

Análisis e Implementación de un Sistema de Respuesta Interactiva (IVR) Virtual en el lenguaje abierto Preprocesador de Hipertexto (PHP) sobre Linux para el voceo de audios que indique el saldo de los abonados prepagos alojados en una red inteligente a través del protocolo de Señalización parte de usuario de Red Digital de Servicios Integrados (ISUP)

Raúl Antonio Camas Oviedo ⁽¹⁾ MSc. José Miguel Menéndez Sánchez ⁽²⁾
Facultad de Ingeniería en Electricidad y Computación
Escuela Superior Politécnica del Litoral
Campus Gustavo Galindo Km 30.5, vía Perimetral
Apartado 09-01-5863, Guayaquil, Ecuador
rcamas@espol.edu.ec ⁽¹⁾ jmenende@fieec.espol.edu.ec ⁽²⁾

Resumen

El artículo a continuación presenta la información correspondiente a la implementación de un Sistema de Respuesta Interactiva, el cual es capaz de ejecutar el voceo de audios correspondientes a un servicio de Prepago de una Red Celular simulada, desarrollado en el lenguaje Preprocesador de Hipertexto (PHP) sobre el sistema operativo de código abierto Linux. La interfaz del IVR está desarrollada bajo el protocolo de Red Digital de Servicios Integrados (ISDN por sus siglas en inglés) Parte de Usuario (ISUP), con el objetivo de brindar una interfaz de consulta comprensible y cómoda para los abonados, donde puedan realizar sus consultas de manera eficiente. El proyecto es parte de una serie de proyectos que componen una red creada con protocolos del Sistema de Señalización Número 7 (SS7). Se busca validar cada uno de los tipos de mensaje que ocurren en una transacción de ISUP, junto con una notificación automatizada por cada mensaje recibido. El artículo busca además denotar la importancia del protocolo a tratar, ISUP, dentro de redes del sistema SS7, en donde este protocolo es fundamental para el intercambio de llamadas desde dos abonados o con sistemas automatizados de respuesta como el que se trata en el artículo presente.

Palabras Claves: ISUP, PHP, SS7, IVR, Señalización.

Abstract

The following article presents the information corresponding to the implementation of an Interactive Voice Response (IVR), which is capable of executing the voicing of corresponding audios belonging to the Prepaid Service of a simulated Mobile Network, designed in the Hypertext Pre-Processor (PHP) language on the Open-Source operating system Linux. The IVR's interface is designed under the Integrated Services Digital Network (ISDN) User Part (ISUP) protocol, with the objective of offering a comprehensible and comfortable inquiry interface for subscribers, where they can make their inquiries in an efficient way. This project is part of a series of projects that compose a network created with protocols belonging to the Signaling System Number 7 (SS7). It seeks to validate each of the message types that happen during an ISUP transaction, complete with an automated notification for each received message. The article seeks as well to denote how important the protocol in question, ISUP, is inside the SS7 system's networks, where this protocol is essential for call exchange from two subscribers or with automated systems like the one reviewed in this article.

Keywords: ISUP, PHP, SS7, IVR, Signaling.

1. Introducción.

Como finalidad principal, este proyecto pretende explicar y demostrar el funcionamiento exitoso del Sistema Interactivo de Respuesta (IVR por sus siglas en inglés) a través del protocolo de Red de Servicios Integrados de Parte de Usuario (o ISUP, por sus siglas en inglés).

Para explicar dicho funcionamiento es necesario presentar cada uno de dichos mensajes que pueden ocurrir en esa transacción, y sus diversas interacciones con otros protocolos e impacto en el sistema generalizado. Más precisamente, se estudia la transacción entre este sistema y una central móvil. Se muestra el proceso de consulta a la base de datos implementada por el sistema prepago, de donde se obtiene la información necesaria para dar una respuesta al abonado, del mismo modo con otros tipos de consulta solicitados por el mismo.

Al mismo tiempo, se beneficia a la sociedad en general, ya que las empresas creadas por instituciones públicas de educación superior pueden atender sectores del país en los cuales no hay cobertura de los distintos servicios de telecomunicaciones por parte del Estado.

