An Analysis of the Skype Peer-to-Peer Internet Telephony Protocol

Salman Baset and Henning Schuzrinne
INFOCOMM 2006

Presenter - Bob Kinicki

Outline

- Skype Overview
- Skype Components
- Review of NATs
- Experimental Set Up
- Skype Functionality
  - Login, Login Server, User Search, Call Establishment, Conferencing
- Super Node Facts
- Conclusions

Skype Overview

- Developed by Kazaa as an overlay P2P (peer-to-peer) network.
- Provides a VoIP client that supports voice calls, instant messaging, audio conferencing and buddy lists. {Currently supports video!}
- Uses TCP for signaling and TCP and UDP for transporting media traffic.
- Uses 256-bit AES encryption.
- Employs wideband codecs (iLBC, ISAC and iPCM) that allow frequencies between 50-8000 Hz.

Skype Network

- Ordinary Host
  - Skype Client (SC)
- Super Node (SN)
  - Skype Client
  - Must have public IP address
  - Has sufficient capacity, CPU and memory
- Skype Login Server

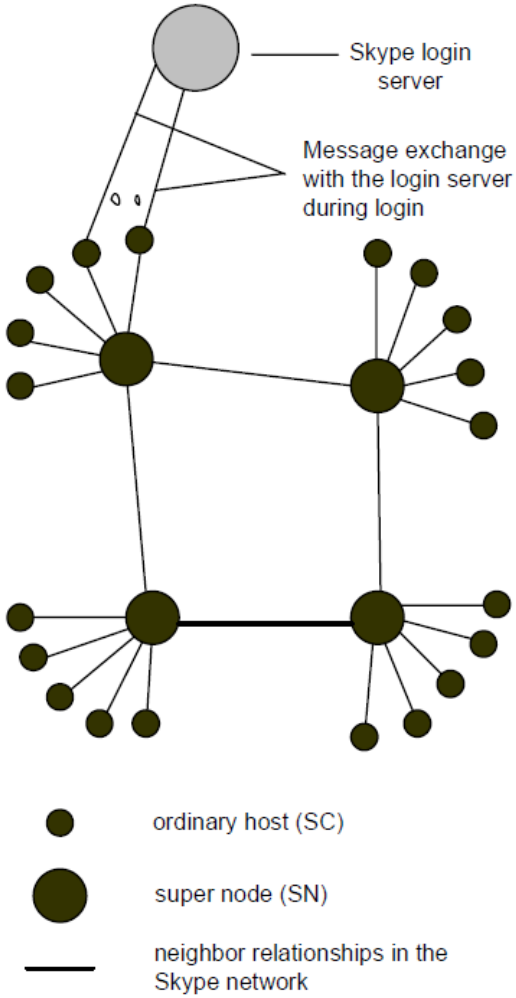


Figure 1. Skype Network. There are three main entities: supernodes, ordinary nodes, and the login server.

Skype Overview

- Ordinary hosts (SC) must connect through a super node (SN) and authenticate itself via the Skype login server.
- Skype handles ordinary hosts behind a port-restricted NAT (Network Address Translation) and/or a UDP-restricted firewall.
- Authors infer from experimentation that variant of STUN (Session Traversal Utilities for NAT [RFC5389) protocol is used by non-centralized Skype servers to determine the type of NAT and firewall the SC is behind.

Skype Components

- SC randomly selects UDP listening port at install.
- SC also opens ports 80 and 443 to listen for incoming HTTP and HTTP-over-TLS* requests, respectively.

* Transport Layer Security supersedes and is an extension of SSL.

SC Ports

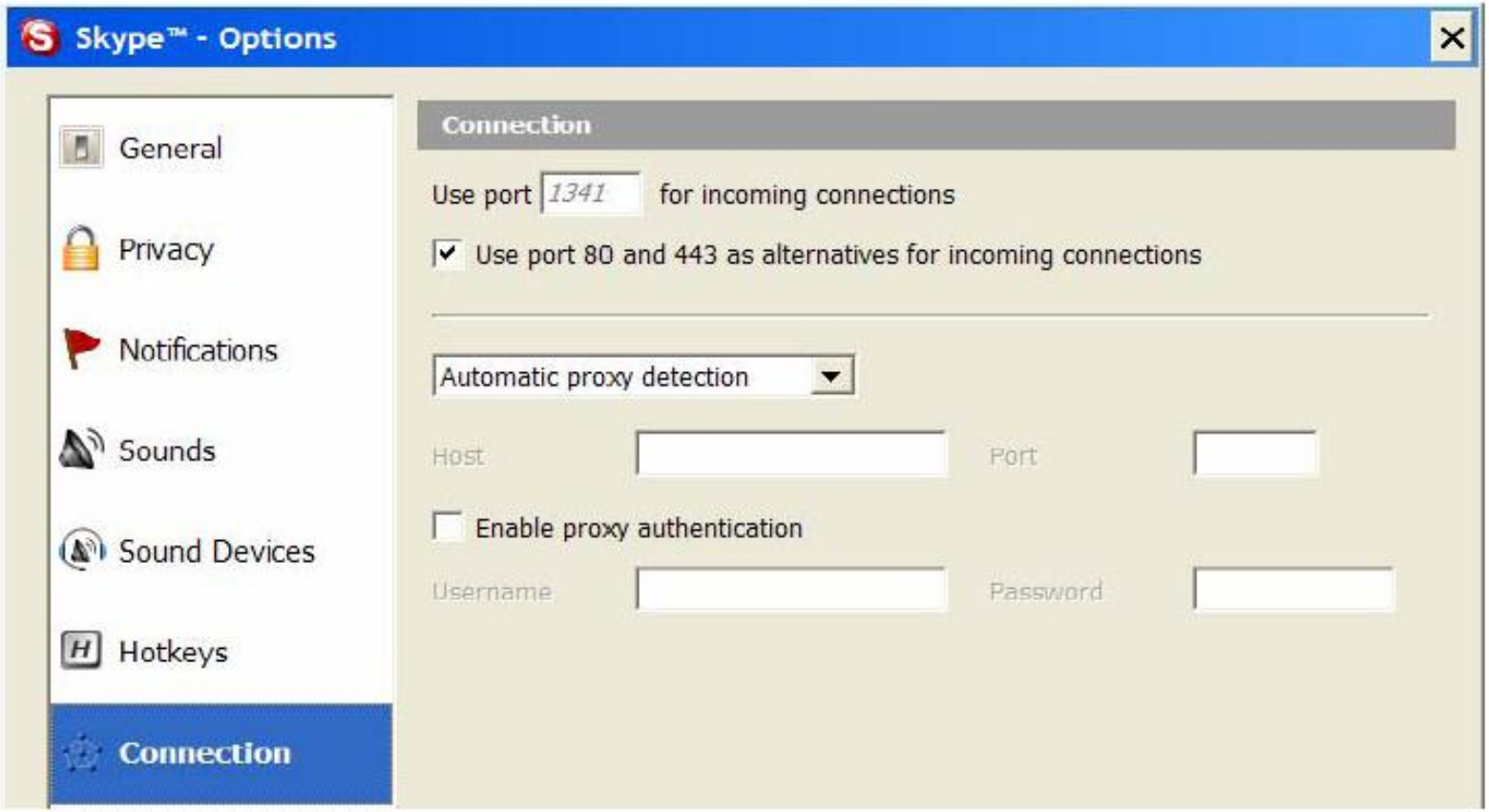
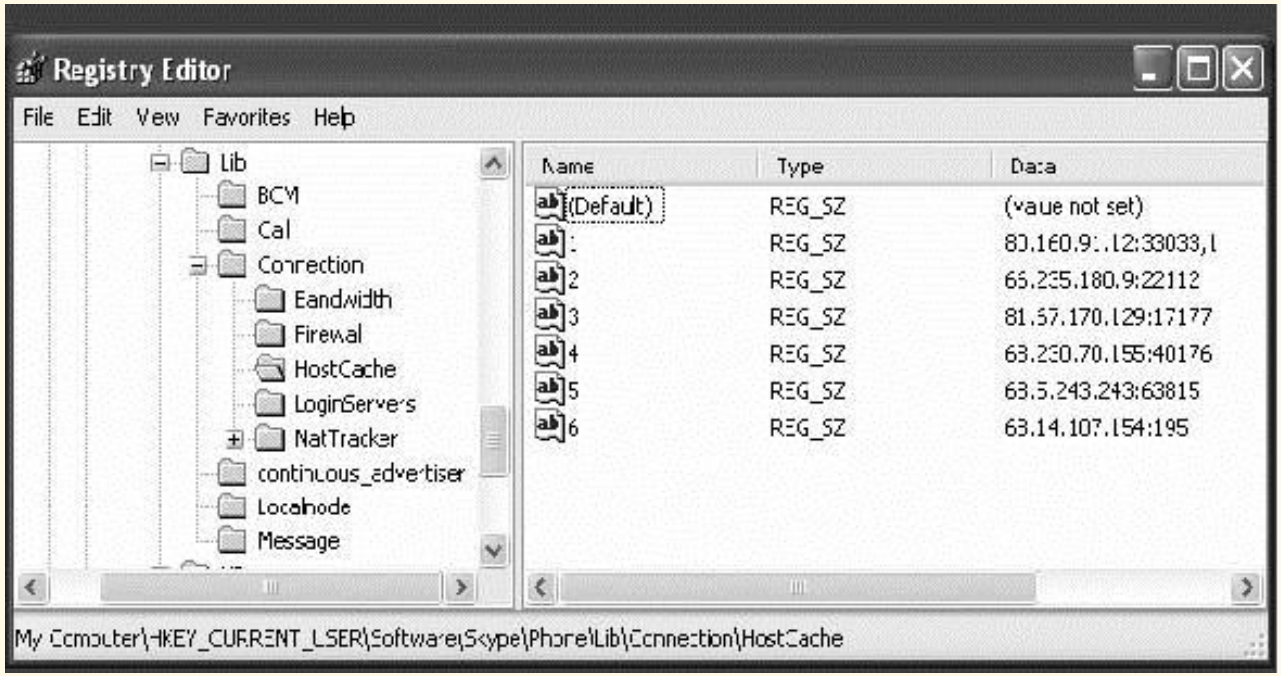


