

Respoke 101

David M. Lee, II

[@leedm777](#)

Lead Devops for Respoke

October 13, 2015



Agenda

- Overview
- Getting Started
- Overview of major APIs
- Walk through a simple example
- Gotchas
- Q&A

[https://github.com/
leedm777/respoke101](https://github.com/leedm777/respoke101)

What is Respoke?

What is Respoke?

Respoke is a web communications platform. It provides messaging, voice and video services for web and mobile applications.

Respoke Features

- Peer to peer messaging
- Group messaging
- WebRTC voice, video and data channel
- Mobile push notifications
- Asterisk integration (chan_respoke)
- PSTN Integration
- Web, Android and iOS client SDKs
- Server side SDKs for Node.js, Ruby, .NET, Java, and more

Respoke Features

- Peer to peer messaging
- Group messaging
- WebRTC voice, video and data channel
- Mobile push notifications
- Asterisk integration (chan_respoke)
- PSTN Integration
- Web, Android and iOS client SDKs
- Server side SDKs for Node.js, Ruby, .NET, Java, and more
- Early Access Features:
 - SIP integration
 - Multi-party conferencing
- Contact support@respoke.io if you are interested in these features.

Getting Started

- <https://www.respoke.io> - links, blog, pricing
- <https://docs.respoke.io> - API docs, examples, tutorials
- <https://portal.respoke.io> - Respoke management portal

Respoke.js API patterns

Respoke.js API patterns

// All async APIs support both callbacks...

```
client.doSomething({  
  onSuccess: function() { /* success */ },  
  onError: function(err) { /* failure */ }  
});
```

// ... and promises

```
client.doSomething()  
  .then(function() {  
    /* success */  
  }).catch(function(err) {  
    /* failure */  
  });
```

Respoke.js API patterns

```
// Event listeners can be given as params
client.createSomething({
  onSomeEvent: function(evt) {
    /* some event handler */
  }
});
```

```
// ... or you can listen later
var something = client.createSomething();

something.listen('some-event', function(evt) {
  /* some event handler */
});
```

Respoke.js API patterns

- Respoke.js supports many styles of front-end development
- In my examples, I'll just stick to one style, and only show the most important options in the API calls
- For detailed information, check the API docs at <https://docs.respoke.io/js-library/respoke.html>

Respoke Applications

- In order to connect your application to Respoke, you will create an *application* via the portal
- The application is identified by its *app-id*
- To securely generate tokens, you will also need the *app-secret*
- Endpoints can only communicate with other endpoints in the same application

To the web!!!

