

SERVICIOS AVANZADOS DE TELEFONÍA IP MEDIANTE SIP



Antonio Abajo Álvarez

*Becario del Departamento de Ingeniería Telemática
Estudiante de Ing. Técnica de Telecomunicación, esp. Telemática. EPSC (UPC)*

Sergio Machado Sánchez

Profesor Visitante del Departamento de Ingeniería Telemática

ABSTRACT

La telefonía IP es una evolución de la telefonía convencional, que funciona mediante conmutación de circuitos, hacia conmutación de paquetes, un tipo de conmutación, en principio no pensada para este fin y que gracias al crecimiento de la popularidad de las redes de acceso de banda ancha aparece como una realidad viable. SIP es el protocolo de señalización utilizado en este tipo de telefonía. La incorporación de servidores de aplicaciones SIP permite la implementación de servicios avanzados en el marco de la telefonía IP que incrementan las posibilidades que ofrece la telefonía convencional. Para disponer de un servidor de aplicaciones dentro de un entorno hay que definir los elementos básicos de la arquitectura a la que pertenece tales como la integración con otros servidores de aplicaciones, con servidores SIP, con puertas de enlace, etc. La interacción de todos estos elementos se muestra en la explicación de un ejemplo real.

1 INTRODUCCIÓN

Actualmente, la evolución en el entorno de desarrollo de servicios de Internet se centra en facilitar el proceso de diseño e implementación de las aplicaciones que ofrecen servicios. Ya desde los primeros tiempos de la popularización de Internet hubo múltiples propuestas en cuanto a la aplicación de estrategias que permitiesen alcanzar un sistema capaz de gestionar comunicaciones utilizando los protocolos pertinentes. HTTP (HyperText Transfer Protocol) es el protocolo básico para acceder a la información disponible en la Web. La evolución del entorno de aplicación de dicho protocolo supuso, a medida que las posibilidades iban incrementándose, una complejidad en el desarrollo, como por ejemplo las páginas Web dinámicas, con lo que para realizar un website con una gestión óptima de los contenidos era necesario tener unos conocimientos elevados sobre desarrollo de aplicaciones servidoras (ejecutadas en el servidor).

La simplificación del desarrollo de estas aplicaciones comenzó con la aparición de los CGI (Common Gateway Interface), un estándar para conectar servidores de información, como los servidores Web a aplicaciones externas. Así, un documento HTML que sirve un daemon Web es estático, sin embargo si ante la petición HTTP el

servidor ejecuta cierta aplicación externa la respuesta obtenida por el cliente puede ser dinámica, observándose pues, que esta tecnología añadía "inteligencia" al puro servicio de acceso a documentos alojados en la Web. La idea de los CGI ha ido evolucionando en diversas tecnologías hijas como PHP, ASP, etc., y sobre todo la tecnología Servlet de la plataforma J2EE (Java 2 Enterprise Edition). Los servlets son una interfaz genérica para el desarrollo de aplicaciones servidoras cuya finalidad es la de ofrecer un sistema de servidor dinámico que gestione modularmente el procesado de peticiones HTTP definiendo unas interfaces capaces de independizar la parte de aplicación con la de transporte de la información.

Hoy en día el amplio despliegue de las redes de acceso como ADSL, cable, PLC (Power Line Communication) ofrecen mayor ancho de banda a los usuarios a un precio asequible. Si a eso sumamos la elevada optimización de los algoritmos de compresión y transmisión, se abre la posibilidad de ofertar servicios de voz y video en tiempo real a través de redes de conmutación de paquetes, redes originalmente no orientadas a dicho fin. En este entorno comienzan a aparecer soluciones de telefonía sobre IP (VoIP, Voz sobre IP). En su proceso de desarrollo de esta tecnología surgen nuevos horizontes a medida que los recursos de la red van incrementándose y que ofrecen algo más que el servicio básico de telefonía. De cara a que un operador pueda ofrecer servicios inteligentes de telefonía sobre IP y aprovechando la experiencia y el éxito de los servlets en el entorno Web, aparece la implementación de esa interfaz genérica para el protocolo SIP (Session Initiation Protocol) utilizado para el transporte de la señalización de las comunicaciones de audio y vídeo.