Figure 14. Skype (v1.4) connection tab. It shows the port on which Skype listens for incoming connections.

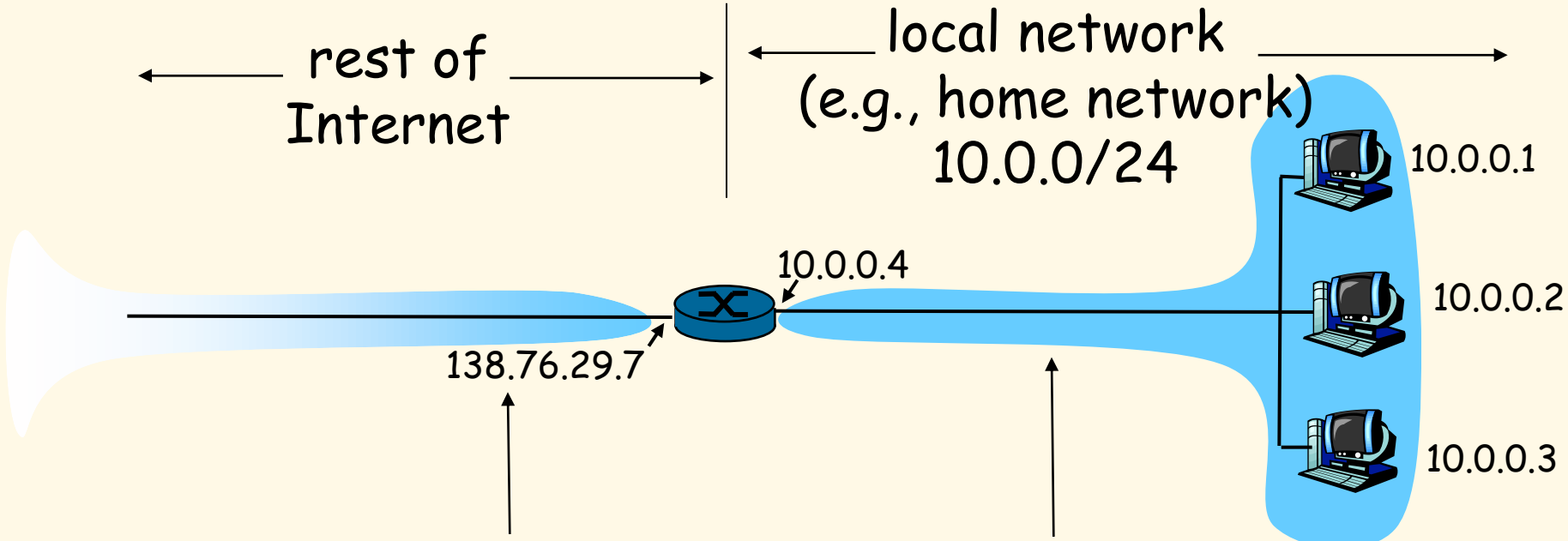
Host Cache

- Local table contains IP address, port pairs for reachable SNs {max is 200 entries}.
  - Host cache is populated on the first login.
  - SNs are periodically added/dropped as Skype runs.



[Keating 09]

NAT: Network Address Translation



All datagrams ***leaving*** local network have **same** single source NAT IP address: 138.76.29.7, different source port numbers

Datagrams with source or destination in this network have 10.0.0/24 address for source, destination (as usual)

NAT: Network Address Translation

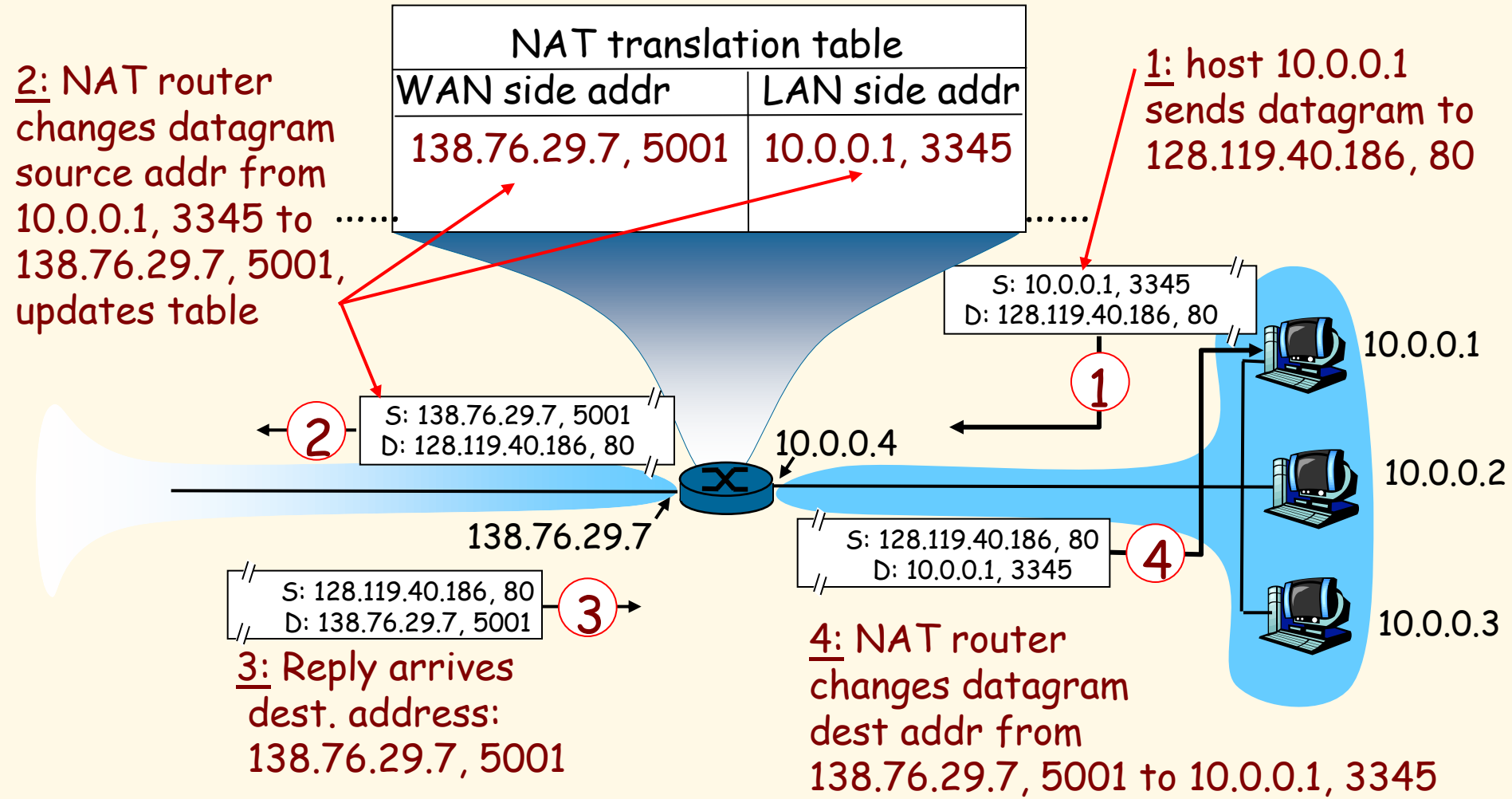
- Motivation: local network uses just one IP address as far as outside world is concerned:
  - range of addresses not needed from ISP: just one IP address for all devices.
  - can change addresses of devices in local network without notifying outside world.
  - can change ISP without changing addresses of devices in local network.
  - devices inside local net not explicitly addressable, visible by outside world (a security plus).

