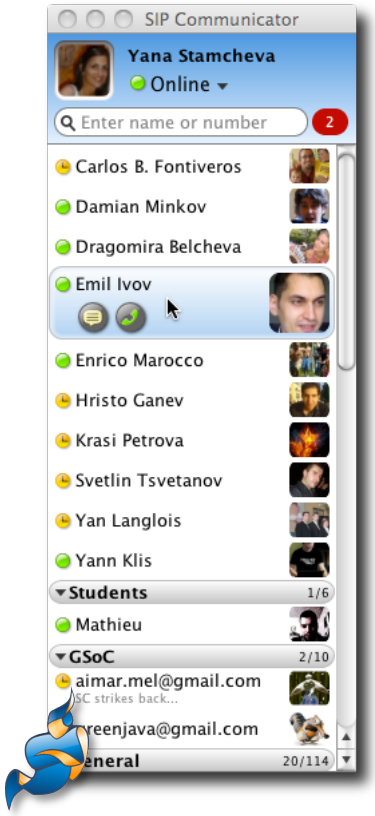
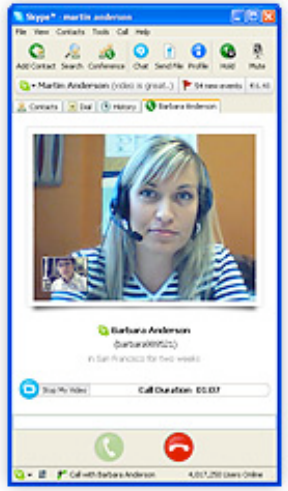
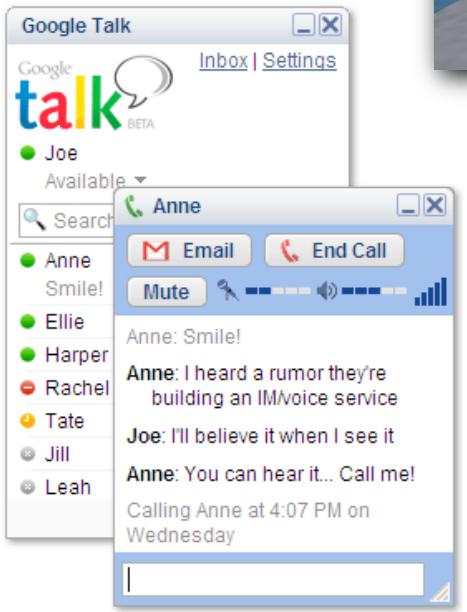
Real-time Communication Protocols

Real-time Communication Applications







sip & xmpp

The basics of IP telephony
Registration

network core
(registrars, proxies, …)

Bob
Address: *B*
Port: *Pb*

- Authentication uses http-style digest MD5 challenges
- Content can protected with TLS but …
  - SIP often uses plain UDP
  - XMPP almost always OK

Alice
Address: *A*
Port: *Pa*

Inter-Server Communication

SIP: SMTP-like (no real limitation)

inter-server communication

XMPP: Requires TLS and use of certificates

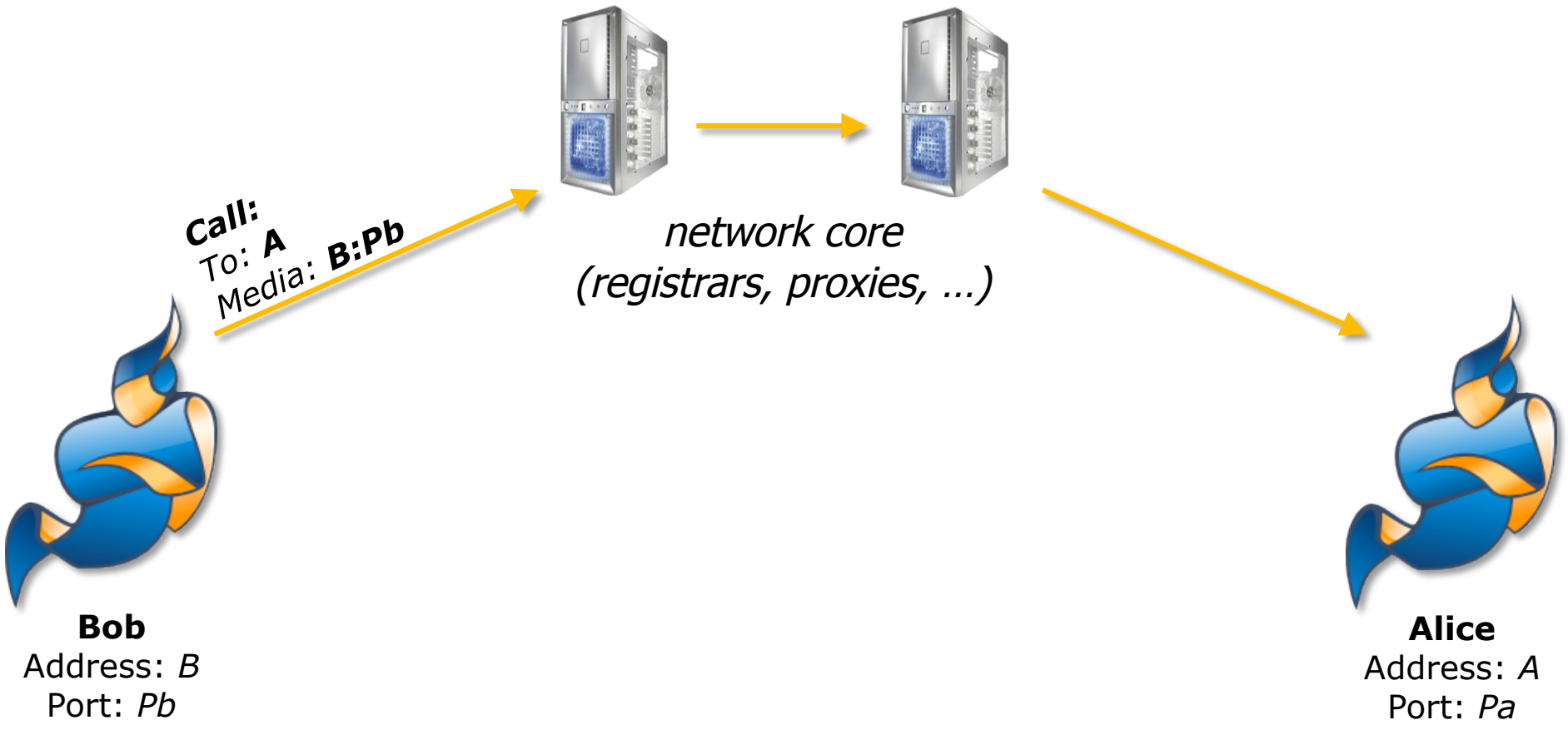
end-to-end encryption for IM uses OTR

(Off The Record messaging)

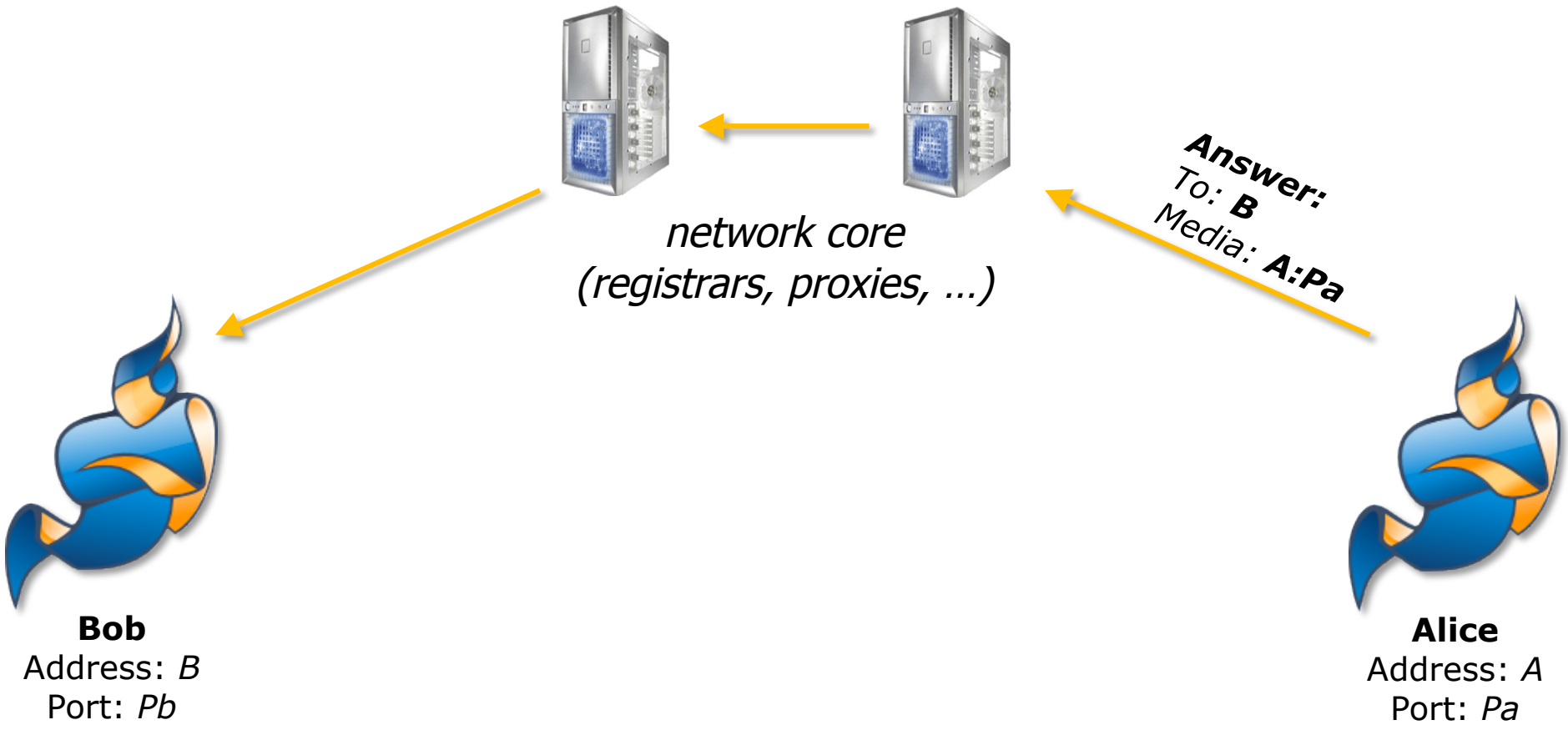
Bob
Address: *B*
Port: *Pb*

Alice
Address: *A*
Port: *Pa*

The basics of IP telephony
A sample call



The basics of IP telephony
A sample call



The basics of IP telephony.

