



Overview of ICE (Interactive Connectivity Establishment)

Dan Wing, Technical Leader
dwing@cisco.com

Agenda – Interactive Connectivity Establishment

- **Introduction to ICE**
- **Simple Call Flow**
- **Complex Call Flow**
 - TURN server**

Introduction to ICE



NAT Traversal Problem Statement

- **Many solutions for NAT traversal for SIP**
 - STUN/TURN**
 - Back-to-Back UA with media (SBC)**
 - Application Layer Gateway (ALG)**
 - UPnP, Bonjour**
- **None are ideal in all scenarios**
 - STUN by itself isn't sufficient**
 - SBCs usually relay all media (expensive), only help relaying SIP, and can interfere with new SIP services**
 - ALGs are buggy and only help specific protocols**
 - Often can't upgrade the NAT**

NAT Traversal Problem Statement

- **Need a SINGLE algorithm that can be *placed into client* which:**
 - Works in all scenarios**
 - Is a good solution in all scenarios**
 - Doesn't require configuration or knowledge of network topology**
 - Doesn't change or control the NAT**

ICE in the Client

- **Advantages of placing into client**

 - VPNs**

 - v4/v6 interworking**

 - SIP and non-SIP applications**

 - Media always takes shortest path**

- **Disadvantages**

 - One client has to implement ICE**

 - To optimize media flows, both clients need ICE**

Relationship: ICE, STUN, TURN

- **STUN (RFC3489) is a request/response protocol**

The response contains IP address and UDP port of request

- **ICE – Procedures for Optimizing Media Flows**

Uses STUN messages for connectivity checks

Defines SDP syntax to indicate ‘candidate addresses’

- **TURN – Media Relay Protocol and Server**

Only used when:

both endpoints are behind ‘symmetric’ NATs, or
one endpoint doesn’t implement ICE

Symmetric NATs are rare

draft-jennings-midcom-stun-results-02.txt

Standardization and Deployment of ICE

- **Standardization**

 - ICE – IETF MMUSIC Working Group**

 - STUN (RFC3489bis), TURN, TURN-IPV6 – IETF BEHAVE Working Group**

- **Deployment**

 - Cisco/Microsoft Press Release**

 - Google (Jabber/XMPP)**

 - Yahoo**

 - Counterpath (softphones)**

ICE Properties

- **If path exists, it will find it**
- **Will use path with fewest media relays**
 - Cheapest for the service provider (and customer!)**
- **Absolutely no knowledge of:**
 - network topology**
 - NAT types**
- **Can guarantee phone only rings if media path is working**
- **Compatible with SIP-Identity, S/MIME**
- **Multi-protocol (SIP, XMPP/Jabber)**

ICE Key Concepts

- **Many addresses to receive media**

Local interfaces

VPN Interfaces

IP Addresses learned from STUN

IP Addresses learned from TURN

IPv6 address(es)

- **Which One(s) Will Work When Talking to a Specific Peer?**

NO WAY TO KNOW AHEAD OF TIME

- **ICE Says: Try Each Candidate**

Each endpoint exchanges candidate and does a “connectivity check”

