此命令的 **no** 形式即可。

示例:

```
ciscoasa(config-if)# no policy-route route-map testmap
```

基于策略的路由示例

以下部分显示路由映射配置示例、基于策略的路由以及现行 PBR 的特定示例。

路由映射配置示例

在以下示例中，由于未指定操作和顺序，因此假设隐式操作为允许且序列号为 10：

```
ciscoasa(config)# route-map testmap
```

在以下示例中，由于未指定匹配条件，因此假设隐式匹配为“any”：

```
ciscoasa(config)# route-map testmap permit 10
ciscoasa(config-route-map)# set ip next-hop 1.1.1.10
```

在本示例中，与 <acl> 匹配的所有流量都将通过外部接口进行策略路由和转发。

```
ciscoasa(config)# route-map testmap permit 10
ciscoasa(config-route-map)# match ip address <acl>
ciscoasa(config-route-map)# set interface outside
```

在本示例中，由于未配置接口或下一跳操作，因此与 <acl> 匹配的所有流量都将根据配置修改 df 位字段和 dscp 字段，并使用普通路由进行转发。

```
ciscoasa(config)# route-map testmap permit 10
ciscoasa(config-route-map)# match ip address <acl>
set ip df 1
set ip precedence af11
```

在以下示例中，与 <acl_1> 匹配的所有流量都使用下一跳 1.1.1.10 进行转发，与 <acl_2> 匹配的所有流量都使用下一跳 2.1.1.10 进行转发，并会丢弃其余流量。“match”条件并不暗示隐式匹配“any”。

```
ciscoasa(config)# route-map testmap permit 10
ciscoasa(config-route-map)# match ip address <acl_1>
ciscoasa(config-route-map)# set ip next-hop 1.1.1.10

ciscoasa(config)# route-map testmap permit 20
ciscoasa(config-route-map)# match ip address <acl_2>

ciscoasa(config-route-map)# set ip next-hop 2.1.1.10
ciscoasa(config)# route-map testmap permit 30
ciscoasa(config-route-map)# set interface Null0
```

在以下示例中，路由映射评估结果将是 (i) 路由映射操作 permit 和 acl 操作 permit 将应用 set 操作 (ii) 路由映射操作 deny 和 acl 操作 permit 将跳至普通路由查找 (iii) 路由映射操作 permit/deny 和 acl 操作 deny 将继续处理下一个路由映射条目。当没有下一个路由映射条目可用时，将不会回退到普通路由查找。

```

ciscoasa(config)# route-map testmap permit 10
ciscoasa(config-route-map)# match ip address permit_acl_1 deny_acl_2
ciscoasa(config-route-map)# set ip next-hop 1.1.1.10

ciscoasa(config)# route-map testmap deny 20
ciscoasa(config-route-map)# match ip address permit_acl_3 deny_acl_4
ciscoasa(config-route-map)# set ip next-hop 2.1.1.10

ciscoasa(config)# route-map testmap permit 30
ciscoasa(config-route-map)# match ip address deny_acl_5
ciscoasa(config-route-map)# set interface outside

```

在以下示例中，当配置了多个 `set` 操作时，将按照上述顺序对其进行评估。仅当 `set` 操作的所有选项都已评估且无法应用时，才会考虑后续 `set` 操作。此排序将确保首先尝试可用性最高且距离最近的下一跳，然后尝试下一个可用性最高且距离最近的下一跳，依此类推。