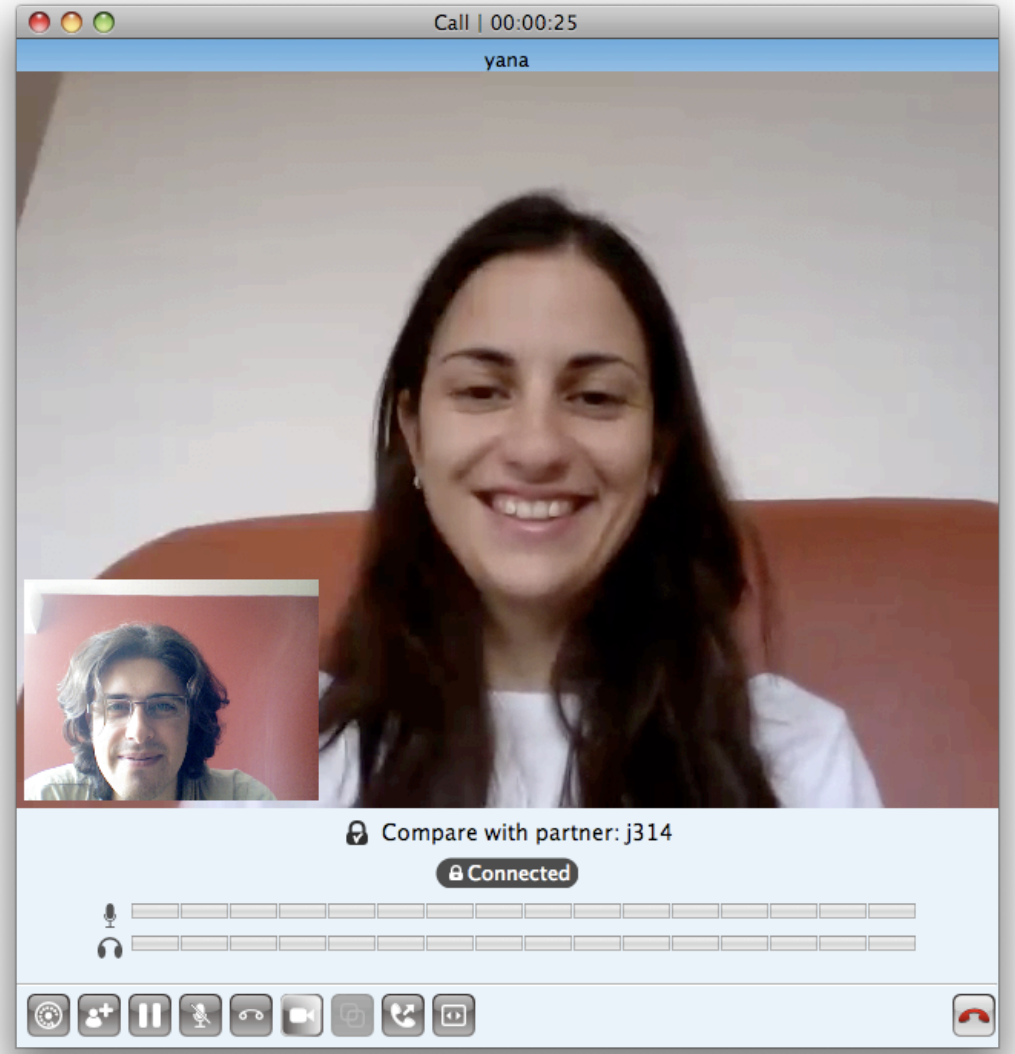
network core
(registrars, proxies, …)

Bob
Address: *B*
Port: *Pb*

Alice
Address: *A*
Port: *Pa*

Encrypting RTP

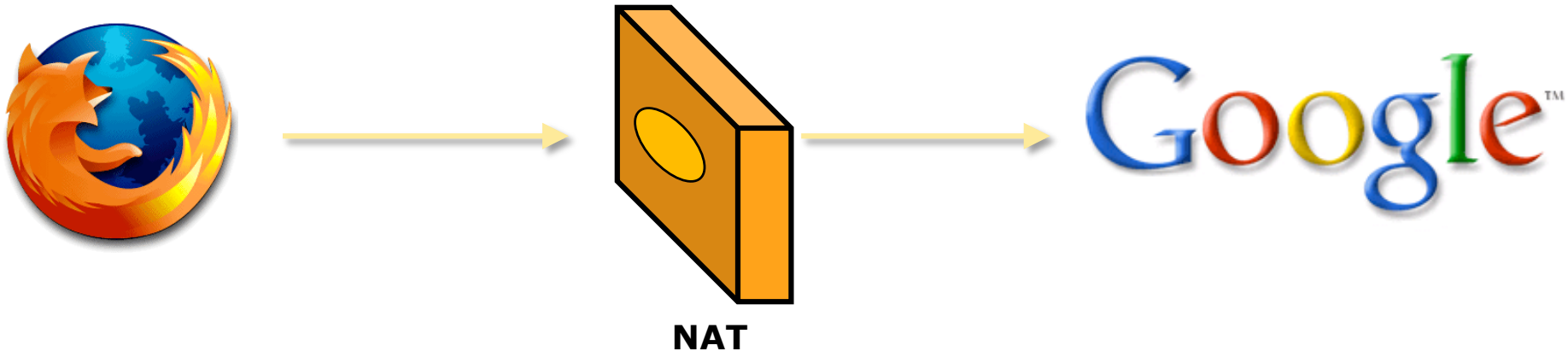
* SRTP provides tools for encrypting RTP flows
* Key management was commonly based on PKIs until
* ZRTP arrived with an alternative



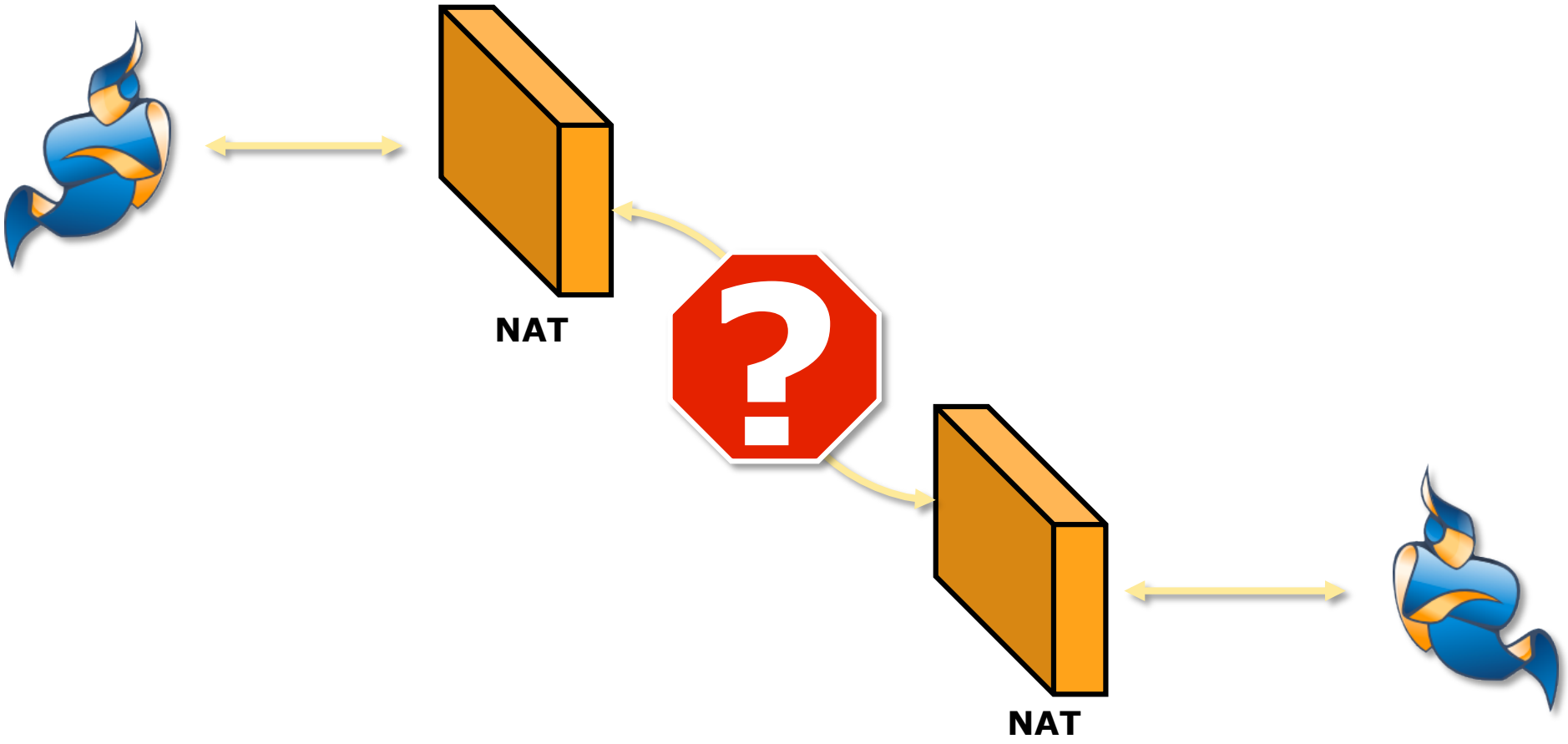
Reality check!

Are we forgetting something?

Standard NAT use



Less standard NAT usage: End – to –end services

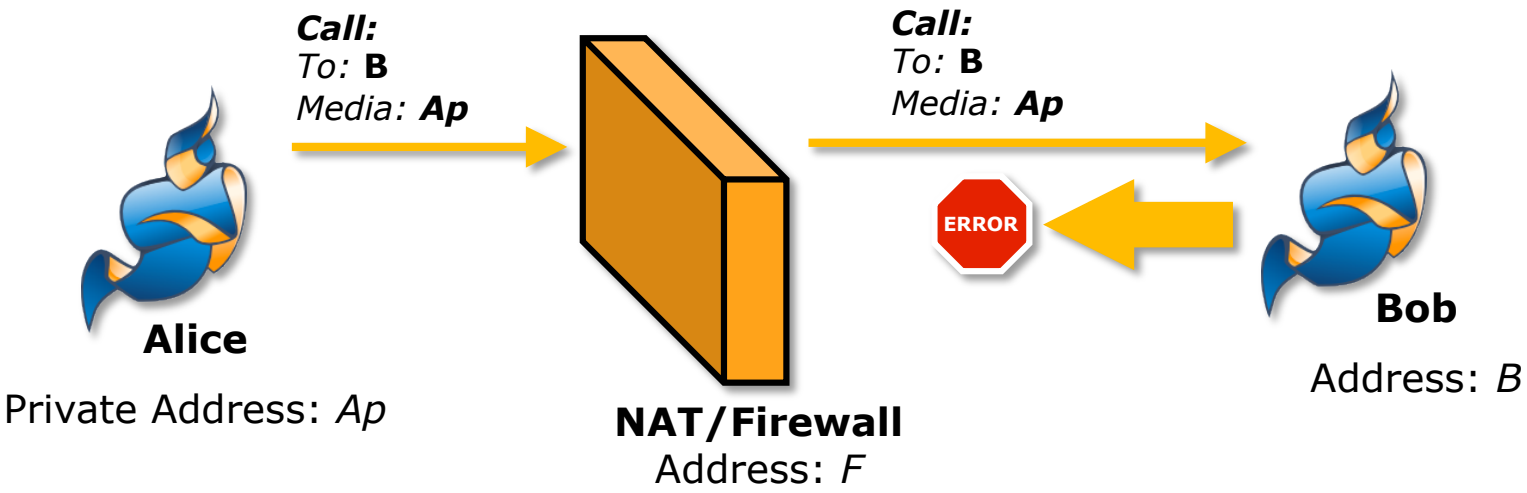


Session initialization with SIP and XMPP

```
INVITE sip:barbara@b.com SIP/2.0

Via: SIP/2.0/UDP 10.43.122.3;branch=1
From: sip:alice@a.com;tag=4ad340f
To: sip:barbara@b.com
Contact: <sip:alice@10.43.122.3>
Call-ID: 1874630@10.43.122.3
Cseq: 12442 INVITE



v=0
o=user 14341433 14341433 IP4 10.43.122.3
s=.
t=0 0
c=IN IP4 10.43.122.3
m=audio 13222 RTP/AVP 0
a=rtpmap:0 PCMU/8000
```