Connectivity checks are STUN messages sent on the RTP and RTCP ports

- **Then, chose the highest priority address that works**

Simple ICE Call Flow

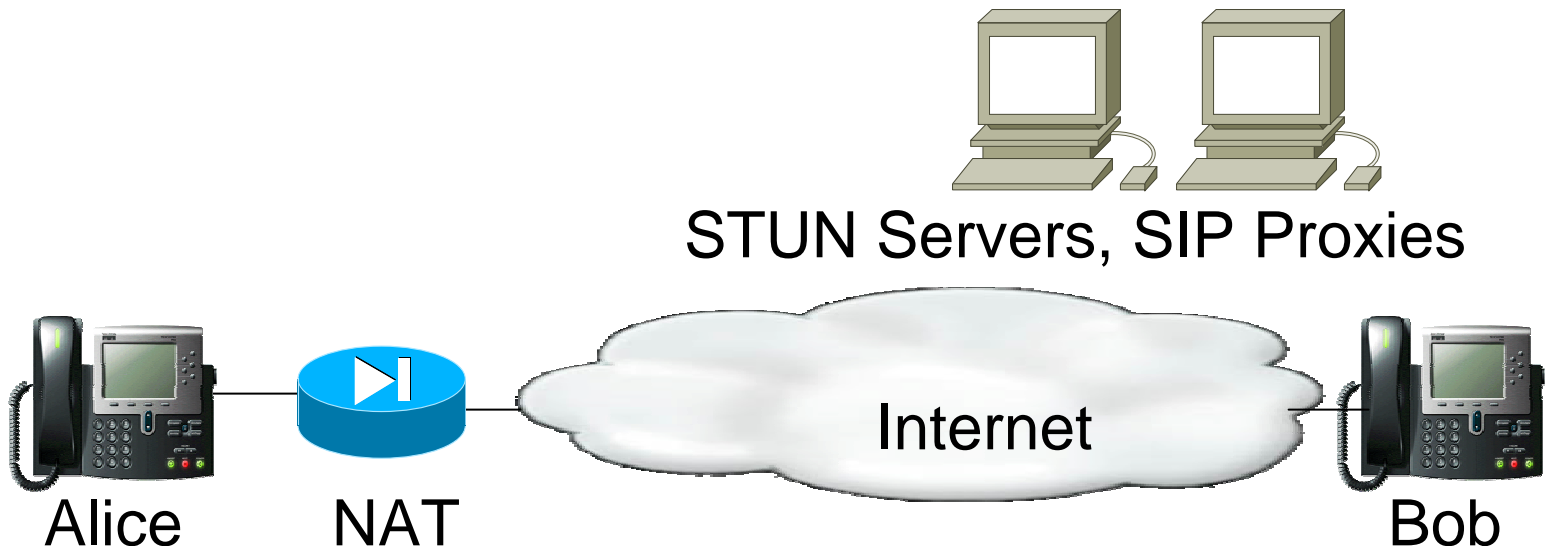


Simple ICE Call Flow

- **Only Alice is behind a NAT**
- **ICE also works if Bob was behind a NAT**

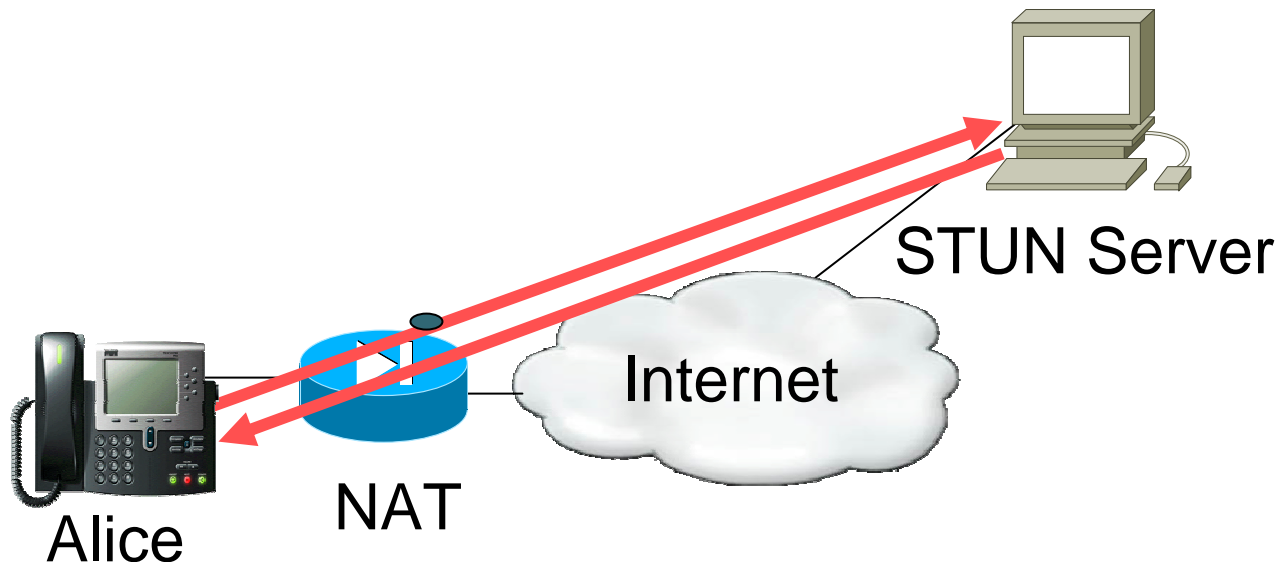
Only difference: Bob would also talk to a STUN server to gather his IP address/UDP port

We'll see this in Complex flow



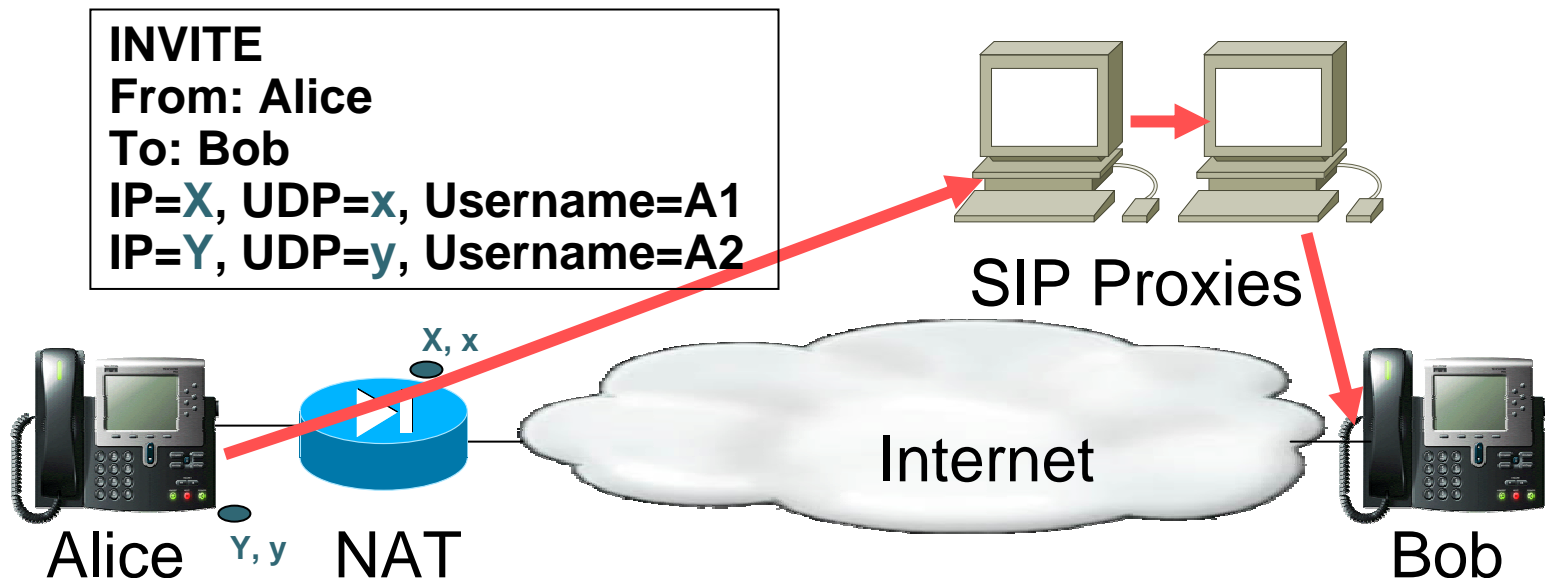
Alice Gathers her IP address: Step 1

- Alice lifts handset offhook
- Alice's phone learns her public IP address and UDP port using STUN (RFC3489)



Alice sends Invite: Step 2

- Alice sends INVITE to her SIP proxy
- INVITE contains all of her candidate addresses
X,x and Y,y
- The SIP proxies sent the INVITE to Bob