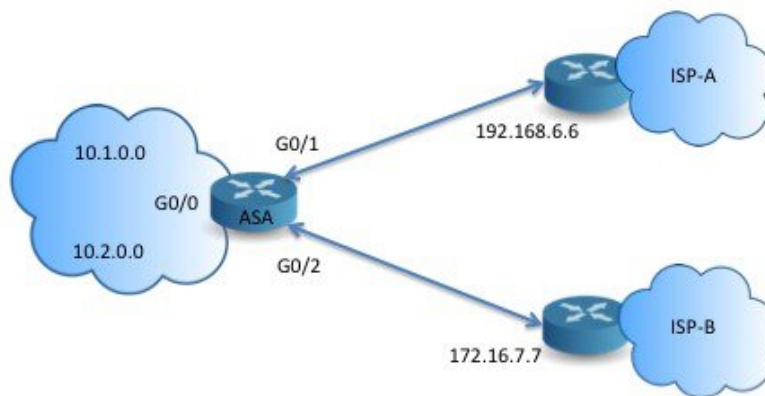
```

ciscoasa(config)# route-map testmap permit 10
ciscoasa(config-route-map)# match ip address acl_1
ciscoasa(config-route-map)# set ip next-hop verify-availability 1.1.1.10 1 track 1
ciscoasa(config-route-map)# set ip next-hop verify-availability 1.1.1.11 2 track 2
ciscoasa(config-route-map)# set ip next-hop verify-availability 1.1.1.12 3 track 3
ciscoasa(config-route-map)# set ip next-hop 2.1.1.10 2.1.1.11 2.1.1.12
ciscoasa(config-route-map)# set ip next-hop recursive 3.1.1.10
ciscoasa(config-route-map)# set interface outside-1 outside-2
ciscoasa(config-route-map)# set ip default next-hop 4.1.1.10 4.1.1.11
ciscoasa(config-route-map)# set default interface Null0

```

PBR 配置示例

本节介绍为以下场景配置 PBR 所需的全套配置：



首先，需要配置接口。

```

ciscoasa(config)# interface GigabitEthernet0/0
ciscoasa(config-if)# no shutdown
ciscoasa(config-if)# nameif inside
ciscoasa(config-if)# ip address 10.1.1.1 255.255.255.0

ciscoasa(config)# interface GigabitEthernet0/1
ciscoasa(config-if)# no shutdown

```



```
ciscoasa(config-if)# nameif outside-1
ciscoasa(config-if)# ip address 192.168.6.5 255.255.255.0

ciscoasa(config)# interface GigabitEthernet0/2
ciscoasa(config-if)# no shutdown
ciscoasa(config-if)# nameif outside-2
ciscoasa(config-if)# ip address 172.16.7.6 255.255.255.0
```

然后，我们需要配置一个访问列表来匹配流量。

```
ciscoasa(config)# access-list acl-1 permit ip 10.1.0.0 255.255.0.0
ciscoasa(config)# access-list acl-2 permit ip 10.2.0.0 255.255.0.0
```

我们需要将上述访问列表指定为匹配条件，并指定需要执行的一系列操作，以此来配置一个路由映射。

```
ciscoasa(config)# route-map equal-access permit 10
ciscoasa(config-route-map)# match ip address acl-1
ciscoasa(config-route-map)# set ip next-hop 192.168.6.6

ciscoasa(config)# route-map equal-access permit 20
ciscoasa(config-route-map)# match ip address acl-2
ciscoasa(config-route-map)# set ip next-hop 172.16.7.7

ciscoasa(config)# route-map equal-access permit 30
ciscoasa(config-route-map)# set ip interface Null0
```

