TRAM
Internet-Draft
Intended status: Informational
Expires: November 30, 2015

P. Martinsen
T. Andersen
G. Salgueiro
Cisco
M. Petit-Huguenin
Impedance Mismatch

May 29, 2015

Traversal Using Relays around NAT (TURN) Bandwidth Probe draft-martinsen-tram-turnbandwidthprobe-00

Abstract

Performing pre-call probing to discover a reasonable value for the available bandwidth, is useful information that can be utilized by bandwidth sensitive or bandwidth intensive network devices (e.g., video encoders). The method described herein is intended to produce an initial bandwidth value. Applications using this mechanism should also employ appropriate rate adaptation techniques. In addition to bandwidth, latency and bufferbloat can also be measured. No modification is needed on the server side.

Status of This Memo

This Internet-Draft is submitted in full conformance with the provisions of BCP 78 and BCP 79.

Internet-Drafts are working documents of the Internet Engineering Task Force (IETF). Note that other groups may also distribute working documents as Internet-Drafts. The list of current Internet-Drafts is at http://datatracker.ietf.org/drafts/current/.

Internet-Drafts are draft documents valid for a maximum of six months and may be updated, replaced, or obsoleted by other documents at any time. It is inappropriate to use Internet-Drafts as reference material or to cite them other than as "work in progress."

This Internet-Draft will expire on November 30, 2015.

Copyright Notice

Table of Contents

1. Introduction

When Interactive Connectivity Establishment (ICE) [RFC5245] and Traversal Using Relays around NAT (TURN) [RFC5766] are used by an endpoint as a firewall/NAT traversal mechanism, the TURN relay can also be used to measure bandwidth and latency prior to call setup.

In normal ICE behavior the client first sends a message (allocate request) to the TURN server to allocate a RELAY address. This address can be used by the endpoint to receive media from other endpoints. The media stream is then received by the TURN server and then relayed back to the endpoint behind the firewall/NAT. For security reasons the endpoint must first set the correct permissions on the TURN server to only allow media from remote participants it wants to communicate with (i.e., addresses taken from the Session Initiation Protocol (SIP) [RFC3261] Session Description Protocol (SDP) [RFC4566] offer/answer exchange [RFC3264]) . The endpoint will also learn its reflexive address on the firewall/NAT when talking to the TURN server.

Combining this with a TCP transfer on the same TURN server can be used to also measure bufferbloat, an important metric for multimedia applications.

Note that only the maximum bandwidth, maximum latency and maximum bufferbloat of the aggregation of both uplink and downlink can be measured. It is not possible with this technique to get the metrics of only one. For most multimedia applications using TURN that is not an issue as they are generally symmetrical, but some other use cases (like conferencing) may need other techniques to measure these metrics separately.

No modification to the TURN server is necessary.

2. Notational Conventions

The key words "MUST", "MUST NOT", "REQUIRED", "SHALL", "SHALL NOT", "SHOULD", "SHOULD NOT", "RECOMMENDED", "MAY", and "OPTIONAL" in this document are to be interpreted as described in [RFC2119].

3. Overview of Operation

Prior to the call (upon registering with the call control server, receiving a configuration, loading application, or a similar event) the endpoint can measure bandwidth and latency between the endpoint and the TURN server.

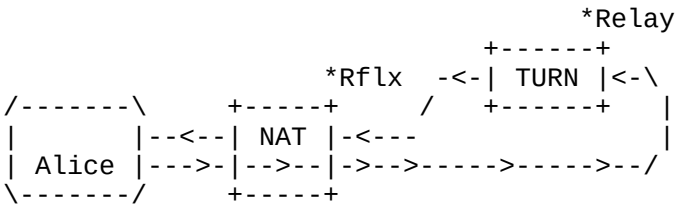

Figure 1

The agent allocates a TURN Relay port on its designated TURN server as described in TURN RFC [RFC5766]. In the process the agent will also learn the outermost NAT address. This is called a reflexive address (Rflx). For more information see Section 2.1 of the TURN RFC regarding candidate gathering in ICE.

The agent must set the permissions on the allocated RELAY port as described in Section 8 of the TURN RFC to allow traffic from the discovered reflexive address.

