

Esquema de redundancia y distribución de carga de alta disponibilidad para la prestación de telefonía IP usando SIP

High availability redundancy and load distribution scheme for the delivery of IP telephony using SIP

Andrés Parra L., Ing.¹ y Fabio G. Guerrero, M.Sc.²

1. Ipsofactum S.A. ESP.

2. Escuela de Ingeniería Eléctrica y Electrónica, Universidad del Valle, Colombia

andresparra@ipsofactum.com; fguerrer@univalle.edu.co

Recibido para revisión 16 de marzo de 2009, aceptado 20 de mayo de 2009, versión final 28 de mayo de 2009

Resumen—En este artículo se presenta un esquema de distribución de carga y redundancia en la infraestructura de un ISP (Internet Service Provider) para la prestación de servicios de telefonía IP (ITSPs) sobre Internet usando el protocolo SIP (Session Initiation Protocol) garantizando niveles de acceso al servicio similares a la PSTN (Public Switched Telephone Network) y brindando una infraestructura de red escalable. La solución se plantea en tres partes: redundancia del servicio y distribución de carga, réplica de bases de datos y redundancia lógica a nivel de protocolo SIP en la interconexión con los proveedores de acceso a las redes conmutadas.

Palabras Clave—Telefonía por Internet, Distribución de Carga, Disponibilidad, SIP.

Abstract—In this paper a redundancy and load balancing scheme at an ISP (Internet Service Provider) infrastructure to provide IP telephony (ITSP) over Internet, guaranteeing levels of availability similar to those of the PSTN (Public Switched Telephone Network) and providing network scalability is presented. The proposed solution is divided in three main stages: redundancy and load balancing, database replication, and SIP level redundancy for interconnection with switched-network carriers.

Keywords—Internet Telephony, Load Balancing, Availability, SIP.

I. INTRODUCCIÓN

Debido a los desarrollos en los últimos 10 años en la tecnología de la VoIP (voz sobre el Protocolo de Internet), los servicios de telefonía IP (Internet Protocol) poseen hoy en día una serie de ventajas comparativas (costos de implementación, facilidades de operación y mantenimiento) y ventajas competitivas (costo para el usuario final y servicios complementarios) con respecto a la telefonía básica conmutada. Uno de los puntos en los cuales la telefonía tradicional aventaja por un margen pequeño, pero considerable para la experiencia de cliente final, a los servicios de telefonía IP es la disponibilidad del servicio.

El paradigma de la alta disponibilidad (*high availability*) en servicios de telefonía supone un servicio por lo menos con una disponibilidad del 99.999% de tiempo efectivo de servicio (comúnmente llamado *cinco nueves*). A lo largo de la historia de la telefonía tradicional por conmutación de circuitos los fabricantes hicieron de este indicador el índice de disponibilidad del servicio característico de la PSTN (Public Switched Telephone Network).

En cuanto a arquitecturas orientadas a ofrecer alta disponibilidad Cisco Systems propone en [1] una arquitectura basada en tecnologías como DNS SRV RR (Domain Name System SRV Records), redundancia de *gateways* (pasarelas entre el

mando SIP y la PSTN) y *proxies* y el Protocolo de Redundancia de Enrutador Virtual (VRRP) [2]. Los conceptos aplicados dependiendo del modelo y tecnología usados son los de inteligencia en el salto anterior el cual depende tanto de la inteligencia de los agentes usuarios como de la infraestructura e inteligencia en el salto redundante el cual depende de la organización de servidores y enrutadores. La aplicación de estas técnicas se desarrolla en tres segmentos. El primero es la interconexión entre los usuarios agentes SIP y el *cluster* de alta disponibilidad de SIP *proxies*, el segundo es la interconexión entre dos *clusters* de SIP *proxies* y finalmente se analiza la conectividad entre el *cluster* de alta disponibilidad de SIP *proxies* y el *cluster* de *gateways*. La red que se ejemplifica son nodos de clientes que se conectan a uno de los *clusters* de SIP *proxies* los cuales para conectarse a usuarios ubicados en otro nodo utilizando una interconexión entre *clusters*. Finalmente se tiene un *cluster* de *gateways* que le dan paso a toda la infraestructura para que se conecte con la PSTN. La solución final está orientada a una implementación dentro de una red de área local (LAN) debido a que se utiliza el protocolo VRRP el cual permite negociar una pertenencia activa de una dirección IP para un nodo entre varios dispositivos. Se argumenta que se prefiere VRRP por encima de DNS SRV RR debido a que el primero presenta menores costos de implementación y no hay una dependencia de la infraestructura DNS de la red LAN. Lo anterior no tiene relevancia en el Internet abierto ya que el sistema de resolución de nombres es un sistema altamente distribuido y automáticamente replicado.

Ohlmeier presenta en [3] un diseño basado en una federación geográficamente distribuida en la cual se poseen tres Proxies SIP [4] ubicados en diferentes ciudades los cuales atienden las solicitudes de cada uno de sus usuarios y a la vez soportan los clientes de sus contrapartes si se presenta una falla en alguno de ellos. Esta fue la primera implementación reportada en la literatura de una solución de redundancia y de alta disponibilidad para VoIP sobre el protocolo SIP basada en la inteligencia del salto anterior lo cual es más adecuado cuando se desarrolla una infraestructura que presta servicios en el Internet abierto.