Client Authentication Dev Mode

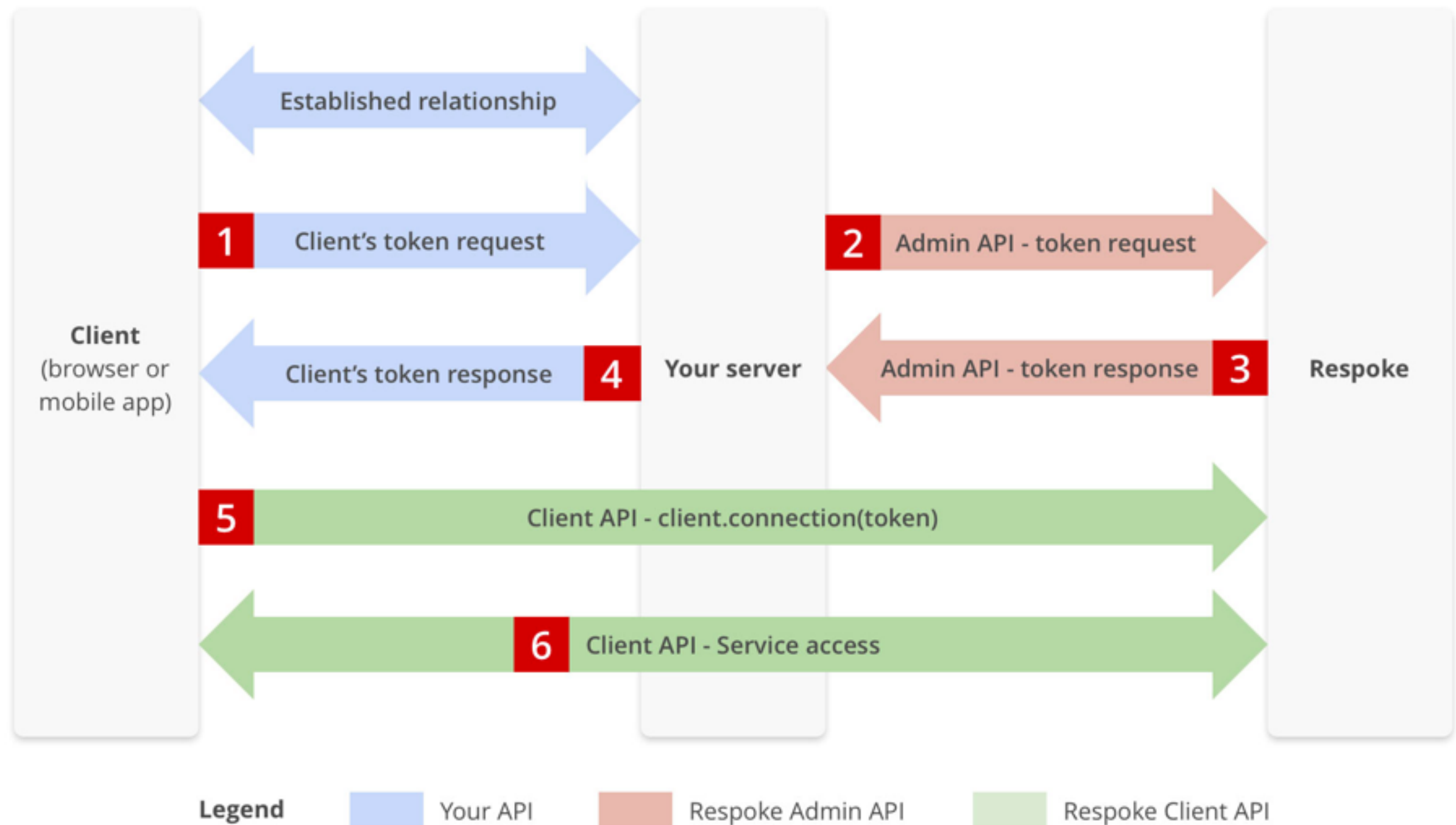
- Enabled at the application level
- Simpler connection process, no server needed
- Permissions default to permissive instead of restrictive
- No credentials needed for connecting: INSECURE!
- Great for playing around with Respoke

Client Authentication

Brokered Auth

- Required when dev mode is disabled
- Restrictive permissions by default, you will need to define permissions for use with your application
- Server required for authentication

