



ESCUELA SUPERIOR POLITÉCNICA DEL LITORAL

Facultad de Ingeniería en Electricidad y Computación

“ANÁLISIS E IMPLEMENTACIÓN EN EL LENGUAJE ABIERTO
PREPROCESADOR DE HIPERTEXTO BAJO LINUX DE UNA CENTRAL DE
CONMUTACIÓN MÓVIL VIRTUAL PARA LA INTERACCIÓN A NIVEL DE
PROTOCOLO DE SEÑALIZACIÓN PARTE DE USUARIO DE RED DIGITAL
DE SERVICIOS INTEGRADOS CON UN SISTEMA DE RESPUESTA
INTERACTIVA”

INFORME DE MATERIA DE GRADUACIÓN

Previa a la obtención del Título de:

INGENIERO EN ELECTRÓNICA Y TELECOMUNICACIONES

Presentado por

Edison Antonio Bayona Murillo
Juan Carlos Marchán Menéndez

Guayaquil - Ecuador

2015

AGRADECIMIENTO

En esta hermosa etapa de nuestras vidas como es la académica hubo muchas personas las cuales día a día nos empujaron a seguir adelante en nuestra lucha hacia el camino del éxito. Muchas gracias:

A Dios por todas sus bendiciones recibidas y por guiarnos día a día en el camino correcto.

A nuestros padres por ser el motor fundamental y nuestro soporte incondicional y quienes siempre nos brindaron todas las facilidades económicas para suplir nuestro estudio.

A nuestros familiares y amigos por siempre estar pendiente de nosotros y una mención especial a nuestros excelentes profesores quienes nos supieron guiar y compartir todas sus vivencias y experiencias.

DEDICATORIA

Hace ya 5 años todo comenzó como un sueño y este sueño ahora se vuelve realidad. Muchas horas de sacrificio y estudio fueron nuestra constante labor. Muchas malas noches y mucho tiempo perdido con la familia se había vuelto parte de nuestras vidas.

Este logro académico que hemos alcanzado se lo dedicamos especialmente a nuestros padres y familiares ya que sin ellos, sin su constante aliento hubiera sido muy difícil poder sobrellevarlo.

Así mismo, el título se lo dedicamos a nuestros amigos politécnicos que en el diario vivir se volvieron más que nuestros compañeros, nuestra propia familia.

TRIBUNAL DE SUSTENTACIÓN

Msc. José Miguel Menéndez

Profesor de la materia de graduación

Msc. Ronald Ponguillo Intriago

Profesor delegado por la unidad académica

DECLARACIÓN EXPRESA

"La responsabilidad del contenido de este Informe, nos corresponde exclusivamente; y el patrimonio intelectual de la misma a la Escuela Superior Politécnica del Litoral".

(Art. 12 del Reglamento de Graduación de la ESPOL)

JUAN CARLOS MARCHÁN MENÉNDEZ

EDISON ANTONIO BAYONA MURILLO

RESUMEN

El presente proyecto simula el funcionamiento de una central de conmutación móvil (MSC) que cuenta con la implementación del protocolo de señalización de la pila SS7 llamado ISUP, para interactuar con un Sistema de Respuesta Interactiva (IVR).

EL proyecto completo fue desarrollado en el lenguaje abierto Preprocesador de Hipertexto (PHP) utilizando diferentes herramientas de programación cuyas principales ventajas, características e instalación se detallan en este documento.

EL resultado final se puede apreciar integrando este proyecto a un sistema que simule el funcionamiento de una red inteligente de telefonía móvil en donde los componentes interactúen a través de ella utilizando los protocolos de la pila SS7.

ÍNDICE GENERAL

AGRADECIMIENTO	II
DEDICATORIA.....	III
TRIBUNAL DE SUSTENTACIÓN.....	IV
DECLARACIÓN EXPRESA.....	V
RESUMEN	VI
ÍNDICE GENERAL.....	VII
ABREVIATURAS	XI
ÍNDICE DE FIGURAS.....	XIII
INTRODUCCIÓN	XV
CAPÍTULO 1	1
1 ANTECEDENTES Y JUSTIFICACIÓN	1
1.1 Antecedentes de ISUP.....	1
1.2 Justificación del uso de ISUP.....	3
1.3 Descripción del proyecto.....	3
1.4 Objetivos	4

1.4.1	Objetivos generales	4
1.4.2	Objetivos específicos	5
1.5	Alcance	5
1.6	Limitaciones	6
CAPÍTULO 2.....		7
2	FUNDAMENTOS TEÓRICOS	7
2.1	Sistemas de Señalización 7 (SS7).....	7
2.1.1	Modalidades de Señalización SS7.....	8
2.2	Redes de Datos	9
2.2.1	Tecnología de Redes	9
2.2.2	Topología de Redes	10
2.3	ISUP en la pila de protocolos SS7	11
2.4	Mensajes básicos ISUP	12
2.5	Flujo de mensajes ISUP.....	13
2.5.1	Descripción de temporizadores.....	15
2.6	Formato de mensajes ISUP	17
2.6.1	Mensaje Inicial de Dirección (IAM, Initial Address Message).....	19
2.6.2	Mensaje de Dirección Completa (ACM, Address Complete Message).....	23
2.6.3	Mensaje de Respuesta (ANM, Answer Message).....	24
2.6.4	Mensaje de Liberación (REL, Release Message)	25

2.6.5	Mensaje de Liberación Completa (RLC, Release Complete Message).....	26
2.7	Otros mensajes ISUP.....	27
2.8	Servicios suplementarios ISUP	28
CAPÍTULO 3.....		30
3	DESARROLLO DEL PROYECTO	30
3.1	Proyecto ISUP aplicado a una MSC	30
3.2	Requisitos para la implementación	31
3.3	Software.....	31
3.3.1	Ubuntu 14.04.1	31
3.3.2	NetBeans IDE 8.0.2	32
3.3.3	JDK	33
3.3.4	XAMPP	33
3.4	Hardware	33
3.5	Proceso de Implementación.....	34
3.5.1	Instalación del JDK	35
3.5.2	Instalación de NetBeans IDE	36
3.5.3	Instalación de XAMPP	37
3.6	Funcionabilidad del sistema implementado	37
3.7	Flujo General para los diferentes escenarios.....	42

3.7.1	Consulta de Saldos *282 opción 1	43
3.7.2	Ingreso de Tarjeta Amigo Exitosa *282 opción 2	44
3.7.3	Ingreso de Tarjeta Amigo Incorrecta *282 opción 2	45
3.8	Topología de la implementación de la Red	46
3.9	Programación en PHP	48
CAPÍTULO 4		53
4	ANÁLISIS DE RESULTADOS	53
4.1	Resultados obtenidos	53
4.2	Interpretación de resultados	54
4.3	Interfaz gráfica de la MSC	60
4.4	Discusión	65
CONCLUSIONES		67
RECOMENDACIONES		70
BIBLIOGRAFÍA		73

ABREVIATURAS Y SIMBOLOGÍA

ACM	Address complete message
ANM	Answer Message
ANSI	American National Standards Institute
CCS	Common Channel Signaling
CIC	Circuit identification code
DPC	Destination Point Code
HLR	Home Location Register
HTTP	Hyper Text Transfer Protocol
IAM	Initial Address Message
ICMP	Internet Control Message Protocol
IP	Internet Protocol
ISDN	Integrated Services Digital Network
ISUP	ISDN User Part
MSU	Message Signal Unit
MTP-1	Message Transfer Part level 1
MTP-2	Message Transfer Part level 2
MTP-3	Message Transfer Part level 3
OPC	Originating Point Code
OSI	Open Systems Interconnection
PSTN	Public Switched Telephone Network

REL	Release message
RLC	Release Complete
SCP	Signaling Control Point
SCCP	Signaling Connection Control Part
SLS	Signaling Link Selection
SP	Signal Point
SS7	Signaling System #7
SSN	Subsystem Number
SSP	Signaling Service Point
STP	Signal Transfer Point

ÍNDICE DE FIGURAS

Figura 2.1 Señalización Asociada.....	8
Figura 2.2 Señalización no Asociada.....	8
Figura 2.3 Señalización quasi Asociada	9
Figura 2.4 Topología Estrella.....	10
Figura 2.5 Topología Anillo	11
Figura 2.6 Flujo de ISUP.....	14
Figura 2.7 Cabecera de enrutamiento de ISUP	17
Figura 2.8 Formato de mensajes ISUP	19
Figura 2.9 Mensaje Inicial de Dirección IAM.....	20
Figura 2.10 Mensaje de Dirección Completa	23
Figura 2.11 Mensaje de Respuesta ANM	25
Figura 2.12 Mensaje de Liberación REL.....	26
Figura 2.13 Mensaje de Liberación Completa RLC	26
Figura 3.1 Computadora portátil	34
Figura 3.2 Instalación JDK.....	35
Figura 3.3 Instalación NetBeans IDE	36
Figura 3.4 Instalación XAMPP	37
Figura 3.5 Diagrama funcional MSC – ISUP.....	41
Figura 3.6 Diagrama de liberación de circuito.....	42
Figura 3.7 Consulta de Saldo *282 opción 1.....	44

Figura 3.8 Ingreso de tarjeta correcto *282 opción 2	45
Figura 3.9 Ingreso de tarjeta incorrecta *282 opción 2	46
Figura 3.10 Topología de la red celular	48
Figura 3.11 Función get parámetro IP	49
Figura 3.12 Programación Mensaje IAM.....	51
Figura 3.13 Programación de Función liberar_canal	51
Figura 3.14 Programación de Funciones Registrar.....	52
Figura 4.1 Logs Consulta de Saldos *282 opción 1a	55
Figura 4.2 Logs Consulta de Saldos *282 opción 1b	56
Figura 4.3 Logs Consulta de Saldos *282 opción 1c	57
Figura 4.4 Logs Ingreso de tarjeta correcta *282 opción 2a	58
Figura 4.5 Logs Ingreso de tarjeta correcta *282 opción 2b	58
Figura 4.6 Logs Ingreso de tarjeta correcta *282 opción 2c.....	59
Figura 4.7 Logs Ingreso de tarjeta correcta *282 opción 2d	59
Figura 4.8 Panel de Control de la aplicación MSC.....	60
Figura 4.9 Interfaz de la señalización MSC visualizada desde la aplicación	61
Figura 4.10 Interfaz Mensaje IAM.....	62
Figura 4.11 Interfaz Mensaje ACM	63
Figura 4.12 Interfaz Mensaje ANM	63
Figura 4.13 Interfaz Mensaje REL	64
Figura 4.14 Interfaz Mensaje RLC	64

INTRODUCCIÓN

Hoy en día vivimos en un mundo regido por la tecnología, en donde las comunicaciones juegan un papel fundamental, ocupando en si gran parte de este mercado que se encuentra constantemente en desarrollo para ofrecer la mayor cantidad de servicios, comodidad, confiabilidad y portabilidad de los dispositivos terminales utilizados para la interacción entre usuarios; dispositivos que con el transcurrir del tiempo se vuelven más complejos y con una serie de características capaces de soportar tantos servicios como las empresas puedan ofrecer. Con el objetivo de poder satisfacer estos requerimientos, no sólo se debe desarrollar los dispositivos terminales, sino también aquellos dispositivos que conforman las redes de telefonía, para que éstos puedan ser capaces de brindar todos los servicios requeridos, por ello las redes actuales son digitales, en donde no sólo se transmite voz, sino también gran cantidad de datos.

Para hacer posible las transmisiones de extremo a extremo se requiere de sobremanera que los equipos de la red se entiendan y comuniquen entre sí, por ello es vital la utilización de protocolos de señalización estandarizados, creados con el fin de controlar y administrar todas las interconexiones de la red. Actualmente se utiliza la pila de protocolos de señalización No7, del cual

nos centraremos en el análisis e implementación de ISUP, el protocolo encargado del establecimiento, gestión y liberación de los circuitos troncales que transportan las llamadas de voz y los datos.

CAPÍTULO 1

1 ANTECEDENTES Y JUSTIFICACIÓN

1.1 Antecedentes de ISUP

Las redes de señalización tiempo atrás eran implementadas haciendo uso de un método denominado Channel Associated Signaling (CAS), que consistía en transmitir voz e información de señalización en el mismo canal. Se utilizaron diversos sistemas capaces de implementar CAS, pero esta señalización conllevaba una serie de limitaciones entre las que destacaba la susceptibilidad al fraude, ya que cualquier persona con previo conocimiento sobre el tema podía alterar la señalización a su conveniencia usando un simple generador de tonos. Así la limitada información de señalización que se podía transmitir y el uso de recursos

ineficientes fueron otros aspectos muy importantes a considerar ya que al solo estar destinada una pequeña porción de la banda de voz del canal, no se podían ofrecer muchos servicios como los que conocemos hoy en día [1].

Debido a la gran cantidad de limitaciones de CAS ante el despliegue tecnológico que se veía venir, se desarrolló un nuevo sistema denominado Common Channel Signaling (CCS), introducido en el año 1976. Este sistema consistía en un canal separado de los canales de voz, dedicado exclusivamente a transmitir datos de señalización para controlar todos los enlaces troncales; además de proveer de gran capacidad a las redes para que puedan transportar una mayor cantidad de información, no solo relacionada a las llamadas, sino también a la prestación de otros tipos de servicios [1]. Para poder utilizar este nuevo sistema, se desarrollaron una serie de protocolos que interactuando entre sí proveían la funcionalidad a la red. Actualmente denominamos a este conjunto de protocolos como Signaling System No. 7, entre los cuales se encuentra ISDN USER PART (ISUP), que es el protocolo de señalización encargado del establecimiento, gestión y liberación de conexiones a nivel de enlaces troncales de la red digital de servicios integrados (Integrated Service Digital Network ISDN) [2].

1.2 Justificación del uso de ISUP

Debido a que este proyecto está enfocado en la implementación del protocolo en una red de telefonía celular actual, ISUP tiene una participación de gran relevancia sobre otros protocolos que cumplen la misma función, siendo utilizado tanto para llamadas ISDN como llamadas no ISDN.

Con el transcurrir del tiempo ISUP ha venido reemplazando al telephone user part (TUP) debido a su gran capacidad para soportar datos dentro de las redes modernas y su mayor robustez frente a TUP que solo puede soportar el servicio de telefonía tradicional. ISUP es fundamental dentro de la simulación de este proyecto debido a que es el encargado de reservar los canales por donde se va a transmitir tanto la voz como datos. Si no se levantara el protocolo ISUP dentro de la red celular no se pudiera establecer un camino para la comunicación. De aquí parte la importancia del uso de nuestro protocolo de señalización.

1.3 Descripción del proyecto

En la actualidad se requiere que los equipos de la red se entiendan y comuniquen entre sí, por ello es vital la utilización de protocolos de señalización estandarizados, creados con el fin de controlar y administrar todas las interconexiones de la red. Para la

implementación de este proyecto se utilizará la pila de protocolos de señalización No7, del cual nos centraremos en el análisis e implementación de ISUP, el protocolo encargado del establecimiento, gestión y liberación de los circuitos troncales que transportan las llamadas de voz y los datos.

