

# Análisis y Funcionamiento de Skype



## *Ingeniería Informática*

UNIVERSITAT DE VALÈNCIA

Estudi general

30 DE MAYO DEL 2006  
*Beatriz Arizo Aracil*

## INDICE

|   |           |
|---|-----------|
| <b>1.- Introducción</b>   | <b>3</b>  |
| <b>2.- Conceptos Previos</b>  | <b>4</b>  |
| 2.1.- VoIP  | 4         |
| 2.2.- Arquitectura peer-to-peer   | 5         |
| 2.3.- Protocolo UDP   | 5         |
| 2.4.- Protocolo TCP   | 5         |
| 2.5.- Herramientas de control de Tráfico                                | 6         |
| <b>3.- Skype</b>  | <b>7</b>  |
| 3.1.- Arquitectura Skype  | 7         |
| 3.2.- Componentes Skype   | 8         |
| 3.3.- Funciones Skype   | 10        |
| 3.3.1.- Arranque  | 10        |
| 3.3.2.- Conexión  | 10        |
| 3.3.2.1.- Proceso de conexión   | 10        |
| 3.3.2.2.- Conexión al servidor  | 12        |
| 3.3.2.3.- Supernodos Bootstrap  | 12        |
| 3.3.2.4.- Determinación de NAT y cortafuegos                            | 12        |
| 3.3.2.5.- Comprobación de la última versión de Skype                    | 13        |
| 3.3.3.- Búsqueda de usuario   | 13        |
| 3.3.4.- Establecimiento de llamada                                      | 14        |
| 3.3.5.- Codificación y transferencia                                    | 14        |
| 3.3.6.- Mensajes de tiempo de vida                                      | 15        |
| <b>4.- Comparación con otras aplicaciones: Yahoo, MSN y Google Talk</b> | <b>16</b> |
| <b>5.- Bibliografía</b>   | <b>18</b> |

## 1. Introducción

La comunicación ha sido una necesidad vital humana desde tiempos remotos. Esta exigencia ha provocado que a lo largo de los siglos se hayan experimentado diversos métodos y dispositivos, que permitieran encontrar una comunicación eficiente a distancia, obteniendo casi siempre resultados mediocres. No es sino a finales del siglo XIX, cuando la historia de comunicación se halla salpicada de sus hitos más importantes, como la telegrafía óptica de Chappe o la eléctrica de Morse, pero es la invención del teléfono por Graham Bell, la que permite entrar de forma plena en la comunicación moderna. A partir de ese momento, la historia de las telecomunicaciones es la historia del teléfono.

El teléfono ha evolucionado, ofreciendo hasta nuestros días, mejoras sustanciales y permitiendo una propagación de la comunicación, alcanzando cuotas inimaginables hasta hace tan solo 50 años, ya son más de 1000 millones de líneas telefónicas existentes por todo el globo, las cuales son testigos de esta evolución. No obstante pese a que su número y sofisticación han aumentado, nuevas barreras económicas representan un obstáculo a este tipo de comunicación en un mundo caracterizado por la globalización, los crecientes costes en esta clase de tecnología hacen necesario un nuevo salto que permita eludir estos nuevos problemas.

Este salto que esta a punto de relegar a un segundo plano al teléfono y ofrecer una alternativa real y barata, ha crecido de forma paralela y silenciosa utilizando la infraestructura telefónica existente para alimentar su evolución, es la joven tecnología denominada Internet.

El desarrollo de la "red de redes", alimentado por redes con altos niveles de calidad, fiabilidad y seguridad que poco tienen que envidiar a la telefonía tradicional, ha generado todo un conjunto de protocolos que rigen su funcionamiento. Entre esta ingente cantidad de protocolos cabe destacar uno llamado Protocolo de Internet (IP). El principal trabajo del protocolo IP es proporcionar un medio para el transporte de datagramas desde un origen a un destino, sin importar si estas máquinas están en la misma red, o si hay otras redes entre ellas. Es precisamente este protocolo el primer elemento que puede catapultar la comunicación personal ofreciendo una alternativa más barata y segura que la telefonía tradicional.

Pero no sólo este protocolo va ser la solución, un segundo elemento se debe unir a la fórmula resultante, para obtener un medio de comunicación rápido y barato. Este nuevo elemento no es otro que la Voz sobre IP (VoIP, Voice over IP) que es una tecnología que permite la transmisión de la voz a través de redes IP en forma de paquetes de datos.

De la combinación de ambos nace la telefonía IP que permite la realización de llamadas telefónicas ordinarias sobre redes IP u otras redes de paquetes utilizando un PC o teléfonos estándares.

Esta clase de telefonía posee varias ventajas que la hacen muy atractiva. Pese a lo que pueda parecer, es una tecnología ya madura, ya que las primeras investigaciones en algoritmos avanzados sobre este campo se remontan a 1970. Además ofrece una posibilidad barata, los costes asociados a una llamada de PC a PC son de un coste ínfimo comparados con una llamada tradicional. Otra ventaja es que se puede adoptar su utilización sin grandes traumas en infraestructuras para usuarios particulares o empresariales, para explotar en su plenitud esta clase de método. Y por último existen numerosas aplicaciones como MSN Messenger, AIM, Google Talk, TeamSpeak y sobre todo la aplicación Skype que facilitan esta transición y ofrecen servicios multimedia variados.

Dentro de este conjunto de herramientas cabe destacar Skype, que es el programa más densamente extendido y con el cual se va a utilizar como hilo conductor para explicar este tipo de programas.

## 2. Conceptos Previos

### 2.1. Voz sobre IP

La Voz sobre IP (*Voice over IP, Voip*) es una tecnología que permite la transmisión de la voz a través de redes IP en forma de paquetes de datos, Internet normalmente, en lugar de ser transportados a través de la red telefónica convencional. La telefonía IP es una aplicación inmediata de la tecnología anterior que permite la realización de llamadas telefónicas ordinarias sobre redes IP u otras redes utilizando un ordenador o un teléfono.

