Understand NAT to Enable Peer-To-Peer Communication on IOS and IOS XE Routers

Contents

Introduction

This document describes the need for Session Traversal Utilities for NAT (STUN) servers, the types of Network Address Translation (NAT) setups with respect to STUN servers, how NAT causes an issue in this setup and the solution.

Background Information

The primary purpose of NAT devices is to allow devices with private IP addresses in a local-area network (LAN) to communicate with devices in public address spaces, such as the Internet. However, although NAT devices are supposed to allow internal hosts to connect with the public space, when it comes to Point-to-Point (P2P) applications like VoIP, gaming, WebRTC, and file-sharing where the end-users need to act as both client and server to maintain 2-way end-to-end communication, NAT provides difficulty to establish those UDP connections. NAT traversal techniques are typically required to make these applications work.

Need for NAT Traversal

Real-time voice and video communication on the Internet are mainstream today with several popular instant messengers (IMs) that support VoIP calls. A big hurdle in the initial adoption of VoIP was the fact that most PCs or other devices sit behind firewalls and use private IP addresses. Multiple private addresses (IP address and port) in the network are mapped to a single public address by a firewall with NAT. But the end device is not aware of its public address, and hence cannot receive voice traffic from the remote party on the private address it advertises in its VoIP communication.

Unilateral Self-Address Fixing (UNSAF) processes are processes whereby some originating endpoint attempts to determine or fix the address (and port) by which it is known to another endpoint - for example, to be able to use address data in the protocol exchange or to advertise a public address from which it receives connections.

The P2P connections under discussion are hence UNSAF processes. One common way P2P applications establish peering sessions and remain NAT-friendly is when they use a publicly addressable rendezvous server for registration and peer discovery purposes.

Session Traversal Utilities for NAT

As per RFC 5389, STUN provides a tool that deals with NATs. It provides a means for an endpoint to determine the IP address and port allocated by a NAT device that corresponds to its private IP address and port. It also provides a way for an endpoint to keep a NAT binding alive.

Types of NAT Implementations

It has been observed that the NAT treatment of UDP varies among implementations. The four treatments observed in implementations are:

Full Cone: A full cone NAT is one where all requests from the same internal IP address and port are mapped to the same external IP address and port. Furthermore, any external host can send a packet to the internal host, and it sends a packet to the mapped external address.

Restricted Cone: A restricted cone NAT is one where all requests from the same internal IP address and port are mapped to the same external IP address and port. Unlike a full cone NAT, an external host (with IP address X) can send a packet to the internal host only if the internal host had previously sent a packet to IP address X.

Port Restricted Cone: A port-restricted cone NAT is like a restricted cone NAT, but the restriction includes port numbers. Specifically, an external host can send a packet, with source IP address X and source port P, to the internal host only if the internal host had previously sent a packet to IP address X and port P.

Symmetric: A symmetric NAT is one where all requests from the same internal IP address and port to a specific destination IP address and port, are mapped to the same external IP address and port. If the same host sends a packet with the same source address and port, but to a different destination, a different mapping is used. Furthermore, only the external host that receives a packet can send a UDP packet back to the internal host.

Consider a topology where the source (A, Pa) (where A is the IP Address, and Pa is the source port) communicates with the destination (B, Pb) and (C, PC) through a NAT device.

Type of NAT implementation	Public source when destined to (B, Pb)	Public source when destined to (C, Pc)	Can destination (for example: (B, Pb)) send traffic to (A, Pa)?
Full Cone	(X1,Px1)	(X1,Px1)	Yes
Restricted Cone	(X1,Px1)	(X1,Px1)	Only if (A, Pa) had first sent the traffic to B
Port Restricted Cone	(X1,Px1)	(X1,Px1)	Only if (A, Pa) had first sent the traffic to (B, Pb)
Symmetric	(X1,Px1)	(X2,Px2)	Only if (A, Pa) had first sent the traffic to (B, Pb)

Issues with NAT Traversal and Symmetric NAT

STUN servers respond to STUN binding requests sent by STUN clients and provide the public

IP/port of the client. Now, this address/port combination is used by the STUN client in its peer-to-peer communication signaling. However, now that the endhost uses the same private address/port (let us assume that is bound to the public IP/port provided in the STUN response) the NAT device translates it to the same IP but a different port if symmetric NAT implementation is used. This breaks the UDP communication because the signalling had established the connection based on the previous port.

Cisco IOS® routers' NAT implementation when it performs PAT is symmetric by default. Therefore, you are expected to see UDP connection issues with these routers that perform NAT.

However, the Cisco IOS-XE routers' NAT implementation when it performs PAT is not symmetric. When you send two different streams with the same source IP and port but to different destinations, the source gets NATED to the same inside global IP and port.

The Solution to the Issue

From this description, it is clear that the issue can be resolved if you perform Endpoint-Independent mapping.

As per RCFC 4787: With Endpoint-Independent Mapping (EIM), the NAT reuses the port mapping for subsequent packets sent from the same internal IP address and port (X:x) to any external IP address and port.

From a client, when the endhost runs the commands **nc -p 23456 10.0.0.4 40000** and **nc -p 23456 10.0.0.5 50000**, on two different terminal windows, here are the results of the NAT translations if you use EIM:

```
Pro Inside global       Inside local             Outside local       Outside global

tcp 10.0.0.1:23456   192.168.0.2:23456  10.0.0.4:40000 10.0.0.4:40000

tcp 10.0.0.1:23456   192.168.0.2:23456  10.0.0.5:50000 10.0.0.5:50000
```

Here you can see that different traffic flows that have the same source address and port get translated to the same address/port regardless of the destination port/address.

On Cisco IOS routers, you can enable Endpoint Agnostic Port Allocation with the command **ip nat service enable-sym-port.**

https://www.cisco.com/c/en/us/td/docs/ios-xml/ios/ipaddr_nat/configuration/15-mt/nat-15-mt-book/iadnat-fpg-port-alloc.html

Summary

Cisco IOS NAT implementation is symmetric by default when you use Port Address Translation (PAT) and it can cause issues when it passes P2P UDP traffic that requires servers like STUN for NAT traversal. You need to explicitly configure EIM on the NAT device to make this work.