PPSP NAT traversal

draft-li-ppsp-nat-traversal-02

Lichun Li, Jun Wang, Wei Chen
{li.lichun1, wang.jun17}@zte.com.cn
chenweiyj@chinamobile.com

Changes since 01

- Make change to PPSP-ICE solution
  - Peer tries to contact remote peer's host addr and reflexive addr first.
  - If it fails, it turns to exchange ICE parameters and perform ICE process.
- Propose PPSP protocol extensions for NAT traversal
  - Mainly based on gu-ppsp-tracker-protocol and gu-peer-protocol
  - Mainly required by PPSP-ICE.
  - RELOAD-ICE may also use some extensions.

NAT traversal solution overview

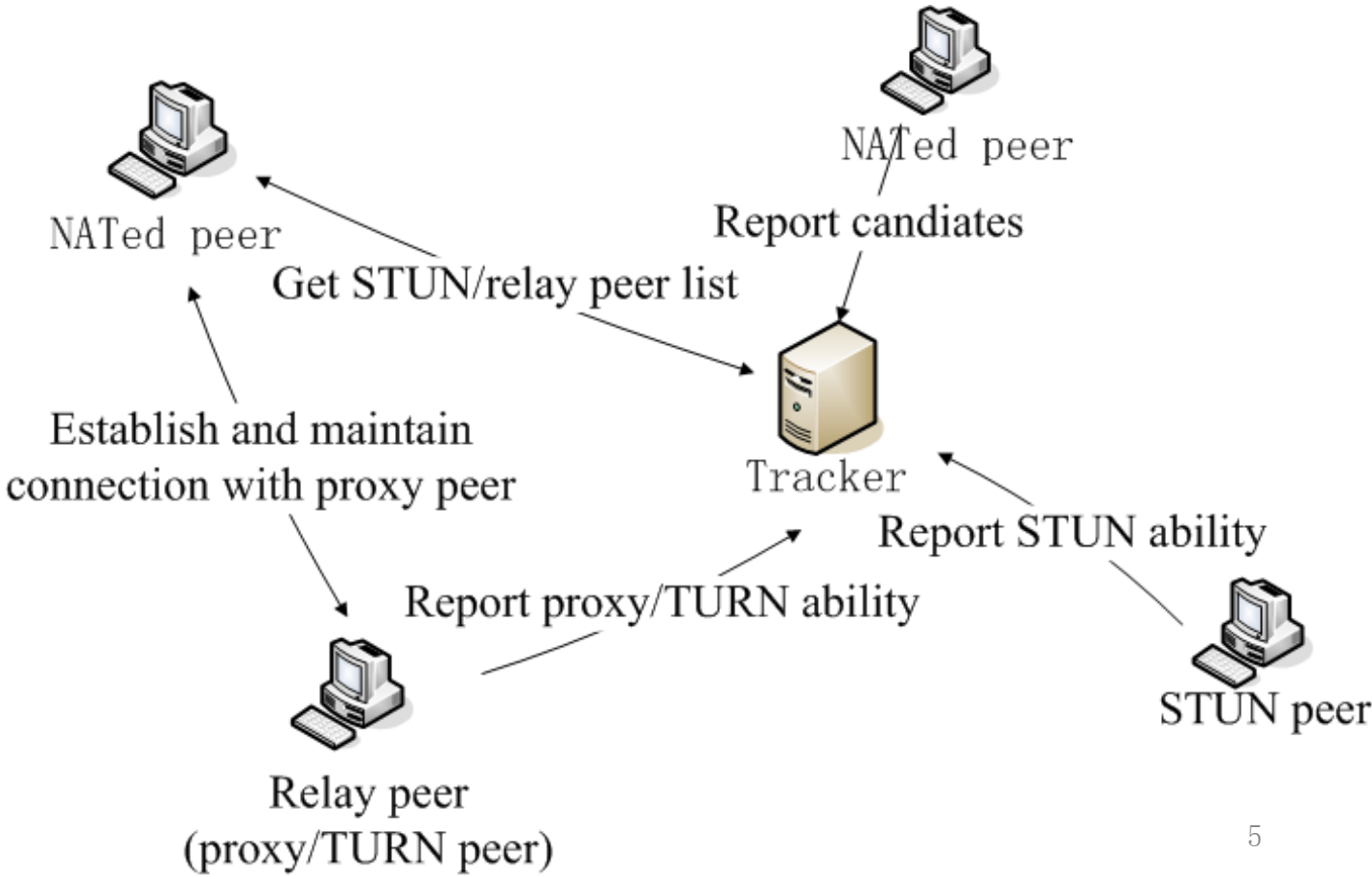
- ICE is the standard NAT traversal solution.
- ICE requires application to exchange ICE parameters.
- Two solutions in the draft: PPSP-ICE and RELOAD-ICE
  - both use ICE
  - but use PPSP and RELOAD separately to exchange ICE parameters (candidates and credentials for connectivity check).
- Candidate
  - Candidate (from ICE RFC5245): A transport address that is a potential point of contact for receipt of media.
  - Assigned by host itself, NAT device or relay node.
  - NATed peers need NAT devices or NAT traversal service nodes to discover/assign candidates.

