

UNIVERSIDAD INTERNACIONAL SEK

FACULTAD DE ARQUITECTURA E INGENIERÍAS

Trabajo de fin de carrera titulado:

**“IMPLEMENTACIÓN DE PLATAFORMA TECNOLÓGICA PARA
COMUNICACIÓN WEBRTC EN EL CAMPUS MIGUEL DE CERVANTES
DE LA UISEK”**

Realizado por:

ALEXANDER AUGUSTO GONZÁLEZ SÁNCHEZ

Director del proyecto:

ING. EDISON ESTRELLA MSC

Como requisito para la obtención del título de:

**INGENIERO DE SISTEMAS EN INFORMÁTICA Y REDES DE
INFORMACIÓN**

Quito, Junio 2015

DECLARACION JURAMENTADA

Yo, ALEXANDER AUGUSTO GONZÁLEZ SÁNCHEZ con cédula de identidad # 172065134-6, declaro bajo juramento que el trabajo aquí desarrollado es de mi autoría, que no ha sido previamente presentado para ningún grado a calificación profesional; y, que he consultado las referencias bibliográficas que se incluyen en este documento.

A través de la presente declaración, cedo mis derechos de propiedad intelectual correspondientes a este trabajo, a la UNIVERSIDAD INTERNACIONAL SEK, según lo establecido por la Ley de Propiedad Intelectual, por su reglamento y por la normativa institucional vigente.

Alexander Augusto González Sánchez

C.C.: 172065134-6

DECLARATORIA

El presente trabajo de investigación titulado:

**“IMPLEMENTACIÓN DE PLATAFORMA TECNOLÓGICA PARA
COMUNICACIÓN WEBRTC EN EL CAMPUS MIGUEL DE CERVANTES
DE LA UISEK”**

Realizado por:

ALEXANDER AUGUSTO GONZÁLEZ SÁNCHEZ

Como requisito para la obtención del Título de:

**INGENIERO DE SISTEMAS EN INFORMÁTICA Y REDES DE LA
INFORMACIÓN**

Ha sido dirigido por el profesor

ING. EDISON ESTRELLA MSC

Quien considera que constituye un trabajo original de su autor

Ing. Edison Estrella MSC.

DIRECTOR

PROFESORES INFORMANTES

Los profesores informantes:

Ing. Daniel Ripalda MSC.

Después de revisar el trabajo presentado, lo han calificado como apto para su defensa oral
ante el tribunal examinador.

Ing. Daniel Ripalda MSC

LECTOR

Quito, Junio 2015

DEDICATORIA

A mis padres y hermanos quienes durante toda mi vida han sido apoyo y motivación en mi desarrollo personal y profesional.

AGRADECIMIENTOS

Mi profundo agradecimiento a Dios y a su Hijo Jesucristo por todas las bendiciones que a diario se manifiestan en mi vida y en la de mi familia. A mi madre por todo el amor y apoyo que siempre he recibido de su parte. A mis hermanos por creer en mí y darme su mano cada vez que la he necesitado. A mis amigos, compañeros y profesores con los que compartí a lo largo de mi carrera universitaria experiencias que por siempre recordaré. A mi novia por todo el cariño y por creer en mí en todo momento. Finalmente a todos los familiares que me han acompañado y alentado a seguir adelante.

TABLA DE CONTENIDO

DECLARACION JURAMENTADA.....	ii
DECLARATORIA	iii
PROFESORES INFORMANTES.....	iv
DEDICATORIA	5
AGRADECIMIENTOS.....	6
RESUMEN.....	13
Palabras claves.	13
ABSTRACT.	14
CAPÍTULO I. INTRODUCCIÓN.	15
1.1 El problema de investigación.	15
1.1.1 Planteamiento del problema.....	15
1.1.2 Objetivo.....	17
1.1.3 Objetivos específicos.	17
1.1.4 Alcance.	17
1.2 Marco Teórico.	19
1.2.1 Fundamentos de Telefonía IP	19
1.2.1.1 Introducción a la VOIP.	19
1.2.1.2 Sistema de Telefonía IP Asterisk.	19
1.2.1.3 La función de los protocolos en la VOIP.....	20
1.2.1.4 Protocolos de señalización.	20
1.2.1.5 Protocolos de transporte de voz.	21

1.2.1.6	Protocolos de plataforma IP.....	21
1.2.1.7	Protocolo TCP.....	22
1.2.1.8	Protocolo UDP.	22
1.2.1.10	Códecs.	23
1.2.2	Fundamentos Web.....	23
1.2.2.1	Servidor Web.....	23
1.2.2.3	Protocolo HTTP.....	24
1.2.2.4	Servidores Web más utilizados.....	24
1.2.2.5	Servidor Web Apache	25
1.2.2.6	HTML y HTML5	25
1.2.2.7	Sistema SIPML5.	26
1.2.2.8	WEBRTC.	26
1.2.3	Fundamentos de Seguridad informática.....	26
1.2.3.1	Definición de Seguridad Informática.	26
1.2.3.2	Confidencialidad, Integridad y Disponibilidad.	27
1.2.3.3	Certificados digitales.....	28
1.2.3.4	Firmas Electrónicas.....	28
1.2.3.5	Certificados de Clave Pública.	29
1.2.3.6	Certificados SSL.	29
1.2.3.7	Protocolo SRTP.	29
1.2.3.8	Estándares de cifrado para SRTP.....	30
1.2.4	Fundamentos de LINUX.....	30
1.2.4.1	Proyecto GNU/Linux.	30
1.2.4.2	Distribución Centos.....	31
1.2.4.3	Asterisk sobre Centos.....	31

1.2.5	Fundamentos de Programación.....	32
1.2.5.1	Introducción a la Programación Informática.....	32
1.2.5.2	Lenguajes de programación basados en Software Libre.....	32
1.2.5.3	Lenguaje Java Script.....	32
CAPÍTULO II. MÉTODO		33
2.1.	Análisis.....	33
2.1.1	Estudio Preliminar (Requerimientos).....	33
2.1.1.1	Telefonía IP.....	33
2.1.1.2	Red de datos.....	33
2.1.1.3	Seguridades.....	33
2.1.1.4	Hardware.....	34
2.1.2	Estudio de factibilidad.....	34
2.1.2.1	Operativa.....	34
2.1.2.2	Tecnológica.....	34
2.1.2.3	Económica.....	35
2.2	Diseño.....	36
2.2.1	Esquema general de la solución técnica.....	36
2.2.2	Esquema a nivel de puertos y protocolos.....	38
2.2.3	Esquema a nivel de Seguridad.....	39
2.2.4	Esquema a nivel de Aplicaciones.....	41
2.2.5	Esquema de la integración sobre la infraestructura de la UISEK.....	42
2.2.6	Esquema de los componentes en el Servidor.....	43
CAPÍTULO III. RESULTADOS.....		44
3.1	Construcción. (Descripción de lo realizado en el Proyecto)	44
3.1.1	Instalación del Sistema Operativo.....	44

3.1.2	Configuración de Red e IP Pública.	44
3.1.2.1	Configuración de las Interfaces, rutas e IP Pública.	44
3.1.2.2	Configuración de la Interfaz de Red “eth0” y “eth1”.	45
3.1.2.3	Configuración de DNS.	46
3.1.2.4	Configuración de Hostname y Gateway del Sistema.	46
3.1.3	Instalación de Componentes adicionales. (SSH, OpenSSL, Iptables).	47
3.1.3.1	Instalación mediante repositorios.	47
3.1.4	Instalación del Servidor WEB.	47
3.1.4.1	Instalación de Apache 2.	47
3.1.5	Instalación del Sistema Transcodificador.	48
3.1.5.1	Instalación de Sistema VoIP Asterisk 11.	48
3.1.6	Implementación de Certificados SSL.	48
3.1.6.1	Generación de Certificados SSL auto firmados.	48
3.1.6.2	Creación del Certificado con sus llaves.	48
3.1.7	Implementación de Aplicativo WEB.	49
3.1.7.1	Implementación del Sistema Sipml5.	49
3.2	Implementación. (Puesta en Marcha).	51
3.2.1	Configuración de Asterisk.	51
3.2.1.1	Creación de extensión y troncal SIP.	51
3.2.1.2	Creación de nuevo contexto.	52
3.2.1.3	Configuración del servicio HTTP para WebRTC.	53
3.2.1.4	Configuraciones generales SIP.	54
3.2.1.5	Inclusión de certificados SSL.	54
3.2.2	Configuración del Firewall (Iptables).	55
3.2.2.1	Configuración del archivo “iptables.conf”.	55

3.2.3	Configuración del sistema Sipml5.	55
3.2.3.1	Configuración del índice Sipml5.	55
3.2.4	Integración con la Central VoIP de la UISEK.	57
3.2.4.1	Creación de troncal SIP en el Central VoIP de la UISEK.	57
3.2.4.2	Creación de ruta entrante.	58
3.2.4.3	Puesta en marcha del servicio en el Portal WEB.	58
3.2.5	Pruebas.	59
3.2.5.1	Designación de ambiente de pruebas.	59
3.2.5.2	Modo de funcionamiento.	60
3.2.6	Producción.	60
3.2.6.1	Resultados finales.	60
CAPÍTULO IV. DISCUSIÓN		61
4.1	Conclusiones.	61
4.2	Recomendaciones.	63
REFERENCIAS BIBLIOGRÁFICAS.		64
ANEXOS.		65
Anexo # 1.		65

ÍNDICE DE FIGURAS.

Figura # 1: Protocolos en llamadas SIP	21
Figura # 2: Esquema general WebRTC	37
Figura # 3: Esquema de los puertos y protocolos utilizados por el sistema WebRTC.	38
Figura # 4: Esquema de seguridad del sistema WebRTC.....	40
Figura # 5 : Esquema de aplicaciones del sistema WebRTC.....	41
Figura # 6: Esquema a implementar en la infraestructura de la UISEK.	42
Figura # 7: Esquema de instalación de componentes del sistema WebRTC.	43
Figura # 8: Configuración de la interfaz de Red “eth0”.....	45
Figura # 9: Configuración de la interfaz de Red “eth1”.....	45
Figura # 10: Configuración de Hostname y Default Gateway	46
Figura # 11: Sistema Demo del sistema Sipml5.....	51
Figura # 12: Configuración de troncal y extensión SIP.....	52
Figura # 13: Creación de contexto “[from-click]”.	53
Figura # 14: Configuración de “http_custom.conf”.	53
Figura # 15: Configuraciones generales en “sip_general_custom.conf”.	54
Figura # 16: Inclusión de certificados SSL en la extensión “6002”.....	55
Figura # 17: Inclusión de extensión SIP “6002” en “indexSek.html”.....	56
Figura # 18: Configuración del número a marcar “152” en “indexSek.html”.	56
Figura # 19: Configuración de troncal SIP “6001” en Central VoIP de la UISEK.	57
Figura # 20: Vinculo al sistema WebRTC desde el portal web de la UISEK.....	58
Figura # 21: Interfaz de usuario final del sistema WebRTC.	59
Figura # 22: Diagrama de clases Sipml5.....	65

RESUMEN.

Las comunicaciones mediante Voz IP (VoIP) se enfocan cada día más hacia los servicios web, dando a los usuarios la posibilidad de prescindir de las llamadas convencionales y de los costos que estas implican. Es por esta tendencia y las nuevas tecnologías en los navegadores web, que surgió la posibilidad de integrar las llamadas convencionales al portal web de las empresas para facilitar la comunicación con sus usuarios. La implementación de la plataforma de comunicación WebRTC, es una integración de varios componentes de Software instalados y configurados en un computador que se encuentra alojado en el Data Center de la UISEK. El propósito principal d

e esta integración es brindar un servicio de llamadas en línea a los usuarios, entre el portal web de la UISEK y la central telefónica de la misma. Las llamadas se generan desde un navegador web por medio de la tecnología WebRTC, que no es más que la integración de ciertos protocolos de transporte de datos y de encriptación, incorporados en un API dentro del mismo navegador. El componente principal de la integración es el motor de telefonía IP Asterisk, el cual gestiona la transcodificación y redirección de llamadas entre el navegador web y la central telefónica IP. Todos los elementos de Software usados en la integración son de código abierto y licencia pública; desde el Sistema Operativo hasta la aplicación web. La seguridad es parte importante de esta integración ya que cuenta con un sistema de cifrado para dar confidencialidad a las llamadas, así como también un sistema de Firewall para mitigar ataques. El funcionamiento del sistema es bastante simple y amigable para el usuario, al contar únicamente con dos botones en su interfaz, uno para llamar y otro para colgar. Las pruebas realizadas en la comunicación, arrojaron resultados positivos en cuanto la calidad del audio y la estabilidad de la llamada. La implementación de la plataforma WebRTC establece un canal de comunicación global, gratuito y seguro entre la universidad y sus usuarios.

Palabras claves.

VoIP, Asterisk, WebRTC, API, GNU/Linux, SSL.

ABSTRACT.