Este proyecto básicamente se enfoca en el análisis e implementación en el lenguaje abierto Preprocesador de Hipertexto (PHP) bajo Linux de una Central de Conmutación Móvil (MSC) virtual para la interacción a nivel de protocolo de señalización Parte de Usuario de Red Digital de Servicios Integrados (ISUP) con un Sistema de Respuesta Interactiva (IVR), elemento que presta el servicio de ingreso y consulta de saldos para los abonados prepago de una red celular.

Para que la comunicación y la transmisión de datos se efectúen necesitaremos homologar las tramas a ser enviadas con el fin de que cada protocolo de señalización en particular pueda entenderlas y permitir el tráfico de ellas.

1.4 Objetivos

1.4.1 Objetivos generales

1. Describir el funcionamiento y operación del protocolo ISUP.

2. Describir los principales mensajes de señalización del protocolo ISUP usados en las diferentes funciones del elemento de una red móvil.

1.4.2 Objetivos específicos

1. Implementar una central de conmutación de telefonía móvil, MSC, virtual que emule el comportamiento a nivel de señalización ISUP.
2. Analizar los aspectos técnicos que intervienen en la operación de la señalización ISUP en una MSC.
3. Establecer una compatibilidad con los demás protocolos de señalización SS7.

1.5 Alcance

Este proyecto tiene como alcance describir la funcionalidad de ISUP como protocolo de señalización de las redes de telefonía celular, y explicar la arquitectura de cada mensaje que lo conforma, emulando su comportamiento en una MSC simulada en un software libre como lo es el sistema operativo Linux. Para lograr esto deberemos formar una red de computadoras interconectadas.

Se pretenderá simular en su totalidad el funcionamiento de una red celular tanto como es la consulta de un abonado al *282 como el

establecimiento de una llamada entre dos celulares y los diferentes escenarios posibles que se pueden presentar enfocándonos básicamente en el protocolo de señalización ISUP.

1.6 Limitaciones

Debido a que SS7 utiliza CCS se requiere que los equipos tengan la capacidad de notificar cuando se interrumpe un canal de voz, ya que las conexiones se establecen por el canal destinado a la señalización.

Una de las limitantes de este proyecto es que no contamos con equipos reales para la simulación de las bases celulares cuando se realiza una llamada, verificación o activación de saldo.

También este proyecto no correrá sobre canales reales de comunicación, todo será virtual y simulado usando unas computadoras portátiles con direcciones IP's fijas, asignadas que cumplirán funciones específicas, en este caso la simulación de una MSC que corre el protocolo ISUP.

CAPÍTULO 2

2 FUNDAMENTOS TEÓRICOS

2.1 Sistemas de Señalización 7 (SS7)

SS7 es un conjunto de protocolos que describen una forma de comunicación entre los conmutadores telefónicos en una red de telefonía pública [3]. Es un tipo de señalización de canal común Common Chanel Signaling (CCS). La organización que se encarga de la administración del protocolo es la ITU-T.

Con el transcurrir del tiempo este protocolo se ha expandido de tal manera que cumple con un gran número de tareas relacionadas con la recolección y el reporte de información necesaria para la transmisión de las llamadas telefónicas [3]. Hoy en día es el responsable del ruteo de las llamadas ya sean nacionales o

internacionales y ha tomado un rol predominante en las redes de telefonía móvil.

2.1.1 Modalidades de Señalización SS7

La conectividad dentro de una red SS7 puede llevarse a cabo de distintas formas. Para esto existen distintos tipos de enlaces:

- **Señalización Asociada:** la señalización sigue el mismo camino que los troncales de voz [3].



Figura 2.1 Señalización Asociada

- **Señalización no Asociada:** la señalización viaja entre dos puntos de señal (SP) mediante al menos dos puntos de transferencia de señal (STP) [3].

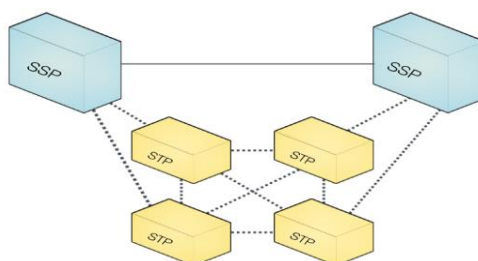


Figura 2.2 Señalización no Asociada

- **Señalización quasi Asociada:** la señalización se da de manera tal que una pareja de SP se conecta a la misma pareja de STP [3].

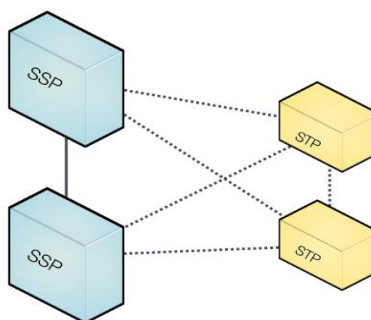


Figura 2.3 Señalización quasi Asociada

2.2 Redes de Datos

Las redes de datos constituyen en la actualidad un apoyo de vital importancia para todas las empresas cuyo éxito dependen del buen manejo de la gran cantidad de información que generan [4].

2.2.1 Tecnología de Redes

Dependiendo del tipo de cobertura, topología, procesamiento que realizan, si son conmutadas o no, tasa de velocidad, se pueden clasificar las redes de la siguiente manera:

- Según su cobertura

- Redes de Área Local (LAN): redes conectadas entre sí dentro de un mismo espacio físico.
 - Redes de Área Metropolitana (MAN): varias redes LAN conectadas entre sí pero que no sobrepasen el ámbito urbano.
 - Redes de Área Amplia (WAN): red que conecta dos o más redes LAN entre ciudades distintas del mismo país [4].
- Redes conmutadas y redes no conmutadas

2.2.2 Topología de Redes

Al hablar de topología nos referimos a la geometría básica de la red, la forma como físicamente están conectadas entre sí. Nos enfocaremos en la descripción tanto de la topología estrella como la topología anillo:

- **Topología Estrella:** consiste en conectar todas las estaciones a un ordenar central que se encarga de controlar la prioridad y procedencia de los mensajes y su distribución [4].

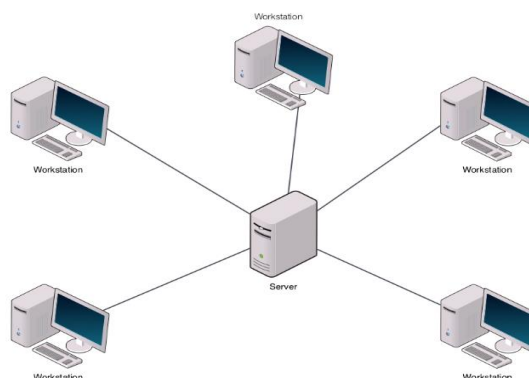


Figura 2.4 Topología Estrella

- **Topología Anillo:** todas las estaciones están conectadas entre sí formando un anillo, se establece una conexión directa entre los dispositivos que conforman la red [4].

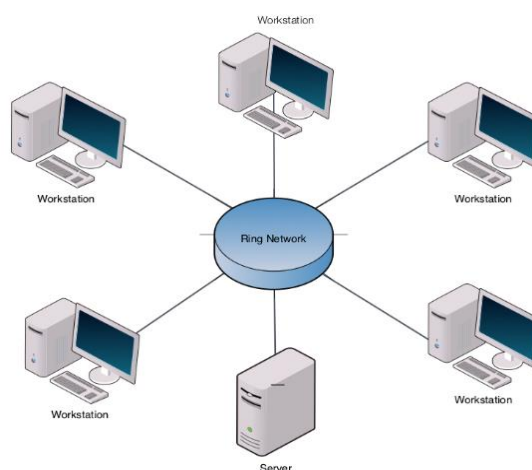


Figura 2.5 Topología Anillo

2.3 ISUP en la pila de protocolos SS7

El SS7 es una tecnología basada en conmutación de paquetes, donde cada uno de estos paquetes contiene un gran número de información muy relevante y necesaria para poder encaminar datos a través de la red sin establecer en sí una conexión directa con el destino. Cabe recalcar que la mayoría de estos procesos son automatizados para lo cual no se necesita mucha intervención del hombre.

El Protocolo para la Parte de Usuario ISDN (ISDN User Part, ISUP) está relacionado a circuito, usado en las llamadas telefónicas y

encargado de establecer y mantener el circuito por el que cursa la llamada. Al ser un protocolo que soporta tanto circuitos digitales como analógicos es el remplazo del ya discontinuado protocolo TUP. Entre los beneficios que ofrece ISUP están su rapidez, ancho de banda, la estandarización de los mensajes a intercambiar y el uso óptimo que le da a los enlaces troncales.

Los mensajes del protocolo ISUP son transportados por la red SS7 y utilizan los servicios de MTP nivel 3 y en ciertos casos, los de SCCP[5].

MTP3 usado por lo general para intercambiar mensajes a través de la red y SCCP con mayor énfasis en señalización extremo a extremo.

2.4 Mensajes básicos ISUP

Mensaje de dirección inicial (IAM): mensaje enviado hacia adelante con el propósito de reservar un circuito y transmitir información relevante con respecto al enrutamiento y al tratamiento de la llamada (número de la parte llamada y otros parámetros) [6].

Mensaje de dirección subsiguiente (SAM): mensaje utilizado para transmitir la información restante que el IAM no pudo abarcar [6].

Mensaje de dirección completa (ACM): mensaje que envía el nodo destino una vez que determine que se ha recibido el número completo de la parte llamada en el IAM [6].

Mensaje de respuesta (ANM): mensaje enviado hacia atrás para indicar que la llamada ha sido respondida [6].

Mensaje de liberación (REL): mensaje enviado en cualquier sentido, para indicar que el circuito se liberará al recibirse el mensaje de liberación completa [6].

Mensaje de liberación completa (RLC): mensaje enviado en cualquier sentido, como acuse de haberse recibido satisfactoriamente el mensaje REL y de haberse liberado el circuito solicitado [6].

2.5 Flujo de mensajes ISUP

Un mensaje ISUP es transmitido por el canal de señalización en una trama MSU. Esta trama tiene la primera parte de su campo SIO codificada 0101 para indicar que transporta una información que proviene del subsistema usuario ISUP [5].

En un flujo básico de ISUP se identifican tres etapas:

- Establecimiento
- Conversación
- Liberación

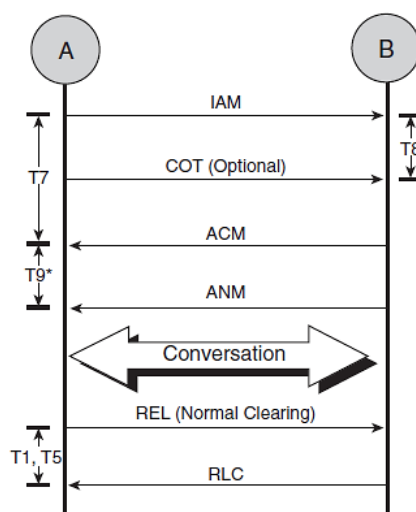


Figura 2.6 Flujo de ISUP

Establecimiento

En esta etapa intervienen tres mensajes. Si la llamada la realiza un abonado conectado al nodo A, este se encarga de enviar los dígitos marcados al nodo de destino mediante un mensaje IAM, el cual también pretenderá reservar un circuito troncal libre hacia el nodo destino para transmitir la voz, mediante una serie de parámetros que conforman su estructura. Una vez enviado el mensaje IAM, el nodo A activa inmediatamente el temporizador T7.

Una vez recibido el IAM en el Nodo B, este responde con un ACM indicando que la llamada hacia el abonado de destino puede tener lugar en el enlace, y a su vez que dicho enlace se ha reservado. Inmediatamente el nodo B envía el tono de llamada hacia el abonado de destino, al igual que lo hace el nodo A con el abonado originador

de la llamada al recibir el ACM, también se activa el temporizador T9 en el nodo A con la llegada de este mensaje.

Cuando el abonado destino, que se encuentra conectado al nodo B descuelga el teléfono, el nodo B envía un mensaje ANM hacia el nodo A para indicar que ahora existe un flujo de voz en el canal troncal [7].

Liberación

En esta etapa intervienen dos mensajes. El nodo que termina la llamada envía el mensaje REL hacia el otro nodo con el propósito de liberar el canal troncal que se está utilizando e inmediatamente activa dos temporizadores llamados T1 y T5.

El nodo que recibe el mensaje REL deberá responder inmediatamente con un mensaje de acuse de recibo RLC para indicar que se ha recibido el REL y que el circuito ha sido liberado con éxito [7].

2.5.1 Descripción de temporizadores

T7 Temporizador de espera de dirección completa [20-30 segundos]: se activa cuando se envía el mensaje de dirección IAM y se cancela cuando se recibe el mensaje ACM. Si este temporizador

expira se procede a liberar el equipo y las conexiones, enviando el mensaje REL [8].

T9 Temporizador de espera de respuesta: se activa cuando se recibe el mensaje ACM y se cancela cuando se recibe el mensaje ANM. En caso de expirar el temporizador, se libera la conexión enviando el mensaje REL [8].

T1 Temporizador de liberación completa [15-60 segundos]: se activa cuando se envía el mensaje REL y se cancela cuando llega el mensaje RLC. Si este temporizador expira, se reenvía el mensaje REL [8].

T5 Temporizador inicial de liberación completa [5-15 minutos]: se activa cuando se envía el mensaje REL y se cancela al recibir RLC. Si este mensaje expira, el nodo realiza las acciones siguientes: [8].

- Emisión de un mensaje de reinicialización de circuito (RSC, Reset Circuit)
- Emisión de una alarma al sistema de mantenimiento
- Desactivación del circuito
- Emisión periódica de un mensaje de reinicialización de circuito todos los T5, hasta la intervención del mantenimiento.

2.6 Formato de mensajes ISUP

Los mensajes ISUP viajan encapsulados en el campo de información de señalización (SIF) de MTP3 en el caso de transporte entre nodos, o de SCCP en caso de transporte de extremo a extremo. El SIF contiene la información de enrutamiento, seguido de los datos transportados [9].

La cabecera de enrutamiento está conformada por los siguientes campos:

- Código de punto de destino (DPC): identifica el nodo de destino.
- Código de punto de origen (OPC): identifica el nodo donde se originó el mensaje.
- Selección de enlace de señalización (SLS): identificador utilizado para compartir la carga a través de los enlaces o grupo de enlaces.

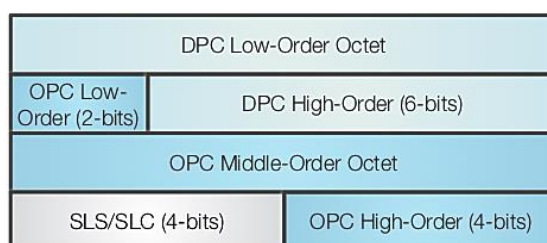


Figura 2.7 Cabecera de enrutamiento de ISUP

Cada mensaje ISUP incluye la siguiente información:

- Código de identificación de circuito (CIC): identifica el circuito al cual el mensaje está relacionado.
- Código de tipo de mensaje: identifica el mensaje.
- Parte obligatoria fija: contiene parámetros obligatorios de longitud fija, determinada por el tipo de mensaje.
- Parte obligatoria variable: contiene parámetros obligatorios de longitud variable, en donde cada uno está compuesto por su longitud y contenido. Se utilizan punteros, cada uno codificado en un solo byte cuyo valor indica el número de bytes entre el propio puntero (incluido) y el primer byte (no incluido) de los parámetros relacionados con el puntero. Todos los punteros se envían consecutivamente al inicio de la parte obligatoria variable incluyendo uno de inicio de la parte facultativa. Existe un orden de envío de estos punteros, de manera que según el tipo de mensaje, se conoce de qué parámetro se trata.
- Parte facultativa: parámetros que pueden estar incluidos en el mensaje, y están compuestos por el nombre del parámetro, su longitud y contenido.
- Fin de la parte facultativa: un octeto de ceros.

