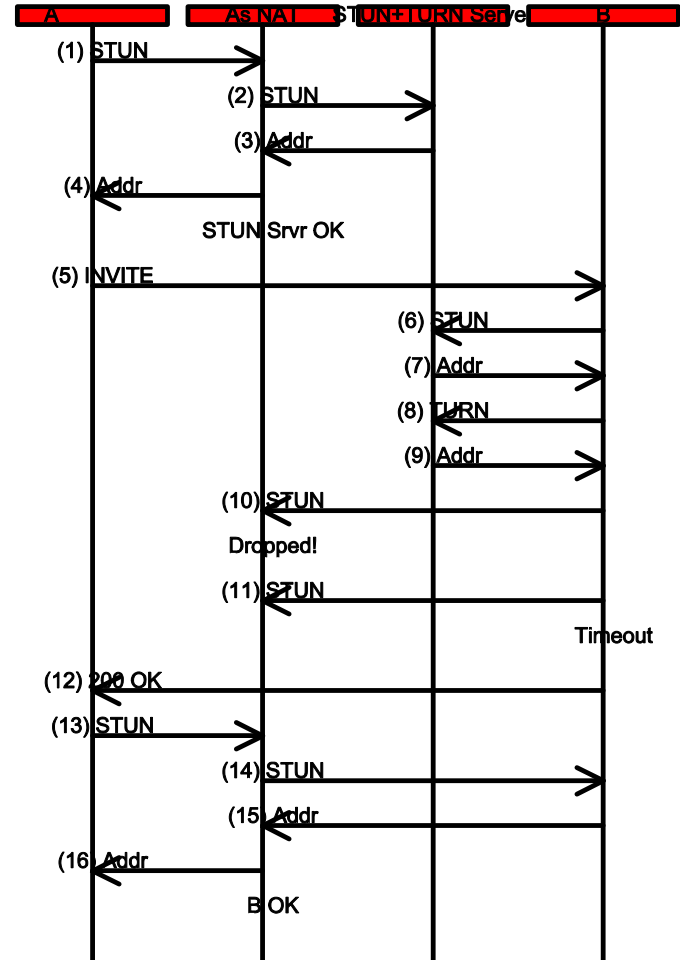


ICE

Jonathan Rosenberg
dynamicsoft

Issue 1: Port Restricted Flow

- This case does not work well with ICE right now
- Race condition
 - Works if message 13 occurs before B timeout
 - Won't happen if B delays 200 OK
 - Should happen if answer is sent immediately
 - Have 9.5s
 - Long enough?
 - ICMP case