2. Justificación del desarrollo de una IVR virtual.

Con la creciente afluencia de los sistemas de telecomunicaciones, en especial la telefonía celular, cada día aumentan más el número de abonados suscritos a estos tipos de servicio, por lo que surge la necesidad de crear un sistema automatizado de consultas de servicios, saldos y promociones, parte de la séptima capa del modelo OSI, interactuando directamente con el usuario a través del teclado del equipo por medio de Multi-Frecuencias de Tono-Dual (DTMF, por sus siglas en inglés).

Es por eso que se considera necesario el desarrollo de aplicaciones de sistemas IVR sobre el protocolo ISUP, permitiendo expandir el alcance del desarrollo de este tipo de aplicaciones dentro de sistemas de señalización más extensos. El lenguaje PHP facilita el desarrollo de aplicaciones en la Web, debido a su flexibilidad y potencia para desarrollo. Mayor parte de los servidores web de licencia libre tienen capacidad para interpretar lenguajes de script, entre ellos PHP.

En el Ecuador, las operadoras de telefonía móvil, Claro, Movistar y la Corporación Nacional de Telecomunicaciones todas constan de un sistema automatizado de respuesta de voz interactiva. Este trabaja sobre protocolos dentro del Sistema de Señalización Número 7 (SS7) comunicándose con una estación móvil para poder acceder a la base de datos del servicio a consultar, que en la presente, será un sistema de Prepago. El IVR y la central se

comunican mediante una transacción ocurrente en el protocolo de ISUP, de donde se establecen las solicitudes iniciales: El abonado que desea realizar la consulta, tras acceder a la IVR recibe el mensaje de que debe escoger una de las opciones disponibles marcando un número asignado del teclado de su equipo, este se detecta en el sistema analizando la frecuencia recibida de la matriz DTMF. Dependiendo de la solicitud realizada, el sistema consulta la base de datos de Prepago a través de mensajes del protocolo de Lógica Mejorada de Aplicaciones Personalizadas para Redes Móviles (o CAMEL, por sus siglas en inglés), recibiendo la información y procesándola de la base de datos para así recibir un vector de coordenadas de la matriz DTMF, denotando el archivo de audio a reproducir por el sistema para acceso del abonado.

El proyecto se enfoca en la transacción ISUP entre la central y el sistema a desarrollar, el cual se logra mediante una aplicación escrita en lenguaje PHP, sobre un sistema operativo "Open Source", en el presente caso, se trabaja sobre el sistema operativo de código abierto Linux. En el transcurso del documento se detalla el uso del servidor XAMPP (X, para cualquiera de los diferentes sistemas operativos, Apache, MySQL, PHP, Perl) que se utiliza para implementar la simulación, el cual consiste en un servidor web (Apache) y la base de datos de MySQL (Lenguaje de Consulta Estructurado por sus siglas en inglés) para la interpretación del lenguaje PHP. Se ha escogido este servidor por poseer licencia de carácter libre, eliminando la preocupación por el costo de alguna licencia de software.

3. El Sistema de Señalización No. 7 (SS7)

SS7, Sistema de Señalización por Canal Común No. 7, es un conjunto de protocolos que cumplen una función muy importante en las redes telefónicas: el establecimiento y la respectiva finalización de llamadas. Previo a los inicios de la señalización por canal común, se trabajaba con señalizaciones "en banda" para todas las redes telefónicas. Estas requerían de un troncal para poder transportarse de un extremo hacia el otro. En la Señalización de Canal Asociado (o CAS por sus siglas en inglés), la información de señalización para una troncal era llevada por la troncal propiamente. En la Señalización de Canal Común (o CCS por sus siglas en inglés) un enlace de señalización común lleva los mensajes para un número de troncales. CCS fue desarrollado para la tercera generación introducida en los años sesenta [1]

3.1 Tipos de Nodos

Existen 3 tipos de nodos en una red de señalización SS7:

- Punto de transferencia de señal (STP)
- Punto de conmutación de servicio (SSP)
- Punto de control de servicio (SCP)



Figura 1. Tipos de Nodos

Existen 3 tipos de nodos en una red de señalización SS7:

STP es un switch de paquetes donde el tráfico de red entre puntos de señalización podrían ser encaminadas. Una STP encamina cada mensaje entrante a una dirección de señalización saliente basada en información de ruteo dentro del mensaje SS7.