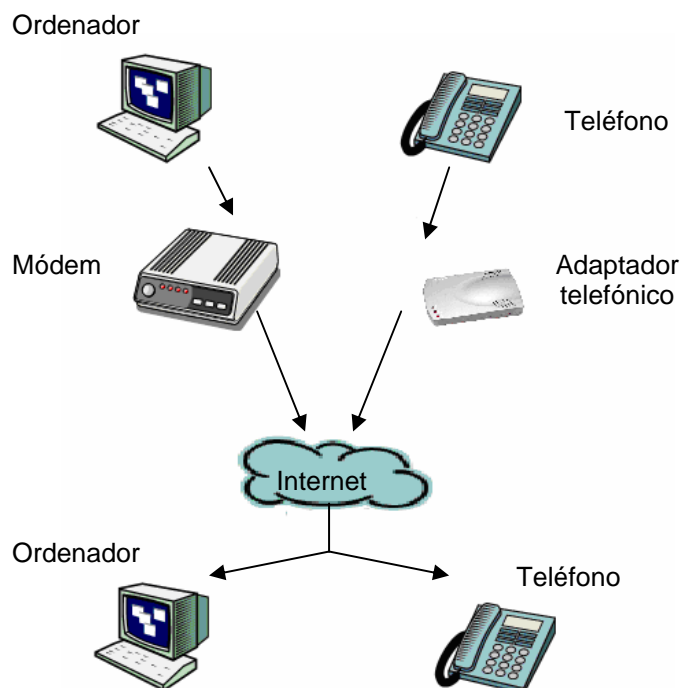


Figura 1. Detalle del funcionamiento de la telefonía IP

En una llamada telefónica normal, la centralita telefónica establece una conexión permanente entre ambos interlocutores, conexión que se utiliza para llevar las señales de voz. Pero en una llamada a través de Internet los pasos son diferentes y se pueden resumir en los siguientes: el emisor emite la voz, ya sea por el teléfono o mediante unos auriculares con micrófono, esta señal analógica se digitaliza, se comprime y por último se envía en paquetes de datos a través de Internet a la dirección IP del destinatario. Cada paquete enviado puede utilizar un camino distinto para llegar. Cuando éstos llegan a su destino se produce el proceso inverso a la emisión. Los paquetes son ordenados, reensamblados, descomprimidos y reconvertidos en la señal de analógica original.

### 2.2. Arquitectura peer-to-peer

En general, una red informática entre iguales (*Peer-to-Peer network, P2P*) hace referencia a una red que no tiene clientes y servidores fijos a diferencia del modelo cliente-servidor tradicional, sino que los nodos que forman la red se comportan a la vez como clientes y como servidores del resto de nodos. Los nodos pueden diferir en configuración local, velocidad de proceso, ancho de banda de conexión y capacidad de almacenamiento.

Debido a que la mayoría de los ordenadores domésticos no tienen una IP fija, sino que le es asignada por proveedor de servicios de Internet (*Internet Service Provider, ISP*) en el momento

de conectarse a Internet, no pueden conectarse entre sí porque no saben las direcciones que han de usar de antemano. La solución habitual es realizar una conexión a un servidor (o servidores) con dirección conocida (normalmente IP fija), que se encarga de mantener la relación de direcciones IP de los clientes de la red, de los demás servidores y habitualmente información adicional, como un índice de la información de que disponen los clientes. Tras esto, los clientes ya tienen información sobre el resto de la red, y pueden intercambiar información entre sí, ya sin intervención de los servidores.

### 2.3. Protocolo UDP

El protocolo de datagrama de usuario (*User Datagram Protocol*, UDP) es un protocolo del nivel 4, cuya especificación oficial se encuentra en el RFC768, tiene como principal función la de permitir una comunicación muy sencilla entre las aplicaciones de dos ordenadores utilizando el protocolo IP para transportar sus mensajes.

El protocolo UDP tiene básicamente dos características destacables, la primera es que es un protocolo no orientado a conexión, ya que no se establece una conexión previa con el otro extremo para transmitir un mensaje UDP. Los mensajes se envían sin más y éstos pueden duplicarse o llegar desordenados al destino. La segunda característica es que es un protocolo no fiable, ya que los mensajes UDP se pueden perder o llegar dañados al no incluir mecanismos que eviten la pérdida de mensajes.

El uso principal del protocolo UDP es para protocolos en los que el intercambio de paquetes de la conexión/desconexión son mayores, o no son rentables con respecto a la información transmitida, así como para la transmisión de audio y vídeo en tiempo real, donde no es posible realizar retransmisiones por los estrictos requisitos de retardo que se tiene en estos casos.

### 2.4. Protocolo TCP

El protocolo de control de transmisión (*Transmission Control Protocol*, TCP), es un mecanismo, especificado en RFC 793, que permite una comunicación fiable entre dos aplicaciones. Para ello utiliza unidades de transferencia de información denominadas segmentos. TCP utiliza para la conexión entre dos estaciones, el par dirección IP: puerto. No puede haber dos conexiones iguales en un mismo instante en toda la red. Aunque bien es posible que un mismo ordenador tenga dos conexiones distintas y simultáneas utilizando un mismo puerto. TCP posee las siguientes características:

- Fiable. A pesar de que el protocolo IP es no fiable, la información que emisor TCP envía llega de forma correcta al destino. Esto se consigue mediante el intercambio de paquetes, es decir, cada vez que llega un mensaje se devuelve una confirmación (*acknowledgement*, ACK) para que el emisor sepa que ha llegado correctamente. Si no le llega esta confirmación pasado un cierto tiempo, el emisor reenvía el mensaje.
- Orientado a conexión. Es necesario establecer una conexión previa entre las dos máquinas antes de poder transmitir ningún dato. A través de esta conexión los datos llegarán siempre a la aplicación destino de forma ordenada y sin duplicados. Finalmente, es necesario cerrar la conexión.