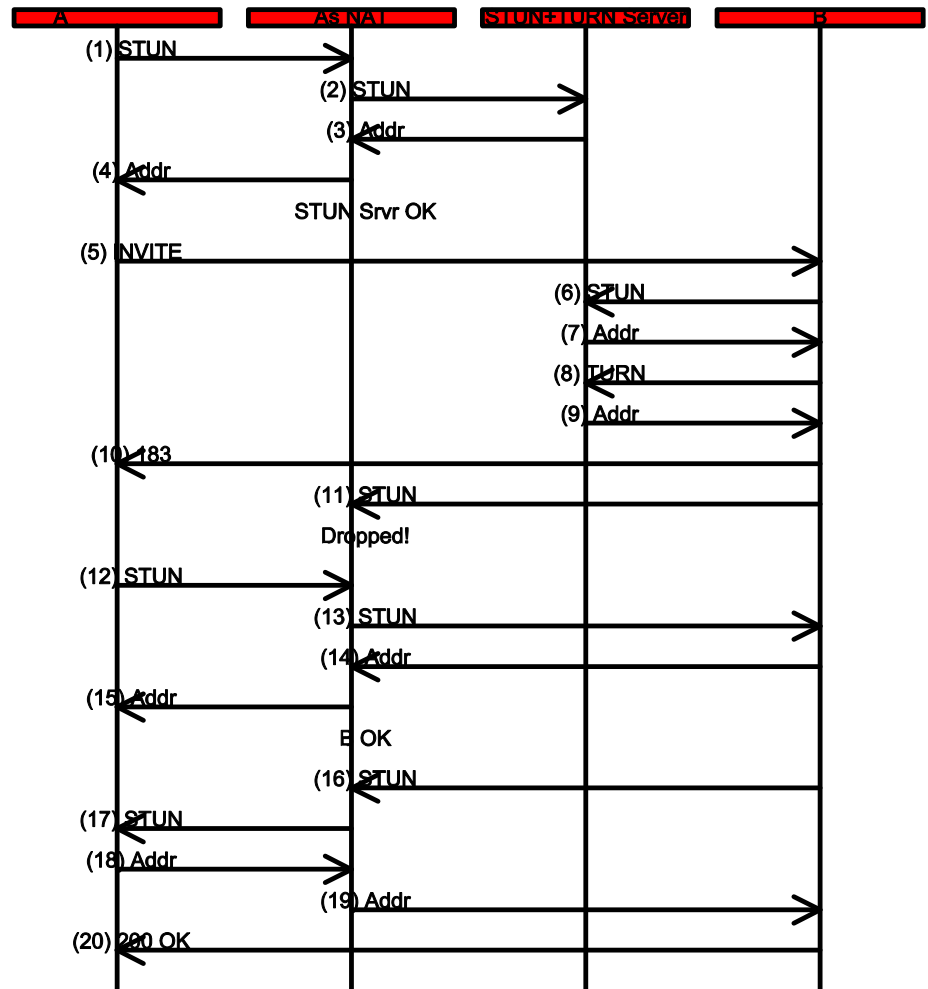


Solutions

- Solution 1
 - Mandate an immediate answer
 - Recommend longer STUN timeout for STUN-allocated addresses
 - In case of INV/200/ACK, advise to continue till ACK
- Solution 2
 - Mandate a second ICE cycle to re-check STUN-allocated addresses
 - Eliminates race condition in all cases
 - Costs more signaling
 - NOT call setup delay

Solution 1 Flow

- Message 12 now beats message 16
- 200 OK can be sent at any time



Proposal: Solution 1

Issue 2: Prioritization

- Current text aims at minimizing relay count and preferring IPv6
- There are other criteria that might work
 - Maximize hops over secure networks
 - Minimize actual path latency (Noop interaction)
 - Minimize router hops
 - Etc.
- Issue: do we need to pick one, if so, which?

Do we need to pick?

- Short answer: no
 - ICE functionality does not depend on policy
 - ICE works so long as there exists at least one address that always works (i.e TURN)
 - After that, its an optimization
 - Its ok to optimize differently
- However
 - There are many parties at the table
 - End user
 - End user's access provider
 - ISP
 - Application service provider
 - Called party
 - Each may have different policies and each is affected by the choice

Session Policy Application

- SIP/PING is pursuing work on session policy
- Allows providers on the call signaling path to assert policies for media handling
 - Session independent
 - Session dependent
- This is another session policy
- Can leverage that work if providers want to distribute policy

Proposal

- Clarify that there can be many axes of optimizations
- Document existing algorithm
- Allow for other specs to define (informationally) other algorithms
- Discuss issues that arise
- Reference session policy as a non-normative approach for finding out other policies

Issue 3: Interaction with NOOP

- Draft-wing-avt-rtp-noop defines a NOOP RTP packet
 - Sent to RTP port
 - Requests an immediate RTCP response
 - Checks for “connectivity” and QoS between endpoints
- This is very similar to the p2p STUN used by ICE
 - Sent to RTP port
 - Generates an immediate STUN response
 - Checks for connectivity, not QoS

Differences

- NOOP uses RTP, not a separate protocol (STUN)
 - Less ugly for RTP – avoids muxing a second protocol onto RTP port
 - STUN works for other media transports too
- NOOP uses RTCP for response
 - Won't work properly through many NAT
 - RTP will be received, but RTCP may be dropped
 - Response ideally sent back to source of request
- STUN also provides address allocation
 - Allows p2p media when there is a nat between A and B
- STUN provides username/pass for validation
 - NOOP would require SRTP

Options

- Use only NOOP
 - Need to incorporate some STUN features
 - Address in response
 - Disambiguation field
 - Would ideally send response in return RTP stream to avoid NAT problem
 - QoS still through RTCP
- Use only STUN
 - Do we still want COT for QoS only?
- In any case, need to clarify relationship

Todos and Open Issues TBD

- Todo
 - Change SDP param, don't use "alt"
 - Align with latest anat
- Unsolved open issues
 - Avoiding sequential tests for STUN tests sent using TURN SEND
 - Symmetrical vs. Assymetric testing
 - Optimization for avoiding extra ICE cycles doesn't always work