UACM
Universidad Autónoma
de la Ciudad de México
Nada humano me es ajeno

COLEGIO DE CIENCIA Y TECNOLOGÍA

LICENCIATURA EN INGENIERÍA EN SISTEMAS ELECTRÓNICOS
Y DE TELECOMUNICACIONES

Diseño e implementación de una infraestructura de red interna en el LACyTES de la UACM-SLT y desarrollo de una aplicación móvil utilizando servicios web para almacenamiento de datos de sistemas experimentales

TRABAJO RECEPCIONAL
PARA OBTENER EL TÍTULO DE LICENCIADO EN
INGENIERÍA EN SISTEMAS ELECTRÓNICOS Y DE TELECOMUNICACIONES

PRESENTA

MARCO ANTONIO MARTÍNEZ LOREDO

Director del trabajo recepcional

Dr. José Joaquín Lizardi Del Angel

Codirector

Dr. Rogelio Mendoza Pérez

Ciudad de México, abril 2017.

UACM
Universidad Autónoma de la Ciudad de México
Nada humano me es ajeno

CA Coordinación Académica

SISTEMA BIBLIOTECARIO DE INFORMACIÓN Y DOCUMENTACIÓN

UNIVERSIDAD AUTÓNOMA DE LA CIUDAD DE MÉXICO
COORDINACIÓN ACADÉMICA

RESTRICCIONES DE USO PARA LAS TESIS DIGITALES

DERECHOS RESERVADOS©

DEDICATORIA

Elían Zahir Martínez Limón.

Hijo mío desde que llegaste a mi vida, lo cambiaste todo, me convertiste en alguien dispuesto a cambiar por tu felicidad. “la vida está llena de momentos difíciles y de momentos bellos". Aprende de todo lo que puedas y sé el hombre que yo sé que puedes ser. Nunca olvides que te amo

Magdalena Limón Trujillo.

No creo que algún día pueda compensarte por todo lo que has hecho por mí, por brindarme tu gran amor y por todo lo que me has tolerado a lo largo de los años; Por eso, hoy simplemente quiero agradecerte y decirte que significas el mundo para mí. ¡Gracias por hacer de la vida algo hermoso! ¡Te Amo!

AGRADECIMIENTOS

A mi padre **José Luis Martínez Moratilla** quien siempre me alentó a seguir mis estudios y mis sueños. Gracias por ser un gran maestro y un excelente guía en mi vida, por haberme forjado como la persona que soy hoy en día.

Agradezco con respeto y admiración mi director de trabajo recepcional **Dr. José Joaquín Lizardi Del Angel**, quien desde el comienzo de este proyecto creyó en mí y durante el proceso compartió conocimientos que estoy seguro serán de gran ayuda para afrontar los problemas de hoy y del mañana.

De igual manera a mi codirector el **Dr. Rogelio Mendoza Pérez,** por permitirme colaborar en los distintos proyectos que se realizaron dentro del LACyTES, por su confianza, su amistad y su apoyo constante en mi formación académica.

A cada uno de mis lectores, por tener la suficiente paciencia a lo largo de mi formación en mi carrera profesional y por ser parte importante de la elaboración de este trabajo recepcional:

M. en I. Diana Aurora Cruz Hernández
M. en C. José Alfredo Del Oso Acevedo
M. en C. Magali Cortez Vázquez
M. en I. Omar Nieto Crisóstomo

Al profesor Joel Yazbek Buendía Gómez por brindarme el apoyo, los conocimientos, y sugerencias acerca del mi trabajo recepcional. Por permitirme realizar algunos ejercicios y pruebas en la red del laboratorio B-404.

A todos y cada uno de mis compañeros del laboratorio LACyTES, Ing. Francisco Cortez Carreón, Ing. Kevin Arnoldo Jiménez Gómez, Ing. Giovanni Albarran Nuñez, Brenda Hernández, Fanny Ávila, Brenda Reyes García, Ma. De los Ángeles Martínez Rojas y a mí buen amigo Alejandro Romero Pacheco.

A la Universidad Autónoma de la Ciudad de México por abrirme las puertas para poder realizar mis estudios profesionales, por el apoyo otorgado para la impresión y empastado de este trabajo recepcional.

A Conacyt por el apoyo recibido del proyecto: **Fomix CONACYT-GDF 189282 2012-2** "Manufactura de paneles fotovoltaicos de CdS/CdTe en áreas de 100cm² con eficiencia de 8% por la técnica de sublimación y procesos preindustriales asociados"

Marco Antonio Martínez Loredo

Índice

Índice de figuras

Índice de tablas

Resumen

El presente documento describe el proyecto realizado en el Laboratorio de Ciencias y Tecnologías Sustentables (LACyTES) perteneciente a la Universidad Autónoma de la Cuidad de México. En su desarrollo se elaboró el análisis, diseño e implementación de una red de datos interna, donde se aseguran los recursos informáticos mediante la implementación de un cortafuego. Así mismo se realizó un análisis referente a la pérdida de información de bitácoras y a la forma en que se capturan los datos de sistemas experimentales sobre éstas. Para ello, se desarrolló una aplicación móvil con herramientas de software libre para agilizar la captura, el envío, consulta y almacenamiento de los datos en un servidor host, mismo que se encuentra resguardado en la red interna.

Se identificaron las necesidades del laboratorio y se definieron los requerimientos de la red para elaborar el diseño a doc. Fue necesario realizar una selección de las distintas infraestructuras de cortafuego y se eligió la más adecuada acorde al diseño y los recursos. En la red interna se implementó un cortafuego que cuenta con un sistema de encaminamiento y logra cubrir las necesidades del laboratorio integrando zona de acceso a usuarios (LAN), así como una zona DMZ para servidores. Adicionalmente, para el desarrollo de la aplicación móvil se utilizaron diversas tecnologías web, como HTML5, CCS3, Node.js, así como el framework de Phonegap, que permitieron resolver la tardía captura, almacenamiento y consulta de información en el portal web. Además la aplicación permite la elaboración de graficas de información de los sistemas experimentales, así como la sincronización con la base de datos de información de los mismos.

Con el desarrollo de estos trabajos se pudo concluir con un diseño de la red interna donde se aseguraron los recursos informáticos mediante la implementación de un cortafuego logrando obtener una red de datos amoldable a las necesidades futuras del laboratorio. La implementación de la aplicación logra reducir la perdida de datos, además ha aumentado la eficiencia y productividad en la manufactura de módulos fotovoltaicos, sobre todo en el uso de los sistemas donde se aplica el uso de ésta.

Palabras clave: infraestructura de red, servidor host, cortafuego, aplicación móvil, tecnologías web.

Capítulo 1

Introducción

La tecnología aplicada en desarrollo de computadoras ha mostrado un gran avance desde los años 70 y con ello las redes de computadoras, las cuales cumplen con la finalidad de transmitir datos e información. Para que una transmisión sea efectiva y correcta, la red de datos debe contar con una excelente administración, la cual puede ser complicada o sencilla dependiendo de su diseño y tamaño.

Cuando una red de datos es compleja requiere mayor mantenimiento y atención a los problemas que ésta pueda presentar, de esta manera se hace imprescindible contar con un diseño físico y lógico adecuado de una red de datos. Con éstas características se logra una buena gestión, se detectan errores, fallas y se emite una pronta solución para los mismos. Además, debe proveer seguridad en la transmisión de la información, así como brindar una comunicación eficiente tanto para la red del interior como para la red externa.

Hoy en día las empresas cuentan con redes de datos y cada vez más dirigen sus estrategias de marketing, servicios y ventas al uso de las tecnologías de la información y la comunicación (TIC), facilitando la comunicación e interacción con clientes a través de computadoras o dispositivos móviles (teléfonos celulares o tabletas). Por otro lado, las TIC proporcionan un gran apoyo a la investigación por ejemplo, muchos laboratorios de instituciones públicas o privadas obtienen beneficio de estas tecnologías gracias a que ofrecen la posibilidad de almacenar información de nivel prioritario, así como permitir desarrollar programas de fácil interacción, consultar datos; elaborar gráficas, estadísticas, cálculos, entre otros. Todo esto gracias al desarrollo de servicios web y aplicaciones móviles, siendo estas últimas las más usadas actualmente.

1.1 Lugar de la problemática

El Laboratorio de Ciencias y Tecnologías Sustentables (LACyTES), perteneciente a la Universidad Autónoma de la Ciudad de México plantel San Lorenzo Tezonco (UACM-SLT), contribuye a la formación de recursos humanos profesionales, realizando investigación y manufacturación de módulos fotovoltaicos inorgánicos, así como celdas fotovoltaicas orgánicas. Cuenta con computadoras de escritorio y portátiles, algunas de estas contienen hardware y software adicional instalado y desarrollado por la empresa **Newport**, con el objetivo de ayudar al desempeño de la investigación que se realiza en la manufacturación de módulos fotovoltaicos. También cuenta con equipos de interconexión, procesamiento y almacenamiento de información, los cuales ayudan a que ésta sea almacenada y resguardada en un servidor host.

Por otra parte, el laboratorio posee equipos para la manufacturación de módulos fotovoltaicos como el Sistema de Transporte de Vapor en Espacio Cerrado basado en Lámparas Infrarrojas (CSVT-IR) y el Sistema de Sublimación en Espacio Cercano basado en Lámparas Infrarrojas (CSS-IR) dónde se realizan procesos de sublimación de materiales como: CdS, CdTe y $CdCl_2$. Estos sistemas contienen sensores de

temperatura, medidores de presión, flujómetro y bombas de vacío de características semejantes, por lo tanto se puede obtener el mismo tipo de información a través sus pantallas.

Para el proceso de sublimación primero se distribuye cierto peso (mg) del material a sublimar en una base de grafito, sobre de ésta se coloca el substrato, se introduce dentro del sistema y se sella completamente para evitar la entrada de partículas y aire. Como segundo paso, se genera una presión atmosférica (mTorr) por determinado tiempo (min) dentro del sistema, con ayuda de las bombas mecánicas las cuales extraen todo el aire y las partículas. Gracias a un sensor se puede conocer el valor de presión que se crea dentro del sistema. Después usando un flujómetro se introducen algunos gases que logran generar una atmosfera dentro del sistema, con el objetivo formar una reacción química al momento de sublimar el material. Finalmente se procede a iniciar el depósito de material, esto se logra elevando las temperaturas Temp. fuente y Temp. substrato con las lámparas infrarrojas que hay dentro del sistema; los controladores de éstas permiten ver en su pantalla la temperatura programada y el cambio de ésta en cada instante.

1.1.1 Sistema de captura

Los datos desplegados a través de la pantalla de cada sensor son capturados de manera escrita en un formulario que recibe el nombre de "hoja de control" que se muestra en la figura 1 (a) y posteriormente aproximadamente un 22% de los datos son transcritos por medio del formulario del portal web del laboratorio, que se muestra en la figura 1 (b).

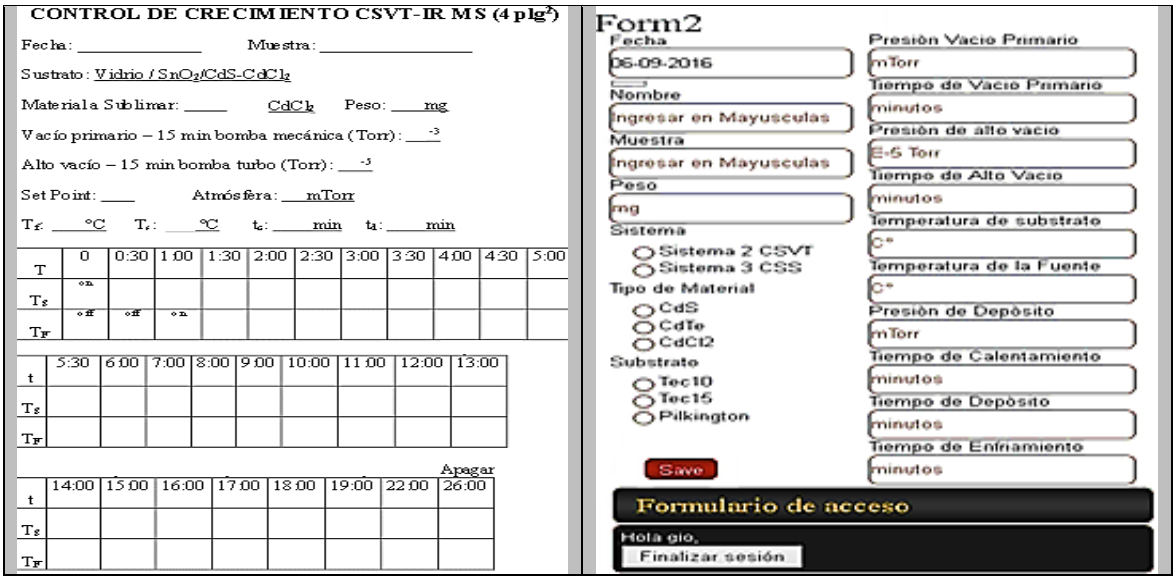


Figura 1 (a) Hoja de control. (b) Formulario del portal web

Una vez que los datos son capturados en el portal web, se almacenan en una base de datos y pueden ser consultados, siempre y cuando el usuario cuente con el nivel de acceso requerido. Como se observa en la figura 1 (b) los datos de Setpoint, atmósfera y temperatura por tiempo de depósito de la hoja de control no se capturan en el portal web; sin embargo; estos datos son necesarios para conocer y dar seguimiento al proceso de manufacturación de módulos fotovoltaicos. Dichos datos son transcritos a un programa de nombre Origin® el cual permite realizar estadísticas y visualizar gráficamente el comportamiento del

depósito (ver figura 2) para poder realizar análisis, estudios y posibles cambios en los parámetros de depósitos futuros, esto con la finalidad de mejorar la eficiencia en los módulos fotovoltaicos.

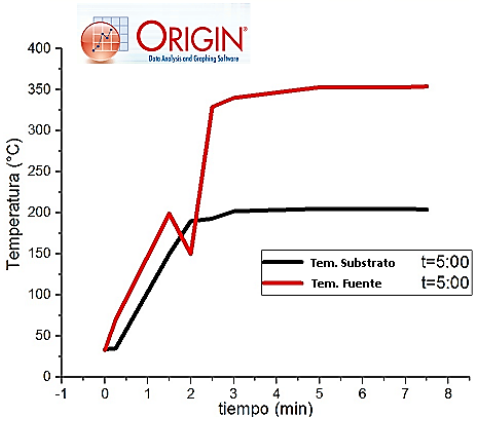


Figura 2 Grafica de Temp vs Teimpo realizada en Origin

Algunas desventajas que se presentan al llevar a cabo estos procedimientos dentro del laboratorio son:

- Repetir nombre de la muestra en hojas de control y en el portal web.
- Pérdida de hojas de control o información.
- Tiempo en transcribir datos al portal web del laboratorio.
- No poder realizar una consulta rápida de los datos.
- No encontrar todos los datos del formulario en el portal web.
- No contar una infraestructura de datos que permita compartir y almacenar información.
- No contar con una red interna de datos que brinde seguridad al momento de transmitir información a usuarios, así como la seguridad de la información de estos.

1.1.2 Estado inicial de la red UACM-LACyTES

En un estudio realizado por José I. Velázquez (2014) se describe lo siguiente: “La UACM cuenta con aproximadamente 100 switchs [1] de 3 distintos fabricantes, esta infraestructura se distribuye en sus 5 campus y otros 3 edificios” [26]. Esta distribución de telecomunicaciones la cual brinda servicios de voz, datos (internet) y video, constituye la red MAN (*Metropolitan Area Network*) de la UACM y distribuye estos servicios a través de MDF (*Main Distribution Facilities*) e IDF (*Intermediate Distribution Facilities*) conectados e instalados en cada uno de los planteles.

El plantel San Lorenzo Tezonco cuenta con una infraestructura de datos que interconecta todas las áreas que hay dentro de éste, incluyendo el LACyTES, donde seis IDF que se distribuyen dentro del laboratorio, brindan conectividad a 5 computadoras y un servidor host. El servidor host del laboratorio, se encuentra conectado al identificador 3P-D55, como se puede observar en la figura 3, al resto de los

[1] Se usara el término de Switch como sinónimo de Conmutador.

IDF se conectan las cinco computadoras también conocidos como sistemas informáticos[2] o sistema terminal (*End System*).

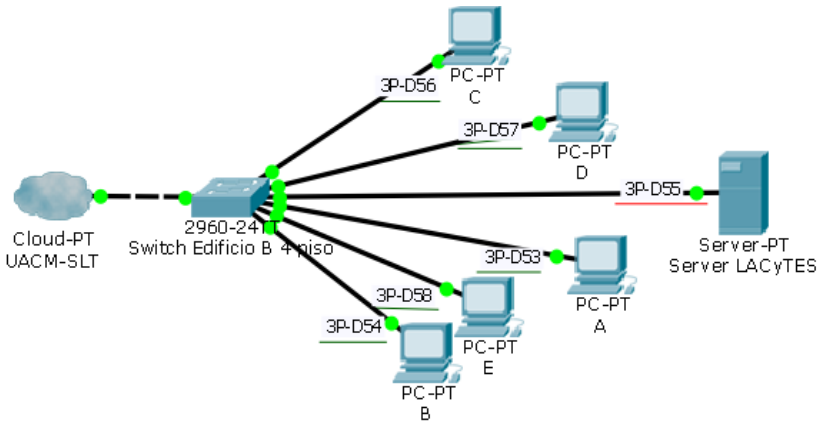


Figura 3 Conexión de computadoras y servidor host, a red de la UACM

Otro punto a abordar es acerca del servidor host el cual ofrece servicios web para dar a conocer a la comunidad las investigaciones del laboratorio. Éste no se encuentra seguro al estar conectado directamente a la red de la UACM, debido a que si por parte de un usuario malicioso se realizara algún tipo de ataque lógico (monitorización, autentificación, DoS, etc.) proveniente de la misma infraestructura, existe la posibilidad que éste elimine, modifique o intercepte los datos del laboratorio como: información de usuarios, resultados de proyectos realizados, investigación de trabajos de grado, entre otros.

Posteriormente la unidad de informática, área encargada de administrar la red de datos, servidores y equipo de cómputo del plantel, modificó y estableció un control a los puertos ethernet a través de la dirección MAC (*Media Access Control*) de cada equipo del laboratorio, con el objetivo de asegurar la conectividad a internet, ejercer un control sobre los sistemas informáticos del plantel y evitar la intrusión de un sistema ajeno a la infraestructura de la UACM, pues de acuerdo a la unidad de informática ya habían detectado el uso incorrecto e indebido de la infraestructura.

Con el tiempo el laboratorio adquirió más equipo de cómputo por lo cual su red se adaptó a las nuevas normas establecidas por la unidad informática. Por consiguiente las nuevas computadoras no podían conectarse a internet, para lo cual como una solución temporal se implementó el uso de un WAP (*Wireless Access Point*) el cual permitiera interconectar los cinco nuevos equipos ya sea por medio cableado UTP o por medio inalámbrico como se observa en la figura 4.

[2] Sistemas que permite procesar, almacenan información y ejecutar programas de usuario, es decir, de aplicación. Estos pueden ser computadoras o servidores.

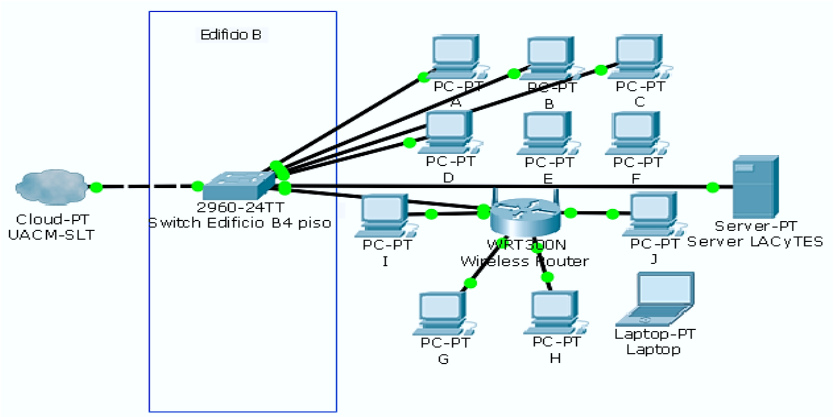


Figura 4 Solución temporal para brindar servicio de internet a los nuevos sistemas

La configuración que de la figura 4 presentaba problemas como: pérdida del servicio de internet, la calidad de los servicios, problemas de ancho de banda, perdida de datos en la transmisión y una comunicación intermitente, lo cual se traduce en una red inestable.

Por otra parte dentro de la adquisición de los nuevos equipos se encuentran los denominados sistemas dedicados y equipos informáticos con mayores características de procesamiento. Los sistemas dedicados son aquellos sistemas de cómputo que se encuentran conectados por medio de una interfaz a un sistema de caracterización, es decir, son sistemas que cuentan con software y hardware adicional al sistema informático, éstos se conectan en tiempos definidos a internet con la finalidad de guardar datos de éstos en el sistema de almacenamiento de OneDrive. Con los equipos informáticos de mayores características de procesamiento se desarrolla un centro de datos, donde se pretende realizar cómputo matemático (mediante el desarrollo de un clúster) además de ofrecer servicios web a la comunidad universitaria.

1.2 Planteamiento del problema

Hoy en día las empresas e instituciones públicas o privadas, debido a la situación económica actual, buscan la optimización y seguridad de sus propios recursos informáticos, infraestructuras tecnológicas, centralización de la información (en caso de tener servicios dispersos), e integración del software que ayude a resolver problemas específicos, como: almacenamiento, organización de información, ventas, entre otros [24].

Las TIC han redefinido el modelo empresarial, pedagógico y de la investigación. A su vez, los rápidos avances en tecnología obligan una actualización constante y de competencia, referidos a la tarea de atender una demanda del entorno [27].

En el LACyTES se lleva a cabo el registro, procesamiento y almacenamiento de información de forma manual y posteriormente en servidores. El estudio de esta información ayuda para realizar mejoras en la manufactura de módulos fotovoltaicos, auxiliar en los objetivos y líneas de investigación del laboratorio; sin embargo, la información no está segura conectada a la red de la UACM, dado que en ocasiones ésta presenta fallas de conexión, además de ser susceptible a fallos y ataques malintencionados, que ponen en riesgo toda la información del laboratorio y sus usuarios.

En la realización del presente trabajo se plantea la siguiente problemática a resolver ¿de qué manera se puede implementar una red de datos interna, confiable y administrable, que integre los diversos sistemas informáticos, las distintas áreas del laboratorio y además se integre una herramienta tecnológica (aplicación móvil) que agilice la captura, almacenamiento, procesamiento y consulta de datos de los sistemas de sublimación de una manera eficiente, fiable y rápida?

1.3 Justificación

Hoy en día los desarrollos tecnológicos pueden facilitar muchas tareas por medio de los dispositivos móviles y sus aplicaciones, debido a que estos proporcionan flexibilidad, facilidad y portabilidad.

Algunas empresas e instituciones han recurrido al desarrollo de aplicaciones móviles propietarias que se usen exclusivamente dentro de su infraestructura de red de datos, protegiendo así toda su información y evitando su extracción. [21]

Con base en la de falta de una red interna y una herramienta de fácil manejo que permita la captura de datos, en el presente trabajo recepcional se decide analizar los problemas existentes de conectividad del laboratorio, de acceso a la información y lo susceptible que son los sistemas informáticos conectados a la red de la UACM. Por otra parte se decide analizar el procedimiento acerca de la captura de datos, su almacenamiento y cómo se elaboran gráficas con determinada información de los formularios.

Dicho lo anterior se brindan propuestas que tengan como propósito auxiliar a las distintas líneas de investigación, por ejemplo, que el laboratorio cuente con un diseño de red adaptable a la distribución física de éste, proteger su información de por medio de una intranet[3] y elaborar una herramienta tecnológica con el uso de tecnologías web para construir una aplicación móvil que agilice la captura, almacenamiento y procesamiento de información.

Por lo tanto, es indispensable que desde la ingeniería se brinden las herramientas tecnológicas tales como el diseño de redes, modelos para comunicación en las redes de datos, arquitectura de software y desarrollo de aplicaciones móviles, las cuales contribuyan a reducir y evitar la pérdida de información del laboratorio.

1.4 Objetivo general

Analizar los problemas de conectividad del laboratorio con el fin de elaborar el diseño e implementación de una red interna que cuente con criterios de seguridad perimetral en redes mediante el uso de un cortafuego por software. Así mismo, analizar la perdida de información de formularios sobre los sistemas de sublimación; formular el desarrollo de una aplicación móvil con el propósito de ofrecer al usuario la posibilidad de realizar el registro, análisis y la consulta de datos que se obtienen a través de las distintas pantallas que ofrecen los sensores de los sistemas CSVT-IR y CSS-IR.

[3] Es una red informática que utiliza tecnología y protocolos de internet para compartir información o servicios web dentro de una organización. Esta red suele ser interna, en vez de pública como internet.

1.4.1 Objetivos específicos

- Identificar los requerimientos para el diseño y configuración de una red de datos.
- Diseñar la red de datos, asegurar los recursos y crear una infraestructura acorde a las necesidades del laboratorio.
- Analizar la metodología empleada a la captura de datos de los sistemas y con base en ello, diseñar una aplicación de fácil manejo.
- Ampliar la tabla donde se almacenan los datos del formulario y que la aplicación pueda realizar consultas.
- Diseño y construcción de una aplicación utilizando las herramientas de software libre, de tal manera que ofrezca ventajas a los usuarios al consultar la información de cada muestra realizada en los sistemas de sublimación.

1.5 Metodología

Para el presente trabajo recepcional se llevó a cabo la siguiente metodología:

1.- Revisión de literatura acerca de los temas presentes en la problemática definida; determinar las bases y delimitar el tema de estudio.
2.- Realizar una investigación acerca del diseño de redes, protocolos de seguridad, hacer uso de los conocimientos adquiridos en el trayecto de la carrera.
3.- Implementar una infraestructura de red interna salvaguardada por un cortafuego que se encuentre basado en el uso de un sistema operativo de distribución libre. Dentro de éste establecer reglas de acceso de manera externa como interna.
4.- Realizar un análisis del portal web: con base a esto determinar el diseño a implementar para el desarrollo y uso de una aplicación móvil.
5.- Hacer uso de software de diseño, animación; así como herramientas de programación, frameworks de desarrollo.
6.- Implementar seguridad (bajo desarrollo script) en las funciones del servidor (funciones PHP) y dentro de la aplicación.
7.- Desarrollar una aplicación móvil donde se logren realizar las mismas funciones del portal web con algunas diferencias marcadas como, el ingreso del 100% de los datos de los sistemas, captura, almacenamiento y consulta de la información de manera eficiente y rápida.
8.- Desarrollo de pruebas de conectividad, funcionamiento y disponibilidad de la red del laboratorio.
9.- Desarrollo de pruebas tales como ejecución de la aplicación en emuladores, portal web y en un dispositivo móvil con sistema operativo Android.

Con la implementación de esta metodología se verán reflejado los beneficios que brindan los avances tecnológicos como el control de acceso a los servicios internos, protección de información y la portabilidad de información en dispositivos móviles, donde actualmente estos pueden contribuir bastante con desarrollo de trabajos e investigaciones.

Capítulo 2

Marco teórico

En el presente capítulo se presentan los conceptos más relevantes necesarios para el desarrollo, diseño e implementación del éste trabajo. Se realizó una recolección de literatura acerca de los temas y se establece información tanto técnica como general del proyecto, por ejemplo, conceptos relacionados con redes, cortafuegos, diseño web y las herramientas para lograr desarrollar una aplicación funcional.

2.1 Antecedentes

En la última década las redes de datos han pasado a ser parte fundamental para cualquier institución u organización por muy pequeña que ésta sea. A causa de la masificación de redes y tecnologías de la información éstas se encuentran bajo amenazas con ataques continuos como: denegación de servicios, saturación de memoria del sistema, etc. Los cuales afectan el rendimiento de los sistemas informáticos, las comunicaciones y la confiabilidad de la red.

De acuerdo con C. A Gunter (1998) una red, es un sistema de interconexión de computadoras que permite a sus usuarios compartir recursos, aplicaciones, datos, voz, imágenes y transmisiones de video. Las redes pueden conectar a usuarios que estén situados en la misma oficina o en países diferentes. La información de la red; se transmite por un sistema de dispositivos autónomos de red, impresoras y aplicaciones de software, interconectados mediante comunicaciones por cable UTP, fibra óptica u ondas de radio [7].

Como mencionan Diana Calderón, Marín Estrella y Manuel Flores (2011) en su trabajo "Implementación de sistema de gestión de seguridad de la información", para mantener la seguridad de la información y de la red de una empresa se requiere la implementación de un sistema que aborde esta tarea de forma metódica y lógica, a través de una red interna (a veces también denominada como intranet) protegida por un cortafuego[4], además señala que es recomendable agregar seguridad interna en los servicios ofrecidos, llámese aplicación de escritorio, portal web, aplicación móvil, correo electrónico y en cualquier servicio o sistema informático que permita compartir recursos tales como periféricos, impresoras, cámaras entre otros [12].

Con esto se asegura la comunicación en distintos niveles de los sistemas, pero no se descarta que pueda llegar a ser vulnerada, hay que recordar que ningún sistema es seguro aun si carece de conexión a la red, pues para asegurar un sistema también es necesario implementar seguridad a nivel físico, por medio de cuartos de telecomunicaciones, bloqueo de puertos de comunicación, o controlando el acceso a equipo de cómputo.

Por otro lado, de acuerdo con María Macías (2012) el diseño de una red ofrece grandes beneficios para una organización, teniendo en cuenta el diseño de red se minimizan costos y se garantiza la disponibilidad

[4] Servidor de seguridad para una red.

mínima requerida de los servicios provistos por la organización. Además para el diseño de red se establecen algunos requisitos (depende cuales requiera cada organización) como: confiabilidad/redundancia, escalabilidad, manejabilidad y ancho de banda [33].

Por otra parte los dispositivos móviles han alcanzado la potencia de una computadora y son capaces de realizar procesos complejos con gran velocidad. Actualmente estos dispositivos están cambiando hábitos en entorno a la sociedad, el ámbito empresarial, y el ámbito pedagógico. Gracias a esta tecnología es posible acceder a internet desde cualquier lugar, revisar el correo electrónico, realizar comunicaciones entre distintos sistemas, uso del GPS (*Global Position System*), entre otros.

Innovanube (2016) en su publicación de "desarrollo de aplicaciones móviles" recuerda como el software empresarial existe desde mucho antes que se llegaran a popularizar los Smartphones por ejemplo menciona lo siguiente:

> "Las aplicaciones de productividad comienzan a migrar de manera consistente desde los ordenadores a dispositivos móviles con la aparición de las PDAs (*Personal Digital Assistant*). Estos terminales, diseñados como agendas electrónicas, con pantalla táctil y un sistema de reconocimiento de escritura algo imperfecto todavía, contenían todo lo necesario para la organización de tareas: calendario, lista de contactos y bloc de notas."

Ahora con el avance de la tecnología en los teléfonos móviles se comenzaron a integrar programas más complejos y hoy en día se ha incrementado la adquisición de estas herramientas tecnológicas, en diversas áreas (educativas, laborales e industriales). Luis A. Chacón y Huber Siche Ricra (2013) hacen referencia a como algunas universidades de latino América han integrado múltiples servicios de su portal web como: gestión de archivos, cursos, altas y bajas de matrículas, horarios, etc. a una aplicación móvil. Además del cómo algunos de sus laboratorios, como el Instituto Químico Sarria, hacen uso de estas tecnologías para el desarrollo de trabajos de investigación [32].

Por su lado Silvia Carrasco Usano (2015) menciona que los dispositivos móviles hacen que la información importante de empresas o usuarios pueda estar disponible en todo momento, acelerando la toma de decisiones y logrando aumentar la productividad. Esto siempre y cuando se establezcan políticas de seguridad adecuadas, solo así podrá beneficiar a las organizaciones, pues estos dispositivos y las aplicaciones ofrecen gran flexibilidad y eficiencia [42].

La conectividad que proveen los dispositivos móviles es un factor muy importante en cualquier tipo de negocio, por ello muchas empresas han planteado la adopción de estas nuevas tecnologías con el propósito de usarlas en aplicaciones con sus clientes, uso corporativo o aplicaciones de productividad (toma de notas, gestión de tareas o herramientas de comunicación). Los principales peligros dentro de una infraestructura de datos son la pérdida de información por medio de accesos no autorizados o infecciones del software. Por esto, es importante la seguridad informática para las empresas que han adoptado o desean adoptar la movilidad por medio de aplicaciones móviles.

Para el desarrollo óptimo de una aplicación, la empresa u organización debe promover el uso correcto de los dispositivos y delimitar su alcance por medio de su propia infraestructura de red [41].

2.2 Redes de datos

Una red es una serie de dispositivos informáticos conectados por medio físico o cualquier otro medio entre sí. De acuerdo a Douglas E. Comer (2000) la comunicación entre redes puedes dividirse en dos tipos básicos: de circuitos conmutados, a veces llamada orientada a la conexión, y por conmutación de paquetes, a veces llamada sin conexión [13]. Sabemos que las computadoras normalmente utilizan las redes de conmutación de paquetes, es decir la información es enviada a través de la red dividida en pequeñas unidades llamadas 'paquetes' que son multicanalizadas a través de las conexiones entre computadoras de alta capacidad [11, 22]

La finalidad del diseño e implementación de una red informática es lograr compartir información y recursos deseados por los usuarios mediante el uso de protocolos de internet. Los protocolos más comunes con estos fines son:

- TCP/IP: *Transmission Control Protocol/Internet Protocol* se define como el protocolo básico de comunicación, de redes, que permite la transmisión de información en redes de computadoras y servidores. Proporciona la base para muchos servicios útiles, incluyendo correo electrónico, transferencia de ficheros y un acceso remoto.
- ARP: *Address Resolution Protocol* permite realizar tareas cuto objetivo es asociar un dispositivo con IP, a un dispositivo de red, el cual a nivel físico posee una dirección física de red. Este protocolo se utiliza regularmente en dispositivos de red local que se conoce como ethernet que es el entorno más extendido hoy en día.
- ICMP*: Internet Control Message Protocol* es usado para comprobar el estado de internet, informa las incidencias en la red pero no toma ninguna decisión. Los mensajes ICMP viajan en el campo de un datagrama (encapsulamiento de paquete IP); los más importantes se enlistan a en la tabla 1.

Tabla 1 Mensajes del protocolo ICMP

Campo de tipo	Tipo de mensaje ICMP
0	Respuesta de eco
3	Destino inaccesible
4	Disminución del tráfico desde el origen
5	Redireccionar (cambio de ruta)
8	Solicitud de eco
11	Tiempo excedido para un datagrama
12	Problema de parámetros
13	Solicitud de marca de tiempo
14	Respuesta de marca de tiempo
15	Solicitud de información
16	Respuesta de información
17	Solicitud de máscara
18	Respuesta de máscara

- FTP: *File Transfer Protocol*es un protocolo para la transferencia remota de archivos, es decir, este protocolo tiene la capacidad de enviar un archivo digital de un lugar loca a uno remoto o viceversa, lo cual lo hace una forma más sencilla la obtención de archivos.

Debido a que las redes de datos tienen un impacto importante en las actividades de distintas instituciones se establece que deben cumplir características importantes: Confidencialidad, Integridad y Disponibilidad de la Información [35].

Confidencialidad: se refiere a asegurar que la información no pueda estar accesible o a disposición de personas o procesos no autorizados. Buenas medidas que se usan para proteger la confidencialidad de los datos es el uso de control de acceso a los sistemas y el cifrado de la información.

Integridad: se refiere a que la información no debe modificarse, alterarse o eliminarse por ninguna persona o sistema, al menos que alguno de estos tenga esa autorización. Si este principio de seguridad se vea afectado la información puede ser falsificada.

Disponibilidad: se refiere a que los sistemas que ofrecen servicios en la red deben mantener disponible la información en todo momento, esta red debe ser eficiente y los sistemas que almacenan la información deben ser capaz de repararse rápidamente en caso de algún fallo.

2.2.1 Tipos de redes

Los tipos de redes se definen de acuerdo a ciertos criterios: En primera instancia a su extensión geográfica o cobertura global. Otra característica es la velocidad de transmisión pero está ultima no la define claramente como tipo de red, puesto que cada una puede usar el medio de transmisiónque le convenga, ya sea conexión DSL, punto a punto o fibra óptica, pues cada uno cuenta con cierta velocidad de transmisión. De acuerdo a estos criterios las redes se dividen en los siguientes tipos: LAN, MAN, WAN [35].

La figura 5 muestra las coberturas y la forma en que se dividen las redes. Así mismo para fines de este trabajo se describe brevemente cada una para mayor comprensión de estas.

Figura 5 Cobertura de los tipos de red

2.2.2 Redes de Área Amplia (WAN)

Son redes que logran cubrir un área geográfica muy amplia (100 – 1000 km), a menudo un país o un continente. Estas redes cuentan con sistemas informáticos los cuales están conectados a una subred de comunicaciones. La finalidad de la subred es trasportar los mensajes de un sistema host a otros sistemas. Estas redes en su mayoría son construidas para una organización o empresa particular y son de uso privado, otras son construidas por los proveedores de Internet para proveer de conexión a los usuarios.

2.2.3 Redes de Área Metropolitana (MAN)

Estas redes son asociadas con dimensiones del tamaño de una ciudad (4 - 500 km) al igual que una red WAN puede ser de uso privado o público. Gracias a que integra tecnología de redes de área local como de área amplia, proporciona la capacidad de integración de múltiples servicios (transmisión de datos, voz y video) sobre medios de transmisión como fibra óptica, par trenzado y algunas MAN incluyen comunicación por antenas. Pueden llegar a una cobertura más amplia mediante la interconexión de diferentes redes MAN.

2.2.4 Redes de Área Local (LAN)

Las redes de tipo LAN (Local Área Network) se encuentran limitadas a una distancia de 200 metros y si se conjunta con repetidores puede llegar a una distancia de 1 Km, son redes pequeñas y regularmente éstas se encuentran en una única ubicación geográfica, como una oficina, fábrica, un campus, etc. Estas redes son usadas para conectar computadoras personales y otros dispositivos informáticos que permitan compartir recursos e intercambiar información. Para la conexión de estas redes se hace uso de *switchs*.