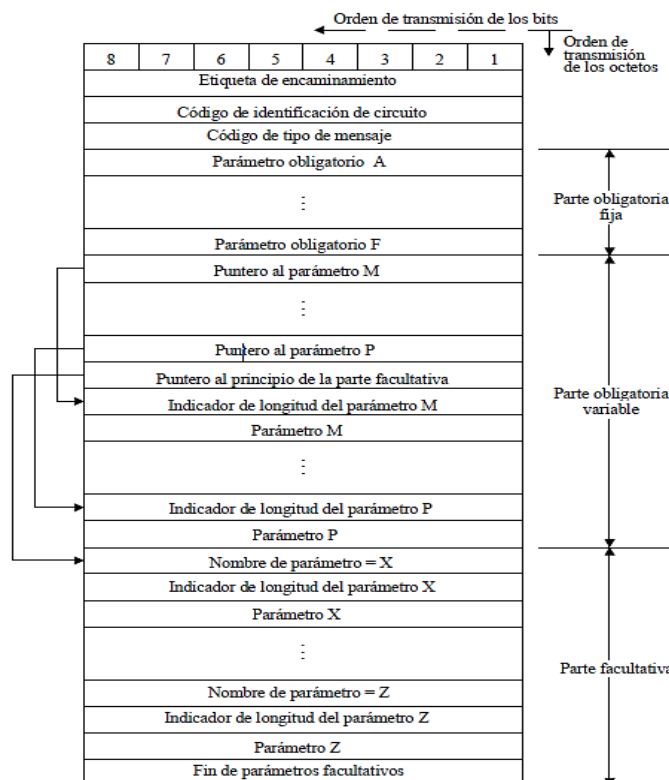


Figura 2.8 Formato de mensajes ISUP

2.6.1 Mensaje Inicial de Dirección (IAM, Initial Address Message)

El IAM es el mensaje más largo de ISUP, puede contener más de 50 parámetros opcionales y contiene la información necesaria para establecer una llamada [9].

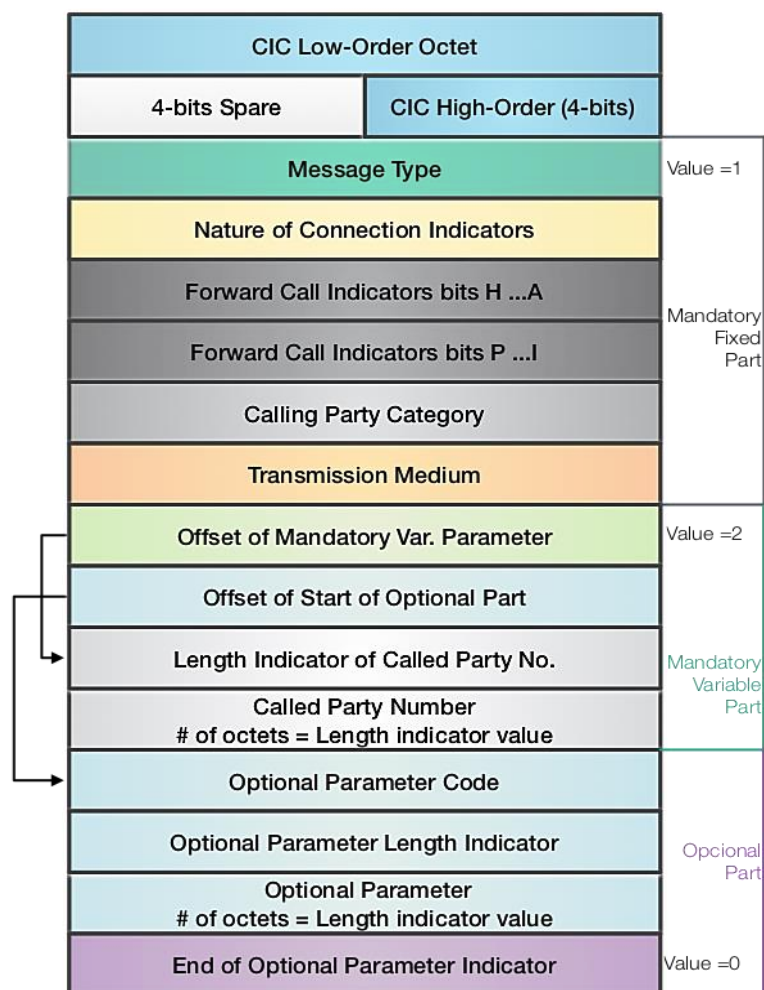


Figura 2.9 Mensaje Inicial de Dirección IAM

Parte obligatoria fija

- Código de tipo de mensaje: 1.
- Indicadores de la naturaleza de la conexión:
 - Indicador de satélite: especifica si han sido usado uno o varios satélites para la conexión que se pretende establecer.

- Indicador de continuidad: determina si se debe realizar una prueba de continuidad en el circuito.
- Indicador de dispositivo de control de eco: especifica si se utiliza la supresión de eco en el circuito.
- Indicadores de llamada hacia adelante:
 - Indicador de llamada nacional/internacional: indica si la llamada entrante es nacional o no.
 - Indicador de método de transferencia de extremo a extremo: especifica el método utilizado para la señalización de extremo a extremo, pudiendo ser de paso largo o SCCP.
 - Indicador de interfuncionamiento: especifica si se ha utilizado SS7 en todas las partes.
 - Indicador de información de extremo a extremo: especifica si hay o no información de extremo a extremo.
 - Indicador de parte usuario de la ISDN: especifica si ISUP ha sido utilizado en todo el trayecto.
 - Indicador de preferencia de la parte usuario de la ISDN: indica si ISUP se requiere en todo el trayecto.
 - Indicador de acceso ISDN: especifica si el acceso de origen es ISDN.

- Indicador de método de la SCCP: especifica si el método de señalización SSP de extremo a extremo es orientado a conexión, no orientado o ambos.
- Categoría de la parte llamante: especifica si el llamante es un abonado ordinario o con alguna prioridad, si es un operador, una llamada de prueba o teléfono público.
- Requisito del medio de transmisión: contiene los requisitos para las capacidades del circuito portador que son necesarios para el establecimiento de la llamada.

Parte obligatoria variable

- Puntero del parámetro obligatorio variable.
- Puntero del inicio de la parte opcional.
- Indicador de la longitud del número de la parte llamada.
- Número de la parte llamada (CdPN).
 - Indicador par/impar: indica un número par o impar de dígitos en el CdPN.
 - Indicador de la naturaleza de la dirección: indica el tipo de número.
 - Indicador de número de red interno: especifica si está permitido el enrutamiento hacia un número de la red interna.

- Indicador de plan de numeración: especifica el tipo de plan de numeración utilizado.
- Señales de dirección: son los dígitos reales que componen el CdPN.

2.6.2 Mensaje de Dirección Completa (ACM, Address Complete Message)

Este mensaje es enviado en dirección contraria al IAM para indicar que el extremo remoto del circuito ha sido reservado [9].

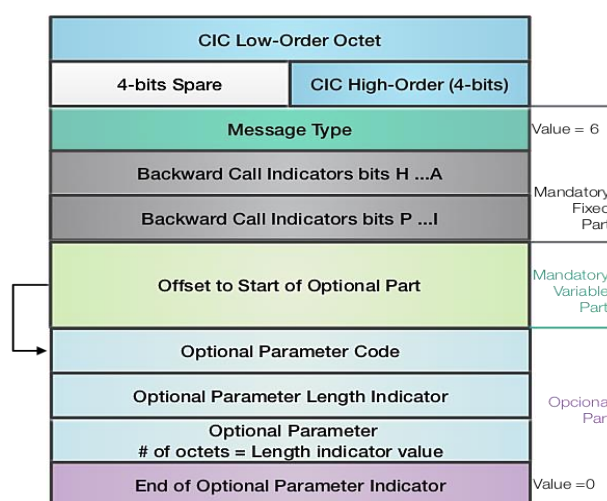


Figura 2.10 Mensaje de Dirección Completa

Parte obligatoria fija:

- Código de tipo de mensaje: 6
- Indicadores de llamada hacia atrás:

- Indicador de tasación: especifica si hay o no tasación de la llamada.
- Indicador de estados de la parte llamada: especifica si es un suscriptor libre.
- Indicador de categoría de la parte llamada: indica si el abonado es prepago o postpago.
- Indicador de método de transferencia de extremo a extremo
- Indicador de interfuncionamiento.
- Indicador de información de extremo a extremo.
- Indicador de la parte de usuario de la ISDN.
- Indicador de retención: Indica si se requiere o no la retención.
- Indicador de acceso.
- Indicador de dispositivo de protección contra el eco.
- Indicador de método SCCP.
- Parte obligatoria variable.
- Puntero del inicio de la parte facultativa.

2.6.3 Mensaje de Respuesta (ANM, Answer Message)

Este mensaje es enviado hacia el nodo de origen cuando el abonado llamado responde (descuelga el teléfono). La parte obligatoria fija solo contiene el octeto de tipo de mensaje cuyo valor es 9. Su parte

obligatoria variable solo está formada por el puntero hacia la parte facultativa en caso de existir parámetros opcionales [9].

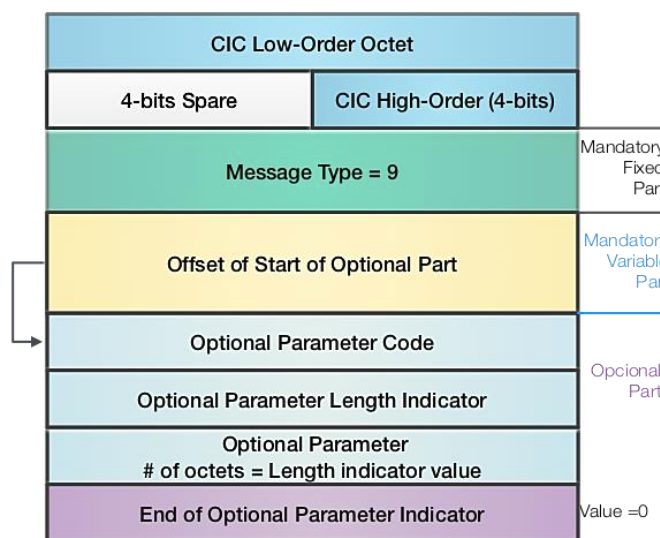


Figura 2.11 Mensaje de Respuesta ANM

2.6.4 Mensaje de Liberación (REL, Release Message)

Indica que el circuito está siendo liberado. Contiene únicamente el parámetro tipo de mensaje igual a 12 en la parte obligatoria fija, y en la parte obligatoria variable se especifica mediante indicadores, la causa por el cual el circuito está siendo liberado [9].

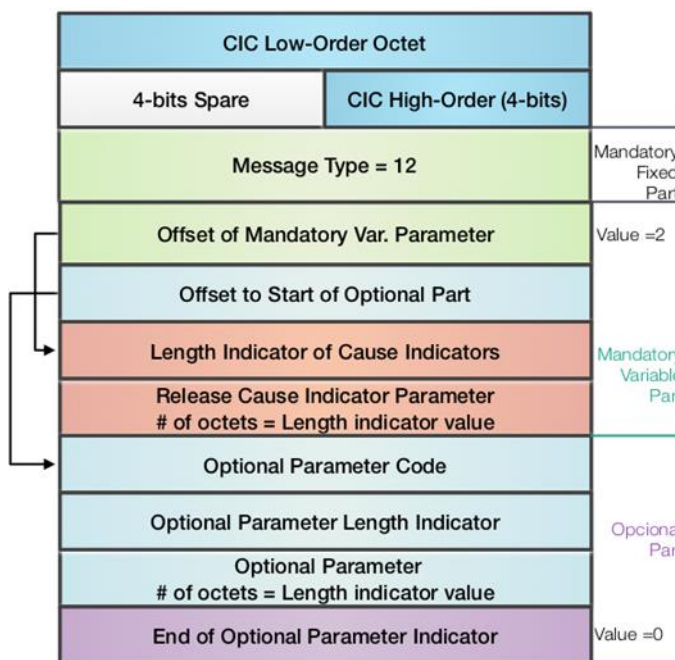


Figura 2.12 Mensaje de Liberación REL

2.6.5 Mensaje de Liberación Completa (RLC, Release Complete Message)

Indica que se ha recibido el mensaje REL. Contiene únicamente el parámetro tipo de mensaje igual a 16.

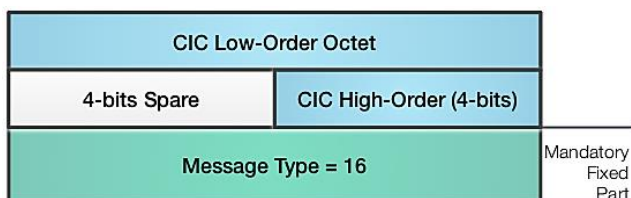


Figura 2.13 Mensaje de Liberación Completa RLC

2.7 Otros mensajes ISUP

En este proyecto de grado nos centramos en el funcionamiento de los mensajes más comunes y básicos para el establecimiento, gestión y liberación de los circuitos como lo son IAM, ACM, ANM, REL, RLC.

Existen otros mensajes que también intervienen en dicho proceso los cuales los detallaremos a continuación: [5].

- Mensaje de Intervención (FOT, Forward Transfer) es utilizado cuando una llamada es establecida automáticamente, entonces se transfiere la llamada a una operadora.
- Mensaje de Petición de Facilidad (FAR, Facility Request Message) se lo emite entre conmutadores con el objetivo de facilitar la activación de algún requerimiento solicitado.
- Mensaje de Rechazo de Facilidad (FRJ, Facility Reject Message) se lo emite cuando el FAR ha sido rechazado.
- Mensaje de Facilidad Aceptada (FAA, Facility Accepted Message) se lo emite cuando el FAR ha sido aceptado.
- Mensaje de Facilidad (FAC, Facility Message) es emitido entre conmutadores cuando se realiza alguna acción ante una solicitud previamente pedida.
- Mensaje de Información de Usuario (USR, User-to-User Information Message) permite establecer la señalización entre

conmutadoras.

- Mensaje Código de Identificación de Circuito no Equipado (UCIC, Unequipped Circuit Identification Code Message) se lo emite cuando el CIC enviado no existe en la base de datos y se es imposible identificar el circuito.
- Mensaje Confusión (CFN, Confusion Message) se lo emite cuando algún tipo de información extraña al circuito no se lo logra reconocer.
- Mensaje Petición de Identificación (IDR, Identification Request Message) se lo emite para poder filtrar aquellas llamadas que son maliciosas.
- Mensaje Respuesta de Identificación (IRS, Identification Responde Message) se lo emite cuando se ha recibido un IDR.
- Mensaje de Petición de Información (INR, Information Request) se emite cuando el IVR necesita exclusivamente el número del llamante y no le ha llegado.
- Mensaje Información (INF, Information Message) se emite en respuesta al INR.

