NAT in VoIP

Contents

Introduction
Prerequisites
Requirements
Components Used
Background Information
Static NAT
Dynamic NAT
NAT overload (PAT)
NAT Command options
NAT pinhole
NAT in VoIP
ALG
Gateways
CME
Local
Local to remote
Remote teleworker
Remote phones with public (read: routable) IP addresses
Remote phones with private IP address
Remote SIP phones
CUBE
Hosted NAT Traversal
NAT SBC
Design Notes
Configuration
Call Flow with SBC NAT
SIP Registration
CUSP
Troubleshooting
Symptoms
Show and debug commands
Things to check
Scenarios
Basic NAT
SIP ALG
References

Introduction

This document describes NAT (Network Address Translation) behavior in routers working as

CUBE (Cisco Unified Border Element), CME or CUCME (Cisco Unified Cummunication Manager Express), Gateways and CUSP (Cisco Unified SIP Proxy).

Prerequisites

Requirements

Cisco recommends that you have knowledge of these topics:

- SIP (Session Initiation Protocol)
- Voice over IP (Internet Protocol)
- Routing Protocols

Components Used

The information in this document is based on

- Any IOS version 12.4T and above.
- Any CME version

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. If your network is live, make sure that you understand the potential impact of any command.

Background Information

Network Address Translation is a commonly used technique to translate IP addresses on packets that flow between networks using different address spaces. The purpose of this document is not to review NAT. Rather, this document aims to provide a comprehensive review of NAT as it is used in Cisco's VoIP networks. Furthermore, the scope is limited to components that make up the MS-Voice technology.

- NAT basically replaces the IP address within packets with a different IP address
- Enables multiple hosts in a private subnet to *share* (i.e. appear like) a single public IP address, to access the Internet.
- Typically, NAT configurations change only the IP address of inside hosts
- NAT is bidirectional- If A gets translated to B on the inside interface, B arriving at outside interface will get translated to A!
- RFC1631

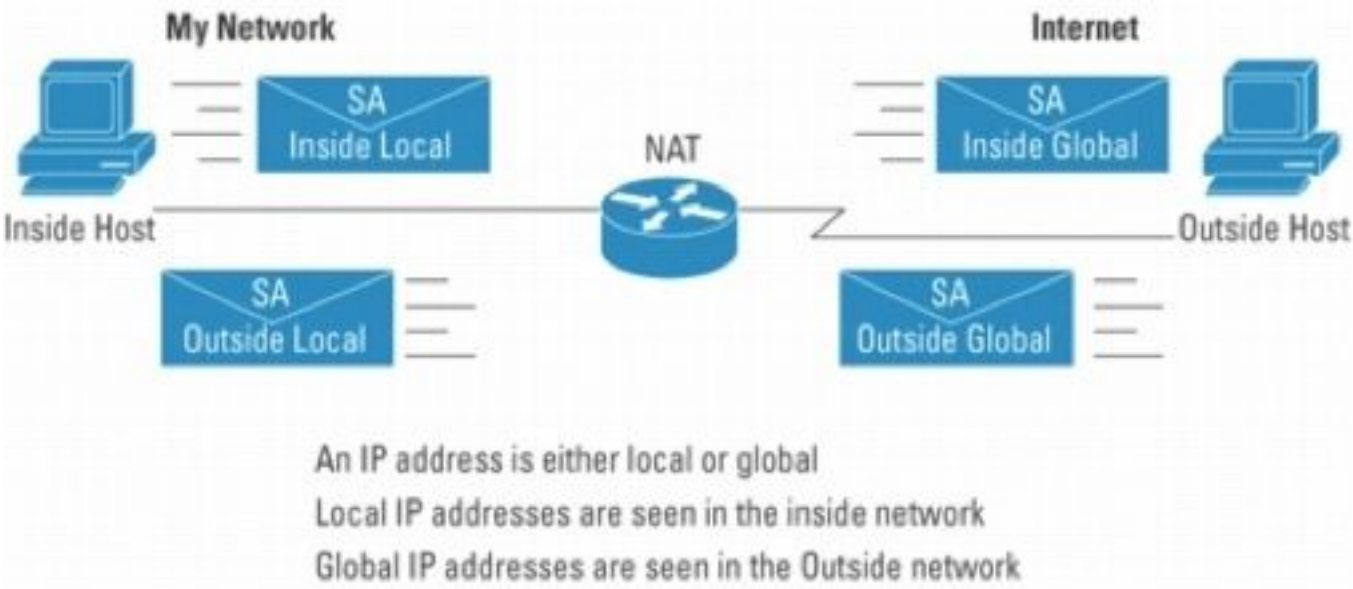


Figure 1

> **Note**: It may help to think of NAT as an aid to route IP packets into and out of networks using private address space. In other words, NAT makes non-routable addresses routable

Figure 2 Shows the topology referenced in the illustrations that follow.