2 INTRODUCCIÓN A LOS SERVIDORES DE APLICACIONES

Los componentes encargados de la ejecución de las aplicaciones externas que añaden inteligencia a los servicios son los servidores de aplicaciones. Se entiende como servidor de aplicaciones a la entidad capaz de resolver peticiones dinámicamente a través de pequeños programas que se ejecutan en el servidor y que responden en función de los parámetros de dichas peticiones. La Fig. 1 muestra un ejemplo de entrada personalizada a un sitio Web usando HTTP. El servidor recibe la petición HTTP, y ejecuta una

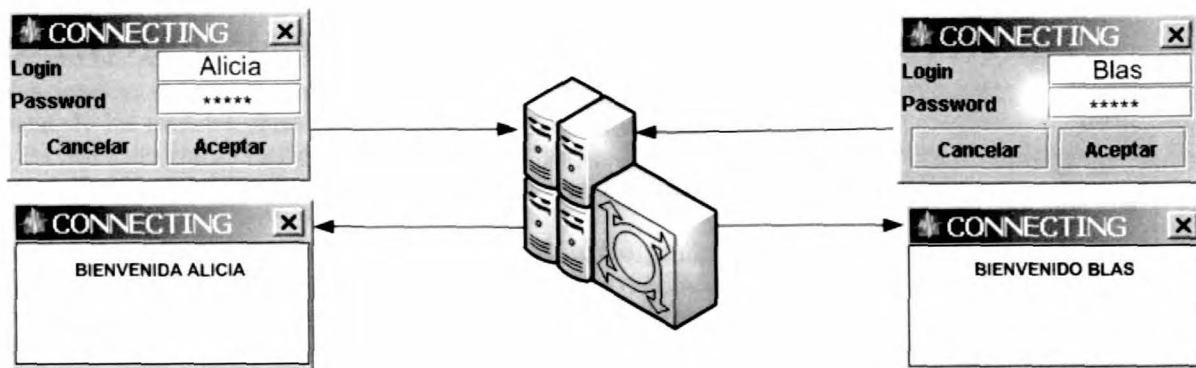


Figura 1 - Ejemplo de funcionamiento de un servidor de aplicaciones

determinada aplicación externa que en función del identificador de usuario y la contraseña entrada por el usuario y transportada dentro de la petición HTTP devuelve una página de bienvenida personalizada que muestra el nombre del usuario que se ha autenticado o, por ejemplo, un error en el caso de que el proceso de autenticación haya fallado por cualquier motivo.

Del mismo modo que sobre HTTP se quería ofrecer respuestas dinámicas en función de los parámetros de la petición, con un servlet SIP se busca ofrecer acciones dinámicas sobre determinados eventos que se pueden producir en el establecimiento o transcurso de una conversación telefónica.

El proceso de estandarización de los servlets SIP se enmarca dentro de una JSR (Java Specification Request). Este proceso consiste en plasmar una idea en un documento como propuesta para que la comunidad JCP (Java Community Process) la evalúe y decida seguir adelante en su estandarización, algo parecido al sistema que sigue el IETF (Internet Engineering Task Force) con los Internet Draft y los Request For Comments (RFC). La especificación "SIP Servlet API" (JSR número 116) se encuentra en estado "Final" y a partir de aquí, los desarrolladores interesados pueden consultarla para implementarla. En el contenido del documento se encuentra detallado el funcionamiento general del entorno, las clases y métodos que deben estar implementados, y el modo en que se relaciona el contenedor con los "SipServlets" y el entorno. Una JSR proporciona además una implementación de referencia, que tiene por objetivo dar un ejemplo del funcionamiento y composición conceptual del sistema para facilitar la comprensión de la especificación así como crear una implementación que sirva como entorno de pruebas de cumplimiento de la especificación. La implementación de referencia de una JSR no suele ser completamente funcional y no está diseñado para ser usado en entornos de producción.