Singh y Schulzrinne [5] proponen una solución basada en una arquitectura híbrida entre una arquitectura punto a punto (P2P) y SIP puro aprovechando la capacidad y robustez de los sistemas P2P distribuidos y las aplicaciones de telefonía que prestan los sistemas y servidores SIP tradicionales. La aplicación que comunica los dos mundos es SIPpeer la cual es, en términos SIP, un Proxy y a nivel P2P un nodo de una DHT (*Distributed Hash Table*) con organización cooperativa utilizando el protocolo *Chord* [6]. Este elemento permite a usuarios agentes SIP comerciales ingresar a la infraestructura sin realizar ningún cambio en su implementación del protocolo.

La estrategia P2P en términos de seguridad no brinda las garantías necesarias para una implementación comercial debido a que nodos maliciosos del anillo DHT pueden afectar la comunicación de una gran cantidad de usuarios y por ende

afectar de manera negativa en términos de disponibilidad a una población importante de clientes [5].

Singh y Schulzrinne presentan en [7] una discusión acerca de las diferentes metodologías de contingencia a fallos (*failover*) y balanceo de carga, los cuales son utilizados para el caso en particular de la telefonía SIP y proponen una arquitectura basada en DNS SRV RR para la contingencia de fallos y dos fases de servidores para garantizar el balanceo de carga. La ubicación del servicio entre usuarios agentes y la primera fase de servidores y la ubicación del servicio entre fases de servidores se plantea utilizando DNS SRV RR.

En la primera fase los SIP *proxy* realizan una selección utilizando el hash de la URI (Uniform Resource Identifier) para ser reenviada la petición a la segunda fase dependiendo de la primera letra del hash. El algoritmo utilizando para este hash es el SHA-1. Estos servidores de la primera etapa están configurados para no mantener los datos de la sesión (Stateless Proxy) lo cual les permite realizar la redirección e independizarse de la misma. Los *proxies* de la segunda etapa están operando para mantener los datos de la sesión consigo (Stateful Proxy) lo cual les obliga a mantenerse en el medio de la señalización entre los usuarios agentes hasta que esta termine.

La ventaja de esta arquitectura es que se distribuye la carga a nivel del servicio de localización [4] uniformemente entre los elementos de la segunda fase ya que cada uno estos guardan los datos de localización de una fracción de la población total de usuarios. La primera fase tiene como objetivo entonces atender las solicitudes entrantes a la infraestructura para luego ser distribuidas estáticamente entre los elementos de la segunda fase. Esta arquitectura no propone soluciones para la interconexión con *gateways*, *carriers* y proveedores de terminación en la PSTN y solo presenta un enfoque orientado a lo que se denomina PC2PC (llamadas entre usuarios agentes SIP de la misma red) el cual es un servicio gratuito que prestan todos los operadores de telefonía por Internet.

La arquitectura que se presenta en este artículo no solo se enfoca en la necesidad de conectividad entre usuarios de la misma red sino además presenta como desafío importante para el proveedor de servicios de telefonía IP (ITSP) garantizar contingencia a fallos en la interconectividad con los proveedores de terminación en la PSTN y *carriers* ya que este aspecto es la principal fuente de ingresos en este tipo de negocio. La arquitectura planteada en este artículo está siendo implementada en la infraestructura del proveedor de servicios de telefonía IP colombiano Ipsofactum S.A E.S.P (<http://www.ipsofactum.com>).

El alcance de este artículo está enfocado en el lado del proveedor de servicios de telefonía por Internet (ITSP) y la organización a nivel de software de la operación de un sistema de telefonía SIP. Se presentan bases prácticas para permitir a un operador de servicios de telefonía IP usando protocolo SIP obtener redundancia, distribución de carga y escalabilidad para lograr niveles de acceso a los servicios de telefonía similares a

los de la PSTN. No se consideran problemas, limitaciones y retos de desarrollo en el lado del cliente, como interrupciones en el servicio de Internet, errores del usuario y fallas en el software, hardware y/o firmware del terminal SIP. El estudio detallado de las medidas de disponibilidad para esta arquitectura está más allá del alcance del presente artículo.

La sección 2 describe brevemente la arquitectura interna de un conmutador de software para VoIP y sus elementos y funcionalidades para proveer servicios de telefonía IP, la sección 3 presenta la aplicación para lograr obtener redundancia con contingencia a fallos y distribución de carga en la infraestructura gracias a la utilización de registros DNS de ubicación de servicio (DNS SRV RR). La sección 4 describe la metodología para lograr la sincronización de las bases de datos con la información de contabilidad y enrutamiento a redes PSTN. Además se propone una estrategia para ofrecer escalabilidad y contingencia a fallos a nivel de bases de datos. En la sección 5 se proponen dos algoritmos para garantizar altos niveles de terminación de llamadas exitosas con redes externas en la conectividad con terceros (proveedores PSTN y *carriers*). Finalmente en la sección 6 se expone la arquitectura propuesta en su totalidad, se analizan casos a nivel de señalización SIP dentro de la misma teniendo en cuenta fallos de conexión y se valida la alta disponibilidad del sistema final.

II. ARQUITECTURA DE UN CONMUTADOR DE SOFTWARE DE VOIP (SOFTSWITCH)

En la Figura 1 se describe la arquitectura de un *softswitch* de VoIP estándar.