现在，此路由映射必须连接至接口。

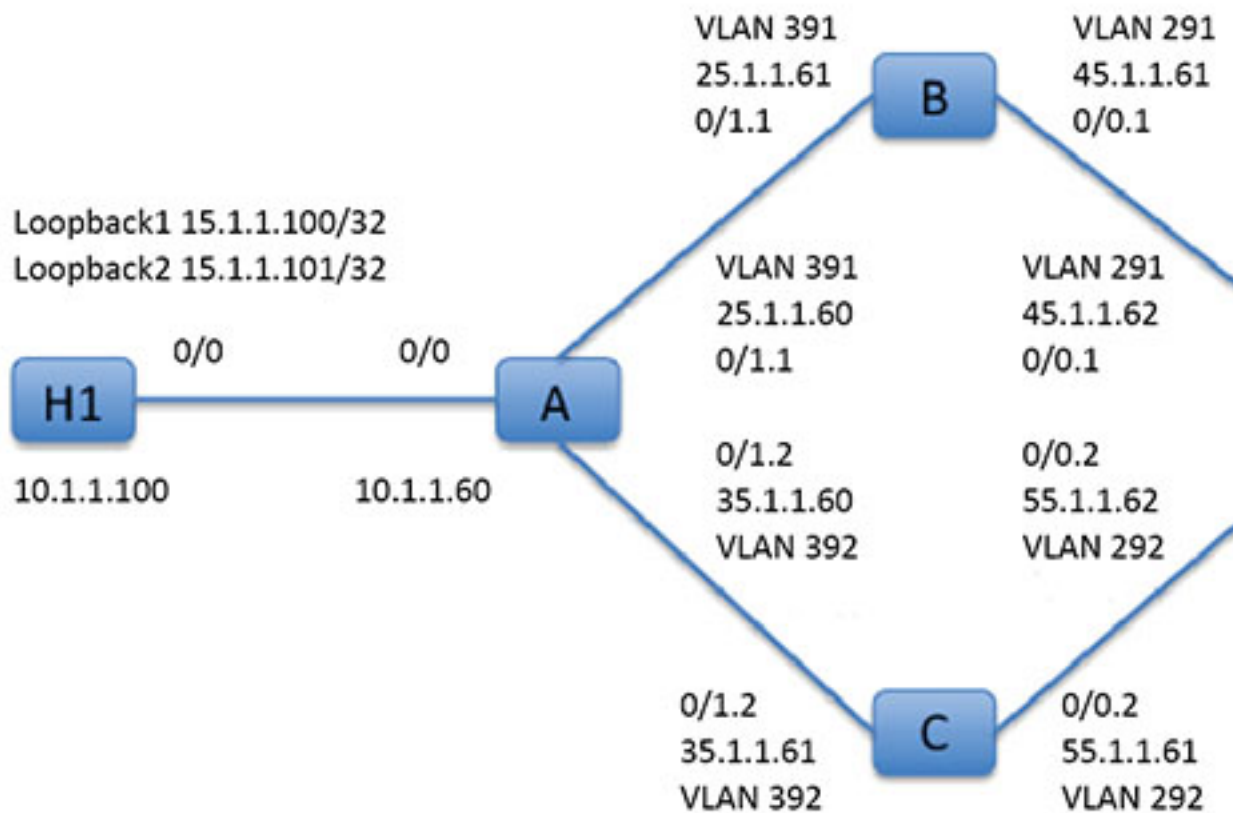
```
ciscoasa(config)# interface GigabitEthernet0/0
ciscoasa(config-if)# policy-route route-map equal-access
```

显示策略路由配置。

```
ciscoasa(config)# show policy-route
Interface          Route map
GigabitEthernet0/0  equal-access
```

正在使用的基于策略的路由

我们将使用此测试设置以不同的匹配条件配置基于策略的路由，并设置操作以了解如何评估和应用这些策略。



首先，我们了解一下设置中所涉及的所有设备的基本配置。这里，A、B、C和D代表ASA设备，H1和H2代表IOS路由器。

ASA-A:

```
ciscoasa(config)# interface GigabitEthernet0/0
ciscoasa(config-if)# nameif inside
ciscoasa(config-if)# security-level 100
ciscoasa(config-if)# ip address 10.1.1.60 255.255.255.0
ciscoasa(config)# interface GigabitEthernet0/1
ciscoasa(config-if)# no shut

ciscoasa(config)# interface GigabitEthernet0/1.1
ciscoasa(config-if)# vlan 391
ciscoasa(config-if)# nameif outside
ciscoasa(config-if)# security-level 0
ciscoasa(config-if)# ip address 25.1.1.60 255.255.255.0

ciscoasa(config)# interface GigabitEthernet0/1.2
ciscoasa(config-if)# vlan 392
ciscoasa(config-if)# nameif dmz
ciscoasa(config-if)# security-level 50
```

```
ciscoasa(config-if)# ip address 35.1.1.60 255.255.255.0
```