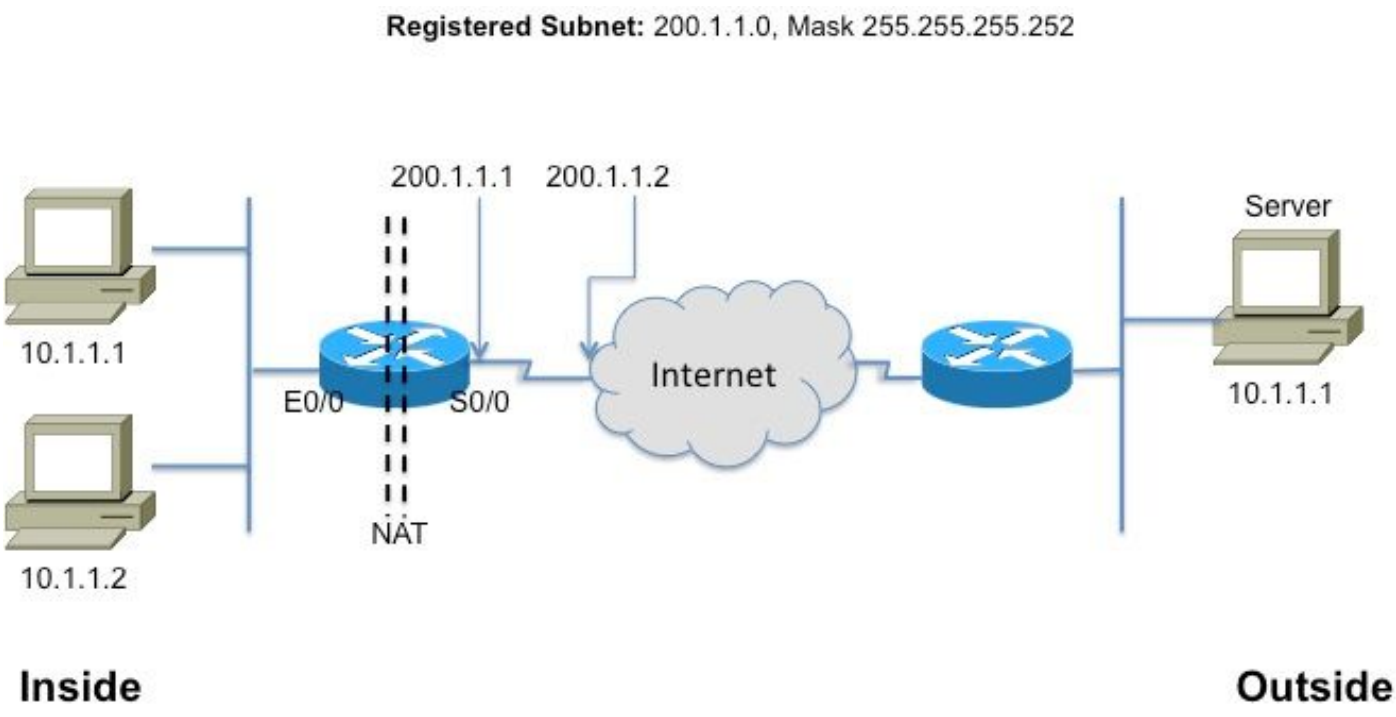


Figure 2

This glossary is fundamental to understand and describe NAT

- **Inside local address**—The IP address assigned to a host on the *inside* network. Typically, the address is from a private address space.
- **Inside global address**—A routable IP address assigned by the NIC or service provider that represents one or more inside local IP addresses to the outside world.
- **Outside local address**—The IP address of an outside host as it appears to the inside

network. Not necessarily a legitimate address, it is allocated from an address space routable on the inside.

- **Outside global address**—The IP address assigned to a host on the outside network by the host owner. The address is allocated from a globally routable address or network space.

Note: Get comfortable with these terms. Any note or doc on NAT is sure to refer to them

Static NAT

This is the simplest form of NAT, where in each inside address is statically translated to an outside address (and vice versa).

Inside Local	Inside Global
10.1.1.1	200.1.1.1
10.1.1.2	200.1.1.2

Figure 3

The CLI to configuration for the above translation is as follows

interface Ethernet0/0

ip address 10.1.1.3 255.255.255.0

ip nat inside

!

interface Serial0/0

ip address 200.1.1.251 255.255.255.252

ip nat outside <-- Required![2]

ip nat inside source static 10.1.1.2 200.1.1.2

ip nat inside source static 10.1.1.1 200.1.1.1

Dynamic NAT

In dynamic NAT, each inside host is mapped to an address from a pool of addresses.

- Allocates an IP address from a pool of inside global addresses.
- If a new packet arrives from yet another inside host, and it needs a NAT entry, but all the pooled IP addresses are in use, the router simply discards the packet.
- Essentially, the pool of inside global addresses needs to be as large as the maximum number

of concurrent hosts that need to use the Internet at the same time

The following CLI illustrates configuring dynamic NAT

```
ip nat pool fred 200.1.1.1 200.1.1.2 netmask 255.255.255.252
!
!
ip nat inside source list 1 pool fred
!
access-list 1 permit 10.1.1.2
access-list 1 permit 10.1.1.1
```

NAT overload (PAT)

When the pool (of IP addresses) is smaller than the set of addresses that need to be translated, this feature comes in handy.

- Several internal addresses NATed to only one or a few external addresses
- PAT (Port Address Translation) uses unique source port numbers on the Inside **Global** IP address to distinguish between translations. Because the port number is encoded in 16 bits, the total number could theoretically be as high as 65,536 per IP address. PAT will attempt to preserve the original source port, if this source port is already allocated PAT will attempt to find the first available port number
- NAT overload can use more than 65,000 ports, allowing it to scale well without needing many registered IP addresses—in many cases, needing only one outside global IP address.

Figure 4 illustrates PAT.

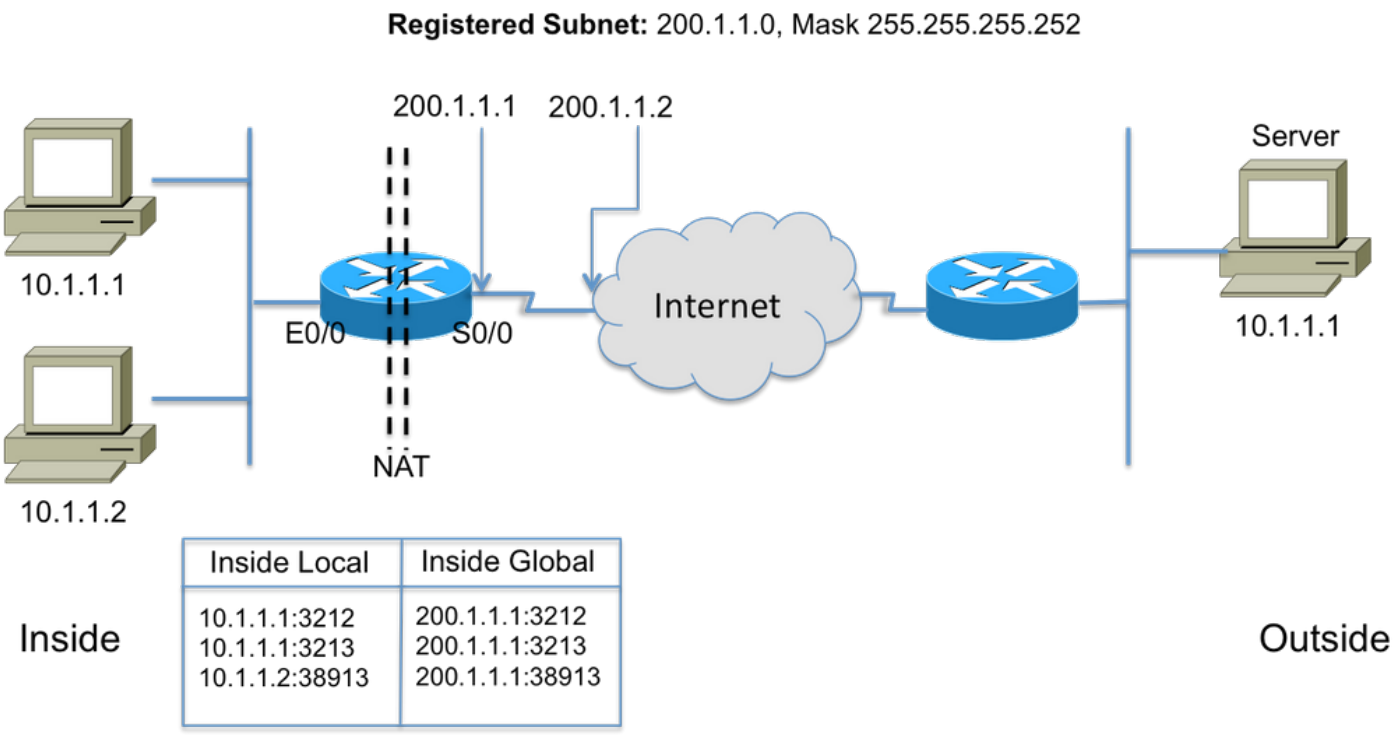


Figure 4

NAT Command options

Cisco NAT implementation is very versatile with a host of options. A few are listed below, but please refer to http://www.cisco.com/en/US/partner/technologies/tk648/tk361/tk438/technologies_white_paper09186a0080091cb9.html for details on the complete list of enhancements.