NAT: Network Address Translation

Implementation: NAT router must:

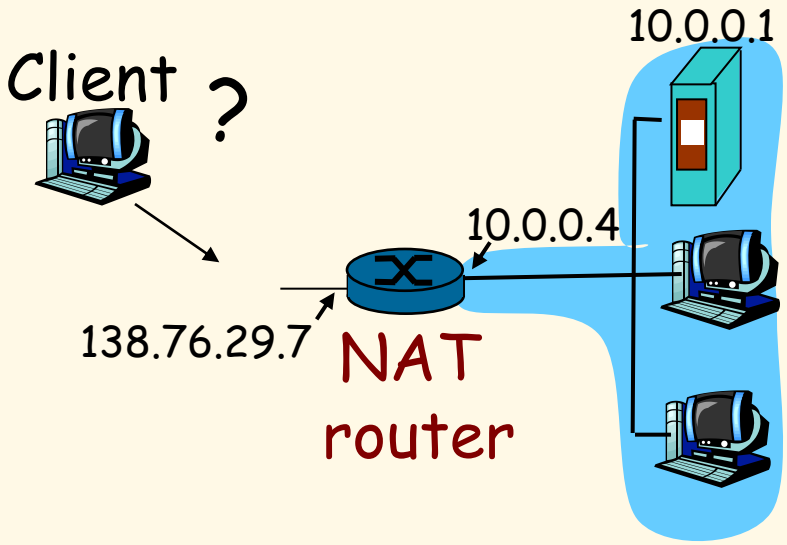
- **outgoing datagrams: replace** (source IP address, port #) of every outgoing datagram to (NAT IP address, new port #)
  - . . . remote clients/servers will respond using (NAT IP address, new port #) as destination address.
- **remember (in NAT translation table)** every (source IP address, port #) to (NAT IP address, new port #) translation pair
- **incoming datagrams: replace** (NAT IP address, new port #) in dest fields of every incoming datagram with corresponding (source IP address, port #) stored in NAT table.

NAT: Network Address Translation



NAT Traversal Problem

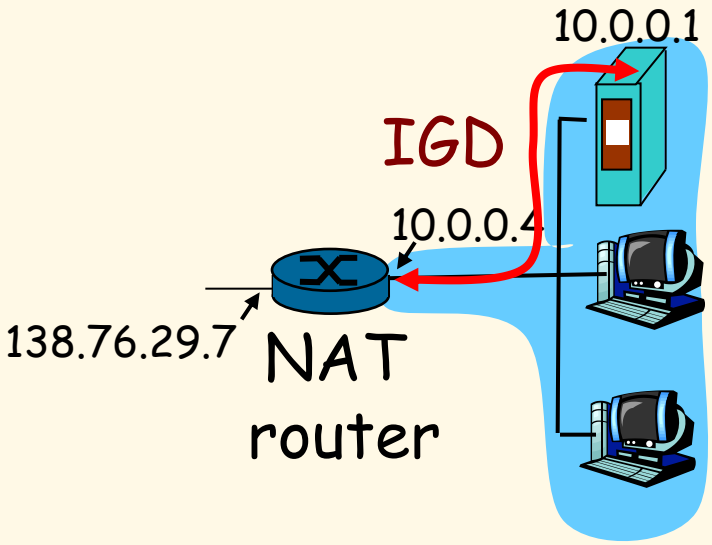
- client wants to connect to server with address 10.0.0.1
  - server address 10.0.0.1 local to LAN (client can't use it as destination addr)
  - only one externally visible NATted address: 138.76.29.7
- Solution 1: statically configure NAT to forward incoming connection requests at given port to server
  - e.g., (123.76.29.7, port 2500) always forwarded to 10.0.0.1 port 25000



NAT Traversal Problem

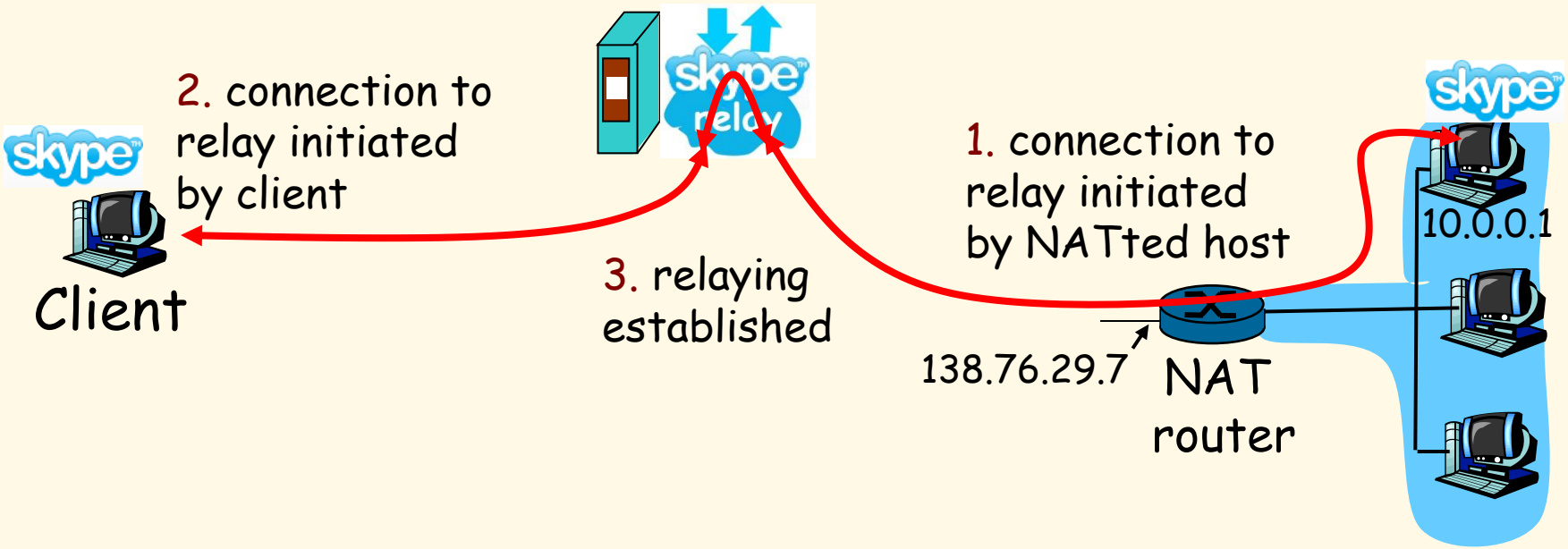
- Solution 2: Universal Plug and Play (UPnP) Internet Gateway Device (IGD) Protocol. Allows NATted host to:
  - learn public IP address (138.76.29.7)
  - add/remove port mappings (with lease times)

