Comunicação de LAN entre hosts que procuram seus endereços IP públicos por trás de um ASA

Contents

Introduction

Este documento descreve diferentes implementações de rede de onde é necessário permitir a comunicação de rede local (LAN) entre hosts que procuram seus endereços IP públicos por trás de um dispositivo de segurança adaptável (ASA).

Prerequisites

Requirements

A Cisco recomenda que você tenha conhecimento destes tópicos:

- Configuração básica do Cisco ASA NAT, versão 8.3 e superior.
- Configuração básica do Cisco ASA NAT, versão 8.2 e anterior.

Componentes Utilizados

As informações neste documento são baseadas nestas versões de software e hardware:

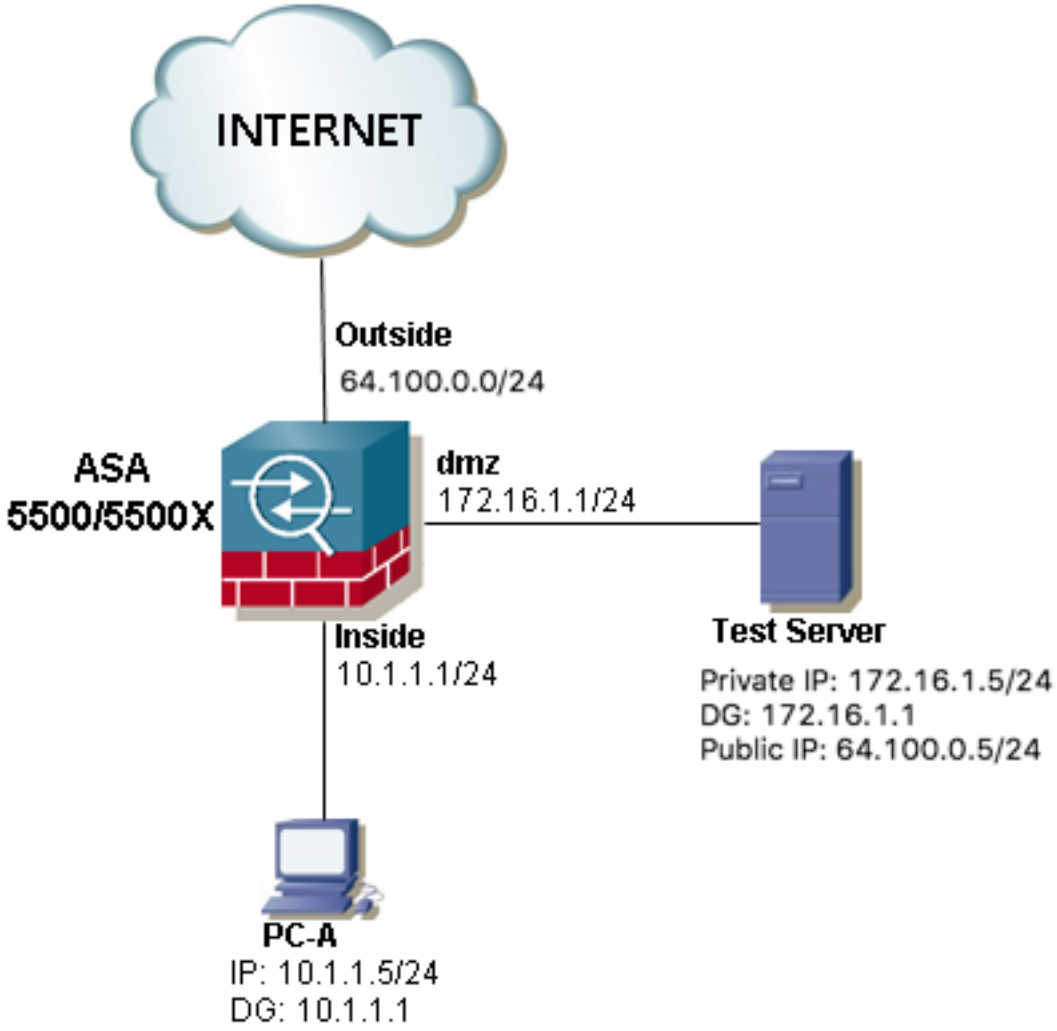
- ASA5500 e ASA5500-X Series.
- Cisco ASA versão 8.3 e superior.
- Cisco ASA versão 8.2 e anterior.

The information in this document was created from the devices in a specific lab environment. All of the devices used in this document started with a cleared (default) configuration. Se a rede estiver ativa, certifique-se de que você entenda o impacto potencial de qualquer comando.