2.8 Servicios suplementarios ISUP

Como hemos mencionado anteriormente, ISUP es un protocolo robusto, de gran capacidad, capaz de ofrecer muchos servicios ya

que está conformado por una gran cantidad de mensajes y a la vez gran cantidad de parámetros que pueden ser transportados a través de la red inteligente. Además de las características mencionadas, ISUP proporciona la flexibilidad suficiente para ajustarse a cualquier red regional gracias a los parámetros opcionales de sus mensajes, ya que hay que tener en cuenta que los servicios varían dependiendo de los operadores de telefonía [7].

CAPÍTULO 3

3 DESARROLLO DEL PROYECTO

3.1 Proyecto ISUP aplicado a una MSC

Como se había mencionado anteriormente, este proyecto tiene como principal objetivo simular de manera interactiva el funcionamiento del protocolo ISUP en una MSC para controlar la comunicación con un IVR en situaciones en las que el abonado requiera consultar su saldo marcando al *282 desde una terminal móvil. Para llevar a cabo esta implementación se prefirió trabajar en un sistema operativo de libre distribución y así evitar cualquier limitación en el desarrollo por parte de distintos fabricantes, además de ser creado con fines educativos y para un posterior desarrollo, motivo por el cual hemos elegido la distribución de Linux llamada Ubuntu por su

entorno amigable, de fácil manipulación y entendimiento para cualquier usuario que haya utilizado alguna vez una computadora con el sistema operativo propietario más común en la actualidad.

3.2 Requisitos para la implementación

Se detallarán los requisitos básicos de software y de hardware que forman parte de la estructura del proyecto, así como también aquellos necesarios durante el desarrollo de la programación, para su uso posterior en cualquier simulación cuya estructura funcional se apegue a la de este proyecto.

3.3 Software

3.3.1 Ubuntu 14.04.1

Elegido como sistema operativo para alojar el proyecto por ser la versión actual que contiene correcciones y mejoras de estabilidad que brindan una mayor seguridad y velocidad en la ejecución de los programas instalados además de seguir prestando soporte oficial para MySQL, Apache y PHP que son herramientas esenciales para el desarrollo del proyecto [10].

3.3.2 NetBeans IDE 8.0.2

Este entorno de desarrollo integrado, libre y gratuito es la herramienta primordial para escribir el código de programación ya que brinda gran soporte para PHP y C/C++ [11] además de poseer editores y herramientas para múltiples lenguajes de programación pudiéndolos compilar, depurar y ejecutar. Además de facilitar la instalación de una gran cantidad de extensiones que se integran fácilmente para desarrollar las herramientas propias requeridas como parte de un gran proyecto.

Una característica importante que influye en la elección de este programa es su gran capacidad y facilidad para poder integrarse con otras herramientas de colaboración en la nube, que permiten un trabajo en grupo organizado y respaldado en todo momento, asegurando cada una de las versiones de la programación a medida que se vayan desarrollando, evitando así la pérdida del trabajo en caso de una futura falla o pérdida de los equipos físicos que ejecutan el código programado. En nuestro caso particular, hemos utilizado los servicios de Assembla para la gestión de nuestro proyecto.

Netbeans puede ser descargado desde la página oficial netbeans.org

3.3.3 JDK

El conjunto de componentes de desarrollo en Java (Java Development Kit) es un entorno de desarrollo para crear aplicaciones utilizando la programación en java [12], siendo necesario para poder instalar NetBeans debido a que es el IDE oficial para Java.

3.3.4 XAMPP

Es una distribución del servidor web HTTP Apache completamente gratuita y fácil de instalar.

Contiene el sistema de gestión de base de datos MySQL, PHP y Perl [13].

3.4 Hardware

Como requerimiento mínimo, es necesario cualquier computadora que disponga de un disco duro cuyo espacio libre sea de 20GB y de 2GB de memoria RAM para el correcto funcionamiento en la ejecución de los programas mencionados anteriormente, además de un router para formar la red de área local mediante la cual se comunicarán los diferentes componentes.

3.5 Proceso de Implementación

En este caso, utilizamos una computadora portátil con las siguientes características:

Marca: Acer

Modelo: Aspire 5920

Memoria RAM: 2GB a 667Mhz

Capacidad de disco duro: 120 GB

Procesador: Intel Core 2 Duo T5450 / 1.66 GHz

Tarjeta de red inalámbrica: 802.11 a/b/g WLAN

Tarjeta de video: Mobile Intel Graphic Media Accelerator X3100



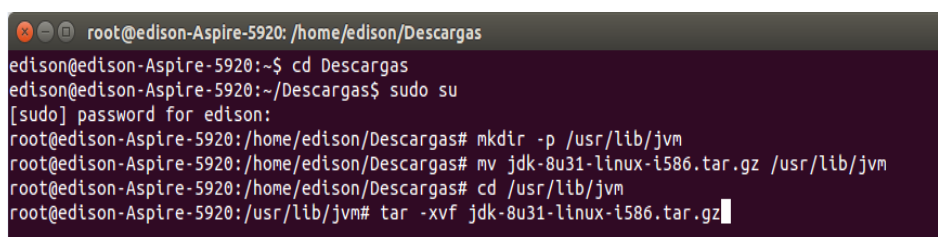
Figura 3.1 Computadora portátil

Como se mencionó anteriormente, se instaló la distribución de Linux, Ubuntu 14.04.1 cuyos archivos se pueden descargar desde el sitio web oficial www.ubuntu.com.

Con Ubuntu instalado en el equipo, se procede a preparar el entorno de desarrollo descargando todos los componentes de software mencionados.

3.5.1 Instalación del JDK

Una vez descargado el archivo de instalación, se debe acceder a la ruta donde se encuentra, desde una sesión en la terminal, seguido de esto es indispensable cambiar el usuario por el súper usuario de Ubuntu para poder realizar cualquier tipo de modificación a los archivos y las instalaciones respectivas. Se debe crear un directorio con la finalidad de alojar los archivos descomprimidos, se mueve el archivo a descomprimir y se debe dirigir hacia el nuevo directorio que lo contiene para proceder con la descompresión del archivo. En la figura 3.2 se muestran los comandos correspondientes para la secuencia explicada.



```
root@edison-Aspire-5920: /home/edison/Descargas
edison@edison-Aspire-5920:~$ cd Descargas
edison@edison-Aspire-5920:~/Descargas$ sudo su
[sudo] password for edison:
root@edison-Aspire-5920:/home/edison/Descargas# mkdir -p /usr/lib/jvm
root@edison-Aspire-5920:/home/edison/Descargas# mv jdk-8u31-linux-i586.tar.gz /usr/lib/jvm
root@edison-Aspire-5920:/home/edison/Descargas# cd /usr/lib/jvm
root@edison-Aspire-5920:/usr/lib/jvm# tar -xvf jdk-8u31-linux-i586.tar.gz
```

Figura 3.2 Instalación JDK

3.5.2 Instalación de NetBeans IDE

Una vez instalado JVK, el equipo se encuentra listo para la instalación de NetBeans. Inicialmente se procede a activar la cuenta del súper usuario para luego acceder al directorio donde se encuentra descargado el archivo de instalación, entonces se le debe dar todos los permisos de lectura, escritura y ejecución para luego poder ejecutar el instalador.

En la figura 3.3 se muestra mediante comandos, cada paso descrito en el párrafo anterior.

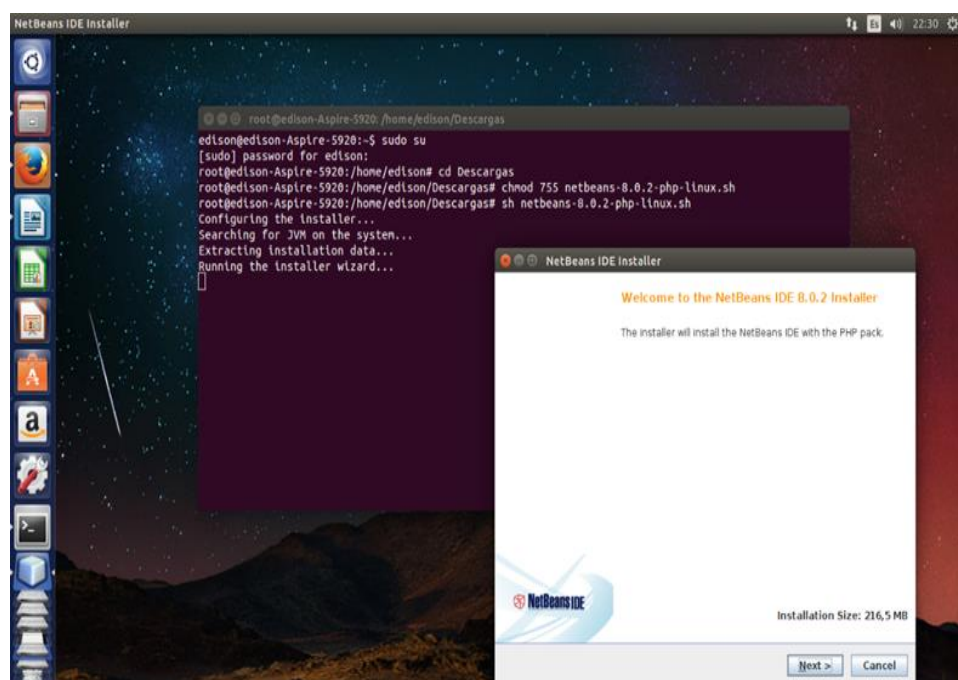


Figura 3.3 Instalación NetBeans IDE

3.5.3 Instalación de XAMPP

Para instalar este componente no es necesario activar la cuenta del súper usuario. Únicamente se debe acceder a la carpeta que contiene el instalador desde la terminal, asignarle permisos al usuario y ejecutar el archivo como se muestra en la figura 3.4.

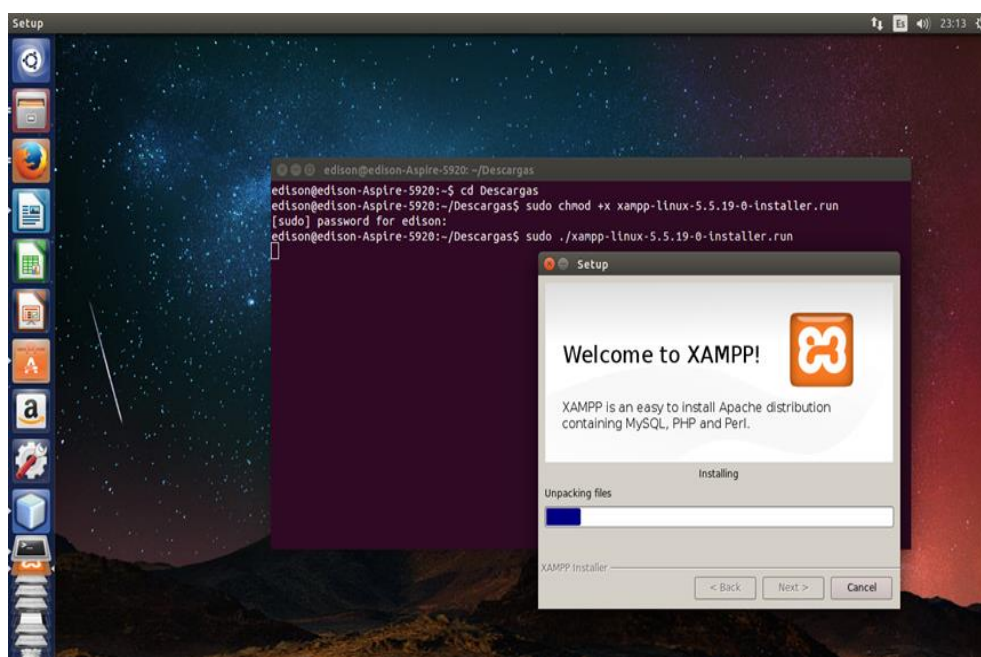


Figura 3.4 Instalación XAMPP

3.6 Funcionabilidad del sistema implementado

Para poder visualizar la funcionalidad del sistema implementado es indispensable tener conectados a la misma red de área local todos los componentes físicos que participan en la emisión-recepción de mensajes de señalización. Cada componente tendrá asignada una IP

fija dada por el router para poder apuntar a una dirección IP en particular de algún otro componente. Todos los mensajes de señalización que gobiernan las conexiones entre la MSC y el IVR serán aquellos pertenecientes al protocolo ISUP.

Para este sistema se utilizarán únicamente los mensajes necesarios que hacen posible la interacción completa entre los componentes mencionados. Cada uno de estos mensajes está estructurado como se mencionó en el capítulo 2 cuyas tramas bien definidas contienen parámetros establecidos por la ITU.

- La comunicación con el IVR inicia cuando la MSC recibe el mensaje ETC correspondiente al protocolo Camel desde el componente que cumple las funciones de una base de datos PrePago. Este mensaje será la señal de que la llamada corresponde a una consulta de saldos, es decir, que el número de destino es *282.
- La siguiente acción a tomar por parte de la MSC será enviar el primer mensaje de conexión IAM, el cual pretenderá reservar un circuito troncal libre hacia el IVR para transmitir la voz, mediante una serie de parámetros que conforman su estructura. Los parámetros variables de mayor relevancia son aquellos que relacionan la llamada del usuario con el canal que se le pretende asignar; estos serán el CIC y el calling party number.

Mencionado esto, aquellos datos irán siendo almacenados y consultados por la MSC en una base de datos creada en MySQL conforme a los requerimientos de los mensajes que se encuentren procesando.

- Una vez enviado el mensaje IAM, la MSC activa inmediatamente el temporizador T7 (20 a 30 segundos) y se cancela cuando se recibe el mensaje ACM. Si este temporizador expira se procede a liberar el equipo y las conexiones, enviando el mensaje REL.
- Cuando se reciban los mensajes ACM y posteriormente ANM desde el IVR, la MSC procederá a confirmar la reservación de ese circuito para la llamada del usuario que pretende consultar su saldo, actualizando el estado del circuito que tenga más tiempo libre, verificando la hora exacta del último uso y almacenando el nuevo número del usuario y el circuito que está utilizando. De esta manera la MSC administra todas las comunicaciones que se llevan a cabo entre el usuario y el IVR.
- También se activa el temporizador T9 en la MSC con la llegada del ACM y se cancela cuando se recibe el mensaje ANM. En caso de expirar el temporizador T9, se libera la conexión enviando el mensaje REL. El mensaje REL hacia el otro nodo se lo envía con el propósito de liberar el canal troncal que se está utilizando. Una vez enviado este mensaje inmediatamente se activan dos

temporizadores llamados T1 (15 a 60 segundos) y T5 (5 a 15 minutos).

- El nodo que recibe el mensaje REL deberá responder inmediatamente con un mensaje de acuse de recibo RLC para indicar que se ha recibido el REL y que el circuito ha sido liberado con éxito. A continuación en las figuras 3.5 y 3.6 podrán visualizar dos diagramas de lo antes descrito anteriormente.