  i.e., automate static NAT port map configuration



NAT Traversal Problem

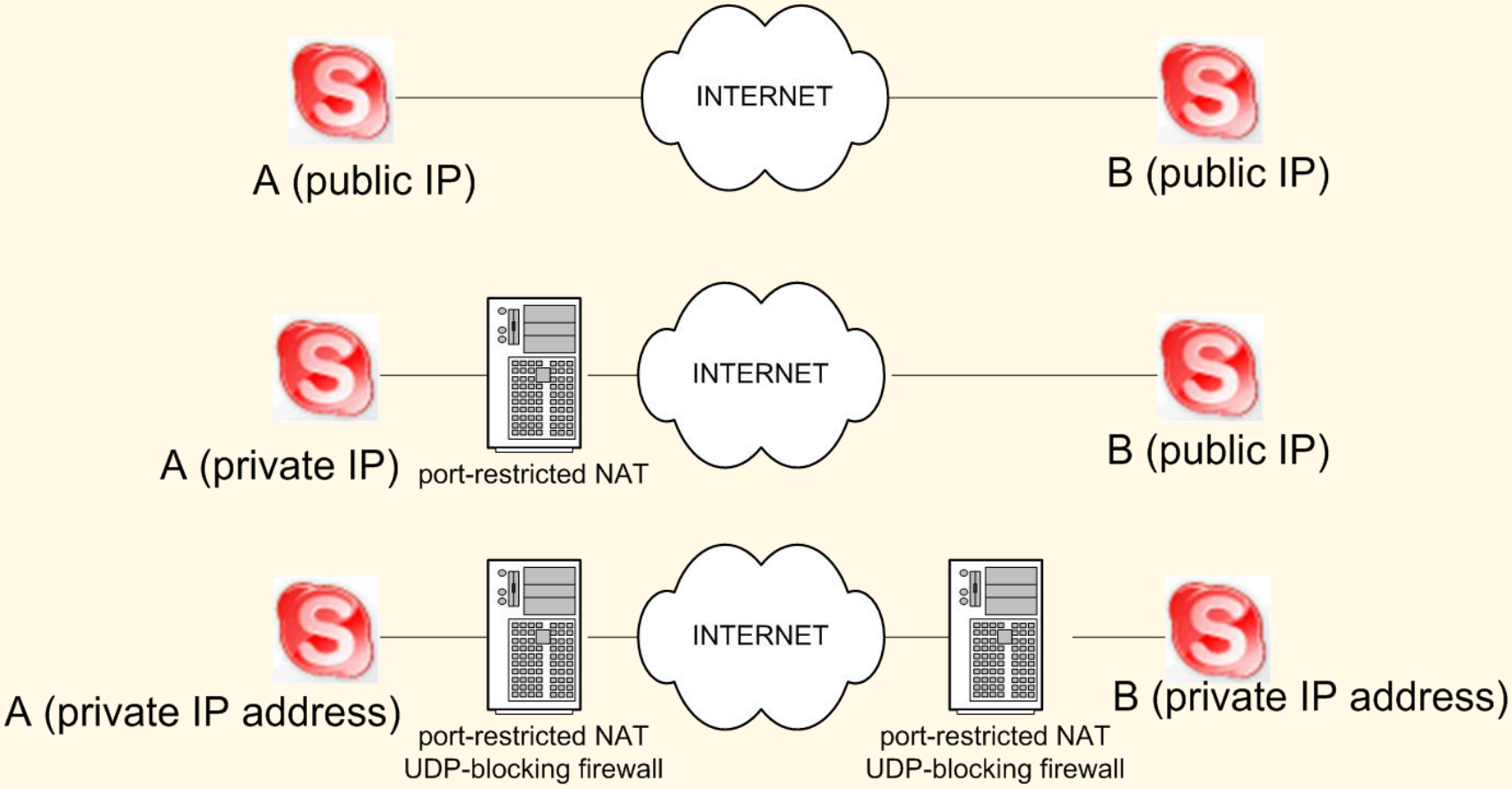
- Solution 3: relaying (used in Skype)
  - NATed client establishes connection to relay
  - External client connects to relay
  - relay bridges packets between to connections



Skype Experimental Setup

- Performed traffic analysis on Windows Skype version 1.4.0.84 and Linux Skype version 1.20.18 in November-December 2005.
- Windows XP machines (3 GHz Pentium 4 CPU, 1GB RAM) with 10/100 Mbps Ethernet card connected to 100 Mbps network.
- (Wireshark) Ethereal network protocol analyzer
  - Captures all traffic passing over a network.
- NetPeeker
  - Used to tune capacity levels.

Experimental Setup



Taken from INFOCOMM06 Presentation

[Keating 09]

Skype Functionality

Login

Login Server

User Search

Call Establishment

Conferencing

Advanced Computer Networks

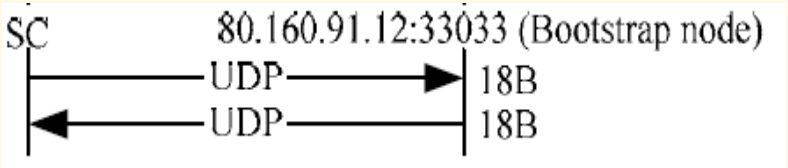
Analysis of Skype 2006

Skype Login

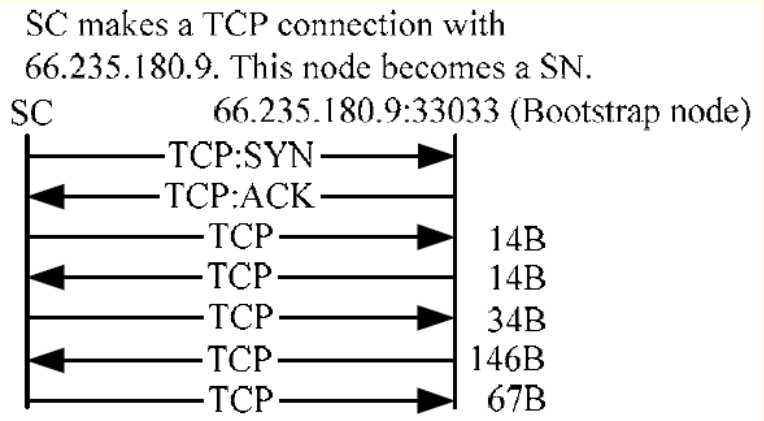
- On the first login, Skype client establishes UDP connection with Bootstrap SuperNode (BN).
  - Hard-coded into Skype client application.
- Logins routed through a SuperNode.
  - If no SuperNodes are reachable, login fails.
- Attempts to use Ports 80 and 443 if behind firewall.

Login {Public IP and NAT}

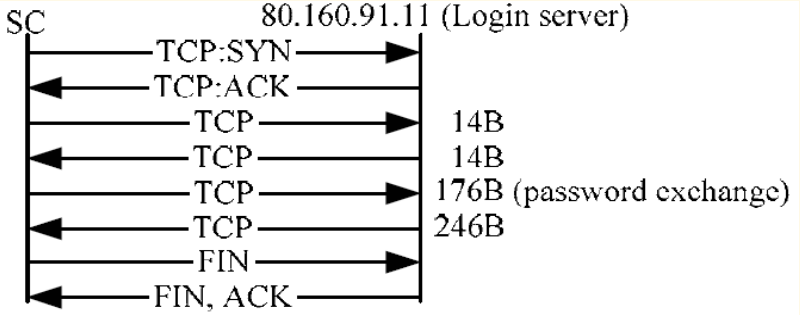
- SC->BN UDP Connection



- SC->SN TCP Connection



- SC->Login Server Auth



- 3-7 seconds

[Keating 09]