- Static translations with ports – Incoming packets addressed to a specific port (e.g. port 25, for SMTP server) sent to a specific server.
- Support for route maps - Flexibility in configuring filters/ACLs
- More flexible pool configurations- to allow discontinuous ranges of addresses.
- Host number preservation - Translate the “network” part, retain the “host” part.

NAT pinhole

A pinhole in NAT parlance refers to the mapping between the <host IP, port> and <global address, *global* port> tuples. It allows the NAT device to use the destination port number (which would be the *global* port) of incoming messages to map the destination back to the host IP and port that originated the session. It is important to note that pinholes time out after a period of non-use and the public address is returned to the NAT pool.

NAT in VoIP

So, what are the issues and concerns with NAT in VoIP networks? Well, recall that NAT that we have discussed so far (losely referredto as basic NAT) only translates the IP address in the IP packet *header* and re-calculates the checksum, of course, but VoIP signaling carry addresses embedded in the *body* of the signaling messages. In other words, at Layer 5

Figure 5 illustrates the effect of leaving the embedded IP addresses un-translated. The call signaling completes successful, but the SIP proxy at the service provider fails trying to route media (RTP) packets to media address sent by the call agent!

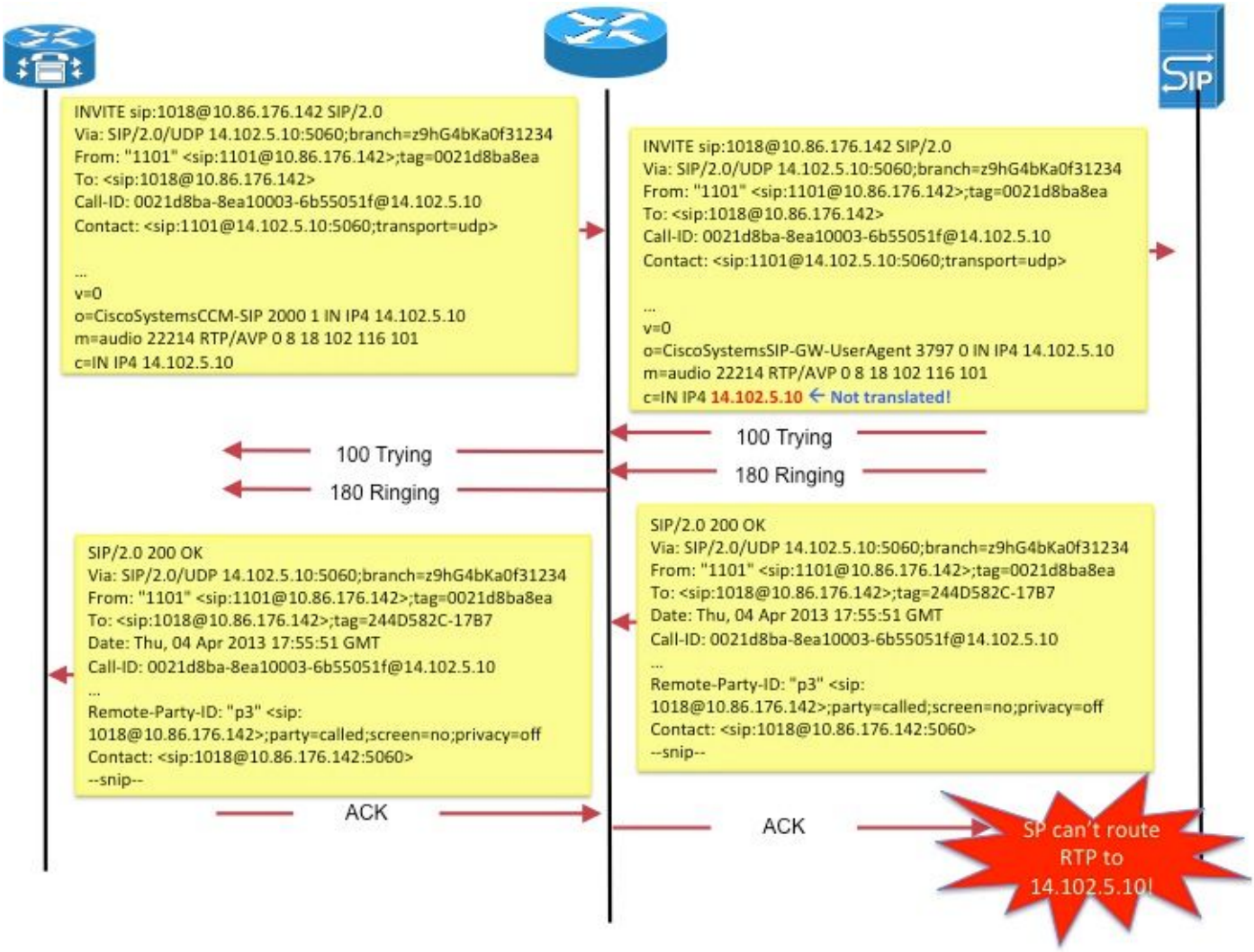


Figure 5

Another example would be SIP endpoint's use of ***Contact:*** field in SDP to communicate the address at which the endpoint would like to receive signaling messages for new requests.

These issues are addressed by a feature called Application Layer Gateway (ALG).

ALG

An ALG understands the protocol used by the specific applications that it supports (e.g. SIP) and does protocol packet-inspection and "fixup" of traffic through it. For a good description of how the various fields are fixed-up for SIP call signaling, refer to http://www.voip-info.org/wiki/view/Routers+SIP+ALG.

On Cisco routers, support for ALG SIP is enabled, by default, on the standard TCP port 5060. It is possible to configure ALG to support nonstandard ports for SIP signaling. Refer to http://www.cisco.com/en/US/docs/ios-xml/ios/ipaddr_nat/configuration/15-mt/nat-tcp-sip-alg.html.