NAT traversal service nodes and candidates

- Dedicated STUN/TURN server
  - TURN: application-agnostic relay, assign relayed candidates to NATed peers
  - STUN: Discover reflexive candidates for NATed peers
- STUN/TURN peer
  - Peer providing STUN/TURN service
- Proxy peer (mandatory for PPSP-ICE)
  - Peer providing application layer relay. Analogy to SIP or RELOAD relay
  - Assign proxy candidates to NATed peers
- STUN-like tracker
  - Tracker providing STUN-like function with PPSP message. E.g. tracker informs peer its reflexive candidate in PPSP CONNECT response.
  - Compared with STUN server/peer, STUN-like tracker saves messages.
  - Analogy to SIP rport

PPSP-ICE: Discover STUN/TURN/proxy peer via tracker

- STUN/TURN/proxy peer report their abilities to tracker
- NATed peer fetches STUN/TURN/proxy peer list from tracker
- NATed peer obtains candidates from STUN/TURN/proxy peer

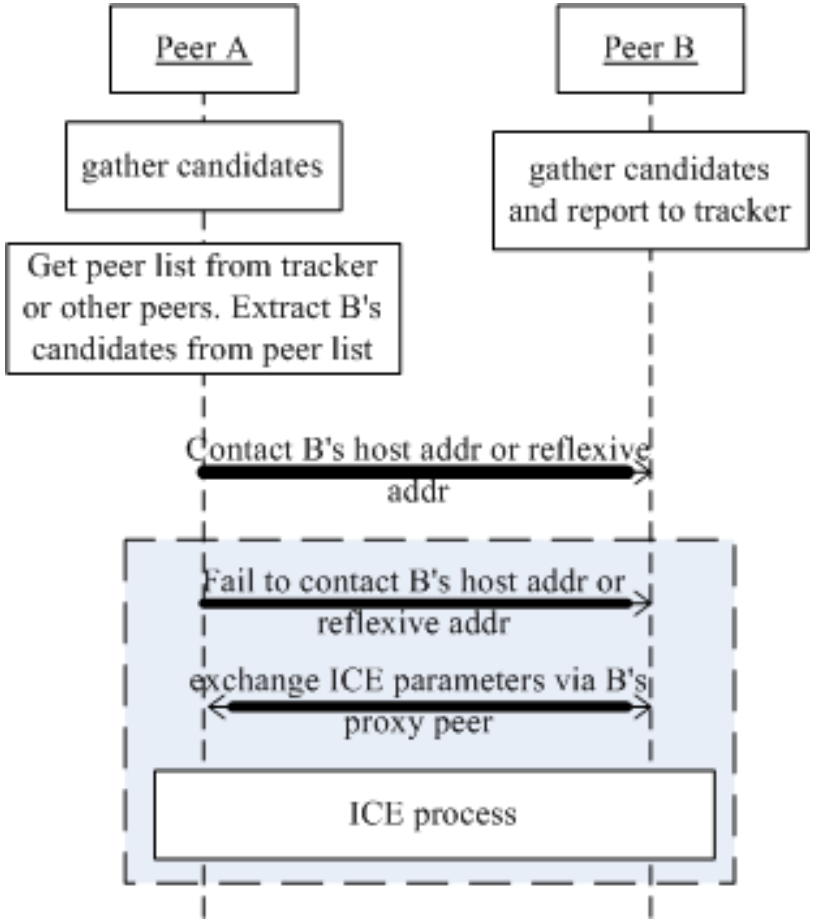


PPSP-ICE solution (signal traversal)

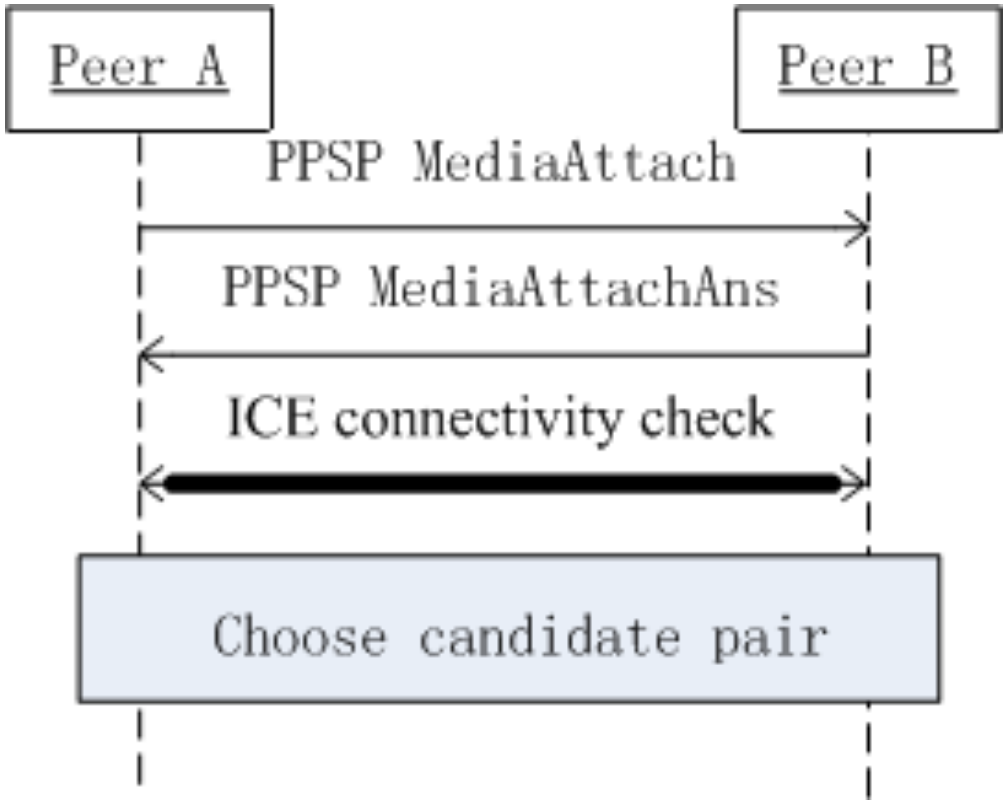
- candidates
  - Host addr, reflexive addr
  - Relay required: relayed addr, proxy addr

  Strategy to connect

  contact host addr and reflexive addr directly first
  - contact proxy addr to exchange ICE parameters
- Exchange ICE parameters with PPSP messages called Attach



PPSP-ICE solution (media traversal)



- After PPSP connection is built, media connection can be built.
- To build media connection, ICE parameters are exchanged using PPSP messages called MediaAttach