The communication through VoIP is currently focused to web services, giving users possibilities to avoid conventional calls and its costs. The trend and new technologies on web navigators let possibility of integrate conventional calls to enterprises web pages, facilitating communications with their users. Implementation of WebRTC communication system is an integration of software components installed, configured and hosted on a computer at UISEK's data center. Main purpose of integration is the online calls service to users between UISEK's web page and UISEK's VoIP telephone exchange. Calls are generated from web navigator through WebRTC technology, that is a data transport and security protocols integration incorporated on web navigator API. Main integration component is Asterisk VoIP engine, it manages every software component between web navigator and VoIP telephone exchange. All of software components used on integration are open source and public general license, from operating system to web application. Security is one of the most important part of integration, because it has an encryption system which provides confidentiality to calls, also it lets firewall support. System operation it's simple and friendly for users, it has only two buttons, one to call another to hang up. Communication testing results are positives about audio quality and stability calls. The WebRTC System set out a global, free and safety communication channel between university and their users.

Keywords.

Web, VoIP, WebRTC, API, Protocol.

CAPÍTULO I.

INTRODUCCIÓN.

1.1 El problema de investigación.

1.1.1 Planteamiento del problema.

Las comunicaciones actualmente han dado un salto significativo en cuanto a su tecnología y movilidad, permitiendo de esta manera unir puntos muy distantes a través de las diferentes redes de telecomunicaciones que existen en el mundo. Sin embargo la comunicación que se maneja a través de Internet (VOIP) aún no estandariza la comunicación entre un navegador web y una central telefónica (PBX), es decir que no se encuentra como un servicio común en los portales web de las empresas un medio que permita realizar una llamada directa hacia su centro de atención al cliente. Lo que el cliente encuentra regularmente son únicamente los números telefónicos de la empresa para así poder comunicarse desde un teléfono convencional o celular, lo cual es útil, pero no en el caso de clientes que ingresan al portal desde otros países, ya que el establecer el contacto representaría un costo significativo por las tarifas en las llamadas internacionales. La respuesta a este problema ha sido la gestión de atención al cliente desde el portal web de la empresa vía Chat en línea. Esta solución es ampliamente utilizada por las empresas del mundo para brindar una atención directa a sus clientes desde sus portales web, sin embargo no están manejando una comunicación directa de audio bidireccional como en una llamada entre dos teléfonos

convencionales, lo cual es una necesidad que muchos clientes tienen al momento de establecer comunicación con una entidad en particular.

El uso de una tecnología basa en VOIP entre el portal web y el centro de atención al cliente supondría una solución más viable a un problema que quizás muchos no se habían planteado por lo remoto que podría significar tener este tipo de tecnología.

En el caso específico de la Universidad Internacional SEK (UISEK), se plantea el problema y la necesidad tanto en el parte académica como administrativa. Empezando por el área de admisiones, ya que al ingresar al portal web de la UISEK lo primero que se despliega es un formulario de admisión, y en el apartado “Contáctenos” se exponen los números telefónicos de los dos campus, es decir que si una persona ingresa al portal web de la UISEK para solicitar información de las carreras debe llenar el formulario o llamar a los números telefónicos.

En otro caso está el exalumno o egresado que se encuentra fuera del país y necesita información que únicamente puede ser proporcionada en los campus de la UISEK, para lo cual el alumno debe llamar a los números de la universidad estableciendo así una llamada internacional que representa un costo según la tarifa aplicada.

Establecemos pues la necesidad de cada usuario de acceder a este tipo de servicio no sólo en la UISEK sino en toda empresa que brinda atención desde su portal web, permitiendo al cliente llamar de forma directa y gratuita desde el navegador web hacia el centro de atención al cliente de la entidad.

1.1.2 Objetivo.

Implementar una plataforma de comunicación basada en WebRTC entre el servidor web de la UISEK y la central telefónica de la misma, para permitir a los usuarios llamar de forma directa y gratuita al centro de atención al usuario de la UISEK desde un navegador web.

1.1.3 Objetivos específicos.

- Garantizar completamente la plataforma mediante el uso de un Firewall y certificados SSL.
- Utilizar únicamente herramientas de Software Libre para la implementación de toda la plataforma.
- Brindar un servicio de calidad, verificado mediante pruebas de integridad, seguridad y disponibilidad.
- Aplicar los conocimientos adquiridos en la facultad y en las prácticas profesionales.

1.1.4 Alcance.

El sistema de comunicación WEBRTC, abarca únicamente la llamada de audio desde el portal web de la UISEK hacia la central telefónica IP de la misma, permitiendo luego de establecida la comunicación, direccionar la llamada hacia cualquier extensión, cola, condición de tiempo, IVR (Interactive Voice Response) o anuncio en el sistema FreePBX. Todos los módulos y componentes en la implementación de esta tecnología son completamente de código abierto y de licencia GPL (General Public License).

Los componentes básicos a instalar para tener un correcto funcionamiento son, el Sistema Operativo Linux-Centos, el motor de telefonía Asterisk y el soporte WebRTC para Asterisk. La comunicación se establecerá con soporte para los protocolos de encriptación DTLS-SRTP y SDES-SRTP para el audio. En el caso de la señalización se utilizará el protocolo WS (Web Socket). Se añadirán certificados de autenticidad SSL auto firmados para el manejo de la encriptación del audio. El códec de compresión de audio a utilizar es G711.

La UISEK proporcionará el ancho de banda necesario para el servicio, en base a la concurrencia de usuarios que estimen manejar, es decir en base al número máximo de llamadas simultaneas que se permitirán, sabiendo que cada llamada ocupa entre 40Kbps y 70 kbps. Todo esto teniendo en cuenta que el número óptimo de llamadas simultáneas se basa en el Hardware proporcionado.

El desarrollo de todo el proyecto se realizará en un equipo PC que cuenta con los recursos de memoria, procesamiento, almacenamiento y Red suficientes para hacer las veces de Gateway transcodificador, entre el servidor web de la UISEK y la central telefónica IP FreePBX de la misma. Se desarrollará una pequeña aplicación en JavaScript basada en el sistema de código abierto SIPML5 para tener un botón de llamar y otro de colgar llamada en el navegador, dicha aplicación se alojará en el servidor web del Hardware a implementar, al cual se direccionarán los usuarios que desean llamar desde el portal web de la UISEK. Se establecerá las reglas del Firewall en el sistema operativo Linux-Centos para permitir únicamente el acceso a los puertos necesarios. Se troncalizará mediante el uso del protocolo SIP a la plataforma WebRTC con la central telefónica IP de la UISEK, y se configurará las rutas de las llamadas en ambas plataformas para darles la dirección correcta. Se establecerá el servicio para al menos tres navegadores web en sus más recientes versiones. La UISEK proporcionará el subdominio para hacer referencia a la IP pública del equipo a implementar.

1.2 Marco Teórico.

1.2.1 Fundamentos de Telefonía IP

1.2.1.1 Introducción a la VOIP.

La voz sobre IP o VoIP consiste en transmitir voz sobre protocolo IP. Dicho así puede sonar simple pero las redes IP fueron diseñadas principalmente para datos y muchas de las ventajas de las redes IP para los datos resultan ser una desventaja para la voz pues ésta es muy sensible a retardos y problemas de transmisión por muy pequeños que estos sean. Por tanto transmitir voz sobre protocolo IP es toda una empresa con muchos problemas técnicos que resolver. Sin embargo la tecnología ha evolucionado y se ha podido resolver notablemente aquellos problemas inherentes a las redes IP que perjudican la calidad de voz.

Edgar Landivar, *Comunicaciones Unificadas Elastix* (2009).

1.2.1.2 Sistema de Telefonía IP Asterisk.

Asterisk es una aplicación para controlar y gestionar comunicaciones de cualquier tipo, ya sean analógicas, digitales o VoIP mediante todos los protocolos VoIP que implementa. Asterisk es una aplicación basada en Software Libre por lo tanto cuenta con las ventajas que ello representa, lo que lo hace libre para desarrollar sistemas de comunicaciones profesionales de gran calidad, seguridad y versatilidad.

Edgar Landivar. *Comunicaciones Unificadas Elastix* (2009).

1.2.1.3 La función de los protocolos en la VOIP.

Hay muchos protocolos involucrados en la transmisión de voz sobre IP incluyendo los protocolos de transporte como TCP y UDP. Sobre ellos se colocan los protocolos de señalización de voz y como si esto fuera poco existen además muchas opciones de protocolos de señalización disponibles, lo que puede hacer que todo suene un poco confuso al principio. Para simplificar las cosas podríamos clasificar a los protocolos utilizados en la VoIP en tres grupos:

- Protocolos de señalización.
- Protocolos de transporte de voz.
- Protocolos de plataforma IP.

Edgar Landivar. *Comunicaciones Unificadas Elastix* (2009).

1.2.1.4 Protocolos de señalización.

Los protocolos de señalización en VoIP cumplen funciones similares a sus homólogos en la telefonía tradicional, es decir tareas de establecimiento de sesión, control del progreso de la llamada, entre otras. Se encuentran en la capa 5 del modelo OSI, es decir en la capa de Sesión. Existen algunos protocolos de señalización, que han sido desarrollados por diferentes fabricantes u organismos, y que se encuentran soportados por Asterisk. Entre los cuales están SIP, IAX, H.323, MGCP, SCCP. Entre estos los más populares en el ámbito de Asterisk son SIP e IAX.

Edgar Landivar. *Comunicaciones Unificadas Elastix* (2009).

1.2.1.5 Protocolos de transporte de voz.

Se refiere al protocolo que transporta la voz propiamente dicha o lo que comúnmente se denomina carga útil. Este protocolo se llama RTP (Real-time Transport Protocol) y su función es simple, transportar la voz con el menor retraso posible. Este protocolo entra a funcionar una vez que el protocolo de señalización ha establecido la llamada entre los participantes.

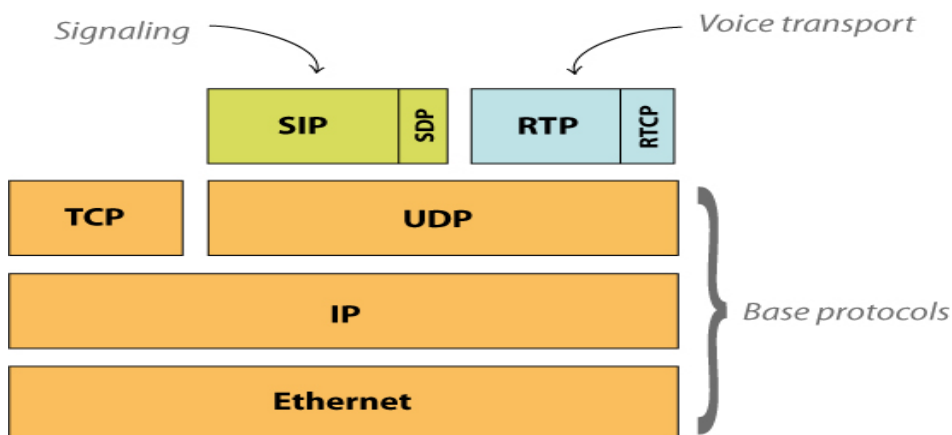
Edgar Landivar. *Comunicaciones Unificadas Elastix* (2009).

1.2.1.6 Protocolos de plataforma IP.

En esta categoría se agrupan los protocolos básicos en redes IP, que forman la base sobre la que se añaden los protocolos de voz, de estos se puede mencionar a Ethernet, IP, TCP y UDP.

Figura # 1: Protocolos en llamadas SIP

Elaborado por: Edgar Landívar.



En la figura anterior se observa al protocolo SIP posando sólo sobre UDP aunque es soportado también por TCP. No se trata de un error sino más bien que en Asterisk la implementación de SIP está únicamente disponible para UDP.

Edgar Landivar. *Comunicaciones Unificadas Elastix* (2009).

1.2.1.7 Protocolo TCP.

Este protocolo se encarga de controlar la transmisión de datos y por esta razón se diseñó lo que se llama “Transmission Control Protocol”. TCP es un protocolo de transporte que se transmite sobre IP. TCP ayuda controlando a que los datos transmitidos se encuentren libre de errores y sean recibidos por las aplicaciones en el mismo orden en que fueron enviados.

José M. Huidobro y David Roldan. *Integración de voz y datos* (2004).

1.2.1.8 Protocolo UDP.

UDP (User Datagram Protocol) es otro protocolo de transporte que se diferencia con TCP en que a este no le importa si los datos llegan con errores o no y tampoco le importa si llegan en secuencia. UDP divide la información en paquetes, también llamados datagramas, para ser transportados dentro de los paquetes IP a su destino, al no ser necesario incluir mucha información de control, el protocolo UDP reduce la cantidad de información extra en los paquetes, por lo que es un protocolo más rápido que TCP y adecuado para transmisión de información que debe ser transmitida en tiempo real como la voz. Es por esta razón que la voz en aplicaciones de VoIP es transmitida sobre este protocolo.

José M. Huidobro y David Roldan. *Integración de voz y datos* (2004).

1.2.1.9 Codificación de la voz.

Para transportar la voz se utilizan algunos protocolos como SIP, IAX y otros como RTP o RTCP, pero la voz es una onda analógica que necesita transformarse a digital en algún formato antes de ser transmitida. Se podría tratar de transmitir la voz tal cual resulta de la conversión analógica digital (ADC) pero al estar en una red de datos se debe empaquetar esta información. Esa búsqueda de un formato óptimo generó algunas alternativas de formatos de transmisión llamadas códecs.

Edgar Landivar. *Comunicaciones Unificadas Elastix* (2009).

