NAT and Firewall Traversal with STUN / TURN / ICE

Simon Perreault
Viagénie

mailto:Simon.Perreault@viagenie.ca
http://www.viagenie.ca

Copyright Viagénie 2008
Credentials

- Consultant in IP networking and VoIP at Viagénie.
- Developed Numb, a STUN / TURN server.
- Ported FreeSWITCH to IPv6.
- Co-ported Asterisk to IPv6.
- Developed many custom VoIP applications.
Plan

- The problem of NAT and firewalls in VoIP
- How STUN, TURN, and ICE solve it
- Asterisk specifics
- Wireshark traces
The Problem of NAT and Firewalls in VoIP

- Network address translators (NATs) are common devices that “hide” private networks behind public IP addresses.
- Connections can be initiated from the private network to the Internet, but not the other way around.
- Having separate addresses for signaling and media makes the situation worse.
A NAT device works by associating a public address and port with a private destination address and port.

<table>
<thead>
<tr>
<th>Public</th>
<th>Private</th>
</tr>
</thead>
<tbody>
<tr>
<td>206.123.31.67 : 55123</td>
<td>192.168.1.2 : 5060</td>
</tr>
</tbody>
</table>

- Valid for duration of flow
  - Meaning of “flow” for UDP?
  - Must be kept alive.

- Useful to discover this address.
STUN

● Session Traversal Utilities for NAT (STUN): simple protocol for discovering the server-reflexive address.
  – Client: Where do you see me at?
  – Server: I see you at 206.123.31.67:55123.

● A STUN server is located in the public Internet or in an ISP's network when offered as a service.
  – Double NATs pose an interesting problem...
STUN Flow Diagram

STUN client 192.168.201.128

STUN Binding Request
Source: 192.168.201.128:45897

NAT 192.168.201.2 - 206.123.31.67

STUN Binding Response
Destination: 192.168.201.128:45897
Payload: 206.123.31.67:55123

STUN server 64.251.14.14

STUN Binding Request
Source: 206.123.31.67:55123

STUN Binding Response
Destination: 206.123.31.67:55123
Payload: 206.123.31.67:55123

Copyright Viagénie 2008
STUN

- It turns out that some NAT devices try to be clever by inspecting the payloads and changing all references to the server-reflexive address into the private address.
- STUN2 obfuscates the address by XORing it with a known value.
- TCP and UDP are supported over IPv4 and IPv6.
Server-Reflexive Address

- A client who knows its server-reflexive address could use it in place of its private address in the SIP headers.
  - Not the intended usage. See *sip-outbound* IETF draft.
- Intended usage: RTP ports.
- RTP port $\rightarrow$ NAT binding $\rightarrow$ STUN request
Symmetric NATs

- Some NAT devices only allow packets from the remote peer to reach the NATed peer:
  - Address dependent
  - Port dependent
  - Both
  - Implication: knowing server-reflexive address is useless.

- These NAT devices are called symmetric NATs.
  - Often “enterprise” NATs ⇒ many devices.
  - Significant presence, must be worked around.
TURN

• Makes devices behind symmetric NATs reachable.
  – Device initiates and maintains connection to relay.

• Traversal Using Relays around NAT (TURN)
  – Protocol between NATed device and relay.
  – Built on top of STUN.

• TURN server is located outside the NAT.
  – On the public Internet
  – or in an ISP's network when offered as a service by the ISP.
TURN Flow Diagram

TURN client
192.168.201.128

TURN Allocate

Allocate Response
Relayed address:
64.251.14.14:51292

Keep-alive

SIP Invite
SDP c= line:
64.251.14.14:51292

TURN Data Indication + RTP packet

NAT

TURN Allocate

Allocate Response
Relayed address:
64.251.14.14:51292

Allocate a port

SIP Invite
SDP c= line:
64.251.14.14:51292

TURN Allocate

TURN Data Indication + RTP packet

TURN server
64.251.14.14

RTP packet

SIP peer

Allocate a port

Copyright Viagénie 2008
Relayed Address

The address allocated by the TURN server is called the *relayed address*.
- TURN server communicates it to TURN client.
- TURN client communicates it to SIP peer.

The TURN client may use it in the SIP headers.

Intended usage: RTP ports.