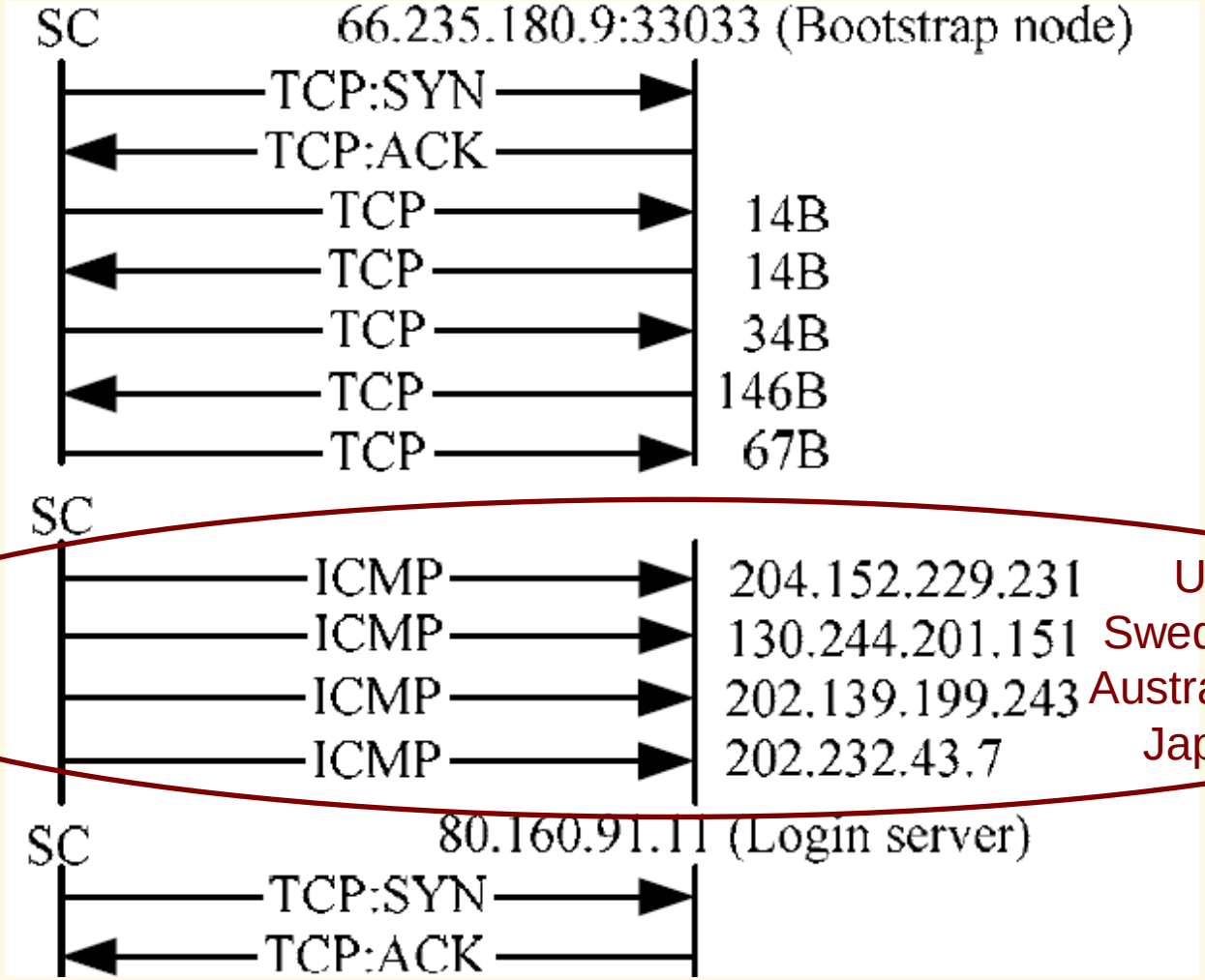
Skype ver 1.4 Login Experiment

- Copy of SC uninstalled; Windows registry cleared of Skype entries; new copy of SC installed.

TABLE I
SKYPE (VER 1.4) LOGIN EXPERIMENT SUMMARY

Skype on a Machine with/behind	Data Exchanged	Time to Login
Public IP address	10 KB	3-7 s
Port-restricted NAT	11 KB	3-7 s
UDP-restricted firewall	7 KB	35 s

Mystery ICMP Packets



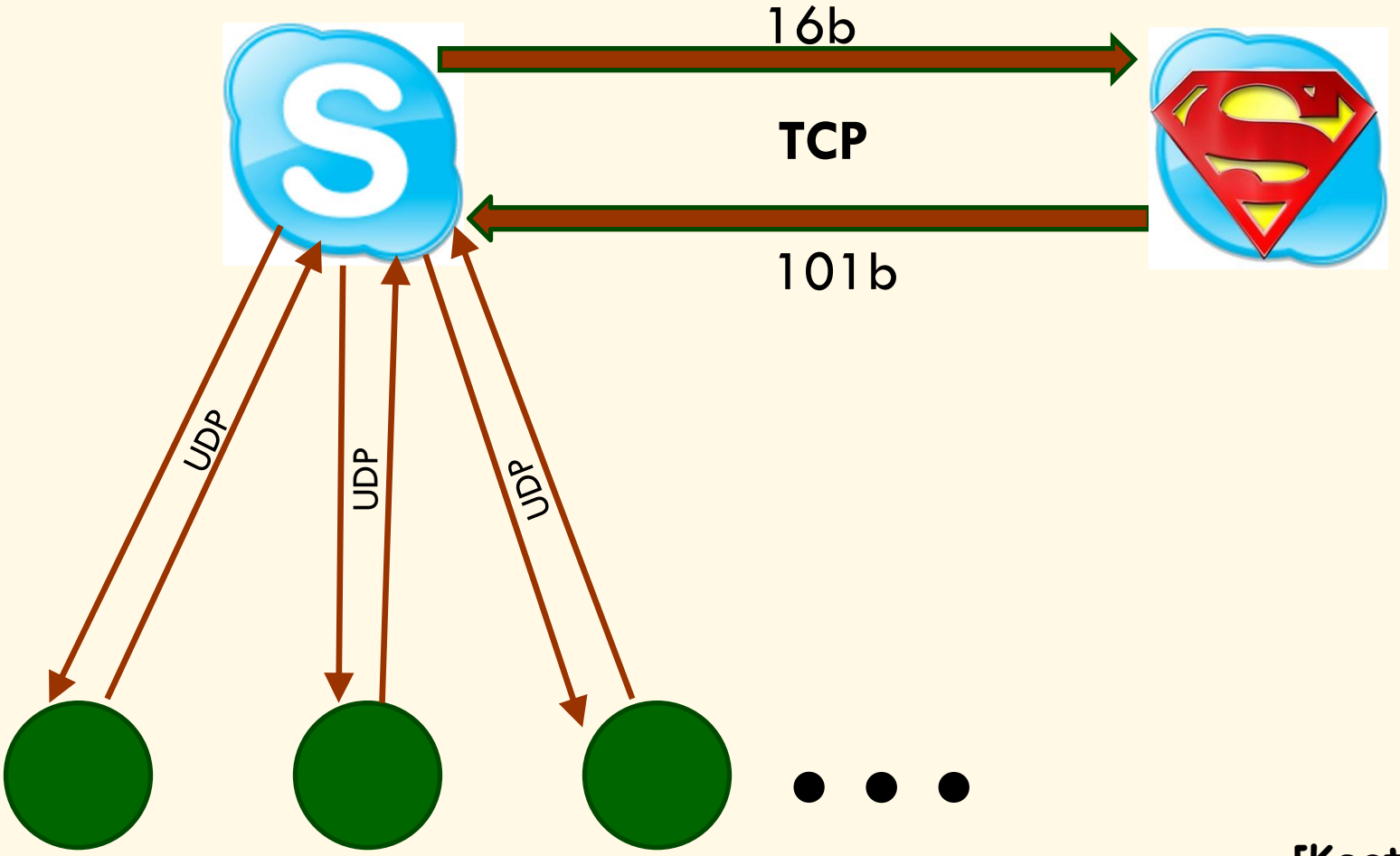
Login Server

- Login Server is ONLY central component in Skype P2P network.
- After SC connects to SN, SC authenticates with Login Server.
- Experiments show SC exchanging data over TCP with 212.72.49.141 or 195.215.8.141 (Login Servers).

Skype User Search

- Uses Global Index technology.
- Skype guarantees it will find any user logged in (public or private IP) in last 72 hours.
- Search depends on where SC resides.
- Experiments show SC performs user information caching at intermediate nodes.