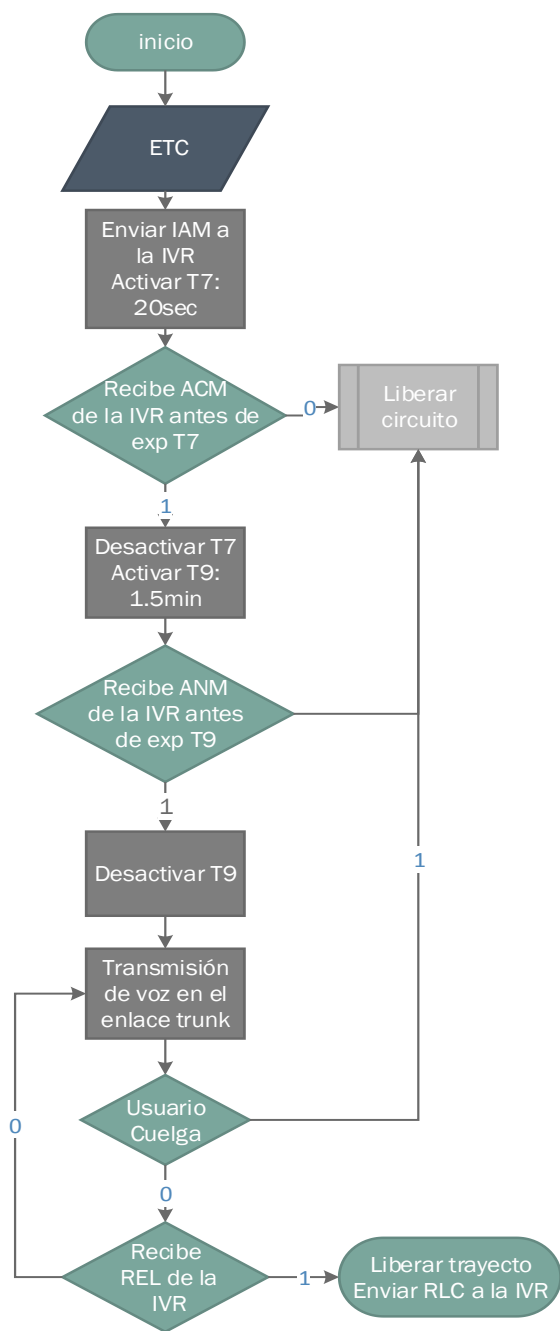


Figura 3.5 Diagrama funcional MSC – ISUP

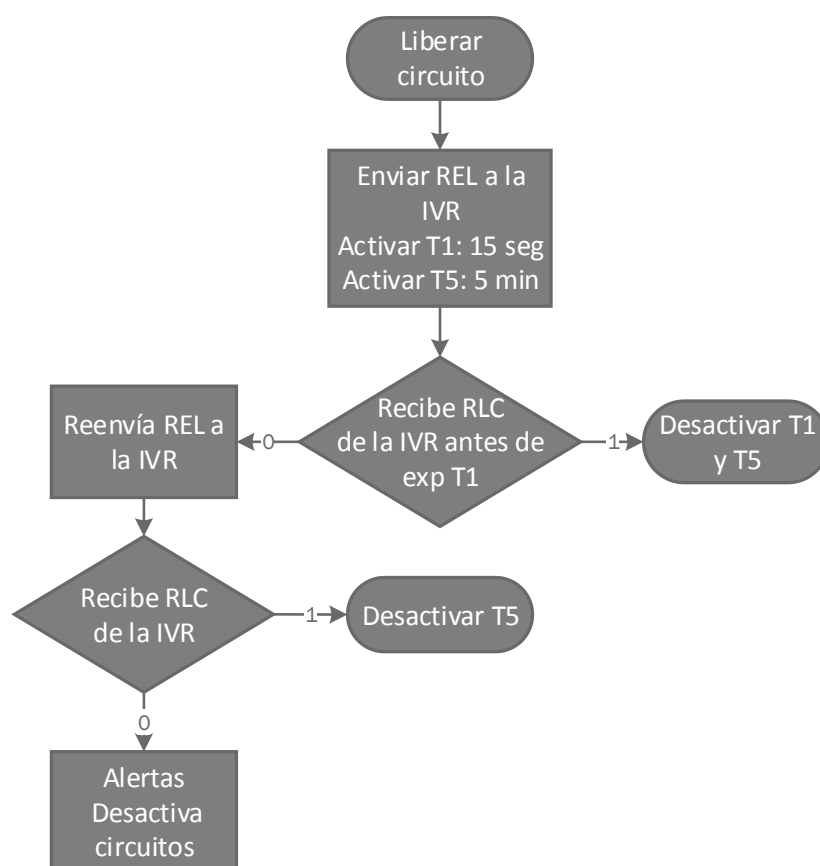


Figura 3.6 Diagrama de liberación de circuito

3.7 Flujo General para los diferentes escenarios

Aunque el proyecto está enfocado en el funcionamiento del protocolo ISUP en la MSC, a continuación se presentarán los flujos generales correspondientes a los diferentes escenarios posibles en la simulación y se procederá a explicar la secuencia de interés en que tiene lugar el mencionado protocolo.

3.7.1 Consulta de Saldos *282 opción 1

Este flujo de mensajes tiene lugar cuando un usuario llama al *282 con la intención de conocer cuánto saldo tiene disponible. Una vez que se ejecuta la llamada y la MSC ha validado que el número marcado corresponde efectivamente al *282, se empieza a comunicar con los demás dispositivos de red que intervienen en esta llamada utilizando los diferentes protocolos de señalización. Se requiere la intervención del IVR para transmitir los audios que posee almacenados al dispositivo del abonado y presentarle las opciones a las que tiene acceso en este servicio.

En este caso, mientras el usuario escuchaba las opciones disponibles, ha digitado el número 1 en su dispositivo, número que una vez llegado a la MSC, viaja a través de los canales de voz hacia el IVR como tono DTMF, para notificarle la opción elegida y así determinar que secuencia de audios se deben transmitir hacia el abonado.

Una vez que el IVR termina de enviar la secuencia de audios a ser escuchada por el abonado, ésta finaliza la comunicación con la MSC enviando un mensaje REL, mensaje que puede ser enviado en sentido contrario en caso que sea el abonado quien cuelgue la llamada antes que el IVR.

Como se puede apreciar en la figura 3.7, se ha destacado con color

azul toda la señalización correspondiente al protocolo ISUP.

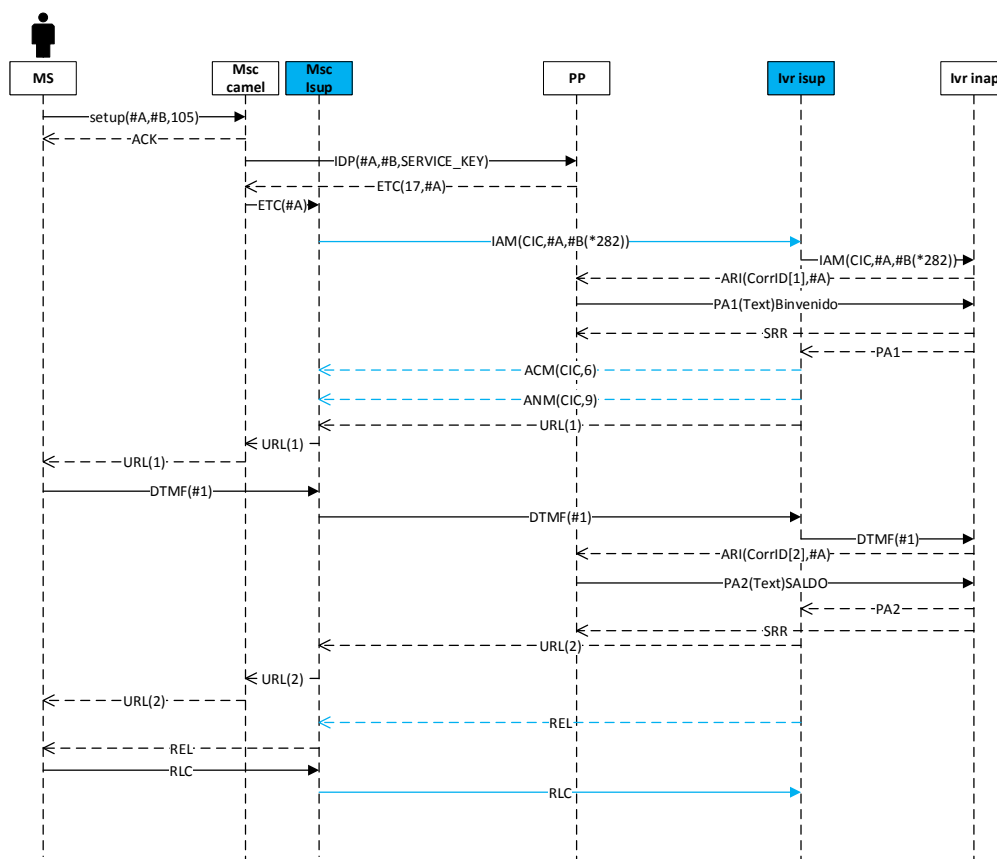


Figura 3.7 Consulta de Saldo *282 opción 1

3.7.2 Ingreso de Tarjeta Amigo Exitosa *282 opción 2

A diferencia de lo explicado en el apartado 3.7.1, en esta ocasión luego de llamar al *282 el abonado digita en su dispositivo el número 2, correspondiente a la opción de recarga de saldo, interviniendo nuevamente el IVR pero utilizándose más recursos de la red, por la

participación de nuevos mensajes de señalización entre otros componentes.

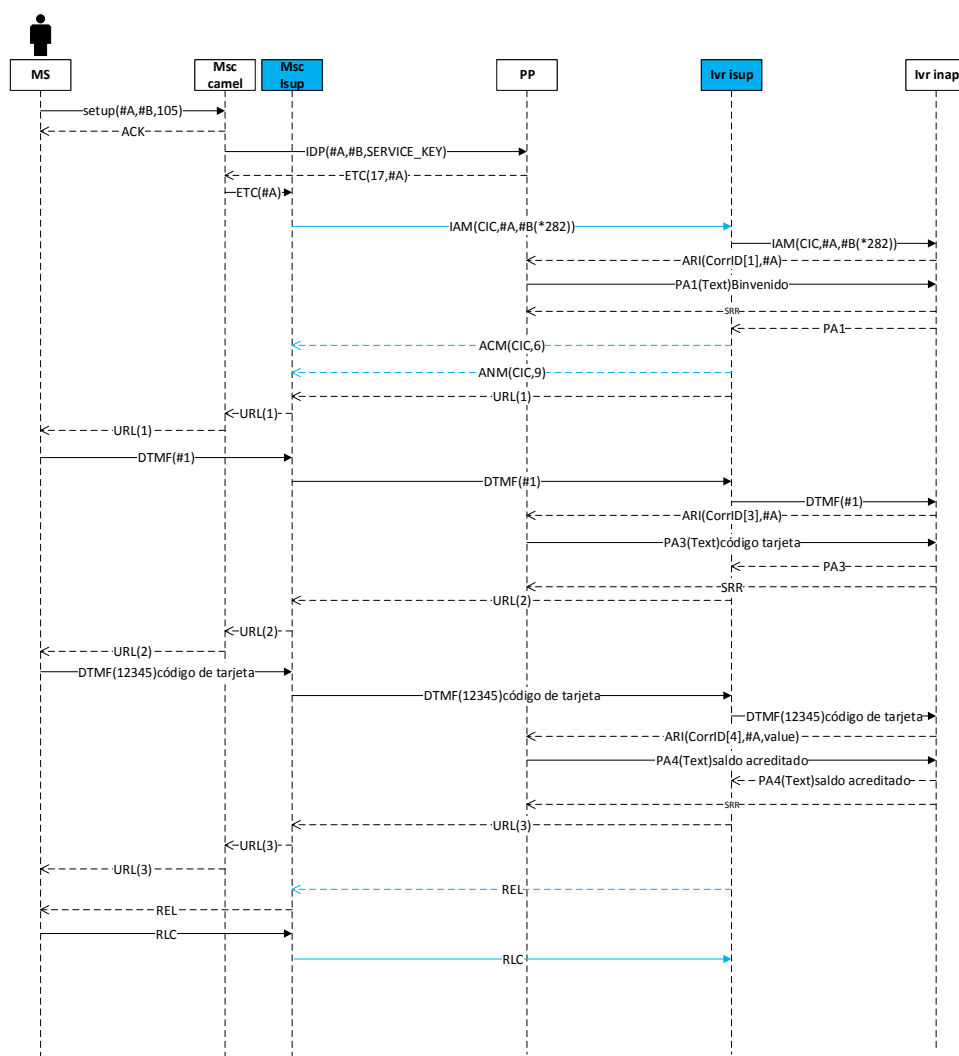


Figura 3.8 Ingreso de tarjeta correcto *282 opción 2

3.7.3 Ingreso de Tarjeta Amigo Incorrecta *282 opción 2

En este escenario, la intervención del protocolo ISUP sigue siendo igual que los apartados anteriores, mas no el procesamiento de otros datos en la MSC y en los otros componentes que intervienen.

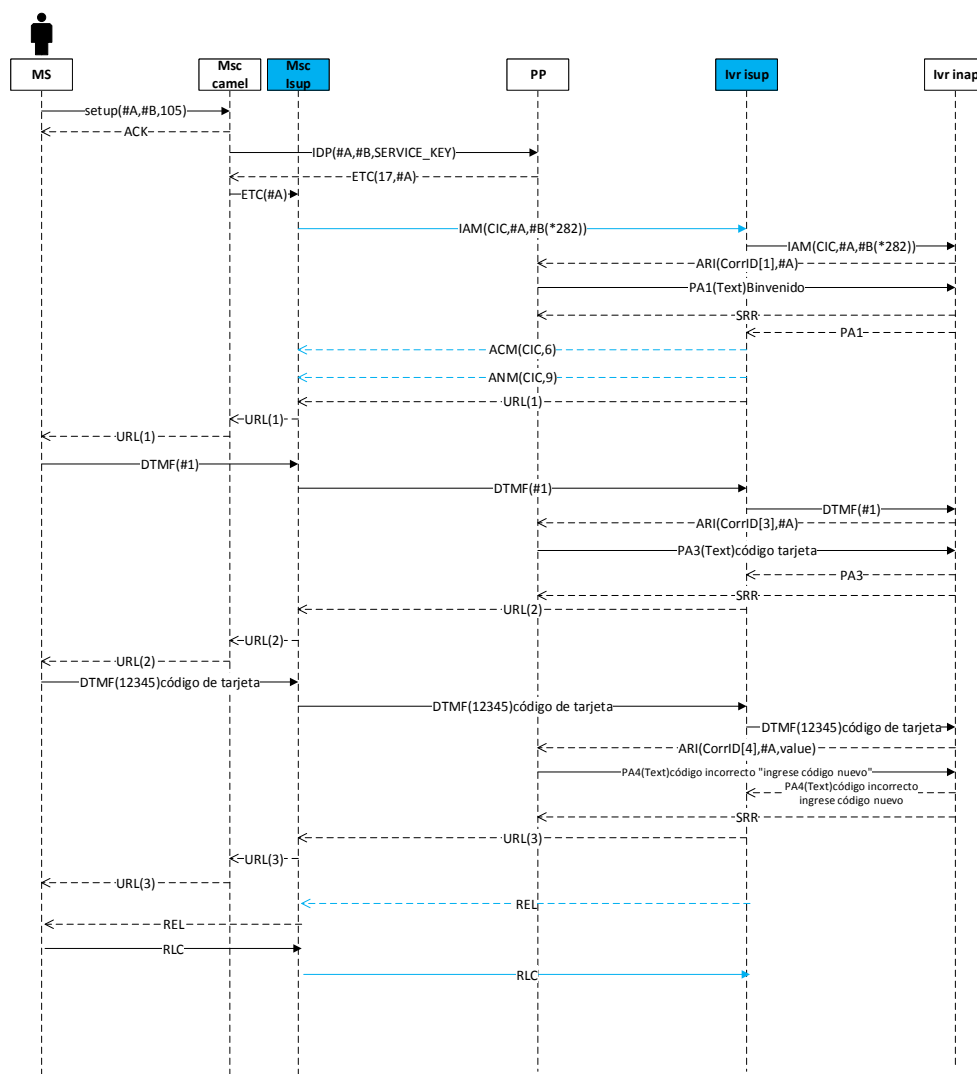


Figura 3.9 Ingreso de tarjeta incorrecta *282 opción 2

3.8 Topología de la implementación de la Red

La Topología de red usada para la simulación de la red celular está basada en una red en estrella y una red inalámbrica Wi-Fi.

