UNIVERSIDAD DEL AZUAY

FACULTAD DE CIENCIAS DE LA ADMINISTRACIÓN

ESCUELA DE INGENIERÍA DE SISTEMAS

Sistema de autentificación con una herramienta basada en tres capas

Tesis previa a la obtención del título de Ingeniero de Sistemas

AUTOR: Jorge Lizandro Brito Ayabaca

DIRECTOR: Ing. Fernando Balarezo Rodríguez

Cuenca-Ecuador

2012

Dedicatoria

Dedico esta tesis a Gladys y Rubén mis Padres.

Juan, Luis y Dunia mis hermanos a quienes aprecio mucho.

Agradecimientos

Ing. Fernando Balarezo por compartirme sus experiencias, conocimientos y dirigirme en este trabajo.

Agradezco a mis amigos y familiares quienes con su apoyo, enseñanzas y paciencia han sido de gran apoyo en todo el transcurso de mi carrera.

Índice de contenidos

Resumen

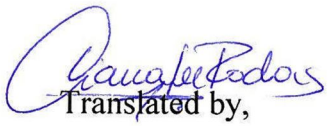
La presente tesis engloba todo el proceso de construcción de una aplicación empleando UML (Lenguaje Unificado de Modelado), con una orientación en tres capas, que permita la gestión de ingreso a redes inalámbricas, concesión de servicios de correo electrónico y registro en servidores DHCP mejorando la productividad de los empleados de una organización en la administración de sus servicios.

El uso en gran parte de software libre minimiza los costos de implementación y agrega flexibilidad a la herramienta. La implementación de esta solución sobre servicios muy conocidos como *sendmail*, *DHCP*, servicios que corren en la mayoría de las distribuciones de Linux, aumenta el espectro donde la herramienta es aplicable.

ABSTRACT

The present thesis encompasses the whole process for the construction of an application that employs UML (Unified Modeling Language). The application is oriented in three layers that allows managing and entering wireless networks, the concession of electronic mail services, and registration in DHCP servers in order to improve the productivity of the employees of an organization during the administration of their services.
The use of free software minimizes in large part the costs of implementation and adds flexibility to the tool. The implementation of this solution to well known services such as *sendmail, DHCP*, which appear in most of Linux distribution, increases the spectrum of application of the tool.

UNIVERSIDAD DEL
AZUAY
DPTO. IDIOMAS



Translated by,
Diana Lee Rodas

Introducción

Las tecnologías enfocadas a la presentación de contenidos a través de un navegador web hoy en día son muy conocidas, una de ellas lo constituye HTML (HyperText Markup Language), sin embargo las personas no somos entes pasivos que solamente consumimos información, sino que necesitamos por naturaleza interactuar. La web 2.0 se constituye como el primer peldaño para generar un web dinámica en la que podamos interactuar ingresando contenidos, comentando artículos, uniéndonos a las redes sociales, etc.

Para cada necesidad surgen nuevas tecnologías destinadas a satisfacerlas, pero el hombre ya no permanece en un solo lugar todo el tiempo, sino que se encuentra en movimiento entre ciudades, países e incluso continentes; de lo expuesto es justificado la necesidad de un sistema que sea accesible desde cualquier parte del mundo y que proporcione la seguridad adecuada, para estos requisitos han nacido las aplicaciones basadas en tres capas. En este esquema se hace uso de tres elementos fundamentales: una base de datos, un servidor de aplicaciones y el navegador web del usuario.

La base de datos como observaremos es nuestro repositorio de información, aunque en ocasiones también almacena la lógica del negocio como en nuestro caso. El objetivo de las tres capas es que el usuario final no tenga acceso directo a la base de datos, haciendo uso para ello del servidor de aplicaciones, equipo que recepta directamente las solicitudes de los usuarios.

El servidor de aplicaciones (Weblogic, Tomcat, Glassfish, etc) es la segunda capa y, en la mayoría de los casos contiene la lógica de negocio aunque no es obligatorio como ya hemos mencionado. La capa final de este esquema y también importante es el browser del usuario (Internet Explorer, Mozilla Firefox, Safari, etc) que es el que prácticamente presenta las interfaces del sistema. Estas interfaces netamente con código HTML no revelan el acceso a la base de datos ni codificación de la aplicación.

En el presente documento se expone precisamente una aplicación construida sobre un esquema en tres capas, que permite la gestión de servicios de una organización desde cualquier equipo informático conectado a la red.

Como se observará esta tesis se desarrolla en cuatro capítulos; el primero de ellos expone los requisitos del software a desarrollar, el análisis efectuado y el desarrollo de los distintos artefactos UML necesarios para describir claramente la aplicación. A partir de los primeros artefactos UML se define la arquitectura y esquema relacional en la base de datos Oracle 11g.

Por su parte el segundo capítulo expone los métodos utilizados para que la aplicación se pueda "comunicar" con servidores remotos. Se podrá observar la potencia del sistema operativo Linux para generar scripts y a través del cliente de Oracle, comunicarse con la base de datos ubicado en otro servidor físico. Servicios como SSH (Secure Shell) y SAMBA cobran suma importancia y su configuración adecuada es vital.

El capítulo tres adiciona un valor agregado al sistema en conjunto, podemos evidenciar la vulnerabilidad de las redes inalámbricas hoy en día y el método para poder "inyectarle" seguridad. Descubriremos la importancia del uso de protocolos seguros en la transmisión de información a través del espectro radioeléctrico, así como la importancia de una adecuada configuración de los equipos "Access Point".

El cuarto capítulo nos lleva a un mundo en donde descubrimos que no todo es integrable, especialmente si hablamos de software informático. Por ello en este capítulo encontraremos los principales problemas de comunicación entre equipos, transmisión de instrucciones; y cuáles fueron las alternativas de solución aplicadas.

Al concluir la lectura de este documento podremos entonces evidenciar las ventajas y desventajas de construir una aplicación para la gestión de múltiples servicios, y su implicación en el incremento de la productividad en las organizaciones actuales.

CAPITULO 1

Diseño y desarrollo del software para administración de cuentas de correo electrónico y accesos a la red inalámbrica.

1.1 Objetivos

1.1.1 Objetivo General

Diseñar y desarrollar un sistema autentificación que gestione cuentas de correo y accesos *wireless* que corran sobre S.O. linux, con una herramienta basada en tres capas.

1.1.2 Objetivos Específicos

- Apoyar al departamento de sistemas de una organización en la gestión de accesos a servicios basados en Linux
- Desarrollar una aplicación que administre la creación, modificación y eliminación de cuentas de correo electrónico y accesos a la red inalámbrica.
- Desarrollar e implementar una interfaz de comunicación entre la aplicación, y los servidores de correo y *Radius.*
- Configurar el servidor *Radius* y los *Access Point* para controlar los accesos a la red inalámbrica.
- "Levantar" el servidor de producción ante un fallo a partir del servidor de *backup* mediante la aplicación, guardando consistencia con la base de datos.

1.2 Análisis del software a desarrollar

Debido a la poca variabilidad de los requisitos se ha elegido el método lineal secuencial, como método de desarrollo. El método secuencial tiene una estructura que inicia por el análisis continúa por el diseño, codificación y termina en el desarrollo de la aplicación. La siguiente figura ilustra lo descrito.

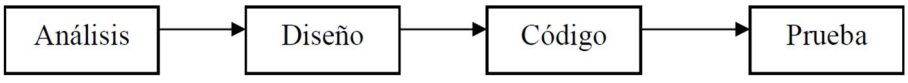


Figura 1-1: FUENTES, J. (2003) Realidad virtual aplicada al tratamiento del trastorno de lateralidad y ubicación espacial (Capítulo 2). Obtenido desde http://catarina.udlap.mx/u_dl_a/tales/documentos/lis/fuentes_k_jf/capitulo2.pdf.

1.2.1 Análisis de Requisitos

El análisis de requisitos de software es una de las fases más importantes e implica la recolección de requisitos tanto funcionales como no funcionales. En esta fase el o los desarrolladores serán alimentados de la mayoría de los requerimientos, los cuales les darán un panorama inicial del software que hay que desarrollar (parámetros de rendimiento, comportamiento e interconexión son definidos en esta etapa), por tal motivo es de suma importancia que se examine minuciosamente cada requisito.

En el análisis de requisitos se establece que debe hacer el sistema pero no como debe implementarse o como debe hacerlo. Para entender claramente que debe hacerse, primero se debe comprender que es un requisito. Requisito puede ser definido como:

- Condición o capacidad que un usuario necesita para poder resolver un problema o lograr un objetivo (IEEE).
- Condición o capacidad que debe exhibir o poseer un sistema para satisfacer un contrato, estándar, especificación, u otra documentación formalmente impuesta (IEEE).
- Una condición o capacidad que debe ser conformada por el sistema (RUP).
- Algo que el sistema debe hacer o una cualidad que el sistema debe poseer (Robertson - Robertson).

Entre los requerimientos para el desarrollo de la aplicación en estudio constan:

1.2.1.1 Requerimientos Funcionales

"Los requerimientos funcionales de un sistema describen lo que el sistema debe hacer. Estos requerimientos dependen del tipo del software que se desarrolle, de los posibles usuarios del software y del enfoque general tomado por la organización al redactar requerimientos. Cuando se expresan como requerimientos del usuario, habitualmente se describen de una forma bastante abstracta. Sin embargo, los requerimientos funcionales del sistema describen con detalle la función de este, sus entradas, salidas, excepciones, etc." Somerville. (Ingeniería de Software).

Los requerimientos funcionales de usuario levantados para el sistema en análisis se describen en el siguiente texto.

1.2.1.1.1 Requerimientos Funcionales del usuario

Se plantea el desarrollo de una aplicación que permita gestionar la autentificación de los usuarios hacia los servicios de correo y acceso inalámbrico de una organización, en la que se permitirá el ingreso a distintos tipos de operadores y que de acuerdo al rol que tienen asignado podrán realizar una u otra operación en el programa. Las operaciones varían de acuerdo al servicio al que tienen acceso a gestionar. Entre las operaciones a considerar en la gestión de correo por menú asignado están:

1.2.1.1.1.1 Menú Administrador

- Añadir, eliminar y modificar una cuenta de correo del servidor mail de acuerdo al dominio que se requiera.
- Agregar, eliminar y modificar las redirecciones entre las distintas cuentas de correo relacionado a uno o varios dominios internos del mismo usuario.
- Agregar, eliminar, y modificar las redirecciones entre distintas cuentas de correo relacionado a uno o varios dominios internos entre distintos usuarios.
- Agregar, eliminar, y modificar las redirecciones entre distintas cuentas de correo relacionado a uno o varios dominios externos entre distintos usuarios.
- Agregar, eliminar y modificar los mensajes de autorespuesta generados por cada dominio.
- Sincronizar los servidores de producción como el de standby.
- Modificación de la clave de correo electrónico. (Implica la modificación de la clave del resto de servicios).
- Deshabilitar una cuenta de correo de cualquiera de los dominios, el cambio de estado en este servicio implica el cambio de estado en el acceso a la red inalámbrica.
- Visualización de tablas del sistema.
- Modificación de los menús de distintos operadores que hacen uso de la aplicación.
- Visualizar un reporte en web de todos los usuarios registrados en el sistema con todos los datos de los mismos.

1.2.1.1.1.2 Menú Operador TI

- Agregar, modificar y eliminar las cuentas de correo en función de los servidores sobre el cual se le ha concedido el permiso.
- Agregar, eliminar y modificar las redirecciones entre las distintas cuentas de correo, en función de los permisos, relacionado a uno o varios dominios internos del mismo usuario.
- Agregar, eliminar, y modificar las redirecciones entre distintas cuentas de correo relacionado a uno o varios dominios internos entre distintos usuarios.
- Agregar, eliminar, y modificar las redirecciones entre distintas cuentas de correo relacionado a uno o varios dominios externos entre distintos usuarios.
- Agregar, eliminar y modificar los mensajes de autorespuesta generados por cada dominio cuando un usuario sale de vacaciones.
- Modificación de la clave de correo electrónico solo de los servidores sobre los cuales tiene permiso. (Implica la modificación de la clave del resto de servicios).
- Deshabilitar una cuenta de correo de cualquiera de los dominios, el cambio de estado en este servicio implica el cambio de estado en el acceso a la red inalámbrica
- Visualizar un reporte de los usuarios registrados, excepto de los usuarios que figuran como administradores.

1.2.1.1.1.3 Menú Operador

Visualizar reporte de los usuarios registrados en la aplicación, el reporte tiene limitaciones en cuanto a los datos visualizados, pues se excluye los siguientes campos:

1. Password actual
2. Tipo de usuario
3. Fecha de concesión
4. Fecha Expiración
5. Mail
6. Fecha de actualización
7. Prefijo telefónico
8. Extensión telefónica.

Adicionalmente el menú de operador podrá realizar la eliminación de un usuario en el sistema debido a su salida de la empresa. En la gestión del servicio de acceso inalámbrico a la corporación cada uno de los roles tienen las siguientes operaciones disponibles.

1.2.1.1.1.4 Rol Administrador

- Agregar, desactivar y modificar una cuenta para acceso a la red inalámbrica.
- Cambio de clave de acceso a la red inalámbrica (implica la modificación de la clave de todos los servicios del usuario).
- Visualización en pantalla de todos los usuarios que pueden ingresar a la red wireless.
- Administración de todas las tablas implicadas en la gestión de acceso a la red inalámbrica.
- Agregar, modificar y eliminar prefijos y extensiones telefónicas.
    - Nota: Ya que lo que se pretende es llevar un registro, la opción de agregar no implica la creación en centralillas de la extensión en cuestión.
- Modificar las asignaciones máximas de una extensión por usuarios.
- Visualizar listado completo de todos los prefijos y extensiones telefónicas de la corporación.
- Gestión de las tablas implicadas en el registro de prefijos y extensiones telefónicas.
- Asignación de extensiones telefónicas a uno o varios usuarios.

1.2.1.1.1.5 Rol Operador TI

- Agregar, desactivar y modificar una cuenta para acceso a la red inalámbrica.
- Cambio de clave de acceso a la red inalámbrica (Implica el cambio de la clave para todos los servicios).
- Visualización en pantalla de todos los usuarios que pueden ingresar a la red inalámbrica, excepto los usuarios que tienen el rol de administrador.
- Agregar, modificar y eliminar prefijos y extensiones telefónicas.
    - **Nota**: Ya que lo que se pretende es llevar un registro, la opción de agregar no implica la creación en centralillas de la extensión en cuestión.

- Modificar las asignaciones máximas de una extensión por usuarios.
- Visualizar listado completo de todos los prefijos y extensiones telefónicas de la corporación.
- Asignación de extensiones telefónicas a uno o varios usuarios.

1.2.1.1.1.6 Rol Operador

- Visualización de todos los usuarios que pueden ingresar a la red inalámbrica.
- No tiene opciones habilitadas para modificar los prefijos y las extensiones telefónicas.

Todas las cuentas de los usuarios tienen una fecha de expiración, si la fecha actual alcanza la fecha antes mencionada el acceso a la aplicación queda bloqueada para el usuario en cuestión.

La aplicación debe mostrar las opciones disponibles para los usuarios en un menú que se presente en la región lateral izquierda. Del requerimiento en el que se solicita el acceso a la aplicación por medio de roles, podemos intuir que el menú necesariamente será dinámico y que solamente el rol administrador podrá otorgar nuevas opciones a los roles de “Operador TI” y “Operador”.

La búsqueda de opciones en un menú se torna complicado cuando el mismo es demasiado grande, por lo tanto el menú debe presentar mecanismos de búsqueda sobre las opciones del que dispone. Contar con la mayor parte de la pantalla para realizar las actividades de trabajo diario es muy importante, es por ello que el menú de navegación debe facilitar la funcionalidad de ser ocultado en cualquier instante y conservar su estado hasta que el usuario defina lo contrario cuando lo visualice nuevamente.
Es necesario que la aplicación cuente con un mecanismo que me permita recibir notificaciones cuando existan inconsistencias entre los servidores de Backup y los de producción las mismas que deberán ser notificadas a los usuarios que estén registrados con el rol de administrador. Es necesario que el software tenga opciones que permitan extraer información de forma que se pueda tratar en hojas de Excel. La posibilidad de manipular los reportes generados por la aplicación es un requisito necesario. A más de

los reportes en formato web que se visualizan en el navegador es necesario que al ingresar a la aplicación se cuente con un Dashboard que muestre un resumen de las actividades realizadas, el formato del mismo debe ser definido por cronómetros.

Como hemos visto los requerimientos de usuario nos presentan un panorama de las necesidades que deben ser implementadas, sin embargo no proporcionan a detalle las funcionalidades del software, para disponer de una especificación detallada es necesario disponer de la especificación de requisitos funcionales del sistema. Es necesario plasmar el hecho de que un requerimiento funcional puede estar enfocado a describir lo que debe hacer el producto final, sin embargo también describirá las restricciones debido a políticas de la organización o a condiciones necesarias para que pueda interconectarse con otros sistemas.

Considerando lo expuesto, podemos entonces analizado los requerimientos funcionales levantados para el software analizado en este documento.

1.2.1.1.2 Requerimientos Funcionales del sistema

1. Añadir, eliminar y modificar una cuenta de correo.
    a. El operador debe seleccionar el usuario desde un popup que visualice los apellidos y nombres así como el nombre de usuario en el sistema registrado.
    b. El operador debe tener la posibilidad de registrar un nuevo usuario en la pantalla de inserción de nuevas cuentas.
    c. El operador debe tener la posibilidad de registrar un nuevo servicio de usuario en la pantalla de inserción de nuevas cuentas.
    d. El operador puede registrar en la misma pantalla el servicio de correo o de *DHCP*.
    e. El operador debe tener la capacidad de crear varias cuentas para un usuario registrado.
2. Añadir, eliminar y modificar las redirecciones entre distintas cuentas
    a. El operador debe poder seleccionar la cuenta de usuario a redireccionar desde un popup que contenga el nombre del empleado y su nombre de usuario registrado en el sistema.

b. El operador puede seleccionar del usuario elegido, el dominio el cual se pretende redireccionar.
c. El operador puede seleccionar en la pantalla de redirecciones cualquiera de los tres tipos disponibles (Redirección Interna, Redirección Externa, Redirección Interna hacia otra cuenta).
    i. Si se seleccionó redirección interna, se debe presentar todos los dominios en los cuales el usuario se encuentra registrado actualmente.
    ii. Para la redirección interna se tendrá únicamente que pasar de un listado a otro los dominios hacia los cuales se pretende ejecutar la redirección.
    iii. Si en caso es necesario realizar la redirección hacia otros usuarios entonces se requiere la presencia de un popup con el listado de los usuarios disponibles. Una vez seleccionado el usuario es obligatorio que se despliegue los dominios en los cuales está registrado, para que el operador pueda seleccionar a cuál de ellos redireccionar.
    iv. Para la opción de redirecciones externas el empleado será capaz de registrar una nueva cuenta de correo externo (Hotmail, gmail, etc.) a la cual se redireccionarán los correos. Si en caso el operador requiere eliminar la redirección externa, solamente se visualizaran las redirecciones creadas.
    v. Es Obligatorio la presentación de la fecha en la que se realiza la operación.

3. El único método de sincronización a emplearse es el full, el cual consiste en replicar el estado actual del servidor en producción con el servidor destino, los datos replicados son los usuarios, grupos, homes del sistema operativo.
    i. Para sincronizar los servidores será necesario especificar en primera instancia el dominio, luego de lo cual se desplegarán los servidores disponibles en el dominio, es importante que el

servidor de origen no sea igual al servidor de destino por cuanto no se estaría realizando ninguna sincronización.

ii. Si el proceso es largo se debe mencionar con un mensaje claro las acciones a llevarse a cabo.

4. Los mensajes de autorespuesta serán definidos al momento en el cual un empleado no haga uso de su dirección electrónica generalmente por motivo de vacaciones. Sin embargo los mensajes de autorespuesta obligatorios deben ser aquellos definidos por: Salida de Vacaciones, Salida de personal de la empresa.
    a. La modificación de la clave de usuario implica la modificación de la clave en todos los servidores en los cuales el empleado tenga registrado un servicio, ya sea este correo electrónico o *DHCP*. La modificación de la clave se realizará en la pantalla de mantenimiento de usuarios.
    b. En el mantenimiento de usuario se puede deshabilitar la cuenta de un usuario a nivel del sistema, lo que implica inhabilitar el acceso al servicio de *Wireless* y el acceso al sistema. En vista de que el usuario tiene correos en una bandeja de entrada no se eliminará la cuenta de mail.
5. Los mantenimientos de las tablas del sistema deben presentar un reporte con los datos actuales y opciones para que el usuario pueda crear, modificar, eliminar un registro de la misma. La pantalla de registro, modificación y eliminación se debe presentar sobre el reporte existente.
6. La modificación de las opciones por el menú se realizará en una grilla con los objetos disponibles en la aplicación.
7. La variación de las opciones disponibles por menú y tipo de usuario implican la modificación del estado de la pantalla, es decir, de acuerdo al tipo de usuario se habilitarán o deshabilitarán ciertas opciones visualizadas.
8. La adición, modificación y eliminación de extensiones telefónicas deberá realizarse en otra ventana, en donde se visualice un reporte de todos los usuarios con extensiones asignadas. El registro de una extensión en el sistema no implica la creación del mismo en la centralilla telefónica.

9. Para realizar búsquedas en el menú se dispondrá de un recuadro de texto en la parte superior del mismo en el cual se podrá digitar letras y en función de ellas se filtraran las opciones del menú.
10. El menú debe presentar opciones para que el usuario pueda desplegar o contraer todas las opciones.
11. Para ocultar el menú se hace necesario la disponibilidad de una barra lateral izquierda que al realizar el primer clic oculta el menú, al segundo clic vuelve a desplegar.
12. Es obligatorio la presencia un Dashboard en la pantalla inicial con cronómetros que visualicen:
    a. Número de actualizaciones en cuentas.
    b. Número de correos creados.
    c. Número de nuevas cuentas de usuario.
    d. Número de correos eliminados.
    e. Número de cuentas expiradas.
    f. Número de cuentas de usuario eliminadas.
13. Los cronómetros deben aumentar automáticamente al 100% su rango cuando los datos presentados alcanzan el 90% del mismo.
14. La pantalla inicial debe mostrar un reporte en web que visualice el estado de las opciones sobre los servidores ordenados por fecha.
    a. El usuario debe poder realizar operaciones de filtrado de información en el reporte presentado.
    b. Los datos del reporte podrán ser exportados a un archivo que sea abierto por Excel.
15. La finalización del contrato de un empleado implica la eliminación de los servicios hacia el mismo, por lo cual todas sus direcciones de mail desaparecerán pero de acuerdo a una fecha establecida en el sistema, hasta entonces los correos podrán ser redireccionados a la persona que será el remplazo.
    a. La redirección puede ser hacia una cuenta interna o hacia una cuenta externa.

b. Además es obligatorio la inclusión de un mensaje de autorespuesta en el cual el operador deberá establecer un texto indicando que la cuenta de usuario será eliminada.

16. Los reportes que el sistema presentará en formato PDF con opción de impresión son:
    a. Reporte de servicios de usuario por ubicación.
    b. Reporte de servidores en los cuales se encuentra registrado un usuario.
    c. Los reportes mencionados deberán presentarse en la web, y existirá alguna opción la cual permita al usuario imprimirlo en PDF.
17. Deberá existir un reporte web de los accesos a la red inalámbrica que el servidor Radius ha registrado.
18. El sistema deberá presentar una opción en la cual pueda registrar los Access Point que el servidor Radius permitirá acceso a la red.
19. Siempre se dispondrá de alguna opción que permita ingresar a la pantalla principal donde se encuentra el Dashboard.

1.2.1.2 Requerimientos No Funcionales

Los requerimientos no funcionales establecen lo que el software debe tener en consideración para su desarrollo. Según Ian Sommerville, los requerimientos no funcionales se pueden identificar como las características del que dispondrá la aplicación. Estas características pueden tener relación directamente con el producto, con la organización o con factores externos que influyen finalmente en la calidad del mismo. Ian Somerville grafica mediante una estructura en árbol esta clasificación.

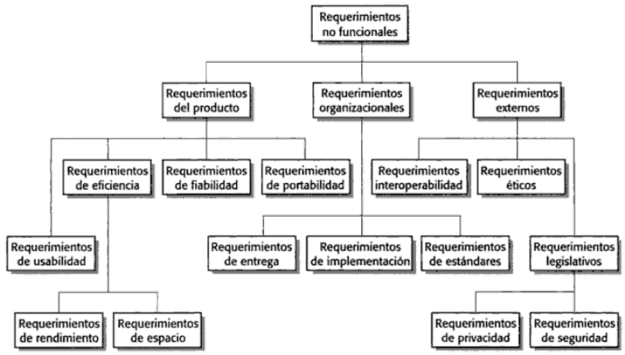


Figura 1-2: SOMMERVILLE, I. (2005). INGENIERÍA DEL SOFTWARE . Obtenido desde http://books.google.com.ec/books?hl=en&lr=&id=gQWd49zSut4C&oi=fnd&pg=PA1&dq=requerimientos+funcionales+y+no+funcionales+de+software&ots=s531ppzCtf&sig=YCLO_nboPgEjscy9MnOwjeRXvU#v=onepage&q=requerimientos%20funcionales%20y%20no%20funcionales%20de%20software&f=false.

Podemos decir también que los requerimientos no funcionales insertan restricciones físicas sobre los requerimientos funcionales. Por ejemplo si un requerimiento funcional es que la aplicación opere sobre un ambiente web, este podría estar restringido si uno de los requerimientos no funcionales establece que solamente se podrá disponer de un equipo el cual no soporta alta transaccionalidad para operaciones web.
Entre los requerimientos no funcionales más habituales constan:

- Usabilidad
    - La facilidad con la que una persona puede hacer uso de la herramienta. La disponibilidad de documentación, de ayuda en línea, las funciones intuitivas. Sin embargo no es una medición cuantitativa por el simple hecho de que todas las personas tenemos distintas percepciones de que es usable y que no.
- Confiabilidad
    - Frecuencia de fallas y tiempo de recuperación, estabilidad del sistema.
- Performance
    - Tiempo de respuesta, porcentaje de procesamiento, capacidad de carga.
- Mantenibilidad
    - Tiempo de respuesta en el que un problema es solucionado.

La criticidad de los requerimientos no funcionales es un factor clave que determina si una aplicación será o no usada, y que por lo general se acuerdan en un contrato de forma textual.

1.2.2 Diseño de software

La etapa de diseño es la fase de ingeniería de software que modela los requerimientos levantados en la especificación de requisitos sean estos funcionales como no funcionales. Si el diseño modela los requisitos levantado en la fase de análisis, entonces, el diseño se constituye como la primera etapa en el desarrollo de un software de ingeniería.

Es conveniente que se cite lagunas definiciones que posibilitarán recordar el concepto de diseño. Según Taylor la etapa de diseño se define como: “Proceso de aplicar distintas técnicas y principios con el propósito de definir un dispositivo, proceso o sistema con los suficientes detalles como para permitir su realización física”.

De las definiciones descritas se puede suponer que la etapa de diseño juega un papel muy importante en la calidad del software final obtenido, entonces se puede decir que el diseño es uno de los primeros elementos técnicos en el producto final obtenido que detallan la arquitectura del software final obtenido. La siguiente figura describe lo descrito:

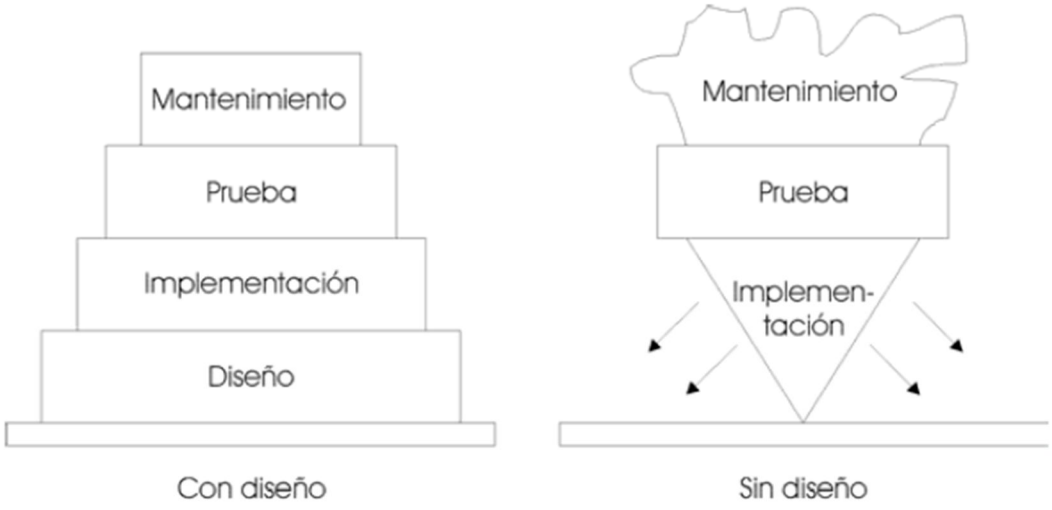


Figura 1-3: TORRES, M. (2006). Fundamentos del diseño. Recuperado de http://indalog.ual.es/mtorres/LP/FundamentosDiseno.pdf.

“Sin diseño, nos arriesgamos a construir un sistema inestable, un sistema que falle cuando se realicen pequeños cambios, un sistema que sea difícil de probar, un sistema cuya calidad no pueda ser evaluada hasta más adelante, cuando quede poco tiempo y ya sea haya gastado mucho dinero.”

“El principio de la sabiduría de un programador está en reconocer la diferencia entre obtener un programa que funcione y uno que funcione correctamente”. Michael A. Jackson

Entre los artefactos utilizados para modelar el software en análisis, se encuentran aquellos que se enfocan en el área estática del sistema o también llamada parte estructural, mientras que otros se orientan netamente a la parte dinámica. Entre los diagramas que modelan el estado estático del sistema se encuentran:

1.2.2.1 Diagrama de clases

Los diagramas de clases permiten describir la estructura estática de un sistema, permitiéndole al usuario final evidenciar los requerimientos funcionales que especificó en la etapa de análisis de requisitos. A partir de un diagrama de clases podemos conocer cuáles serán los objetos implicados y sus relaciones tanto de dependencia, generalización, composición y asociación con otros objetos.

Existen diferentes formas de representar las clases, en nuestro caso hemos adoptado la especificación en la que definimos el nombre de la clase, encerrado en un rectángulo, las relaciones y el número de instancias posibles de cada una de ellas. Como veremos los objetos son parte fundamental para la definición de actores (operadores, sistema, servidores, etc.) y los elementos base con los que se construye los diagramas de componentes muy útiles en la visualización del funcionamiento general del sistema.

Para graficar los diagramas de clases hacemos uso de los símbolos descritos a continuación:

Rectángulo: Está dividido en tres secciones donde la primera contiene el nombre de la clase, la segunda sección encierra los atributos y la última sección tiene los eventos que ejecuta la clase.

Flechas dirigidas: Representan generalización, composición, dependencia, asociación

-  Asociación
-  Dependencia
-  Generalización
-  Composición

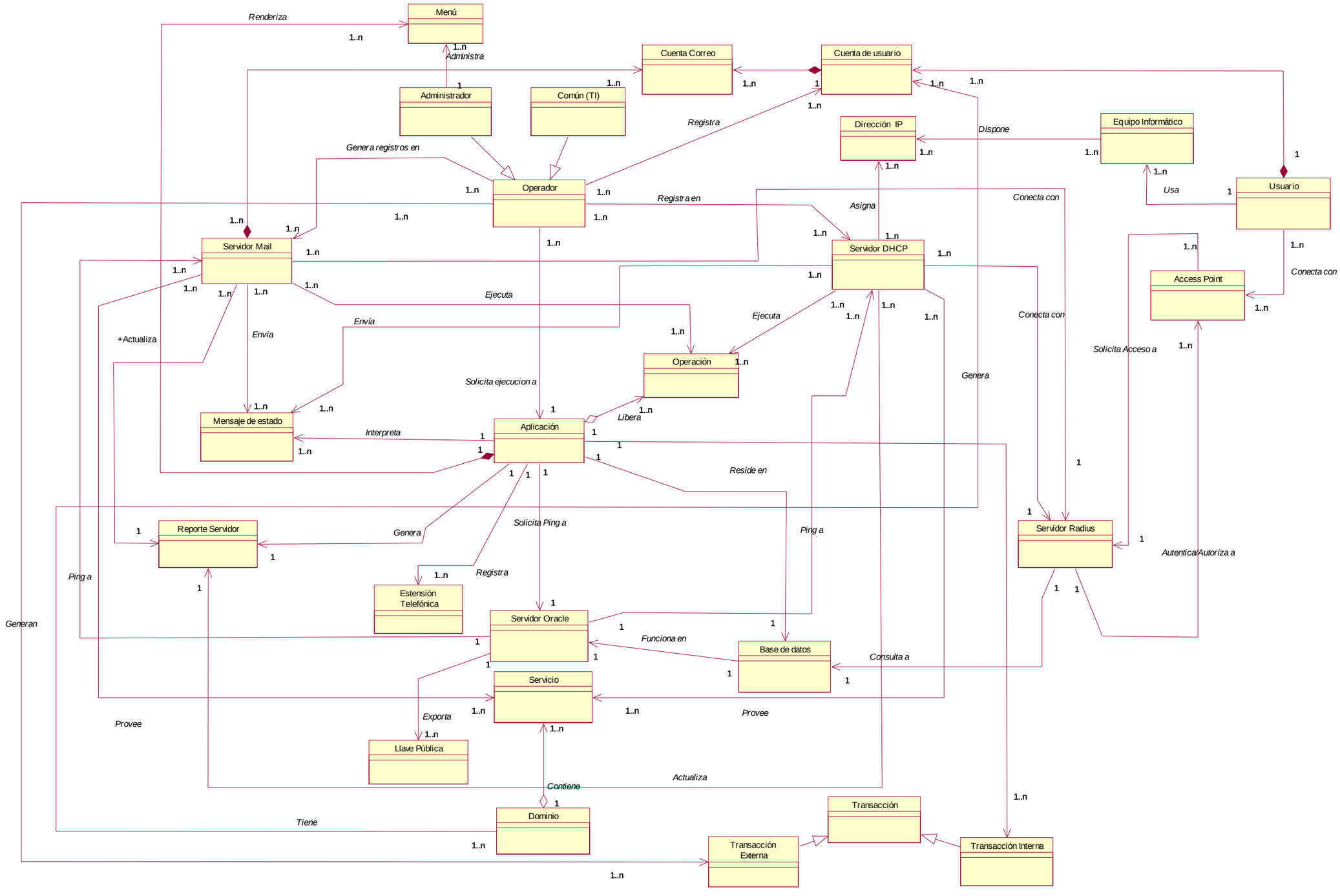

Menú
Renderiza
1..n
1..n
Administra
Cuenta Correo
Cuenta de usuario
Administrador
Común (TI)
1..n
1..n
1
1..n
Registra
Dirección IP
Dispone
Equipo Informático
Genera registros en
1..n
1..n
Usa
1
Usuario
Operador
1..n
Registra en
Asigna
Conecta con
1..n
1..n
1..n
1..n
1..n
Servidor Mail
Servidor DHCP
1..n
1..n
1..n
1..n
1..n
1..n
1..n
Access Point
1..n
1..n
Conecta con
Ejecuta
Envía
Ejecuta
Conecta con
+Actualiza
1..n
Operación
1..n
Solicita Acceso a
1..n
Solicita ejecucion a
Genera
1..n
1..n
Mensaje de estado
1
Aplicación
Libera
1..n
Interpreta
1..n
1
1
1
1
1
1
Reside en
1
1
Servidor Radius
1
Autentica/Autoriza a
Genera
Solicita Ping a
Ping a
Reporte Servidor
1
1
1
Ping a
1..n
Registra
Estensión Telefónica
1
Servidor Oracle
Generan
1
1
1
1
Funciona en
Base de datos
Consulta a
1
1
Servicio
1..n
1..n
Provee
Provee
Exporta
1..n
Llave Pública
Actualiza
Contiene
1
Transacción
1..n
Dominio
Tiene
1..n
Transacción Externa
Transacción Interna
1..n

Figura 1-4

A pesar de que el diagrama de clases presenta una representación de los requerimientos funcionales, no visualiza cual es la interacción de los actores con el sistema. Para visualizar la interacción, es necesario hacer uso de un diagrama llamado: “Diagrama de casos de uso”.

1.2.2.2 Diagrama de casos de uso

Los diagramas de caso de uso permiten modelar el funcionamiento del sistema vista desde los usuarios. Es muy utilizada para facilitar la comunicación con los clientes pues les muestra que hará la aplicación. “Un caso de uso es una forma de expresar cómo alguien o algo externo a un sistema lo usa. Cuando decimos “alguien o algo” hacemos referencia a que los sistemas son usados no sólo por personas, sino también por otros sistemas de hardware y software.

Por ejemplo, un sistema de ventas, si pretende tener éxito, debe ofrecer un servicio para ingresar un nuevo pedido de un cliente. Cuando un usuario accede a este servicio, podemos decir que está “ejecutando” el caso de uso “ingresando pedido.”
Los casos de uso generados para la aplicación descrita en este documento son:

Tabla 1-1: Verificación credenciales sistema

CU-001	**Verificación credenciales sistema**	
Instancia de usuarios Participantes		Operador, Operador No TI
Descripción		El sistema debe autorizar el acceso al mismo.
Precondiciones		
Flujo de eventos	1	El operador ingresa sus credenciales.
	2	El sistema valida los datos ingresados y la vigencia de la cuenta.
	3	El sistema presenta un menú lateral izquierdo, un Dashboard inicial y el estado de las últimas operaciones realizadas en cada servidor.
Excepciones		

Tabla 1-1

Continuación

	2	Si los datos ingresados no son válidos o la cuenta se encuentra expirada entonces el sistema despliega un mensaje de error notificando que el operador no está autorizado. El sistema dirige al operador a la pantalla de ingreso para que digite nuevamente sus credenciales.
Importancia		Vital

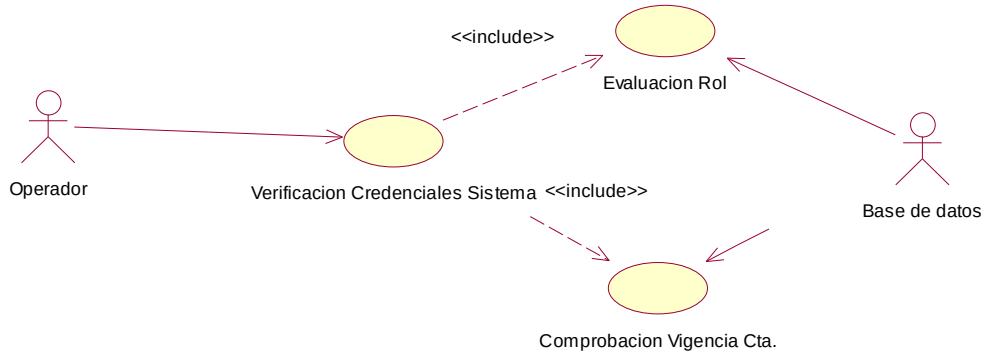


Figura 1-5

Tabla 1-2: Actualizar Usuario

CU-002		**Actualizar Usuario**
Instancia de usuarios Participantes		Operador Administrador, Operador TI
Descripción		Se modifica los datos de un usuario registrado en el sistema
Precondiciones		El usuario debe haberse logueado. Caso de uso CU-001
Flujo de eventos		
	1	En el menú se selecciona la opción Usuarios.
	2	Se presenta un listado de los usuarios, donde en cada registro se da la opción de edición.
	3	El operador puede modificar los datos del usuario.

Tabla 1-2:

Continuación

	4	El sistema verifica si el operador tiene permisos sobre los servidores asociados al usuario registrado.
	5	El sistema verifica conexión con cada servidor que tendrá que ser modificado.
	6	El sistema verifica si la fecha de concesión es menor en tres días a la fecha de expiración.
	7	El sistema ingresa una transacción y bloquea la tabla de transacciones para evitar incoherencias en el registro, por actividades concurrentes de otros usuarios.
	8	El sistema actualiza el usuario en cada servidor.
	9	Actualización de todas las redirecciones a cuentas internas del mismo usuario, redirecciones internas a otras cuentas y redirecciones a cuentas externas.
	10	Actualización de asignaciones DHCP.
	11	El servidor de mail actualiza el estado de la transacción generada por el sistema a un estado completado.
Excepciones		
	4	Si el operador no tiene permiso para modificar los servidores asociados al usuario, el sistema devuelve un error sin modificar el usuario en sus datos sensibles: usuarios_sistema y password, sino solo los datos adicionales de registro.
	5	Si el operador tiene permiso sobre los servidores asociados al usuario entonces verifica conexión a los mismos y si al menos uno de ellos se encuentra sin conexión, entonces no modifica el registro en sus datos sensibles.
	7	Si la tabla de transacciones se encuentra actualmente bloqueada, entonces el sistema realiza la espera por el

		recurso. Si la espera por el bloqueo es de más de un usuario entonces todas las actividades a partir del segundo bloqueo se descartan.
	10	Si existe un corte en la red o el sistema queda fuera del aire, el servidor falla en la actualización y se le presenta al usuario coloreado el servidor con problemas en el reporte inicial.
	10	El usuario puede cancelar la operación en cualquier momento, con un botón de cancelar.
Importancia		Importante

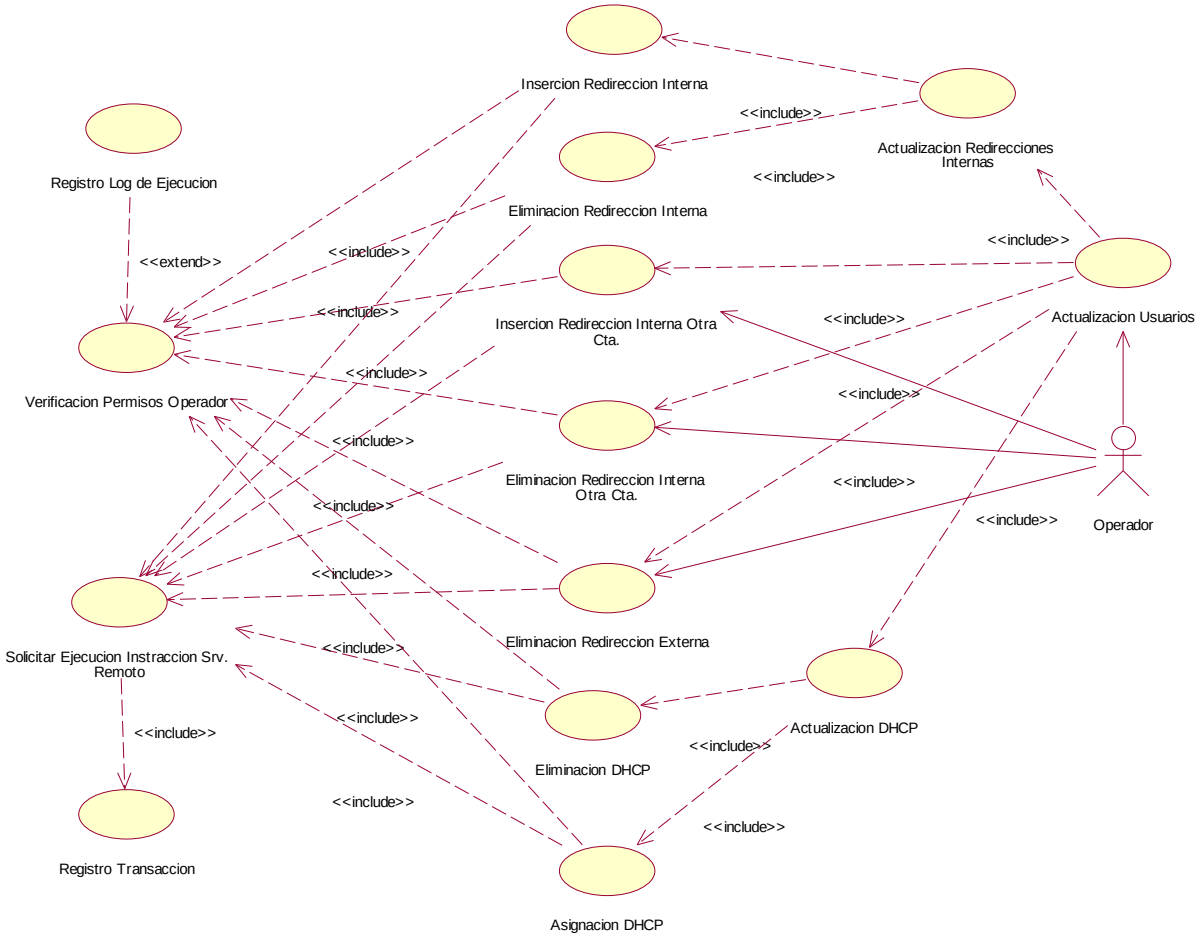


Figura 1-6

Tabla 1-3: Eliminación de un usuario

CU-003		**Eliminación de un usuario**
Instancia de usuarios		
Participantes		Operador, Operador No TI
Descripción		El sistema debe permitir la eliminación del usuario de modo que todos los servidores de correo asociados queden coherentes.
Precondiciones		El usuario debe haberse logueado. Caso de uso CU-001
Flujo de eventos		
	1	El operador selecciona la opción "Usuarios" del menú lateral izquierdo, pantalla en la cual encontrara el botón de eliminar
	2	El usuario deberá hacer clic en el botón para eliminar el usuario.
	3	El sistema notifica al operador que esta acción eliminara el usuario de los servidores asociados.
	4	El sistema verifica si existe una transacción en curso.
	5	El sistema genera una nueva transacción por la operación solicitada.
	6	El sistema verifica que el operador tenga permisos sobre los servidores asociados al usuario que pretende eliminar.
	7	El sistema valida la conexión de cada servidor para proceder con la eliminación del usuario.
	8	Se elimina las redirecciones a cuentas internas, redirecciones a cuentas de otros usuarios y a cuentas de correo externo. Se elimina las asignaciones DHCP.
	9	El sistema genera un log por la operación realizada.
	10	El sistema visualiza un mensaje de que el usuario ha sido eliminado exitosamente.
	11	Los servidores implicados actualizan el estado de la operación a "COMPLETADO"

Excepciones		
	3	Si el operador no acepta la notificación presentada entonces el sistema no realiza acción.
	4	Si el sistema se encuentra con una transacción en curso entonces generara un espera hasta que se libere el recurso, a partir de la segunda transacción en espera todas las solicitudes serán descartadas.
	6	Si el sistema valida que el operador no tiene permisos sobre los servidores implicados en la operación, entonces presenta un mensaje de error visualizando el servidor inalcanzable.
	7	Si alguno de los servidores que serán modificados no se encuentran al alcance, entonces el sistema visualiza un mensaje con la dirección IP del servidor implicado.
	11	Si existe un corte en la red o el sistema queda fuera del aire, el servidor falla en la actualización y se le presenta al usuario coloreado el servidor con problemas en el reporte inicial.
Importancia		Importante

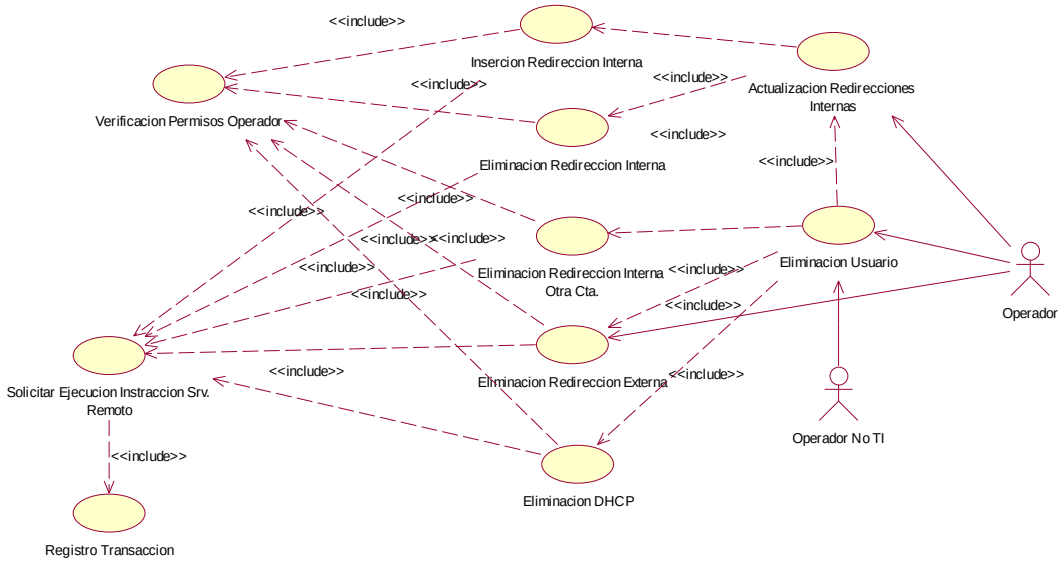


Figure 1-7

Tabla 1-4: Creación de cuenta de correo

CU-004		**Creación de cuenta de correo.**
Instancia de usuarios Participantes		Operador
Descripción		El sistema debe permitir la creación de una nueva cuenta de correo
Precondiciones		El usuario debe haberse logueado. Caso de uso CU-001
Flujo de eventos		
	1	El sistema presenta un menú en donde consta la opción de "Concesión de servicios".
	2	El operador deberá seleccionar la opción "Asignación Servicio Usuario"
	3	El sistema presenta una pantalla en donde el operador debe seleccionar el usuario
	4	El operador selecciona el usuario
	5	El sistema visualiza la opción de selección de servicio.
	6	El operador selecciona el servicio y puede además crear uno nuevo.
	7	El sistema visualiza los dominios disponibles para el servicio seleccionado.
	8	El operador deberá seleccionar la opción de "Actualizar Asignaciones" con lo cual se creara o eliminara las cuenta dependiendo de la lista de dominios seleccionados.
	9	El sistema validará si existe una transacción en curso para el usuario implicado.
	10	El sistema valida que se dispone de permisos para afectar a los servidores que tienen relación con los dominios seleccionados.
	11	El sistema valida que los servidores se encuentren al alcance.

Tabla 1-4:

Continuación

	12	El sistema genera un registro en el log por la operación realizada.
	13	El sistema crea o elimina las cuentas de correo de acuerdo a los dominios seleccionados.
	14	Los servidores de mail implicados actualizan la transacción generada.
	16	El sistema presenta un mensaje con el resultado exitoso en la creación de la cuenta.
Excepciones		
	11	Si existe una transacción en curso para el usuario en el mismo dominio, el sistema generara una demora hasta que el usuario sea liberado. Para un mismo usuario en un mismo dominio y en un mismo servicio solo pueden existir dos transacciones en espera.
	12	Si el operador no dispone de permisos entonces el sistema visualiza un mensaje con la IP del servidor al cual no tiene acceso.
	13	Si el servidor a ser afectado por la operación solicitada no se encuentra al alcance del sistema, entonces no se realiza la operación, y se presenta un mensaje con la dirección IP del mismo.
	16	Si existe un corte en la red o el sistema queda fuera del aire, el servidor falla en la actualización y se le presenta al usuario coloreado el servidor con problemas en el reporte inicial.
Importancia		Importante

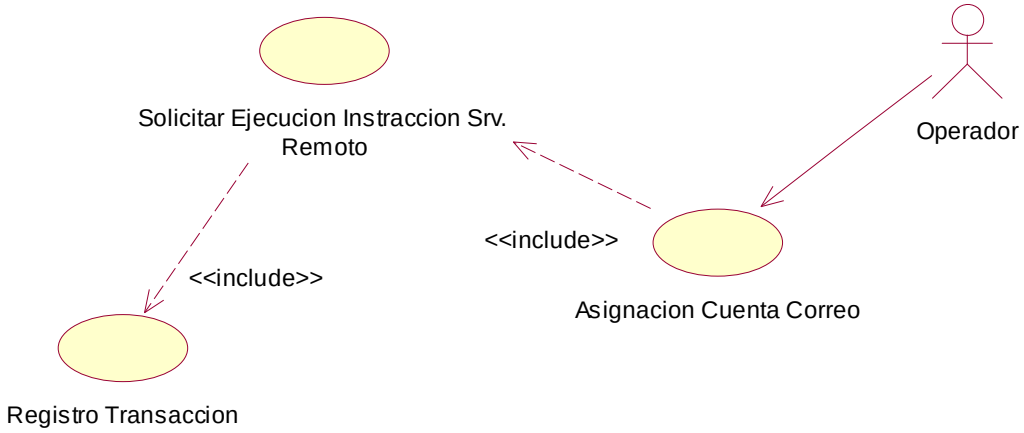


Figura 1-8

Tabla 1-5: Actualización redirecciones a cuentas internas del mismo dominio

CU-005	**Actualización redirecciones a cuentas internas del mismo usuario**
Instancia de usuarios Participantes	Operador
Descripción	El sistema debe permitir la inserción de redirecciones entre cuentas del mismo usuario.
Precondiciones	El usuario debe haberse logueado. Caso de uso CU-001
Flujo de eventos	
	1.-El sistema visualiza un menú con la opción de redirecciones.
	2.-El operador selecciona la opción de redirecciones
	3.-El sistema genera una pantalla donde se debe seleccionar el usuario a redireccionar.
	4.-El operador selecciona el usuario y el dominio que desea redireccionar.
	5.-El sistema presenta la opción de redirección interna.
	6.-El operador selecciona la opción de redirección interna.

Tabla 1-5: Continuación

	7.-El sistema genera el listado de dominios en los cuales el usuario se encuentra registrado.
	8- El operador selecciona los dominios que se desea redireccionar.
	9.- El operador selecciona la opción de actualizar redirección.
	10.- El sistema verifica si existe una transacción para el mismo usuario, dominio y servicio.
	11.-El sistema valida que el operador tenga permisos para realizar operaciones sobre los servidores asociados a los dominios seleccionados.
	12.- El sistema valida que exista conexión entre el sistema y los servidores a ser afectados.
	13.- El sistema genera un log con las operaciones realizadas.
	14.- Los servidores afectados actualizan el estado de la transacción, una vez completada la operación.
Excepciones	
	1.- Si el usuario no tiene ningún dominio registrado entonces el sistema no presenta las opciones de redirección y la operación termina en esos momentos.
	10.- Si el sistema encuentra que existe una transacción en curso, para el mismo usuario, dominio y servicio entonces genera una espera hasta que la transacción en curso termine su operación.
	11.- Si el operador no tiene permisos sobre los servidores a ser afectados no se procede con la solicitud, y el sistema presenta un mensaje de error con la IP del servidor implicado.
	12.- Si el sistema determina que no se tiene acceso a los

	servidores a ser afectados por la operación, entonces presenta un mensaje de error con la dirección IP del servidor implicado.
	13.- Si existe un corte en la red o el sistema queda fuera del aire, el servidor falla en la actualización y se le presenta al usuario coloreado el servidor con problemas en el reporte inicial.
Importancia	Importante

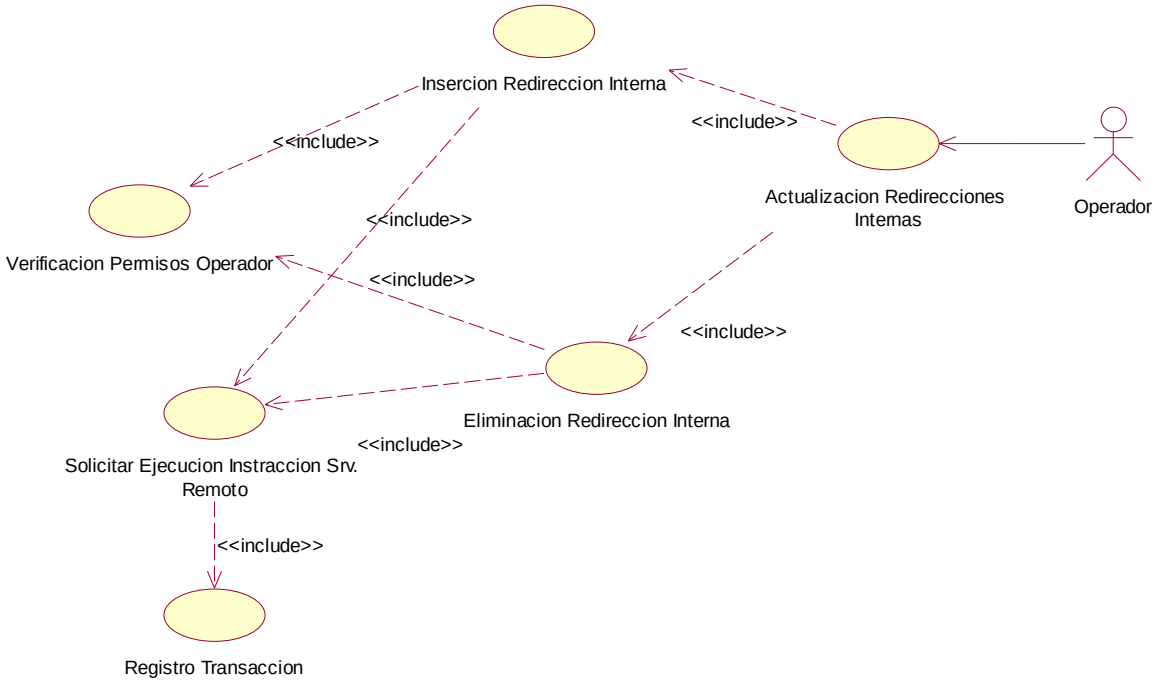


Figura 1-9

Tabla 1-6: Inserción de una redirección interna hacia otra cuenta

CU-006		**Inserción de una redirección interna hacia otra cuenta.**
Instancia de usuarios Participantes		Operador
Descripción		El sistema debe permitir el ingreso de redirecciones entre las cuentas internas de los usuarios.

Tabla 1-6:

Continuación

Precondiciones		El usuario debe haberse logueado. Caso de uso CU-001
Flujo de eventos		
	1	El operador selecciona del menú lateral izquierdo la opción de "Redirecciones".
	2	El sistema presenta una pantalla en la cual el operador debe seleccionar el usuario y el dominio que desea redireccionar.
	3	El operador selecciona el usuario y el dominio.
	4	El sistema visualiza un listado de usuario al cual se pueden redireccionar.
	5	El operador selecciona el usuario destino y el dominio que tenga asociado.
	6	El sistema presenta el botón para crear redirecciones internas a otros usuarios.
	7	El sistema valida si existe una transacción en curso.
	8	El sistema valida si el operador tiene permisos para realizar operaciones en los servidores asociados al dominio.
	9	El sistema valida si los servidores a ser afectados se encuentran al alcance del mismo.
	10	El sistema genera un log por la operación realizada.
	11	El servidor de mail actualiza el estado de la transacción ha completado.
	12	El sistema presenta un mensaje de que la operación se ha realizado exitosamente.
Excepciones		
	3	Si el usuario seleccionado no dispone de dominio el cual se va a redireccionar, entonces termina la operación, ya que el sistema no visualiza el resto de opciones

		necesarias.
	7	Si existe una transacción en curso para el mismo usuario, dominio y servicio entonces el sistema genera una demora hasta que termine la misma. En caso de existir más de dos transacciones en espera para el mismo usuario, dominio y servicio entonces el sistema descarta transacciones a partir de la segunda.
	8	Si el operador no dispone de permisos sobre todos los servidores a ser afectados entonces el sistema presenta un mensaje y no realiza la operación solicitada.
	9	Si el sistema no tiene conexión con los servidores de mail entonces cancela la operación y presenta un mensaje con el servidor que se encuentra con problemas.
	12	Si existe un corte en la red o el sistema queda fuera del aire, el servidor falla en la actualización y se le presenta al usuario coloreado el servidor con problemas en el reporte inicial.
Importancia		Importante

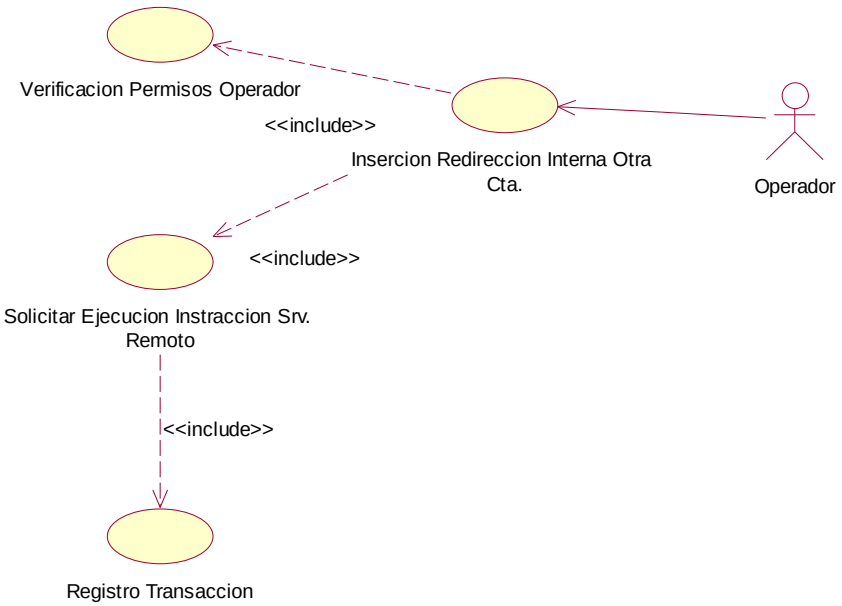


Figura 1-10

Tabla 1-7: Registro Usuario

CU-007	**Registro Usuario**
Instancia de usuarios Participantes	Operador, Operador No TI
Descripción	El sistema debe permitir al operador registrar un nuevo usuario sin necesidad de asociar aun a un dominio en particular con el password autogenerado por defecto.
Precondiciones	El usuario debe haberse logueado. Caso de uso CU-001
Flujo de eventos	
	4.- El operador debe seleccionar en el menú lateral izquierdo la opción de Usuarios.
	5.- El sistema devuelve el listado completo de los usuarios con una opción de edición de cada registro así como opciones para el registro de nuevos usuarios.
	6.- El operador selecciona el botón de crear con el cual ingresara un nuevo usuario.
	7.- El sistema renderiza la pantalla con los elementos para los cuales el usuario tiene permiso.
	8.- El sistema limita el acceso a los elementos de acuerdo a los permisos que tiene el operador.
	9.- El password es autogenerado por el sistema cuando se ingresa, sin embargo el operador lo podrá cambiar de según crea conveniente.
	10.- El sistema valida si la fecha de expiración es mayor en tres días a la fecha de concesión.
	11.- Si todo va bien el sistema registra el usuario y visualiza un mensaje de éxito.
Excepciones	
	10.- Si la fecha de concesión es mayor a la fecha de

	expiración o la fecha de expiración no es mayor en tres días a la fecha de concesión, entonces se presenta un mensaje de error y el sistema para su ejecución.
	11.- En cualquier momento el usuario puede cancelar para lo cual puede pulsar el botón de cerrar del popup presentado.
Importancia	Importante

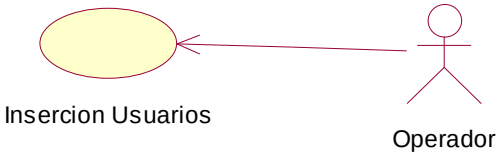


Figura 1-11

Tabla 1-8: Eliminación de redirecciones internas a otras cuentas

CU-008		**Eliminación de redirecciones internas a otras cuentas**
Instancia de usuarios Participantes		Operador
Descripción		El sistema debe permitir eliminar las redirecciones de correo
Precondiciones		El usuario debe haberse logueado. Caso de uso CU-001
Flujo de eventos		
	1	El operador o administrador selecciona la opción de redirecciones.
	2	El operador o administrador debe disponer de un listado de todos los usuarios, del cual seleccionara el deseado.
	3	Seleccionado el usuario, el sistema me debe dar la opción de escoger el dominio el cual requiero redireccionar.
	4	Seleccionado el dominio, el sistema presenta las opciones de redirección, del cual seleccionamos la opción redirección interna a otro usuario.