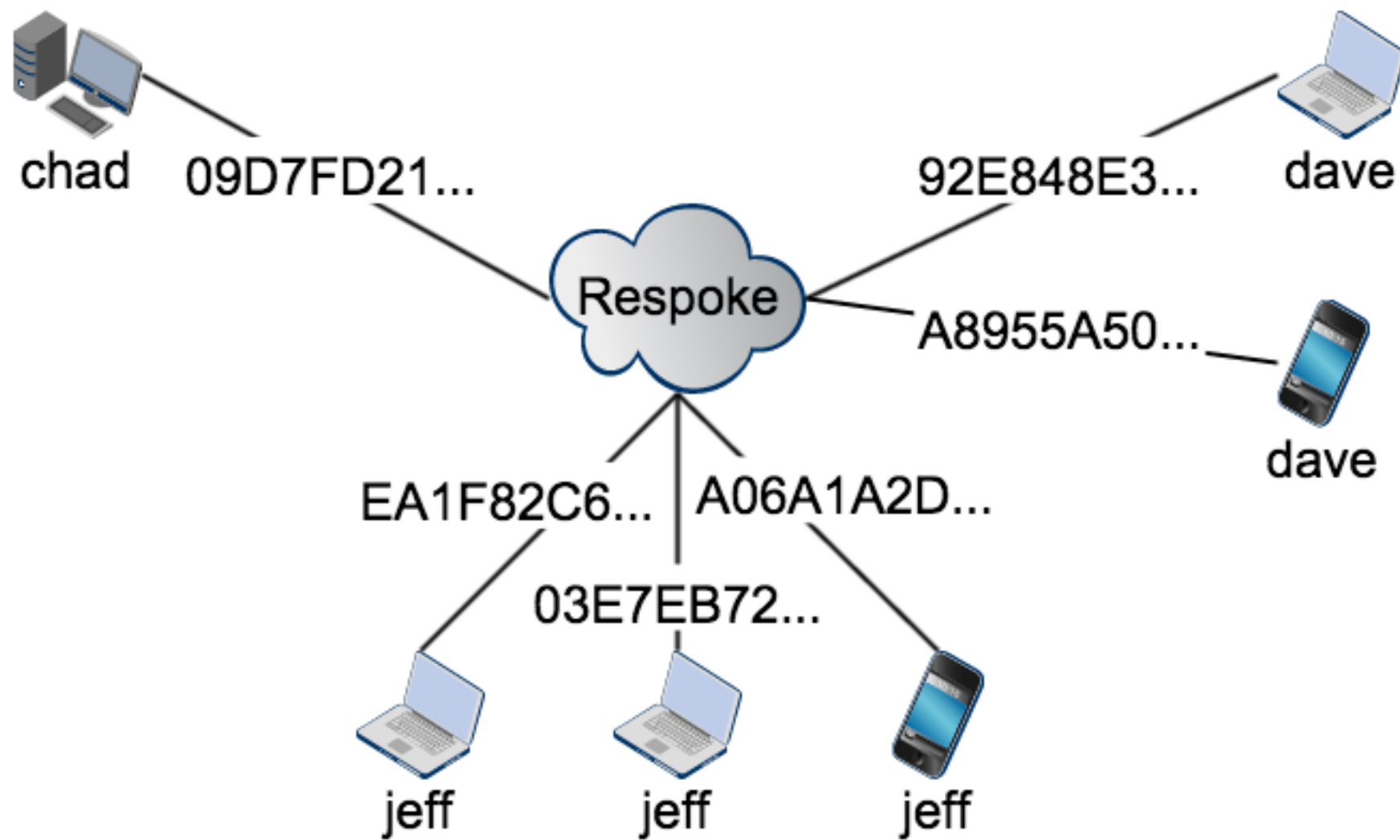
Why brokered auth?



Endpoints and Connections

- When a client connects to Respoke, you supply an *endpoint-id*
- That connection is assigned a unique *connection-id*