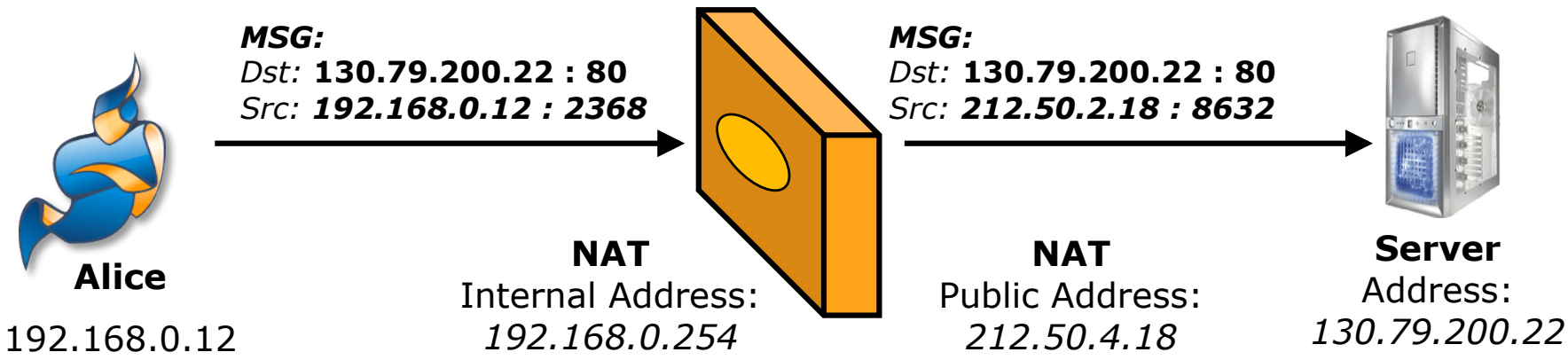
```
<iq from='juliet@capulet.lit/balcony'
    id='hs81w639'
    to='romeo@montague.lit/orchard'
    type='set'>
  <jingle xmlns='urn:xmpp:jingle:1'
          action='session-accept'
          initiator='romeo@montague.lit/orchard'
          responder='juliet@capulet.lit/balcony'
          sid='a73sjjvkla37jfea'>
    <content creator='initiator' name='voice'>
      <description xmlns='urn:xmpp:jingle:apps:rtp:1' media='audio'>
        <payload-type id='18' name='G729'/>
      </description>
      <transport xmlns='urn:xmpp:jingle:transports:raw-udp:1'>
        <candidate component='1'
                   generation='0'
                   id='z7sdjb01hf'
                   ip='208.68.163.214'
                   port='9876'/>
        <candidate component='2'
                   generation='0'
                   id='hg92lsn10b'
                   ip='208.68.163.214'
                   port='9877'/>
      </transport>
    </content>
  </jingle>
</iq>
```

And then NATs were born …



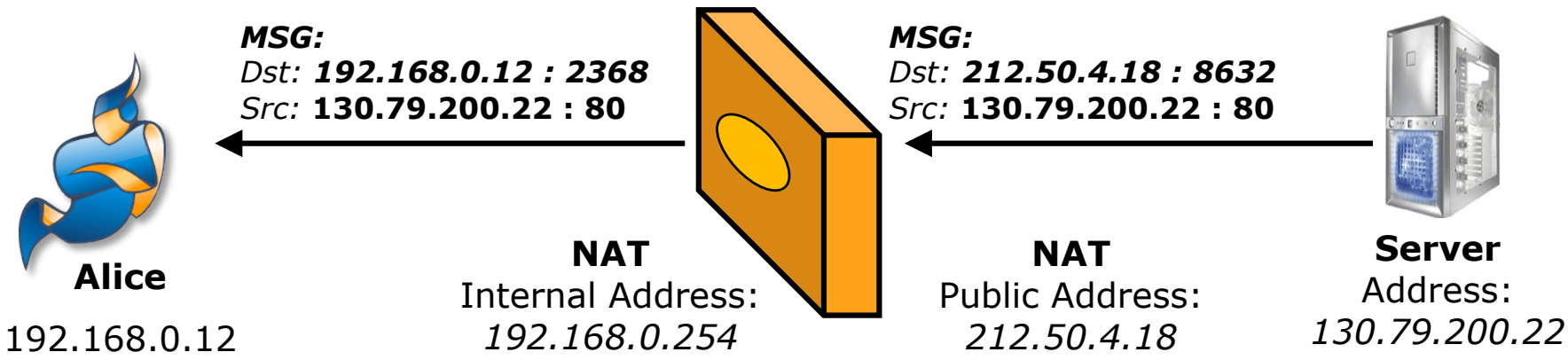
How do NATs work …

Internal **host:port**	NAT **port**
192.168.0.12 : 2368	***8632***



How do NATs work …

Internal **host:port**	NAT **port**
192.168.0.12 : 2368	***8632***



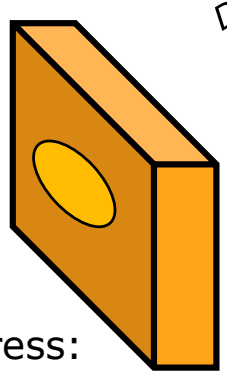
How do NATs work …

Internal host:port	NAT port
192.168.0.12 : 2368	***8632***

Endpoint-Independent Mapping
Endpoint-Independent Filtering

Bob
Address:
60.55.68.53

MSG:
Dst: ***212.50.4.18 : 8632***
Src: ***60.55.68.53 : 9595***



MSG:
Dst: ***192.168.0.12 : 2368***
Src: **60.55.68.53 : 9595**

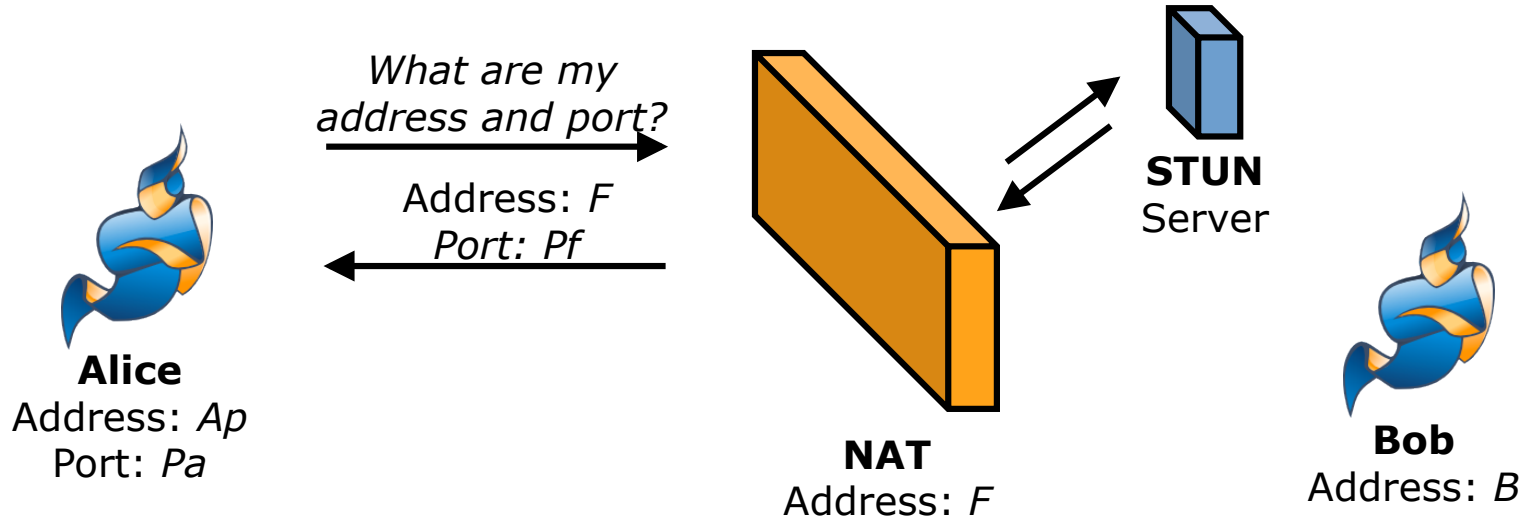
Alice
192.168.0.12

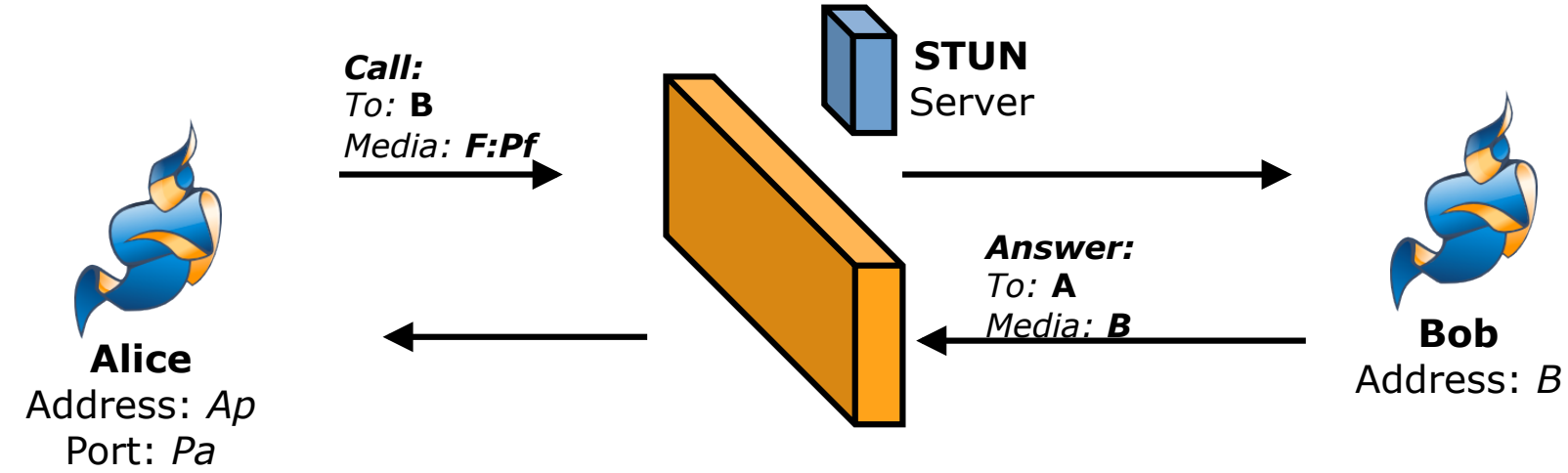
NAT
Internal Address:
192.168.0.254

NAT
Public Address:
212.50.4.18

Server
Address:
130.79.200.22

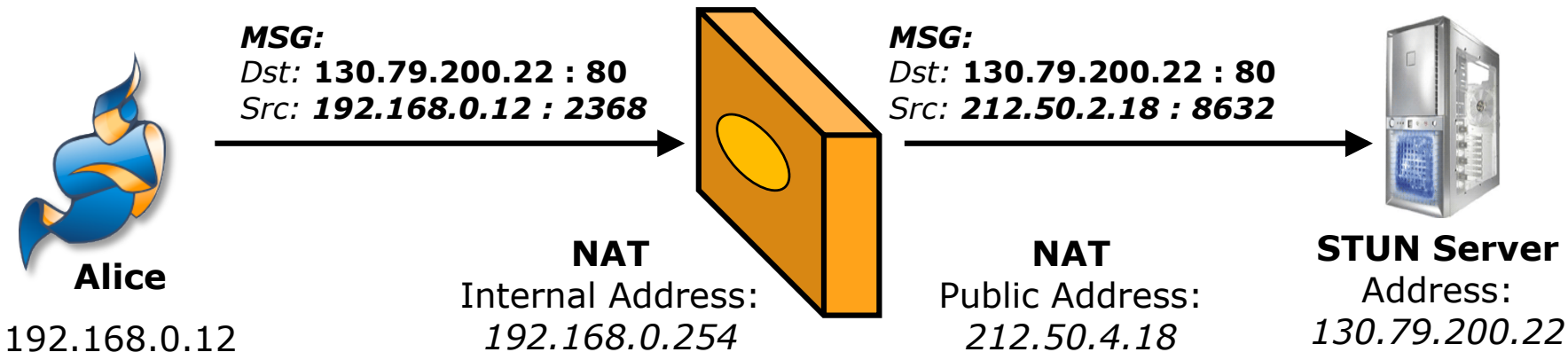
Basic Firewall and NAT Traversal
STUN





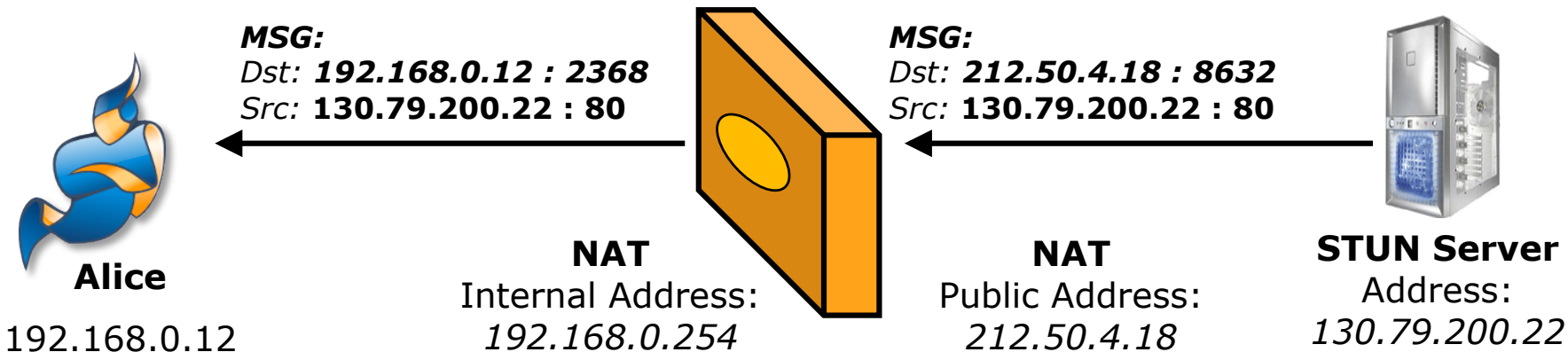
How do NATs work …
Address (and port) dependent filtering