1.2.1.10 Códecs.

La palabra códec proviene de abreviar las palabras codificación y decodificación, su función principal es la de adaptar la información digital de la voz para obtener algún beneficio. Este beneficio en muchos casos es la compresión de la voz de tal manera que se pueda utilizar menos ancho de banda del necesario. Algunos códecs, soportados por Asterisk y comúnmente usados en comunicaciones de VoIP, son G.711, G.729, GSM, iLBC, entre otros.

Edgar Landivar. *Comunicaciones Unificadas Elastix* (2009).

1.2.2 Fundamentos Web.

1.2.2.1 Servidor Web.

Un servidor web es una máquina equipada con el software para procesar cualquier aplicación web, realizando conexiones bidireccionales y/o unidireccionales, síncronas o asíncronas generando una respuesta en cualquier lenguaje o aplicación del lado del cliente que

utiliza el protocolo de HTTP para responder a las peticiones de los clientes web en una red TCP/IP.

Lujan M. *Programación de aplicaciones web* (2002).

1.2.2.2 Protocolos Web

Un servidor web es un programa que sirve datos en forma de páginas web, hipertextos o páginas HTML (HyperText Markup Language), textos complejos con enlaces, figuras, formularios, botones y objetos incrustados como animaciones o reproductores de sonidos. La comunicación de estos datos entre cliente y servidor se hace por medio un protocolo, concretamente del protocolo HTTP. Con esto, un servidor web se mantiene a la espera de peticiones HTTP, que son ejecutadas por un cliente HTTP, lo que solemos conocer como un navegador web.

1.2.2.3 Protocolo HTTP.

Una de las características del Protocolo HTTP es la de no ser permanente, es decir, cada operación HTTP implica una conexión con el servidor, que es liberada al término de la misma. Por ejemplo, un documento HTML con 10 imágenes necesita 11 conexiones distintas (10 imágenes más la página HTML en sí).

1.2.2.4 Servidores Web más utilizados.

- Apache
- IIS

- Tomcat
- Cherokee
- Webfsd

1.2.2.5 Servidor Web Apache

Está basado en Software Libre y ha sido diseñado para ser un servidor web potente y flexible que pueda funcionar en la más amplia variedad de plataformas y entornos. Las diferentes plataformas y entornos, hacen que a menudo sean necesarias diferentes características o funcionalidades. Apache se ha adaptado siempre a una gran variedad de entornos a través de su diseño modular. Este diseño permite a los administradores de sitios web elegir qué características van a ser incluidas en el servidor seleccionando que módulos se van a cargar, ya sea al compilar o al ejecutar el servidor. Este es el más común y más utilizado en todo el mundo.

Jorge Sánchez Asenjo. *Servidores de Aplicaciones Web* (2012).

1.2.2.6 HTML y HTML5

Es un lenguaje de descripción de hipertexto compuesto por una serie de comandos, marcas o etiquetas, también denominadas Tags que permiten definir la estructura lógica de un documento web y establecer los atributos del mismo (color de texto, contenidos multimedia, hipervínculos, etc.). HTML5 es la quinta revisión importante del lenguaje básico de la World Wide Web, HTML. HTML5 especifica dos variantes de sintaxis para HTML: un clásico HTML (text/html), la variante conocida como HTML5 y una variante XHTML conocida como sintaxis XHTML5. Herrera. *Arrancar con HTML5* (2011).

1.2.2.7 Sistema SIPML5.

Es el primer cliente SIP para HTML5 basado en software libre y desarrollado completamente en JavaScript para integración en las comunicaciones Web entre Páginas, Servidores Web y dispositivos de telefonía IP que utilicen el protocolo SIP. Siplml5 fue desarrollado por la empresa Doubango.

Siplml5.org. *Guía del programador* (2014).

1.2.2.8 WEBRTC.

WebRTC (Web Real Time Communications) es un proyecto basado en software libre que provee a los navegadores web la capacidad de establecer comunicaciones en tiempo real, para transmisión de datos, voz y video mediante el uso de API's (Interfaz de programación de aplicaciones), que incluyen el motor WebRTC para establecer los canales de comunicación. El proyecto fue diseñado por Google y finalmente lo completó la W3C.

Alan Johnston y Dan Burnett. *WebRTC: The Web Way to Communicate* (2013).

1.2.3 Fundamentos de Seguridad informática.

1.2.3.1 Definición de Seguridad Informática.

La seguridad informática consiste en garantizar que los recursos del sistema de información de una organización se utilizan de la manera que se decidió y que el acceso a la información allí contenida así como su modificación solo sea posible a las personas que se encuentren acreditadas y dentro de los límites de su autorización.

Álvarez Marañón. *Seguridad Informática para empresas y particulares* (2004).

1.2.3.2 Confidencialidad, Integridad y Disponibilidad.

Si bien es cierto que todos los componentes de un sistema informático están expuestos a un ataque, son los datos y la información los sujetos principales de protección de las técnicas de seguridad. La seguridad informática se dedica principalmente a proteger la confidencialidad, integridad y disponibilidad de la información, por tanto, actualmente se considera que la seguridad de los datos y la información comprende 3 aspectos fundamentales:

- *Confidencialidad*
- *Integridad*
- *Disponibilidad*

Hay que tener en cuenta que tanto las amenazas como los mecanismos para contrarrestarla suelen afectar a estas 3 características de forma conjunta por tanto un fallo del sistema que haga que la información no sea accesible puede llevar consigo una pérdida de integridad. Generalmente tiene que existir los 3 aspectos descritos para que haya seguridad. Dependiendo del entorno en el que trabaje un sistema, a sus responsables les interesara dar prioridad a un cierto aspecto de la seguridad. Junto a estos 3 conceptos fundamentales se suele estudiar conjuntamente la autenticación y el no repudio.

Álvarez Marañón. *Seguridad Informática para empresas y particulares* (2004).

1.2.3.3 Certificados digitales.

Los certificados constituyen una tecnología imprescindible para la gestión de datos en línea, contienen datos sobre la identidad de su titular, como su nombre, organización, país, dirección de correo electrónico, etc. Dado que para acceder a ellos es necesario el conocimiento de una clave secreta o el acceso a un dispositivo físico como una tarjeta inteligente, presentándolos se asume que su poseedor es quien el certificado atestigua que es.

Álvarez Marañón. *Seguridad Informática para empresas y particulares* (2004).

1.2.3.4 Firmas Electrónicas.

El fundamento sobre el que se apoyan las firmas electrónicas es la criptografía, esta disciplina matemática no sólo se encarga del cifrado de textos para lograr su confidencialidad, protegiéndolos de ojos indiscretos, sino que también proporciona mecanismos para asegurar la integridad de los datos y la identidad de los participantes en una transacción. El cifrado consiste en transformar mediante un algoritmo un texto que es claro inteligible por todos en un texto cifrado, totalmente ininteligible excepto para el legítimo destinatario del mismo. Se distinguen dos métodos generales de cifrado: el cifrado simétrico y el cifrado asimétrico. En general, el cifrado simétrico, por ser muy rápido, se utiliza para cifrar grandes volúmenes de datos. El cifrado asimétrico, siendo mucho más lento debido a la complejidad de las operaciones matemáticas que realiza, se emplea para cifrar las claves de sesión utilizadas por algunos algoritmos de cifrado simétrico para cifrar documentos.

Álvarez Marañón. *Seguridad Informática para empresas y particulares* (2004).

1.2.3.5 Certificados de Clave Pública.

Las infraestructuras de clave pública PKI (Public Key Infrastructure) constituyen una de las formas más avanzadas y fiables de dotar seguridad a aplicaciones distribuidas en red. Ofrecen un conjunto de aplicaciones y servicios para poder utilizar de forma eficiente y sencilla las complejidades de la criptografía de clave pública. Su mayor logro reside en posibilitar las comunicaciones seguras con otros usuarios o redes sin necesidad de conocerlos con anterioridad ni establecer relaciones previas con ellos.

Álvarez Marañón. *Seguridad Informática para empresas y particulares* (2004).

1.2.3.6 Certificados SSL.

Los certificados SSL (Secure Sockets Layer) son ampliamente usados en la Web, ya que permiten realizar las transacciones de datos más sensibles que se ejecutan en internet, como acceder a una cuenta bancaria o realizar una transacción comercial segura. Su funcionamiento consiste en validar mediante una entidad certificadora, que el servidor web al que se está accediendo es un sitio seguro, además cifra los datos que se transaccionan entre el servidor Web y el usuario mediante una llave pública. Tanto la llave pública como la entidad certificadora las gestiona directamente el navegador Web.

Álvarez Marañón. *Seguridad Informática para empresas y particulares* (2004).

1.2.3.7 Protocolo SRTP.

El protocolo SRTP (Security Real Time Protocol), se utiliza para comunicaciones seguras entre redes de datos, principalmente en aquellas comunicaciones que implican audio y video. El funcionamiento de este protocolo básicamente es el de aplicar una clave simétrica de cifrado al protocolo RTP (Real Time Protocol), que utiliza uno o varios canales de

comunicación entre redes de datos. Dos de estos métodos para la gestión de cifrado en SRTP son SDES y DTLS.

Edgar Landivar. *Comunicaciones Unificadas Elastix* (2009).

1.2.3.8 Estándares de cifrado para SRTP.

DTLS (Datagram Transport layer Security) y SDES (Session Description Protocol Security) son capas de cifrado sobre UDP, que se usan para negociar la clave simétrica del protocolo SRTP, en la comunicación de redes web.

1.2.4 Fundamentos de LINUX.

1.2.4.1 Proyecto GNU/Linux.

Linux es un sistema operativo gratuito y de libre distribución inspirado en el sistema Unix, escrito por Linus Torvalds con la ayuda de miles de programadores en Internet. Unix es un sistema operativo desarrollado en 1970, una de cuyas mayores ventajas es que es fácilmente portable a diferentes tipos de ordenadores, por lo que existen versiones de Unix para casi todos los tipos de ordenadores, desde PC y Mac hasta estaciones de trabajo y superordenadores. Al contrario que otros sistemas operativos, como por ejemplo MacOS (Sistema operativo de los Apple Macintosh), Unix no está pensado para ser fácil de emplear, sino para ser sumamente flexible. Por lo tanto Linux no es en general tan sencillo de emplear como otros sistemas operativos, aunque, se están realizando grandes esfuerzos para facilitar su uso. Pese a todo la enorme flexibilidad de Linux y su gran estabilidad (y el bajo coste) han hecho de este sistema operativo una opción muy a tener en cuenta por aquellos usuarios que

se dediquen a trabajar a través de redes, naveguen por Internet, o se dediquen a la programación.

Javier García de Jalón , Iker Aguinaga y Alberto Mora. *Aprenda Linux* (2000).

1.2.4.2 Distribución Centos.

De sus siglas en Ingles Community Enterprise Operating System, Centos es un Sistema Operativo basado en Red Hat Enterprise Linux, el cual a su vez está basado en el proyecto GNU/Linux. Centos ha sido ampliamente usado en el ámbito de servidores informáticos, catalogándose como un sistema robusto, estable y fácil de instalar y utilizar. Es completamente de código abierto y su actualización y soporte dependen de la comunidad de desarrolladores a nivel global.

1.2.4.3 Asterisk sobre Centos.

El motor de telefonía IP Asterisk, el cual está basado en Software Libre, funciona sobre la mayoría de Distribuciones del Sistema Operativo GNU/Linux. En el caso específico de Centos se podrán encontrar diversos desarrollos e integraciones para la gestión de telefonía IP mediante Asterisk implementadas sobre este sistema operativo, entre los más usados se encuentran, FreePBX, TrixBox, Elastix, y AsteriskNow. Dichos Sistemas han confiado en la estabilidad y compatibilidad de Centos para su óptimo funcionamiento y adaptación de nuevo hardware.

1.2.5 Fundamentos de Programación.

1.2.5.1 Introducción a la Programación Informática.

La programación informática, acortada como programación, es el proceso de diseñar, codificar, depurar y mantener el código fuente de programas computacionales. El código fuente es escrito en un lenguaje de programación. El propósito de la programación es crear programas que exhiban un comportamiento deseado. El proceso de escribir código requiere frecuentemente conocimientos en varias áreas distintas, además del dominio del lenguaje a utilizar, algoritmos especializados y lógica formal.

1.2.5.2 Leguajes de programación basados en Software Libre.

Existen varios compiladores y plataformas de desarrollo basadas en Software Libre, tales como JAVA, PHP, MySQL, Javascripts, entre otros. La mayor parte de los desarrollos basados en Software Libre han sido compilados en las plataformas mencionadas.

1.2.5.3 Lenguaje Java Script.

Es un lenguaje de secuencias de comandos más ampliamente admitido por la mayoría de los exploradores, es utilizado principalmente para añadir funcionalidades a una página web.

Emmanuel Gutiérrez. *JavaScript, conceptos básicos y avanzados* (2009).

CAPÍTULO II.

MÉTODO

2.1. Análisis.

2.1.1 Estudio Preliminar.

2.1.1.1 Telefonía IP.

Para la puesta en marcha de la plataforma de comunicación WebRTC, es necesaria la integración con una central telefónica IP, que además cuente con el soporte de protocolo SIP.

2.1.1.2 Red de datos.

Es fundamental que la infraestructura de Red donde se implementará la plataforma de comunicación WEBRTC esté estructurada y segmentada para voz y datos.