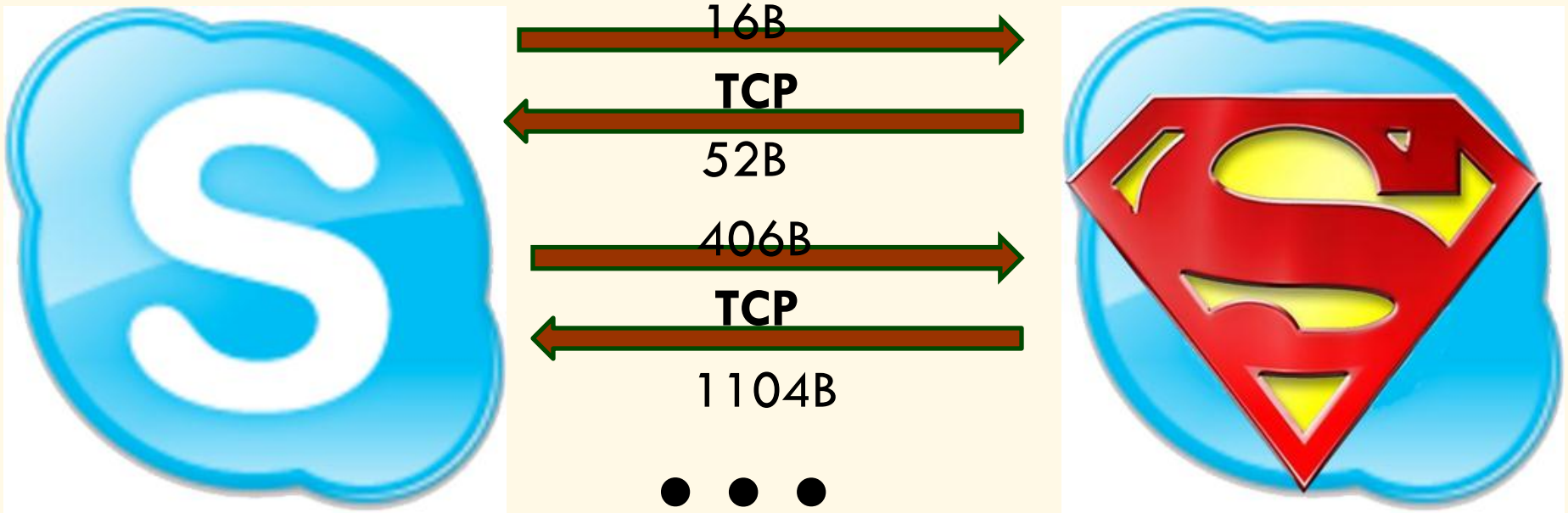
User Search from Public IP/NAT



[Keating 09]

User Search from UDP-Restricted Firewall

- SuperNode performs search



[Keating 09]

Call Establishment

- Skype uses buddy list.
- Call signaling carried out with TCP.
- Initial message exchanges uses a "challenge-reponse" mechanism.

Caller and Callee Public IP Addresses

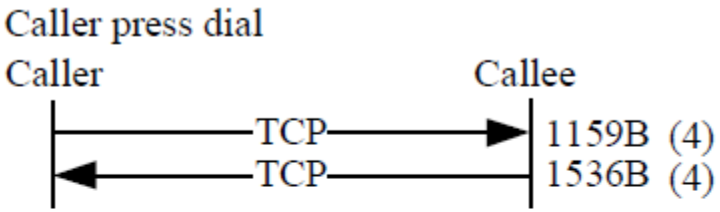


Figure 8. Message flow for call establishment when caller and callee SC v1.4 are on machines with public IP addresses and the callee is present in the buddy list of the caller. 'B' stands for bytes. Messages have been aggregated for space. Message size corresponds to the approximate cumulative size of messages exchanged between caller and a callee and vice versa. The number in paranthesis shows the total number of messages sent in that direction.

Caller behind NAT

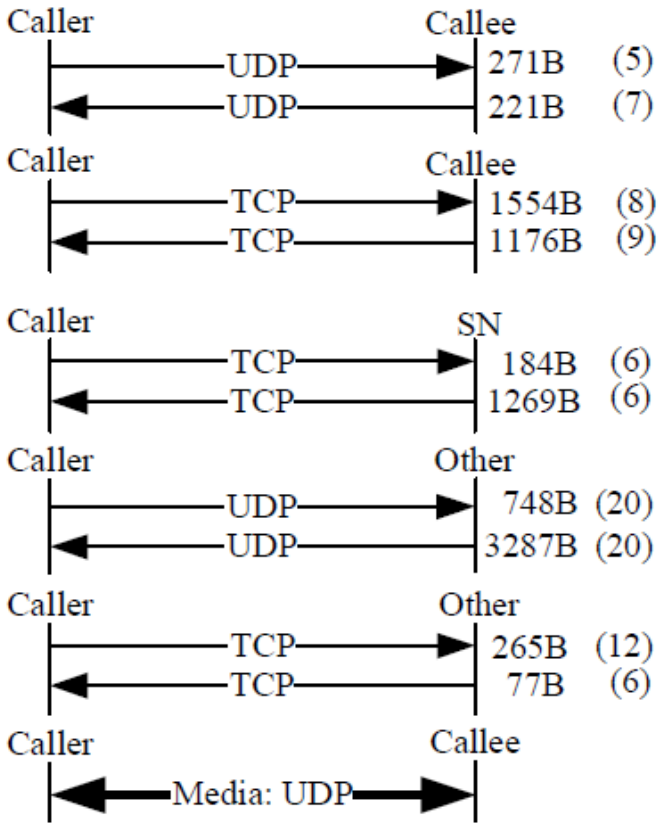
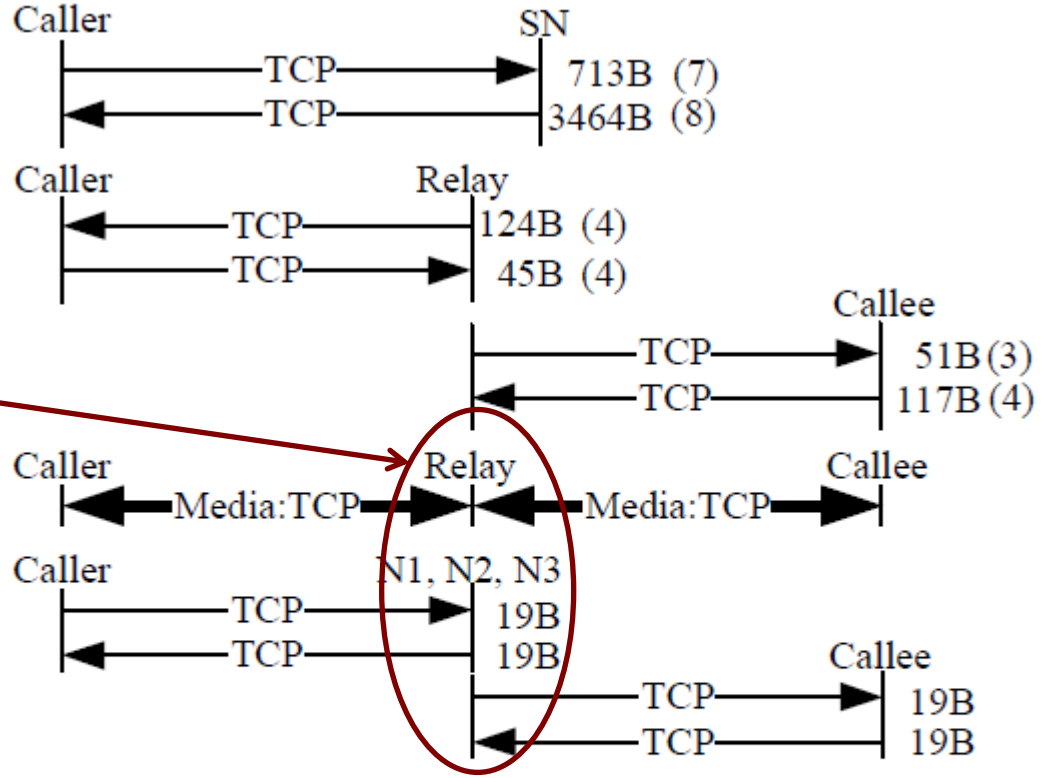


Figure 9. Message flow for call establishment when caller SC is behind a port-restricted NAT and callee SC is on a public IP address. 'B' stands for bytes. Not all messages are shown. Messages have been aggregated for space. Message size corresponds to the approximate cumulative size of messages exchanged between caller, callee, SN, other nodes and vice versa. The number in paranthesis shows the total number of messages sent in that direction.