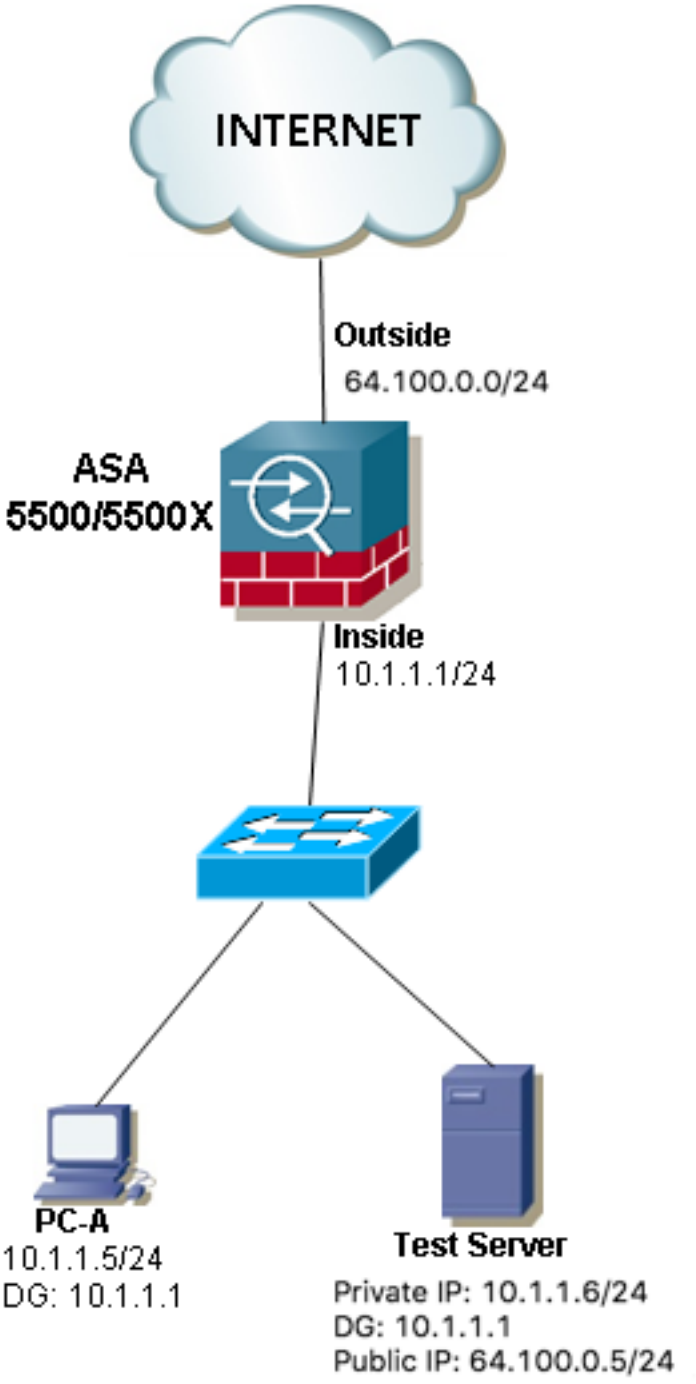
Problema: Comunicação de LAN entre hosts que procuram seus endereços IP públicos por trás de um ASA

Na próxima seção, você pode ver três exemplos de topologia que mostram esse requisito de comunicação para permitir a comunicação de LAN entre hosts que procuram seus endereços IP públicos por trás de um ASA.

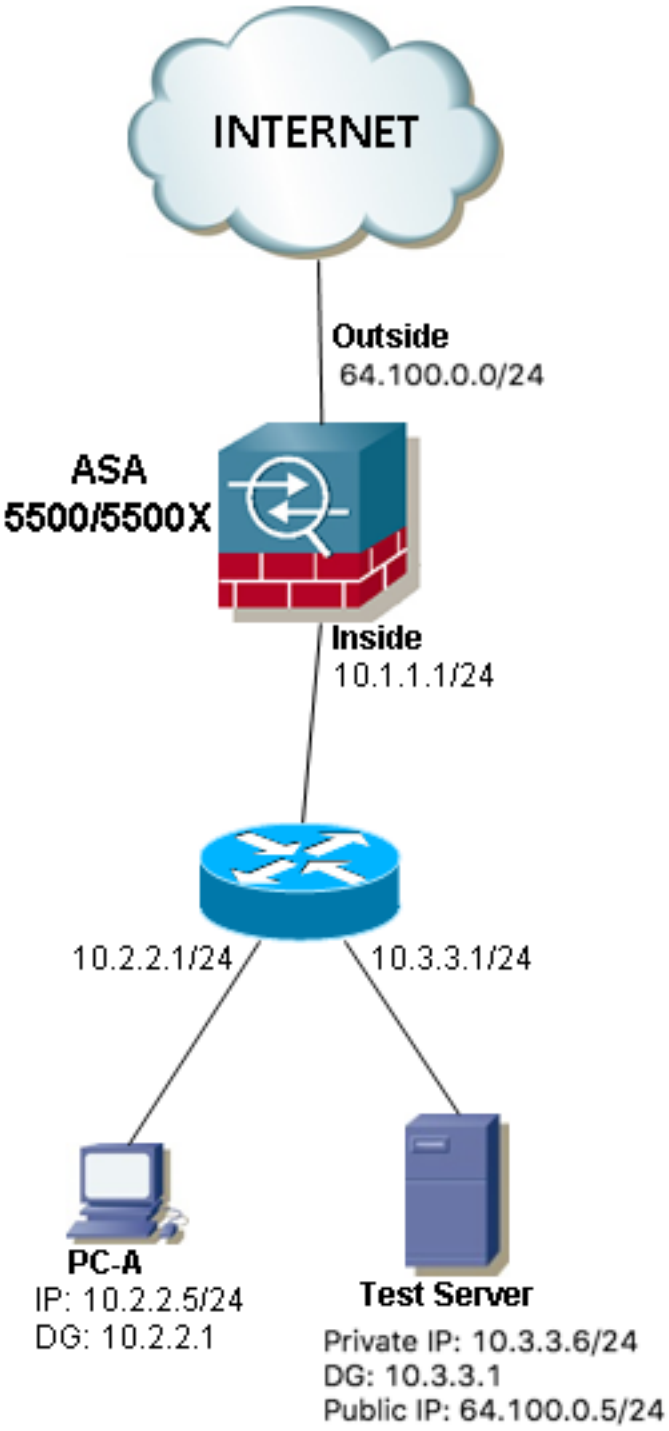
Exemplo 1. O PC-A do host de origem está conectado à interface interna do ASA, enquanto o Servidor de teste do host de destino está conectado à interface DMZ.



Exemplo 2. Os hosts origem e destino PC-A e Test Server estão conectados à mesma interface interna do ASA.



Exemplo 3. Os hosts origem e destino PC-A e Test Server estão conectados à interface interna do ASA, mas atrás de outro dispositivo da camada 3 (pode ser um roteador ou um switch multicamada).