Finalmente, la conexión de dos o más redes se denomina como *interred*, un claro ejemplo es la red de internet de alcance mundial. Hay que recordar que la distancia es importante para la clasificación de la red, por lo cual es necesario conocer estos conceptos, de esta forma se determina qué tipo de red se desea establecer para el objetivo de este proyecto así como el tipo de interconexión que se determinara con la red de la universidad.

2.3 Elementos de la red de datos

La red de datos se construye con base a dos elementos clave: hardware de red y software [28].

El hardware de red se refiere a los equipos que facilitan y constituye el uso de una red informática, dentro de estos se incluyen los siguientes elementos:

- Estaciones de trabajo: Son sistemas de cómputo o informáticos que acceden al servidor y hacen uso de los recursos que éste proporciona. También son conocidos como sistema en red, sistema receptor, sistema remoto o cliente. En nuestro caso se llamaran sistemas host

- Servidor: Cuando hablamos de un servidor nos referimos a una computadora que está integrada a una red informática con el objetivo de permitir a los miembros de la red la posibilidad de acceder a los servicios que se encuentran dentro de éste (correo electrónico, base de datos, aplicaciones web, información, entre otros) o los recursos que comparta.
- Dispositivos de acceso a la red (o de interconexión). Son dispositivos diseñados para conectar computadoras y periféricos a una red de datos, algunos elementos de conexión son: tarjetas de red, concentradores (*hubs)*, repetidores, encaminadores (*routers)*, conmutadores (*switches*). La mayoría de estos permiten conectar diversos dispositivos a una red y son capaces de enviar datos a través de la red, cumpliendo con el objetivo de compartir recursos y acceder a los servicios que se ofrecen dentro de ésta.
- Cortafuego: Es un dispositivo que se conecta entre dos redes, la red no confiable (red externa) y la red confiable (red interna o local), y aplica una política de seguridad donde la tarea principal es proteger a la red confiable de la red que no lo es. Esto lo logra a base de reglas establecidas por el administrador de dicha red local.
- Cableado: Es el sistema mediante el cual los datos pueden llegar desde un equipo de cómputo a otros que se encuentren conectados a la red. Algunos ejemplos son cable coaxial, cable UTP categoría 3, 4, 5, 5E, 6 y 6E y fibra óptica (multimodales, mono modales).
- Punto de acceso inalámbrico. El aumento de redes inalámbricas ha significado que se haga una vasta oferta de dispositivos de red para lograr la comunicación sin cables. Los llamados dispositivos Wifi pueden pertenecer a dos grandes categorías: dispositivos terminales (Smartphone, Tablet-Pc, Laptops, etc.) y dispositivos de distribución (*routers* inalámbricos, puntos de acceso y repetidores inalámbricos).

Estos elementos se utilizan en este trabajo para definir y elaborar un diseño de red interna de datos, la cual se configurará a través de un cortafuego-*router*, donde a través de un *swtich* o dispositivos de acceso se interconectaran los sistemas host y la red inalámbrica del laboratorio.

El otro elemento clave de una red es el software, este se presenta como un conjunto de aplicaciones las cuales también permiten establecer una conexión lógica entre dos o más dispositivos. Se conocen también como sistema operativo de red (NOS) y aplicaciones de red.

Él NOS permite a los usuarios interactuar de forma más sencilla y eficiente a través de las computadoras, es decir, se encargan de compartir recursos, gestionar cada uno de los procesos básicos del equipo y es la base para que otras aplicaciones se puedan ejecutar, un buen ejemplo de estos sistemas seria Windows y Ubuntu. Estos sistemas operativos son parte elemental de este trabajo, debido a que las aplicaciones de algunos usuarios se ejecutan sobre estos sistemas.

Las aplicaciones de red son programas que se ejecutan sobre él NOS y permite un mejor aprovechamiento de las redes, proporcionan seguridad controlando el acceso a los datos y periféricos de la computadora o dispositivo, un claro ejemplo de estas aplicaciones son las que se encuentran en los cortafuegos de software los cuales permiten filtrar, retener o permitir el paso de la información.

2.4 Modelo ISO/OSI

El modelo de Interconexión de Sistemas Abiertos, mejor conocido como "modelo de referencia OSI" es una propuesta elaborada por la Organización Internacional de Normalización (ISO, por sus siglas en inglés) como primer paso para la normalización, generó un sistema de interconexión abierto de modo que hubo compatibilidad en las redes de computadoras[2, 20].

Se utiliza con frecuencia para elaborar diseño de redes y elaborar la ingeniería de las soluciones de red, es decir, conforma las redes del mundo real, aunque existen diferencias entre la teoría que lo sustenta y la práctica real en la mayoría de las redes. Este modelo divide métodos y protocolos necesarios en una conexión de red en siete distintas capas, como se muestra en la figura 6. Las cuales describen cómo se transmite la información desde las aplicaciones a un equipo de cómputo, a través de los medios de la red hacia una aplicación que se ejecuta en otro equipo [18].

Los principios básicos de este modelo son:

1. Se debe crear una capa siempre que se necesite un nivel diferente de abstracción.
2. Cada capa debe realizar una función específica.
3. La función de cada capa se debe elegir pensando en la definición de protocolos estandarizado internacionalmente.
4. Los límites de las capas deben elegirse a modo de minimizar el flujo de información a través de las interfaces.
5. La cantidad de capas debe ser suficiente para no tener que agrupar funciones distintas en la misma capa y lo bastante pequeña para que la arquitectura no se vuelva inmanejable.

También es necesario señalar que cada capa proporciona sus servicios a la capa superior o a la capa inferior (dependiendo de la dirección que llevan los datos) garantizando así que la información llegue a su destino y sea presentada de forma adecuada.

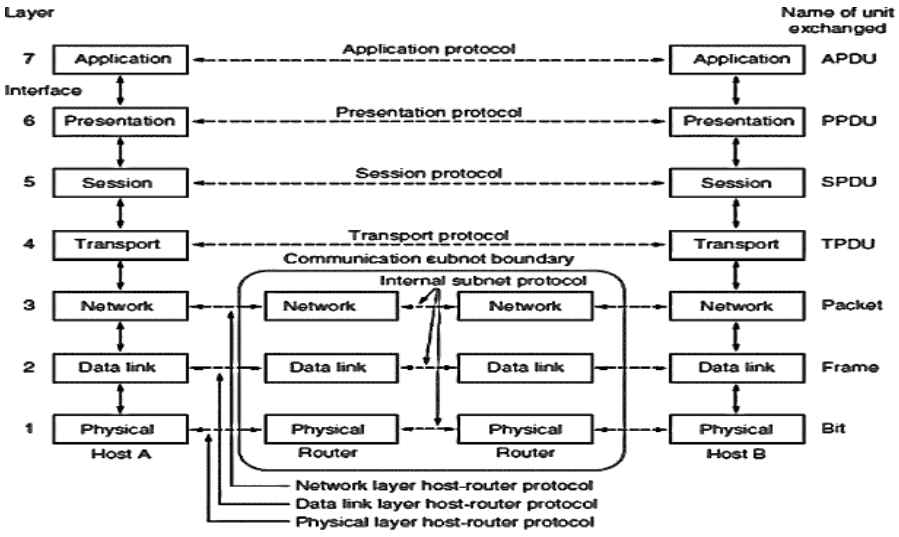


Figura 6 Capas y funcionamiento del modelo OSI

Los principios de cada capa se pueden resumir brevemente de la siguiente manera [4, 2]:

2.4.1 Capa 1 (física)

Define las propiedades del medio físico y mecánico en la transmisión que se utiliza para llevar a cabo la conexión de la red. Esta capa es la encargada de la transmisión de bits a través de un canal de comunicación, mediante medios de transmisión guiados y no guiados, en los cuales se establecen las características materiales (componentes y conectores mecánicos) y eléctricas (niveles de tensión) que se van a usar en la transmisión de los datos por los medios físicos.

Los principales medios guiados son los medios magnéticos, par trenzado, cable coaxial (banda base y banda ancha) y fibra óptica, donde regularmente se emplean señales eléctricas. Así mismo los medios de transmisión no guiados incluyen el espectro electromagnético, radiotransmisión, transmisión por microondas, infrarrojas y haz de luz (láser a través del aire) se hace uso de señales luminosas o de radio.

Esta capa también es responsable por definirse si la transmisión puede o no, ser realizada en dos sentidos simultáneamente, pero esto se precisa en el protocolo que se desea usar. Los protocolos más conocidos en esta capa son: IEE 1394, DSL, RDSI, Bluethooth, GSM, USB y ADSL.

2.4.2 Capa 2 (enlace de datos)

Esta capa es la responsable de la transferencia de mensajes (tramas) a través del canal físico. A su vez toma el medio de transmisión de datos y lo transforma en una línea libre de errores de transmisión para la capa de red. Esto lo realiza, haciendo que el emisor divida y agrupe los datos de entrada en marcos de datos (unos cuantos miles de bytes), posteriormente transmite los marcos en forma secuencial y finalmente procesa los marcos de acuse de recibido (ACK) que devuelve el receptor.

Corresponde a la capa de enlace de datos controlar el flujo de envió de la información, esto lo consigue creando y reconociendo los límites de cada marco. Para ser más específicos añade patrones de bits al principio y al final del marco, es decir, delimita el inicio y final de cada marco para que el receptor reconozca y procese cada trama[5] de forma individual.

La capa de enlace de datos puede ofrecer varias clases de servicio distintas a la capa de la red, se denominan subcapas o subniveles.

Subcapa de control de enlace lógico (LLC – Logical Link Control). Evita que un transmisor de calidad alta en su velocidad de transmisión sature de datos a un receptor que no tiene la misma capacidad, normaliza el tráfico mediante un mecanismo de regulación de tráfico para que el transmisor sepa cuando espacio de almacenamiento temporal (buffer) tiene el receptor en ese preciso momento. Coloca información en la trama que identifica qué protocolos de capa de red está siendo utilizado por la trama.

[5] Unidad de medida de la información en capa de enlace de datos del modelo ISO/OSI, es la segmentación de los datos trasladándolos por medio de paquetes.

Esta información permite que varios protocolos de la capa 3, como IP e IPX, utilicen la misma interfaz de red y el mismo medios de transmisión.

LLC puede ser usado sobre todos los protocolos IEEE del sub nivel MAC, como por ejemplo, el IEEE 802.3 (Ethernet), IEEE 802.4 (*Token Bus*) e IEEE 802.5 (*Token Ring*).

Subcapa de MAC (*Medium Access Control*). Define los procesos para el acceso al medio, proporciona a la capa de enlace de datos el direccionamiento y la delimitación de datos de acuerdo con los requisitos de señalización física del medio y al tipo de protocolo, es decir, distinguir entre una red local o una red global.

Las direcciones locales son asignadas por el administrador de red y no tienen significado fuera de la red local. Las direcciones globales son asignadas por la IEEE (*Institute of Electrical and Electronics Engineers*) para asegurar que no existen dos estaciones (host) en ningún lugar del planeta que contengan la misma dirección global. La tarea de la capa de red es encontrar la manera de localizar el destino.

MAC tiene protocolos importantes como el IEEE 802.3 (Ethernet), IEEE 802.4 (*Token Bus*) e IEEE 802.5 (*Token Ring*). Algunos dispositivos que trabajan en esta capa son: Servidores, computadoras, teléfonos IP, teléfonos móviles, impresoras, tabletas y *Switches* que trabajen con Vlan's.

Los protocolos más conocidos en esta capa son: ARP, PPP, LAPB, SLIP, HDLC, LAPD, Frame Relay e IEEE.

2.4.3 Capa 3 (red)

La capa de red provee los medio funcionales y de procedimiento para que se realice la transferencia de datos, desde un sistema host origen que se encuentra en una red de datos hasta un host de destino, la cual se encuentra en una red de datos diferente, esto lo hace proporcionando procesos importantes para el transporte de extremo a extremo como el direccionamiento, el encapsulado, encaminamiento y desencapsulado de los paquetes a lo largo de las redes.

El direccionamiento se refiere cuando los paquetes o datagramas transmitidos por un sistema host origen deben dirigirse a un sistema host final, este paquete debe contener una dirección IP única a la cual transitara, para lo cual primero se debe hacer un mapeo de direcciones. ARP (Address Resolution Protocol) es uno de los protocolos que permite conocer la dirección de cada host que hay en la red, lo hace enviando un paquete desde el sistema host origen a todas las estaciones de trabajo y pide al host destino responder con su dirección IP. Cuando esto sucede ARP guarda en un *cache* local la dirección, de esta manera antes de enviar un paquete a un host destino, busca en su *cache* si ya tiene la dirección.

Los otros protocolos de control que se utilizan son ICMP que se encarga de reportar errores y mensajes de control en la capa de red, RARP usa un *end system* para encontrar su propia dirección IP, al mapear una dirección física hacia una dirección lógica.

La capa de red debe proveer el encapsulado de los datos debido a que los dispositivos host no deben ser identificados solo con una dirección. Las PDU (Protocol Data Units) agregan un encabezado y una etiqueta donde además de contener la dirección del host destino, contiene la dirección del host origen. A cada capa del modelo OSI le corresponde una PDU siguiendo por lo tanto el orden de encapsulamiento que se muestra en la tabla 2.

Tabla 2 Correspondencia de PDU respecto a capa del modelo ISO/OSI

Capas	PDU
Aplicación	Datos
Presentación	
Sesión	
Transporte	Segmentos
Red	Paquetes
Enlace de datos	Tramas
Física	Bits

Una vez que el paquete está listo la capa de red debe proveer también los servicios de encaminamiento, con el objetivo de que el paquete tenga una ruta óptima y rápida de llegar a su destino. Dentro de esta ruta, puede que el host final no se encuentre dentro de la misma red, por lo cual el paquete debe recorrer diferentes redes a través de dispositivos que se conocen como ruteadores o encaminadores, los cuales ayuden al paquete llegar a su objetivo.

Finalmente, cuando el paquete llega al host destino este examina la dirección de destino para verificar que el paquete fue direccionado a ese host. Si la dirección es correcta el paquete es desencapsulado por la capa de red y la PDU contenida en el paquete pasa hasta el servicio adecuado por medio de la capa de transporte.

Los *routers* son dispositivos capaces de tomar decisiones lógicas con respecto a la mejor ruta para el envío de datos entre redes y redirigen los paquetes hacia el segmento y puerto de salida adecuados. Además pueden tomar decisiones basándose en la densidad del tráfico y el ancho de banda.

2.4.4 Capa 4 (transporte)

Esta capa acepta el mensaje transmitido por la capa de sesión, define cuando y como debe utilizarse la retransmisión para asegurar su llegada. Para ello divide el mensaje (datagramas) y gestiona cada porción para que se envíe a la capa de red, por otro lado, en la recepción hace el proceso inverso, junta los paquetes enviados por la capa de red en segmentos para la capa de sesión. Otro aspecto es que los datos no sólo deben entregarse sin errores sino que también en la secuencia correcta, para lo cual numera las porciones del mensaje correlativamente y los entrega a la capa de red para su envió.

Un sistema host puede tener múltiples aplicaciones que se comuniquen a través de la red. Cada una de estas aplicaciones se comunica con más aplicaciones en host remotos, por lo tanto es responsabilidad de la capa de transporte mantener los diversos canales de transmisión y comunicaciones entre dichas aplicaciones. Pero, si la conexión de transporte requiere un volumen de transmisión alto, la capa de transporte podría crear múltiples conexiones de red, dividiendo los datos entre las conexiones para

aumentar el volumen. La capa de transporte asigna un identificador haciendo uso de los encabezados de mensajes y de los mensajes de control, para indicar con que aplicación se asocian los datos.

Los protocolos más comunes en la capa de transporte son el protocolo de control de transmisión (TCP, por sus siglas en inglés) y el protocolo de datagramas de usuario (UDP, por sus siglas en inglés) ambos protocolos gestionan la comunicación de múltiples aplicaciones.

- TCP es un protocolo orientado a conexión, usado para la realización de conexiones entre sistemas por medio de redes de información. Hace uso de recursos para ganar funciones, dichas funciones son: el orden de entrega, entrega confiable y control de flujo, TCP funciona en conjunto con el protocolo que se conoce como el conjunto de protocolo de enlace de tres vías que se encarga de establecer una sesión entre sistemas en la comunicación y haciendo uso de Acuse de Recibo o ACK, lo cual permite que en el tiempo de la comunicación los mensajes lleguen completos y sin alguna modificación, en caso de haber un error se solicita una retransmisión. Este protocolo realiza un mecanismo donde divide los datos en segmentos para su transmisión, pero en la recepción, este mismo protocolo se encarga del reensamblaje de la información. Entre las aplicaciones que suelen utilizar este protocolo se encuentran los navegadores web, la transferencia de archivos y el envío de correo electrónico.
- UDP es un protocolo sin conexión, proporciona la entrega de datos sin verificar las conexiones entre los host origen y destino. Resulta muy útil en aplicaciones que envían pequeñas cantidades de datos. Las porciones de comunicaciones en UDP reciben el nombre de datagramas. Entre las aplicaciones que utilizan UDP se incluyen: el *Domain Name System* (DNS), Streaming de Video y Voz sobre IP(VoIP)

2.4.5 Capa 5 (sesión)

La capa de sesión permite que los usuarios de diferentes sistemas host puedan establecer sesiones entre ellos. A través de una sesión se puede llevar a cabo un transporte de datos ordinario, tal y como lo hace la capa de transporte, pero mejorando los servicios que esta proporciona y que se utilizan en algunas aplicaciones.

Una sesión podría permitir al usuario acceder a un sistema de tiempo compartido a distancia, o transferir un archivo a distancia.

Esto se logra implementando varios mecanismos de control, control a nivel de la conectividad y la conversación, es decir, determinar quién debe comunicar y en qué momento, para así poder realizar las "negociaciones" relativas a los parámetros de sesión. El control de dialogo y la separación de dialogo permite a las aplicaciones comunicarse entre el sistema host y sistema remoto.

Los protocolos más conocidos de esta capa son: SMTP, SSH, FTP, RCP, ZIP, SAP, y SCP.

2.4.6 Capa 6 (presentación)

Recibe, asegura la información proporcionada en las capas inferiores, la procesa a fin de que ésta pueda presentarse al sistema. Define de forma estándar la codificación y presentación de la información además maneja el procesamiento tal como la encriptación, compresión y codificación. Dicha codificación puede tener propiedades de eficiencia (comprensión) o de seguridad (cifrado), un buen ejemplo son la conversión de datos a código ASCII para EBCDIC, la criptográfica también se realiza dentro de esta capa, por lo tanto, podría decir que esta capa es capaz de traducir entre varios formatos de datos utilizando un formato común, estableciendo la sintaxis y la semántica de la información enviada.

2.4.7 Capa 7 (aplicación)

Es la capa donde se presenta la interfaz, los programas y servicios dirigidos al usuario y que provee servicios de red para las aplicaciones del usuario como trasferencia de archivos, correo electrónico, entre otros. En esta capa ocurre toda interacción entre el usuario y el sistema host. La responsabilidad de esta capa es identificar y establecer la disponibilidad de comunicación del host destino. Dentro de esta capa existen dos formas de procesos que proporcionan acceso a la red: aplicación y servicio.

Aplicación: son los programas de tipo software que utiliza el usuario, éstas se comunican directamente con la red.

Servicios de aplicación: proporcionan algún recurso en la red a los programas de aplicación, como la trasferencia de archivos estos servicios preparan los datos para su trasferencia, aunque estas acciones son totalmente transparentes para el usuario. Los datos pueden ser de distinto formato ya sea texto, grafico o video. Cada servicio utiliza protocolos que definen los estándares y formatos a utilizarse.

Los protocolos más conocidos de esta capa son:

- DNS: Este protocolo se encarga de traducir nombres de dominio (direcciones web que representan un host) a sus respectivas direcciones IP.
- HTTP (*Hipertext Transfer Protocol*): Es el método utilizado para la transferencia y transmisión de información en la *World Wide Web*.
- FTP (*File Transfer Protocol*): Utilizado en la transferencia de archivos entre sistemas de red conectados.
- DHCP (*Dynamic Host Configuration Protocol*): Es un protocolo utilizado para solicitar y asignar direcciones IP, dirección del equipo de salida y dirección de servidor DNS en caso de haber uno configurado.
- SMTP (*Simple Mail Transfer Protocol*): Protocolo utilizado para las transmisiones de mensajes de correo electrónico a través de Internet.

Estos protocolos de comunicación manejan el intercambio de datos entre dos entidades o sistemas host.

2.5 Direccionamiento IP

IP es un protocolo orientado a no conexión, usado para la comunicación de datos. Su función es la de enviar paquetes de datos tanto a nivel local como a través de redes. La dirección IP es un número decimal que identifica a un sistema host en particular dentro de una red o múltiples subredes, lo hace de manera lógica y jerárquica (nombres, direcciones y rutas.). Un nombre indica lo se busca, una dirección indica dónde está y una ruta indica cómo llegar hasta él. Para lo cual se fragmentan los paquetes para su envió a través de encaminamiento y re-ensamblado.

Este protocolo corresponde a la capa de red del modelo ISO/OSI, el valor de estas direcciones numéricas se utiliza para encaminar paquetes de sistema host a un sistema remoto o viceversa [11]. Utiliza cuatro mecanismos clave para proveer su servicio: tipo de servicio, tiempo de vida, opciones y cabecera de *checksum*, éstos están explicados a detalles en el RFC 791.

La estructura que se muestra en la figura 7 describe el direccionamiento IP versión 4 (IPv4), esta utiliza direcciones de red que se componen de 32 bits divididos en cuatro campos, es decir, cada número antes del punto está construido por un juego de ocho bits [37].

Otra versión de este protocolo IP es Ipv6 que consta con una longitud de 128 bits, su desarrollo se llevó a cabo pensando en el pronto agotamiento de direcciones Ipv4, la adopción de este protocolo está dándose cada vez mas, debido a que los proveedores de servicios de internet (ISP por sus siglas en inglés) han ido preparando su infraestructura para el uso de IPv6 [31].

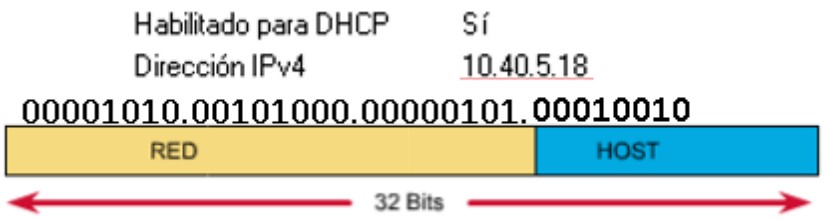


Figura 7 Esquema del formato de direccionamiento IP

Dichos números son una dirección única de red (tres primeros juegos de bits), cuando se agrega esta dirección a un dispositivo se le denomina sistema host (último juego de bits). De esta manera la capa de red provee mecanismos para direccionar estos sistemas. Estos mecanismos son:

Encapsulado: La capa de red también debe proveer el sistema de encapsulado. Los dispositivos no pueden ser identificados sólo con una dirección: las UDP provenientes de la Capa 4, deben contener en su encabezado información (HDR) acerca de los host de origen y destino, esta información se agrega como encabezado a la capa 3 para lograr crear la PDU de esta misma capa. Después el paquete se envía a la capa de transporte para la preparación de su envío a través del medio [44].

Encaminamiento: Consiste en dirigir los paquetes al host destino. Si el sistema host no está dentro de la misma red que el host destino, los paquetes tendrán que hacer el recorrido por varias redes. A lo largo de cada ruta los paquetes son guiados para que lleguen a su destino. Los dispositivos intermediarios en

la ruta que conectan las redes reciben el nombre de *routers*, el cual tiene por función seleccionar las rutas y dirigir los paquetes a su destino. Este proceso se conoce como encaminamiento y trabaja con protocolos como: RIP, RIPv2, IGRP, EIGRP, OSPF e IS-IS.

Desencapsulamiento: Este es el último proceso donde el paquete llega al host destino y donde la capa 3 lo procesa, lo interpreta y examina la dirección destino para corroborar que el paquete fue direccionado correctamente, de ser lo contrario no lo recibe. Cuando el paquete llega a su destino sin error alguno este es desencapsulado y se envía al servicio adecuado mediante la capa de transporte.

La función de la capa de red en modelo ISO/OSI es la transferencia de datos desde el host que origina los datos hacia el host que los usa. Si el host de destino está en la misma red que el host de origen el paquete se envía sin necesidad de un *router*, sin embargo, si el host de origen no está en la misma red, el paquete puede llevar una PDU a través de muchas redes, si es así la información que contiene no es alterada por ningún dispositivo de interconexión cuando se toman las decisiones de envío. En cada salto, las decisiones de envío se basan en la información del encabezado del paquete. El cual se mantiene intacto a través de todo el proceso, desde el sistema host origen hasta el destino, para que la comunicación sea exitosa, y se dé entre dos redes diferentes, el paquete se envía hacia su puerta de enlace[6]. Si este es enviado a un segundo *router* es responsabilidad del dispositivo el reenvío del paquete. En los protocolo ARP, RARP e ICMP se coloca la dirección IP del host de origen en el encabezado del paquete, así como la dirección de destino.

2.6 División de una red en subredes (subneting)

El protocolo IP tiene un formato de dos partes que consta de la dirección de red y la dirección local. La primera identifica la red a la que está conectado el host, la dirección local identifica a un host en particular dentro de la red.

Los dispositivos necesitan de direcciones IP para poder enviar información dentro de una red interna. Es necesario dar a conocer que también necesita un esquema de direccionamiento que permita a los sistemas host enviar datos a través de una red de modo eficiente. Existen razones por las cuales es necesario el diseño de redes múltiples o subredes. Estas son: el aumento de tamaño de cada red y el aumento de la cantidad de redes. Si una red LAN, MAN o WAN aumenta sus proporciones demasiado, se debe controlar el tráfico de la red; una solución es que ésta tiene que ser dividida en porciones más pequeñas, lo que se denomina como "segmentos de red". Cuando se realiza da como resultado un grupo o segmento de redes, donde cada uno tiene una dirección individual. Hay que señalar que una vez separadas estas redes deben estar en comunicación entre sí a través de internet. El segmentado de red permite una mejor administración y seguridad, por ejemplo, reduce el tráfico de *broadcast* (difusión dentro de la subred) de la red, se localizan más rápido los problemas, etc. Pero existen desventajas al momento de segmentar pues al realizar esta implementación en ocasiones se desperdician muchas direcciones IP [16].

[6] La puerta de enlace es normalmente un equipo informático que se configura para otorgar a las máquinas de una red de área local conectadas a él un acceso hacia una red exterior.

2.6.1 Clases de red

Cuando se implementa la segmentación de la red es necesario saber los tipos de clases de redes que existen (A, B, C, D y E). Estas clases permiten crear una red de interconexión de subredes, cada una con un único identificador de red. Cada clase de red tiene determinada una máscara de red por defecto, un rango de direcciones IP, cierta cantidad de redes y de host. Estas clases se describen en la tabla 3.

Tabla 3 Clases de redes

Clase	**Rango de direcciones**	**Máscara de red por defecto**	**Número de redes y host por red**
A	0.0.0.0-127.255.255.255	255.0.0.0	128 redes, 16777214 host/red
B	128.0.0.0-191.255.255.255	255.255.0.0	16384 redes, 65534 host/red
C	192.0.0.0-223.255.255.255	255.255.255.0	2097150 redes, 254 host/red
Clase	**Uso**		**Rango de direcciones**
D	Utilizadas en grupos milticast en la red		224.0.0.0-239.255.255.255
E	Para investigación y desarrollo. No son asignables para redes IPv4		240.0.0.0-254.255.255.255

La clase A se define para redes muy grandes. Es un rango de direcciones donde el primer octeto determina la dirección de red, los octetos 2,3 y 4 (los siguientes 24 bits) se pueden utilizar para la división de la red en subredes y host, como se estime conveniente para el administrador.La dirección de IP **0.0.0.0** es usada por los host cuando encienden, pero no se usa después. Regularmente todas las direcciones de la forma **127.xx.xx.xx** son reservadas para pruebas de realimentación, estas se procesan localmente y se tratan como paquetes de entrada, estas direcciones nunca viajan fuera del host.

Las direcciones de esta clase se utilizan para una capacidad de hasta 16,777,216 hosts y se pueden tener hasta 128 redes distintas de esta clase.

La clase B se define para redes de tamaño medio. Esta red es donde los dos primeros octetos son la parte de red, los octetos 3 y 4 son empleados para definir la subred y host. Tiene un rango de direcciones de red principal de **128.0.0.0 – 191.255.255.255**. Las direcciones de esta clase son empleadas para definir redes que cuentan con 256 y 65,536 host.

La clase C se define para redes pequeñas, los primeros tres octetos se consideran para la parte de red. Tiene una dirección principal de **192.0.0.0 – 223-255.255.255**. El cuarto octeto (últimos ocho bits) es empleado para redes locales y host, se utiliza esta clase para redes con menos de 254 host [22].

Se han definido que cada clase de direcciones IP contiene un rango de direcciones especiales, las cuales se denominan “privadas” o “reservadas” donde el RFC 1918 asigna tres bloques de direcciones IP para uso interno y privado(ver tabla 4), logrando responder al concepto de aprovechar mejor el espacio de direccionamiento IP. Se propuso a consecuencia del crecimiento de internet y la creciente escases de direcciones IP públicas, por lo cual se desarrollaron nuevos esquemas de direccionamiento, tales como *Classless InterDomain Routin* (CIDR), para ayudar a resolver este problema. Por tal motivo se hizo

necesario recurrir a alguna alternativa para maximizar el uso del direccionamiento IP, como el direccionamiento público y el direccionamiento privado. Las direcciones públicas son enrutables en internet y las direcciones privadas no. Lo que se traduce a que las primeras son asignadas a equipos que deben ser alcanzables desde la red de internet y permitir el acceso a los servicios desde cualquier red; sin embargo, las direcciones privadas son utilizadas sobre equipos que no requieren o es limitado su acceso a internet.

Tabla 4 Direcciones IP privadas

Clase	Rango
Clase A	10.0.0.0 – 10.255.255.255
Clase B	172.16.0.0 – 172.31.255.255
Clase C	192.168.0.0 – 192.168.255.255

No obstante, los dispositivos con direcciones privadas pueden tener acceso a internet por medio de algunos servicios, tal es el caso de NAT (Traducción de Direcciones de Red) el cual permite que las direcciones privadas sean encaminadas o direccionadas a internet por medio de una dirección homologada.

Otra alternativa es la implementación de subredes, la cual permite la división de una red en varias subredes para uso interno, pero aún actuar como una sola red ante el mundo exterior. Fuera de la red, la subred no es visible, por lo que la asignación de una subred nueva no requiere comunicación con el NIC (*Network Interface Card*) ni la modificación de base de datos externas. El NIC se encarga de traducir los datos (información) producidos por un host a un formato que se pueda transmitir por la red local.

2.6.2 Máscara de red

Con la máscara de red se logra jerarquizar aún más la estructura IP. Está constituida por parte de red más parte de hosts, incluyendo un nuevo nivel de jerarquía que se denomina número de subred (también llamado subnet) [10].

Esta máscara de red anteriormente era fija (máscara por defecto o máscara natural) para cada clase de red lo que significa que todos los host en la misma red física compartían la misma parte de red.

Para la red de clase ‘A’ la máscara por defecto es **255.0.0.0**, en bits se representa **11111111.00000000.00000000.00000000**, donde los “1” representan la porción de red, la cual será común para todos los host. La porción de host está representada por bits “0” e indica la cantidad de host que formaran parte de la red. Algo muy similar sucede con las otras dos clases mencionadas. En la clase B los dos primeros octetos pertenecen a la porción de red que va a ser común para todos los host y los dos últimos octetos pertenecen al número de host que formaran parte de la red. Y para la clase ‘C’ los tres primeros octetos pertenecen al identificador de la red que será común para todos los host y el último octeto representa al número de host que contendrá este tipo de clase de red. Para entender esto se puede analizar la figura 8 donde se

muestra en la parte enmarcada con rojo las porciones de red, las porciones sombreada de gris representa los bits que se utilizaran para los host dentro de la red [6].

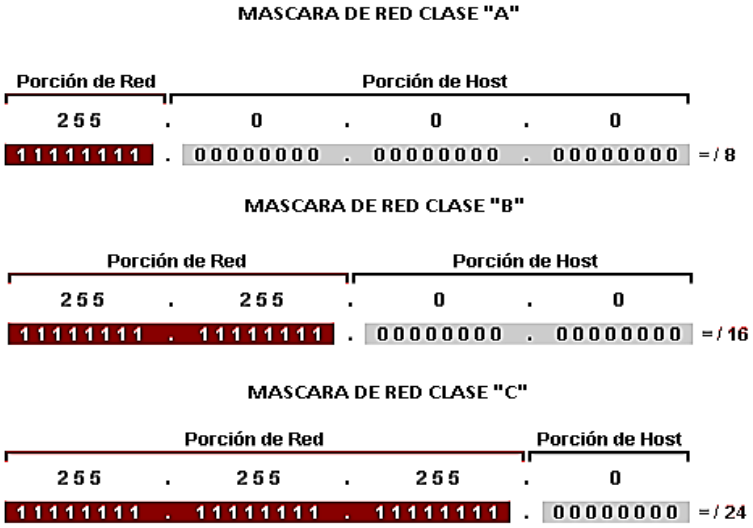


Figura 8 Porciones de red y porción de host de cada clase

2.6.3 Máscara de subred de longitud variable (VLSM)

Cuando una red se divide en subredes, todas estas cuentan con la misma máscara de subred, lo que significa que cada sistema informático contará con el mismo prefijo de mascara de subred. Por ejemplo, en el caso de las direcciones IP de clase C, la mascara de red es **255.255.255.0** y es la de mayor tamaño, la cual especifica 254 sistemas host para la red. Cuando las empresas contaban con más de 254 host en una red y para conservar las direcciones IP de los demás sistemas, los proveedores de internet internet emitían varias direcciones IP de clase C a sus clientes para que las utilizaran. Esto complicaba la configuración de las rutas y otras cuestiones.

Esto era funcional en algunas ocasiones, pero la mayoría de las veces representaba un desperdicio de direccionamiento, e ineficiencia en la asignación de direcciones IP debido al crecimiento exponencial de internet. CIDR reduce el tamaño de direccionamiento y hace que haya más direcciones IP mediante el método de máscara de subred de longitud variable (VLSM, por sus siglas en inglés) el cual permite emplear diferentes máscaras de red para cada subred [4].

Como se sabe los octetos con "**0**" en la máscara de red siempre se consideran para representar la cantidad de host dentro de una red, pero dentro de estos octetos se pueden crear subredes. Se realiza de manera que se toman los bits que pertenecen a la porción de host, junto al identificador de red, y se adapta la máscara de red por defecto a la subred.

Como describe Eduard Lara (2014), VLMS permite adoptar un esquema de direccionamiento en función de las necesidades de la red, es decir, permite emplear diferentes sub máscaras para cada subred, lo cual permite a su vez hacer uso eficiente del espacio de direcciones IP [14].

Las subredes permiten crear múltiples redes lógicas que existen dentro de las redes de las clases A, B y C, y en el caso de no usar las subredes solamente se tendrían disponibles las redes con máscara de red por defecto. Pero como se mencionó se tendría un gran desperdicio de direcciones para hosts.

En una determinada red o subred se recomienda que no sean asignadas ni la primera dirección, es una dirección reservada por ser la dirección de la subred, ni la última dirección la cual se reserva para realizar el *broadcast*, este sirve para enviar anuncios dentro de una subred. Cuando se envían los paquetes a la última dirección los reciben todos los host o dispositivos que pertenecen a la red. Para la realización de subredes se consideran las siguientes dos ecuaciones que indican cómo es posible obtener el número de subredes y el número de host por cada subred [35].

Numero de subredes = $2^{x} - 2$. Ecuación (1)
Donde "x" representa el número de bits asignados a las subredes.

N host por subredes = $2^{y} - 2$. Ecuación (2)
Donde "y" representa el número de bits asignados a la porción de host.

2.7 Redes inalámbricas

Una red inalámbrica es un sistema basado en un medio de transmisión no guiado, el cual utiliza ondas electromagnéticas para comunicarse con dispositivos, lo cual permite la flexibilidad para trabajar en la red según la cobertura de ésta, es decir, se evita estar atados a un cable dispositivos como impresora, teclado, entre otros.

Algunas universidades hoy en día ya cuentan con antenas por todo sus planteles para permitir a los estudiantes realizar consultas del catálogo de la biblioteca o revisar otros servicios que esta ofrezca. La implementación de una red inalámbrica es fundamental para este trabajo, debido a que si algún investigador del laboratorio desea hacer una consulta desde su teléfono móvil, solo necesita estar dentro de la cobertura de la red inalámbrica del laboratorio y lograra acceder a la información que desea.

2.7.1 WLAN

Wireless Local Area Network se refiere a la comunicación e interacción entre dos o más host sin medios cableados. Estas redes permiten flexibilidad y portabilidad, a diferencia de las redes LAN, una WLAN comunica y conecta distintos host (computadoras, impresoras, dispositivos móviles, tabletas) por medio de un punto de acceso inalámbrico (WAP). Una red WLAN debe estar constituida por dos elementos clave, una estación de cliente (STA) y la otra por puntos de acceso. Donde la comunicación se realiza directamente entre estas dos. El intercambio de información solo se logra cuando existe autentificación entre el STA y el AP.

El WAP transmite señales de gestión periódicas, el STA las recibe e inicia la autentificación mediante él envió de una trama de autentificación, por consecuente la estación de trabajo envía una trama asociada

y el AP responde con otra. Una vez realizado este proceso se obtiene la comunicación entre ambos, de esta manera es posible tener acceso a los servicios y aplicaciones de red.

El estándar IEEE 802.11A fue desarrollado en 1999 por la IEEE pero fue comercializado hasta mediados del 2002, este especifica una capa física con una velocidad de datos de 54 Mbps. Pero el rendimiento máximo de un usuario podría llegar a ser de 30 Mbps (5 veces más que IEEE 802.11B – 11 Mbps).

Por otro lado la velocidad de transferencia de datos en una red LAN puede alcanzar hasta 10 Mbps en una red Ethernet y 1 Gbps en FDDI o Gigabit Ethernet.