ASA-B:

```
ciscoasa(config)# interface GigabitEthernet0/0
ciscoasa(config-if)# no shut

ciscoasa(config)# interface GigabitEthernet0/0.1
ciscoasa(config-if)# vlan 291
ciscoasa(config-if)# nameif outside
ciscoasa(config-if)# security-level 0
ciscoasa(config-if)# ip address 45.1.1.61 255.255.255.0

ciscoasa(config)# interface GigabitEthernet0/1
ciscoasa(config-if)# no shut

ciscoasa(config)# interface GigabitEthernet0/1.1
ciscoasa(config-if)# vlan 391
ciscoasa(config-if)# nameif inside
ciscoasa(config-if)# security-level 100
ciscoasa(config-if)# ip address 25.1.1.61 255.255.255.0
```

ASA-C:

```
ciscoasa(config)# interface GigabitEthernet0/0
ciscoasa(config-if)# no shut

ciscoasa(config)# interface GigabitEthernet0/0.2
ciscoasa(config-if)# vlan 292
ciscoasa(config-if)# nameif outside
ciscoasa(config-if)# security-level 0
ciscoasa(config-if)# ip address 55.1.1.61 255.255.255.0

ciscoasa(config)# interface GigabitEthernet0/1
ciscoasa(config-if)# no shut

ciscoasa(config)# interface GigabitEthernet0/1.2
ciscoasa(config-if)# vlan 392
ciscoasa(config-if)# nameif inside
ciscoasa(config-if)# security-level 0
ciscoasa(config-if)# ip address 35.1.1.61 255.255.255.0
```

ASA-D:

```
ciscoasa(config)# interface GigabitEthernet0/0
ciscoasa(config-if)# no shut

ciscoasa(config)# interface GigabitEthernet0/0.1
ciscoasa(config-if)# vlan 291
ciscoasa(config-if)# nameif inside-1
ciscoasa(config-if)# security-level 100
ciscoasa(config-if)# ip address 45.1.1.62 255.255.255.0

ciscoasa(config)# interface GigabitEthernet0/0.2
ciscoasa(config-if)# vlan 292
ciscoasa(config-if)# nameif inside-2
ciscoasa(config-if)# security-level 100
ciscoasa(config-if)# ip address 55.1.1.62 255.255.255.0
```

```
ciscoasa(config)# interface GigabitEthernet0/1
ciscoasa(config-if)# nameif outside
ciscoasa(config-if)# security-level 0
ciscoasa(config-if)# ip address 65.1.1.60 255.255.255.0
```

H1:

```
ciscoasa(config)# interface Loopback1
ciscoasa(config-if)# ip address 15.1.1.100 255.255.255.255

ciscoasa(config-if)# interface Loopback2
ciscoasa(config-if)# ip address 15.1.1.101 255.255.255.255

ciscoasa(config)# ip route 0.0.0.0 0.0.0.0 10.1.1.60
```

H2:

```
ciscoasa(config)# interface GigabitEthernet0/1
ciscoasa(config-if)# ip address 65.1.1.100 255.255.255.0

ciscoasa(config-if)# ip route 15.1.1.0 255.255.255.0 65.1.1.60
```

我们将在 ASA-A 上配置 PBR 以路由源自 H1 的流量。

ASA-A:

```
ciscoasa(config-if)# access-list pbracl_1 extended permit ip host 15.1.1.100 any

ciscoasa(config-if)# route-map testmap permit 10
ciscoasa(config-if)# match ip address pbracl_1
ciscoasa(config-if)# set ip next-hop 25.1.1.61

ciscoasa(config)# interface GigabitEthernet0/0
ciscoasa(config-if)# policy-route route-map testmap

ciscoasa(config-if)# debug policy-route
```

H1: ping 65.1.1.100 repeat 1 source loopback1

```
pbr: policy based route lookup called for 15.1.1.100/44397 to 65.1.1.100/0 proto 1 sub_proto
 8 received on interface inside
pbr: First matching rule from ACL(2)
pbr: route map testmap, sequence 10, permit; proceed with policy routing
pbr: evaluating next-hop 25.1.1.61
pbr: policy based routing applied; egress_ifc = outside : next_hop = 25.1.1.61
```