Figura 1. Arquitectura del Softswitch. Arquitectura diagramada de un softswitch de voz sobre ip de nueva generación segmentado desde las capas de protocolos hacia la capa de presentación y aplicación.

A continuación se describe brevemente cada parte:

A. Protocolos

Este bloque se encarga de brindar compatibilidad con los diferentes protocolos de señalización de VoIP, protocolos de la capa de transporte orientados a VoIP, y de transmisión como por ejemplo TDM (Time Division Multiplex) y PRI (Primary Rate Interface) de la red de servicios digitales integrados. La discusión en esta investigación se centra en el protocolo de inicialización de sesión SIP.

B. Motores principales

En este bloque se encuentran ubicados los diferentes motores básicos a nivel de protocolos de VoIP (SIP y H.323). Estos elementos permiten realizar todo tipo de conmutaciones, conversiones de protocolo y además como en el caso del B2BUA (Back to Back User Agent) [4] generar por si solos nuevas sesiones de voz. En la arquitectura planteada en este artículo son importantes los motores SIP *Proxy/Location/Registrar* y el B2BUA.

C. Funcionalidad

Aquí se encuentran todas las herramientas y aplicaciones para el tratamiento de numeración, planes de marcación, terminación, enrutamiento y AAA (Autenticación, autorización y contabilidad). El convertidor de codificadores de Voz (RTP Proxy) no es relevante para la presente investigación ya que se debe asumir que los dispositivos son compatibles en sus codificadores de voz.

D. Funcionalidad Opcional

Esta interfaz de programación de aplicaciones (API) permite realizar el desarrollo del control y toma de datos en tiempo real de la arquitectura de alta disponibilidad propuesta.

E. Interfaz de Administración y usuarios

En este bloque se encuentran las herramientas de presentación vía WEB para los registros detallados de llamadas (CDRs) y control de parámetros del sistema por parte del administrador del mismo.

III. REDUNDANCIA DE SERVICIO Y DISTRIBUCIÓN DE CARGA

Para el manejo de la redundancia planteado en este artículo se emplean las técnicas DNS utilizando registros de servicios DNS SRV RR [8] para ubicar servidores SIP [9] los cuales permiten distribuir carga a los servidores, asignar prioridad a nodos centrales SIP e ignorar servidores caídos en el momento de resolver el dominio del servicio de telefonía. Esta técnica se conoce como inteligencia en el salto anterior. Los registros de servicio DNS SRV RR tienen el siguiente formato:

Servicio._protocolo.nombre TTL SRVPrioridad Peso Puerto Destino.

En el formato anterior se observan los diferentes campos que componen un registro SRV, donde Prioridad significa en qué orden se debe contactar al destino (el de menor número se contacta primero) y Peso es un mecanismo de selección de servidores (el de mayor número tienen mayor probabilidad de recibir peticiones). Gracias a la naturaleza de este tipo de registros se puede ofrecer a priori redundancia en el acceso al servicio siempre y cuando el cliente SIP soporte DNS SRV RR lo cual se cumple para la mayoría de fabricantes de dispositivos y softphones SIP (Cisco, Linksys, Sipura, Grandstream, CounterPath, Polycom, etc). Para lograr distribución de carga se necesita configurar el campo Peso con respecto a las

capacidades de cada uno de los destinos (Servidores o Granjas de Servidores) y las necesidades de la infraestructura (Velocidad de Conexión). Para el caso de la Figura 2 se observa que se tienen definidos dos destinos para que los clientes busquen el servicio SIP (sobre TCP, UDP o TLS): sip1.ipsofactum.com y sip2.ipsofactum.com. Además *sip1* y *sip2* son registros DNS tipo A que podrían ser declarados como una granja de servidores.

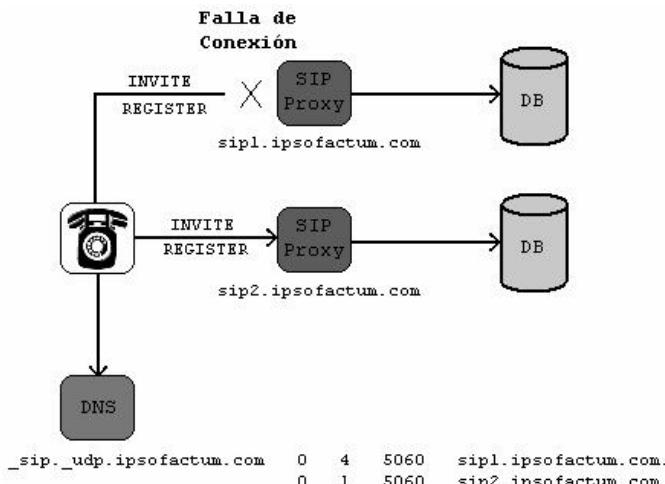


Figura 2. Ejemplo redundancia y distribución de carga de servidores SIP utilizando DNS SRV RR.

En la Figura 2 se observa que el destino *sip1* tiene 4 veces más de peso que el destino *sip2*. Esto significa que el 80% de las veces las peticiones van a ser enviadas a *sip1* y el 20% a *sip2*. Esta es una metodología de distribución de carga utilizando el campo Peso del registro DNS SRV la cual tal como se define en el RFC2782[6] representa información estática del estilo “El servidor A posee una mejor CPU” o “El servidor B cuenta con un canal de datos más rápido”.