Internal **host:port**	NAT **port**	Active connections **host:port**
192.168.0.12 : 2368	***8632***	**130.79.200.22 (: 80)**



How do NATs work …
Address (and port) dependent filtering

Internal **host:port**	NAT **port**	Active connections **host:port**
192.168.0.12 : 2368	***8632***	**130.79.200.22 (: 80)**



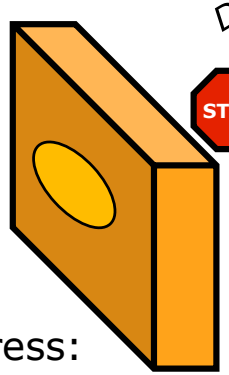
How do NATs work …
Address (and port) dependent filtering

Internal **host:port**	NAT **port**	Active connections **host:port**
192.168.0.12 : 2368	***8632***	**130.79.200.22 (: 80)**

Bob
Address:
60.55.68.53

Endpoint-Independent Mapping
Endpoint-Dependent Filtering

MSG:
Dst: **212.50.4.18 : 8632**
Src: **60.55.68.53 : 9595**



Alice
192.168.0.12

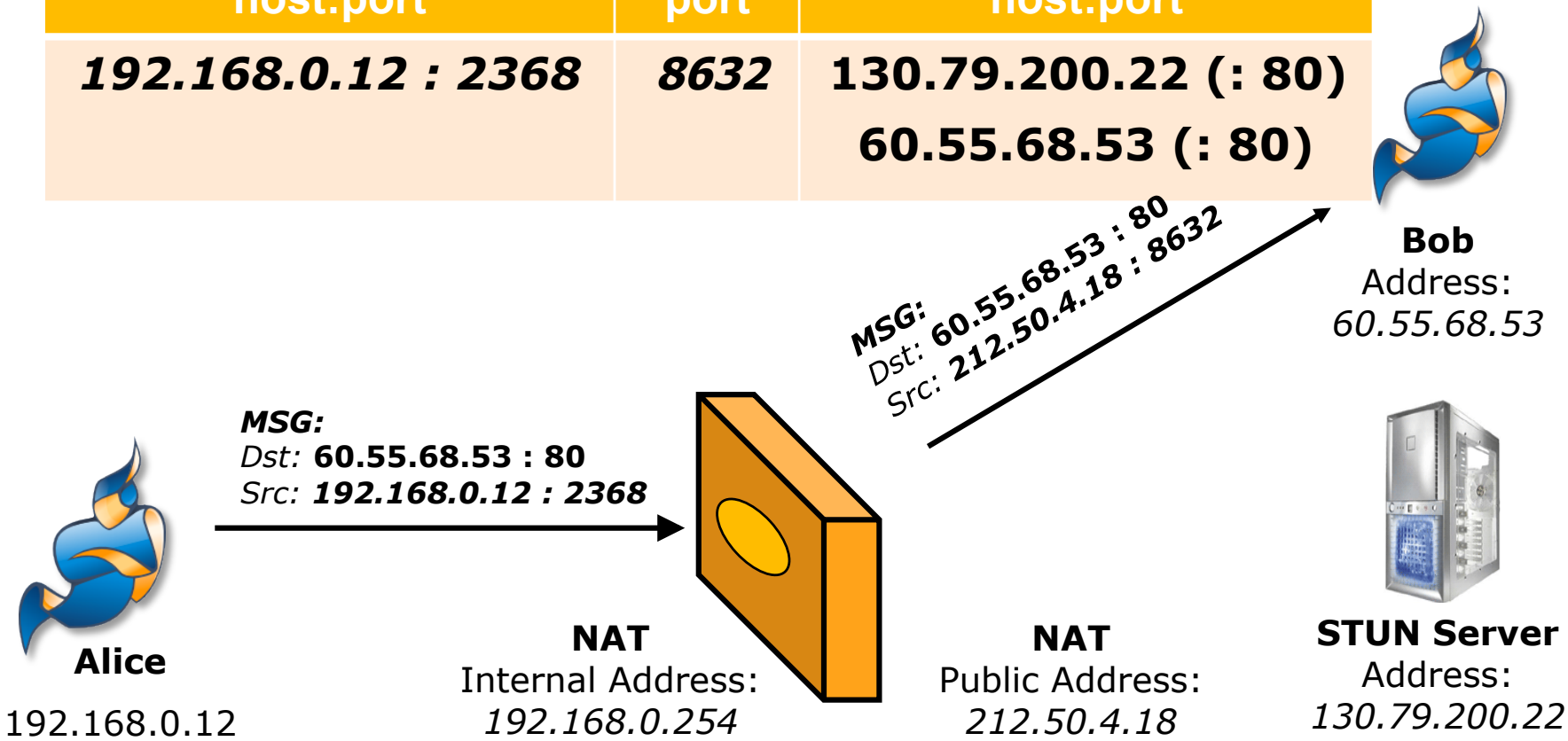
NAT
Internal Address:
192.168.0.254

NAT
Public Address:
212.50.4.18

STUN Server
Address:
130.79.200.22

How do NATs work …
Address (and port) dependent filtering

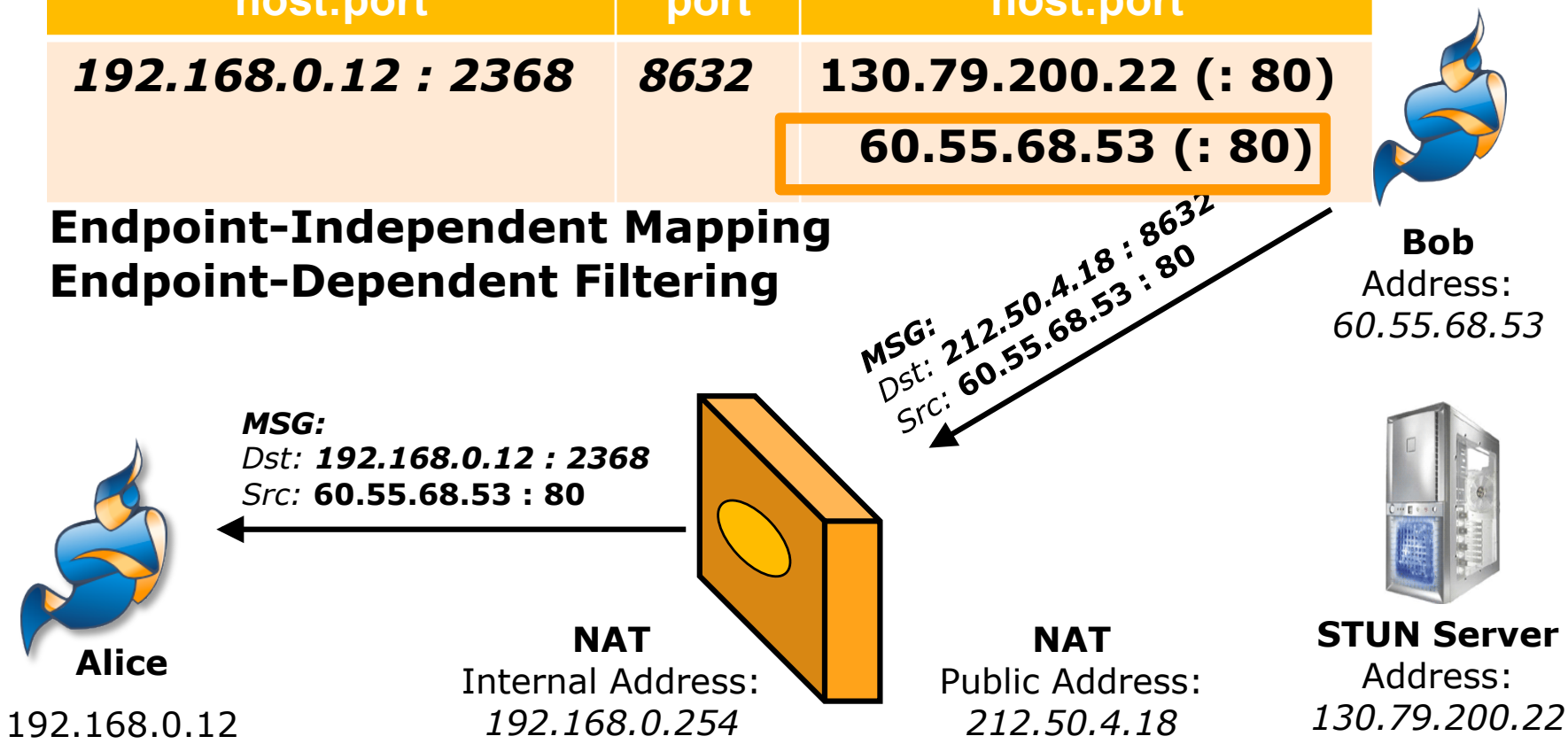
Internal **host:port**	NAT **port**	Active connections **host:port**
192.168.0.12 : 2368	***8632***	**130.79.200.22 (: 80)** **60.55.68.53 (: 80)**



How do NATs work …
Address (and port) dependent filtering

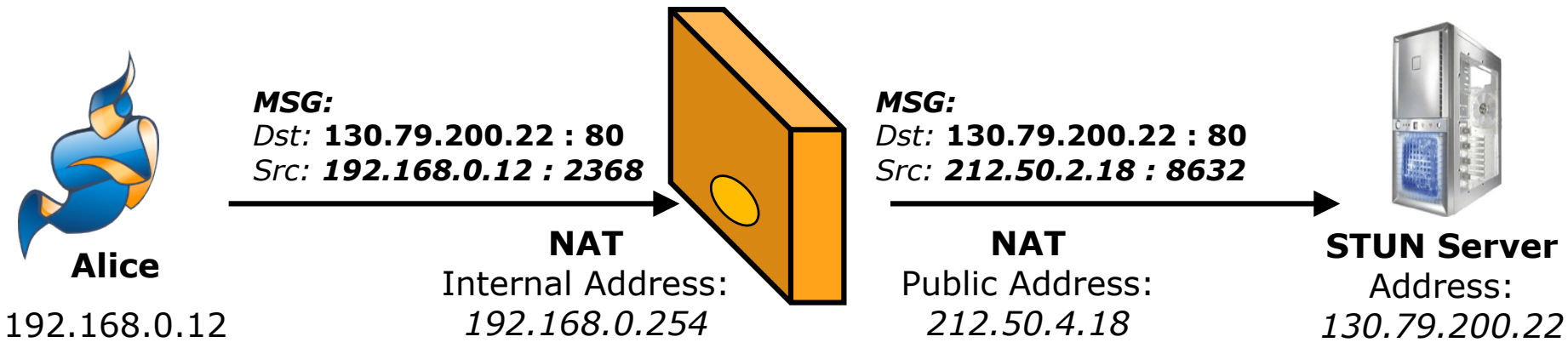
Internal **host:port**	NAT **port**	Active connections **host:port**
192.168.0.12 : 2368	***8632***	**130.79.200.22 (: 80)**
		60.55.68.53 (: 80)