- Red en Estrella:

Red en la cual las estaciones están conectadas directamente a un punto central y todas las comunicaciones se realizan a través de un concentrador. En este caso las Centrales de conmutación (MSC) y Registrador de Localización de Abonado (HLR) están conectados directamente entre sí con sus respectivas IP's a través de un Switch. La topología en estrella reduce la posibilidad de fallo de red conectando todos los nodos a un nodo central.

- Red Inalámbrica Wi-Fi:

Este tipo de redes hace posible la conexión de una red local (Red en Estrella) con cualquier otro dispositivo sin necesidad de cables, obteniendo una simulación más cercana a como se conecta una Estación Base (Router) con un celular (MS). La emisión y recepción de datos se realiza a través de radiofrecuencia.

En la figura 3.10 se muestra la topología de la red implementada. La MSC ISUP programada para la implementación del proyecto descrito tiene como dirección IP asignada la **192.168.0.4**. Cualquier tipo de comunicación que se quiera establecer con dicha central tiene que realizarse apuntando directamente a la IP antes mencionada.

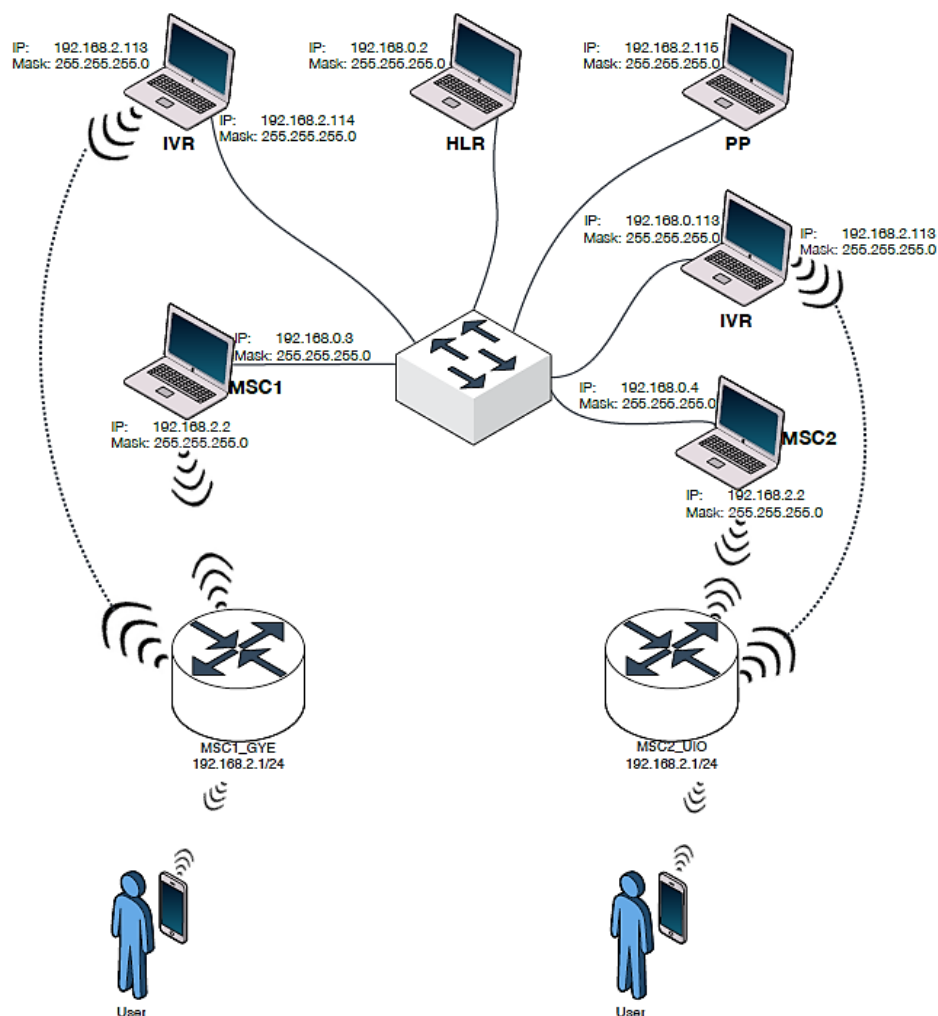


Figura 3.10 Topología de la red celular

3.9 Programación en PHP

La programación en PHP se le ejecuta bajo el software Netbeans IDE. Para el correcto funcionamiento del proyecto integrado fue necesario implementar ciertas funciones las cuales nos permitieron establecer conexión ya sea alámbrica o inalámbrica con los demás dispositivos de la red celular.

Básicamente se trabajó con direccionamiento IP fijo asignado a los diferentes componentes, en nuestro caso la MSC-ISUP con dirección IP 192.168.0.4.

Se implementó la función **get_parametro_ip_msc_isup()** la cual contiene la dirección IP de la MSC-ISUP que es de útil uso al momento que otro componente quiera establecer conexión con nuestra central.

```
1 <?php
2
3
4 function get_parametro_ip_ivr_isup() {
5     return "192.168.0.113";
6 }
7
8 function get_parametro_ip_ivr_inap()
9 {
10 //     return "192.168.0.114";
11     return "192.168.0.113";
12 }
13
14 function get_parametro_ip_msc_isup() {
15     return "192.168.0.4";
16 }
```

Figura 3.11 Función get parámetro IP

La programación se basa en mensajes nuevos y mensajes que son invocados como respuesta a un mensaje recibido. Para poder establecer conexión con la base de datos usamos el parámetro **mysql_connect** [14]; así mismo el parámetro **mysql_query()** [15] se lo uso para consultar a la base de datos información de la misma como son el número del celular que llamaba y el canal reservado

para el proceso de señalización.

Para el envío de mensajes nuevos se implementó la función **enviar_nuevo(\$equipo, \$ip, \$mensaje, \$parametros)** la cual contiene como parámetros:

- equipo: equipo de la red al cual se le va a entregar el mensaje
- ip: IP del equipo de la red al cual se le va a entregar el mensaje
- mensaje: mensaje que se va a entregar al equipo destinatario
- parámetros: diferentes parámetros que se envían en el mensaje

En el proceso de implementación se necesitaba que al momento de finalizar una comunicación se libere el canal que fue reservado para el proceso de señalización y transmisión de datos. Para esto se implementó la función **liberar_canal** en la cual básicamente se asocia el número del celular que llama con el código de identificación del circuito ya que este es el mismo número del circuito reservado.

A continuación se muestra en la figura 3.12 la programación donde se procesa el mensaje ETC proveniente del componente denominado PP que utiliza el protocolo Camel en la comunicación con la MSC para proceder a elaborar y enviar el mensaje IAM hacia el IVR. Además se puede visualizar composición de la función **enviar_nuevo**, y en la figura 3.13 la programación de la función **liberar_canal**.

```

1 <?php
2 include_once './util/registras.php';
3 include_once './util/enviar_mensaje.php';
4 include_once './util/parametros.php';
5 include_once './util/conversiones.php';
6
7 registrar_recibido_nuevo("etc", $_POST);
8
9 $callingPartyNumber = $_POST["calling_party_number"];
10
11
12 $conexion = mysql_connect("localhost", "root");
13 if (!$conexion) {
14     //die("No se puede conectar a la base de datos");
15     enviar_respuesta(array("ERROR", "No se puede conectar a la base de
16     datos"));
17     return;
18 }
19 mysql_select_db("msc_isup");
20 $resultado = mysql_query("select * from canal where reservado is null order
21     by ultimo uso asc");
22 //echo mysql_error();
23 $fila = mysql_fetch_assoc($resultado);
24 if (!$fila) {
25     enviar_respuesta(array("ERROR", "No hay canal disponible"));
26     return;
27 }
28 $cicNumero = $fila['codigo'];
29 mysql_query("update canal set reservado = '$callingPartyNumber' where codigo
30     = $cicNumero");
31 registrar_info("El canal seleccionado es " . $cicNumero);
32 $cicHexadecimal = str_pad(cadena_decimal_a_hexadecimal($cicNumero), 3, "0",
33     STR_PAD_LEFT); // "002";
34 // $callingPartyNumberHexadecimal = ""
35 $iam_ack = enviar_nuevo("ivr/isup", get_parametro_ip_ivr_isup(), "iam",
36     array("iam => "00${callingPartyNumber}82F21001040002020A0000010$
37     {cicHexadecimal}10004001"));
38
39 $urls = array();
40
41 for($i=0; $i<count($iam_ack); $i++)
42 {
43     $urls[] = $iam_ack[$i];
44 }
45 enviar_respuesta($urls);
46
47

```

Figura 3.12 Programación Mensaje IAM

```

1 <?php
2
3 function liberar_canal($calling_party_number){
4
5
6     $conexion = mysql_connect("localhost", "root");
7     mysql_select_db("msc_isup");
8
9     // preguntar que canal esta asociado al numero que llega
10    $resultado = mysql_query("select a.codigo from canal a where a.reservado
11        = $calling_party_number");
12    $fila = mysql_fetch_assoc($resultado);
13
14    if (!$fila)
15    {
16        return enviar_respuesta(array("ERROR", "Numero no tiene canal
17        asociado"));
18    }
19
20    //Se obtiene el canal asociado al numero
21    $cic=$fila["codigo"];
22    // notificar a la ivr que se libera el canal
23    $rel_ack = enviar_nuevo("ivr/isup", get_parametro_ip_ivr_isup(), "rel",
24        array("rel" => $cic));
25
26    //marcar el canal como liberado.
27    mysql_query("update canal set reservado = null, ultimo_uso = now() where
28        codigo = $cic and reservado is not null");
29
30    return enviar_respuesta(array("OK"));
31 }

```

Figura 3.13 Programación de Función liberar_canal

Para poder reproducir los prompts lo que se realiza es mediante HTTP apuntar a los url's donde se encuentran almacenados los audios.

Como parte de la programación se implementaron una serie de funciones, cuyo principal objetivo es proveer a un registro todos los mensajes que se envían y que se reciben en la MSC, así como también aquellas funciones encargadas de presentar los datos en la terminal de Ubuntu de manera ordenada para entender el comportamiento que tiene en tiempo real la MSC y al mismo tiempo poder identificar cualquier problema en caso de que exista.

En la figura 3.14 se muestra la programación de algunas funciones para registrar los datos y presentar en pantalla.

```

1 <?php
2
3 include_once 'parametros.php';
4 function registrar_central($componente, $stipo, $mensaje, $fields)
5 {
6     $fields_string = "";
7     //url-ify the data for the POST
8     $fields_string = "meta-componente=$componente&meta-tipo=$stipo&meta-
9     mensaje=$mensaje&";
10    foreach($fields as $key=>$value) { $fields_string .= $key.'='.$value.'&';
11    }
12    rtrim($fields_string, '&');
13
14    //open connection
15    $ch = curl_init();
16
17    //set the url, number of POST vars, POST data
18    $url = "http://192.168.0.4:8080/log";
19    curl_setopt($ch, CURLOPT_URL, "$url?$fields_string");
20
21    curl_setopt($ch, CURLOPT_RETURNTRANSFER, 1);
22    //execute post
23    $result = curl_exec($ch);
24    if ( ! $result )
25    {
26        trigger_error(curl_error($ch));
27    }
28
29    //close connection
30    curl_close($ch);
31    return $result;
32 }
33
34 function registrar($mensaje)
35 {
36     $archivo = "registro.log";
37     // La nueva persona a añadir al fichero
38     // Escribir los contenidos en el fichero,
39     // usando la bandera FILE_APPEND para añadir el contenido al final del
40     // fichero
41     // y la bandera LOCK_EX para evitar que cualquiera escriba en el fichero
42     // al mismo tiempo
43     file_put_contents($archivo, $mensaje . "\n", FILE_APPEND | LOCK_EX);
44 }
45
46 function registrar_completo($mensaje)
47 {
48     date_default_timezone_set("America/Guayaquil");
49     $hora = date("Y/m/d H:i:s").substr(microtime(), 1, 9);
50     registrar("$hora - $mensaje");
51 }
52
53 function presentar_array($arreglo)
54 {
55     $resultado = var_export($arreglo, true);
56     $resultado = str_replace(">=", "<=", str_replace("array", "",
57     $resultado));
58     $resultado = substr($resultado, 3, strlen($resultado) - 6);
59     return $resultado;
60 }
61
62 function registrar_recibido_nuevo($mensaje, $parametros)
63 {
64     registrar_central(getComponente(), "REC-NUE", $mensaje, $parametros);
65     registrar_completo("REC NUE : ");
66     registrar(presentar_array(array("mensaje" => $mensaje, "parametros" =>
67     $parametros)));
68 }
69
70 function registrar_envio_nuevo($destinatario, $ip, $mensaje, $parametros)
71 {
72     registrar_central(getComponente(), "ENV-NUE", $mensaje, $parametros);
73     registrar_completo("ENV NUE : ");
74     registrar(presentar_array(array("destinatario"=>$destinatario, "ip" =>
75     $ip, "mensaje" => $mensaje, "parametros" => $parametros)));
76 }
77
78 function registrar_recibido_respuesta($sack)
79 {
80     registrar_central(getComponente(), "REC-RES", "", $sack);
81     registrar_completo("REC RES :");
82     registrar(presentar_array(array("ack" => $sack)));
83 }
84
85 function registrar_envio_respuesta($sack)
86 {
87     registrar_central(getComponente(), "ENV-RES", "", $sack);
88     registrar_completo("ENV RES :");
89     registrar(presentar_array(array("ack" => $sack)));
90 }
91
92 function registrar_info($info)
93 {
94     registrar_completo("INFO :$info");
95 }

```

Figura 3.14 Programación de Funciones Registrar

CAPÍTULO 4

4 ANÁLISIS DE RESULTADOS

4.1 Resultados obtenidos

Al finalizar este proyecto se obtiene como resultado un equipo de procesamiento portátil, capaz de emular una parte del funcionamiento de una MSC que utiliza ISUP como protocolo fundamental para el transporte de voz a través de un canal de comunicación con un IVR, parte que al ser integrada con los demás proyectos desarrollados en paralelo y que funcionan con protocolos de la misma pila SS7, conforman una red que simula de manera interactiva, a una red de telefonía móvil constituida por diferentes

componentes que interactúan entre sí mediante el uso de diferentes protocolos.

