How can application and desktop sharing, initiated by SIP, be realised in existing SIP infrastructure with the least possible impact on that infrastructure?
RFC 3261: The Session Initiation Protocol

- **User location**
  Wherever they are on the internet.
  Addresses are easy to remember.
- **User availability**
- **User capabilities**
- **Session setup**
- **Session management**
  For example: michiel@nlnet.nl
What is SIP?

RFC 3261: The Session Initiation Protocol

- User location
- User availability
- User capabilities
- Session setup
- Session management

Originally, “User not found”

Nowadays also presence information, like in instant messaging clients.
RFC 3261: The Session Initiation Protocol

- User location
- User availability
- User capabilities
- Session setup
- Session management

The session types supported:
- Voice
- Video
- Instant Messaging
- Desktop sharing
What is SIP?

RFC 3261: The Session Initiation Protocol

- User location
- User availability
- User capabilities
- Session setup
- Session management

- Calling
- Redirections

Willem Toorop (willem.toorop@os3.nl)
RFC 3261: The Session Initiation Protocol

- User location
- User availability
- User capabilities
- Session setup
- Session management
- Transfers
- Hangups
What is SIP?

RFC 3261: The Session Initiation Protocol

- User location
- User availability
- User capabilities
- Session setup
- Session management

SIP does not do the session itself!
Why desktop sharing with SIP?

- No host names or IP-addresses to remember or find out about
Why desktop sharing with SIP?

- No host names or IP-addresses to remember or find out about
- No VPN’s to private networks needed
Why desktop sharing with SIP?

- No host names or IP-addresses to remember or find out about
- No VPN’s to private networks needed
- Simply call your problem solver and offer your desktop
How does SIP work?

Alice's User Agent

1. INVITE sip:bob@example.com
   Contact: Alice

2. SIP/2.0 100 Trying

3. SIP/2.0 180 Ringing
   Contact: Bob

4. SIP/2.0 200 OK
   Contact: Bob

Proxy for example.com

Bobs User Agent

INVITE sip:bob@example.com
Contact: Proxy

SIP/2.0 180 Ringing
Contact: Bob

SIP/2.0 200 OK
Contact: Bob

ACK Bob

ACK Bob

Media session

RTP over UDP
The NAT-Traversal problem

Client 10.0.0.8 → NAT Binding table 10.0.0.8:2345 → 74.125.79.104:80

NAT Binding table

<table>
<thead>
<tr>
<th>Internal</th>
<th>External</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.0.0.8:2345</td>
<td>192.0.0.1:6789</td>
</tr>
</tbody>
</table>

NAT 192.0.0.1:6789 → 74.125.79.104:80

Host 74.125.79.104
How does SIP deal with it?

Alice's User Agent \(\rightarrow\) NAT \(\rightarrow\) Alice's Outbound Proxy \(\rightarrow\) Proxy for example.com \(\rightarrow\) Bobs User Agent

INVITE alice@example.com

100 Trying

180 Ringing

200 OK

ACK

Media session??? How to connect?
Industry solutions

Application-level Gateway
Industry solutions

Application-level Gateway

Session Border Controller
Full cone NAT

The NAT-Traversal problem

The IETF answer

Willem Toorop (willem.toorop@os3.nl)

Desktop sharing with SIP

February 25, 2009
Address restricted cone NAT

<table>
<thead>
<tr>
<th>Internal</th>
<th>External</th>
<th>Servers</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.0.0.8:2345</td>
<td>192.0.0.1:6789</td>
<td>74.125.79.104</td>
</tr>
<tr>
<td></td>
<td></td>
<td>145.100.96.70</td>
</tr>
</tbody>
</table>
Port restricted cone NAT

<table>
<thead>
<tr>
<th>Internal</th>
<th>External</th>
<th>Server &amp; port</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.0.0.8:2345</td>
<td>192.0.0.1:6789</td>
<td>74.125.79.104:80</td>
</tr>
<tr>
<td></td>
<td></td>
<td>145.100.96.70:80</td>
</tr>
</tbody>
</table>

Willem Toorop (willem.toorop@os3.nl)
Symmetric NAT

```
<table>
<thead>
<tr>
<th>Internal from</th>
<th>External to</th>
<th>External from</th>
</tr>
</thead>
<tbody>
<tr>
<td>10.0.0.8:2345</td>
<td>74.125.79.104:80</td>
<td>192.0.0.1:6789</td>
</tr>
<tr>
<td>10.0.0.8:2345</td>
<td>145.100.96.70:80</td>
<td>192.0.0.1:5555</td>
</tr>
</tbody>
</table>
```
STUN & TURN

The NAT-Traversal problem

The IETF answer

Client → NAT ← STUN Server

What IP:port do you see?

IP-address 1

IP-address 2

Media session

Media session

NAT

NAT

Willem Toorop (willem.toorop@os3.nl)

Desktop sharing with SIP

February 25, 2009 12 / 21
STUN & TURN

The NAT-Traversal problem

The IETF answer

Client

Reply from other IP please →

NAT

→ Sure!

STUN Server

IP-address 1

IP-address 2

Client
STUN & TURN

The NAT-Traversal problem

Client

NAT

Reply from other IP please

→

Sure!

STUN Server

IP-address 1

IP-address 2

Client

Media session

NAT

TURN Server

IP-address 1

IP-address 2

NAT

Media session

Client

Desktop sharing with SIP

February 25, 2009
ICE & ICE-TCP

draft-ietf-mmusic-ice-19: Interactive Connectivity Establishment

Defines a procedure for SIP User Agents to get the best connection.

Uses STUN for discovery and TURN as a last resort solution.