> **Caution**: Beware! There is no RFC or other standard that spells out which embedded fields should be translated for the various VoIP protocols. As a result, implementations vary, among equipment vendors, resulting in interop issues (and TAC cases).

Gateways

Since gateways, by definition, are not ip-to-ip devices, NAT is not applicable.

CME

This section of the document review call scenarios with CME to understand why NAT must be used.

Scenario 1. Local phones

Scenario 2. Remote phones (with public IP addresses)

Scenario 3. Remote teleworker

> **Note**: In all cases, for audio to flow, the CME IP address needs to be routable

Local

In this scenario (Figure 6), the two phones involved in the call are skinny phones with private IP addresses.



Figure 6

> **Note**: Remember that skinny phone that is connected in a call with another skinny phone in the same CME system sends its media packets directly to the other phone; i.e. RTP for local-phone to local-phone does NOT flow through CME.

Therefore, NAT is not applicable or required in this case.

> **Note**: CME determines if media (RTP) should directly or not based on whether the two phones involved in a call are both skinny *and* in the same network segment. Otherwise, CME inserts itself in the RTP path.

Local to remote

In this scenario (Figure 7), CME inserts itself into the RTP stream such that RTP from the phones will be terminated on the CME. CME will re-originate the streams towards the other phone. Since CME sits on both the inside (private) network and the outside network and sends its inside address to the inside phone and outside (public) address to the outside phone, NAT is not required here either.

Note however, that the UDP/TCP ports (signaling as well as RTP) must be open between remote IP phone and CME source IP address. This means that the firewalls or other filtering devices are configured to allow the ports in question.

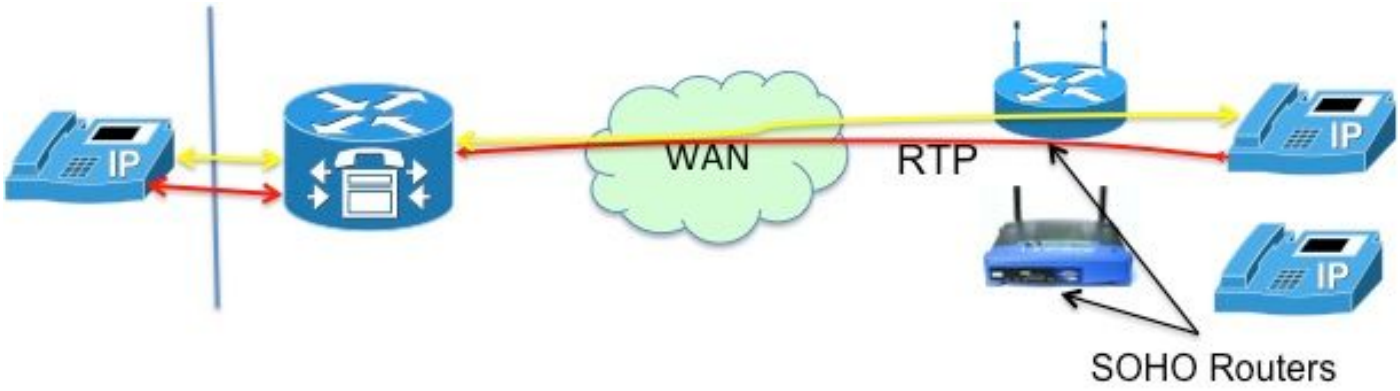


Figure 7

> **Note**: Note that signaling [messages] are always terminated on CM

Remote teleworker

This refers to IP phones connecting to CME over a WAN to support teleworkers who have offices that are remote from the CME router. The most common designs are those involving phones with routable IP addresses and phones with private IP addresses.

Remote phones with public (read: routable) IP addresses

If both the phones involved in the call are configured with public, routable IP addresses, media can flow between the phones directly Figure 8). Therefore, once again, no need for NAT!

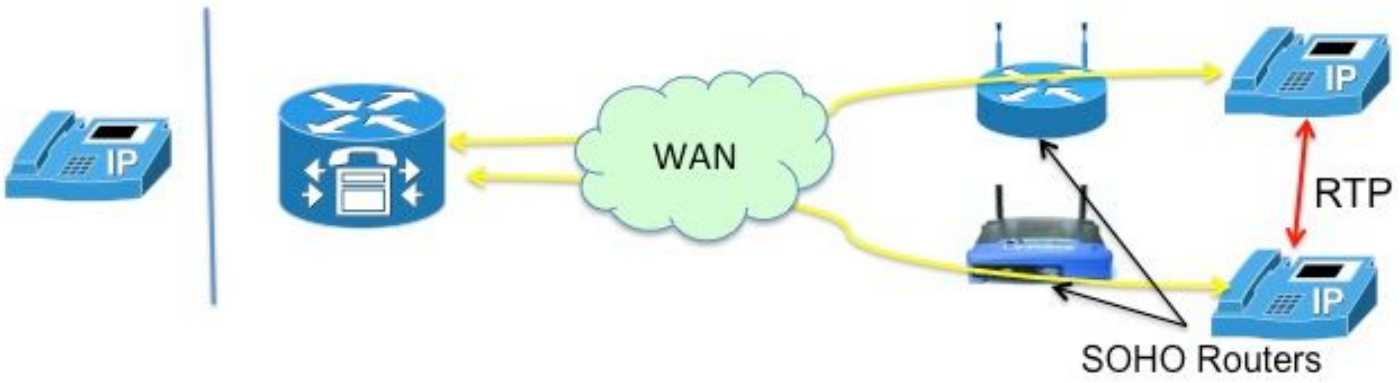


Figure 8

Remote phones with private IP address

In this scenario, the call is signaled between skinny phones configured with private IP addresses. The home office (SOHO) routers, in general, tend to not be“SCCP aware”. i.e. incapable of translating the IP addresses embedded in the SCCP messages. This means that, upon call set up completion, the phones end up with each other’s private IP address. Since both the phones are private, CME will signal the call between them such that the audio flows directly between the phones. This however, will result in one-way or no-way audio (since private IP addresses, by definition, cannot be routed to on the Internet!), unless one of the following workarounds is implemented**-**

· Configure static routes on the SOHO routers

· establish an IPsec VPN connection to the phones

A better way to resolve this would be to configure “mtp”. The mtp command ensures that media (RTP) packets from remote phones transit through the CME router (Figure 9).

