SG VIRTUAL CONFERENCE
6ta edición

Construcción de un Sistema de Comunicación en Tiempo Real

Presentado por:
KOOMBEA

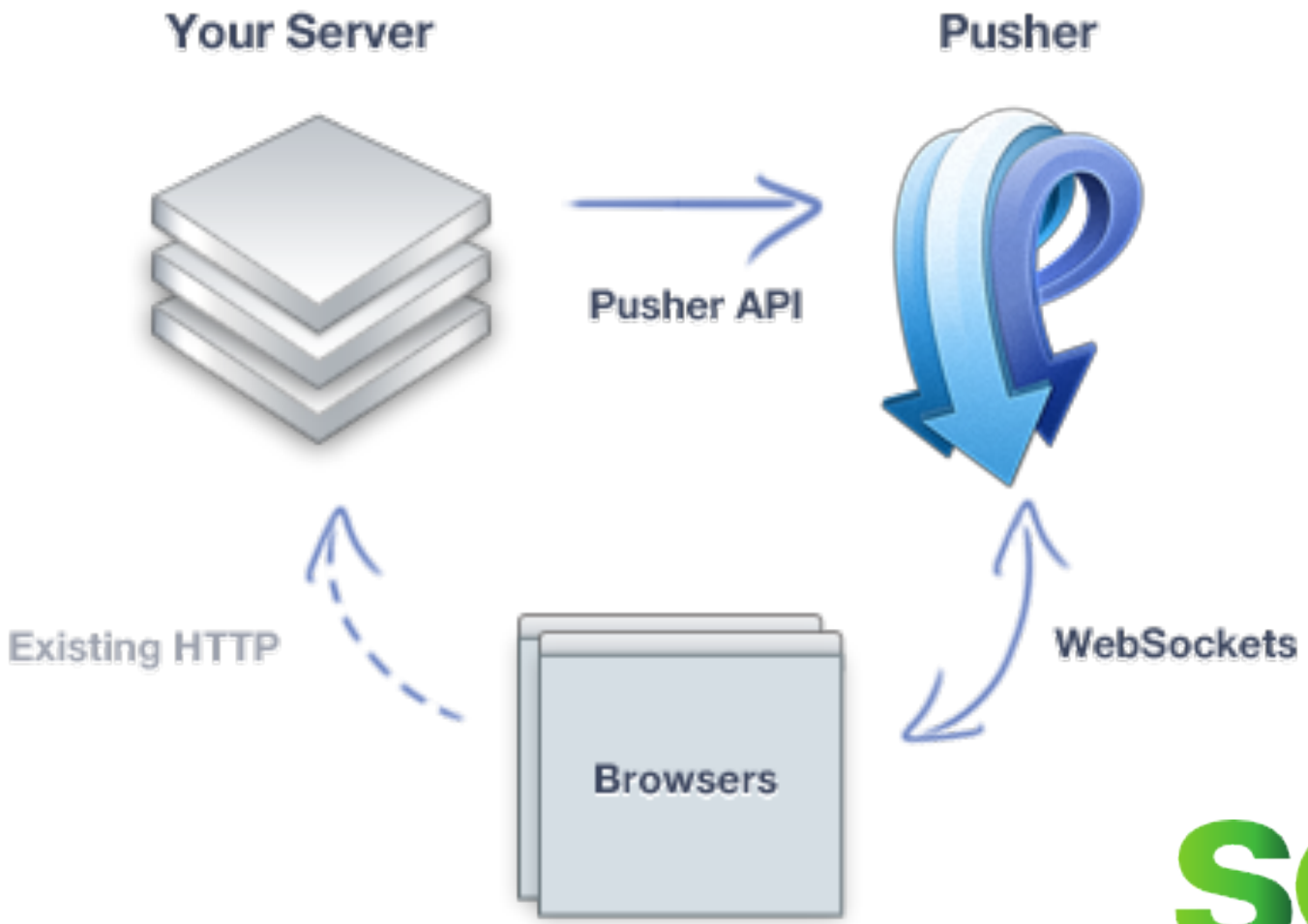
Agenda

- Que es Pusher ?
- Manejador Web de Aplicaciones de Pusher
- Arquitectura de Pusher
- Concepto Básico de Pusher
- Demo (chat interactivo entre dos aplicaciones android).