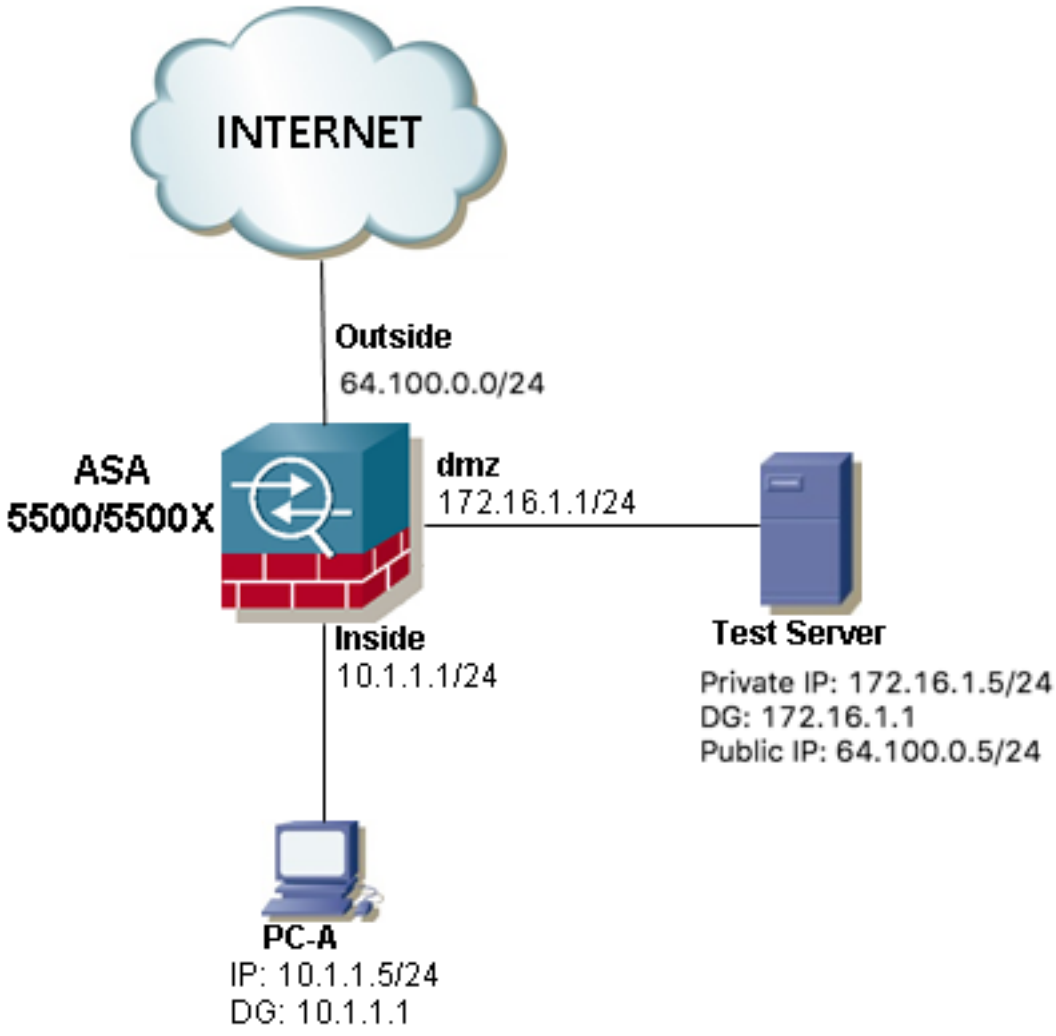
Note: O **Servidor de Teste** nas três imagens tem uma NAT (Network Address Translation) estática configurada no ASA, essa conversão de NAT estático é aplicada de fora para a interface interna correspondente para permitir que o **Servidor de Teste** possa ser alcançado de fora com o endereço IP público 64.100.0.5, então isso é convertido para o endereço IP privado interno do **Servidor de Teste**.

Solução

Para permitir que o PC-A do host de origem acesse o Servidor de teste de destino com seu endereço IP público em vez do privado, precisamos aplicar uma configuração de NAT duas vezes. A configuração de duas vezes NAT nos ajuda a converter os endereços IP origem e destino dos pacotes quando o tráfego passa pelo ASA.

Aqui estão os detalhes da configuração de duas nat necessárias para cada topologia:

Exemplo 1. O PC-A do host de origem está conectado à interface interna do ASA, enquanto o Servidor de teste do host de destino está conectado à interface DMZ.



Configuração

Duas vezes NAT para ASA versões 8.3 e posterior:

```
object network obj-10.1.1.5
host 10.1.1.5

object network obj-172.16.1.5
host 172.16.1.5

object network obj-64.100.0.5
host 64.100.0.5

nat (inside,dmz) source static obj-10.1.1.5 interface destination static obj-64.100.0.5 obj-
172.16.1.5
```

NOTE: After this NAT is applied in the ASA you will receive a warning message as the following:

```
WARNING: All traffic destined to the IP address of the outside interface is being redirected.
```

```
WARNING: Users may not be able to access any service enabled on the outside interface.
```

Duas vezes NAT para ASA versões 8.2 e mais antigas:

```
access-list IN-DMZ-INTERFACE extended permit ip host 10.1.1.5 host 64.100.0.5
static (inside,dmz) interface access-list IN-DMZ-INTERFACE

access-list DMZ-IN-INTERFACE extended permit ip host 172.16.1.5 host 172.16.1.1
static (dmz,inside) 64.100.0.5 access-list DMZ-IN-INTERFACE
```