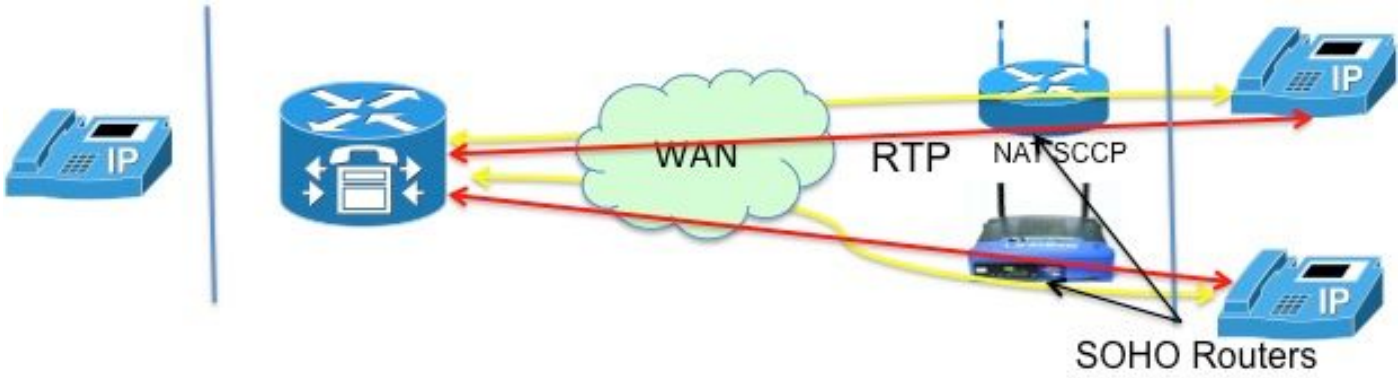


Figure 9

The “mtp” solution is better because of complications with opening up firewall ports. The media packets flowing over a WAN may be obstructed by a firewall. This means that you need to open ports on the firewall, but which ones? With CME relaying the audio, firewalls can be easily configured to pass the RTP packets. CME router uses a ***specific*** UDP port(2000!) for media packets. So, by just allowing packets to and from port 2000, ALL RTP traffic can be passed.

Figure 10 illustrates how to configure mtp.

```
ephone 1
  mac 1111.2222.3333
  type 7965
  mtp
  button 1:1
```

Figure 10

All is not wonderful with mtp. There are situations where mtp may not be desirable

- MTP is not gentle on CPU utilization
- Multicast MOH generally cannot be forwarded over a WAN- The Multicast MOH feature checks to see if MTP is enabled for a phone and if it is, does not send MOH to that phoneL.

Thus, if you have a WAN configuration that **can** forward multicast packets and you can allow RTP packets through your firewall, you can decide not to use MTP.

Remote SIP phones

Note that SIP phones were not mentioned in the above scenarios. This is because of the fact that if one of the phones is a SIP phone, CME inserts itself into the audio path. This then becomes the local-to-remote scenario described earlier, wherein NAT is not required.

CUBE

The CUBE inherently performs NAT and PAT functions as it terminates and re-originates all sessions. The CUBE substitutes its own address for the address of any endpoint it communicates with, thus effectively hiding (translating) the address of that endpoint.

Thus, NAT is not required with the CUBE function. There is a VoIP service scenario in which NAT is required on the CUBE, as described in the next section.

Hosted NAT Traversal

A brief background on Hosted telephony service will help understand the rationale for this feature.

Hosted telephony service is a new form of VoIP service in which most of the gear reside at the service provider’s location. They work with the home gateways (HGW), which implement only basic NAT (i.e. NAT at L3/L4). E.g. Verizon installs the Optical Network Terminal (ONT), which provides FiOS services in the home; voice call is signaled using a SIP process built into the ONT. The SIP signaling is made over Verizon's private IP-network to new soft switches, which provide the service and control to establish voice communications to other FiOS Digital Voice customers, or to traditional phone customers.

Among the key provider requirements for the hosted telephony service include,

- Remote NAT traversal: the ability to deliver Class 5 services to endpoints utilizing NAT (which can only do NAT layer 3!) and firewall devices (by doing "ALG" remotely!)
- Co-media support: the ability to send media between co-located devices where it does not make sense to route the media back to the IP network
- No added equipment, eliminating the need to add any CPE.

Given the above, what options exist to implement such a service ?

- Replace the HGW with an expensive ALG,
- Use a session border controller (SBC) to modify the embedded SIP headers for packets. This involves a network-hosted, carrier-grade product supporting SIP in a very secure, fault-tolerant configuration. This solution is referred to NAT SBC.

The NAT SBC option satisfies the provider requirements listed above.

NAT SBC

The NAT SBC works as follows (Figure 11)

1. Access Router translates only the L3/L4 IP Address
2. IP Address in the SIP message not translated
3. SBC NAT intercepts and translates the embedded IP Address. The moment the SBC sees SIP packets destined to **200.200.200.10**, it kicks in the nat-sbc code.
4. Media is not translated and goes directly between the phones[5]