2.1.1.3 Seguridades.

La seguridad para el servicio de llamadas web requiere de un firewall en la red de datos y un firewall local en la infraestructura a implementar. Además los certificados SSL que se requieren para la encriptación del audio, se estipula que serán autogenerados con la posibilidad de comprar certificados verificados por una entidad certificadora.

2.1.1.4 Hardware.

El equipo de Hardware donde se implementará la infraestructura de llamadas web, debe cumplir con las siguientes características:

- 2 GB de memoria RAM.
- 50 GB de Disco Duro.
- Procesador Dual Core.
- 2 Tarjetas de Red de 100Mb.

2.1.2 Estudio de factibilidad.

2.1.2.1 Operativa.

La plataforma de comunicación WebRTC permitirá a todos los usuarios que ingresan al Portal web de la UISEK realizar llamadas de audio de forma gratuita y desde cualquier lugar donde se conecten, directamente hacia el centro de atención al usuario de la universidad. Todo este proceso contará con las seguridades pertinentes para evitar la pérdida del servicio o de la información. Brindando de esta manera un servicio útil e innovador para alumnos, docentes, padres de familia, egresados y postulantes.

2.1.2.2 Tecnológica.

La tecnología WebRTC al ser basada en Software Libre tiene características que la hacen completamente operativa en varios tipos de infraestructura tecnológica. La plataforma web de la UISEK actualmente permite manejar varios servicios online para los alumnos, docentes y administrativos, en cuanto a temas de campus virtual se refiere.

La infraestructura de telefonía IP que posee la UISEK está conformada por dos servidores Apliance DELL tanto en el campus de Carcelén como en el de Guápulo, cada uno con el sistema de comunicaciones FREE PBX basado en el motor de telefonía IP ASTERISK. Ambos servidores se encuentran troncalizados a través del protocolo IAX2, por el cual se permite llamadas entre las extensiones de los dos campus. Por tanto la troncalización mediante el protocolo SIP entre la plataforma WebRTC y la plataforma de telefonía IP de la UISEK es factible. La UISEK cuenta con un enlace dedicado de datos entre ambos campus para el manejo de datos y voz. Dado que existe un sistema de telefonía IP totalmente operativo, se asume que la infraestructura tecnológica de la UISEK cuenta con los requerimientos necesarios en cuanto a red de datos se refiere. La UISEK cuenta con sistemas de Seguridad a nivel de Red que le permiten filtrar contenido y limitar los ataques de usuarios maliciosos, de esta manera cumple con los requerimientos de seguridad para la implementación de la plataforma WebRTC. El Hardware requerido para la implementación será provisto por el departamento de tecnología de la UISEK cumpliendo con los requerimientos estipulados. Por tanto la implementación y administración de la plataforma WebRTC es plausible en la infraestructura tecnológica de la UISEK.

2.1.2.3 Económica.

La implementación en general de la infraestructura de comunicación WebRTC en la UISEK contará con elementos de Software, Hardware y Red, lo cuales requieren- de inversión por parte de la Universidad. A continuación se listarán cada uno de estos elementos.

- Software.
 - Certificados SSL (Opcional).
- Hardware.

- Equipo Dell Dual Core.
- Red.
 - Tarjeta Ethernet auxiliar.
 - IP Pública.

Cada uno de los elementos mencionados representa un costo que asume la UISEK, por un monto aproximado de \$ 1400 dólares americanos.

2.2 Diseño.

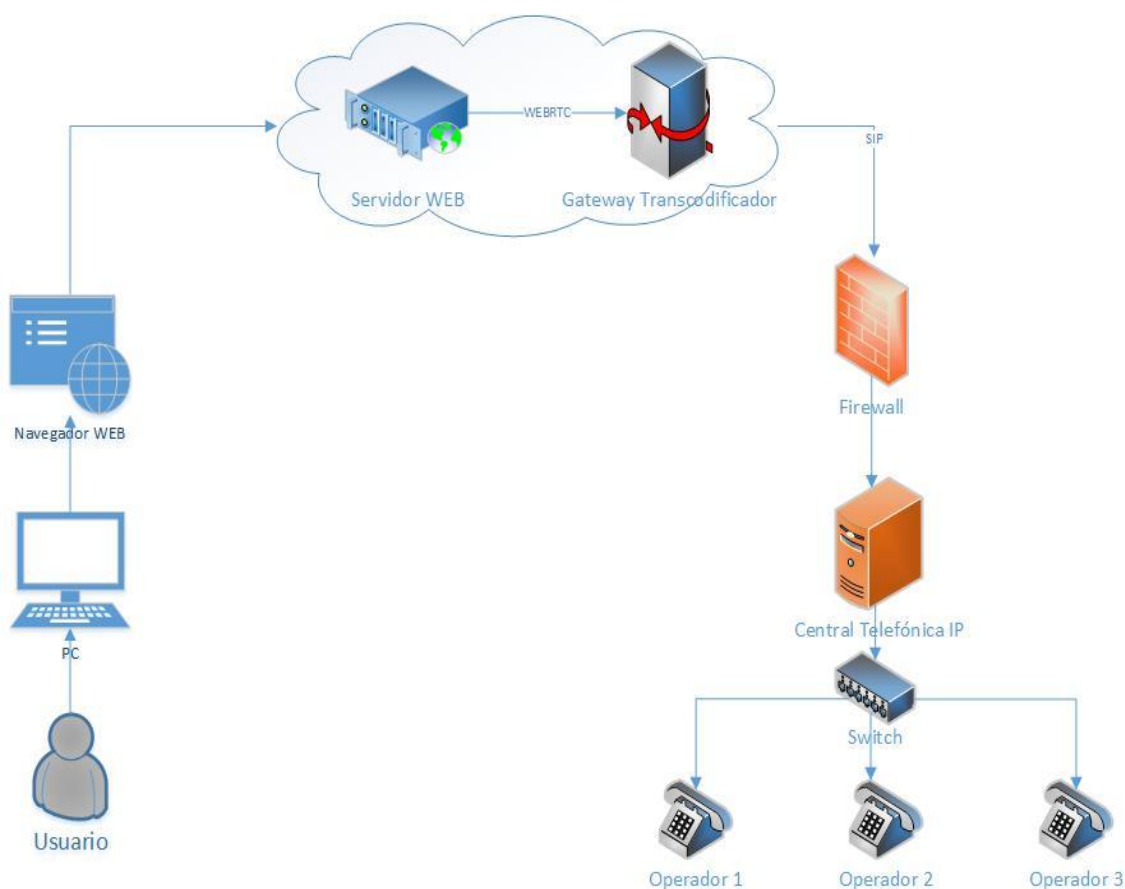
2.2.1 Esquema general de la solución técnica.

La implementación de la plataforma WebRTC en sentido global es una integración de componentes basados en software, que trabajan en conjunto para interactuar en las redes de datos, transmitiendo, codificando y decodificando la comunicación. La integración se ha diseñado de tal manera que la mayor parte de sus componentes se puedan implementar en un solo equipo de hardware, el cual hará al mismo tiempo las veces de servidor web, central VoIP y servidor de infraestructura.

A continuación se ilustra de forma general el funcionamiento del sistema WebRTC.

Figura # 2: Esquema general WebRTC

Elaborado por: Alexander González.



En la ilustración se puede apreciar de forma general como opera el sistema WebRTC. El usuario desde su PC, Laptop, Tablet o Smartphone ingresa al portal web de la universidad y en el apartado “contáctenos” puede llamar directamente dando clic en un botón, inmediatamente la llamada se direcciona desde el Servidor web hacia el Servidor VoIP que hace las veces de Gateway Transcodificador, seguido, la llamada pasa a la troncal SIP establecida con la central VoIP de la universidad, la misma que redirecciona la llamada a sus agentes telefónicos. De esta manera se establece la comunicación desde un navegador web y una Central VoIP a través de WebRTC y el protocolo SIP.

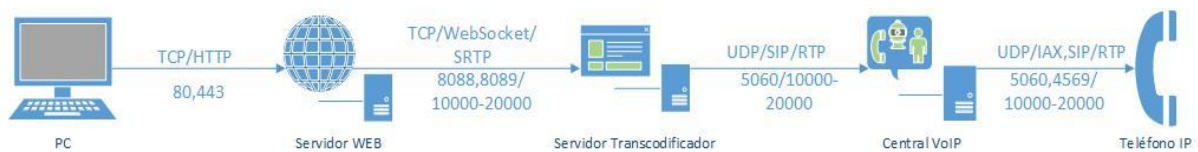
2.2.2 Esquema a nivel de puertos y protocolos.

Para que la integración pueda establecer la correcta comunicación de audio deben interactuar una serie de protocolos a nivel de Red y de VoIP, que definen la señalización, los canales y las seguridades. Los protocolos que convergen en la integración, operan en las distintas capas de los protocolos TCP y UDP. Cada protocolo tiene definido como estándar su propio Puerto en la Red de datos, mediante el cual se podrá filtrar el tráfico en el Firewall de la Red.

A continuación se ilustra el esquema del sistema a nivel de puertos y protocolos.

Figura # 3: Esquema de los puertos y protocolos utilizados por el sistema WebRTC.

Elaborado por: Alexander González.



El proceso para establecer la comunicación mediante WebRTC, a nivel de puertos y protocolos se detalla de la siguiente manera.

- El navegador web realiza la petición de conexión al Servidor web mediante el protocolo HTTP, a través del puerto 80, en el caso de utilizar HTTPS será el puerto 443.
- Cuando el usuario da clic en el botón para llamar, el Servidor WEB se conecta mediante el protocolo WebSocket en el puerto 8088 al Servidor Transcodificador (Asterisk).

- El Servidor Transcodificador direcciona la llamada a la troncal establecida con la Central VoIP a través del protocolo SIP en el puerto 5060.
- La central VoIP redirecciona la llamada a sus agentes telefónicos. Dependiendo de la tecnología de la central utilizará los protocolos SIP, IAX, g729. Etc.
- Una vez establecida la conexión, se establecen los canales de comunicación a través de los protocolos SRTP y RTP los cuales operan en un rango de puertos del 10000 al 20000.

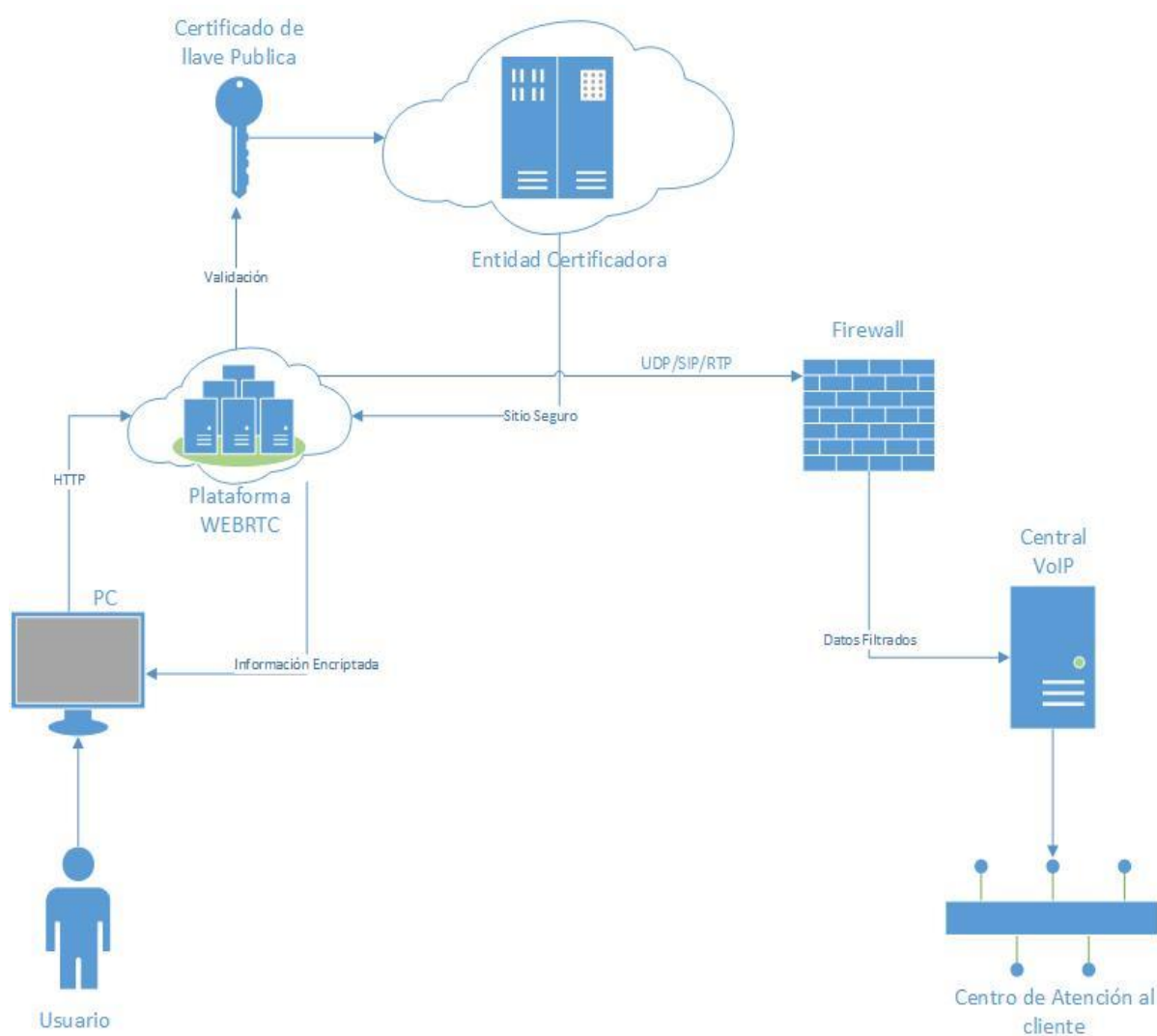
2.2.3 Esquema a nivel de Seguridad.