Que es Pusher ?

Pusher es a simple API que adiciona comunicación bidireccional en tiempo real vía WebSockets para web y aplicaciones móviles.

Arquitectura de Pusher



SG VIRTUAL CONFERENCE

Conexiones

La conexión de pusher es el medio más fundamental de la comunicación con el servicio. Esta es una con conexión bidireccional capaz de enviar y recibir mensajes.

```
var pusher = new Pusher(applicationKey, options);
```

Los parámetros **options** son opcionales sobre la instancia del constructor de pusher.

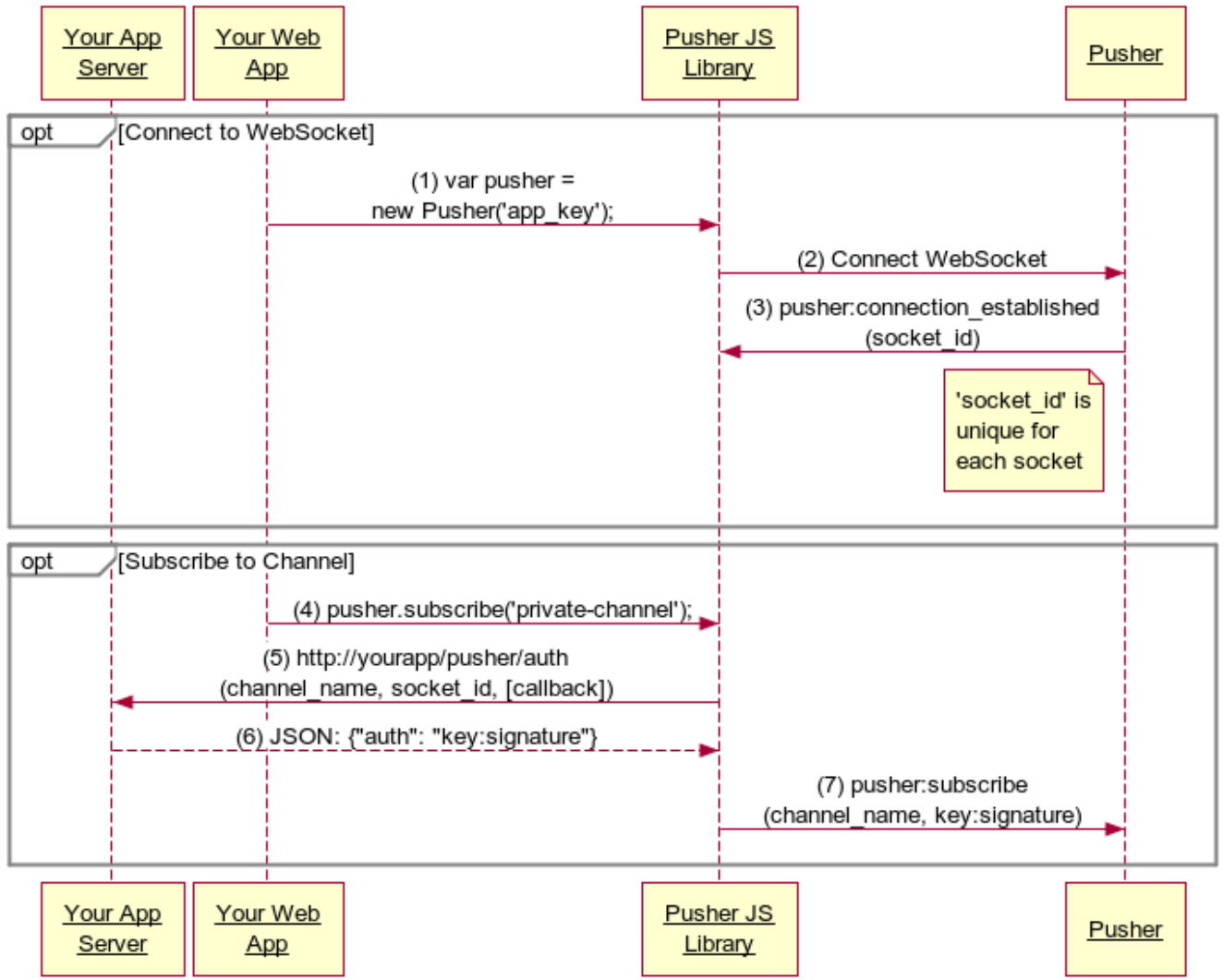
Estructura de los Parámetros

```
{
  encrypted: true, // true/false
  auth: {
    params: { // {key: value} pairs
      param1: 'value1',
      param2: 'value2'
    },
    headers: { // {key: value} pairs
      header1: 'value1',
      header2: 'value2'
    }
  }
}
```