### 2.5. Herramientas de control de tráfico

- **Ethereal** es un potente analizador libre de protocolos de redes, para máquinas Unix y Windows. Permite capturar los datos directamente de una red u obtener la información a partir de una captura en disco.
- **NetPeeker** proporciona un control exacto y exhaustivo de todas las aplicaciones que hacen uso de la conexión a Internet en el PC. Permite limitar el ancho de banda usado por cada

una de ellas, pudiendo especificar limitaciones incluso por IP remota y por número de conexiones.

### 3. Skype

Skype es un cliente P2P VoIP desarrollado por KaZaa que también es una red peer-to-peer overlay. Skype permite que sus usuarios realicen llamadas de voz y envíen mensajes del texto a otros usuarios clientes de Skype. Esencialmente, es muy similar a las aplicaciones MSN y Yahoo IM, pues tiene capacidades para llamadas de voz, mensajería instantánea, audio conferencia y listas de contactos. Sin embargo, los protocolos y las técnicas subyacentes que emplea son absolutamente diferentes.

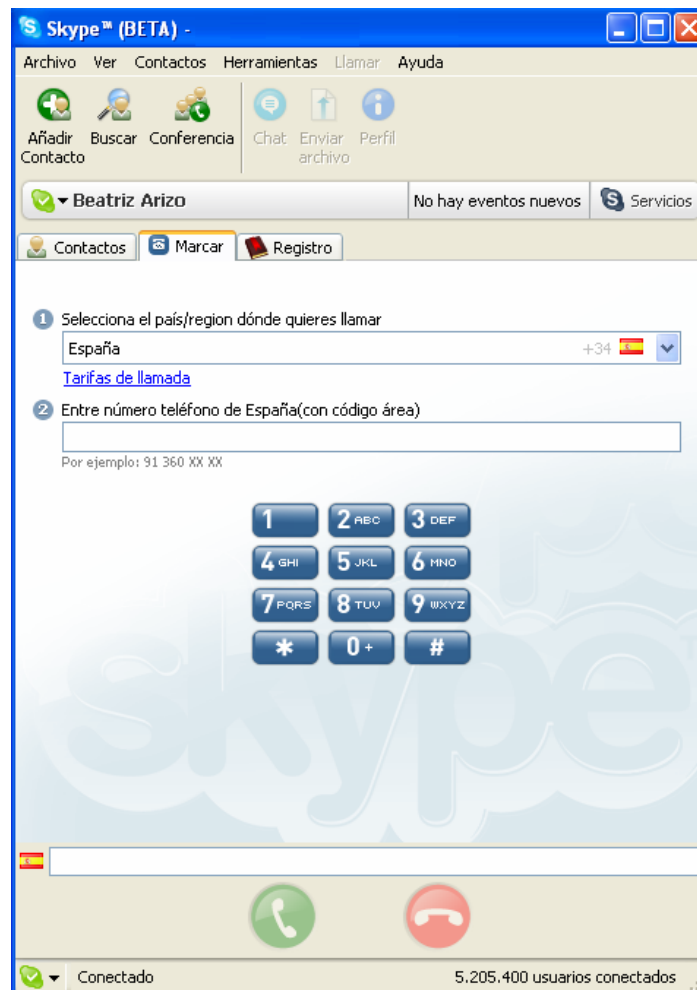


Figura 2. Imagen del inicio de Skype

#### 3.1. Arquitectura Skype

La arquitectura Skype se basa en un tipo de red llamada "overlay network" o red de aplicación que es un ejemplo de red P2P.

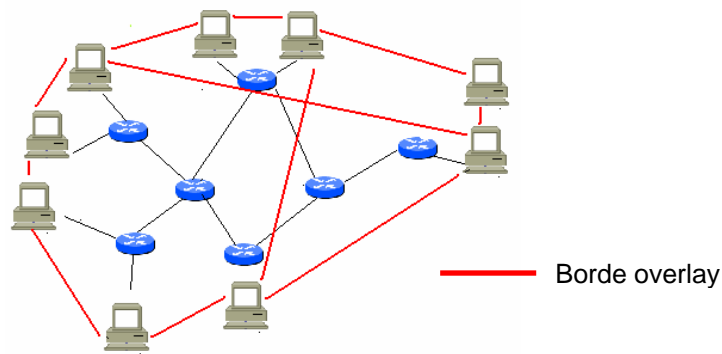


Figura 3. Ejemplo de red overlay

La red overlay de Skype tiene 3 tipos de nodos:

- a) **Nodo Normal:** nodo donde se ejecuta una aplicación o cliente Skype (SC) que se puede utilizar para realizar llamadas de voz y para enviar mensajes del texto
- b) **Supernodo:** nodo con una dirección IP pública que tiene suficiente CPU, memoria, y ancho de banda de red.
- c) **Skype login Server (Servidor de conexión):** Servidor único y única entidad central dentro del esquema de Skype asegurándose de que los nombres de la conexión sean únicos a través del espacio de nombres de Skype. Su principal función es la de permitir la autenticación del usuario y almacenar los nombres de forma única y las contraseñas del usuario.