SSP es un switch que se encarga de la inicialización, duración, terminación y redirección de llamadas. Una SSP envía mensajes de señalización a otros SSP para manejar o configurar circuitos de voz necesarios para ejecutar una llamada.

SCP es una base de datos centralizada que determina la ruta de una llamada tras ser notificado por la SSP a través de un mensaje. La SCP responde a la SSP enviándole la información del número al cual se marcó (Número B) [2]

3.2 Estructura

Fuertemente bajo la influencia del modelo de Interconexión de Sistemas Abiertos (OSI, por sus siglas en inglés), la red SS7 está constituida por varios protocolos, los cuales se agrupan en cuatro niveles principales, que vendrán a ser equivalentes a las capas del modelo OSI. Dichos niveles son:

- Nivel 1: Nivel físico
- Nivel 2: Nivel de enlace de datos
- Nivel 3: Nivel de red
- Nivel 4: Nivel de partes de usuario

Los tres primeros niveles de la red son constituidos por un solo protocolo, el protocolo de Parte de Transferencia de Mensajes (MTP por sus siglas en inglés), que es el principal responsable del tráfico de los mensajes de la red SS7 y provee los servicios de transferencia de mensajes para los usuarios. Este protocolo se constituye de tres niveles, cada uno formando parte de los tres primeros niveles de la arquitectura de SS7. MTP de niveles 1, 2 y 3 correspondientes a los niveles Físico, Enlace de Datos y Red, respectivamente. El cuarto nivel de SS7, llamado el nivel de Parte de Usuario está constituido por diversos protocolos.

- Parte de usuario de telefonía (TUP)
- Parte de usuario RSDI (ISUP)
- Parte control de conexión de señalización (SCCP)

- Capacidades de transacción (TC)
- Entidades de aplicación (AE)
- Elementos de servicio de aplicación (ASE)

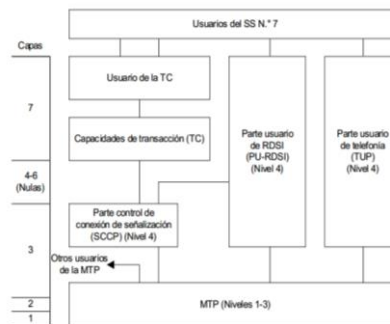


Figura 2. Estructura de SS7

4. El Protocolo Parte de Usuario de Red de Servicios Integrados (ISUP)

ISUP es utilizado tanto para llamadas que son parte de la Red Integrada de Servicios como para llamadas que no son parte de dicha red. ISUP es basado en el protocolo telefónico de parte de usuario (TUP) el cual también forma parte del cuarto nivel de SS7, el nivel de Partes de Usuario, podría decirse que ISUP es básicamente un TUP compatible con redes inteligentes de servicios integrados [3]

La señalización de ISUP ocurre durante llamadas cualquiera, sean o no parte de una Red Inteligente de servicios. En una llamada básica ocurren intercambios de mensajes de notificación de inicialización, los cuales deben ser contestados con un aviso de respuesta. Estos mensajes contienen datos a ser entregados al destino, cada uno abarcando una determinada cantidad de bits, estos pueden enviarse ya sea en un “flujo” de cifras binarias o en una sola cadena hexadecimal [4]

La llamada transcurre en cinco pasos específicos, en los cuales ocurren los intercambios de los mensajes de señalización indicados junto a cada paso:

1. Envío del mensaje de inicialización (IAM)
2. Confirmación de dirección enlazada (ACM)
3. Confirmación de respuesta (ANM)
4. Desconexión del circuito (REL)
5. Confirmación de desconexión (RLC)

En ciertos casos, la llamada suele ser a un terminal asignado por la correspondiente operadora para servicios como, consultas, promociones, conexión a centros de servicio, entre otras. Estos tipos de llamada tienen un flujo con algunas diferencias al flujo convencional de mensajes de una llamada normal.