Attach & MediaAttach

<DestPeerID>***</DestPeerID>

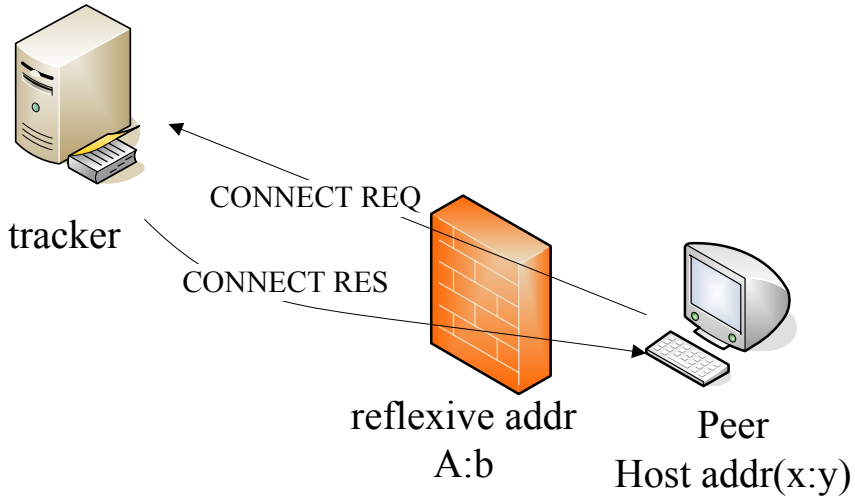
<PeerID>***</PeerID>

<SDP> ... </SDP>

- SDP contains ICE parameters.
- Mandatory for PPSP-ICE solution
- Also required if there is no NAT