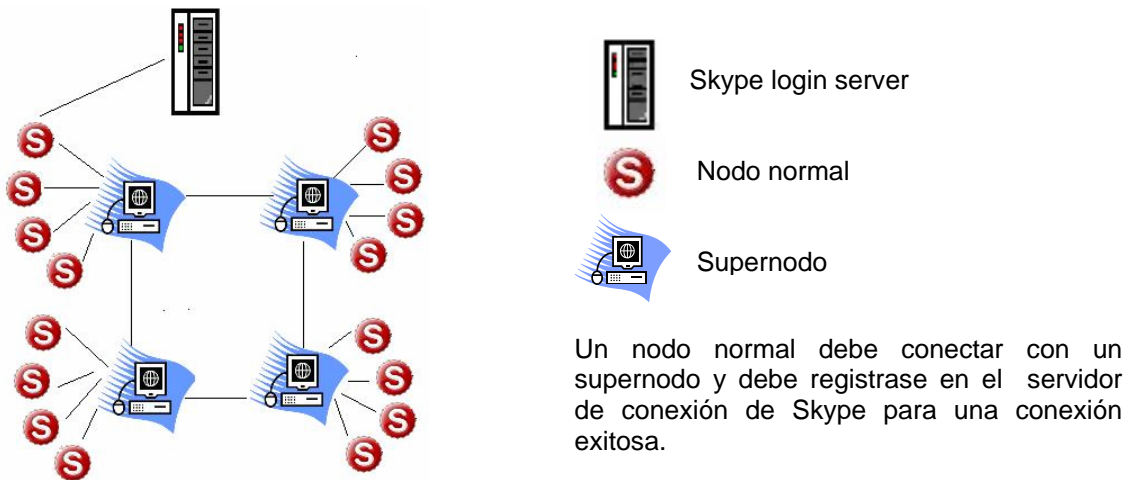


Figura 4. Red overlay de Skype

Aparte del servidor de conexión, no hay servidor central en la red de Skype. La información en línea y fuera de línea del usuario se almacena y se propaga de manera descentralizada y así como las búsquedas de los usuarios.

### 3.2. Componentes de Skype

Como toda aplicación de red, un cliente Skype abre puertos de escucha TCP y UDP, pero Skype a diferencia de otros protocolos como HTTP, no tiene puertos de escucha por defecto, sino que el número de puerto del cliente es generado aleatoriamente durante la instalación. Adicionalmente el SC abre los puertos de escucha TCP en los puertos 80 (HTTP) y 443

(http\_over\_TLS). La figura 5 muestra el cuadro de diálogo de conexión de Skype mostrando los puertos en los cuales un SC escucha las conexiones entrantes.

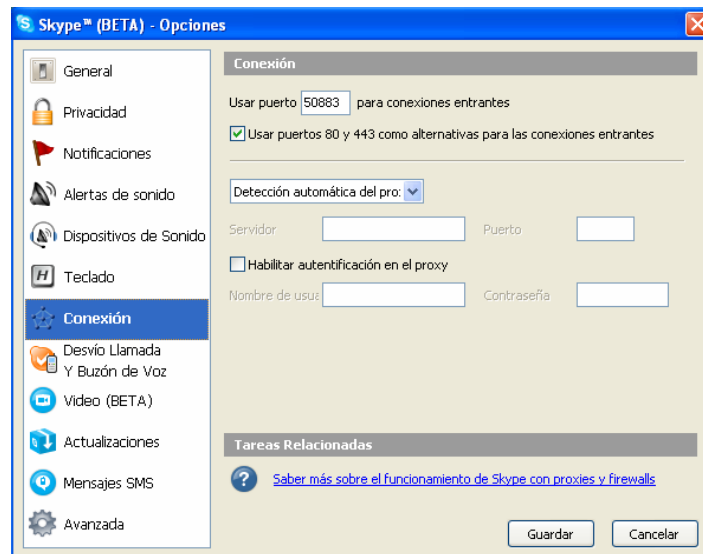


Figura 5. Tabla de conexiones de Skype, donde se muestran los números de puertos

Skype mantiene en la caché del cliente (*Host Cache*, HC) una tabla con pares de direcciones IP de supernodos y puertos, que se construye y refresca regularmente llegando a almacenar un máximo de 200 entradas. El mantener la caché del cliente es una de las operaciones más críticas de Skype. En las primeras versiones de Skype era necesario que al menos hubiera una entrada válida en la tabla, ya que durante el proceso de identificación tenía que establecer una conexión TCP e intercambiar información con cualquier entrada de la caché. De no ocurrir este intercambio, el programa daba un error en el proceso de identificación. A partir de la versión 1.2, este error se subsanó, de manera que si resultaba imposible realizar una conexión TCP con alguna entrada de la caché del cliente, entonces se intentaba establecer una conexión TCP e intercambiar información con alguna de los siete pares de direcciones IP y puertos alojados en el *Bootstrap* que se hallan codificados en el ejecutable de Skype.

Un cliente Skype en Windows XP almacena su caché de cliente en un fichero XML 'shared.xml' situado en 'C:\Documents And Settings\\Application Data\Skype'. En Linux se almacena el fichero XML en '\$(HOME)/.Skype'.

```
<?xml version="1.0" ?>
- <config version="1.0" serial="6625" timestamp="1135714201.11">
- <Lib>
+ <Account>
+ <BCM>
- <Connection>
- <Bandwidth>
  <CurSlotLength>6008</CurSlotLength>
  <LastRtTestTime>1135714068</LastRtTestTime>
  <OutHistory>7974</OutHistory>
</Bandwidth>
<DisablePort80>0</DisablePort80>
+ <EventServers>
- <Firewall>
  <TcpInHistory>-1431655768</TcpInHistory>
  <UdpInHistory>-1431655768</UdpInHistory>
  <UdpOutHistory>1431655807</UdpOutHistory>
</Firewall>
- <HostCache>
  <_1>140.115.23.23:62601</_1>
  <_10>87.69.48.254:1586</_10>
  <_100>140.121.135.224:3256</_100>
  <_101>217.199.108.68:35749</_101>
  <_102>217.199.108.67:59107</_102>
```

Figura 6. Trozo del contenido de la tabla de la caché del cliente. (shared.xml)

Para la codificación de audio Skype se utilizan los códecs iLBC, iSAC y iPCM, todos ellos desarrollados por GlobalIPSound que permiten utilizar frecuencias comprendidas entre los 50 y los 8000 Hz. Este rango de frecuencias es característico de una codificación de banda ancha.