Tabla 1-8:

Continuación

	5	El sistema visualiza una región en la que constan todos los usuarios registrados y que se encuentran activos. Además visualiza todas las redirecciones del usuario, registrados actualmente.
	6	El operador selecciona el usuario de un listado.
	7	El sistema despliega los dominios del usuario destino.
	8	Seleccionado el dominio, el sistema muestra el botón para eliminar la redirección.
	9	El sistema valida la existencia de otras transacciones.
	10	El sistema verifica permisos sobre servidores para el operador.
	11	El sistema verifica que los servidores se encuentren activos.
	12	El sistema elimina la redirección de los servidores correspondientes.
Excepciones		
	9	Si existen transacciones en curso para el mismo usuario, dominio y servicio, entonces el sistema realiza una espera. Si existen más de dos transacciones en espera entonces el sistema descarta la transacción a partir de la segunda en espera.
	10	Si el sistema determina que el operador no tiene permisos sobre los servidores a ser modificados, entonces no realiza acción alguna, y deshace todos los cambios.
	11	Si los servidores destino no están activos o están apagados, entonces el sistema no realiza ninguna operación hasta que todos estén listos.
Importancia		Importante

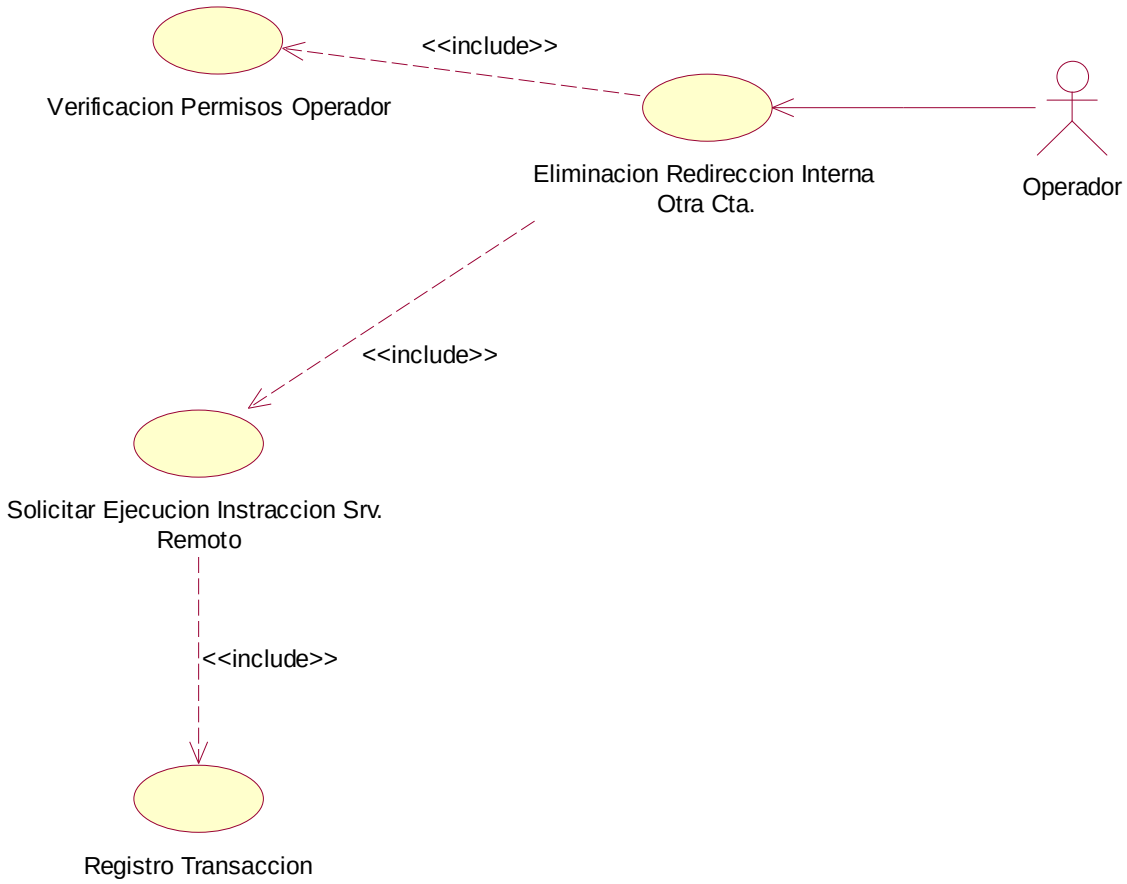


Figura 1-12

Tabla 1-9: Creación de redirecciones externas

CU-009		**Creación de redirecciones externas**
Instancia de usuarios Participantes		Operador
Descripción		El sistema debe permitir la creación de redirecciones a cuentas de mail externos, Ej. Hotmail, Yahoo, gmail. Etc.
Precondiciones		El usuario debe haberse logueado. Caso de uso CU-001
Flujo de eventos		
	1	El usuario selecciona la opción de redirecciones externas en el menú del sistema
	2	El sistema presenta un listado de usuarios donde el operador puede seleccionar el usuario a redireccionar.

Tabla 1-9:

Continuación

	3	Seleccionado el usuario, el sistema visualiza los dominios disponibles.
	4	Si el operador selecciono un dominio, el sistema despliega las opciones de redirección.
	5	Seleccionado la opción de redirección externa el sistema lista todas las cuentas de correo externo, y si no existe la cuenta el sistema debe dar la opción de crear una nueva.
	6	El sistema visualiza un botón para que el operador proceda con la creación de la redirección externa.
Excepciones		
	5	Si la dirección externa ya existiera, entonces el sistema presenta un mensaje de dirección duplicada y no permite la creación de la misma.
	6	El sistema verifica la existencia de transacciones, si existen una en curso entonces el sistema realiza una espera. A partir de la segunda transacción en espera las actividades son descartadas.
	6	Si el operador no dispone de permisos sobre todos los servidores implicados en la operación, entonces el sistema no procede con solicitud enviada.
	6	Si los servidores implicados en la redirección no están activos, el sistema cancela la operación.
Importancia		Importante

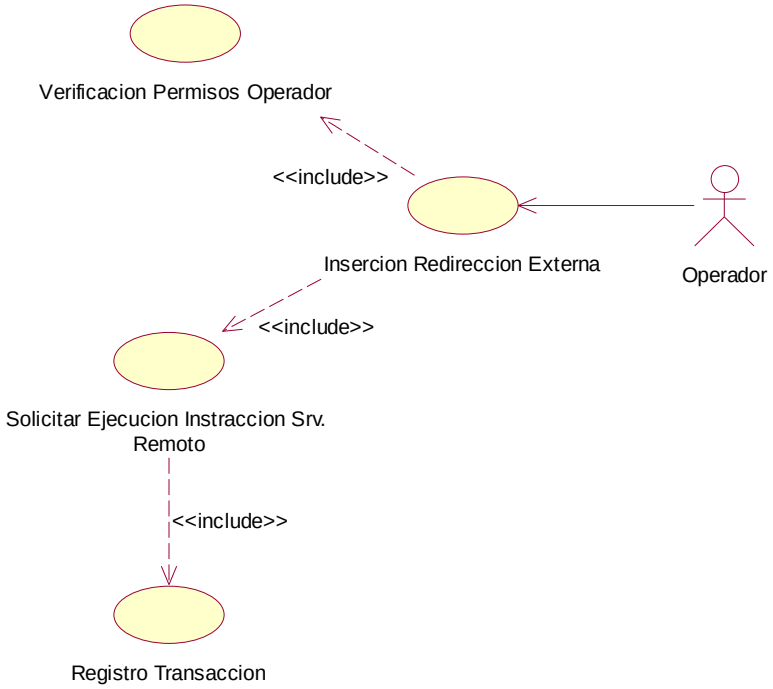


Figura 1-13

Tabla 1-10: Eliminación de redirecciones externas

CU-010		**Eliminación de redirecciones externas**
Instancia usuarios Participantes		Operador
Descripción		El sistema debe permitir la eliminación de las redirecciones de correo hacia cuentas externas: hotmail, gmail, yahoo, etc.
Precondiciones		El usuario debe haberse logueado. Caso de uso CU-001
Flujo de eventos		
	1	El operador o administrador debe seleccionar la opción de redirecciones, en el menú presentado por el sistema
	2	El sistema visualiza un listado de usuarios del cual el operador puede seleccionar el deseado.
	3	El sistema visualiza los dominios asociados al usuario seleccionado.

	4	El sistema visualiza una vez seleccionado el dominio un método para eliminar la redirección.
	5	El sistema verifica transacciones actuales.
	6	El sistema valida permisos del operador.
	7	El sistema valida conectividad a los servidores.
	8	El sistema ejecuta la operación solicitada.
Excepciones		
	5	Si existe alguna transacción en curso, el sistema provoca una espera. A partir de la segunda transacción en espera, las mismas serán descartadas.
	6	Si el operador no tiene permisos, el sistema no realiza la operación solicitada.
	7	Si los servidores no responden con un estado activo, entonces el sistema cancela la operación solicitada.
Importancia		Importante

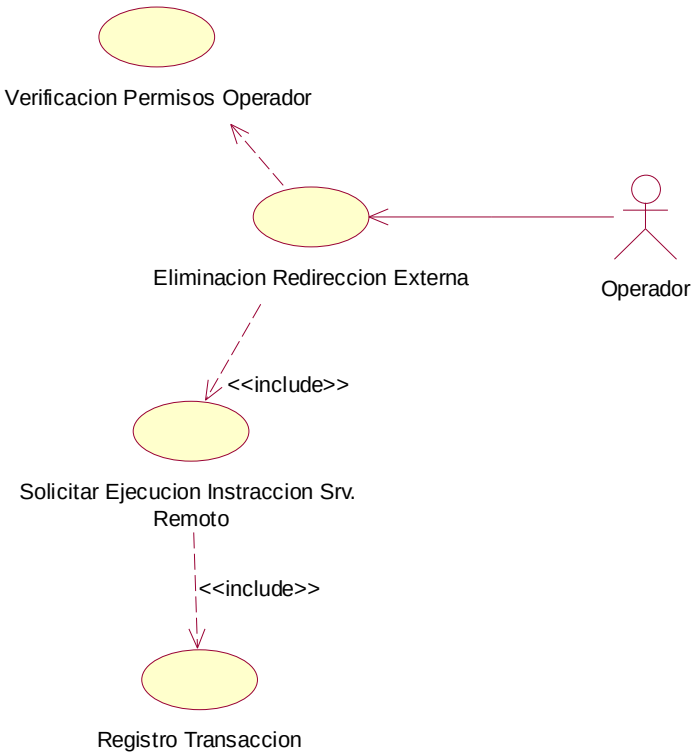


Figura 1-14

Tabla 1-11: Crear asignación DHCP

CU-011		**Crear asignación DHCP**
Instancia de usuarios Participantes		Operador
Descripción		El sistema debe permitir el registro de un usuario en el o los servidores DHCP, asociados a un dominio.
Precondiciones		El usuario debe haberse logueado. Caso de uso CU-001
Flujo de eventos		
	1	El operador debe seleccionar la opción de "Concesión de servicios"
	2	El sistema presenta un listado de todos los usuarios disponibles.
	3	El operador selecciona el usuario y el sistema despliega los servicios asociados al mismo, si el usuario no dispone del servicio "Internet Directo", entonces lo puede crear en ese mismo instante.
	4	Seleccionado el servicio, el usuario necesita disponer una tarjeta de red.
	5	Una vez seleccionado la tarjeta, el sistema debe presentar los dominios existentes.
	6	Seleccionado el dominio, el sistema visualiza el botón que registrara la asignación.
	7	El sistema registra el usuario en los servidores asociados al dominio seleccionado.
Excepciones		
	5	Si no existe una tarjeta de red asociado a un usuario, entonces el sistema no desplegará los dominios disponibles.
	6	Si no existe al menos un servidor asociado al dominio de DHCP, entonces el sistema no visualiza dicho dominio.

Tabla 1-11:

Continuación

	7	Si el sistema tiene una transacción en curso, entonces la solicitud deberá esperar para ser procesada. A partir de la segunda transacción el sistema descartara las operaciones.
	7	Si el operador no tiene permisos sobre el o los servidores asociados a un dominio, entonces el sistema cancela la operación.
	7	Si los servidores asociados a un dominio no se encuentran activos entonces, el sistema no procesa la solicitud.

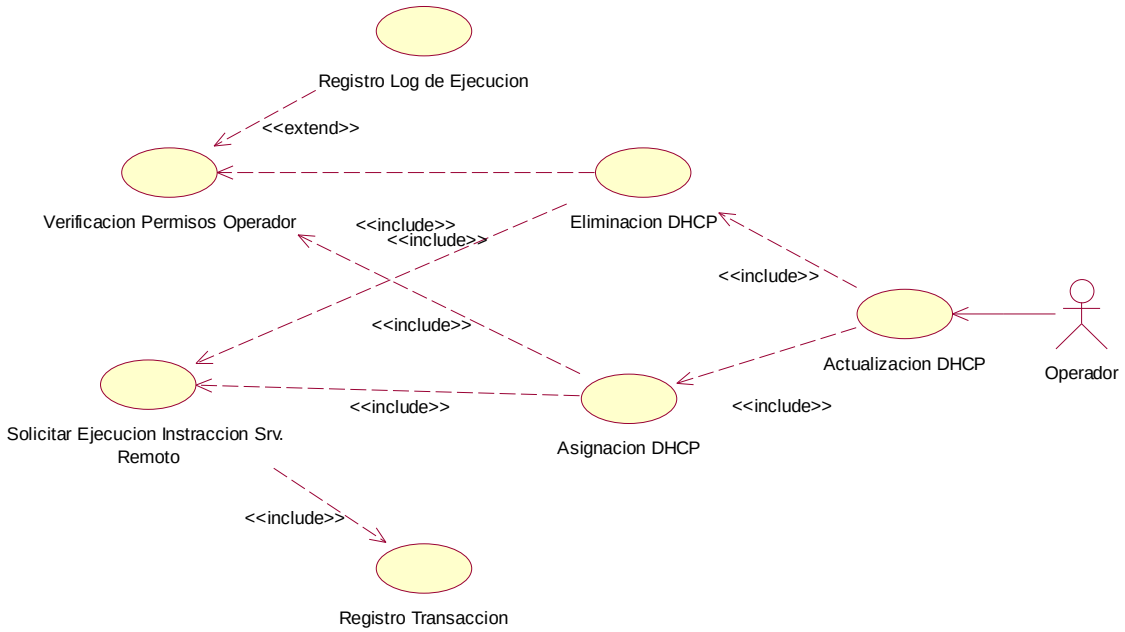


Figura 1-15

Tabla 1-12: Eliminación de asignación DHCP

CU-012		**Eliminación de asignación DHCP.**
Instancia de usuarios Participantes		Operador
Descripción		El sistema debe permitir eliminar la asignación de una IP en el DHCP un usuario en particular.

Tabla 1-12:

Continuación

Precondiciones		El usuario debe haberse logueado. Caso de uso CU-001
Flujo de eventos		
	1	El usuario seleccionar del menú la opción de concesión de servicios.
	2	El sistema despliega una pantalla en la que se visualiza alguna opción que permita la selección del usuario.
	3	Después de que el operador seleccione el usuario, el sistema visualiza el listado de servicios disponibles para el usuario.
	4	El sistema una vez que el operador seleccione el servicio de "Internet Directo", renderizará las tarjetas de red del usuario.
	5	El sistema debe posibilitar el registro de la tarjeta de red en uno o varios servidores
	6	EL operador debe solicitar la acción presionando la opción de registro.
	7	El sistema debe proporcionar alguna alternativa que alerte al usuario antes de proceder con la operación.
Excepciones		
	3	Si el servicio no existe, entonces el usuario podrá realizar el registro en ese momento.
	4	Si el usuario no dispone de una tarjeta de red, entonces el sistema no despliega la opción de registro y el usuario no puede continuar.
	5	Si existen servidores asociados al servicio y a un dominio, entonces el sistema presenta una lista con las opciones disponibles.
	6	Si existe una transacción en curso, entonces el sistema espera hasta que la misma termine y continúa su trabajo.

	6	Si existe más de una transacción en espera, entonces el sistema aborta la operación a partir de la segunda.
	6	El sistema verifica que el operador disponga de permisos que le permitan manipular los servidores, si no existe entonces cancela la operación y visualiza un mensaje con el problema suscitado.
	6	El sistema verifica conectividad con cada servidor, en caso de no existir al menos con uno, el sistema cancela la transacción.
	7	El usuario puede seleccionar la opción cancelar y el sistema no ejecuta la operación solicitada.
Importancia		Importante

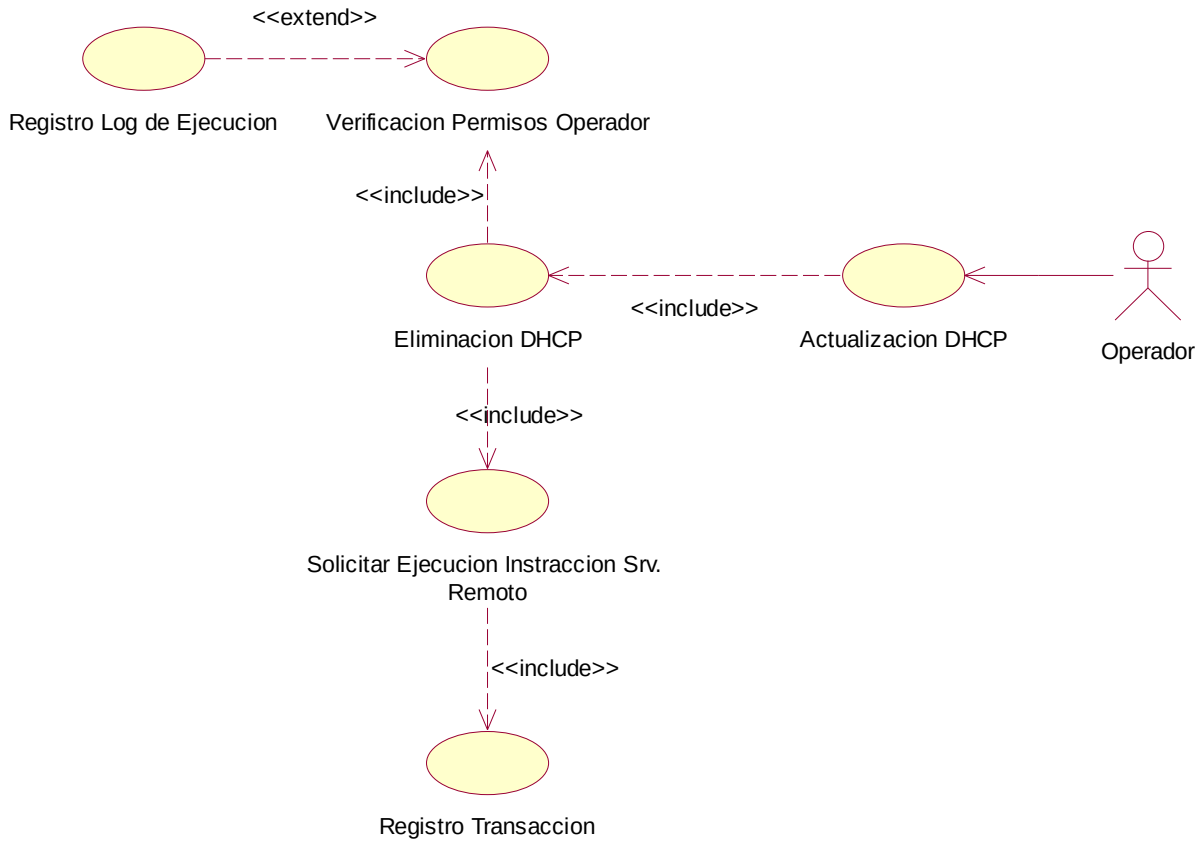


Figura 1-16

Tabla 1-13: Notificación estado Srv. Remoto

CU-013		**Notificación estado servidor remoto.**
Instancia usuarios Participantes		Servidor Mail, Servidor DHCP, Sistema
Descripción		Captura de notificación acerca del estado de los servidores remotos.
Precondiciones		
Flujo de eventos		
	1	El servidor verifica cuáles son sus: -Nombre de equipo. - Servicio activo - Dirección MAC - Dirección IP
	2	El servidor notifica al sistema de que se encuentra activo, enviándole para ello los datos capturados en la fase anterior adjunto además el estado de UP.
	3	La aplicación verifica la existencia del servidor con los datos que se envió.
	4	El sistema realiza un ping al servidor del cual receptó la información.
	5	El sistema actualiza el estado del servidor en función de su repuesta y los datos proporcionados.
Excepciones		
	2	Si la base de datos no se encuentra activo o se produce un error de conexión con ella, entonces el servidor guarda el último error producido.
	3	Si los datos enviados por el servidor no concuerdan con los registrados en el sistema, entonces la aplicación no realiza actualización alguna.
	4	Si el servidor no responde al ping entonces la aplicación se encarga de cambiar el estado al servidor registrado.
	5	Si al menos uno de los datos: "intradius" y "key_server" se encuentra con "0" entonces el sistema detecta que existe un error y reporta al servidor como fallido.

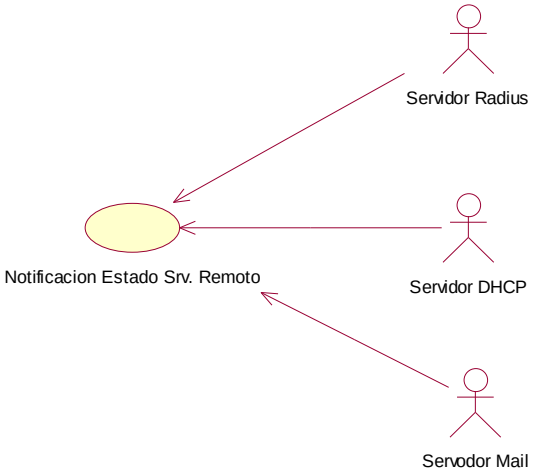


Figura 1-17

Tabla 1-14: Sincronización Servidores

CU-014		**Sincronización Servidores**
Instancia de usuarios Participantes		Operador
Descripción		Es requerido la sincronización de cuentas de correo entre los servidores que constan como miembros de un dominio.
Precondiciones		El usuario debe haberse logueado. Caso de uso CU-001
Flujo de eventos		
	1	El operador selecciona del menú la opción de sincronizaciones.
	2	El sistema despliega una pantalla en la que el usuario puede seleccionar los dominios disponibles.
	3	Una vez que el operador seleccione el dominio, la aplicación se encargara de desplegar los servidores disponibles.

Tabla 1-14: Continuación

	4	Si el operador ha seleccionado el servidor origen de la sincronización, el sistema visualiza el listado de los servidores destino.
	5	Cuando el operador haya seleccionado todos los parámetros requeridos, el sistema validara permisos del operador sobre los servidores seleccionados.
	6	El sistema valida disponibilidad de conexión con los servidores.
	7	El sistema notifica que es importante que la llave pública del servidor origen deba estar registrada en el destino.
	8	El proceso termina con la sincronización y un mensaje notificando que el servidor destino será reiniciado.
Excepciones		
	5	Si el operador no dispone de permisos entonces el sistema cancela la operación y visualiza un mensaje haciendo referencia al problema suscitado.
	6	Si al menos uno de los servidores implicados en la sincronización está inactivo, entonces la aplicación cancela la operación.
	7	Si la llave pública no está registrada en el servidor destino, entonces la aplicación falla.

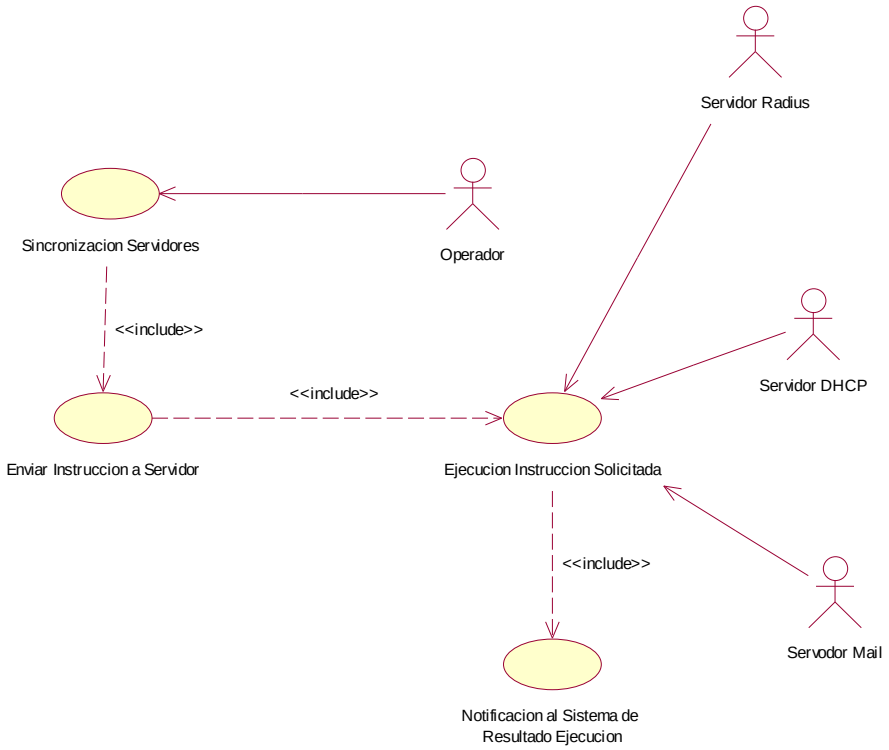


Figura 1-18

Tabla 1-15: Salida de personal

CU-015		**Salida de personal**
Instancia de usuarios Participantes		Operador, Servidor Mail, Servidor DHCP
Descripción		Cuando un colaborador abandona alguna de las empresas del grupo, es necesario eliminar los servicios que fueron asignados a él.
Precondiciones		El usuario debe haberse logueado. Caso de uso CU-001
Flujo de eventos		
	1	El operador selecciona la opción "Finalización de Contrato" en el menú.
	2	El sistema visualiza una pantalla en la que el operador puede seleccionar el usuario que desea que se desvincule de la empresa.

Tabla 1-15:

Continuación

	3	Seleccionado el usuario, el sistema solicitara que el operador seleccione otro usuario que estará receptando los correos del empleado que sale.
	4	El empleado que receptara los correos puede tener múltiples dominios, es por ello que los mismos son listados para que el operador defina cual necesita usar.
	5	El sistema proporciona la posibilidad para que el operador decida la fecha en la cual los servicios del usuario saliente, se eliminen definitivamente de los servidores asociados.
	6	Establecidos estos parámetros, el sistema verifica transacciones en curso.
	7	El sistema ejecuta la acción solicitada.
	8	El servidor Radius bloquea todos los intentos de conexión a la red inalámbrica.
Excepciones		
	6	Si existen transacciones en ejecución, entonces el sistema realiza una espera antes de proceder con la ejecución.
	6	Si hay más de una transacción en espera, entonces el sistema cancela las mismas a partir de la segunda.
	7	El sistema verifica los permisos del operador, si al menos en uno de los servidores que se necesita afectar el operador no dispone de permisos entonces el sistema visualiza un mensaje de error con la dirección IP del servidor en conflicto.
	7	El sistema antes de proceder con la ejecución de la operación solicitada verifica que todos los servidores asociados se encuentren activos, si al menos uno de ellos se encuentra inactivo, la aplicación cancela la operación.

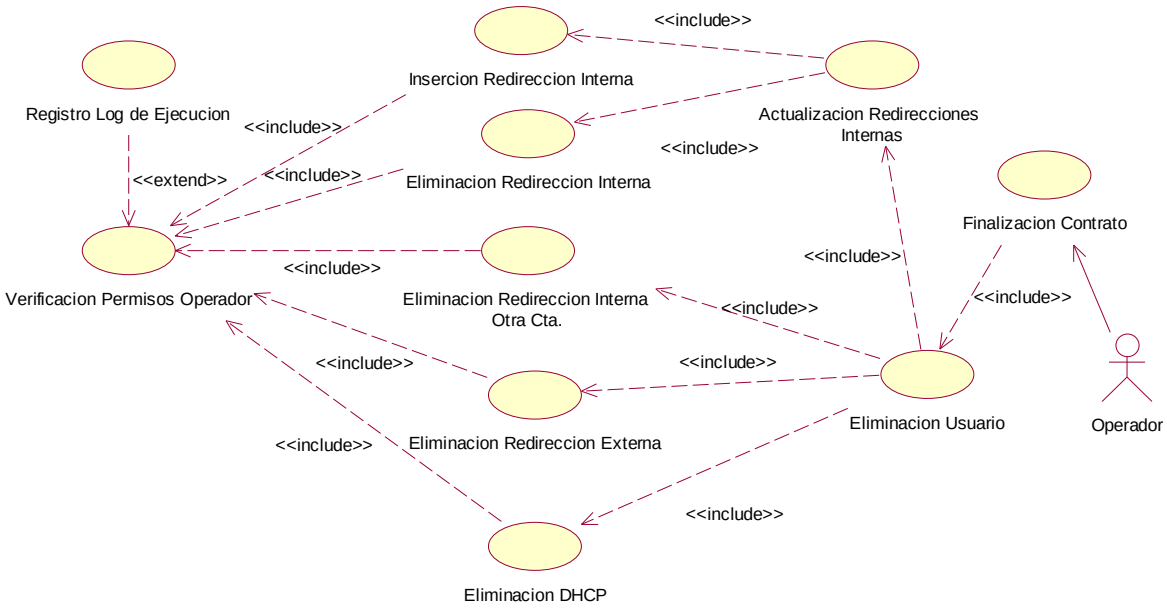


Figura 1-19

Tabla 1-16: Salida de vacaciones del personal

CU-016		**Salida de vacaciones del personal.**
Instancia de usuarios Participantes		Operador, Servidores mail, Servidor Oracle, Base de datos, Aplicación
Descripción		El abandono temporal de un empleado por uso de sus días de vacación genera una situación en la que el sistema debe generar una redirección temporal de sus correos.
Precondiciones		El usuario debe haberse logueado. Caso de uso CU-001
Flujo de eventos		
	1	El operador selecciona la opción "Salida de vacaciones"
	2	El sistema visualiza un listado de todos los usuarios activos.
	3	El operador escoge de entre los usuarios el empleado que está por hacer uso de sus vacaciones.
	4	El sistema le da dos opciones al operador: - Para cuando el usuario salga de vacaciones, su coreo puede ser redireccionado a una cuenta externa: hotmail, yahoo, etc. o a la cuenta de correo de otro usuario que pertenece a la organización.

Tabla 1-16:

Continuación

	5	El sistema obligatoriamente solicita el mensaje que será enviado automáticamente a las personas que escriban a la cuenta de correo del empleado que se encuentra tomando sus vacaciones.
	6	El sistema procesa la operación solicitada.
	7	El sistema internamente genera una redirección hacia la misma cuenta del usuario para conservar sus correos.
	8	El sistema guarda un registro de la actividad realizada.
	9	El servidor Radius bloquea todas las conexiones entrantes para la cuenta del empleado.
Excepciones		
	5	Si no se define el mensaje de auto respuesta, el sistema no procesa la solicitud y cancela la operación.
	6	Si el operador no dispone de permisos sobre los servidores implicados en la operación, entonces cancela la misma.
	6	Si al menos uno de los servidores que son afectados por la operación no se encuentra activo, el sistema cancela la operación.

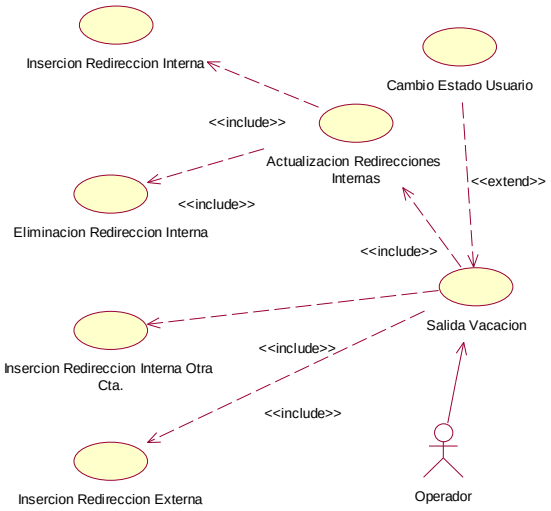


Figura 1-20

Tabla 1-17: Regreso de vacación de empleado

CU-017		**Regreso de vacación de empleado**
Instancia de usuarios Participantes		Operador
Descripción		El sistema debe permitir la reactivación de un usuario que salió de vacaciones.
Precondiciones		El usuario debe haberse logueado. Caso de uso CU-001
Flujo de eventos		
	1	El operador debe seleccionar la opción regreso de vacación del menú.
	2	El sistema visualiza una pantalla en donde se debe seleccionar el usuario que se encuentra de vacaciones.
	3	El sistema una vez seleccionado el usuario despliega un botón para ejecutar la operación.
	4	El sistema genera una nueva transacción
	5	El servidor Radius desbloquea al usuario para que la cuenta pueda nuevamente autenticarse en la red inalámbrica.
Excepciones		
	3	Si el operador no dispone de permisos sobre los servidores a ser afectados, entonces el sistema cancela la operación y visualiza un mensaje de error.
	3	Si cualquiera de los servidores que están implicados en la operación no se encuentra activo, entonces el sistema cancela la operación notificando que servidor se encuentra con problemas.
	4	Si existen más de dos transacciones en espera, el sistema descarta las operaciones a partir de la segunda en cola.
Importancia		Importante

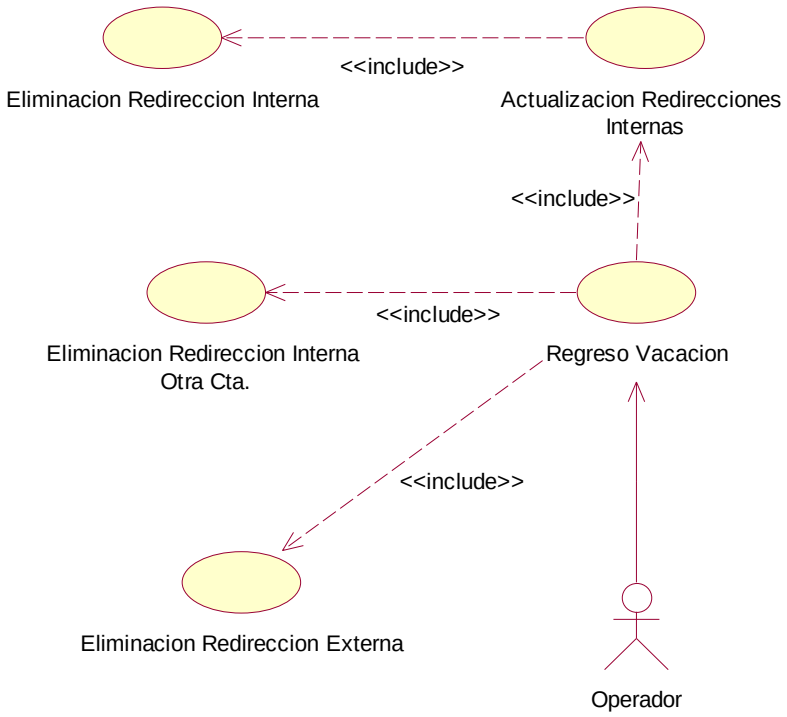


Figura 1-21

Tabla 1-18: Verificación Credenciales BD

CU-018		**Verificación Credenciales BD**
Instancia de usuarios Participantes		Operador, Servidor Mail, Servidor DHCP, Servidor Radius
Descripción		Debido a que los servidores que forman parte del sistema necesitan ingresar a la base de datos para realizar operaciones, es necesario definir métodos de autenticación que les permitan ingresar.
Precondiciones		
Flujo de eventos		
	1	Los servidores mail, DHCP, Radius se comunican con el cliente de Oracle instalado en el Sistema Operativo.
	2	Instancian las variables de ambiente.

Tabla 1-18:

Continuación

	3	Se abre en el sistema Operativo una "Tubería" (Pipeline), el cual contiene el resultado del intento de conexión realizado por SQL Plus.
	4	El servidor se conecta y ejecuta las operaciones que se requieran.
Excepciones		
	2	Si no se instancian las variables de ambiente la ejecución del intento de conexión falla.
	2	Si el listener de la base de datos se encuentra parado, entonces el intento de conexión nuevamente falla.
	3	Si existe cualquier error en la ejecución de querys, procedimientos, funciones el pipeline reenvía el mensaje ORACLE de error al pipeline.

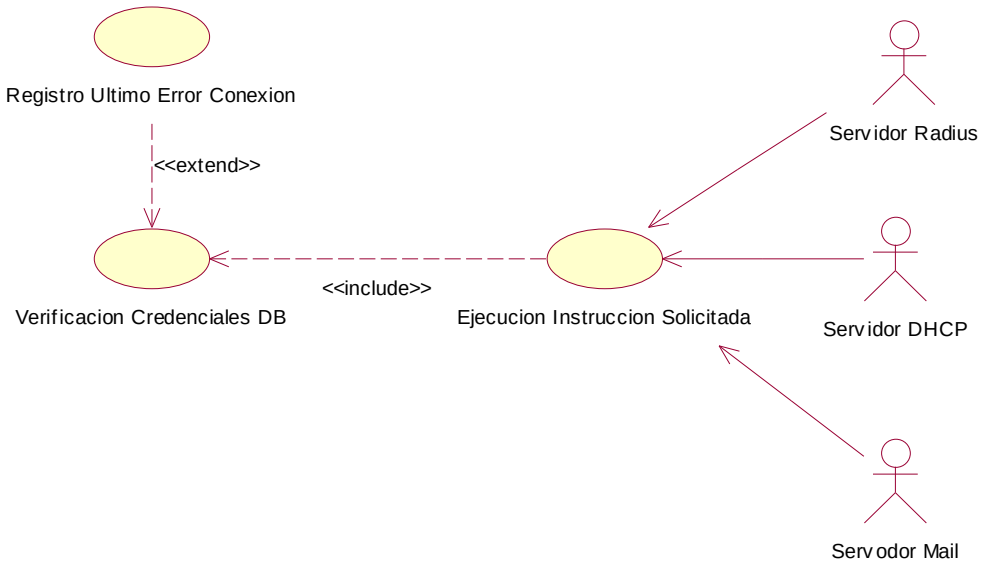


Figura 1-22

Tabla 1-19: Registro Log Ejecución

CU-019		**Registro Log Ejecución**
Instancia de usuarios Participantes		Sistema
Descripción		El sistema debe registrar un log con las operaciones realizadas y el operador que las ejecutó.
Precondiciones		Debe haberse iniciado una transacción.
Flujo de eventos		
	1	El operador selecciona cualquier actividad: - Registro Usuario - Registro Cuenta Correo - Registro Redirecciones - Registro DHCP - Registro Salida Vacaciones - Finalización Contrato - Eliminación Usuario - Eliminación Cuenta - Eliminación Redirecciones - Eliminación relación DHCP
	2	El sistema genera una transacción
	3	El sistema guarda el log de las operaciones realizadas.

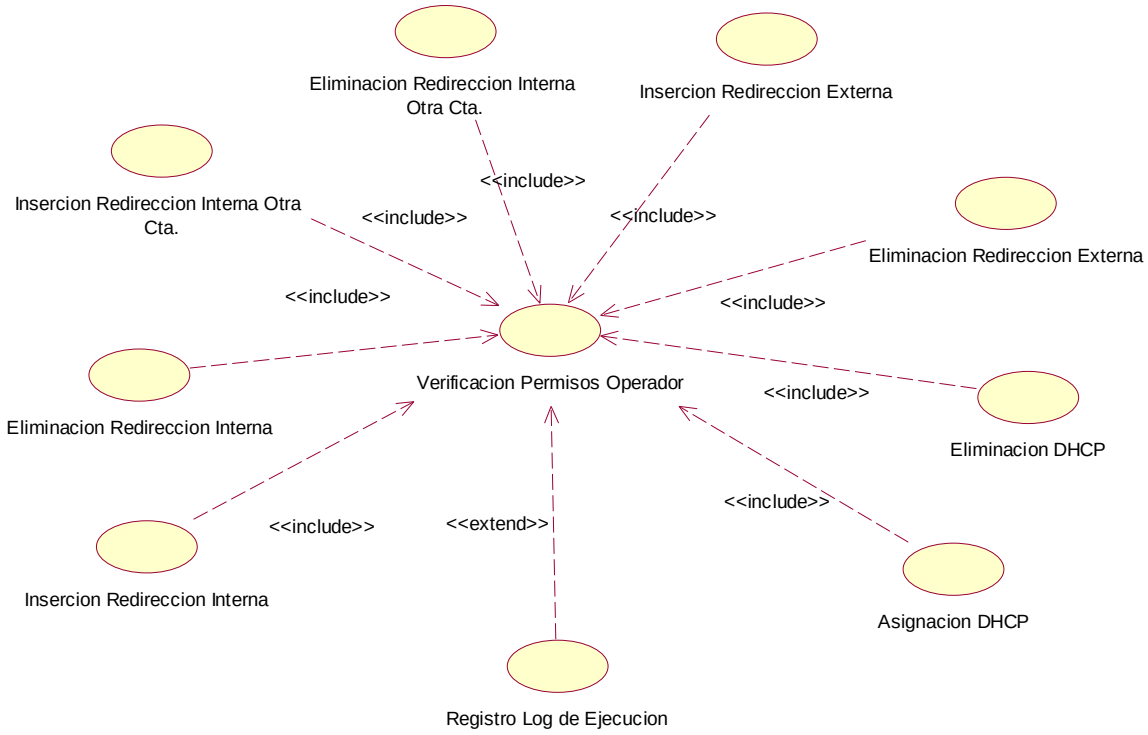


Figura 1-23

Tabla 1-20: Solicitar Ejecución Instrucción Srv. Remoto

CU-020		**Solicitar Ejecución Instrucción Srv. Remoto**
Instancia de usuarios Participantes		Aplicación, Servidor Mail, Servidor DHCP, Servidor Oracle
Descripción		El sistema solicita la ejecución de una instrucción en los servidores remotos para completar la tarea enviada por el operador.
Precondiciones		Debe haberse iniciado una transacción.
Flujo de eventos		
	1	El sistema verifica que se haya iniciado una transacción.
	2	El sistema espera si encuentra transacciones en ejecución.
	3	Validar la existencia del directorio INTRADIUS
	4	Verificar la disponibilidad de la llave publica del servidor. Remoto.

Tabla 1-20:

Continuación

	5	El sistema envía una instrucción a los servidores implicados.
Excepciones		
	2	El sistema detecta alguna restricción por índices duplicados.
	5	El sistema puede quedar inconsistente si en el transcurso de la ejecución de una instrucción existen cortes de energía, etc.

Tabla 1-21: Gestión de menús

CU-021		**Gestión de menús**
Instancia de usuarios Participantes		Operador Administrador
Descripción		Es necesario que el administrador pueda modificar los menús y sus opciones que estarán disponibles para los usuarios.
Precondiciones		El usuario debe haberse logueado. Caso de uso CU-001
Flujo de eventos		
	1	El operador debe seleccionar la opción "objetos" en donde se define las opciones que tiene los menús del usuario. Entre las operaciones permitidas están: inserción, eliminación y modificación de cualquier opción.
	2	El operador debe seleccionar la opción "Menús-Objetos" en donde el espacio de menú es asociado el árbol generado en la actividad anterior.
	3	El operador debe seleccionar una opción en la que se asocie el menú a un usuario.
Excepciones		

Tabla 1-20:

Continuación

	2	Si no se generó el árbol de opciones del menú, entonces no será posible asociar a un espacio.
	3	Si no se asignó un espacio con las opciones, entonces no es posible asignar el menú a un usuario.

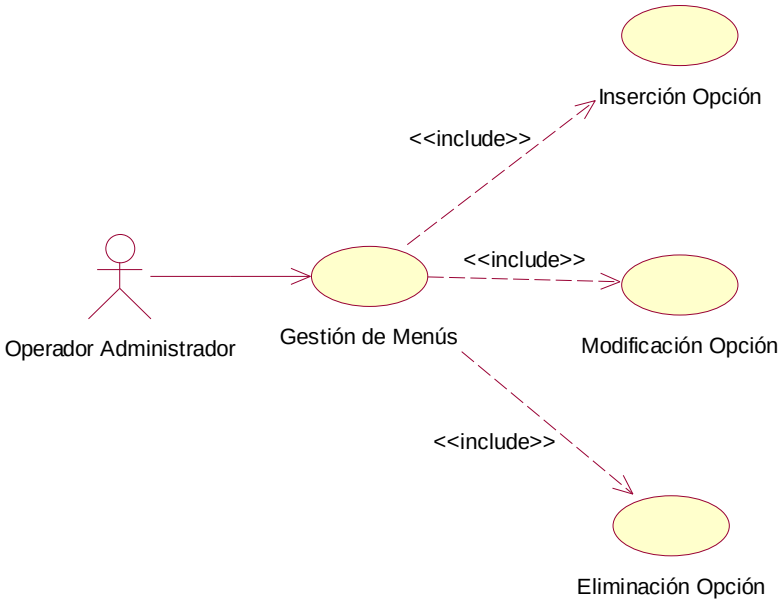


Figura 1-24

Tabla 1-22: Mantenimiento Tablas Base

CU-022		**Mantenimiento Tablas Base**
Instancia de usuarios Participantes		Operador Administrador
Descripción		El sistema debe ser capaz de proporcionar métodos con los que se permita dar mantenimiento a todas las tablas que lo conforman.
Precondiciones		El usuario debe haberse logueado. Caso de uso CU-001
Flujo de eventos		
	1	El sistema genera un menú con opciones que representan

		las tablas del sistema.
	2	El operador selecciona cualquier opción de la sección de mantenimiento
	3	El sistema visualiza un reporte con todos los registros listados. Además visualiza un link por cada registro con el que se pueda eliminar, modificar e insertar nuevos datos.
Excepciones		
	3	Ninguno de los registros puede estar duplicado por cuanto se genera una restricción de información duplicada.

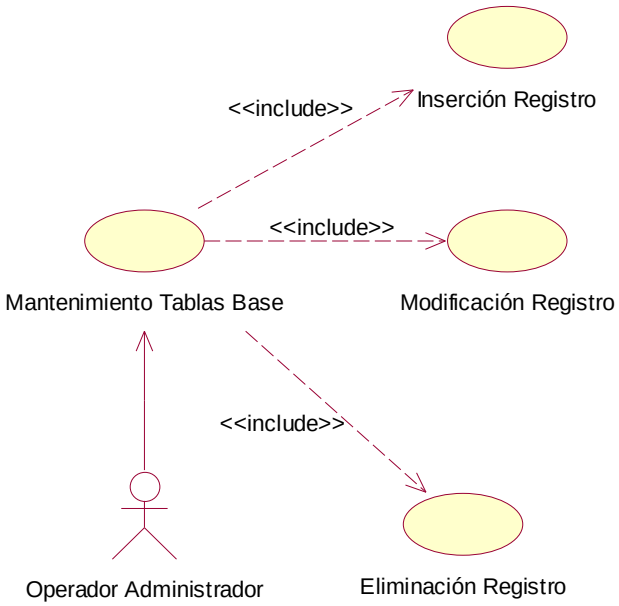


Figura 1-25

Tabla 1-23: Registro Último Error Conexión

CU-023		**Registro Último Error Conexión**
Instancia de usuarios Participantes		Servidor Mail, Servidor DHCP

Tabla 1-23:

Continuación

Descripción		Cada servidor debe tener la posibilidad de guardar el último error generado por un intento fallido de conexión a la base.
Precondiciones		
Flujo de eventos		
	1	El servidor genera o recibe cualquier solicitud para ejecutar una instrucción.
	2	El servidor busca en su sistema de archivos los scripts necesarios para verificar conexión hacia la base de datos.
	3	El servidor trata de conectarse a la base de datos para ejecutar un procedimiento almacenado
Excepciones		
	3	Si el listener de la base de datos no se encuentra activo, entonces los script de conexión reportan el error en el archivo "output.log"

Tabla 1-24: Enviar Instrucción a Servidor

CU-024		**Enviar Instrucción a Servidor**
Instancia de usuarios Participantes		Sistema
Descripción		El sistema debe garantizar que se genere adecuadamente la instrucción a ser ejecutada por los servidores remotos, y además debe tener la capacidad de enviarlos.
Precondiciones		Debe haberse iniciado una transacción.
Flujo de eventos		
	1	El sistema genera una transacción por cualquier operación que requiera la intervención de los servidores remotos.

Tabla 1-24:

Continuación

	2	El sistema verifica que los servidores remotos consten como activos.
	3	El sistema valida la existencia de modulo para envío de instrucciones
	4	El sistema envía la instrucción a los servidores externos a través del sistema operativo que ejecuta la base de datos.
Excepciones		
	3	Si el modulo que gestiona el envío de instrucciones remotas tiene errores toda la transacción se invalida.
	3	Si el sistema operativo hospedero de la base de datos tiene problemas de conexión de red, puede generarse inconsistencias entre la base de datos y los registros reales en los servidores remotos.

Tabla 1-25: Solicitud Autenticación Wireless Usuario

CU-025		**Solicitud Autenticación Wireless Usuario**
Instancia de usuarios Participantes		Servidor Radius, Usuarios, Access Point, Base de datos
Descripción		El sistema debe ser capaz de visualizar los intentos de conexión a la red inalámbrica.
Precondiciones		Debe existir conexión a la base de datos. CU-026
Flujo de eventos		
	1	El servidor verifica la existencia de la tabla que contiene los Access Point autorizados para pedir autenticación.
	2	El servidor valida que la configuración del mismo este correcta.
	3	El servidor consulta la tabla de usuarios y verifica la existencia del usuario que solicita autenticación.

Tabla 1-24:

Continuación

	4	El servidor Radius guarda un log del equipo que solicitó autorización de acceso a la red inalámbrica.
Excepciones		
	3	Si el usuario no existe entonces el servidor Radius no autoriza el acceso al solicitante, a la red inalámbrica.

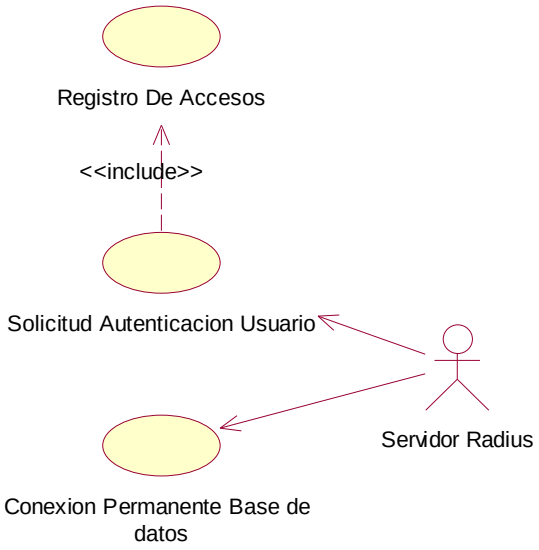


Figura 1-26

Tabla 1-26: Conexión permanente Base de Datos

CU-026		**Conexión Permanente Base de datos**
Instancia de usuarios Participantes		Servidor Radius
Descripción		El servidor Radius inicia una conexión persistente a la base de datos
Precondiciones		El servidor Radius debe poder ejecutar exitosamente el comando ping al servidor que contiene la base de datos.
Flujo de eventos		
	1	El servidor verifica la existencia de dos librerías Oracle que son las que permitirán la conexión hacia la base de datos.

Tabla 1-26:

Continuación

	2	El servidor usa las credenciales proporcionadas y trata de iniciar conexión hacia Oracle.
	3	Una vez que el servidor de base de datos autorizó el acceso entonces el servidor Radius mantiene conexión.
Excepciones		
	1	Si no existen las librerías entonces el servidor Radius genera un error grave y para su ejecución.
	2	Si las credenciales no son correctas entonces falla la conexión.

1.2.2.3 Diagramas de secuencia

Un diagrama de secuencias permite diseñar las interacciones entre los objetos de un sistema y el orden en el que ocurrirán mediante el envío de mensajes. Así mismo este diagrama por lo general no contiene casos de decisión, etc. puesto que el diagrama sería tan grande que no se podría comprender. Se podría decir que el diagrama de secuencias proporciona una alternativa en el que se pueda describir las operaciones de los casos de uso en forma más detallada.

Los diagramas de secuencias tienen tres componentes fundamentales: objetos, mensajes y líneas de "vida" entre los objetos. Es importante que describamos estos tres componentes para poder entender los diagramas.

Mensajes: Son las actividades solicitadas desde un objeto a otro y se representa mediante una flecha horizontal. Existen diferentes tipos que lo podemos observar en la siguiente ilustración.

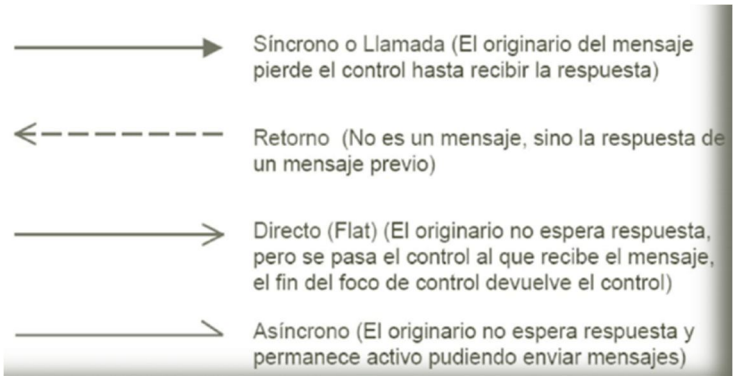


Figura 1-27: Diagramas de secuencias comportamiento de los sistemas.

http://www.scribd.com/doc/15493687/DIAGRAMAS-DE-SECUENCIA

Objetos: Los objetos se representan mediante un rectángulo que se coloca en la parte superior del diagrama y que tiene a su vez asociado una línea de vida.

Línea de vida: La línea de vida se representa mediante una línea vertical que nace desde un objeto y del cual se originan o llegan los mensajes.