4.2 Interpretación de resultados

Mediante la ejecución del archivo ***registro.log*** se puede visualizar de manera amigable el paso a paso de la iteración del proyecto implementado. Utilizaremos la aplicación ***terminal*** de Ubuntu para observar la ejecución en tiempo real, de la interacción entre los componentes de acuerdo a la programación en PHP del protocolo, desarrollada en Netbeans. Se mostrarán uno a uno tanto los mensajes que enviamos como los que recibimos. A continuación se mostrarán los logs del proceso de consulta de saldos marcando el *282 opción 1.

```

registro.log x
,
2015/02/08 03:10:58.66286200 - REC NUE :
'mensaje' : 'etc',
'parametros' :
(
'calling_party_number' : '0988272881',
)
2015/02/08 03:10:58.74187800 - INFO      :El canal seleccionado es 5
2015/02/08 03:10:58.74297200 - ENV NUE :
'destinatario' : 'ivr/isup',
'ip' : '192.168.0.113',
'mensaje' : 'iam',
'parametros' :
(
'iam' : '00098827288182F21001040002020A000001000510004001',
)
)
2015/02/08 03:10:59.06695400 - REC NUE :
'mensaje' : 'acm',
'parametros' :
(
'trama_acm' : '00001506000510004001',
'cic' : '5',
'message_type' : '6',
'opc' : '1',
'dpc' : '1',
'sls' : '1',
)
)
2015/02/08 03:10:59.06815700 - ENV RES :
'ack' :
(
0 : 'OK',
)
)
2015/02/08 03:10:59.07619800 - REC NUE :
'mensaje' : 'anm',
'parametros' :

```

Figura 4.1 Logs Consulta de Saldos *282 opción 1ª

Como era de esperarse se puede visualizar en la figura 4.1 que se recibe el mensaje ETC por parte del protocolo CAMEL, utilizado por la MSC para comunicarse con otro componente llamado prepago, que no forma parte de este análisis, y como respuesta se envía al IVR el mensaje IAM con su trama completa en la cual los campos más importantes para esta implementación fueron el identificador del circuito y el tipo de mensaje ya que con el identificador de circuito se reserva el canal que para el ejemplo mostrado es el canal 5. También se puede verificar la recepción de los mensajes ACM y ANM que le llegan a la MSC. En la figura 4.2 se visualiza los url's de los prompts

a tocar por parte del IVR. Una vez finalizada la consulta de saldos el teléfono cierra la llamada para lo cual se enviará un mensaje REL al IVR la cual nos responderá con un acuse RLC con el cual se libera el canal establecido para la comunicación. En la figura 4.3 se puede visualizar este envío y recepción de mensajes.

```
registro.log x
'parametros' :
(
  'trama_anm' : '0009000510004001',
  'cic' : '5',
  'message_type' : '9',
  'opc' : '1',
  'dpc' : '1',
  'sls' : '1',
)
2015/02/08 03:10:59.08345600 - ENV RES :
'ack' :
(
  0 : 'OK',
)
2015/02/08 03:10:59.09229100 - REC RES :
'ack' :
(
  0 : 'http://192.168.2.113/ivr/isup/audio/101.mp3',
  1 : 'http://192.168.2.113/ivr/isup/audio/102.mp3',
  2 : 'http://192.168.2.113/ivr/isup/audio/103.mp3',
  3 : 'http://192.168.2.113/ivr/isup/audio/2.mp3',
)
2015/02/08 03:10:59.09407700 - ENV RES :
'ack' :
(
  0 : 'http://192.168.2.113/ivr/isup/audio/101.mp3',
  1 : 'http://192.168.2.113/ivr/isup/audio/102.mp3',
  2 : 'http://192.168.2.113/ivr/isup/audio/103.mp3',
  3 : 'http://192.168.2.113/ivr/isup/audio/2.mp3',
)
2015/02/08 03:11:00.51203900 - REC NUE :
'mensaje' : 'rel',
'parametros' :
(
  'calling_party_number' : '0988272881'
)
Texto plano ▾ Anchura de la pestaña: 8 ▾ Ln 61010, Col 4 INS
```

Figura 4.2 Logs Consulta de Saldos *282 opción 1b


```

registro.log x
  2 : 'http://192.168.2.113/ivr/isup/audio/103.mp3',
  3 : 'http://192.168.2.113/ivr/isup/audio/2.mp3',
)
2015/02/08 03:10:59.09407700 - ENV RES :
'ack' :
(
  0 : 'http://192.168.2.113/ivr/isup/audio/101.mp3',
  1 : 'http://192.168.2.113/ivr/isup/audio/102.mp3',
  2 : 'http://192.168.2.113/ivr/isup/audio/103.mp3',
  3 : 'http://192.168.2.113/ivr/isup/audio/2.mp3',
)
2015/02/08 03:11:00.51203900 - REC NUE :
'mensaje' : 'rel',
'parametros' :
(
  'calling_party_number' : '0988272881',
)
2015/02/08 03:11:00.51434100 - ENV NUE :
'destinatarlo' : 'ivr/isup',
'lp' : '192.168.0.113',
'mensaje' : 'rel',
'parametros' :
(
  'rel' : '4',
)
2015/02/08 03:11:00.52454000 - REC RES :
'ack' :
(
  0 : 'rlc',
)
2015/02/08 03:11:00.56399600 - ENV RES :
'ack' :
(
  0 : 'OK',
)
)

```

Texto plano ▾ Anchura de la pestaña: 8 ▾ Ln 61030, Col 9 INS

Figura 4.3 Logs Consulta de Saldo *282 opción 1c

Quando el usuario final quiere hacer una recarga de una tarjeta amigo en su celular se puede validar en los logs del proyecto que los prompts a tocar son otros por parte del IVR ya que el número a marcar ya no es 1 sino 2. En la figura 4.6 podemos validar lo descrito anteriormente. Al momento de finalizar la recarga el proceso de liberar el circuito es similar al que se ejecuta en la consulta de saldos. Se enviará un mensaje REL al IVR la cual nos responderá con un acuse RLC con el cual se libera el canal establecido para la comunicación. En la figura 4.7 se puede visualizar este envío y recepción de mensajes.

```

registro.log x
2015/02/08 03:02:48.05746700 - REC NUE :
'mensaje' : 'etc',
'parametros' :
(
'calling_party_number' : '0988272881',
)
2015/02/08 03:02:48.11303000 - INFO :El canal seleccionado es 4
2015/02/08 03:02:48.11416400 - ENV NUE :
'destinatario' : 'ivr/lsup',
'ip' : '192.168.0.113',
'mensaje' : 'iam',
'parametros' :
(
'iam' : '00098827288182F21001040002020A000001000410004001',
)
2015/02/08 03:02:48.48809400 - REC NUE :
'mensaje' : 'acm',
'parametros' :
(
'trama_acm' : '00001506000410004001',
'cic' : '4',
'message_type' : '6',
'opc' : '1',
'dpc' : '1',
'sls' : '1',
)
2015/02/08 03:02:48.48977100 - ENV RES :
'ack' :
(
0 : 'OK',
)
2015/02/08 03:02:48.49954000 - REC NUE :
'mensaje' : 'anm',
'parametros' :
(

```

Figura 4.4 Logs Ingreso de tarjeta correcta *282 opción 2a

```

registro.log x
'parametros' :
(
'trama_anm' : '0009000410004001',
'cic' : '4',
'message_type' : '9',
'opc' : '1',
'dpc' : '1',
'sls' : '1',
)
2015/02/08 03:02:48.50337800 - ENV RES :
'ack' :
(
0 : 'OK',
)
2015/02/08 03:02:48.50957100 - REC RES :
'ack' :
(
0 : 'http://192.168.2.113/ivr/lsup/audio/101.mp3',
1 : 'http://192.168.2.113/ivr/lsup/audio/102.mp3',
2 : 'http://192.168.2.113/ivr/lsup/audio/103.mp3',
3 : 'http://192.168.2.113/ivr/lsup/audio/2.mp3',
)
2015/02/08 03:02:48.51071000 - ENV RES :
'ack' :
(
0 : 'http://192.168.2.113/ivr/lsup/audio/101.mp3',
1 : 'http://192.168.2.113/ivr/lsup/audio/102.mp3',
2 : 'http://192.168.2.113/ivr/lsup/audio/103.mp3',
3 : 'http://192.168.2.113/ivr/lsup/audio/2.mp3',
)
2015/02/08 03:02:51.94791500 - REC NUE :
'mensaje' : 'dtmf',
'parametros' :
(
'tecla' : '1'

```

Figura 4.5 Logs Ingreso de tarjeta correcta *282 opción 2b

```

registro.log x
(
  'tecla' : '1',
  'calling_party_number' : '0988272881',
)
2015/02/08 03:02:51.94964800 - ENV NUE :
'destinatario' : 'ivr/isup',
'ip' : '192.168.0.113',
'mensaje' : 'dtmf',
'parametros' :
(
  'tecla' : '1',
  'calling_party_number' : '0988272881',
)
)
2015/02/08 03:02:52.18787700 - REC RES :
'ack' :
(
  0 : 'http://192.168.2.113/ivr/isup/audio/105.mp3',
  1 : 'http://192.168.2.113/ivr/isup/audio/15.mp3',
  2 : 'http://192.168.2.113/ivr/isup/audio/107.mp3',
  3 : 'http://192.168.2.113/ivr/isup/audio/108.mp3',
  4 : 'http://192.168.2.113/ivr/isup/audio/89.mp3',
  5 : 'http://192.168.2.113/ivr/isup/audio/109.mp3',
  6 : 'rel',
)
)
2015/02/08 03:02:52.23854500 - ENV RES :
'ack' :
(
  0 : 'OK',
)
)
2015/02/08 03:02:52.24097200 - ENV RES :
'ack' :
(
  0 : '',
  1 : 'http://192.168.2.113/ivr/isup/audio/105.mp3',
  2 : 'http://192.168.2.113/ivr/isup/audio/15.mp3',
)
)

```

Texto plano ▾ Anchura de la pestaña: 8 ▾ Ln 60824, Col 5 INS

Figura 4.6 Logs Ingreso de tarjeta correcta *282 opción 2c

```

registro.log x
(
  0 : '',
  1 : 'http://192.168.2.113/ivr/isup/audio/105.mp3',
  2 : 'http://192.168.2.113/ivr/isup/audio/15.mp3',
  3 : 'http://192.168.2.113/ivr/isup/audio/107.mp3',
  4 : 'http://192.168.2.113/ivr/isup/audio/108.mp3',
  5 : 'http://192.168.2.113/ivr/isup/audio/89.mp3',
  6 : 'http://192.168.2.113/ivr/isup/audio/109.mp3',
  7 : 'rel',
)
)
2015/02/08 03:02:59.52542900 - REC NUE :
'mensaje' : 'rlc',
'parametros' :
(
  'calling_party_number' : '0988272881',
)
)
2015/02/08 03:02:59.52691500 - ENV NUE :
'destinatario' : 'ivr/isup',
'ip' : '192.168.0.113',
'mensaje' : 'rlc',
'parametros' :
(
  'opcode' : '1000',
  'CIC' : '',
)
)
2015/02/08 03:02:59.54313300 - REC RES :
'ack' :
(
  0 : 'OK',
)
)
2015/02/08 03:02:59.54407000 - ENV RES :
'ack' :
(
  0 : 'OK RLC',
)
)

```

Texto plano ▾ Anchura de la pestaña: 8 ▾ Ln 60854, Col 5 INS

Figura 4.7 Logs Ingreso de tarjeta correcta *282 opción 2d

4.3 Interfaz gráfica de la MSC

Para poder visualizar la ejecución del proyecto utilizaremos una interfaz que se diseñó con el objetivo de que la interacción con el usuario final sea mucho más amigable y comprensible.



Figura 4.8 Panel de Control de la aplicación MSC

Desde el menú de inicio en el panel de control de la aplicación se tiene acceso a la señalización de la MSC. Estas interfaces fueron creadas con el fin de obtener información más detallada de cada uno de los mensajes de señalización de los diferentes protocolos SS7. En nuestro caso nos enfocaremos en mostrar la información detallada de cada uno de los mensajes ISUP como la hora en la que fueron enviados, recibidos; así como los diferentes parámetros que cada uno de estos mensajes posee.

A continuación en la figura 4.9 se muestra la interfaz gráfica de la señalización de la MSC ISUP en la cual se muestra en líneas entre cortadas aquellos mensajes que no son parte del protocolo; más sin embargo fueron necesarios al momento de la programación en PHP.

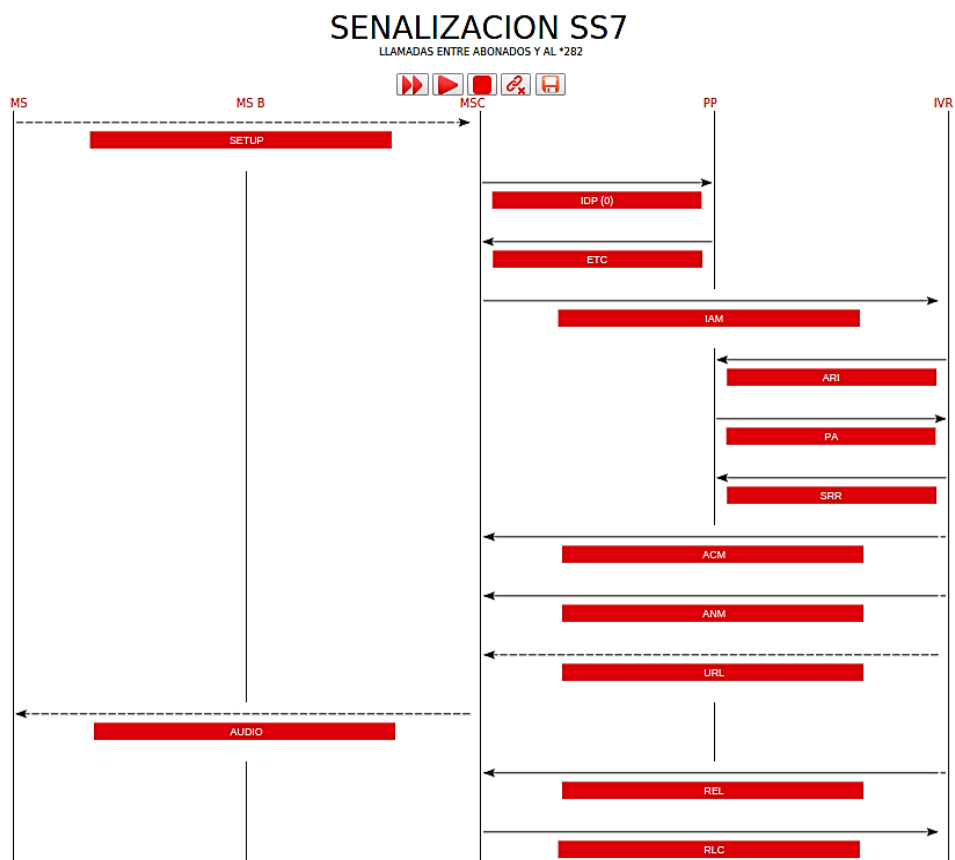


Figura 4.9 Interfaz de la señalización MSC visualizada desde la aplicación

La información detallada de estos mensajes la obtenemos al momento de dar clic sobre cada uno de los casilleros que las encierran. Cabe recalcar que la señalización se va mostrando paso a paso para lo cual se puede dar clic en el botón ya sea de reproducir o parar al momento que se considere pertinente.