Las comunicaciones mediante WebRTC establecen como una de sus principales ventajas a la seguridad, ya que el sistema utiliza encriptación de forma obligatoria para realizar las llamadas. El uso de certificados y protocolos seguros, permite que ésta tecnología cifre las comunicaciones entre el navegador web y la central telefónica a la que se conecta.

A continuación la ilustración del esquema del sistema WebRTC a nivel de Seguridad.

Figura # 4: Esquema de seguridad del sistema WebRTC.

Elaborado por: Alexander González.



Como se puede observar en la ilustración, el certificado de llave pública permite validar a través de una entidad certificadora si el servicio web es seguro o no. Además dicho certificado es necesario para la encriptación del audio y la señalización de las llamadas.

2.2.4 Esquema a nivel de Aplicaciones.

Las aplicaciones que interactúan en toda la integración del sistema WEBRTC son las siguientes:

- Navegador WEB:
 - Mozilla Firefox
 - Chrome/Chromium.
 - Opera.
- Servidor WEB:
 - Apache 2.
- API WebRTC JavaScript:
 - SipML5.
- Sistema Transcodificador:
 - Asterisk.
- Central telefónica IP.
 - FreePBX, Asterisk.

Figura # 5 : Esquema de aplicaciones del sistema WebRTC.

Elaborado por: Alexander González.

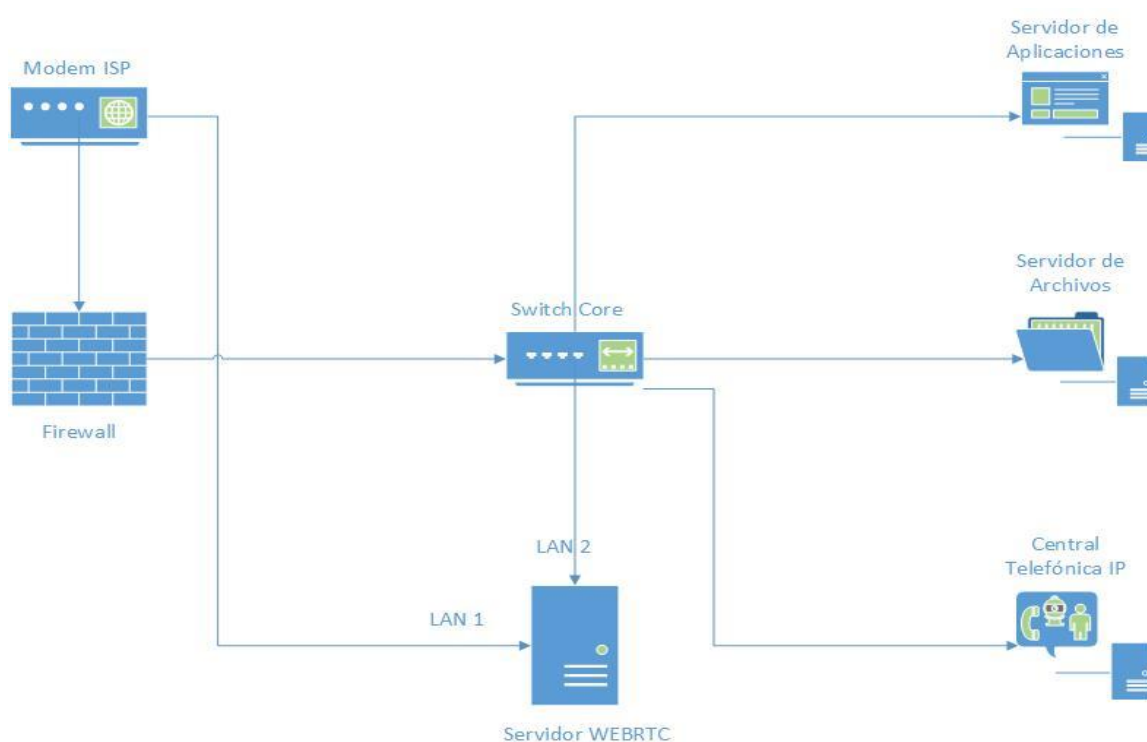


2.2.5 Esquema de la integración sobre la infraestructura de la UISEK.

La plataforma de comunicaciones se implementará en la infraestructura tecnológica de la UISEK, y contará con dos interfaces de Red, las cuales se conectarán al Modem del proveedor de Internet y al Switch de Core respectivamente. La razón por la cual se conectará la plataforma a dos redes distintas, es porque contará con una IP pública que permitirá el acceso al servicio desde internet y a la vez estará conectada al Switch de Core para integrarse con la central telefónica IP de la universidad. A continuación la ilustración para la implementación del sistema en la infraestructura de la UISEK.

Figura # 6: Esquema a implementar en la infraestructura de la UISEK.

Elaborado por: Alexander González.

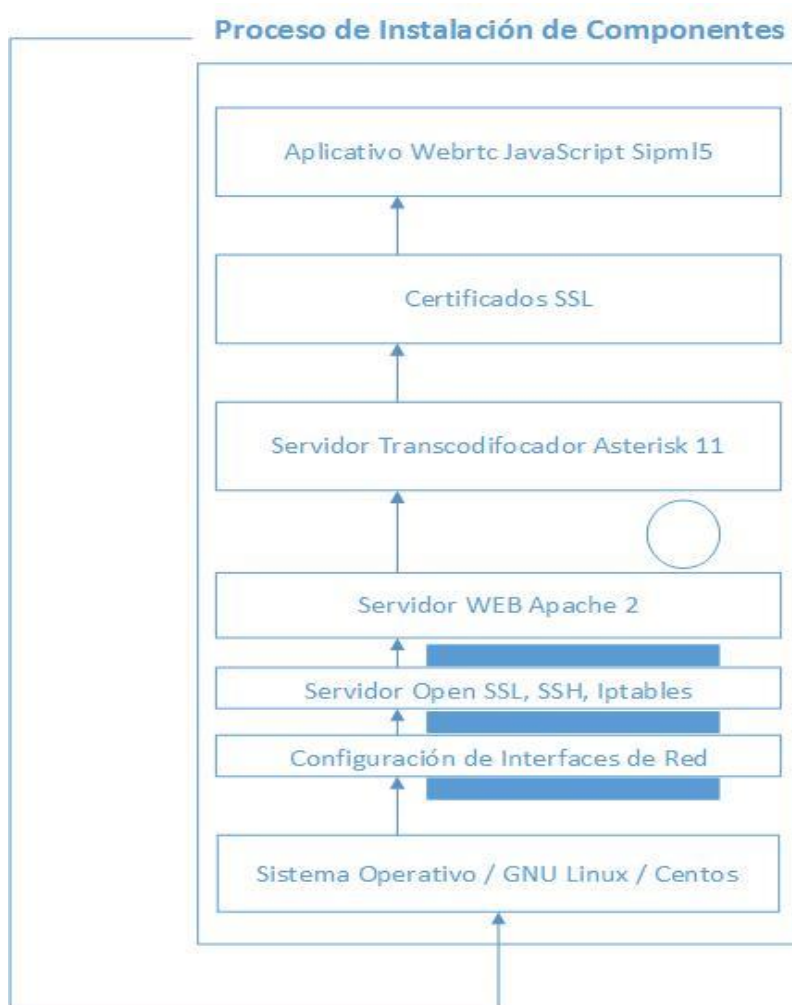


2.2.6 Esquema de los componentes en el Servidor.

Los componentes que comprenden la integración total de la plataforma de comunicaciones WebRTC, serán instalados en un equipo de Hardware que hará las veces de Servidor Maestro para el sistema. La implementación en el equipo mencionado se ilustra a continuación en el orden que se instalará cada componente.

Figura # 7: Esquema de instalación de componentes del sistema WebRTC.

Elaborado por: Alexander González



CAPÍTULO III.

RESULTADOS.

3.1 Construcción.

3.1.1 Instalación del Sistema Operativo.

3.1.1.1 Instalación de la última versión estable de Centos (Centos 7).

La instalación del Sistema Operativo Centos es la implementación inicial del proyecto, que a su vez se constituye en la base de toda la integración. Para efectos de su instalación se debe seguir los pasos de una instalación típica del modo “Servidor de Infraestructura”, de esta manera se obtiene una implementación de Centos lista para recibir la integración de los demás componentes. Los pasos para este tipo de instalación pueden ser consultados en la documentación oficial de del proyecto Centos, en la página www.centos.org/docs/.

3.1.2 Configuración de Red e IP Pública.

3.1.2.1 Configuración de las Interfaces, rutas e IP Pública.

Las configuraciones de Red en Centos y de forma general en GNU/Linux se realizan en archivos de configuración dentro de los diferentes directorios y subdirectorios del Sistema Operativo. En este caso se necesita configurar dos interfaces de red, una con IP local y otra con IP pública.

3.1.2.2 Configuración de la Interfaz de Red “eth0” y “eth1”.

En el directorio “/etc/sysconfig/network-scripts/” se encuentra el archivo “ifcfg-eth0”, dentro del cual se configuran los datos de IP, Mascara de Subred, Red, dirección de Broadcast, entre otras. Por tanto se ingresa con la ayuda de un editor de texto, puede ser a dicho archivo y se realiza la configuración. De la misma manera que se configuró la primera interfaz de red “eth0”, se procede a configurar la Interfaz “eth1”, pero esta vez se lo realiza en el archivo “ifcfg-eth1. A continuación las ilustraciones de cada configuración.

Figura # 8: Configuración de la interfaz de Red “eth0”.

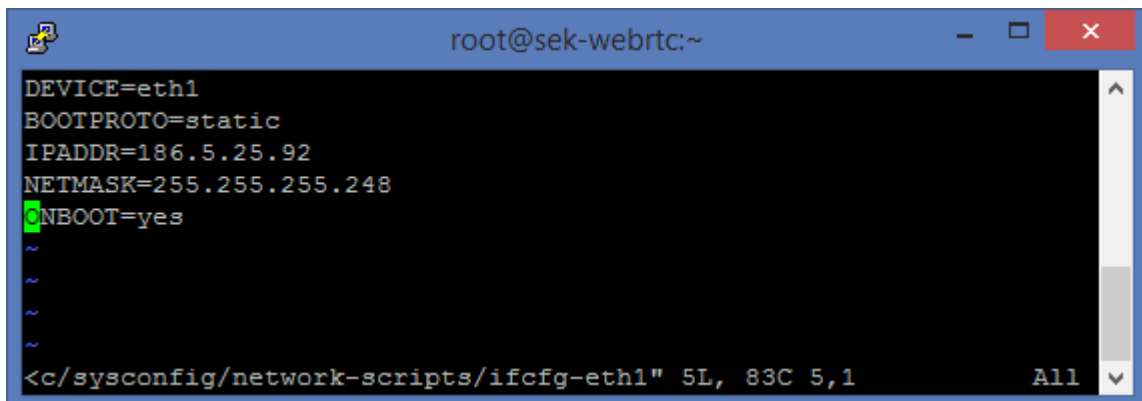
Elaborado por: Alexander González.



```
root@sek-webrtc:~  
DEVICE="eth0"  
BOOTPROTO="static"  
IPADDR=172.16.32.6  
NETMASK=255.255.248.0  
HWADDR="00:1D:09:91:20:AB"  
NM_CONTROLLED="yes"  
ONBOOT="yes"  
TYPE="Ethernet"  
UUID="051668ef-ca89-43a0-b4e9-72db3178b4b9"
```

Figura # 9: Configuración de la interfaz de Red “eth1”

Elaborado por: Alexander González.



```
root@sek-webrtc:~  
DEVICE=eth1  
BOOTPROTO=static  
IPADDR=186.5.25.92  
NETMASK=255.255.255.248  
ONBOOT=yes  
~  
~  
~  
~  
<c/sysconfig/network-scripts/ifcfg-eth1" 5L, 83C 5,1
```

3.1.2.3 Configuración de DNS.

El archivo de configuración para los DNS's se llama “resolv.conf” y se encuentra en el directorio “/etc/”, con el comando del editor de textos se ingresa a dicho archivo y se configura el “NameServer”. La información debe quedar plasmada en dicho archivo de la siguiente manera:

- “nameserver 172.16.32.2”

Aquí también se pueden agregar más servidores de DNS.

3.1.2.4 Configuración de Hostname y Gateway del Sistema.

El archivo para la configuración del Hostname y para el “Default Gateway” del sistema se encuentra en el directorio “/etc/sysconfig” y se llama “network”, una vez dentro del archivo se puede configurar el Gateway del Sistema. Los datos del archivo deben quedar de la siguiente manera.