La lista de contactos o *buddy list* se almacena encriptada en otro fichero XML llamado 'config.xml'. En Windows XP este fichero se aloja en 'C:\Documents And Settings\

```
<CentralStorage>
  <LastBackoff>0</LastBackoff>
  <LastFailure>0</LastFailure>
  <LastSync>1135714076</LastSync>
  <NeedSync>0</NeedSync>
  <SyncSet>
    <u>
      <skypebuddy1>2f1b8360:2</skypebuddy1>
      <skypebuddy2>d0450f12:2</skypebuddy2>
```

Figura 7. Trozo del contenido de la lista de contactos. (config.xml)

En cuanto a la encriptación de datos, Skype utiliza AES (*Advanced Encryption Standard*), también conocida como Rijndael que es usado también por las organizaciones de protección de datos del gobierno de los Estados Unidos. Esta encriptación utiliza 256 bits que permiten un total de  $1.1 \times 10^{77}$  posibles claves para cifrar cada llamada o mensaje instantáneo.

### 3.3. Funciones Skype

Dado que Skype no es un programa de código abierto, para averiguar las comunicaciones que se realizan entre clientes Skype, Basset y Schulzrinne utilizan dos clientes Skype usando las herramientas Ethereal y NetPeeker para analizar y controlar el tráfico de red. Especialmente se utiliza NetPeeker para analizar el funcionamiento de Skype en una red congestionada.

#### 3.3.1. Arranque

Cuando el cliente Skype funciona la primera vez después de la instalación, envía un *HTTP 1.1 GET Request* al servidor de Skype (skype.com). La primera línea de esta petición contiene la palabra clave 'installed'.

#### 3.3.2. Conexión

La conexión es quizás la función más crítica de la operación de Skype. Durante este proceso un cliente Skype autentifica su nombre y contraseña de usuario con el servidor de conexión, que anuncia su presencia a otros nodos y a sus contactos. Determina que tipo de NAT y de cortafuegos está detrás y descubre los nodos Skype con direcciones del IP públicas que están conectados.

##### 3.3.2.1. Proceso de conexión

Para comprobar el funcionamiento del proceso de conexión, primero se borra el fichero XML del cliente. Después se observa que éste manda un paquete UDP de 18 bytes de longitud a cada una de las siete entradas de la Bootstrap, todas ellas en el puerto 33033. Si no recibe respuesta en 5 segundos, el cliente intenta establecer una conexión TCP sobre cada una de las siete entradas sobre el mismo puerto. Si este intento de conexión falla se repite el proceso después de 6 segundos. Skype puede quedarse infinitamente

realizando estos intentos de conexión hasta que lo consiga, pues nunca devuelve un fallo de conexión.

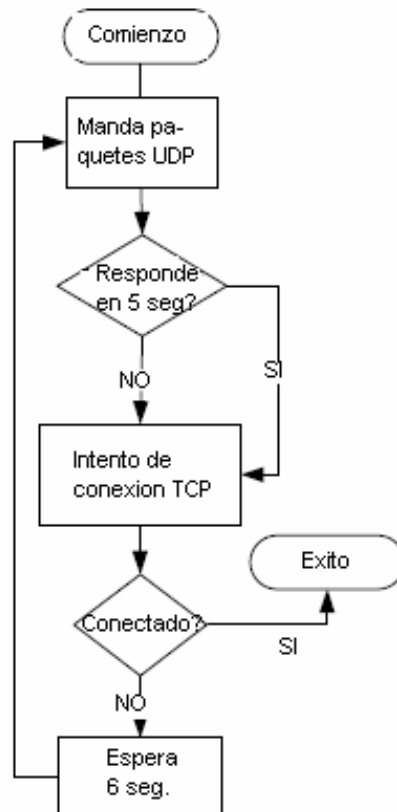


Figura 8. Diagrama de flujo del intento de conexión

Cuando se realiza la conexión se observa cual es el número mínimo de mensajes intercambiados. El primer y el segundo mensaje intercambiado con el servidor de conexión son siempre los mismos para diferentes intentos de conexión e incluso para distintos usuarios de Skype.

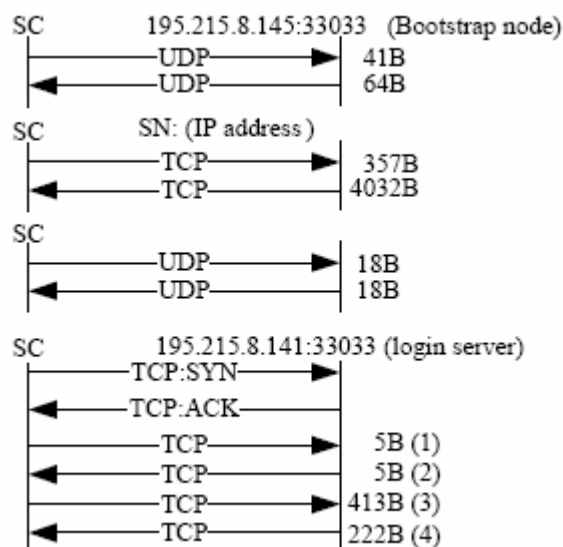


Figura 9. Diagrama de flujo del intento de conexión

La representación decimal con el Ethereal del mensaje (1) es 22 3 1 0 0 y la representación del mensaje (2) es 23 3 1 0 0. En la mayoría de los experimentos realizados, sólo cuatro mensajes se intercambian entre el servidor de conexión y el cliente Skype, siendo estos normalmente de la misma longitud. Los mensajes (3) y (4) son distintos para cada intento de conexión, aun así mantienen 4 bytes en común que son los mismos que los primeros 4 bytes de los mensajes (1) y (2) respectivamente, correspondientes a la cabecera del mensaje. El valor 22 de la cabecera corresponde al tipo de mensaje SSL, lo cual indica que Skype utiliza SSL para los mensajes de conexión.

Cuando la conexión falla, la longitud de los mensajes (1), (2) y (3) es la misma que en el caso anterior, mientras que el mensaje (4) tiene una longitud de 18 bytes indicando el fallo en la conexión.