La utilización de los SRV RR permite definir redundancia, peso y prioridad para la ubicación de servicios específicos definidos por la IETF en el RFC1700 (*Assigned Numbers STD2*) y actualizados en la base de datos de la organización IANA (Internet Assigned Numbers Authority). Basándose en las pruebas realizadas se tiene como conclusión que se pueden definir por DNS cualquier cantidad de servidores redundantes o inclusive “granjas” lo cual traslada el problema final a la sincronización de los mismos, el dimensionamiento de la infraestructura y los límites propios de esta.

IV. SINCRONIZACIÓN DE BASES DE DATOS

La base de datos es un elemento crítico para un sistema de VoIP ya que en la misma se almacena toda la información para realizar AAA (Autenticación, Autorización y Contabilidad) en tiempo real, además de almacenar los parámetros de

enrutamiento, costos, tarifas y proveedores. Todo softswitch posee controladores para la conexión con gestores de bases de datos como MySQL, PostgreSQL, LDAP y Oracle.

Para este estudio en particular, se utiliza el gestor MySQL [10] ya que este presenta ventajas de estabilidad, portabilidad, su uso es libre bajo licencia GNU GPL (Licencia Pública General) y ha probado su compatibilidad con el conmutador de software de Ipsofactum.

El motor de bases de datos MySQL posee funciones y modos de operación que permiten realizar réplicas de bases de datos. Esto se logra gracias a que el demonio *mysqld* puede ejecutarse generando registros (*log*) binarios de todas sus operaciones y comandos SQL (*Structured Query Language*). En un esquema de réplicas como el que utiliza MySQL existen dos agentes diferenciados: el *Maestro*, sobre el cual se ejecutan todas las tareas de lectura y escritura y el *Esclavo*, sobre el cual se realizan las lecturas de los datos y cuya información permanece actualizada gracias a que es replicada desde el *Maestro*. Esto permite un balanceo de carga para todas las solicitudes de los *Agentes Usuarios*, una solución en la cual se puede garantizar escalabilidad en términos de la conectividad de nuevos agentes a la infraestructura. En la Figura 3 se describe el esquema propuesto para garantizar escalabilidad. En [11] se plantea un esquema similar de replicas de dos vías el cual es escalable a solamente dos elementos redundantes.

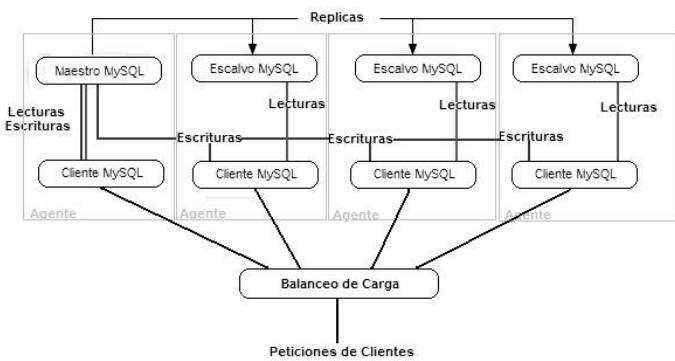


Figura 3. Esquema de replicas garantizando escalabilidad.

Como se observa en la Figura 3, uno de los agentes está configurado como *Maestro* de réplicas y los restantes poseen una denominación de *Esclavos* para ese momento en particular. Las operaciones de escritura se dan únicamente sobre el *Maestro*. Las operaciones de lectura se pueden dar tanto sobre el *Maestro* como los *Esclavos*. Como regla general para evitar sobrecargas en el agente denominado circunstancialmente *Maestro*, el cliente MySQL de los agentes redundantes que posean un *Esclavo MySQL* deben realizar las operaciones de lectura sobre la base de datos local y no sobre la base de datos de otro agente. Por lo tanto solo el agente redundante al que pertenece el denominado *Maestro MySQL* realiza operaciones

de lectura sobre la base de datos del *Maestro MySQL* de la arquitectura.

Debido a que un solo agente no puede garantizar una alta disponibilidad por sí mismo para el servicio de bases de datos de MySQL eventualmente el *Maestro* puede salir de línea. Para este caso, el motor MySQL cuenta con funciones para reconfigurar el servicio de bases de datos de cada agente cambiando el *Maestro* de réplicas e informar a los demás agentes para que estos se reconfiguren y procedan a leer los registros binarios (*logs binarios*) del nuevo *Maestro* como se muestra en la Figura 4.

Se debe tener en cuenta que una vez el agente que se encontraba en falla vuelve en línea, este no debe recibir peticiones hasta que esté sincronizado. El *Maestro* que tomó el control debe mantenerlo hasta que se presente una falla nuevamente. Ahora, si alguno de los esclavos falla es necesario solo la resincronización del *Esclavo* en falla para que el sistema entre en operación normal.

Con este enfoque y aprovechando las características y funciones de un motor de bases de datos con MySQL se garantiza que la información esté sincronizada en todos los agentes redundantes para mantener los datos del sistema AAA y las estadísticas del servicio de telefonía IP constantemente actualizados y disponibles, mientras el servicio de localización SIP que permite comunicación entre usuarios de la misma red y el servicio de enrutamiento a redes externas permanezcan activos.