4.1 Estructura de los mensajes

Cada uno de los mensajes que se envían o reciben en ISUP cuentan con un determinado número de parámetros, estos incluyen identificadores del tipo de mensaje a enviar, información del circuito, entre otros. Todo mensaje de ISUP contiene en su formato la siguiente información predeterminada [5]:

- **Código de Identificación del Circuito (CIC):** Identifica el circuito de voz relacionado al mensaje.

- **Tipo de Mensaje:** Contiene una cifra que determina el tipo de mensaje ISUP relacionado.

- **Parte Fija Obligatoria:** Constan de todos los parámetros de mensaje de una longitud fija.

- **Parte Variable Obligatoria:** Contiene parámetros de longitud variable, sus variables serían longitud del parámetro y su contenido.

- **Parte Opcional:** Campos opcionales que pueden incluirse, como nombre del parámetro y su longitud. Estos no son obligatorios [6]

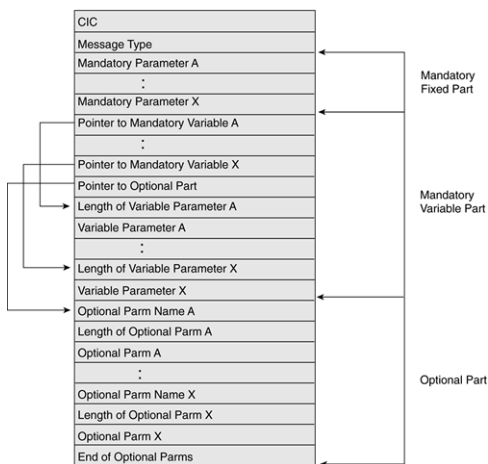


Figura 3. Estructura del mensaje ISUP

4.2 Mensajes básicos de la llamada

La sección a introducirse contiene los parámetros obligatorios en cada uno de los mensajes básicos de una transacción revisados en la sección anterior, junto con un diagrama que muestra dichos parámetros. Dichos parámetros se encuentran establecidos y debidamente documentados en la recomendación Q.763 de la ITU-T.

IAM contiene la información suficiente necesitada para establecer la llamada. Es el mensaje más grande puesto a que es compatible con varios parámetros opcionales que pueden ser incluidos. IAM es un mensaje de Tipo 1.

ACM se envía por el nodo de destino para notificar la satisfactoria conectividad de un circuito de voz ya reservado, tras haber recibido antes un IAM. ACM también señala información sobre el estado de la llamada a través de indicadores de llamada hacia atrás (BCI por sus siglas en inglés), que son contrarios a los parámetros FCI contenidos en el IAM. ACM es un mensaje de tipo 6.

ANM se envía una vez la persona llamada descuelgue el terminal para responder a la llamada. Este mensaje contiene ciertos parámetros opcionales, sin embargo no contiene parámetros obligatorios a más de su tipo de mensajes. ANM es de tipo 9.

REL notifica la liberación del circuito. Este tipo de mensaje puede ser enviado en cualquier dirección del circuito, ya sea del llamante a quien reciba o viceversa. Contiene parámetros obligatorios llamados Indicadores de Causa, que denotan el motivo de la liberación de la llamada. REL es un mensaje de tipo 12.

RLC actúa como el acuse de recibo de un REL. Una vez se ha receptado, el circuito entra en estado de libre. RLC es un mensaje de tipo 16.