### 3.3.2.2. Conexión al servidor

Después de que un cliente Skype se conecte a un supernodo, el cliente debe autenticarse mediante un nombre de usuario y contraseña con un servidor Skype. La conexión al servidor es el único componente centralizado existente en red P2P de Skype. Este almacena los nombres de usuarios y contraseñas asegurándose que son únicos dentro del espacio de nombres de Skype. Durante los experimentos realizados se observó que siempre intercambia datos sobre TCP con los nodos cuya dirección IP son 212.72.49.141 o 195.245.8.141, suponiendo por tanto que estos son los servidores de conexión de Skype.

### 3.3.2.3. Supernodos Bootstrap

Se listan las direcciones IP y los números de puertos de los siete supernodos por defecto que se observaron durante un intento fallido de conexión. Estas direcciones se muestran en la Tabla 1, obteniendo el *hostname* y la primera entrada de la sección de autoridad, mediante una búsqueda por *reverse lookup*. De aquí se muestra que uno sólo de los supernodos es mantenido por Skype.

| Dir IP:puerto       | Resultados del reverse lookup       | Sección de autoridad            |
|---------------------|-------------------------------------|---------------------------------|
| 66.235.180.9:33033  | sssl.skype.net                      | ns1.hopone.net                  |
| 66.235.181.9:33033  | No PTR result                       | ns1.hopone.net                  |
| 212.72.49.143:33033 | No PTR result                       | ns07.customer.e<br>u.level3.net |
| 195.215.8.145:33033 | No PTR result                       | ns3.DK.net                      |
| 64.246.49.60:33033  | rs-64-246-49-<br>60.evl.net         | ns2.evl.net                     |
| 64.246.49.61:33033  | rs-64-246-49-<br>61.evl.net         | ns2.evl.net                     |
| 64.246.48.23:33033  | ev1s-64-246-48-<br>23.evservers.net | ns1.evl.net                     |

Tabla 1. Direcciones IP del Bootstrap y los hostname obtenidos por reverse lookup

### 3.3.2.4. Determinación del NAT y del cortafuegos

Se supone que el cliente Skype es capaz, durante el proceso de conexión, de determinar si esta detrás de un cortafuegos o de un NAT. Bassett y Schulzrinne suponen que sólo hay dos maneras de saberlo. Una posibilidad es que el cliente intercambie mensajes con su supernodo utilizando una variación de protocolo STUN. Otra posibilidad es que durante la conexión, el cliente mande y reciba datos de algunos nodos después de establecer conexión con su supernodo, suponiendo entonces que el cliente utilice la variación del protocolo STUN para determinar que tipo de NAT o cortafuegos hay detrás.

Una vez determinado, el cliente almacena la información en el fichero 'shared.xml' refrescando esta información periódicamente.

### 3.3.2.5. Comprobación de la última versión de Skype

Durante la conexión, el cliente Skype solamente envía un *HTTP 1.1 GET Request* al servidor Skype (skype.com) para determinar si una nueva versión está disponible. La primera línea de esta petición contiene la palabra clave 'getlatestversion'. Junto con la petición del HTTP enviada al principio del arranque, éstos son los únicos mensajes basados en texto enviados por Skype.

### 3.3.3. Búsqueda de un usuario

La tecnología de búsqueda de un usuario utilizada por Skype es Global Index (GI). Mediante esta tecnología se consigue una búsqueda distribuida que garantiza la localización de un usuario que exista y que se haya conectado a Skype durante las últimas 72 horas.

Skype no es un protocolo abierto y sus mensajes están encriptados. Mientras que para la conexión se podían deducir las diferentes entidades involucradas, para la búsqueda de usuarios no es factible puesto que no se pueden seguir los pasos de los mensajes más allá del supernodo.

Un cliente Skype tiene una ventana de dialogo para la búsqueda (Figura 10). Después de introducir el usuario Skype a buscar y presionar el botón de Buscar, el cliente comienza la búsqueda.

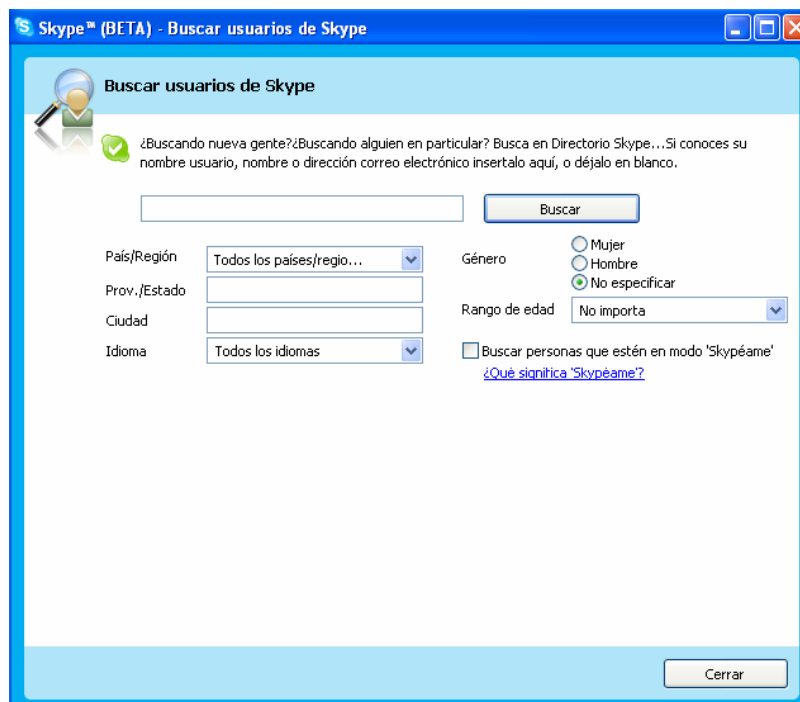


Figura 10. Ventana de dialogo para la búsqueda de un usuario

El cliente Skype manda un paquete TCP a su supernodo (SN). A continuación el supernodo le manda al cliente una lista de ocho nodos con sus direcciones IP y sus puertos como respuesta. Tras este intercambio, el cliente manda paquetes UDP a los ocho nodos. Si no se encuentra al usuario en estos nodos, se vuelve a establecer una conexión TCP con el supernodo, devolviéndole este las direcciones de 16 nodos, repitiendo el proceso anterior. Este proceso

continúa hasta que se encuentra al usuario o hasta que se resuelva que el usuario no existe. Un ejemplo de la búsqueda viene ejemplificado en la figura 11.