Troubleshoot

Saída do Packet Tracer versões 8.3 e posteriores:

```
ASA# packet-tracer input inside tcp 10.1.1.5 123 64.100.0.5 80

Phase: 1
Type: ACCESS-LIST
Subtype:
Result: ALLOW
Config:
Implicit Rule
Additional Information:
MAC Access list

Phase: 2
Type: UN-NAT
Subtype: static
Result: ALLOW
Config:
nat (inside,dmz) source static obj-10.1.1.5 interface destination static obj-64.100.0.5 obj-
172.16.1.5
Additional Information:
NAT divert to egress interface dmz
Untranslate 64.100.0.5/80 to 172.16.1.5/80

Phase: 3
Type: NAT
Subtype:
Result: ALLOW
Config:
nat (inside,dmz) source static obj-10.1.1.5 interface destination static obj-64.100.0.5 obj-
172.16.1.5
Additional Information:
Static translate 10.1.1.5/123 to 172.16.1.1/123

Phase: 4
Type: NAT
Subtype: per-session
Result: ALLOW
Config:
Additional Information:

Phase: 5
Type: IP-OPTIONS
Subtype:
Result: ALLOW
Config:
Additional Information:

Phase: 6
```

```
Type: NAT
Subtype: rpf-check
Result: ALLOW
Config:
nat (inside,dmz) source static obj-10.1.1.5 interface destination static obj-64.100.0.5 obj-
172.16.1.5
Additional Information:

Phase: 7
Type: NAT
Subtype: per-session
Result: ALLOW
Config:
Additional Information:

Phase: 8
Type: IP-OPTIONS
Subtype:
Result: ALLOW
Config:
Additional Information:

Phase: 9
Type: FLOW-CREATION
Subtype:
Result: ALLOW
Config:
Additional Information:
New flow created with id 167632, packet dispatched to next module

Result:
input-interface: inside
input-status: up
input-line-status: up
output-interface: dmz
output-status: up
output-line-status: up
Action: allow
```

Saída do Packet Tracer versões 8.2 e mais antigas:

```
ASA#packet-tracer input inside tcp 10.1.1.5 123 64.100.0.5 80

Phase: 1
Type: UN-NAT
Subtype: static
Result: ALLOW
Config:
static (dmz,inside) 64.100.0.5 access-list DMZ-IN-INTERFACE
match ip dmz host 172.16.1.5 inside host 172.16.1.1
static translation to 64.100.0.5
translate_hits = 0, untranslate_hits = 1
Additional Information:
NAT divert to egress interface dmz
Untranslate 64.100.0.5/0 to 172.16.1.5/0 using netmask 255.255.255.255

Phase: 2
Type: IP-OPTIONS
Subtype:
Result: ALLOW
Config:
Additional Information:
```

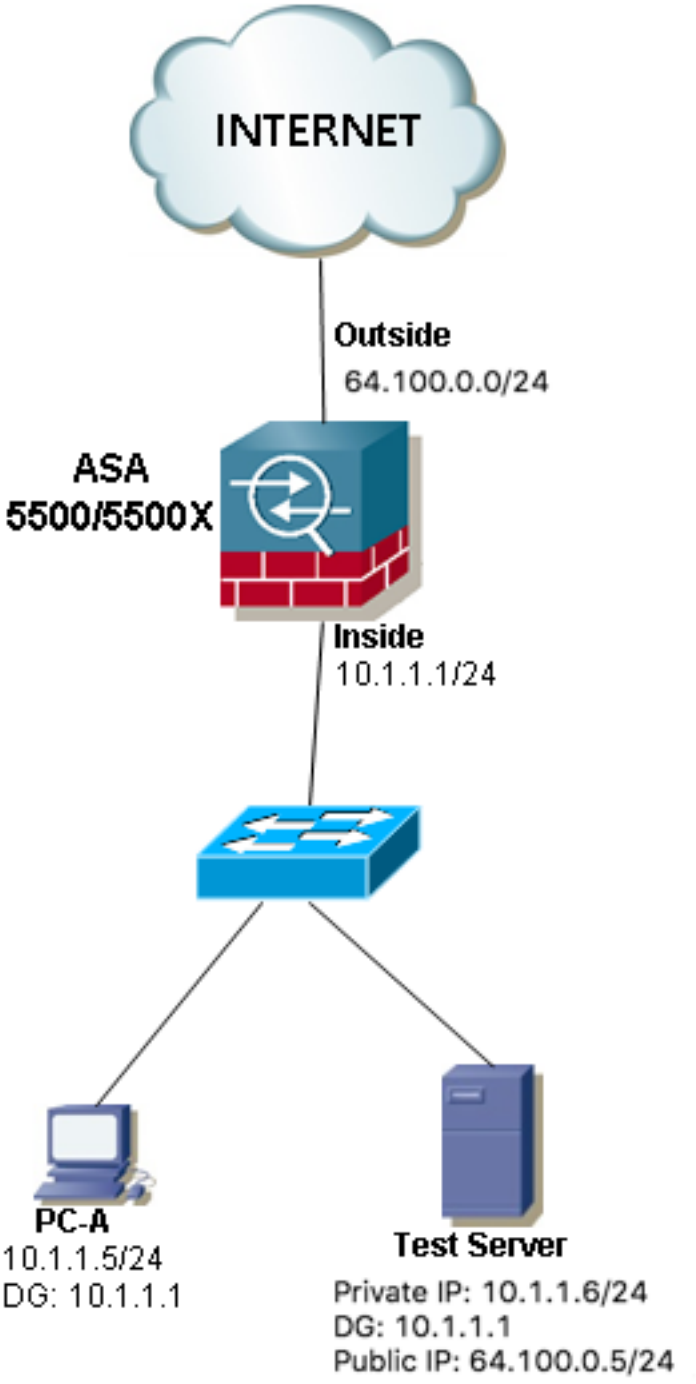
```
Phase: 3
Type: NAT
Subtype:
Result: ALLOW
Config:
static (inside,dmz) interface access-list IN-DMZ-INTERFACE
match ip inside host 10.1.1.5 dmz host 64.100.0.5
static translation to 172.16.1.1
translate_hits = 1, untranslate_hits = 0
Additional Information:
Static translate 10.1.1.5/0 to 172.16.1.1/0 using netmask 255.255.255.255

Phase: 4
Type: NAT
Subtype: host-limits
Result: ALLOW
Config:
static (inside,dmz) interface access-list IN-DMZ-INTERFACE
match ip inside host 10.1.1.5 dmz host 64.100.0.5
static translation to 172.16.1.1
translate_hits = 1, untranslate_hits = 0
Additional Information:

Phase: 5
Type: NAT
Subtype: rpf-check
Result: ALLOW
Config:
static (dmz,inside) 64.100.0.5 access-list DMZ-IN-INTERFACE
match ip dmz host 172.16.1.5 inside host 172.16.1.1
static translation to 64.100.0.5
translate_hits = 0, untranslate_hits = 1
Additional Information:

Phase: 6
Type: NAT
Subtype: host-limits
Result: ALLOW
Config:
static (dmz,inside) 64.100.0.5 access-list DMZ-IN-INTERFACE
match ip dmz host 172.16.1.5 inside host 172.16.1.1
static translation to 64.100.0.5
translate_hits = 0, untranslate_hits = 1
Additional Information:

Phase: 7
Type: IP-OPTIONS
Subtype:
Result: ALLOW
Config:
Additional Information:

Phase: 8
Type: FLOW-CREATION
Subtype:
Result: ALLOW
Config:
Additional Information:
New flow created with id 503, packet dispatched to next module

Result:
input-interface: inside
input-status: up
input-line-status: up
```

```
output-interface: dmz
output-status: up
output-line-status: up
Action: allow
```