Para nuestra aplicación han surgido los siguientes diagramas de secuencia:

Verificación Credenciales sistema

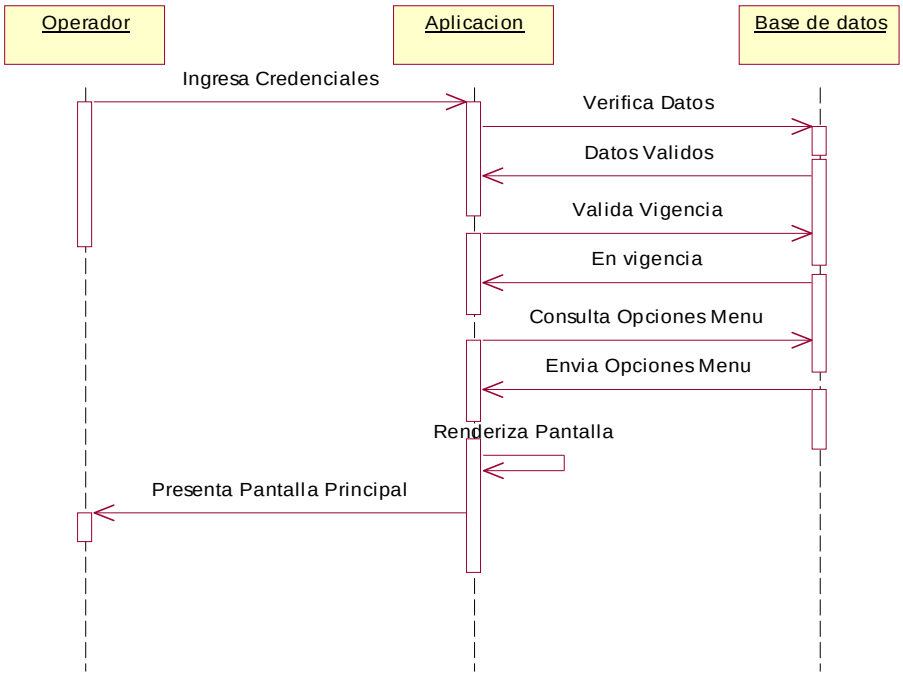


Figura 1-28

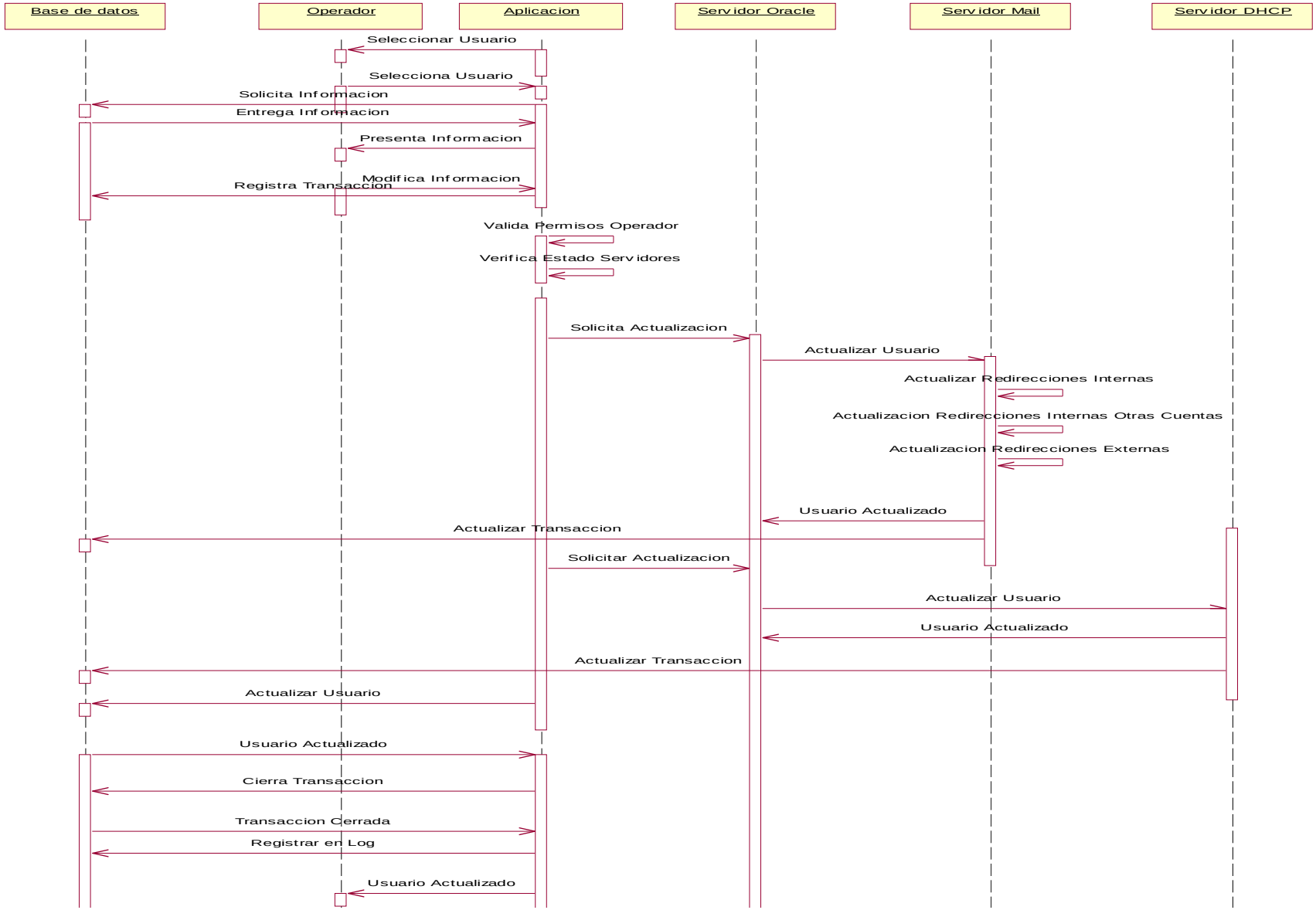

Base de datos
Operador
Aplicacion
Servidor Oracle
Servidor Mail
Servidor DHCP
Seleccionar Usuario
Selecciona Usuario
Solicita Informacion
Entrega Informacion
Presenta Informacion
Modifica Informacion
Registra Transaccion
Valida Permisos Operador
Verifica Estado Servidores
Solicita Actualizacion
Actualizar Usuario
Actualizar Redirecciones Internas
Actualizacion Redirecciones Internas Otras Cuentas
Actualizacion Redirecciones Externas
Usuario Actualizado
Actualizar Transaccion
Solicitar Actualizacion
Actualizar Usuario
Usuario Actualizado
Actualizar Transaccion
Actualizar Usuario
Usuario Actualizado
Cierra Transaccion
Transaccion Cerrada
Registrar en Log
Usuario Actualizado

Figura 1-29

Eliminación de usuario

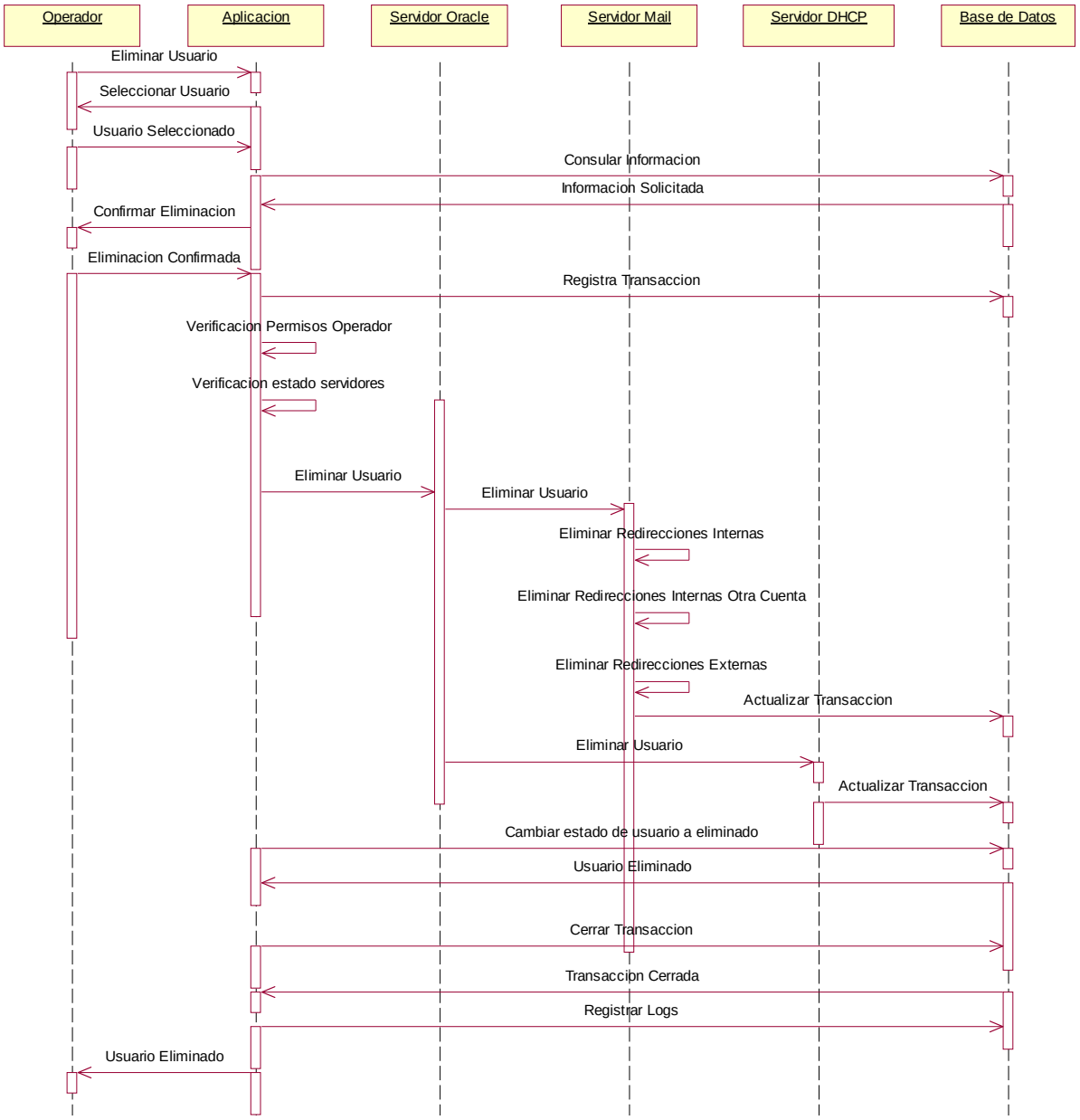


Figura 1-30

Creación cuenta de correo

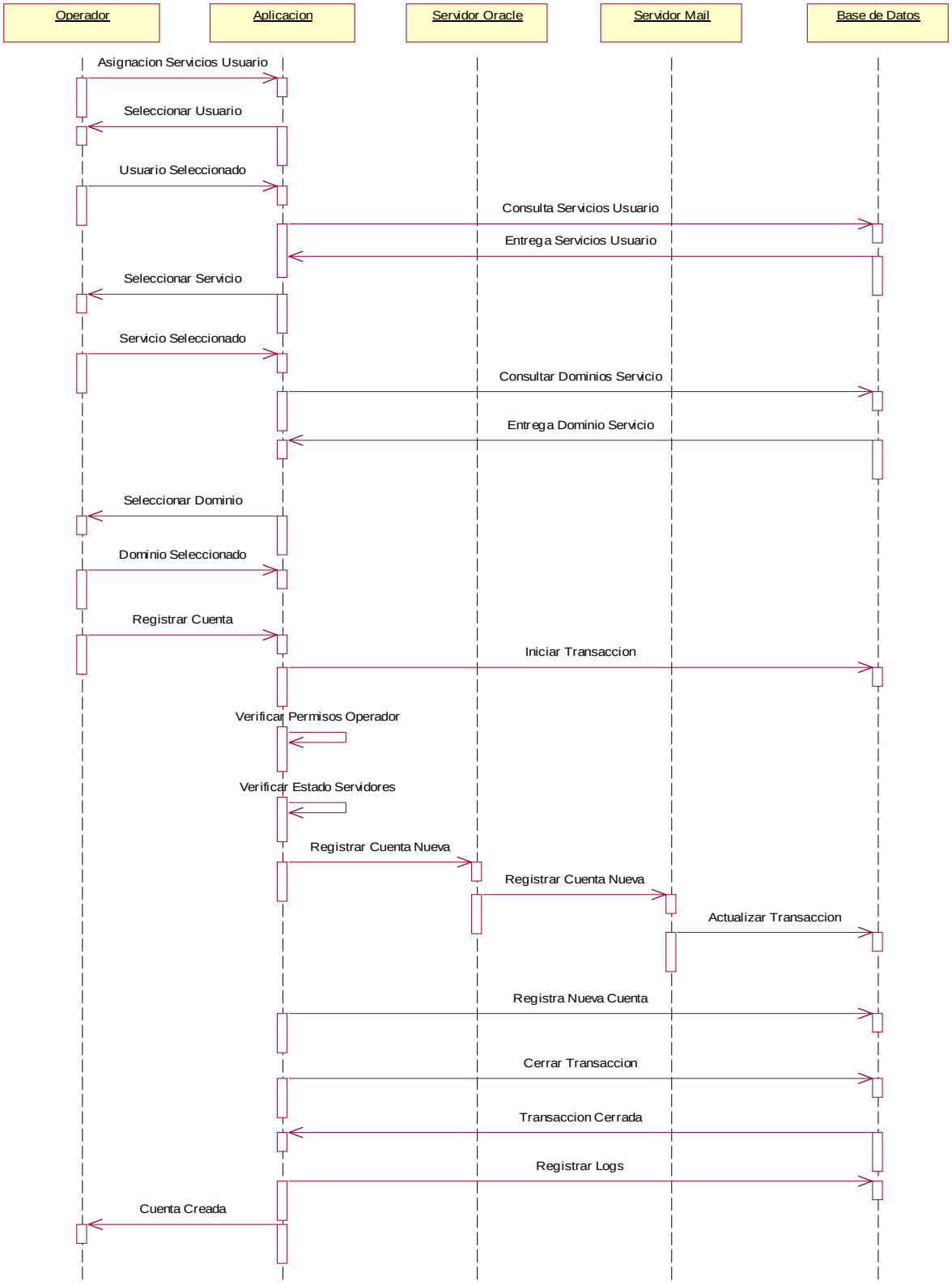


Figura 1-31

Actualización de redirecciones a cuentas internas del mismo usuario

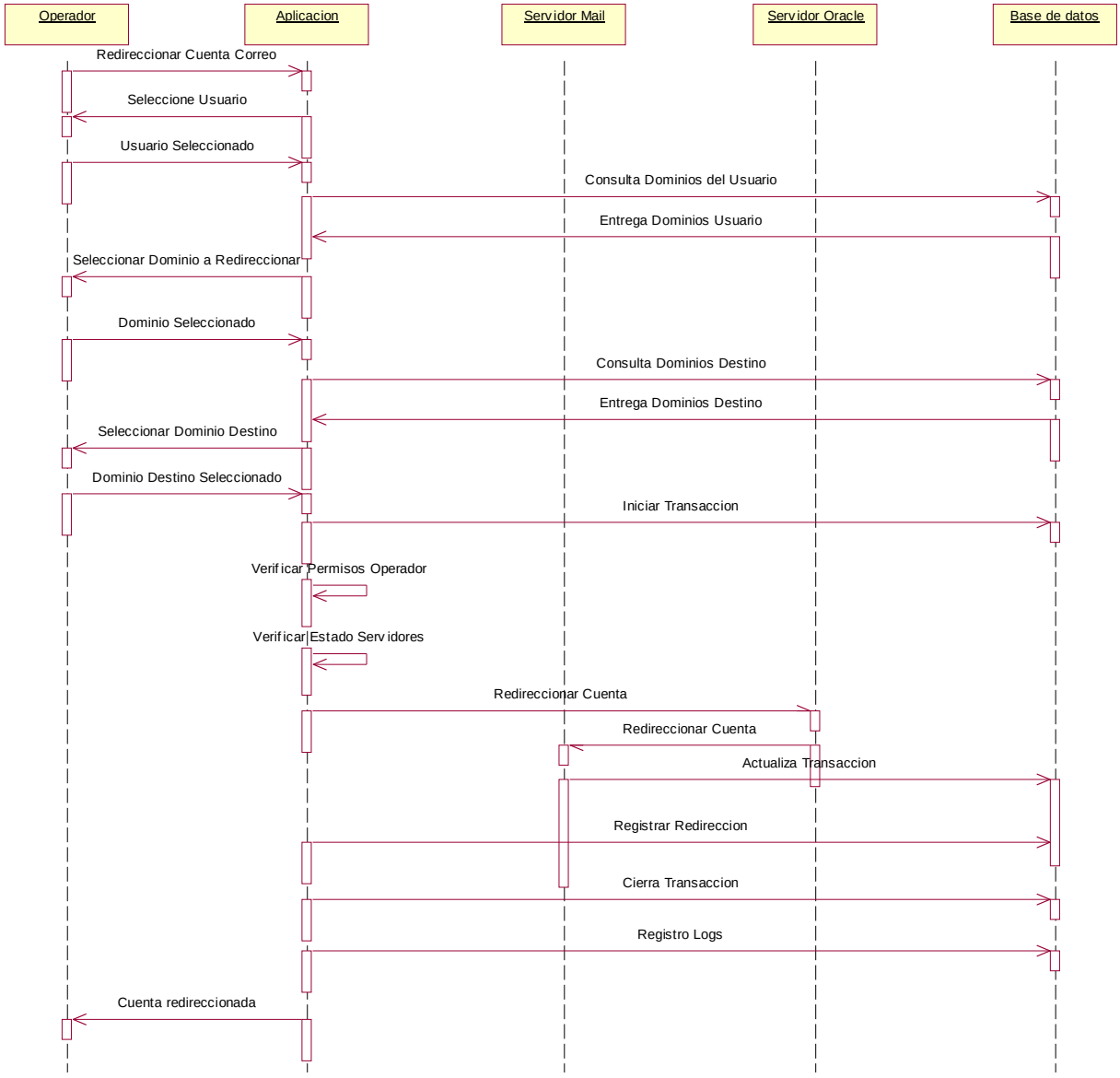


Figura 1-32

Inserción de una redirección interna hacia otra cuenta.

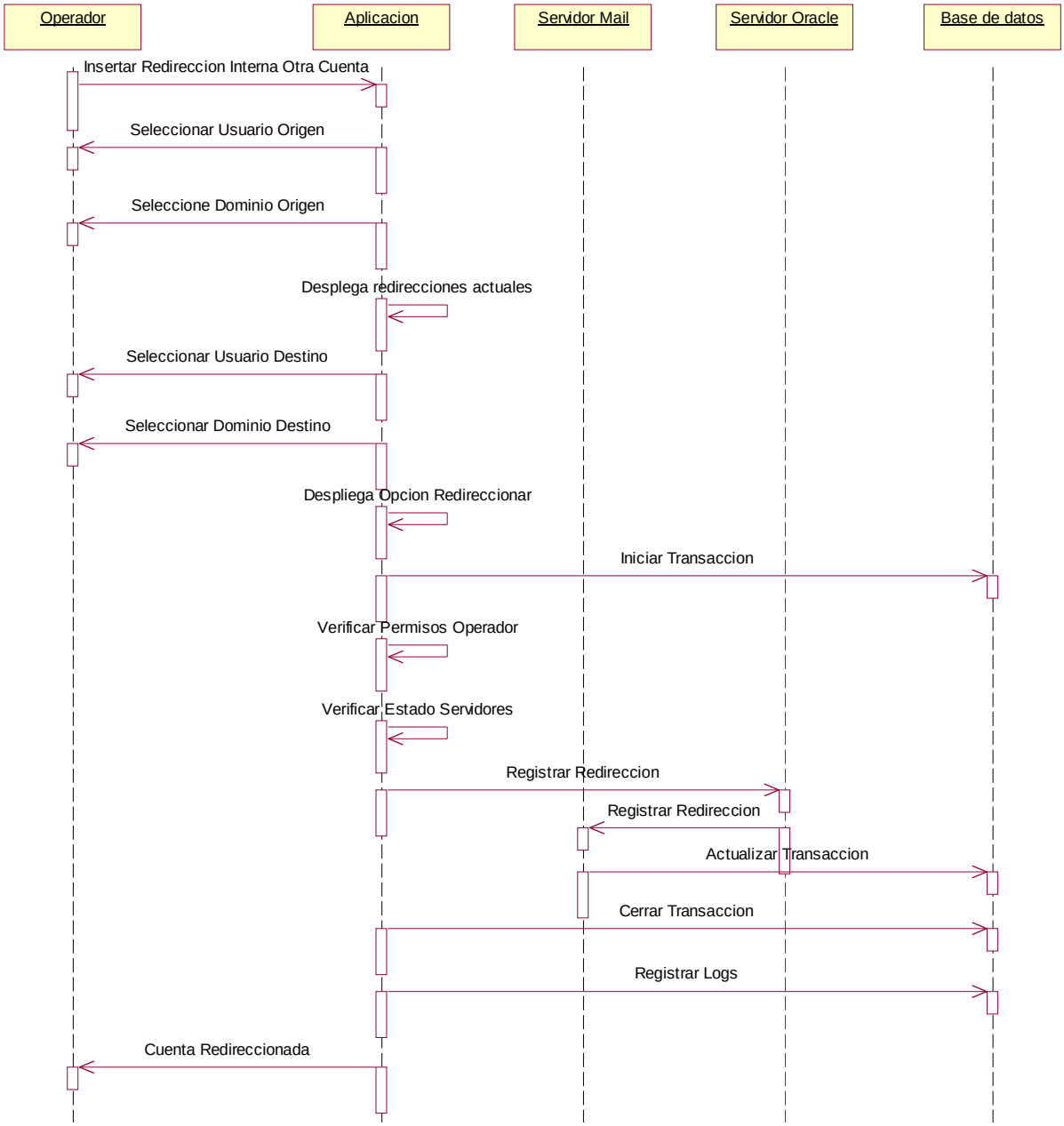


Figura 1-33

Registro usuario

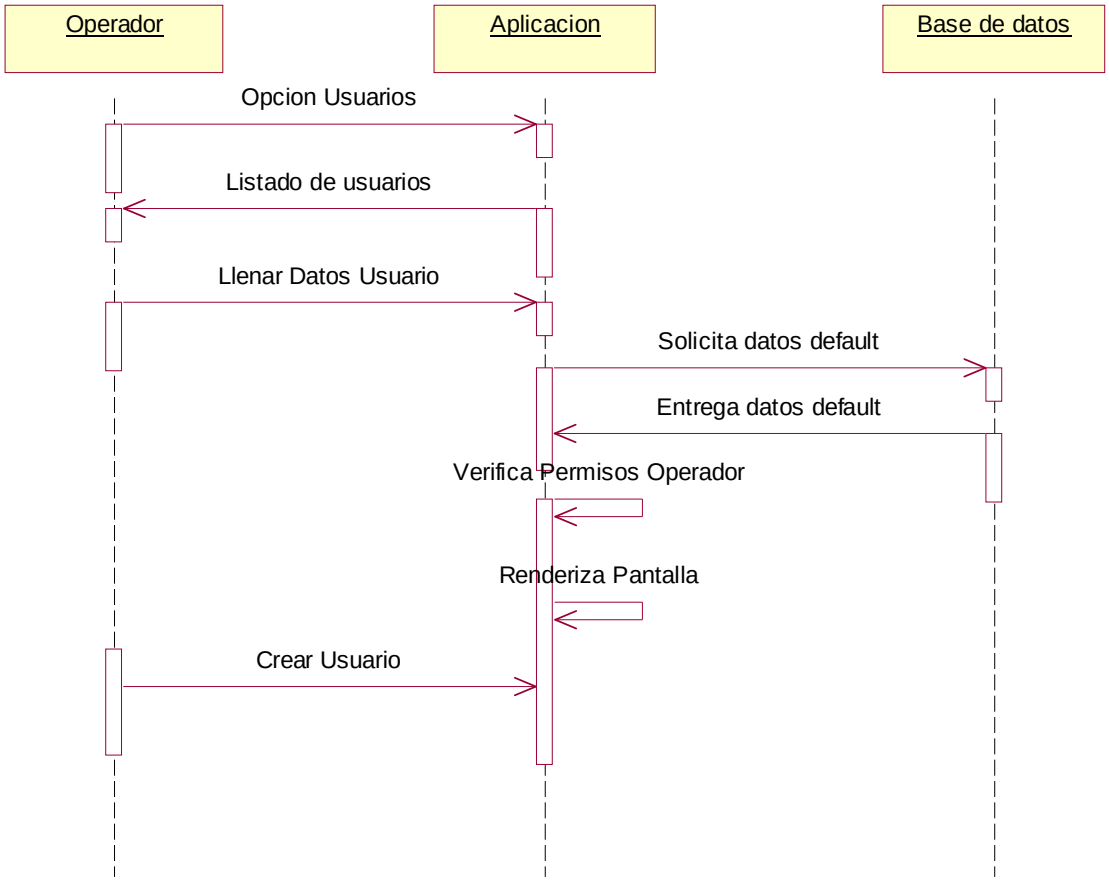


Figura 1-34

Eliminación redirecciones internas a otras cuentas

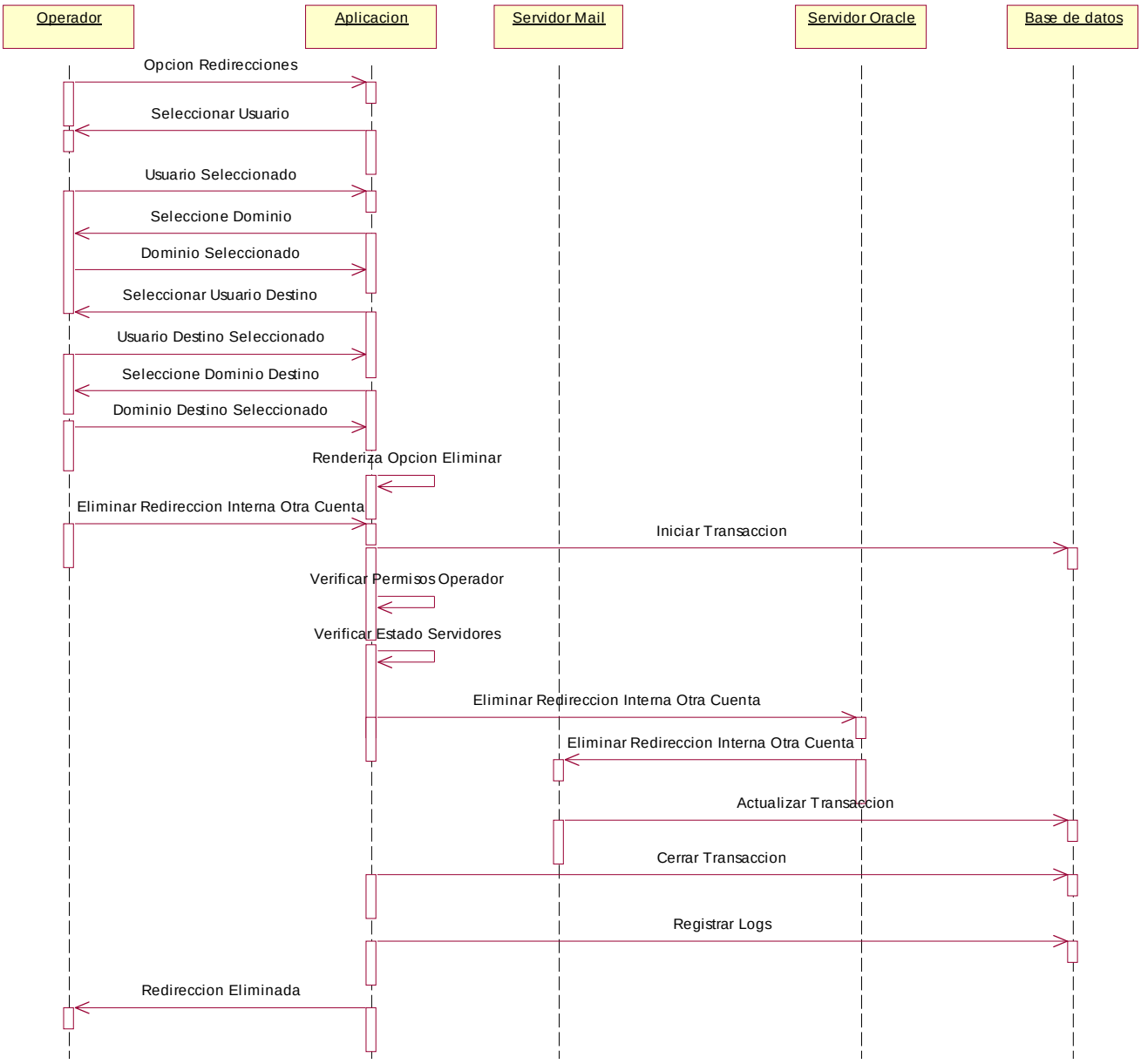


Figura 1-35

Creación de redirecciones externas

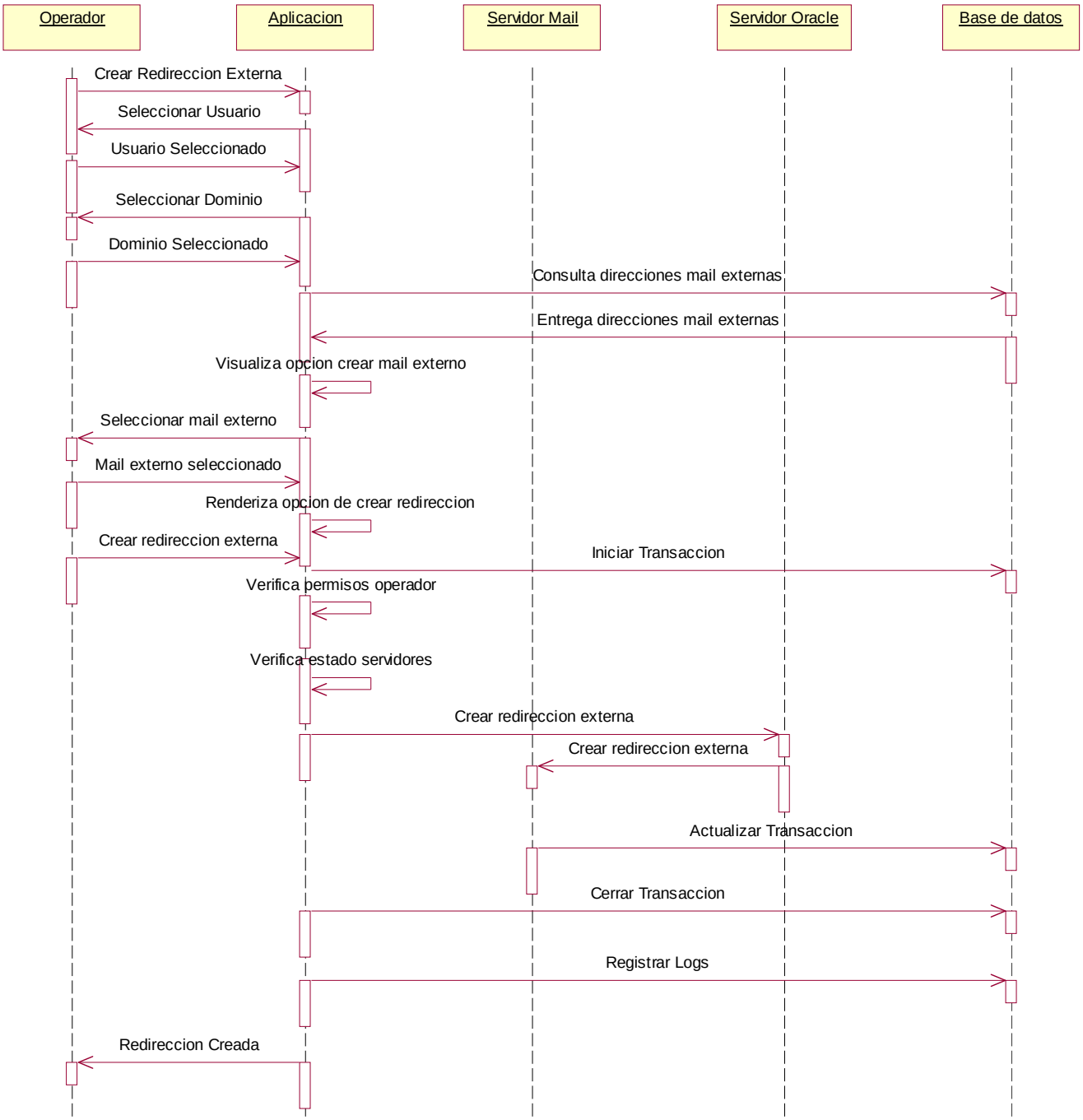


Figura 1-36

Eliminación redirecciones externas

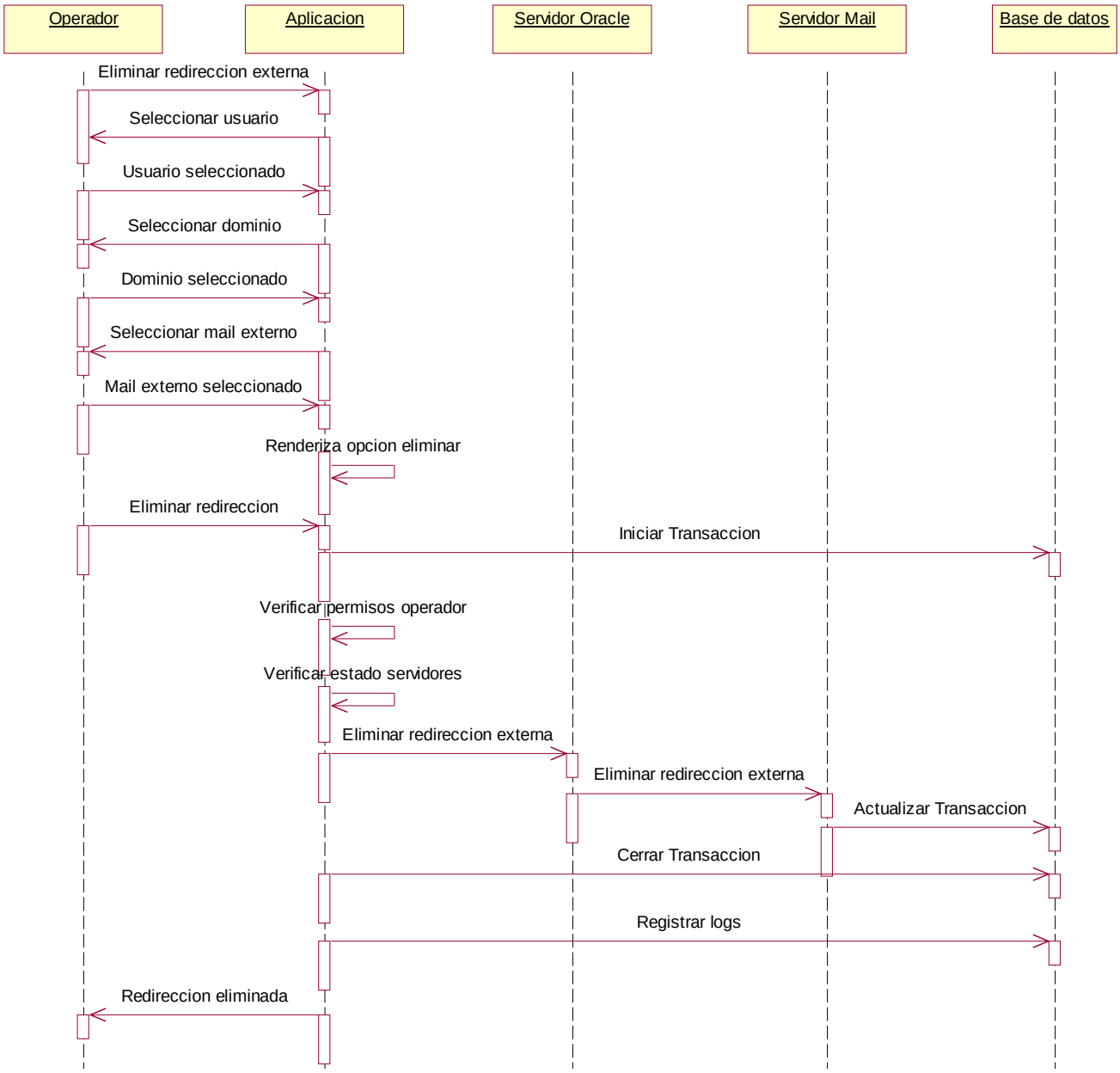


Figura 1-37

Crear asignación DHCP

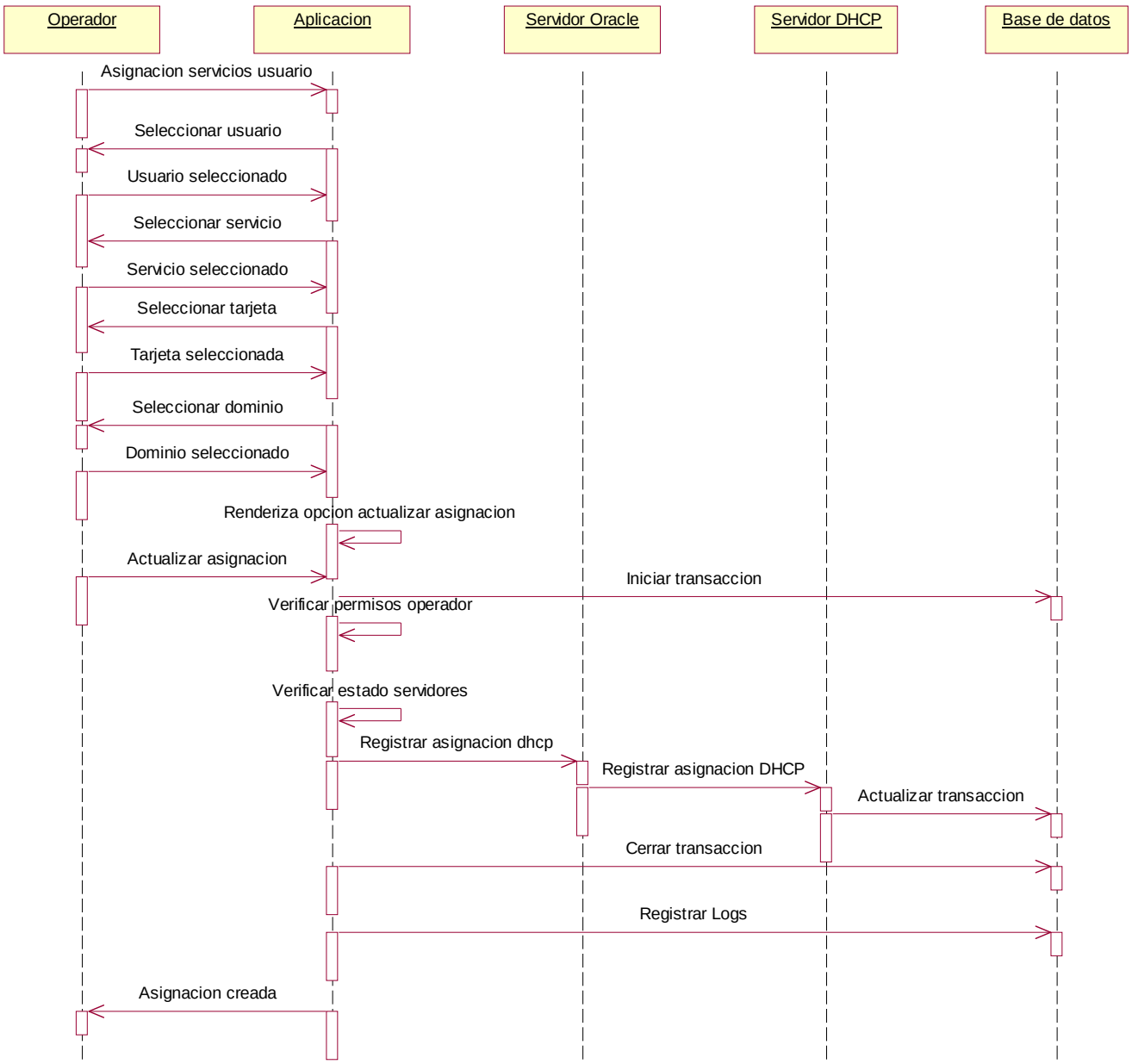


Figura 1-38

Eliminar asignación DHCP

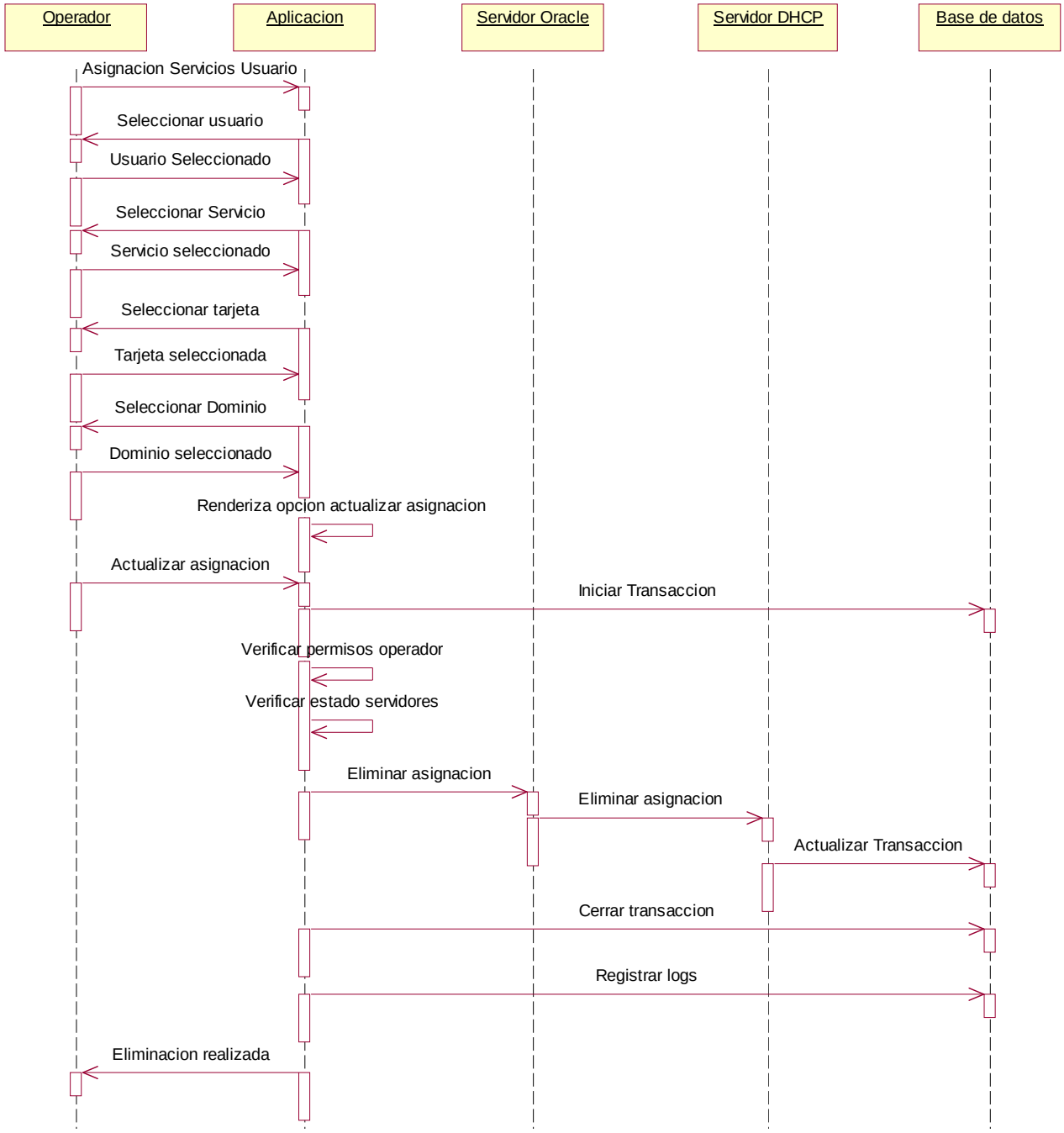


Figura 1-39

Notificación estado Srv. Remoto

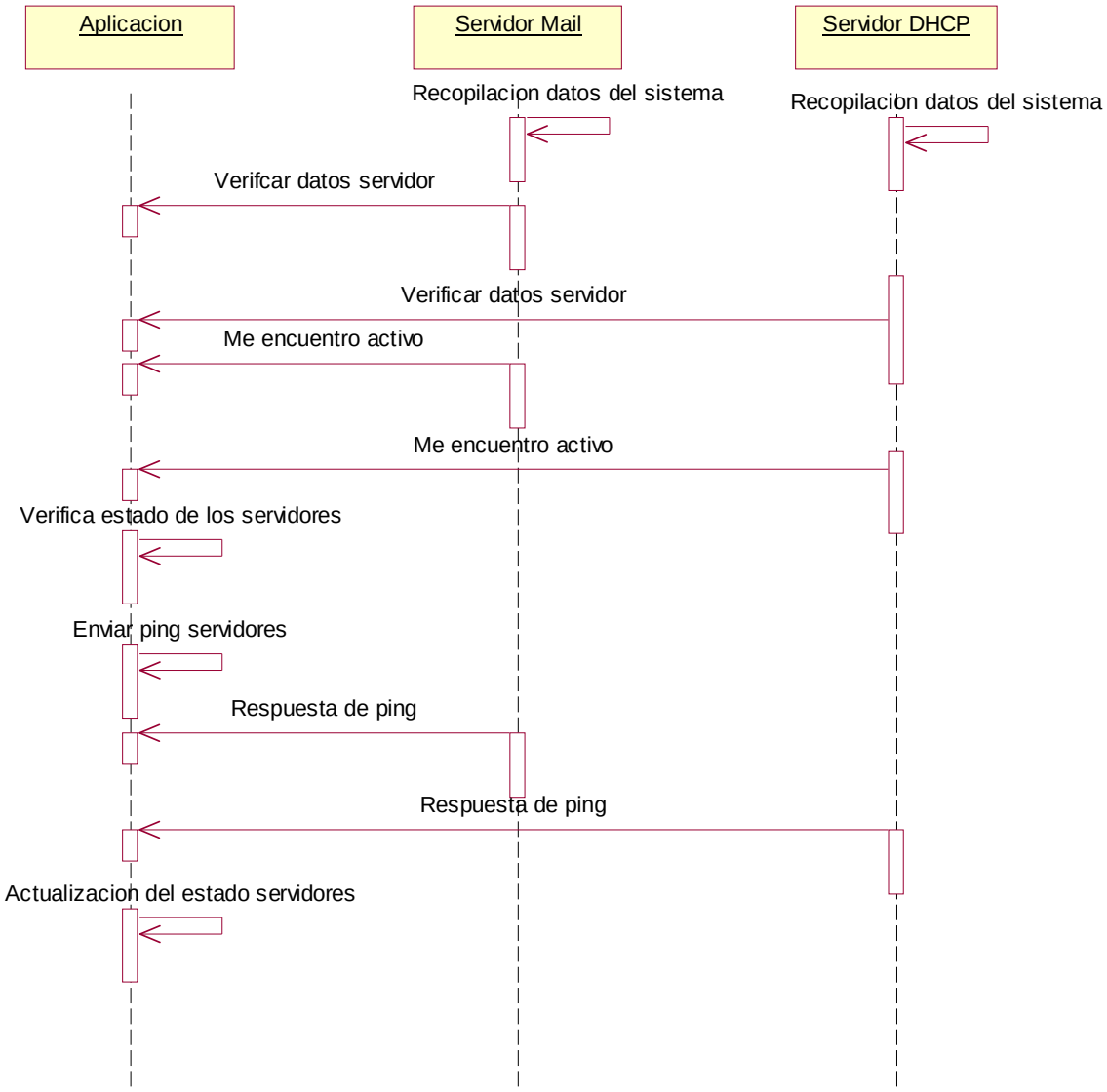


Figura 1-40

Sincronización servidores

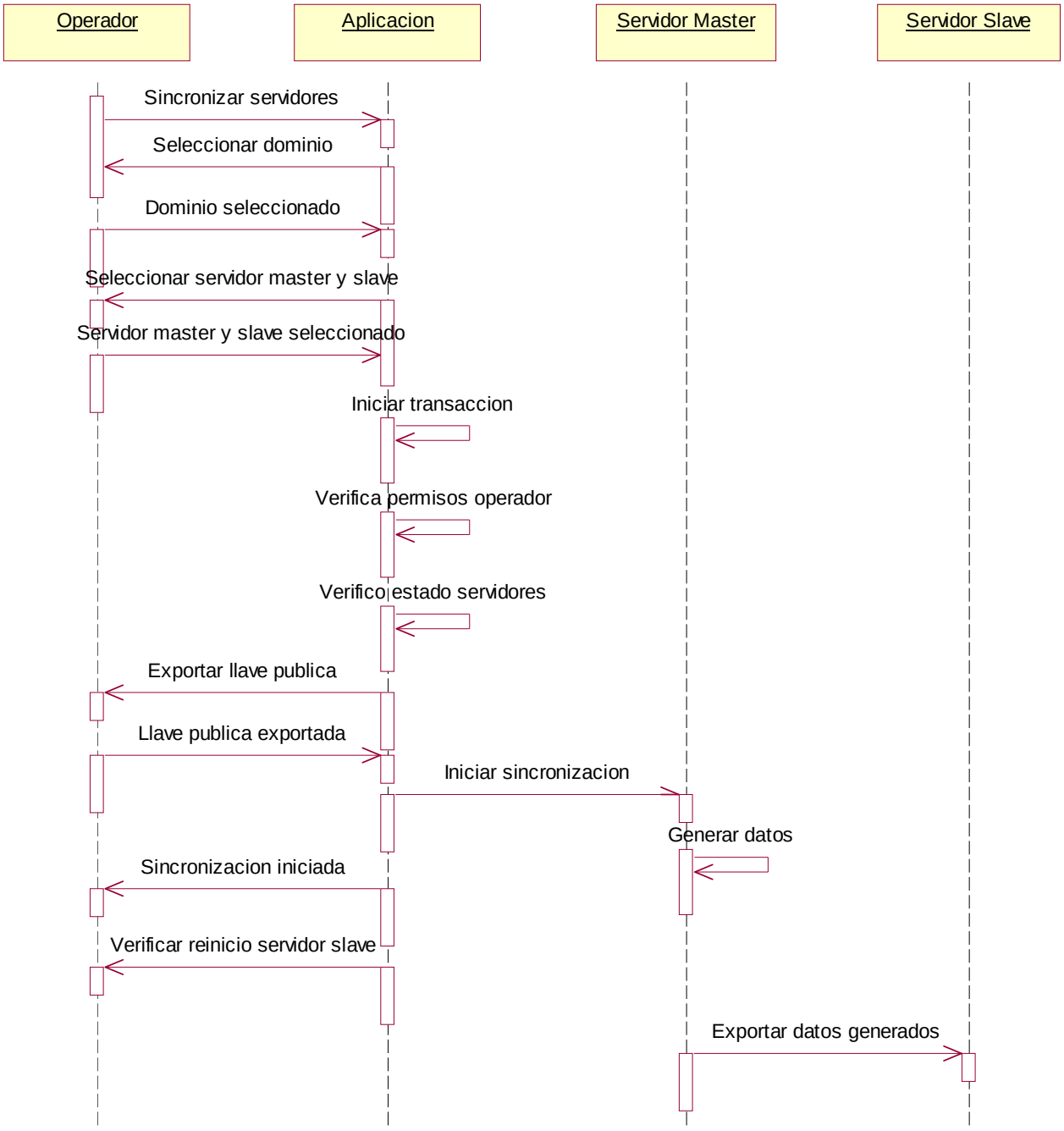


Figura 1-41

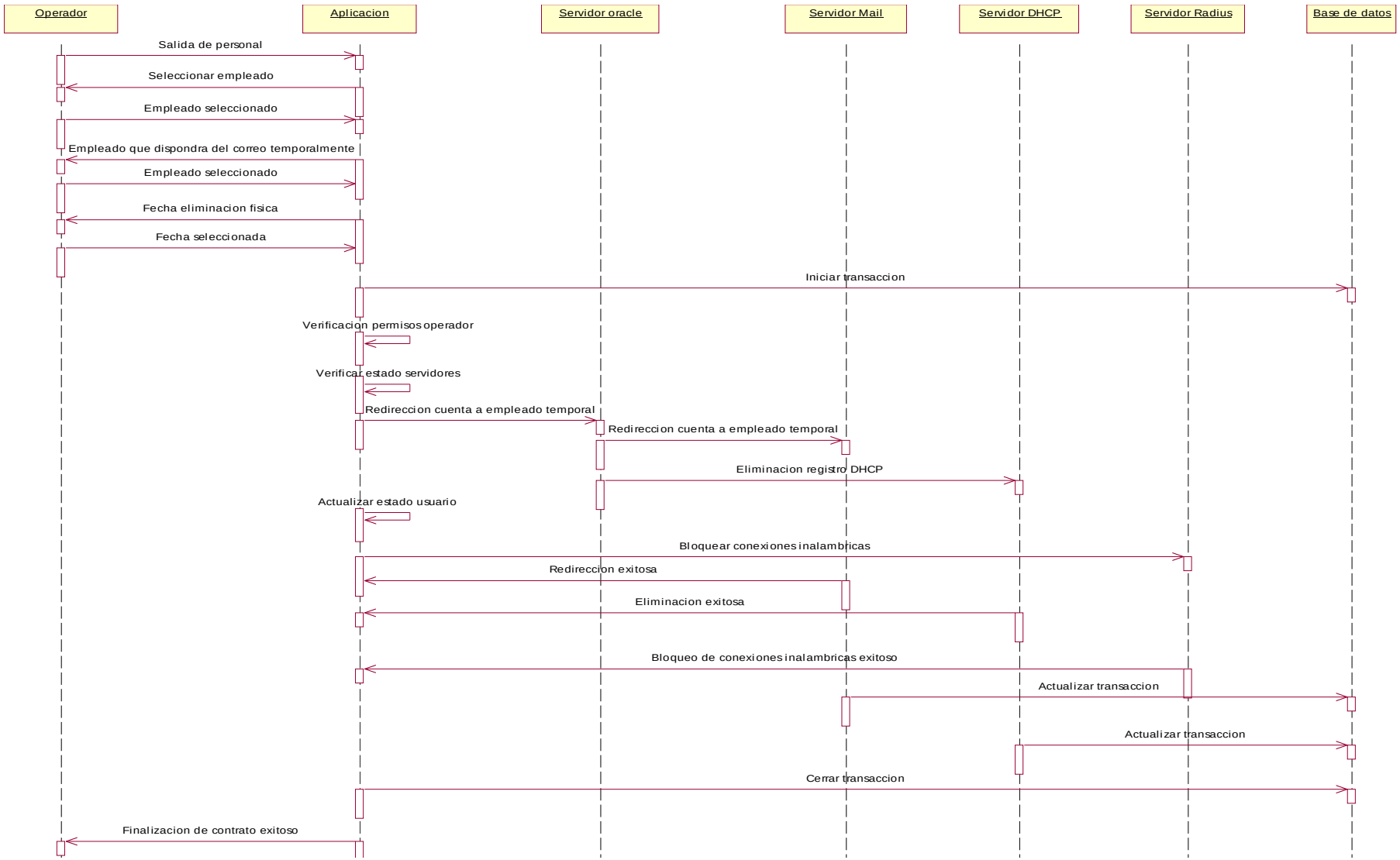

Operador
Aplicacion
Servidor oracle
Servidor Mail
Servidor DHCP
Servidor Radius
Base de datos
Salida de personal
Seleccionar empleado
Empleado seleccionado
Empleado que dispondra del correo temporalmente
Empleado seleccionado
Fecha eliminacion fisica
Fecha seleccionada
Iniciar transaccion
Verificacion permisos operador
Verificar estado servidores
Redireccion cuenta a empleado temporal
Redireccion cuenta a empleado temporal
Eliminacion registro DHCP
Actualizar estado usuario
Bloquear conexiones inalambricas
Redireccion exitosa
Eliminacion exitosa
Bloqueo de conexiones inalambricas exitoso
Actualizar transaccion
Actualizar transaccion
Cerrar transaccion
Finalizacion de contrato exitoso

Figura 1-42

Salida de vacaciones del personal

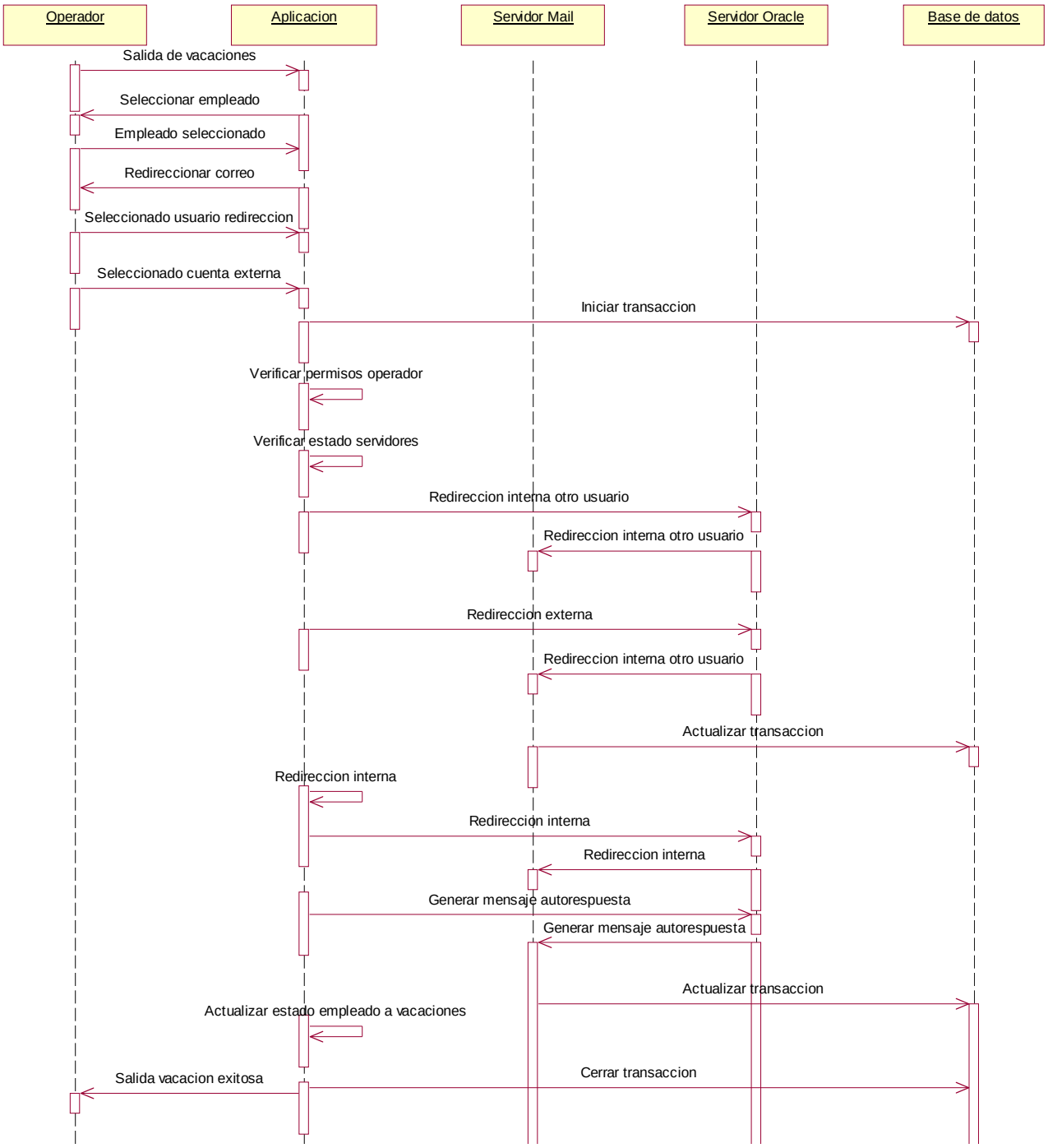


Figura 1-43

Regreso de vacaciones del empleado

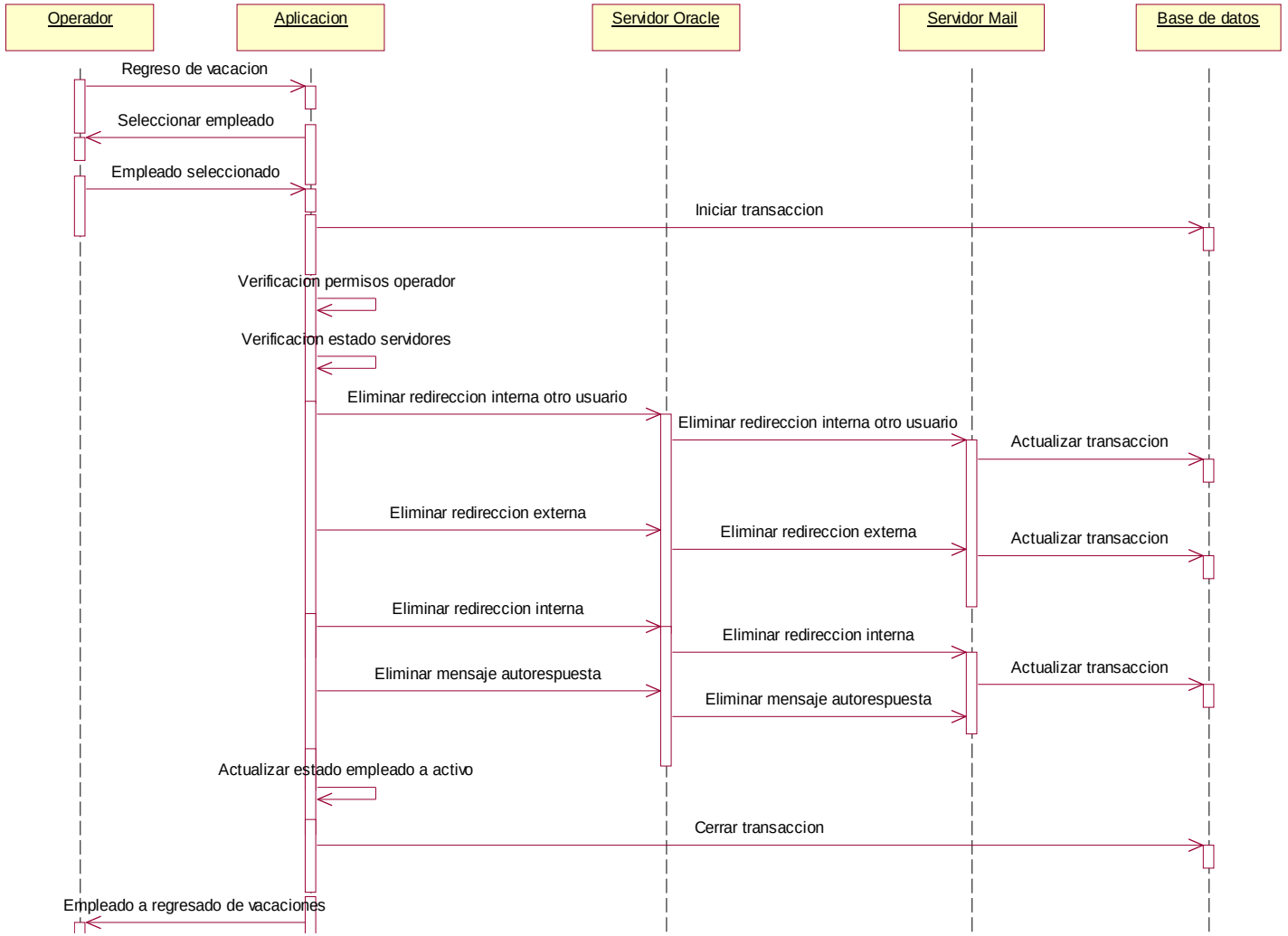


Figura 1-44

Verificación de credenciales BD.