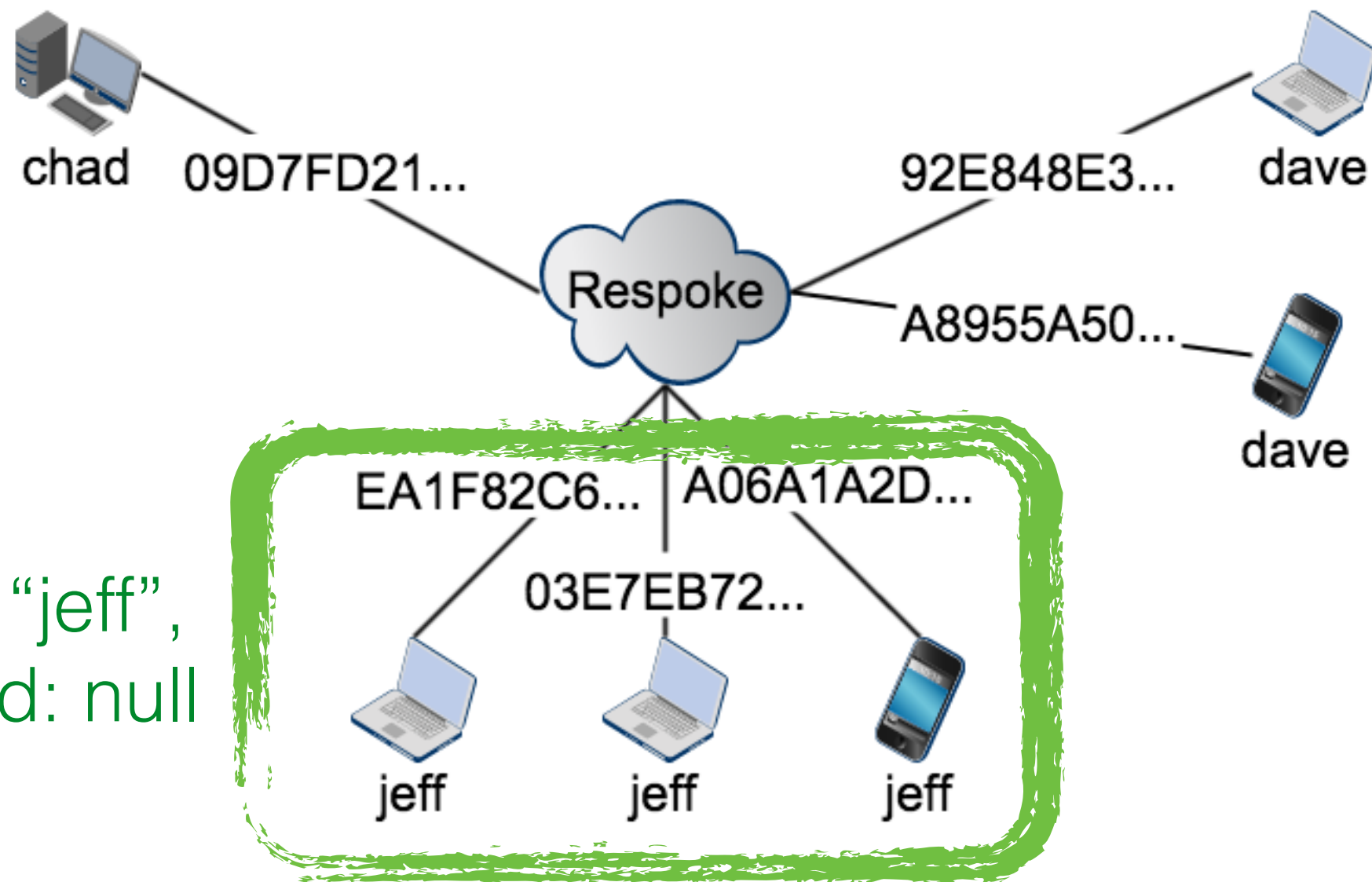
Endpoints and Connections



Sending Messages

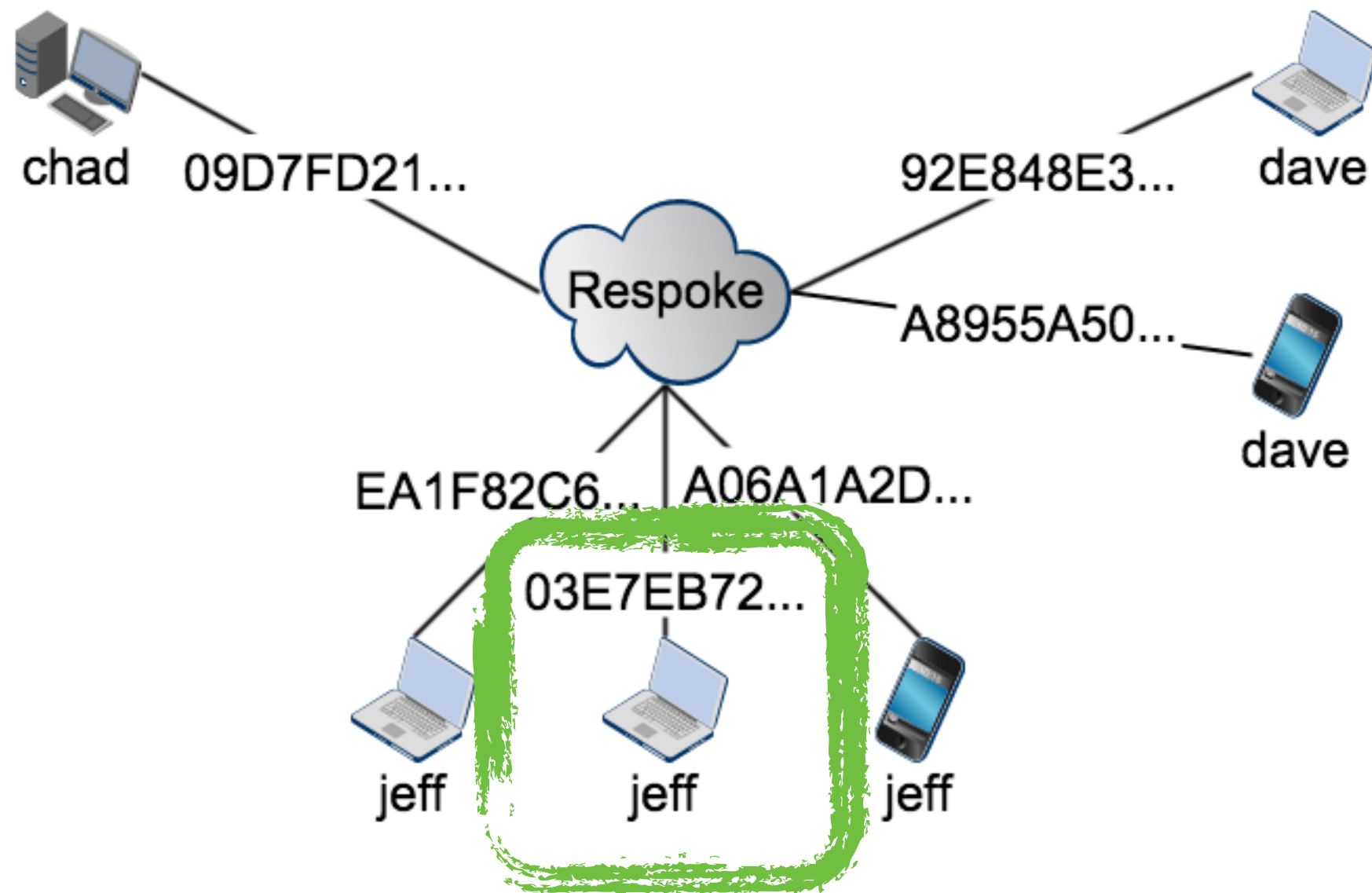
```
client.sendMessage({  
  endpointId: '...',  
  connectionId: '...', // optional  
  message: '...',  
});
```