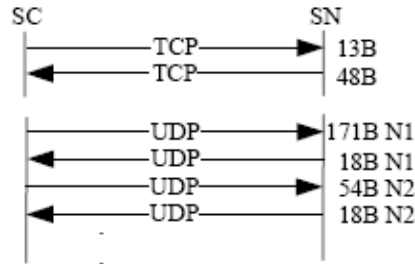


Figura 11. Flujo de mensajes en una búsqueda

Como media, un cliente encuentra al usuario buscado tras entrar en contacto con más de 24 nodos.

### 3.3.4. Establecimiento de llamada

Para el establecimiento de llamada, se considera únicamente que se realiza con un usuario de la lista de contactos. Si se realizase con un usuario que no estuviera en la lista de contactos, entonces sería equivalente a una búsqueda más un establecimiento de llamada.

Cuando un usuario establece una llamada a otro usuario, ambos en las listas de contacto propias, el usuario emisor establece una conexión TCP con el usuario receptor, de manera que toda la información entre los usuarios se intercambia sobre TCP tal como se muestra en la figura 12.

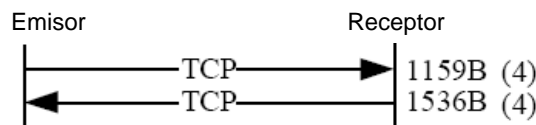


Figura 12. Flujo de mensajes en un establecimiento de llamada. El número de bytes corresponden al tamaño acumulativo de los mensajes intercambiados, siendo el número entre paréntesis, el número total de mensajes enviados en esa dirección.

El intercambio inicial de mensajes entre emisor y receptor indica la existencia de un mecanismo *challenge-response*. El emisor también manda algunos mensajes UDP (no mostrados en la figura 12) a nodos Skype alternativos.

Durante la finalización de la llamada también se realiza intercambio de mensajes sobre TCP entre emisor y receptor. Los mensajes observados durante el intercambio se muestran en la figura 13.

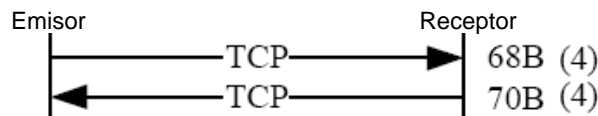


Figura 13. Flujo de mensajes en una finalización de la llamada. El número de bytes corresponden al tamaño acumulativo de los mensajes intercambiados, siendo el número entre paréntesis, el número total de mensajes enviados en esa dirección.

### 3.3.5. Codificación y transferencia.

En la transferencia de voz, el tráfico fluye sobre UDP en el puerto configurado en el cuadro de dialogo *Opciones*. El tamaño del paquete de voz varía entre 40 y 120 bytes. Para dos usuarios conectados a Internet mediante una Ethernet de 100Mb/s sin saturación, se llegan a

intercambiar alrededor de 85 paquetes de voz en un segundo. El total del ancho de banda de subida y de bajada es 5 Kbytes/s, que conviene con el ancho de banda demandado por Skype que se encuentra entre 3-16 Kbytes/s. La codificación utilizada es iSAC.

### 3.3.6. Mensajes de tiempo de vida

El cliente Skype refresca su conexión con el supernodo mediante mensajes TCP espaciados en el tiempo por dos minutos.

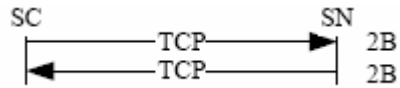


Figura 14. Flujo de mensajes para el refresco de la conexión

## 4. Comparación con otras aplicaciones: Yahoo, MSN y Google Talk

Los resultados del estudio comparativo realizado por Basset y Schulzrinne de las aplicaciones más utilizadas en el entorno de Internet, arrojan ciertos valores significativos a la hora de valorar su funcionamiento. Las aplicaciones utilizadas para el experimento son las siguientes:





El *Yahoo! messenger* o *Yahoo! Instant Messaging* (YIM) es una aplicación de mensajería instantánea (IM) de la compañía Yahoo. Esta aplicación permite el intercambio en tiempo real de mensajes entre dos o más usuarios en forma de texto. Pero en su última versión, además, permite la llamada a teléfonos fijos y móviles de todo el mundo, con una calidad de voz muy similar a la de Skype.

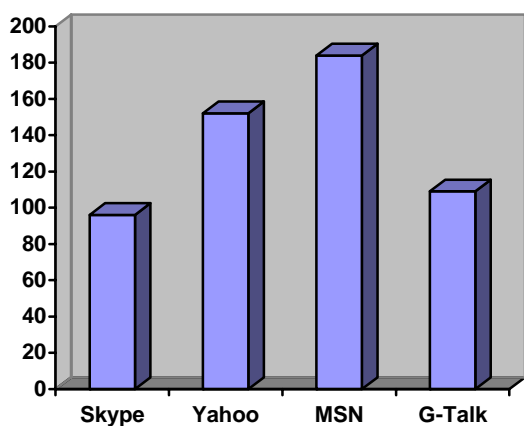
El MSN Messenger es el cliente de mensajería instantánea de Microsoft. Este cliente utiliza el protocolo de notificación de estado móvil (*Mobile Status Notification Protocol*, MSNP) sobre TCP y opcionalmente sobre HTTP para conectar al .NET Messenger Service, servicio ofrecido en el puerto 1863. La versión actual del protocolo no es totalmente secreta ya que es susceptible de captura de comandos mediante *sniffers* como *Ethereal*.





GoogleTalk es un programa que utiliza el protocolo Jabber y XMPP que permite la comunicación en tiempo real con otros usuarios mediante mensajería instantánea y llamadas de voz. Por un lado, Jabber es un protocolo abierto basado en el estándar XML para el intercambio en tiempo real de mensajes y presencia entre dos puntos en Internet. Mientras por otro XMPP, de las siglas inglesas *Extensible Messaging and Presence Protocol*, es un protocolo abierto y extensible basado en XML ideado para mensajería instantánea.