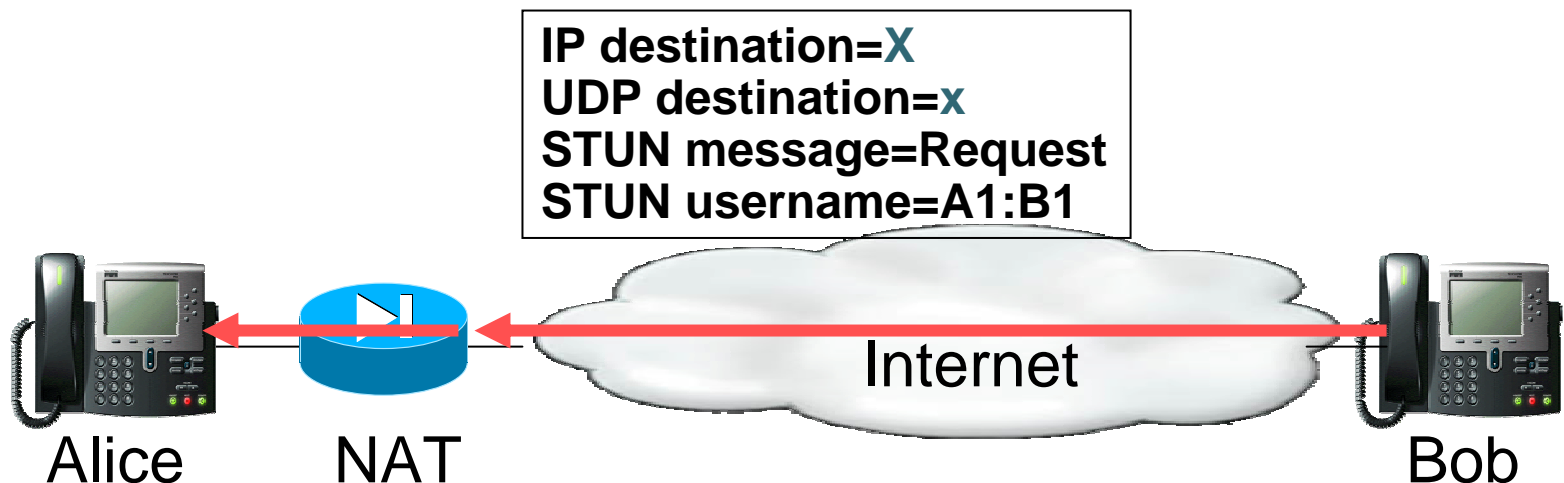


Bob Sends STUN Request: Step 3

- **Bob sends STUN connectivity check message to each of Alice's candidate addresses**

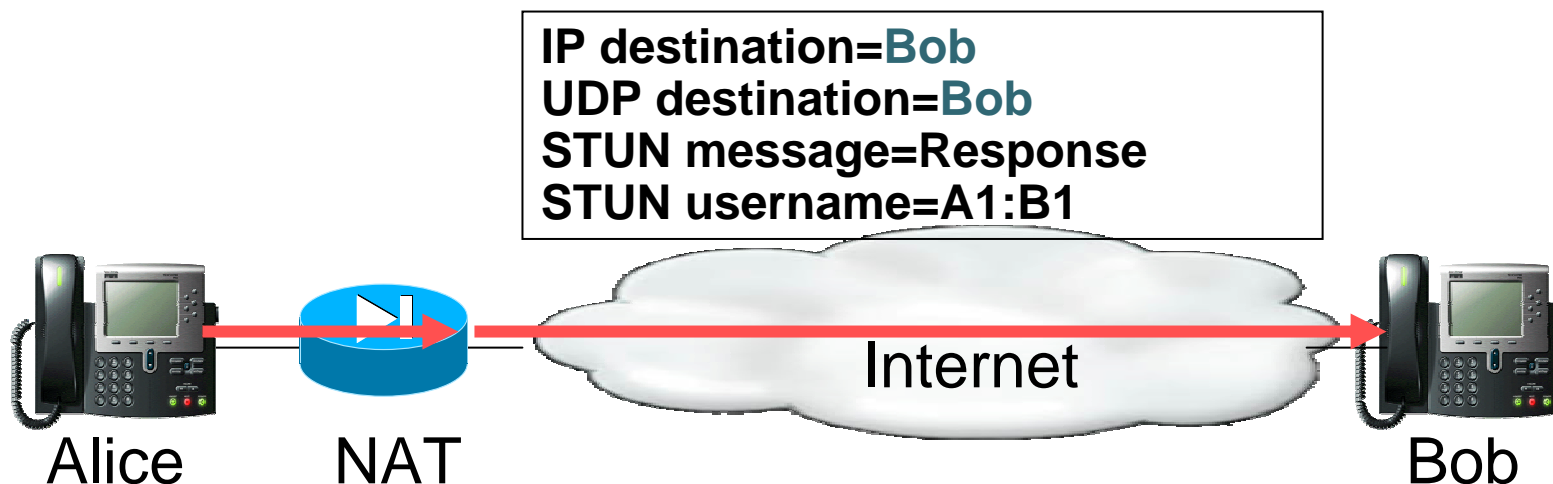
STUN Binding Request

- **Alice's NAT already had a UDP binding, so it sends the connectivity check message to Alice**



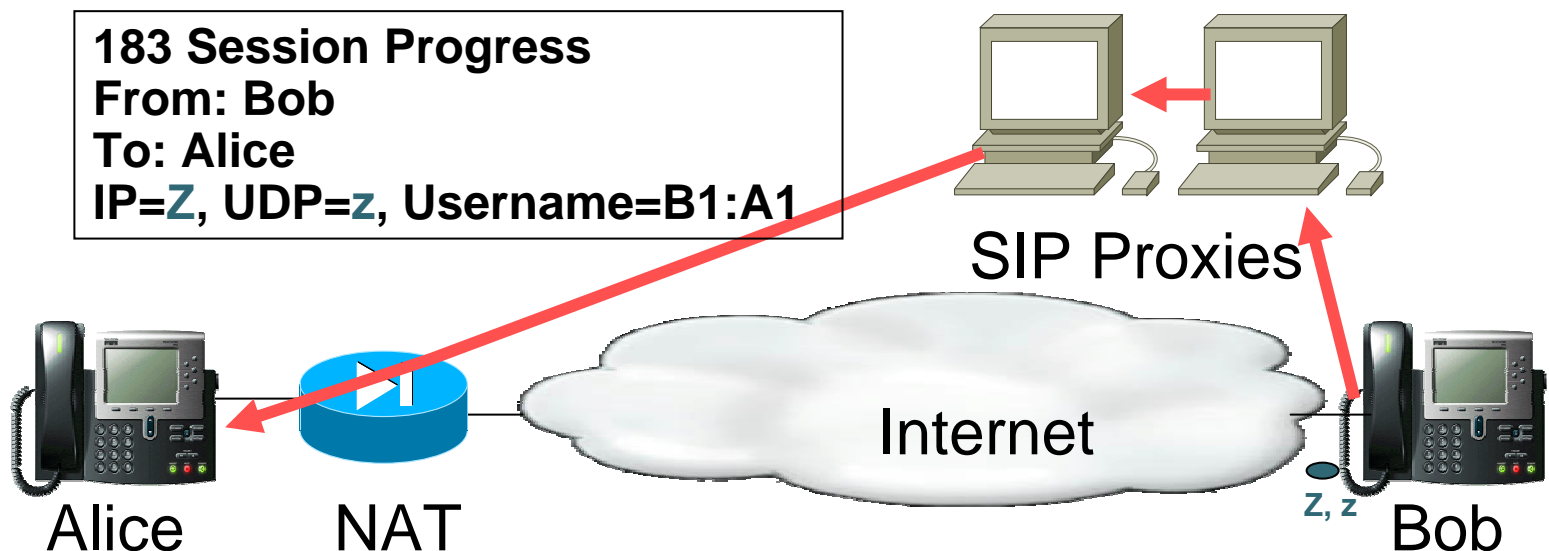
Alice Sends STUN Response: Step 4

- Alice sends STUN Binding Response
 - MESSAGE-INTEGRITY will require Alice to wait for Bob's SDP
- Bob's connectivity check succeeds