Endpoint-Independent Mapping
Endpoint-Dependent Filtering



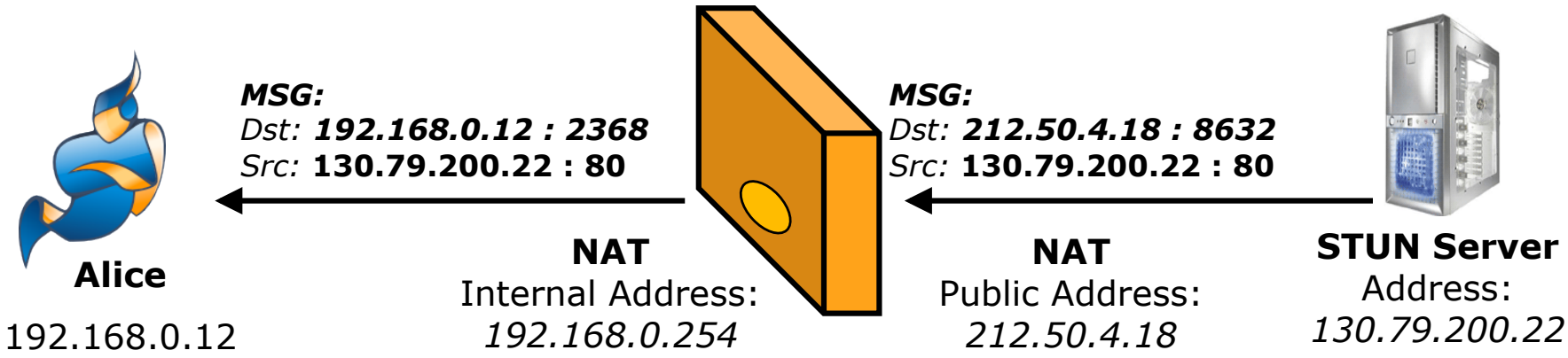
How do NATs work …
Endpoint dependent mapping

Internal **host:port**	NAT **port**	Active connections **host:port**
192.168.0.12 : 2368	***8632***	**130.79.200.22 (: 80)**



How do NATs work …
Endpoint dependent mapping

Internal **host:port**	NAT **port**	Active connections **host:port**
192.168.0.12 : 2368	***8632***	**130.79.200.22 (: 80)**



How do NATs work …
Endpoint dependent mapping

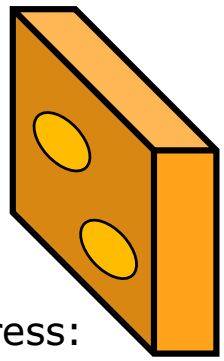
Internal **host:port**	NAT **port**	Active connections **host:port**
192.168.0.12 : 2368	***8632***	**130.79.200.22 (: 80)**
192.168.0.12 : 2368	***9391***	**60.55.68.53 (: 80)**

Bob
Address:
60.55.68.53

MSG:
Dst: **60.55.68.53 : 80**
Src: ***212.50.4.18 : 9391***

MSG:
Dst: **60.55.68.53 : 80**
Src: ***192.168.0.12 : 2368***

Alice
192.168.0.12



NAT
Internal Address:
192.168.0.254

NAT
Public Address:
212.50.4.18

STUN Server
Address:
130.79.200.22

How do NATs work …
Endpoint dependent mapping

Internal **host:port**	NAT **port**	Active connections **host:port**
192.168.0.12 : 2368	***8632***	**130.79.200.22 (: 80)**
192.168.0.12 : 2368	***9391***	**60.55.68.53 (: 80)**

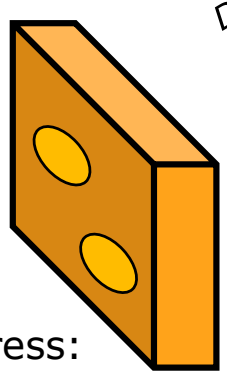
Endpoint-Dependent Mapping
Endpoint-Dependent Filtering

Bob
Address:
60.55.68.53

MSG:
Dst: ***212.50.4.18 : 8632***
Src: **60.55.68.53 : 80**

MSG:
Dst: ***192.168.0.12 : 2368***
Src: **60.55.68.53 : 80**

Alice
192.168.0.12



NAT
Internal Address:
192.168.0.254

NAT
Public Address:
212.50.4.18

STUN Server
Address:
130.79.200.22

Universal Plug and Play (UPnP)

www.upnp.org

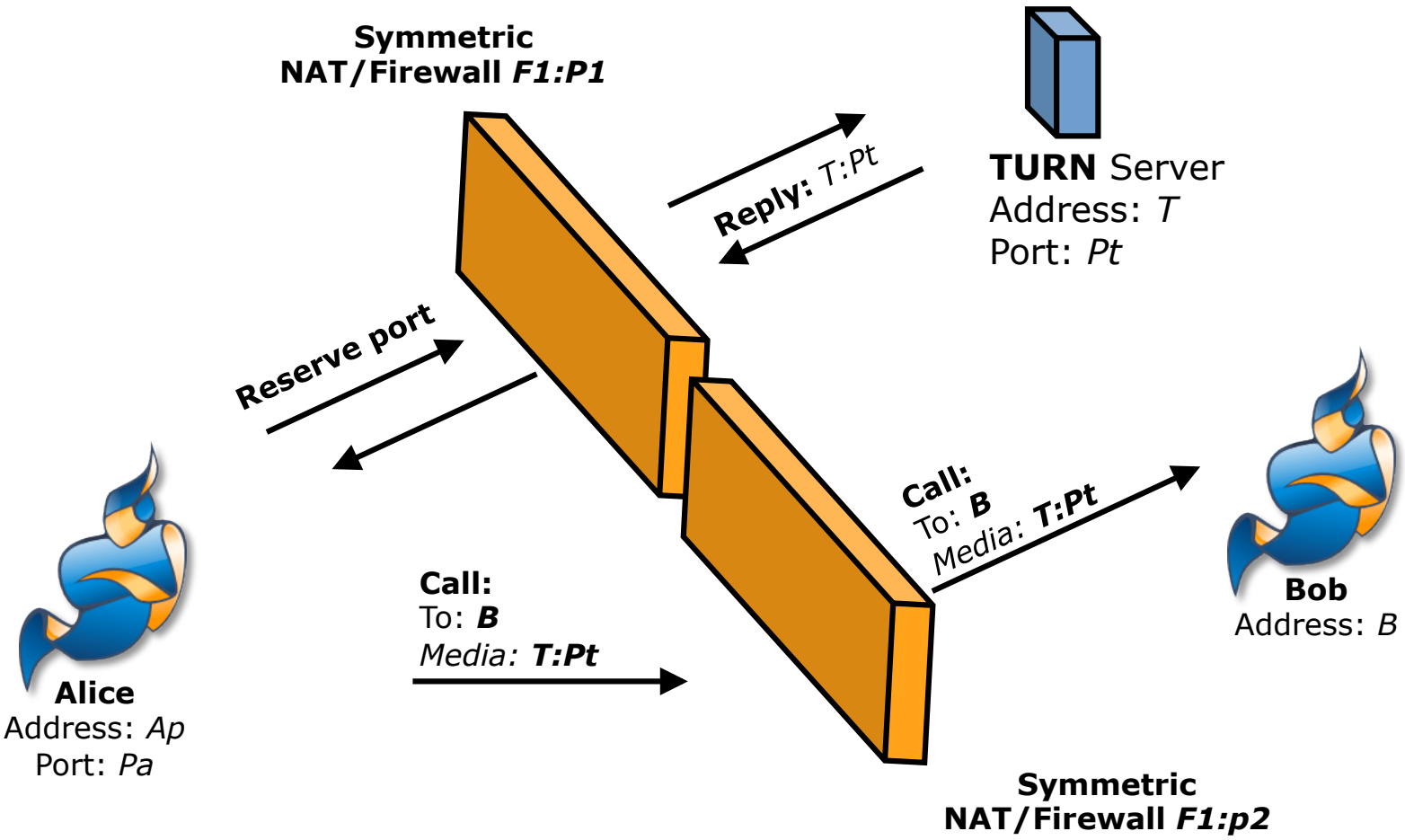
Designed for zero-configuration networking and to allow devices to:

* dynamically join a network and obtain an IP address
* announce its name
* advertise capabilities
* discover other devices and their capabilities
* Standardized as a 73-part International Standard, ISO/IEC 29341, in December, 2008.

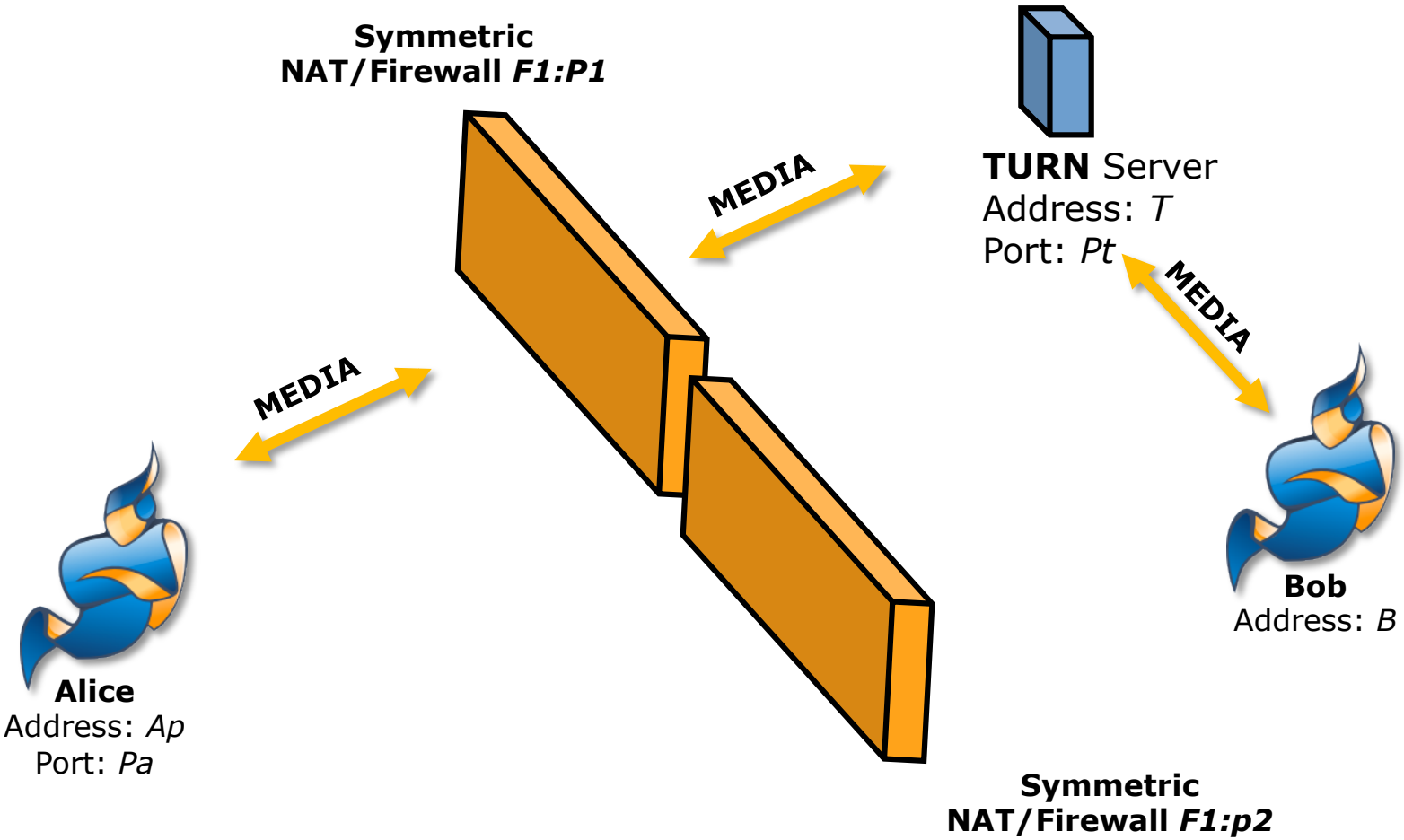
Makes it easy to:

* Learn the external (public) address of an internet gateway
* Enumerate existing port mappings
* Add and remove port mappings
* Assign lease times to mappings

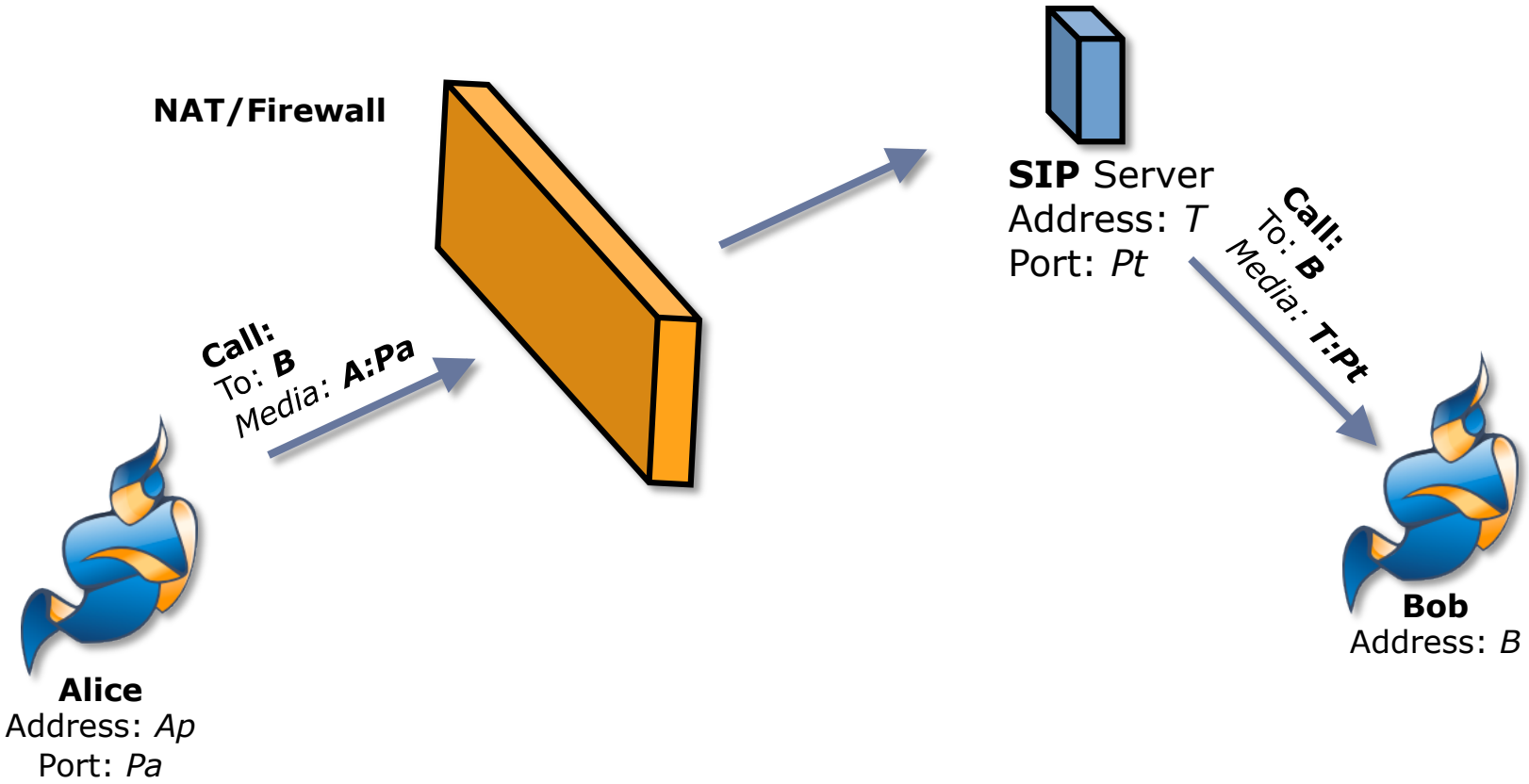
Relaying Media



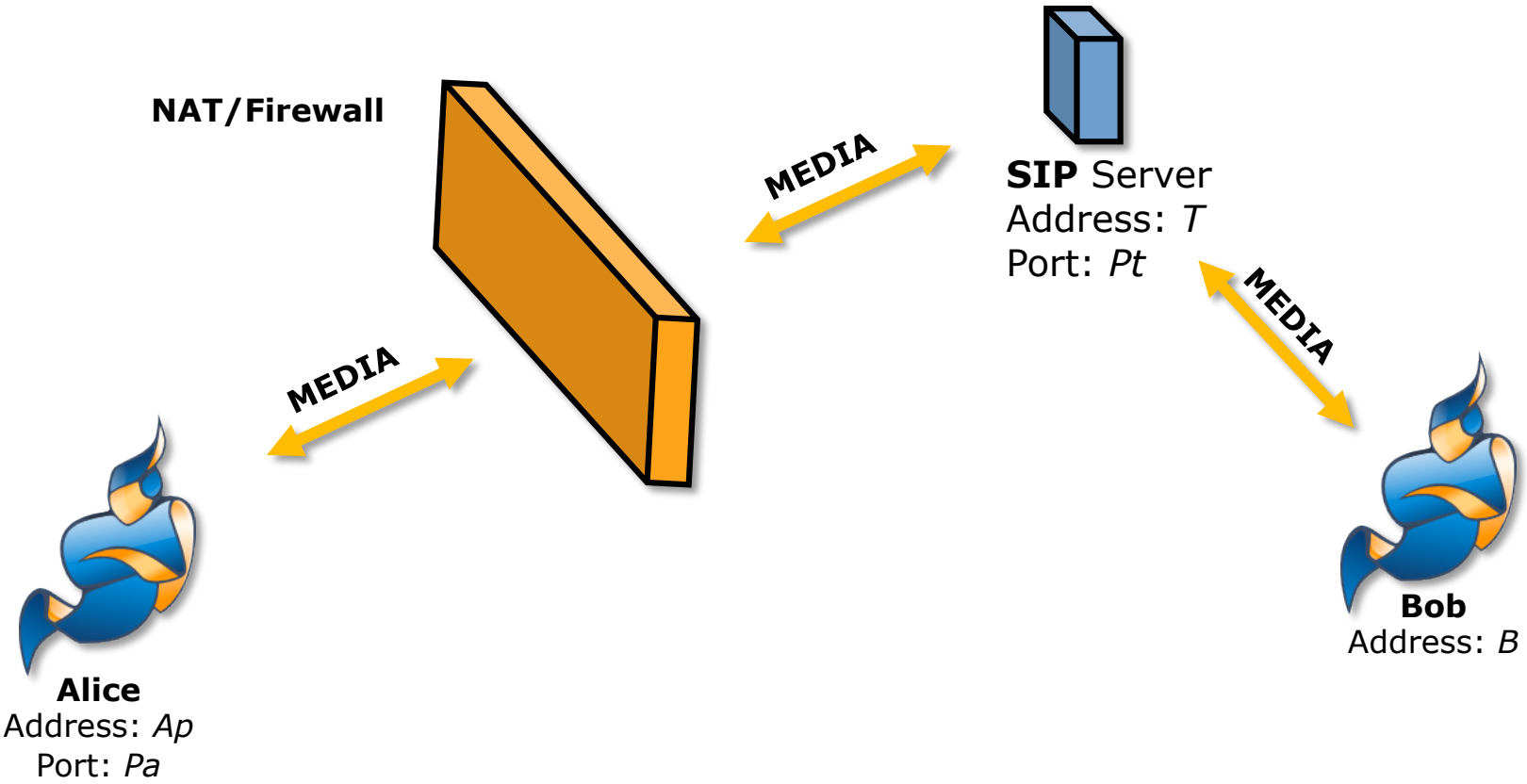
Relaying Media



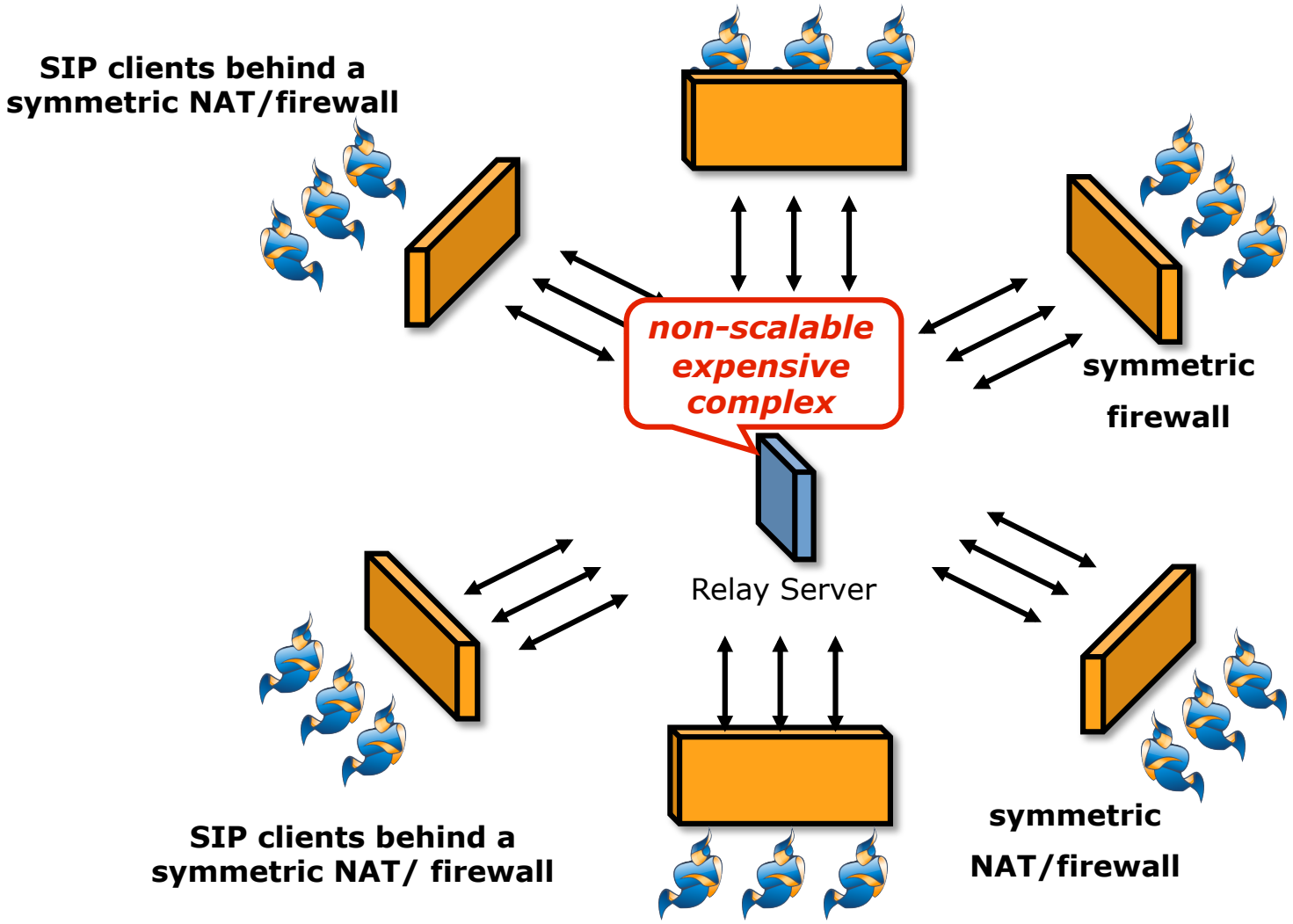
Relaying Media The SIP Way
Latching



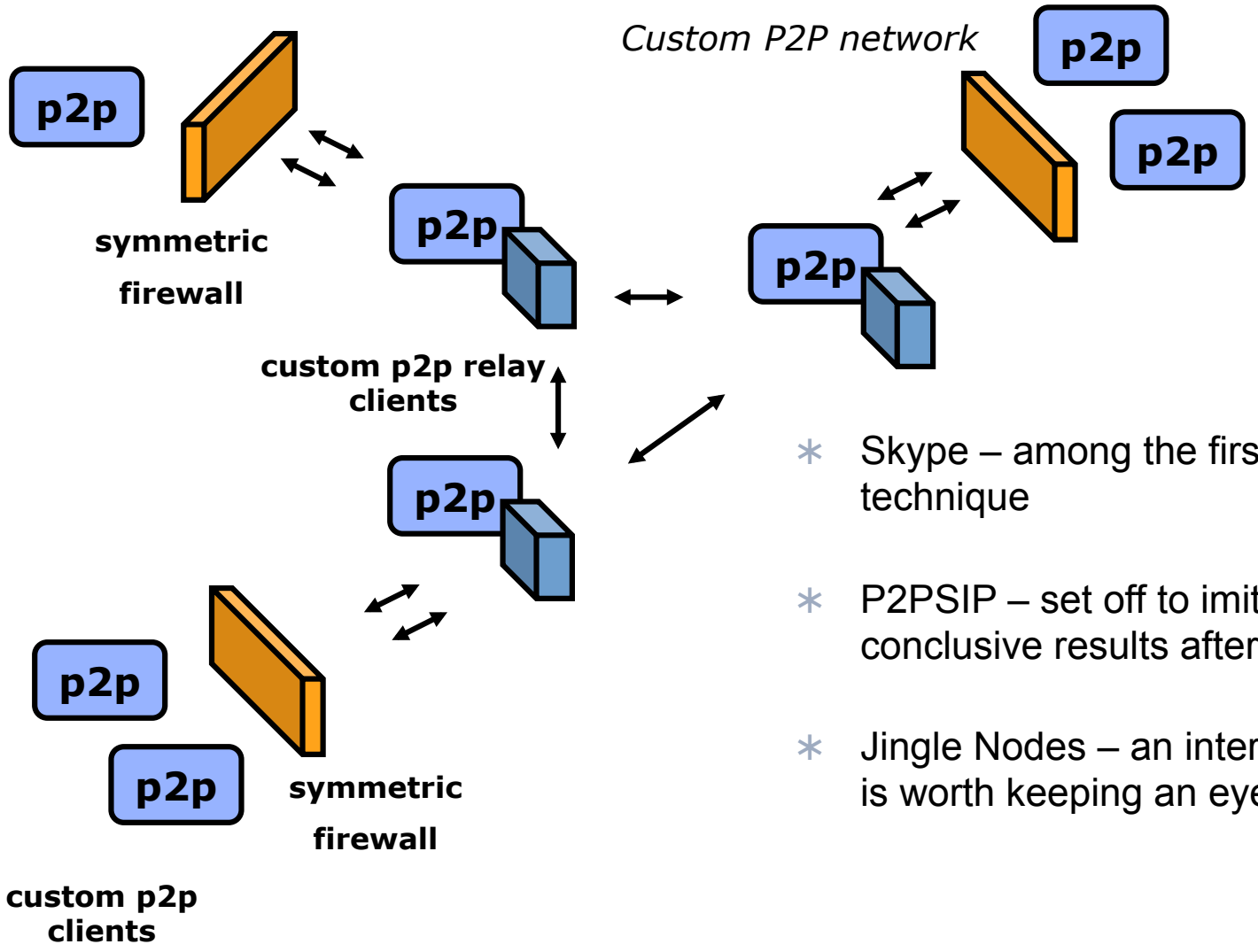
Relaying Media The SIP Way
Latching



Relaying Media