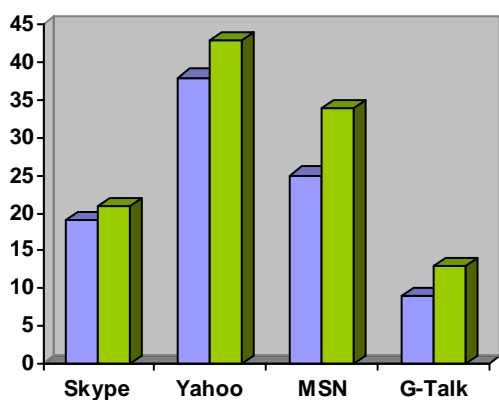
Para realizar la comparativa, los autores utilizan la latencia boca/oído, que se define como la diferencia entre el tiempo en que el emisor emite las palabras con su voz y el tiempo en el que el cliente receptor las escucha, se presupone que ambos clientes se encuentran en la misma sesión de voz. Esta latencia o retraso se calcula mediante la correlación entre las dos señales utilizando la transformada rápida de Fourier.





Se realizaron las pruebas con dos ordenadores portátiles, utilizando un archivo de audio SUN estéreo y pregrabado. Este archivo tiene una extensión .au de 8000 HZ y codificación lineal de 16 bits con una duración 24 seg. Se utiliza el programa *Ethereal* para hacer las capturas del emisor y receptor. Y se obtuvieron los siguientes resultados:

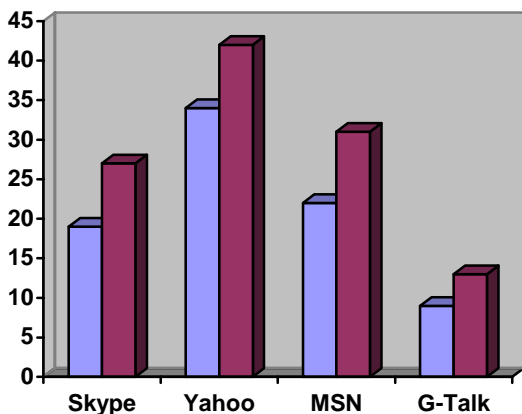
| Aplicaciones MI/Voz   | Versión   | Prioridad del proceso antes de la llamada. | Prioridad del proceso después de la llamada. |
|---|-----------|--|--|
|  | 1.4.0.84  | Normal                                     | Alta   |
|  | 7.0.0.437 | Normal                                     | Normal                                       |
|  | 7.5       | Normal                                     | Normal                                       |
|  | 1.0.0.80  | Normal                                     | Normal                                       |





| Aplicaciones MI/Voz   | Latencia boca/oído | Desviación estándar de la latencia |
|---|--------------------|------------------------------------|
|  | 96 ms              | 4                                  |
|  | 152 ms             | 12                                 |
|  | 184 ms             | 16                                 |
|  | 109 ms             | 10                                 |



| Aplicaciones MI/Voz Emisor  | Uso de Memoria antes de la llamada | Uso de Memoria después de la llamada |
|---|------------------------------------|--------------------------------------|
|    | 19MB                               | 21 MB                                |
|   | 38MB                               | 43 MB                                |
|  | 25 MB                              | 34 MB                                |
|  | 9 MB                               | 13 MB                                |



| Aplicaciones MI/Voz Receptor  | Uso de Memoria antes de la llamada | Uso de Memoria después de la llamada |
|---|------------------------------------|--------------------------------------|
|  | 19MB                               | 27 MB                                |
|  | 34MB                               | 42 MB                                |
|  | 22 MB                              | 34MB                                 |
|  | 9 MB                               | 13 MB                                |

## 5. Bibliografía

Skype. <http://www.skype.com>

Kazaa. <http://www.kazaa.com>

Ethereal. <http://www.ethereal.com>

Net Peeker. <http://www.net-peeker.com>

Salman A. Baset y Henning Schulzrinne. Columbia University. edn. 15 Septiembre 2004 "An Analysis of the Skype Peer to Peer Internet Telephony Protocol".

<http://www1.cs.columbia.edu/~salman/publications/cucs-039-04.pdf>

Oscar Ardaiz, Luis Díaz de Cerio. Universitat Politècnica de Catalunya. "Sistemas Distribuidos y CSCL".

GlobalIPSound. <http://www.globalipsound.com>

Codec iLBC. [http://www.globalipsound.com/solutions/solutions\\_whiteprs.php](http://www.globalipsound.com/solutions/solutions_whiteprs.php)

Global Index (GI): [http://www.skype.com/skype\\_p2pexplained.html](http://www.skype.com/skype_p2pexplained.html)

Estepa, R. (2005) "Internet de nueva generación." Área de Ingeniería telemática. Universidad de Sevilla. [http://www.it.uniovi.es/material/cursos/InternetNG\\_EU\\_072005/VoIP\\_1.pdf](http://www.it.uniovi.es/material/cursos/InternetNG_EU_072005/VoIP_1.pdf)

Wikipedia. Enciclopedia Libre (Español) [http://es.wikipedia.org/wiki/Protocolo\\_IP](http://es.wikipedia.org/wiki/Protocolo_IP)

[http://www.solont.com/z-net/tcp-04/tcp\\_04.htm](http://www.solont.com/z-net/tcp-04/tcp_04.htm)

<http://www.monografias.com/trabajos7/protoip/protoip.shtml> #FDIPC

Montañana, R. (2005) Apuntes Asignatura Redes de Ordenadores. Escola Tècnica Superior d'Enginyeria. Universidad de Valencia

Wikipedia. Enciclopedia Libre (Español) [http://es.wikipedia.org/wiki/User\\_Datagram\\_Protocol](http://es.wikipedia.org/wiki/User_Datagram_Protocol)