But...
ICE & ICE-TCP

draft-ietf-mmusic-ice-19: Interactive Connectivity Establishment

Defines a procedure for SIP User Agents to get the best connection.

Uses STUN for discovery and TURN as a last resort solution.

But...

- It is still a draft
ICE & ICE-TCP

draft-ietf-mmusic-ice-19: Interactive Connectivity Establishment
Defines a procedure for SIP User Agents to get the best connection.
Uses STUN for discovery and TURN as a last resort solution.

But...
- It is still a draft
- SBCs work well
Media specific solutions: MSRP

- For instant messaging
Media specific solutions: MSRP

- For instant messaging
- Instant messaging is popular!
Media specific solutions: MSRP

- For instant messaging
- + Instant messaging is popular!
- + Has TCP as the underlying transport
### MSRP Messages

**Alice →**

<table>
<thead>
<tr>
<th>Message-ID</th>
<th>Content-Type</th>
<th>From</th>
<th>To</th>
<th>Content-Type</th>
<th>Message Body</th>
</tr>
</thead>
<tbody>
<tr>
<td>QZ3ts6C3Ed</td>
<td>message/cpim</td>
<td>Alice</td>
<td>Bob</td>
<td>text/plain</td>
<td>Hi Bob</td>
</tr>
<tr>
<td>BczlzlN3Vf</td>
<td>message/cpim</td>
<td>Bob</td>
<td>Alice</td>
<td>text/plain</td>
<td>Hi Alice</td>
</tr>
</tbody>
</table>

**← Bob**

<table>
<thead>
<tr>
<th>Message-ID</th>
<th>Content-Type</th>
<th>From</th>
<th>To</th>
<th>Content-Type</th>
<th>Message Body</th>
</tr>
</thead>
<tbody>
<tr>
<td>QZ3ts6C3Ed</td>
<td>message/cpim</td>
<td>Alice</td>
<td>Bob</td>
<td>text/plain</td>
<td>Hi Bob</td>
</tr>
<tr>
<td>BczlzlN3Vf</td>
<td>message/cpim</td>
<td>Bob</td>
<td>Alice</td>
<td>text/plain</td>
<td>Hi Alice</td>
</tr>
</tbody>
</table>

Content-Type is agreed upon by SIP (User Capabilities)
RFB over MSRP!

vncviewer →

MSRP ydD6J6w SEND
Byte-Range: 1-10/10
Message-ID: QZ3ts6C3Ed
Content-Type: application/x-rfb

RFB data
-------ydD6J6w$

MSRP t4gk7Sv 200 OK
-------t4gk7Sv$

← vncserver

MSRP ydD6J6w 200 OK
-------ydD6J6w$

MSRP t4gk7Sv SEND
Message-ID: Bcz1lz1N3Vf
Content-Type: application/x-rfb

RFB data
-------t4gk7Sv$
SIP SIMPLE LIBRARY

A Python based library

- It does SIP

The solution:

Willem Toorop (willem.toorop@os3.nl)
SIP SIMPLE LIBRARY

A Python based library

- It does SIP
- It does MSRP

The solution:
A Python based library

- It does SIP
- It does MSRP

Ends in a MSRPSession object for reading and writing

The solution:
A Python based library

- It does SIP
- It does MSRP

Ends in a MSRPSession object for reading and writing

The solution:

- Copy data from object to vnc software
- Copy data from vnc software to object

But how to connect...
- vncserver is already running
Connecting endpoint

- vncserver is already running
- others can connect too
Connecting endpoint

- vncserver is already running
- others can connect too
- password protection
Listening endpoint

+ No such problems here
Reverse VNC connection

- No others that can connect to the server
Implemented solution

Reverse VNC connection
- + No others that can connect to the server
- + No password is required
Implemented solution

Python based vncviewer

- + No programs to start
Implemented solution

Python based vncviewer

- + No programs to start
- + Encapsulation in Python based GUI
Implemented solution

Python based vncviewer

- No programs to start
- Encapsulation in Python based GUI
- But a real viewer might be nicer
Conclusion

How can application and desktop sharing, initiated by SIP, be realised in existing SIP infrastructure with the least possible impact on that infrastructure?

draft-boyaci-avt-app-sharing-00:
RTP Payload format for Application and Desktop Sharing

- Operates over RTP over UDP
- Will probably work on existing infrastructure (SBCs)
Conclusion

How can application and desktop sharing, initiated by SIP, be realised in existing SIP infrastructure with the least possible impact on that infrastructure?

draft-boyaci-avt-app-sharing-00:
RTP Payload format for Application and Desktop Sharing

- Operates over RTP over UDP
  Will probably work on existing infrastructure (SBCs)
- Defines a new application sharing protocol
  (Is it worth the effort?)
Conclusion

How can application and desktop sharing, initiated by SIP, be realised in existing SIP infrastructure with the least possible impact on that infrastructure?

draft-boyaci-avt-app-sharing-00: RTP Payload format for Application and Desktop Sharing

- + Operates over RTP over UDP
  Will probably work on existing infrastructure (SBCs)
- - Defines a new application sharing protocol
  (Is it worth the effort?)

RFB over MSRP

- + Instant messaging is popular (Added value to offer costumers)
Conclusion

How can application and desktop sharing, initiated by SIP, be realised in existing SIP infrastructure with the least possible impact on that infrastructure?

draft-boyaci-avt-app-sharing-00:
RTP Payload format for Application and Desktop Sharing

- Operates over RTP over UDP
  Will probably work on existing infrastructure (SBCs)
- Defines a new application sharing protocol
  (Is it worth the effort?)

RFB over MSRP

- Instant messaging is popular (Added value to offer costumers)
- Uses existing public RFB standard