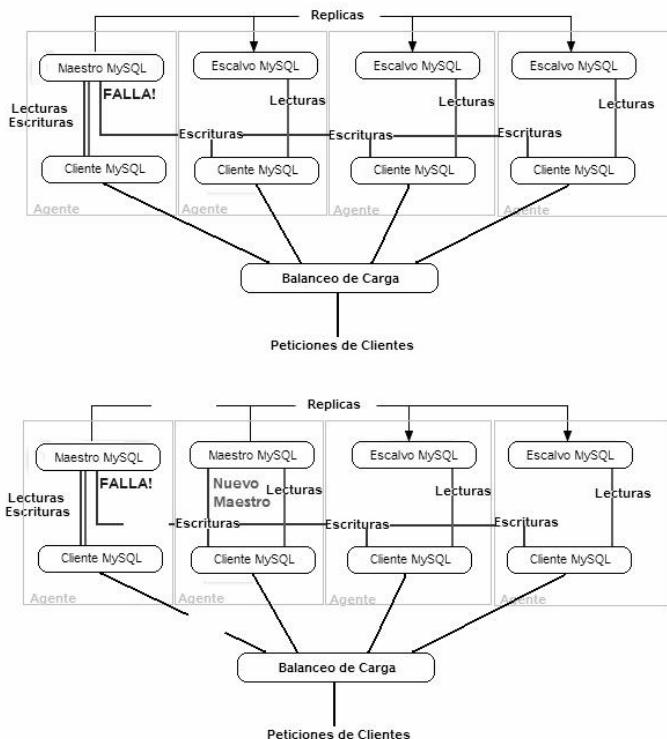


Figura 4. Contingencia del sistema de réplicas en caso de falla de maestro.

V. REDUNDANCIA DE PROVEEDORES DE TERMINACIÓN EN LA PSTN

Uno de los retos a enfrentar es lograr maximizar el porcentaje de terminación de llamadas a redes externas, esto se logra con una estrategia de proveedores redundantes. Internamente en todo *softswitch* el motor principal del conmutador debe generar un reporte del estado de las llamadas y de cada canal. Los estados generados por los intentos se describen a continuación.

- 1) “ANSWERED”: Este estado describe que la llamada fue correctamente contestada.
- 2) “NOANSWER”: La llamada no fue contestada. Depende del tiempo máximo de timbrado permitido en el sistema.
- 3) “BUSY”: El abonado está ocupado.
- 4) “CANCEL”: El intento de llamada fue cancelado por el usuario.
- 5) “CHANUNAVAIL”: El canal no está disponible ya que la numeración es incorrecta.
- 6) “CONGESTION”: La ruta está congestionada.

El algoritmo desarrollado tiene la capacidad de detectar en tiempo real el estado del canal y basándose en este realizar reintentos secuencialmente o en paralelo para maximizar el porcentaje de terminación de llamadas exitosas.

El comando de marcación posee las siguientes variables en la base de datos:

- Host: Dirección IP o nombre canónico.
- Protocol: SIPoTCP (SIP sobre transporte TCP), SIPoUDP (SIP sobre transporte UDP) o SIPoTLS (SIP sobre transporte TLS).
- Port: Puerto de conexión.
- Prefix: Prefijo de marcación.
- Telnumber: Número marcado por el cliente.

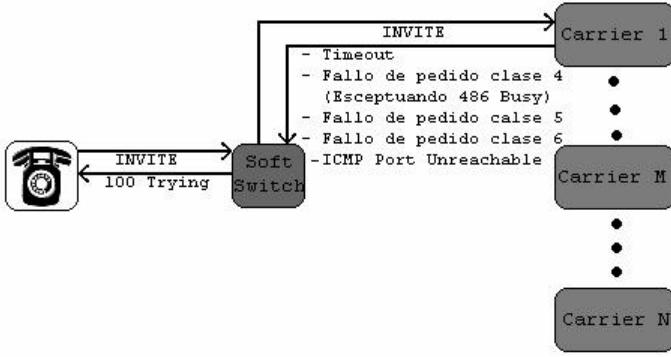
En el softswitch se ejecuta el comando de marcación con el formato:

Dial (Protocol / Prefix Telnumber @ Host:Port)

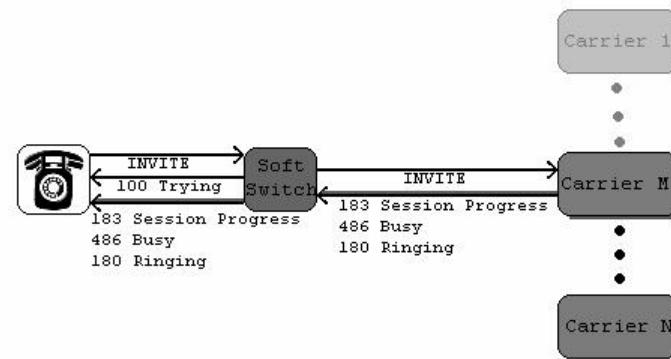
A. Algoritmo para enrutamiento serial

Se inicia ejecutado el comando de marcación en el softswitch el cual crea una canal de conexión. Una vez el canal es cerrado se debe establecer la razón de terminación (o estado) del mismo. Si el estado del canal es “BUSY” o “ANSWERED” el algoritmo termina y se procede a generar el reporte de la llamada al sistema de contabilidad. Si por otra parte el estado del canal es diferente a los estados mencionados anteriormente se procede a ejecutar otro comando de marcación con la información del proveedor que sigue en el orden de prioridad definido en el sistema de enrutamiento del softswitch para cada ruta. Este proceso se

realiza tantas veces como proveedores inscritos se tengan en el sistema de redundancia como se muestra en la Figura 5. Este método permite definir una preferencia estática dependiendo de calidad y costos para cada una de las rutas, por lo tanto se recomienda utilizar este enfoque para priorizar la utilidad.