2.8 Seguridad en la red

Los servicios de seguridad ayudan a mejorar la protección de los sistemas informáticos, evitando el acceso a ataques mediante la utilización de uno o varios mecanismos de seguridad con el objetivo de resguardar la información. A medida que estos sistemas incrementan su utilidad para los usuarios, la seguridad de los mismos debe hacerlo; así bien debido a la interconectividad que se genera entre estos y la generación de redes más amplias se tiene el siguiente concepto de seguridad de la red:

Conjunto de herramientas y normas de seguridad para la protección de equipo activo dentro de una red.

Para administrar y controlar lo que sucede en la red se deben implementar medidas para evitar amenazas manteniendo la seguridad de los sistemas y aplicaciones. Un mecanismo comúnmente utilizado es la seguridad perimetral donde se define el perímetro como la frontera de la red de área local. Sin embargo hoy en día existen dispositivos que rompen con el concepto tradicional de seguridad perimetral por lo cual las redes se han convertido en redes dinámicas. Se considera entonces que el perímetro comienza donde la trasferencia de datos se realiza. Para fines de este trabajo se contempla el perímetro como se explica en la definición inicial sumando que se deben implementar controles técnicos como etiquetado de equipo, control de acceso por dirección MAC, entre otros que evalúen permanentemente los servicios que la red ofrece, tanto propios como externalizados [19].

2.9 Cortafuego (firewall)

Se definen como un sistema o grupo de sistemas que refuerzan la política de control de acceso entre dos redes (la red confiable o red local y la red no confiable o red externa). Las acciones principales de un cortafuego es proteger a la red confiable de la red no confiable, y dar acceso solo a las conexiones autorizadas de esta última. [35, 19, 36].

Este sistema debe proteger todos los elementos de la red interna, incluyendo hardware, software e información, no sólo de cualquier intento de acceso no autorizado desde el exterior, sino también de ciertos ataques desde el interior que puedan dañar o vulnerar la red. Es un dispositivo que ofrece seguridad en la red ya que permite la detección accesos no autorizados, bloquea el acceso a puertos de comunicación que se encuentren abiertos en algún sistema host de la red local, esto con la finalidad de que el atacante no se aproveche de vulnerabilidades de la red.

Existen distintos tipos de cortafuegos entre los más comunes están el cortafuego de hardware y el cortafuego de software.

El cortafuego de hardware se utiliza regularmente en empresas y grandes corporaciones. Normalmente son dispositivos que se colocan entre el dispositivo *router* y la conexión a internet. Al ser dispositivos dedicados de seguridad, se encuentran optimizados para realizar la función de cortafuego, y además no consumen los recursos de los sistemas personales. Su mayor inconveniente es el mantenimiento, ya que son difíciles de actualizar y de configurar correctamente. La instalación de éste es compleja y es hecha gracias a un navegador que tiene acceso a internet.

El cortafuego de software se caracteriza por su fácil instalación en una computadora o un servidor de grupo de trabajo, sin embargo, presenta inconvenientes que son inherentes a su condición; por ejemplo, consume recursos de la máquina, en ocasiones no se ejecuta correctamente y pueden ocasionar errores de compatibilidad con otros tipos de software que se encuentren instalados en el equipo.

Hoy en día sistemas operativos como Windows y Linux integran soluciones básicas de firewall. En algunos casos, como en el software libre, son muy potentes y flexibles, pero requieren un gran conocimiento en redes y puertos necesarios para las funciones de algunas aplicaciones. Para poder simplificar su configuración, suelen ser habituales los interfaces web que simplifican su manejo al usuario, aunque también se pierde gran parte de su funcionalidad.

Si es un cortafuego gratuito puede ser utilizado con total libertad. Su objetivo es rastrear y no permitir acceso a ciertos datos a las computadoras de los usuarios. En caso de ser un cortafuego de coste, este posee el mismo funcionamiento de un gratuito, solo que en ocasiones se le suma mayor nivel de control y protección. Regularmente son vendidos con otros sistemas de seguridad como antivirus en el caso particular de Windows o se encuentran disponibles en servidores FTP como FreeBSD, que está hecho para ser compatible con la norma POSIX.

Un cortafuego de hardware o software, puede ser un sistema que funcione sobre alguna de las capas del modelo ISO/OSI, capa de enlace, capa de red, capa de sesión, capa de transporte y capa de aplicación.

- Capa 2, de enlace de datos. Filtrado por dirección física (MAC).
- Capa 3, de red. Filtrado por protocolo de red (por direcciones IP).
- Capa 4 y 5, de transporte y sesión. Filtrado por protocolo de transporte y por puertos.
- Capa 7, de aplicación. Caso de los servidores proxy.

Es necesario tener claro cuáles objetivos determinaran la seguridad de una red interna, para poder elegir la herramienta y definir las reglas de filtrado. Estas se utilizan regularmente para salvaguardar la información y la red.

2.9.1 Tipos de cortafuegos por capa OSI

Existe distintos tipos de cortafuegos, algunos actúan desde capa de red a capa de aplicación del modelo OSI y se puede implementar conforme convenga a la organización o empresa, según la forma en que se desee implementar sus política de seguridad de red. A continuación se describen los tipos básicos de cortafuegos.

Los cortafuegos de capa 3 pueden ser considerados como filtros de paquetes ya que lo que realizan es un filtrado de los intentos de conexión atendiendo a las direcciones IP de origen, destino y puertos TCP, UDP.

Además es capaz de identificar paquetes por propiedades relativas a capas superiores ya que portan información en sus cabeceras, como por ejemplo el sistema operativo que generó el paquete, tipo de aplicación (voz IP, *streaming* de audio, video, etc). El cortafuego encaminará todos los paquetes que ingresen a su interior, las reglas sólo definen claramente qué paquetes entran o no a éste. Algunos de estos cortafuegos se encuentran en *routers* comerciales.

Otra implementación de cortafuegos son aquellos que funcionan a nivel de transporte del modelo OSI, estos examinan si el/los paquete(s) pertenece(n) o no a una sesión que se encuentre abierta en ese momento, si el paquete no pertenece a ninguna conexión establecida, niega el acceso a la red. Estos cortafuegos reciben el nombre de cortafuegos dinámicos o *Stateful Packet Inspection* (SPI).

Los cortafuegos de capa 7 también reciben el nombre de *Deep Packet Inspection* (DPI) ofrecen mayores opciones para brindar seguridad, pero como consecuencia exigen mayor trabajo al dispositivo donde se implementa. Este tipo de cortafuegos es capaz de realizar filtrados a nivel aplicación, es capaz de inspeccionar el tráfico a cualquier capa del modelo OSI, incluyendo la capa de red, nombre de dominio, URL, palabras clave; este último denominado filtrado de contenido. Los cortafuegos de aplicación pueden identificar secuencias de comandos que tengan un comportamiento amenazador, como longitud de nombre, peticiones a una base de datos, detección de troyano, spyware, malware, entre otros.

Podemos decir que hay dos maneras básicas para diseñar y construir las listas de acciones que revisara el cortafuego para que pueda controlar el tráfico de la red. Una es negar todo de manera predeterminada, permitir sólo los paquetes seleccionados. O aceptarr todo de manera predeterminada, denegar sólo los paquetes seleccionados

En una red “seria” se debe implementar la primera opción, por defecto bloqueamos todo el tráfico, tanto el que sale como el que accede, y solo se permitirá el tráfico que le interese al administrador de la red.

2.9.2 Arquitecturas de cortafuegos

Existen diversas formas de utilizar un cortafuego en una red de datos y en cada arquitectura se toman tres decisiones básicas para el diseño y la configuración del cortafuego.

La primera de ellas hace referencia a la política de seguridad de la organización donde se encuentre establecida la red. Estas políticas establecen que tráfico externo hacia el domino de su propiedad se bloquea, que dominós se bloquean para que usuarios no pierdan el tiempo, por ejemplo, en algunas oficinas del gobierno no se permite el acceso a páginas de redes sociales (Facebook, YouTube, Twitter).

La segunda decisión es el nivel de monitorización, redundancia y control deseado en la organización, es decir, básicamente qué se va a permitir y qué se va a denegar. Por último se debe tomar y valorar cuestiones económicas pues en función del valor estimado sobre lo que se desea proteger, se debe gastar o invertir [36].

Un cortafuego puede generar gastos extras pero existen aproximaciones funcionales para proteger una red interna, como utilizar una computadora o PC como un cortafuego de software de distribución libre como FreeBSD, en caso de desear proteger una red más grande se pueden utilizar sistemas propietarios, como, Intel, Cisco, Extreme Network , que suelen ser algo costosos y a veces requieren más tiempo de configuración que los cortafuegos sobre Unix, los cuales también cuentan con cortafuegos de software y hardware con mayores características a un costo considerable.

A continuación se describen algunas de las arquitecturas más utilizadas en el diseño de redes con cortafuego.

- *Dual-Homed Host* (*Multi-Home-Host*): Esta arquitectura sitúa al cortafuego justo en medio de la red interna y la red externa no segura (internet), el servidor de seguridad contempla por lo menos dos interfaces de red (NIC), como se muestra en la figura 9 . Donde una da acceso a la primera red y otra da salida a la red de internet. El fin de esta configuración es no permitir que el tráfico procedente de la red no segura se encamine a la red segura, es decir se obtiene un doble filtrado o un filtrado en cada tarjeta. Esta configuración también se conoce como *proxy*, donde las peticiones de un usuario de la red interna se solicitan al éste y son atendidas por el mismo.

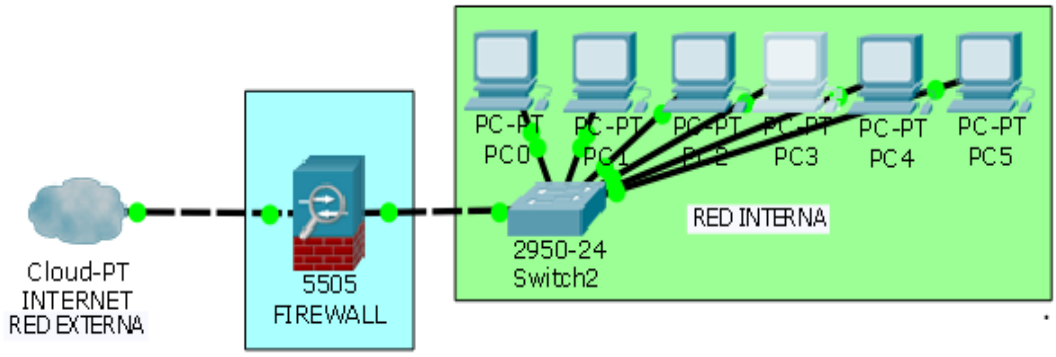


Figura 9 Arquitectura de cortafuego Dual-Homed Host

- *Screened host*: En esta arquitectura se utiliza un host denominado bastión como el que se muestra en la figura 10, con el objetivo de que los sistemas externos se conecten a éste en vez de permitir la conexión directa a otros sistemas internos y con menos seguridad. Para lograr esto, un *router* de filtrado de paquetes se configura para que todas las conexiones generadas desde la red externa se dirijan hacia el host bastión mientras tanto en el resto de la red interna se filtran los paquetes y

sólo algunos servicios serán permitidos, pero todos los dispositivos en ésta se mantendrán inaccesibles desde el exterior.

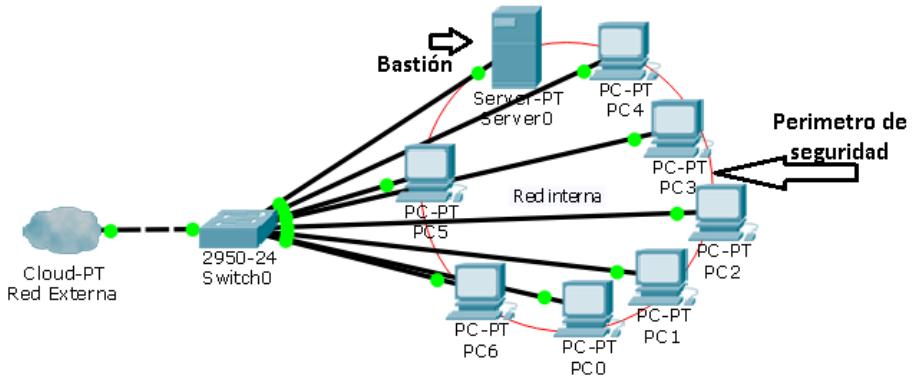


Figura 10 Arquitectura de cortafuego Screened Host

- *Screened Subnet* y DMZ**:** Es una variación de los cortafuegos de capa 3 o filtrado de paquetes y del modelo de arquitectura *dual-homed*. En esta arquitectura es necesario determinar más de un perímetro de red como se muestra en la figura 11, por lo que surge el concepto de DMZ o zona desmilitarizada. Se trata de una red local que se ubica entre la red externa y la red interna, es decir, para cualquiera que trate de ingresar hacia la red interna, la zona desmilitarizada se convierte en una trampa, la arquitectura DMZ intenta aislar el sistema host bastión en una red perimétrica de forma que si algún intruso accede a esta máquina no consigue acceder totalmente a la subred protegida.

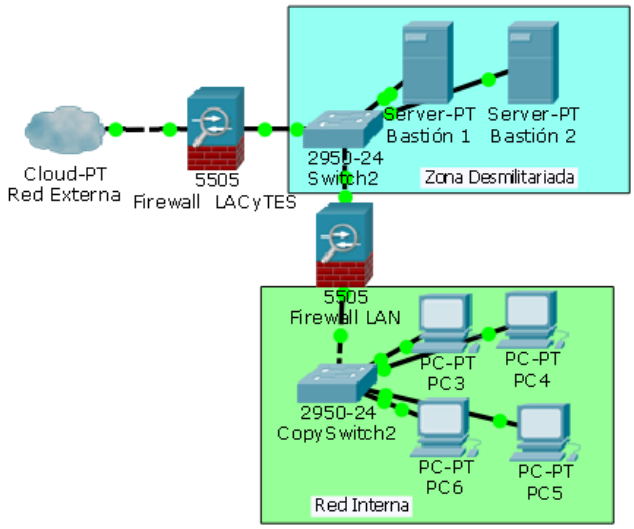


Figura 11 Arquitectura de cortafuego Front-End

Por lo regular dentro de la DMZ se encuentran ubicados los servidores host (web, DNS, correo electrónico o proxy) hacia la red externa como se observa en la figura 11 que describe una arquitectura *Front-End*. Esta cuenta con dos cortafuegos configurados independientemente de distinta forma, por ejemplo, el primer cortafuego se configura para que los servicios fiables pasen directamente sin acceder al host bastión y el segundo cortafuego se configura para recibir los servicios, pero realiza un filtrado de que paquetes acceden a la red y a cuales se les niega el acceso.

Existe una variante conocida como *Three-Legs*, cuenta con tres niveles de defensa lo que la hace una arquitectura o una configuración más segura (eliminando aplicaciones, protocolos y puertos innecesarios) y la creación de una red perimetral (DMZ), la cual puede consistir de un simple host bastión o de más servicios. Este tipo de configuración regularmente consta de un cortafuego que contiene 3 tarjetas de redes, es decir, cada red física se conecta a un puerto distinto de éste, como se muestra en la figura 12. Las reglas de filtrado sobre estos elementos pueden ser complicadas de establecer y comprobar, cada red perimetral debe seguir diferentes reglas de filtrado lo que mejora la seguridad de los sistemas [9].

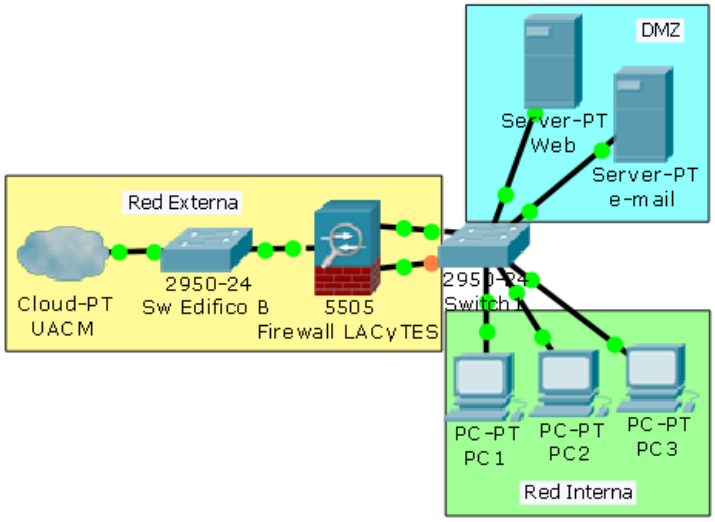


Figura 12 Arquitectura de cortafuego Screened Subnet con variante Three-Legs

Según la configuración que se elija, ya sea *Three-Legs*, que consta de la implementación de un solo cortafuego o Front-End que consiste en el uso de dos cortafuegos, serán los encargados de proteger tanto a los equipos bastión de la DMZ como a la red interna de las posibles intrusiones.

Es fundamental conocer estas arquitecturas, dado que con base en estas se determina cual conviene establecer para el presente trabajo. Es necesario tomar en cuenta las reglas que se desean establecer en la configuración del cortafuego (aislamiento, seguridad de contenidos, autenticación u ocultamiento del rango de direccionamiento interno), conocer los recursos del laboratorio para poder implementar una arquitectura funcional y segura.

2.10 Android

Android es un software escrito en los lenguajes C/C++ y JAVA, fue adquirida por Google en 2005 y se presentó como tal en 2007 a través del consorcio de grandes firmas de la informática y telecomunicaciones, el Open Handset Alliance. Android está basado en GNU (*General Public License*)/Linux un sistema operativo muy popular, de hecho ambos comparten la filosofía de ser software libre y gratuito, esto es una gran ventaja para el uso de la plataforma y el desarrollo de aplicaciones sobre Android. Cuenta con una arquitectura de cuatro capas, donde cada una contiene la característica de estar basadas en software libre, estas son: Kernel de Linux, bibliotecas, Frameworks y aplicaciones.

2.10.1 Arquitectura de Android

Continuando con el sistema operativo de Android a continuación se describe cada una de sus capas mencionadas y cómo cada una de estas utiliza servicios ofrecidos por las capas inferiores y a su vez ofrece sus servicios a capas de niveles superiores. En la figura 13 se muestra cómo se encuentra distribuida la plataforma Android.

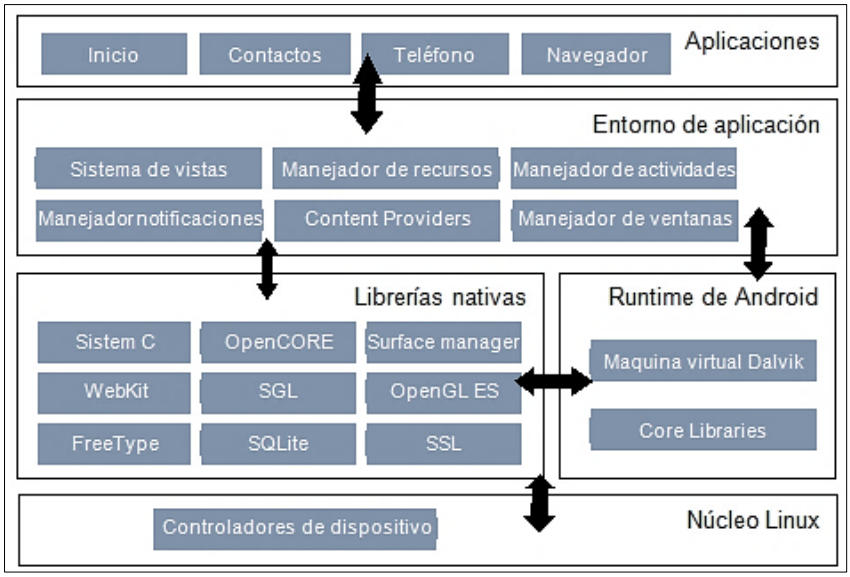


Figura 13 Capas del sistema operativo de Android

El núcleo de Android está formado por el sistema Linux 2.6. Esta capa proporciona servicios como la seguridad en el manejo de memoria, comunicación entre procesos, pila de protocolos, control de *drivers* para dispositivos y la administración de energía. Provee una capa de abstracción entre el hardware y el resto de la pila de protocolos

Las bibliotecas se ejecutan sobre el kernel, Android incluye un conjunto de estas bibliotecas en C/C++ como libC y SSL. Estas características se exponen a los desarrolladores a través del entorno de aplicación de Android algunas de estas son: biblioteca para reproducción de archivos de audio y video, administrador de superficie para la gestión de la visualización, bibliotecas graficas que incluyen SGL y OpenGL para gráficos 2D y 3D. Soporte para bases de datos nativas con SQLite. SSL y Webkit para el navegador web integrado y la seguridad en internet.

En el *Runtime* o tiempo de ejecución de Android las aplicaciones son codificadas principalmente en Java y compiladas en un formato específico para que la máquina virtual *Dalvik* las ejecute en *byte code*. Cada aplicación se ejecuta una única vez, de esta forma estarán listas para distribuirse y poder ejecutarse en cualquier dispositivo con Android.

El entorno de aplicaciones representa el conjunto de herramientas de desarrollo de cualquier aplicación. Los desarrolladores tienen acceso total a la API del entorno base o del núcleo, es decir reutilizan las características desarrolladas por Google o por otros usuarios.

Por último la capa de aplicaciones está formada por el conjunto de aplicaciones del dispositivo, permite acceder a la información y funcionalidad básica del móvil. Por lo general estas aplicaciones son escritas en Java y C/C++ pero hoy en día Android contiene varios frameworks que permiten desarrollar aplicaciones hibridas, por ejemplo, en los resultados de una encuesta realizada por *developereconomics* (2013) se señala que Phonegap es el framework más usado por desarrolladores.

> "PhoneGap encabeza el ranking de CPT (*Cross-Platform Tool's*), que se utiliza en un 34% de los desarrolladores, seguido por Appcelerator con un 21%, Adobe AIR con 19% y el resto del porcentaje lo contienen Sencha Touch, Qt, Unity y Mono…".

Para este trabajo eligió el framework de Phonegap que permite desarrollar aplicaciones móviles con características nativas a partir de lenguajes de programación y diseño Web. Se trata de una opción para desarrolladores web que no tiene conocimientos en Java, C/C++ u otras tecnologías requeridas por el sistema operativo móvil de Android. Para algunos desarrolladores de aplicaciones móviles puede o no ser un mejor camino utilizar esta opción el cual ofrece ventajas como:

- Facilidad. Si se conoce desarrollo y diseño web, desarrollar una aplicación con Phonegap resultara mucho más fácil que hacerlo a través de su lenguaje Java o SDK asociados a la plataforma.

- Velocidad de desarrollo. Phonegap, codifica la aplicación una sola vez y permite adaptarse a múltiples plataformas, en vez de programar una aplicación diferente para cada una.

- Aplicación nativa. La aplicación que se desarrolle tendrá todas las ventajas de las aplicaciones nativas, es decir, puede utilizar libremente los elementos hardware del dispositivo, como cámara, acelerómetro, NFC, elementos de conectividad, entre otros.

- Libre y gratuito. Phonegap es un *open source* y se puede utilizar sin costo para el desarrollo de aplicaciones.

- Extensible. Existen muchos *plugins* para extender las funcionalidades de este framework y acceder a características adicionales, como agenda, notificaciones, geolocalización, etc.

Por otro lado, PhoneGap permite crear aplicaciones para una amplia cantidad de sistemas operativos: desde Android, iOS y Windows Phone hasta otros menos populares, como Bada y Tizen. Sin embargo también presenta algunas desventajas para los desarrolladores que utilizan este framework, tales como las malas prácticas de los desarrolladores, ya que algunos se acostumbran a crear aplicaciones con Phonegap, sin preocuparse por aprender a utilizar las características de cada plataforma.

Otra desventaja es el mal rendimiento del framework, muchos desarrolladores que utilizan Phonegap descubren que las aplicaciones tardan tiempo en cargar. Esto se debe a la estructura del framework, sin embargo hoy en día las actualizaciones han resuelto considerablemente este problema. Para lo cual Phonegap puede ser una buena opción para implementar efectos visuales complejos y desarrollar videojuegos.

Con base en las preferencias de la encuesta mencionada, los conocimientos de tecnologías web y el cursos online de desarrollo de aplicaciones móviles tomados por este autor, se decide utilizar Phonegap sobre la plataforma Android para la elaboración de la segunda parte este trabajo, además de ser compatible con jQuery Mobile, que es seleccionado como el framework para la construcción de interfaces gráficas [8].

2.11 Aplicaciones móviles

Antes de entender y definir el concepto de aplicación móvil, es necesario conocer la definición de algunos aspectos que llegan a constituir una aplicación.

El software se define como el conjunto de instrucciones lógicas o rutinas que soportan el hardware de los sistemas informáticos, sirve como interfaz entre los usuarios y las maquinas; el objetivo es realizar una o varias tareas de forma única o múltiple. En otras palabras es el que indica a los componentes del hardware la forma en que se almacenaran y procesaran las tareas que el usuario desea.

Tomando en cuenta esta definición de software y de acuerdo al estándar ISO/IEC/IEEE 24765:2010 establecida por la IEEE se concluye que el software cuenta con un propio ecosistema de análisis, diseño, pruebas, implementación los cuales están en funcionamiento constante con los diferentes dispositivos informáticos.

Con esto se define que las aplicaciones móviles son los conjuntos de instrucciones lógicas, procedimientos, reglas, documentación, datos e información asociada a éstas que funcionan específicamente en dispositivos móviles, por ejemplo teléfonos inteligentes (SmartPhone), televisores inteligentes (Smart-tv) o dispositivos móviles cuya característica principal es el uso de una pantalla táctil como dispositivo de entrada primordial y cuentan con un hardware ligeramente inferior a una laptop [43].

Las aplicaciones móviles se desarrollaron con diferentes lenguajes de programación y funcionan específicamente en sistemas operativos móviles, los lenguajes más usados para desarrollar aplicaciones móviles son: Java, Objetic C, Xcode C#, C++, HTML5, entre otros. Existen tres tipos de desarrollo de aplicaciones móviles que se describen a continuación:

- Aplicaciones nativas: Están desarrolladas con la finalidad de ejecutarse en un sistema operativo móvil específico. Estas aplicaciones funcionan de una forma más eficiente debido a que su código es optimizado (el código fuente es compilado para crear un programa ejecutable), específicamente para la plataforma donde serán instaladas. Para desarrollar una aplicación nativa debe considerarse que cada plataforma tiene su propio lenguaje a utilizar. Por ejemplo, Android utiliza JAVA, iOS hace uso de Objetive C. Una aplicación nativa tiene grandes ventajas pues permite el acceso a las características nativas de cada dispositivo, cámara, sensor, giroscopio, entre otras. Se integra a la interfaz del sistema operativo de una mejor manera. Como desventaja su desarrollo tienden a ser difícil y costoso.

- Aplicaciones web móviles: están son desarrolladas para correr en un navegador web del teléfono. Son realizadas en base al estándar de diseño web pero están preparadas para verse de manera óptima en dispositivos móviles. El bajo costo es una de sus ventajas, es así como su portabilidad y no deben someterse al proceso de aprobación de las tiendas de aplicaciones oficiales, carecen de acceso a características nativas ya que sólo funcionan sobre un navegador web, por lo cual depende en gran medida de la conexión a internet, no tiene acceso a funciones como por ejemplo: cámara, acelerómetro, agenda y la velocidad depende de la conexión a internet.
- Aplicaciones hibridas: Estas son una mezcla de desarrollo de aplicaciones nativas y web móviles es decir; contiene lo mejor de los dos mundos anteriores de desarrollo de aplicaciones. Este tipo de aplicación permite el uso de tecnologías web y multiplataforma como HTML5, JavaScript, CSS3, entre otros lenguajes. Permiten acceder a gran parte de las características nativas de los dispositivos, lo cual brinda mejor relación entre costo de desarrollo y experiencia de usuario. El proceso de desarrollo se vuelve un poco complicado si no se tiene conocimiento de JavaScript y al igual que las aplicaciones nativas, se compila a un ejecutable, para después ser instalado.

En la figura 14 se muestra como interactúa cada tipo de aplicación con la arquitectura del sistema operativo Android.

Figura 14 Taxonomía de desarrollo de aplicaciones

2.11.1 Phonegap/Cordova

Hoy en día el desarrollo de aplicaciones móviles basadas en tecnologías y servicios web ha facilitado la creación de aplicaciones multiplataforma. El uso de tecnologías ampliamente conocidas en el desarrollo web, más la capacidad de los nuevos sistemas operativos móviles de ejecutar aplicaciones desarrolladas con estas tecnologías, ha generado que muchos desarrolladores puedan trasladar sus conocimientos e ideas a un mundo de aplicaciones móviles.

Phonegap es una implementación de código abierto y se utiliza para desarrollo de aplicaciones móviles hibridas, basado sobre todo en estándares de tecnologías web. Esto lo logra creando una instancia del

navegador del sistema llamado Webkit[7]. Además ofrece una única API (*Application Programming Interface*) basada en JavaScript para acceder a diversas bibliotecas nativas del dispositivo, de esta manera Phonegap actúa como un puente entre las aplicaciones hibridas y dispositivos móviles.

Como se muestra en la figura 15 la API del framework maneja la comunicación e interactúa con las distintas funcionalidades del hardware del sistema operativo.

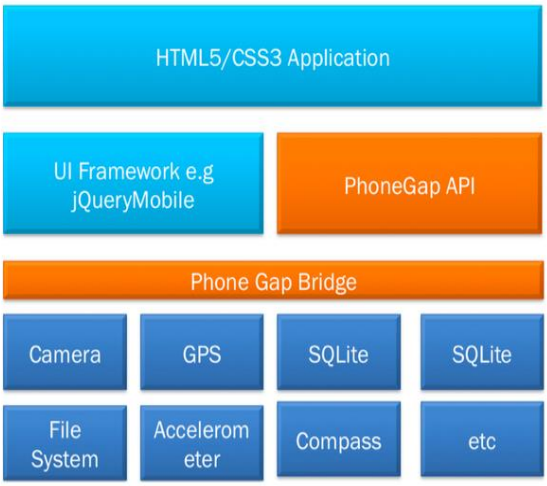


Figura 15 Comunicación de la API de Phonegap con las bibliotecas nativas de Android

Como función extra, el mecanismo de comunicación permite el desarrollo de plugins nativos. Estos permiten escribir clases propias y sus correspondientes interfaces JavaScript para ser usadas con las aplicaciones a desarrollar. Phonegap permite desarrollar aplicaciones para los siguientes sistemas operativos:

- Android.
- iOS6 e iOS7.
- Windows Phone 7 y 8.
- BlackBerry OS y BlackBerry 10.
- Web OS.
- Symbiam.
- Tizen
- Bada,
- Ubuntu Phone,
- Firefox Os

La base de una aplicación desarrollada con Phonegao es la vista en el sistema Webkit el cual a diferencia del navegador normal, permite utilizar la pantalla completa, sin mostrar controles de navegación.Dicha vista es la misma utilizada por el sistema operativo nativo. En iOS, esta es la biblioteca se llama clase *UIWebView* de Objetive C, en Android es *android.webkit.WebView*. De todas formas existen ciertas

[7] Webkit: es un proyecto de código abierto que sirve como motor a muchos de los principales navegadores web.

diferencias para la visualización de aplicaciones hibridas en las diferentes plataformas, lo que se debe tener en cuenta en el desarrollo de la interfaz de usuario [1].

PhoneGap soporta la manipulación del DOM (*Document Object Model*) así que las interfaces de la aplicación se pueden hacer con frameworks compatibles como: jQuery Mobile y Sencha Touch.

Provee varias bibliotecas JavaScript desarrolladas en el lenguaje específico de cada plataforma y por ello permite acceder a algunas características nativas de cada plataforma como *GPS*, *file system*, base de datos, contactos, acelerómetro, etc. El desarrollo de aplicaciones es muy parecido al desarrollo de una página web debido que se requiere de HTML5, JavaScript y CSS3. Tiene la ventaja de que la aplicación entera se puede desarrollar usando la mayoría de los navegadores para ver la visualización de la aplicación sin usar el emulador de cada plataforma [38].

2.11.2 Tecnologías subyacentes de Phonegap

Las tecnologías y herramientas que se relacionan para desarrollo de la aplicación hibridan de este trabajo, se enlistan a continuación:

- HTML5: Es un lenguaje de hipertexto que permite escribir de forma estructurada, en esta versión 5 se agrupan diversas especificaciones relacionadas con las tecnologías web, se agregan nuevas etiquetas para contenido multimedia (audio, video, gif, etc.), se añaden APIs, funciones como Drag & Drop (arrastrar y soltar) se incluyen en esta versión mediante uso de una API y permite el uso de una base de datos local, utilizada en la arquitectura cliente/servidor en los sistemas de información [30].
- CSS3: Las hojas de estilo en cascada permiten definir las reglas para controlar el aspecto y la presentación en diferentes dispositivos, pantallas de equipos de escritorio, portátil, móvil, impresoras o cualquier dispositivo capaz de mostrar contenido web. Es decir CSS3 es un lenguaje para definir el estilo o la apariencia de las páginas web escritas con HTML [30].

  Antes de la adopción de CSS3, los desarrolladores definían el estilo dentro de la etiqueta HTML pero actualmente se define una etiqueta en la cual se pueden insertar todos los aspectos que se desean. En su versión 3 se incluyen nuevas opciones para dar estilos, bordes de diferentes tamaños, se amplió la gama de colores, permite opacidad de los elementos, utilizar fuentes externas y se incluye el uso de Medio Queries que es utilizada para el desarrollo de diseños adaptables a los tamaños de los distintos dispositivos.

- JavaScript: Es un lenguaje de programación que nos permite construir sitios web y hacerlos más interactivos. Este lenguaje puede interactuar con el código HTML, de esta manera permite a los programadores web utilizar contenido dinámico.
- Ajax (Asynchronous JavaScript And XML): Es un estilo de programación que permite al desarrollador hacer una página web interactiva con el servidor que la aloja. Ajax usa HTML y CSS como lenguaje de estructura y diseño, JavaScript como lenguaje de programación, Json (JavaScript Object Notation) como formato de transporte de datos desde y hacia el servidor y un

lenguaje de servidor como PHP para la lógica del servidor y el acceso a base de datos. Ajax es una tecnología asíncrona que permite solicitar al servidor los datos y estos se envían en un segundo plano, logrando de esta manera no interferir con la visualización ni el comportamiento de la página

- jQuery Mobile: Es un Framework desarrollado por jQuery, se combinan dos tecnologías HTML5 y jQuery de esta manera permite trabajar en plataformas de teléfonos inteligentes y de escritorio. Este framework nos proporciona herramientas que nos facilitan crear el diseño, debido a que cuenta con bibliotecas de CSS que permiten dar funcionabilidad y diseño a las aplicaciones. Está preparado para funciones móviles como deslizar, multitoque, transiciones animadas y todo gracias a Ajax [29].

- PHP (Hypertext Preprocessor): Es un lenguaje de código libre, su programación se realiza del lado del servidor host, y su servicio se brinda en el portal web, es decir, la ejecución la realiza el servidor y no el cliente. Permite embeber fragmentos de código dentro de una página HTML y realizar determinadas acciones de una forma fácil y eficaz, ofrece varias funciones para la explotación de bases de datos de una manera sencilla.

  Este lenguaje es popular y comúnmente utilizado para acceder a bases de datos, procesar datos de formularios, enviar correos desde interfaz web, crear sitios web y aplicaciones dinámicas. Existen muchas bibliotecas y frameworks compatibles con este lenguaje de programación, de este modo se facilita la programación y disminuye el código a programar. Hay mucha compatibilidad con las distintas bases de datos más comunes tales como MySQLi, mSQLi, Oracle, Informix, ODBC [34].

2.11.3 Herramientas de entorno

A continuación se describen algunas de las herramientas utilizadas durante el desarrollo de la aplicación, estas herramientas ayudaron a la construcción, compilación y estructuración de la misma.

- Node.js: Es un entorno de programación que se ejecuta en la parte del servidor basado en el lenguaje de programación JavaScript, al estar desarrollado en este lenguaje, no requiere el aprendizaje de un nuevo lenguaje para desarrollar servicios. Node.js permite instalar y ejecutar Phonegap a través de una consola de comando permitiendo preparar y compilar proyectos, es decir, proporciona una manera fácil para desarrollar programas que contengan las propiedades esenciales de escalabilidad. [3]
- ThemeRoller: Es una herramienta que permite a los desarrolladores diseñar y elaborar temas jQuery personalizados para la aplicación móvil. Permite seleccionar el estilo, los colores y generar componentes como tabs, calendarios, cuadros de dialogo para la interfaz, sin la necesidad de realizar código CSS3. El manejo de Themeroller es bastante sencillo como personalizar colores, fuentes y texturas, y descargar la plantilla para empezar a usarla.
- Highcharts: Es un plugin desarrollado con estilos CSS3 y control JavaScript, tiene una configuración sensible que permite crear gráficos de manera que éstos se puedan adaptar a distintos tamaños de pantalla. Ofrece una gran variedad de estilos mediante atributos CSS3 y SVG. Este plugin es compatible con la mayoría de los navegadores web incluyendo los navegadores nativos de los distintos sistemas operativos móviles.

- Ripple: Existen maneras de hacer uso de las aplicaciones comunes en los *smartphones* en la comodidad de la PC. La forma más sencilla es instalar programas emuladores de Android, que introducen en el ordenador entornos similares que suelen verse en el móvil. Pero estos emuladores requieren de mucho procesamiento de la computadora donde se instala y afecta el rendimiento de la misma. Ripple es una extensión de uso "explosivo" en el navegador web de Google Chorme, ésta crea un entorno idéntico a un dispositivo móvil. Para poder emular sensores, esta extensión dispone de un entorno móvil total, es decir, se puede hacer uso de acelerómetro, cámara, posición global, entre otros más [39].

2.12 Arquitectura cliente-servidor

Esta arquitectura es la más usada en desarrollo de software gracias al trabajo distribuido que realiza. La idea básica de ésta es que el servidor, gestiona un recurso compartido y realiza determinadas funciones solo cuando las pide otro, el cliente, el cual interactúa con el usuario. Regularmente estos dos programas, el cliente y el servidor, se encuentran en distintos sistemas o computadoras. Las principales características de la arquitectura "cliente servidor" son: el servidor siempre presenta a sus clientes una interfaz única y definida (comúnmente basadas en HTML, PHP o Java Web), el cliente no tiene la necesidad de conocer la lógica del servidor sólo su interfaz interna, el cliente no depende de la ubicación física del servidor ni su sistema operativo, los cambios en el servidor implican pocos o ningún cambio en el cliente.