SG VIRTUAL CONFERENCE

Proceso de Autenticación

- El siguiente diagrama muestra el proceso de autenticación de pusher.



www.websequencediagrams.com

SG VIRTUAL CONFERENCE

Endpoint para Canales Privados

Pusher tiene varios ejemplos en diferentes lenguajes, de cómo implementar un endpoint de autenticación para canales privados.

Los lenguajes son :

- Rails
- Node.js
- Php/Drupal
- Php/Wordpress
- ASP.NET
- Python

SG VIRTUAL CONFERENCE

Endpoint de Autenticación en Node.js

```
var express = require( 'express' );
var Pusher = require( 'pusher' );

var app = express( express.logger() );
app.use( express.bodyParser() );

var pusher = new Pusher( { appId: APP_ID, key: APP_KEY, secret: APP_SECRET
} );

app.post( '/pusher/auth', function( req, res ) {
  var socketId = req.body.socket_id;
  var channel = req.body.channel_name;
  var auth = pusher.auth( socketId, channel );
  res.send( auth );
} );

var port = process.env.PORT || 5000;
app.listen( port );
```

SG VIRTUAL CONFERENCE

Canales

Los canales son de gran importancia en pusher, y su concepto es fundamental para la comunicación.

En pusher cada aplicación tiene un limite de canales y el cliente escoge a que canal desea conectarse.

SG VIRTUAL CONFERENCE

Tipos de Canales

Pusher tiene disponible 3 tipo de canales :

- Public channel
- Private channel
- Presence channel

Canales Públicos

Los canales públicos son usado para acceder a cualquier tipo de información sin ningún tipo de restricción, y no requieren de autorización para poder subscribirse.

var publicChannel = pusher.subscribe(ChannelName);

pusher.unsubscribe(channelName);

SG VIRTUAL CONFERENCE

Canales Privados

Para un usuario suscribirse a un canal privado necesita un permiso para ser autorizado. Esta autenticación ocurre vía Http request a un endpoint configurable cuando el método **subscribe** es llamado.

Los canales privados debe empezar con el prefijo **private-** seguido del nombre del canal.

SG VIRTUAL CONFERENCE

Canales de Presencia

Canales de presencia se basan en la seguridad de los canales privados, y tienen una gran funcionalidad que permiten saber que usuarios estan suscritos a un determinado canal de presencia.

Los canales de presencia debe empezar con el prefijo **presence-** seguido del nombre del canal.

SG VIRTUAL CONFERENCE

Eventos

Los eventos son el principal método de empaquetado de los mensajes en el sistema de pusher. Ellos hacen parte de la comunicación y un evento puede ser visto como una notificación de alguna acción en tu sistema.

Los eventos no pueden ser utilizados como filtros (Los canales hacen este trabajo más eficiente).

Eventos

Los eventos cuentan con métodos para notificar el resultado de una operación o envío de mensajes.

- Binding to events
- Unbinding from events
- Pusher channel events
  1. pusher:subscription_succeeded event
  2. pusher:subscription_error event

- Triggering Client Events

SG VIRTUAL CONFERENCE

Demo

Recursos

- www.pusher.com
- http://www.genymotion.com/
- https://github.com/
- http://repo1.maven.org/maven2/com/pusher/pusher-java-client/

SG VIRTUAL CONFERENCE
6ta edición

Oscar Rodríguez

oscar.rodriguez@koombea.com