Intento con Carrier: 1 FALLA



Intento con Carrier M: Éxito

Figura 5. Ejemplo de algoritmo de proveedores redundantes en serie.

B. Algoritmo para enrutamiento en paralelo

Se ejecuta simultáneamente tantos comandos de marcación como proveedores estén inscritos por ruta tal como se ilustra en la Figura 6. Los proveedores se inscriben en el sistema de enrutamiento interno del softswitch. El primer proveedor que responda la llamada queda en propiedad del canal e inmediatamente se cierran los otros canales. Dado que es muy probable que la mejor ruta disponible y por tanto posiblemente la más costosa, enlace primero la llamada, se debe tener en cuenta que este enfoque es adecuado para mejorar los índices de terminación de llamadas exitosas (%ASR) y tiempo o latencia de post-marcado (PDD), lo cual es penalizado en términos de la utilidad monetaria.

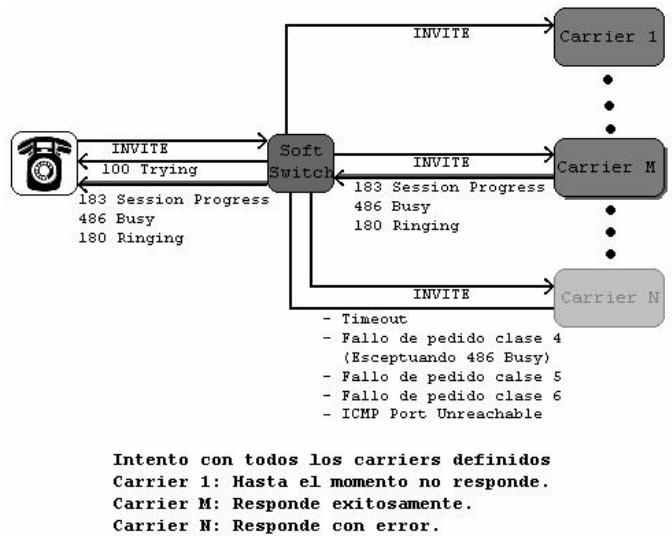


Figura 6. Ejemplo de algoritmo de proveedores redundantes en paralelo.

Utilizando este tipo de algoritmos se maximiza la capacidad de ofrecer interconexión y terminación con redes

PSTN optimizando desde un enfoque práctico en la prestación de servicios de telefonía por Internet el índice de terminación de llamadas exitosas ASR (Answer Seizure Ratio), tiempo o latencia de post-marcación PDD (Post Dial Delay) y brindando la capacidad de mejorar los porcentajes de utilidad de este tipo de negocios.

VI. ARQUITECTURA PROPUESTA Y CÁLCULO DE LA DISPONIBILIDAD

La figura 7 muestra la arquitectura planteada la cual se basa en agentes redundantes, cada uno de los cuales posee un *softswitch* con su respectiva base de datos. A nivel SIP tal como se expuso en la sección 2, el *softswitch* posee como uno de sus motores principales un *Proxy SIP* que se encarga de recibir directamente todas las peticiones SIP para procesarlas y redireccionarlas a su destino. Como regla general cada cliente del proveedor de telefonía IP posee solamente un dispositivo SIP (*softphone*, *hardphone* o *analog telephone adaptor*) por línea IP registrada en el sistema el cual es aprovisionado mediante TFTP (Trivial File Transfer Protocol) o HTTP (HyperText Transfer Protocol). Esta característica evita violaciones de seguridad sobre los usuarios del proveedor ya que los mismos no conocen la dupla usuario-password SIP con la que se autentica el dispositivo SIP con la infraestructura ni los parámetros de configuración y además libera a la infraestructura de lidiar con múltiples dispositivos para un mismo usuario SIP, la cual se conoce como *forking*. La figura 7 muestra la arquitectura planteada en su totalidad indicando todos los elementos presentes discutidos.

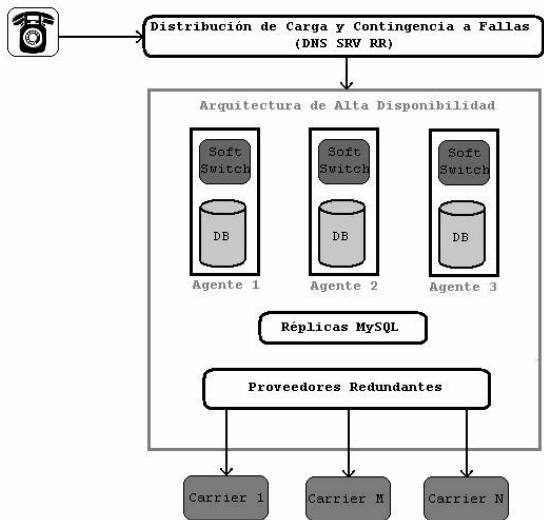


Figura 7. Arquitectura de alta disponibilidad propuesta.