3 EL PROTOCOLO SIP (SESSION INITIATION PROTOCOL)

SIP es un protocolo creado por el IETF y se encuentra especificado en el RFC 3261. Su propósito es transportar

la señalización de comunicaciones de voz y video a través de una red de conmutación de paquetes. Utiliza unos patrones para definir una lógica de intercambio de mensajes de cara a realizar una serie de funcionalidades que se consideran importantes para posibilitar que dos usuarios en Internet sean capaces de transmitir y recibir flujos de voz y video.

3.1 Situación de SIP en la pila TCP/IP

En la Fig. 2 se puede observar la situación de SIP en la pila TCP/IP. Una implementación SIP debe ser capaz de trabajar a nivel de transporte TCP/IP tanto con TCP (Transmission Control Protocol) como con UDP (User Datagram Protocol), si bien en el RFC se recomienda el uso de UDP por motivos de eficiencia en las comunicaciones y por el hecho de que no es necesario controlar la pérdida de mensajes a nivel de transporte ya que SIP posee un control propio.

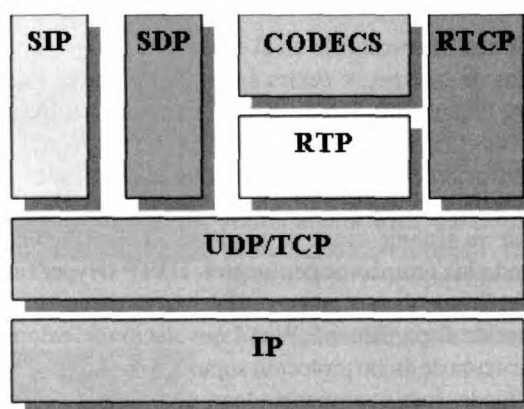


Figura 2 - SIP en la pila TCP/IP

Destacar que el intercambio de información de los flujos de voz y video no viaja sobre el protocolo SIP, si no que para su transmisión se utiliza un protocolo de transporte de información de tiempo real, típicamente Real-time Transport Protocol (RTP). Dentro de la especificación de RTP se incluye la definición del Real-Time Transport Control Protocol (RTCP) para la supervisión de la calidad de servicio durante la transmisión. Otra entidad

que aparece en esta pila son los codecs, necesarios para adaptar los streams de audio y video a un determinado formato. SIP únicamente interviene para señalar la conversación y para realizar la gestión previa entre los usuarios mediante la cual éstos se intercambian información de configuración de dispositivos, de los codecs que soportan, de la conectividad, etc., necesaria para que la comunicación se realice satisfactoriamente mediante SDP (Session Description Protocol).

Mensajes SIP

#	Tipo	Funcionalidad
1	INVITE	Alicia envía al Servidor SIP un mensaje indicando que quiere llamar a Blas
2	100 Trying	Servidor SIP responde a Alicia que está intentando establecer la llamada
3	INVITE	Servidor SIP reenvía el mensaje SIP hacia Blas para indicarle que Alicia le llama
4	100 Trying	Blas le responde que está procesando la petición.
5	180 Ringing	Blas responde al Servidor SIP que su terminal telefónico está sonando
6	180 Ringing	El Servidor SIP indica a Alicia que el terminal de Blas suena
7	200 OK	Blas notifica que ha descolgado el teléfono.
8	200 OK	Alicia recibe que Blas ha descolgado el teléfono.
9	ACK	Mensaje directo de Alicia a Blas previo a la transmisión de voz y video.
10	BYE	Blas cuelga el teléfono y envía este mensaje directo a Alicia
11	200 OK	Al recibir que Blas cuelga Alicia también cuelga notificándolo con este mensaje.