2.13 Patrón modelo – vista - controlador

Bazini (2014) menciona que el patrón MVC es probablemente el patrón más citado en el diseño y elaboración de portales web. Este propone la utilización de tres componentes: Modelo, Vista, Controlador, los cuales permiten controlar la compatibilidad con el mayor número de dispositivos posibles (multiplataforma) [15].

El modelo se encarga de acceder al almacenamiento de datos, gestiona los accesos a la información. Así mismo, se encarga de enviar a la vista la información solicitada por el usuario. Las peticiones realizadas por el usuario llegan al modelo por medio del controlador. La vista es la presentación grafica del modelo, nos permite visualizar la información y lógica del portal web. Además es la encargada de enviar las peticiones realizadas por el usuario al controlador para interactuar con el modelo. Y el controlador se encarga de realizar las acciones sobre el modelo o la vista, en respuesta a eventos desencadenados en la aplicación, es decir, responde a los eventos y las acciones realizada por usuarios, e invoca las peticiones al modelo y la vista (para cada acción del sistema debe haber un controlador).

Capítulo 3

Diseño de red

En este capítulo se presenta la primera parte de este trabajo, donde se realizó un breve análisis acerca de las condiciones en las que se encontró el laboratorio, con respecto a infraestructura de red de datos. Así mismo, se describe la metodología para mejorar su seguridad a través del diseño de una red interna, protegida por un cortafuego y el método con el cual se realizó el diseño de la red.

3.1 Análisis de red

Previo al desarrollo de este trabajo la red fue modificada y controlada por medio de filtrado de MAC, la solución de brindar conexión a los usuarios por medio de un WAP no resultó ser la mejor opción para resolver los problemas de conectividad y mucho menos para los servicios web que el laboratorio pretende ofrecer a la comunidad. Además hubo un incremento considerable de sistemas informáticos.

El estudio abarca la red de datos y la seguridad de dicha red en producción para las distintas áreas del laboratorio. Como primer punto se realizó un análisis del perímetro, las áreas de manufacturación, investigación, cubículo y centro de datos que componen el laboratorio (ver figura 16). Fue necesario identificar los dispositivos que se integran a la red, los dispositivos de seguridad la misma, su ubicación, el espacio que ocupa cada uno, así como los servicios que ofrecen cada uno.

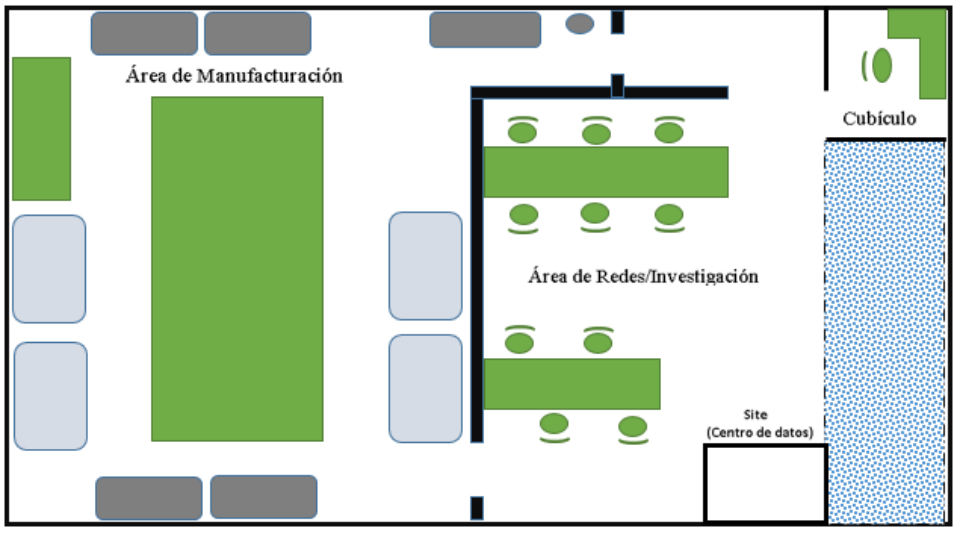


Figura 16 Áreas del laboratorio.

Se encontraron los siguientes servicios: el laboratorio cuenta con un servidor host que cuenta con un portal web acerca de las investigaciones que se realizan dentro de éste, cuenta con seguridad interna bajo los protocolos SSL/TLS, dentro del mismo se almacena información en bases de datos gestionada por el sistema MariaDB. Además éste servidor cuenta con la protección de un cortafuego de arquitectura *dual-homed*, que contiene un filtrado de tipo *packet filter* y tan solo contenía un set de 10 reglas básicas para

filtrado. Finalmente el servicio de red inalámbrica se proveía con un *Wireless Access* el cual contaba con cuatro puertos RJ45 para conexión cableada y una antena para red inalámbrica en la cual la contraseña estaba con una encriptación WEP y no gestionaba el control de acceso por medio de filtrado MAC.

La problemática más común en la conexión a internet era la pérdida repentina del servicio y la calidad del mismo, generando pérdida de datos e inconformidad de los usuarios. Por lo que se realiza un segundo análisis enfocado a la realización de una arquitectura de red de datos, logrando identificar y resolver los inconvenientes ocurridos en la transmisión de la información. Dicha arquitectura de red contempla una subred LAN y una subred DMZ a lo largo del laboratorio. Además ambas subredes serán distribuidas a través de un *swtich* de capa 2 y 3 del modelo ISO/OS, con el cual se regularán las comunicaciones entre las distintas redes locales, se aíslan física y lógicamente las subredes dedicadas a servidores, investigadores y usuarios. Aumentando así la seguridad en los distintos servicios y ejercicios que se realizan dentro del laboratorio.

Como se puede observar, dentro del laboratorio no existía una zona desmilitarizada como tal, dado que los servicios que se ofrecen a la comunidad convivían con la red de la UACM y el tráfico se mezclaba en uno o más dispositivos, lo cual representaba una falla en la capa de distribución en la red, por lo cual el funcionamiento en red se veía comprometido a ataques por cualquier método y a obtener acceso al servidor u otros equipos.

3.2 Diseño de red interna

Para el diseño se tomaron en cuenta los equipos que integraran la red interna hoy en día y cuáles permanecerán eventualmente fuera de ésta (sistemas dedicados). En relación a esto, se realizó un diseño de estructura física general del laboratorio, así mismo, se realizó la distribución de los elementos con los cuales se llevaron a cabo las prácticas de conectividad, orientando todo esto la mejor utilización y aprovechamiento del espacio.

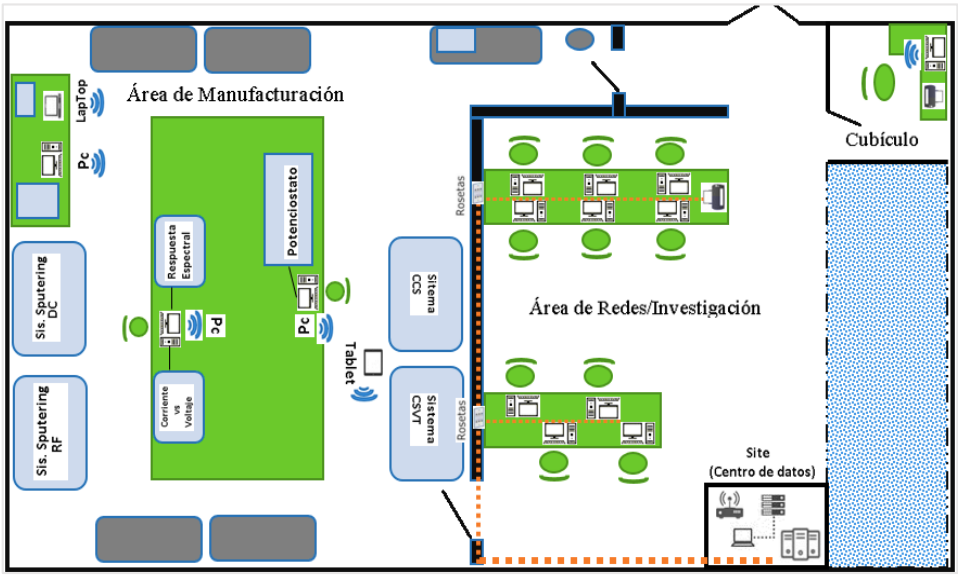


Figura 17 Areas del laboratorio con sus respectivos sistemas informaticos

En la figura 17 se muestra el laboratorio, las distintas áreas que lo integran y los sistemas informáticos que forman parte de cada una de éstas. El área de investigación se forma con 11 computadoras y una impresora funcionando en perfectas condiciones, el área de manufacturación cuenta con 5 equipos informáticos (3 Computadoras con tarjeta inalámbrica, una Laptop y una tableta electrónica), el área de cubículo cuenta con una computadora y una impresora, ambos sistemas cuentan con tarjeta inalámbrica para la conexión a la red inalámbrica del laboratorio. Por último, el centro de datos cuenta con 9 servidores que brindaran servicios web a la comunidad universitaria y servicios para cómputo matemático.

Dentro de la producción de esta red se reemplazo el cortafuego que estaba por uno que cuente con arquitectura *Screened Subnet* con la variante *Three-Legs* con un hardware de tamaño estándar, agregando servicios de seguridad, red y enrutamiento. El set de reglas se establece para beneficiar la calidad del servicio, asignando ancho de banda de manera uniforme para hacer eficiente el tiempo de respuesta de la red, se aseguran los servicios mediante un cortafuego con software OpenBSD con el cual se elimina la necesidad de realizar cambios en la configuración de los servidores, debido a que éste tipo de software los protege y aísla el tráfico entrante hacia ellos, evitando ataques maliciosos hacia los mismos.

Los requerimientos para la arquitectura de una red interna, deben de cumplir con lo que se desea lograr. Para lo cual se procedió a realizar un diseño, tomando en cuenta que la instalación de infraestructura interna requerirá permisos a posibles ampliaciones futuras.Además se tomaron en cuenta las necesidades particulares para el laboratorio, espacios y tipo de conexión (cableado o inalámbrico).

En la tabla 5 se especifican los dispositivos informáticos con los que hoy en día cuenta el laboratorio, y con los que se procedió a la elaboración del diseño, tanto físico como lógico de la red.

Tabla 5 Componentes para diseño de red

Cantidad	Dispositivo	Marca (distribuidor)
1	Switch	Hp 1910-24P-PoE
1	Punto de Acceso	Linksys by Cisco
15	Computadoras	Acer, Optiplex, LG, Lenovo
1	Laptop	Gateway
1	Cortafuego/PfSense	Dell-Evo
1	Servidor	Mac- Apple
1	Tablet/Android	Hp
5	Servidores	SuperMicro, Lufac
2	Impresoras de red	Hp
2	Tarjetas de red Fast y Gb Ethernet PCI	3com y Tp-Link

Para la infraestructura de red en el LACyTES, se tomaron en cuenta aspectos de diseño de una red, los cuales asocian las necesidades de los usuarios y para qué está destinada la red.

En el proceso se tomaron en cuenta las metas del laboratorio, las técnicas de almacenamiento, captura, caracterización y el proceso de datos de sistemas tales como: aplicaciones y servicios.

Dentro de las aplicaciones se encuentran las desarrolladas por Profiler, Nova 1.10 y Newport que son programas diseñados para trabajarn en los sistemas dedicados, estas aplicaciones hacen uso de la red inalámbrica conectándose eventualmente a ésta, con el propósito de evitar que sean atacados o infectados por un virus informático.

Los servicios se ofrecen a través de un portal web o aplicación móvil y son necesarios para almacenamiento y consulta de datos. Herramientas como *switch*, punto de acceso y la infraestructura de la red son clave para lograr acceder a estos servicios. Además de ayudar en el funcionamiento y desempeño de la red, logrando la meta y objetivo de este trabajo.

3.2.1 Evaluación de las necesidades

En lo que se refiere al diseño y la elaboración de la red del laboratorio se realizaron evaluaciones de los requerimientos usados para la producción de una red. Se tomó en cuenta los medios de transmisión, elementos de seguridad física, suministro eléctrico, cableado de red, puntos de red y equipos de interconexión que son utilizados para la elaboración y diseño de la red, basándose en la infraestructura del laboratorio y recursos disponibles.

Para la red de datos se emplearon los elementos de la tabla 6, la norma que se estableció para el cableado es la TIA/EIA 568-B-1, que va dirigida a: cableado horizontal, cuartos de telecomunicaciones, backbone y parámetros técnicos Categoría 3,4 y 5.

Tabla 6 Elementos para red de datos

Los requerimientos de la red
- Cable UTP Cat 5e.
- Conectores RJ-45
- Conectores RJ-45 hembra
- Rosetas de punto de red
- Switch Uplink de 24 puertos
- Wireless router 802.11
- Tarjetas de red para el diseño de las subredes
- En general lo que se ha determinado como *End Systems*

En la tabla 7 se referencian los elementos de la seguridad física, mecanismos de prevención y protección que permiten proteger físicamente cualquier recurso del sistema y el entorno en que se encuentra. Se siguieron los requisitos mínimos indispensables para la seguridad física según lo determina el estándar UNE-ISO / IEC 27002.

- Las paredes no deben permitir filtraciones de agentes externos que contaminen el entorno.
- Las ventanas deben contar con una protección y no permitir la visualización interna del laboratorio.

- Contar con equipo que controle la humedad y la temperatura la cual no debe superar los 25°, detectores y alamas de calor, humos y humedad, sistemas de extinción de incendios.
- Definir reglas de acceso de personal autorizado al laboratorio.
- El piso debe contar con un material antideslizante para prevenir accidentes.

Para los elementos de hardware

- Se debe contar con un sistema que controle la sobre carga de energía (UPS).
- Los componentes deben estar etiquetados con el fin de identificarlos.
- Generar copias de seguridad para brindar protección a la información y prevenir pérdidas de la misma.
- Mantener la información confidencial del laboratorio en un lugar donde que se encuentre al alcance del personal autorizado y no sólo manejarla por medio electrónico, sino también en medio físico.

Tabla 7 Elementos de seguridad física

Requerimientos de Seguridad Física
- UPS (Suministro de Energía No Interrumpible)
- Aire Acondicionado
- Extintores
- Cuarto de Telecomunicaciones (Site)
- Rack
- Tablero de distribución de electricidad

Así mismo se elaboró la tabla 8 que es un listado de requisición para módulos faltantes.

Tabla 8 Requisiciones faltantes para la red

Módulos faltantes
- Panel de conexiones
- Etiquetador de cables RJ47
- Fibra Óptica
- *Switch Core*
- Organizadores verticales y Horizontales
- Cinhos de velcro
- Canaletas para cableado estructurado
- *Tester* de cable de red RJ45

Dentro de la red diseñada, se pretende satisfacer la demanda de ancho de banda, conectividad, privacidad y esto se implementa mediante elementos de seguridad perimetral, tomando en cuenta los equipos que se encuentran en el laboratorio. En la tabla 9 se especifican estos elementos.

Tabla 9 Elementos de seguridad perimetral

Requerimientos de Seguridad a Red de datos	
- Cortafuegos (Software)	- pfSense, se establecerán controles de acceso y filtro paquetes.
- Control de acceso:	• El acceso a la red solo se permite al personal del laboratorio. • Cada usuario del laboratorio debe contar con usuario y contraseña para acceder al servidor puesto en red. • Los mecanismos de protección para acceder a la red y los recursos deben estar controlados por el administrador de seguridad. • El acceso a internet solo debe poder hacerse por el personal autorizado. • El acceso a sitios de internet potencialmente peligrosos deben estar restringidos.
- Control de Tráfico	• Determinar los servicios que se implementan en el laboratorio con el fin de permitir solo estos en la red. • El uso del ancho de banda debe estar regulado para no sobrecargar la red del laboratorio. • La red debe utilizarse solo con fines estratégicos del laboratorio. • Segmentar el tráfico de la red con el fin de administrar mejor los servicios.
- Control de Actualizaciones	• Actualizar constantemente la documentación. • Mantener los sistemas y programas actualizados.

Requerimientos de las estaciones de trabajo.

Los usuarios del laboratorio llevan a cabo distintos estudios como depósito de películas delgadas, caracterización espectroscópica, nuevos materiales, conductancia, sistemas electrónicos, etc. Por lo cual se entiende que las necesidades en cada uno son diferentes, esto ayudó mucho en la construcción de la red. Cada estación de trabajo cuenta con las aplicaciones de software necesarias para la productividad del usuario. Las plataformas o Sistemas Operativos (S.O.) que se usan más para elaborar dichos estudios son:

- Microsoft Windows: alrededor de 10 equipos de cómputo cuentan con este sistema operativo en su versión Windows 8. Éste se caracteriza por el uso de un servicio de directorio llamado Active Directory para administrar los recursos de la red, además cuenta con aplicaciones como: Microsoft Office.
- Lunix: Este sistema operativo está basado en UNIX, este sistema incluye distribuciones como Red Hat, Debian, Fedora, Ubuntu, entre otro más. Debían es la versión implementada en el servidor, y algunos otros usuarios utilizan distribuciones similares para el desarrollo de proyectos de ingeniería, física o matemáticos.

- Android: Este sistema regularmente se encuentra en dispositivos móviles cuenta con seguridad básica. Para reforzar su seguridad es necesario la instalación de aplicaciones propietarias. La tableta electrónica forma parte de los sistemas dedicados que se encuentran en el laboratorio.

La seguridad en las estaciones de trabajo es esencial. En un caso particular, para brindar más seguridad se cuenta con un antivirus ESET NOD32 para el caso del S.O. Windows y para los usuarios que cuentan con algún S.O. UNIX se usa Bitdefender, que proporcionan paquetes de seguridad diseñados a medida con condiciones aceptables. Para la tablet se puede instalar antivirus, por ejemplo, AVG Security, CM Security entre otros más que se pueden encontrar en la tienda virtual.

3.3 Cálculo de subredes por método de VLSM

Para lograr que las distintas áreas que conforman la red del laboratorio funcionen como una red unificada, se comparte el espacio de direccionamiento, es decir, a partir de una dirección IP privada se generaron subredes con direcciones igualmente privadas que cumplen con los requerimientos de direcciones requeridas para cada área.

Para la elaboración del diseño la unidad de informática del plantel, recomendó asignar el identificador de subred **10.40.6.X/8** (Dirección Clase A, Privada) para fines de control, recordando tomar en cuenta la cantidad de sistemas informáticos del laboratorio y la posible futura adquisición de nuevos sistemas para el cálculo de la submáscara de red. Además recomendó asignar una dirección de red como red WAN que se encontrara dentro del rango de direcciones **172.17.120.145 – 172.17.120.158** y se le notificara de la dirección seleccionada con el propósito de ejercer control sobre ésta y que otras redes no tomen la misma IP.

Tomando en cuenta las recomendaciones se procedió a elaborar el cálculo de acuerdo a las **Ecuaciones (1)** y **(2)** que indican cómo obtener el número de subredes y el número de host para cada subred, recordando que estos no exceden de 256 host. Por lo que, se consideran los valores binarios para la máscara de subred que se observan en la tabla 10.

Tabla 10 Subred de clase C de 7 casos matemáticos hay 5 casos prácticos

Mascara de subred en IPv4 (último octeto en binarios)	Mascara de subred en IPv4 (decimal)	Bits para Subredes/#subredes	Bits para host/host
10000000	255.255.255.128	1/2	7/126
11000000	255.255.255.192	2/4	6/62
11100000	255.255.255.224	3/8	5/30
11110000	255.255.255.240	4/16	4/14
11111000	255.255.255.248	5/32	3/6
11111100	255.255.255.252	6/64	2/2
11111110	255.255.255.254	7/128	1/0

Analizando la recomendación, el segmento de red **10.40.6.X** es clase A del bloque de direcciones IP privadas. Esta dirección cuenta con la máscara natural de red 255.0.0.0 que proporciona 16,777,216 direcciones IP (2^{24}). Por consiguiente se eligió un direccionamiento que se adapte al número de sistemas que se encuentran en cada área del laboratorio para poder mantener homogéneo el direccionamiento, esto con la finalidad de ayudar a la sumarización[8] de redes en un futuro y lograr optimizar las actualizaciones del protocolo de encaminamiento que se implementen posteriormente.

Se tomó como partida mascara de red **255.255.255.0** para iniciar la división de subredes. Mediante los cálculos realizados se obtiene la siguiente cantidad de subredes y de sistemas host por red. Los cuales son asignados a cada área del laboratorio posteriormente.

- Para red LAN se necesitan 12 host, conectados vía cable UTP.

Numero de nodos = 12	8<12<16	16 host por subred
Máscara decimal	Número de redes	Número de Host/red
255.255.255.240	$2^4 - 2 = 14$	$2^4 - 2 = 14$

- Para la Red DMZ se necesitan 5 host, considerando su expansión se toman en cuenta 9 host, conectados vía cable UTP.

Numero de nodos = 9	8<9<16	16 host por subred
Mascara decimal	Número de redes	Número de Host/red
255.255.255.240	$2^4 - 2 = 14$	$2^4 - 2 = 14$

El número de bits utilizados para la porción de red LAN y DMZ es de 28 bits, tomando los 4 bits restantes para la generación de 16 direcciones de host utilizables dentro de cada subred.

Cada subred cuenta con 16 direcciones (no se cuenta la primera, ni la última dirección IP del rango de la subred) y en cada una se configuraran direcciones de manera estática y algunas de manera dinámica por medio de un servidor DHCP a través de un *router* que funciona y se adapta a través de un cortafuego por software. Este cortafuego-*router* enlaza las subredes internas con la red externa (WAN) la cual pertenece a la infraestructura de red de la UACM. El listado de subredes y su rango de direcciones IP para una máscara de subred 255.255.255.240 se muestra en la figura 18, donde solo se utilizaran las subredes SN1, SN2 y el resto se dejan para una posible extensión de red.

[8] Permite reducir considerablemente las entradas en la tabla de enrutamiento al resumir la información de direccionamiento de dos o más subredes en un solo bloque IP

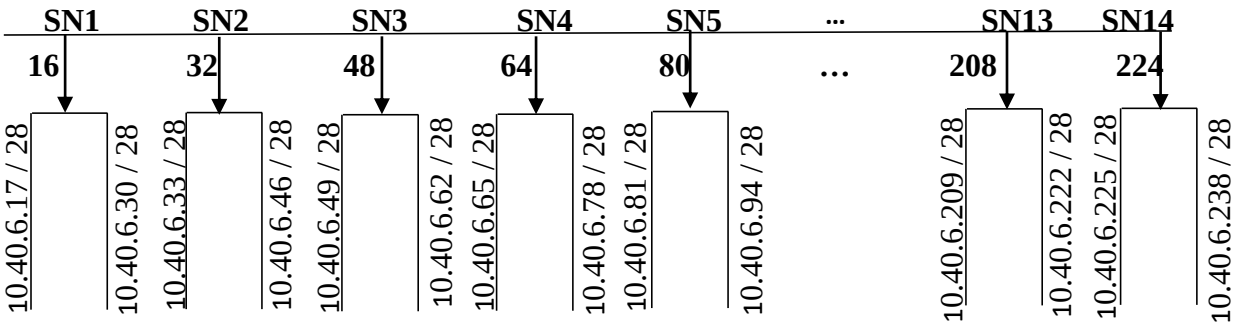


Figura 18 Esquema lógico de subredes disponibles para el laboratorio

3.3.1 Topología física de red

Debido a que el laboratorio hace uso de servidores para almacenar, compartir y gestionar datos internamente, se hace uso de la DMZ para montar estos servicios en el servidor. Por tal razón, se hace uso de la topología que se muestra en figura 19, la cual brinda seguridad a los servidores internos, fundamentando en que están filtrados de internet por el cortafuego-*router*.

Esto permite simplificar la forma en la cual se administrará y configurará la seguridad, debido a que ésta estará centralizada en el cortafuego con el software pfSense.

Como se observa en la figura 19 la topología toma en cuenta tres zonas que comprenden la red interna, además de la red externa. La interfaz de enlace WAN del cortafuego se conecta a la red de la UACM, como si ésta fuera un equipo de cómputo más. Los equipos destinados a cada subred se conectan a un conmutador (*switch*), donde se definieron las VLANs para cada subred y posteriormente se conectaron los equipos y donde se realizaron las pruebas de conectividad.

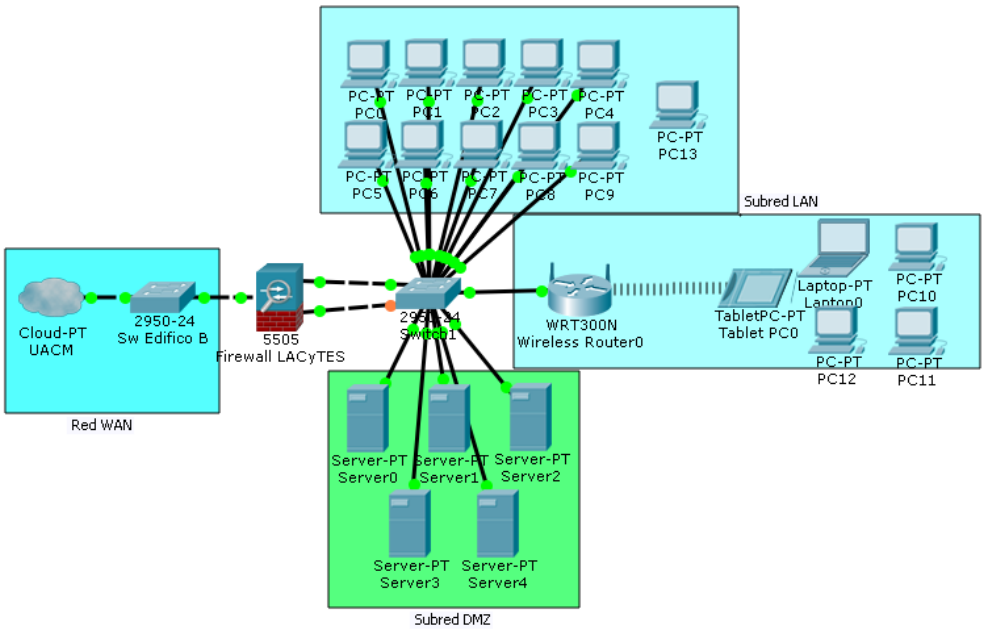


Figura 19 Topología física de infraestructura interna de LACyTES.

Para lograr distinguir las distintas subredes (LAN y DMZ) se elaboraron particiones en el *switch*, tres segmentos independientes mediante VLANs. Se utilizó la herramienta *Web Mannagement Platform*, que permitió definir las VLAN y asignar cada puerto ethernet a una VLAN de manera gráfica, a través del navegador, como se observa en la figura 20.

Figura 20 VLANs en conmutador para las redes diseñadas.

La subred LAN se definió con 12 puertos (recuadro verde de la figura 20) al puerto número 1 del conmutador se conectó, por medio de cable UTP, la interfaz LAN que proviene del cortafuego, a partir de este puerto se distribuyen las direcciones IP al resto de los puertos de la VLAN. En el puerto número 12 se conectó el *wireless router* que brinda conexión a los sistemas inalámbricos (dentro de éstos se encuentran los sistemas dedicados y la tableta electrónica que contará con la aplicación diseñada). La subred DMZ (recuadro rojo de la figura 20) cuenta también con 12 puertos disponibles, donde se conecta por cable UTP la interfaz DMZ proveniente del cortafuego al puerto número 13 de *switch*, a partir de éste se distribuyen las direcciones IP hacia los demás puertos de la subred DMZ donde se conectaron los distintos servidores.

Las VLANs definidas logran establecer comunicación entre ellas a través del cortafuego, dentro de éste se establecen protocolos y reglas para que ambas redes logren tener comunicación sólo en el sentido de la subred DMZ a la subred LAN, de esta manera se determina que el cortafuego también funciona como un *router*.

3.3.2 Red inalámbrica para sistemas dedicados

Esta red se conecta directamente a la VLAN de la subred LAN que se distribuye en el *switch* y para el diseño de ésta red inalámbrica se consideraron dos tipos de configuración, en base a las características del *router* inalámbrico Linksys, para aquellos sistemas que cuentan con tarjeta inalámbrica dentro del área de manufacturación**.**

La primera configuración es en modo puente o AP y la segunda configuración en modo *router*. La configuración en modo *router* se encarga de comunicar dos o más redes independientes y realiza la búsqueda del camino más rápido para la comunicación, esta opción es más compleja debido a que requiere realizar direccionamiento NAT dentro del *router*, posteriormente se establecen las comunicaciones entre las VLANs por medio de troncales y se finaliza estableciendo reglas de control,

acceso y filtrado dentro del cortafuego, por lo que se genera un sistema más complejo y tardío. Por simplicidad se decidió realizar la configuración en modo puente y realizar sólo el direccionamiento en el cortafuego.

Como se muestra en la figura 21 la configuración de seguridad inalámbrica cuenta con un cifrado WPA2 con algoritmo TKIP (802.11i), además de controlar el acceso a la red por medio de filtrado MAC, para lo cual se pidió a cada usuario proporcionar la dirección MAC de sus dispositivos móviles y portátiles, se creó la lista de acceso solo a usuarios del laboratorio, para prevenir cualquier intrusión o ataque proveniente de un sistema ajeno.

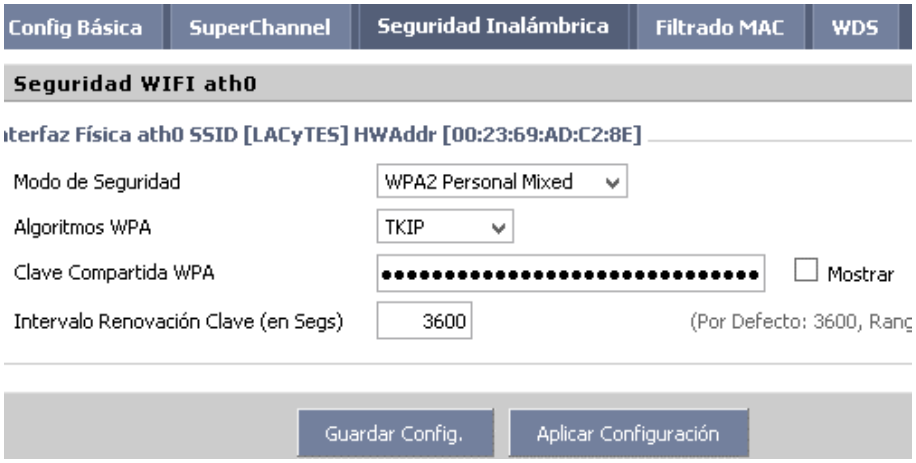


Figura 21Configuracion de seguridad en la red inalambrica del laboratorio

3.4 Implementación de filtrado de contenido con pfSense

PfSense es una distribución basada en *FreeBSD*, cuyo objetivo principal es obtener un cortafuego y enrutador fácil de configurar por medio de una interfaz amigable vía web (WebGUI) con el administrador de la red, se instala en cualquier computadora, incluyendo computadoras con una sola tarjeta de red. Permite proteger a una red de computadoras de las intrusiones provenientes de otras redes, además de filtrar el tráfico que entra a la red interna y el que sale a la red externa.

Desde su interfaz web se pueden ampliar las funcionalidades de pfSense, permite elegir que paquetes o módulos de seguridad se deseas instalar, el sistema lo descarga y lo instala automáticamente. El cortafuego forma parte del Kernel del sistema, esto permite que el filtrado de paquetes se realice a nivel de kernel, llamado *packet filter* (PF) el cual permite asignar prioridades por tipo de tráfico.

PfSense se puede configurar como un cortafuego permitiendo y denegando determinado tráfico de las redes tanto entrantes como salientes a partir de una dirección IP ya sea de una red o de un sistema host origen, también ofrece el filtrado avanzado de paquetes por protocolo y puerto. PfSense agrega otras funcionalidades, las cuales permiten hacer de éste una herramienta muy potente, que a futuro se puede emplear. Teniendo en cuenta los conocimientos que se tienen acerca del software, la facilidad de empleo y la seguridad que ofrece al diseño de la red, se elige este software como cortafuego para cumplir los objetivos de este trabajo.

Una vez obtenida la división en subredes se prosigue con la instalación de pfSense que se llevó a cabo en una computadora cuyas características técnicas principales son:

- 3 tarjetas de Red (RealTek Gb Ethernet, 3Com Fast Ethernet y Broadcom Gb Ethernet)
- Disco Duro de 40GB
- Memoria Ram de 1GB
- Teclado y Monitor Dell

Para ello se ha seguido los pasos del documento anexo A-1 que se puede consultar al final de este documento.

El motivo de las tarjetas de red de dichas marcas es para ejercer la compatibilidad de hardware con PfSense, dado que hay una lista de hardware de marcas y modelos que son ideales para la operación ideal de éste software. Como ya se vio la red WLAN se conecta directamente a la VLAN de la red LAN definida en el *switch* por tal motivo no se usó una interface de red en el cortafuego.

Al concluir la instalación se designó a cada interfaz de red las direcciones, prefijo de subred e IP en el cortafuego que se muestran en la tabla 11.

Tabla 11 Direcciones de red de acuerdo a Interfaces

Interface.en cortafuego	Subred	Dirección red	IP cortafuego
bge0	WAN	172.17.120.0/28	172.17.120.146
re0	LAN	10.40.6.0/28	10.40.6.17
xl0	DMZ		10.40.6.33

3.4.1 Direccionamiento DHCP

Sirve principalmente para distribuir direcciones IP dentro de una red, permite que una computadora que se encuentre conectada a una red pueda obtener su configuración en forma dinámica, es decir, sin intervención de un administrador o algún particular. Por lo cual se aprovecha el cambio de la infraestructura de red interna con la finalidad de que todas las direcciones de red de todos los equipos se gestionen de manera centralizada. De esta manera se utilizará el protocolo DHCP para las redes LAN y DMZ. Este protocolo es un protocolo de configuración de red centralizado, es decir, que cuando un equipo configurado por DHCP enciende envía una trama de red solicitando una configuración (*broadcast*). El servidor la recibe, y verifica si hay alguna configuración establecida, de no ser así asignará una configuración determinada al equipo. Este protocolo tiene ciertas ventajas pues en caso de cambios en la topología de red, no es necesario ir a configurar los parámetros equipo por equipo. Como desventaja tiene el hecho de que estando los equipos configurados para recibir direcciones, si algún dispositivo arranca un servicio DHCP para asignar direcciones no validas, puede causar problemas como denegación de servicio (DDoS).

Para el caso del laboratorio, los usuarios tienen un puesto de trabajo asignado, con lo que el servicio DHCP se decide implementarlo, agregando además funcionalidades que ofrece PfSense como el control de IP con MAC lo cual permite administrar la red de una forma más segura. De esta manera se asegura qué, si algún sistema host ajeno al laboratorio se le obstaculice el acceso a internet aun cuando el usuario coloque la dirección IP en su computadora o dispositivo.

Los servidores que se ubicaron en la red DMZ se encuentan sujetos al rango de direcciones de red de la subred SN2, que son de la dirección **10.40.6.33/28** a **10.40.6.46/28**. En cuanto a la red LAN todas las direcciones del rango de subred SN1, **10.40.6.17/28 - 10.40.6.30/28** fueron reservadas para los equipos que no funcionen como servidor, es decir, computadoras, laptop y dispositivos móviles de usuarios del laboratorio, cuentan con conectividad hacia el exterior. Esto se puede entender un poco más en la figura 23.

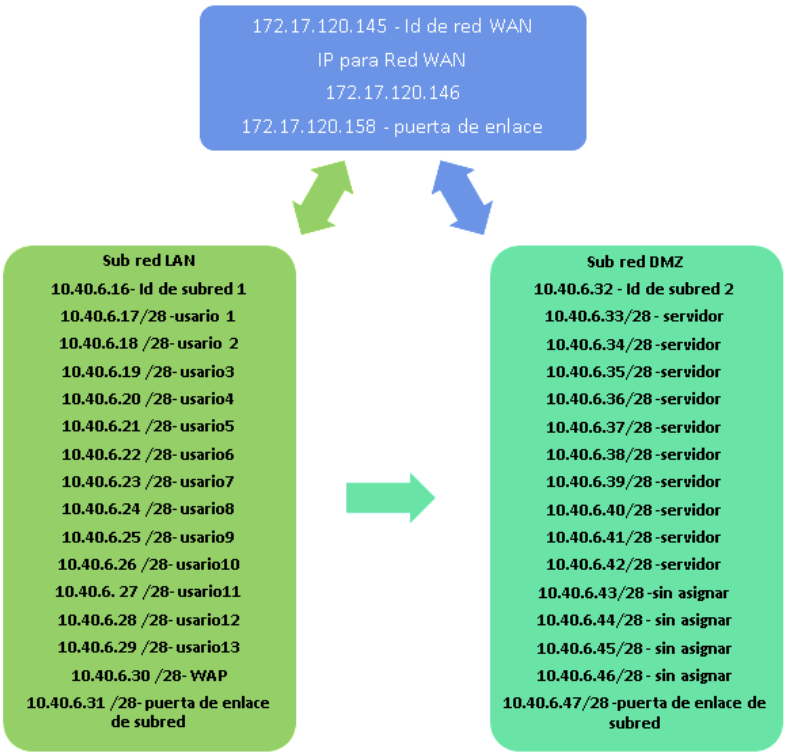


Figura 22 Diagrma de comunicación entre las subredes diseñadas

Para la red inalámbrica, se establecieron direcciones IP dentro del rango **192.168.1.1/24 – 192.168.1.19/24** dado que es el número máximo de usuarios que contiene el laboratorio. Estas direcciones fueron asignadas libremente para los distintos dispositivos, es decir esta red inalámbrica ejerce perfectamente el protocolo DHCP. Por lo que cada Smartphone, tabletas electrónicas o sistemas dedicados podrán conectarse siempre y cuando se encuentre en la lista que permite el acceso, de no ser así el servicio de conectividad será negado.

