Universidad Autónoma del Estado de México

Facultad de Ingeniería

Diseño e Implementación de un Dominio Seguro Basado en TAM y LDAP para el Servicio de la Banca en Línea en una Institución Bancaria en Venezuela

MEMORIA

Que para obtener el título de

Ingeniero en Computación

Presenta:

Jaime Alberto Varela Calderón

Asesora:

M. en I. Sara Vera Noguez

Toluca, México
Octubre de 2013

Toluca, Edo. de México a 6 de Septiembre del 2013

C. JAIME ALBERTO VARELA CALDERÓN
PASANTE DE INGENIERÍA EN COMPUTACIÓN
P R E S E N T E

En respuesta a su solicitud, a continuación transcribo el tema aprobado por esta Dirección, que propuso la **M. EN I. SARA VERA NOGUEZ,** con el fin de que lo desarrolle en la modalidad de **MEMORIA,** le informo que se autoriza la **impresión de su trabajo** para presentar su Evaluación Profesional.

"DISEÑO E IMPLEMENTACIÓN DE UN DOMINIO SEGURO BASADO EN TAM Y LDAP PARA EL SERVICIO DE LA BANCA EN LÍNEA EN UNA INSTITUCIÓN BANCARIA EN VENEZUELA"

ÍNDICE

Ruego a usted tomar nota de que, en cumplimiento a lo especificado por la Ley de Profesiones, deberá prestar Servicio Social durante un tiempo mínimo de seis meses, como requisito indispensable para sustentar su Evaluación Profesional.

Asimismo, para la elaboración de la **MEMORIA** y demás trámites, deberá sujetarse a la reglamentación respectiva de esta Universidad.

A T E N T A M E N T E
PATRIA, CIENCIA Y TRABAJO
"2013, 50 Aniversario Luctuoso del Poeta Heriberto Enríquez"

M. EN I. RAUL VERA NOGUEZ
DIRECTOR

FACULTAD DE INGENIERÍA
U.A.E.M

**/agk

Agradecimientos

A mi madre, por haberme dado la vida y brindarme por siempre su amor, cuidado, y apoyo.

A mi hijo Jaime Santiago, quien me inspiró a dar lo mejor de mí en este trabajo.

Introducción

Esta memoria fue construida con base en una serie de experiencias obtenidas a partir del proceso de diseño e implementación de una solución de seguridad informática orientada al control de accesos para servicios de la banca en línea publicados en internet para una institución bancaria en Venezuela. El propósito principal de esta memoria es mostrar al lector la manera en como se ejecutaron las tareas necesarias para construir una solución de dominio seguro, resaltando la importancia de mantenerse alineado a las metodologías *ITIL v3.0* y *COBIT v4.1.*

Por otro lado, cabe mencionar que se hace especial énfasis en los límites de responsabilidad para cada uno de los miembros dentro del proyecto, ya que la solución involucra productos de diversos fabricantes de tecnología, y un factor importante para lograr el éxito del proyecto fue justamente el asignar y delimitar correctamente las responsabilidades y obligaciones de cada participante.

La memoria se divide en tres capítulos principales, el primero de ellos describe de manera general el planteamiento de la problemática o situación original, dentro de la cuál se presentan algunas imágenes que muestran la solución anterior, así como algunas gráficas de rendimiento.

El segundo capítulo consta de una gran cantidad de información ya que precisamente es en donde se describe el procedimiento para hacer el dimensionamiento de capacidades requeridas para cada componente de la solución, así como también se presenta la definición del modelo de datos. Para la fase de implementación es importante señalar que se describen los procesos técnicos y administrativos para ejecutar la transición sin mencionar explícitamente los comandos capturados, ya que esto se presenta en el apartado de anexos.

El tercer y último capítulo contiene una serie de gráficas y tablas que nos sirven para poder comparar el rendimiento de la solución implementada con respecto a la solución anterior. Dentro de este capítulo se menciona de manera breve la experiencia profesional adquirida, así como una serie de comentarios finales y recomendaciones para el lector con el propósito de acotar el uso de esta memoria.

Cabe señalar que es recomendable, más no requerido que lector cuente con conocimientos básicos sobre soluciones de autenticación y autorización para una mejor comprensión del contenido.