Capturas de pacotes:

```
ASA# sh cap
capture capin type raw-data interface inside [Capturing - 1300 bytes]
match ip host 10.1.1.5 host 64.100.0.5
capture capout type raw-data interface dmz [Capturing - 1300 bytes]
match ip host 172.16.1.1 host 172.16.1.5

ASA# sh cap capin

10 packets captured
1: 12:36:28.245455 10.1.1.5 > 64.100.0.5: icmp: echo request
2: 12:36:28.269441 64.100.0.5 > 10.1.1.5: icmp: echo reply
3: 12:36:28.303451 10.1.1.5 > 64.100.0.5: icmp: echo request
4: 12:36:28.333692 64.100.0.5 > 10.1.1.5: icmp: echo reply
5: 12:36:28.372478 10.1.1.5 > 64.100.0.5: icmp: echo request
6: 12:36:28.395563 64.100.0.5 > 10.1.1.5: icmp: echo reply
7: 12:36:28.422402 10.1.1.5 > 64.100.0.5: icmp: echo request
8: 12:36:28.449241 64.100.0.5 > 10.1.1.5: icmp: echo reply
9: 12:36:28.481420 10.1.1.5 > 64.100.0.5: icmp: echo request
10: 12:36:28.507435 64.100.0.5 > 10.1.1.5: icmp: echo reply
10 packets shown

ASA1# sh cap capout

10 packets captured
1: 12:36:28.245730 172.16.1.1 > 172.16.1.5: icmp: echo request
2: 12:36:28.269395 172.16.1.5 > 172.16.1.1: icmp: echo reply
3: 12:36:28.303725 172.16.1.1 > 172.16.1.5: icmp: echo request
4: 12:36:28.333646 172.16.1.5 > 172.16.1.1: icmp: echo reply
5: 12:36:28.372737 172.16.1.1 > 172.16.1.5: icmp: echo request
6: 12:36:28.395533 172.16.1.5 > 172.16.1.1: icmp: echo reply
7: 12:36:28.422661 172.16.1.1 > 172.16.1.5: icmp: echo request
8: 12:36:28.449195 172.16.1.5 > 172.16.1.1: icmp: echo reply
9: 12:36:28.481695 172.16.1.1 > 172.16.1.5: icmp: echo request
10: 12:36:28.507404 172.16.1.5 > 172.16.1.1: icmp: echo reply
10 packets shown
```

Exemplo 2. Os hosts origem e destino PC-A e Test Server estão conectados à mesma interface interna do ASA.



Configuração

Duas vezes NAT para ASA versões 8.3 e posterior:

```
object network obj-10.1.1.5
host 10.1.1.5

object network obj-10.1.1.6
host 10.1.1.6

object network obj-64.100.0.5
host 64.100.0.5

nat (inside,inside) source static obj-10.1.1.5 interface destination static obj-64.100.0.5 obj-
10.1.1.6
```

NOTE: After this NAT is applied in the ASA you will receive a warning message as the following:

```
WARNING: All traffic destined to the IP address of the outside interface is being redirected.
WARNING: Users may not be able to access any service enabled on the outside interface.
```

Duas vezes NAT para ASA versões 8.2 e mais antigas:

```
access-list IN-OUT-INTERFACE extended permit ip host 10.1.1.5 host 64.100.0.5
static (inside,inside) interface access-list IN-OUT-INTERFACE

access-list OUT-IN-INTERFACE extended permit ip host 10.1.1.6 host 10.1.1.1
static (inside,inside) 64.100.0.5 access-list OUT-IN-INTERFACE
```

> **Note**: A principal intenção da conversão de NAT para o endereço IP origem de 10.1.1.5 para o endereço IP da interface interna do ASA 10.1.1.1 é forçar as respostas que vêm do host 10.1.1.6 a retornar ao ASA, isso é altamente necessário para evitar o roteamento assimétrico e permitir que o ASA processe todo o tráfego entre os hosts interessados, se não transpormos depois do endereço IP de origem como fizemos neste exemplo, o ASA bloqueará o tráfego interessado devido ao roteamento assimétrico.

Troubleshoot

Saída do Packet Tracer versões 8.3 e posteriores:

```
ASA# packet-tracer input inside tcp 10.1.1.5 123 64.100.0.5 80

Phase: 1
Type: UN-NAT
Subtype: static
Result: ALLOW
Config:
nat (inside,inside) source static obj-10.1.1.5 interface destination static obj-64.100.0.5 obj-
10.1.1.6
Additional Information:
NAT divert to egress interface inside
Untranslate 64.100.0.5/80 to 10.1.1.6/80

Phase: 2
Type: NAT
Subtype:
Result: ALLOW
Config:
nat (inside,inside) source static obj-10.1.1.5 interface destination static obj-64.100.0.5 obj-
10.1.1.6
Additional Information:
Static translate 10.1.1.5/123 to 10.1.1.1/123

Phase: 3
Type: ACCESS-LIST
Subtype:
Result: ALLOW
Config:
Implicit Rule
Additional Information:

Phase: 4
Type: NAT
Subtype: per-session
```

```
Result: ALLOW
Config:
Additional Information:

Phase: 5
Type: IP-OPTIONS
Subtype:
Result: ALLOW
Config:
Additional Information:

Phase: 6
Type: NAT
Subtype: rpf-check
Result: ALLOW
Config:
nat (inside,inside) source static obj-10.1.1.5 interface destination static obj-64.100.0.5 obj-
10.1.1.6
Additional Information:

Phase: 7
Type: NAT
Subtype: per-session
Result: ALLOW
Config:
Additional Information:

Phase: 8
Type: IP-OPTIONS
Subtype:
Result: ALLOW
Config:
Additional Information:

Phase: 9
Type: FLOW-CREATION
Subtype:
Result: ALLOW
Config:
Additional Information:
New flow created with id 167839, packet dispatched to next module

Result:
input-interface: inside
input-status: up
input-line-status: up
output-interface: inside
output-status: up
output-line-status: up
Action: allow
```