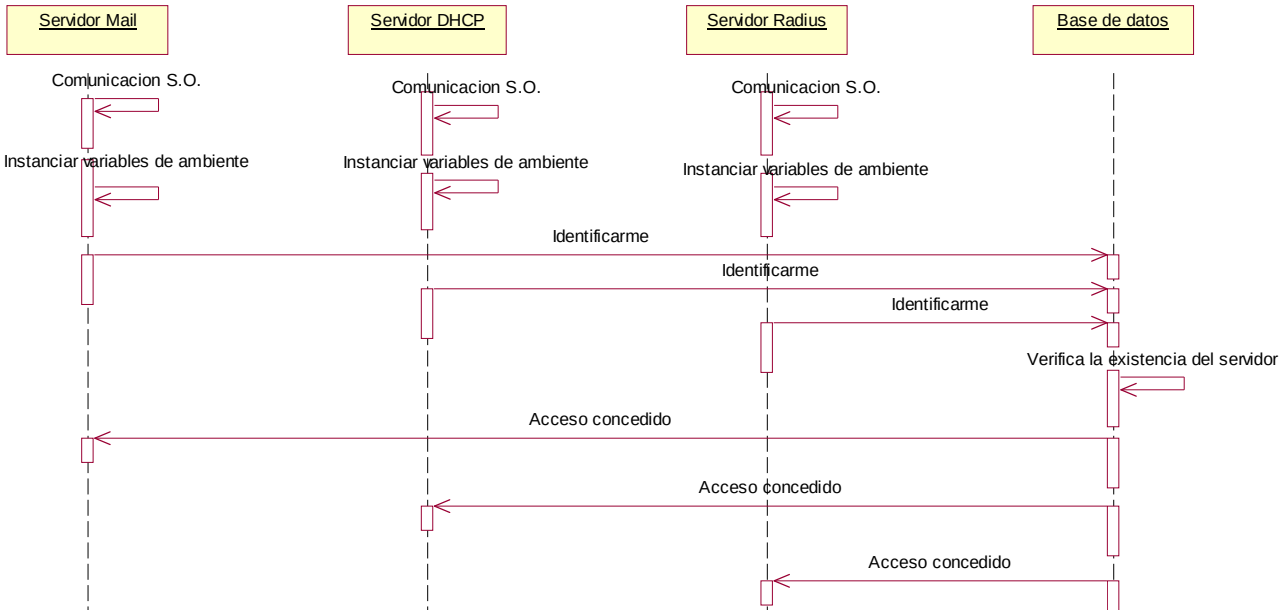


Figura 1-45

Registro log ejecución

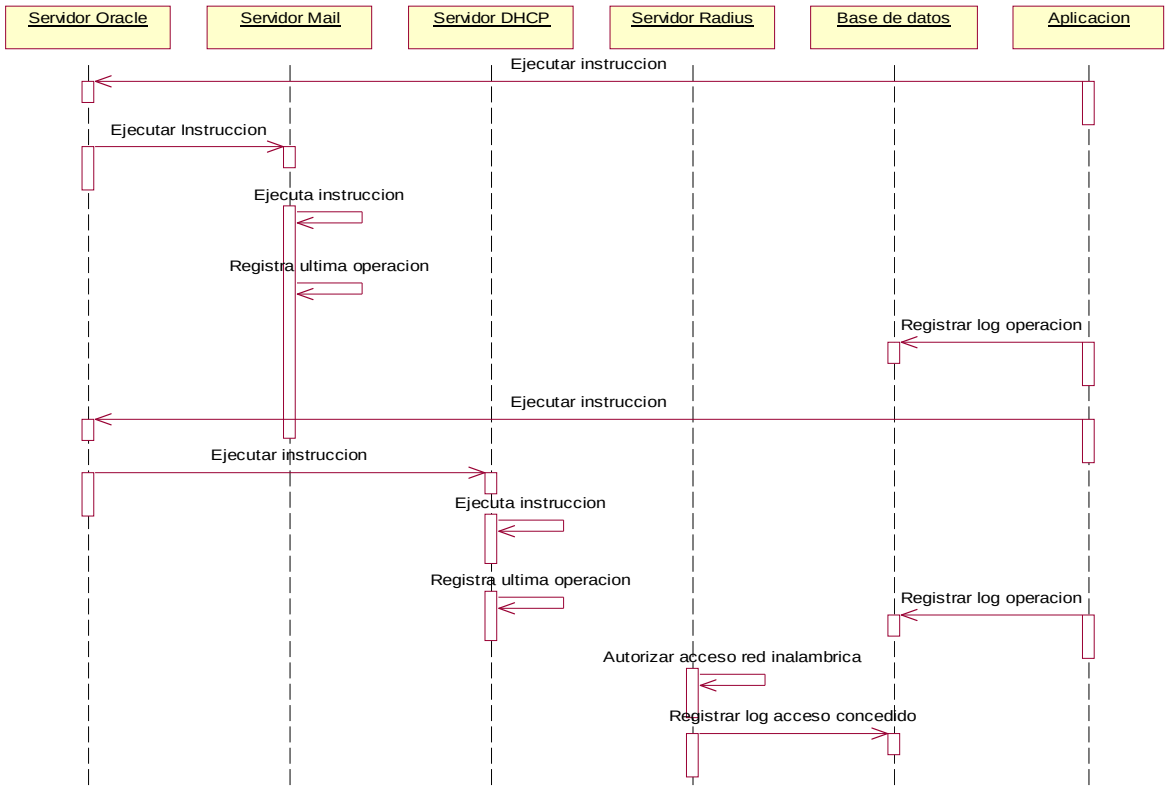


Figura 1-46

Solicitar Ejecución Instrucción Srv. Remoto

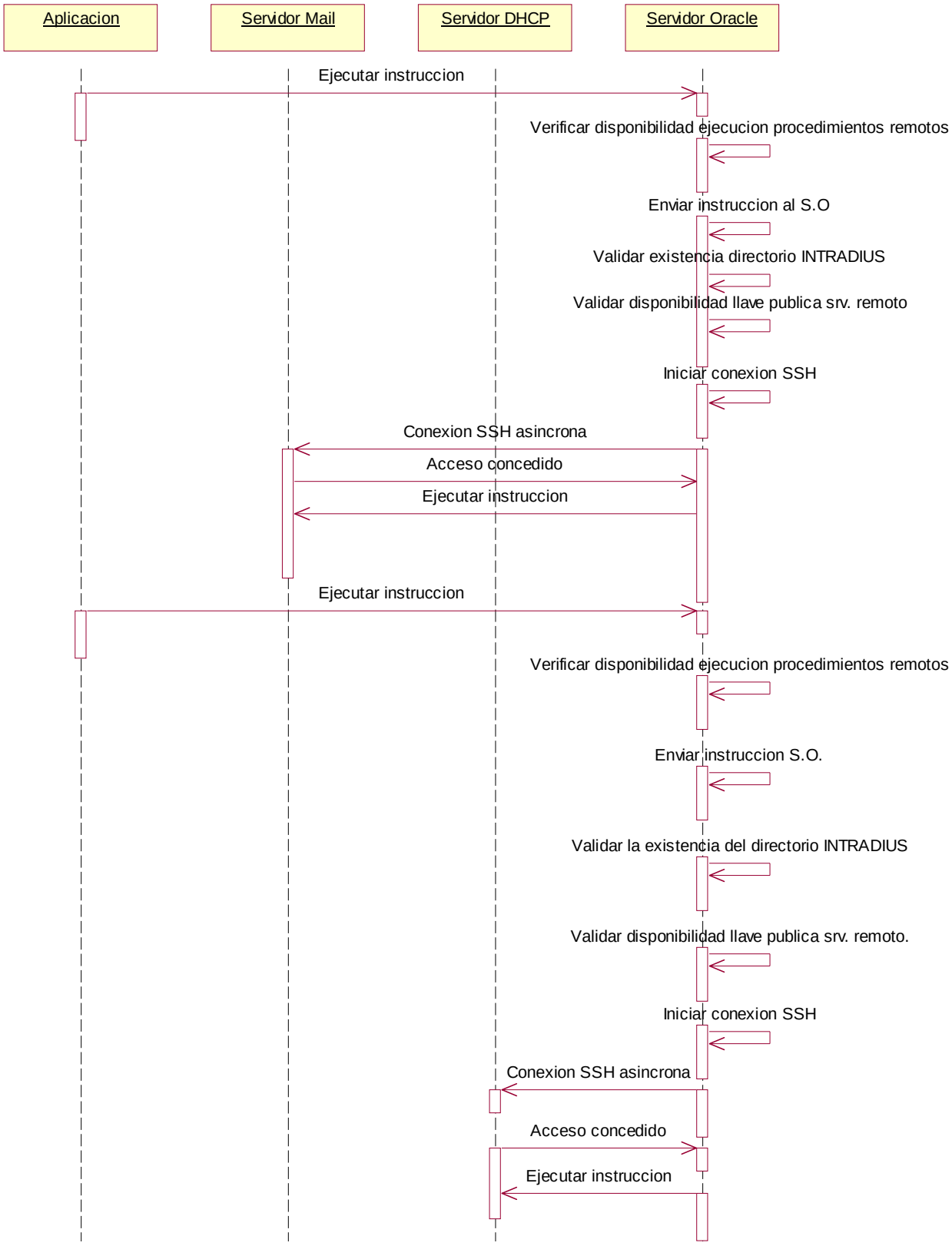


Figura 1-47

Registro ultimo error de conexión

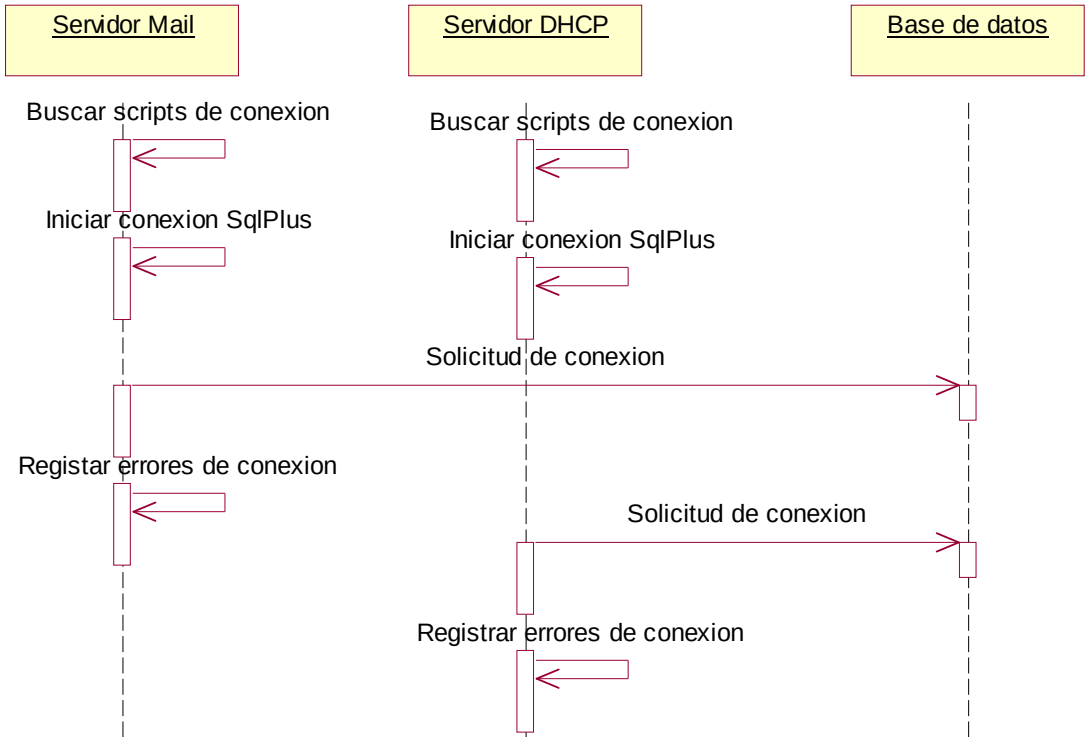


Grafico 1-48

Solicitud Autenticación Wireless Usuario

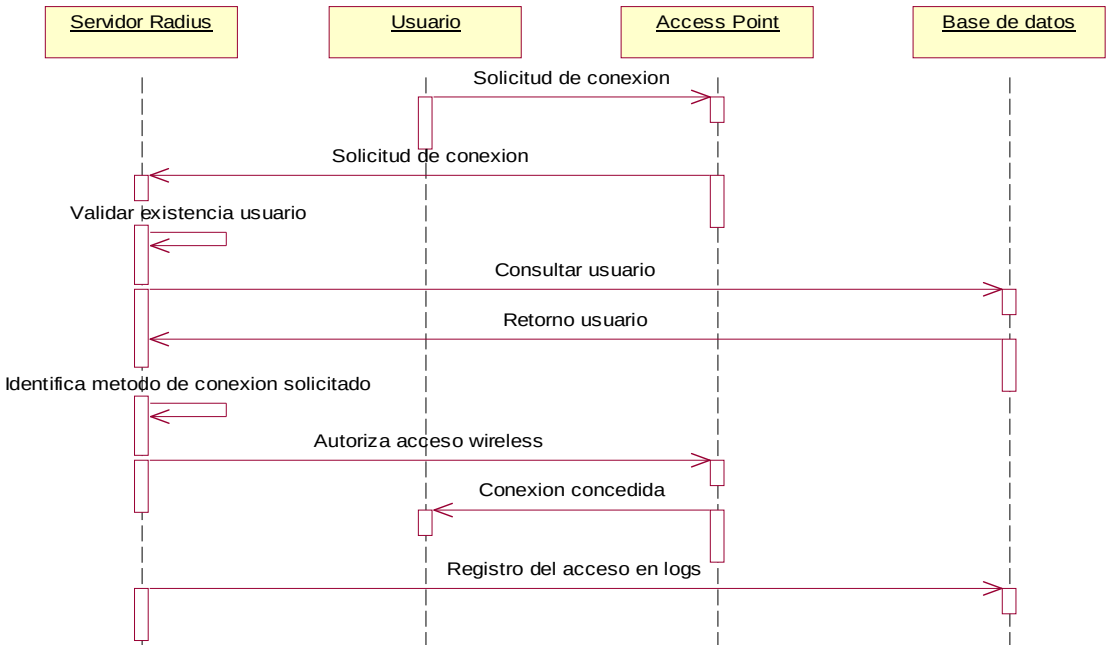


Figura 1-49

Conexión permanente base de datos

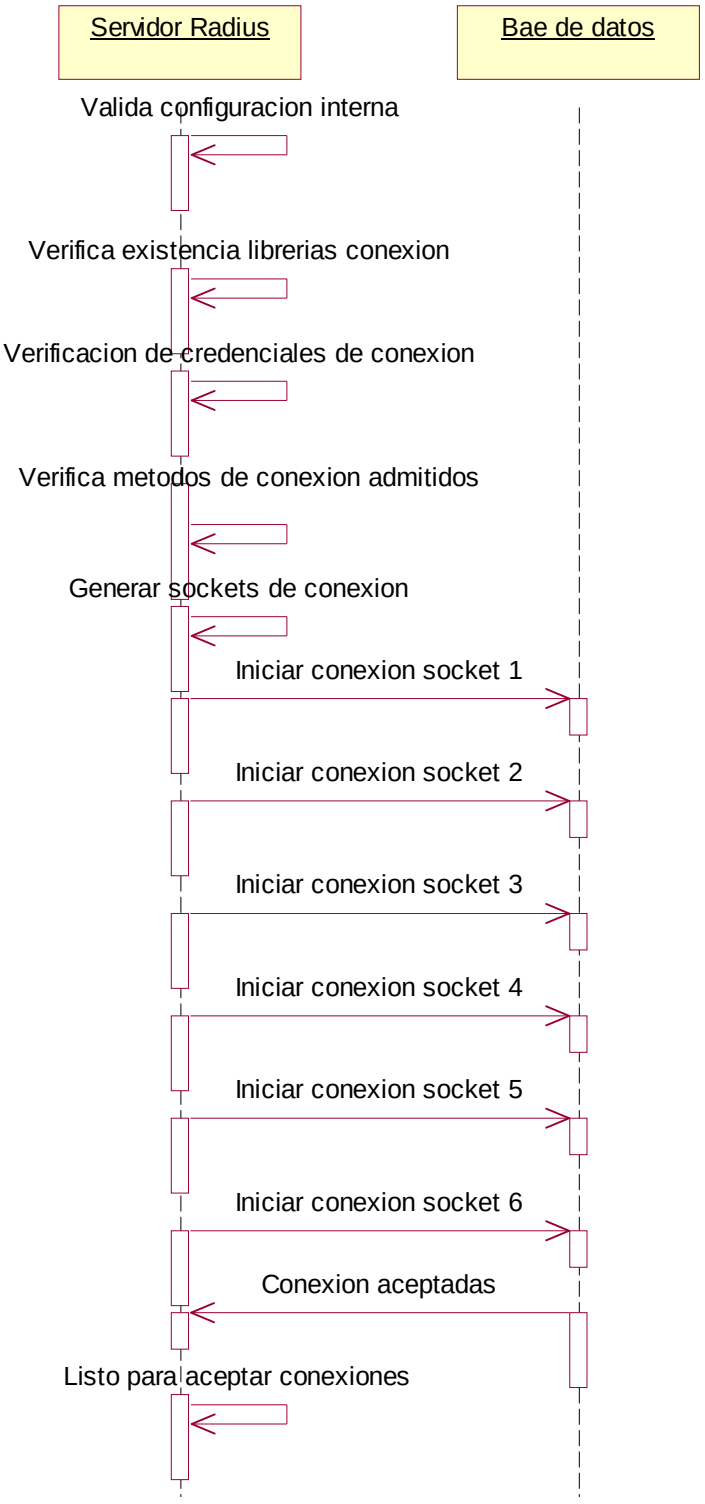


Figura 1-50

1.2.2.4 Diagramas de colaboración

El objetivo de los diagramas de colaboración es la representación del comportamiento de las instancias de las clases de un sistema para conseguir un objetivo, para ello hacen uso de el envío de mensajes que son numerados secuencialmente. Se podría decir también que un diagrama de colaboración es un grafo, donde los nodos son los objetos y los arcos los enlaces que transportan los mensajes.

"Elementos de un diagrama de colaboración

- Objetos o Roles: nodos del grafo.
- Enlaces o comunicaciones: arcos del grafo.
- Mensajes: llevan número de secuencia y flecha dirigida.
- Anidamiento: se utiliza la numeración decimal Ej.: 1, 1.1, 1.1.1 ....
- Iteración: colocar un * antes del número de secuencia y una cláusula de condición, si es necesario. ej. *[x>0].
- Bifurcación: los caminos alternativos tendrán el mismo número de secuencia, seguido del número de subsecuencia, y se deben distinguir por una condición."

Para poder elaborar un diagrama de colaboración es a menudo requerido la disponibilidad de:

- Diagrama de casos de uso
- Diagramas de Secuencias
- Diagramas de clases

Para el sistema en que nos hallamos diseñando surgen los diagramas de colaboración visualizado en las figuras inferiores.

Eliminación usuario

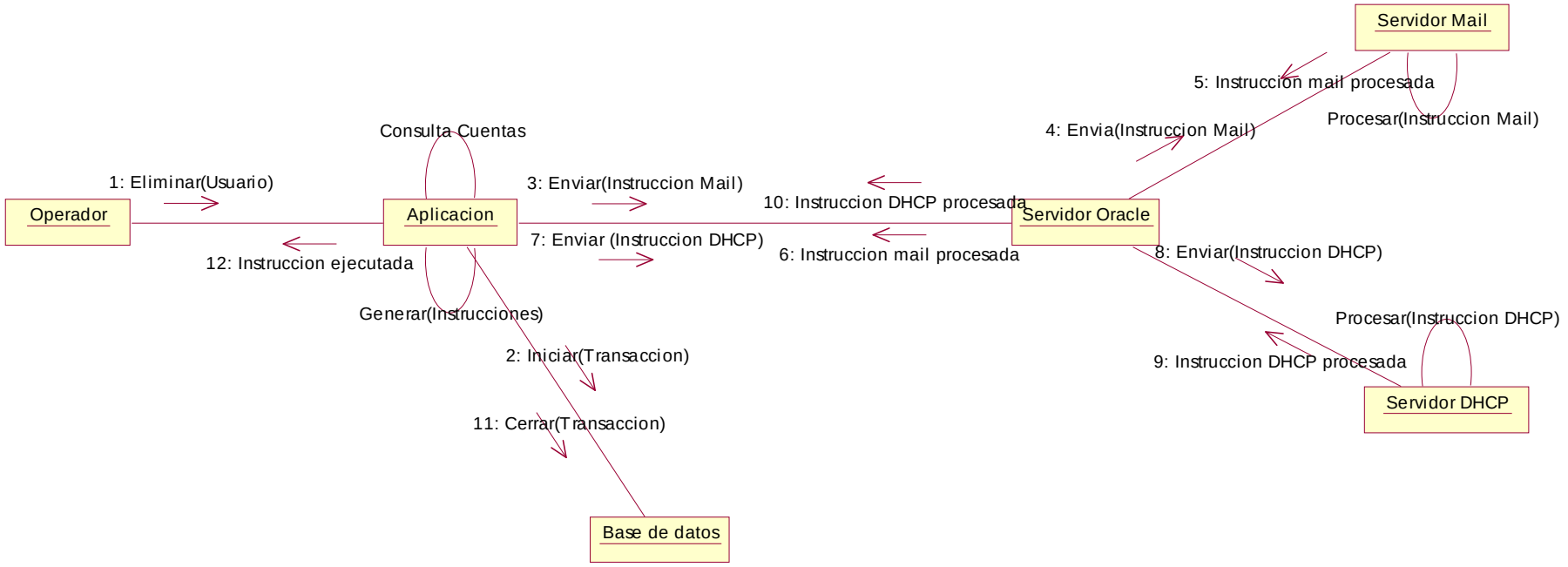


Figura 1-51

Finalización Contrato

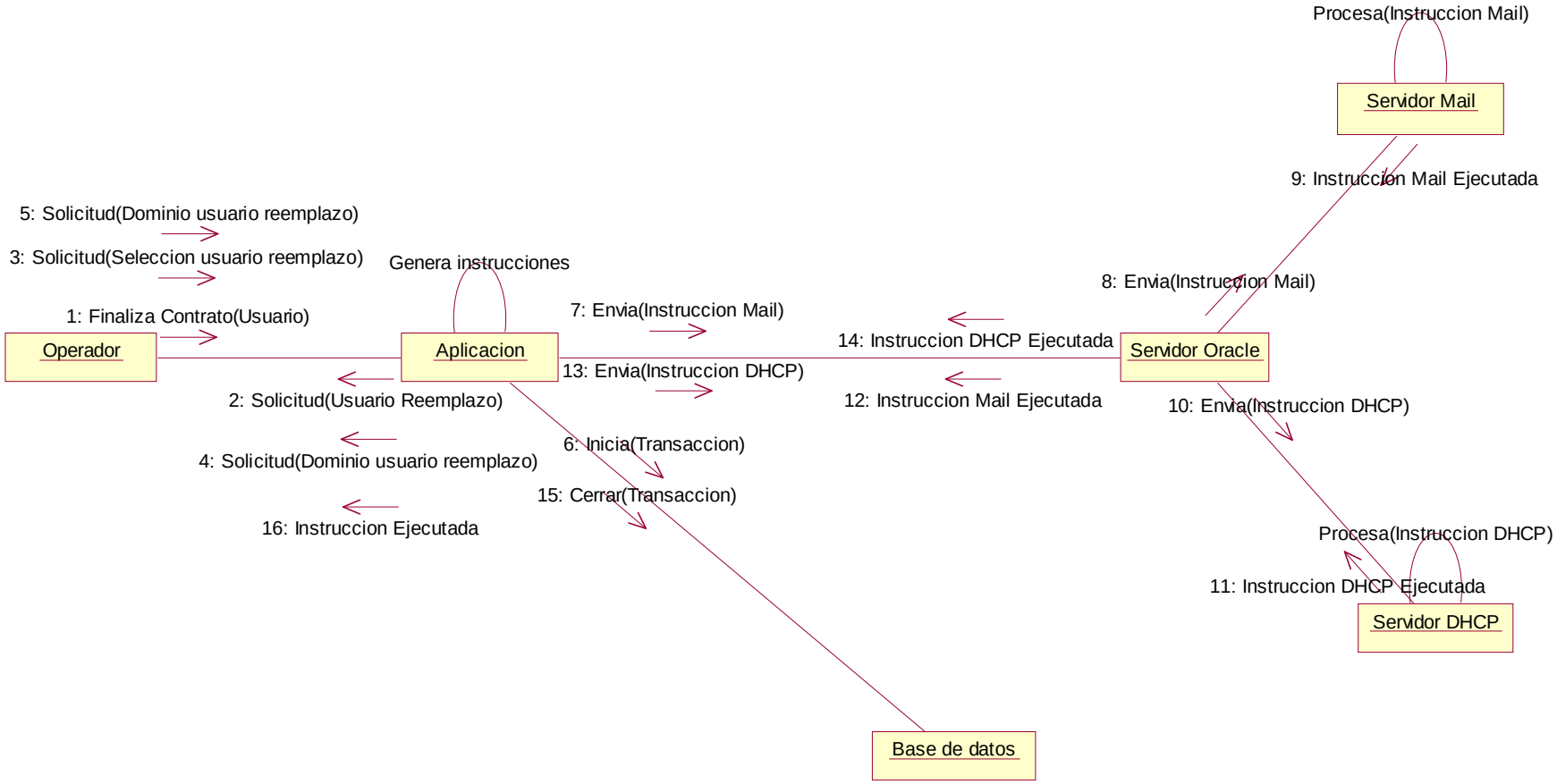


Figura 1-52

Redireccionar cuenta de correo

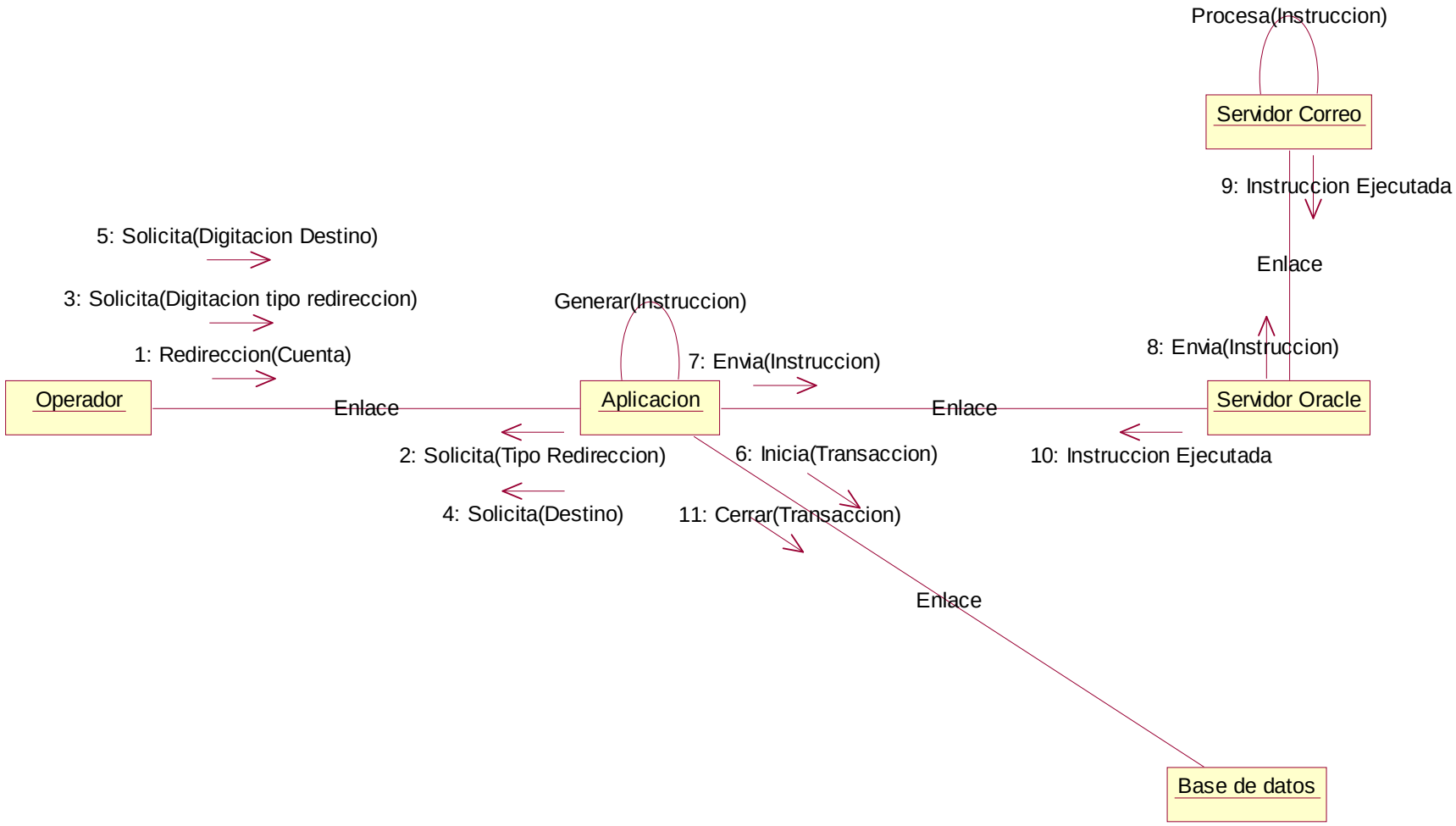


Figura 1-53

Registrar cuenta de correo

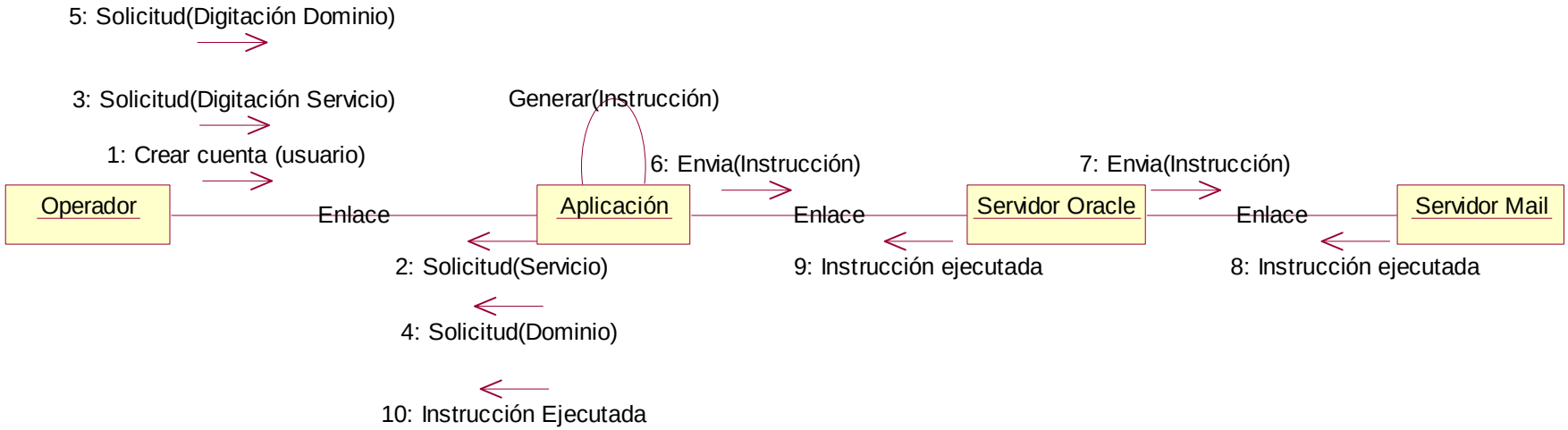


Figura 1-54

Registro DHCP

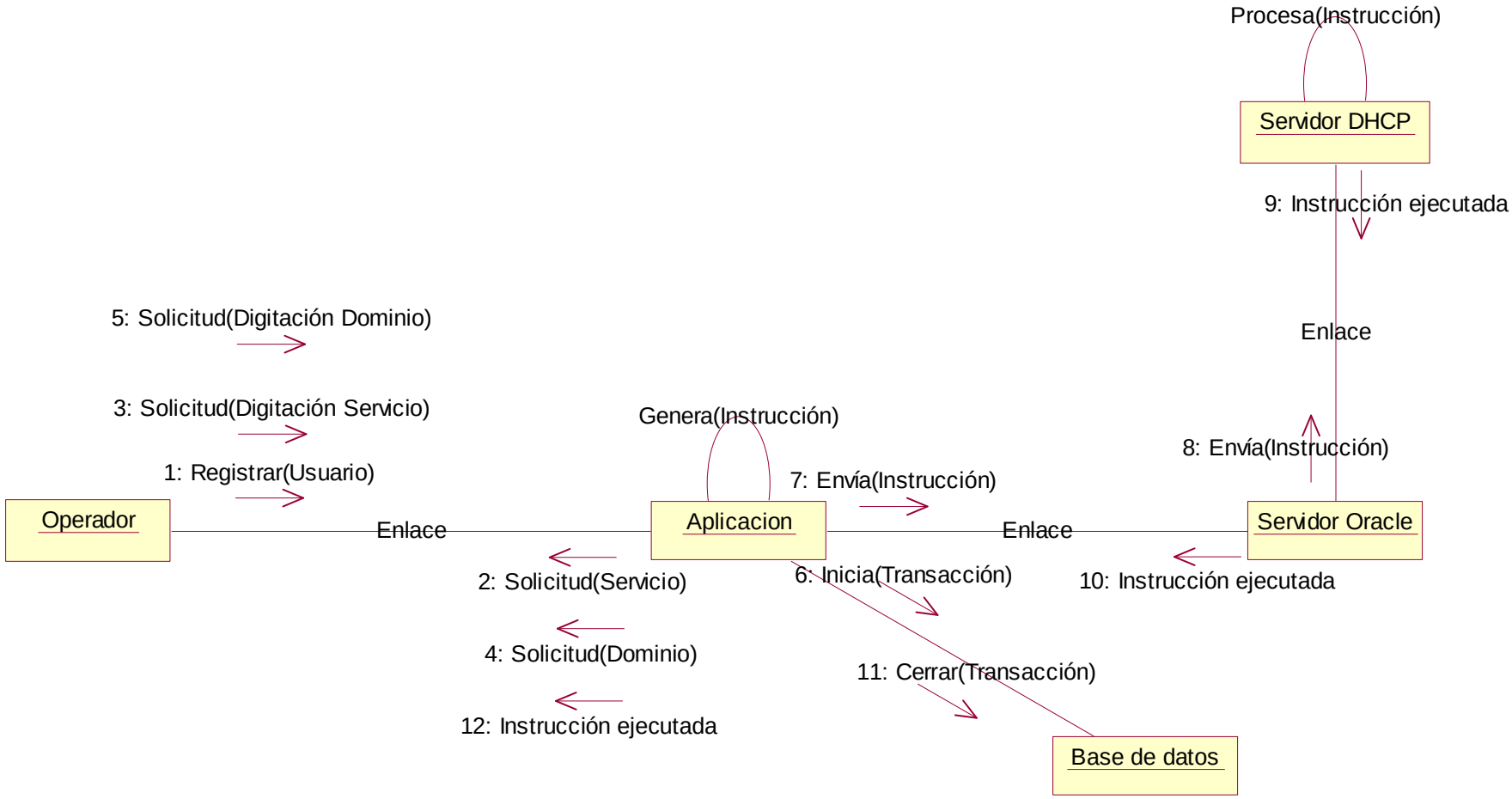


Figura 1-55

Regreso de vacaciones

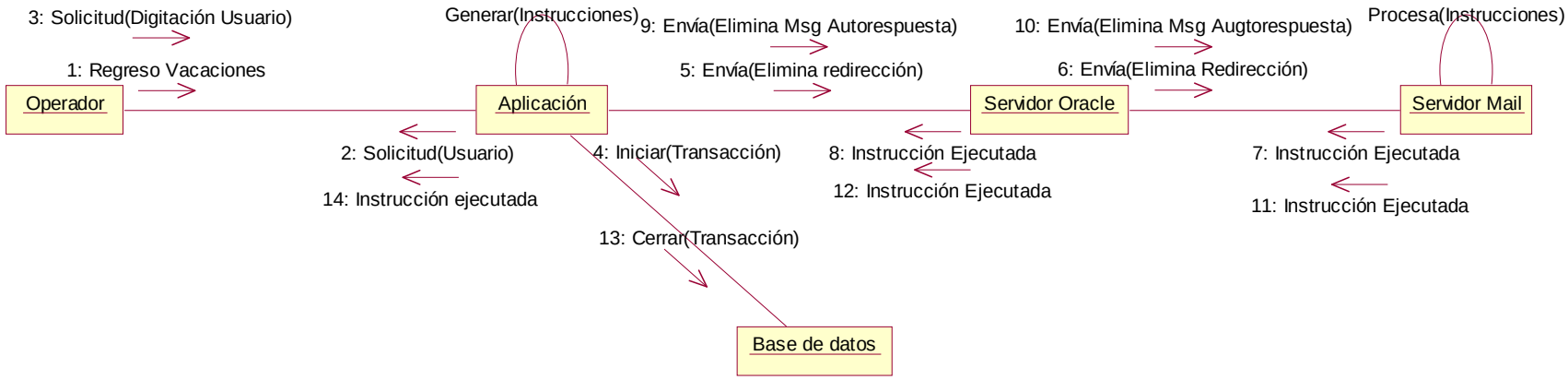


Figura 1-56

Salida Vacaciones

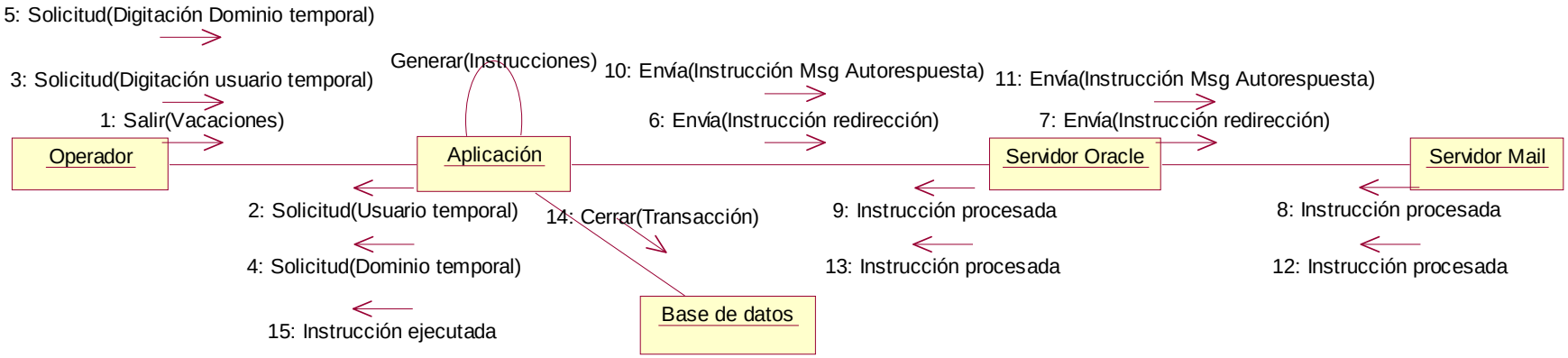


Figura 1-57

1.2.2.5 Diagramas de estado

Estos diagramas son muy útiles cuando es necesario representar los objetos y los eventos que cambian su estado en determinadas circunstancias; extienden la representación de los casos de uso.

Un evento es un suceso importante a considerar en un sistema. Un estado es la condición de un objeto en un momento determinado: el tiempo que transcurre entre eventos. Una transición es una relación entre dos estados, e indica que, cuando ocurre un evento, el objeto pasa del estado anterior al siguiente.

En UML, los estados se representan mediante rectángulos con sus vértices redondeados. Las transiciones se representan mediante flechas con el nombre del evento respectivo. Se suele incluir un estado inicial el mismo que es representado por un círculo de relleno negro.

En el sistema en estudio se ha determinado los siguientes estados:

Aplicación

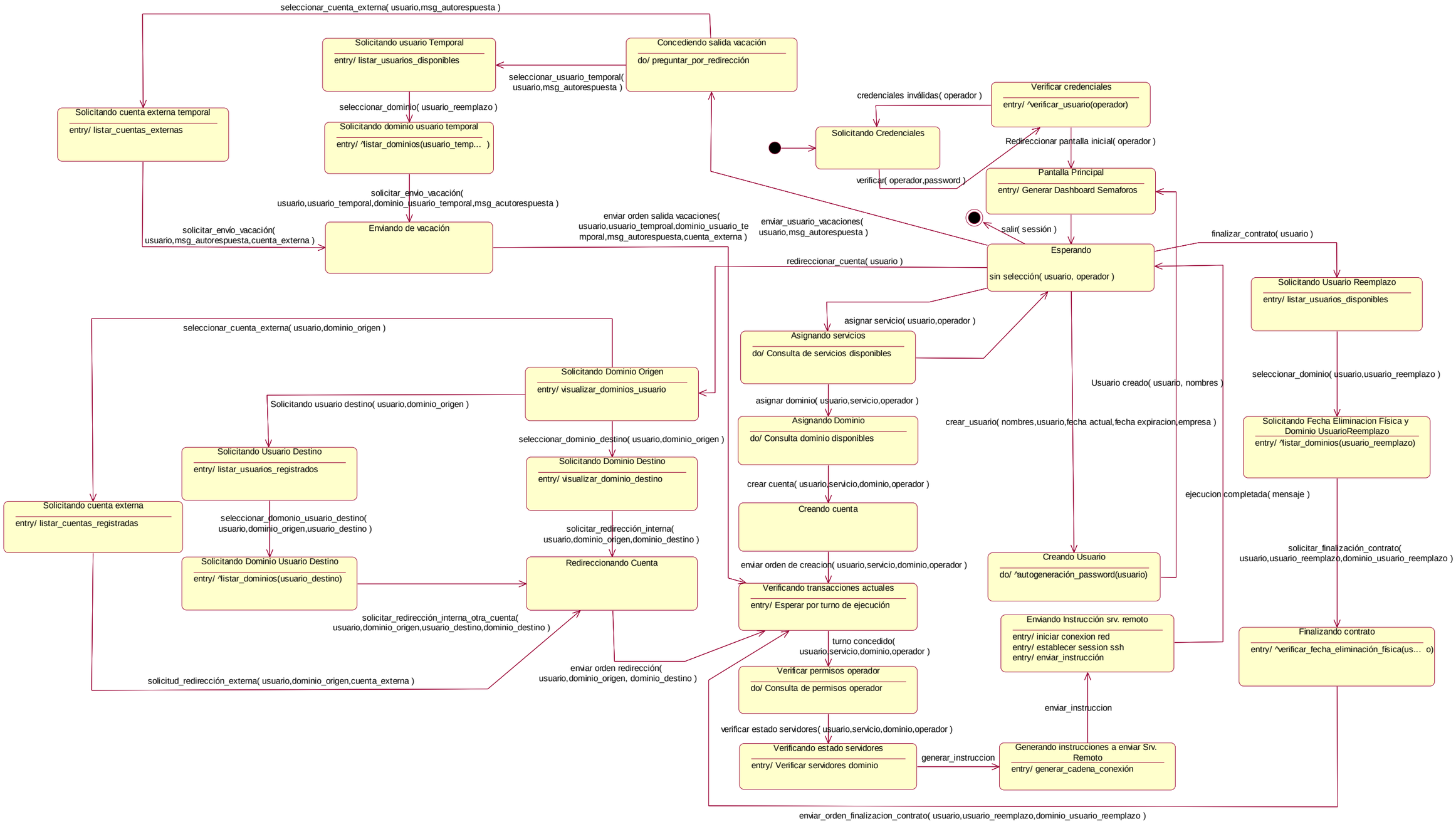


Figura 1-58

Cuentas Correo

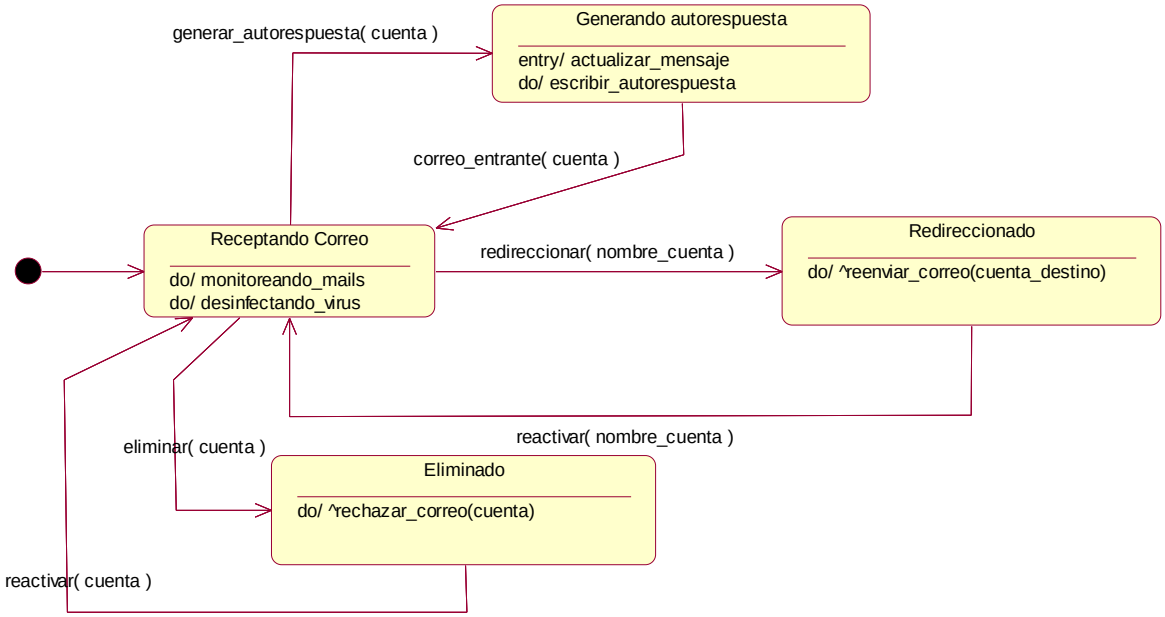


Figura 1-59

Menú Aplicación

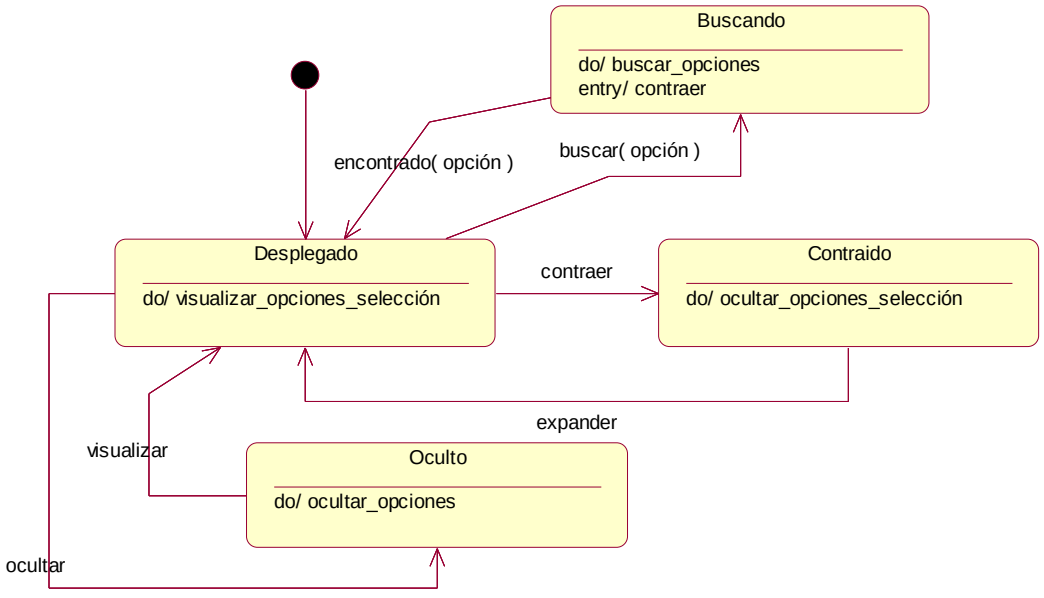


Figura 1-60

Servidor DHCP

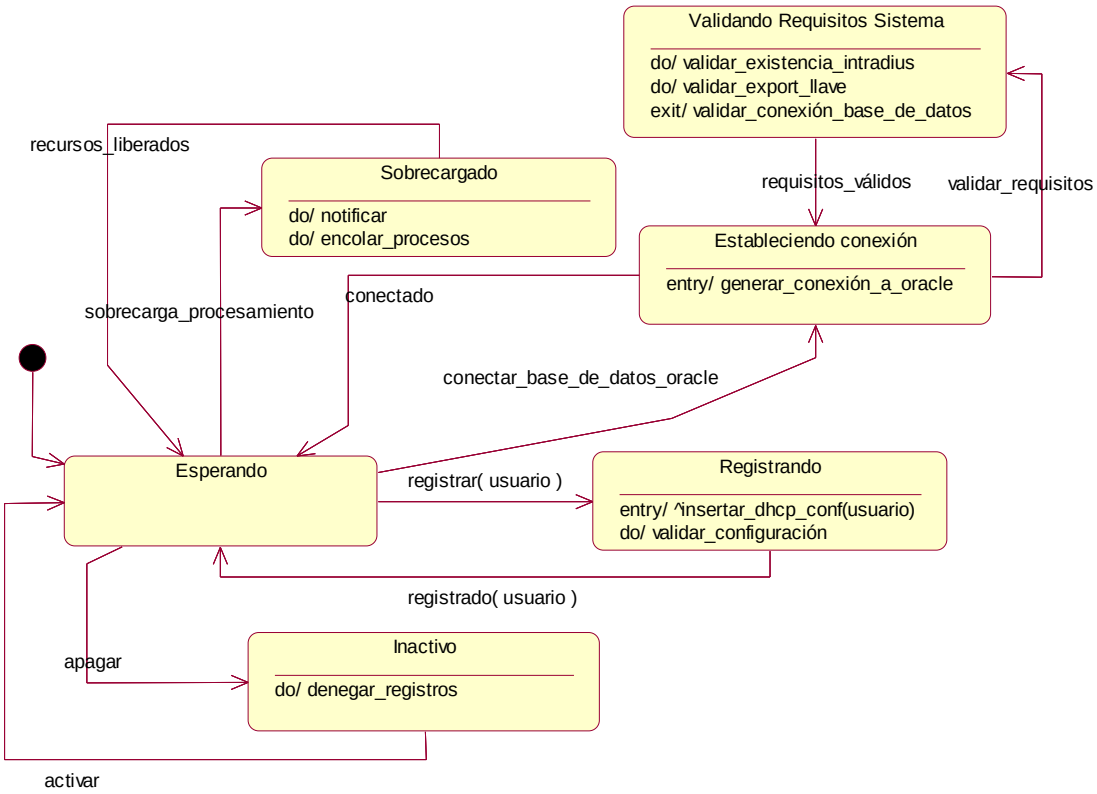


Figura 1-61

Servidor Mail

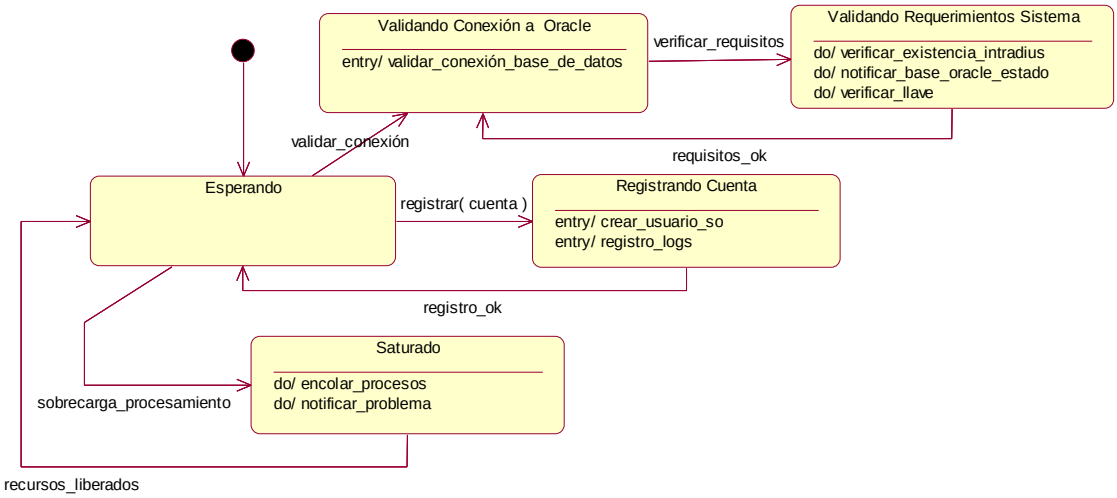


Figura 1-62

Servidor Oracle

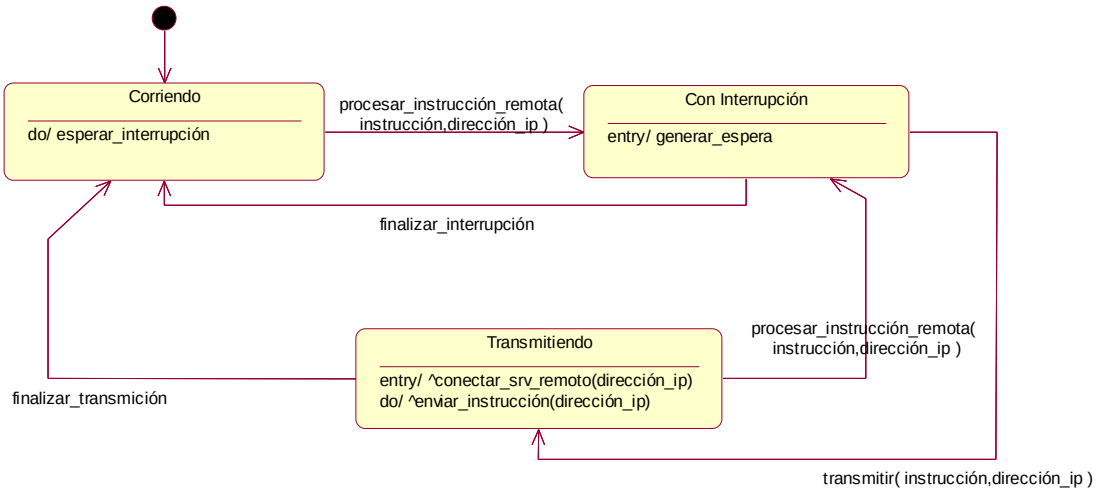


Figura 1-63

Servidor Radius

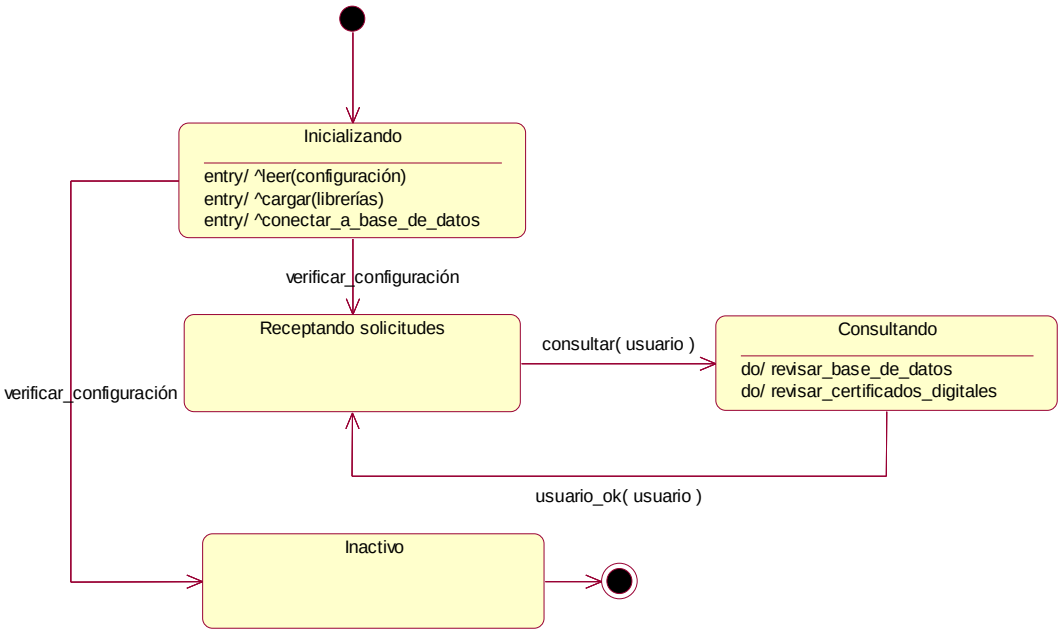


Figura 1-64

Access Point

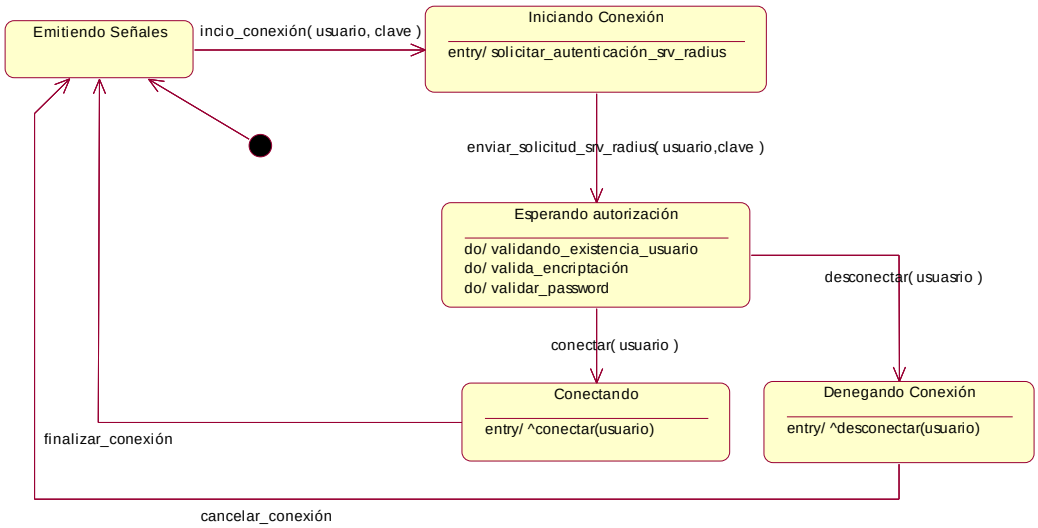


Figura1-65

1.2.2.6 Diagramas de componentes

Los diagramas de componentes permiten disponer de una visión macro de la arquitectura adoptada por el software construido, porque evidencia cómo se organiza los elementos constituyentes (archivos, código, módulos, etc.) y las relaciones existentes entre ellos. Podríamos decir entonces que un diagrama de componentes pretende representar a un sistema en particular como un conjunto de elementos necesarios para su funcionamiento.

Los diagramas de componentes están constituidos por una serie de artefactos (componentes, Interfaces, relaciones), que organizados de forma adecuada son un clara representación de cómo funcionará el sistema cuando este sea implementado. Es necesario entender el concepto de cada elemento que conforma un diagrama de componentes para poder comprender como está construida esta aplicación de autentificación.

- **Componente:** Representación del código fuente, archivos, ejecutables o interfaces que conforman un sistema.
- **Interfaz:** Puerta de comunicación por el cual los componentes indican dependencia sobre otros componentes existentes en el sistema.
- **Relación:** Identifica la dependencia existente entre componentes.

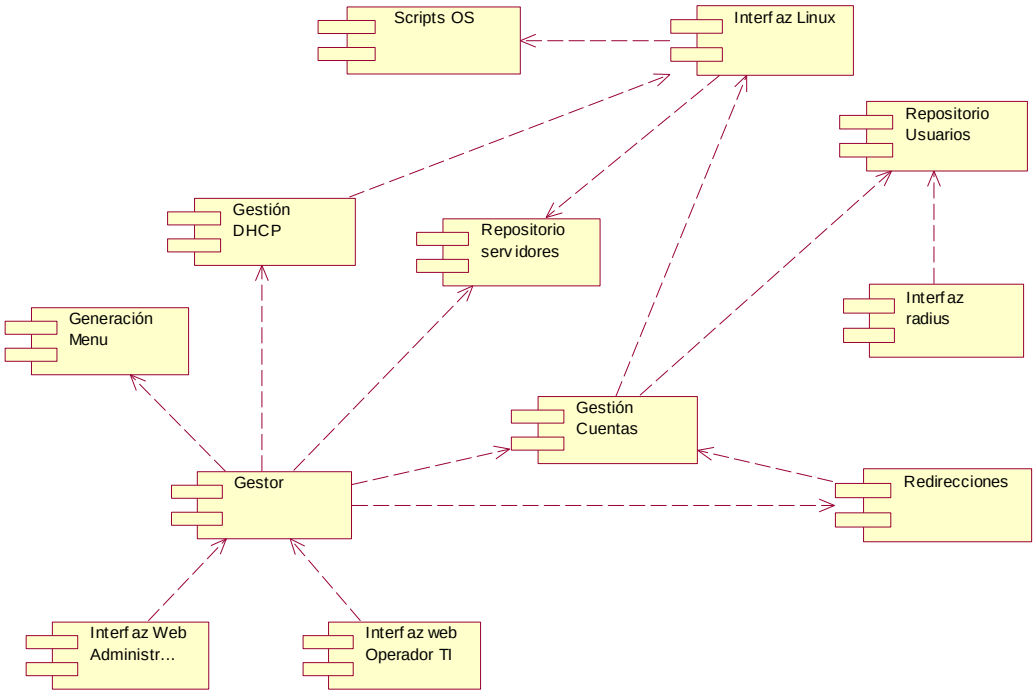


Figura 1-66

1.2.2.7 Diagramas de despliegue

Para representar cuales son los dispositivos físicos implicados en el sistema y sus relaciones, se hace uso de los diagramas de despliegue. Estos diagramas también permiten reflejar la disposición de las instancias de componentes de ejecución en instancias de nodos conectados por enlaces de comunicación.

"Los nodos se interconectan mediante soportes bidireccionales que pueden a su vez estereotiparse. Esta vista permite determinar las consecuencias de la distribución y la asignación de recursos. Las instancias de los nodos pueden contener instancias de ejecución, como instancias de componentes y objetos. El modelo puede mostrar dependencias entre las instancias y sus interfaces, y también modelar la migración de entidades entre nodos u otros contenedores."

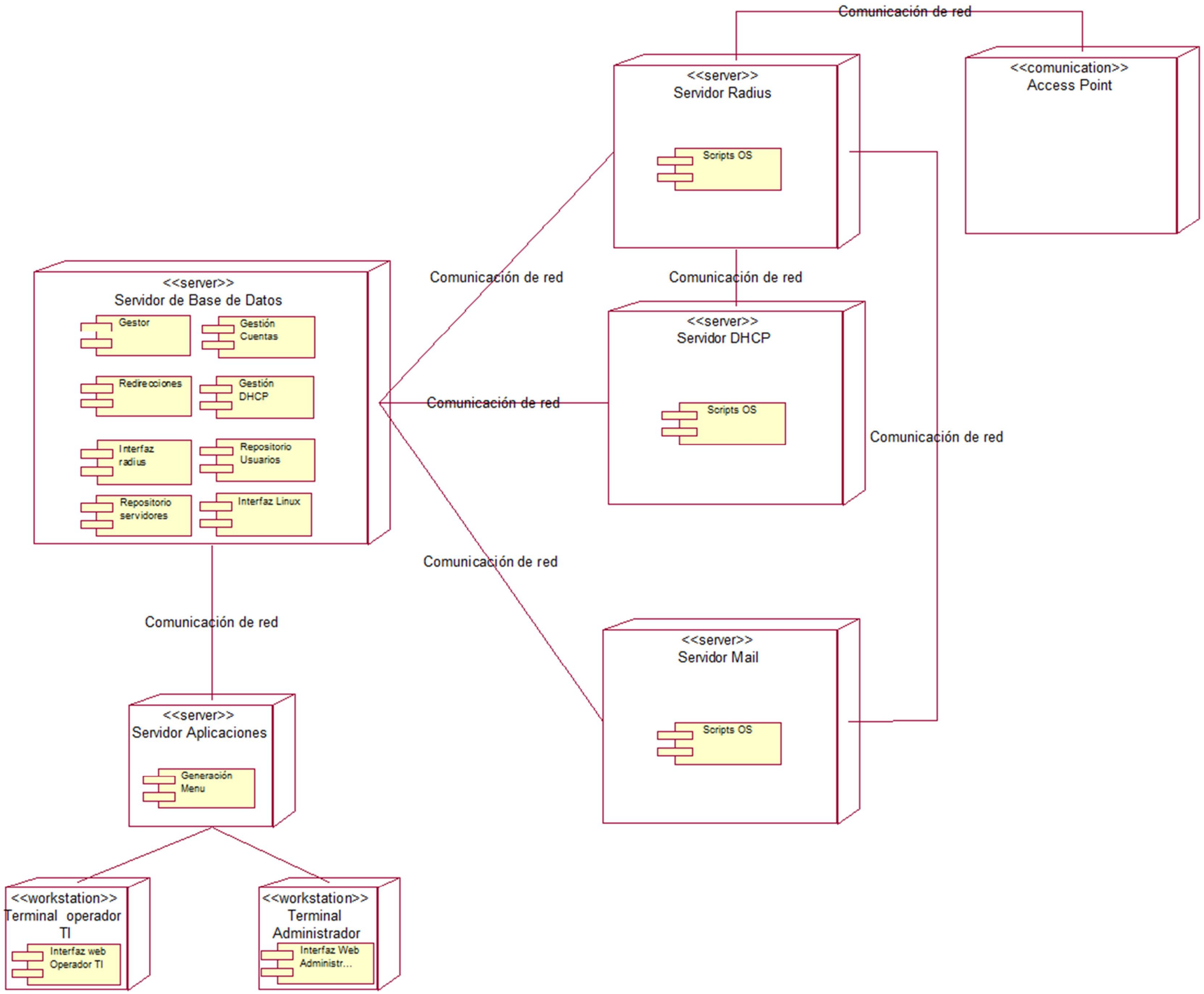

Comunicación de red
<<server>>
Servidor Radius
Scripts OS
<<comunication>>
Access Point
Comunicación de red
Comunicación de red
<<server>>
Servidor de Base de Datos
Gestor
Gestión Cuentas
Redirecciones
Gestión DHCP
Interfaz radius
Repositorio Usuarios
Repositorio servidores
Interfaz Linux
Comunicación de red
<<server>>
Servidor DHCP
Scripts OS
Comunicación de red
Comunicación de red
Comunicación de red
<<server>>
Servidor Mail
Scripts OS
Comunicación de red
<<server>>
Servidor Aplicaciones
Generación Menu
<<workstation>>
Terminal operador TI
Interfaz web Operador TI
<<workstation>>
Terminal Administrador
Interfaz Web Administr...

Figura 1-6

1.2.2.8 Diagrama de contexto

Cuando deseamos representar el flujo de datos hacia el sistema existen los diagramas de contexto, aunque como veremos en las siguientes secciones también es posible la especialización de cada proceso pero esto sucede con otro tipo de diagramas. Los diagramas de contexto representan al sistema como una burbuja que representa todo el sistema y desde el cual se origina y llegan los flujos de datos. En este flujo no detallamos cuales son los subprocesos del sistema, sino solamente cuales son las entradas y salidas hacia agentes externos que lo limitan, en conclusión el diagrama de contexto visualiza al sistema como una caja negra en el cual no conocemos su funcionamiento interno pero si los agentes externos.

Los elementos que conforman un diagrama de contexto son:

- **Entidades (Agentes externos):** Se representan mediante rectángulos y se comunican con el sistema utilizando flujos de datos.
- **Almacenes de datos:** Constituyen el repositorio donde los datos están almacenados y se representan mediante dos barras paralelas con el nombre del repositorio entre ellas.
- **Flujos de datos:** Se representan mediante una flecha dirigida que indica el sentido del flujo de los datos.

Los diagramas de contexto se constituyen como la primera iniciativa para iniciar el desarrollo de los diagramas de flujo de datos que veremos a continuación.

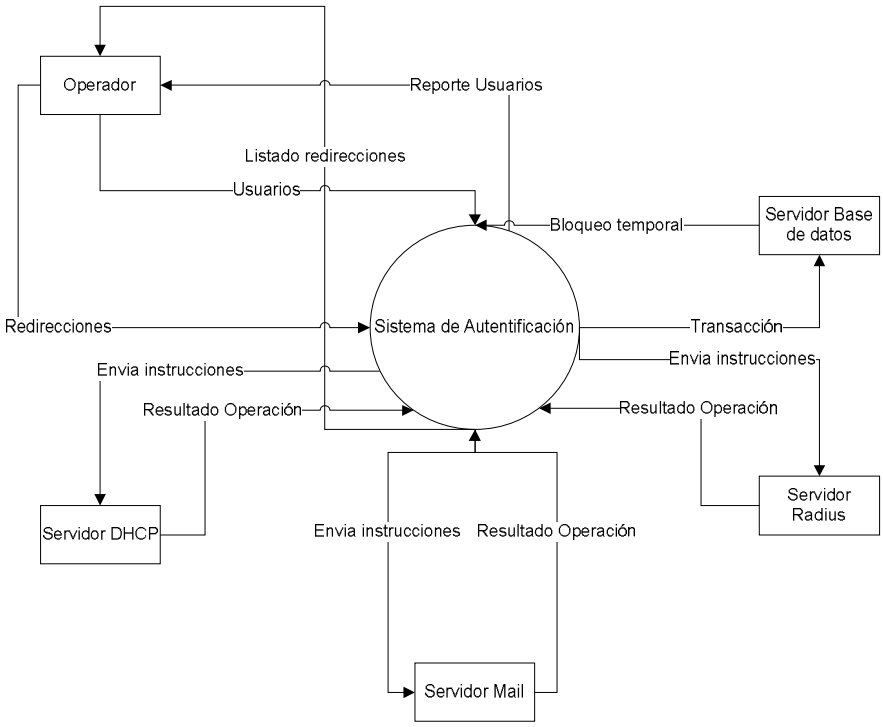


Figura 1-68

1.2.2.9 Diagramas de Flujo de Datos

Una manera de concebir como los datos fluyen a través de la organización lo constituyen los diagramas de flujo de datos, que permiten identificar los distintos procesos, entidades internas y externas que influyen en la transformación de la información.