When sending packets to the allocated RELAY port on the TURN server, the packets will be forwarded back to the agent in a data indication packet. See Section 10 of the TURN RFC for details on how the TURN server can relay packets back to the allocating agent. Available bandwidth can be measured by sending varying number of packets and detecting the amount of packet loss. Each packet sent affects both upstream and downstream links.

To make it easier to calculate the available bandwidth a TIMESTAMP attribute is defined in this document (see Section Section 6.1) and can be added to the Session Traversal Utilities for NAT (STUN) [RFC5389] probe packets. The PADDING attribute from the NAT Behavior Discovery Using STUN RFC [RFC5780] can be used to vary the packet size.

Discovering the MTU and network path (using the STUN-PMTUD [I-D.petithuguenin-tram-stun-pmtud] and STUN Traceroute [I-D.martinsen-tram-stuntrace] mechanisms) can also be performed when probing for the bandwidth available between the client and the TURN server.

4. Base Protocol Procedures

In order to perform the STUN bandwidth probing mechanism described in this document, the client MUST take the following general steps (explained in greater detail in the following subsections).

- Allocate TURN RELAY address
- Set correct permissions on the allocated TURN RELAY address
- Originating client sends data to itself through the TURN server and measures bandwidth throughput and latency

When initiating a bandwidth probe it is important to not do so when a device powers up or some similar initiating events. If a power failure has happened and all devices within an area are rebooted concurrently the bandwidth probing of all the devices can have a DDOS-like effect. Measures should be taken to avoid such scenarios (e.g., random delays to initiate bandwidth probing, etc).

Discovery of the TURN server as well as the determination of what TURN server to uses is entirely at the discretion of the client and outside the scope of this document. A client MUST be prepared to be

redirected to another TURN server if it receives an ALTERNATE-SERVER response.

While allocating the TURN RELAY port the client will learn its outermost NAT address or reflexive address. This is the address the TURN server will receive the bandwidth probing packets from.

The bandwidth mechanism can use either a UDP transport or a TCP. Secure transports (i.e. TLS or DTLS) may be used to discover if an intermediary network element tries to process flows differently when they are secured.

4.1. UDP Procedures

The client allocates a TURN RELAY port as described in the TURN RFC. The client then use a CreatePermission request with the obtained reflexive address encoded in a XOR-PEER-ADDRESS attribute as described in Section 9.1 of the TURN RFC.

It is recommended to create a TURN channel as soon as possible to lower the overhead of the packets exchanged.

If the transport address used to send the UDP packets to the TURN relay is identical to the transport address used to create the TURN allocation, then a TURN Channel can be created immediately by using the reflexive transport address learned during the Allocate.

If not, the TURN Channel can be created as soon the first Data indication is received.

The client can then send UDP packets to the relay transport address and receive then over the TURN Channel.

Immediately after this the client can send UDP packets over the TURN channel and receive them directly, as an additional way of averaging the impact of the difference of encapsulation for the packets. Note that the client still need to periodically send packets over the TURN Channel to persist eventual NAT bindings.

Note that the client cannot use a TCP transport to the server with a UDP allocation because there would be no way to retrieve the UDP reflexive address for the CreatePermission request.

4.2. TCP Procedures

The client allocates a TURN RELAY port as described in TURN Extensions for TCP Allocations [RFC6062]. The client then use a CreatePermission request with the obtained reflexive address encoded

in a XOR-PEER-ADDRESS attribute as described in Section 9.1 of the TURN RFC.

The client then establishes a TCP connection to the relay transport address. The client will receive a ConnectAttempt indication that will trigger a new TCP connection to the TURN server, and the sending of a ConnectBind.

After completion of this procedure, data sent over the direct TCP connection will be received over the bound TURN connection, and vice-versa, although there is no difference of overhead in that case.

4.3. Sending Data to Measure Available Bandwidth and Latency

The specific calculation and measurement of the bandwidth is client dependent and implementation-specific and is thus outside the scope of this document.