数据包使用路由映射中的下一跳地址按预期转发。

当配置了下一跳时，将在输入路由表中执行查找，以确定到所配置的下一跳的已连接路由，并使用对应的接口。此处显示了本例的输入路由表（匹配路由条目已亮显）。

```
in 255.255.255.255 255.255.255.255 identity
in 10.1.1.60      255.255.255.255 identity
in 25.1.1.60     255.255.255.255 identity
in 35.1.1.60     255.255.255.255 identity
```

```
in 10.127.46.17 255.255.255.255 identity
in 10.1.1.0 255.255.255.0 inside
in 25.1.1.0 255.255.255.0 outside
in 35.1.1.0 255.255.255.0 dmz
```

接下来，我们将 ASA-A 配置为将数据包从 H1 loopback2 路由到 ASA-A dmz 接口外。

```
ciscoasa(config)# access-list pbracl_2 extended permit ip host 15.1.1.101 any
```

```
ciscoasa(config)# route-map testmap permit 20
ciscoasa(config-route-map)# match ip address pbracl
ciscoasa(config-route-map)# set ip next-hop 35.1.1.61
```

```
ciscoasa(config)# show run route-map
!
route-map testmap permit 10
  match ip address pbracl_1
  set ip next-hop 25.1.1.61
!
route-map testmap permit 20
  match ip address pbracl_2
  set ip next-hop 35.1.1.61
!
```

H1: ping 65.1.1.100 repeat 1 source loopback2

调试如下所示:

```
pbr: policy based route lookup called for 15.1.1.101/1234 to 65.1.1.100/1234 proto 6 sub_proto
 0 received on interface inside
pbr: First matching rule from ACL(3)
pbr: route map testmap, sequence 20, permit; proceed with policy routing
pbr: evaluating next-hop 35.1.1.61
pbr: policy based routing applied; egress_ifc = dmz : next_hop = 35.1.1.61
```

从输入路由表中所选的路由条目如下所示:

```
in 255.255.255.255 255.255.255.255 identity
in 10.1.1.60 255.255.255.255 identity
in 25.1.1.60 255.255.255.255 identity
in 35.1.1.60 255.255.255.255 identity
in 10.127.46.17 255.255.255.255 identity
in 10.1.1.0 255.255.255.0 inside
in 25.1.1.0 255.255.255.0 outside
in 35.1.1.0 255.255.255.0 dmz
```

基于策略的路由的历史记录

表 1: 路由映射的历史记录

功能名称	平台版本	功能信息
基于策略的路由	9.4(1)	<p>基于策略的路由 (PBR) 是一种机制，基于该机制，流量可以使用 ACL，通过带有指定 QoS 的特定路径进行路由。基于数据包的第 3 层和第 4 层报头的内容，ACL 可以对流量进行分类。管理员通过此解决方案可向不同的流量提供 QoS，在低带宽、低成本永久路径与高带宽、高成本交换式路径之间分发交互式和批处理流量，并允许互联网运营商和其他组织通过明确定义的互联网连接来路由源自各类用户的流量。</p> <p>引入了以下命令：set ip next-hop verify-availability、set ip next-hop、set ip next-hop recursive、set interface、set ip default next-hop、set default interface、set ip df、set ip dscp、policy-route route-map、show policy-route 和 debug policy-route</p>
为策略型路由提供 IPv6 支持	9.5(1)	<p>策略型路由现在支持 IPv6 地址。</p> <p>引入了以下命令：set ipv6 next-hop、set default ipv6-next hop、set ipv6 dscp</p>
为策略型路由提供 VXLAN 支持	9.5(1)	<p>现在您可以在 VNI 接口中启用策略型路由。</p> <p>未修改任何命令。</p>
为身份防火墙和思科 TrustSec 提供策略型路由支持	9.5(1)	<p>您可以先配置身份防火墙和思科 TrustSec，然后再在策略型路由的路由图中使用身份防火墙和思科 TrustSec ACL。</p> <p>未修改任何命令。</p>