Para comprender correctamente un diagrama de flujo de datos, es necesario comprender cuales son los componentes que los conforman. Por ejemplo para representar una entidad se hace uso una caja cuadrada la cual tiene en su interior el nombre de la entidad que puede ser una persona, grupo de personas o un sistema. Para representar el procesamiento de información se emplea un círculo con el nombre del proceso. Finalmente podemos ubicar al flujo de datos como una flecha y el almacén de datos con dos paralelas con el nombre entre ellas.

Diagrama de Flujo de datos Primer nivel

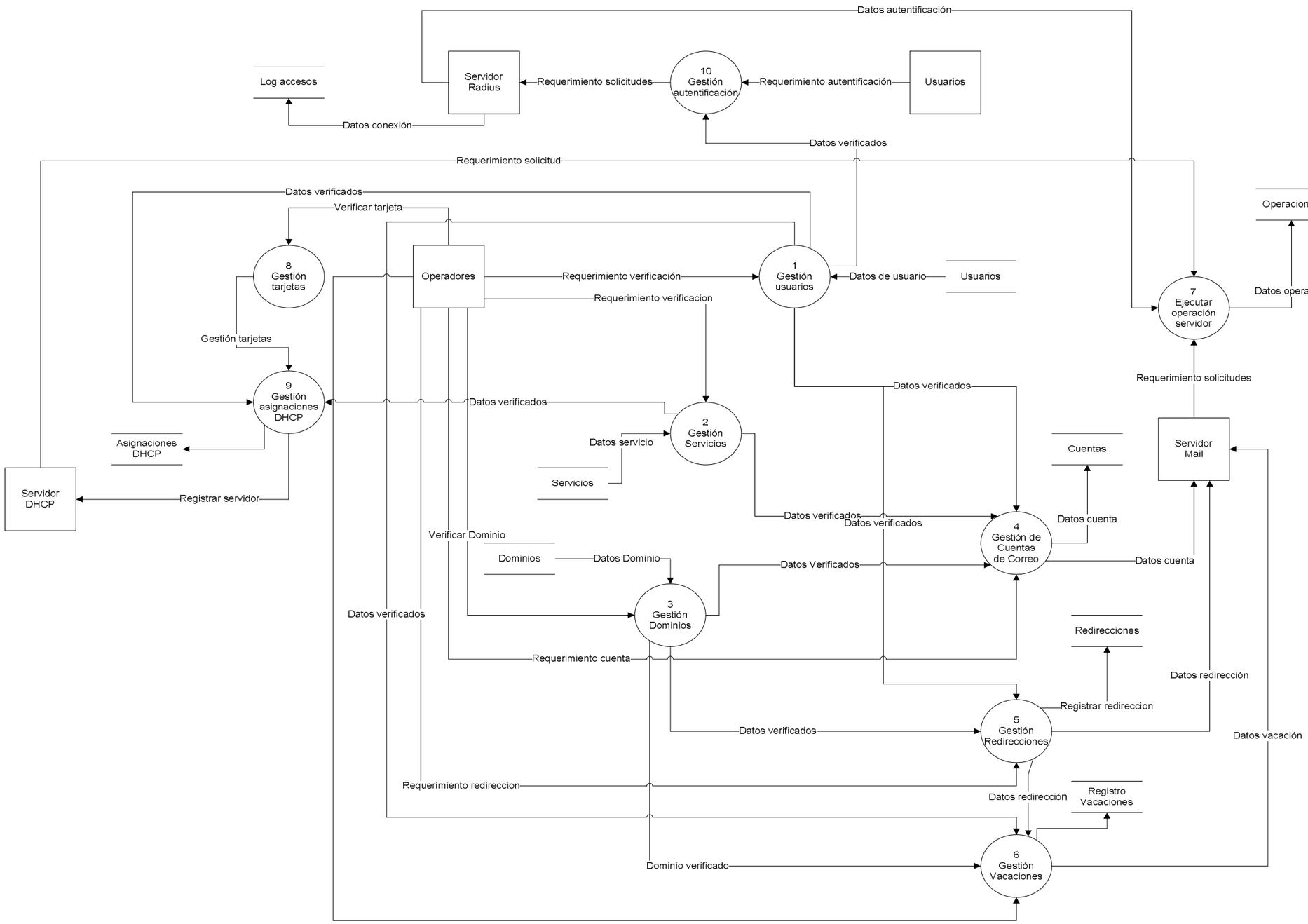


Figura 1-69

DFD Proceso 1 – Segundo Nivel

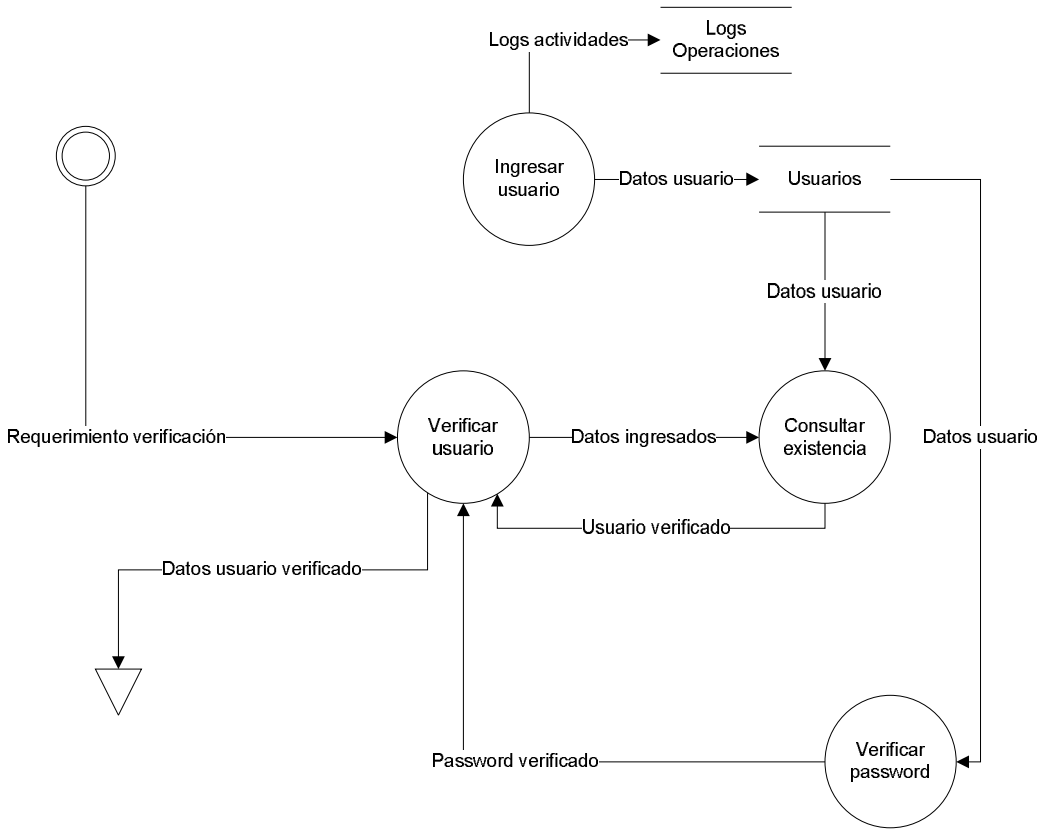


Figura 1-70

DFD Proceso 2 – Segundo Nivel

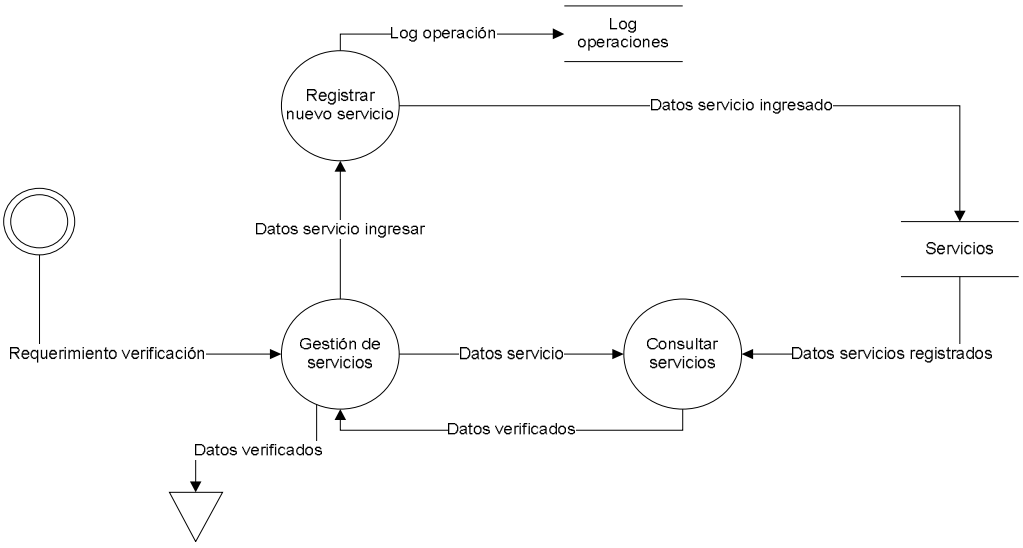


Figura 1-71

DFD Proceso 3 – Segundo Nivel

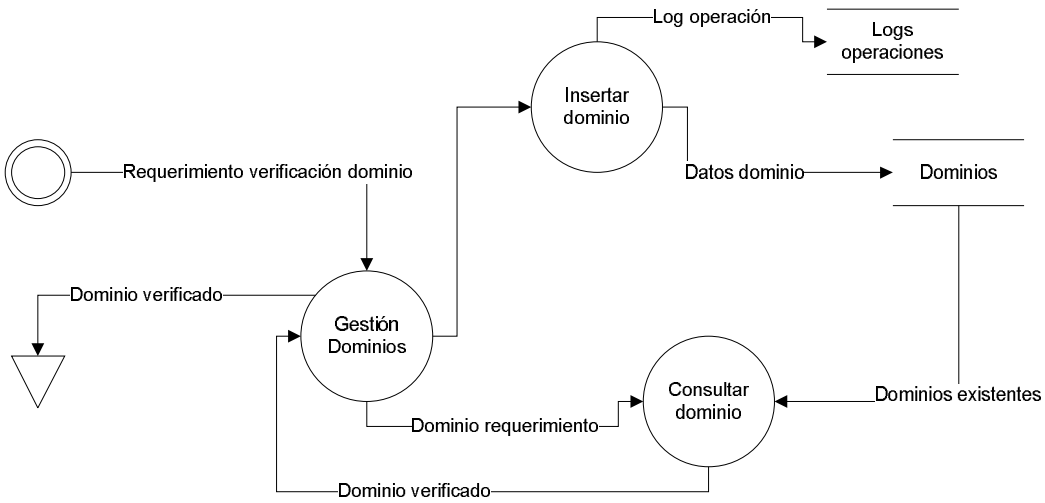


Figura 1-72

DFD Proceso 6 – Segundo Nivel

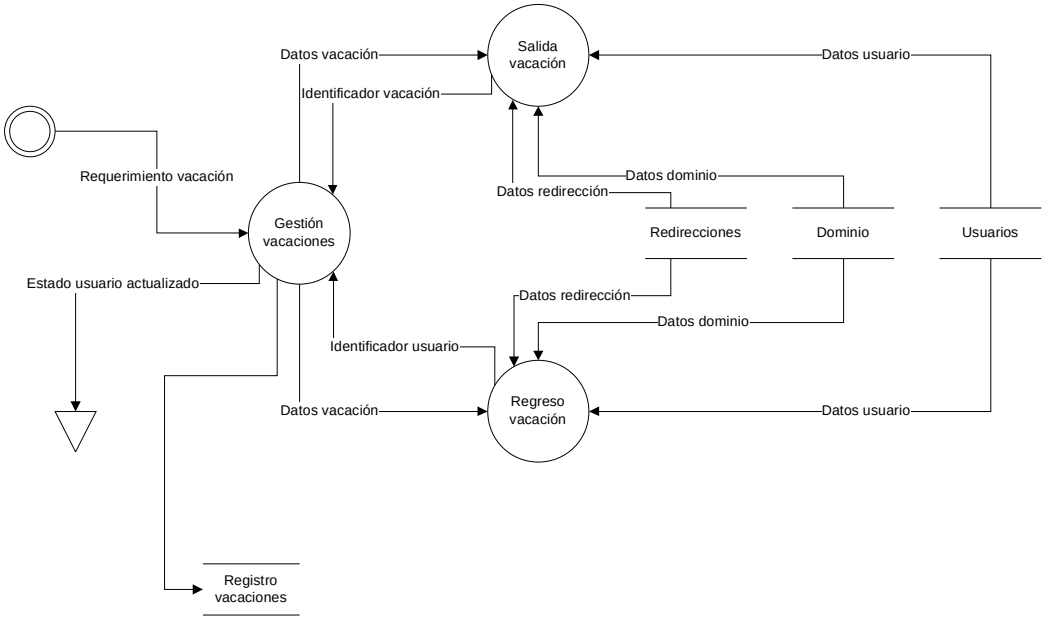


Figura 1-73

DFD Proceso 5 – Segundo Nivel

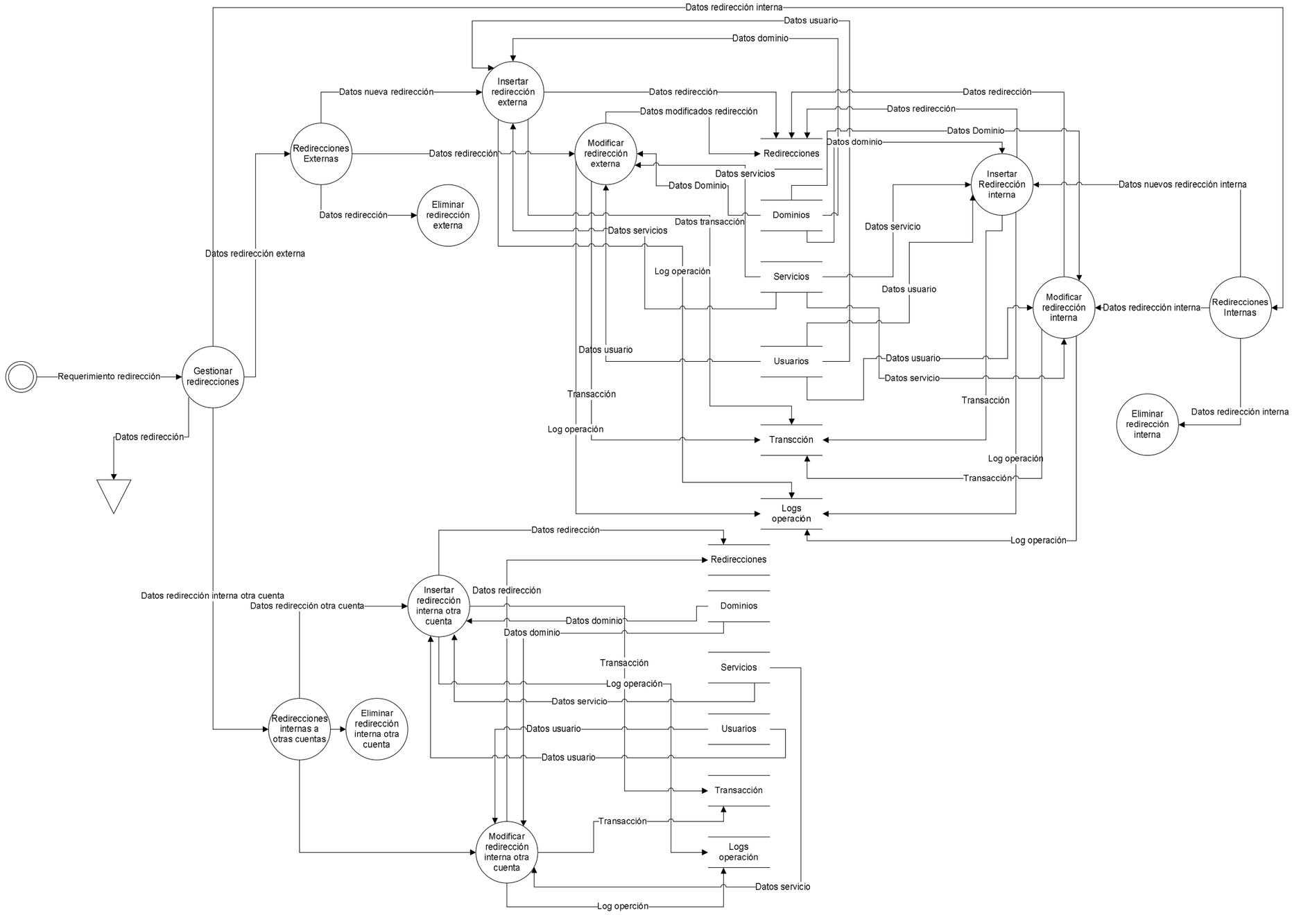


Figura 1-74

DFD Proceso 7 – Segundo Nivel

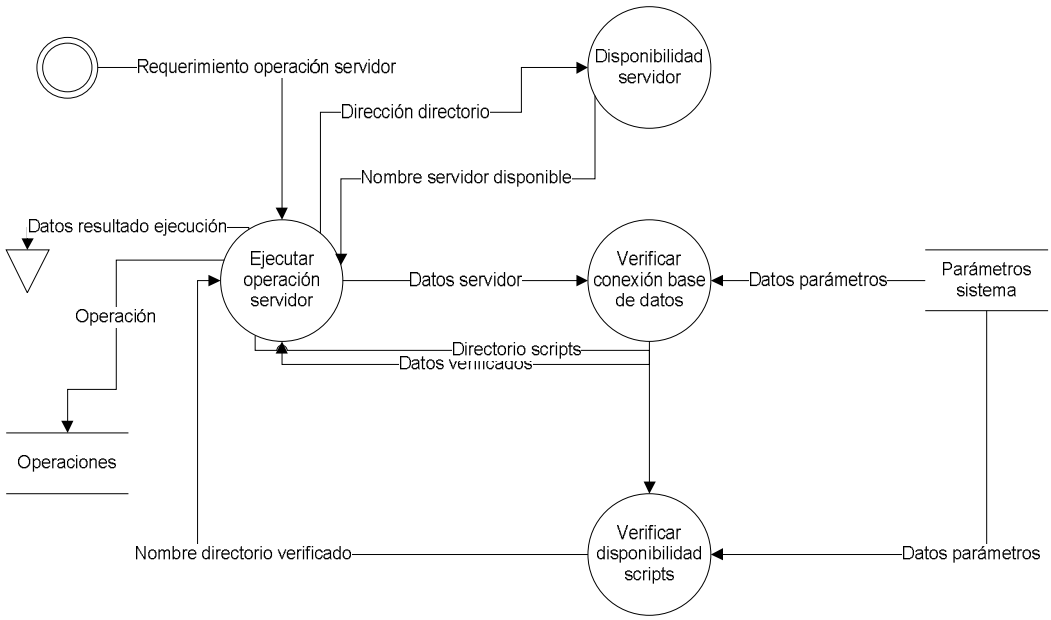


Figura 1-75

DFD Proceso 8 – Segundo Nivel

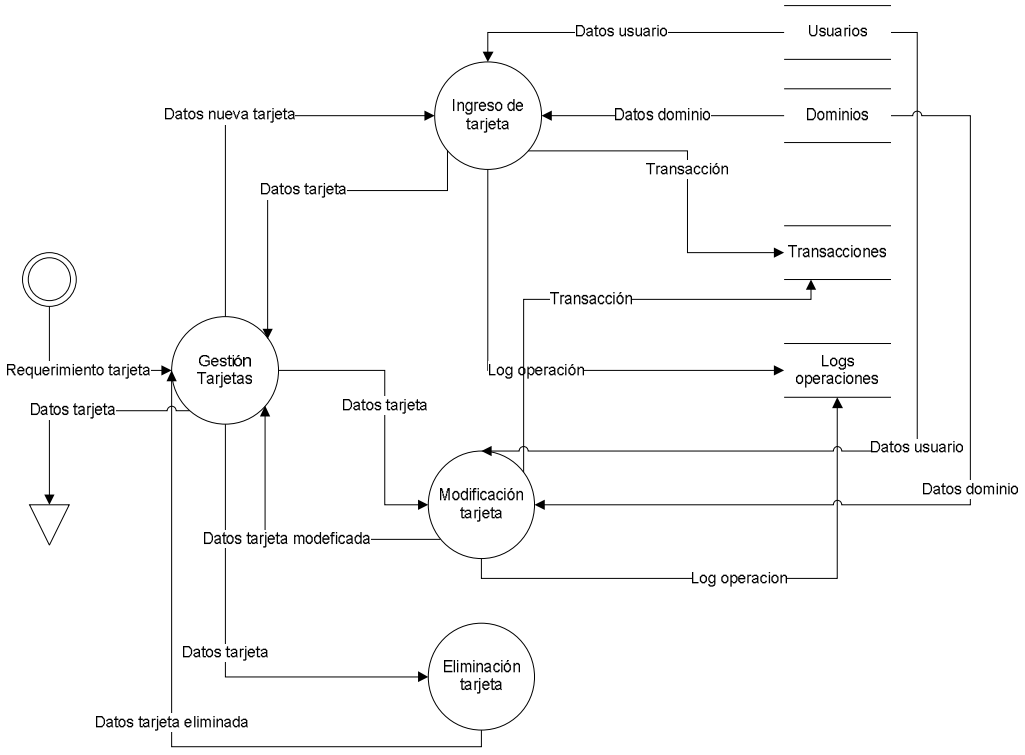


Figura 1-76

DFD Proceso 9 – Segundo Nivel

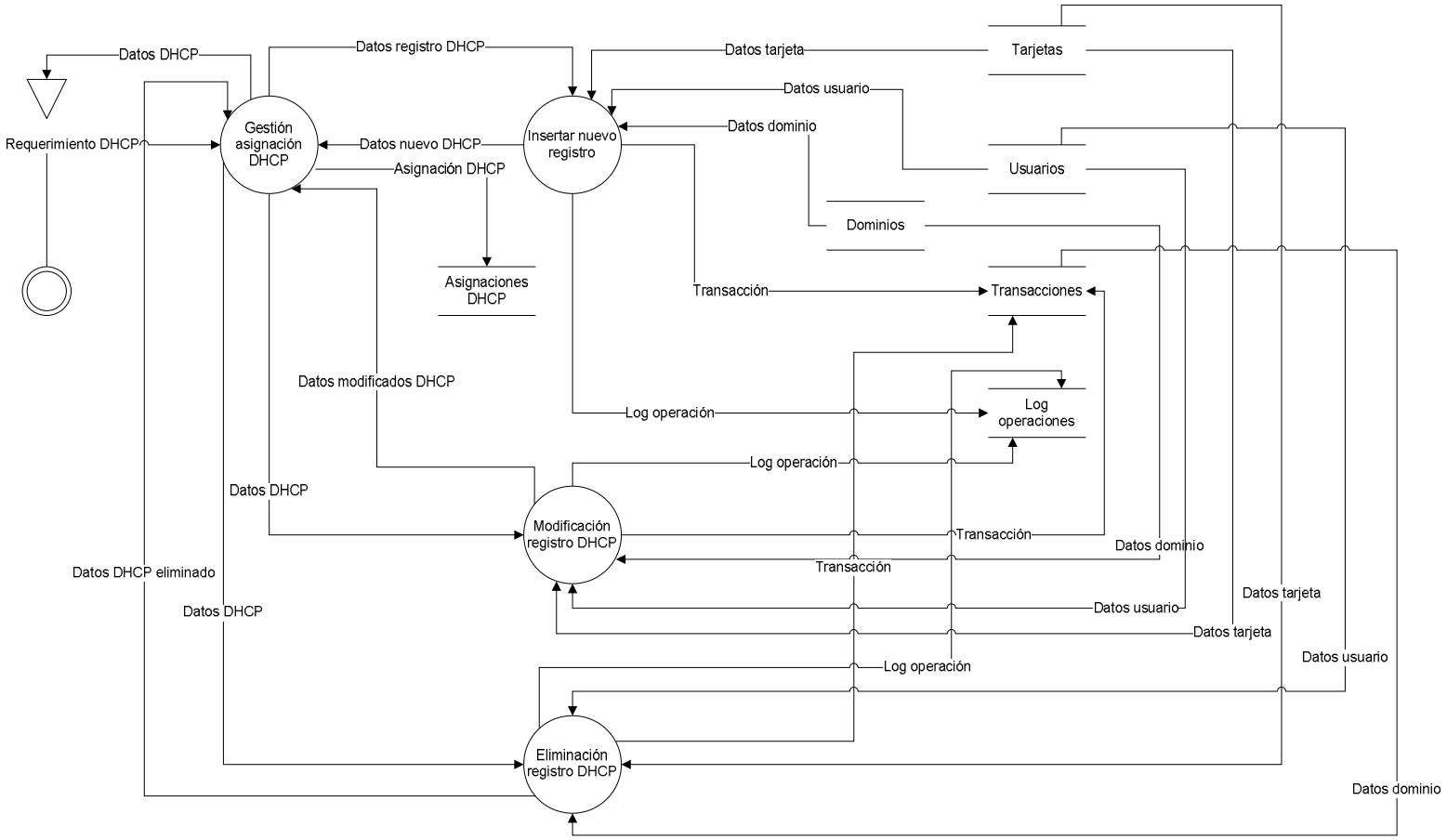


Figura 1-77

DFD Proceso 10 – Segundo Nivel

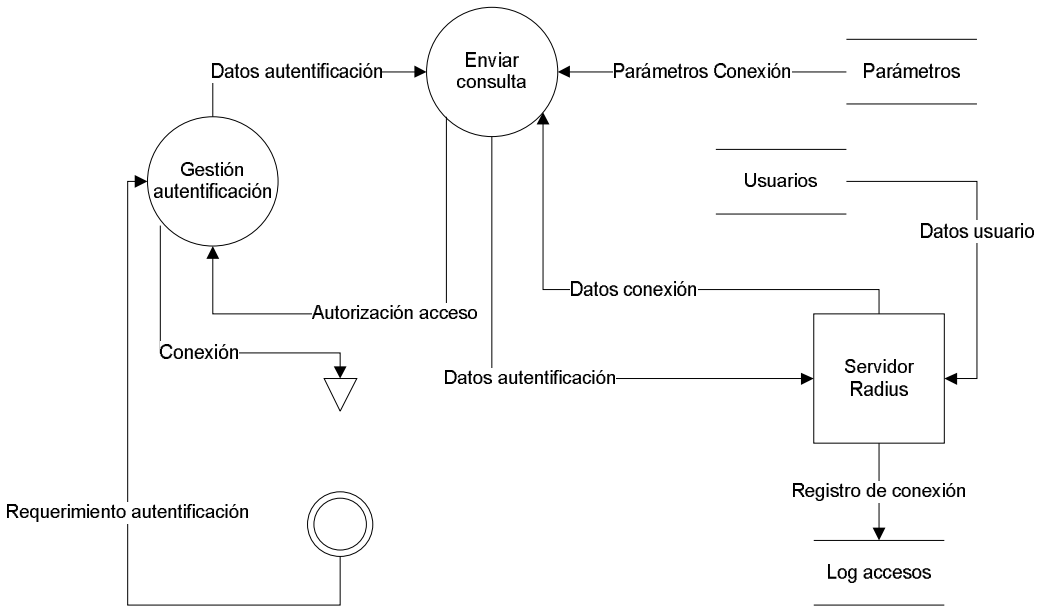


Figura 1-78

CAPÍTULO 2

Interfaz de comunicación entre la aplicación y los servidores

2.1 Descripción general Centos 5

Centos 5(Community Enterprise Operating System) es una variante del sistema operativo RHEL y nace a partir del código fuente liberado por redhat. Los desarrolladores de Centos utilizan el código fuente liberado por redhat y dan origen a Centos como un sistema operativo libre pero que no tiene soporte técnico por parte de redhat. Aunque centos usa binarios de redhat, también hace uso de otras fuentes como es el comando yum que surge de la distribución Fedora.

Las versiones del sistema operativo Centos se liberan cada 2 años y disponen de actualizaciones cada 6 meses, una versión liberada se mantiene durante un lapso de 7 años a través de actualizaciones de seguridad, después de lo cual no existirá actualizaciones o mejoras en el sistema operativo.

Es importante referirnos a la historia de este sistema operativo que nace a finales del año 2003 con una versión denominada CentOs 3 build4-rc0 pero no es sino hasta el año 2004 específicamente el 14 de mayo, que se lanza la versión denominada Centos 2 basada en la versión 2.1 de RHEL. Para el año 2007 se lanza la versión 5 nuevamente tomando como referencia la versión 5 de RHEL.

Este sistema operativo es excelente y muy usado en aplicaciones empresariales como soporte de servicios web, bases de datos, servidores de aplicaciones, etc. De acuerdo a una encuesta publicada por la *Web Technology Surveys* Centos se posiciona con el 30% del mercado siendo usado principalmente para publicación de servicios web, dejando de lado a distribuciones como debían, Ubuntu, Fedora entre otras, y es justamente este sistema operativo el que se emplea en el desarrollo de esta tesis.

2.2 Shell Linux

Para comprender el funcionamiento de la aplicación citada en este documento, es necesario comprender en primera instancia uno de los pilares principales del software, lo constituye la programación de scripts Shell pertenecientes a sistemas operativos basados en Unix.

Para ejecutar scripts Shell es necesario que el sistema operativo cuente con un programa que los pueda interpretar, este programa se llama *"Command Shell"* que también puede ejecutar comandos digitados manualmente. Es importante comprender que el *Command Shell* no es para nada un sistema operativo sino que se constituye como una interfaz con el sistema operativo utilizado para la ejecución de comandos. Es importante también que mencionemos que el *command Shell* es la interfaz de comunicación entre nosotros los operadores y el *kernel* del sistema operativo.

El *Command Shell* original de Unix es el Bourne Shell en honor Steve Bourne, aunque existen diferentes tipos de Command Shell que regularmente lo disponen la mayoría de distribuciones Linux. Entre los *command Shell* más conocidos encontramos:

2.2.1 Shell Bourne (bsh)

Es el terminal más conocido debido a que nació con el sistema operativo Unix aunque en la actualidad este terminal no es muy usado debido principalmente a que el Shell *bash (Bourne Again Shell)* realiza todas sus funciones y adicionalmente otras.

2.2.2 Bourne Again Shell (bash)

Es el Shell más utilizado en las distribuciones Linux en el caso de Red Hat esta es su terminal por defecto. Surge principalmente para ser usado por el proyecto GNU y es por ello que reside en la mayoría de distribuciones Linux. Incorpora características útiles de KSH y CSH y otras propias como la edición de comandos, tamaño ilimitado de historial de comandos, etc.

2.2.3 Korn Shell (ksh)

Es muy utilizada en la distribución HP-UX y en BSD (Berkley Software Distribution) combina lo mejor de lo terminales *csh y bsh* posibilitando la disponibilidad de características adicionales de programación.

2.2.4 C-Shell

Surge del sistema operativo *BSD* que habíamos mencionado anteriormente e incluye características de programación orientadas mayormente al lenguaje C.

Si hemos usado anteriormente distribuciones Linux seguramente nos hemos dado cuenta de que este sistema operativo es sensible a mayúsculas o minúsculas así por ejemplo no es igual escribir el comando EXIT que el comando exit, para el primer caso obtendremos un mensaje de error mientras que en el segundo caso el comando se encuentra bien escrito y no tendremos mensaje inesperados.

Entre las características que debemos tener en consideración antes de escribir un comando en el terminal de un distribución Linux constan por ejemplo: El carácter "/" en el caso de un sistema Linux representa una separación de directorio mientras que en un sistema operativo Windows representa el carácter necesario para la separación de argumentos de un comando a ejecutar. En un sistema Linux los nombres de archivos también son sensibles a mayúsculas o minúsculas y no poseen una extensión como sucede en Windows.

Al igual que en el sistema operativo Windows, Linux tiene caracteres especiales que permite la representación de uno o varios caracteres, redirección de la salida de un comando, separación de comandos, etc. Enumeraremos los más comunes:

- \ => Carácter que permite las inserciones de caracteres especiales como espacios, etc.
    - /root/Desktop/Carpeta\ Prueba
- . => El punto sirve para representar el directorio actual, aunque cuando es usado al inicio de una cadena también sirve para ocultar ficheros.
    - cp /root/Desktop/archivo . (Copia el archivo "archivo" al directorio actual).
    - .nombre_archivo (El punto inicial permite ocultar el archivo).
- .. => Dos puntos seguidos nos sirve para indicar que estamos haciendo referencia al directorio padre del actual.
    - cd .. (Permite cambiarnos al directorio padre).
- * => El asterisco al igual que en sistemas operativos Windows permite la representación de 0 o más caracteres.

    - locate arch* (Encuentra todos los archivos que comiencen por “arch” y terminen en cualquier carácter).
- ? => Así como el caso anterior este comando es usado en Windows y también sirve para representar un solo carácter dentro del nombre de un archivo.
    - locate arch?vo (Encuentra todos los caracteres que comiencen con los caracteres “arch” seguido de cualquier carácter y terminado en “vo”).
- | => Denominado *pipe* este carácter es usado para redireccionar la salida de un comando como entrada de otro comando.
    - ls –la | grep nuevo (Lista todos los documentos que contenga la cadena “nuevo”).
- > => El signo de “mayor que” es utilizado para redireccionar la salida de un comando a un archivo.
    - echo “hola” > archivo1 (Inserta el contenido “hola” dentro del archivo “archivo1”)
- & => Cuando nos encontramos en una situación en la que necesitamos ejecutar un comando pero requerimos que el terminal no sea bloqueado, entonces este carácter es el que necesitamos ya que permitirá la ejecución de un comando en background.
    - dd if=/dev/sda1 of=/de v/sdb1 & (Ejecuta el comando dd que pasa byte por byte el contenido de sda1 a sdb1).

En el sistema operativo Linux podemos ejecutar comandos tales como “ls, cat, find, …” sin embargo esto es solamente posible cuando se ha instanciado la variable PATH. Cuando digo instanciado me refiero a que debe existir la variable de ambiente PATH que es la que contiene la ruta donde el terminal buscará por comandos (ls, cat, …), las rutas contenidas en la variable PATH generalmente son: /bin, /usr/bin, /usr/X11R6/bin entre otras.

Algunos comandos del sistema operativo requieren que se les pase parámetros que normalmente tienen el siguiente formato.

- comando [-argumento1] [-argumento2] [-argumento3] [file]
    - ls –la
    - df –h

No es objetivo de este documento la presentación detallada de todos los comandos como si se tratara de un libro de referencia, más aún se pretende presentar nociones básicas necesarias para comprender el resto del documento. En todo caso si nos surge la necesidad de información más detallada de algún comando en particular, existe para ello disponible comandos que visualizan información de ayuda de otros comandos.

Si deseamos obtener ayuda de un comando en particular se hace uso de las siguientes opciones:

- grep --help(Despliega ayuda embebida en el comando)
- man grep (Despliega ayuda de manuales de referencia)
- info grep (Despliega información rápida de un comando en particular)

Dependiendo de la opción que usemos se visualizara información limitada o información bastante extensa y detallada. La estructura de directorios utilizada en este sistema operativo es diferente a la usada en sistemas como Windows, hago referencia constantemente a Windows debido a que es el sistema operativo que con más frecuencia se usa en equipos de escritorio. Digo que la estructura es diferente porque en Linux no tememos unidades C:/ o D:/ etc., sin embargo si dispone de directorios como por ejemplo:

- /etc. => Archivos de configuración del sistema operativo.
- /bin => Comandos ejecutados en el terminal del sistema operativo
- /usr => Directorio que contiene los archivos que se pueden compartir, como documentación y librerías.
- /lib => Librerías para la mayoría de programas instalados.
- /var => Almacena archivos de longitud variable como logs, mails
- /sbin => Contiene los comandos esenciales para el sistema operativo.

Todos estos directorios son muy importantes porque básicamente constituyen el sistema operativo.

2.2.5 Ejecución de scripts Shell

Hemos llegado a un punto en el que el intérprete de comando *bash* demuestra su verdadera potencia, debido a que además de ejecutar comandos también puede ejecutar scripts que en su interior contienen comandos de sistema operativo y también estructuras de control (if, while, for, do while entre otras). Lo que probablemente nos estemos preguntando es: ¿Qué es un script?. Pues bien un script es un archivo de texto que en muchas ocasiones inicia con la línea: *#!/bin/bash* la misma que nos indica de qué tipo de terminal vamos a hacer uso, en este caso un terminal tipo bash. El resto de contenido del archivo de texto como ya mencionamos anteriormente son comandos e instrucciones de control.

Todos los Scrjpts Shell terminan con los caracteres: ".sh" y necesariamente deberán tener permisos de ejecución para poder ser ejecutados. Para asignar los permisos de ejecución, por lo regular se hace uso de: *chmod 755 script.sh*. Antes de visualizar el código de los diferentes scripts generados para la herramienta es necesario revisar las estructuras de control que se usan en los scripts, empecemos por las estructuras de control y luego pasamos a las estructuras repetitivas.

2.2.6 Estructuras de Control

Como ya se conoce, la primera estructura de control que nos viene a la mente es:

```
if [ condición ]
then
        acciones
fi
```

La estructura de control IF nos permite realizar comparaciones lógicas en donde "condición" puede adoptar los siguientes valores:

- -eq es igual
- -ne no es igual
- -lt menor que
- -le menor que o igual
- -gt mayor que
- -ge mayor que o igual

Ejemplo:

```
if [ $# -ge 2 ]
then
        echo “Mayor o igual que 2”
else
        echo “No es mayor o igual que 2”
fi
```

En este ejemplo hacemos uso de “$#” que nos permite determinar cuál ha sido el número de parámetros pasados a un script, como podemos observar en el ejemplo hacemos una comparación simple en donde consultamos si el número de parámetros es mayor o igual a dos en cuyo caso se envía el mensaje “Mayor o igual que 2” caso contrario visualizará “No es mayor o igual que 2”. Al igual que en cualquier lenguaje de programación, la sentencia IF puede estar anidada en otras sentencias IF.

Otra sentencia de control usada muy a menudo pero no muy conocida es la instrucción case que podemos ver en el ejemplo inferior.

```
seleccion=”B”;
case $seleccion in
 a|A)
        instruccion 1;
        instruccion 2;
 b|B)
        instruccion3;
        instruccion4;
esac;
```

En este ejemplo podemos observar que la variable “$seleccion” es comparada con “a|A”, “b|B” si en caso resulta la condición afirmativa para B entonces ejecutara las instrucciones 3 y 4. La condición puede ser tanto uno como varios valores para lo cual es usa la siguiente notación:

- A) => Se compara que tenga la letra “A.”

- a|A => Se valida que tenga la letra “a” o “A”.
- [a-z] => Se compara que este en el rango de la letra “a” hasta la letra “z”.

Ahora pasaremos a revisar las estructuras de repetición tanto desde contadores o condiciones lógicas como las iteraciones generadas por la lectura de un archivo.

2.2.7 Estructuras Repetitivas

Empecemos por la estructura en la que el bucle se repite siempre y cuando existan palabras en la instrucción.

Ejemplo:

```
for variable in palabra1 palabra2
do
        acciones
done
```

En este caso la estructura repetitiva se ejecutara dos veces, una para la palabra1 y otra para la palabra2. De acuerdo a la documentación se menciona que podríamos prescindir de las palabras “do” y “done” las mismas que pueden ser remplazadas por llaves. Para ingresar a un bucle en el que solamente se escape a través de una comparación lógica usaremos la siguiente sintaxis:

```
while ($VARIABLE=valor)
 do
   comandos
 done
```

Este bucle continuará ejecutándose siempre y cuando la condición sea verdadera, es obvio que para poder “romper” el bucle es necesario alterar “$VARIABLE” de modo que la condición sea falsa.

La última estructura de repetición que es muy conocida y por ende utilizado es aquella en la que el bucle se ejecuta a partir de la lectura de un archivo, esta

estructura se mantendrá ejecutando siempre y cuando haya contenido en el archivo, caso contrario termina su operación.

Ejemplo:

```
while read ARCHIVO
 do
   echo $ARCHIVO;
 done /home/prueba.txt
```

Este ejemplo leerá el archivo /home/prueba.txt y visualizará su contenido por medio del comando *echo* siempre que exista contenido en el archivo. Por último nos queda explicar brevemente cómo funciona el paso de parámetros para la ejecución de scripts Shell.

2.2.8 Paso parámetros a Script Shell

El uso de parámetros constituye el método para enviar valores al interior de un script Shell. Es importante mencionar que existe un conjunto pequeño de parámetros que son especiales y por tanto tienen una representación distinta. Para enviar parámetros al interior de un script es necesario establecerlos a continuación del nombre del script separados por un espacio.

Ejemplo:

```
sh nombre_script.sh parámetro1 parámetro2 parámetro3 parametro4
```

Para hacer uso de un parámetro al interior de un script se utiliza el símbolo de dólar ($) seguido por un número secuencial que empieza en 1.

Ejemplo:

```
$1   => Parámetro 1
$2   => Parámetro 2
$3   => Parámetro 3
$4   => Parámetro 4
```

Cuando el número de parámetros supera el al noveno entonces los parámetros se tienen que recuperar u obtener su valor a través de una representación que incluye a más de del símbolo de dólar, llaves que encierran los dígitos secuenciales.

Ejemplo

${10} => Parámetro 10
${11} => Parámetro 11

Parámetros que son especiales y que retornan valores como el *PID (Process ID)* también inician con el símbolo de dólar pero el siguiente carácter difiere y también es un carácter especial a excepción del 0. Veamos un listado y cuál es su función:

Tabla 1-2: Caracteres especiales usados al interior de un script

Símbolo	Descripción uso
$0	Representa la llamada que se realizó al script Ejemplo: sh script.sh
$#	Es el número de parámetros que se pasa.
$@ y $*	Contiene el valor de todos los parámetros pasado al script.
$$	El PID que tiene nuestro proceso.
$?	Es en valor de salida del último comando ejecutado. Si todo sale bien entonces retorna 0 caso contrario puede retornar 1.
$_	Nos da el último argumento que se ha empleado.

Es posible el uso de variables dentro de un script de tipo texto que tienen que ser definidos de la siguiente manera: ${variable} si variable no es definida entonces podemos hacer uso de ${variable=palabra} en donde la variable "variable" tiene un valor por defecto que viene dado por el valor que tenga palabra.

2.2.8 Permisos Sistema Operativo sobre los Scripts

Es común encontrarse en la ejecución de scripts Shell, con el problema que tiene como mensaje “permiso denegado”. Este problema se presenta comúnmente cuando el terminal no halla los permisos de ejecución sobre un script Shell.

Para solucionar este problema se hace necesario ejecutar la siguiente instrucción:

```
chmod u+x script.sh
```

Para entender que hace el comando ejecutado anteriormente es necesario explicar brevemente cómo funcionan los permisos gestionados por el sistema operativo Linux.

2.3 Permisos Sistema Operativo.

El sistema operativo Linux fundamenta su sistema de permisos mediante el empleo de usuario y grupos a quienes se les proporciona acceso sobre ficheros y carpetas. Una de las características de Linux que permiten justamente mitigar el problema de los virus es disponer de esta estructura de permisos por usuarios y por grupos, para que un virus pueda infectar un sistema operativo necesita permisos sobre lo archivos pero si no los tiene no podrá realizarlo, prácticamente se corta la “reproducción” del virus.

2.3.1 Permisos del que dispone Linux

- r => Read (Lectura): Si el permiso de lectura está asignado a un directorio quiere decir que se podrá listar los contenidos del mismo, pero si este permiso está asignado a un archivo entonces eso quiere decir que se podrá visualizar el contenido del archivo.
- w => Write(Escritura): Si este permiso está activo para un directorio entonces se podrá crear y eliminar archivos, sin embargo si está asignado a un archivo entonces este permiso significa que se podrá modificar su contenido.
- x => Execute(Ejecución): Si está activo sobre un archivo permitirá la ejecución del mismo, sin embargo si está activo sobre un directorio quiere decir que se podrá realizar actividades adicionales a las de listar, crear y

eliminar contenido como por ejemplo podría ingresar al directorio, aunque si no tiene permisos de lectura no podrá visualizar el contenido del directorio.

Existen diferentes formas de ver los permisos de los archivos y directorios, sin embargo haremos referencia a dos formas:

- ls –l
- ll

Ambos comandos listarán archivos y directorios con sus respectivos permisos.

Ejemplo:

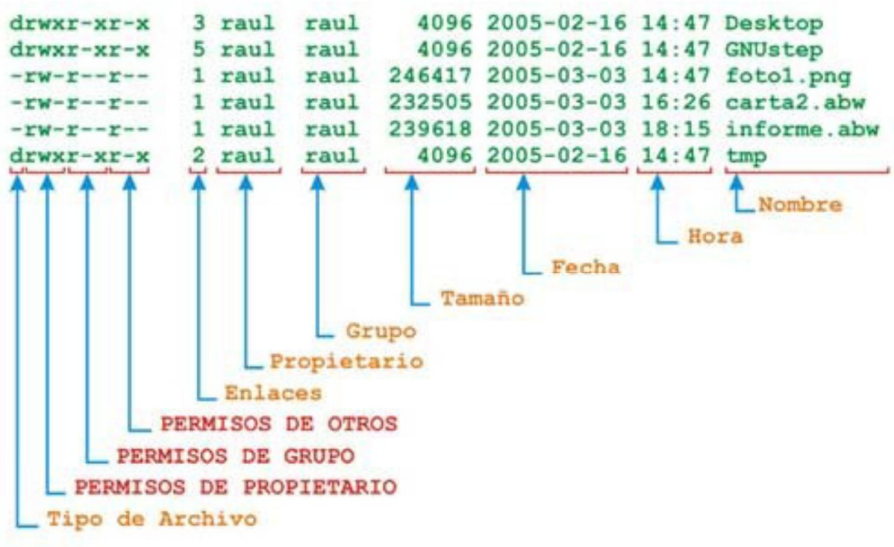


Figura 2-1: Descripción permisos de archivos en Linux. Obtenido de: http://mmujica.files.wordpress.com/2007/05/permisos-linux.pdf.

Para cambiar los permisos de los archivos y directorios es necesario hacer uso de comandos del sistema operativo que lo permitan, entre los comandos utilizados constan:

- chmod permisos archivo opciones
- chown nuevo_usuario_propietario archivo opcion**es**
- chgrp nuevo_grupo archivo opciones

Para cambiar permisos de los archivos se puede hacer uso de números como también de caracteres, observemos.

Tabla 2-2: Significado de símbolos en la sección de permisos

Caracter	Descripción
U	Usuario propietario del archivo o directorio.
G	Grupo propietario del archivo o directorio.
O	Todos los demás usuarios que no están en el grupo y no es el propietario.
A	Todo el mundo en donde constan el propietario, el grupo y los otros.
R	Permiso de lectura
W	Permiso de escritura
X	Permiso de ejecución
+	Adiciona permisos
-	Elimina permisos
=	Asigna permisos

Para comprender mejor cómo funcionan los permisos es necesario que describamos un ejemplo como: chmod a+rwx archivo1 => Permite asignar permisos de lectura, escritura y ejecución a todo el mundo sobre el archivo 1, los permisos en este comando pueden también ser representados por medio de números: chmod 777 archivo1.

La siguiente tabla ilustra la relación que existe entre los números y los permisos que podemos asignar.

Tabla 2-3: Nomenclatura de permisos

#	Permiso	Nomenclatura
0	Significa que no se han asignado permisos.	---
1	Significa permiso de ejecución sobre archivos y directorios.	--x
2	Significa permiso de escritura.	-w-

3	Significa permiso de escritura y ejecución.	-wx
4	Significa permisos de sólo lectura.	r--
5	Permisos de lectura y ejecución.	r-x
6	Permisos de lectura y escritura.	rw-
7	Todos los permisos: Lectura, escritura y ejecución.	rwx

No es propósito de esta tesis el listar todo el código desarrollado, sin embargo es necesario citar al menos uno de los scripts para evidenciar que la teoría descrita se cumple.

2.3.2 Script ABC_MAIL_MX.sh

```
#!/bin/bash
#########################################################################
#####
########Script de actualización de archivo de servidor MAIL##############
#########################################################################
#####
#####definición de variables para identificar bloques de eliminación#####
usuario=$1
password=$2
opcion=$3
operador=$4
srv_servicio=$5
dominio=$6
dir_ipv4=""
dir_mac=""
nom_host=""
pasa=0
fecha=""
usu_oracle="radius_p"
pass_oracle="radius_p"
ru_bin_ora="/u01/app/oracle/product/11.2.0/client_1/bin"
export ORACLE_HOME="/u01/app/oracle/product/11.2.0/client_1"
fecha=`date +%m/%d/%Y\ %H:%M:%S`
#################################################################
#########Obtencion de datos a registrar en la base############
#################################################################
while read lst
do
      if [ "$pasa" -lt "1" ]
      then
            extraccion=`echo $lst | grep IPADDR`
            dir_ipv4=`echo $extraccion | awk -F = '{print $2}'`
            if [ "$dir_ipv4" != "" ]
            then
                  pasa=`expr $pasa + 1`
```

```
            fi
        fi
done < /etc/sysconfig/network-scripts/ifcfg-eth0
pasa=0
#Obtencion de la direccion mac de la tarjeta del equipo
while read lst
do
        if [ "$pasa" -lt "1" ]
        then
            extraccion=`echo $lst | grep HWADDR`
dir_mac=`echo $extraccion | awk -F = '{print $2}'`
            if [ "$dir_mac" != "" ]
            then
                pasa=`expr $pasa + 1`
            fi
        fi
done < /etc/sysconfig/network-scripts/ifcfg-eth0
pasa=0
#Obtencion del nombre de la maquina
while read lst
do
        if [ "$pasa" -lt "1" ]
        then
            extraccion=`echo $lst | grep HOSTNAME`
            nom_host=`echo $extraccion | awk -F = '{print $2}'`
            if [ "$nom_host" != "" ]
            then
                pasa=`expr $pasa + 1`
            fi
        fi
done < /etc/sysconfig/network

if [ "$opcion" = "I" ]
then
        useradd $usuario
        echo "$password" | passwd --stdin $usuario
fi

if [ "$opcion" = "E" ]
then
        #Matamos posibles sessiones SSH iniciadas por el usuario
        ps -ef | grep ssh > /tmp/sesi_ssh.txt
        while read sshs
        do
            user=`echo $sshs | awk '{print $1}'`
            pidu=`echo $sshs | awk '{print $2}'`
            if [ "$usuario" = "$user" ]
            then
                kill -9 $pidu
            fi
```

```
        done < /tmp/sesi_ssh.txt
        userdel -r $usuario
fi

if [ "$opcion" = "M" ]
then
        echo "$password" | passwd --stdin $usuario
fi
sleep 3
#########################################################################
###Envio de respuesta exitosa a la base de datos#################################
. oraenv
/u01/app/oracle/product/11.2.0/client_1/bin/sqlplus
$usu_oracle/$pass_oracle@udatesis << EOF > ./output_response.log
        execute
radius_p.pq_radius_mantenimientos.pr_upd_usu_ser_dom_res('$dir_ipv4','$nom_host','$dir_mac','$srv_servicio','$usuario','$operador','COMPLETADO','$fecha','$dominio');
EOF
```

2.4 Cliente *Oracle* Linux 11g

En la sección de scripts Shell hemos hablado temas de permisos, comandos insertados en archivos de texto, etc. pero no comentamos como podemos conectarnos a una base de datos desde un script Shell. Pues bien para lograr ese objetivo es necesario que se haga uso del cliente de Oracle, en nuestro caso el 11g.

Es de suma importancia que se defina claramente que es el cliente de Oracle 11g que estamos usando en esta tesis. El cliente de Oracle es un conjunto de archivos ejecutables que permiten el establecimiento de conexión mediante un protocolo denominado *Oracle Net* hacia la base de datos Oracle. Este cliente en su interior dispone de algunos archivos que es necesario que lo entendamos.

2.4.1 ¿Qué es un listener?

Un Listener es un archivo que se encuentra ubicado en el servidor donde fue instalada la base de datos. Este archivo es el responsable de contener la información necesaria que permitirá establecer y mantener la conexión entre las aplicaciones Cliente y Servidor de la Base de Datos.

Este fichero se encuentra ubicado en la ruta que nos indica la variable de ambiente ORACLE_HOME, más la ruta: */network/admin*. Por lo general en sistemas Linux esta variable contiene: */u01/app/oracle/product/11.2.0/db_home1/* que es el lugar donde se encuentra todos los binarios de la base de datos.

Para nuestro caso tenemos un archivo listener que contiene la siguiente información:

```
listener_extproc=
 (address=(protocol=ipc)(key=extproc))
sid_list_listener_extproc=
 (sid_list=
  (sid_desc=
   (sid_name=extproc)
   (oracle_home=/u01/app/oracle/product/11.2.0/dbhome_1)
   (program=extproc)
  )
)
```

Como podemos observar existen palabras claves que representan los parámetros que deben estar configurados para que podamos acceder a la base de datos, veamos una descripción de lo que representan cada uno de ellos:

- LISTENER: Direcciones del protocolo en las que se aceptarán solicitudes de conexión.
- DESCRIPTION: Contiene protocolos a ser usados por el listener.
- ADDRESS: Especifica protocolos individuales.
- HOST: Dirección IP del servidor de la base de datos.
- PORT: Puerto de escucha de la base de datos, por defecto el 1521.
- SID_LIST_LISTENER: Información acerca del servicio de base de datos, incluyendo el nombre de la base de datos global, ubicación del directorio raíz y el identificador del sistema.
- SID_LIST: Lista de descripciones.
- SID_DESC: Información para una instancia específica.
- GLOBAL_DB_NAME: Nombre de la base de datos.
- ORACLE_HOME: Directorio de instalación de Oracle.
- SID_NAME: SID (identificación) de la Base de Datos.

Tomado de aesinformatica.com

2.4.2 ¿Qué son los archivos TNSNAMES?

"Los archivos TNSNAMES puede existir tanto en el servidor como en los clientes que se conectan al servidor". Contiene información necesaria para que los clientes se puedan conectar a la base de datos, por lo general contienen los nombres de servicio.

El directorio donde se encuentra ubicado en la presente tesis es: */u01/app/oracle/product/11.2.0/dbhome_1/network/admin* y contiene la información listada a continuación, sin embargo este archivo en el servidor de la base de datos puede diferir en algo su contenido a los archivos que se encuentran en los clientes en otros equipos.

Archivo tnsnames.ora en servidor de base de datos

```
extproc_connection_data=
  (description=
  (address=(protocol=ipc)(key=extproc))
  (connect_data=(sid=extproc)))
```

Archivo tnsnames.ora en clientes

```
UDATESIS=
  (DESCRIPTION =
    (ADDRESS = (PROTOCOL = TCP)(HOST = 192.168.1.19)(PORT = 1521))
    (CONNECT_DATA =
      (SERVER = DEDICATED)
      (SERVICE_NAME = udatesis)
    )
  )
```

Al igual que en el caso anterior es necesario describir sus parámetros:

- DESCRIPTION: Es obligatorio y describe las características de conectividad de la base de datos
- ADDRESS: Contiene el protocolo de comunicación (IPC) en red.
- CONNECT_DATA: Conexión de datos.
- SID: Identificación única de la base de datos.

2.4.3 ¿Para qué sirve el archivo SQL Net?

Archivo de configuración, en el cliente y el servidor, con detalles básicos para la identificación de las conexiones de usuarios a la base de datos.

Este archivo para nuestro servidor de base de datos se encuentra ubicado en la ruta: */u01/app/oracle/product/11.2.0/dbhome_1/network/admin* y contiene:

NAMES.DIRECTORY_PATH= (TNSNAMES, EZCONNECT)
ADR_BASE = /u01/app/oracle

Para sistemas operativos Windows varia un poco su contenido básicamente porque ya no encontramos la línea ADR_BASE sino la línea SQLNET.AUTHENTICATION_SERVICES= (NTS).

De acuerdo a lo citado en la página aesinformatica.com tenemos la siguiente descripción de este archivo.

NTS: Autentificación de Windows sin contraseña.
NONE: Cualquier usuario valido con contraseña accede.
ALL: Cualquier método de autentificación es válido.
NAMES.DIRECTORY_PATH: Especifica el orden y método para resolver el Identificador de Conexión, y el Nombre del Servicio de Red. Si no lo resuelve, utiliza el siguiente de la lista.
TNSNAMES: Resuelve el nombre del servicio de red, mirando en el archivo Tnsnames.ora del cliente. Aconsejado para redes de distribución con un número reducido de servicios, que cambian en pocas ocasiones.
EZCONNECT: Permite a los clientes conectarse al servidor de la base de datos, sin ninguna configuración. Los clientes usan una dirección TCP/IP sencilla, con un nombre de Host, Puerto y servicio opcional (*Connect usuario/password@:puerto/nombre_servicio*).
LDAP: Resuelve un servicio de bases de datos o un nombre de servicio de red, en un descriptor de conexión, almacenado en un servidor de directorios central. Se agrega automáticamente durante la instalación.
HOSTNAME: El usuario se puede conectar a un servidor de la base de datos, mediante un alias del nombre del Host, en un entorno TCPIP, resuelto en un sistema DNS (Sistema de Nombres de Dominio), en un Servidor de Información de Red (NIS) o en un juego de archivos de */etc./host*, mantenidos centralmente.
NIS: Servicios de Información de Red. Para sistemas Sun Microsytems.

CDS: Servicio de directorio de celdas. Permite a los usuarios, de forma transparente las herramientas y aplicaciones de Oracle, para acceder a las base de datos Oracle, en un entorno distribuido (DCE).

ONAMES: Servicio de directorios de Oracle, compuesto por un sistema ORACLE NAMES SERVERS (almacena información de red para un servicio, junto con un nombre sencillo) y así se puede hacer la conexión con un nombre sencillo, en lugar de una dirección de red larga.

2.4.4 Edición del archivo .oraenv

Cuando el sistema desarrollado necesita enviar una solicitud para que se creen cuentas de correo tiene que seguir el siguiente procedimiento para llevar a cabo la conexión con el servidor remoto. Así mismo el servidor remoto necesitara conectarse a la base de datos y notificarle que el procedimiento fue exitoso o falló.

Procedimiento

1. La aplicación, después de verificar transacciones en curso, permisos de operador, disponibilidad de servidores, etc., genera una cadena de conexión similar a la siguiente:
    a. /usr/bin/ssh -n -f -t -t root@192.168.1.14 "sh /intradius/ABC_MAIL_MX.sh jorge.brito XXXXXXX E usuario 1 1"
2. Podemos observar que se ejecuta un comando SSH en modo background, esto lo detallaremos en la siguiente sección.
3. Cuando el script inicia su ejecución realizar algunas tareas que implica la ejecución de algunos comandos del sistema operativo.
4. Cuando el script trata de informar a la base de datos que ha terminado la ejecución ya sea con falla o con error, entonces tiene que invocar al cliente de Oracle para que este realice la actividad necesaria.
5. Sin embargo, el cliente de Oracle no puede funcionar en background si previamente no se han instanciado las variables de ambiente necesarias (ORACLE_HOME, ORACLE_BASE).
6. Para proceder a instanciarlas el script ejecuta el archivo *oraenv.*
7. Si no se ha configurado previamente con los valores de las variables de ambiente de ORACLE, entonces falla.

Si hemos leído detenidamente las actividades que se llevan a cabo, entonces podemos intuir que el archivo *oraenv* debe contener los valores necesarios para las variables de ambiente. Sin embargo en una instalación "pura" del cliente de oracle, cada vez que se ejecuta el archivo *oraenv* este pregunta por los valores a asignar a las variables ORACLE_HOME y ORACLE_BASE, lo cual evidentemente representa un problema.

Para resolverlo inevitablemente es necesario modificar el archivo de modo que estos valores estén "quemados" y no causen problemas. La ruta en la que se encuentra este archivo es la siguiente: */usr/local/bin/oraenv*.

2.5 Protocolo SSH(Secure Shell)

Cuando surge la necesidad de conectarse a un servidor remoto de forma segura a través del envío de los datos encriptados, entonces el uso de este protocolo es la solución. SSH se diseñó como una alternativa para remplazar a sistemas menos seguros como Telnet o RSH. Así como nace SSH como alternativa de conexión segura, también nace SCP para transferencia de archivos de forma segura.

El protocolo SSH proporciona algunas características de protección, una de ellas es poder "recordar" si una máquina ya realizó una conexión anteriormente. Otra característica es la mejora en la seguridad mediante el empleo de una encriptación de 128 bits. Una de las funcionalidades adicionales es la posibilidad de reenviar aplicaciones *X11*, lo que se traduce en la posibilidad de usar aplicaciones gráficas sobre la red.

Es necesario que comprendamos como se establece la conexión entre dos equipos mediante este protocolo. En primera instancia cuando el cliente inicia la conexión el servidor envía un *handshake* encriptado para que el cliente pueda verificar la comunicación con el servidor correcto. Después de ello se encripta la capa de transporte entre el cliente y el servidor mediante un código simétrico a la vez que comprime los datos que se transmitan a través de ella. Ahora el cliente tiene que loguearse o autenticarse contra el servidor, si la autenticación es correcta entonces se inicia la interacción entre cliente y servidor.

Surge una pregunta del procedimiento descrito anteriormente y es ¿Cuál es el procedimiento empleado para la encriptación de la capa de transporte?. Podemos decir entonces que al iniciar la conexión se realiza un intercambio de claves, seguidamente se determina el algoritmo de encriptación de la clave pública, algoritmo de encriptación simétrica, algoritmo de autenticación de mensajes y el algoritmo *hash* que se empleará. El servidor se identifica con una clave de host única y después del intercambio de claves se crea un valor *hash* para el intercambio y un valor secreto compartido. Estos dos últimos valores varían cada cierto tamaño de bytes transmitidos.

2.5.1 Generación de llaves privadas y públicas

Cuando se inicia una sesión SSH el servidor notifica al cliente cuales son los métodos que tiene disponible para autenticarle, entre uno de esos métodos consta el uso de llaves privadas y públicas, en la cual el cliente genera una llave privada que es utilizada por él y adicionalmente se genera un llave pública que es exportada al equipo servidor encriptadas con 128 bits.

El procedimiento de generación de llaves públicas y privadas posibilita a los clientes el ingreso a un equipo remoto sin la necesidad de hacer uso de una contraseña, en su lugar hace uso de la llave privada que se generó la misma que es verificada por el servidor con su llave pública. De lo descrito podemos entonces intuir que el acceso es unidireccional, ya que el cliente exporta su llave al servidor, pero en ningún momento el servidor exporta una llave al cliente. Por el contrario si necesitásemos que el servidor también se conecte al cliente, entonces el servidor se convertiría en cliente y exportaría su llave al servidor que antes era cliente.

Este método se convierte en un camino seguro para realizar conexiones entre equipos que usen el protocolo SSH, en nuestro sistema la posibilidad de realizar una conexión a un equipo remoto sin necesidad de usar clave automatiza los procesos y ahorra tiempo.

2.5.2 Establecimiento de conexiones en Background

Cuando el sistema operativo del servidor de base de datos realiza la ejecución de un comando, necesariamente genera una espera de CPU hasta que el comando sea procesado. Sin embargo, en una base de datos que es transaccional estas esperas pueden causar más de un dolor de cabeza generando lentitud en todo el sistema y si las esperas de CPU son constantes terminaran por colapsar la base de datos. Debido a que las bases de datos actuales se encuentran ejecutando múltiples sistemas, entonces es crítico considerar lo descrito.