A continuación se muestran dos ejemplos de flujo de llamadas sobre la infraestructura usada por Ipsofactum. En la Figura 8 se puede observar la capacidad de contingencia a fallas de la arquitectura en eventos en los cuales se realiza una llamada dentro de la misma red y en la Figura 9 uno en el cual la llamada tiene como destino la PSTN.

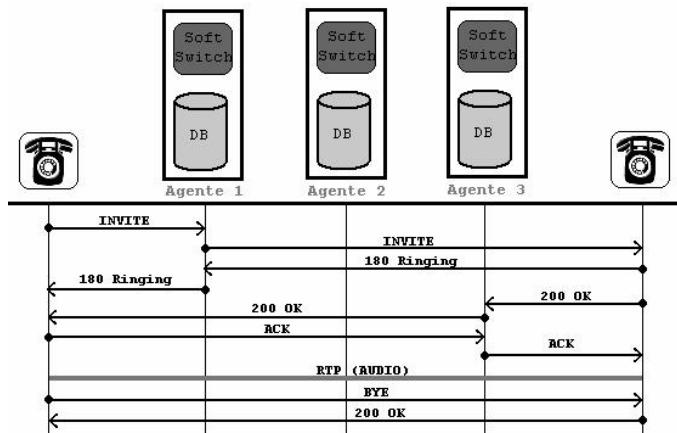


Figura 8. Flujo de llamada entre usuarios de la red.

En el diagrama de la Figura 8 se observa que el proxy no mantiene el estado de la sesión (stateless proxy) lo cual significa que la sesión se establece sin importar el agente redundante que recibe las peticiones SIP.

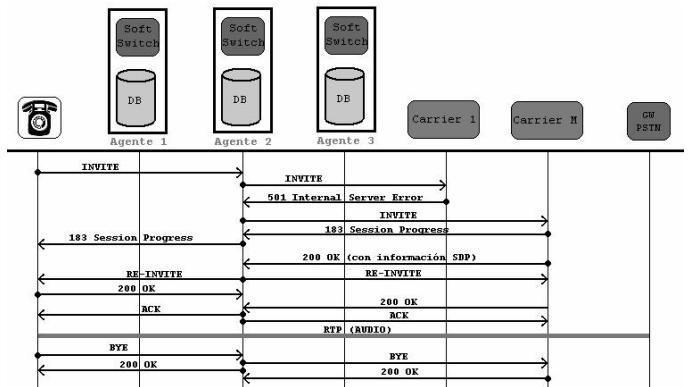


Figura 9. Flujo de llamada hacia la PSTN.

En el ejemplo de la Figura 9, el proxy mantiene el estado de la sesión (*stateful proxy*) ya que esta es redireccionada internamente al B2BUA que realiza el enrutamiento a las redes PSTN y ejecuta el algoritmo de redundancia de proveedores así como los scripts AAA. Un B2BUA es un dispositivo SIP que recibe peticiones como un usuario agente servidor (UAS) y las procesa como un usuario agente cliente (UAC), por lo tanto por cada sesión que recibe se genera una nueva sesión hacia el destino. Este comportamiento es extremadamente útil para realizar tareas de contabilidad en tiempo real, autorización de sesiones dependiendo de datos instantáneos como saldos o permisos y para proteger información reservada como la dirección IP de autenticación con el carrier y los prefijos de marcación.

En los dos casos anteriores se puede observar que la sesión de audio inicializada no es comprometida por fallos en la infraestructura ya que el camino de la información RTP (Real Time Protocol) [12] está definido entre las direcciones IP de la fuente y el destino sin pasar en ningún momento por los elementos redundantes de la infraestructura. En el caso particular de una sesión de voz hacia la red PSTN, si se presenta una falla en el agente redundante que inicia esta sesión, se procede a generar un registro de falla de la sesión y se corrige la contabilidad una vez se reciban los reportes CDR (Call Detail Records) del proveedor.

Para realizar el cálculo de la disponibilidad del sistema se debe asumir que el tiempo para que falle un sistema (MTTF - Mean Time to Failure) es mucho menor que el tiempo para que se repare (MTTR - Mean Time to Recovery). Debido a que se poseen agentes redundantes en los cuales la base de datos y el softswitch comparten el mismo servidor la disponibilidad del sistema es $(1-(1-Da)^A)$ donde Da es el factor de disponibilidad de cada agente y A es el número de agentes presentes. Por lo tanto, la arquitectura garantiza disponibilidad de *cinco nueves* si se tienen tres agentes redundantes cada uno con una disponibilidad de al menos 98% (esto es 3.6 días de tiempo muerto al año) lo cual en términos prácticos es bastante

deficiente para un servidor dedicado dado que los proveedores de este tipo de servicios garantizan por contrato un mínimo de 99.9%, así con este último factor bastarían dos agentes redundantes para alcanzar *cinco nueves* de disponibilidad.

