MIDCOM
Internet-Draft
Expires: August 22, 2005

J. Rosenberg
Cisco Systems
R. Mahy
Airspace
C. Huitema
Microsoft
February 21, 2005

Traversal Using Relay NAT (TURN)
draft-rosenberg-midcom-turn-07

Status of this Memo

Copyright Notice

Abstract

Traversal Using Relay NAT (TURN) is a protocol that allows for an element behind a NAT or firewall to receive incoming data over TCP or UDP connections. It is most useful for elements behind symmetric NATs or firewalls that wish to be on the receiving end of a

connection to a single peer. TURN does not allow for users to run servers on well known ports if they are behind a nat; it supports the connection of a user behind a nat to only a single peer. In that regard, its role is to provide the same security functions provided by symmetric NATs and firewalls, but to "turn" them into port-restricted NATs.

Table of Contents

1. Introduction

Network Address Translators (NATs), while providing many benefits, also come with many drawbacks. The most troublesome of those drawbacks is the fact that they break many existing IP applications, and make it difficult to deploy new ones. Guidelines [9] have been developed that describe how to build "NAT friendly" protocols, but many protocols simply cannot be constructed according to those guidelines. Examples of such protocols include multimedia applications and file sharing.

Simple Traversal of UDP Through NAT (STUN) [1] provides one means for an application to traverse a NAT. STUN allows a client to obtain a transport address (and IP address and port) which may be useful for receiving packets from a peer. However, addresses obtained by STUN may not be usable by all peers. Those addresses work depending on the topological conditions of the network. Therefore, STUN by itself cannot provide a complete solution for NAT traversal.

A complete solution requires a means by which a client can obtain a transport address from which it can receive media from any peer which can send packets to the public Internet. This can only be accomplished by relaying data though a server that resides on the public Internet. This specification describes Traversal Using Relay NAT (TURN), a protocol that allows a client to obtain IP addresses and ports from such a relay.

Although TURN will almost always provide connectivity to a client, it comes at high cost to the provider of the TURN server. It is therefore desirable to use TURN as a last resort only, preferring other mechanisms (such as STUN or direct connectivity) when possible. To accomplish that, the Interactive Connectivity Establishment (ICE) [13] methodology can be used to discover the optimal means of connectivity.

2. Terminology

In this document, the key words MUST, MUST NOT, REQUIRED, SHALL, SHALL NOT, SHOULD, SHOULD NOT, RECOMMENDED, MAY, and OPTIONAL are to be interpreted as described in RFC 2119 [2] and indicate requirement levels for compliant TURN implementations.

3. Definitions

TURN Client: A TURN client (also just referred to as a client) is an entity that generates TURN requests. A TURN client can be an end system, such as a Session Initiation Protocol (SIP) [6] User Agent, or can be a network element, such as a Back-to-Back User

Agent (B2BUA) SIP server. The TURN protocol will provide the TURN client with IP addresses that route to it from the public Internet.

TURN Server: A TURN Server (also just referred to as a server) is an entity that receives TURN requests, and sends TURN responses. The server is capable of acting as a data relay, receiving data on the address it provides to clients, and forwarding them to the clients.

Transport Address: An IP address and port.

4. Applicability Statement

TURN is useful for applications that require a client to place a transport address into a protocol message, with the expectation that the client will be able to receive packets from a single host that will send to this address. Examples of such protocols include SIP, which makes use of the Session Description Protocol (SDP) [7]. SDP carries and IP address on which the client will receive media packets from its peer. Another example of a protocol meeting this criteria is the Real Time Streaming Protocol (RTSP) [8].

When a client is behind a NAT, transport addresses obtained from the local operating system will not be publically routable, and therefore, not useful in these protocols. TURN allows a client to obtain a transport address, from a server on the public Internet, which can be used in protocols meeting the above criteria. However, the transport addresses obtained from TURN servers are not generally useful for receiving data from anywhere. They are only useful for communicating with a single peer. This is accomplished by having the TURN server emulate the behavior of a port-restricted NAT. In particular, the TURN server will only relay packets from an external IP address and port towards the client if the client had previously sent a packet through the TURN server towards that IP address and port. As a result of this, when a TURN server is placed in front of a symmetric NAT, the resulting combined system has identical security properties to a system that just had a port-restricted NAT. Since clients behind such devices cannot run public servers, they cannot run them behind TURN servers either.

5. Overview of Operation

The typical TURN configuration is shown in Figure 1. A TURN client is connected to private network 1. This network connects to private network 2 through NAT 1. Private network 2 connects to the public Internet through NAT 2. On the public Internet is a TURN server.

DOCKET ALARM

Explore Litigation Insights

Docket Alarm provides insights to develop a more informed litigation strategy and the peace of mind of knowing you're on top of things.

Real-Time Litigation Alerts

Keep your litigation team up-to-date with **real-time alerts** and advanced team management tools built for the enterprise, all while greatly reducing PACER spend.

Our comprehensive service means we can handle Federal, State, and Administrative courts across the country.

Advanced Docket Research

With over 230 million records, Docket Alarm's cloud-native docket research platform finds what other services can't. Coverage includes Federal, State, plus PTAB, TTAB, ITC and NLRB decisions, all in one place.

Identify arguments that have been successful in the past with full text, pinpoint searching. Link to case law cited within any court document via Fastcase.

Analytics At Your Fingertips

Learn what happened the last time a particular judge, opposing counsel or company faced cases similar to yours.

Advanced out-of-the-box PTAB and TTAB analytics are always at your fingertips.

API

Docket Alarm offers a powerful API (application programming interface) to developers that want to integrate case filings into their apps.

LAW FIRMS

Build custom dashboards for your attorneys and clients with live data direct from the court.

Automate many repetitive legal tasks like conflict checks, document management, and marketing.

FINANCIAL INSTITUTIONS

Litigation and bankruptcy checks for companies and debtors.

E-DISCOVERY AND LEGAL VENDORS

Sync your system to PACER to automate legal marketing.

WHAT WILL YOU BUILD? | **sales@docketalarm.com** | **1-866-77-FASTCASE**

fastcase®
Smarter legal research.