Saída do Packet Tracer versões 8.2 e mais antigas:

```
ASA# packet-tracer input inside tcp 10.1.1.5 123 64.100.0.5 80

Phase: 1
Type: UN-NAT
Subtype: static
Result: ALLOW
Config:
static (inside,inside) 64.100.0.5 access-list OUT-IN-INTERFACE
match ip inside host 10.1.1.6 inside host 10.1.1.1
static translation to 64.100.0.5
```

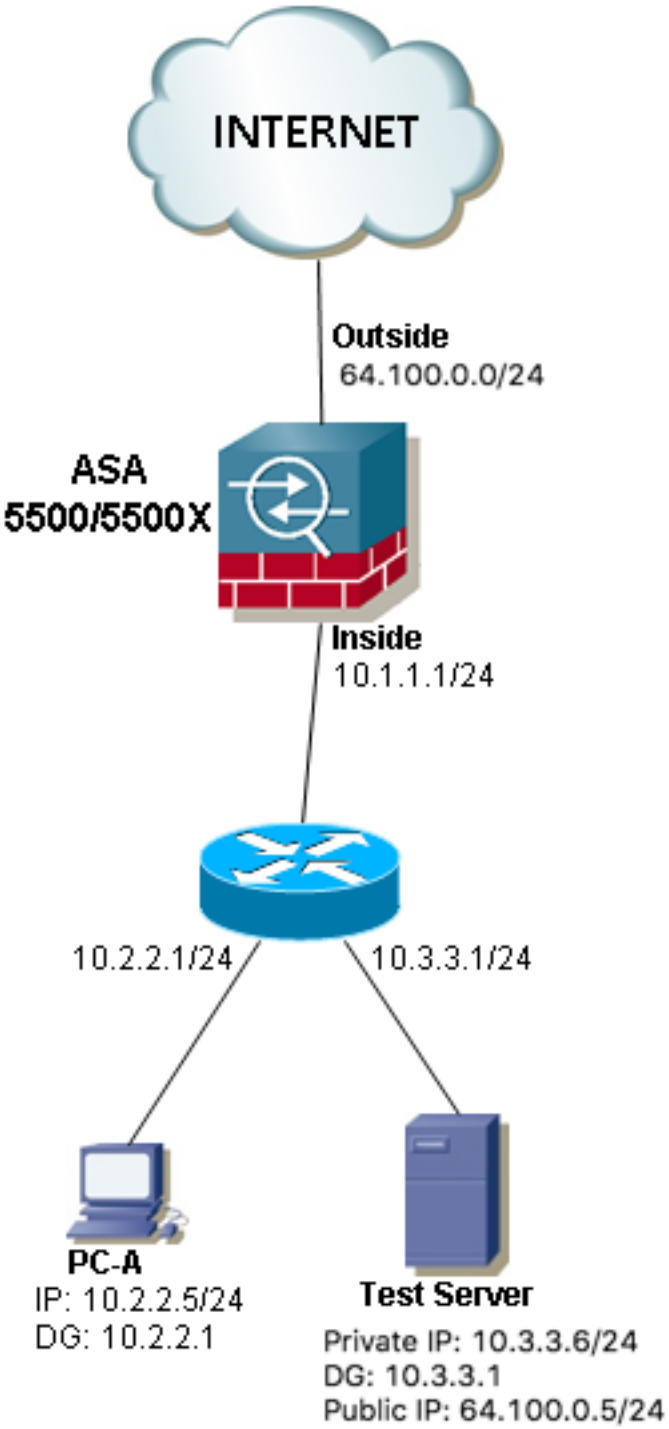
```
translate_hits = 0, untranslate_hits = 1
Additional Information:
NAT divert to egress interface inside
Untranslate 64.100.0.5/0 to 10.1.1.6/0 using netmask 255.255.255.255

Phase: 2
Type: ACCESS-LIST
Subtype:
Result: ALLOW
Config:
Implicit Rule
Additional Information:

Phase: 3
Type: IP-OPTIONS
Subtype:
Result: ALLOW
Config:
Additional Information:

Phase: 4
Type: NAT
Subtype:
Result: ALLOW
Config:
static (inside,inside) interface access-list IN-OUT-INTERFACE
match ip inside host 10.1.1.5 inside host 64.100.0.5
static translation to 10.1.1.1
translate_hits = 1, untranslate_hits = 0
Additional Information:
Static translate 10.1.1.5/0 to 10.1.1.1/0 using netmask 255.255.255.255

Phase: 5
Type: NAT
Subtype: host-limits
Result: ALLOW
Config:
static (inside,inside) interface access-list IN-OUT-INTERFACE
match ip inside host 10.1.1.5 inside host 64.100.0.5
static translation to 10.1.1.1
translate_hits = 1, untranslate_hits = 0
Additional Information:

Phase: 6
Type: NAT
Subtype: rpf-check
Result: ALLOW
Config:
static (inside,inside) 64.100.0.5 access-list OUT-IN-INTERFACE
match ip inside host 10.1.1.6 inside host 10.1.1.1
static translation to 64.100.0.5
translate_hits = 0, untranslate_hits = 1
Additional Information:

Phase: 7
Type: NAT
Subtype: host-limits
Result: ALLOW
Config:
static (inside,inside) 64.100.0.5 access-list OUT-IN-INTERFACE
match ip inside host 10.1.1.6 inside host 10.1.1.1
static translation to 64.100.0.5
translate_hits = 0, untranslate_hits = 1
Additional Information:
```

```
Phase: 8
Type: IP-OPTIONS
Subtype:
Result: ALLOW
Config:
Additional Information:

Phase: 9
Type: FLOW-CREATION
Subtype:
Result: ALLOW
Config:
Additional Information:
New flow created with id 727, packet dispatched to next module

Result:
input-interface: inside
input-status: up
input-line-status: up
output-interface: inside
output-status: up
output-line-status: up
Action: allow
```