Tabla de Contenido

Índice de Figuras

Índice de Tablas

Tabla de Acrónimos

ACI.- Access Control Item

ACL.- Access Control List

API.- Application Programming Interface

BCS.- Business Continuity Strategy

BIA.- Business Impact Analysis

BTU.- British Thermal Unit

CA.- Certificate Authority

CAB.- Change Advisory Board

CFO.- Chief Financial Officer

CIO.- Chief Information Officer

CMS.- Configuration Management System

CN.- Common Name

COBIT.- Control Objectives for Information and Related Technology

CPU.- Control Processing Unit

CSI.- Continual Service Improvement

CISO.- Chief Information Security Officer

DIT.- Directory Information Tree

DN.- Distinguished Name

DRP.-Disaster Recovery Plan

EAR.- Enterprise ARchive

ELS.- Early Life Support

FIPS.- Federal Information Processing Standards

FMA.- Fault Management Architecture

GSSAPI.- Generic Security Services API

HBA.- Host Bus Adapter

HTML.- HyperText Markup Language

HTTP.- HyperText Transfer Protocol

HTTPS.- HyperText Transfer Protocol Secure

DB2.- Database 2

ICANN.- Internet Corporation for Assigned Names and Numbers

IETF.- Internet Engineering Task Force

ISO.- International Standards Organization

ITDS.- IBM Tivoli Directory Server

ITU-T.- International Telecommunication Union - Telecommunication

IP.- Internet Protocol

ISACA.- Information Systems Audit and Control Association

ITIL.- Technology Infrastructure Library

KPI.- Key Performance Indicator

LAN.- Local Area Network

LDAP.- Lightweight Directory Access Protocol

LDIF.- LDAP Data Interchange Format

MD5.- Message Digest 5

MTTR.- Mean Time To Repair

MTBF.- Mean Time Between Failure

NDA.- Non Disclosure Agreement

OU.- Organization Unit

PCI.- Payment Card Industry

PDU.- Power Distribution Unit

PKI.- Public Key Infrastructure

QOS.- Quality Of Service

RACF.- Resource Access Control Facility

RAM.- Random Access Memory

RBAC.- Role Based Access Control

RFC.- Request For Change

RFP.- Request For Proposal

RISC.- Reduced Instruction Set Computer

ROI.- Return On Investment

RUV.- Replication Update Vector

SAF.- System Authorization Facility

SAN.- Storage Area Network

SASL.- Simple Authentication Security Layer

SHA.- Secure Hash Algorithm

SKMS.- Service Knowledge Management System

SLA.- Service Level Agreement

SPARC.- Scalable Processor ARChitecture

SSL.- Secure Sockets Layer

TAM.- Tivoli Access Manager

TCO.- Total Cost of Ownership

TCP.- Transport Control Protocol

TIM.- Tivoli Identity Manager

TLS.- Transport Layer Security

TPM.- Trusted Platform Module

UDP.- User Datagram Protocol

URL.- Uniform Resource Locator

UML.- Unified Modeling Language

VPN.- Virtual Private Network

VAC.- Voltage in Alternating Current

XML.- Extensible Markup Language

1.- PLANTEAMIENTO DEL PROBLEMA

En este primer capítulo se describe la situación original que se tenía en la plataforma tecnológica sobre la que operaba el servicio de la banca en línea al interior de una Institución bancaria previo a la implementación de nuestra solución.

1.1.- Introducción

La institución bancaria para la cuál se desarrolló el proyecto es una de las instituciones líder del Sistema Financiero Venezolano. Dicha institución bancaria fue fundada el 15 de Octubre de 1953, con un capital inicial de Bs.15,000,000[1]. En 1983, la entidad alcanzó el liderazgo en el país, manteniendo una posición destacada en la mayoría de los segmentos en los que se desempeña. En Noviembre de 1996, dicha institución bancaria se convierte en el primer banco universal del país, ampliando su enfoque de negocios para incluir actividades propias de la banca especializada. En 1997, un grupo bancario de gran envergadura y presencia global, adquiere la mayoría accionaria de esta institución bancaria, como parte de su estrategia de expansión en América Latina, incrementando progresivamente su participación hasta 55.14%.

Actualmente la entidad ha hecho grandes inversiones en mejoras tecnológicas que le permiten hoy ofrecer a sus clientes el acceso al Banco a cualquier hora, desde cualquier lugar y de la manera más conveniente, a través de una de las redes de distribución más extensa del país, la cuál hasta Octubre de 2011, contaba con 314 oficinas y 1,042 cajeros automáticos, así como una novedosa red de atención telefónica y el acceso a servicios de *Home Banking* vía Internet.

A lo largo de los años dicha institución bancaria ha ido evolucionando y revolucionando la manera en cómo se obtiene al máximo beneficio en el uso de los servicios financieros. Hoy en día la manera más ágil y rápida para que un usuario pueda consular su saldo, hacer una transferencia, pagar impuestos o servicios, es mediante una aplicación web a la que accede a través de un navegador, desde la comodidad de su casa u oficina, o incluso por medio de un teléfono inteligente con acceso a internet, sin necesidad de desplazarse físicamente a una sucursal bancaria para ser atendidos.