Caller and Callee behind NAT

Users generally do not like that arbitrary traffic can flow across their machine!!



Caller and callee on the average exchange 3 msg/s over TCP with N1, N2 and N3 after call has been established.

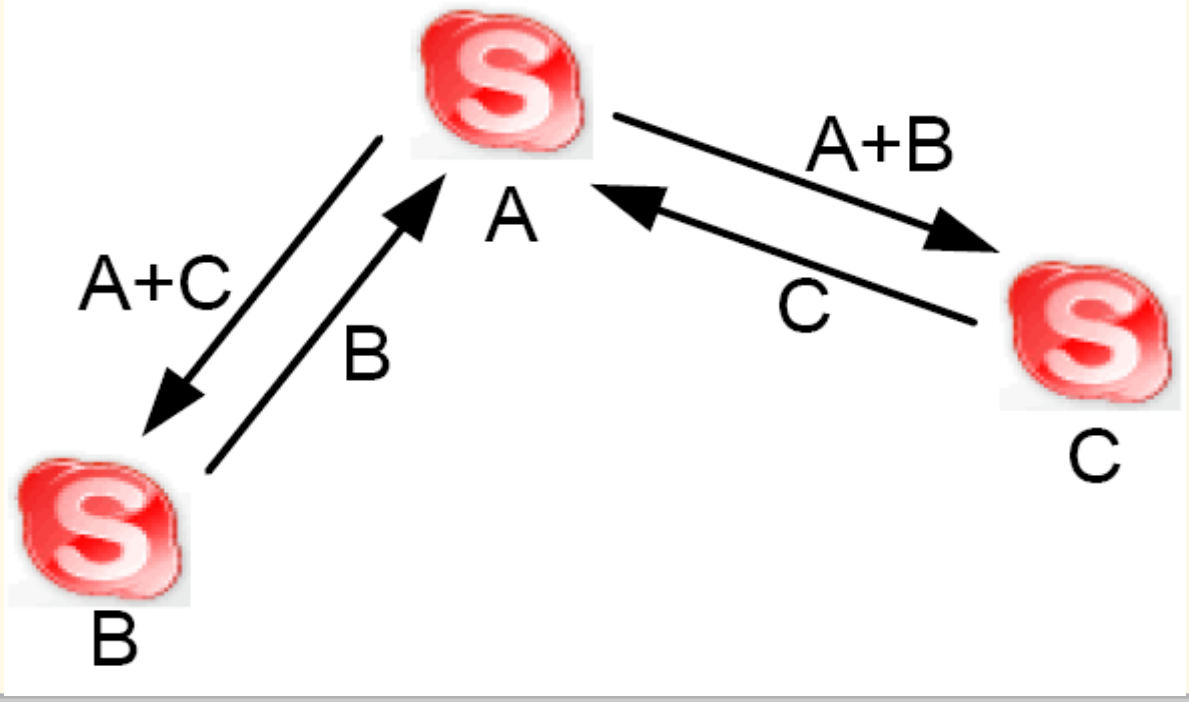
Figure 10. Message flow for call establishment when caller and callee SC are behind a port-restricted NAT and UDP-restricted firewall. 'B' stands for bytes and 'N' stands for a node. Not all messages are shown. Messages have been

Media Transfer

- Internet Speech Audio Codec (iSAC)
- Frequency range: 50-8000Hz
- Public IPs communicate directly.
  - NAT/firewall users use an SN relay node.
- Uses UDP Transport if possible.
  - 5 kilobytes/sec
  - UDP-restricting firewall users communicate over TCP
- Does not perform Silence Suppression.

Skype Conferencing

- A: 2GHz P4 w/ 512MB RAM
- B, C: 300MHz P2 w/ 128MB RAM
- A acts as mixer for both B and C



Skype Super Nodes

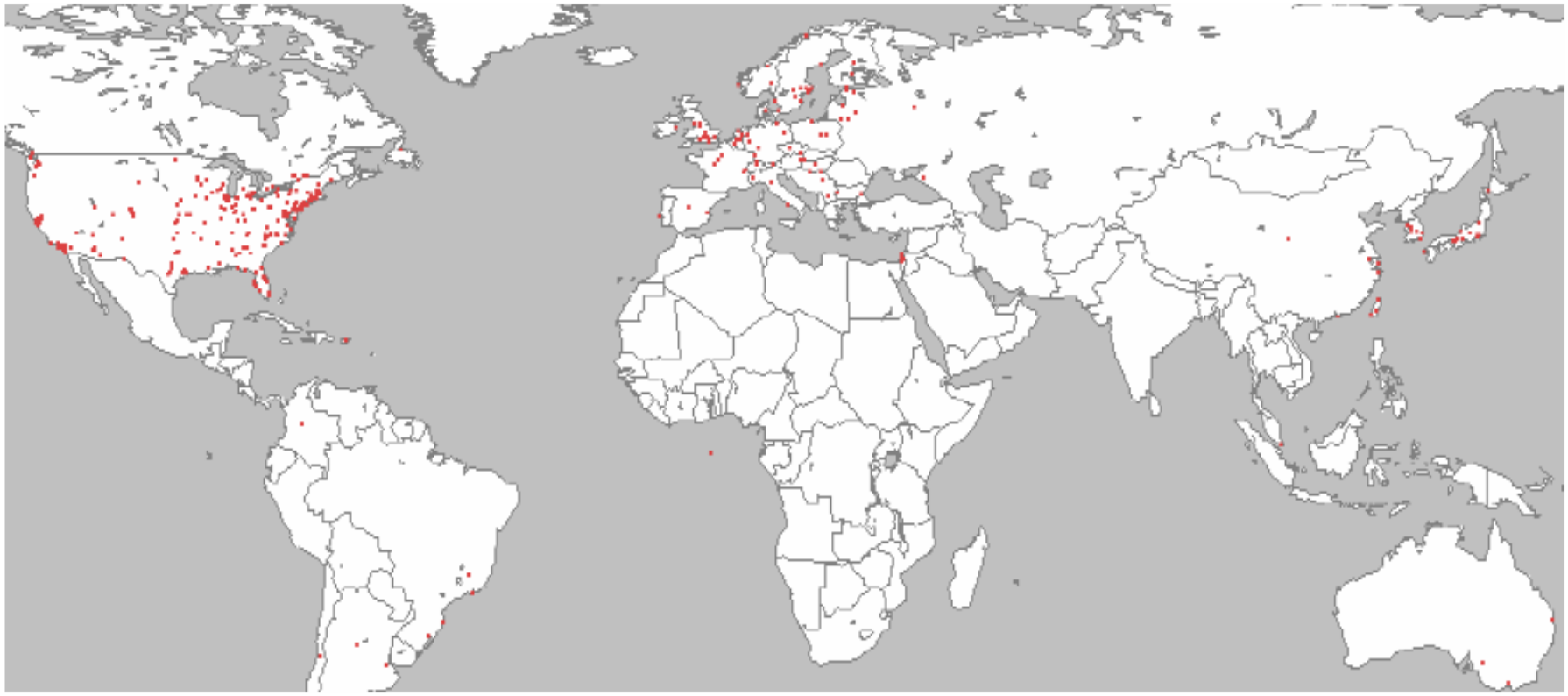
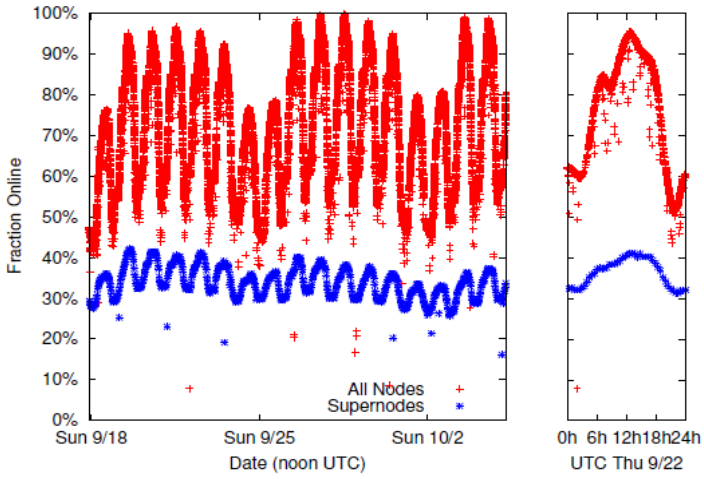