A continuación se muestran los mensajes que tomamos en consideración para la implementación del proyecto correspondiente al protocolo ISUP que son IAM, ACM, ANM, ACM y RLC.

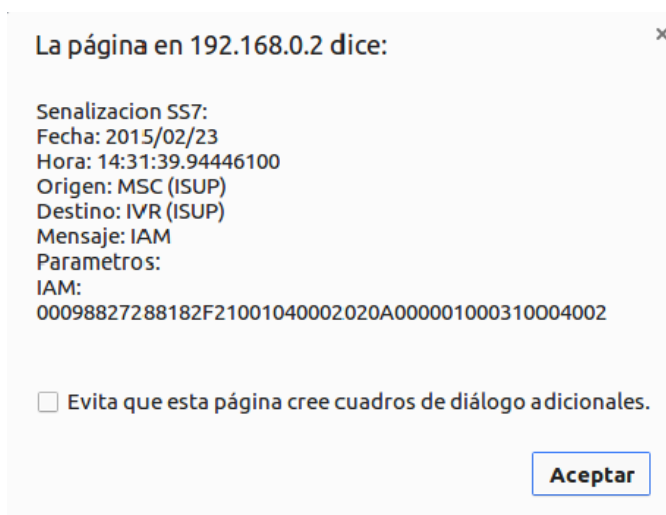


Figura 4.10 Interfaz Mensaje IAM

Cada uno de estos mensajes tiene su propia identificación como se lo explicó en capítulos anteriores y esta es una buena herramienta para poder detectar cualquier tipo de error ya que en la interfaz se muestra información detalla de cada uno de los parámetros que poseen los mensajes y al momento de haber alguna incongruencia se la detectará con facilidad.

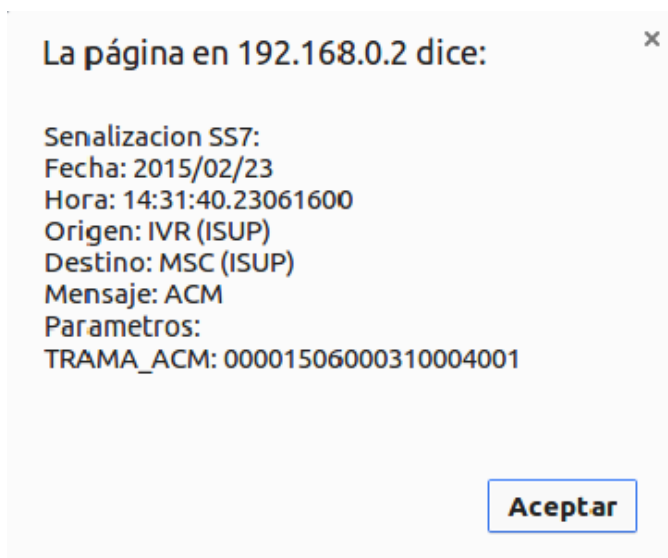


Figura 4.11 Interfaz Mensaje ACM

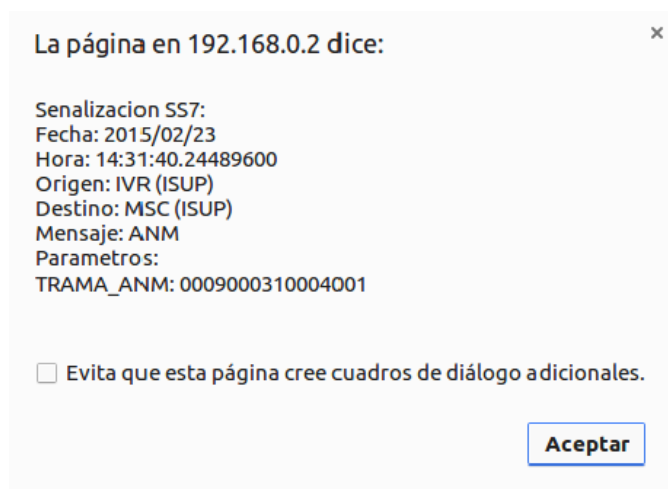


Figura 4.12 Interfaz Mensaje ANM

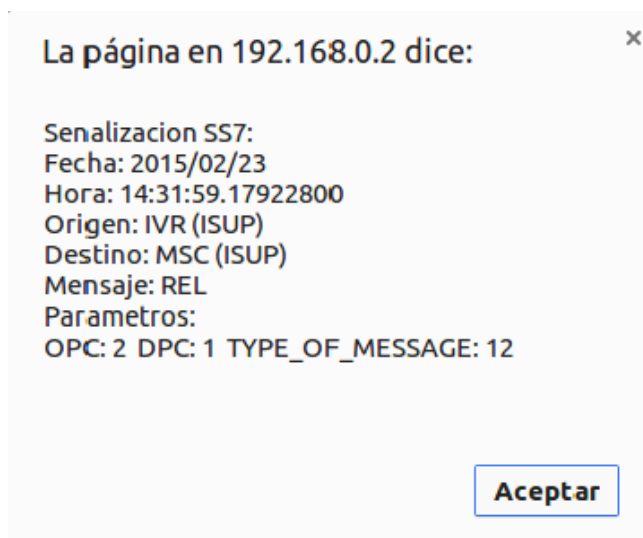


Figura 4.13 Interfaz Mensaje REL

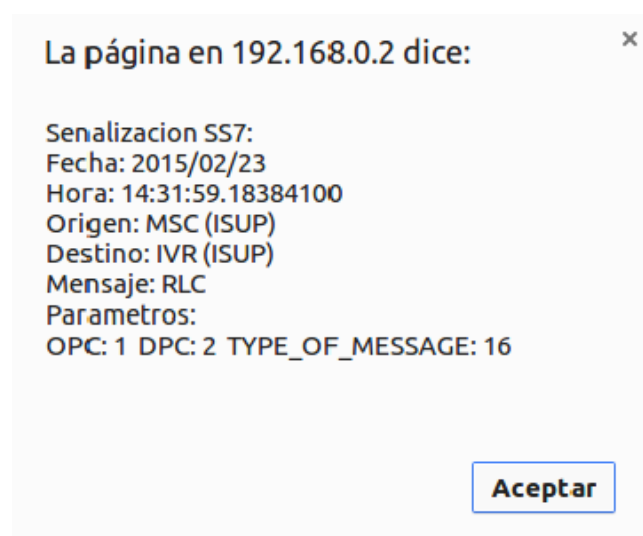


Figura 4.14 Interfaz Mensaje RLC

4.4 **Discusión**

De acuerdo con la apreciación de los resultados obtenidos en el apartado 4.2, existen muchos aspectos a analizar dentro de la interpretación de las pruebas realizadas.

La acción de los temporizadores descritos en el capítulo 2, administrados por la MSC, tienen poco impacto en la implementación del proyecto debido a que los tiempos de respuesta para el envío y recepción de mensajes del protocolo ISUP son muy cortos y no serían apreciados, razón por la cual la implementación se la realizó a través de un único temporizador. Cabe recalcar que tanto el temporizador T1 como el temporizador T5 cumplen un papel importante dentro del funcionamiento del protocolo ISUP, ya que al momento de expirar dichos temporizadores los canales que se reservaron o se pretendían reservar serían liberados. Con lo cual no se establecería ningún tipo de comunicación.

En la ejecución del proyecto se visualizan algunos *logs* o registros de mensajes que no pertenecen en sí al protocolo ISUP; mas sin embargo su creación y reproducción fueron necesarios al momento de realizar la programación en PHP ya que estos mensajes son indispensables para el correcto funcionamiento de la MSC y para el establecimiento de la conexión entre los abonados.

Por otro lado, es necesario aclarar que, en la interfaz gráfica que se muestra al operador o usuario final de la aplicación desarrollada, los mensajes que no pertenecen a ISUP son mostrados con líneas entrecortadas para poder diferenciarlos de los mensajes propios del protocolo.

CONCLUSIONES

Al haber culminado con éxito el presente proyecto, se puede concluir sobre algunos aspectos relacionados a la implementación lógica – física del sistema en base a la funcionalidad obtenida, siendo de mucho interés para futuros lectores que deseen desarrollar un sistema que simule una red gobernada por protocolos de telefonía.

1. La MSC como central inteligente de la red en la que converge una gran cantidad de canales de comunicación, es el elemento principal, a ser cuidadosamente programado para entender los mensajes provenientes de diferentes protocolos, y darles el procesamiento adecuado, tomando las decisiones en base a la información que contienen. Como caso específico, siempre que se recibe el mensaje ETC correspondiente al protocolo CAMEL, ésta entenderá automáticamente que debe levantar un

enlace de voz con el IVR cuyo DPC coincida con el destino de la comunicación a establecerse utilizando el protocolo ISUP, procediendo a ejecutar las operaciones respectivas para el envío de mensajes de señalización de acuerdo al protocolo.

2. La gran cantidad de parámetros opcionales que permiten transportar los mensajes del protocolo ISUP, brinda la flexibilidad de adaptar el sistema implementado a un vasto número de requerimientos, pudiendo ser capaz de simular el comportamiento real de diferentes servicios presentes en la telefonía alrededor del mundo.
3. Para que un canal de señalización se reserve por completo es necesario recibir por parte del IVR el mensaje ACM, como respuesta al mensaje IAM que fue enviado inicialmente por la MSC para reservar dicho canal. La activación de los temporizadores permite controlar de manera adecuada la liberación o reservación de un canal. Si los temporizadores activados llegaran a expirar se produce inmediatamente la liberación del circuito, quedando nuevamente disponible para ser reutilizado.
4. Durante el proceso de establecimiento de la comunicación con el IVR, la MSC realiza una búsqueda del canal que tiene más tiempo libre en su base de datos para asignarlo a la comunicación que pretende establecer, organizando así de manera racional la utilización de todos los canales disponibles para optimizar la carga que maneja la central.

5. La correcta programación en PHP desarrollada en Netbeans permite de manera muy fácil acceder a la base de datos en MySQL utilizando funciones de sencilla implementación, lo que permite administrar de óptima manera la base de datos del proyecto implementado. Para poder establecer conexión con la base de datos se utilizaron los parámetros `mysql_connect` y `mysql_query` para realizar consultas y actualizaciones en la base de datos, respectivamente.

RECOMENDACIONES

Con la finalidad de realizar una correcta implementación, y desarrollo de una aplicación cuya estructura se base en la detallada en este proyecto, se indicarán a continuación algunas recomendaciones para facilitar su implementación y comprender la relevancia de algunos programas utilizados.

1. Al trabajar con un proyecto integral donde cada componente debe ser desarrollado simultáneamente, un factor muy importante a tener en cuenta es el uso de herramientas capaces de brindar un respaldo en la nube en todo momento de cada una de las actualizaciones del código programado, así como también el poder compartir las modificaciones con todos los equipos de trabajo, evitando de esta manera aumentar la probabilidad de incompatibilidad en la comunicación existente entre los componentes que se desarrollan en diferentes computadoras.

2. Se debe tener conocimientos previos de Linux para poder realizar cualquier tipo de manipulación con los archivos requeridos en el proceso de implementación, así también como conocimientos avanzados de programación en PHP y de herramientas de desarrollo que facilitan la detección de errores mediante la ejecución de pruebas directas al código programado.
3. Realizar pruebas de conectividad de la red antes de ejecutar la simulación, asegurándose que todos los dispositivos se encuentren conectados ya sea alámbrica o inalámbricamente a sus respectivas redes, realizando pruebas de conectividad de LAN a LAN mediante el protocolo ICMP (Internet Control Message Protocol), específicamente con las instrucciones *pingy traceroute* en el terminal de Ubuntu, así como también verificar que el cableado utp utilizado se encuentre en óptimas condiciones.
4. Al momento de instalar las diferentes herramientas como Java, Netbeans, Ubuntu, XAMPP, hacerlo bajo la misma arquitectura ya sea 32 o 64 bits ya que de lo contrario se pueden generar conflictos al momento de establecerse la comunicación.
5. Es primordial tener levantado el servicio Apache al momento de la ejecución del proyecto. De lo contrario no se podrá establecer comunicación entre los diferentes dispositivos de la red. Existen comandos que se pueden ejecutar en el terminal de Ubuntu para que este

proceso sea automático y así evitar pérdida de conectividad entre los dispositivos.

BIBLIOGRAFÍA

- [1] Van Bosse & Devetak, Signaling in Telecommunication Networks, John Wiley & Sons 2nd Ed, 2006
- [2] Performance Technologies Inc., Tutorial on Signaling System 7, <http://www.pt.com>, fecha de consulta octubre del 2014
- [3] Pignataro, Natalia, Sistema de Señalización Número 7, https://eva.fing.edu.uy/pluginfile.php/67113/mod_resource/content/2/SS7_2014%20%28presentacion%29.pdf, fecha de consulta marzo 2015
- [4] Enrique Herrera Pérez, Tecnología y Redes de Transmisión de Datos, Grupo Noriega 1^{ra} Ed, 2003
- [5] Etudes et FORMations en Télécommunications, El Protocolo ISUP, http://www.efort.com/media_pdf/ISUP_ES_EFORT.pdf, fecha de consulta octubre 2014
- [6] ITU-T, Q.762 Signalling System No. 7 – ISDN User Part general functions of messages and signals, ITU, 1999
- [7] Dryburgh & Hewett, Signaling System No. 7 (SS7/C7): Protocol, Architectures, and Services, Cisco Press, 2005
- [8] ITU-T, Q.764 Signalling System No. 7 – ISDN User Part signalling procedures, ITU, 1999
- [9] ITU-T, Q.763 Signalling System No. 7 – ISDN User Part formats and codes, ITU, 1999

[10] Canonical Ltd., Ubuntu PC Operating System,
<http://www.ubuntu.com/desktop>, fecha de consulta diciembre 2014

[11] The Smarter and Faster Way to Code,
<https://netbeans.org/features/index.html>, fecha de consulta diciembre 2014

[12] Java Team, Development Kit,
<http://www.oracle.com/technetwork/java/javase/jdk-8-readme-2095712.html>,
fecha de consulta diciembre 2014

[13] Apache Friends, Xampp para Windows
<https://www.apachefriends.org/es/download.html>, fecha de consulta
diciembre 2014

[14] Manual de PHP, The Php Group,
<http://php.net/manual/es/function.mysql-connect.php>, fecha de consulta
febrero 2015

[15] Manual de PHP, The Php Group,
<http://php.net/manual/es/function.mysql-db-query.php>, fecha de consulta
febrero 2015