Tabla 1 - Peticiones SIP

Código de Respuesta	Descripción
1xx	Informativos: respuesta recibida, continuando para procesar la petición.
2xx	Sucesos: Acción recibida correctamente, entendida y aceptada.
3xx	Redirección: Acción futura requerida para completar la petición
4xx	Error del cliente: Peticiones contiene una sintaxis errónea o no puede ejecutarse
5xx	Error del servidor: Error del servidor al ejecutar una petición aparentemente válida
6xx	Error global: la petición no puede ejecutarse en cualquier servidor

Tabla 2 - Tipo de respuestas SIP

La Tabla 1 muestra el conjunto de peticiones definidos por SIP, mientras que la Tabla 2 hace lo propio con los mensajes que el usuario puede obtener como respuesta cuando envía una petición. Los mensajes que especifica el estándar SIP no son muy elevados en número en comparación al estándar H.323 que es el que hoy en día está

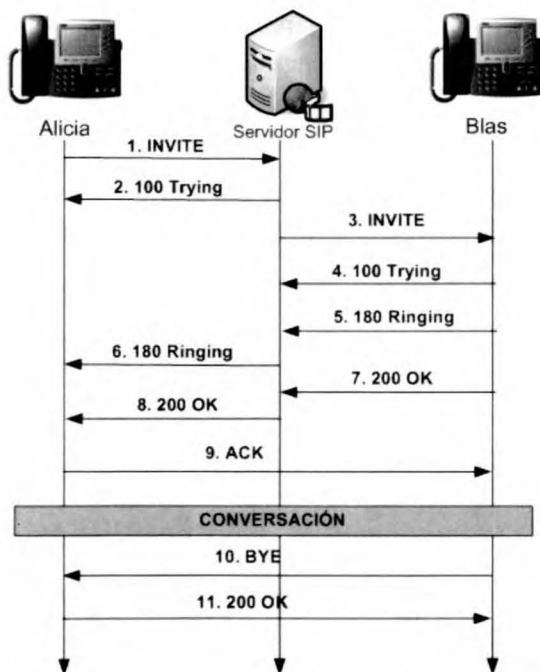


Figura 3 - Establecimiento de una conversación de voz con SIP

vigente en la mayoría de dispositivos destinados a actuar como clientes de voz y video sobre Internet como teléfonos VoIP, aplicaciones de mensajería instantánea tipo Messenger, etc.

La Fig. 3 ejemplariza el inicio de una conversación de voz utilizando SIP y que ayuda a observar el intercambio de mensajes entre los dos comunicantes y el servidor SIP.

La comunicación SIP de la figura 3 se corresponde a un inicio de llamada y a su finalización, ya que la conversación no forma parte del protocolo SIP. En la Tabla 3 se recoge una descripción de los mensajes intercambiados entre los dos comunicantes.

PETICIONES SIP	Descripción
INVITE	El usuario o el servicio está siendo invitado a participar en una sesión
ACK	El cliente ha confirmado la recepción de un INVITE
OPTIONS	Para preguntar al servidor sobre sus características o capacidades.
BYE	Indicar que el cliente quiere finalizar la llamada.
CANCEL	Cancela una petición pendiente
REGISTER	Un cliente registra su dirección en un servidor.

Tabla 3 - Funcionalidad de mensajes del ejemplo

4 ARQUITECTURA DE SERVICIOS AVANZADOS CON SERVLETS SIP

La finalidad de un servlet SIP es posibilitar la programación de servicios avanzados de valor añadido a nivel de aplicación. Como ejemplos de servicios que se pueden desarrollar en este entorno se podrían encontrar lo siguientes: un servicio de presencia (conocer si los usuarios del servicio se encuentran o no disponibles), mensajería instantánea, centralita de telefonía automática, buzón de voz, encaminamiento de llamadas, etc.

Para implementar un servlet SIP, el programador parte de una interfaz ya definida por la especificación. Una interfaz no es más que la declaración de métodos o funciones que ha de realizar una determinada clase o programa por el simple hecho de heredar de ella y que sirve como punto de aislamiento entre clases usuarias de esa interfaz y la implementación de dicha interfaz. Para simplificarlo más una interfaz es la plantilla que han de seguir los programadores para llevar a cabo una parte del programa que estén desarrollando. En la Fig. 4 se recoge el diagrama UML de la interfaz que todo servlet SIP tiene que implementar.