Bob Sends Invite: Step 5

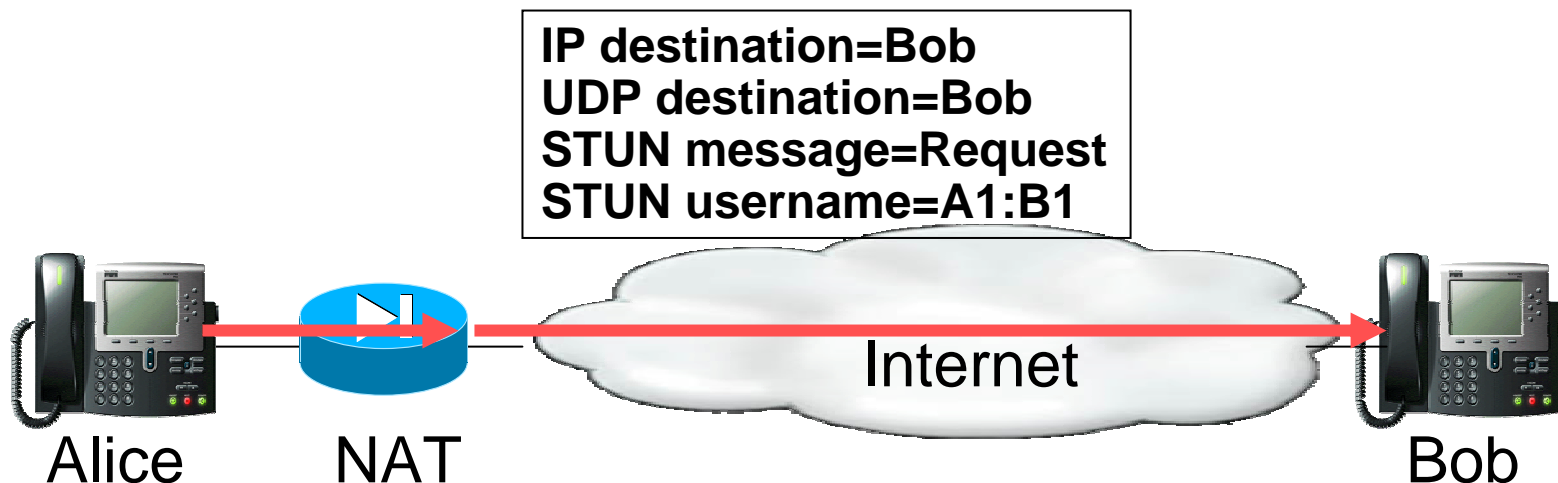
- Bob sends his SDP, containing his IP address(es)
- Alice receives SDP from Bob
- Alice now knows Bob's IP addresses



Alice sends STUN Request: Step 6

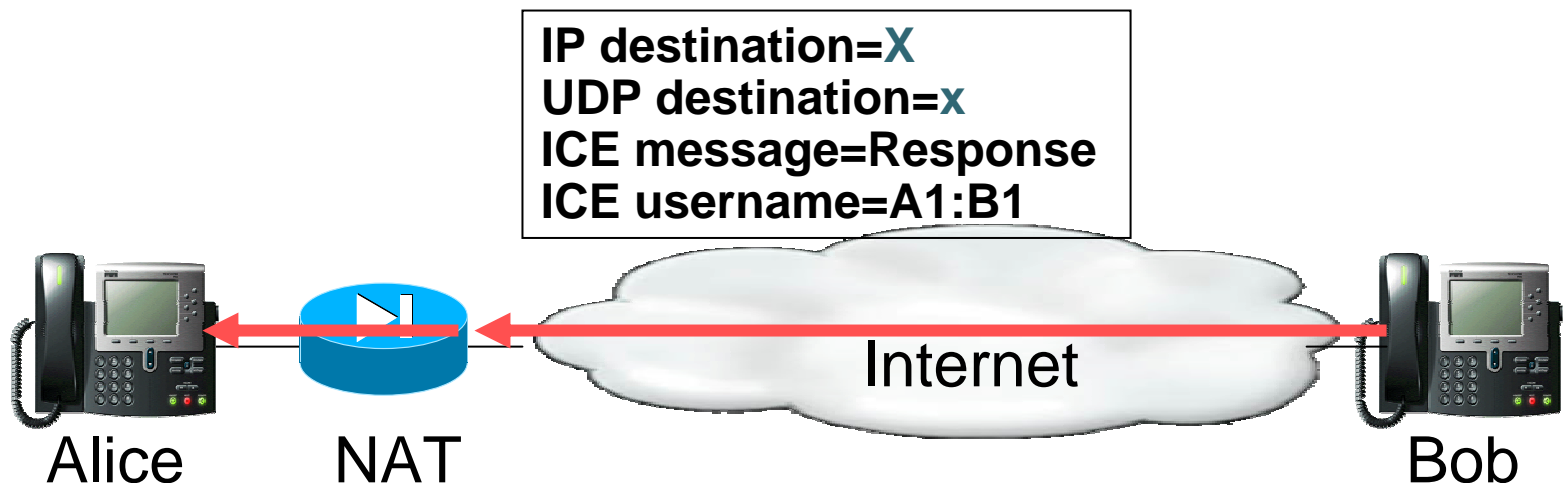
- Alice sends STUN connectivity check message to each of Bob's candidate addresses