3.4.2 Control de IP por MAC en pfSense

En el anexo A-2 se muestra el procedimiento para activar el servicio DHCP en cada subred. Dentro de este mismo servicio pfSense ofrece distintas funcionalidades basadas en dnsmasq[9] tales como:

- La posibilidad de asignar direcciones IP estáticas en función de la dirección MAC del dispositivo.
- La posibilidad de capturar fácilmente las direcciones MAC, sin tener que introducirlas manualmente.
- La posibilidad de cerrar las lista de direcciones MAC, impidiendo la conexión de dispositivos desconocidos.
- Poder encender (wake-up) dispositivos de la red para tareas de mantenimiento remoto, siempre y cuando el dispositivo remoto cuente con esta característica.
- Tener una pantalla donde se tiene la relación de todos los equipos de una red.
- La movilidad de equipos entre redes.
- El uso de DHCP para evitar la configuración de conexiones de red de cada computadora o dispositivo.

Se consideraron las tres primeras funcionalidades y a continuación se describe cómo se llevó a cabo la implementación de éstas.

En este caso particular se implementó un sistema con las tres interfaces de red que se interconectan a la red WAN con las subredes (LAN y DMZ). La interface de la red WAN cuenta con una dirección estática (**172.17.120.146**) y permite la comunicación con la red de internet.

Para la realización de la siguiente configuración fue necesario efectuar un inventario completo de todos los equipos, obteniendo las direcciones MAC de cada sistema host, impresora, servidor, *switch* o *wireless router*. Esto permitió ejercer un control y seguridad de los sistemas que hay en el laboratorio, ya que de esta manera se puede denegar el acceso a equipos desconocidos que puedan conectarse a la red (conectándose directamente por medio de conexión RJ45).

Para el procedimiento de configuración de control IP por MAC se realizó lo siguiente:

Hasta este punto, el sistema se encontraba configurado para tener conectividad en las subredes definidas y contaba con un sistema de conexión de direcciones automático (DHCP), que fueron asignadas cuando los sistemas encendieron. Para la realización del control de asignación de direcciones IP por MAC de los equipos se accedió a la computara que contiene el cortafuego a través del WebGUI donde usuario y contraseña se muestran en el anexo A-1. Se ingresó al menú '**Services/DHCP Server**', por defecto ubica en la interfaz de subred LAN, pero las siguientes tareas también se realizaron para la interface DMZ.

En esta parte de la configuración se permitió, como opción, que la conexión de red tenga una reserva de dirección IP asignada por DHCP. Debido a que la red LAN diseñada cuenta con 12 equipos y la red

[9] Dnsmasq: permite poner en marcha un servidor DNS y un servidor DHCP de una forma muy sencilla. Está diseñado para proporcionar DNS y opcionalmente DHCP a una red pequeña.

DMZ cuenta con 9 (considerando una posible expansión de equipos) se configuró para que ambas tengan la reserva de dirección IP para un solo sistema informático ajeno al laboratorio.

Como se observa en la figura 23, pfSense proporciona gráficamente los parámetros como el prefijo de dirección de red, la máscara de subred, el rango de IP disponibles así como el rango que deseamos escoger para asignar IP estática o dinámica a los equipos,.

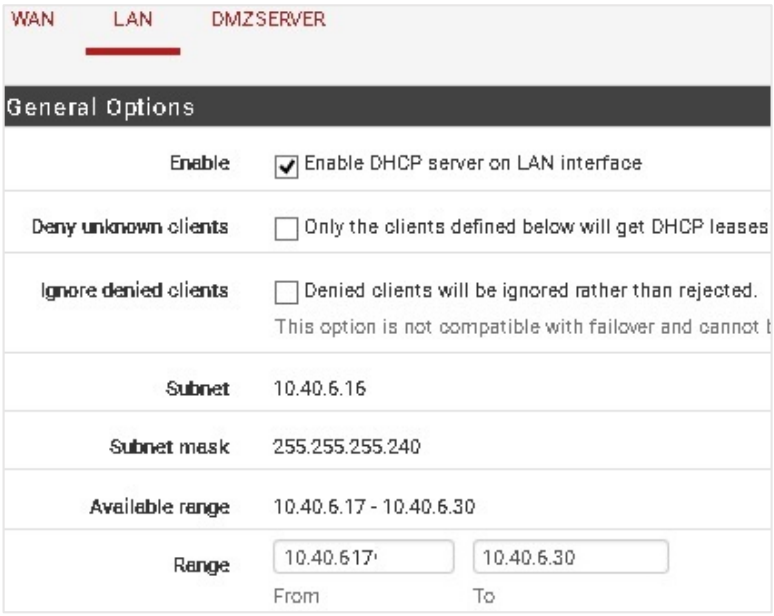


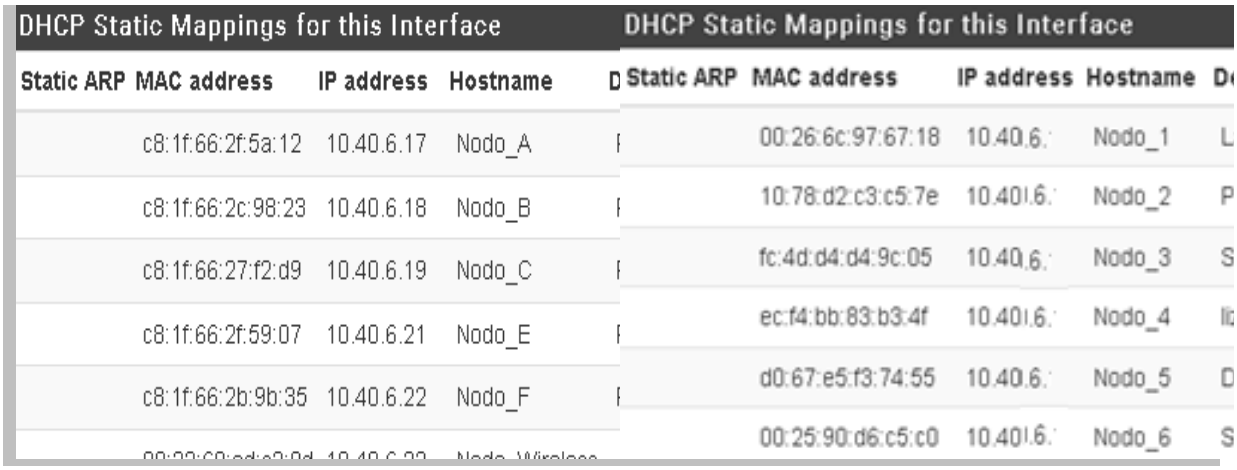
Figura 23 Parámetros mostrados al acceder al menú DHCP Server

Para activar los rangos de direcciones, se activó '**Deny unknown client**', esta opción evita el acceso por medio DHCP a cualquier sistema host que no esté en la lista de direcciones MAC permitidas para tener acceso a internet. El primer rango permite asignar direcciones IP a los clientes o equipos por medio de dirección MAC de cada equipo, es decir, se deniega el servicio DHCP y a cada equipo se le asigna una dirección IP estática. El segundo rango asigna direcciones IP automáticamente mediante el servicio DHCP. Por lo cual para controlar la asignación de direcciones IP se define un rango para control de equipos por dirección MAC.

Posteriormente al desplazar hacia abajo en la WebGUI y se encuentra la opción "**ARP entries**". La cual permite establecer que sólo los equipos que figuren en la lista de direcciones MAC puedan comunicarse con el cortafuego además de contar con el servicio de internet. Dado que se consideró un sólo sistema host para asignación de IP por DHCP esta opción fue descartada, pero se deja su configuración para un cambio futuro.

Los cambios son guardaron después de que se establezca la lista de todos los equipos en cada red. Para la asignación de las IP estáticas, nuevamente se realizó un desplazamiento hacia abajo donde se ubica la tabla de nombre '**DHCP Static Mappings for this Interface**' que permitió administrar las direcciones IP con respecto a la dirección MAC de cada sistema, en esta tabla se debe agregar un nombre (Hostname) y una breve descripción del equipo como se muestra en la figura 24. Donde además se muestran resultados de las interfaces configuradas para cada porción de subred que tiene la topología configurada

de manera general, del lado derecho de la figura se observa la subred LAN y los nodos de ésta, los cuales reciben una etiqueta de manera alfabética. Del lado izquierdo se observan los nodos de la subred DMZ que de manera similar reciben un etiquetado pero ésta es de forma numérica.



DHCP Static Mappings for this Interface

Static ARP	MAC address	IP address	Hostname	D
	c8:1f:66:2f:5a:12	10.40.6.17	Nodo_A	
	c8:1f:66:2c:98:23	10.40.6.18	Nodo_B	
	c8:1f:66:27:f2:d9	10.40.6.19	Nodo_C	
	c8:1f:66:2f:59:07	10.40.6.21	Nodo_E	
	c8:1f:66:2b:9b:35	10.40.6.22	Nodo_F	

DHCP Static Mappings for this Interface

Static ARP	MAC address	IP address	Hostname	D
	00:26:6c:97:67:18	10.40.6.	Nodo_1	L
	10:78:d2:c3:c5:7e	10.40.6.	Nodo_2	P
	fc:4d:d4:d4:9c:05	10.40.6.	Nodo_3	S
	ec:f4:bb:83:b3:4f	10.40.6.	Nodo_4	
	d0:67:e5:f3:74:55	10.40.6.	Nodo_5	D
	00:25:90:d6:c5:c0	10.40.6.	Nodo_6	S

Figura 24 Configuración de equipos en ambas subredes. (Izq subred LAN, Der subred DMZ)

3.4.3 Reglas de filtrado

Generalmente los cortafuegos asignan rutas a los paquetes antes de que logren entrar al interior de la red. El filtrado de éstos es implementado previo a atravesar al cortafuego (dependiendo de cómo se encuentren configuradas las reglas). Las reglas que se establecieron de acuerdo a cada interfaz de red (WAN, LAN y DMZ), controlan el tráfico entrante al cortafuego y por lo tanto su conectividad. Además es necesario mencionar que al haber definido las reglas, éstas se ejecutan en orden descendente (de arriba hacia abajo), por lo tanto, fue necesario establecer el orden adecuado para una buena respuesta del cortafuego y las distintas reglas que lo componen.

Las reglas realizan una serie de acciones, que asignan criterios determinados los cuales son:

- Aceptar: si el paquete cumple con las condicione, se acepta y el cortafuego lo encamina hacia su destino.
- Bloquear: en caso de que el paquete ingrese al cortafuego y no cumple las condiciones se desecha, el cortafuego no realiza ninguna acción.
- Rechazar: el paquete se desecha y se envía al emisor un paquete comunicando que su petición fue rechazada.

Regularmente cuando no se desea permitir un tráfico, se bloquean los paquetes, sin dar una explicación al emisor. Las condiciones en cada regla varía respecto a lo que se desea permitir, por ejemplo: ¿de qué manera?, ¿que no se permite?, etc.

Para la configuración de reglas, fue necesario trabajar con lo que se tenía ya implementado hasta el momento en pfSense (Anexos A-1. y A-2.).

Hay dos conceptos que se analizaron para realizar el proceso de configuración de reglas, el primer concepto simplifica mucho la administración, pfSense lo denomina 'Alias'. Alias es un identificador, que puede ser uno o un grupo de redes, direcciones IP y puertos. Cuando se define un alias con estas características el número de reglas es mucho menor y por lo tanto la administración es más sencilla.

El segundo concepto es más extendido, debido a que se define un identificador por cada dirección IP, puerto y red. La diferencia entre ambos conceptos es que a un alias definido y empleado por pfSense no permite eliminarlo y modificarlo, pues se debe considerar, qué dirección, puerto o protocolo se elimina o se modifica, sin embargo; cuando se define una regla particular, ofrece la ventaja de poder eliminarla, modificarla, desactivar y organizar el orden en que se desea ejecutar la regla, lo cual para algunos resulta mucho mejor.

Para empezar a trabajar las reglas, primero se consideró la existencia de los servidores y equipos que solo funcionan como clientes, de tal forma que estos tengan conectividad externa e interna.

En las redes WAN y LAN se encuentran reglas definidas por defecto después de la instalación, estas no deben ser modificadas (cambiar puerto, protocolo, fuente, acción) o movidas de la fila en que se encuentra. Estas reglas son de alta prioridad dado que brindan conexión a las demás redes y cualquier otra regla definida se debe ordenar por debajo de éstas.

Configuración de reglas:

Para la definición de nuevas reglas se accedió nuevamente a la interface del WebGUI, en el menú '**Firewall**/**Rules**', donde se aprecian las redes que forman parte del cortafuego y además se encontraran las reglas definidas por defecto mencionadas anteriormente, ver figura 25. Se dio clic en el botón '**Add**' y se muestran las características para para la creación de una regla.

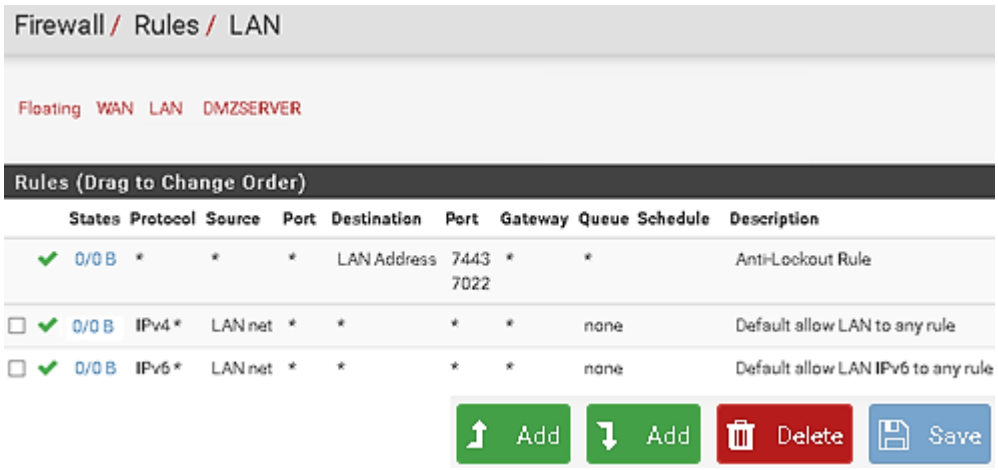


Figura 25 Interfaces de red y regla definidas por defecto.

Las características se enlistan a continuación, las utilizadas en este trabajo se muestran con un ✓ al inicio y las que no se utilizaron se marcaran con un • al inicio de su descripción.

- ✓ Acción: permite elegir qué hacer con los paquetes que cumplen los criterios especificados (aceptar, bloquear o rechazar) para este trabajo se selecciona “Pass” para permitir los paquetes que contaran con los criterios de los siguientes puntos.
- Inhabilitar: si hay una regla ya definida, se puede desactivar sin eliminarla de la lista, para este caso no hay una regla que se quiera deshabilitar.
- ✓ Interfaz: se elige la interfaz donde los paquetes deben llegar a coincidir con la regla.
- ✓ Familia de dirección: versión del Protocolo de Internet que se aplicara en la regla, puede ser IPv4, IPv6 o IPv4 + IPv6.
- ✓ Protocolo: protocolo sobre el que se verificaran los criterios estos pueden ser: TCP, UDP, TCP/UDP, ICMP, entre otros más que se encuentran en la lista despegable de esta opción.
- ✓ Fuente: se especifica el puerto de origen o un rango de puertos.
- ✓ Destino: se especifica el puerto destino o rango de puertos.
- Intervalo de puertos de destino: se especifica el puerto de destino o rango de puertos para la regla.
- ✓ Registro de evento (log). cuando se cumple la regla se genera un registro, con los datos del paquete y la acción realizada. Para conocerlos se selecciona esta opción.
- ✓ Descripción: permite ingresar un texto que funcione como identificador o referencia administrativa de la regla.
- Opciones avanzadas: se utiliza para limitar la regla en caso de muchas conexiones, banderas de TCP, horario de ejecución de la regla, entre otras opciones más.

Estas características son auto explicativas, lo que permitió definir las reglas que se muestran en la figura 26 donde se muestran los resultados de las reglas establecidas en una interfaz del cortafuego. Se puede identificar la importancia del orden de las reglas, la primera regla permite el acceso a toda la red de la universidad, que abarca las mayoría de las comunicaciones (internas principalmente). Seguidamente se observan reglas que se definieron para el acceso y los filtros que se establecen en cada una.

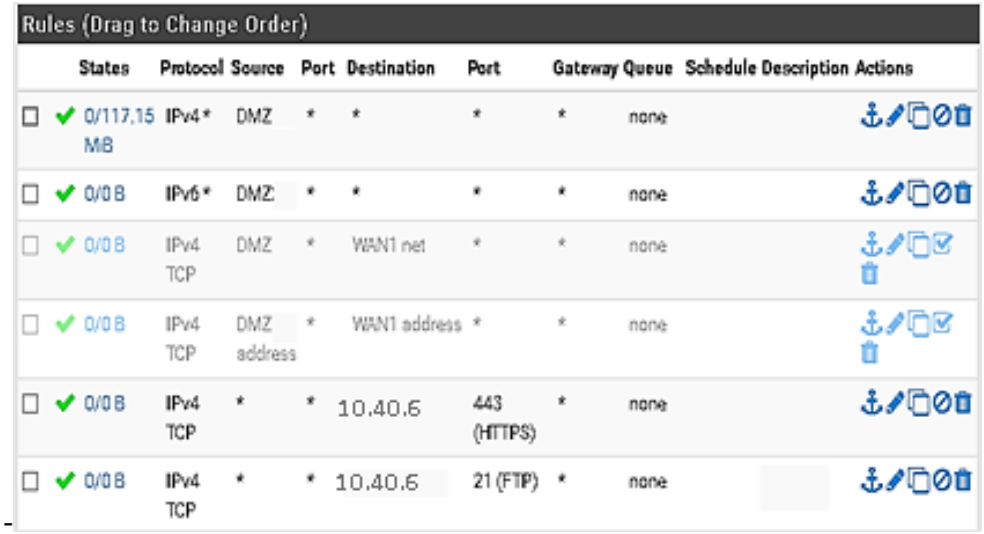


Figura 26 Listado de reglas en una de las interfaces de pfSense.

3.4.4 Acceso externo a servicios de LACyTES mediante NAT

El direccionamiento que se utilizó en la red interna es direccionamiento privado, lo que se entiende que éste no puede atravesar más allá del cortafuego-router que lo tiene configurado. Para lograr acceder a los servicios host del laboratorio, el cortafuego debe traducir las peticiones externas y enviarlas al servidor correspondiente. A este mecanismo o reglas se les denomina NAT (*Network Address Translator*), el cual asocia una dirección externa hacia una dirección interna y viceversa. PfSense ofrece varias opciones para realizar este direccionamiento como son: PortForward, 1:1 y Outbound.

- PortForward: define las reglas NAT *pre-routing*. Cuando se agrega un puerto de salida, también se debe agregar una regla en el cortafuego para permitir el tráfico a la dirección IP interna designada por el puerto, es decir, asocia la dirección IP y puerto externos con una dirección IP y puertos internos, por lo que únicamente los puertos definidos en estas reglas del cortafuego serán accesibles hacia el servidor para un usuario remoto, lo cual aporta seguridad y control.
- 1:1: permite definir reglas por medio de la dirección IP de un servidor host y que éste sea totalmente visible desde el exterior, es decir vincula una dirección específica interna (o subred) a una dirección externa específica. El tráfico entrante desde internet a la dirección IP especificada se dirige hacia la IP interna asociada. El tráfico de salida a internet desde la IP interna especificada se originará en la IP externa asociada.
- Outbound: define las reglas NAT *post-routing* que permiten enmascarar la LAN con la dirección IP de la WAN, con la finalidad de que los equipos pertenecientes a la LAN tengan acceso a internet por medio del enmascaramiento de la dirección pública.

Para la elaboración del direccionamiento de servicios se realizaron sólo las configuraciones PortForward y Outbound que aportan lo necesario para esta parte del trabajo. A continuación se describen los procedimientos que se siguieron para la configuración de reglas NAT Outbound:

a) Configuración Outbound:

Para que pfSense permita que las redes tengan comunicación a internet fue necesario configurar las reglas *post-routing*. En este cortafuego se generaron automáticamente las reglas después de haber implementado el siguiente proceso.

Se accedió al menú '**Firewall/NAT**', y se seleccionó la pestaña '**Outbound**'. Luego se observa que aparecen varias opciones de las cuales se escogió '**Manual Outbound NAT rule generation**'. Al seleccionar '**Save**' automáticamente el cortafuego genera las regla NAT *post-routing*, si se desea se pueden editar, eliminar o agregar según sea necesario. La configuración y resultado para estas reglas se observan en la figura 27 donde se muestra en la parte inferior las reglas que enmascaran cada subred, a partir del prefijo inicial de cada sub red, con la dirección de IP de WAN.

	Interface	Source	Source Port	Destination	Destination Port	NAT Address	NAT Port	Static Port	Description
☐ ✔	WAN	127.0.0.0/8	*	*	500	WAN address	*	✔	Auto created rule for ISAKMP - localhost to WA
☐ ✔	WAN	127.0.0.0/8	*	*	*	WAN address	*	⇄	Auto created rule - localhost to WAN
☐ ✔	WAN	10.40.6.16/28	*	*	500	WAN address	*	✔	Auto created rule for ISAKMP - LAN to WAN
☐ ✔	WAN	10.40.6.16/28	*	*	*	WAN address	*	⇄	Auto created rule - LAN to WAN
☐ ✔	WAN	10.40.5.32/28	*	*	500	WAN address	*	✔	Auto created rule for ISAKMP - DMZSERVER to WAN
☑ ✔	WAN	10.40.5.32/28	*	*	*	WAN	*	⇄	Auto created rule - DMZSERVER to WAN

Figura 27 Opciones de *Post-routing*

Estas reglas no controlan el tráfico de salida proveniente de las interfaces de red, solo determinan cómo se maneja éste a medida que sale.

Una vez que se llevó a cabo la configuración, se continua con los procedimientos de la siguiente forma de direccionamiento y de igual manera se describen los procesos para la configuración de reglas NAT *pre-routing* o como lo define pfSense *PortForward*.

b) Configuración PortForward:

La forma más recomendable de dar a conocer los servicios web que ofrece el laboratorio es a través de una dirección IP que logre el direccionamiento pre-routing, que como se mencionó permite la salida a ciertos servicios del servidor host y redireccionar las peticiones a diferentes puertos de este mismo.

Para definir estas reglas fue necesario ubicarse en el mismo menú de la configuración anterior '**Firewall/NAT'** y se seleccionó la opción. '**PortForward**' donde se procedió a definir una nueva regla NAT, para esto se presiona en la opción '**Add**' donde consecutivamente muestra las características para la creación de estas reglas.

Las características se despliegan a continuación, las opciones utilizadas para la creación de una regla se muestran con un ✓ al inicio de su descripción, las que no se utilizaron se marcaran con un •.

- Inhabilitar: Si hay una regla NAT ya definida, se puede desactivar sin eliminarla de la lista, para este caso no se utiliza pues de momento no hay reglas que se deseen deshabilitar.
- No RDR (NOT): Esta característica esta sin marcar y así permanecerá para este trabajo. Su función es negar la redirección de tráfico que coincide con los valores especificado es esta regla.
- ✓ Interfaz: Se elige la interfaz donde se origina el tráfico y en donde aplicara esta regla, por lo general se especifica WAN.
- ✓ Protocolo: Se elige el protocolo del tráfico proveniente de una red externa que se desea permitir. La mayoría de los casos es TCP, pero se puede escoger otro más de la lista desplegable.

- Fuente: Permite especificar la fuente origen de tráfico para esta regla. Regularmente se escoge 'ninguna' lo que significa que permite a todos los equipos conectarse a los servicios.
- ✓ Destino: Específica la dirección IP de destino original del tráfico, normalmente será la dirección WAN.
- ✓ Intervalo de puertos destino: Especifica el puerto destino original del tráfico, además se puede especificar por rango de puertos para esta regla. Estos puertos se sitúan para transmitir datos hacia el exterior y que puedan recibir datos por estos mismos puertos.
- ✓ Redirigir IP de destino: Se define la dirección IP interna a la que se trasmitirá el tráfico, es decir, si una dirección externa pregunta por la dirección de WAN la contestara la dirección IP a la que se re direccione.
- ✓ Redirigir puerto de destino: Se especifica el puerto en el servidor con la dirección IP ingresada anteriormente, este puerto regularme suele ser idéntico al puerto destino que se describió más arriba.
- ✓ Descripción: Si se decide, se realiza una descripción de referencia a la regla NAT realizada.
- Reflexión NAT: Permite la reflexión NAT para ser activado o desactivado delante de cada puerto.
- ✓ Filtro de asociación por regla: En caso de la existencia de reglas previas, se pueden asociar entre estas.

Estas características de reglas NAT son posiblemente un poco más complejas que las de filtrado. Al terminar de crear una regla de '**Port Forward'** el sistema pfSense ofrece la opción de crear la regla automáticamente, es decir, la regla NAT aparecerá establecida como si fuera una "regla de filtrado" más, pero en realidad, esta regla es derivada de la regla de direccionamiento NAT, como se observa en la figura 28.

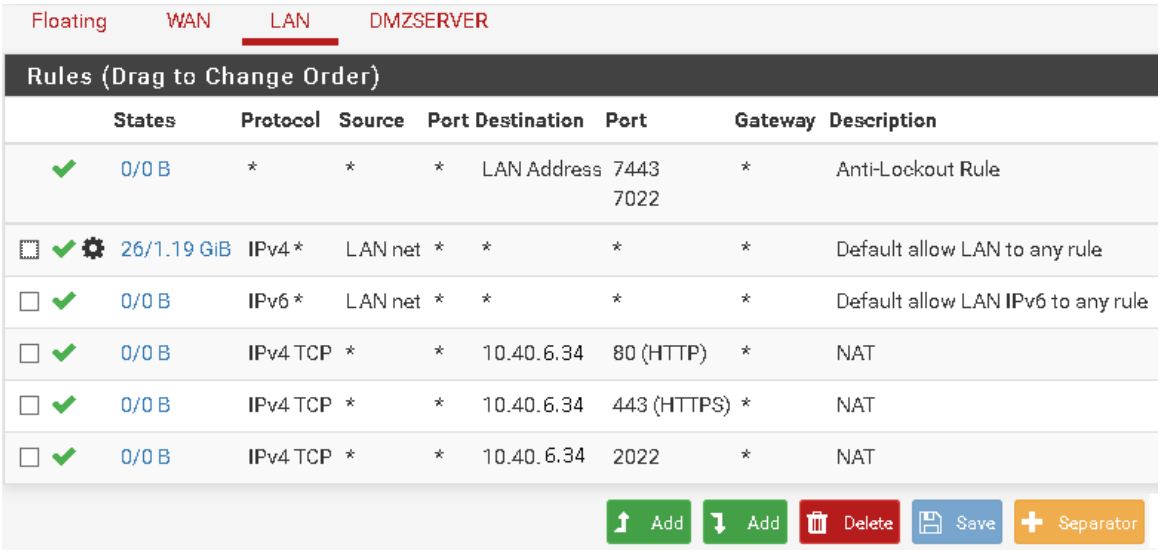


Figura 28 Reglas NAT definidas.

De esta manera se formó un conjunto de reglas de filtrado y reglas NAT, lo cual ofrece una seguridad de amplio nivel. En este conjunto de reglas de entrada del cortafuego, se aprecian todos los servicios que están disponibles en la red por medio de ambas reglas.

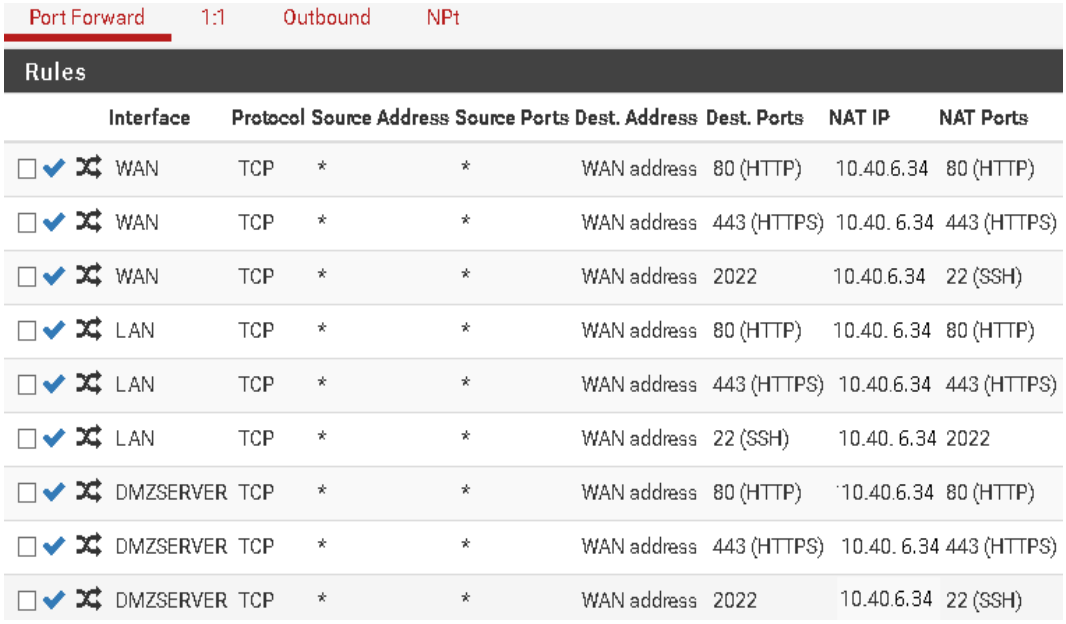


Figura 29 Conjunto de reglas NAT y de filtrado

En las reglas NAT (figura 29) sólo se podrán observar de manera global las reglas que definieron toda la red, los puertos destino y puertos origen. En estas reglas la mayoría realiza una redirección hacia los servicios que se encuentran en el servidor de LACyTES, sólo se podrán acceder a dichos servicios por medio de puertos establecidos, como SSH, HTTP y HTTPS. Por ejemplo, si algún usuario o cliente externo pretende ingresar por medio del puerto FTP o TELNET no tendrá respuesta alguna, ya que estos puertos no se encuentran habilitados.

Este es un excelente método para realizar el direccionamiento hacia puertos exclusivos y evitar que ingresen paquetes no deseados con protocolos no permitidos. Todo esto permite mantener de manera más segura los servicios puestos para disposición de usuario y se proteja la LAN que como se mencionó, es la red donde se encuentran asignados la mayoría de los equipos de los investigadores y estudiantes de proyecto, por lo tanto, proteger su información, trabajos, proyectos y datos es prioridad.

3.5 Pruebas de conectividad, funcionamiento y disponibilidad de la red LACyTES.

El diseño elaborado de red es importante para el LACyTES y como se vio en el apartado 3.3.1, se cuentan con dos redes distintas en el laboratorio que conforman la red interna; la subred LAN donde se encuentran todos los equipos de investigación, desarrollo y sistemas dedicados, y la subred DMZ donde se encuentran los servicios ofrecidos (portal web, procesamiento de computo matemático y aplicación móvil).

En la LAN, la mayoría de los equipos de cómputo cuentan con el sistema operativo Windows y en la DMZ los equipos en su mayoría cuentan con sistema operativo en base UNIX/Linux. Dentro de cada subred los equipos se pueden comunicar entre sí ya que pertenecen a la misma red, pero para descartar dudas se realizan pruebas de conectividad mediante el comando *Ping*. El *ping* opera mediante el envío

de paquetes por medio del protocolo ICMP al host de destino y la espera de la respuesta logra registrar el tiempo de ida y vuelta, además de la pérdida de algunos paquetes.

Antes de realizar las pruebas, primero se debió obtener la configuración TCP/IP de un equipo de cada red. En el sistema operativo Windows primero se ingresó a 'Símbolo del sistema'. Una vez en esta interfaz, se ingresó el comando **ipconfig**. Para los sistemas Unix/Linux basta con acceder a la interfaz de consola del equipo y realiza el comando **ifconfig**.

En la figura 30 se observan los resultados obtenidos por estos comandos, se pueden apreciar los datos de la configuración de cada equipo, como la conexión a la red donde se encuentra cada uno, dirección IP, máscara de subred y su puerta de enlace (Gateway).

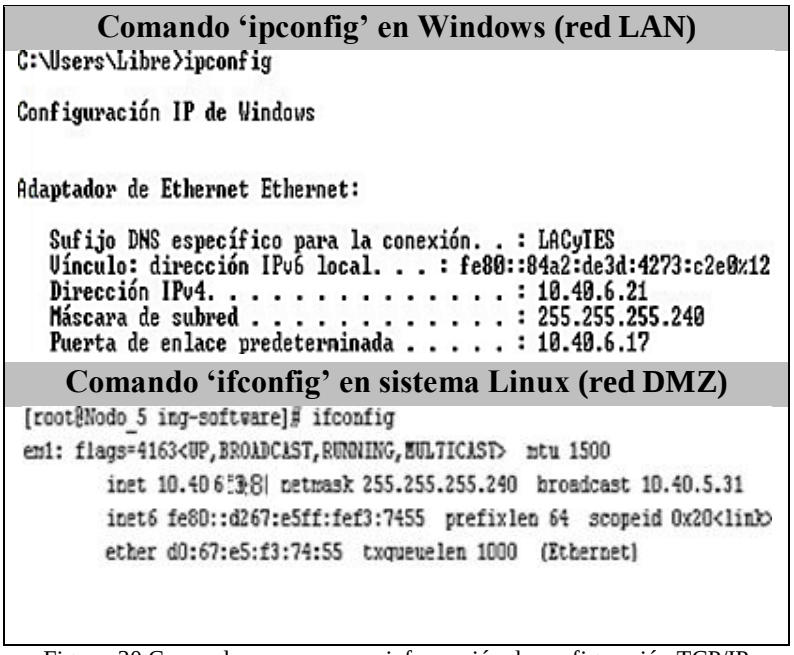


Figura 30 Comandos para conocer información de configuración TCP/IP

3.5.1 Pruebas de conectividad interna en cada red

Para la prueba de conectividad, funcionamiento y disponibilidad dentro de cada red se realizó desde los equipos que contienen estas IP vistas en la figura 30. Como se observa en la siguiente figura 31 en cada equipo se realiza con el comando ping seguido de la dirección IP de otro equipo perteneciente a la misma red.

Como se mencionó anteriormente este comando indica el tiempo que tardan los paquetes de datos en ir y regresar por medio de la red desde un equipo informático a otro.

Comando 'ping' + 'IP' en Windows

```
C:\Users\Libre>ping 10.40.6.22

Haciendo ping a 10.40.6.22 con 32 bytes de datos:
Respuesta desde 10.40.6.22: bytes=32 tiempo<1m TTL=128
Respuesta desde 10.40.6.22: bytes=32 tiempo<1m TTL=128
Respuesta desde 10.40.6.22: bytes=32 tiempo<1m TTL=128
Respuesta desde 10.40.6.22: bytes=32 tiempo<1m TTL=128

Estadísticas de ping para 10.40.6.22:
    Paquetes: enviados = 4, recibidos = 4, perdidos = 0
    (0% perdidos),
Tiempos aproximados de ida y vuelta en milisegundos:
    Mínimo = 0ms, Máximo = 0ms, Media = 0ms
```

Comando 'ping' + 'IP' en sistema Linux

```
[root@Nodo_5 ing-software]# ping 10.40.6.40
PING 10.40.6.40 (10.40.6.40) 56(84) bytes of data.
64 bytes from 10.40.6.38: icmp_seq=1 ttl=128 time=0.552 ms
64 bytes from 10.40.6.38: icmp_seq=2 ttl=128 time=0.529 ms
64 bytes from 10.40.6.38: icmp_seq=3 ttl=128 time=0.501 ms
64 bytes from 10.40.6.38: icmp_seq=4 ttl=128 time=0.465 ms
64 bytes from 10.40.6.38: icmp_seq=5 ttl=128 time=0.538 ms
64 bytes from 10.40.6.38: icmp_seq=6 ttl=128 time=0.488 ms
^C
--- 10.40.5.18 ping statistics ---
6 packets transmitted, 6 received, 0% packet loss, time 4999ms
```

Figura 31 Respuestas de conectividad en red interna.

Esta prueba logra verificar la conectividad de extremo a extremo, es decir, corrobora que dentro de cada subred elaborada, los equipos se comunican entre sí. De no ser así se verifica el ponchado de los cables UTP, las rosetas UTP, asi como cualquier característica que pueda interferir en la conectividad.

3.5.2 Prueba de conectividad de la LAN a DMZ.

Para demostrar la conectividad de la subred LAN hacia la subred DMZ, esta se realiza de manera análoga pero desde la LAN; esto es, se mandó un ping de un equipo perteneciente a la red LAN hacia uno que se encuentra en la subred DMZ. Esta vez el equipo de la red LAN es diferente, esto se hace con la finalidad de comprobar que cualquier equipo puede realizar esta prueba. Con la misma interfaz de 'Símbolo de sistema' de Windows se realiza el comando ping.

Como se observa en la figura 32 la IP pertenece a las direcciones de la red LAN, y el ping se realiza a un servidor ubicado en la subred DMZ, Cuando se ejecuta el comando se observa que se obtiene una respuesta del servidor con lo cual se comprueba que la LAN puede realizar comunicación con la subred DMZ y que ambas subredes logran comunicarse en este sentido (LAN hacia DMZ) por medio del cortafuego donde se establecieron las reglas necesarias.

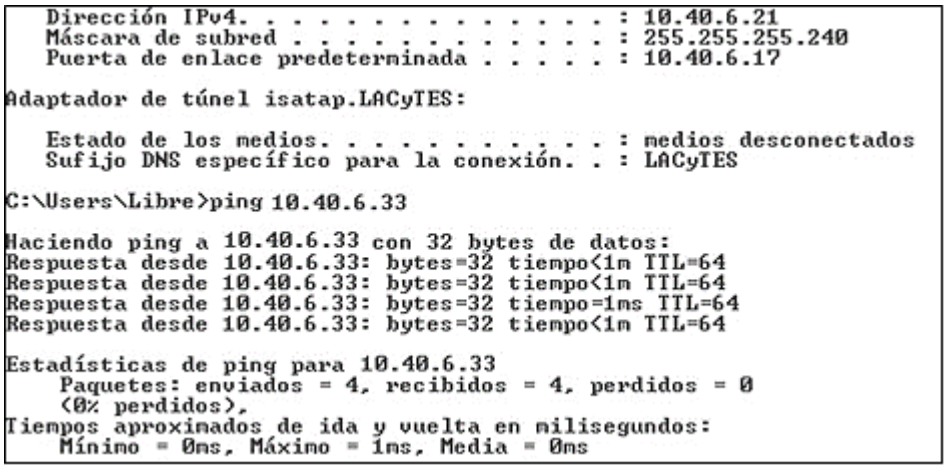


Figura 32 Respuesta de conectividad LAN a DMZ.