Endpoints and Connections



endpointId: "jeff",
connectionId: null

Endpoints and Connections



endpointId: "jeff",
connectionId: "03E7EB72..."

Group Messaging

- Connections may join/leave groups
- Messages sent to a group are forwarded to all members of that group
- Permissions can be used to fine tune who can send/receive messages, inspect group members,

Group Messaging

```
client.join({  
  id: '...',  
});
```

```
group.sendMessage({  
  message: '...',  
});
```


Voice and Video

```
client.startAudioCall({  
  endpointId: '...',  
  // ...  
});
```

```
client.startVideoCall({  
  endpointId: '...',  
  // ...  
});
```

Asterisk Integration

- chan_respoke: https://github.com/respoke/chan_respoke
- Once Asterisk is connected with Respoke, you can call it just like you would any other endpoint

Examples!!!

Actual code

Other Cool Stuff

That I couldn't fit into the training

Screen Sharing

- Requires browser extension/plug in

```
endpoint.startScreenShare({  
  // ...  
});
```

PSTN Integration

- Contact support@respoke.io to enable

```
client.startPhoneCall({  
  number: "+12564286254",  
  callerId: "+15555555555"  
});
```

Mobile Push Notifications

- Requires some setup for your mobile app push credentials

```
client.sendMessage({  
  // ...  
  push: true  
});
```

Direct Messaging

- Uses WebRTC Data Channel for secured peer-to-peer messaging.

```
endpoint.startDirectConnection();
```

```
directConnection.send({  
  message: { /* POJO */ }  
});
```


Multi-party Audio Conferencing

- Early access feature; contact support@respoke.io to get in on the beta

```
client.joinConference({  
  id: '...',  
  // ...  
});
```