La interfaz de la figura 4 recoge todos los métodos que se pueden implementar en un servlet SIP, entre los más importantes encontramos los métodos doX(), donde X es el tipo de mensajes que se recibe en el servidor de aplicaciones. La idea es que toda la programación a partir de este método entrará en funcionamiento en cuanto se reciba un mensaje que origine el evento que redirigiera dicho mensaje a ese punto del programa.

SipServlet



Figura 4 - Diagrama UML de la interfaz SIPServlet

4.1 Componentes de la Arquitectura

La Fig. 5 muestra una arquitectura básica de la red de un operador de telefonía IP que use un servidor de aplicaciones SIP como solución para ofrecer servicios avanzados. Con esta arquitectura un operador puede ofrecer servicios integrados a sus usuarios, servicio de Voz sobre IP e integración con la red de telefonía conmutada

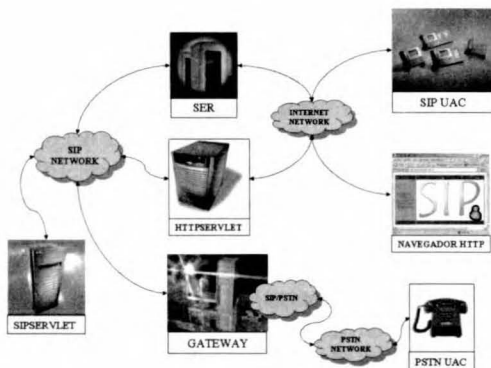


Figura 5 - Arquitectura básica de una red con servicios avanzados SIP

clásica o la red de telefonía de conmutación de paquetes (PSTN, Public Switched Telephone Network).

El operador dispone de un servidor SIP, en este caso se ha utilizado (SIP Express Router). La misión de este dispositivo es la gestión de las llamadas SIP entre agentes de usuario (dispositivos que poseen los usuarios para utilizar los servicios), que funcionen a través de la red de conmutación paquetes mediante la señalización proporcionada por SIP. Para poder comunicar un teléfono IP con abonados de otras redes se necesita disponer de gateways (puerta de enlaces) que adapten la señalización de las dos redes. Para ejecutar los servicios avanzados se dispone del servidor de aplicaciones SIP. Otra entidad que aparece es la de un servidor de aplicaciones HTTP. Tener este componente en la red del operador ofrece la ventaja de su posible integración en la red SIP, ya que las características dinámicas que ofrece HTTP facilitan la convergencia con SIP a través de un adaptador. Con esta arquitectura el operador dispone de una red que integra tanto telefonía convencional como aplicaciones HTTP y SIP.

5 EJEMPLO DE SERVICIO AVANZADO

En el departamento de Ingeniería Telemática estamos desarrollando una aplicación de servicio avanzado de videoconferencia utilizando servlets SIP que denominamos VideoMeeting y en la que existe una convergencia entre SIP y HTTP. Los servicios mínimos que se pretenden proporcionar son:

-**Servicio de Presencia.** Obtener información de la presencia de los usuarios, es decir, si están conectados o no, así como proporcionar información de utilidad como sus datos personales, su dirección de correo electrónico, etc.

-**Creación de reuniones.** Posibilitar la opción de crear reuniones de videoconferencia con los demás usuarios, gestionándose la aceptación de estas reuniones por parte de los usuarios implicados.

-**Videoconferencia.** Una vez acordada la videoconferencia para una determinada fecha y hora se establecerá la comunicación entre los usuarios que hayan sido invitados a esa reunión, utilizando el servlet para optimizar la transmisión de los flujos.