Para que la aplicación estudiada en este documento sea capaz de mitigar el problema, tuvo que necesariamente hacer uso de la ejecución de procesos en background. El problema originado por la ejecución de procesos en background es la dificultad para descubrir si una instrucción fue procesada exitosamente.

Observemos como un adecuado uso del protocolo SSH permite que los comandos puedan ser ejecutados en background sin causar esperas de CPU en la base de datos y por ende sus causas negativas. Entre los parámetros necesarios constan algunos que permiten redireccionar la salida del comando como es el caso del parámetro *"-n"*, y otros en cambio permiten la ejecución del comando en background, veamos un ejemplo y descripción de los parámetros para entender mejor.

Ejemplo:

ssh -n -f -t -t root@192.168.1.14 "sh /intradius/ABC_MAIL_MX.sh jorge.brito XXXXXXX E usuario 1 1"

De acuerdo al manual de especificación de SSH me permito citar:

-n => Redirects stdin from /dev/null (actually, prevents reading from stdin). This must be used when ssh is run in the background.
–f => Requests ssh to go to background just before command execution. This is useful if ssh is going to ask for passwords or passphrases, but the user wants it in the background.
–t => Force pseudo-tty allocation.

Para este ejemplo se invoca al comando *ssh* que solicita la ejecución del script *ABC_MAIL_MX.sh* ubicado en el directorio *intradius* del servidor remoto, después de la ejecución de esta instrucción habremos eliminado un usuario del sistema operativo del servidor 192.168.1.14.

2.6 Samba

Samba es un protocolo libre que permite a servidores Linux, MacOS y Unix ser visibles en entornos Windows tanto como clientes así como servidores de carpetas compartidas. Podríamos decir que Samba es una implementación libre del protocolo *SMB* de Microsoft, aunque en la actualidad este protocolo en sistemas Linux ha adoptado el nombre de CIFS. Este software es escrito, y mantenido hasta la actualidad, por Andrew Tridgell desde 1991. De lo descrito podemos entonces concluir que este software nace como producto del empleo de ingeniería inversa para conectar servidores distintos al de *Microsoft (Windows)* con un protocolo tipo caja negra como SMB.

Samba no solamente provee el servicio para carpetas compartidas sino también algunos otros servicios, entre ellos *WINS*, controlador de dominio, servicio de impresión, pero el servicio que nos lleva a implantarlo es el de carpetas compartidas debido a la necesidad de disponer de una arquitectura en la que cada servidor remoto utilice los mismos scripts para generar nuevas cuentas de correo, redirecciones, etc. La posibilidad de disponerlo en varios sistemas operativos permite que el sistema tratado en esta tesis sea más flexible. Aunque no es tema de esta tesis tratar la instalación del software *Samba*, se expondrá brevemente y ahondaremos un poco sobre algunos aspectos de la configuración.

2.6.1 Instalación del servicio Samba

Para realizar la instalación tenemos dos formas, la primera implica descargar los binarios desde los repositorios del sistema operativo y la segunda necesariamente hace uso de las fuentes del programa que deben ser compiladas para un sistema operativo en particular. Nosotros emplearemos los repositorios del sistema operativo, para ello ejecutamos la siguiente instrucción:

*yum install samba**

La instrucción citada en la línea anterior permite la instalación del servidor samba y todas sus dependencias. Una vez que se haya instalado el servicio se habilitará dos componentes:

- **nmbd**: Provee el servicio de Netbios o nombre de equipo sobre la red.
- **smbd**: Provee el servicio de carpetas compartidas e impresión sobre la red.

La instalación del programa inserta en el sistema operativo algunos archivos, algunos de ellos son de configuración, otros son ejecutables y también existen algunos que guardan un log de lo que está sucediendo. El archivo necesario para iniciar el software *Samba* se encuentra en */usr/local/bin* regularmente, el archivo que inicia el software *Samba* se llama *smb* aunque también podemos iniciar el software con la instrucción:

service smb start

Para configurar samba es necesario modificar el archivo *smb.conf,* en el cual se establece que redes tienen acceso, opciones de registro de logs, nombre de servidor, servicio *WINS* en definitiva todos los servicios que provee samba. Para configurar las carpetas compartidas la sección debe empezar con el nombre del recurso a compartir el mismo que debe estar escrito entre corchetes y las definiciones de permisos, directorio a compartir, usuarios que tienen acceso, etc. se escriben después del nombre entre corchetes.

2.6.2 Ejemplo de configuración samba para esta tesis:

```
[global]
        workgroup = Radius
        server string = Radius_Interfaz
        netbios name = Radius
        ; interfaces = lo eth0 192.168.12.2/24 192.168.13.2/24
        hosts allow = 127. 192.168.1. 10.9.0.
        #----------------------------Líneas para monitorear los acceso al servidor----------
---
```

```
        syslog = 10
        vfs object = extd_audit


[intradius]
        comment = Interfaz Conexion Radius
        path = /intradius
        guest ok = no
        browseable = no
        readonly = no
        force group = intradius
        force create mode = 0771
        force directory mode = 0771
        create mask = 0771
        directory mask = 0771
        valid users = intradius
```

Como podemos observar el sistema de permisos del que se hace uso es similar al ya explicado en secciones anteriores.

Es necesario que observemos gráficamente cual es la estructura empleada para nuestra tesis en el uso de carpetas compartidas, donde el servidor *Radius* también actúa como servidor Samba y comparte los scripts a todos los servidores sobre los cuales se va a realizar alguna operación ya sea esta con el correo o con el servidor de DHCP. En el siguiente capítulo discutiremos algunos detalles adicionales así como la finalidad del servidor Radius.

Con el empleo del servicio Samba solucionamos el problema de la disponibilidad de los mismos scripts en cada servidor remoto, sin embargo es necesario que se defina explícitamente que cada servidor conecte el recurso compartido cuando arranque después de un reinicio del servidor, para obtener este resultado es necesario que se edite el archivo */etc./fstab* que contiene las definiciones necesarias para que las unidades de disco se monten cuando se inicie el equipo.

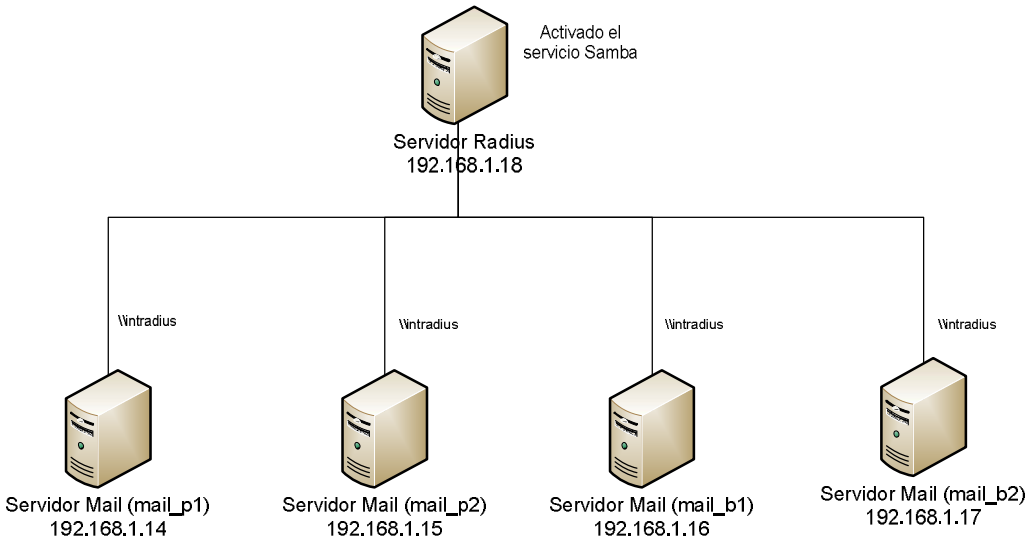


Figura 2-2

La línea de configuración que debe estar presente en este fichero es similar a la siguiente:

\\192.168.1.18\intradius /intradius cifs gid=0,uid=0,username=intradius,password=intrad3005,rw 0 0

La dirección IP 192.168.1.18 corresponde a la dirección del servidor Samba e *\intradius* corresponde al recurso que ha sido compartido. También existe otra forma en la que podemos montar directorios compartidos pero solamente es temporal, es decir, si el servidor se reiniciar entonces el recurso deja de estar disponible. Enuncio el comando necesario en más de una ocasiones nos puede ayudar.

smbmount //servidor/compartido /mnt/destino -o username=usuario%password

Con el comando anterior estamos diciendo que se monte la carpeta compartida *"compartido"* dentro del directorio */mnt/destino* utilizando para ello el usuario y password *usuario%password*. Hasta aquí hemos hablado del por qué y cómo usar el software *Samba*. En el siguiente capítulo nos centraremos más en detalle el uso del servidor *Radius* y la importancia que tiene el mismo en nuestra arquitectura.

CAPÍTULO 3

Configuración de servidores Radius y Access Point

3.1 Generalidades servidor FreeRadius

Cuando hablamos de un servidor que provee servicios de autenticación, autorización y administración, estamos hablando de un servidor que implementa el protocolo *Radius(Remote Authentication Dial-In User Server)* en su funcionamiento. En este momento lo más probable es que estemos preguntándonos que es el protocolo *Radius*; pues bien el protocolo *Radius* es una implantación que permite mediante el canal UDP en el puerto 1812 la autenticación y autorización de usuarios de modo que los mismos puedan hacer uso de los servicios de red.

La capacidad de este servidor para establecer sesiones identificando fecha y hora de inicio y fin, posibilitan que este servidor sea usado como fuente de información para realizar algún proceso de facturación por servicios prestados, muy usado en *ISPs*(*Internet Service Provider*). Para que se pueda realizar el inicio de sesión, el suplicante tiene que ingresar su usuarios y contraseña, datos que son validados haciendo uso de esquemas de validación como CHAP, PAP, MSCHAPV2, EAP o PEAP disponiendo cada uno de ellos de características particulares que discutiremos más adelante.

Existen algunas implementaciones de este protocolo tanto embebidas en el hardware como software de instalación sobre algún sistema operativo en particular, sin embargo la alternativa, para el control de acceso a la red inalámbrica, que se ha adoptado en esta tesis es FreeRadius un software de código abierto que opera sobre sistemas operativos Linux.

Debido al extenso empleo de este protocolo y la importancia derivada de ello, realizaré una explicación un tanto profunda donde empezaremos inicialmente

definiendo conceptos como *AAA, NAS* y los algoritmos de protección existentes para el control de acceso a las redes inalámbricas, para terminar exponiendo la configuración del servidor como de un equipo que necesita ingresar a la red.

El servidor *Radius* también es denominado como *AAA* debido a que proporciona los servicios de autenticación, autorización y registro (*accounting*). Cuando hablamos de autenticación nos estamos refiriendo al proceso en el que se solicita al usuario las credenciales necesarias para poder acceder a consumir los recursos, este proceso se caracteriza principalmente por la solicitud de un usuario y una contraseña. Cuando hemos logrado pasar la primera etapa de autenticación, pasamos a una segunda en donde se autoriza a un usuario el uso de los servicios que se le han asignado, pudiendo ser estos internet, mail, etc. Finalmente la etapa de *accounting* o registro posibilitar llevar una bitácora de cuales fueron los accesos con tiempos de inicio y finalización, cliente que ha ingresado, el *access point* por el cual ingreso (en el caso de tratarse de autentificación de red inalámbrica), etc.

Mediante el protocolo *Radius,* los servidores de autenticación pueden agruparse en jerarquías, de este modo se puede disponer de múltiples servidores distribuidos geográficamente, pero todos consultado a un servidor central para poder autorizar el acceso y posterior consumo de los servicios de red.

Es necesario entender que un computador portátil de un usuario cualquiera no puede consultar directamente al servidor que se encuentra corriendo el servicio de *Radius,* para poder llegar al mencionado servidor tiene que hacer uso de un equipo *NAS(Network Access Server)*.

Podemos decir entonces que un *NAS* es un intermediario que no contiene datos de ningún usuario que requiera acceso a los servicios de red, sino es un equipo que recibe las peticiones del usuario y las retransmite, de forma encriptada, al servidor *Radius* quien a su vez envía una respuesta de vuelta al *NAS* indicándole si debe o no debe dar acceso a los servicios requeridos. Si lo miramos como una arquitectura cliente-servidor el cliente seria el *NAS* y el servidor seria el servidor *Radius,* sin embargo existe una diferencia marcada porque el cliente es el que proporciona el

acceso, mientras que el servidor autoriza ese acceso y el usuario es el que consume los servicios que fueron concedidos y autorizados.

De lo descrito anteriormente, y conociendo que para nuestro caso el *NAS* será los puntos de acceso o *Access Point*, entonces se hace necesario que hablemos un poco acerca de la seguridad de las redes inalámbricas y la vulnerabilidad por posibles ataques en que se incurre, sino se encuentra bien configurada.

3.2 Seguridad en las redes inalámbricas

Los dispositivos de acceso inalámbrico o también llamados *Access Point* permiten a los computadores portátiles conectarse a una red, pero a diferencia de una red que tiene como medio físico de transmisión de datos un cable *UTP*, fibra óptica, la transmisión se realiza por medio de ondas electromagnéticas que se dispersan por el aire. Entonces podemos decir que de cierta manera los datos están "flotando en el aire" y por ello todos los dispositivos que pueden reconocerlos estarían en la capacidad de poder interpretarlos. A lo largo del desarrollo de esta tecnología se han desarrollado métodos para que solamente los usuarios destinatarios de la información puedan leerlos, sin embargo la curiosidad y hasta malicia propicia que otras personas no autorizadas deseen ver la información que se encuentra transmitiendo.

Para evitar que personas no autorizadas vean información que no está dirigido a ellas se han desarrollado métodos de encriptación para proteger la información. Entre los métodos más conocidos constan: WEP, WPA-PSK, WPA-Enterprise, WPA2-PSK, WAP2-Enterprise. Veamos una descripción de cada uno de ellos para entre el motivo por el cual fueron creados.

3.2.1 WEP(Wired Equivalent Privacy)

Es el primer método desarrollado para proteger las redes inalámbricas basadas en el estándar IEEE 802.11 estándar que fue diseñado para redes LAN (Local Area Network). Es el protocolo más inseguro para la protección de una red inalámbrica, "está basado en el algoritmo de encriptación RC4, con una clave secreta de 40 o 104 bits, combinada con un Vector de Inicialización (IV) de 24 bits para encriptar el

mensaje de texto M y su checksum – el ICV (Integrity Check Value). El mensaje encriptado C se determinaba utilizando la siguiente fórmula:

C = [M || ICV(M)] + [RC4(K || IV)]" Guillaume Lehembre.

Existen fallos de seguridad en este protocolo generados por la debilidad del algoritmo *RC4* en la construcción de claves, así mismo los vectores que conforman la clave son demasiado cortos lo que ocasiona que la misma sea descubierta con un número reducido de paquetes trasmitidos a la red. Actualmente todas estas vulnerabilidades ya están explotadas por programas como *Aircrack, AirSnort, etc*. razón por la cual es un crimen que se use este protocolo en redes empresariales.

Existen diferentes formas con las que se puede "romper" la seguridad *WEP*, por ejemplo tenernos el método de suplantación que es un ataque en el que el intruso pretende tomar la identidad de un usuario autorizado en muchos de los casos haciendo uso de la dirección *MAC*, spoofing es un ejemplo. Otro ataque conocido lo representa la denegación de servicio en donde el intruso consigue que los usuarios autorizados no puedan conectarse a la red haciendo uso para tal fin la generación de interferencias.

3.2.2 WAP y WAP2 (Wireless Protected Access)

Después de que se determinó por completo la inseguridad de *WEP*, en el año 2003 se propone el acceso protegido a la red *WAP* (Wireless Access Protected) son sus siglas que después de un año sería certificado con el nombre de *WPA2*. *WAP y WAP2* son protocolos construidos para trabajar con y sin servidores de llaves, en el caso de que no se use un servidor de llaves entonces todas las estaciones de la red usan una llave previamente compartida(*PSK – Pre-Shared-Key*) y este modo con *PSK* se conoce como personal(WAP-Personal, WAP2-Personal). Cuando se usa un servidor de llaves entonces se le conoce a este protocolo como *WAP o WPA2*-Corporativo.

WAP2 como habíamos dicho es la versión certificada de *WAP* pero incluye dos cambios importantes según Julio Cesar Ardita.

- Remplazo del algoritmo Michael por un código de autenticación conocido como el protocolo "Counter-Mode/CBC-Mac" (CCMP), que es considerado criptográficamente seguro.

- Remplazo del algoritmo RC4 por el “Advanced Encryption Standard (AES)” conocido también como Rijndael.

De acuerdo a lo citado por Julio Cesar Ardita, en caso de necesitar confidencialidad mediante el cifrado a nivel de la capa de enlace, la mejor alternativa es hacer uso de *WPA2* corporativo (*WPA-Enterprise*).En caso de adoptar la opción *WPA*-Personal entonces deberemos tener en consideración cual es la clave PSK o pre compartida que usaremos.

Aunque el protocolo *WAP2* es el protocolo más seguro que existe actualmente no deja de tener sus “puertas abiertas” o “huecos de seguridad” según un artículo publicado por Md Sohail Ahmad. Veamos como la fortaleza del protocolo *WPA2* se debilita bajo ciertas circunstancias.
WAP2 al igual que cualquier protocolo de seguridad está expuesto a ataques, estos se vienen dando desde hace algunos años como lo ilustra el gráfico publicado por Ahmad.

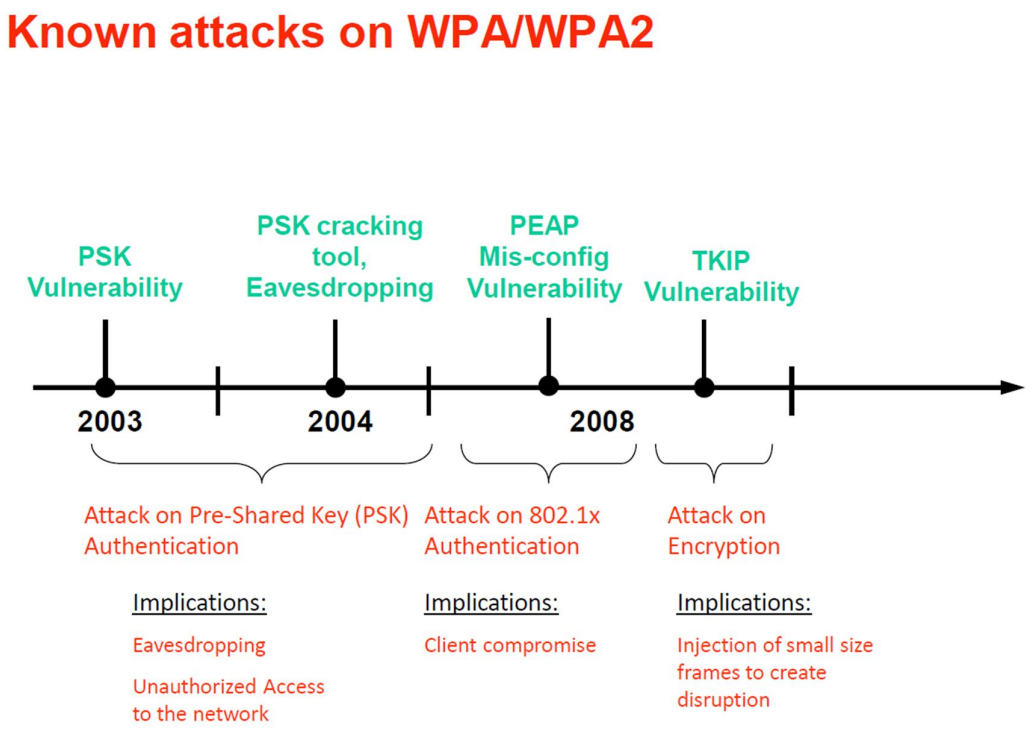


Figura 3-1: AHMAD S. (2012). WAP TOO!. Obtenido desde http://www.defcon.org/images/defcon-18/dc-18-presentations/Ahmad/DEFCON-18-Ahmad-WPA-Too.pdf.

Para resolver la vulnerabilidad *PSK*, se sugiere no usar esta autenticación en redes que no sean privadas o de hogares. Sin embargo esta solución no ayuda a las empresas que usan redes inalámbricas, para ello se sugiere como solución no ignorar la validación de certificados en la configuración de los clientes, además como última alternativa se propone que se use la encriptación *AES* en lugar de *TKIP*.

Debido a las vulnerabilidades presentadas es apropiado que nos cuestionemos si el protocolo es en realidad seguro. Para responder a la interrogante planteada es necesario que describamos más a fondo este protocolo. *WPA2* hace uso de dos tipos de encriptación de datos:

- **Pairwise Key (PTK):** Usado para proteger las tramas de tipo **unicast.**
- **Group Key (GTK):** Usado para proteger grupos de datos como los usados en el envío de paquetes broadcast que normalmente usa tramas *ARP*. Estos paquetes son enviados únicamente por el *Access Point*.

Nuevamente la ilustración proporcionada por Ahmad ayuda para que entendamos más claramente los conceptos de *PTK y GTK*.

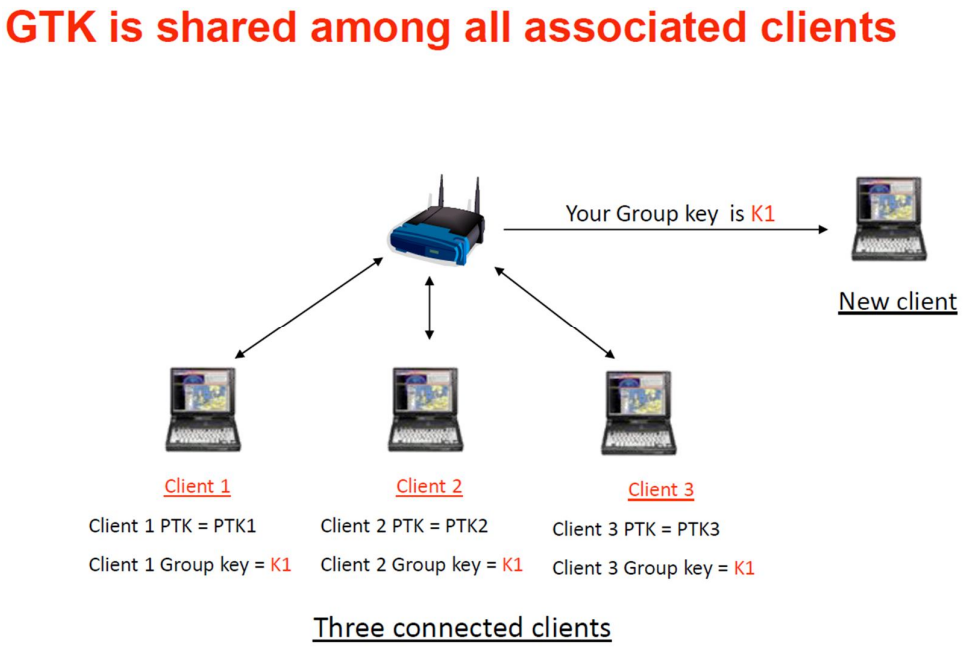


Figura 3-2: AHMAD S. (2012). WAP TOO!. Obtenido desde http://www.defcon.org/images/defcon-18/dc-18-presentations/Ahmad/DEFCON-18-Ahmad-WPA-Too.pdf.

Como habíamos dicho antes los únicos equipos que generan los paquetes *GTK* son los *Access Point*, sin embargo pueden generar los clientes paquetes *GTK*, la respuesta es afirmativa pero bajo ciertas condiciones:

1. Debe conocer el *GTK* generado por el *Access Point.*
2. Se debe conocer el *KeyID*
3. Se debe conocer el *PN* (*Packet Number*).

Usando programas adecuados podemos obtener esa información, siempre y cuando ya nos conectemos, pues todos los usuarios conectados lo están receptando cuando transmiten paquetes al *Access Point.*
Entonces con los datos obtenidos el atacante puede generar los paquetes necesarios e "inyectar" a clientes sin que ellos se den cuenta de lo que está sucediendo. A este tipo de ataque lo llamaremos "interno" debido principalmente a la necesidad de que le usuario se haya previamente conectado a la red para iniciar sus actividades maliciosas.

Si se llega a ingresar en la red por un tiempo adecuado se podría realizar un espionaje del tráfico con el enmascaramiento de direcciones *ARP* que permitiría redireccionar el tráfico en la red, se podría también redireccionar los paquetes *TCP*, escaneo de puertos, inyección de malware, etc.

Con el ataque de "*ARP Poisoning*" el atacante tiene la capacidad de envenenar los paquetes *ARP* de un cliente de modo que le puede "decir" que el *Gateway* es su máquina (máquina del atacante) y con ello escanear todo el tráfico que se encuentre generando en busca de información; el siguiente grafico ilustra lo descrito:

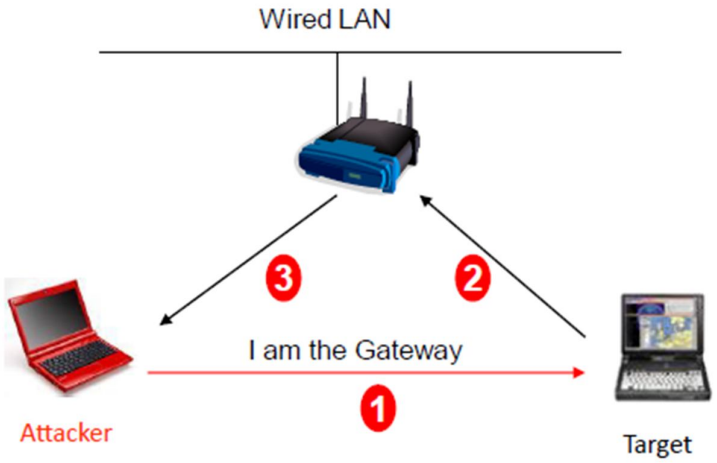


Figura 3-3: AHMAD S. (2012). WAP TOO!. Obtenido desde http://www.defcon.org/images/defcon-18/dc-18-presentations/Ahmad/DEFCON-18-Ahmad-WPA-Too.pdf.

Para que el atacante no sea descubierto entonces la mejor opción para ejecutar el envenenamiento de paquetes *ARP* es que el *Access Point* que está conectado a la red cableada, y que puede disponer de alguna herramienta de detección de envenenamiento de chache *ARP*, no procese ningún paquete; para ello se establece una comunicación directa entre la máquina del atacante y el cliente. El *Access Point* para tratar de mitigar este problema realiza un refresco del *PN (Packet Number)* cada cierto tiempo enviando un número *PN* mayor al actual.

La figura en la parte inferior ilustra lo descrito:

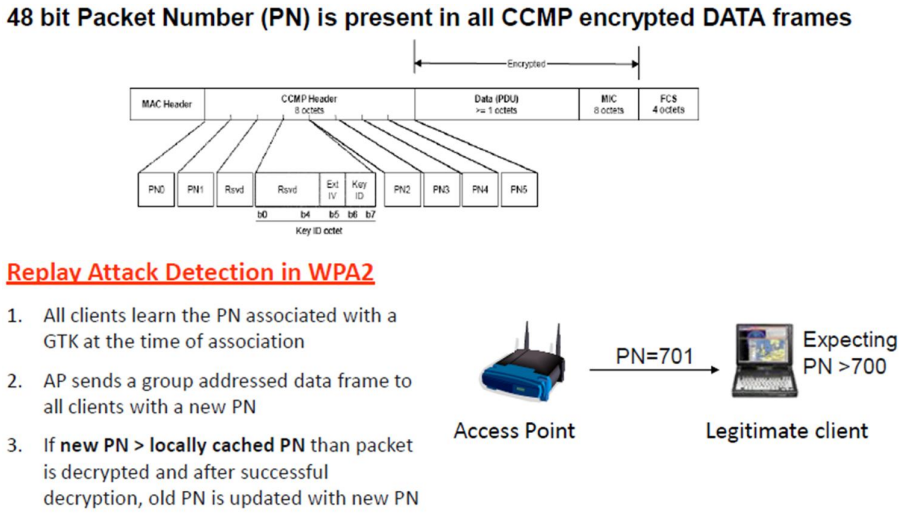


Figura 3-4 AHMAD S. (2012). WAP TOO!. Obtenido desde http://www.defcon.org/images/defcon-18/dc-18-presentations/Ahmad/DEFCON-18-Ahmad-WPA-Too.pdf.

Otra de las formas para solucionar el problema consiste en usar *PSPF* (Public Secure Packet Forwarding) que permite cortar la comunicación entre el *Access Point* y el cliente sin embargo este procedimiento no siempre funciona. Otra de las desventajas de usar *PSPF* es que no todos los dispositivos lo soportan y en muchas ocasiones se hace necesario realizar un upgrade a los equipos *Access Point.*

Otra de las alternativas a ser analizadas por su uso en la seguridad de redes inalámbricas es *EAP (Extensible Authentication Protoco*l). "EAP proporciona algunas funciones comunes y un método para negociar el mecanismo de autenticación a usar. Actualmente hay más de 40 métodos distintos. En esta práctica haremos uso del denominado EAP protegido (PEAP) para la autenticación de nuestro usuario en la red inalámbrica". Nosotros usaremos un suplicante con sistema operativo Windows 7 y el método de autenticación MSCHAPv2, la versión de PEAP es empleada por Microsoft.

Método de autenticación Mschapv2 (Microsoft Challenge Handshake Authentication Protocol versión 2)

Protocolo desarrollado por la casa Microsoft para permitir la autenticación a redes inalámbricas haciendo uso de usuarios y contraseñas. Cuando se usa este método de autenticación tanto el servidor como el cliente deben demostrar que conocen la contraseña del usuario para que la autenticación sea correcta. "Con EAP-MS-CHAP v2, tras una autenticación correcta, los usuarios pueden cambiar sus contraseñas y se les notifica cuándo caducan". Para poder hacer uso de *MS-CHAP v2* es necesario que esté disponible la opción PEAP en la configuración del equipo del cliente.

PEAP es un método que mejora la seguridad de autenticación al tradicional ***EAP,*** *PEAP* proporciona características de seguridad tales como:

- Un canal de cifrado para proteger los métodos EAP.
- Generación de clave dinámicas.
- Almacenamiento de claves en cache lo que permite a un equipo moverse entre distintos puntos de acceso (*Access Point*) y conectarse rápidamente a la red.
- Autenticación contra el servidor de los puntos de acceso. Garantiza que *Access Point* no autorizados no tengan acceso a la red.

De acuerdo a la documentación técnica remitida por la Microsoft en su página web, el método de autenticación PEAP hace uso para autenticar dos fases principales:

- **Autenticación de servidor y creación de un canal de cifrado de TLS.** El servidor se identifica ante un cliente proporcionando a éste información del certificado. Una vez que el cliente comprueba la identidad del servidor, se genera un secreto maestro. A continuación, las claves de sesión obtenidas del secreto maestro se utilizan para crear un canal de cifrado de TLS que cifra toda la comunicación posterior entre el servidor y cliente inalámbrico.

- **Conversación de EAP y autenticación de usuario y de equipo cliente.** Mediante el canal de cifrado de TLS se encapsula una conversación de EAP completa entre el cliente y el servidor. Con PEAP puede utilizarse cualquiera de los diversos métodos de autenticación de EAP, como contraseñas, tarjetas inteligentes y certificados, para autenticar al usuario y al equipo cliente.

Cuando el administrador de la red establece que la conexión inalámbrica debe ser refrescada cada cierto número de minutos, entonces debe también habilitar la opción de reconexión rápida para no causar que los usuarios tengan que digitar nuevamente las claves para ingreso a la red. Existe además otra ventaja en el uso de *PEAP* con *MSCHAP v2* y es que cuando los usuarios se desplacen hacia otros puntos de acceso estos ya no volverán a pedir credenciales para conexión a la red sino que esos datos se extraen de la cache a menos que las credenciales en el cache haya expirado. Es necesario que observemos cual es el proceso adoptado por *MS-CHAP v2*, un proceso unidireccional con contraseña cifrada, para la autenticación de un usuario:

1. El autenticador (el servidor de acceso remoto o el servidor IAS o el servidor *Radius*) envía al cliente de acceso remoto un desafío formado por un identificador de sesión y una cadena de desafío arbitraria.
2. El cliente de acceso remoto envía una respuesta que contiene:
    a. El nombre del usuario.
    b. Una cadena de desafío arbitraria del mismo nivel.

c. Una codificación unidireccional de la cadena de desafío recibida, la cadena de desafío del mismo nivel, el identificador de sesión y la contraseña del usuario.

3. El autenticador comprueba la respuesta del cliente y devuelve una respuesta que contiene:
    a. Una indicación de si el intento de conexión ha sido correcto o no.
    b. Una respuesta autenticada basada en la cadena de desafío enviada, la cadena de desafío del mismo nivel, la respuesta codificada del cliente y la contraseña del usuario.
4. El cliente de acceso remoto comprueba la respuesta de autenticación y, si es correcta, utiliza la conexión. Si la respuesta de autenticación no es correcta, el cliente de acceso remoto termina la conexión.

Para que el servidor pueda autenticar haciendo uso del protocolo *MS-CHAP v2* en necesario que se habilite la opción, aunque en servidores Windows está habilitada por defecto. Además es necesario que el equipo *Access Point* también disponga de la opción para autenticar haciendo uso de *MS-CHAP v2*

MS-CHAP v2 entre una de sus características hace uso de la contraseña del usuario para generar una clave que permita encriptar la información que está siendo trasmitida, a más de ello este protocolo hace una validación para averiguar que el servidor conozca o tenga acceso a la contraseña del usuario que en caso de ser afirmativo entonces se procede con el proceso de autenticación. Por último para garantizar la seguridad en la información este protocolo genera dos claves de cifrado distintas tanto para enviar como para recibir información.

3.3 Módulo de conexión con Oracle

La arquitectura de este sistema amerita que el repositorio donde deben residir las credenciales de acceso de los usuarios es una base de datos Oracle. Para que ello suceda el servidor que corre el servicio de *Radius* debe ser capaz de hablar el mismo idioma que utiliza el RDBMS, sin embargo de forma nativa no lo hace. Para lograr ese objetivo es necesario que se instale el cliente de Oracle en el servidor *Radius* y se configure los archivos de consulta hacia nuestro sistema.

A continuación detallaremos el proceso de instalación y configuración del servidor *Radius* así como del módulo de conexión con la base de datos Oracle.

3.4 Instalación y configuración Servidor que proporciona el servicio *Radius*

El sistema operativo donde se propone realizar la instalación es Centos 5 de 64 bits con los paquetes por defecto proporcionados por el fabricante. Una vez que dispongamos del sistema operativo funcionando y con acceso a internet entonces instalaremos los siguientes paquetes; tomar en consideración que el símbolo “>” representa el prompt de una terminal del sistema operativo.

3.4.1 Paquetes instalados

> yum install openssl

> yum install openssl-devel

> yum install gcc

> yum install gcc-c++

> yum install libiodbc

Ahora descomprimimos el paquete libeap-ikev2 y lo instalamos con el siguiente procedimiento:

> tar –zxvf libeap-ikev2-0.2.1.tar.gz

> cd libeap-ikev2-0.2.1

> ./configure

> make

> make install

Instalado estos paquetes es necesario que se instale el cliente de Oracle para nuestro caso de la base de datos 11g y crear dentro del directorio “lib” que se encuentra en el “*ORACLE_HOME*” los siguientes enlaces simbólicos:

> ln -s libnnz11.a libnnz10.a

> ln -s libnnz11.so libnnz10.so

Listo entonces ahora es hora de descomprimir los binarios del software FreeRadius y nos ingresemos a la carpeta descomprimida, para ello seguimos el siguiente procedimiento:

> tar –zxvf freeradius-server-2.1.10.tar.gz
> cd freeradius-server-2.1.10

Para poder instalar el software FreeRadius en el sistema operativo desde las fuentes es necesario que primero se configure para ello ejecutamos:

> ./configure –with-raddbdir=/etc/raddb –with-logdir=/var/log/radius –with-oracle-lib-dir=/u01/app/oracle/product/11.2.0/dbhome_1/lib/ --with-oracle-include-dir=/u01/app/oracle/product/11.2.0/dbhome_1/rdbms/public

Nos saldrá 2 errores con lTNC y con iodbc los cuales no los usaremos debido a esto continuamos, terminada la configuración el siguiente paso es la compilación e instalación:

> make
> make install

Si el usuario “radiusd” no fue creado con la instalación es necesario crearlo con el usuario root en el S.O. Una vez creado el usuario es necesario cambiar el propietario de la carpeta “raddb” el comando a continuación ejecutamos dentro del directorio /etc.

>chown root:radiusd –R raddb/

Para configurar el software FreeRadius tenemos que ubicar los archivos *radiusd.conf, sql.conf, eap.conf* y el archivo *dialup.conf* ubicado en el directorio */etc./raddb/sql/oracle*. En el archivo *radiusd.conf* tenemos que descomentar la línea “$INCLUDE sql.conf” mientras que en el archivo *sql.conf* tenemos que editar los datos relacionados con la conexión a la base de datos:

>radius_db="(DESCRIPTION=(ADDRESS=(PROTOCOL=TCP)(HOST=192.168.20.62)(PORT=1521))(CONNECT_DATA=(SID=pdmcert)))"

En la línea descrita el parámetro HOST hace referencia al nombre de dominio o dirección IP de la máquina que corre el servicio de base de datos. PORT nos ilustra cual es el puerto que el servidor de base de datos está usando y por último debemos describir que el SID es el identificador de la base de datos que en caso de no conocerla el administrador de la base de datos nos puede ayudar.

Para "decirle" al software FreeRadius que debe consultar la información necesaria para autenticar y autorizar a un usuario desde la base de datos es necesario que descomentemos las líneas que contenga la palabra *sql* dentro de los archivos */etc./raddb/site-enablesd/default* y */etc./raddb/site-enabled/default.*

Debido a que el sistema que hemos desarrollado tiene condiciones para acceder a la red inalámbrica entonces se hace necesario que editemos el archivo *dialup.conf* y modifiquemos las líneas de consulta de los *Access Point* aceptados y de los usuarios aceptados de modo que queden de la siguiente forma:

3.4.2 Access Point

nas_query = "SELECT NAS_ID, NAS_IPV4, NAS_NOMBRE_CORTO, NAS_TIPO, NAS_CONTRASENIA_SERVIDOR, NAS_SERVIDOR FROM ${nas_table}"

3.4.3 Usuarios Autorizados

authorize_check_query="SELECT USU_ID,USU_USUARIO_SISTEMA,USU_ATTRIBUTE,USU_PASSWORD,USU_OP FROM ${authcheck_table} WHERE USU_USUARIO_SISTEMA = '%{SQL-User-Name}' AND USU_ESTADO='ACTIVO' ORDER BY USU_ID"

Para terminar existen algunos pasos adicionales como el cambio de permisos de algunos directorios, cambio de propietario de las carpetas, servidor virtual y la

generación de certificados digitales. Enumeremos los pasos que nos permitirán disponer de un servidor *Radius* funcional.

1. Nos queda únicamente cambiar los permisos de las carpetas:
    a. usr de 755 a 775
    b. /usr/local de 755 a 775
    c. /usr/local/var de 755 a 775
    d. /usr/local/var/run de 755 a 775
    e. /usr/local/var/run/radiusd de 755 a 775
2. Cambiamos el propietario de la carpeta radiusd
    a. chown root:radiusd radiusd
3. Es necesario crear un servidor virtual para ello dentro del directorio: /etc./raddb/sites-enabled se deberá crear el fichero: 192.168.1.18 (ip_servidor_radius) con el siguiente contenido.

```
server 192.168.20.5 {
        $INCLUDE    ${confdir}/sites-available/default
}
```

Guardamos los cambios.

Nota: Cabe mencionar que si vamos a utilizar la autenticación a través de EAP necesitaremos crear unos certificados que serán instalados en el cliente, existen en el directorio cert unos certificados creados automáticamente que solamente deben ser utilizados en ambientes de prueba y no de producción, para la creación de certificados para en ambientes de producción o implementación usaremos:

1. Comentamos la línea make_cert_command, del archivo eap.conf
2. Primero limpiamos los certificados creados por defecto
    a. Ingresamos al directorio /etc./raddb/certs y ejecutamos
        i. rm –f *.pem *.der *.csr *.crt *.key *. p12 serial* index.txt*
    b. Editamos los archivos correspondientes a ca.cnf, server.cnf y client.cnf
        i. ca.cnf => Editamos los campos country, state, etc. De la sección [certificate_autority], así como los valores para las contraseñas en “input_password” and “output_password” de la

sección [req]. El valor de los password debe ser igual del archivo eap.conf

- Ejecutamos make ca.pem
- Ejecutamos make ca.der

ii. Server.cnf Editamos los campos country, state, etc. De la sección [server], así como los valores para las contraseñas en "input_password" and "output_password" que debe ser igual al de ca.cnf, el valor de conmíname debe ser diferente del de ca.cnf

- make server.pem
- Para verificar que los certificados sean compatibles con Windows Xp podemos ejecutar: make server.csr

iii. Cliente.cnf => Editamos los campos country, state, etc. De la sección [server], así como los valores para las contraseñas en "input_password" and "output_password"el password debe ser diferente del de los archivos anteriores y las que se usaran para ingresar a la red inalámbrica. El valor de commonName se usara como nombre de usuario para ingresar a la red.

- make client.pem

c. Si en caso de presentase errores al momento de crear el certificado de server.pem, entonces antes deberemos correr:
   i. make clean
   ii. make destroycerts
   iii. make index.txt
   iv. make serial
   v. make random
d. Finalmente creamos el archivo dh con el siguiente comando:
   i. openssl dhparam –out dh 1024 o make dh
e. Y nuevamente ejecutamos los comandos para crear los certificados del ca, server y cliente.

3. Antes de iniciar el servidor Radius es necesario instanciar las variables de ambiente de oracle, para ello se debe ejecutar el siguiente comando:
   a. . oraenv

4. Ahora iniciamos el servidor radius(Ojo que la configuración del tnanames debe estar realizada para que el cliente se conecte a la base)
    a. radiusd -X
5. Realizamos una prueba de conectividad al servidor radius
    a. radtest jbrito acjb9906 localhost 1812 testing123

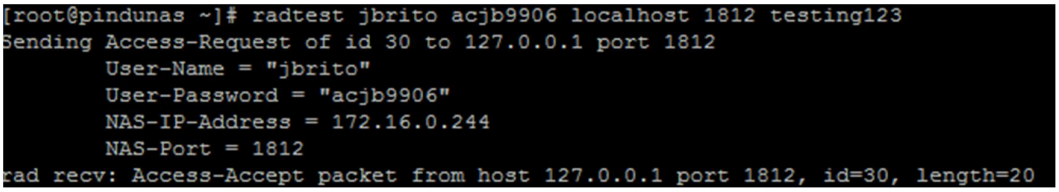


Figura 3-5

Si hemos recibido como respuesta del servidor un Access-Accept todo ha ido bien.

3.5 Configuración de Access Points

Otro de los pasos a seguir en la implementación del sistema en estudio es la configuración de los clientes del servidor *Radius* que pueden ser los dispositivos emisores de ondas inalámbricas o *Access Point* o en su defecto los *NAS(Network Access Server).* Entre las configuraciones como veremos consta la definición del protocolo a usar, el servidor *AAA* y la clave secreta de comunicación entre servidor y cliente.

Hemos tomado el dispositivo *Access Point* TP-Link modelo TL-WR841N como ejemplo para las pantallas a continuación.

Para iniciar la configuración ingresamos a la consola desde la dirección *URL* proporcionada por el fabricante aunque en nuestro caso la hemos modificado para que concuerde con la red en la que se encuentra el servidor *Radius.* Todos los fabricantes de equipos tienen sus propias referencias de cómo acceder al panel de administración, es necesario que no remitamos a los mismos para iniciar esta tarea en caso de no conocer este dato.

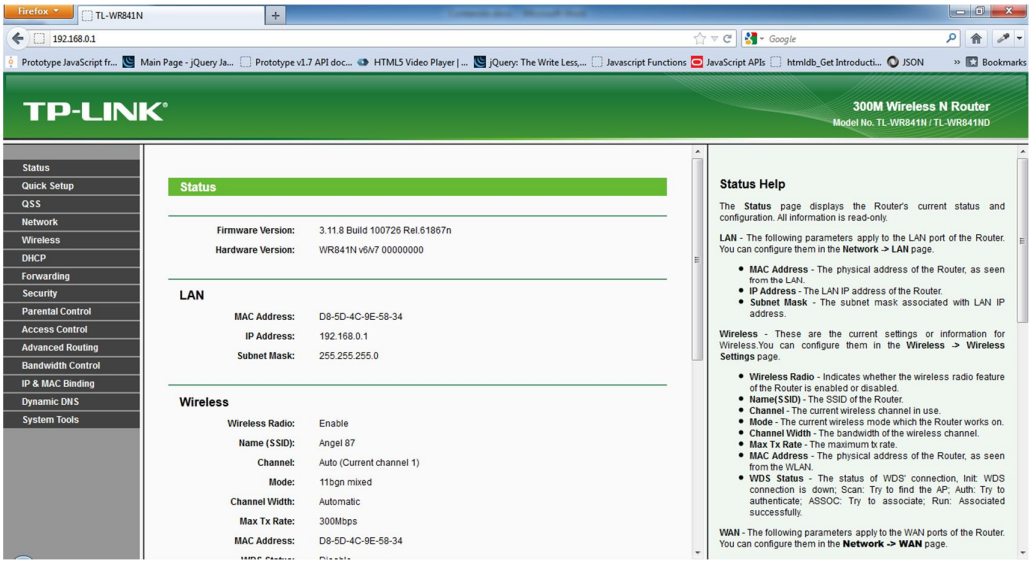


Figura 3-6: [Imagen de estado de access point]. Obtenido desde: Consola de administración de access point.

Lo que nos interesa es ingresar a la opción del dispositivo en la que podamos insertar los datos relacionados con el servidor *AAA.* Dependiendo del equipo es probable que a más de las opciones de configuración para autenticación también sea necesario establecer los valores correspondientes para permitir al servidor *Radius* registrar los accesos concedidos.

De lo que podemos observar en la figura inferior notamos que se debe especificar un puerto y una contraseña del servidor *Radius,* estos parámetros son establecidos en el servidor al momento de su configuración. Por último es necesario que nos percatemos de la línea que dice: ***Group Key Update Period,*** este parámetro como habíamos visto en la sección anterior permite de cierta manera mitigar el problema de seguridad, generado cuando una pc portátil ingresa a la red, llamado "ataques al interior" pues permite el refresco del parámetro *PN(Packet Number).*

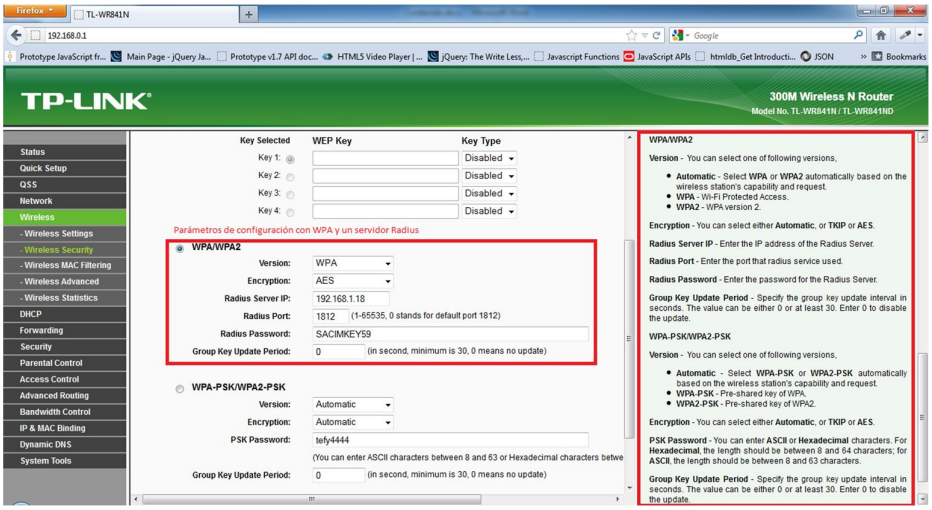


Figura 3-7: [Imagen de configuración seguridad wifi]. Obtenido desde: Consola de administración de access point.

Esta es la configuración completa para el cliente, sin embargo también es necesario que se detalle cual es la configuración para que un usuario (PC Portátil) pueda ingresar a la red. Se ha tomado como ejemplo un equipo con Windows 7 de 64 bits para demostrar cuales son los parámetros que deben ser establecidos.

3.6 Configuración Cliente

Para iniciar la configuración es necesario que ingresemos a la configuración de red de Windows. Para ingresar a esta sección tenemos que irnos al panel de control, cuando estemos en el panel de control tenemos que seleccionar la opción *Centro de redes y recursos compartidos* lo cual abre una nueva ventana con un menú lateral izquierdo en el cual daremos clic en *Administrar redes inalámbricas* lo cual despliega la pantalla capturada en la imagen inferior.

Figura 3-8: [Imagen de adición de red wireless]. Obtenido desde: Consola de administración Windows 7.

Seleccionamos la opción de añadir o en este caso *Add* por estar nuestro sistema operativo en Ingles lo cual visualiza la pantalla a continuación.

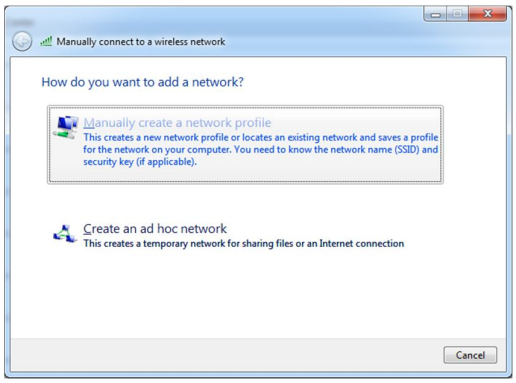


Figura 3-9: [Imagen de configuración de red wireless]. Obtenido desde: Consola de administración Windows 7.

Damos clic en la primera opción que nos permite crear la configuración manualmente, deberemos establecer el nombre de la red inalámbrica en este caso es "prueba", el tipo de seguridad que es "WPA-Enterprise" o "WPA-Empresa" y el tipo de encriptación que podemos usar TKIP o AES, en nuestro caso es TKIP.

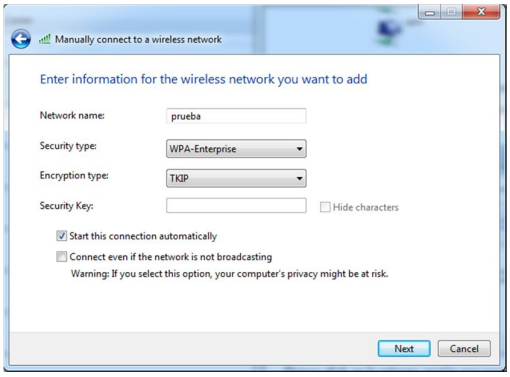


Figura 3-10: [Imagen de configuración de red wireless]. Obtenido desde: Consola de administración Windows 7.

Terminada esta configuración se solicitarán datos adicionales para lo cual se habilita otra opción que nos dice en nuestro caso: **Change connection setting** o en español algo como: cambiar las configuraciones de conexión, daremos clic en la opción y continuaremos.

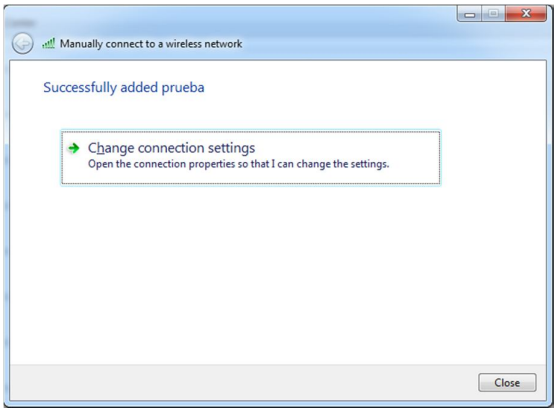


Figura 3-11: [Imagen de configuración de red wireless]. Obtenido desde: Consola de administración Windows 7.

Ahora definimos algunos aspectos relacionados con el método de autenticación para lo cual debemos seleccionar la pestaña de seguridad en nuestro caso "Security" y posteriormente el botón de configuración (Settings).

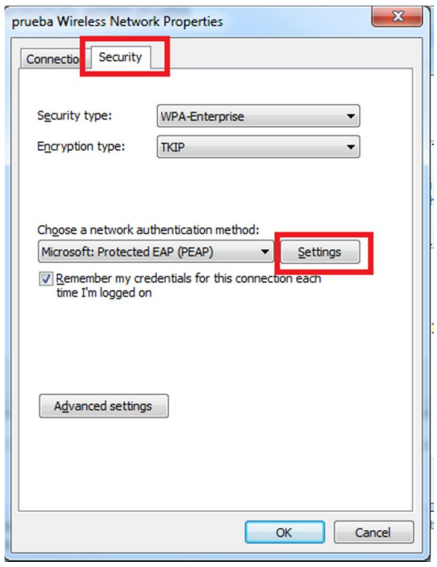


Figura 3-12: [Imagen de configuración de red wireless]. Obtenido desde: Consola de administración Windows 7.

Dentro de esta opción tenemos configuraciones que permiten definir si usaremos certificados digitales o un usuario y contraseña para ingresar a la red. En nuestro caso haremos uso de usuarios y contraseñas para ello desmarcamos la opción que solicita la validación de certificados del servidor en nuestro caso presenta el mensaje: "Validate server certificate". Por defecto en sistemas operativos Windows se tiene

habilitado el uso del nombre de usuario y clave con el que se ingresa al sistema operativo para que el servidor *Radius* valide e ingrese a la red con esas credenciales, nosotros deshabilitaremos esa opción puesto que usamos un repositorio de usuarios y claves que cada usuario conoce.

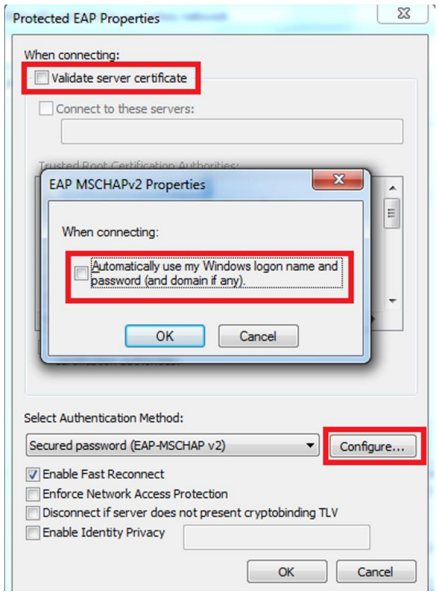


Figura 3-13: [Imagen de configuración de red wireless]. Obtenido desde: Consola de administración Windows 7.

Aceptaremos todos los cambios y por último le "decimos" al sistema operativo que use siempre los datos proporcionados por el usuario, para conseguirlo es necesario que se ingrese al botón de configuraciones avanzadas "Advanced Settings" y seleccionemos el modo de autenticación a "Autenticación de Usuario".

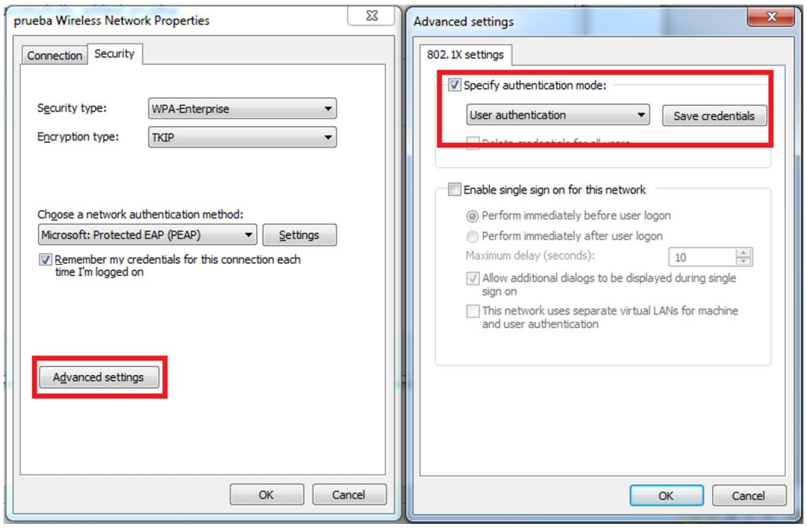


Figura 3-14: [Imagen seguridad wireless]. Obtenido desde: Consola de administración Windows 7.

3.7 Conclusión

Podemos concluir de este capítulo que las redes inalámbricas proporcionan una flexibilidad enorme para interconectar dispositivos que hacen uso del aire como medio de transmisión de datos, sin embargo esta flexibilidad tiene un costo en seguridad porque la información se encuentra a disposición de todos los equipos receptores dentro del rango de transmisión, y aunque actualmente se disponga de métodos para proteger la información que "flota" en el espectro electromagnético siempre existirán personan que deseen ver la información que no está autorizada a ellos.

Del problema evidenciado en el párrafo anterior surge la alternativa de autenticación con *Radius* que no sólo mejora considerablemente la seguridad en la red wireless sino que también posibilita que desarrollos web como el tratado en este documento, pueda administrar y monitorear los usuarios que tendrán acceso a la red.

Las ventajas, además de las ya mencionadas, que proporciona la encriptación de la información con métodos criptográficos seguros hacen que el disponer de un servidor *Radius,* como centro de autenticación y registro, sean de uso obligatorio en todas la redes inalámbricas empresariales. Las soluciones dependerán del presupuesto, sin embargo el uso de FreeRadius (herramienta open source) es una muy buena alternativa a adoptar si no se dispone del presupuesto.

CAPÍTULO 4

Implementación, análisis de resultados y solución de problemas encontrados.

4.1 Implementación

Para implementar el sistema se hizo uso de algunas herramientas en su mayoría de tipo OpenSource, sin embargo también se emplea herramientas que al ser llevadas a un uso comercial estarán sujetos a la aplicación de licencias por parte del proveedor. Es conveniente que se enumere cuáles son las herramientas usadas para obtener el sistema en total funcionamiento. Empezaremos primeramente por los sistemas operativos para continuar con el servidor de aplicaciones y por último con los servicios de *Radius, DHCP, etc.*

4.1.1 Sistemas operativos

Para aumentar el número de usuarios que pueden hacer uso de la herramienta, todos los sistemas operativos son de código abierto, para nuestro caso en particular son Centos 5.5, 6.2 de 64 y 32 bits. Se ha seleccionado Centos por ser un sistema operativo robusto de tipo empresarial, y compatible totalmente con Redhat Enterprise Linux pues nace a partir de este.

Tabla 4-1: Sistemas operativos instalados

Sistema Operativo	Hostname	Ver.	Arquitect.	Virt.	IP	Mem.
CentOS release 6.2 (Final)	BowepCorp	6.2	x86_64	NO	192.168.1.11	16384
CentOS release 5.5 (Final)	mailp1	5.5	x86_64	SI	192.168.1.14	512
CentOS release 5.5 (Final)	mailp2	5.5	x86_64	SI	192.168.1.15	512
CentOS release 5.5 (Final)	mailb1	5.5	x86_64	SI	192.168.1.16	512
CentOS release 5.5 (Final)	mailb2	5.5	x86_64	SI	192.168.1.17	512
CentOS release 5.5 (Final)	radius	5.5	x86_64	SI	192.168.1.18	2048
CentOS release 5.5 (Final)	Database	5.5	x86_64	SI	192.168.1.19	5120
CentOS release 5.5 (Final)	weblogic	5.5	i386	SI	192.168.1.20	4096
CentOS release 5.5 (Final)	dhcpp1	5.5	x86_64	SI	192.168.1.21	512

Aunque el sistema operativo citado en la primera fila del recuadro anterior no tiene un papel completamente activo, lo hemos enumerado porque haciendo uso del hypervisor *qemu*, soporta todas las máquinas virtuales utilizadas en el sistema.

4.1.2 Servidor de Aplicaciones

Uno de los elementos importantes en la arquitectura adoptada por el sistema en estudio lo constituye el servidor de aplicaciones, debido principalmente a que es expuesto directamente a los usuarios que pueden ser internos (al interior de una empresa) o externos (expuesto al internet) y porque sirve de soporte para la aplicaciones web. La fiabilidad requerida entonces cobra mucha importancia, y un servidor de aplicaciones como *Weblogic* es una muy buena alternativa para dar solución a este requerimiento. Sin embargo, la fiabilidad viene dado de la mano con el costo de licenciamiento de esta herramienta que puede variar entre algunos miles de dólares.

Aunque existen otras alternativas como *Tomcat*, más ajustadas a un bajo presupuesto, estas no son 100% compatibles con *APEX (*herramienta de Oracle*)* que es la plataforma donde se ha desarrollado el sistema en estudio. Debido a la importancia que tiene Weblogic entonces detallaré los pasos más importantes a seguir para concretar la instalación de esta herramienta.

4.1.2.1 Instalación servidor de aplicaciones Weblogic

Se tiene que descargar los binarios de instalación desde la página de Oracle, una vez que lo tengamos tenemos que seguir el siguiente procedimiento:

1. Necesitamos cambiar los permisos del archivo descargado de modo que pueda ser ejecutado.
    a. chmod a+x wls1035_oepe111172_linux32.bin
2. Habilitamos las conexiones al display con el usuario root:
    a. xhost +
3. Ahora lo que tenemos que realizar es crear los grupos necesarios por la aplicación para que pueda ser instalada.
    a. groupadd dba
    b. groupadd oinstall
    c. useradd –g oinstall –G dba oracle

4. Para que el programa sea instalado sin problemas, el sistema operativo debe tener algunos paquetes instalados.
    a. yum install gcc gcc-c++ setarch sysstat libaio libaio-devel libstdc++ libstdc++-devel compat-libstdc++-296 compat-db control-center glibc-common binutils make openmotif22 openmotif unixODBC unixODBC-devel
5. Además de los paquetes instalados también es necesario que el sistema operativo sea tuneado, para ello se debe modificar algunos parámetros del fichero */etc./security/limits.conf*
    a. soft nofle 4096
    b. hard nofile 4096
6. Asignamos una clave al usuario *oracle* que hemos creado anteriormente.
    a. passwd oracle
7. Ahora es necesario que nos cambiemos de usuario e ingresemos como usuario *oracle*.
    a. su – oracle
8. Ahora ejecutaremos el archivo de instalación.
    a. ./wls1035_oepe111172_linux32.bin -Djava.security.egd=file:/dev/./urandom
9. Después de ejecutado el archivo se despliega la siguiente pantalla.