Capturas de pacotes:

```
ASA# sh cap
capture capin type raw-data interface inside [Capturing - 1300 bytes]
match ip host 10.1.1.5 host 64.100.0.5
capture capout type raw-data interface inside [Capturing - 1300 bytes]
match ip host 10.1.1.1 host 10.1.1.6

ASA# sh cap capin

10 packets captured
1: 12:50:39.304748 10.1.1.5 > 64.100.0.5: icmp: echo request
2: 12:50:39.335431 64.100.0.5 > 10.1.1.5: icmp: echo reply
3: 12:50:39.368389 10.1.1.5 > 64.100.0.5: icmp: echo request
4: 12:50:39.389368 64.100.0.5 > 10.1.1.5: icmp: echo reply
5: 12:50:39.398432 10.1.1.5 > 64.100.0.5: icmp: echo request
6: 12:50:39.418176 64.100.0.5 > 10.1.1.5: icmp: echo reply
7: 12:50:39.419732 10.1.1.5 > 64.100.0.5: icmp: echo request
8: 12:50:39.425103 64.100.0.5 > 10.1.1.5: icmp: echo reply
9: 12:50:39.434395 10.1.1.5 > 64.100.0.5: icmp: echo request
10: 12:50:39.438423 64.100.0.5 > 10.1.1.5: icmp: echo reply
10 packets shown

ASA2# sh cap capout

10 packets captured
1: 12:50:39.305282 10.1.1.1 > 10.1.1.6: icmp: echo request
2: 12:50:39.335386 10.1.1.6 > 10.1.1.1: icmp: echo reply
3: 12:50:39.368663 10.1.1.1 > 10.1.1.6: icmp: echo request
4: 12:50:39.389307 10.1.1.6 > 10.1.1.1: icmp: echo reply
5: 12:50:39.398706 10.1.1.1 > 10.1.1.6: icmp: echo request
6: 12:50:39.418130 10.1.1.6 > 10.1.1.1: icmp: echo reply
7: 12:50:39.419762 10.1.1.1 > 10.1.1.6: icmp: echo request
8: 12:50:39.425072 10.1.1.6 > 10.1.1.1: icmp: echo reply
9: 12:50:39.434669 10.1.1.1 > 10.1.1.6: icmp: echo request
10: 12:50:39.438392 10.1.1.6 > 10.1.1.1: icmp: echo reply
10 packets shown
```

Exemplo 3. Os hosts origem e destino PC-A e Test Server estão conectados à interface interna do ASA, mas atrás de outro dispositivo da camada 3 (pode ser um roteador ou um switch multicamada).



Configuração

Duas vezes NAT para ASA versões 8.3 e posterior:

```
object network obj-10.2.2.5
host 10.2.2.5

object network obj-10.3.3.6
```

```
host 10.3.3.6

object network obj-64.100.0.5
host 64.100.0.5

nat (inside,inside) source static obj-10.2.2.5 interface destination static obj-64.100.0.5 obj-
10.3.3.6
```

NOTE: After this NAT is applied in the ASA you will receive a warning message as the following:

```
WARNING: All traffic destined to the IP address of the outside interface is being redirected.
WARNING: Users may not be able to access any service enabled on the outside interface.
```

Duas vezes NAT para ASA versões 8.2 e mais antigas:

```
access-list IN-OUT-INTERFACE extended permit ip host 10.2.2.5 host 64.100.0.5
static (inside,inside) interface access-list IN-OUT-INTERFACE

access-list OUT-IN-INTERFACE extended permit ip host 10.3.3.6 host 10.1.1.1
static (inside,inside) 64.100.0.5 access-list OUT-IN-INTERFACE
```

> **Observação**: a principal intenção da conversão de NAT para o endereço IP de origem de 10.1.1.5 para o endereço IP da interface interna do ASA (10.1.1.1) é forçar as respostas que vêm do host 10.1.1.6 a retornarem ao ASA, isso é altamente necessário para evitar o roteamento assimétrico e permitir que o ASA processe todos os hosts entre os interessados se não traduzirmos o endereço IP de origem como fizemos neste exemplo, o ASA bloqueará o tráfego interessado devido ao roteamento assimétrico.

Troubleshoot

Saída do Packet Tracer versões 8.3 e posteriores:

```
ASA# packet-tracer input inside tcp 10.2.2.5 123 64.100.0.5 80

Phase: 1
Type: UN-NAT
Subtype: static
Result: ALLOW
Config:
nat (inside,inside) source static obj-10.2.2.5 interface destination static obj-64.100.0.5 obj-
10.3.3.6
Additional Information:
NAT divert to egress interface inside
Untranslate 64.100.0.5/80 to 10.3.3.6/80

Phase: 2
Type: NAT
Subtype:
Result: ALLOW
Config:
nat (inside,inside) source static obj-10.2.2.5 interface destination static obj-64.100.0.5 obj-
10.3.3.6
Additional Information:
Static translate 10.2.2.5/123 to 10.1.1.1/123

Phase: 3
Type: ACCESS-LIST
Subtype:
```

```
Result: ALLOW
Config:
Implicit Rule
Additional Information:

Phase: 4
Type: NAT
Subtype: per-session
Result: ALLOW
Config:
Additional Information:

Phase: 5
Type: IP-OPTIONS
Subtype:
Result: ALLOW
Config:
Additional Information:

Phase: 6
Type: NAT
Subtype: rpf-check
Result: ALLOW
Config:
nat (inside,inside) source static obj-10.2.2.5 interface destination static obj-64.100.0.5 obj-
10.3.3.6
Additional Information:

Phase: 7
Type: NAT
Subtype: per-session
Result: ALLOW
Config:
Additional Information:

Phase: 8
Type: IP-OPTIONS
Subtype:
Result: ALLOW
Config:
Additional Information:

Phase: 9
Type: FLOW-CREATION
Subtype:
Result: ALLOW
Config:
Additional Information:
New flow created with id 167945, packet dispatched to next module

Result:
input-interface: inside
input-status: up
input-line-status: up
output-interface: inside
output-status: up
output-line-status: up
Action: allow
```