If the client want to use STUN packets as the basis for the probing packets, then a TIMESTAMP attribute is defined in this specification (see Section Section 6.1) to simplify measurement of round-trip time (RTT) and available bandwidth. A PADDING attribute is already defined in RFC 5780 [RFC5780] that makes it easy to vary the size of the STUN probing packet.

The probing packet will be sent upstream to the TURN server and later received downstream from the TURN server. Available bandwidth would typically be determined to be the lowest of the bandwidth values calculated for the upstream and downstream directions.

If the RTP [RFC3550] loop-back mechanism described in RFC 6849 [RFC6849] is in use the method described here can extend the use-cases mentioned in RFC6849 Section 1.1 to enable the "loopback source" and "loopback mirror" to be running on the same device. Using RTP would permit to reuse the standards RTP tools for calculating latency, jitter and other metrics. It may also permit to get better results if some intermediary network element has preferential treatment for media packets.

The client should take care to reuse the same congestion control mechanisms it uses when sending media to avoid unnecessary strain on the network.

5. IANA Considerations

This specification defines a new STUN attribute. IANA added this new attribute to the STUN Attributes sub-registry of the Session

Traversal Utilities for NAT (STUN) Parameters registry. (This is still an ID draft so not assignment yet)

6. New STUN Attribute

This STUN extension defines the following new attribute:

0xXXX0: TIMESTAMP

6.1. TIMESTAMP

The TIMESTAMP attribute has a length of 80 bits. Padding is needed to hit the required 32 bit STUN attribute boundary.

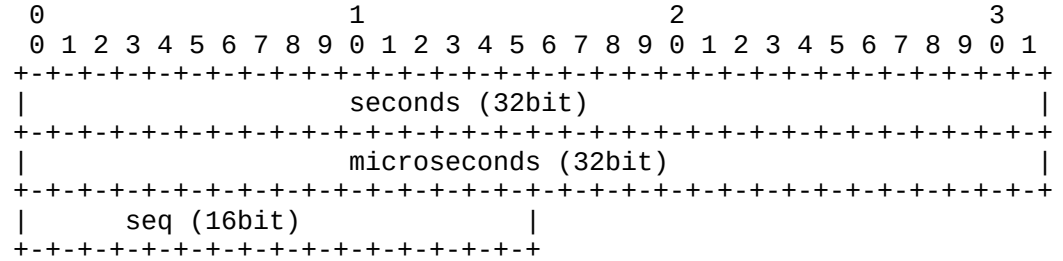

Figure 2: TIMESTAMP Attribute

The seconds and microseconds fields reflect what would be returned in the struct timeval when calling getTimeofDay() function. Note that the size of that struct may vary based on platform, but 32 bits is more than sufficient to obtain the required accuracy for the feature described in this document. It is RECOMMENDED to initialize these fields with a random value that later can be subtracted to get the right timing.

The seq field is a 16 bit sequence number. It is increased by one for each bandwidth probe STUN packet sent. It is RECOMMENDED to choose a random starting value.

7. Implementation Status

[[Note to RFC Editor: Please remove this section and the reference to [RFC6982] before publication.]]

This section records the status of known implementations of the protocol defined by this specification at the time of posting of this Internet-Draft, and is based on a proposal described in RFC 6982

[RFC6982]. The description of implementations in this section is intended to assist the IETF in its decision processes in progressing drafts to RFCs. Please note that the listing of any individual implementation here does not imply endorsement by the IETF. Furthermore, no effort has been spent to verify the information presented here that was supplied by IETF contributors. This is not intended as, and must not be construed to be, a catalog of available implementations or their features. Readers are advised to note that other implementations may exist.

According to RFC 6982 [RFC6982], "this will allow reviewers and working groups to assign due consideration to documents that have the benefit of running code, which may serve as evidence of valuable experimentation and feedback that have made the implemented protocols more mature. It is up to the individual working groups to use this information as they see fit"

7.1. Cisco Collaboration Endpoint (CE)

Organization: Cisco

Name: Cisco Collaboration Endpoints (CE) software

Description: Hard video endpoint part of the Cisco collaboration portfolio

Level of maturity: In released products

Coverage Implementation of base procedures of the functionality described in this specification

Licensing: Proprietary

Implementation experience: Straight forward, but implementation was don prior to writing up the spec

Contact: Paal-Erik Martinsen (palmarti@cisco.com)

8. Security Considerations

When setting permissions this is done on a per IP address basis. Port number is not part of the permission. This is necessary limitation of the TURN protocol [RFC5766] and not something introduced by this specification.