Using P2P networks for NAT Traversal



* Skype – among the first to implement the technique
* P2PSIP – set off to imitate Skype. No conclusive results after four years
* Jingle Nodes – an interesting alternative that is worth keeping an eye on

Could we please have IPv6 now?

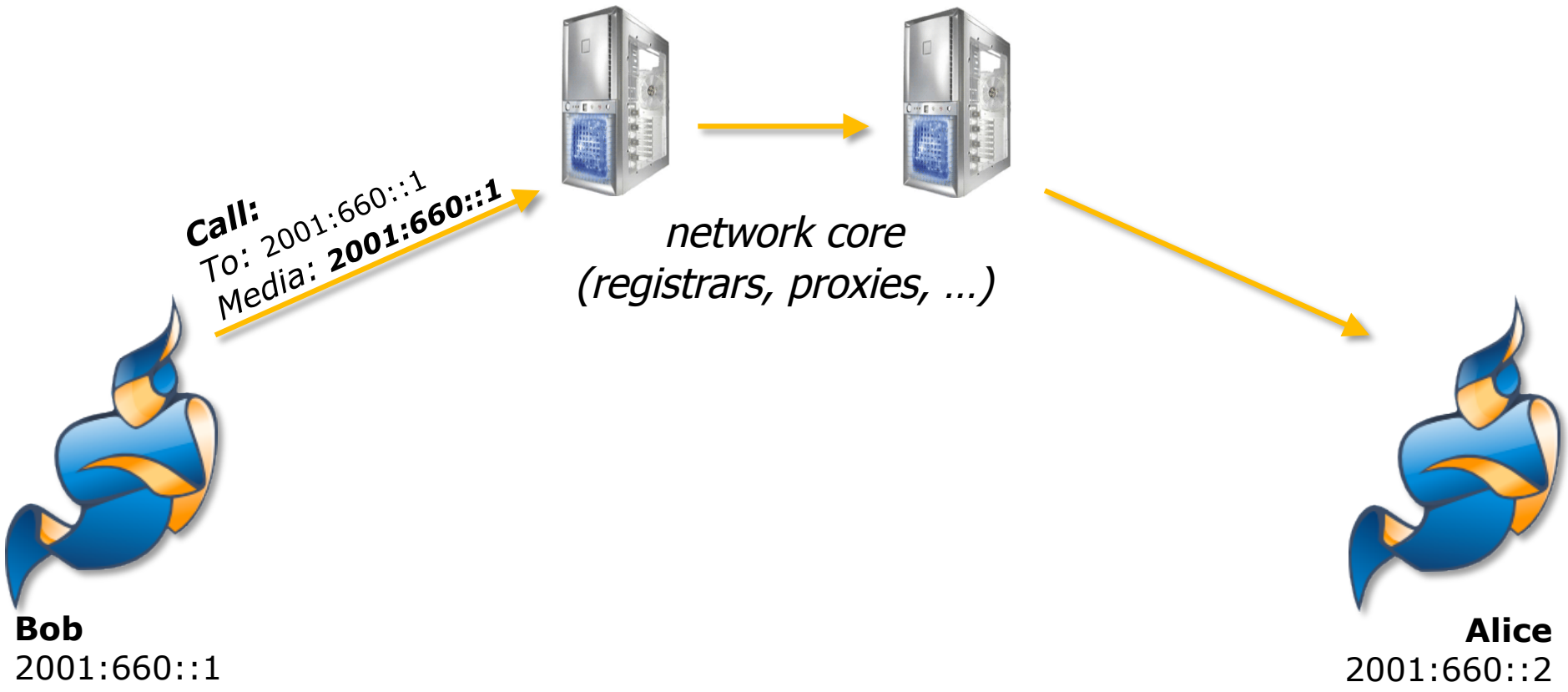
… ok, it's probably high time we moved to IPv6 …

Could we please have IPv6 now?

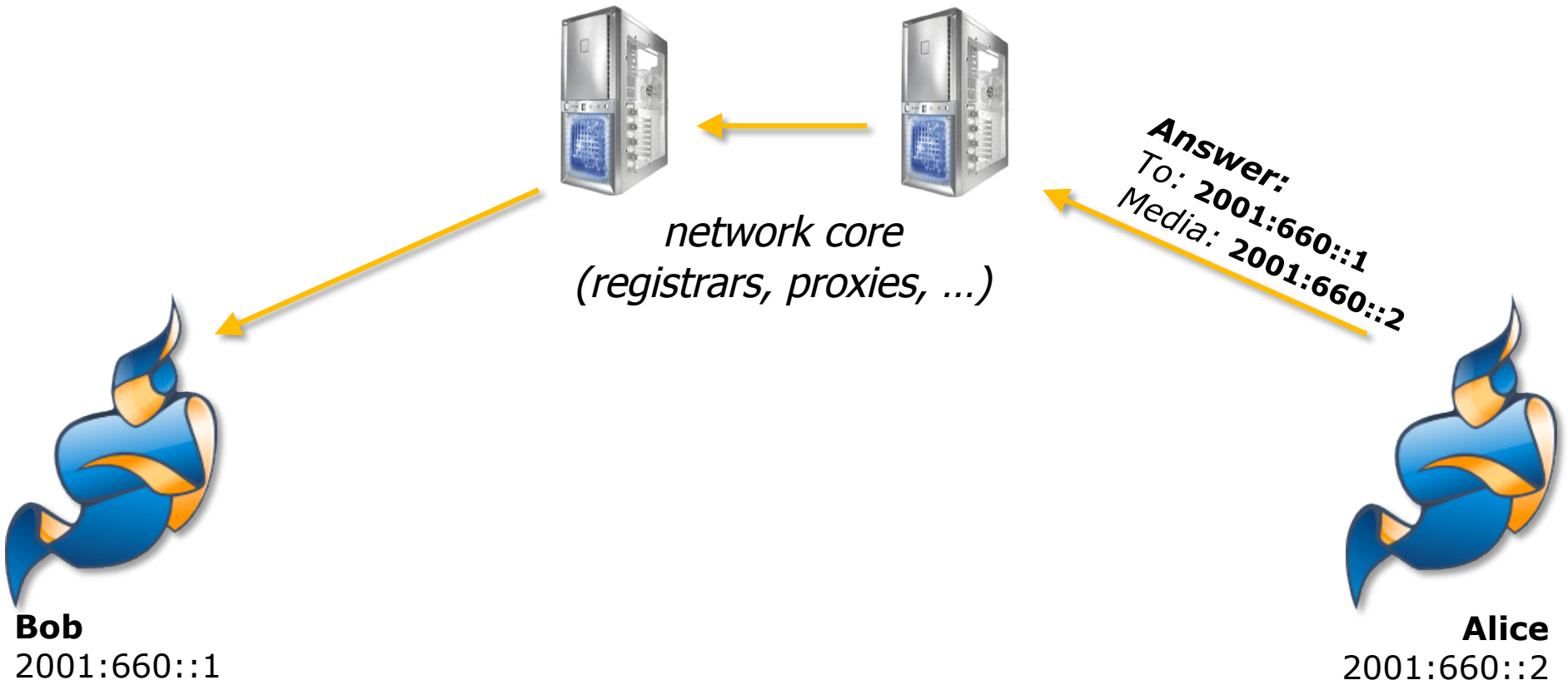
… this should simplify VoIP

… shouldn’t it?