VII. CONCLUSIONES

En este artículo se ha presentado una arquitectura que permite ofrecer servicios de telefonía IP sobre Internet utilizando el protocolo SIP añadiendo a la infraestructura capacidades de distribución de carga y brindando escalabilidad de una forma sencilla para permitir mayor capacidad y alta disponibilidad del servicio.

Al segmentar el problema en tres grandes frentes la infraestructura resultante se vuelve más tolerante a fallos en el momento de la desconexión de uno o varios nodos de la red ya que los clientes SIP tienen acceso ininterrumpido a alguno de los agentes redundantes gracias a que se aplican las técnicas DNS SRV RR para protocolo SIP. La información propia del servicio está disponible y actualizada en todas las bases de datos debido al esquema escalable de réplicas planteado. En el evento que alguno de los proveedores de terminación en la PSTN presente fallos, los cuales no se pueden controlar desde el punto de vista de la infraestructura del ITSP, el sistema está en capacidad de encaminar la llamada utilizando los algoritmos de redundancia de proveedores descritos.

Se concluye finalmente que introduciendo estos desarrollos en la infraestructura de un proveedor de servicios de telefonía IP (ITSP) este puede mejorar la capacidad de acceso, la escalabilidad, la tolerancia a fallos, los porcentajes de terminación de llamadas y por lo tanto la adopción de este tipo de telefonía como natural sustituto tecnológico de la telefonía tradicional por commutación de circuitos.

REFERENCIAS

- [1] Cisco, 2000. High-availability solutions for sip enabled voice-over-ip networks. White Paper.
- [2] Knight, S., Weaver, D., Whipple, D., Hinden, R., Mitzel, D., Hunt, P., Higginson, P., Shand, M. and Lindem, A., 1998. Virtual router redundancy protocol. Request for Comments 2338. <ftp://ftp.ietf.org/rfc/rfc2338.txt>.
- [3] Ohlmeier, N., 2003. Design and implementation of a high availability sip server architecture. Diplomarbeit, Technische Universität Berlin, Berlin.
- [4] Rosenberg, J., Schulzrinne, H., Camarillo, G., Johnston, A., Peterson, J., Sparks, R., Handley, M. and Schooler, E., 2002. Sip: Session initiation protocol. Request for Comments 3261, Junio 2002. <ftp://ftp.ietf.org/rfc/rfc3261.txt>.
- [5] Singh, K. and Schulzrinne, H., 2004. Peer-to-peer internet telephony using sip. Department of Computer Science, Columbia University, New York.
- [6] Stoica, I., Morris, R., Karger, D., Frans Kaashoek, M. and Balakrishnan, H., 2001. Chord: A Scalable Peer-to-peer Lookup Service for Internet Applications. MIT Laboratory for Computer

Science. San Diego, CA.

- [7] Singh, K. and Schulzrinne, H., 2006. Failover, load sharing and server architecture in sip telephony, Department of Computer Science, Columbia University, New York.
- [8] Gulbrandsen, A., Vixie, P. and Esibov, L., 2000. A DNS RR for specifying the location of services (DNS SRV). Request for Comments 2782. <ftp://ftp.ietf.org/rfc/rfc2782.txt>.
- [9] Schulzrinne, H. and Rosenberg, J., 2002. Session initiation protocol (sip): Locating sip servers. Request for Comments 3263. <ftp://ftp.ietf.org/rfc/rfc3263.txt>.
- [10] MySQL, 2009. Open Source SQL server, <http://www.mysql.com>.
- [11] Singh, K. and Schulzrinne, H., 2004. Failover and load sharing in sip telephony. Reporte Técnico. CUCS-011-04, Columbia University, Computer Science Department, New York, NY, USA.
- [12] Schulzrinne, H., Casner, S., Frederick, R. and Jacobson V., 1889. RTP: A Transport Protocol for Real-Time Applications. Request for Comments 1889. <ftp://ftp.ietf.org/rfc/rfc1889.txt>.

Andrés Parra Londoño Ingeniero Electrónico de la Universidad Autónoma de Occidente, Cali, Colombia (2004). Estudiante del Programa de Maestría en Ingeniería con Énfasis en Electrónica en la Universidad del Valle, Cali, Colombia. Es miembro del grupo de investigación en sistemas de telecomunicaciones de la Universidad del Valle (SISTEL-UV). Actualmente se desempeña como Gerente de Telefonía y Telecomunicaciones de Ipsofactum S.A E.S.P. y es jefe de desarrollo de la infraestructura de VoIP de esta compañía. Sus áreas de interés son: Telefonía IP, compatibilidad de sistemas telefónicos NGN con plataformas SS7 y sistemas distribuidos aplicados a la prestación de servicios de VoIP.

Fabio G. Guerrero Ingeniero en electrónica y telecomunicaciones de la Universidad del Cauca, Popayán, Colombia (1992). M.Sc. Real Time Electronic Systems, Bradford University, United Kingdom (1995). Actualmente se desempeña como profesor asistente de la Escuela de Ingeniería Eléctrica y Electrónica de la Universidad del Valle (Cali, Colombia) en el Área de Telecomunicaciones. Coordinador del grupo de investigación en sistemas de telecomunicaciones de la Universidad del Valle (SISTEL-UV). Sus áreas de interés son: redes de próxima generación, sistemas de comunicaciones móviles y comunicación digital.