To prevent replay attacks or other attacks that rely on static sequence number initialization it is important to randomly initialize the seq number in the TIMESTAMP Attribute. Likewise it is important

to hide the time information by assigning a random value to the seconds and microseconds fields. That random value can be added and subtracted by the client when sending and receiving packets to get the correct value. This prevents any information leakage regarding time from the client.

9. Acknowledgements

Thanks to Dan Wing for input.

10. References

10.1. Normative References

[RFC2119] Bradner, S., "Key words for use in RFCs to Indicate Requirement Levels", BCP 14, RFC 2119, March 1997.

[RFC3550] Schulzrinne, H., Casner, S., Frederick, R., and V. Jacobson, "RTP: A Transport Protocol for Real-Time Applications", STD 64, RFC 3550, July 2003.

[RFC5245] Rosenberg, J., "Interactive Connectivity Establishment (ICE): A Protocol for Network Address Translator (NAT) Traversal for Offer/Answer Protocols", RFC 5245, April 2010.

[RFC5389] Rosenberg, J., Mahy, R., Matthews, P., and D. Wing, "Session Traversal Utilities for NAT (STUN)", RFC 5389, October 2008.

[RFC5766] Mahy, R., Matthews, P., and J. Rosenberg, "Traversal Using Relays around NAT (TURN): Relay Extensions to Session Traversal Utilities for NAT (STUN)", RFC 5766, April 2010.

[RFC5780] MacDonald, D. and B. Lowekamp, "NAT Behavior Discovery Using Session Traversal Utilities for NAT (STUN)", RFC 5780, May 2010.

[RFC6062] Perreault, S. and J. Rosenberg, "Traversal Using Relays around NAT (TURN) Extensions for TCP Allocations", RFC 6062, November 2010.

[RFC6849] Kaplan, H., Hedayat, K., Venna, N., Jones, P., and N. Stratton, "An Extension to the Session Description Protocol (SDP) and Real-time Transport Protocol (RTP) for Media Loopback", RFC 6849, February 2013.

[RFC6982] Sheffer, Y. and A. Farrel, "Improving Awareness of Running Code: The Implementation Status Section", RFC 6982, July 2013.

10.2. Informative References

[I-D.martinsen-tram-stuntrace]
Martinsen, P. and D. Wing, "STUN Traceroute", draft-martinsen-tram-stuntrace-00 (work in progress), February 2015.

[I-D.petithuguenin-tram-stun-pmtud]
Petit-Huguenin, M., "Path MTU Discovery Using Session Traversal Utilities for NAT (STUN)", draft-petithuguenin-tram-stun-pmtud-00 (work in progress), January 2015.

[RFC3261] Rosenberg, J., Schulzrinne, H., Camarillo, G., Johnston, A., Peterson, J., Sparks, R., Handley, M., and E. Schooler, "SIP: Session Initiation Protocol", RFC 3261, June 2002.

[RFC3264] Rosenberg, J. and H. Schulzrinne, "An Offer/Answer Model with Session Description Protocol (SDP)", RFC 3264, June 2002.

[RFC4566] Handley, M., Jacobson, V., and C. Perkins, "SDP: Session Description Protocol", RFC 4566, July 2006.

Authors' Addresses

Paal-Erik Martinsen
Cisco Systems, Inc.
Philip Pedersens Vei 22
Lysaker, Akershus 1325
Norway

Email: palmarti@cisco.com

Trond Andersen
Cisco Systems, Inc.
Philip Pedersens Vei 22
Lysaker, Akershus 1325
Norway

Email: trondand@cisco.com

Gonzalo Salgueiro
Cisco Systems, Inc.
7200-12 Kit Creek Road
Research Triangle Park, NC 27709
US

Email: gsalguei@cisco.com

Marc Petit-Huguenin
Impedance Mismatch

Email: marc@petit-huguenin.org