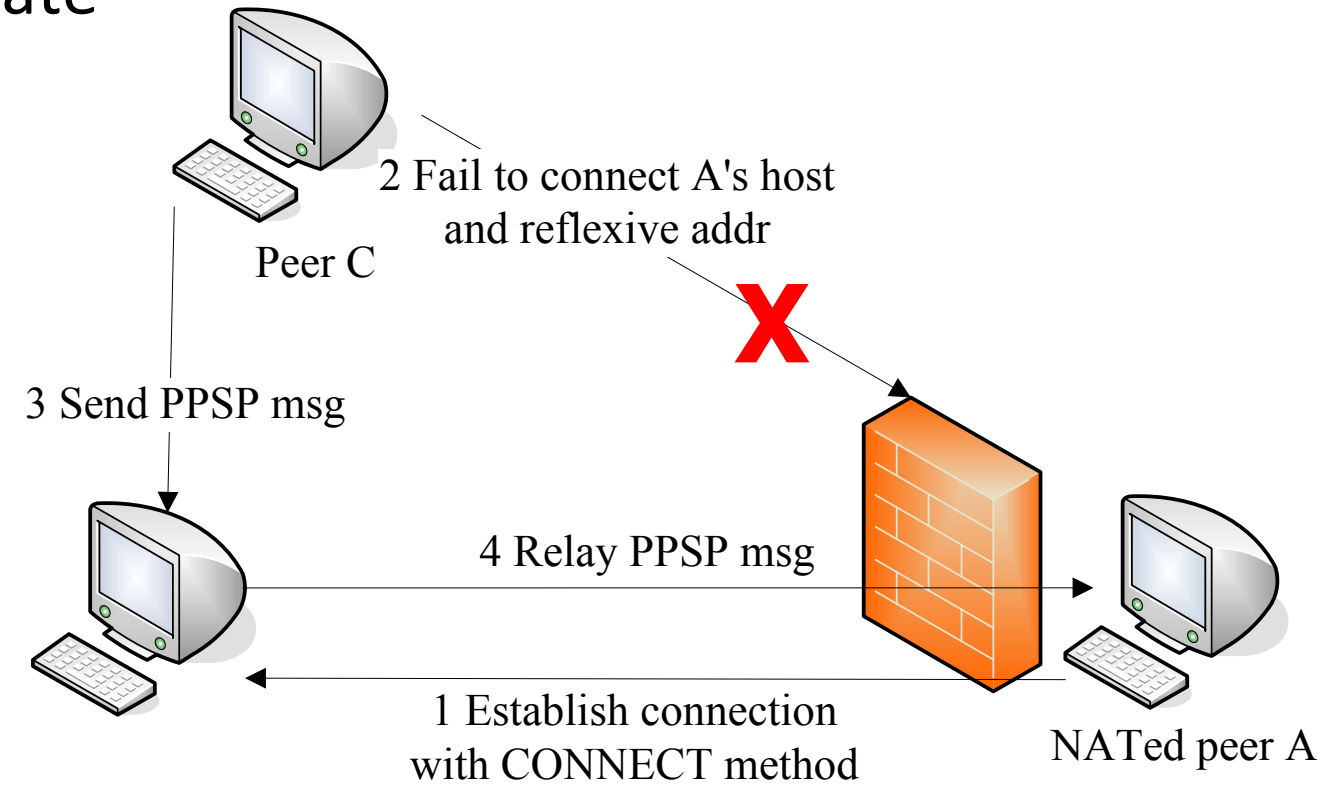
Tracker's STUN-like Function

- Optional for both PPSP-ICE and RELOAD-ICE solutions.
- Peer discovers its reflexive addr with tracker's help.
- Put <ReflexiveAddr> IP and port </ReflexiveAddr> in CONNECT response.



Proxy peer

- Mandatory for PPSP-ICE solution
- NATed peer reuses CONNECT to establish connection with proxy peer, and obtain proxy cadidate



STUN/TURN/proxy Ability Report

- Mandatory for PPSP-ICE solution, optional for RELOAD-ICE solution
- gu-ppsp-tracker-protocol already supports STUN/TURN ability report with STAT method
- Add a STAT type “proxy” to support proxy ability report

Find STUN/TURN/proxy Peer List

- Mandatory for PPSP-ICE solution, optional for RELOAD-ICE solution
- Fetch STUN/TURN/proxy peer list from tracker with modified FIND method
- Remove <SwarmID> and <ChunkID> from FIND request

<PeerID>***</PeerID>

<Peernum>***</Peernum>

<Stats>

<Stat property="STUN">true</Stat>

... more stats ...

</Stats>

Report Candidates to Tracker

- Peer puts <PeerAddresses> in JOIN/JOIN_CHUNK request

```
<PeerAddresses>
  <PeerAddress ip="***" port="***" priority="***" type="host"/>
  <PeerAddress ip="***" port="***" priority="***" type="reflexive"/>
  <PeerAddress ip="***" port="***" priority="***" type="proxy"/>
</PeerAddresses>
```

- Mandatory for PPSP-ICE solution
- Also required to support multi-homed peer

About NAT traversal extension

- Leave the choice of NAT traversal to implementation
  - In some scenarios, NAT traversal is not necessary.
    - no NAT
    - QoE is satisfied without NAT traversal.
  - There are multiple NAT traversal solution/methods requiring different protocol extension.
- Base protocol is mandatory to implement, while NAT traversal extension is not?
  - Base protocol should work without NAT traversal, and allow NAT traversal extension

Next Step

- Get more comments
- Call on participation in the draft
- Refine NAT-traversal extensions

Thank you!

Questions?

Thank you!

Questions?

Thank you!

Questions?

NAT traversal necessity

- Scenarios need NAT traversal: The ratio of NATed peer is high in the swarm. Without NAT traversal, some peers can't download or take long time to download needed chunks.
- Scenarios don't need NAT traversal:
  There is no NAT or the QoE is satisfied without NAT traversal solution.
- NAT traversal is necessary at least in some P2P streaming systems (e.g. UUSee).

Implementation consideration

- The decision of supporting NAT traversal or not should be left to implementation.
- The choices of NAT traversal solution/method, NAT traversal service node and NAT traversal service discovery method should be left to implementation too.
- Implementation considerations: the ratio of NATed peer, the ratio of each NAT type, implementation overhead, etc.