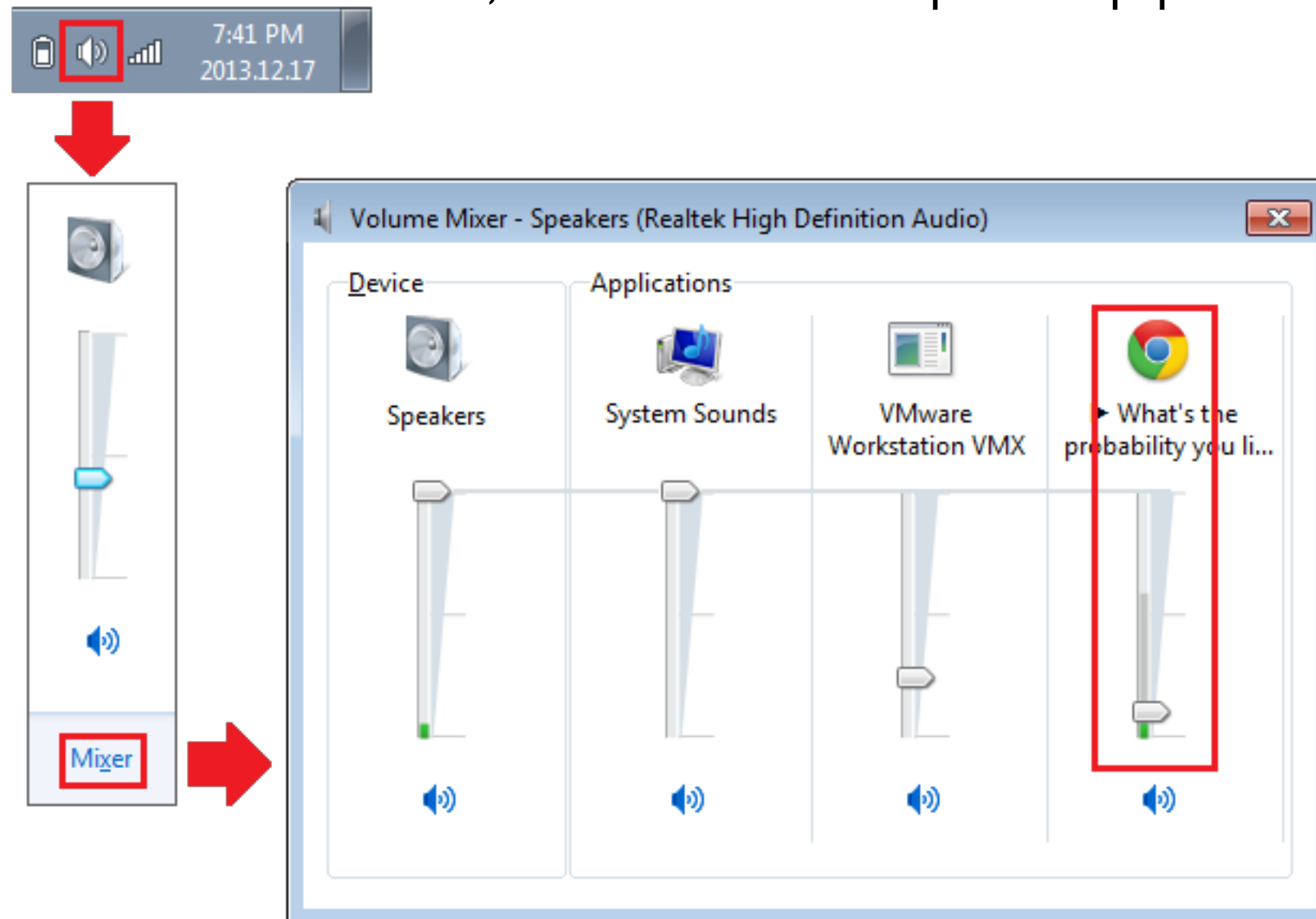
Gotchas

No audio?

- Check your volume settings.
- Check which input devices the browser is using.
- Add a simple sound file to you app.
 - Even just a **ring.wav** that you play before calls.
- If you can hear the sound play, then most of the simple causes for no audio problems have been eliminated.

No audio?

- Check the volume settings
- And on Windows, this can be per-app



No audio?

- When using chan_respoke
- Collect Asterisk debug information: <https://wiki.asterisk.org/wiki/display/AST/Collecting+Debug+Information>

No audio or video?

- Check your Content-Security-Policy, if you are using one
- Behavior changed in Chrome 45, to no longer include **blob:** URLs by default
- To fix, add `media-src blob: 'self';` to your CSP
- Not a concern if you haven't enabled CSP

No audio or video?

- CSP header:

```
Content-Security-Policy:default-src 'self';  
script-src 'self' 'unsafe-inline';  
img-src data: 'self';  
media-src blob: 'self';  
connect-src * 'self';  
style-src 'self' 'unsafe-inline'
```

No audio or video?

- Chrome is about to deprecate `getUserMedia` from “insecure origins”
- This means you can only get the camera and mic from `https://` or `localhost` URLs

No audio or video?

- Sometimes, it's a network problem and you have to dive into WebRTC's ICE negotiation to see what's happening
 - Chrome: <chrome://webrtc-internals>
 - FireFox: <about:webrtc>
 - We can help! Post questions to community.respoke.io

Questions?

Thanks!



<https://www.respoke.io>

Twitter: [@respoke](https://twitter.com/respoke)

GitHub: <https://github.com/respoke>