VoIP and IPv6 – demo version

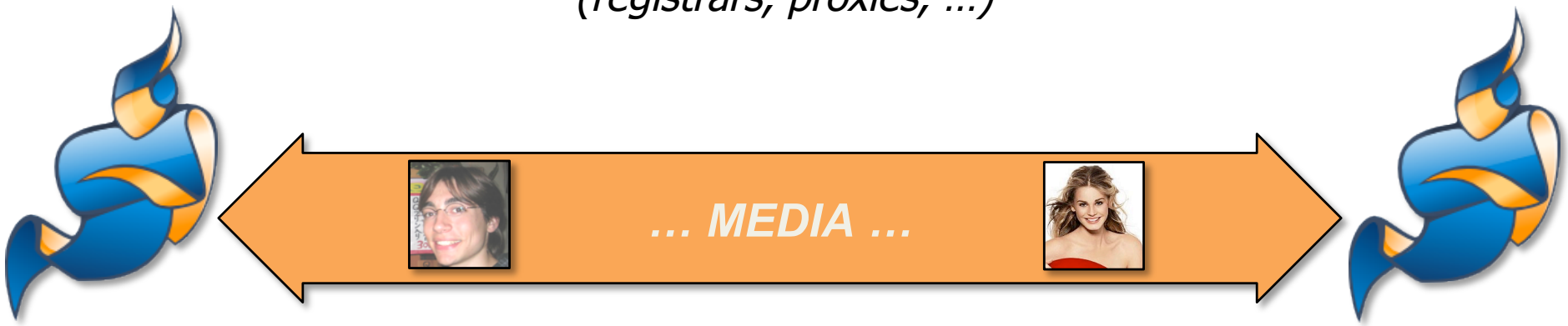


VoIP and IPv6 – demo version



VoIP and IPv6 – demo version

network core
(registrars, proxies, …)



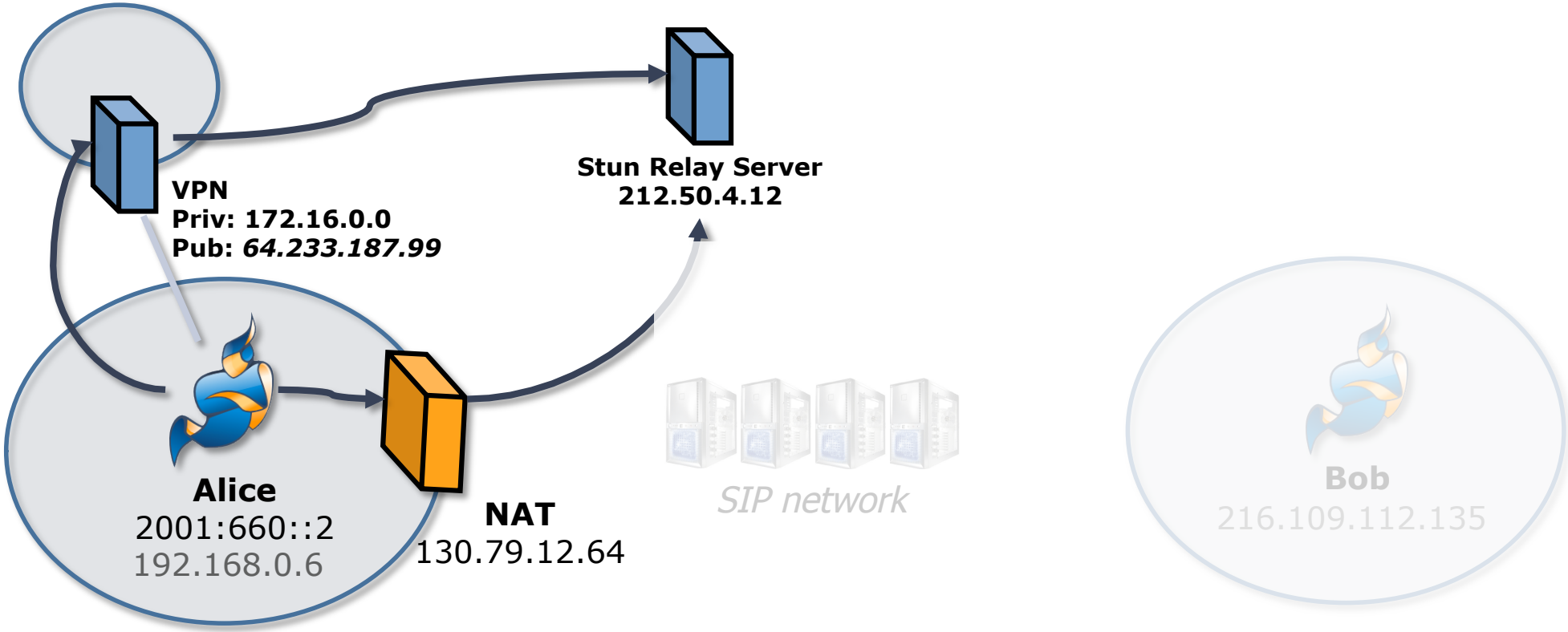
Bob
2001:660::1

Alice
2001:660::2

Reality check!

Reality check!

Reality check!



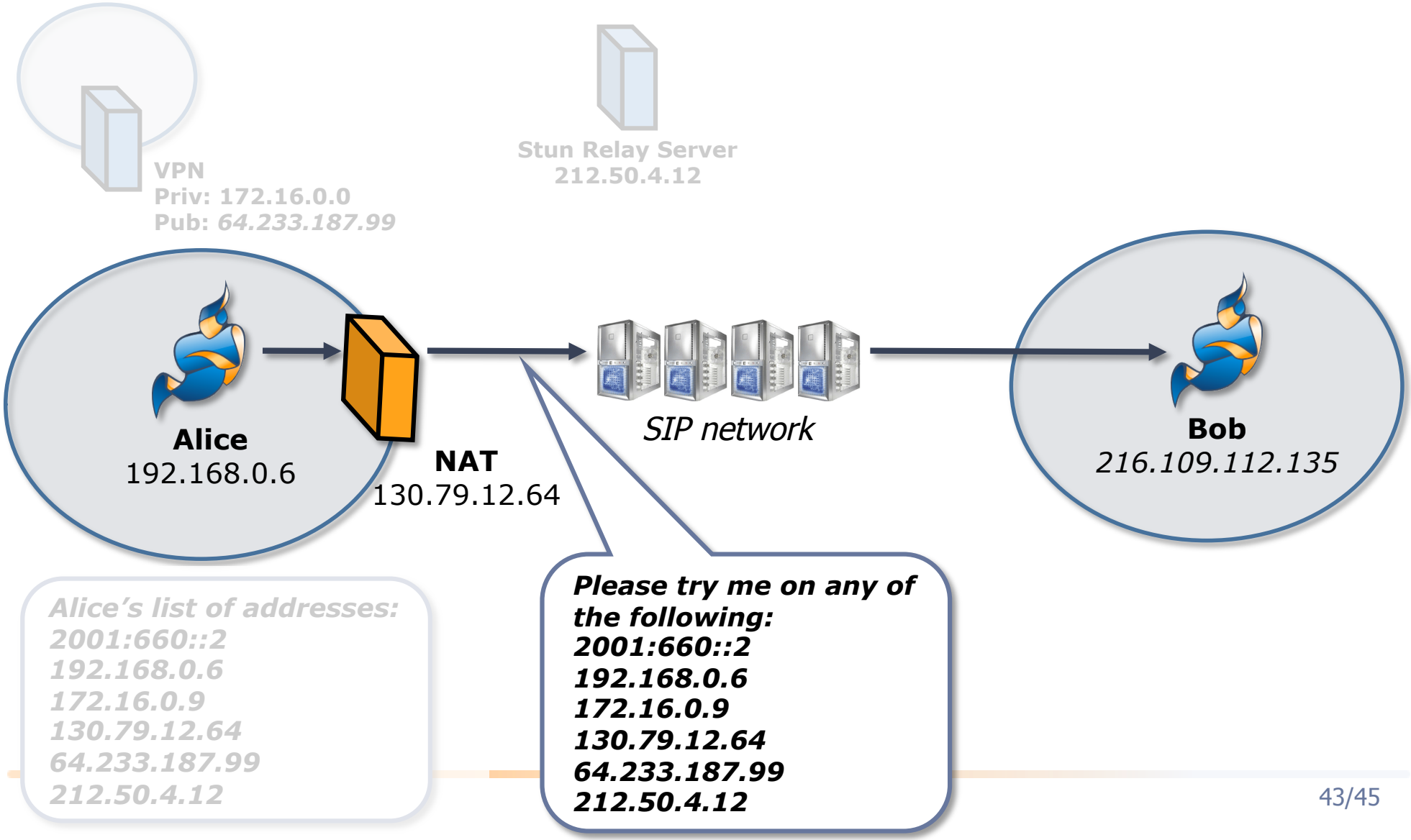
Alice's list of addresses:
2001:660::2
192.168.0.6
172.16.0.9
130.79.12.64
64.233.187.99
212.50.4.12

How to avoid relaying?

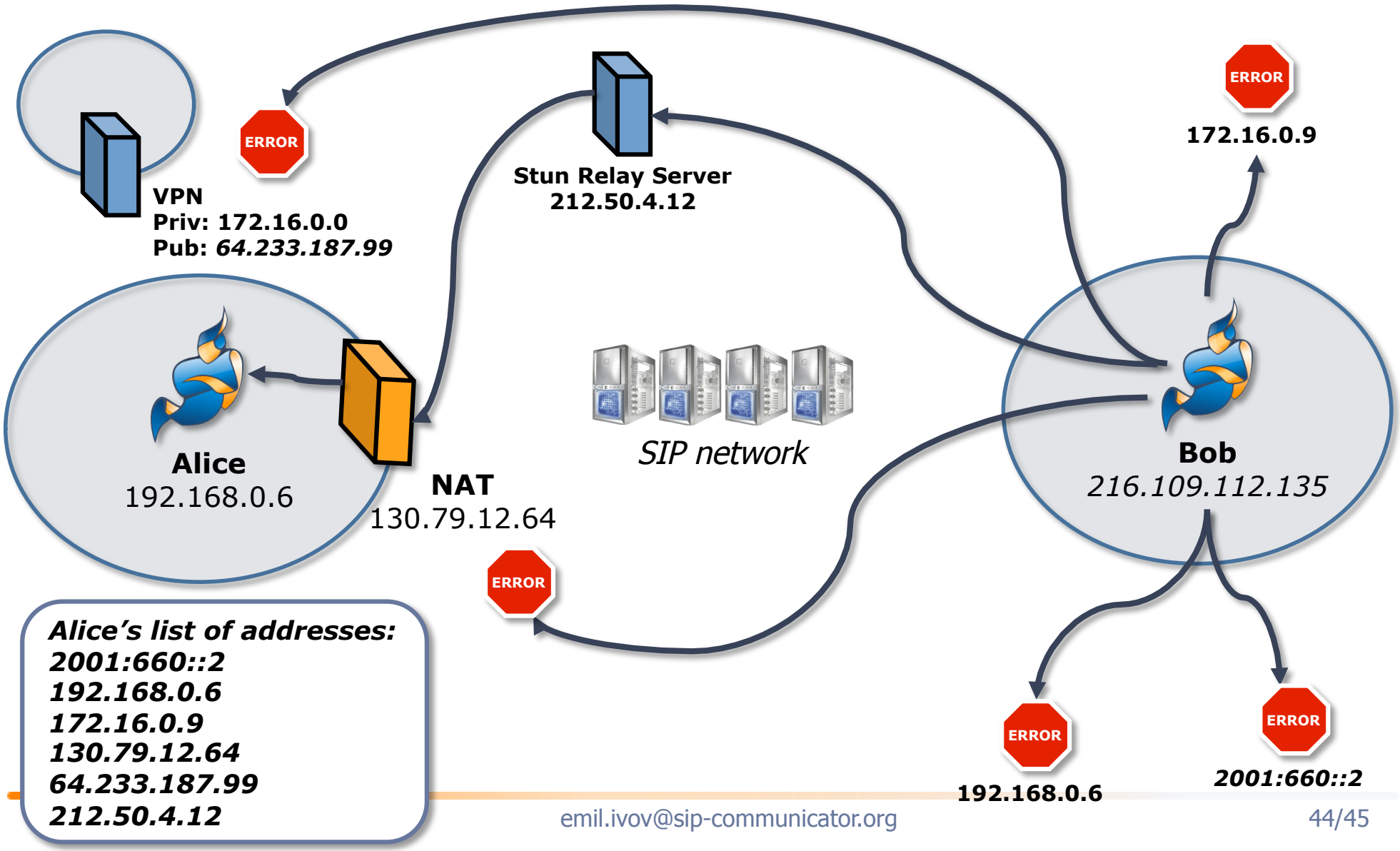
Interactive Connectivity Establishment (ICE)

An IETF RFC brought to you by Cisco's Jonathan Rosenberg

Address management with ICE



Address management with ICE



Real-time Communication 101

NAT and Firewall Traversal