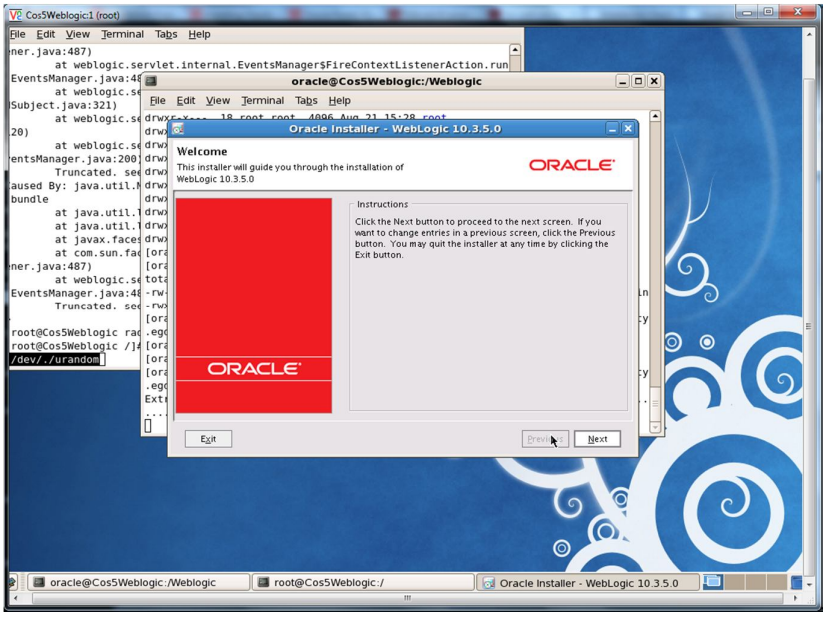


Figura 4-1: [Imagen instalación weblogic]. Obtenido desde: Wizard de instalación del software.

Damos clic en siguiente y especificamos cual es la ruta donde se realizará la instalación del programa, después definimos el tipo de instalación que para nuestro caso es *custom*. Cuando seleccionamos la opción *custom* es necesario que en la siguiente ventana se seleccione las opciones: SUN jdk 1.6.0 y también *Oracle JRockit 1.6.*

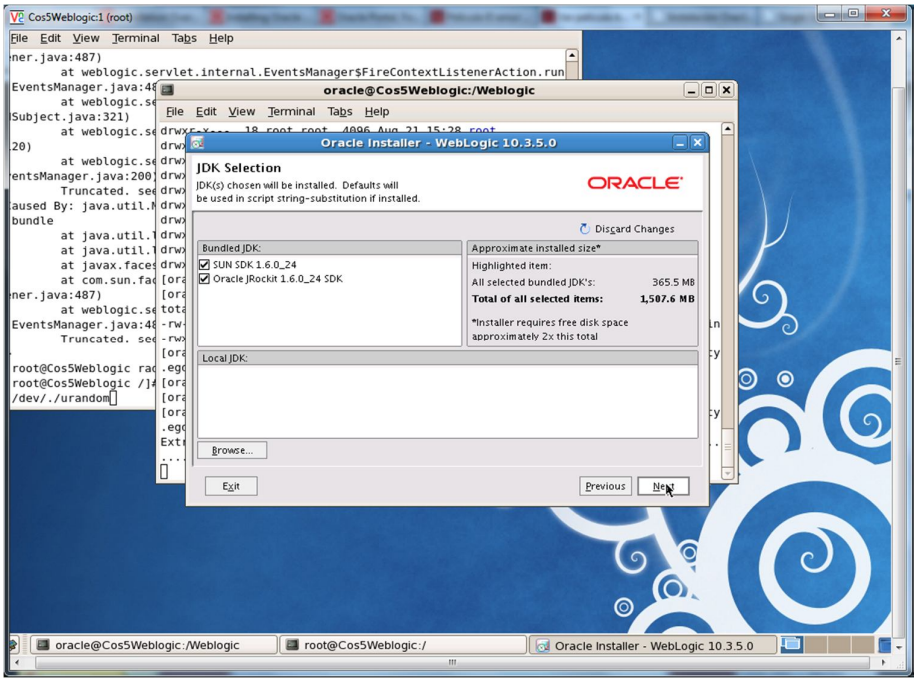


Figura 4-2: [Imagen instalación weblogic]. Obtenido desde: Wizard de instalación del software.

Para cada una de las siguientes pantallas aceptaremos las opciones por defecto y daremos clic en el botón etiquetado con la palabra “siguiente” hasta terminar.

En vista de que es necesario presentar algunos reportes, que son llamados desde *APEX*, tenemos que instalar el software *reports* en el servidor tomando en consideración los pasos listados en las siguientes páginas.

Para instalar el software es necesario que descarguemos los binarios y ejecutemos el archivo de instalación runInstaller.

- ./ runInstaller

En la primera ventana se debe especificar el directorio de instalación y el grupo del sistema operativo del cual se va a hacer uso, después de lo cual se solicita ejecutar algunos scripts como el usuario root. Una de las pantallas más importantes del wizard para realizar la instalación es la selección del tipo de instalación, es obligatorio que la opción *Install-Do not configure* sea la adoptada.

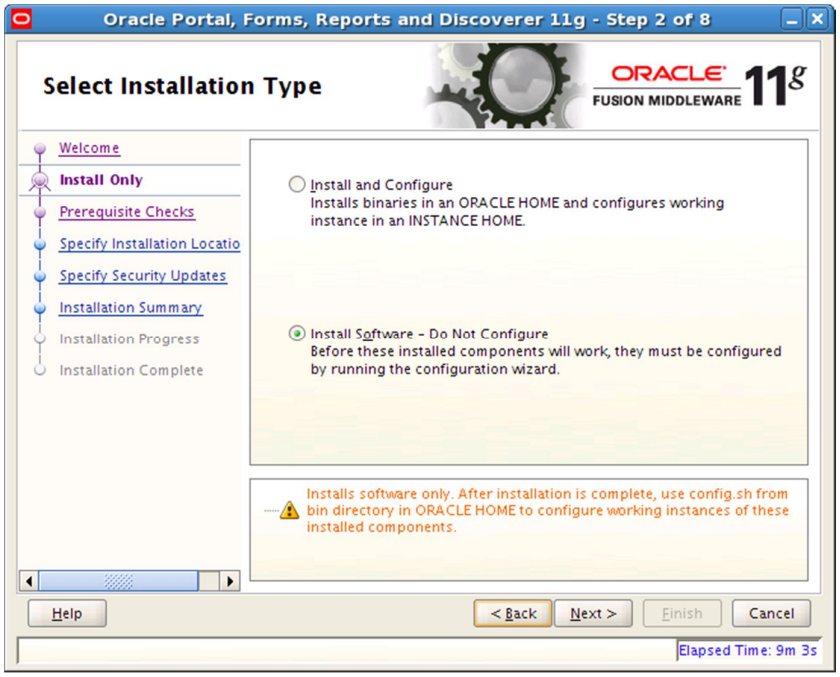


Figura 4-3: [Imagen instalación weblogic]. Obtenido desde: Wizard de instalación del software.

Pulsamos el botón de siguiente y el instalador procede con la validación de los paquetes necesarios del sistema operativo. En caso de que no se disponga de alguno se tiene que instalar manualmente. Para resolver el problema de valores del kernel se puede editar el archivo */etc./security/limits.conf* y agregar:

oracle hard nofile 4096
oracle soft nofile 4096

Si se ha solucionado los problemas relacionados con los paquetes instalados y con la configuración del *Kernel* del sistema operativo entonces el *wizard* nos preguntará por la dirección donde se encuentra instalado el servidor de aplicaciones porque se necesita de algunos binarios que este dispone para que el software de Forms y Reports funcione.

Figura 4-4: [Imagen instalación weblogic]. Obtenido desde: Wizard de instalación del software.

Si hemos realizado todos los pasos correctamente entonces el software para forms y reports se instalara sin problemas. Terminada la instalación del software como tal, es necesario que se configure definiendo algunos parámetros como el nombre del dominio, su usuario y clave, etc.

4.1.2.2 Configuración de Forms y Reports.

Para iniciar la configuración es necesario se ejecute el siguiente script que se encuentra en el home de la instalación de Weblogic.

- sh /$ORACLE_HOME/bin/config.sh

Entre los primeros datos que el wizard consulta son el nombre, usuario y clave del dominio que necesita Weblogic. Para nuestro caso en particular hemos usado las siguientes credenciales.

- User Name: Radius_Domain
- User Password: logic_weblogic
- Domain Name: rad_domain

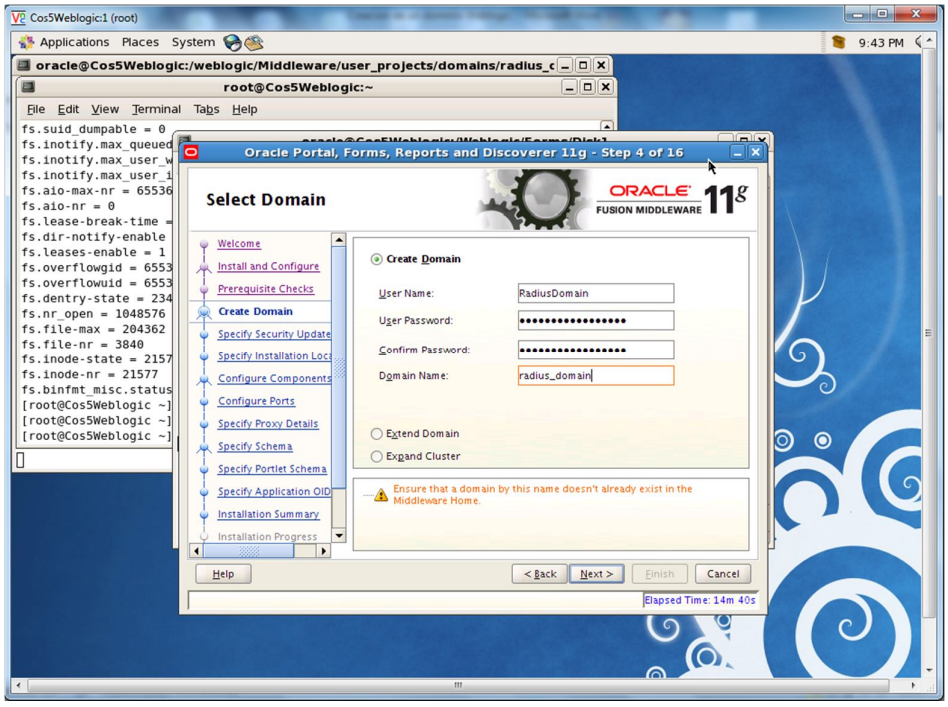


Figura 4-5: [Imagen instalación weblogic]. Obtenido desde: Wizard de instalación del software.

En las siguientes pantallas el sistema nos consulta sobre cuáles son los paquetes que deseamos instalar, para nuestro caso solo seleccionamos Forms y Reports, Oracle portal es una alternativa que nos permite la generación de Intranets pero en nuestro caso no haremos uso de esta herramienta. Para que Forms y Reports funcione correctamente es necesario que el servicio Oracle HTTP Server esté instalado y corriendo. Por ultimo definimos los puestos de escucha para forms y reports los mismos serán utilizados en los programas generados en *APEX*. Las siguientes pantallas ilustran el proceso antes descrito.

4.1.2.2.1 Selección directorios en los que se encuentra instalado el servidor Weblogic y los binarios de Forms y Reports.

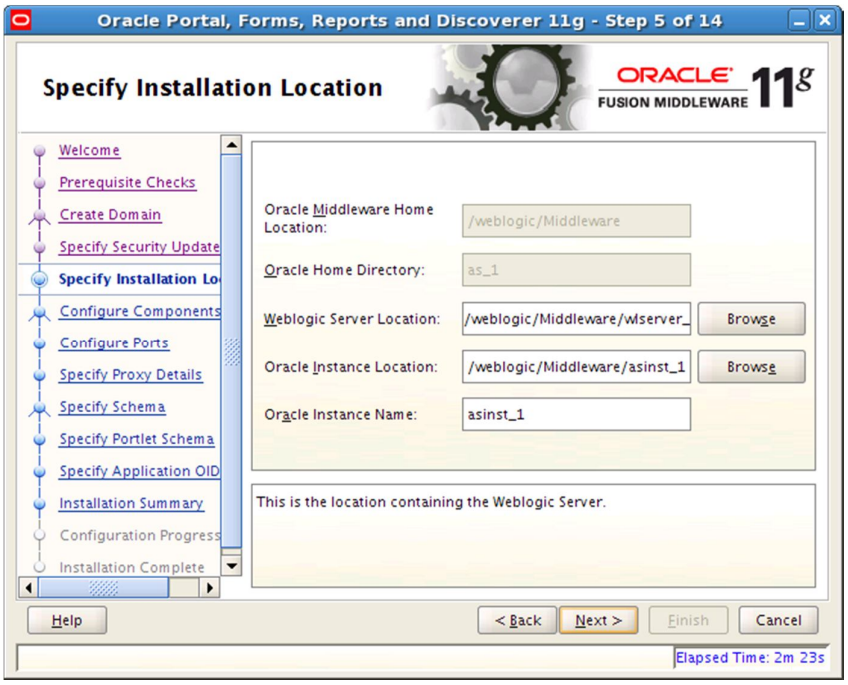


Figura 4-6: [Imagen instalación weblogic]. Obtenido desde: Wizard de instalación del software.

4.1.2.2.2 Selección de los paquetes a instalar.

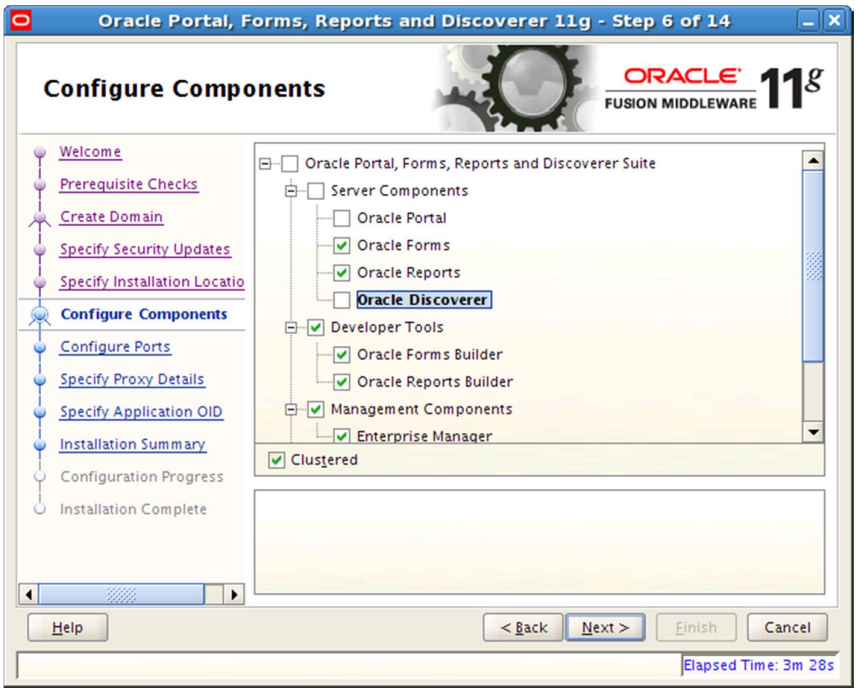


Figura 4-7: [Imagen instalación weblogic]. Obtenido desde: Wizard de instalación del software.

4.1.2.2.3 Definición del puerto de escucha.

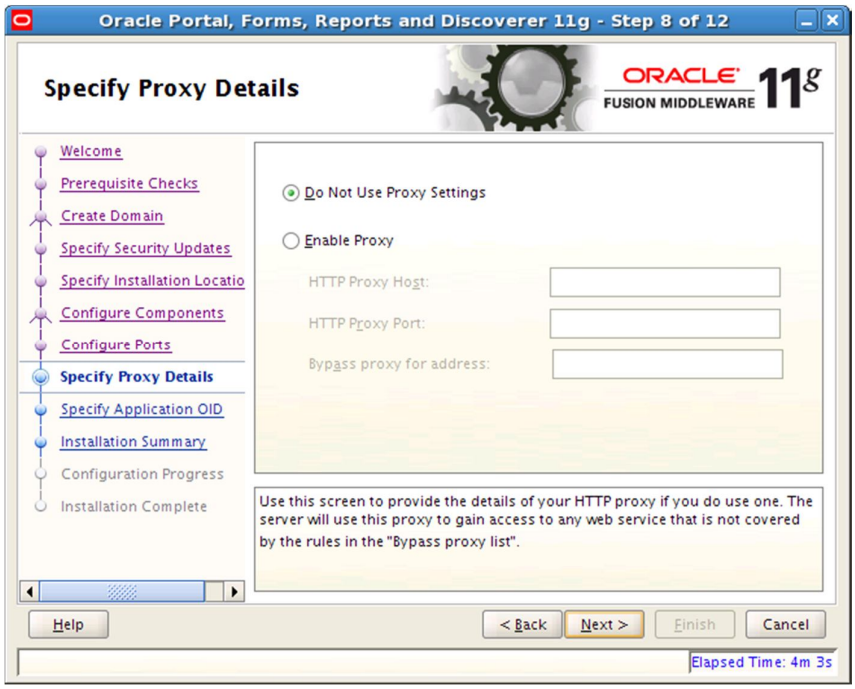


Figura 4-8: [Imagen instalación weblogic]. Obtenido desde: Wizard de instalación del software.

4.1.2.2.4 En la siguiente pantalla nos pide que especifiquemos un identificado de aplicación(OID).

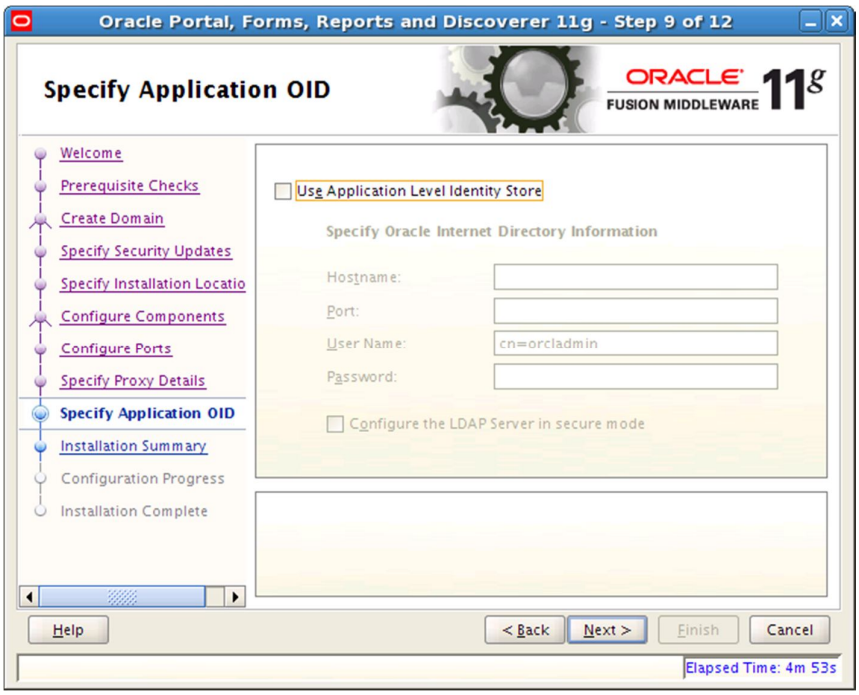


Figura 4-9: [Imagen instalación weblogic]. Obtenido desde: Wizard de instalación del software.

Después de que hemos establecido el OID solamente tendremos que revisar el resumen de las configuraciones y herramientas que se procederán a instalar, aceptamos y finalizamos con la instalación. Si se produce un error en la configuración del software Forms y Reports, entonces deberemos cancelar esta operación de configuración y ejecutar un proceso de actualización de los binarios instalados para el software Forms y Reports; después de que hemos actualizado los binarios entonces procedemos nuevamente con la configuración.

Si hemos finalizado correctamente la configuración entonces el resultado de la aplicación es:

Type: Oracle Portal, Forms, Reports and Discoverer Installation

Configuration Options

Middleware Home Location: /weblogic/Middleware

Oracle Home Location: /weblogic/Middleware/as_1

Oracle Instance Location: /weblogic/Middleware/asinst_1

Oracle Instance: asinst_1

Domain Option: Create Domain

Domain Name: RadiusDomain

Domain Home: /weblogic/Middleware/user_projects/domains/RadiusDomain

Domain Host Name: Cos5Weblogic

Domain Port No: 7001

User Name: RadiusDomain

Automatic Port Detection: true

Administrator Console: http://Cos5Weblogic:7001/console

EM Console: http://Cos5Weblogic:7001/em

EMAgent URL: http://Cos5Weblogic:5155/emd/main

Forms URL: http://Cos5Weblogic:8090/forms/frmservlet

Reports URL: http://Cos5Weblogic:8090/reports/rwservlet

Las primeras y más importantes instrucciones que debemos aprender son aquellas que permiten iniciar y parar el servidor de aplicaciones, el servicio de Reports y el servicio de Oracle Http.

4.1.2.3 Instrucción para iniciar el servicio de weblogic

sh

/weblogic/Middleware/wlserver_10.3/server/bin/startNodeManager.sh

Es muy probable que se nos solicite un password, el dato que debemos ingresar corresponde al que digitamos al momento de la instalación que para nuestro caso es: logic_weblogic

4.2.1.4 Instrucción para parar el servicio de Weblogic

Si hemos iniciado el servicio de weblogic desde un consola sin ejecutarlo en background entonces solo bastará con presionar las teclas: Ctrl+c, en cambio si hemos iniciado el proceso en background entonces debemos primero buscar el número del proceso con el comando:

ps –ef | grep Weblogic

El comando anterior retorna el número de proceso y para "matar" ese proceso debemos ejecutar:

kill #proceso

4.2.1.5 Instrucciones para iniciar y parar el servicio de Reports y Forms

Al igual que para poder arrancar el servicio de weblogic es necesario ejecutar un Script, para poder arrancar el servicio de reports y forms también requerimos de un script. Estos scripts ejecutan algunas instrucciones entre las cuales se encuentran inicialización de variables de ambiente, comprobación del usuario que arranca el servicio, verificación de nombre de equipo, etc.

4.2.1.6 Instrucción para iniciar Reports y Forms

Para iniciar el proceso de *Forms* se debe ejecutar la instrucción descrita en la siguiente línea.

sh

/weblogic/Middleware/user_projects/domains/radius_domain/bin/startManagedWeblogic.sh WLS_FORMS "t3://localhost:7001/"

Nota: En lugar de usar localhost se podría usar la dirección IP de la máquina que tiene instalado el servicio de Weblogic. Para iniciar el servicio de *Reports* se ejecuta:

sh
/weblogic/Middleware/user_projects/domains/radius_domain/bin/startManagedWebl
ogic.sh WLS_REPORTS “t3://localhost:7001/”

Para que el servicio de Forms y Reports funcione adecuadamente es necesario que se inicie el servicio *http(Hypertext Transfer Protocol)* de Oracle y se instancie las variables de ambiente necesarias. Para lograr este objetivo tenemos que localizar y ejecutar los siguientes scripts como usuario Oracle.

sh
/weblogic/Middleware/user_projects/domains/rad_domain/bin/setDomainEnv.sh
sh /weblogic/Midleware/asinst_1/bin/opmnctl startall

Si todos los pasos antes listados se realizaron correctamente entonces deberíamos llegar hasta este punto sin problemas. Ahora ya tenemos listo el servidor de aplicaciones, sin embargo no podemos ingresar a *APEX* porque no se ha realizado el despliegue del *listener*. Antes de ahondar en cómo se realiza la instalación del listener es necesario que se entienda que es y cómo se integra con weblogic esta herramienta.

4.1.3 APEX Listener

Cuando decimos “Servidor de Aplicaciones” estamos hablando de un servidor que es capaz de correr aplicaciones desarrolladas haciendo uso del estándar *J2EE*(Java 2 Platform Enterprise Edition) de modo que ya no es necesario hacer uso del servicio Oracle HTTP Server (OHS) y *mod_plsql* que existen generalmente en un servidor *OAS*(Oracle Application Server).

La disponibilidad de *JDBC(Java Database Connectivity)* integrado con *APEX Listener* ocasiona que la configuración para acceder a la base de datos sea más simple debido principalmente a que ya no es necesario el directorio *ORACLE HOME.*

De acuerdo a la documentación proporcionada por Oracle existen dos metodologías para realizar la instalación de esta aplicación, la estándar que hemos adoptado y que

demostraremos el procedimiento de cómo realizarlo, y la forma avanzada que proporciona un nivel mayor de seguridad pero que para esta tesis no se la ha visto necesaria puesto que el sistema no estará expuesto al internet. Es conveniente sin embargo que se explique brevemente en qué consisten estas dos alternativas de instalación, algún día nos podría servir.

En la configuración estándar el *APEX* listener es instalado considerando que las peticiones no serán receptadas desde fuera del firewall y los clientes acceden directamente al servicio que en su totalidad reside en el servidor de aplicaciones. Lo enunciado en este párrafo se ilustra en el grafico inferior.

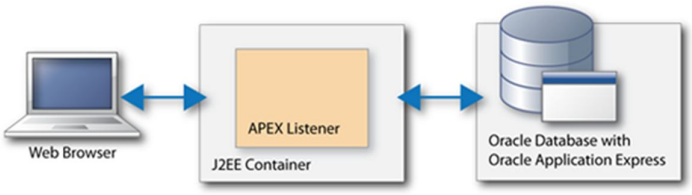


Figura 4-10: [Arquitectura Apex Listener]. Obtenido desde: http://www.oracle.com/technetwork/developer-tools/apex-listener/overview/index.html

En una configuración avanzada lo que se pretende es habilitar una opción con la que se pueda receptar las peticiones tanto desde fuera del firewall como desde adentro, para conseguirlo el *APEX* listener es instalado en el contenedor J2EE dentro del firewall y un *listener* de peticiones HTTPP es instalado fuera del firewall. De este modo todas las peticiones de los usuarios se dirigen al listener HTTP y este redirecciona las peticiones al *APEX* listener dentro de la Red.

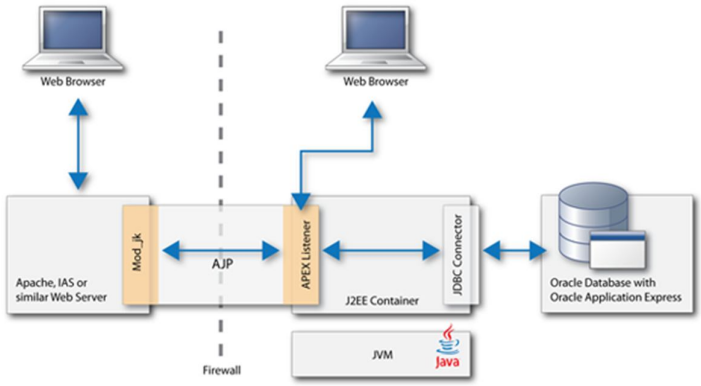


Figura 4-11: [Arquitectura Apex Listener]. Obtenido desde: http://www.oracle.com/technetwork/developer-tools/apex-listener/overview/index.html

4.1.3.1 Instalación *APEX listener*

La instalación de una aplicación JAVA sobre un servidor de aplicaciones se traduce en desplegar una aplicación en el servidor, sin embargo en el caso del *APEX* listener es necesario se configuren parámetros relacionados con los roles administrativos y de control. Los pasos que hemos desarrollado tomando como referencia la página de Oracle se puede resumir con los pasos descritos en las siguientes páginas.

Para instalar *APEX* listener es necesario tener activo el servidor weblogic con un dominio creado, para esta tesis es rad_domain, y disponer de las credenciales que permitan acceder a la consola de Weblogic para realizar las configuraciones necesarias. Aunque las credenciales requeridas las podemos hallar en la sección de instalación del servidor de aplicaciones las cito nuevamente.

Username: Radius_Domain
Password: logic_weblogic.

Aunque *APEX* está instalado en la base de datos y se puede extraer tanto la lógica como las imágenes, se recomienda que las imágenes no sean obtenidas a partir del *RDBMS* porque afecta el rendimiento de la aplicación. Para solventar este problema ubicaremos las imágenes en el servidor de aplicaciones. La fuentes de *APEX* no traen incluido un archivo *.war* que contenga las imágenes, en su lugar debemos generarlo, pero para poder realizar esta tarea necesitamos del comando *jar*.

El comando *jar* está incluido en el paquete de desarrollo de java, y para disponer de él es necesario ejecutar las siguientes instrucciones que instalan los JDK necesarios.

```
yum install java-1.6.0-openjdk
yum install java-1.6.0-openjdk-devel
```

El sistema operativo también requiere de algunas librerías adicionales, al igual que en el caso anterior deberemos ejecutar el comando yum.

```
yum install giflib
yum install jpackage-utils
```

yum install tzdata-java

Para crear el archivo *.war* hacemos uso de la instrucción inferior:

jar –cvf0 <temp directory>/i.war –C <apex directory>/images .

Donde <temp directory> es el directorio temporal donde se localizará el archivo .war que contendrá las imágenes, y <apex directory> es el directorio que contiene los binarios de *APEX*. Para iniciar la instalación seleccionamos la opción “despliegues” en la sección izquierda, se visualiza en el lado derecho de la pantalla el listado de aplicaciones desplegadas, allí existe un menú, distribuido horizontalmente en pantalla, que permite la instalación de nuevas aplicaciones, seleccionamos la opción de instalar.

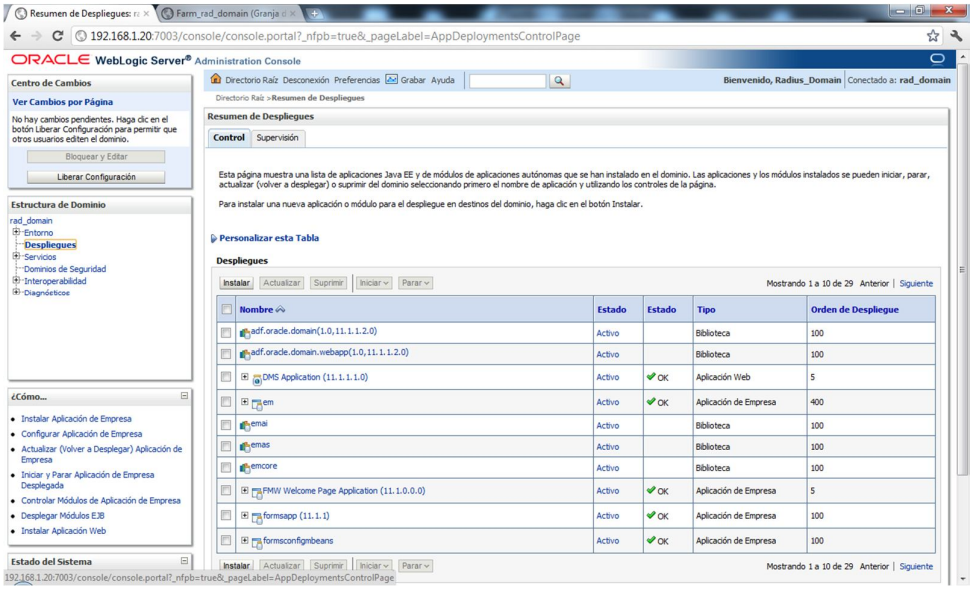


Figura 4-12: [Pantalla administración weblogic]. Obtenido desde: Consola administración servidor weblogic

Una vez hayamos hecho clic en el botón de instalar, la consola de Weblogic solicita la ruta donde se encuentra el archivo apex.war, podemos digitar la ruta o usar el botón de selección proporcionado para buscar el archivo deseado. El momento que tengamos la ruta correcta damos clic en el botón etiquetado con la palabra siguiente y continuamos.

Figura 4-13: [Pantalla administración weblogic]. Obtenido desde: Consola administración servidor weblogic

Cuando hayamos dado clic en “siguiente” la consola de Weblogic pregunta qué tipo de instalación se va a realizar, en vista de que el archivo seleccionado corresponde a una aplicación java entonces seleccionaremos “Instalar despliegue como aplicación” y damos nuevamente clic en el botón etiquetado como “Siguiente”.

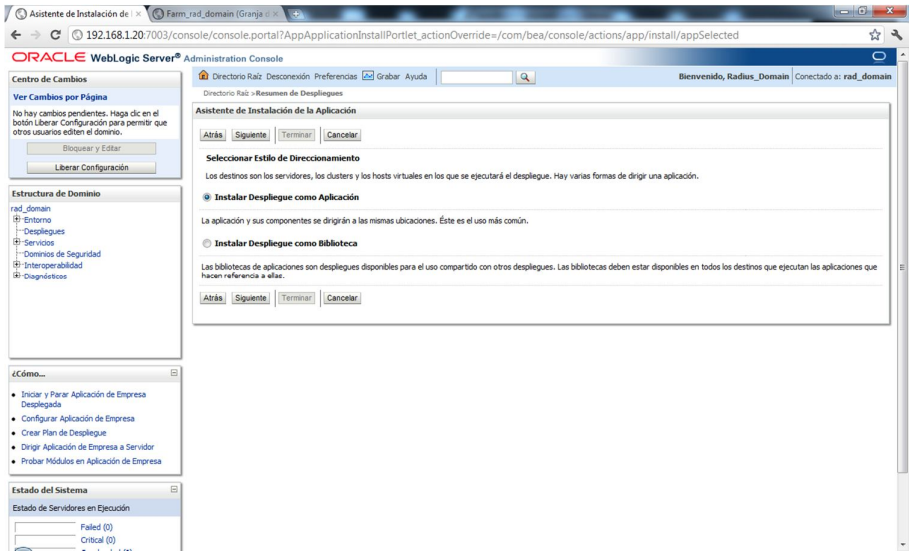


Figura 4-14: [Pantalla administración weblogic]. Obtenido desde: Consola administración servidor weblogic

El servidor *Weblogic* dispone de algunas instancias donde podemos desplegar la aplicación entre ellas constan: *AdminServer*, *WLS_FORMS* y *WLS_REPORTS*, sin embargo la que nos interesa es **AdminServer,** la seleccionamos y damos clic en siguiente.

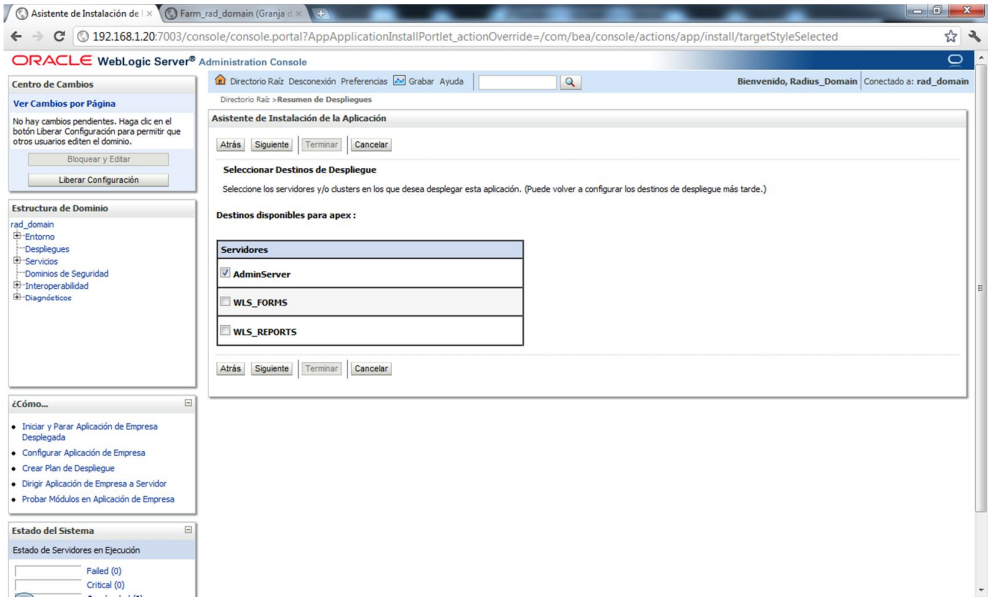


Figura 4-15: [Pantalla administración weblogic]. Obtenido desde: Consola administración servidor weblogic

Después de seleccionado el destino es necesario definir el nombre del despliegue que en nuestro caso es apex, también es necesario establecer las políticas de seguridad, roles personalizados y por último en accesibilidad de origen seleccionamos: "*Usar valores por defecto definidos por destinos de despliegue*".

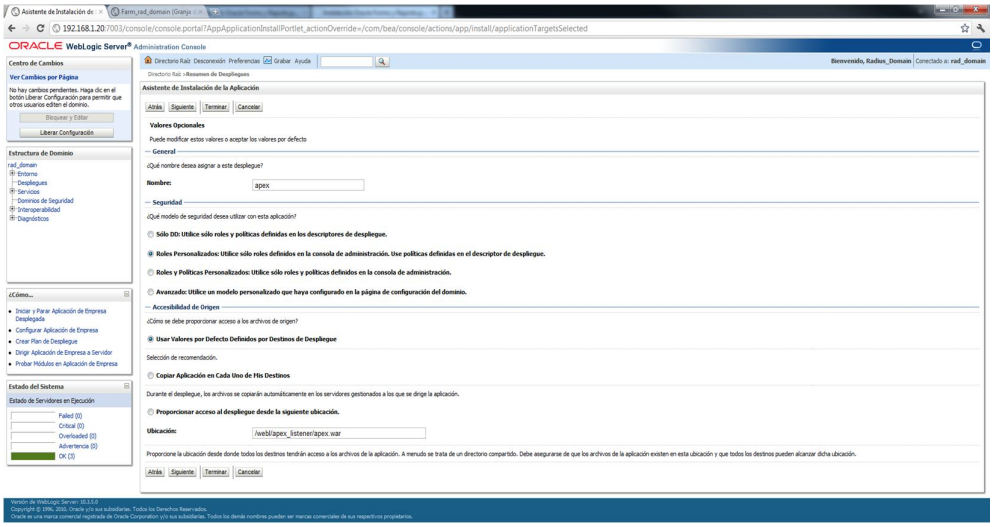


Figura 4-16: [Pantalla administración weblogic]. Obtenido desde: Consola administración servidor weblogic

Antes de finalizar tenemos un cuadro resumen que nos muestra las acciones a realizar, en el mismo seleccionamos la opción de que no deseo revisar la configuración sino que lo realizaré más tarde.

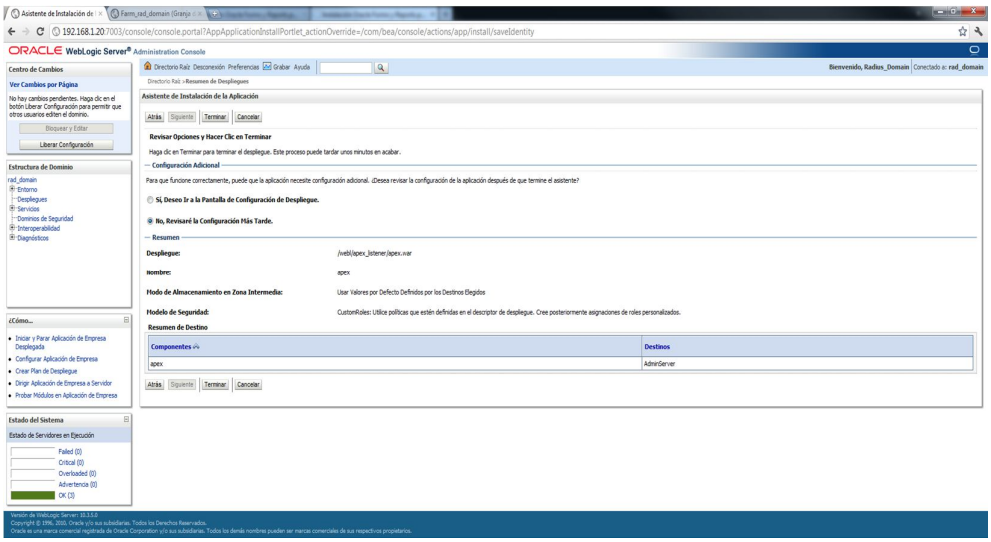


Figura 4-17: [Pantalla administración weblogic]. Obtenido desde: Consola administración servidor weblogic

A continuación damos clic en terminar y la instalación da lugar. Para instalar las imágenes seguiremos los pasos anteriores pero con unas pequeñas modificaciones, veamos:

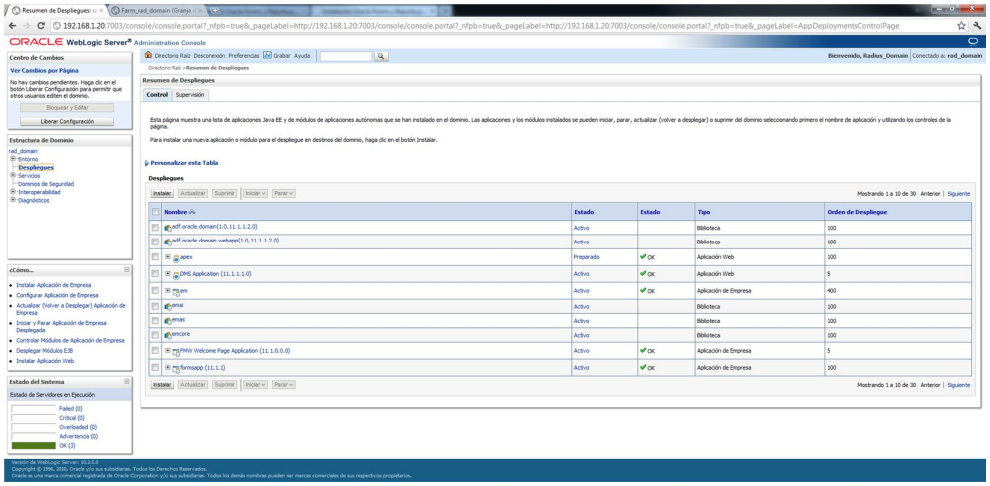


Figura 4-18: [Pantalla administración weblogic]. Obtenido desde: Consola administración servidor weblogic

Igual que el procedimiento para instalar apex, damos clic en instalar y en la ventana siguiente seleccionamos el archivo i.war

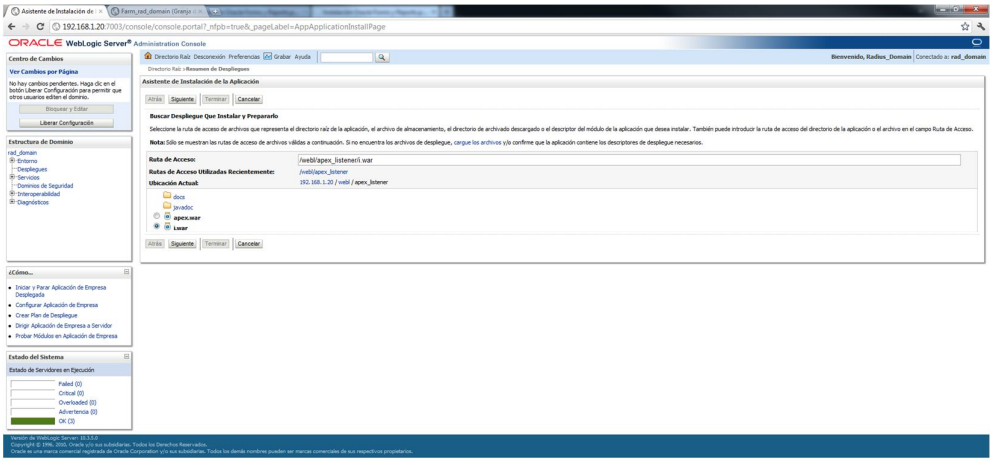


Figura 4-19: [Pantalla administración weblogic]. Obtenido desde: Consola administración servidor weblogic

En las pantallas que corresponden al tipo de instalación e instancia del servidor se deben seleccionar exactamente las mismas opciones que seleccionamos cuando se instaló la aplicación “apex”, es decir, se deberían seleccionar: Despliegue como aplicación e instalación en la instancia AdminServer.

En la siguiente pantalla se tiene que hacer un cambio en las opciones que serán seleccionadas, es decir, en lugar de seleccionar “Opciones personalizadas” utilizaremos la opción: “Solo roles y políticas definidas en los descriptores de despliegue”.

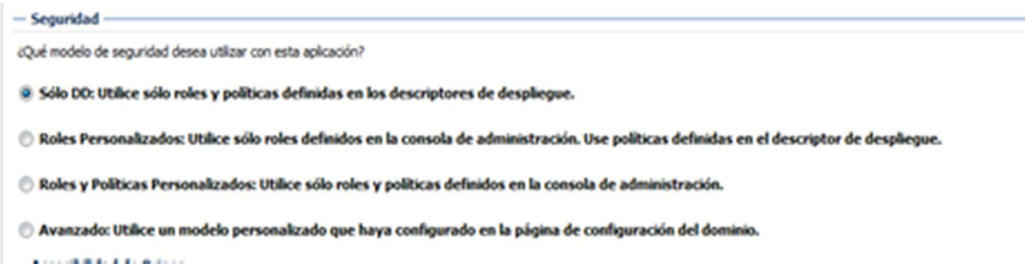


Figura 4-20: [Pantalla administración weblogic]. Obtenido desde: Consola administración servidor weblogic

En las pantallas siguientes, al igual que el procedimiento adoptado para la instalación de la aplicación “apex”, le “decimos” a la interfaz web de *Weblogic* que no deseamos revisar la configuración para proceder con la instalación del archivo i.war. Guardamos los cambios dando clic en la parte superior izquierda en el botón “Aplicar Cambios”. Para verificar que las aplicaciones se han instalado con éxito,

seleccionamos la opción despliegues y verificamos que las aplicaciones instaladas aparezcan con un estado OK en el listado derecho.

Para iniciar los servicios instalados, seleccionamos con el botón izquierdo las aplicaciones deseadas y luego nos dirigimos al botón iniciar localizado en la parte inferior de la pantalla.

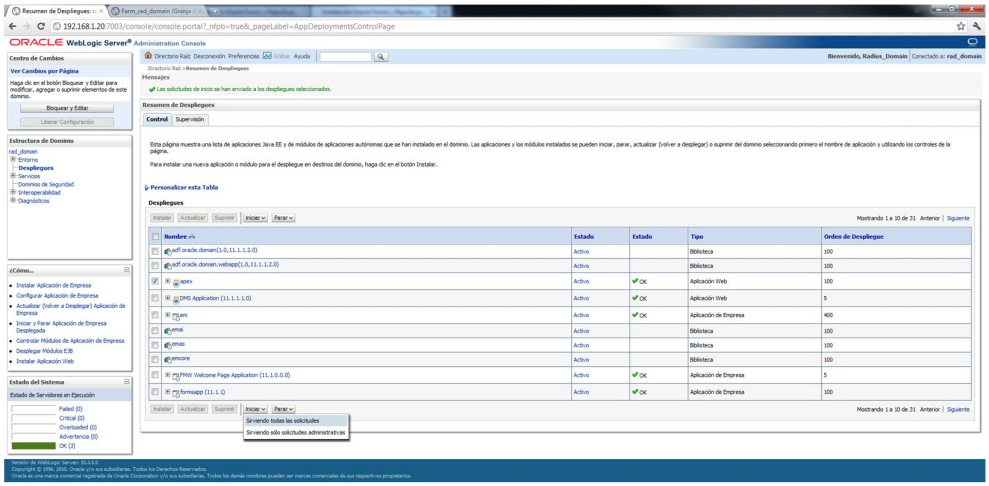


Figura 4-21: [Pantalla administración weblogic]. Obtenido desde: Consola administración servidor weblogic

Para administrar el *APEX listener*, administración que incluye la definición de parámetros como la dirección IP y SID de la base de datos, weblogic proporciona mecanismos de seguridad que son generados a partir de la creación de usuarios y roles. De acuerdo a la documentación de Oracle se han creado dos roles llamados: *Manager y Administrador.* La siguiente sección describe como crear y configurar los usuarios y roles de administración del *APEX listener.*

4.1.3.2 Creación Usuarios y Roles

Así como el servidor de aplicaciones Tomcat dispone de mecanismos de seguridad administrados a través del archivo "*tomcat_users.xml*" ubicado en el directorio de configuración del servidor, weblogic también dispone de alternativas que permiten definir quiénes y a través de que roles podrán realizar actividades de administración. Para el servidor de weblogic esta configuración se la realiza a través de la interfaz web.

En el menú izquierdo damos clic en Dominios de seguridad, en primera instancia crearemos los usuarios por lo que damos clic en myrealm:

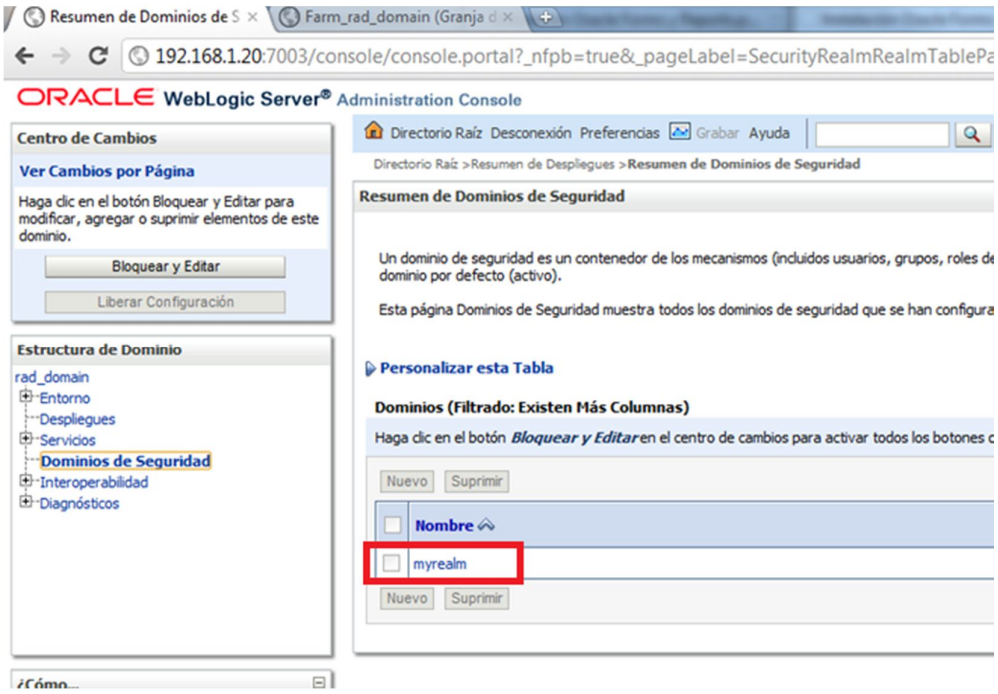


Figura 4-22: [Pantalla administración weblogic]. Obtenido desde: Consola administración servidor weblogic

Dentro de myrealm nos dirigimos a usuarios y grupos:

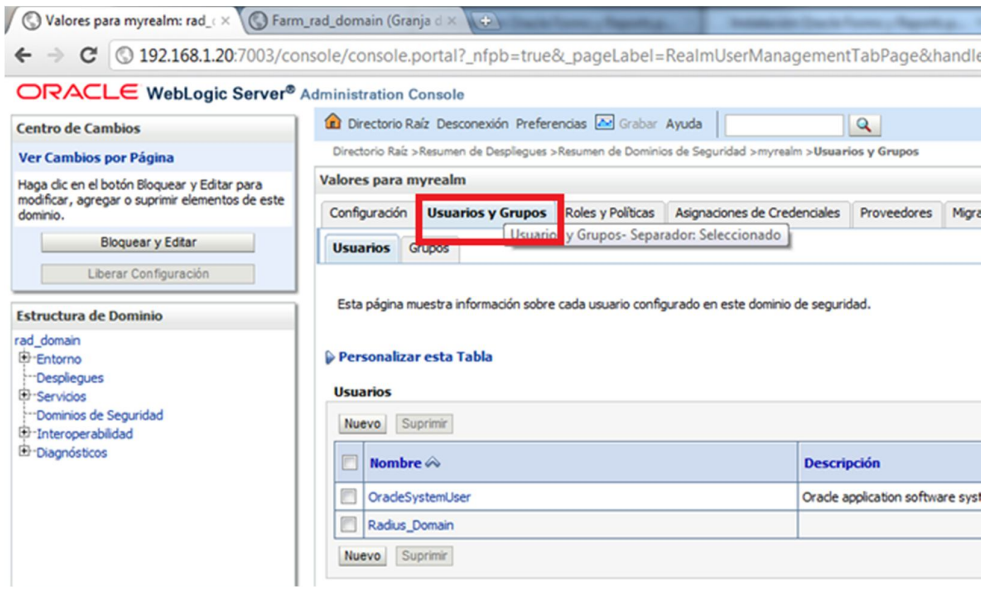


Figura 4-23: [Pantalla administración weblogic]. Obtenido desde: Consola administración servidor weblogic

Damos clic en la pestaña usuarios y a continuación seleccionamos nuevo:

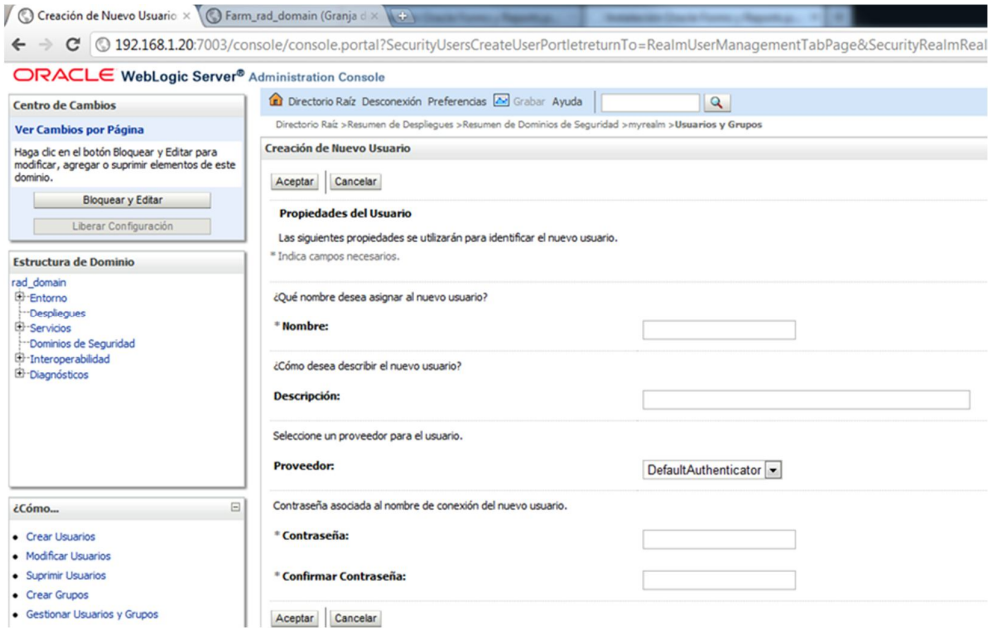


Figura 4-24: [Pantalla administración weblogic]. Obtenido desde: Consola administración servidor weblogic

En el nombre de usuario introduciremos: "**adminlistener**" que es el nombre de usuario que se encargará de configurar el *listener*, por su parte en la descripción, que es un campo informativo, digitaremos *"APEX Listener administrator user"*. En el tipo de autenticación seleccionamos: DefaultAutenticator y en el campo password digitamos: ***logic_apex_weblogic***

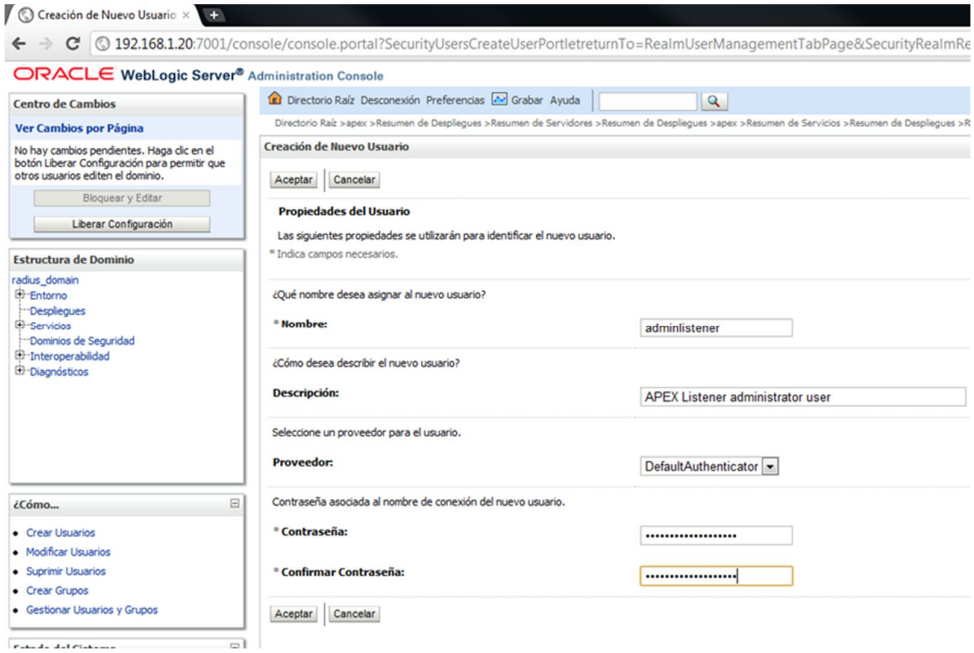


Figura 4-25: [Pantalla administración weblogic]. Obtenido desde: Consola administración servidor weblogic

Damos clic en OK y procedemos con la creación de otro usuario (manager), siguiendo los mismos pasos que para la creación del usuario administrador, que podrá acceder solamente para visualizar la página de estado del Listener Apex. La clave del mismo será: ***logic_apex_manager_weblogic*** y el usuario es ***managerlisterner***.

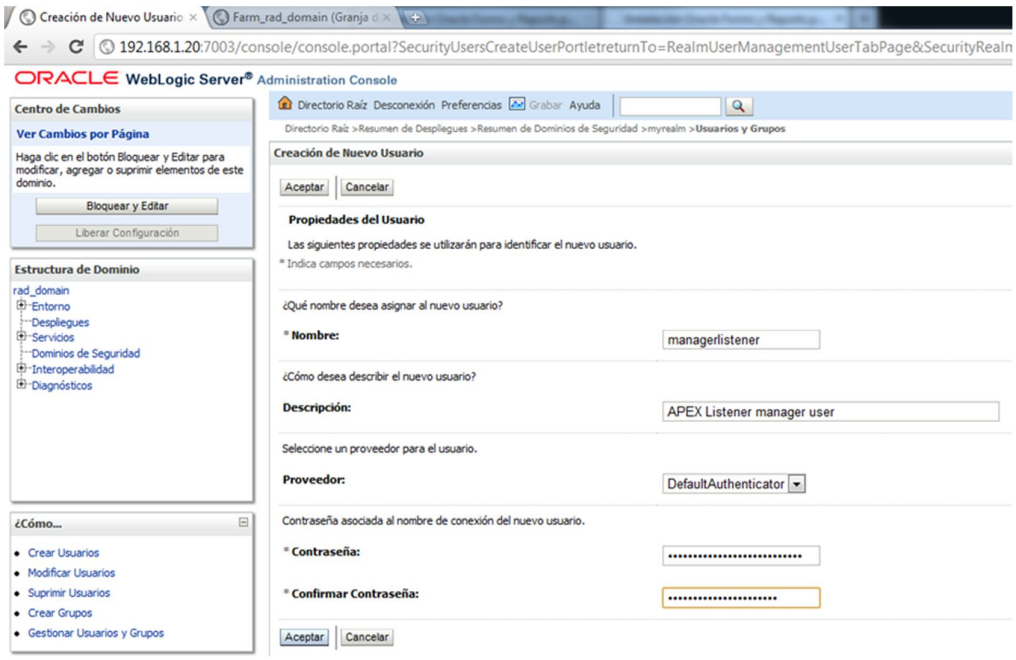


Figura 4-26: [Pantalla administración weblogic]. Obtenido desde: Consola administración servidor weblogic

Nos queda ahora por definir los roles que contienen los permisos que datemos a los usuarios administrador y manager. Al igual que la creación de usuarios es necesario ingresar a dominios de seguridad, seleccionar myrealm y posteriormente seleccionar la pestaña "*roles y políticas*" y dentro de ella seleccionamos roles de dominio. Después de dar clic en la opción "Roles de Dominio" aparece un listado de donde seleccionamos el símbolo (+) junto al texto "Despliegues" y del sub-listado resultante nos dirigimos al link "apex".