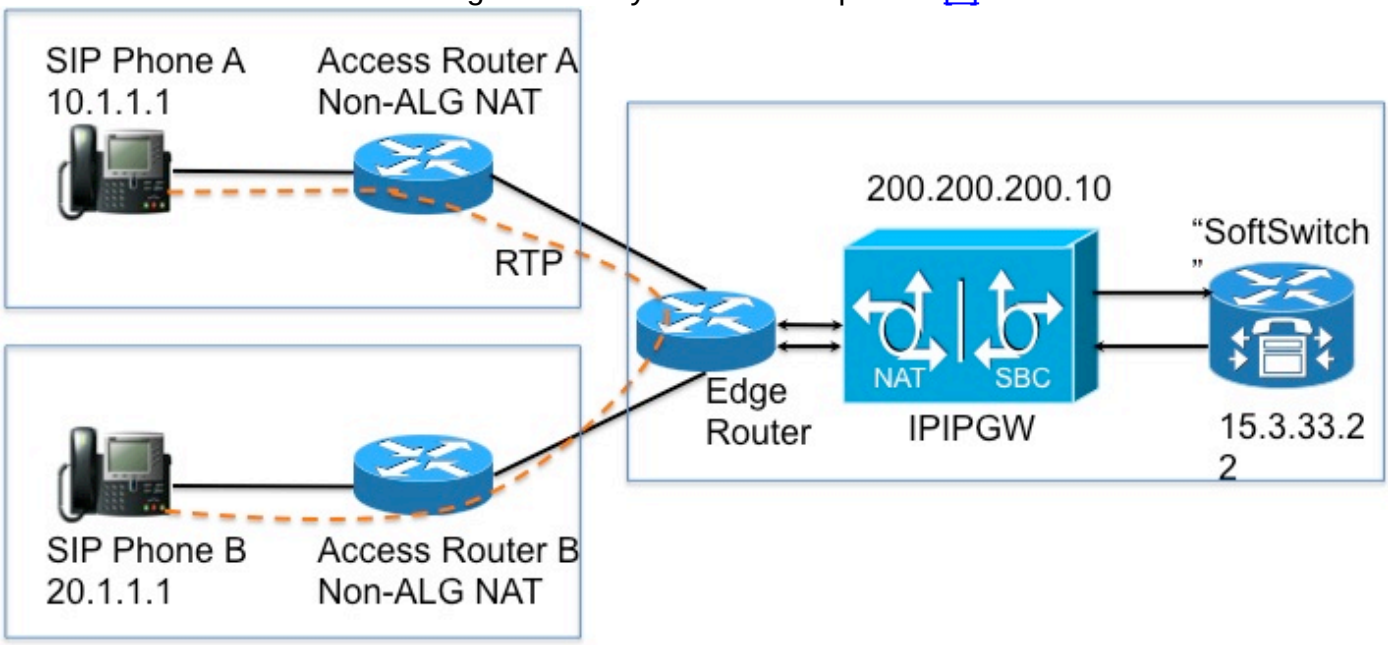


Figure 11

Design Notes

- The IP address **200.200.200.10** (Figure 12) is not assigned to any interface on the NAT SBC. It is configured as the address of the “proxy” to which SIP Phone A and SIP Phone B send signaling messages.
- Home devices do not translate certain SIP/SDP *address-only* fields (e.g. Call-Id: ,O= , Warning: headers & branch= parameter. maddr= and received= parameters were handled in certain scenarios only.). These fields are handled by the NAT SBC, except for the proxy-authorization and authorization translation, because these will break the authentication.
- If the home devices are configured to do PAT, the user agents (phones and proxy) must support symmetric signaling[6] and symmetric and early media. You must configure the override port on the NAT SBC router.
- In the absence of support for symmetric signaling and symmetric and early media, the intermediate routers must be configured without PAT and the override address should be configured in the NAT SBC.

Configuration

Sample configuration for a typical NAT SBC follows.

```
ip nat sip-sbc

 proxy 200.200.200.10 5060 15.3.33.22 5060 protocol udp

 call-id-pool call-id-pool

 session-timeout 300
```

```
  mode allow-flow-around
  override port
!
ip nat pool sbc1 15.3.33.61 15.3.33.69 netmask 255.255.0.0
ip nat pool sbc2 15.3.33.91 15.3.33.99 netmask 255.255.0.0
ip nat pool call-id-pool 1.1.1.1 1.1.255.254 netmask 255.255.0.0
ip nat pool outside-pool 200.200.200.100 200.200.200.200 netmask 255.255.255.0
ip nat inside source list 1 pool sbc1 overload
ip nat inside source list 2 pool sbc2
ip nat outside source list 3 pool outside-pool add-route
ip nat inside source list 4 pool call-id-pool
!
access-list 1 permit 10.1.1.0 0.0.0.255
access-list 1 permit 171.1.1.0 0.0.0.255
access-list 2 permit 20.1.1.0 0.0.0.255
access-list 2 permit 172.1.1.0 0.0.0.255
access-list 3 permit 15.4.0.0 0.0.255.255
access-list 3 permit 15.5.0.0 0.0.255.255
access-list 4 permit 10.1.0.0 0.0.255.255
access-list 4 permit 20.1.0.0 0.0.255.255
```

Call Flow with SBC NAT

Figure 13 and Figure 14 illustrate the call flow in terms of the translations. The following points should be noted-

- Upon registration, the soft switch notes down the two phones as
  -- SIP Phone A – 15.3.33.62 2001

  -- SIP Phone B – 15.3.33.62 2002

- In this call flow, SBC NAT effectively leaves the media IP address un-translated.