RTP port $\rightarrow$ NAT binding $\rightarrow$ TURN allocation

**TURN guarantees** communication in all NAT cases unless there is an explicit firewall policy to prohibit its use.
Disadvantages of TURN

• TURN server is in forwarding path.
  – Requires a lot of bandwidth.
  – Server must remain available for the whole duration of the allocation.
  – Triangle routing results in longer path.

• Encapsulation.
  – Lowers MTU (not so much a problem for VoIP packets).
  – Additional headers consume a bit more bandwidth.
  – Firewall must inspect payload to discover real sender.

• Allocation must be kept alive.
Disadvantages of TURN

- ICMP not relayed.
  - No path MTU discovery.
- TTL not properly decremented.
  - Possibility of loops.
- DiffServ (DS) field not relayed.
- As of now only IPv4 and UDP.
Mitigating Mechanisms

- Availability and scalability provided by anycast.
  - Only used for discovery, server must remain up for the duration of the allocation.

- Channel mechanism for minimizing header size.
  - 4 bytes only.

- Permission mechanism enforced by TURN server.
  - Only peers previously contacted by client may send data to relayed address.
  - Firewall may “trust” the TURN server, no payload inspection.

- Keep TURN server close to NAT device.
  - Offered as a service by ISPs.
IPv4 and IPv6 Interoperability

- TURN will also be used to relay packets between IPv4 and IPv6.
- Alleviates load from the B2BUA.
  - Designed for relaying performance.
  - Anycast ensures scalability and reliability.
- TURNv6 draft still in progress.
Numb

- Numb is a STUN and TURN server developed by Viagénie.
  - Supports IPv4 and IPv6 in mixed scenarios.
  - Supports anycast.
- Free access at http://numb.viagenie.ca
- To install it in your own network, contact us: info@viagenie.ca
Connectivity Establishment

• Many addresses may be available:
  – Host addresses.
  – Server-reflexive address.
  – Relayed address.
  – Each in IPv4 and IPv6 flavour!
  – Each in UDP and TCP flavour!

• Which one to choose?

• Need for an automatic *connectivity establishment* mechanism.
Interactive Connectivity Establishment (ICE)

- Conceptually simple.
  - Gather all candidates (using STUN/TURN).
  - Order them by priority.
  - Communicate them to the callee in the SDP.
  - Do connectivity checks.
  - Stop when connectivity is established.

- Gnarly details:
  - Keep candidates alive.
  - Agree on priority.
  - Reduce delays and limit packets.
Peer-Reflexive Address

• Remember: Server-reflexive address useless with symmetric NAT.

• Address as seen from peer (instead of STUN server) is peer-reflexive address.
  - Works even with a symmetric NAT.
    • but not two of them (TURN still necessary).

• During ICE connectivity checks, peer-reflexive candidates are gathered and prepended to check list.

• Information reuse between ICE instances.
Examples

DNS server
206.123.31.2
2620:0:230:8000:2

STUN server
64.251.14.14
64.251.22.149

NAT + DNS server
206.123.31.67
2620:0:230:c000:67

SIP registrar
206.123.31.98
2620:0:230:c000:98

192.168.201.2
192.168.201.128
Asterisk Specifics

- NAT traversal in 1.6 was greatly enhanced
  - Can define internal NATed network (*localnet*)
  - Can determine external address either...
    - directly (*externip*)
    - via dynamic DNS (*externhost*)
    - with a **STUN client** (*stunaddr*)
- RFC 3581 rport mechanism (*nat = yes*)
- Don't re-INVITE internal <-> external calls (*canreinvite = nonat*)
Deployment

- ISPs are deploying STUN / TURN servers within their network.
- TURN a part of the IPv6 migration.
- SIP client vendors are implementing ICE.
- B2BUAs also should implement ICE.
Conclusion

• Discussed
  – The problem of NAT and firewalls in VoIP
  – How STUN, TURN, and ICE solve it
    • Obtaining a server reflexive address via STUN
    • Obtaining a relayed address via TURN
    • Telling the other party about these addresses via ICE
    • Making connectivity checks
    • Obtaining peer reflexive addresses
  • STUN / TURN / ICE stack too thick? Use IPv6!
Questions?

Simon.Perreault@viagenie.ca

This presentation: http://www.viagenie.ca/publications/
STUN / TURN server: http://numb.viagenie.ca

References:
ICE draft: http://tools.ietf.org/html/draft-ietf-mmusic-ice

Copyright Viagénie 2008