5. Sistema de Respuesta Interactiva (IVR)

IVR, de las siglas Respuesta de voz interactiva en inglés, hace referencia a aplicaciones interactivas entre los dispositivos móviles, teniendo como entrada tonos o voz según sea el caso de implementación. Es capaz de interactuar con el ser humano a través de grabaciones de voz, obteniendo información del mismo para cumplir a cabalidad ciertos procesos [7]

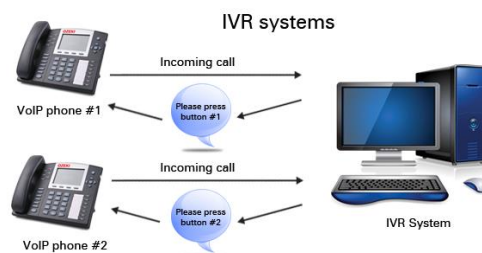


Figura 4. Funcionamiento de un IVR

El móvil realiza una llamada un número de consulta proporcionado por la operadora. El sistema IVR recibe ese número y en respuesta al usuario envía grabaciones de una base de datos disponible para la transacción. El usuario recibe esta información y procede a responder con las teclas de marcado necesarias para continuar el proceso correspondiente, en nuestro caso la consulta de saldos de su línea telefónica.

6. Diseño de la red y componentes

Cada una de las máquinas está configurada para ser parte de una pequeña red en donde se muestra el comportamiento de un sistema de señalización, estas poseen una interfaz donde se puede apreciar el intercambio de mensajes y sus respectivos parámetros en tiempo real. Todos los cambios realizados y archivos se actualizan y están disponibles para cada equipo de esta pequeña red. Los ordenadores deben tener instalado un sistema operativo basado en Linux. Para el presente proyecto se utiliza la distribución Ubuntu.

6.1 Software

XAMPP es un servidor web el cual es libre e independiente de alguna plataforma. Se trata de una distribución sin costo alguno del servidor web Apache, mismo que es de código abierto y trabaja sobre HTTP (Protocolo de Transferencia de Hipertexto por sus siglas en inglés), y también contiene la base de datos MySQL (Mi Lenguaje Estructurado de Consulta por sus siglas en inglés), y es capaz de interpretar “scripts” de los lenguajes de programación PHP y Perl. [8]

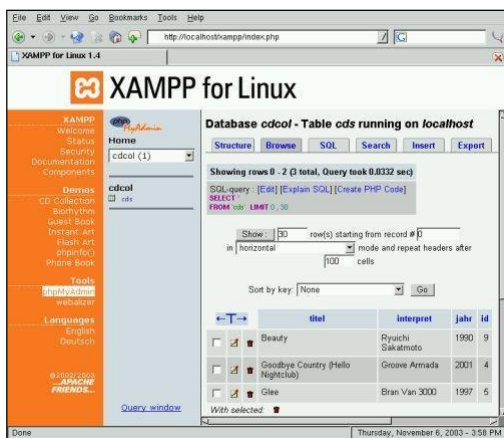


Figura 5. Sitio principal XAMPP

NetBeans es un Entorno de Desarrollo Integrado (IDE por sus siglas en inglés) cuyo propósito principal es el desarrollo en el lenguaje Java, concebido en conjunto por Sun Microsystems y Oracle. Es un proyecto fundado en el año 2000 y que hoy en día cuenta con casi 100 socios alrededor del mundo, junto con una creciente comunidad de usuarios.

6.2 Hardware

El componente correspondiente a IVR se encuentra implementado en una máquina, esta interactúa con otras dos máquinas mediante dos protocolos distintos. IVR ejecuta transacciones con la central MSC mediante ISUP, y con la base de datos del sistema prepago a través del protocolo Parte de Aplicación de Red Inteligente (INAP, por sus siglas en inglés).

La máquina donde se simula una IVR cuenta con las siguientes especificaciones:

- Modelo: Dell Inspiron 14R 5420
- Procesador: Inter Core i5-3210M 2.5 GHz
- RAM: 6.00 GB HDD: 500 GB
- Sistemas operativos: Windows 8.1 Pro. Ubuntu LTS 14.04

La máquina se encuentra conectada a una red inalámbrica de área local (WLAN) personalizada utilizando un pequeño enrutador inalámbrico marca D-Link, de donde se conectan los equipos.