Cuando demos clic en el link "apex" se visualiza otra ventana con opciones que permiten la creación de roles, podemos intuir entonces que los roles son creados por aplicación y se asignan a uno o varios usuarios. Las siguientes pantallas ilustran lo descrito:

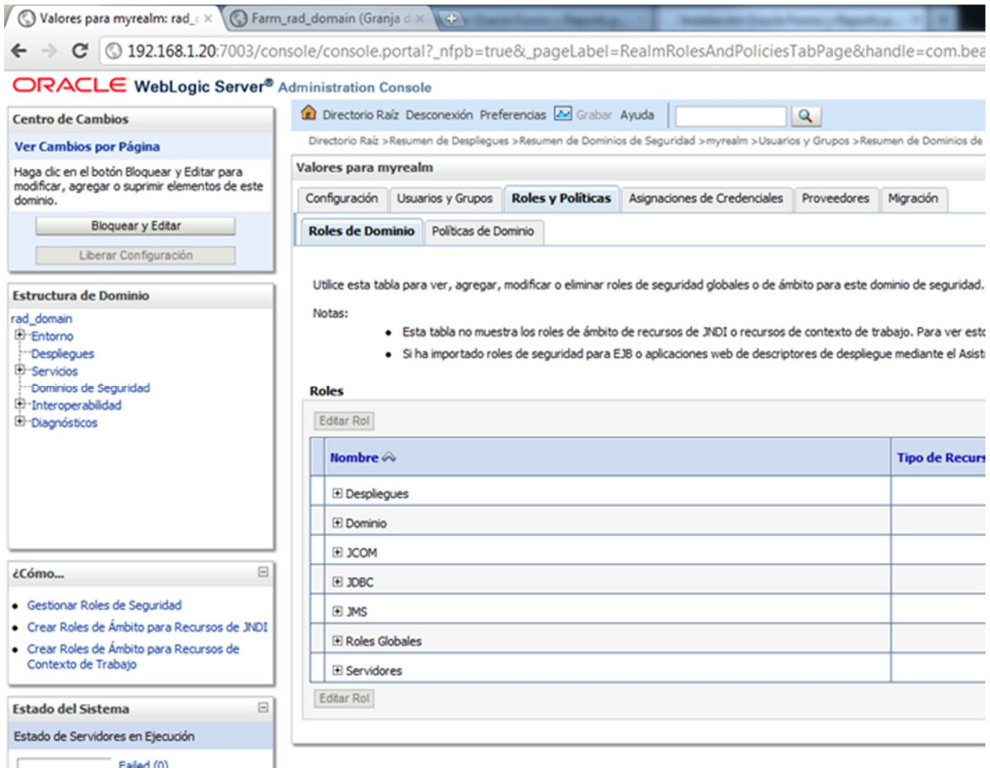


Figura 4-27: [Pantalla administración weblogic]. Obtenido desde: Consola administración servidor weblogic

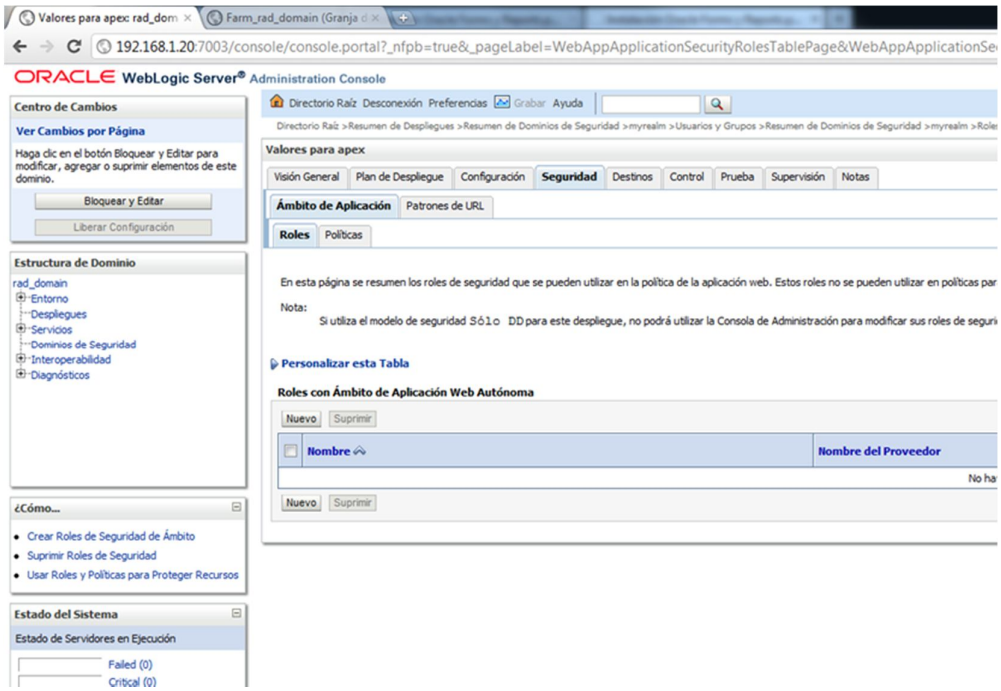


Figura 4-28: [Pantalla administración weblogic]. Obtenido desde: Consola administración servidor weblogic

En la pantalla que tiene como título: "Roles con Ámbito de Aplicación Web" Autónoma damos clic en nuevo:

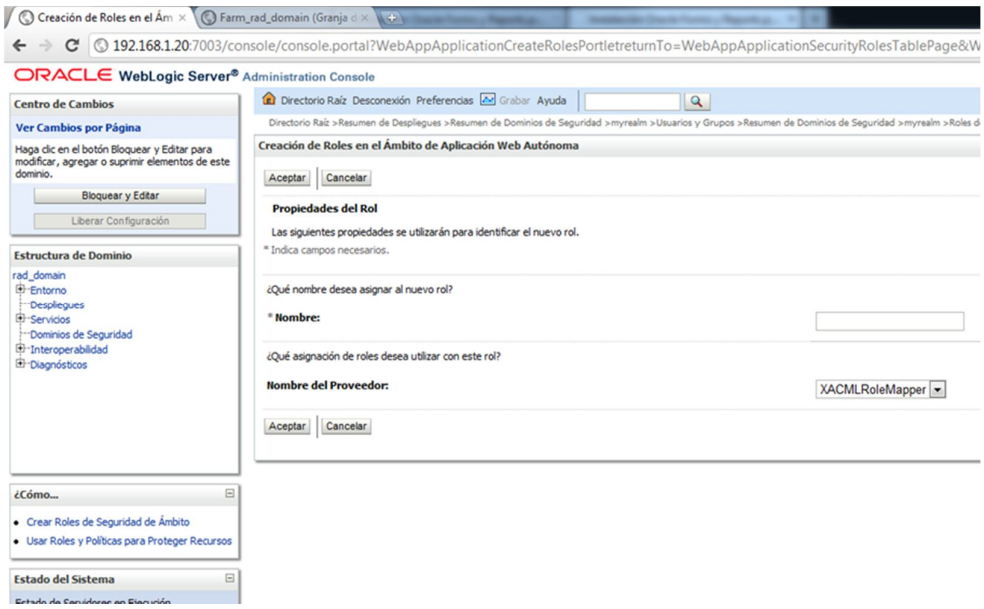


Figura 4-29: [Pantalla administración weblogic]. Obtenido desde: Consola administración servidor weblogic

En la pantalla que se visualiza establecemos los valores correspondientes a nombre igual a “Admin” y “Nombre del Proveedor” establecemos a XACMLRoleMapper similar a lo representado por la imagen inferior.

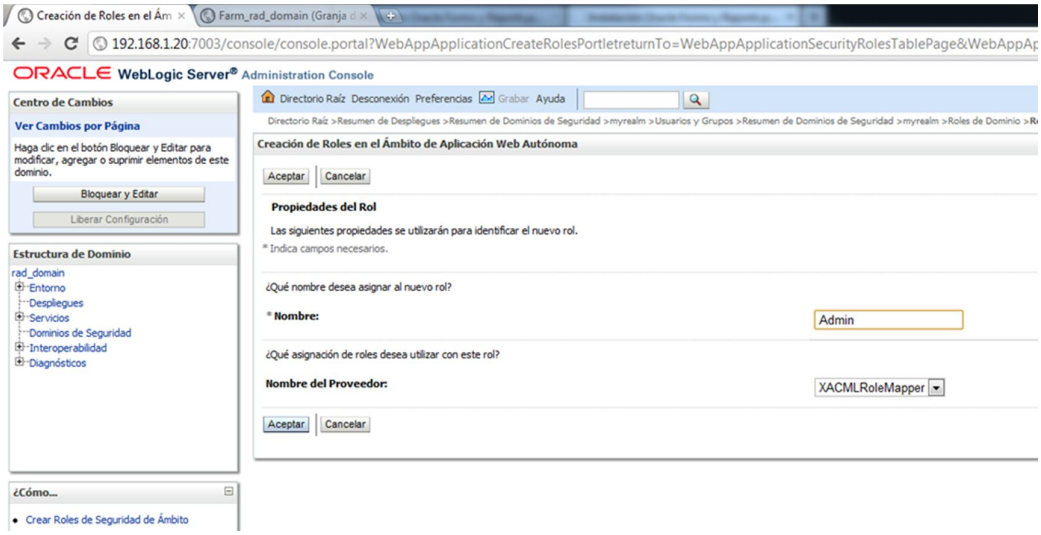


Figura 4-30: [Pantalla administración weblogic]. Obtenido desde: Consola administración servidor weblogic

Damos clic en aceptar y creamos a continuación una condición para el rol creado, esta condición nos permite asociar el usuario “administrador” con el rol Admin. Para

crear la condición seleccionamos el rol “Admin” y a continuación el botón con etiqueta “Agregar Condiciones” en la sección condiciones de rol.

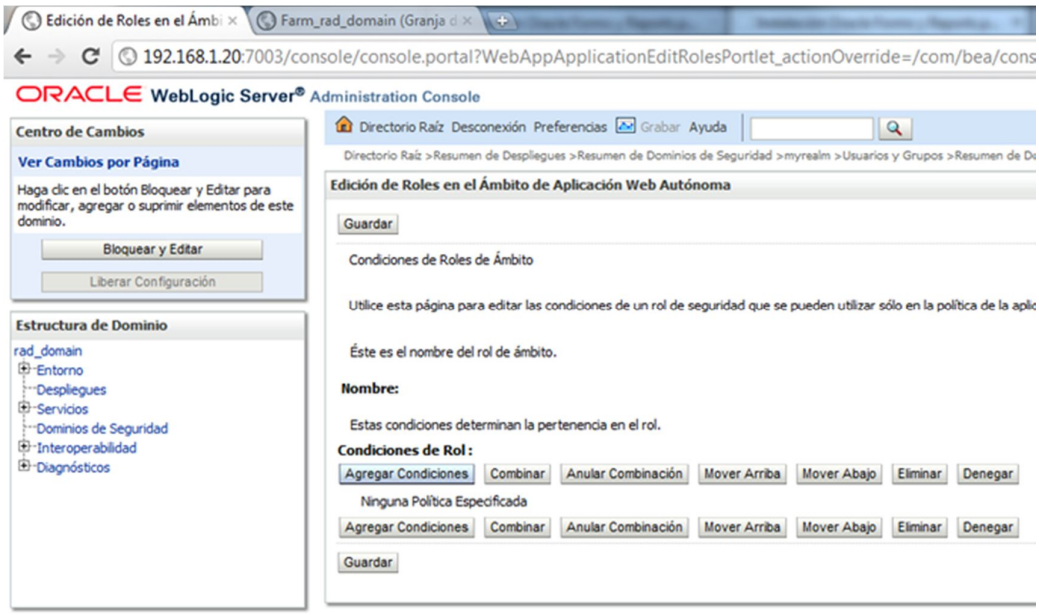


Figura 4-31: [Pantalla administración weblogic]. Obtenido desde: Consola administración servidor weblogic

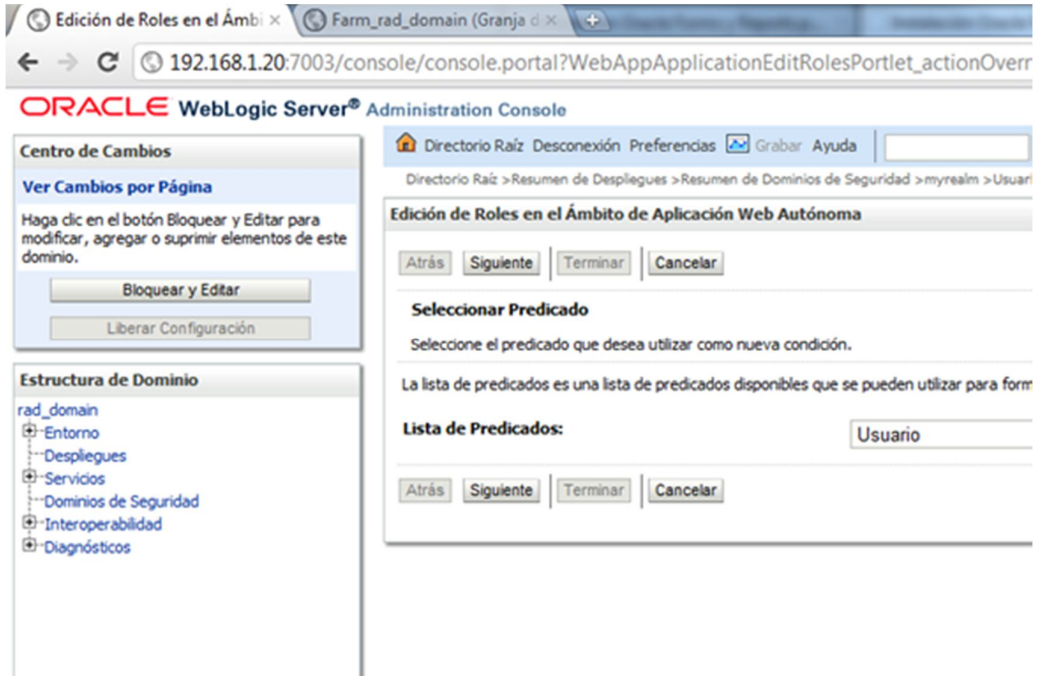


Figura 4-32: [Pantalla administración weblogic]. Obtenido desde: Consola administración servidor weblogic

Seleccionado del “select list” de la pantalla anterior la opción usuario damos clic en siguiente, y agregamos al usuario administrador creado en las operaciones anteriores.

Finalmente pulsamos el botón de terminar y en la siguiente pantalla guardar, con esto hemos generado un rol de administrador y también se ha asociado un usuario a ese tol.

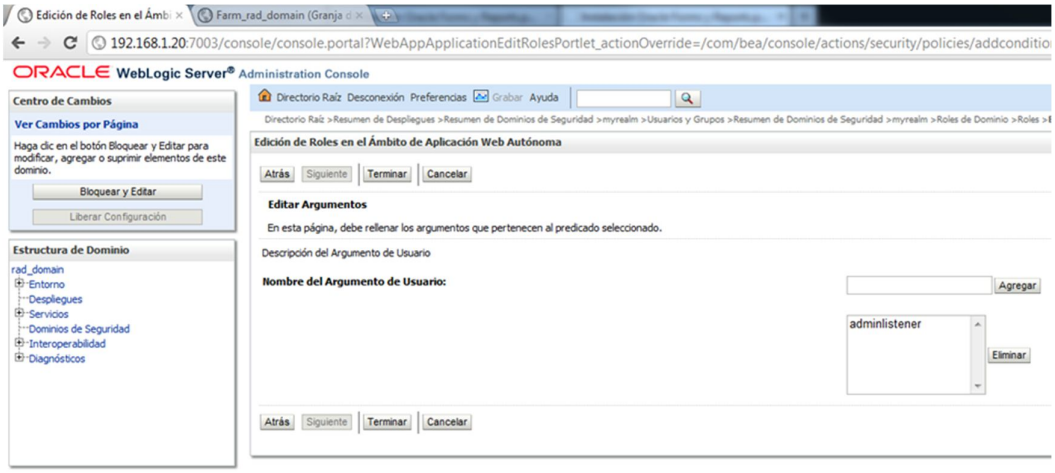


Figura 4-33: [Pantalla administración weblogic]. Obtenido desde: Consola administración servidor weblogic

Para el usuario "manager" también se tiene que crear un rol, como los pasos son similares a los descritos para la creación del rol administrador, solamente me remitiré a presentar los pantallazos necesarios para aclarar el proceso.

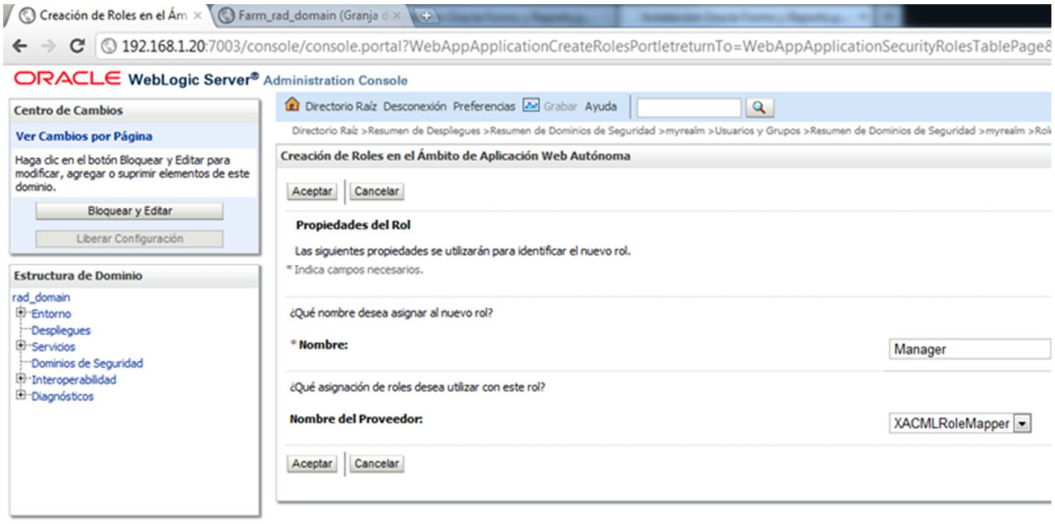


Figura 4-34: [Pantalla administración weblogic]. Obtenido desde: Consola administración servidor weblogic

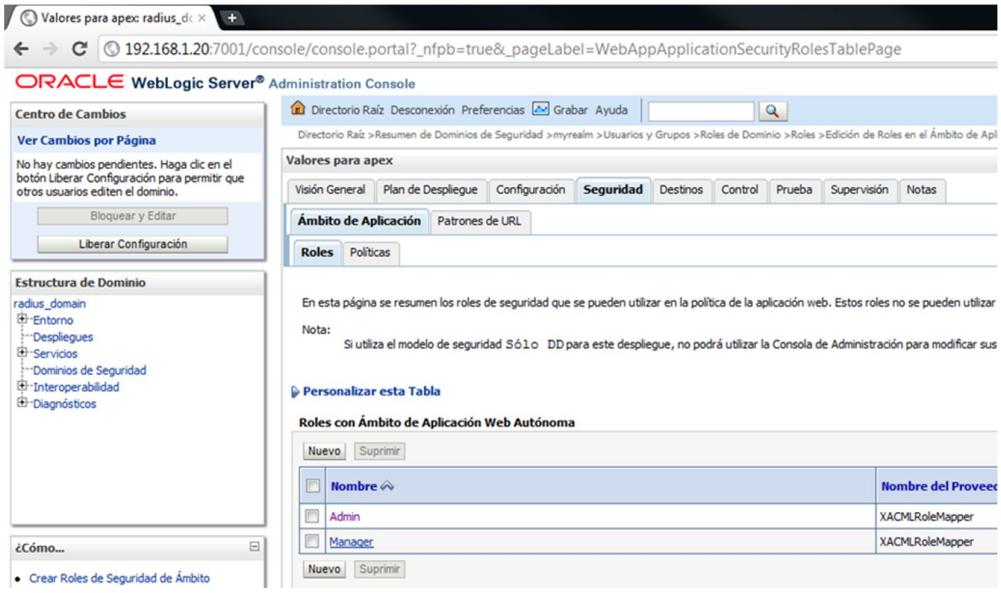


Figura 4-35: [Pantalla administración weblogic]. Obtenido desde: Consola administración servidor weblogic

Damos clic en Manager para editarlo y agregar una condición:

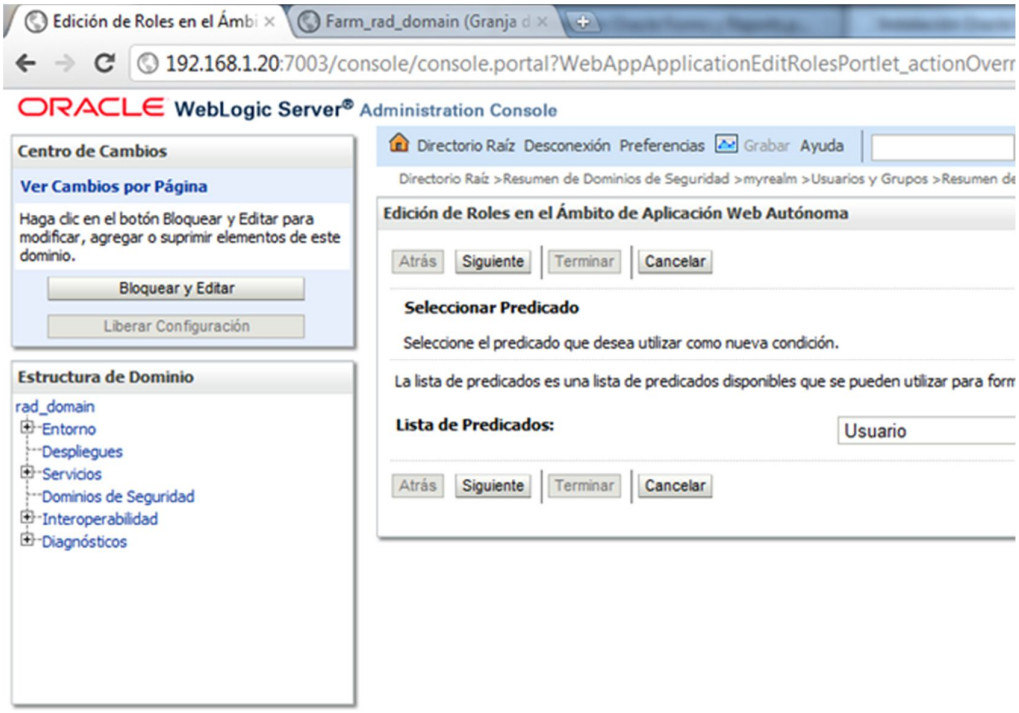


Figura 4-36: [Pantalla administración weblogic]. Obtenido desde: Consola administración servidor weblogic

Seleccionamos usuario y en la siguiente ventana digitamos y agregamos "managerlistener". Con esto hemos terminado con la creación de usuarios, roles y los permisos correspondientes a cada uno de ellos.

Ahora ingresaremos a la página de administración de Apex Listener para lo cual debemos digitar una ruta que tiene la sintaxis siguiente: http://<host>:<port>/apex/listenerConfigure y que para nuestro caso es: http://192.168.1.20:7001/apex/listenerConfigure. Cuando hayamos ingresado, el navegador visualiza la pantalla de configuración inferior.

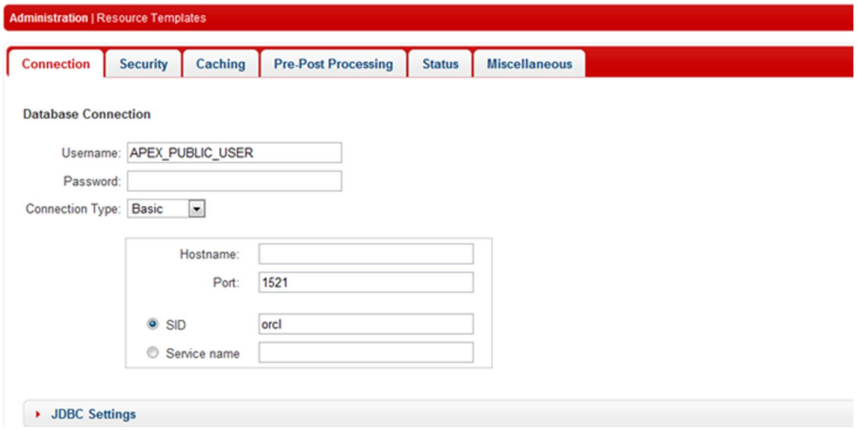


Figura 4-39: [Pantalla administración weblogic]. Obtenido desde: Consola administración servidor weblogic

La primera vez que se accede no solicita credenciales, sin embargo una vez configurado la próxima entrada sí solicita las credenciales de administrador.

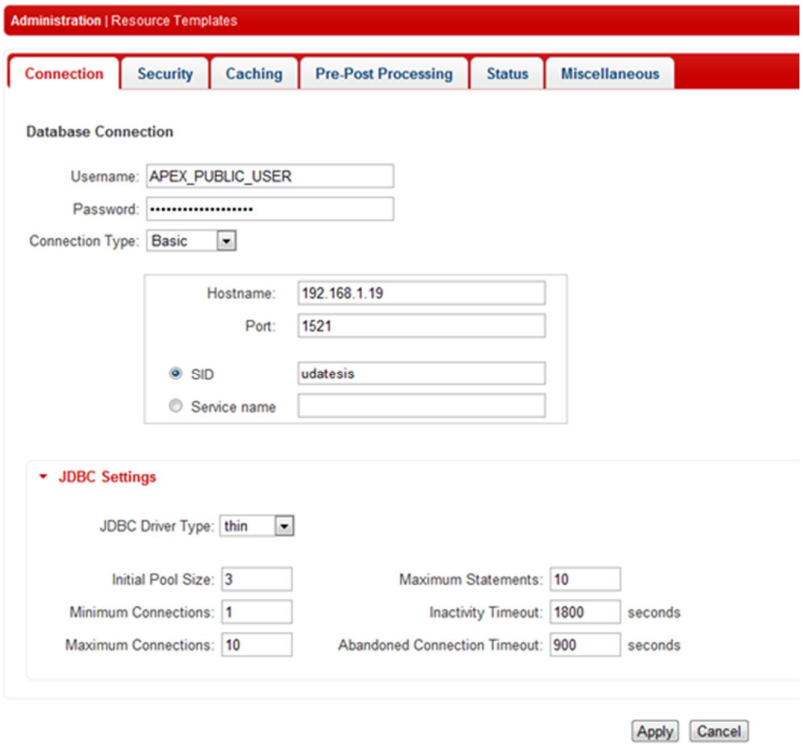


Figura 4-38: [Pantalla administración weblogic]. Obtenido desde: Consola administración servidor weblogic

4.2 Problemas de Integración

Cuando se planifica el desarrollo de cualquier aplicación la etapa de diseño juega un papel muy importante porque es en ella donde se pueden evidenciar futuros problemas, que en algunos casos son asumidos, pero que en otros casos necesariamente deben ser solucionados para no perjudicar el proyecto en su conjunto. Los problemas pueden variar desde la "simple" tarea de elegir la base de datos, el lenguaje de desarrollo o el servidor de aplicaciones que se debería utilizar.

Además de los requerimientos, que en primera instancia exponen algunas restricciones sobre las herramientas de desarrollo a emplear, también se debe considerar la arquitectura de la aplicación, es decir, si es una aplicación cliente-servidor, una aplicación web o una aplicación de tres capas. Otro de los factores importantísimos y que se deben tener en consideración es la infraestructura utilizada porque no solo determina cual es nuestro entorno de aplicación, sino que también determina los factores de costo y presupuestos asociados.

El propósito principal de esta tesis es el análisis y desarrollo de una aplicación que permita la administración de recursos, en nuestro caso servidores de correo y DHCP, a través de una arquitectura de tres capas. Considerando lo expuesto podemos intuir que para cumplir ese objetivo tendríamos que concebir un mecanismo para la comunicación entre la aplicación web y cada uno de los servidores de correo y DHCP. Si bien se desarrolló un mecanismo este no es perfecto y probablemente contenga errores que no han sido identificados en el proceso de pruebas. A continuación explico cuál es el mecanismo desarrollado y cuáles fueron los riesgos mitigados y también los asumidos.

Cuando nos enfrentamos a una aplicación cuya arquitectura es de tres capas disponemos de un servidor de base de datos, que es nuestro repositorio de información, también está presente un servidor de aplicaciones que se expone a los usuarios y por último se identifica el navegador o browser de los usuarios, que es el medio por el cual en última instancia se ingresa a la aplicación.

Cuando un usuario solicita la ejecución de una tarea en su navegador web se llevan a cabo algunas actividades que no tiene conocimiento. Primero su navegador web realiza la ejecución de procedimientos JavaScript que puede incluir validación de campos, ejecución de cálculos, o incluso la solicitud de información a través de la ejecución de procesos AJAX(Asynchronous JavaScript And XML), para este último necesariamente tendrá que contactar con el servidor de aplicaciones. Después de todas las validaciones y demás actividades del lado del cliente, el browser o navegador del cliente contacta con el servidor de aplicaciones para enviarle o receptar datos desde o hacia el cliente, este proceso normalmente causa el refresco de toda la página web visualizada. En el tiempo que transcurre durante el refresco de toda la página web del cliente, el servidor de aplicaciones contacta con la base de datos para enviarle los datos rellenados en la interfaz web por el usuario, para nuestro caso la base de datos no solamente almacena información sino que también la procesa a través del motor PLSQL, los datos resultantes del procesamiento son enviados al servidor de aplicaciones y este su vez envía al cliente.

En algunas arquitecturas de tres capas la lógica del negocio se lleva a cabo en el servidor de aplicaciones, regularmente se encuentran desarrolladas en java, sin embargo de lo descrito en el párrafo anterior podemos evidenciar que la lógica para la aplicación desarrollada en esta tesis esta almacenada en la base de datos. En nuestro caso como la lógica esta almacenada en la base de datos, entonces el servidor que alberga la base de datos debe ser capaz de contactar a los equipos remotos. Para lograr esta tarea como hemos explicado en el capítulo 3 se ha hecho uso la ejecución de procedimientos externos.

La aplicación desarrollada no tiene la característica de que la ejecución de sus procedimientos sea atómica, es decir, que se realice toda la tarea o no se realice nada. Esto debido principalmente a las esperas de CPU causadas en el servidor de base de datos cuando uno de los servidores remotos (correo, DHCP) se encuentra saturado de trabajo y no pueden responder rápidamente a las solicitudes enviadas por la aplicación.

Cuando una de las tareas que se encuentra en ejecución sobre el motor PLSQL de la base de datos requiere contactar a un servidor remoto se siguen los siguientes pasos:

1. Se verifica que servidores van a ser afectados, se extrae su dirección IP registrada, y se adiciona el comando a ejecutar.
2. De los datos obtenidos en el primer paso se genera una instrucción que contacta a través del protocolo *SSH (Secure Shell)* al servidor remoto, en donde ejecuta un script en background que siempre está presente en la ruta "/intradius" y que es el encargado de crear una nueva cuenta del sistema operativo, cuenta que será el nombre de usuario de la dirección de correo electrónico. La instrucción SSH es similar a la siguiente.
    a. '/usr/bin/ssh -n -f -t –t root@192.168.1.14 "sh /intradius/ABC_MAIL_MX.sh jorge.brito XXXXXXX I usuario 1 1'
3. Para que el Script sea ejecutado primeramente se realizan dos validaciones.
    a. Se verifica que tiene la llave pública del servidor donde se encuentra la base de datos, el registro debe ser similar al siguiente y cómo podemos observar en la última línea se encuentra el usuario que exportó la llave y el *hostname* del equipo que corre el motor RDBMS.
        i. ssh-dss

        AAAAB3NzaC1kc3MAAACBAJEoRxIKDBAhAj549sZgGVuqth1UEyGpJez7If/70zux3Y/cldvYggKwWQcKDs/SDDORcCexqGVmNBwXi+o2DF3D+r4pJ+7ORGcj1AQPGuLN9FaKDTTcT9gtD+LZ4ibvJvNcE5ULnSV4ps6DtRTZZzio3V+G4jLOKIjDz2sZuI2FAAAAFQCLoWZA37ZmJgW/Vo64R8EGFrrVgwAAAIEAhyVjEbJ8kfRbXWNn2HA0Gmv5XUV5Yqrd6JxP81lZi1rC1UxBz5GcSlIV5UxiC/9snZ/Tn56upJlIcvLlxowVjw0gnuPKjIFrxbIa4/54F0KAMLWR9bigRqIVD6dxcRt7pm/0Tq4q633cjHejWzFN0CHX3CA9yGZ+H3bgPyFE/3kAAACAGAWL5Y6F8MVcMpjhJzsxFM863EObt79HD0Ydg8AgfTWSZh0vhTWpJMCv+n7WuFPG2ji/Yzl3mKMZsXFmEk6tifMKrnljvT7mV6yCHWIdEkJ5tiublxi+H8/U20t22iAJUKPVNOnKmRHQjvC22D5N5dq1LgQMLY8g3Alr6nCJmn8= **oracle@Database**
    b. Se verifica que el directorio "/intradius" se encuentre montando, es decir, que esté conectado con el servicio samba que corre en el servidor Radius.
4. Ejecuta el proceso solicitado desde el motor PLSQL de la base de datos.
5. Notifica a la base de datos el estado de ejecución del proceso, es decir, si se ejecutó correctamente o existió algún problema. Esta notificación es indispensable debido a que el proceso fue lanzado en background y *APEX* ha aplicado un *commit* a la transacción en Oracle. Si existe algún error, reportado por esta notificación, entonces el administrador u operador deberán deshacer los cambios y reintentar el proceso que enviaron.

El problema principal que evidenciamos en la arquitectura es bastante claro, en pocas palabras podemos resumir este problema de integración en:

1. Si existe un corte de red lo suficientemente prolongado entonces es muy probable que la operación sea errónea, por lo tanto tendremos que volver a realizarla.
2. Si algunos de los scripts del S.O. remoto se corrompe entonces la operación es siempre errónea.
3. Si uno cualquiera de los scripts desarrollados es modificado por una persona que no conozca la arquitectura, lo más probable es que todas las operaciones a partir de la modificación del script sean inconsistentes o no se ejecuten.
4. Como el proceso se corre en background la aplicación no podría saber en si el servidor remoto se encuentra encendido o apagado.

Para solventar en parte esta serie de problemas se ha desarrollado algunos métodos, para algunos casos la solución es total sin embargo para otros es parcial y el riesgo se ha asumido. En una de las secciones siguientes de este mismo capítulo explico cuáles fueron las soluciones aplicadas y en qué grado solventaron el problema.

4.3 Problemas de Implementación

APEX es una aplicación instalada en el motor de base de datos de Oracle y debido a ello podemos suponer que es un esquema más del RDBMS. La aplicación *APEX* es un conjunto de procedimientos, paquetes, *triggers*, secuencias, etc. que permiten que la pantalla final que se presenta al usuario sea visualizada. Como la pantalla que se presenta al usuario tiene colores, imágenes, hojas de estilo, archivos JavaScript, etc. todo este contenido multimedia provoca que el rendimiento del gestor de base de datos se vea afectado. Ese es uno de los primeros problemas que tenemos con la implementación de esta herramienta, sin embargo no es de preocuparse porque existe una alternativa de solución, un servidor de aplicaciones, que se explicará más a detalle en la sección de soluciones aplicadas.

Hoy en día HTML por si solo ya no constituye una solución para aplicaciones web, y aunque HTML permite el envío de datos a través de formularios, eso no es suficiente para las necesidades actuales de las empresas. Las empresas actualmente necesitan

una verdadera aplicación que funcione en web, aplicación que permita registrar transacciones, generar reportes, etc. Para lograr ese objetivo entonces nacen algunas tecnologías como PHP, ASP, etc., sin embargo el tiempo de desarrollo que se necesitaría para desarrollar una aplicación era extremadamente alto y no justificaba los gastos asociados. Para solventar este problema nacen tecnologías como *APEX* que permite el desarrollo de aplicaciones transaccionales a través de un ambiente de desarrollo que también se encuentra en la web.

Con *APEX* se logra disponer de una herramienta de desarrollo rápida y fácil de aprender. Con esta herramienta somos capaces de consultar datos simplemente con establecer el SQL en el entorno de desarrollo, el *html* para visualizar el resultado ya está preestablecido y se genera dinámicamente de acuerdo a los datos visualizados, aunque si queremos podríamos establecer nuestros diseños. Sin embargo cada consulta que se realiza al servidor genera que toda la página sea recargada por completo y este comportamiento obviamente da origen a un nuevo problema de rendimiento que se puede resolver a través de llamadas asíncronas como veremos en la sección siguiente.

Con *html* simple y javascript estático se torna complicado generar menús que sean dinámicos, para ayudarnos *APEX* nuevamente implementa una solución a este inconveniente y nos da plantillas con las que podemos generar un menú es cuestión de minutos. Pero aunque se generan de forma dinámica no tienen funciones de búsqueda, visualización de hojas, etc. Se buscó una solución a este problema, la solución emplea una librería externa a *APEX* llamada "*EXT js*" que se integra perfectamente, explicaré más a detalle en la siguiente sección.

Uno de los últimos problemas de implementación nace a través de la necesidad de visualizar indicadores de actividad a través del uso de cronómetros, el problema es mitigado por *APEX* a través del uso de la tecnología *FLASH* y procedimientos almacenados en la base de datos para retornar Querys dinámicos.

A pesar de que *APEX* proporciona wizards para generar consultas asíncronas al servidor de aplicaciones y este a su vez a la base de datos, estos métodos no son suficientes para satisfacer nuestros requerimientos de performance. El uso de

JavaScript con llamadas a procedimientos “bajo demanda” como se denomina en *htmldb* o actualmente *APEX*, nos permiten ejecutar procedimientos almacenados en la base de datos sin la necesidad de recargar toda la página web mejorando significativamente el rendimiento, velocidad o performance de nuestra aplicación. Para confirmar que en realidad se están generando consultas hacia la base de datos, actualmente navegadores como Firefox, Chrom o Safari ayudan a los desarrolladores con herramientas para visualizar las llamadas http generadas por código JavaScritpt.

Aunque el uso de JavaScritpt mejora considerablemente el rendimiento, esta cualidad se puede perder si la máquina del usuario final no dispone de características necesarias, este efecto se presenta principalmente porque JavaScritpt es un lenguaje ejecutado netamente en el browser del usuario final. Entre las características que nuestro equipo debe tener para que la aplicación funcione normalmente están:

1. CPU: Core duo 2.4Gzh
2. Memoria: 2GB
3. Browser: Firefox 13,14. Chrom 20, Safari últimas versiones(No se recomienda Internet Explorer, sin embargo si es una navegador de uso obligatorio se podría usar la versión 9)
4. Plugin *Flash* instalado en el navegador web.

4.4 **Soluciones aplicadas**

En los últimos años la trasmisión de datos a través de la red ha mejorado considerablemente, el conocimiento acerca de las interferencias, y en definitiva todo factor externo que perjudica la trasmisión de información a través de impulsos eléctricos, ha permitido el desarrollo de medios de transmisión más efectivos. Partiendo de este supuesto podemos entonces suponer que el problema en la ejecución de scripts por cortes de red ya ha sido reducido, sin embargo, si un servidor remoto se encuentra en una ubicación en donde no existan métodos fiables de comunicación como transmisión por radio, entonces el sistema se puede ver afectado.

Para solventar este problema se diseñó el sistema de modo que constantemente se esté comprobando conectividad con los servidores remotos, si uno de ellos falla entonces su estado es cambiado a “*DOWN*”, cuando el sistema intenta atender una

solicitud del operador u administrador y encuentra el estado del servidor en el estado down entonces no realiza ninguna operación y se le notifica al usuario acerca del problema de conexión con el equipo remoto.

Aunque la verificación solventa problemas de red esta no puede determinar si la ejecución de un script fue exitosa o fallida, para atenuar este inconveniente entonces cada uno de los scripts tienen la capacidad de conectarse directamente con la base de datos y registrar el resultado de la operación. Aunque hayamos registrado el resultado de la operación no podemos realizar un *rollback* de la transacción en la base de datos es por ello que el usuario deberá revisar el reporte de ejecución, si se encuentra coloreado el registro la operación tuvo un error y se debería deshacer los cambios para nuevamente intentar la ejecución del procedimiento; sin embargo si el reporte registra un resultado correcto es porque en realidad la operación fue exitosa, esta metodología agrega confiabilidad a nuestra aplicación.

La modificación de los scripts que receptan las instrucciones enviadas por el motor PLSQL es muy peligrosa porque puede afectar directamente a la funcionalidad del sistema generando inconsistencias entre lo que la base de datos almacena y lo que se ejecutó en los servidores remotos. Para solucionar este problema, que fue detallado en la secciona anterior, solamente se dispondrá de respaldos que el administrador de la aplicación lo haya realizado, si bien la solución a este problema no es tan óptima pero ayuda a que el administrador u operador de vez en cuando ingresen al servidor Radius que contiene los scripts.

Existen dos métodos con los cuales la aplicación podrá determinar que un servidor remoto se encuentra apagado, el primero podría ser la verificación del servidor al momento en que se intenta ejecutar una instrucción u otra es no verificar al momento en que se intenta ejecutar la instrucción sino disponer de esta información previamente. Hemos adoptado la segunda, es decir, el sistema constantemente envía instrucciones ICMP a servidor registrado, si el servidor remoto se encuentra apagado entonces su estado es cambiado. Cuando el servidor remoto arranca y se encuentra totalmente cargado entonces se ejecuta un evento que notifica al sistema que ya se encuentra activo, este evento cambia el estado del servidor registrado.

Unos de los problemas de implementación que habíamos mencionado es que el servidor de base de datos se encargaba de procesar toda la información necesaria para presentar una página web a cada usuario, esto inevitablemente es un problema. Para resolver este problema se tiene que implementar un servidor de aplicaciones de modo que: la base de datos solamente contenga los datos y lógica de la aplicación y todos los binarios sean trasladados a este nuevo servidor.

El resultado de dividir la lógica y datos en un equipo y binarios en otro, hace que cuando un usuario solicite una página se retorne casi simultáneamente datos como binarios mejorando el performance. Sin embargo si tanto base de datos y servidor de aplicaciones están en el mismo servidor físico entonces esta ventaja se pierde porque el sistema operativo tiene que proporcionar accesos a disco de forma secuencial. Aunque binarios es la palabra adecuada para describir archivos de imágenes, videos, etc. comprende demasiados ficheros, es por ello que detallo cuales son los archivos que el servidor, para nuestra tesis, alberga:

1. Archivos JavaScript de apex.
2. Archivos JavaScript JQuery.
3. Archivos de imagen.
4. Archivos de video.
5. Archivos EXT js.
6. Archivos CSS

En la sección de problemas de integración e implementación también habíamos hablado del problema de recargar constantemente toda la pantalla del navegador y las implicaciones de rendimiento consiguientes, pues ahora expongo cual es la solución que nos ayuda y que porcentaje. El uso de AJAX permite que carguemos solamente la información necesaria y no actualicemos toda la pantalla, el empleo de procesos bajo demanda es un método que hace uso de AJAX para funcionar. Mientras más conocimientos de JavaScript tengamos entonces podremos mejorar el desarrollo de las interfaces y por consiguiente el performance de la aplicación.

Cuando los menús en cualquier aplicación son extremadamente grandes es preferible disponer de métodos de búsqueda de las opciones del que dispone, esta funcionalidad

es completamente entregada por el componente EXT js, que es un plugin que se instala en *APEX* y que necesita solamente un query sql para generar el menú deseado.

Todas las librerías, imágenes, etc. de este componente pueden ser descargadas desde la página web del fabricante www.sencha.com. Para instalar se tiene que compilar un paquete en el esquema propietario de la aplicación, después de ello se tiene que copiar las librerías *javascript* que pueden estar en la base de datos o en el servidor de aplicaciones, para sistemas en producción se recomienda que se ubique en el servidor de aplicaciones.

Se puede configurar el componente para que cada nodo sea carga con llamadas AJAX, sin embargo si el número de nodos es elevado entonces es preferible que se configure de modo que todos los nodos sean cargados al momento de desplegar cualquier página.

Por motivos de interoperabilidad es recomendable que todo el contenido visualizado por cualquier aplicación web sea desarrollado en HTML y no en flash, sin embargo debido a la complejidad del desarrollo de cronómetros o tacómetros indicadores de actividad del sistema, se hizo necesario usar esta tecnología para satisfacer este requerimiento. *APEX* genera dinámicamente las animaciones flash en base a un query que puede ser dinámico o estático, los valores mínimos y máximos del tacómetro son también establecidos.

Aunque la virtualización no es parte de esta tesis podemos al menos citarlo por ser parte fundamental y solución empleada para levantar cada uno de los servidores de prueba utilizados. El hypervisor que se usó para levantar cada servidor de prueba es *qemu*, uno de los sistemas open source más eficientes que existen en la actualidad. Con el empleo de la virtualización, se pudo utilizar un solo servidor físico con procesador Intel Xeon de 2.4 Ghz y 16 Gb de RAM, para correr 9 servidores virtuales, aunque es necesario aclarar que al menos los servidores de correo no deberían ser utilizados, con la capacidad que se asignó en esta tesis, para estar en operación sobre ambientes de producción; para este tipo de ambientes se recomienda

analizar cuidadosamente cuales son los requisitos de red, disco entre otros factores relevantes y que influyen directamente en la calidad final del servicio brindado.

Conclusiones y Recomendaciones

Podemos entonces concluir que es necesario que tengamos siempre presente que cada vez que se desarrolla una aplicación, ya desde la etapa de diseño podemos pre visualizar cuáles serán las implicaciones de usar una tecnología u otra. Independientemente del uso de cualquier arquitectura siempre estarán presentes problemas de implementación y de integración, deberemos entonces nosotros determinar cuáles de ellos serán solucionados y cuáles serán aceptados, siempre tomando en consideración el factor costo beneficio.

Las ventajas de una aplicación web son numerosas pero constantemente vendrán dada de la mano por la velocidad (entre otros factores), y la velocidad estará ligada estrechamente a los métodos adoptados para mejorar el rendimiento, métodos que pueden varias desde el tipo de discos duros, cantidad y uso de memoria del servidor, servidores de aplicaciones, calidad del código fuente y el manejo de interfaces web a través del uso de *AJAX*.

Como recomendación principal citaremos que cada vez que se realice una aplicación que implique la comunicación con equipos remotos, se emplee todo el tiempo necesario para que las interfaces entre los equipos sean lo más robustas posible. Mientras más trabajemos en el aspecto interacción o comunicación, con menos problemas nos encontraremos en etapas de desarrollo, disminuyendo las pérdidas innecesarias de valioso tiempo y optimizando nuestros procesos de “fabricación” de software.

La documentación en el código fuente debería ser obligatoria, es decir, por cada procedimiento se debe especificar claramente que función desempeña, esta información es vital para procesos de mantenimiento futuros. En el anexo uno encontraremos el diagrama físico de tablas, campos y relaciones, comúnmente llamado diagrama entidad relación.

Se recomienda también se medite sobre las tecnologías a emplear, lo más probable, para que un software sea de calidad, es que se requiera contratar personal especializado en cada una de ellas y esta condición afectará directamente en los costos asociados al desarrollo, finalmente lo que determina si un desarrollo es viable son los costos asociados versus la calidad de producto entregado.

Finalmente me permito recomendar que para que un sistema funcione adecuadamente, independientemente de la arquitectura, lenguajes, etc. lo fundamental es contar con los “fierros” o hardware adecuado. Por más optimización de código, uso de servidores de aplicaciones, etc. que se use, si mi hardware no es lo suficientemente potente el proyecto fracasará. Es recomendable para sistemas en producción se use solamente equipos certificados de compañías grandes y con experiencia en este tipo de “terrenos”, con esto estamos reduciendo los riesgos agregando valor al software final.

Bibliografía

AHMAD Sohail. WAP TOO! [En Línea]. Obtenido desde http://www.defcon.org/images/defcon-18/dc-18-presentations/Ahmad/DEFCON-18-Ahmad-WPA-Too.pdf. 2010.

CÁCERES Jesús. DIAGRAMAS DE SECUENCIA [En línea]. Obtenido desde http://www2.uah.es/jcaceres/capsulas/DiagramaSecuencia.pdf. Madrid 2008.

CERIA Santiago, Ingeniería de software I, Casos de Uso [En línea]. Obtenido desde http://www-2.dc.uba.ar/materias/isoft1/2001_2/apuntes/CasosDeUso.pdf. Buenos Aires 2009.

CONWAY Harry, AULT Mike, BURLESON. Oracle Tuning Power Scripts: With 1--+ High Performace SQL Scripts, Donald. Ediciones Don Burleson, North Carolina-USA, 2005.

COOPER Mendel. Advanced Bash-Scripting Guide [En línea]. Obtenido desde http://tldp.org/LDP/abs/abs-guide.pdf. Abril 2012.

FUENTES, Joaquín. Realidad virtual aplicada al tratamiento del trastorno de lateralidad y ubicación espacial (Capítulo 2) [En línea]. Obtenido desde http://catarina.udlap.mx/u_dl_a/tales/documentos/lis/fuentes_k_jf/capitulo2.pdf. Cholula, Puebla, México 2003

GEDRIS Victor. An Introduction to the Linux Command Shell For Beginners [En línea]. Obtenido desde http://vic.gedris.org/Manual-ShellIntro/1.2/ShellIntro.pdf. Ottawa 2003.

GELBMANN Matthias. Highlights of web technology surveys, July 2010: CentOS is now the most popular Linux distribution on web servers [En línea]. Obtenido desde http://w3techs.com/blog/entry/highlights_of_web_technology_surveys_july_2010. Julio 2010.

HALL Timothy. Oracle PL/SQL Tuning: Expert Secrets for High Performance Programing, Ediciones Don Burleson, North Carolina- USA, 2006.

HUGUET Vicente, DOMÉNECH Sonia. Linux Sistema de Ficheros [En línea]. Obtenido desde http://mural.uv.es/oshuso/831_tipos_de_shell.html. València Abril 2011.

INTEL-Corporation. Guía del usuario de la utilidad Intel PROSet/Wireless Wifi [En línea]. Obtenido desde http://support.elmark.com.pl/rgd/drivery/U12C/WLAN/Win7/Docs/ESN/overview.htm#mchap2. Julio 2010.

LEHEMBRE Guillaume. Seguridad Wi-Fi –WEP, WAP y WAP2 [En línea]. http://www.hsc.fr/ressources/articles/hakin9_wifi/hakin9_wifi_ES.pdf. Francia 2006.

LOAYZA Carlos. SEGURIDAD PARA LA RED INALÁMBRICA DE UN CAMPUS UNIVERSITARIO [En línea]. Obtenido desde http://www.utpl.edu.ec/seguridad/wp-content/uploads/2008/10/seg_wifi.pdf. Loja 2008.

LÓPEZ José. Protocolo SSH [En línea]. Obtenido desde http://sopa.dis.ulpgc.es/ii-aso/portal_aso/leclinux/seguridad/ssh/ssh.pdf. 2005.

MARTINEZ Antonio. INTRODUCCION A SHELL SCRIPT [En línea]. Obtenido desde http://www.elviajero.org/antoniux/tutos/shell_intro.pdf. España 2005.

MICROSOFT-TechNet. MS-CHAP v2 [En línea]. Obtenido desde http://technet.microsoft.com/en-us/library/cc957983.aspx. [s.a]

MICROSOFT-TechNet. MS-CHAP versión 2 [En línea]. Obtenido desde http://technet.microsoft.com/es-es/library/cc739678%28v=ws.10%29.[s.a]

MUDAY Jeff. A Brief Introduction to File and Print Sharing on Linux Using Samba [En línea]. Obtenido desde http://users.wfu.edu/mudayja/presentations/samba-intro-2002.pdf. 2003.

MUJICA Manuel. Permisos en GNU/Linux [En línea]. Obtenido desde http://mmujica.files.wordpress.com/2007/05/permisos-linux.pdf. Mayo 2007.

NIEMIEC Richard. ORACLE DATABASE 10g Performace Tunning: Tips & Techniques. Ediciones McGRAW – HILL/OSBORNE. 2007

ORACLE Corporation, Oracle Application Express: APEX Listener [En línea]. Obtenido desde http://www.oracle.com/technetwork/developer-tools/apex-listener/overview/index.html. Julio 2012.

Oracle Corporation, Oracle Application Express Listener: Installation and Devleoper Guide [En línea]. Obtenido desde http://download.oracle.com/otn/java/appexpress/1.1/docs/AELIG/E21058_01.pdf?AuthParam=1344309981_0c5d8e1fe3fe9e335b864e51f2343428. [s.a]

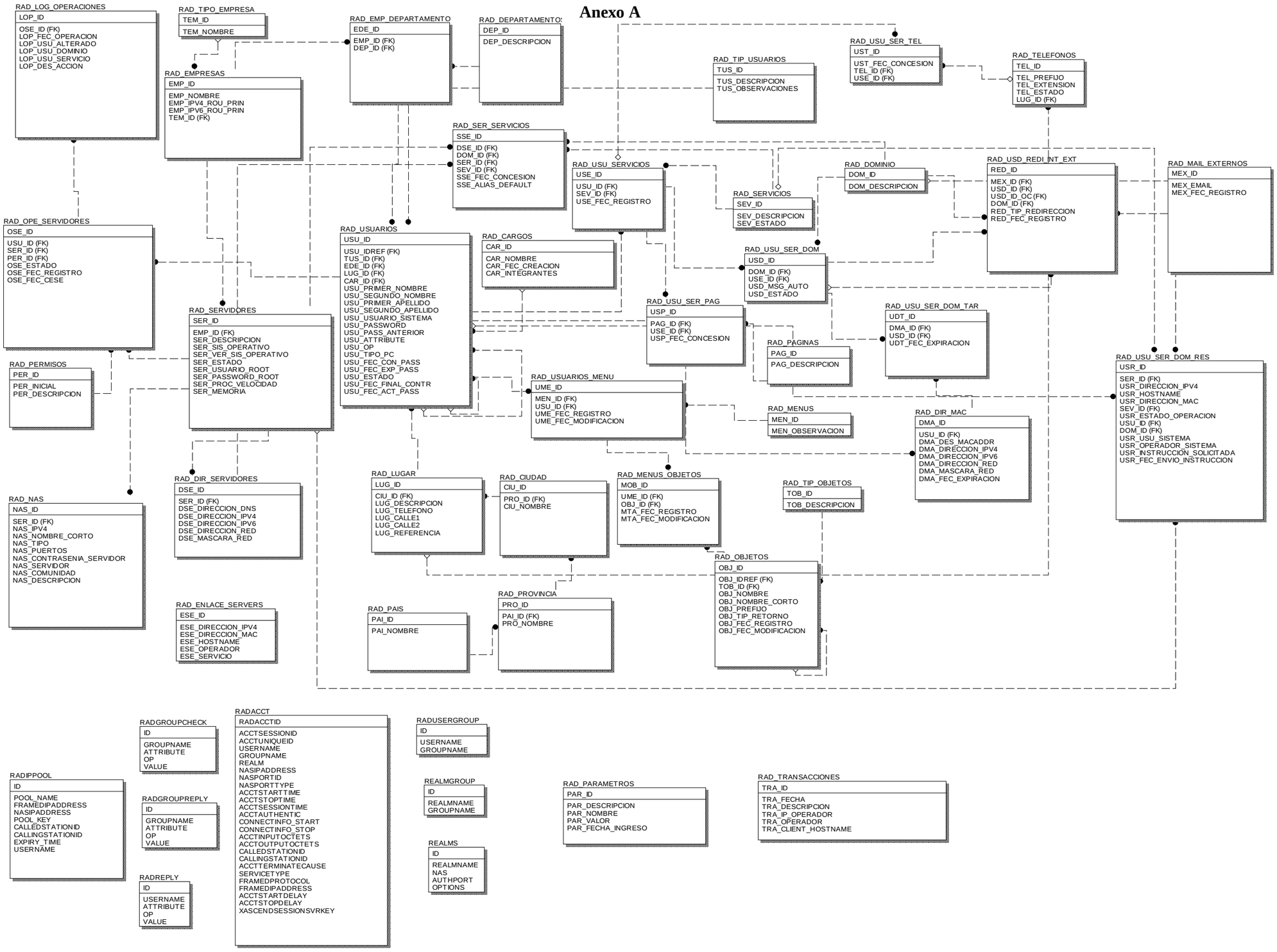
OTERO Mari. Diagramas de estado [En línea]. Obtenido desde http://caceros.docentes.upbbga.edu.co/I2012/SIM_I2012/laboratorios/lab1/Diagramas_estado.pdf [s.a]

SOMMERVILLE Ian,. INGENIERÍA DEL SOFTWARE [En línea]. Obtenido desde http://books.google.com.ec/books?hl=en&lr=&id=gQWd49zSut4C&oi=fnd&pg=PA1&dq=requerimientos+funcionales+y+no+funcionales+de+software&ots=s531ppzCtf&sig=YCLO_nboPgEjscy9MnOwjeRXvU#v=onepage&q=requerimientos%20funcionales%20y%20no%20funcionales%20de%20software&f=false. Madrid 2005.

TORRES Manuel. Fundamentos del diseño [En línea]. Obtenido desde http://indalog.ual.es/mtorres/LP/FundamentosDiseno.pdf. España 2006.

URÍA Álvaro. Shell Scripting: programación con comandos de shell [En línea]. Obtenido desde http://www.e-ghost.deusto.es/docs/shellScriptin.html. Agosto 2003.

Anexo A


RAD_LOG_OPERACIONES
LOP_ID
OSE_ID (FK)
LOP_FEC_OPERACION
LOP_USU_ALTERADO
LOP_USU_DOMINIO
LOP_USU_SERVICIO
LOP_DES_ACCION
RAD_TIPO_EMPRESA
TEM_ID
TEM_NOMBRE
RAD_EMPRESAS
EMP_ID
EMP_NOMBRE
EMP_IPV4_ROU_PRIN
EMP_IPV6_ROU_PRIN
TEM_ID (FK)
RAD_EMP_DEPARTAMENTO
EDE_ID
EMP_ID (FK)
DEP_ID (FK)
RAD_DEPARTAMENTOS
DEP_ID
DEP_DESCRIPCION
RAD_TIP_USUARIOS
TUS_ID
TUS_DESCRIPCION
TUS_OBSERVACIONES
RAD_USU_SER_TEL
UST_ID
UST_FEC_CONCESION
TEL_ID (FK)
USE_ID (FK)
RAD_TELEFONOS
TEL_ID
TEL_PREFIJO
TEL_EXTENSION
TEL_ESTADO
LUG_ID (FK)
RAD_SER_SERVICIOS
SSE_ID
DSE_ID (FK)
DOM_ID (FK)
SER_ID (FK)
SEV_ID (FK)
SSE_FEC_CONCESION
SSE_ALIAS_DEFAULT
RAD_USU_SERVICIOS
USE_ID
USU_ID (FK)
SEV_ID (FK)
USE_FEC_REGISTRO
RAD_SERVICIOS
SEV_ID
SEV_DESCRIPCION
SEV_ESTADO
RAD_DOMINIO
DOM_ID
DOM_DESCRIPCION
RAD_USD_REDI_INT_EXT
RED_ID
MEX_ID (FK)
USD_ID (FK)
USD_ID_OC (FK)
DOM_ID (FK)
RED_TIP_REDIRECCION
RED_FEC_REGISTRO
RAD_MAIL_EXTERNOS
MEX_ID
MEX_EMAIL
MEX_FEC_REGISTRO
RAD_OPE_SERVIDORES
OSE_ID
USU_ID (FK)
SER_ID (FK)
PER_ID (FK)
OSE_ESTADO
OSE_FEC_REGISTRO
OSE_FEC_CESE
RAD_USUARIOS
USU_ID
USU_IDREF (FK)
TUS_ID (FK)
EDE_ID (FK)
LUG_ID (FK)
CAR_ID (FK)
USU_PRIMER_NOMBRE
USU_SEGUNDO_NOMBRE
USU_PRIMER_APELLIDO
USU_SEGUNDO_APELLIDO
USU_USUARIO_SISTEMA
USU_PASSWORD
USU_PASS_ANTERIOR
USU_ATTRIBUTE
USU_OP
USU_TIPO_PC
USU_FEC_CON_PASS
USU_FEC_EXP_PASS
USU_ESTADO
USU_FEC_FINAL_CONTR
USU_FEC_ACT_PASS
RAD_CARGOS
CAR_ID
CAR_NOMBRE
CAR_FEC_CREACION
CAR_INTEGRANTES
RAD_USU_SER_DOM
USD_ID
DOM_ID (FK)
USE_ID (FK)
USD_MSG_AUTO
USD_ESTADO
RAD_USU_SER_PAG
USP_ID
PAG_ID (FK)
USE_ID (FK)
USP_FEC_CONCESION
RAD_PAGINAS
PAG_ID
PAG_DESCRIPCION
RAD_USU_SER_DOM_TAR
UDT_ID
DMA_ID (FK)
USD_ID (FK)
UDT_FEC_EXPIRACION
RAD_SERVIDORES
SER_ID
EMP_ID (FK)
SER_DESCRIPCION
SER_SIS_OPERATIVO
SER_VER_SIS_OPERATIVO
SER_ESTADO
SER_USUARIO_ROOT
SER_PASSWORD_ROOT
SER_PROC_VELOCIDAD
SER_MEMORIA
RAD_PERMISOS
PER_ID
PER_INICIAL
PER_DESCRIPCION
RAD_USUARIOS_MENU
UME_ID
MEN_ID (FK)
USU_ID (FK)
UME_FEC_REGISTRO
UME_FEC_MODIFICACION
RAD_MENUS
MEN_ID
MEN_OBSERVACION
RAD_DIR_MAC
DMA_ID
USU_ID (FK)
DMA_DES_MACADDR
DMA_DIRECCION_IPV4
DMA_DIRECCION_IPV6
DMA_DIRECCION_RED
DMA_MASCARA_RED
DMA_FEC_EXPIRACION
RAD_USU_SER_DOM_RES
USR_ID
SER_ID (FK)
USR_DIRECCION_IPV4
USR_HOSTNAME
USR_DIRECCION_MAC
SEV_ID (FK)
USR_ESTADO_OPERACION
USU_ID (FK)
DOM_ID (FK)
USR_USU_SISTEMA
USR_OPERADOR_SISTEMA
USR_INSTRUCCION_SOLICITADA
USR_FEC_ENVIO_INSTRUCCION
RAD_DIR_SERVIDORES
DSE_ID
SER_ID (FK)
DSE_DIRECCION_DNS
DSE_DIRECCION_IPV4
DSE_DIRECCION_IPV6
DSE_DIRECCION_RED
DSE_MASCARA_RED
RAD_LUGAR
LUG_ID
CIU_ID (FK)
LUG_DESCRIPCION
LUG_TELEFONO
LUG_CALLE1
LUG_CALLE2
LUG_REFERENCIA
RAD_CIUDAD
CIU_ID
PRO_ID (FK)
CIU_NOMBRE
RAD_MENUS_OBJETOS
MOB_ID
UME_ID (FK)
OBJ_ID (FK)
MTA_FEC_REGISTRO
MTA_FEC_MODIFICACION
RAD_TIP_OBJETOS
TOB_ID
TOB_DESCRIPCION
RAD_OBJETOS
OBJ_ID
OBJ_IDREF (FK)
TOB_ID (FK)
OBJ_NOMBRE
OBJ_NOMBRE_CORTO
OBJ_PREFIJO
OBJ_TIP_RETORNO
OBJ_FEC_REGISTRO
OBJ_FEC_MODIFICACION
RAD_PROVINCIA
PRO_ID
PAI_ID (FK)
PRO_NOMBRE
RAD_PAIS
PAI_ID
PAI_NOMBRE
RAD_NAS
NAS_ID
SER_ID (FK)
NAS_IPV4
NAS_NOMBRE_CORTO
NAS_TIPO
NAS_PUERTOS
NAS_CONTRASENIA_SERVIDOR
NAS_SERVIDOR
NAS_COMUNIDAD
NAS_DESCRIPCION
RAD_ENLACE_SERVERS
ESE_ID
ESE_DIRECCION_IPV4
ESE_DIRECCION_MAC
ESE_HOSTNAME
ESE_OPERADOR
ESE_SERVICIO
RADACCT
RADACCTID
ACCTSESSIONID
ACCTUNIQUEID
USERNAME
GROUPNAME
REALM
NASIPADDRESS
NASPORTID
NASPORTTYPE
ACCTSTARTTIME
ACCTSTOPTIME
ACCTSESSIONTIME
ACCTAUTHENTIC
CONNECTINFO_START
CONNECTINFO_STOP
ACCTINPUTOCTETS
ACCTOUTPUTOCTETS
CALLEDSTATIONID
CALLINGSTATIONID
ACCTTERMINATECAUSE
SERVICETYPE
FRAMEDPROTOCOL
FRAMEDIPADDRESS
ACCTSTARTDELAY
ACCTSTOPDELAY
XASCENDSESSIONSVRKEY
RADGROUPCHECK
ID
GROUPNAME
ATTRIBUTE
OP
VALUE
RADGROUPREPLY
ID
GROUPNAME
ATTRIBUTE
OP
VALUE
RADREPLY
ID
USERNAME
ATTRIBUTE
OP
VALUE
RADIPPOOL
ID
POOL_NAME
FRAMEDIPADDRESS
NASIPADDRESS
POOL_KEY
CALLEDSTATIONID
CALLINGSTATIONID
EXPIRY_TIME
USERNAME
RADUSERGROUP
ID
USERNAME
GROUPNAME
REALMGROUP
ID
REALMNAME
GROUPNAME
REALMS
ID
REALMNAME
NAS
AUTHPORT
OPTIONS
RAD_PARAMETROS
PAR_ID
PAR_DESCRIPCION
PAR_NOMBRE
PAR_VALOR
PAR_FECHA_INGRESO
RAD_TRANSACCIONES
TRA_ID
TRA_FECHA
TRA_DESCRIPCION
TRA_IP_OPERADOR
TRA_OPERADOR
TRA_CLIENT_HOSTNAME