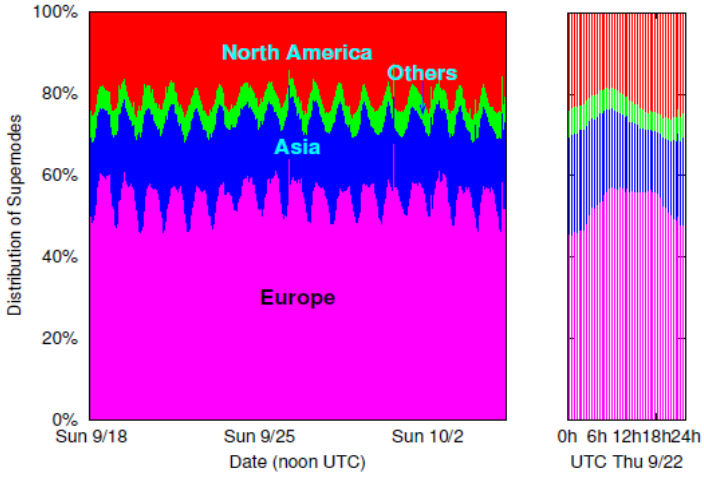
Figure 16. Worldmap of super nodes to which Skype establishes a TCP connection at login.

Super Node Behavior

[Guha 06]



(a)



(b)

Figure 1: (a) Percentage of all nodes, and supernodes active at any time. (b) Geographic distribution of supernodes as observed over the duration of our trace.

Super Node 'Churn'

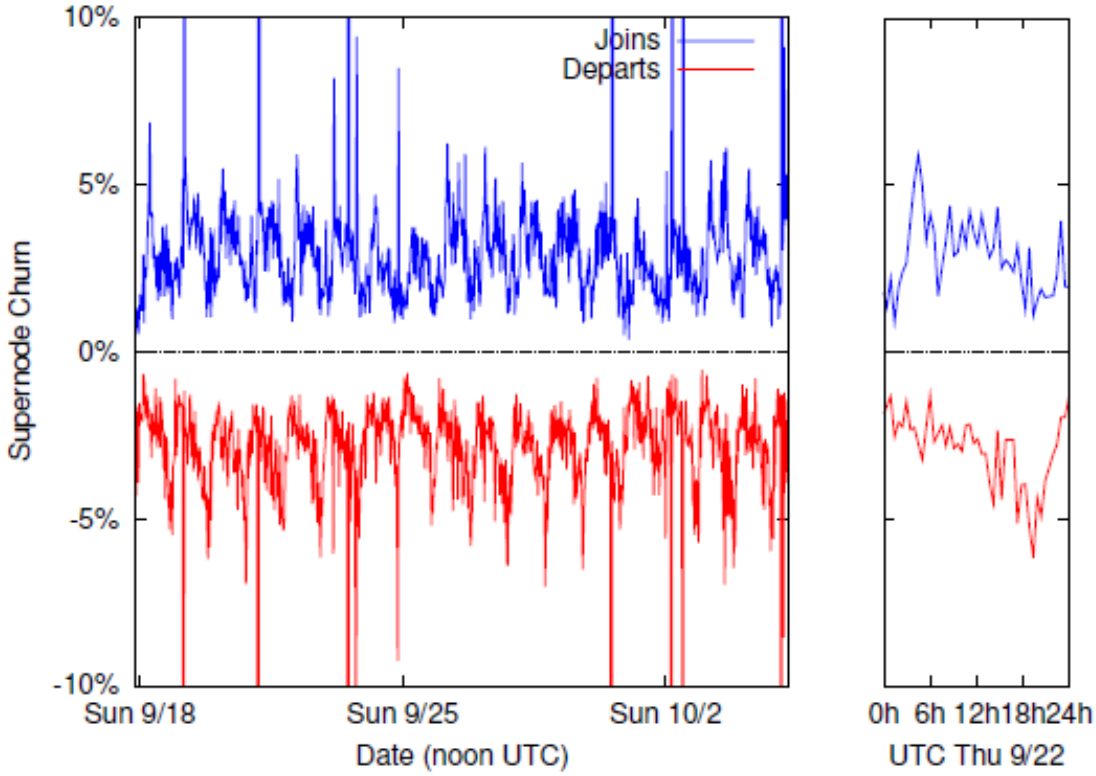


Figure 2: Fraction of supernodes joining or departing the network over the duration of our trace.

[Guha 06]

Super Node Bandwidth Consumption

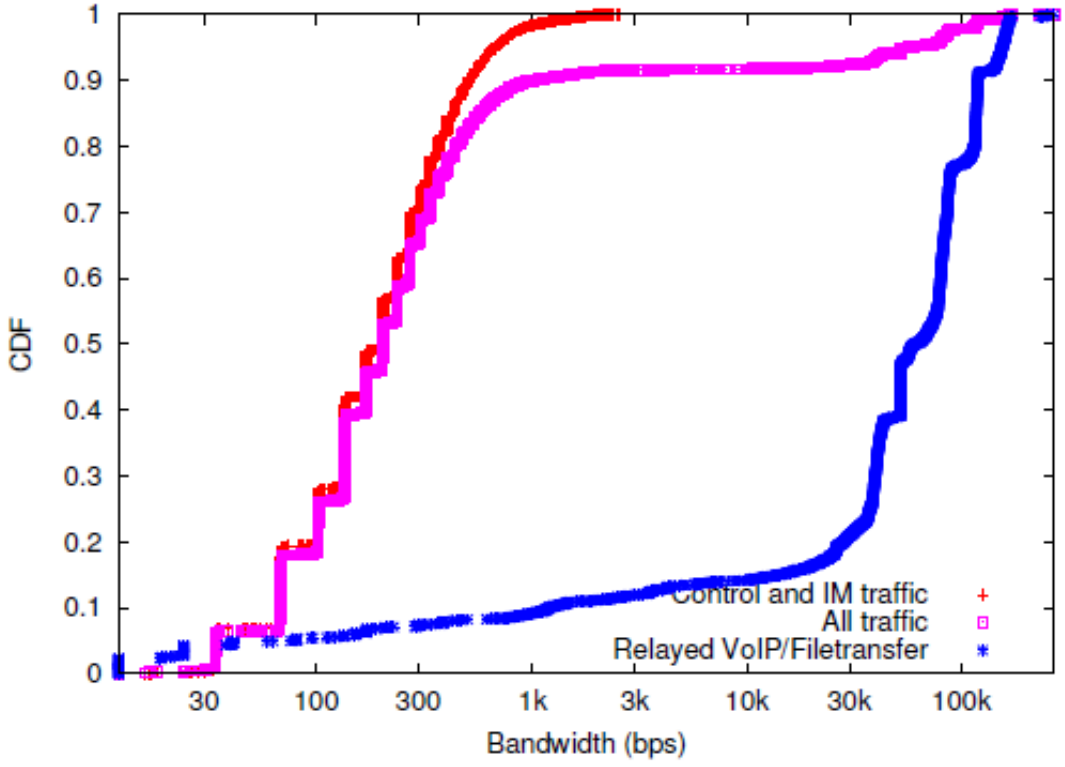


Figure 4: Semi-log plot of CDF of bandwidth used by our supernode.

[Guha 06]

Conclusions

- Skype can work behind NATs and firewalls using STUN protocol.
- Skype architecture relies on Super Nodes with public IP addresses.
- Skype uses TCP for signaling and prefers to use UDP for media transfer.
- Skype packets are encrypted and Skype uses a central Login Server to authenticate Skype users.

References

- [Guha 06] S. Guha, N. Dawani and R. Jain, "An Experimental Study of the Skype Peer to Peer VoIP System", The Fifth International Workshop on Peer-to-Peer Systems (IPTPS06), Santa Barbara, CA, February 2006.
- [Keating 09] Andrew Keating presentation in CS577 Fall 2009.

Thanks!

Questions?