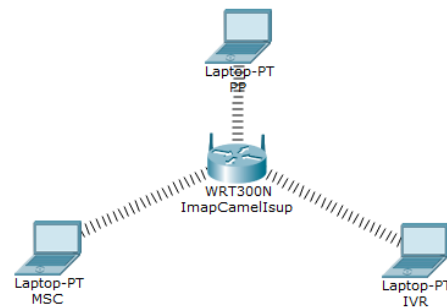


Figura 6. Ejemplo de topología inalámbrica con tres máquinas

6.3 Implementación

Cada mensaje se programa en lenguaje PHP en el IDE NetBeans el cual debe ser instalado en Linux para posibles ediciones del proyecto, con su respectiva cabecera y funciones especiales. El proyecto posee una carpeta especial de archivos de “utilidades”.

conversiones.php realiza conversiones de números, con funciones para convertir cadenas binarias o decimales a hexadecimales. **enviar_mensaje.php** controla el envío de mensajes nuevos desde el IVR y sus respectivos acuses de recibo. **ips.php** captura las direcciones IP de cada máquina al momento de necesitarse su declaración. **parametros.php** captura los parámetros de los mensajes alojados en la base de datos del proyecto. **registrar.php** declara el registro de intercambios en un archivo .log creado para el proyecto.

El **IAM** es recibido desde la MSC (Centro de Conmutación Móvil por sus siglas en inglés) como una trama hexadecimal con su respectivo CIC y números de llamante y llamado. Se registra la trama y se convierte la trama a binaria para realizar la sustracción de los parámetros de interés para la transacción (se utiliza la función substr). Luego de extraer cada dígito de los dos números involucrados estos se concatenan en una nueva variable a ser enviada a través de nuevos mensajes declarados para la parte de INAP. Los mensajes **ACM** y **ANM** son enviados a MSC con la

sentencia `enviar_nuevo` declarada en `enviar_mensaje.php`. Se crea un lazo `for` donde se declara la dirección URL que direcciona a los archivos de audio alojados en IVR, donde se involucran la IP de la máquina y el nombre de la pista recibido desde Prepago a través de INAP.

Se define como **timeout** al tiempo transcurrido tras la reproducción del audio en el que el IVR debe cerrar la llamada automáticamente. Cuando se recibe el `timeout` de parte de MSC significa que el audio terminó de reproducirse, y se debe devolver un mensaje REL para culminar la llamada. Se programa también el recibimiento de un RLC que confirme la liberación de la llamada.

En el archivo **DTMF.php** se crea un diagrama de estados a través de la sentencia `switch`, donde se declaran los casos para el ingreso de teclas 1 o 2. INAP procesa este ingreso para solicitar un servicio a Prepago. De no cumplirse ninguno de los dos casos, el audio se reproducir nuevamente por lo que se declara el lazo de las pistas similar al del mensaje IAM. Al llegar al estado de liberación de la llamada (REL), se envía el mensaje, o de lo contrario se vuelve a solicitar el ingreso de teclas.

```

$estado = $fila['estado'];
$cadenaDTMF = $fila['cadena_dtmf'];
$spistas = null;
//$rel = false;
switch($estado)
{
  case "282":
    if ($tecla==1)
    {
      $spistas= enviar_nuevo("ivr/inap", get_parametro_ip_ivr_inap(), "dtmf1",array("tecla"=>$tecla));
      $estado = "rel";
      // $estado = "282_1";
    }
    else if($tecla == 2)
    {
      $spistas= enviar_nuevo("ivr/inap", get_parametro_ip_ivr_inap(), "dtmf2",array("tecla"=>$tecla));
      $estado = "282_2";
    }
    else
    {
      // enviar_respuesta(array("reproducir otra vez url 1,2,3,14"));
      $spistas = array(182, 183, 2);
    }
    break;
}

```

Figura 7. Extracto del código de DTMF

6.4 Simulación

La máquina utilizada fue asignada con la dirección IP 192.168.0.113 para conexión alámbrica al enrutador, y 192.168.2.113 para la conexión inalámbrica que le permite a las centrales reproducir audios al llegar al equipo necesario.