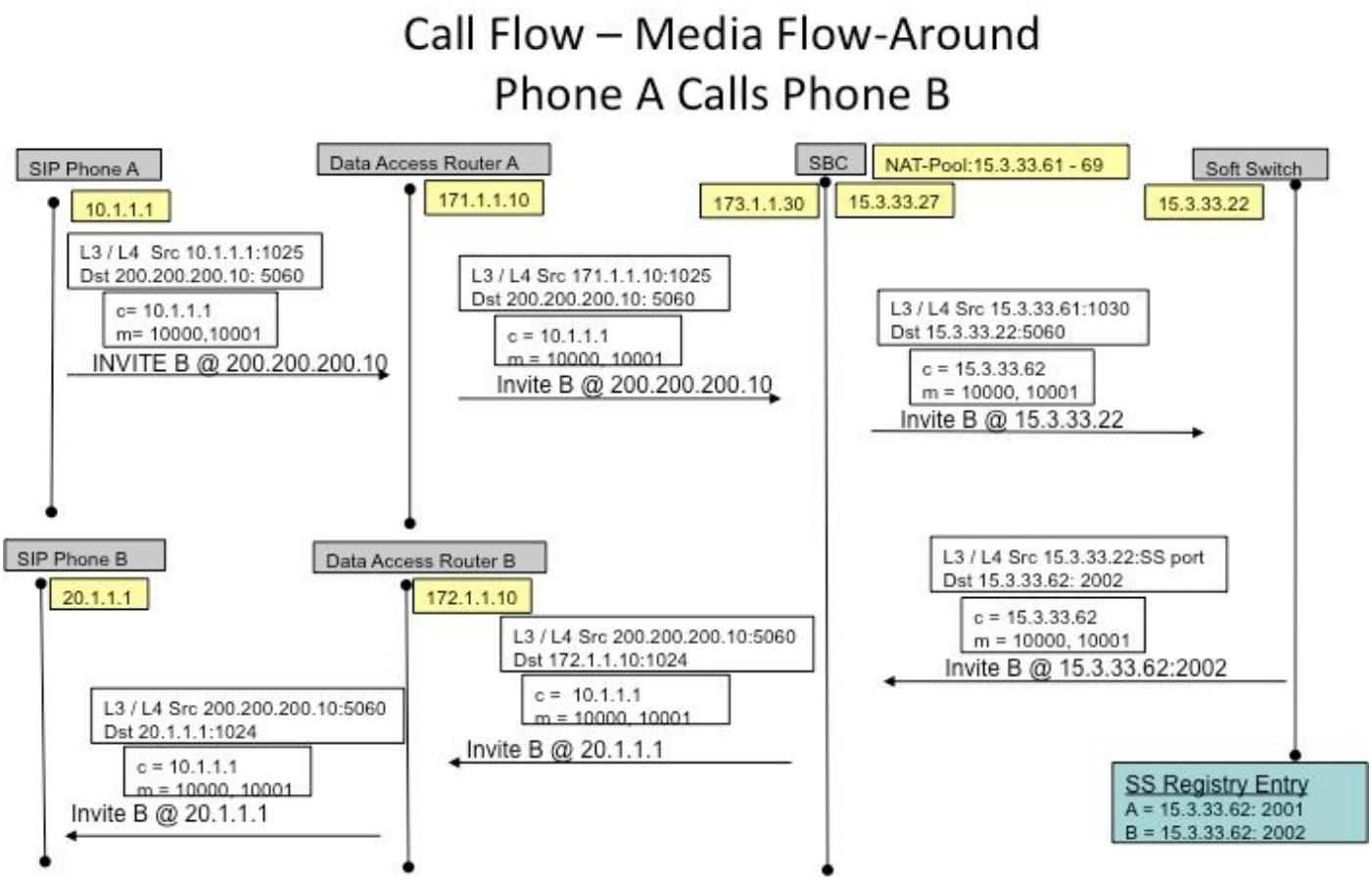


Figure 13

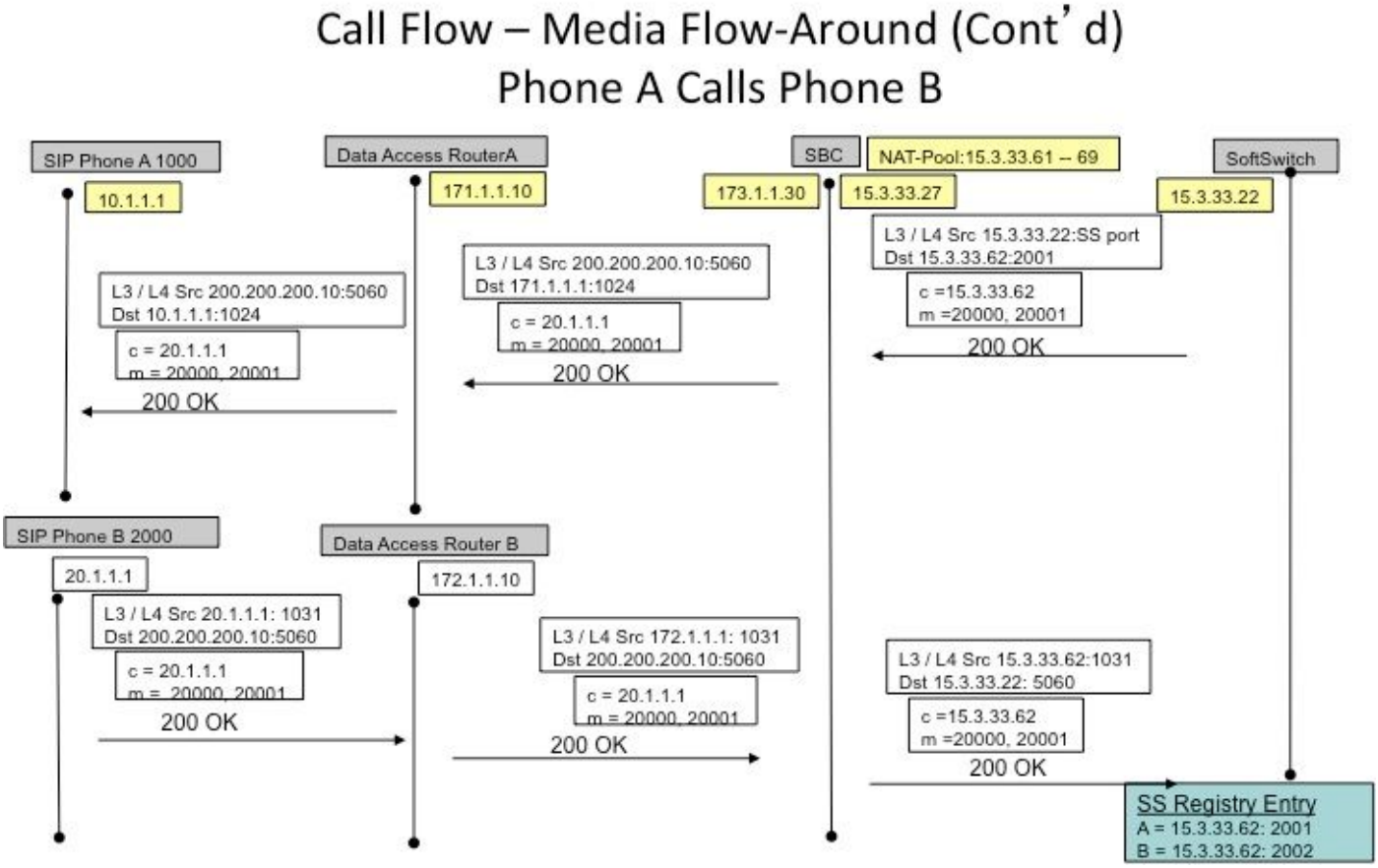


Figure 14

SIP Registration

In earlier versions (of SBC NAT), SIP endpoints had to send *keep-alive* packets to keep the SIP Registration pinhole open (to allow out->in traffic to flow, e.g. inbound calls). *keep-alive* packets could be any SIP packet sent by the endpoint or the registrar (soft switch). Recent versions obviate the need for this, with the NAT-SBC itself (as opposed to soft switches) forcing the endpoints to re-Register frequently to keep the pinholes open.

> **Note**: Symptoms of an expired registration pinhole can be obscure, with random call signaling failures.

CUSP

CUSP has the notion of a logical network, which refers to a collection of local interfaces that are treated similarly for (e.g. interface, port, transport for listening) routing purposes. When configuring a logical network on CUSP, you can configure it to use NAT. Once configured, SIP ALG is automatically enabled. This is useful when certain logical networks.

Troubleshooting

Symptoms

An obvious symptom might be that a call fails in one or both directions. Less obvious symptoms might include,

- One-way audio
- One-way audio on transfer
- No-way audio
- Losing SIP registration

Show and debug commands

- `deb ip nat  [sip | skinny]`
- `show ip nat statistics`
- `show ip nat translations`

Things to check

- Ensure that the configuration includes the **ip nat inside** or **ip nat outside** interface subcommand. These commands enable NAT on the interfaces, and the inside/outside designation is important.
- For static NAT, ensure that the **ip nat source static** command lists the inside local address first and the inside global IP address second.
- For dynamic NAT, ensure that the ACL configured to match packets sent by the inside host match that host’s packets, before any NAT translation has occurred. For example, if an inside local address of 10.1.1.1 should be translated to 200.1.1.1, ensure that the ACL matches source address 10.1.1.1, not 200.1.1.1.