3.5.3 Prueba de conectividad de la DMZ a LAN

La siguiente prueba se realiza desde la subred DMZ hacia la subred LAN, en esta prueba se comprueba que no hay conexión con la red LAN, pues esta red debe ser totalmente ajena a las demás redes, esto sólo se puede cambiar si se establecen reglas en el cortafuego que permitan dicha conexión.

Para determinar la conectividad se envía un ping desde esta red hacia la red LAN. Para realizar el ping solo es necesario el comando y la dirección hacia un equipo perteneciente a la red LAN donde se desea verificar conectividad.

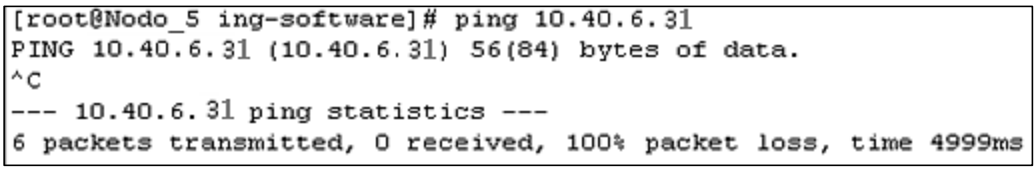


Figura 33 Respuesta nula de conectividad de red DMZ a LAN.

Como se observa en la figura 33 desde la red DMZ hacia la red LAN no hay conectividad y esto es correctamente lo que se configuro en el cortafuego.

3.5.4 Prueba de conectividad y redireccionamiento a servicios.

La última prueba se realizó para verificar el direccionamiento NAT que se implementó en el cortafuego, es decir por medio de un navegador y el comando ping se verificó que la dirección de la red WAN se direccione correctamente al servidor que contiene el portal web del laboratorio.

Para esta prueba se solicitó el uso de un equipo que se encuentra ubicada dentro de la red MAN de la UACM en el área de profesores, primero se obtuvo su configuración de red para demostrar que el equipo no se encuentra dentro de la red diseñada. El resultado de la configuración TCP/IP se puede verificar en la parte superior figura 35.

Ipconfig de red externa

```
C:\Documents and Settings\profesores>ipconfig

Configuración IP de Windows


Adaptador Ethernet Conexión de área local          :

        Sufijo de conexión específica DNS :
        Dirección IP. . . . . . . . . . . . : 172.17.120.179
        Máscara de subred . . . . . . . . . : 255.255.255.240
        Puerta de enlace predeterminada    : 172.17.120.190
```

Ping de red externa a red WAN

```
Microsoft Windows XP [Versión 5.1.2600]
(C) Copyright 1985-2001 Microsoft Corp.

C:\Documents and Settings\profesores>ping 172.17.120.146

Haciendo ping a 172.17.120.146 con 32 bytes de datos:

Respuesta desde 172.17.120.146: bytes=32 tiempo=1ms TTL=63
Respuesta desde 172.17.120.146: bytes=32 tiempo=1ms TTL=63
Respuesta desde 172.17.120.146: bytes=32 tiempo=1ms TTL=63
Respuesta desde 172.17.120.146: bytes=32 tiempo=1ms TTL=63

Estadísticas de ping para 172.17.120.146:
    Paquetes: enviados = 4, recibidos = 4, perdidos = 0
    (0% perdidos),
Tiempos aproximados de ida y vuelta en milisegundos:
    Mínimo = 1ms, Máximo = 1ms, Media = 1ms
```

Figura 34 Ping de equipo externo a red del laboratorio.

En la parte inferior de la figura 34 se observa claramente que la dirección WAN de la red interna del laboratorio responde al ping, pero para verificar el direccionamiento NAT se usó el navegador web.

Desde el equipo prestado se ingresa a un navegador web y se escribe la dirección WAN de la red diseñada.

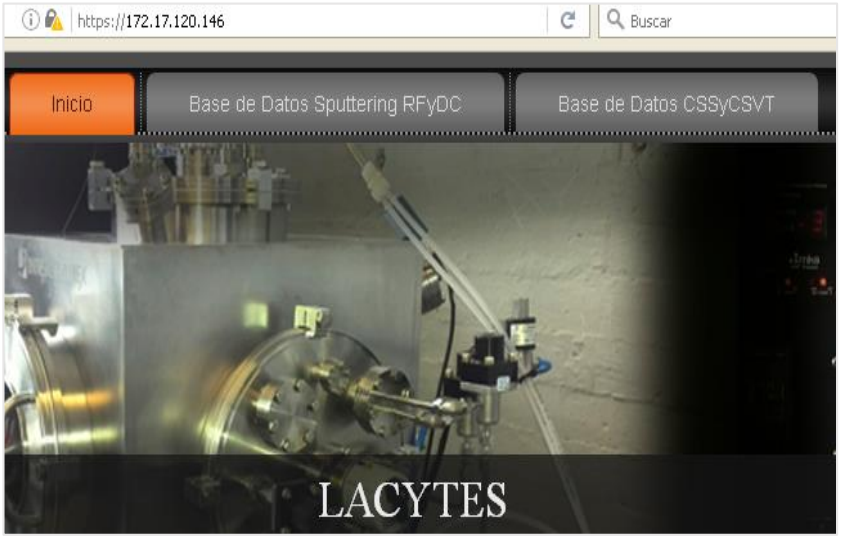


Figura 35 Respuesta del servidor por medio de reglas NAT.

Como resultado se obtiene la figura 35 donde se observa que la dirección WAN redirecciona al portal web, el cual se encuentran en el servidor host con una dirección local de la red DMZ (10.40.6.34), es decir, que las reglas NAT que se definieron para hacer redireccionamiento a los puertos 53, 80, 22 y 443, logran su objetivo. Cualquier petición hacia otro puerto es negada pues no se definió en la configuración

y pfSense cierra el resto los puertos por defecto, esto se comprueba mediante el escaneo de puertos a través de Nessus, donde se comprueba que los puertos configurados para acceso son visibles desde cualquier parte de la UACM, ver figura 36.

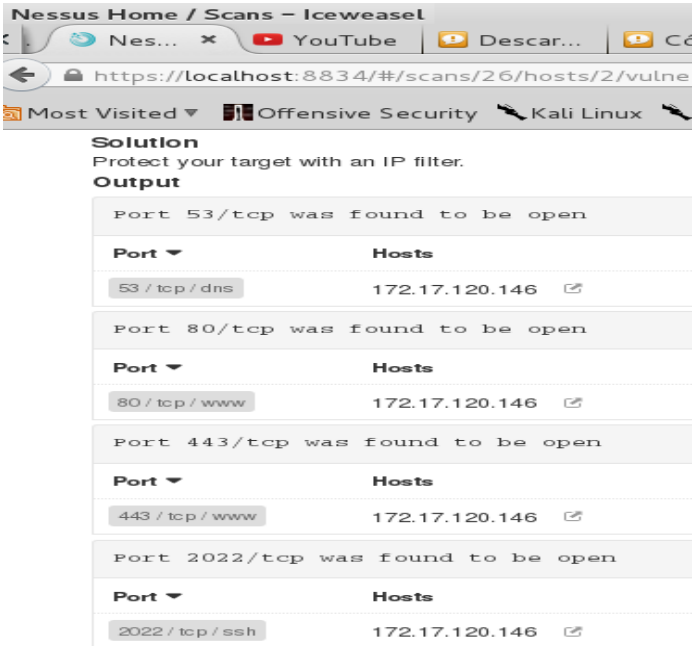


Figura 36 Escaneo de puertos con Nessus

3.6 Resultados de red LACyTES

Finalmente la infraestructura se implantó con éxito en el laboratorio, por lo cual los objetivos planteados en un inicio respecto a la red de datos se cumplieron. Así mismo el análisis elaborado permitió conocer las necesidades del laboratorio y resolver los problemas de seguridad que existían con la interconexión a la red de la UACM. Estos problemas pudieron ser remediados mediante el uso de determinadas técnicas y controles.

La topología de red implantada brinda la posibilidad de interpretar fácilmente el diseño de la red, la seguridad de esta es administrada de forma centralizada, permitiendo gestionarla y configurarla de una forma sencilla. Sin embargo, esta configuración tiene un riesgo de seguridad puesto que si el servidor de seguridad se viera comprometido gravemente, se llegaría a afectar por algún ataque externo o interno, la red podría no funcionar y el laboratorio tendría problemas de conectividad por lo cual es necesario realizar respaldos del sistema. Una alternativa para proteger la red sería la implementación de un cortafuego en paralelo.

El direccionamiento que se estableció es estático y solo se permite una conexión por DHCP, logrando así evitar cualquier otra intrusión de un sistema ajeno, de acuerdo a las pruebas realizadas la reglas de seguridad en el cortafuego funcionan correctamente de agregarse más, podrían generar algunos cuellos de botella y la red funcionaría de forma lenta, pues cada regla debería revisar que las condiciones de esta se cumplen y permitir el paso hacia la siguiente regla.

Las pruebas de conectividad que se realizaron, logran verificar que los sistemas de cómputo se encuentran en funcionamiento y que éstos cuentan con comunicación interna y externa (salida a internet) sin problema alguno. Las reglas NAT se encuentran funcionando en su totalidad, logrando direccionar el acceso externo hacia el servidor web lo cual demuestra que su configuración es correcta, en caso de que se deban agregar más puertos para tener acceso a la información (FTP, Tenet, POP2, smtp) se pueden activar, pero se tendría un sistemas más susceptible a ataques externos.

El comandos ping puesto en ejecución para la realización de pruebas de conectividad pueden no funcionar si se presentan fallas en la topología, el cableado UTP o por configuración del dispositivo *switch.*

La configuración del punto de acceso es utilizable y no presenta problema alguno, se limitaron las conexiones a un total de 20 usuarios, puesto que el laboratorio cuenta con este número aproximado de usuarios y la mayoría cuenta con un sólo dispositivo móvil (teléfonos celulares, tabletas electrónicas o laptops) los cuales se conectan a la red WLAN. De llegar a desear agregar más usuarios es necesario ingresar a la configuración del WAP para agregar los dispositivos nuevos.

La tableta electrónica donde se ejecuta la aplicación se encuentra conectada a esta red y se recomienda que esta conexión siempre deba ser así, debido a que es el único medio de comunicación por el cual se pueden enviar los datos al servidor. Los sistemas dedicados se interconectan al punto de acceso de manera eventual, con el objetivo de solo enviar o almacenar datos de los sistemas de caracterización para su análisis.

Gracias al conjunto de estos servicios los dispositivos de la red interna se encuentran salvaguardados. Con esto se logró obtener una red de datos amoldable a las necesidades del laboratorio y ofrece la posibilidad de adaptarse a sus futuras necesidades del laboratorio.

Capítulo 4

Diseño y desarrollo de aplicación hibrida

En el presente capítulo se presenta la segunda parte de este trabajo, donde se toma en cuenta el sistema de captura de datos mediante el portal web de LACyTES, con la finalidad de determinar las diferencias que existen entre éste y el desarrollo de una aplicación. Además se describe como se desarrolló la aplicación, que tiene como objetivo la captura datos de los sistemas experimentales CSVT-IR y CSS-IR. Se muestra la maquetación para el desarrollo (sketching), cómo se estableció la conexión con la base de datos, la seguridad establecida internamente, el funcionamiento de la aplicación mediante un emulador y posteriormente en un dispositivo móvil.

4.1 Diseño de aplicación

Como parte del proceso metodológico de recolección de información que permita llegar a uno o más de los objetivos expuestos en el presente trabajo, se realizó un análisis de diseño y contenido que tendrá la aplicación móvil.

En el apartado **1.1.1** se explicó la forma en que se realizaba la captura de datos, esto se realizaba mediante la escritura de los datos en "hojas de control" donde los datos de los sistemas visto a través de sus pantallas se escribían con pluma sobre estas y posteriormente sólo un 22% de los datos eran transcritos a un portal web que posee el laboratorio. Este portal web ofrece únicamente la posibilidad de almacenar y consultar los datos acerca de la manufacturación de módulos fotovoltaicos de dicho porcentaje, sin embargo; el desarrollo de la aplicación móvil para este trabajo debe ser capaz de capturar el 100% de los datos, consultarlos y procesar algunos de los datos para elaborar graficas dinámicas.

El diseño de la aplicación está basado en estructurar los formularios que se encuentran en el portal web de una manera fácil, que contengan un aspecto visual atractivo y sencillo para los investigadores. Como primer paso, se identifican las necesidades de los usuarios para qué el desarrollo de la aplicación logre o ayude a resolverlas. Estas necesidades se han identificado preguntando directamente a los usuarios de los sistemas CSVT-IR y CSS-IR y se enlistan de la siguiente manera:

- Ahorrar tiempo al transcribir los datos al portal web.
- Tener una consulta rápida de los datos.
- No repetir una muestra con el mismo nombre.
- Asegurar la información en una base de datos dentro de un servidor host.
- Poder elaborar graficas de los datos y realizar un análisis sobre estos.
- Movilidad con el dispositivo que cuente con la aplicación.
- Portabilidad, que la aplicación también funciones en un navegador web.

Como segundo paso, se elaboró el prototipo a base de dibujos (sketching) donde se representó la estructura visual de la aplicación. Ésta suele carecer de aspectos visuales como, estilo tipográfico, colores e imágenes, ya que su propósito fue servir como apoyo para discutir el contenido, requerimientos, funcionalidad y comportamiento de los eventos de la aplicación [23].

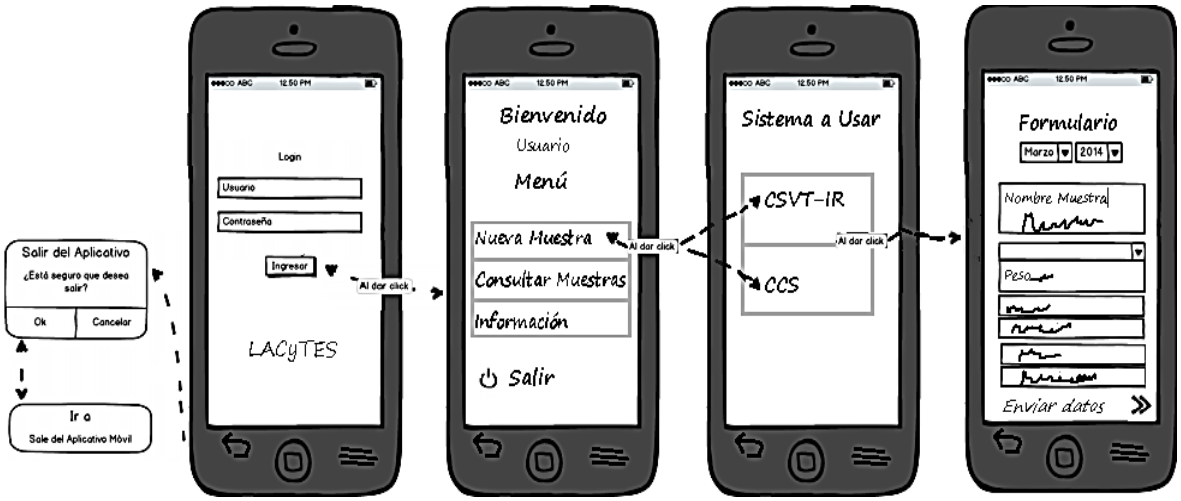


Figura 37 Vistas que muestran algunas funciones de la aplicación.

En la figura 37 se muestra, la simulación de una autenticación correcta de un usuario al cual se le muestra una pantalla de bienvenida y un menú con distintos escenarios. Cuando se selecciona la opción "**Nueva Muestra**" se despliega una pantalla con dos opciones, sistema CSVT-IR o CSS-IR, y al escoger una de éstas, se visualizará el respectivo formulario para la captura de los datos experimentales de la muestra a realizar y al finalizar los datos serán almacenados en formato digital en el servidor host y podrán ser consultados.

Figura 38 Visualización de Grafica

Como se muestra en la figura 38 en el menú principal se encontrará una opción que permite realizar las consultas acerca de las muestras realizadas y almacenadas en el servidor host. Al seleccionar esta opción se visualizara una pequeña caja de texto donde al colocar sólo el nombre de la muestra que se desea

consultar se mostraran los datos de esta y los datos de temperaturas con respecto al tiempo se podrán representar de manera gráfica. Finalmente el menú principal se encuentra una opción más, esta aporta información acerca de laboratorio como se muestra en la figura 39.

Figura 39 Skecth de opción información del laboratorio

Como se puede observar en las figuras 37, 38 y 39 el prototipo de funciones para una aplicación es bastante útil para el desarrollo de ésta. Para este trabajo ayudo a realizar un buen diseño gráfico, ver las posibles funciones a realizar dentro de ésta, pero sobre todo el de transmitir la idea del concepto de la aplicación a los usuarios.

Cabe señalar que durante el desarrollo de la aplicación el prototipo tuvo algunas modificaciones éstas se realizaron con facilidad y rapidez, se debieron a que surgieron cambios como funcionalidades de algunos botones, algunas casillas tipo radio, tablas para colocar la información, etc. Estos cambios hicieron a la aplicación más funcional y más ligera, es decir, no requiere demasiada memoria RAM del dispositivo. Se priorizó el contenido dentro de la misma y esto redujo el tiempo de desarrollo.

4.2 Diagrama de comunicación entre servidor y aplicación.

El diagrama de comunicación de la petición realizada por la aplicación, hacia el servidor web usando el protocolo HTTP corresponde con la figura 40, donde el usuario por medio el dispositivo móvil realiza una petición al servidor a través de la aplicación. Para esto primero se procesa la petición, el framework la encamina con dirección al controlador correspondiente, éste realiza un llamado al recurso o acción por medio del protocolo HTTPS hacia el modelo que se encuentra instanciado en el servidor host. Después el controlador procesa esta acción e instancia los objetos necesarios a través del modelo. Una vez realizada esta acción, envía la información a la vista para poder mostrarla al usuario.

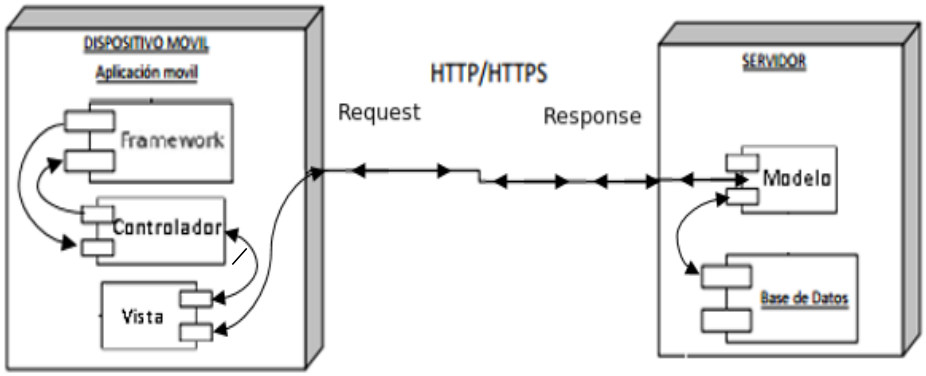


Figura 40 Diagrama de flujo de una petición desde la aplicación hacia el servidor.

El modelo realiza la consulta a la base de datos para lograr instanciar el objeto solicitado, si no es el caso instancia uno nuevo. El modelo envía los datos al el controlador una vez más por medio del protocolo HTTPS, este los envía también a la vista ya generada y se muestra al usuario como respuesta.

Con el entendimiento del diagrama y gracias al desarrollo de frameworks hoy en día las tecnologías web permiten diseñar su estructura visual y lógica para el desarrollo de aplicaciones móviles hibridas. Se puede determinar que una aplicación se organiza de una forma simple, de modo que la navegación a través de esta se hace intuitiva y esto facilita la interacción del usuario. Una vez que se termina el desarrollo de la aplicación mediante tecnologías web, se compila para así obtener una aplicación hibrida, más adelante se detalla este proceso.

4.3 Ampliación de la base de datos de los sistemas CSVT-IR y CSS

El almacenamiento de los datos experimentales se hace en una base de datos. Esta cuenta con la tabla de nombre ‘tabladatos2’ ya diseñada para el almacenamiento del 22% los datos. Sin embargo, dentro de esta no se registran algunos datos importantes cuando se realiza un nuevo módulo.

Se estudió en la literatura acerca de cómo se realiza una buena estructura de base de datos, se recomendó modificarla para el almacenamiento de los datos, administración y gestión de éstos, pero por parte del director del trabajo recepcional recomendó seguir la misma estructura y sólo agregar los datos que hacían falta, para que por medio de la aplicación se capturaran y almacenaran.

Por consiguiente se solicitó al administrador del servidor el acceso para analizar la estructura de la tabla, se agregaron y adaptaron nuevas columnas dentro de esta, con el propósito de que las funciones establecidas en la aplicación almacenen los nuevos datos que se muestran en la figura 41.

Figura 41 Datos agregados en la base de datos

El envío de estos datos se hace mediante funciones desarrolladas en JavaScript, es decir, los datos son transportados desde la aplicación hacia el servidor host, en el cual el lenguaje PHP (lógica del servidor) permite el acceso a la base de datos y a la tabla.

4.4 Elaboración de la estructura para la aplicación

A continuación se explica de forma detallada cómo se llevó a cabo el desarrollo de cada parte de la aplicación y dónde han sido utilizadas las tecnologías web, por ejemplo CSS3 se ocupara para brindar una forma visual atractiva a los usuarios de la aplicación, HTML5 se ocupara para elaborar el menú y los formularios, los cuales estarán elaboraros de una manera que el usuario los entienda, JavaScript se emplea para la comunicación entre el servidor y la aplicación.

Como primera parte del desarrollo de la aplicación fue necesario instalar Node.js, ya que como se mencionó anteriormente este software permite instalar y ejecutar Phonegap a través de una consola de comando, permitiendo construir de una manera fácil y escalable la base del proyecto.

Una vez instalado Node.js se procedió a abrir el programa. Dentro de éste se realizó la instalación del framework phonegap mediante el comando.

```
# npm install -g phonegap@latest
```

Al ejecutarse este comando se comienza la descarga el módulo Node de Phonegap y se genera la base para crear el proyecto de la aplicación, como se observa en la figura 42.

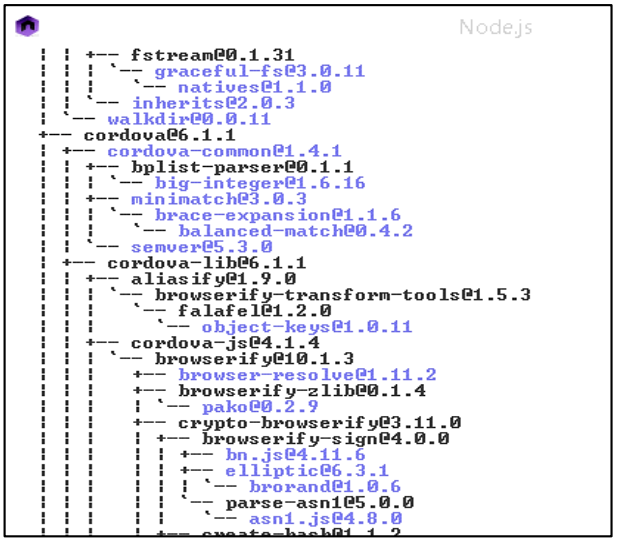


Figura 42 Proceso de Instalación de Phonegap en su versión 6.3.1.

Al finalizar la instalación se procedió a crear la carpeta donde se crea la aplicación mediante el siguiente comando. `create <aplicacion> <identificador> <nombre a desplegar>`

Este comando debe contener el nombre de la aplicación, un identificador y un nombre a desplegar para la aplicación. El identificador representa a una organización, por ejemplo org.lacytes.uacm y el nombre a desplegar es aquel que recibe el icono de instalación de la aplicación una vez que ya se tiene en el dispositivo, por ejemplo App-SiSu. La primera presentación de la aplicación se crea con una plantilla que contiene un simple "HelloWord" con un HTML y JavaScript que espera hasta que el dispositivo se encuentre listo para usar las funciones nativas. El comando que se utilizó para la creación del proyecto es:

```
# phonegap create app-sisu edu.lacytes.uacm SiSu
```

La figura 43 muestra la creación de la estructura del proyecto 'app-sisu' en el sistema de archivos, como se observa en la parte inferior de la misma figura se crea el proyecto con una planilla de 'hello-world.'

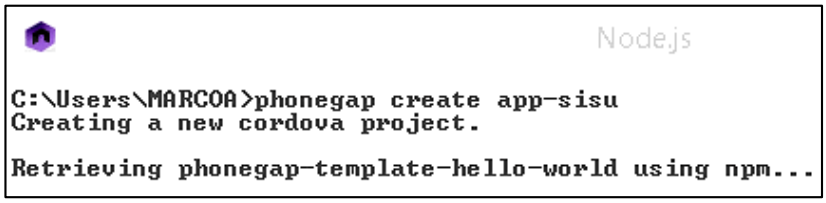


Figura 43 Proyecto de aplicación app-sisu generado.

Hasta este momento solo se generó el proyecto de la aplicación, ya se cuentan con la base y los archivos necesarios para el desarrollo óptimo del trabajo.

Para poder visualizar la estructura de la aplicación se ingresó a la carpeta del proyecto mediante el uso de los comandos siguientes:

```
# cd app-sisu
# dir
```

```
C:\Users\MARCOA>cd app-sisu

C:\Users\MARCOA\app-sisu>dir
 El volumen de la unidad C no tiene etiqueta.
 El número de serie del volumen es: D646-916A

 Directorio de C:\Users\MARCOA\app-sisu

14/09/16  14:51    <DIR>          .
14/09/16  14:51    <DIR>          ..
14/09/16  14:51             1,229 .bithoundrc
14/09/16  14:51                16 .npmignore
14/09/16  14:51             9,087 config.xml
14/09/16  14:51               733 CONTRIBUTING.md
14/09/16  14:51               721 COPYRIGHT
14/09/16  14:51    <DIR>          hooks
14/09/16  14:51            11,362 LICENSE
14/09/16  14:51               365 NOTICE
14/09/16  14:51               583 package.json
14/09/16  14:51    <DIR>          platforms
14/09/16  14:51    <DIR>          plugins
14/09/16  14:51             2,849 README.md
14/09/16  14:51    <DIR>          www
```

Figura 44 Estructura del proyecto en el sistema de archivos.

Como se observa en la figura 44 la estructura de la aplicación está compuesta por distintas carpetas (*hooks, platforms, plugins* y *www*) y el archivo núcleo (*config.xml*).

- `Hooks`: Es utilizada para el manejo del ciclo de vida de una aplicación, es decir, rutinas que se quieran incluir antes o después de algún evento, como por ejemplo el evento *preview, post view, create plugin*, o script personalizados, la recomendación es que estos sean hechos en Node.js para permitir la compatibilidad con diversas plataformas.
- `Platforms`: Esta carpeta agrega las plataformas para las cuales se desean desarrollar aplicaciones.
- `Plugins`: Son archivos o documentos XML y existen diversos ya desarrollados por usuarios permitiendo que sean gratis en la página oficial de Phonegap/Cordova. Se permite hacer uso de características del sistema siempre y cuando se programen de acuerdo a las especificaciones de la guía de desarrollo de plugins en Cordova.
- `www`: Carpeta de recursos web de servidores apache tales como HTML, JavaScript, CSS3, etc. En esta carpeta se almacenan los recursos web que serán servidos por la aplicación.

Las carpetas que se usaron para el desarrollo de este trabajo son las 3 últimas descritas. Por lo cual es necesario agregar la plataforma Android sobre la cual se trabajara, mediante el comando platform add. <nombre de plataforma>

phonegap platform add android

```
C:\Users\MARCOA\app-sisu>phonegap platform add android
Adding android project...
Creating Cordova project for the Android platform:

        Path: platforms\android
        Package: com.phonegap.helloworld

        Name: Hello_World
        Activity: MainActivity
        Android target: android-23

Android project created with cordova-android@5.1.1

Discovered plugin "cordova-plugin-battery-status" in config.xml. Installing to t
he project

Fetching plugin "cordova-plugin-battery-status@~1.1.1" via npm

Installing "cordova-plugin-battery-status" for android
```

Figura 45 Agregación de plataforma Android y plugin del mismo sistema

Como se muestra en la figura 45 el comando *platform* además agrega automáticamente los plugins que permiten acceder a las diferentes características del sistema de Android. Así mismo se escogió la plataforma de Android debido a la facilidad que ofrece al momento de desarrollar aplicaciones y realizar pruebas directamente en dispositivos reales, sin la necesidad de descargarlas de un mercado de aplicaciones.

El archivo config.xml se crea automáticamente al crear el proyecto de la aplicación. Este archivo XML es independiente de la plataforma y se arregla basado en la especificación del W3C.

Los elementos que se observan en el archivo config.xml (figura 46) se admiten en todas las plataformas que soporten Phonegap/Cordova (Android, iOS, Windows Phone, Blackberry, Tizen y Firefox OS).

```
<widget id="com.example.hello" version="0.0.1">
    <name>HelloWorld</name>
    <description>
        A sample Apache Cordova application that responds to the deviceready event.
    </description>
    <author email="dev@callback.apache.org" href="http://cordova.io">
        Apache Cordova Team
    </author>
    <content src="index.html" />
    <access origin="*" />
</widget>
```

Figura 46 Configuración global del núcleo

- El elemento *<name>* especifica nombre formal de la aplicación, como aparece en la pantalla principal del dispositivo y dentro de la tienda app interfaces. Para este trabajo se nombró App-SiSi (Aplicación de Sistemas de Sublimación).
- Los elementos *<description>* y *<author>* especifican metadatos e información de contacto que puede aparecer en anuncios de la tienda app.
- Opcional *<content>* elemento define la página de inicio de la aplicación en el directorio web de alto nivel de activos. El valor predeterminado es `index.html`, que habitualmente aparece en el directorio de nivel superior *www* de un proyecto.
- Elementos *<access>* definen el conjunto de dominios externos que puede comunicarse con la aplicación. El valor predeterminado que se muestra en la figura 46 permite acceder a cualquier servidor, mientras más se limite el acceso, la aplicación se puede considerar con mayor seguridad.

El archivo XML se encuentra en el directorio *app-sisu/config.xml*, debido a la creación del proyecto, pero algunos usuarios en ocasiones colocan éste archivo dentro de la carpeta *app-sisu/www/config.xml.*

Dentro de la carpeta *www* se procedió a realizar la aplicación mediante etiquetas HTML5 que dará el entorno de vista, realización de eventos y ejecución de controladores de la aplicación.

4.5 Página de Inicio

En la página `index.html` se muestra un *Splash screen* el cual es una ventana temporal donde se podrá apreciar el logotipo que representa al laboratorio, detrás de éste se aprecia el fondo que es un arreglo de celdas fotovoltaicas, no permanece constante durante toda la aplicación ya que se perdería de vista algunos de los objetos de las siguientes pantallas dentro de esta. Como se muestra en la figura 47 este fondo fue editado con Photoshop para resaltar el tema de investigación de energías renovables.

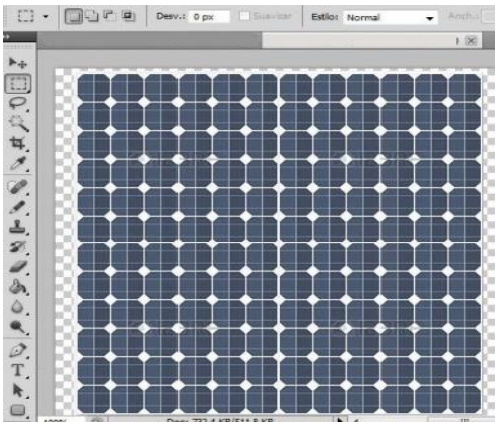


Figura 47 Fondo de Inicio de aplicación creado en Photoshop

Dentro de esta primera página se definieron elementos CSS3 que permitieron personalizar el color, el tamaño y los márgenes del texto, incluso se ha añadido sombra para que se observe claramente el texto. Para la imagen de LACyTES también se usó código CSS3 logrando que ésta sea reescalable por medio de un área determinada.

Cuando se accede a la aplicación, instantáneamente se ejecuta el plugin `cordova-plugin-network-information` de JavaScript, éste verifica la conexión a la red inalámbrica de LACyTES. Si existe conexión, la aplicación ejecuta un script *timeout* que da paso a la pantalla de autentificación de usuarios. En caso de no tener conexión la aplicación envía un mensaje de alerta.

El resultado del conjunto del diseño del fondo con elementos CSS3, JavaScript y HTML5 se aprecia claramente en la figura 48.

Figura 48 Inicio de aplicación y verificación de conexión Wifi

4.6 Selección de temas para la interface

Para la página de autenticación de usuarios, menú, formularios, consulta de muestras e información se cuenta con un diseño muy similar, para éstas se hizo uso de las bibliotecas de jQuery, jQuery Mobile y diseño de ThemeRoller. Para poder obtener el diseño se ingresó a la página oficial de ThemeRoller donde se logró crear de una manera sencilla tres distintos *swatch* para la aplicación, se consiguió modificar características como sombreado, colores en botones, enlaces, figuras, texto, etc. Una vez obtenido el diseño deseado ThemeRoller permite descargar el código fuente de los *swatch* y agregarlas a nuestro código CCS3 de la aplicación.

El resultado de estos diseños se aprecia en la autentificación de usuarios, menú principal, formularios, botones y pantallas del resto de la interfaz.

4.7 Autentificación de usuarios

Previo a iniciar con el desarrollo de la página de autentificación es necesario recordar que los usuarios ya cuentan con un registro en la plataforma web del laboratorio, por lo cual se solicitó trabajar con la base de datos donde se encuentren estas credenciales y poder ingresar a la aplicación por medio de éstas y evitar un doble registro.

Como se observa en la figura 49 el sistema de inicio de sesión consta de un diseño muy similar a cualquier aplicación que permite el acceso a correo electrónico, redes sociales, etc. Cuenta con un botón de registro que al presionar sobre éste permitirá al usuario ingresar al portal web de LACyTES, lo hace gracias al plugin **inappbrowser** que fue instalado dentro de la aplicación. Permite abrir una ventana de navegador web sobre la aplicación, sin la necesidad de salir de ésta para poder realizar el registro necesario.

Figura 49 Inicio de sesión

A pesar de que el dispositivo móvil se encuentra dentro de la subred LAN del laboratorio, la petición se hace mediante la dirección IP de la red WAN del cortafuego, esto con la finalidad de que se ocupe el

enrutamiento a través de la red diseñada. Lo cual tiene como ventaja que si algún usuario del laboratorio desea consultar datos experimentales sólo se podrá hacer dentro de la red de la UACM, sin embargo; los datos solo se podrán ingresar si el dispositivo que contiene la aplicación se encuentra conectado a la red inalámbrica del laboratorio, de esta manera se asegura que ninguna persona externa haga muestras falsas.

Para lograr evitar algún tipo de bypass o intrusión en la aplicación se realizó un script, que se encuentra en cada página de la aplicación, éste evita ejecutar el resto de la aplicación cuando el usuario no se ha autenticado ver figura 50.

```
<script type="text/javascript" src="js/sesion.js" ></script> <!--sesion-->

<script type="text/javascript">
    var url = window.location.pathname;var filename = url.substring(url.lastIndexOf('/')+1);
    if(localStorage.login=="true" && filename == 'login.html')
   {
    window.location.href = "form.html";
   }
    else if(localStorage.login=="false" && filename != 'login.html')
   {
    window.location.href = "login.html";
   }
   </script>
```

Figura 50 Script que evita la instrucción sin autentificación

Así mismo, por medio de las funciones `addslashes` o `stripslashes` de PHP se evita la inyección SQL que es un método de infiltración de código intruso que se vale de una vulnerabilidad informática presente en el nivel de validación de credenciales. Así mismo se realiza un filtro anti-XSS el cual protege a aplicación de inyección de código malicioso, regularmente escrito en JavaScript, todo esto con la finalidad de salvaguardar la información de los usuarios.

```
    //Filtro anti-XSS
$caracteres_malos = array("<", ">", "\"", "'", "/", "<", ">", "'", "/");
$caracteres_buenos = array("& lt;", "& gt;", "& quot;", "& #x27;", "& #x2F;",
 "& #060;", "& #062;", "& #039;", "& #047;");
$consultaBusqueda = str_replace($caracteres_malos, $caracteres_buenos, $consultaBusqueda);
    // se agrega addslashes para evitar la inyeccion sql
    //se puede con el uso de real_escape_string()
    $username = addslashes(trim($_POST["username"]));
    $password = addslashes(trim($_POST["password"]));
```

Figura 51 Filtros de inyección SQL, XSS realizados en PHP y almacenados en el servidor

El la figura 51 se observa que la función PHP `$caracteres_malos` que se encuentra en el servidor cuenta con un arreglo de caracteres que se utilizan para ejecución de códigos maliciosos desarrollados en javascript, y se elimina su ejecución. Más abajo de la figura se observa la función PHP `addslashes` el cual ingresar o ejecutar cualquier tipo de comillas (‘’ ””) simplemente las quitara. Las combinaciones de estos caracteres son muy usados para la inyección de código SQL, el cual podría lograr el ingreso o autenticación a la aplicación.

Al ingresar las credenciales de usuario y contraseña correctas y presionar el botón "**iniciar sesion**" se presenta una pantalla de la interfaz donde se encuentra el menú principal, como se puede ver en la figura 52. Los iconos que aparecen en este menú fueron editados con ayuda de Photoshop a excepción del logo del laboratorio. Gracias al estilo que ofrece jQuery Mobile se realizó el menú mediante etiquetas de HTML5.

Figura 52 Menú principal

Éste cuenta con distintas opciones como son: realización de nuevo módulo, consulta de resultados, información acerca del laboratorio y la opción de salir. Al ingresar a alguna opción los script jQuery Mobile y Ajax implantados esperarán el evento indicado para poder realizar la transición hacia la opción que el usuario indique.