Figura # 10: Configuración de Hostname y Default Gateway

Elaborado por: Alexander González.

A screenshot of a terminal window with a blue title bar. The title bar contains a small icon on the left, the text "root@sek-webrtc:~" in the center, and standard window control buttons (minimize, maximize, close) on the right. The terminal content shows the following configuration: NETWORKING=yes, HOSTNAME=sek-webrtc, and GATEWAY=186.5.25.89. Below these lines, there are two tilde (~) characters and a green cursor. At the bottom right of the terminal, the text "4,0-1" and "All" are visible next to a scrollbar.

```
root@sek-webrtc:~  
NETWORKING=yes  
HOSTNAME=sek-webrtc  
GATEWAY=186.5.25.89  
~  
~  
4,0-1 All
```

NOTA: Una vez que se ha realizado toda la configuración de Red, se debe reiniciar el servicio de la misma para que se apliquen todos los cambios, esto se realiza ejecutando el comando “service network restart”.

3.1.3 Instalación de Componentes adicionales. (SSH, OpenSSL, Iptables).

3.1.3.1 Instalación mediante repositorios.

La instalación de estos componentes se realiza vía repositorios con la ayuda del servicio de Centos “YUM”, de forma práctica y automática, instalándose cada componente completamente y en su última versión. Los siguientes son los comandos para la instalación de cada componente.

- SSH.
 - “Yum -y install ssh”

- OpenSSL.
 - “Yum -y install openssl”

- Iptables.
 - “Yum -y install iptables”

3.1.4 Instalación del Servidor WEB.

3.1.4.1 Instalación de Apache 2.

De la misma manera que se instalaron los componentes anteriores, la instalación del Servidor Web Apache 2, se puede realizar mediante los repositorios oficiales de Centos, de la siguiente manera:

- “yum -y install apache 2”

Una vez instalado el Servidor Web, se agrega el directorio “/var/www/html/”, en donde se alojan las aplicaciones y páginas web que se ejecutaran en el Servidor.

3.1.5 Instalación del Sistema Transcodificador.

3.1.5.1 Instalación de Sistema VoIP Asterisk 11.

La implementación del motor de telefonía IP Asterisk, es sin lugar a duda el componente más importante de toda la implementación. Existen dos formas de instalar Asterisk, la primera es descargando todos los paquetes y compilarlos en el Sistema Operativo, la segunda es mediante los repositorios oficiales de Centos 7. Para obtener una instalación segura y poder actualizarla posteriormente lo ideal es instalar Asterisk mediante repositorios. El siguiente es el comando para la instalación de Asterisk 11 vía repositorios.

- “yum -y install asterisk-11”

La opción “-y” da una respuesta positiva a todas las preguntas que surgen en el proceso de instalación. Una vez se instala Asterisk en el Servidor, aparece el subdirectorio “/etc/asterisk/”, con todos los archivos de configuración listos para ser modificados.

3.1.6 Implementación de Certificados SSL.

3.1.6.1 Generación de Certificados SSL auto firmados.

Los certificados de cifrado se pueden comprar directamente en una entidad certificadora autorizada en internet, también se los puede generar para auto firmarlos. El sistema WebRTC permite el uso de certificados auto firmados. El servicio OpenSSL permite generar los certificados y firmarlos. A continuación se detalla cómo se generan dichos certificados con el servicio OpenSSL.

3.1.6.2 Creación del Certificado con sus llaves.

La llave privada es la primera en generarse. Una vez creado, el certificado SSL dependerá de esta llave para la implementación del mismo en cualquier servicio que requiera una conexión segura.

En el siguiente ejemplo se muestra el comando para generar una llave privada de 128 bits:

- “openssl genrsa -out server.key 128”

El siguiente paso es la creación del CSR (Certificate Signing Request) que es la base para un certificado SSL, en él se definen datos como el dominio, organización, ubicación, información de contacto, entre otros. Es importante destacar que estos pasos también son necesarios cuando se adquiere un certificado SSL de un proveedor autorizado, durante la gestión del mismo, el proveedor va a solicitar este archivo para crear el certificado. Por lo tanto, se debe tener cuidado de que la información que es ingresada sea la correcta. El CRS se genera utilizando como uno de sus parámetros la llave privada que ya se generó, con la siguiente instrucción:

- **“openssl req -new -key server.key -out server.csr”**

Para generar el certificado SSL se requiere tanto de la llave privada como el CRS que se ha generado. Para generarlo se ejecuta el siguiente comando:

- **“openssl x509 -req -days 365 -in server.csr -signkey server.key -out server.crt”**

Una vez generado el certificado SSL, los tres archivos (server.crs, server.key, server.crt) se almacenarán en un directorio específico. Lo siguiente será unir el certificado de llave privada y el CSR, para obtener el certificado SSL con extensión “.PEM”. Para unir ambos certificados se ejecuta los siguientes comandos:

- **“openssl rsa -in server.key -text > private.pem”**
- **“openssl x509 -inform PEM -in server.crt > public.pem”**

De esta manera se obtiene el Certificado SSL con extensión “.PEM”, y ya está listo para ser utilizado por Asterisk en la interconexión de las llamadas WebRTC.

3.1.7 Implementación de Aplicativo WEB.

3.1.7.1 Implementación del Sistema Sipml5.

El sistema Sipml5 está compuesto por una serie de librerías, que contienen de forma general el código HTML, JavaScript y CSS junto con los archivos de audio e imágenes que usa la aplicación. Sipml5 es un desarrollo de código abierto y cuenta con un portal desde el cual se puede descargar el API como tal, y también el proyecto global como demo, en este caso se procede a descargar e implementar el proyecto completo para luego modificarlo

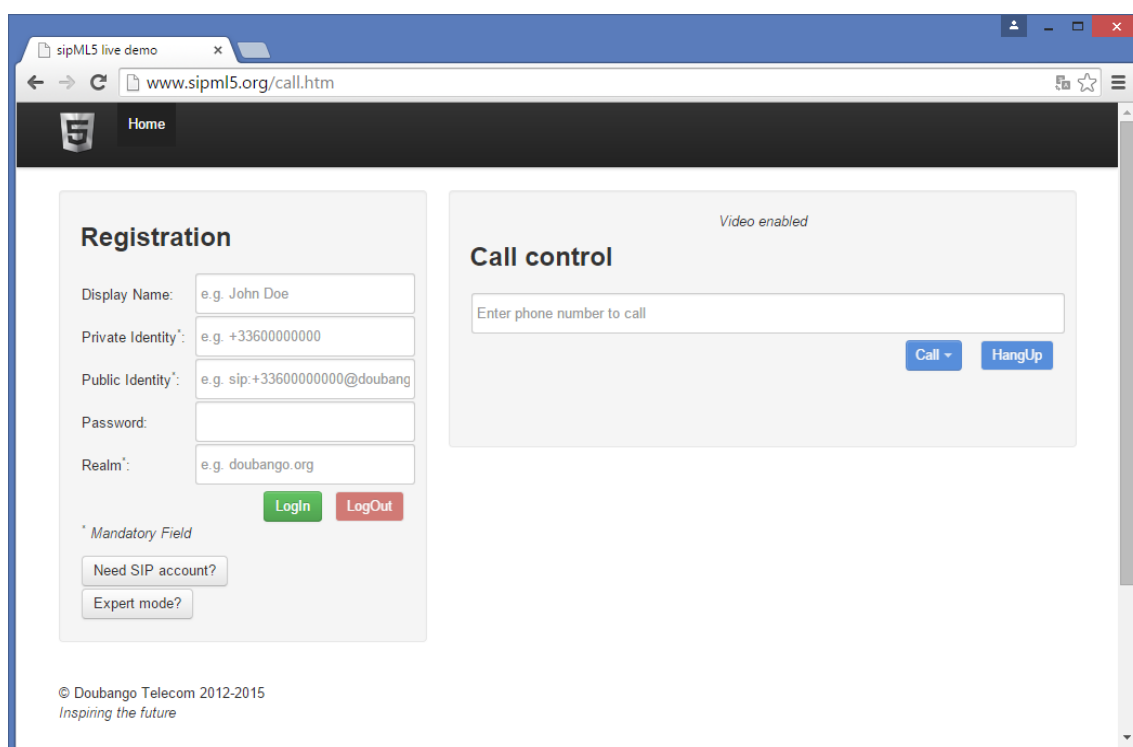
acorde a los requerimientos de la UISEK. A continuación se detallan los pasos para la implementación del sistema Sipml5.

- Lo primero es ingresar al portal web del proyecto Sipml5 y descargar la versión más reciente del API Sipml5:
 - **“<http://sipml5.org>”**
 - ” **<https://code.google.com/p/sipml5/wiki/Downloads>”**
- Una vez descargada la última versión del API Sipml5, se deberá descargar el proyecto demo Sipml5, que incluye todas las librerías para la gestión del aplicativo Web, en la siguiente dirección:
 - ” **<http://sipml5.org/call.htm?svn=230>”**
- Teniendo todos los archivos descargados, se procederá a copiarlos al servidor, dentro de la siguiente librería:
 - **“[/var/www/html/sipml5/](#)”**
- Finalmente se le da los permisos de ejecución a todo el directorio, para que puede ser usado por Asterisk, con el siguiente comando:
 - **“`chown -R asterisk:asterisk /var/www/html/sipml5/`”**
- El sistema Sipml5 está implementado y listo para ser configurado.

Una vez implementado el sistema Demo Sipml5, se lo puede revisar desde el navegador web, ingresando al subdirectorío donde se lo implementó. La siguiente figura ilustra la interfaz web de Sipml5.

Figura # 11: Demo del sistema Sipml5.

Elaborado por: Alexander González.



3.2 Implementación.

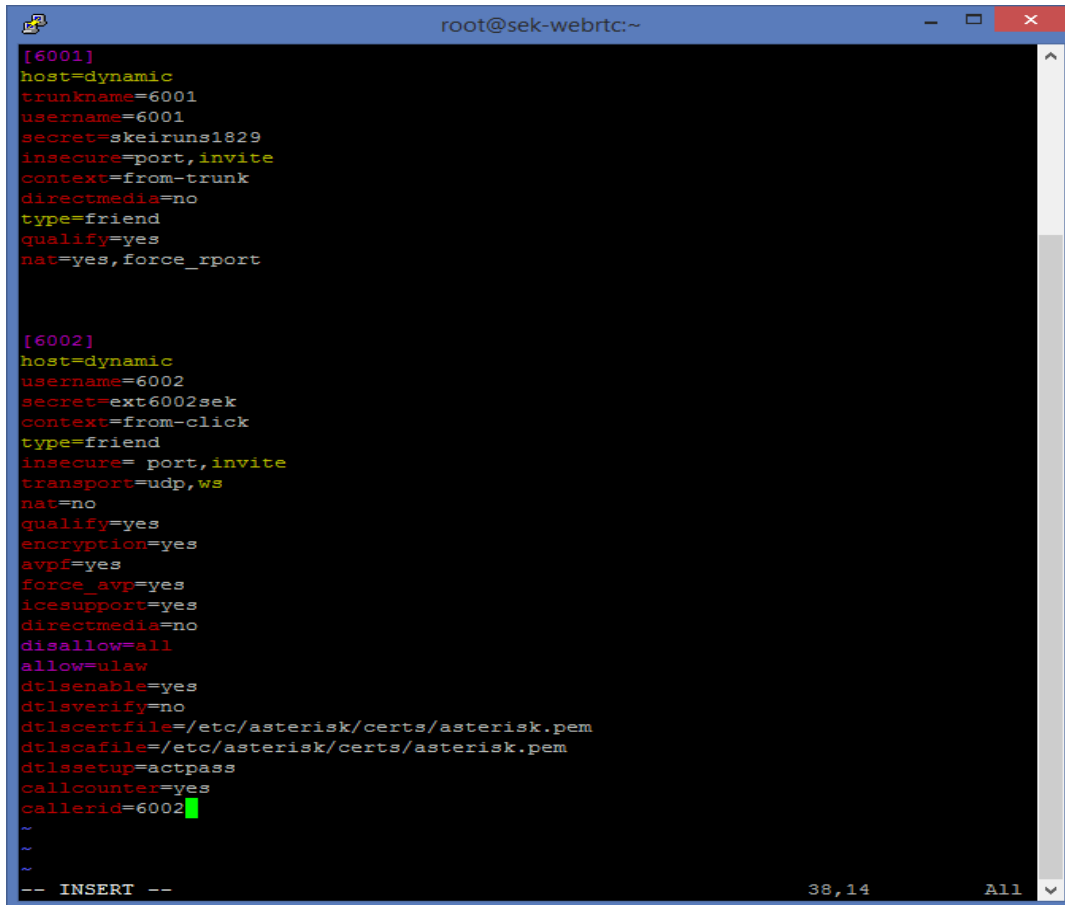
3.2.1 Configuración de Asterisk.

3.2.1.1 Creación de extensión y troncal SIP.

La creación de los objetos SIP como extensiones y troncales se debe configurar directamente en un archivo particular que maneja Asterisk llamado “sip_custom.conf”, el cual se encuentra en el subdirectorio “/etc/asterisk/”. La siguiente figura ilustra la configuración de la extensión y la troncal SIP en el servidor WebRTC de la UISEK. La troncal corresponde al objeto “[6001]” y la extensión al objeto “[6002]”

Figura # 12: Configuración de troncal y extensión SIP.