- For dynamic NAT without PAT, ensure that the pool has enough IP addresses. Symptoms of not having enough addresses include a growing value in the second misses counter in the **show ip nat statistics** command output, as well as seeing all the addresses in the range defined in the NAT pool in the list of dynamic translations.
- For PAT, it is easy to forget to add the **overload** option on the **ip nat inside source list** command. Without it, NAT works, but PAT does not, often resulting in users' packets not being translated and hosts not being able to get to the Internet.
- Perhaps NAT has been configured correctly, but an ACL exists on one of the interfaces, discarding the packets. Note that IOS processes ACLs before NAT for packets entering an interface, and after translating the addresses for packets exiting an interface.
- Don't forget to configure "ip nat outside" on the interfacing facing the WAN (even if not translating outside address)!
- As soon as NAT is configured, show ip nat translations doesn't show anything. Ping once and then check again.
- Grab **wireshark Traces** on inside and outside interfaces of the NAT-SBC

Scenarios

Debug output for a couple of scenarios are shown below. They are mostly self-explanatory!

Basic NAT

Configuration and debug lines for basic NAT are shown below.

```
interface Loopback0
 ip address 10.1.1.1 255.255.255.0
 ip nat inside
 ip virtual-reassembly in
!
interface Serial0/1/0
 description **Line to FRS**
 ip address 100.10.10.1 255.255.255.0
 ip nat outside
 ip virtual-reassembly in
 encapsulation ppp
ip nat inside source list 91 interface Serial0/1/0 overload
access-list 91 permit 10.1.1.1

R1#show ip nat translations
Pro Inside global      Inside local       Outside local      Outside global
icmp 100.10.10.1:7     10.1.1.1:7         200.200.200.2:7    200.200.200.2:7
icmp 100.10.10.1:8     10.1.1.1:8         200.200.200.2:8    200.200.200.2:8

R1#ping 200.200.200.2
Type escape sequence to abort.
Sending 5, 100-byte ICMP Echos to 200.200.200.2, timeout is 2 seconds:
!!!!!
R1# sho log
000044: *Apr 17 00:13:00.027: NAT: s=10.1.1.1->100.10.10.1, d=200.200.200.2
[40]
000045: *Apr 17 00:13:00.027: NAT*: s=200.200.200.2, d=100.10.10.1->10.1.1.1
[40]
```

Debug line for NAT on Incoming packet

SIP ALG

Output lines from **debug ip nat sip** are shown. In this case, embedded IP address on an outgoing packet is translated.

```
ip nat inside source static 10.1.1.1 20.1.1.1
------------------------------------------------------------------------
Sent: INVITE sip:1018@10.86.176.142:5060 SIP/2.0
Via: SIP/2.0/UDP 10.1.1.1:5060;branch=z9hG4bK23C1ED01
Remote-Party-ID: "3196" <sip:3196@10.1.1.1>;party=calling;screen=no;privacy=off
From: "3196" <sip:3196@10.1.1.1>;tag=A9F3DB34-EEE
To: <sip:1018@10.86.176.142>
Date: Tue, 23 Apr 2013 17:53:02 GMT
Call-ID: 7A3AC014-AB7511E2-BE6BB2A0-B6AF1B2B@10.1.1.1
--snip--
Contact: <sip:3196@10.1.1.1:5060>
--snip--
v=0
o=CiscoSystemsSIP-GW-UserAgent 9771 5845 IN IP4 10.1.1.1
s=SIP Call
c=IN IP4 10.1.1.1
t=0 0
m=audio 16384 RTP/AVP 18 100 101
c=IN IP4 10.1.1.1
--snip-
------------------------------------------------------------------------
068441: Apr 23 13:53:02.477: NAT: SIP: [0] processing INVITE message
068442: Apr 23 13:53:02.477: NAT: SIP: [0] register:0 door_created:0
--snip--
068447: Apr 23 13:53:02.477: NAT: SIP: [0] translated embedded address 10.1.1.1->20.1.1.1
068448: Apr 23 13:53:02.477: NAT: SIP: [0] register:0 door_created:0
068449: Apr 23 13:53:02.477: NAT: SIP: [0] register:0 door_created:0
068450: Apr 23 13:53:02.477: NAT: SIP: Contact header found
068451: Apr 23 13:53:02.477: NAT: SIP: Trying to find expires parameter
068452: Apr 23 13:53:02.477: NAT: SIP: [0] translated embedded address 10.1.1.1->20.1.1.1
068453: Apr 23 13:53:02.477: NAT: SIP: [0] register:0 door_created:0
068454: Apr 23 13:53:02.477: NAT: SIP: [0] message body found
068455: Apr 23 13:53:02.477: NAT: SIP: Media Lines present:1
068456: Apr 23 13:53:02.477: NAT: SIP: Translated m= (10.1.1.1, 16384) -> (20.1.1.1, 16384)
068457: Apr 23 13:53:02.477: NAT: SIP: old_sdp_len:307 new_sdp_len :307
068458: Apr 23 13:53:02.477: //158107/79BF74A6BE66/SIP/Msg/ccsipDisplayMsg:
```

References

Overview:

- http://www.cisco.com/en/US/partner/technologies/tk648/tk361/tk438/technologies_white_paper09186a0080091cb9.html
- **Anatomy:** http://www.cisco.com/web/about/ac123/ac147/archived_issues/ipj_7-3/anatomy.html
- http://www.cisco.com/en/US/tech/tk648/tk361/technologies_tech_note09186a0080094831.shtml

VoiP and NAT

- https://supportforums.cisco.com/docs/DOC-5406
- http://www.juniper.net/techpubs/software/junos-security/junos-security95/junos-security-swconfig-security/id-60290.html

NAT Feature Matrix

- http://www.cisco.com/en/US/tech/tk648/tk361/technologies_tech_note09186a0080b17919.shtml
- http://www.cisco.com/en/US/technologies/tk648/tk361/tk438/technologies_white_paper09186a00801af2b9.html
- http://www.cisco.com/en/US/tech/tk648/tk361/technologies_tech_note09186a0080b17919.sht

ml

Hosted NAT traversal:

- **www.tmcnet.com/it/0804/FKagoor.htm**

NAT SBC

- EDCS-611622
- EDCS-526070

ALG:

- http://www.cisco.com/en/US/docs/ios-xml/ios/ipaddr_nat/configuration/15-0s/iadnat-applvlgw.html
- http://www.voip-info.org/wiki/view/Routers+SIP+ALG
- http://www.commpartners.us/knowledge/attachments/voip-nat.pdf
- http://www.cisco.com/en/US/partner/docs/ios-xml/ios/ipaddr_nat/configuration/15-mt/nat-tcp-sip-alg.html

CME

- http://www.cisco.com/en/US/docs/voice_ip_comm/cucme/srnd/design/guide/security.html#wp1077376
- http://www.cisco.com/en/US/docs/routers/asr1000/configuration/guide/sbcu/sbc_cucm.html