STUN Binding Request



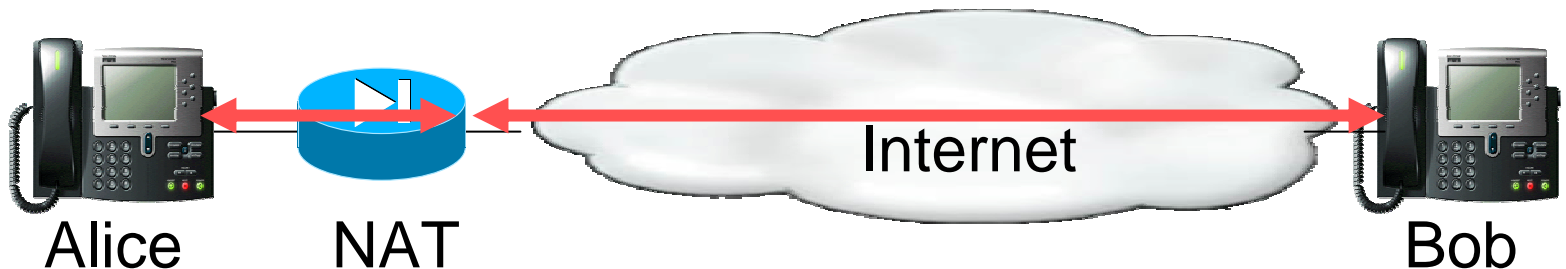
Bob sends STUN Response: Step 7

- Bob sends Binding Response
- Alice's connectivity check succeeds



Bi-Directional Media: Step 8

- **Bi-Directional media is confirmed and established**

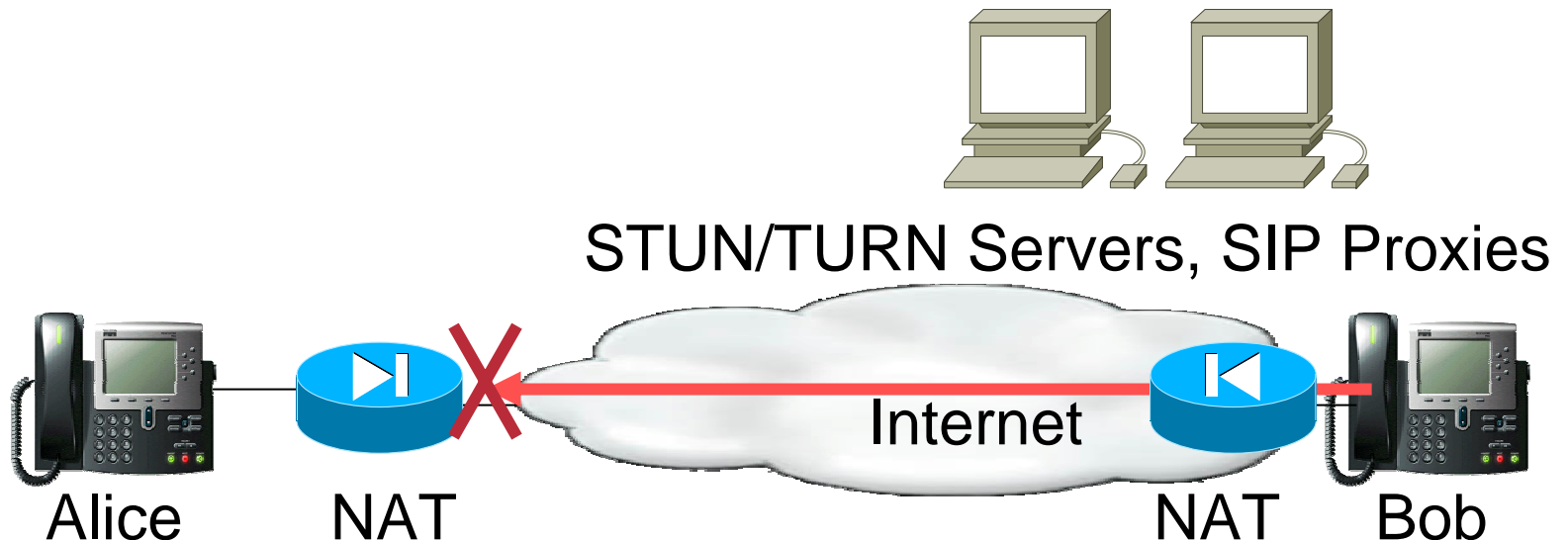


Complex ICE Call Flow



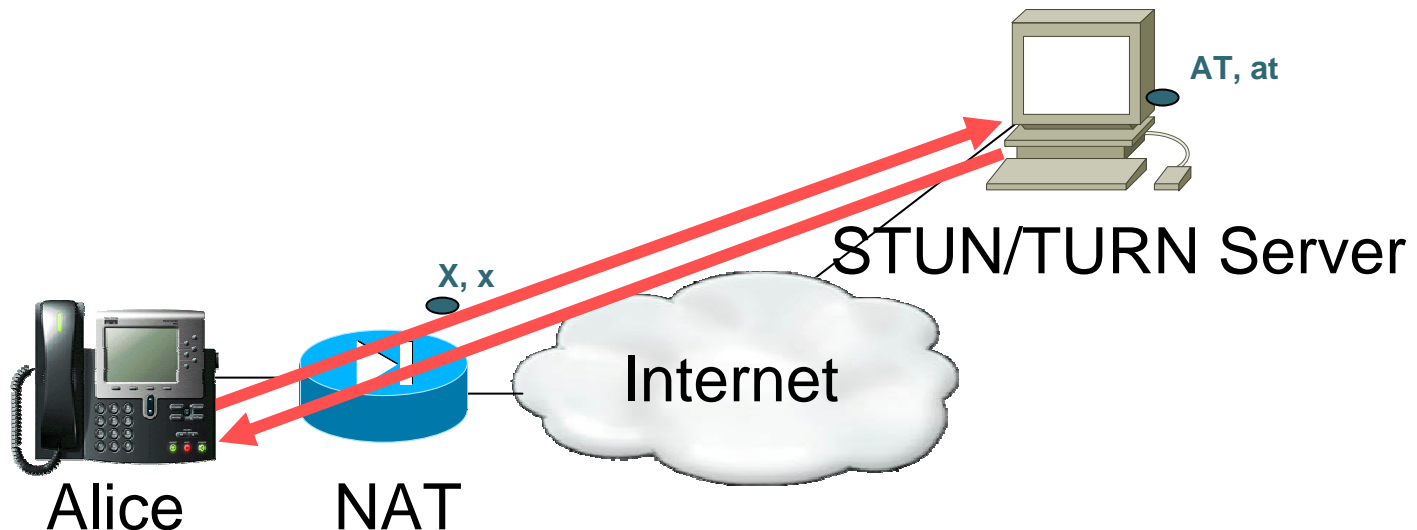
Complex ICE Call Flow

- For simplicity, only Alice is behind a NAT
- Both Alice's and Bob's NATs are "Symmetric"
Requires a media relay
- This flow only occurs when both NATs are 'Symmetric'