El sistema completo utiliza dos centrales, una para la ciudad de Quito y otra para Guayaquil, cada una posee un duplicado de IVR para reproducir audios correctamente al realizarse llamadas al *282 desde cualquier ciudad. El código desarrollado permite al sistema procesar el mensaje de inicialización de la transacción (IAM) y devolver los correspondientes mensajes de confirmación (ACM y ANM) y las direcciones de la red para acceder a los archivos de audio a ser reproducidos por el simulador. IVR contiene alojado en su sistema de archivos los audios correspondientes al servicio de consulta de saldos y recargas.

Una interfaz ha sido diseñada para mostrar la señalización que cursa entre las partes presentes del sistema. La simulación en cualquiera de los escenarios inicia con la marcación del *282 desde la máquina del celular, desde tras solicitarse un ingreso, el usuario marca ya sea la opción 1 o 2, dependiendo de qué es lo que se desee hacer. Tras escucharse el respectivo audio, e ingresarse por teclado la opción correspondiente, al usuario se solicita digite el código de tarjeta seguido del signo # para el caso de opción 2. En cada caso la simulación de la señalización es mostrada en la interfaz.

Este es muy útil para detectar errores en el momento de realizar pruebas. IVR emplea intercambio ISUP para devolver las direcciones correspondientes a los audios a reproducir, y se encarga de terminar la llamada de no receptarse ingreso alguno del teclado.

Estas se muestran con una flecha de línea recta continua apuntando a la dirección del mensaje, junto con un recuadro que muestra el nombre del mensaje respectivo. Toda la señalización correspondiente al protocolo ISUP se encuentra presente en la interfaz, incluyendo el envío y la recepción del marcado desde el teclado y el direccionamiento al audio del equipo.

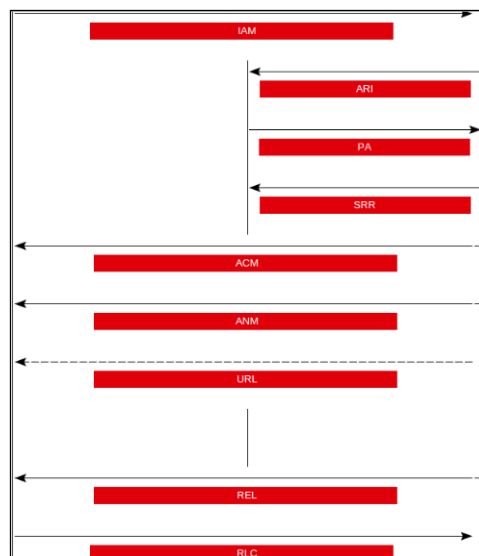


Figura 8. Señalización ISUP como se muestra en la aplicación web

Si se opta por dar un clic sobre el nombre de cada uno de los mensajes, un recuadro pequeño se muestra con los parámetros del mensaje. Para ISUP, lo que se muestra es la respectiva trama hexadecimal para los mensajes de IAM, ACM y ANM, los cuales contienen información tal como el Identificador de Circuito (CIC), códigos de punto de origen y destino (OPC y DPC), y en el caso del IAM, los números de llamante y llamado (*282 en estos casos).



Figura 8. Interfaz del mensaje IAM

Se deben tener bien claros cuáles son los posibles conflictos a suceder dentro de la simulación de la transacción. Antes de tratar de operar la aplicación web, se considera necesario tener conocimientos acerca de los protocolos involucrados directamente con el IVR, principalmente el protocolo INAP, que se comunica con el sistema de Prepago para solicitar información de su base de datos alojada. No estaría de más mencionar la operación de la aplicación del celular, desde donde se realizan los ingresos por teclado para el IVR y desde donde se accede al mismo, proyectos que van de la mano y por lo tanto se necesita conocimiento de su funcionamiento para mejor comprensión del comportamiento del equipo y la aplicación.

7. Conclusiones

Aunque en el presente proyecto se utilizó con la finalidad de comunicarse con un sistema virtualizado, por lo que únicamente pasan archivos de audio por el circuito de voz, ISUP opera de modo correcto para establecer la conexión de voz con un IVR, el cual brinda los servicios de modo eficaz y satisfactorio.