Elaborado por: Alexander González.



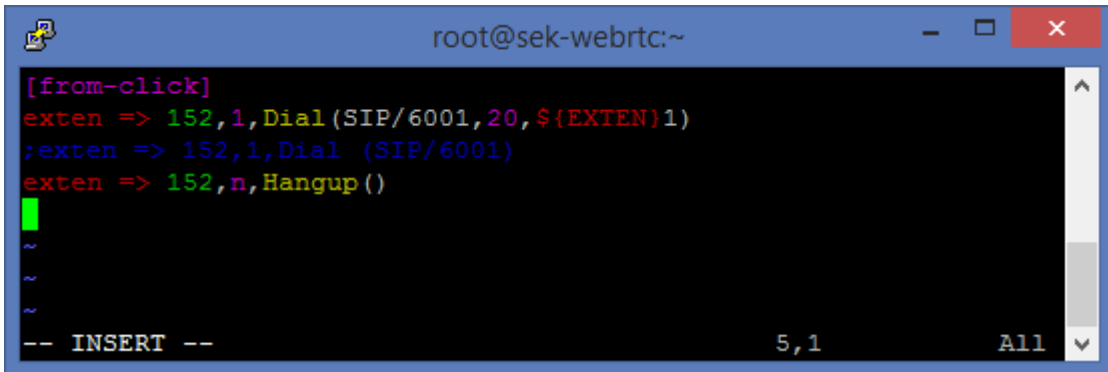
```
root@sek-webrtc:~  
[6001]  
host=dynamic  
trunkname=6001  
username=6001  
secret=skeiruns1829  
insecure=port,invite  
context=from-trunk  
directmedia=no  
type=friend  
qualify=yes  
nat=yes,force_rport  
  
[6002]  
host=dynamic  
username=6002  
secret=ext6002sek  
context=from-click  
type=friend  
insecure= port,invite  
transport=udp,ws  
nat=no  
qualify=yes  
encryption=yes  
avpf=yes  
force_avp=yes  
icesupport=yes  
directmedia=no  
disallow=all  
allow=ulaw  
dtlsenable=yes  
dtlsverify=no  
dtls-certfile=/etc/asterisk/certs/asterisk.pem  
dtls-cafile=/etc/asterisk/certs/asterisk.pem  
dtls-setup=actpass  
callcounter=yes  
callerid=6002  
~  
~  
-- INSERT --
```

3.2.1.2 Creación de nuevo contexto.

El contexto es un parámetro de una extensión o troncal que define el plan de marcado que está va a usar, pero debe ser previamente elaborado en el archivo de configuración “extensions_custom.conf” en el sub directorio “/etc/asterisk/”. El siguiente es el contexto “from-click” creado específicamente para la extensión “6002” del servidor WebRTC de la UISEK y que redirecciona la llamada hacia la troncal “6001” cuando se digita un número específico, en este caso el “152”.

Figura # 13: Creación de contexto “[from-click]”.

Elaborado por: Alexander González.



```
root@sek-webrtc:~  
[from-click]  
exten => 152,1,Dial(SIP/6001,20,${EXTEN}1)  
;exten => 152,1,Dial (SIP/6001)  
exten => 152,n,Hangup()  
~  
~  
~  
-- INSERT -- 5,1 All
```

3.2.1.3 Configuración del servicio HTTP para WebRTC.

Para que Asterisk pueda gestionar este servicio se debe configurar el archivo “http_custom.conf” en el sub directorio “/etc/asterisk/”, para que se habilite el uso del protocolo “http” en el puerto “8080”. El archivo debe estar configurado de la siguiente manera.

Figura # 14: Configuración de “http_custom.conf”.

Elaborado por: Alexander González.



```
root@sek-webrtc:~  
[general]  
enabled=yes  
bindaddr=0.0.0.0  
bindport=8088  
enablestatic=yes  
~  
~  
-- INSERT -- 5,1 All
```

3.2.1.4 Configuraciones generales SIP.

El motor de telefonía IP Asterisk a partir de la versión “11.13”, integra el soporte para el API WebRTC, esta integración básicamente incluye el soporte del protocolo WebSocket y el sistema de cifrado DTLS-SRTP, Los cuales deben ser activados en el archivo “sip_general_custom.conf” en el subdirectorio “/etc/asterisk/” de la siguiente manera.

Figura # 15: Configuraciones generales en “sip_general_custom.conf”.

Elaborado por: Alexander González.

A screenshot of a terminal window titled "root@sek-webrtc:~". The terminal displays the following configuration lines:

```
tcpenable=yes
tcpbindaddr=0.0.0.0
transport=udp,ws
nat=yes,force_rport
icesupport=yes
encryption=yes
avpf=yes
dtlsenable=yes
```

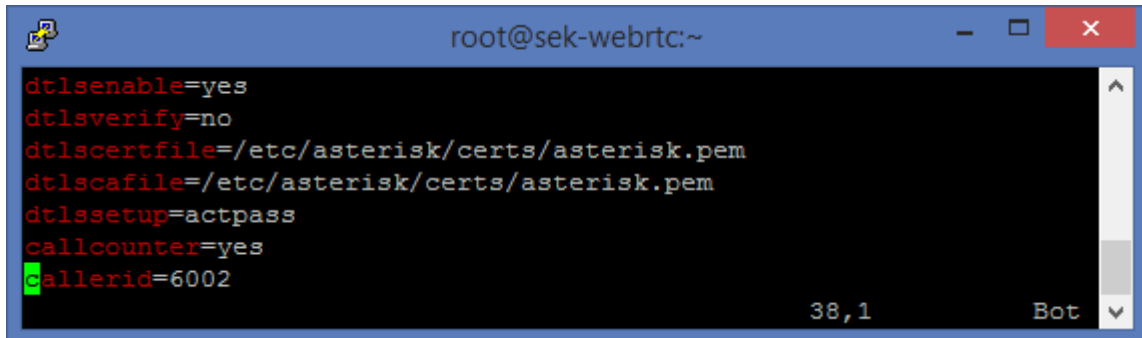
 At the bottom of the terminal, there is a green cursor, the text "-- INSERT --", and the coordinates "9,1" and "All".

3.2.1.5 Inclusión de certificados SSL.

Los certificados de llave pública y privada, se deben incluir en la extensión SIP creada en Asterisk, dentro de los parámetros referentes al protocolo de cifrado DTLS, ya que dicha extensión gestionará el cifrado desde el Servidor Web de cara al usuario final. Los certificados se encuentran configurados en la extensión SIP “6002” de la siguiente manera.

Figura # 16: Inclusión de certificados SSL en la extensión “6002”.

Elaborado por: Alexander González.



```
root@sek-webrtc:~  
dtlsenable=yes  
dtlsverify=no  
dtlscertfile=/etc/asterisk/certs/asterisk.pem  
dtlscafile=/etc/asterisk/certs/asterisk.pem  
dtlssetup=actpass  
callcounter=yes  
callerid=6002  
38,1 Bot
```

3.2.2 Configuración del Firewall (Iptables).

3.2.2.1 Configuración del archivo “iptables.conf”.

La configuración del Firewall en el sistema propio de GNU/Linux “Iptables”, se lo realiza en el archivo de configuración “iptables.conf” en el sub directorio “/etc/sysconfig/”. Para la implementación del servidor WebRTC se permite la entrada y salida de datos únicamente por los siguiente puertos y protocolos.

- SSH: 22
- HTTP: 8080
- WS: 8088
- RTP: 10000 – 20000
- SIP: 5060

3.2.3 Configuración del sistema Sipml5.

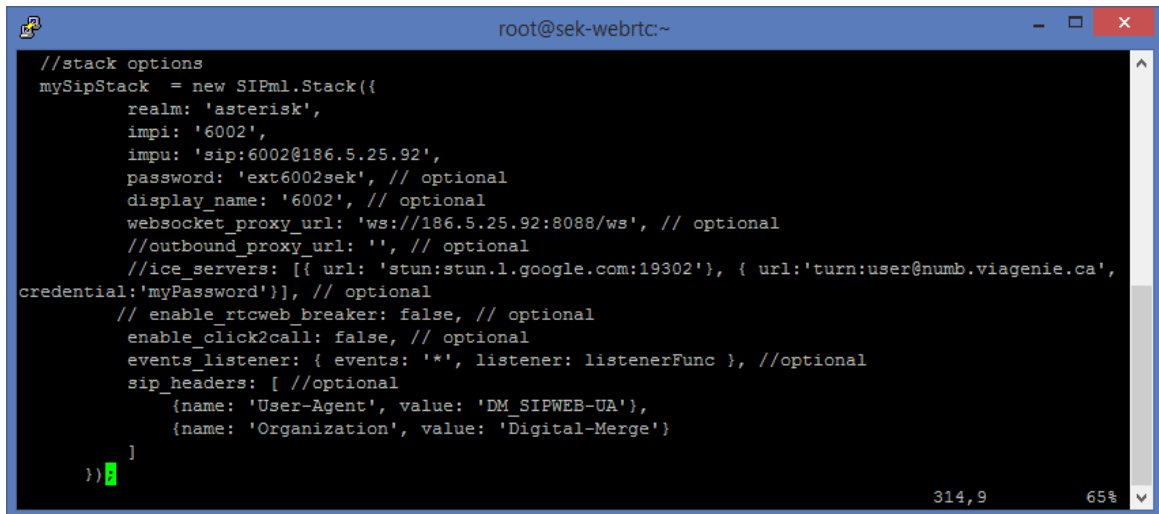
3.2.3.1 Configuración del índice Sipml5.

Para la configuración de sistema Sipml5, se debe elaborar un nuevo Index a partir del que ya integra el sistema, este nuevo Index simplifica las opciones en el sistema a sólo dos botones, uno de “llamar” y otro de “colgar”. El esquema de clases para este pequeño desarrollo se encuentra en el **Anexo # 1**.

Además de simplificar el Index, se debe incluir los parámetros de la extensión SIP “6002” configurada en Asterisk, para que el sistema Sipml5 pueda gestionar la comunicación por medio de dicha extensión. También se debe incluir el número que se define en el contexto “from-click” para que el sistema tenga un número al cual marcar. La configuración de la extensión SIP y del número a marcar en el sistema Sipml5 se la realiza en el nuevo índice dentro del sub directorio “/var/www/html/sipml5/” de la siguiente manera.

Figura # 17: Inclusión de extensión SIP “6002” en “indexSek.html”.

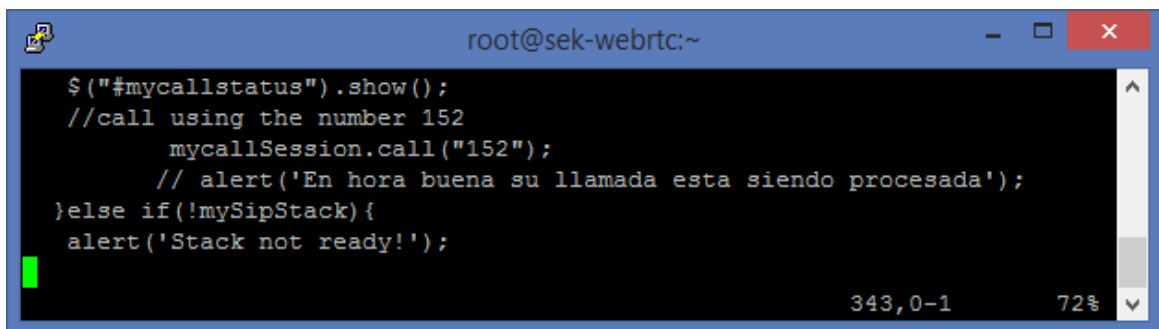
Elaborado por: Alexander González.



```
root@sek-webrtc:~  
//stack options  
mySipStack = new SIPml.Stack({  
  realm: 'asterisk',  
  impi: '6002',  
  impu: 'sip:6002@186.5.25.92',  
  password: 'ext6002sek', // optional  
  display_name: '6002', // optional  
  websocket_proxy_url: 'ws://186.5.25.92:8088/ws', // optional  
  //outbound_proxy_url: '', // optional  
  //ice_servers: [{ url: 'stun:stun.l.google.com:19302'}, { url: 'turn:user@numb.viagenie.ca',  
  credential: 'myPassword'}], // optional  
  // enable_rtweb_breaker: false, // optional  
  enable_click2call: false, // optional  
  events_listener: { events: '*', listener: listenerFunc }, //optional  
  sip_headers: [ //optional  
    {name: 'User-Agent', value: 'DM_SIPWEB-UA'},  
    {name: 'Organization', value: 'Digital-Merge'}  
  ]  
});
```

Figura # 18: Configuración del número a marcar “152” en “indexSek.html”.