Alice Gathers her IP address: Step 1

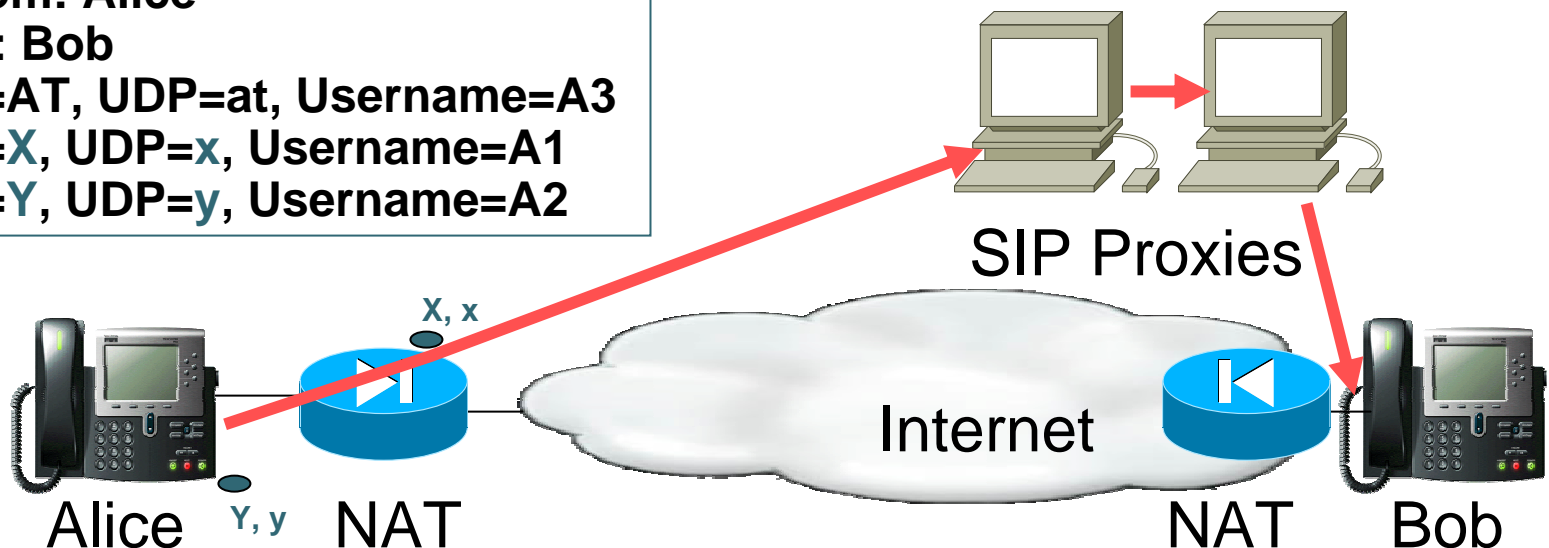
- Alice lifts handset offhook
- Alice's phone learns her public IP address and UDP port using STUN (RFC3489)
- And learns TURN address



Alice sends Invite: Step 2

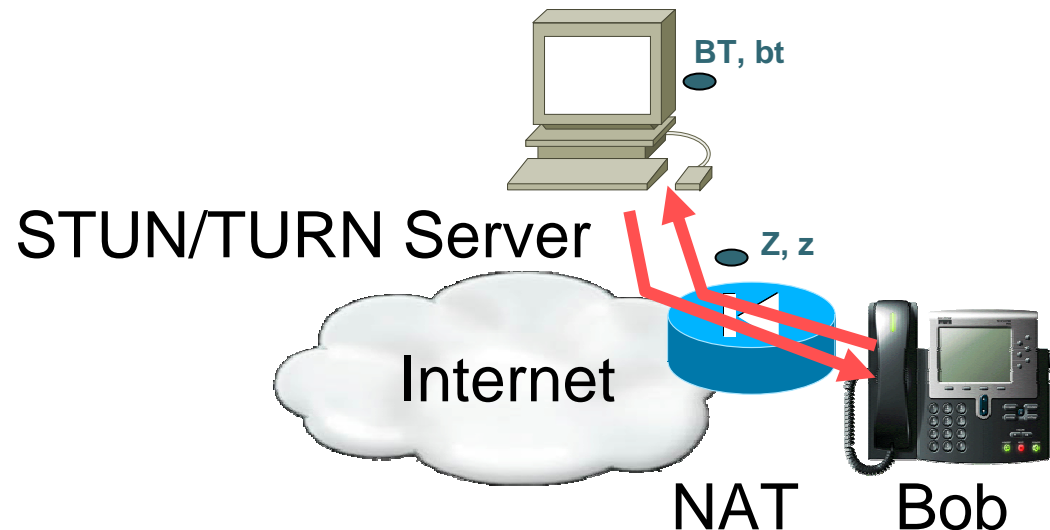
- Alice sends INVITE to her SIP proxy
- INVITE contains all of her candidate addresses
X,x, Y,y, and AT,at

```
INVITE
From: Alice
To: Bob
IP=AT, UDP=at, Username=A3
IP=X, UDP=x, Username=A1
IP=Y, UDP=y, Username=A2
```



Bob Gathers his IP addresses: Step 3

- Bob's phone learns his public IP address and UDP port using STUN (RFC3489)
- Bob learns his TURN address (BT, bt)
- Bob tells his TURN server to relay with Alice's TURN server



Bob Sends SDP: Step 4

- Bob sends his SDP, containing his IP addresses
- Alice receives Bob's SDP
- Alice now knows Bob's IP addresses