Saída do Packet Tracer versões 8.2 e mais antigas:

```
ASA# packet-tracer input inside tcp 10.2.2.5 123 64.100.0.5 80
```

```
Phase: 1
Type: UN-NAT
Subtype: static
Result: ALLOW
Config:
static (inside,inside) 64.100.0.5 access-list OUT-IN-INTERFACE
match ip inside host 10.3.3.6 inside host 10.1.1.1
static translation to 64.100.0.5
translate_hits = 0, untranslate_hits = 1
Additional Information:
NAT divert to egress interface inside
Untranslate 64.100.0.5/0 to 10.3.3.6/0 using netmask 255.255.255.255

Phase: 2
Type: ACCESS-LIST
Subtype:
Result: ALLOW
Config:
Implicit Rule
Additional Information:

Phase: 3
Type: IP-OPTIONS
Subtype:
Result: ALLOW
Config:
Additional Information:

Phase: 4
Type: NAT
Subtype:
Result: ALLOW
Config:
static (inside,inside) interface access-list IN-OUT-INTERFACE
match ip inside host 10.2.2.5 inside host 64.100.0.5
static translation to 10.1.1.1
translate_hits = 1, untranslate_hits = 0
Additional Information:
Static translate 10.2.2.5/0 to 10.1.1.1/0 using netmask 255.255.255.255

Phase: 5
Type: NAT
Subtype: host-limits
Result: ALLOW
Config:
static (inside,inside) interface access-list IN-OUT-INTERFACE
match ip inside host 10.2.2.5 inside host 64.100.0.5
static translation to 10.1.1.1
translate_hits = 1, untranslate_hits = 0
Additional Information:

Phase: 6
Type: NAT
Subtype: rpf-check
Result: ALLOW
Config:
static (inside,inside) 64.100.0.5 access-list OUT-IN-INTERFACE
match ip inside host 10.3.3.6 inside host 10.1.1.1
static translation to 64.100.0.5
translate_hits = 0, untranslate_hits = 1
Additional Information:

Phase: 7
Type: NAT
```

```
Subtype: host-limits
Result: ALLOW
Config:
static (inside,inside) 64.100.0.5 access-list OUT-IN-INTERFACE
match ip inside host 10.3.3.6 inside host 10.1.1.1
static translation to 64.100.0.5
translate_hits = 0, untranslate_hits = 1
Additional Information:

Phase: 8
Type: IP-OPTIONS
Subtype:
Result: ALLOW
Config:
Additional Information:

Phase: 9
Type: FLOW-CREATION
Subtype:
Result: ALLOW
Config:
Additional Information:
New flow created with id 908, packet dispatched to next module

Result:
input-interface: inside
input-status: up
input-line-status: up
output-interface: inside
output-status: up
output-line-status: up
Action: allow
```

Capturas de pacotes:

```
ASA# sh cap
capture capin type raw-data interface inside [Capturing - 1300 bytes]
match ip host 10.2.2.5 host 64.100.0.5
capture capout type raw-data interface inside [Capturing - 1300 bytes]
match ip host 10.1.1.1 host 10.3.3.6

ASA# sh cap capin

10 packets captured
1: 13:06:09.302047 10.2.2.5 > 64.100.0.5: icmp: echo request
2: 13:06:09.315276 64.100.0.5 > 10.2.2.5: icmp: echo reply
3: 13:06:09.342221 10.2.2.5 > 64.100.0.5: icmp: echo request
4: 13:06:09.381266 64.100.0.5 > 10.2.2.5: icmp: echo reply
5: 13:06:09.421227 10.2.2.5 > 64.100.0.5: icmp: echo request
6: 13:06:09.459204 64.100.0.5 > 10.2.2.5: icmp: echo reply
7: 13:06:09.494939 10.2.2.5 > 64.100.0.5: icmp: echo request
8: 13:06:09.534258 64.100.0.5 > 10.2.2.5: icmp: echo reply
9: 13:06:09.564210 10.2.2.5 > 64.100.0.5: icmp: echo request
10: 13:06:09.593261 64.100.0.5 > 10.2.2.5: icmp: echo reply
10 packets shown

ASA# sh cap capout

10 packets captured
1: 13:06:09.302367 10.1.1.1 > 10.3.3.6: icmp: echo request
2: 13:06:09.315230 10.3.3.6 > 10.1.1.1: icmp: echo reply
3: 13:06:09.342526 10.1.1.1 > 10.3.3.6: icmp: echo request
4: 13:06:09.381221 10.3.3.6 > 10.1.1.1: icmp: echo reply
```

```
5: 13:06:09.421517 10.1.1.1 > 10.3.3.6: icmp: echo request
6: 13:06:09.459174 10.3.3.6 > 10.1.1.1: icmp: echo reply
7: 13:06:09.495244 10.1.1.1 > 10.3.3.6: icmp: echo request
8: 13:06:09.534213 10.3.3.6 > 10.1.1.1: icmp: echo reply
9: 13:06:09.564500 10.1.1.1 > 10.3.3.6: icmp: echo request
10: 13:06:09.593215 10.3.3.6 > 10.1.1.1: icmp: echo reply
10 packets shown
```

Informações Relacionadas

- Guia de configuração do ASA 8.3: Pré-requisito para duas NAT
- Guia de configuração do ASA 8.4: DNS e NAT
- Exemplos de configuração de NAT ASA Pre-8.3 a 8.3