4.8 Opciones de Menú

El menú principal se elaboró mediante el etiquetas de HTML5 como `<ul data-role="listview">` dentro del cual se insertan etiquetas de listas `<ul>` e `<li>` cada uno de los elementos `<li>` contiene el texto que tiene que mostrar y además de un link o enlace con la pagina concreta que tiene que enseñar. Por lo tanto, cada opción de la lista del menú envía a una página distinta, dentro de las cuales se elaboran funciones específicas.

A continuación, se detallan de manera independiente la selección de cada opción del menú principal debido a que el proceso y desarrollo de cada una varía respecto a la selección del usuario.

4.8.1 Opción "Nuevo Módulo"

Para esta opción se diseñaron las partes del formulario, en los cuales se ingresan los primeros datos, para este trabajo se definieron como "Datos básicos". Son enviados al servidor con la finalidad de almacenarlos. Al presionar sobre esta opción se ejecutan funciones de Ajax insertados en las bibliotecas

de jQuery Mobile que permiten la transición hacia la primera parte del formulario que tiene como título "**Datos Básicos**".

Figura 53 Primera parte de formulario

Esta parte del formulario (ver figura 53), se debe llenar correctamente mediante parámetros indicados en cada caja de texto. Cabe señalar que cada caja de texto responderá al tipo de etiqueta HTML5 `<input>` por ejemplo, si se desea que la entrada del dato sea numérico el tipo de la etiqueta se define como `<input type =number>` lo cual al seleccionar esta caja de texto en el dispositivo móvil se mostrará el teclado donde solo se observaran los números de 0 a 9, lo mismo sucede de forma similar en cada tipo de entrada.

Al terminar esta primer parte del formulario los datos se envían al servidor por medio del programa, elaborado en JavaScrpit, el cual recupera los datos en un arreglo y los envía por el método POST al servidor. Los datos enviados son leídos en el programa PHP que se desarrolló en el servidor, y al llegar a la operación `SELECT` los datos ingresan a la estructura selectiva `if-else` donde se lleva a cabo la verificación del nombre de la muestra a realizar.

```
var formulario_1 = $("#formulario_1").serializeArray();
$.post(url_v,formulario_1).done(function(respuesta)
{
    if (respuesta == "true")
    {
        window.location.href = "form.html#identica";

    }
    else
    {
        window.location.href = "form.html#enviados";

        $("#ultimo_id").val(respuesta);
    }
    limpiarformulario("#formulario_1");
},"json");
```

Figura 54 Script de respuestas

Como se aprecia en la figura 54 sí la respuesta devuelta del programa PHP a JavaScrit es “true” o “false” se ejecutara la función “`window.location.href`” la cual mostrará un cuadro de dialogo correspondiente,

Por ejemplo, si la respuesta que devuelve la función PHP es “true” significa que el nombre de la muestra ya está registrada, por lo tanto se mostrará el cuadro de dialogo con el mensaje “**El nombre de la muestra ya existe y éste debe ser cambiado**”. En caso contrario, los datos ingresan a la operación “`INSERT INTO`” (ver figura 55), y son almacenados en la tabla de la base de datos, es decir, no ocurre ningún error por lo tanto los datos son enviados y almacenados.

```
$v_muestra = "SELECT id  FROM tabladatos WHERE  Muestra2='$mues   '";
$valida = $datos->query($v_muestra);//Como la muestra es UNIQUE si valida tiene m
    if(mysqli_num_rows($valida) != 0)
    {
        echo "true";//Error al registrar! - muestra Duplicada - Ingresa otra
    }
    else//Si no hubo muestra repetida se agrega
    {   $q = "INSERT INTO tabladatos2 (Fech, Nomb, Mues, Pes, Sis, Mater, Subs )
        VALUES ('$date ','$nom','$mues','$ps', '$sist', '$mate' , '$subs'); ";

        $result = $datos->query($q);
        $id=mysqli_insert_id($datos);
        echo "$id";
    }
```

Figura 55 Funciones en PHP.

Cuando PHP envía la respuesta “false” a JavaScript, se ejecutará la función que muestra un cuadro de dialogo con el mensaje “**Datos enviados Correctamente, es necesario que termines la muestra sin regresar”** y al presionar **“aceptar”**, se ofrece al usuario una nueva pantalla donde se encuentra la segunda parte del formulario denominada “**Presiones y Temperaturas**”.

Algo importante que hay que señalar es que para lograr continuar escribiendo datos sobre la misma fila y columnas de la muestra, es necesario recuperar un identificador de ésta y así lograr pasar el dato a la siguiente parte del formulario. Para esto PHP ofrece la función “`mysqli_insert_id`” el cual devuelve el id auto agregado que se utilizó en la última inserción de datos. Éste dato es devuelto a JavaScript, lo entrega a HTML5 en la segunda parte del formulario mediante la etiqueta `<input type="hidden">`, aquí el identificador es recuperado por tanto no es necesario ingresarlo, además éste dato no será visible para el usuario, pues no es relevante para él; sin embargo es utilizable solo para la lógica del servidor.

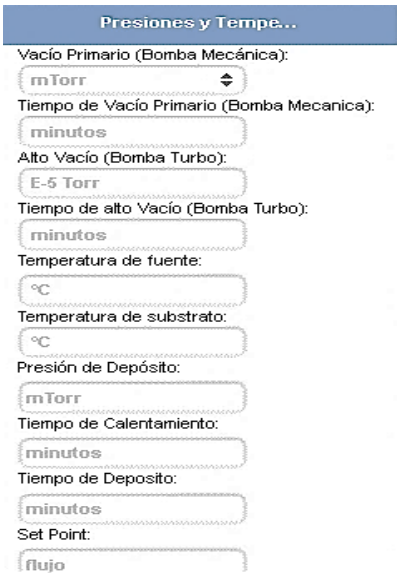


Figura 56 Segunda parte del formulario

Como se ve en la figura 56 los datos en esta parte son de tipo numérico y cuando se ingresan todos, el envío de éstos (incluyendo el identificador) hacia al servidor se hace de la misma manera y por el mismo método que se utilizó en el formulario "**Datos Básicos**".

Otro programa elaborado en PHP recibe los datos y a través de la estructura selectiva "`if-else`" se verifica que el dato del identificador vaya incluido en el arreglo. En algunas ocasiones el identificador no es recuperado correctamente, puede ser debido a que se haya perdido la conexión inalámbrica o el usuario haya salido de la aplicación. Si esto sucede, significa que el dato se perdió al enviarse y por tal motivo se enviará la respuesta "true" a JavaScript el cual ejecutará de igual manera la función `window.location.href` y por lo tanto, se mostrara el cuadro de dialogo "**Hubo un error al recuperar el Id**".

En caso de que el identificador y el resto de los datos lleguen correctamente, se aplicará la operación `UPTADE` en el programa PHP. Los datos capturados son almacenados en las columnas de la fila que contiene el mismo identificador de la muestra. JavaScript recibe la respuesta "false" la cual da paso al siguiente formulario, desplegando un cuadro de dialogo con el mensaje "**Datos enviados correctamente, continúe por favor**".

```
$muestra_ag = "SELECT *  FROM tabladatos WHERE id='$reg_id'";
$valida_ag = $agregar->query($muestra_ag);
$resu=mysqli_num_rows($valida_ag);
if( $resu == 0)
{
    echo "true";
}
else
{
    $up = "UPDATE tabladatos2 SET
    PreVacPrim2='$v_pri',
     TiemVacPrim2='$t_pri',
      PresAltVac2='$a_va',
      TemAltVac2='$t_av',
       TemSubs2='$t_sub',
       TempFuente2='$t_ft',
       PresDep2='$pre_d',
       TiemCalem2='$t_ca',
       TiemDep2='$t_de',
       SetPoint='$sp',
       Atmosfera='$atm'
        WHERE id='$reg_id'";

    $re_up = $agregar->query($up);
    echo "$reg_id";
}
```

Figura 57 Actualización de datos sobre misma muestra.

Como se muestra en la figura 57 para recuperar el identificador solo se retoma el arreglo de datos y se realiza un "`echo`" para así solo recuperar el identificador y enviarlo nuevamente a JavaScript, el cual lo entrega a la caja de texto del HTML5 de la última parte del formulario, donde nuevamente no será mostrado. Este dato ofrece la misma función; la de escribir los siguientes datos sobre el nombre de la misma fila de datos.

La parte final del formulario recibe el nombre de "**Temperatura vs Tiempo**" y es el formulario más extenso del proyecto debido a que la captura de datos se debe realizar cada cierto tiempo, es decir, del minuto "0" al minuto "12" la captura es consecutiva, posteriormente la captura se realiza cada cuatro minutos hasta llegar al minuto "60", el tiempo de captura está definida por los investigadores del laboratorio y usuarios de los sistemas.

Para lograr comprender el proceso del siguiente formulario, como primera parte se desarrolló el plugin del temporizador, el cual realiza un conteo ascendente y cuenta con un conector básico para funcionar con jQuery. Las funciones establecidas se realizan a base de botones como, iniciar el temporizador, detener o ponerlo a cero y el estilo de estos botones fue definido mediante el lenguaje CSS (ver figura 58).

Después, la ubicación del formulario se encuentra dentro de una tabla en forma vertical la cual permite visualizar los minutos y las dos temperaturas a capturar. Este formulario se visualiza mediante el desplazamiento suave de la tabla utilizando HTML5 y CSS3. Para lograrlo, se utilizaron propiedades de posicionamiento, así como `-web-kit-overflow-scrolling` de CSS3. Esta última propiedad genera una vista desplazable como si fuera una aplicación nativa.

Por último el envío de esta parte del formulario se realiza de manera similar, la función que permite recuperar el identificador ya no es utilizada, pero también se realiza una verificación para determinar si el identificador de la parte anterior del formulario llegó correctamente. Si este no llega, se mostrara el

cuadro de dialogo mencionando el error, en caso contrario el cuadro de dialogo mostrara el mensaje "**Datos enviados correctamente, puede verificar los datos en la opción 'Resultados' del menú principal**".

Figura 58 Última parte del formulario

4.8.2 Opción "Resultados"

La opción "**Resultados de Muestras**" se elaboró en dos partes. La primera constituye una caja de texto de tipo `<type="search">` donde se cuenta con la función de autocompletar, es decir, al ingresar algún carácter relacionado con el nombre de la muestra, aparecen sugerencias a partir de la información almacenada en la tabla donde se encuentra la información (base de datos). El desarrollo de esta función se llevó a cabo mediante el widget *Autocomplete* de jQueryUi, PHP almacena en la variable "muestra" el dato que se va a buscar, este valor se obtiene de la variable global $_GET con el nombre "term". Se crea una conexión a la base de datos y se ejecuta la consulta que devolverá una lista de muestras con los caracteres ingresados, estas muestras se almacenan en un arreglo que lleva por nombre "muestras", se selecciona la deseada y al final el arreglo se pasa a json`_encode` y se imprime en la caja de texto gracias a JavaScript.

Al presionar el botón "**Buscar Muestra**" se ejecuta la función "buscar.php" la cual devuelve la información completa de la muestra con base al nombre que se encuentra en la caja de texto. Este nombre se recibe mediante el método $_POST, se crea nuevamente la conexión a la base de datos y se ejecuta la consulta. Posteriormente se agrega a la variable "respuesta" los datos de la muestra y se imprime en la tabla gracias a `json` a través de Ajax. Como se puede ver en la figura 59 en la tabla se visualizaran los datos básicos, seguidos de los resultados de presiones y temperaturas, después se muestra los datos de las temperaturas capturadas en cada minuto y así mismo se muestra el gradiente, el cual es calculado mediante una función implementada en el servidor.

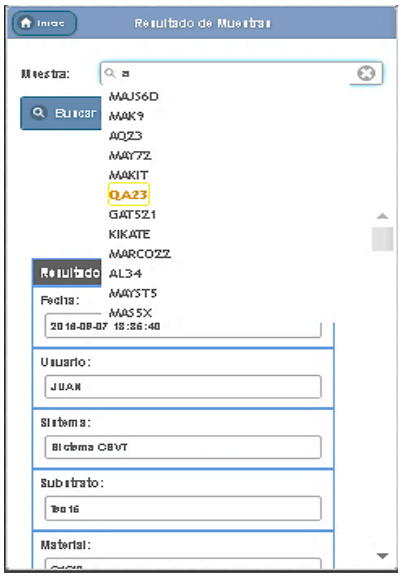


Figura 59 Consulta de resultados

Como segunda parte los datos de las temperaturas se pueden graficar gracias al plugin Highchartst.js el cual por medio de funciones elaboradas en JavaScript y PHP coloca los datos de la muestra consultada y gracias a los los estilos CSS3 pueden ser visibles de una manera agradable y sobre observar la correlación entre las curvas de temperaturas, como se ilustra en la figura 60 donde la temperatura fuente es de color rojo y la curva de temperatura del sustrato de color verde.

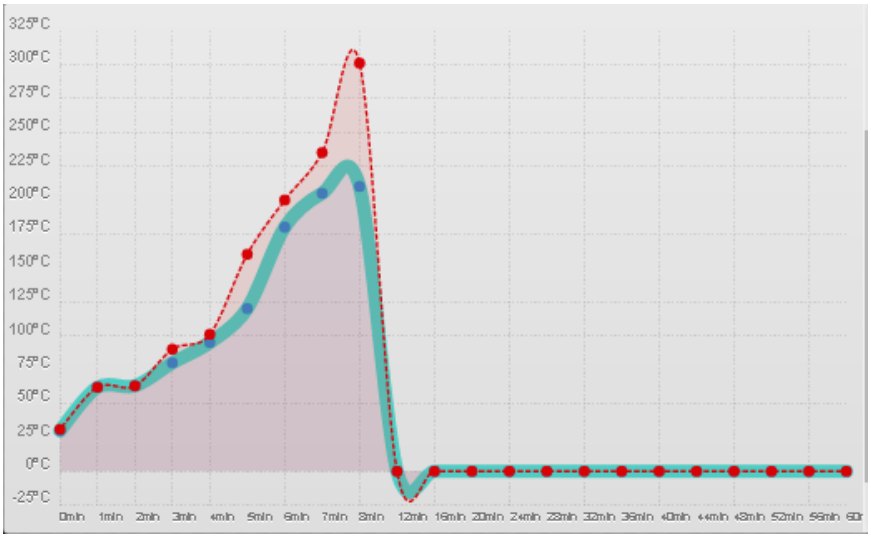


Figura 60 Grafica de temperaturas.

4.8.3 Opción “Información”

En esta opción, el usuario se encontrará con información detallada acerca del laboratorio y sus programas de investigación mediante distintos controles desplegables, en los cuales se encuentra la información. Al seleccionar un control despegable se mostrará la información que contiene éste. Al pie de la pantalla se encontrara lo que se conoce como “footer” dentro del cual se visualizaran tres opciones. Al presionar la opción “UACM” permitirá ingresar a la página oficial de la universidad, si se selecciona la opción

“Inicio”, enviara al usuario al menú principal y por último la opción “Contacto” desplegara un cuadro de dialogo con los datos específicos del laboratorio. Estos controles descritos se pueden apreciar en la figura 61.

Figura 61 Acordeones que despliegan la Informacion

4.8.4 Opción “Salir”

Cuando se selecciona esta opción del menú principal se ejecuta un script que envía la petición a PHP, donde se destruye la sesión y después el mismo scrpit direcciona a la página de inicio de sesión al presionar en el botón “**Aceptar**” como se ve en la figura 62.

Figura 62 Dialogo de salida de aplicación.

4.9 Firma de aplicación

El firmado de la aplicación es importante ya que de lo contrario al querer integrarla a un dispositivo móvil aparecerá un error de tipo "'*Package file was not signed correctly*", este problema es debido a que la firma del archivo APK, no ha sido realizada.

Para realizar la firma se usó la interfaz consola de node.js, la cual permite generar una *keystore.* Para esto se pone el siguiente comando (se puede cambiar el nombre del *keystore* y del *alias* el resto permance de la misma forma):

```
keytool -genkey -v -keystore lacytestore.keystore -alias lacytes -keyalg RSA -keysize 2048 -validity 10000
```

Después la consola pregunta dos veces por la contraseña del *keystore* y realiza una serie de preguntas, que se deben responder adecuadamente como: nombre, apellido, el nombre de la unidad de organización, nombre de la organización, ciudad, provicia y código del país como se muestra en la figura 63. Al terminar muestra los datos que se ingresaron y si está todo correcto se escribe '`si`'. Por ultimo preguntara por la contraseña de alias, para mantener la misma contraseña para el *keystore* solo se presiona '**Enter**', de lo contrario el desarrollador deberá asignar una nueva contraseña.

```
C:\Users\MARCOA>keytool -genkey -v -keystore lacytestore.keystore -alias lacytes
 -keyalg RSA -keysize 2048 -validity 10000
Introduzca la contraseña del almacén de claves:
Volver a escribir la contraseña nueva:
¿Cuáles son su nombre y su apellido?
  [Unknown]:  lacytes uacm b406
¿Cuál es el nombre de su unidad de organización?
  [Unknown]:  lacytes
¿Cuál es el nombre de su organización?
  [Unknown]:  UACM
¿Cuál es el nombre de su ciudad o localidad?
  [Unknown]:  Cuidad de Mexico
¿Cuál es el nombre de su estado o provincia?
  [Unknown]:  Ciudad de Mexico
¿Cuál es el código de país de dos letras de la unidad?
  [Unknown]:  MX
¿Es correcto CN=lacytes uacm b406, OU=lacytes, O=UACM, L=Cuidad de Mexico, ST=Ci
udad de Mexico, C=MX?
  [no]:  si

Generando par de claves RSA de 2.048 bits para certificado autofirmado (SHA256wi
thRSA) con una validez de 10.000 días
        para: CN=lacytes uacm b406, OU=lacytes, O=UACM, L=Cuidad de Mexico, ST=C
iudad de Mexico, C=MX
Introduzca la contraseña de clave para <lacytes>
        (INTRO si es la misma contraseña que la del almacén de claves):
[Almacenando lacytestore.keystore]
```

Figura 63 Creación de firma de seguridad para aplicación móvil

Como se observa en al final de figura 63, al finalizar se crea un archivo 'lacytestore.keystore' en la carpeta donde se haya ejecutado el comando. Es necesario copiar este archivo en la raíz del proyecto de la palicacion.

Después, para colocar la firma en la aplicación es necesario dirigirse a la raíz del proyecto que se desea firmar y allí dentro se crea un archivo de texto con el nombre 'build.json'. Dentro del texto es necesario usar los nombres, contraseñas de acuerdo a como se creó el keystore, estos parámetros se colocan como se muestra en el siguiente contenido.

```
{
      "android": {
          "debug": {
              "keystore": "lacytestore.keystore",
              "storePassword": "tutorial",
              "alias": "lacytes",
              "password" : "telcom05",
              "keystoreType": ""
          },
          "release": {
              "keystore": "lacytestore.keystore",
              "storePassword": "telcom05",
              "alias": "lacytes",
              "password" : "telcom05",
              "keystoreType": ""
          }
      }
  }
```

Finalmente, se tiene todo lo necesario para poder construir la aplicación y al finalizar el desarrollo de esta, se podrá compilar a un formato .apk. Para ello se realizó usando el siguiente comando:
cordova build android –release

De esta forma se obtiene la aplicación compilada y firmada en la carpeta `/platforms/android/build/outputs/apk/ con el nombre 'app-sisu-release.apk'.`

4.10 Pruebas de la aplicación

El diseño de la aplicación consta de varias tecnologías web y fue importante realizar pruebas que permitieron conocer el funcionamiento de la aplicación.

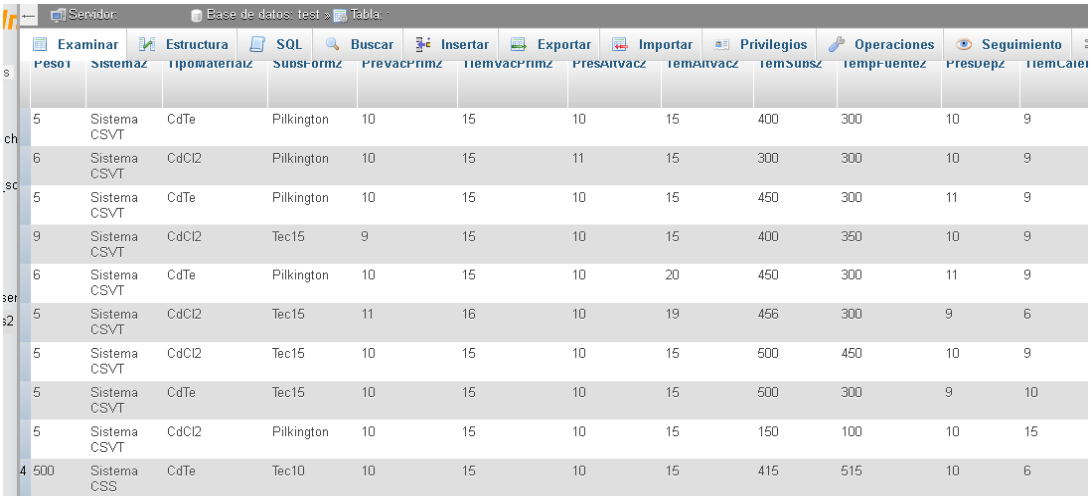
Las pruebas que a continuación se detallan se realizan en distintos escenarios como: en la base de datos, aplicación móvil, las funciones PHP desarrolladas en el servidor y confirmar que los scrpits de la aplicación funcionan. Estas pruebas se realizan en un emulador y en la la aplicación instalada en la tableta electrónica, con el objetivo de verificar que las funciones implantadas se encuentran funcionando adecuadamente y los datos sean almacenados de manera precisa.

4.10.1 Prueba de envió de datos

Para esta prueba se efectúa la captura de una muestra aleatoriamente de los datos y comprueba que están siendo enviados de manera precisa y correcta al servidor; para ello, se ingresa a la base de datos en donde se encuentra la tabla que almacena los datos.

Como se observa en la figura 64, la información de la muestra capturada por medio de la aplicación efectivamente es enviada y colocada en su respectivo arreglo de fila y columna. Además, se verifica que ninguno de los datos falte, sólo a excepción de aquel donde el usuario haya olvidado registrar alguno,

por lo cual es conveniente realizar o implementar (por desarrollo de script) una verificación en el formulario donde se contenga los datos requeridos.



5	Sistema CSVT	CdTe	Pilkington	10	15	10	15	400	300	10	9
6	Sistema CSVT	CdCl2	Pilkington	10	15	11	15	300	300	10	9
5	Sistema CSVT	CdTe	Pilkington	10	15	10	15	450	300	11	9
9	Sistema CSVT	CdCl2	Tec15	9	15	10	15	400	350	10	9
6	Sistema CSVT	CdTe	Pilkington	10	15	10	20	450	300	11	9
5	Sistema CSVT	CdCl2	Tec15	11	16	10	19	456	300	9	6
5	Sistema CSVT	CdCl2	Tec15	10	15	10	15	500	450	10	9
5	Sistema CSVT	CdTe	Tec15	10	15	10	15	500	300	9	10
5	Sistema CSVT	CdCl2	Pilkington	10	15	10	15	150	100	10	15
500	Sistema CSS	CdTe	Tec10	10	15	10	15	415	515	10	6

Figura 64 Datos almacenados en el servidor

4.10.2 Prueba de funciones PHP

En el desarrollo del formulario de sesión, se utilizaron funciones de PHP las cuales evitan ejecutar los códigos SQL (apartado 4.7 ‘Autentificación de usuarios’) de forma segura, por lo cual se pretende verificar la funcionabilidad de éstas, realizando dos pruebas sobre la aplicación. La primera prueba consta de ingresar credenciales (usuario y contraseña) validadas en el servidor y así comprobar que el sistema de autentificación funciona al momento de enviar los datos. Esta prueba se realizó en el emulador ‘ripple’ instalado en el navegador web Chorme, su instalación se describe en el anexo ‘B-1’ en conjunto con la ejecución del framework phonegap. Como se muestra en la figura 65 al presionar el botón “iniciar sesión” la información se envía al servidor donde se corrobora y permite el ingreso al menú principal.

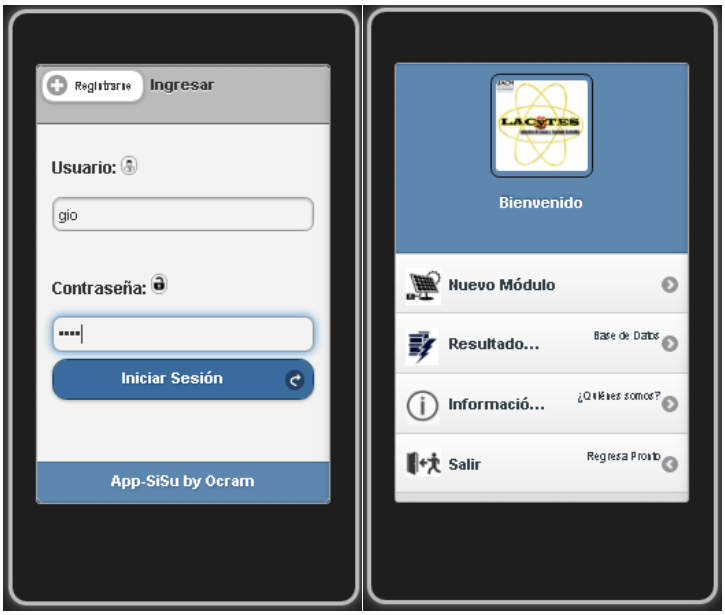


Figura 65 Ingreso al menú después de autentificar correctamente

La segunda prueba se realizó insertando código SQL 'or'1'='1 en las cajas de texto de usuario y contraseña como se muestra en la figura 66. Al presionar el botón "iniciar sesión" este arroja el dialogo "**error, usuario o contraseña invalida**" lo cual quiere decir que las funciones PHP realizan su labor

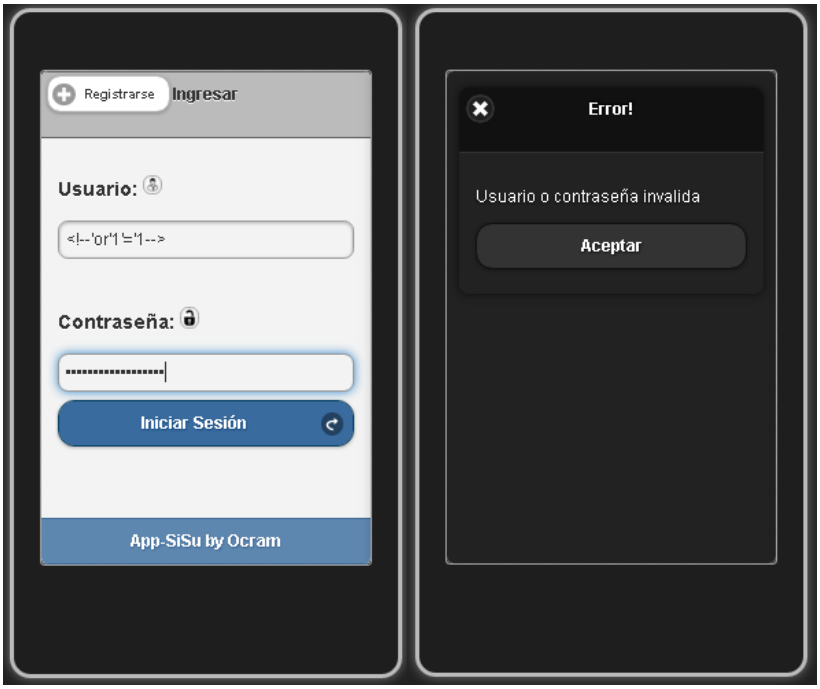


Figura 66 Inyección SQL

Sin embargo, se puede elaborar un formulario con captación de captchas[10] para validar las peticiones de los usuarios, se utilizan para impedir que se pueda tener acceso a la función de un script de forma automática.

4.10.3 Prueba de gráfica

Esta prueba se realizó para verificar la sincronización de los datos con la petición de elaborar la gráfica, es decir, revisar que realmente está graficando la muestra consultada.

Primeramente se consultó una de las muestras 'AL34' y se tomaron sus datos uno a uno en tabla 12. Se elaboró la gráfica en la aplicación y se verifico que los datos mostrados por esta son graficados correctamente y que sean exactamente los mismos.

Tabla 12 Datos de muestra.

Datos Temperaturas Vs Tiempo											
Minutos	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**12**	**16**
°C Substrato	30	62	63	80	95	120	180	205	210	0	0
°C Fuente	31	62	63	90	101	160	200	235	301	0	0

[10] Es una prueba para comprobar que quien introduce la información es una persona y no un sistema informático.

Minutos	20	24	28	32	36	40	44	48	52	56	60
°C Substrato	0	0	0	0	0	0	0	0	0	0	0
°C Fuente	0	0	0	0	0	0	0	0	0	0	0

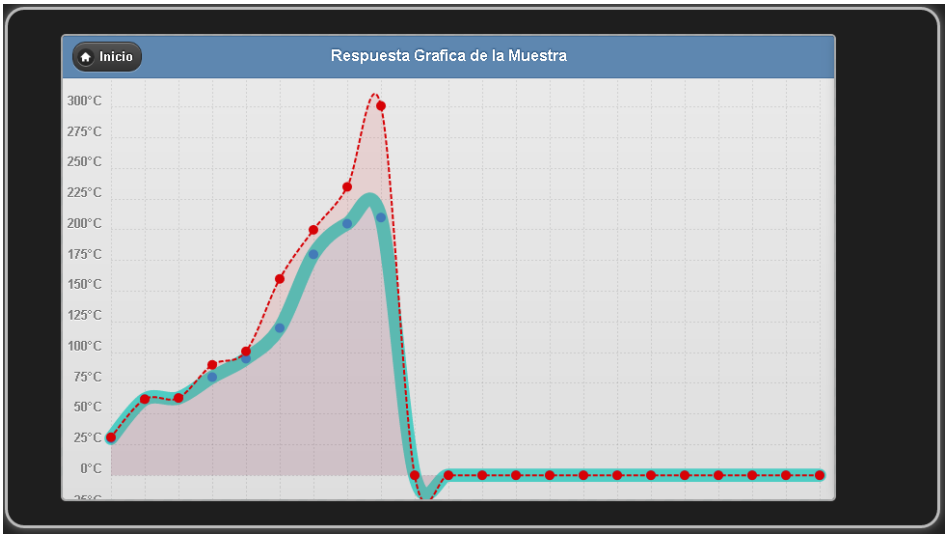


Figura 67 Gráfica a través de la aplicación de la muestra AL34

El resultado que se aprecia en la figura 67 concuerda perfectamente con los datos de la tabla 10. Por lo cual se concluye que esta porción de la aplicación funciona correctamente.

4.10.4 Prueba de Bypass

Cuando se desarrolla una aplicación hibrida a ésta se le puede inyectar código XSS el cual como ya se sabe está desarrollado a base de código JavaScript. Este aprovecha las variables que se pasan entre páginas usando una URL o substrayendo las cookies. Para lo cual con la implementación de las funciones PHP y un script elaborado en JavaScript se logra evitar acceder a la aplicación sin la validación de credenciales. Para la realización de esta prueba se hizo uso del emulador antes mencionado.

En la barra de navegador web se coloca la página de inicio de sesión.

Figura 68 Re direccionamiento tras realizar Bypass

En esta barra de navegación se ingresa alguna dirección que se encuentre dentro de la aplicación. Por ejemplo `https://localhost/MyLacytes/www/form.html#reloj` que es la dirección que muestra el formulario donde se encuentra el reloj. Como se muestra en la figura 68, el resultado fue el direccionamiento hacia la página de inicio de sesión, por lo cual la única manera de acceder a las distintas direcciones de la aplicación es por mediante un ingreso correcto de credenciales.

4.11 Resultados de aplicación móvil

Los resultados obtenidos tras la realización del diseño e implementación de la aplicación sobre el dispositivo móvil han cumplido con los objetivos planteados en un inicio, el análisis que se realizó sobre el portal web y sobre las bitácoras de información fue de gran ayuda, ya que debido a estos se pudo diseñar una aplicación con características similares.

Gracias al prototipo de la aplicación se logró identificar las necesidades de los usuarios, en éste se especificaron los requerimientos y se detallaron funcionalidades, donde además se definió y diseño el aspecto de la navegación. Se cumple así el objetivo de tener una buena arquitectura de la información.

Con la ayuda de los frameworks (Node.js y Phonegap) la aplicación fue construida de una manera factible y rápida, donde las tecnologías web que se usaron: jQuery, JavaScript, CSS3 y HTML5, permitieron desarrollar la vista sin mucha complejidad de código y se logró obtener una aplicación hibrida compatible con navegadores.

Tomando en cuenta la ampliación de la base de datos anteriormente ya existente, el almacenamiento de éstos se realizó de manera remota, es decir, se almacenan en un servidor y no en el celular, esto gracias

a que se elaboraron funciones mediante JavaScript y PHP logrando realizar una comunicación exitosa entre cliente (tablet) y Servidor.

El tiempo de atención al momento de realizar la captura de los datos se reduce y agiliza en mayor tiempo. Además, el problema de pérdida de información de bitácoras sobre los sistemas CSVT-IR y CSS-IR se ha resuelto en gran medida pero es necesario vaciar los datos que aún se encuentran en las bitácoras de papel, para que el 100% de la información de los sistemas se encuentre en el servidor.

Así mismo el poder realizar una gráfica de la muestra consultada ayuda mucho a los investigadores dado que con esto ya no es necesario estar ingresando dato por dato en otro programa para elaborar un análisis del comportamiento de las temperaturas de los depósitos realizados.

Las pruebas realizadas en el emulador aportan mucha información ya que la aplicación se comporta adecuadamente en este pero para descartar errores reales también fue necesario realizarlas en un dispositivo físico como se muestra en la figura 69.

Figura 69 Aplicación en dispositivo móvil

La prueba realizada en el emulador arroja los mismos resultados obtenidos, la aplicación responde de la misma manera y la navegabilidad es muy fluida. La aplicación tiene su manejo eficiente, el acceso a la información sobre muestras realizadas y del mismo modo el proceso de elaborar un gráfico (Temperaturas vs Tiempo) se realiza de manera rápida.

La compilación de la aplicación se realizó mediante Node.js y la aplicación tiene un tamaño de 73.6 MB por lo cual es una aplicación muy ligera respecto a las características de nuestro dispositivo móvil.

Conclusiones

De acuerdo a los resultados obtenidos, después de realizar las pruebas de conectividad, de disponibilidad de la red y funcionamiento de la aplicación se puede concluir lo siguiente:

Se logró implementar una red interna en base el diseño propuesto, mediante el cálculo de las subredes por el método VLSM con el cual se realizó una red de datos adaptable a las necesidades del laboratorio. Además gracias al uso de un cortafuego-router de software libre (pfSense) la red puede ser administrada, configurada y modificada desde un sólo equipo, dado que el cortafuegos interactúa con las distintas redes y subredes.

Gracias a las reglas de filtro y reglas NAT que se definieron en el cortafuego se demuestra la importancia de éstas, dado que logran minimizar en un porcentaje los riesgos de seguridad asociados a la red de datos, así mismo de asegurar los recursos informáticos mediante el uso del cortafuego. Hoy en día el laboratorio cuenta con una red de datos amoldable a las necesidades futuras y se ponen en uso el funcionamiento y la descripción de la capa 1, 2, 3 del modelo ISO/OSI.

Gracias al uso de tecnologías web y de software como Phonegap, Node.js, HTML5, JavaScript, CSS3, PHP se logró una sinergia de éstos para lograr construir una aplicación hibrida que permite realizar la captura y almacenamiento de información. La comunicación que se realiza entre el servidor y la aplicación se ejecutan gracias a funciones elaboradas en JavaScript, que permiten acceder a la biblioteca Webkit nativa del dispositivo donde Phonegap actúa como un puente entre las aplicaciones hibridas y dispositivos móviles.

Al elaborar una serie de pruebas del funcionamiento de la aplicación se demuestra que ésta agiliza el envío y almacenamiento de datos experimentales, permite consultar los datos y ofrece la opción de elaborar graficas gracias a la implementación de funciones a nivel lógico y de control en el desarrollo de la aplicación. Además se demuestra la utilización de la infraestructura de red diseñada por medio de la aplicación dado que se utilizan el envío de la información a través de la red (capa 4), la gestión y finalización de las conexiones con usuarios finales (capa 5), la codificación y presentación de la información por medio de una interfaz (capa 6 y 7), por lo que se puede afirmar que se logró establecer la arquitectura cliente-servidor.

Finalmente con los objetivos alcanzados de este trabajo se logra demostrar que las técnicas de las ingenierías pueden contribuir a minimizar los riesgos de seguridad asociados a la red de datos, protección de información y consulta de ésta. Se deja el link donde se puede consultar la programación de los archivos HTML5, Javascript, CCS3 y jQuery, este aporte da paso a continuar con la estipulación de la filosofía del proyecto GNU en el cual los usuarios tienen la libertad de estudiar y modificar el código fuente de la aplicación, además implica la redistribución de la copia exacta y la distribución de versiones modificadas.

Link: https://github.com/OcramZahir/

Trabajo a futuro

De acuerdo a las conclusiones y experiencias obtenidas durante la realización de este trabajo, se enlistan algunos de los trabajos que pueden ser desarrollados con la finalidad de realizar mejoras en la red interna implementada así como en la aplicación móvil.

- Aislar la red LAN mediante de una arquitectura Front-*End*: esta arquitectura tiene como objetivo aislar las subredes cada una con un respectivo cortafuego, lo cual lograría ofrecer mayor seguridad en la red de datos.

- Implementación de un cortafuego en paralelo: la red desarrollada por este trabajo cuenta con un solo cortafuego, sin embargo es necesario tener un respaldo en caso de que el primero llegue a presentar algún daño (externo o interno), pudiendo entrar el segundo como respaldo además de ofrecer un nivel más de seguridad.

- Descentralizar los servicios web: el servidor host que ofrece servicios web almacena cada uno de los requerimientos (PHP, base de datos, servicios web, email, almacenamiento, etc.) y es necesario distribuir estos servicios con la finalidad de no exponer todos en un solo sistema.