183 Session Progress

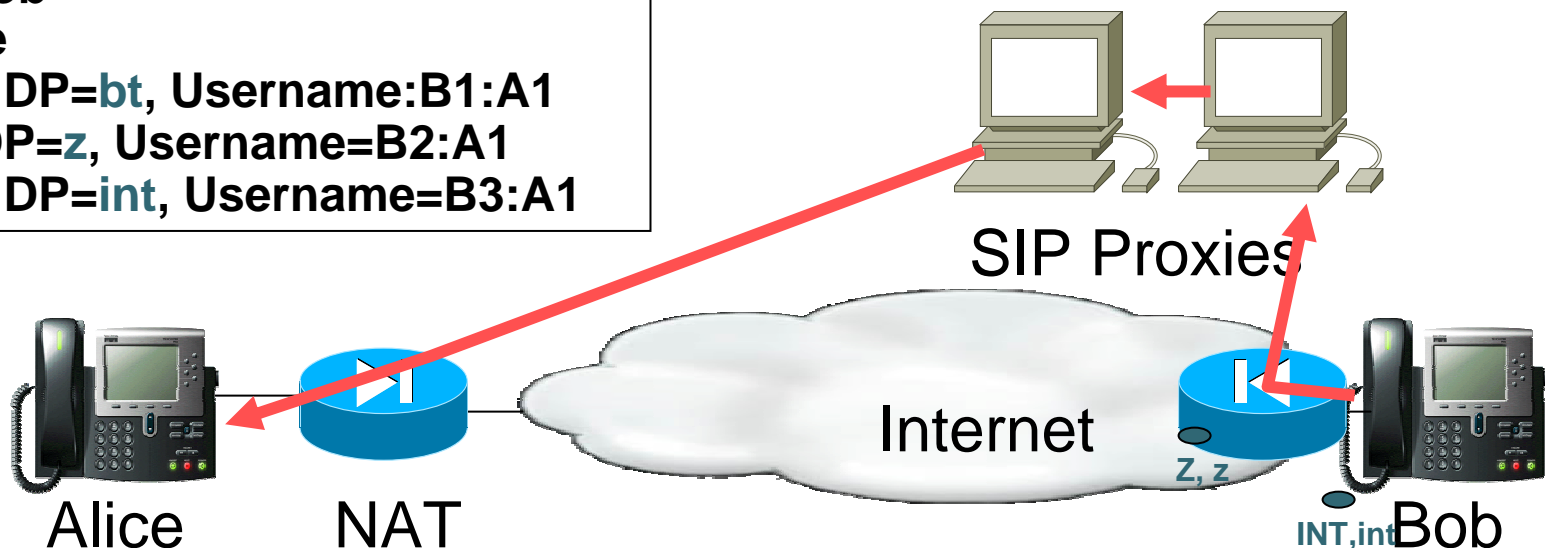
From: Bob

To: Alice

IP=BT, UDP=bt, Username=B1:A1

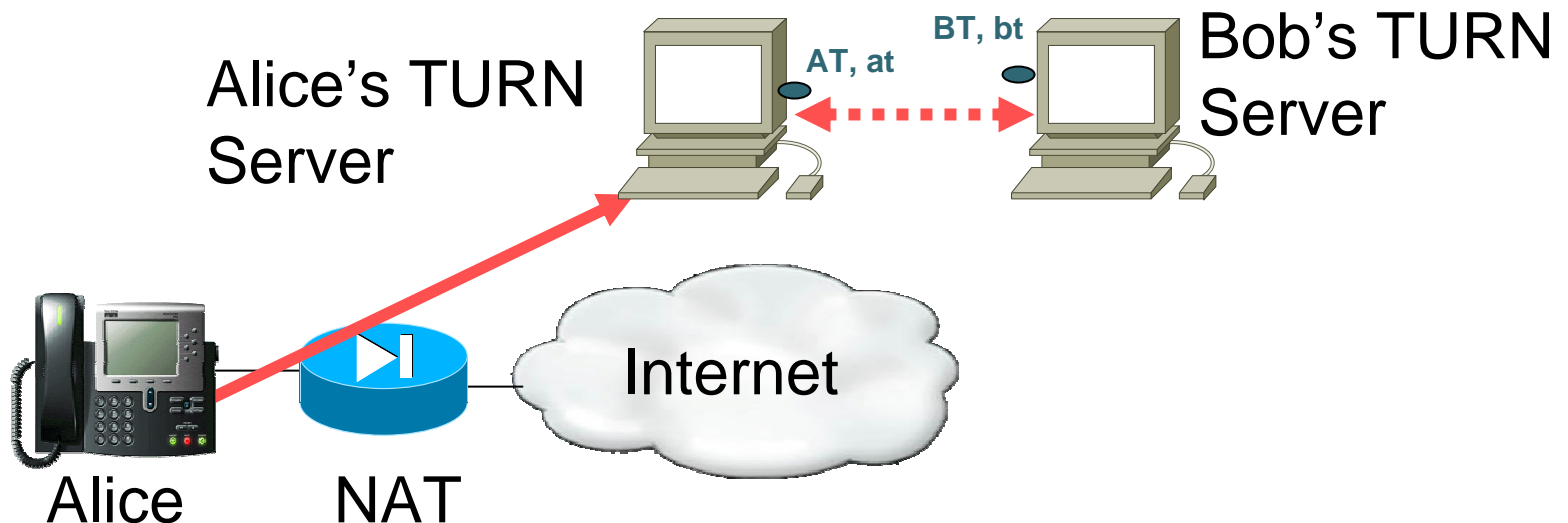
IP=Z, UDP=z, Username=B2:A1

IP=INT,UDP=int, Username=B3:A1



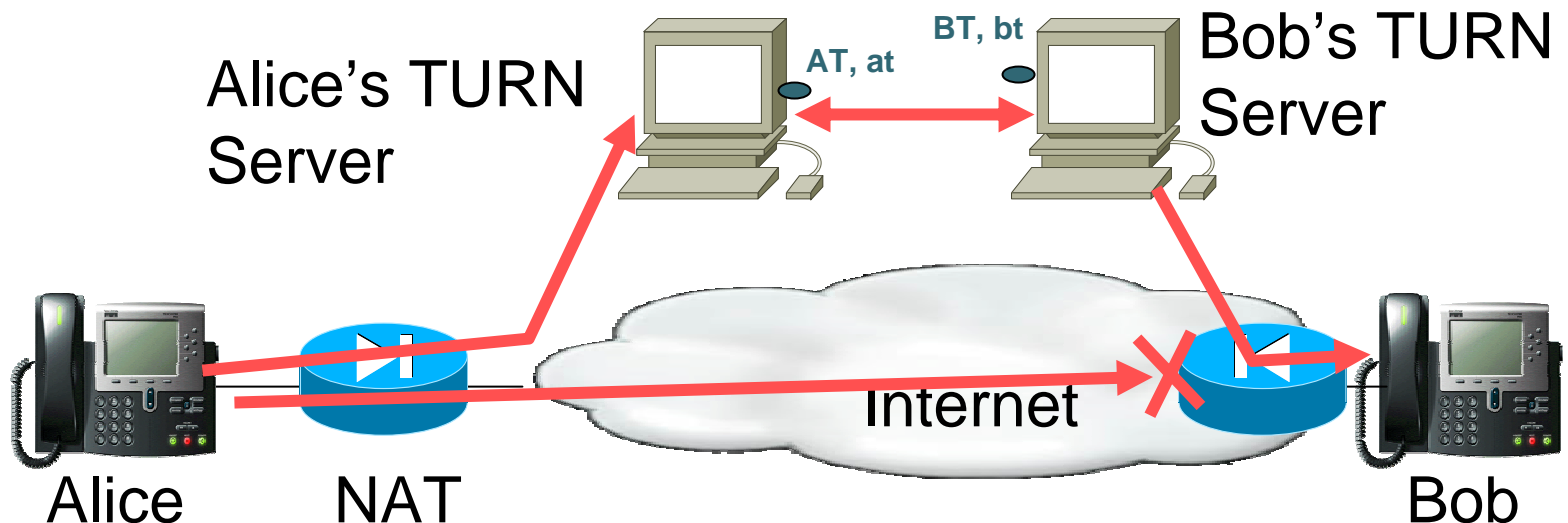
Alice Tells her TURN server: Step 5

- Alice tells her TURN server to relay packets to and from Bob's address



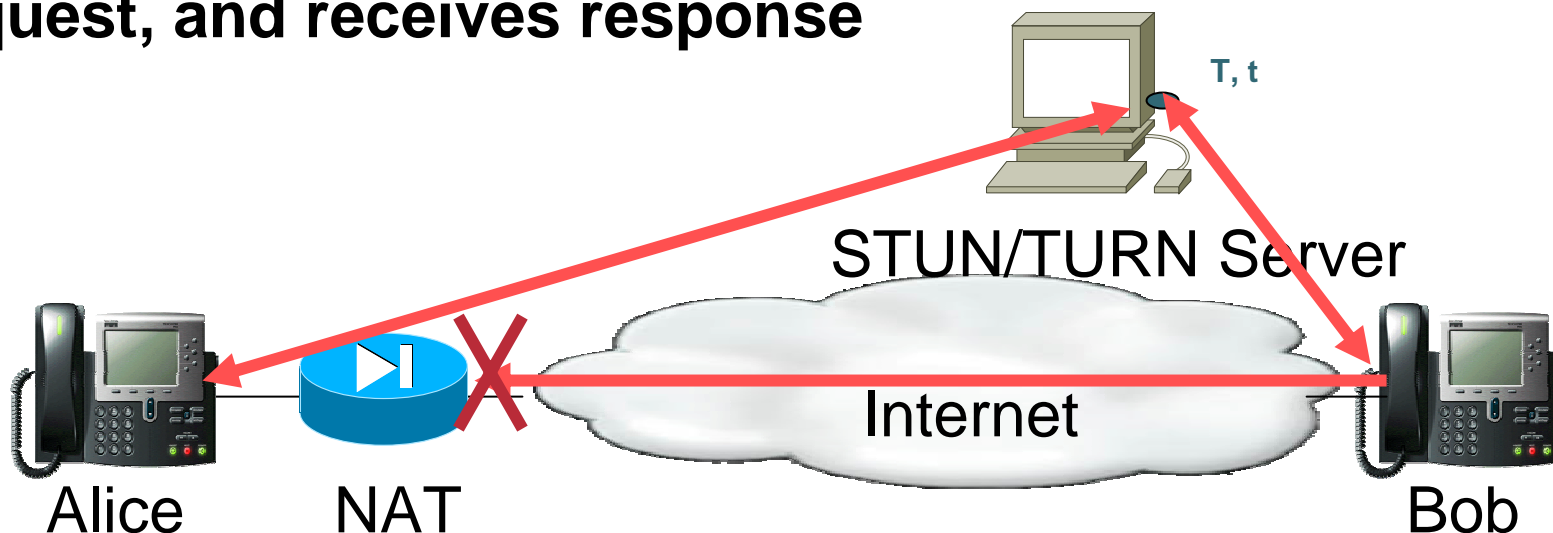
Alice sends STUN Request: Step 6

- Alice sends STUN connectivity check message to each of Bob's candidate addresses
- Checks sent directly to Bob will fail (symmetric NAT)
- Check sent through her TURN server will succeed



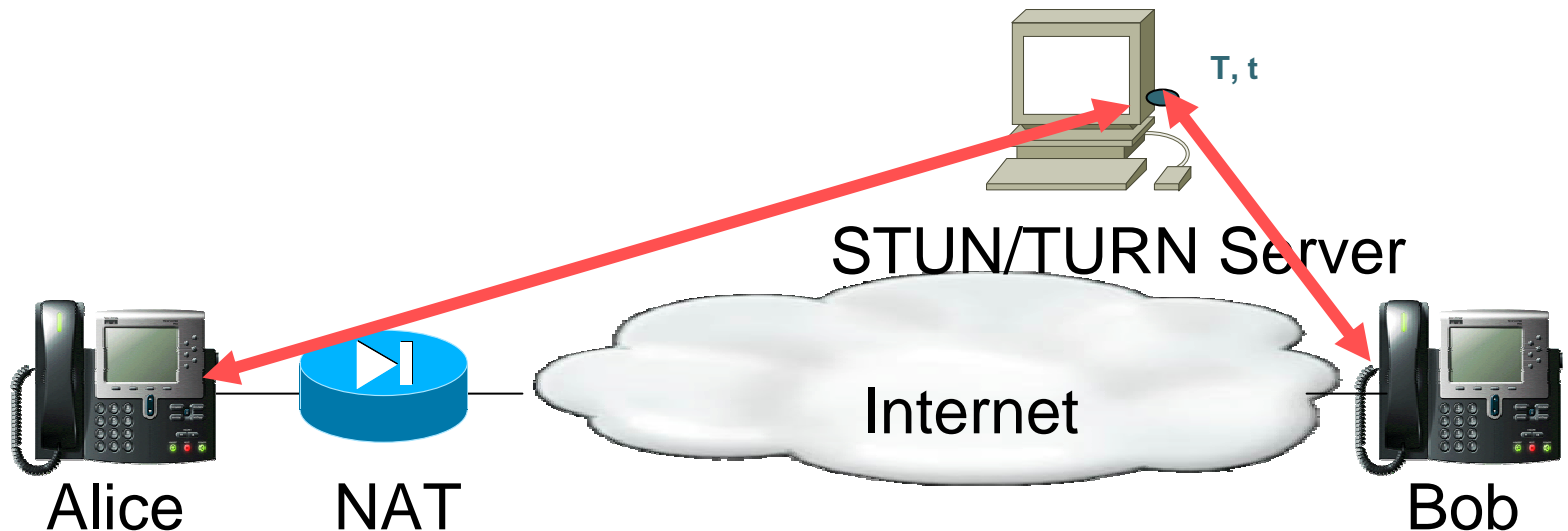
Bob Replies and sends his own STUN Request: Step 7

- **Meanwhile, Bob has been sending STUN connectivity checks to each of Alice's addresses**
 - They have all been failing
- **Bob receives Alice's STUN Request via TURN server**
- **He sends Response, and sends his own STUN Request, and receives response**



Bi-Directional Media: Step 8

- **Bi-directional media is confirmed and established through the media relay (TURN server)**



Summary



*** Note: more than one image can be used with these segue and Q and A slides, but should never extend past the half grid of the page**

Complaints About ICE

- **Slow Call Setup**

 - STUN queries are lighter weight than DNS queries**

 - Calling party's delays can be hidden (do STUN query at offhook)**

 - Called party delay can be hidden (don't alert until STUN query has succeeded)**

- **ICE code in endpoints**

Strengths of ICE

- **Always finds shortest, most optimal path**
- **Places control into endpoints**
SIP-Identity, S/MIME
- **Applicable to many protocols**
SIP, XMPP/Jabber, peer-to-peer gaming

CISCO SYSTEMS