El sistema IVR se implementó con éxito dentro de una red telefónica virtual, funcionando bajo los protocolos establecidos. Este se ha conectado a una red correspondiente al servicio de una ciudad (Quito o Guayaquil) y utiliza mensajes de ISUP para comunicarse con su respectiva Central Móvil, desde donde llegan las solicitudes del usuario. Este IVR brinda los servicios de consulta de saldos y recarga de los mismos.

El sistema emplea todos los mensajes de señalización acordes a la estandarización ISUP, en todos los casos pertinentes se envían y reciben los mensajes correspondientes a cada caso estudiado, los cuales también son mostrados e identificados en la interfaz. Este protocolo es capaz de establecer con éxito cualquier tipo de llamada siendo de vital importancia para cualquier red móvil de telefonía

8. Recomendaciones

1. Se recomienda tener un conocimiento intermedio de Linux para su correcto manejo, principalmente comandos de asignación de permisos de directorios, instalaciones de software y ejecutables, en especial en la distribución de Ubuntu, sobre donde se ha trabajado el proyecto.
2. Tener un conocimiento avanzado del lenguaje PHP para manipular sin problemas los ficheros de cada mensaje, poder detectar rápidamente algún conflicto que se genere con algún archivo y resolverlo de igual manera.
3. Al trabajar dentro de la red correspondiente al sistema, siempre descargar cambios realizados de archivos relacionados al IVR a través de la herramienta Update de la subversión en NetBeans, y también subir los cambios hechos en el mismo equipo para la disponibilidad de los otros equipos a través de un Commit.
4. No solo limitarse a los respaldos en la nube hechos al repositorio, también es altamente recomendable realizar más respaldos del código fuente de todos los mensajes y utilidades ya sea en alguna memoria física o bien en otros almacenamientos en nube, dese el caso de que se pase por alto algún respaldo en el repositorio, o si los cambios realizados para una prueba no resulten efectivos y se altere el funcionamiento del sistema que haya sido funcional en versiones anteriores.
5. Tener sumo cuidado con la conectividad en la topología. Primero realizando una prueba física, viendo si todos los enrutadores o switches se encuentran encendidos y funcionales, y las IP estén asignadas correctamente. Luego ejecutando pruebas de conectividad a través de un ping en el terminal de comandos.
6. Conocer a fondo la operación de cada mensaje de ISUP como IAM, REL, RLC, ACM y ANM, cada uno sus respectivas tramas, se recomienda estudiar las recomendaciones de la ITU acerca de este protocolo y sus mensajes.

9. Bibliografía

- [1] Clase de Signaling System No. 7. Reseña Histórica. EVA. [Online]. Available: https://eva.fing.edu.uy/pluginfile.php/67113/mod_resource/content/2/SS7_2014%20%28presentacion%29.pdf [Accessed: Dic-2014]
- [2] Signaling System 7 (SS7) Definition and Diagram. [Online]. Available: http://www.telecomdictionary.com/telecom_dictionary_ss7_definition.html [Accessed: Dic-2014]

- [3] Pradip Bhuyan's SS7 Tutorial. The SS7 Protocol Stack,
http://www.angelfire.com/indie/pradip_bhuyan/ss7/SS7.html [Accessed: Dic-2014]

- [4] SS7 Overview.
<http://www.techfest.com/networking/wan/ss7.htm> [Accessed: Dic-2014]

- [5] Interwork Connected Number between SIP and SS7,
https://www.dialogic.com/webhelp/BorderNet20/1.1.0/WebHelp/iw_connected_number_sip_ss7.htm [Accessed: Dic-2014]

- [6] Inform IT. ISUP Message Format,
http://www.informit.com/library/content.aspx?b=Signaling_System_No_7&seqNum=66
[Accessed: Dic-2014]

- [7] What is IVR? Interactive Voice Response.
<http://www.aspect.com/glossary/what-is-ivr-interactive-voice-response> [Accessed: Dic-2014]

- [8] Apache Friends, XAMPP.
<https://www.apachefriends.org/index.html>
[Accessed: Ene-2015]