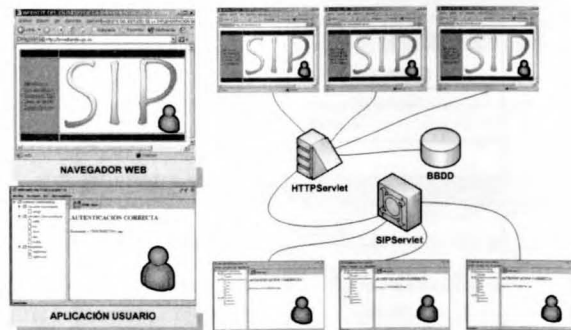


Figura 6 - Esquema de la aplicación Videomeeting

La Fig. 6 representa esquemáticamente el escenario que se persigue en el desarrollo de esta aplicación. Tenemos un servidor de aplicaciones Web con un servlet HTTP que se encarga de ejecutar la lógica de la gestión del servicio de reuniones a través de videoconferencia. Entre otras tareas este servidor se encarga de la autenticación, del registro de usuarios, del alta de reuniones, utilizando como repositorio una base de datos MySQL para almacenar la información de los usuarios. Por otro lado, tenemos un servlet SIP que se encarga de atender las peticiones SIP, funcionando como proxy hacia el servlet HTTP e implementando la lógica de procesado y reenvío de mensajes. Un usuario podrá hacer toda la operativa de gestión tanto a través de cualquier navegador Web como de la aplicación cliente Videomeeting que estamos desarrollando. Esta aplicación, implementada en Java y que utiliza una librería Java de procesado de mensajes SIP llamada NIST (National Institute of Standards and Technology) que permite generar mensajes SIP de forma cómoda, además de, como ya se ha dicho, permitir la gestión, será la que se utilizará para reproducir los flujos de audio, vídeo y texto que envíen el resto de participantes de la reunión.

6 CONCLUSIONES

La creciente popularidad de las redes de acceso de banda ancha proporciona la posibilidad de ofrecer a los usuarios servicios de comunicaciones de voz y vídeo. Gracias a esta capacidad de acceso y a la estandarización de protocolos de señalización como SIP, la telefonía IP se prevé como una alternativa factible a la telefonía convencional. Los servlets SIP, permiten crear servicios avanzados sobre el protocolo SIP lo cual permitirá a los operadores de telefonía IP la gestión y el desarrollo de servicios de valor añadido, dotándose de una herramienta muy potente y sencilla de utilizar, que garantizará dinamismo, eficiencia y un incremento en la calidad de los servicios ofrecidos al usuario.

Para más información:

BIBLIOGRAFÍA:

- [1] Practical VoIP using VOCAL, ed. O'Reilly
- [1] How TOMCAT works, ed. Brainysoftware

PÁGINAS WEB:

- [2] Documentos relacionados con SIP : <http://www.ietf.org/html.charters/sip-charter.html>

- [3] Request For Comments 3261 : <http://www.ietf.org/rfc/rfc3261>
- [4] Java Specification Request 116 (Sip Servlet) : <http://www.jcp.org/en/jsr/detail?id=116>
- [5] Java Specification Request 53 (Java Servlet) : <http://www.jcp.org/en/jsr/detail?id=53>
- [6] JAIN : <http://java.sun.com/products/jain/>
- [7] SLEE tutorial : <http://java.sun.com/products/jain/JAIN-SLEE-Tutorial.pdf>
- [8] SLEE - SipServlet technical comparison : <http://java.sun.com/products/jain/JSLEE-SIPServlet.pdf>
- [9] NIST <http://www.nist.gov/index.html>
- [10] Site Tomcat <http://jakarta.apache.org/tomcat/index.html>
- [11] Site SER: <http://iptel.org>
- [12] Site VOCAL <http://www.vovida.org/vocal>
- [13] Implementación de Referencia Sip-RI : <http://www.sipservlet.org>

AUTORES



Antonio Abajo Álvarez, nacido en L'Hospitalet de Llobregat (Barcelona) en 1980. Obtuvo el título de Técnico Superior en Sistemas de Telecomunicaciones e Informática en 2001. Actualmente está realizando el proyecto final de carrera para la obtención del título de Ingeniero Técnico de Telecomunicación, especialidad Telemática, en la EPSC, mientras trabaja como becario del Departamento de Ingeniería Telemática

Sergio Machado Sánchez, nacido en Barcelona. Obtuvo el título de Ingeniero Superior en Telecomunicaciones en 1998. Trabajó hasta 2003 en IBM España, fecha en la que se incorporó como profesor al Departamento de Ingeniería Telemática impartiendo asignaturas en la EPSC y en la EUETIT. Actualmente está realizando su tesis doctoral centrada en temas de transmisión de vídeo.