- Incluir el resto de los sistemas experimentales: la aplicación se puede ampliar para elaborar la captura de los datos así como la generación de bitácoras digitales de los demás sistemas que se encuentran en el laboratorio.

- Descargar datos de la aplicación en formato .pdf: Gracias a la implementación de plugins (jspdf) es posible descargar los datos deseados.

- Comparación de datos: elaborar una gráfica donde se muestren distintas consultas y se grafiquen simultáneamente para elaborar comparaciones de eficiencia o analizar el comportamiento del deposito

Referencias

[1] Agustin Bender Gabriel, David Chavez. (2014). Desarrollo de aplicaciones hibridas multiplataforma con Phonegap para dispositivos HandHeld. Argenitna: UNPSJB

[2] Alejandro Corletti Estrada. (2011). Seguridad por niveles. Madrid: Learning Consulting. S.L.

[3] Alex Handy,[en linea];junio 2011, Node.js pushes JavaScript to the server-side,[consulta: 16 de octubre 2016], disponible: http://www.sdtimes.com/NODE_JS_PUSHES_JAVASCRIPT_TO_THE_SERVER_SIDE/By_Alex_Handy/About_JAVASCRIPT_and_NODEJS/35668

[4] Andrew S. Tanenbaum. (1997). Redes de computadoras. México: Prentice hall.

[5] Barry M. Leiner, Vinton G. Cerf, David D. Clark, Leonard Kleinrock, [en línea]; 2010. Breve historia de internet, [consulta 13 de marzo del 2016]. De: Internet Society, disponible: http://www.internetsociety.org/es/breve-historia-de-internet

[6] Bruce Hallberg. (2007). Fundamentos de redes. México: McFraw-Hill

[7] C. A. Gunter, (1998). Un lenguaje de programación para redes activas. Actas de la Conferencia Internacional sobre Programación Funcional (ICFP) '98.

[8] Camilo Rodríguez, Héctor Enríquez. (2014). Caracteristicas del desarrollo en Frameworks multiplataforma para móviles. Ingenium vo. 15 n° 30.

[9] Cintia Quezada Reyes, Ma. Jaquelina López Barrientos. (2004). Fundamentos de Seguridad Informatica. México: U.N.A.M

[10] Cisco System Inc. [en línea]; 18 octubre 2015. Direccionamiento de IP y conexión en subredes para los usuarios nuevos. [consulta: 16 de diciembre del 2015]. De: Cisco Sitio web, disponible: http://www.cisco.com/cisco/web/support/LA/102/1025/1025418_3.html

[11] Daniel Mayan. (2014). Redes y Subredes de San justo, Buenos Aires, Argentina: Universidad Nacional de la Matanza

[12] Diana Calderon Onofre, Marín Estrella Ochoa y Manuel Flores Villamarín. (2011). Implementación de sistema de gestion de seguridad de la informacion aplicada al area de recursos humanos de la empresa DECEVALE S.A. Escuela Superior Pilitecnica del Litoral.

[13] Douglas E. Comer (2000). Redes globales de informacion con Internet y TCP/IP Principios basicos, protocolos y arquitecura. México: Prentice-Hall

[14] Eduard Lara [en línea]; 2014. Documentacío de xarxes de computadors i sistemes operatius. [consulta 16 de enero del 2016], disponible: http://elara.site.ac.upc.edu/

[15] Erik Rolando Vidal Bazini (2014),Beneficios de usar tecnologia movil para la industria de distribuidoras orientado a pequeñas y medianas empresas. Guatemala: Universidad de San Carlos de Guatemala

[16] Ernesto Ariganello, Enrique Barrientos S. (2010).Redes Cisco: CCNP a fondo, guia de estudio para profesionales. Ra-Ma,S.A

[17] Firtman, Maximiliano. (2015) AJAX. Web 2.0 con jQuery para profesionales. Alfa Omega

[18] Fundamentos de redes. (2011) Microsoft Official Academic Course. EU: Wiley

[19] Henry Cristhian Mancheno Torres e Ivette Lorena Robles Coronel (2013). Vulnerabilidades y Seguridad en redes TCP/IP. Ecuador: Universidad Católica de Santiago Guayaquil.

[20] Hubert Zimmerman, (abril de 1980). OSI Reference Model – The ISO Model of Architecture for Open Systems Interconnection

[21] Innovanube, [en línea]; 7 de junio 2016. El desarrollo de aplicaciones móviles [consulta: 20 Febrero 2016]. De: TICbeat, disponible:http://www.innovanube.com/docs/ticbeat%20-%20desarrollo_de_apliaciones_moviles.pdf.

[22] Internet Society, [en línea]; Agosto de 2012 . Brief History of the Internet [consulta: 13 de Marzo 2015], disponoble: http://www.internetsociety.org/es/breve-historia-de-internet.

[23] Jesse James Garret, (2002). The Elements of User Experience. AIGA, New Riders

[24] Jorge García Molinero, (2013). Red informática corporativa para empresa comercializadora de electricidad. España: Universidad Abierta de Cataluña.

[25] José I. Castillo Velázquez, (2013). Infraestructura de la red de datos de telecomunicaciones en la UACM trabajo presentado en la Reunion Internacional de Otoño, ROC&C'2013, Acapulco, Gro.
[26] Jose I. Castillo Velázquez, (2014). Ingenería inversa parcial y simulación de la infraestructura de una red de datos MAN. Ciudad de México.
[27] José Santiago Merino, (2015). Introducción a la Investigación de Mercados. España: Universidad Compútense de Madrid
[28] José Zulú y Guevara Julca, (2002). Sistemas de comunicaciones orientadas a la descentralizacion de las entidades públicas del país. Lima: Perú. Universidad Nacional Mayor de San Marcos
[29] jQueryMobile, [en línea];febrero 2013,[consulta: 6 de noviembre 2016], disponible: http://jquerymobile.com
[30] Juan Diego Guachat. (2011), El gran libro de HTML5, CSS3 y JavaScrpit. España:MARACOMBO, S.A.
[31] Luis Mengual Galán. [en línea]; 2014. Mecanismos de Seguridad en Protocolo IPv6. [consulta: 26 de febrero del 2016] UPM, disponible: http://www.personal.fi.upm.es/~lmengual/ARQ_REDES/Seguridad_IPV6.pdf
[32] Luis Teodoro Aguirre Chacón y Huber Jhonn Siche Ricra , Diseño de una aplicación móvil para consulta academica de la FIIS-UTP., Peru, Peru: UTP, 2013.
[33] Ma. Eugenia Macías Ríos,[en línea]; 3 de septiembre 2012, Administración de Redes [cosnulta: 3 de diciembre 2015], de: Facultad de Ingeneria U.N.A.M, disponible: http://redyseguridad.fi-p.unam.mx/pp/maru/labpracticas/Planeacion_AdmonRedes_1_1.pdf
[34] PHP, [en linea];diciembre 2009,PHP 7 Manual,[consulta: enero 2016] disponible:https://php.net/manual/es/index.php
[35] Miguel González Pomposo (2013). Implementación de una red de datos por puerto extendido. México: U.N.A.M
[36] Mónica Ferrer Berbegal. (2006). Firewalls software: Estudio, instalación, configuración de escenarios y comparativa. Barcelona. U.P.C.
[37] Nicolás B. Vázquez, Carlos F. y Sonia F. Cid (2005). Redes de computadores y arquitecturas de comunicaciones, Supuestos practicos. Madrid: Pearson Prentice Hall
[38] Phonegapspain. [en llínea]; Julio de 2012. Frameworks compatibles con Phonegap. [consulta: Noviembre de 2015], de: Phonegapspain, disponible: http://www.phonegapspain.com
[39] Raúl Rosso [en línea]; Los cinco mejores emuladores de Android para pc 2016 [consulta: 7 de mayo de 2016], disponible: http://www.desarrolloweb.com/de_interes/themeroller-editor-plantillas-jquery-1861.html
[40] Robert E. Gunther, Yoram (Jerry) Wind y Paul Kleindorfer (1998) El reto de las redes. Reino unido: Wharton School Publishing.
[41] Sandra Sánches,[en linea]; 13 de octubre del 2012. El boom de las empresas de 'apps' en España, [consulta: 19 de marzo del 2016]. de: El mundo, disponible: http://www.elmundo.es/elmundo/2012/10/11/economia/1349945204.html
[42] Silvia Carrasco Usano. (2015) Analisis de la aplicación de la tecnología móvil en las empresas, Valencia, España: Universidad Politécnica de Valencia.
[43] Systems and software engineering - Vocabulary .(2010).ISO/IEC/IEEE 24765:2010, [consulta: 1 Abril de 2015], disponible : https://www.iso.org/obp/ui/#iso:std:iso-iec-ieee:24765:ed-1:v1:en
[44] Yiruan Rojo, [en línea]; 26 de septiembre del 2012. Capa de red del modelo OSI. [consulta: 26 de abril del 2016].De: Socializando redes, disponible: http://socializandoredes.blogspot.mx/2012/09/capa-de-red-del-modelo-osi.html

Anexos

Los siguientes documentos describen, los procedimientos seguidos para implementar, administrar y recuperar (en caso de fallas y problemas) toda la infraestructura de red diseñada para el laboratorio.

Estos documentos están referidos dentro del trabajo para que el lector tenga una comprensión técnica de cómo se realizó la instalación del cortafuego.

A-1. Instalación de pfSense y configuración básica

A continuación se describe los pasos iniciales para llevar a cabo la instalación de pfSense en un sistema de manera permanente para que el sistema sea funcional.

Se requiere un equipo estándar con arquitectura computacional, al cual se le borraran todos los datos del disco duro mediante un formateo a bajo nivel. Asimismo Se necesitaran diversas tarjetas de red para poder agregar estas interfaces al sistema operativo del cortafuego.

Como primer paso, es necesario preparar la instalación para lo cual se debe descargar de la página web www.pfSense.org oficial el fichero IOS (imagen de un CD) de la última versión estable de pfSense (v. 2.3.2). Una vez que se obtiene la descarga se realiza la grabación del ISO, se puede realizar en un CD o DVD para este caso se hizo en una memoria flash mediante UNetbootin herramienta para crear dispositivos USB en modo de arranque en vivo.

El proceso de la instalación se realiza de forma muy similar a las versiones más actuales de los sistemas operativos en base Linux, es decir, una vez que se inicia el arranque de la computadora, el ISO se carga directamente en la memoria RAM de la computadora (al reiniciar la computadora se perderá toda configuración realizada). Cuando termina esto, pfSense presenta al usuario algunas opciones y un temporizador de cuenta regresiva (ver figura 1-A). Si no se elige cualquier opción, por defecto se ejecutara la opción 1.

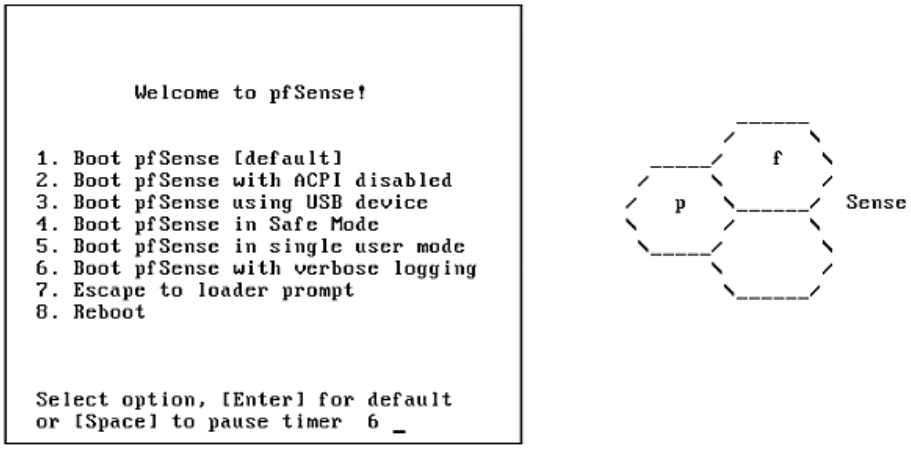


Figura 1-A

Esto comenzará a detectar el hardware del equipo y a configurar todos los controladores. A continuación se presiona “I” para instalar una copia de pfSense en la computadora.

Después como se muestra en la figura 2-A aparecerá la pantalla “**Configure Console**”, se pedirá seleccionar una opción para realizar la configuración inicial simplemente se pulsa en “**Accept these Settings**” para seguir adelante con el proceso de instalación de pfSense.

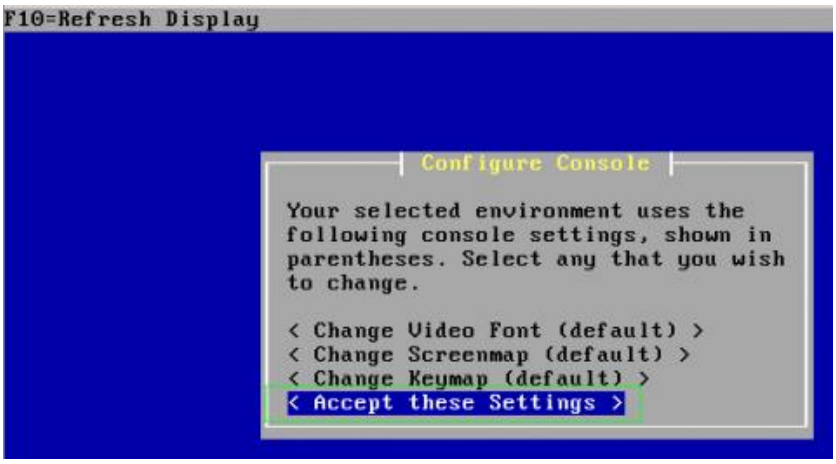


Figura 2-A

La instalación se realiza de manera personalizada, es decir, a partir de ahora se selecciona el disco en que se desea instalar pfSense, si se desea particionar el disco duro, formatear y finalmente en qué partición permanecerá el núcleo del sistema. Al finalizar estas configuraciones la computadora realiza un reinicio.

Al iniciar nuevamente el sistema (ya sin la memoria flash que contiene el ISO) se obtiene la siguiente pantalla, ver figura 3-A, que muestra las interfaces disponibles para llevar a cabo la configuración de la red.

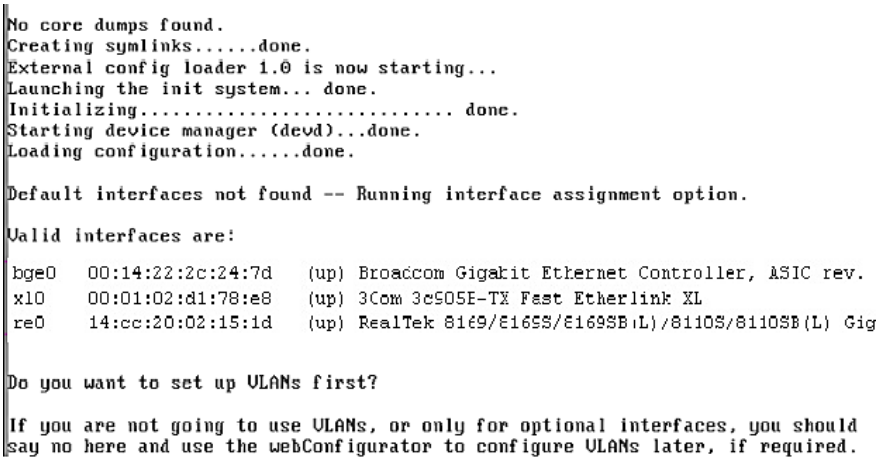


Figura 3-A

La configuración de la topología de red se realiza en este menú y se recomienda configurar sólo las interfaces de red local (LAN), zona desmilitarizada (DMZ) y acceso exterior (WAN) para más adelante

realizar una configuración más detallada de cada una de éstas. Lo primero que pregunta es acerca de configurar VLAN en caso de seleccionar que “no”, asignan interfaces de red directamente.

Para seleccionar por cuenta propia cada interfaz del menú de consola del sistema se escoge la opción 1 “**Assign Interfaces**” y se asigna la interfaz bge0 como la interfaz de red externa (WAN), re0 para la subred interna (LAN) y xl0 como interfaz para la subred desmilitarizada. Después de configurar las interfaces, se obtendrá el menú pfSense como se muestra en la figura 4-A.

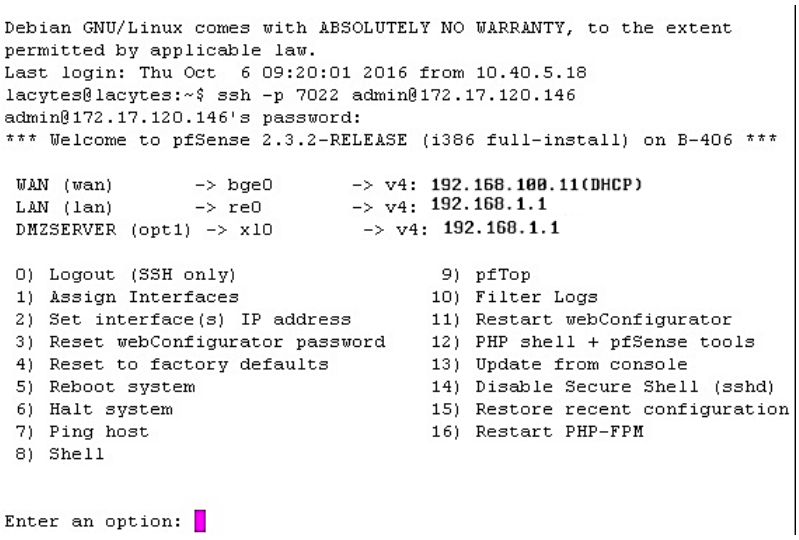


Figura 4-A

Se asignan las direcciones IP a la interfaz configurada como WAN, para poder acceder por web al sistema. Para ello, en el menú de la figura 4-A, se selecciona la opción 2 “**Set Interfaces(s) IP address**”.

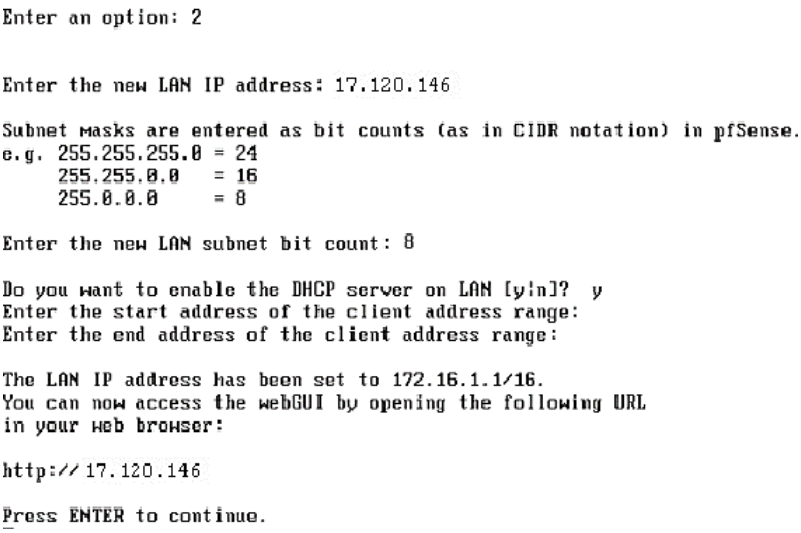


Figura 5-A

Una vez realizadas todas las configuraciones, se mostrara un enlace para lograr acceder a la interfaz “**Web Configurator**” por medio de un navegador web como se muestra en la figura 5-A. Desde otra computadora conectada al mismo segmento de subred LAN, se coloca la dirección IP en el navegador

web y se ingresa el Username como ‘admin’ y ‘pfSense’ como contraseña como se muestra en la figura 6-A.

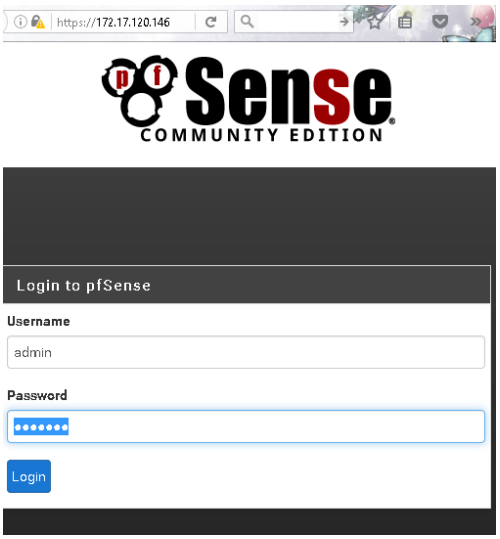


Figura 6-A

Una vez que se verifican las credenciales, el asistente de configuración aparece y este guiara a través del resto de la configuración de pfSense.

Como se observa en la figura 7-A en los primeros parámetros que se ingresan a través del portal web de son el “Hostname” que se asignara al cortafuego, el “Domain” y los servidores DNS (primario y secundario). Los DNS son proporcionados por la unidad de informática de la UACM, éstos ofrecen la posibilidad de navegar en internet y a través de la red de la universidad.

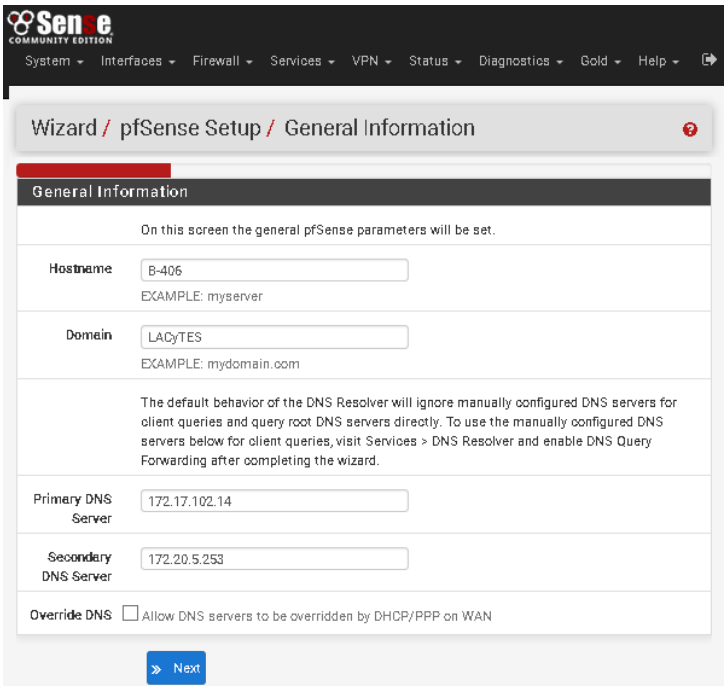


Figura 7-A

Después se continua con la configuración se piden los datos de un servidor de hora (Servidor NTP) con el que se sincronizara la hora del sistema. Se elige la zona horaria y se pasa a la siguiente configuración que trata acerca de la interfaz WAN. Si se cuenta con una conexión PPOE o se desea usar esta computadora como un router, es necesario elegir PPOE desde el menú desplegable, de lo contrario solo se tiene que seleccionar "static" e insertar la dirección IP, Gateway y Subnet Mask como se muestra en la figura 8-A.

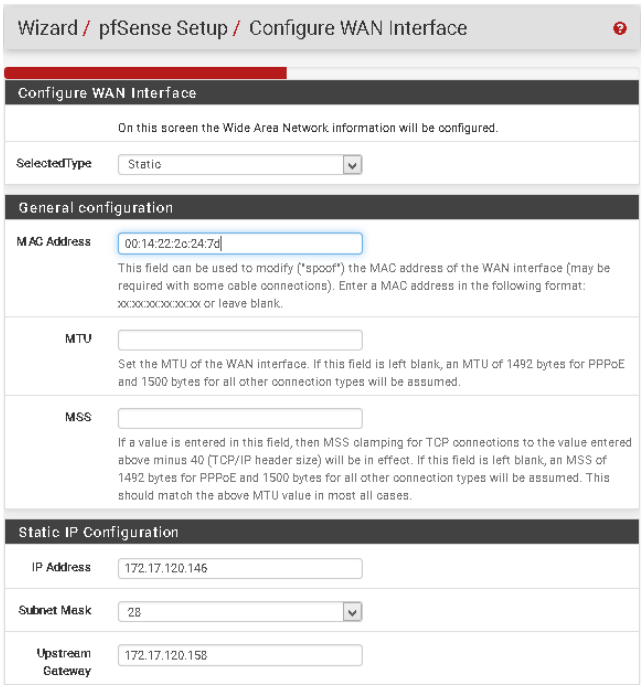


Figura 8-A

El siguiente paso es la configuración de la interfaz LAN, donde sólo se asignan lo parámetros de dirección IP y la máscara de subred como se muestra en la figura 9-A

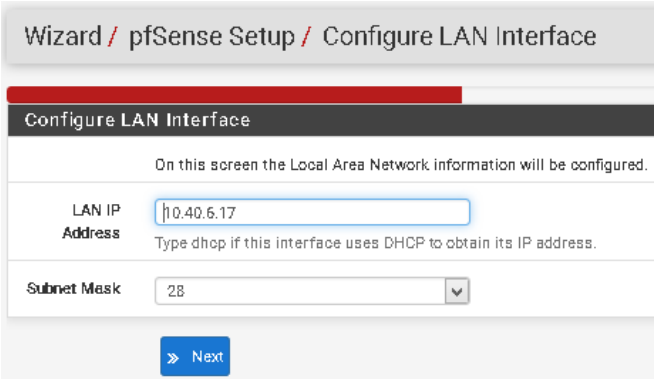


Figura 9-A

Por último se realiza el cambio de "Username" y "password", después el navegador web se volverá cargar y el sistema realizara los cambios. PfSense puede utilizarse como *router* o cortafuego con muchas características avanzadas tales como moldeador de tráfico de la red, equilibrador de carga y muchas más. Puede ser utilizado en pequeña escala a gran escala, dependiendo las características y el tamaño que se desea establecer la red.

A-2. Configuración de Interfaces y subredes en pfSense

Este documento describe la configuración de las interfaces de red para las subredes diseñadas y configuradas en el sistema.

Es requisito disponer del sistema pfSense instalado, tal como se vio anteriormente en el documento A-1 y la configuración de los equipos pertenecientes a la red que permitan realizar la topología diseñada.

Para este trabajo se implementó un sistema con 3 interfaces de red, las dos que se instalan por defecto son WAN y LAN, la tercera red de interfaz es para los servidores (DMZ).

La red WAN cuenta con una dirección IP estática la cual permite la comunicación con el exterior y posteriormente se configuraran las redes LAN y DMZ.

Como primer paso se necesita acceder al sistema donde se encuentra instalado pfSense, a través de la consola s de éste se configuran las direcciones de cada subred como se ve en la figura 10-A. Para esto del menú se selecciona "**Assign Interfaces**" y "**Set interfaces(s) Ip address**" y se configura de acuerdo a como se describió en el documente A-1.

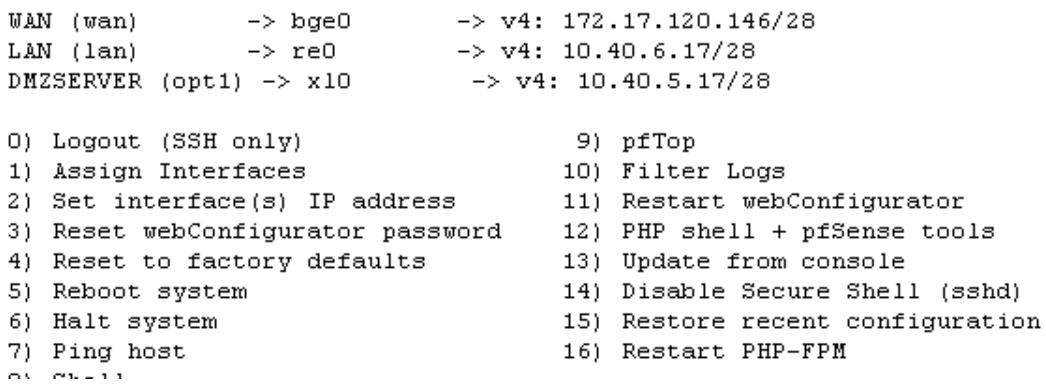


Figura 10-A Consola de pfSense con redes configuradas

Después, se accede desde un navegador web al sistema de pfSense, en la pestaña "**Interface**s" se selecciona cada una de ellas para configurar sus parámetros así como cambiar el nombre de la interfaz.

Cuando se selecciona una interfaz aparecen campos a llenar perfectamente con lo que se desea, tipo de dirección (estática, dinámica, PPPoE), dirección MAC de nuestra tarjeta de red, velocidad y modo dúplex, la dirección IP que tendrá la interfaz (solo si se seleccionó anteriormente que el tipo sea IP estática) y puerta de enlace para la interfaz de red.

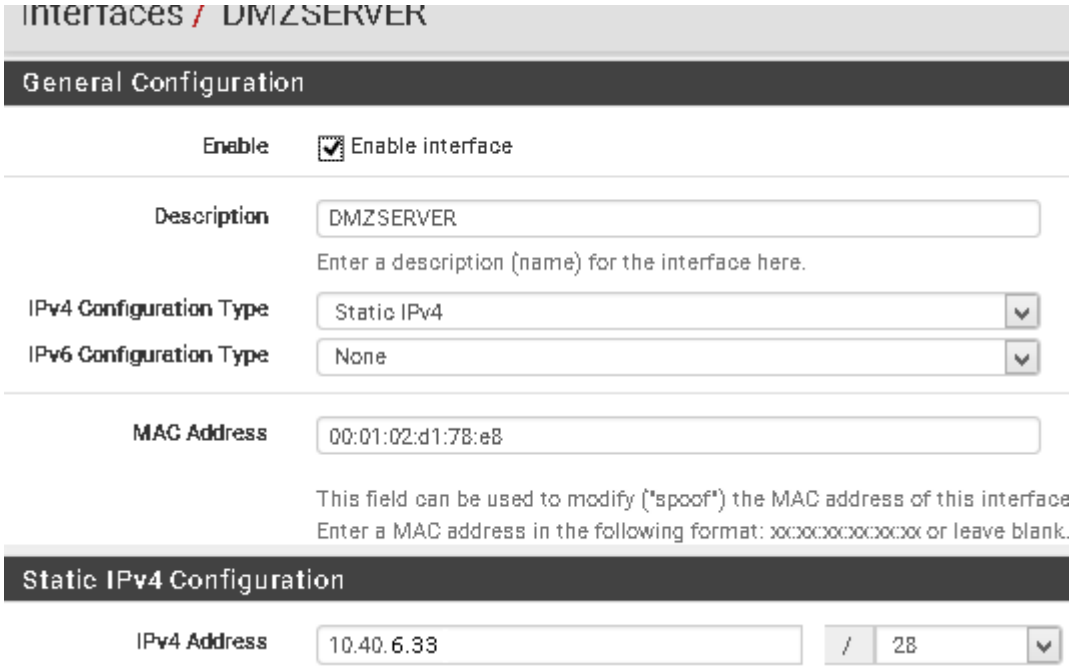


Figura 11-A Configuración de interfaz

Los resultados de la configuración de una interfaz se muestran en la figura 11-A. Así mismo de manera análoga se realiza la configuración en cada una de las interfaces. Para verificar la configuración y los parámetros insertados, se puede consultar desde el menú **"Status/Interfaces"** Donde se observa a detalle la configuración de cada red y una estadística del trafico enviado y recibido, ver figura 12-A.

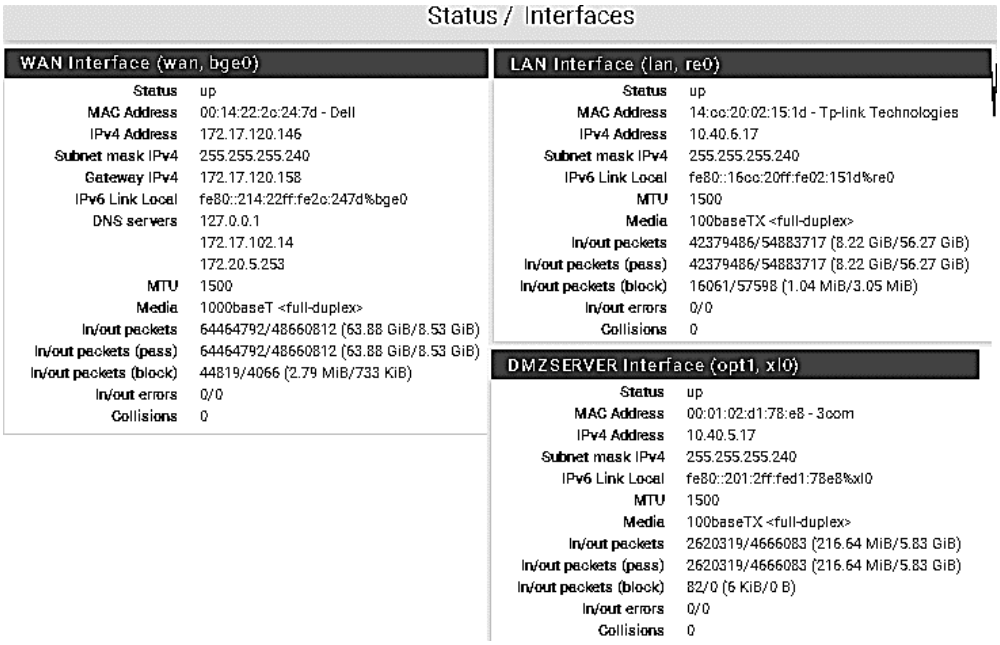


Figura 12-A

Hasta este punto el sistema se encuentra configurado para la conectividad de cada una de las redes, ahora es necesario implementar un sistema de concesión de direcciones automático (DHCP). Para hacer esto se accede al menú "**Services/ DHCP Server**".

Esta opción brinda valores para poder asignar a los clientes de la red una dirección IP. Además permite la creación de reservas de direcciones, las cuales permiten asociar una configuración con determinado cliente o equipo de cómputo, determinando su dirección MAC.

Para activar el servicio DHCP basta con seleccionar la casilla "**Enable DHCP server on LAN interface**" dirigirse a "**Range**" y seleccionar un rango de IP para que a los equipos pertenecientes a la red se les asigne una dirección IP dinámica. Esto se realiza de manera análoga en la interfaz DMZ.

Así con las interfaces ya configuradas y el servidor DHCP activado para cada subred se logra tener la topología diseñada para el laboratorio.

B-1 Instalación y configuración de Ripple

Ripple es una extensión que se instala en el navegador Google Chorme . Ésta funciona como un emulador de dispositivo multi-plataforma móvil y está hecho a medida para las pruebas móviles, recrea el entorno y sensores de un dispositivo móvil real dentro del navegador web. Cuenta con un sistema avanzado de simulación para probar aplicaciones basadas en PhoneGap, su instalación se realiza ingresando a la siguiente dirección web desde el navegador web mencionado, donde se moestrara el contenido que se muestra en la figura 1-B.

https://chrome.google.com/webstore/detail/ripple-emulator-beta/geelfhphabnejjhdalkjhgipohgpdnoc

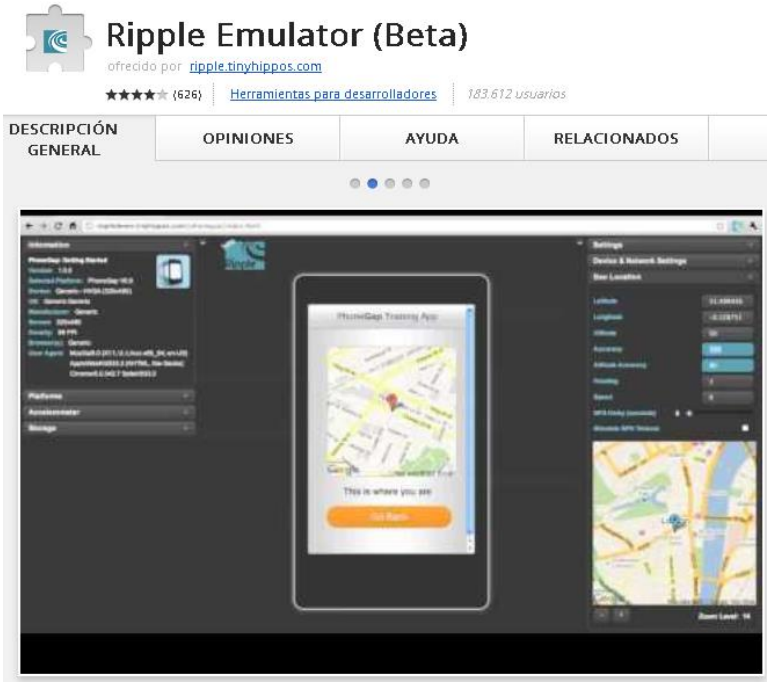


Figura 1-B

Después de añadir la extensión, aparecerá un botón con la imagen del logo de Ripple en la parte superior derecha del navegador, se presiona sobre el botón y se selecciona la opción *enable*, esta extensión funciona específicamente con paginas http:// y https://.
Como se muestra en la figura 2-B solicitará la plataforma que se desea utilizar para hacer la simulación. Esto se puede cambiar en cualquier momento, dependiendo del entorno que se desea emular.

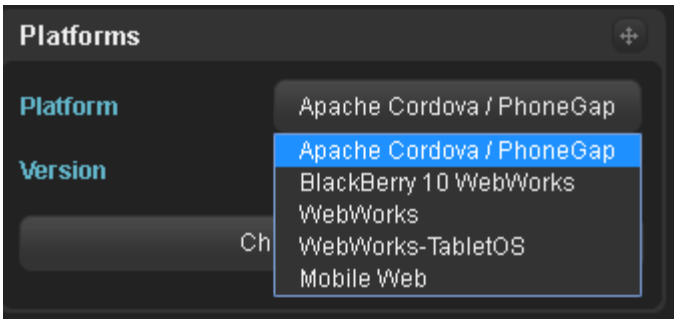


Figura 2-B

Al ejecutar la extensión se podrá apreciar lo siguiente:

- En panel de la parte izquierda del navegador se podrá observar la configuración del dispositivo y algunas de sus características, además de la orientación y la opción de "shaking". También se puede cambiar la plataforma.

- En la parte central del navegador, se observa la emulación del dispositivo escogido en el panelde la izquierda. La página que se visualiza debe mostrarse dentro del dispositivo emulado.

- En el panel de la derecha del navegador, se muestran algunas características respecto al tipo conexión del dispositivo, así como la posibilidad de simular geoposicionamiento.