Elaborado por: Alexander González.



```
root@sek-webrtc:~  
$("#mycallstatus").show();  
//call using the number 152  
mycallSession.call("152");  
// alert('En hora buena su llamada esta siendo procesada');  
}else if(!mySipStack){  
  alert('Stack not ready!');  
}
```

3.2.4 Integración con la Central VoIP de la UISEK.

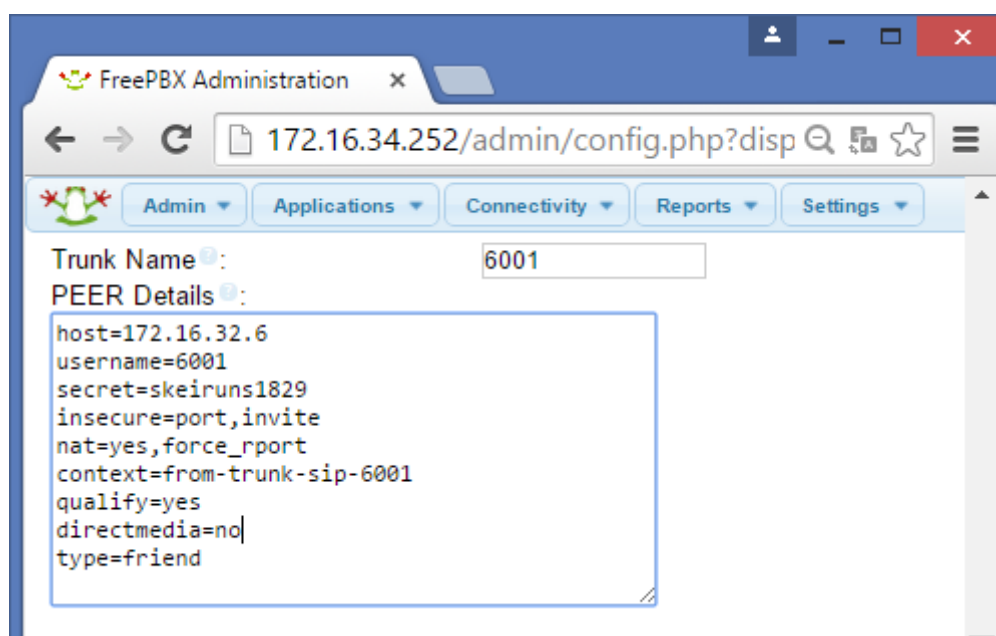
Una vez que el servidor WebRTC se encuentra completamente implementado y configurado, se procede a la troncalización SIP con la central VoIP de la UISEK, para lo cual se requiere la creación y configuración de una troncal SIP y una ruta entrante.

3.2.4.1 Creación de troncal SIP en el Central VoIP de la UISEK.

EL sistema FreePBX es el que gestiona la comunicación VoIP de la UISEK, y para la configuración de la troncal SIP en el mismo se deben establecer parámetros similares a la troncal “6001” creada en Asterisk en el servidor WebRTC, variando únicamente en los parámetros “context” y “register_string”. Por lo tanto la nueva troncal en la central VoIP de la UISEK queda parametrizada de la siguiente manera.

Figura # 19: Configuración de troncal SIP “6001” en Central VoIP de la UISEK.

Elaborado por: Alexander González.



3.2.4.2 Creación de ruta entrante.

La ruta que se debe crear en la central VoIP de la UISEK debe estar condicionada al caller ID del llamante, en este caso el “6002”, que es el caller ID de la extensión SIP creada en el servidor WebRTC. Al condicionar este caller ID se puede establecer el destino que toma la llamada una vez que ingresa, es decir que cuando la central VoIP de la UISEK detecta una llamada entrante del caller ID “6002 la redirecciona hacia un destino también definido en la nueva ruta entrante, que puede ser una cola de atención, un IVR o una extensión en particular.

3.2.4.3 Puesta en marcha del servicio en el Portal WEB.

Una vez completa la integración entre el servidor WebRTC y la central VoIP de la UISEK, se procede con la vinculación al sistema WebRTC desde la página Web de la universidad (www.uisek.edu.ec) en el apartado “Contáctenos”. Las ilustraciones de vinculación y de usuario final se muestran en las siguientes figuras.

Figura # 20: Vinculo al sistema WebRTC desde el portal web de la UISEK.

Elaborado por: Alexander González.

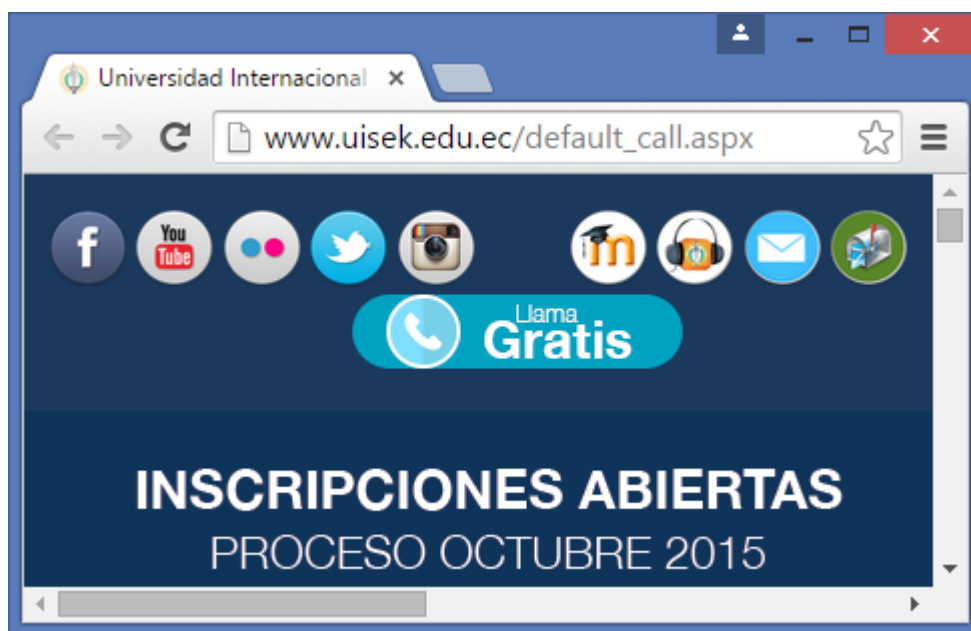


Figura # 21: Interfaz de usuario final del sistema WebRTC.

Elaborado por: Alexander González.



La UISEK proporciona los subdominios “llamada.uisek.edu.ec” y “llamagratias.uisek.edu.ec”, para la IP pública del servidor WebRTC, dichos dominios son los que se vinculan en el portal Web de la UISEK.

3.2.5 Pruebas.

3.2.5.1 Designación de ambiente de pruebas.

El ambiente de pruebas se define de tal manera que el vínculo en el portal Web sólo puede ser accedido por los administradores de TI de la UISEK, para realizar- pruebas de estabilidad en la comunicación, calidad del audio, seguridad, rendimiento del ancho de banda y compatibilidad con los tres navegadores establecidos en diferentes dispositivos. Este ambiente de pruebas se establece en un lapso de tiempo de alrededor de dos semanas antes de salir a producción.

3.2.5.2 Modo de funcionamiento.

El funcionamiento es bastante simple y amigable para el usuario final. Para empezar el usuario que ingresó en el portal Web de la UISEK al apartado “Contáctenos”, puede ingresar al sistema de comunicación WebRTC en un Link que lo especifica, una vez en la interfaz del sistema WebRTC debe hacer click en el botón “llamar”, y dar “aceptar” en el mensaje que el navegador lanza para permitir el uso del micrófono, inmediatamente la llamada se establece y el mensaje de bienvenida de la UISEK empieza a reproducirse hasta que un agente atiende la llamada. Finalmente el usuario puede dar click en el botón “colgar” para terminar la llamada.

3.2.6 Producción.

3.2.6.1 Resultados finales.

El resultado final de las pruebas es positivo en cuanto a la estabilidad, calidad, seguridad, compatibilidad y rendimiento del ancho de banda en las llamadas. Esto permite a los administradores del área de TI de la UISEK poner finalmente a disposición de los usuarios el servicio de llamadas web entre el portal web de la universidad y el centro de atención al usuario de la misma, de forma totalmente global y gratuita, desde los navegadores Chrome, Firefox y Opera.

CAPÍTULO IV.

DISCUSIÓN

4.1 Conclusiones.

- Las ventajas que proporcionan los sistemas de servicios VoIP basados en Software Libre son notables a nivel empresarial. Como premisa está la calidad y estabilidad del servicio, seguido por el ahorro de recursos en Hardware y licencias de Software.
- El motor de telefonía Asterisk al integrar los soportes de encriptación y protocolos de transporte necesarios para la gestión de la comunicación WebRTC, se convierte en un potente sistema transcodificador, que hace las veces de Gateway entre la tecnología WebRTC y el protocolo SIP.
- La seguridad en este tipo de implementación es fundamental para proteger la información transmitida en las llamadas que se originan desde Internet, así como para el blindaje de la plataforma en general. Por lo tanto ha sido de utilidad el uso del Firewall propio de Centos para disminuir la posibilidad de ataques a la plataforma y el uso de los certificados SSL para la encriptación de la comunicación.
- Los conocimientos aplicados en la implementación de esta plataforma están relacionados

con las siguientes áreas de estudio dentro de la Informática:

- Redes de Datos (Networking).
 - Voz sobre IP (Asterisk).
 - Sistemas Operativos (GNU/Linux).
 - Desarrollo Web (HTML, JavaScript).
 - Seguridad Informática (Firewall, Encriptación).
-
- La calidad del audio en la comunicación dependerá entre varios factores del ancho de banda de lado del usuario como del lado del servidor. Aunque el servicio ocupa muy poco ancho de banda (40 kbps – 70 kbps) y no es un problema del lado del servidor por la calidad del enlace que posee la UISEK, si es necesario que el usuario cumpla con el mínimo recomendable de ancho de banda (128 kbps).
 - La disponibilidad del servicio dependerá principalmente de la conexión a Internet que posea la UISEK, esta deberá estar siempre activa, de lo contrario el sistema no será accesible para los usuarios. De esta forma uno de los parámetros para definir el nivel de servicio en cuanto a la disponibilidad, debe estar sujeto al nivel de servicio proporcionado por el ISP.
 - El servidor no almacena información respecto al registro de llamadas que han sido gestionadas por el mismo. Dicha información es almacenada por el sistema VoIP de la UISEK, donde se podrá consultar la información del registro de llamadas recibidas desde el portal Web.

- El número de llamadas simultáneas del sistema se define en base a la capacidad de Hardware respecto al procesamiento y memoria del servidor, así como los requerimientos y estimación de llamadas que se esperan recibir mediante este medio. Por tanto el número de llamadas simultáneas permitidas para la plataforma WebRTC de la UISEK es de cinco.

4.2 Recomendaciones.

- La plataforma WebRTC implementada en la UISEK está compuesta de varios elementos de Software, por lo que necesita llevar un proceso de actualización de los mismos, y así evitar problemas con las versiones de cada componente. Este proceso se puede realizar de forma manual o automatizada.
- Se recomienda el uso de certificados SSL emitidos por una entidad certificadora pública, para brindar seguridad del tipo “HTTPS” en la página Web del sistema, de esta manera se brinda mayor confianza al usuario y se garantiza al servidor Web de la plataforma WebRTC.
- El ancho de banda puede ser optimizado si se utilizan los códecs de compresión de audio G.729 y OPUS, los cuales tienen un costo de licenciamiento, por tal razón no están incluidos en esta implementación.
- El sistema puede aumentar su capacidad de gestión mediante la ampliación de recursos de hardware como memoria y procesador. Si el requerimiento de gestión de llamadas con es del triple o mayor, se recomienda la integración de un nuevo Hardware.

REFERENCIAS BIBLIOGRÁFICAS.

- José M. Huidobro, David Roldan. (2004). Integración de voz y datos, *Call Centers, Tecnología y Aplicaciones*. UISEK.
- Edgar Landivar. (2012). Comunicaciones Unificadas Elastix, *Volumen 1*.
http://www.razametal.org/asterisk/elastix/ElastixBook-Comunicaciones_Unificadas_con_Elastix_Vol1_v0.8.pdf
- Javier García de Jalón • Iker Aguinaga• Alberto Mora. (2000). Aprende Linux, *como si Estuviera en primero*. Escuela Superior de Ingenieros Industriales de San Sebastián UNIVERSIDAD DE NAVARRA.
<http://mat21.etsii.upm.es/ayudainf/aprendainf/Linux/Linux.pdf>
- Alan Johnston – Dan Burnett. (2013). WebRTC: The Web Way to Communicate.
<http://webrtcbook.com/presentations/WebRTCIEEE04-02-13.pdf>
- Doubango.org. Guía del programador (2014).
<http://bookgedebook.tk/pdf.php?file=sipml5-programmer-guide.pdf>
- Jorge Sánchez Asenjo. (2012). Servidores de Aplicaciones Web, Unidad 1, versión 2.01.
<http://www.jorgesanchez.net/web/iaw/iaw1.pdf>
- Álvarez Marañón. (2004). *Seguridad Informática para empresas y particulares*. UISEK
- Sebastián Sánchez Prieto, Oscar García Población. (2009). *Linux Guía práctica*. UISEK
- Herrera. (2011). Arrancar con HTML5. UISEK
- Lujan M. (2002). Programación de aplicaciones web, *historia, principios básicos y clientes web*. UISEK
- Emmanuel Gutierrez (2009). JavaScript, *conceptos básicos y avanzados*. UISEK

ANEXOS.

Anexo # 1.

Esquema de clases del sistema Sipml5.

Figura # 22: Diagrama de clases Sipml5

Elaborado por: Doubango.