A partir del surgimiento y uso de estas tecnologías web dentro de la industria bancaria se han presentado varios eventos desafortunados que han logrado vulnerar la seguridad de los sistemas informáticos, entre los que se encuentran principalmente robos de identidad, incluyendo números de tarjeta y/o pin, que han costado millones de dólares, con lo cuál se han dado a conocer públicamente algunas vulnerabilidades muy particulares de algunos productos y la pobre seguridad con la que muchos sistemas informáticos han sido desarrollados. A raíz de estos eventos, la institución bancaria se ha preocupado por la creación de una área específica responsable de la administración de la seguridad de la información, en donde las principales actividades tienen como objetivo primordial, proteger la información de accesos no autorizados,

[1] *El símbolo Bs. hace referencia a la moneda oficial de la República Bolivariana de Venezuela*

que puedan derivar en el mal uso, revelación al público, disrupción, modificación, registro, robo, y destrucción de la misma.

Hoy en día la institución bancaria cuenta con un área perteneciente al departamento de informática, llamada seguridad lógica, cuya misión es salvaguardar el activo más importante del banco, que es la información. Cabe mencionar que existen 3 aspectos básicos que fungen como marco de trabajo para la seguridad informática (Andress, 2011), (Peltier *et al*, 2005), los cuales se mencionan a continuación:

- Confidencialidad.- Es el término usado para prevenir la revelación o publicación de información a individuos, entidades, o procesos no autorizados. Es decir, visto desde otra óptica, significa asegurar que la información sólo puede ser accedida por aquellas entidades autorizadas.
- Integridad.- Se refiere a que la información no debe ser modificada por agentes ajenos al conjunto de elementos autorizados. Es decir, se debe tener la certeza de que la información permanece sin alteraciones no autorizadas, y que es 100% confiable y legítima.
- Disponibilidad.- Significa que la información debe estar disponible cuando es requerida, es decir, bajo un modelo de demanda, ya que el valor de la misma radica precisamente en la influencia que puede tener al momento de tomar decisiones cuando esta es solicitada.

Bajo este contexto resultaba de gran importancia que el servicio de la banca en línea contara con altos niveles de seguridad para los usuarios, y con ello, de manera inherente surgían dos elementos críticos que se mencionan a continuación.

- Autenticación.- Es el proceso por medio del cuál se verifica la identidad digital de un usuario a través de elementos únicos, como son, nombres de usuario, números de tarjeta, contraseñas, *tokens*, etc.
- Autorización.- Es el proceso por medio del cuál se autoriza el acceso a determinados recursos para un usuario que previamente ha sido identificado.

1.2.- Descripción de la infraestructura original

La arquitectura original de autenticación y autorización del servicio de la banca en línea de la institución bancaria en la que se desarrolló el trabajo, misma que en lo sucesivo será referida únicamente como el cliente fue originalmente implantada en Septiembre del año 2006, y con el paso del tiempo fue necesario ejecutar de manera constante algunas actualizaciones hacia versiones más recientes de los productos, así como la aplicación de parches para diferentes componentes. Sin embargo, a pesar de que en todo momento se hizo lo posible por mantener actualizada la infraestructura, para Octubre de 2010 ya habían pasado 4 años y la arquitectura era obsoleta e insuficiente para las cargas de trabajo que se ejecutaban en ese momento.

Evidentemente el personal que operaba la solución contaba con el conocimiento suficiente sobre la plataforma y existía cierta resistencia a un eventual cambio, pero desafortunadamente para ellos, algunos de los componentes de la infraestructura de hardware y software que soportaba el servicio de autenticación y autorización estaban incluso a punto de llegar al fin de vida de soporte, creando un alto riesgo de que en cualquier momento presentaran fallas graves que impidieran la disponibilidad del servicio. A continuación se presentan los elementos más importantes que

conformaban la capa de autenticación y autorización:

- Módulo de Autenticación en Java.- Este módulo era considerado como el núcleo de la capa de autenticación y formaba parte de la aplicación principal del servicio de la banca en línea. Estaba desarrollado sobre la plataforma *Java EE 5.0*. Dicho módulo se encontraba integrado dentro del archivo EAR (*Enterprise ARchive*) que era desplegado en el servidor de aplicaciones *IBM WebSphere Application Server*, y su función primordial era la ejecución de rutinas lógicas de autenticación, a través de credenciales que básicamente se componían de un nombre de usuario y una contraseña. Este módulo contaba con su propio repositorio de usuarios, el cuál estaba sincronizado con el software *IBM Tivoli Directory Server.*
- Servidor de Aplicaciones.- El servidor de aplicaciones utilizado era *IBM WebSphere Application Server v7.0.* Este componente era utilizado para el despliegue de la aplicación principal del servicio de la banca en línea, y era el primer punto de contacto para el usuario después de pasar a través del *firewall* y de los balanceadores de carga web. El servidor de aplicaciones se encontraba configurado en modo *cluster,* el cuál estaba conformado por 4 nodos, cada uno de ellos con una instancia de la aplicación, de tal manera que todas las peticiones de los usuarios eran distribuidas entre los 4 nodos y en caso de que alguno de ellos sufriera algún percance, todas las conexiones eran redistribuidas entre los nodos restantes. El balanceo de carga hacia los 4 nodos se hacía a través de un par de balanceadores web marca *Cisco.*
- *Mainframe*.- Este componente era una parte fundamental dentro del núcleo del proceso de autenticación. El equipo *mainframe* al que nos referimos es un *IBM System z9,* y este albergaba una serie de aplicaciones de diversa índole y servicios, entre los cuales se encontraban los componentes *RACF*, *SAF*, y *ITDS*, todos ellos responsables de interactuar con el módulo de autenticación para gestionar y controlar el acceso de los usuarios, así como la asignación de los recursos, dependiendo del nivel de autorización.
- *Resource Access Control Facility* (*RACF*).- Es un producto de software de la marca *IBM*, que provee el control de accesos por medio de la identificación y verificación de usuarios a través de certificados digitales, nombres de usuario y contraseñas. Es el encargado de garantizar la protección de los recursos y usa el repositorio de usuarios *IBM Tivoli Directory Server.*
- *System Authorization Facility* (*SAF*).- Es un producto de software marca *IBM*, que funciona como una interfaz definida que permite la integración del módulo de autenticación con el software *RACF*. Esta basado en mapas de perfiles que permiten controlar el nivel de acceso a ciertos recursos. Este componente también utiliza el repositorio de usuarios *IBM Tivoli Directory Server.*
- *IBM Tivoli Directory Server* (*ITDS*).- Es un producto de software de marca *IBM*, que en aquél momento se encontraba en el *Release 5.2*. Basicamente es una implementación del protocolo *Lightweight Directory Access Protocol*, mejor conocido como *LDAP* por sus siglas en inglés, basado en el modelo X.500, el cuál se encuentra bajo estándares completamente abiertos, y cuyo repositorio de usuarios se encuentra sobre una base de datos *IBM DB2*. Como se ha mencionado, para este caso en particular el producto se encontraba instalado sobre *mainframe.*
- *IBM System x3650.*- Es un producto de hardware y en aquél momento habían 4 servidores de este modelo, los cuales eran utilizados para albergar los servidores de aplicaciones. En cada uno de estos servidores se ejecutaba una instancia de la aplicación, ya que como se menciono anteriormente estaban configuradas bajo un esquema de alta disponibilidad.
- *Cisco Application Control Engine.*- Es un switch balanceador capa 7 para

protocolo *http/https*. En aquél momento existían 2 dispositivos de estos, cuya función principal era repartir y distribuir la carga de peticiones entre los 4 nodos del *cluster* del servidor de aplicaciones *IBM WebSphere Application Server*. El algoritmo utilizado para lograr dicha finalidad era *round-robin.*

A continuación se presenta la figura 1.1 que muestra un diagrama conceptual, incluyendo la estructura lógica y la distribución de componentes de la solución de autenticación y autorización que se tenía previamente:

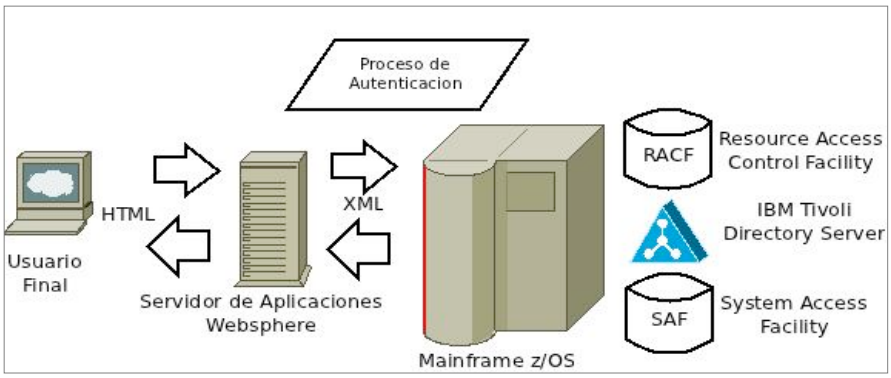

Figura 1.1 Diseño conceptual de la solución original

La imagen anterior muestra la distribución lógica de cada uno de los elementos encargados de la ejecución del flujo de autenticación y autorización de los usuarios. Sin embargo no se muestra la distribución física de dichos elementos, para lo cuál a continuación se presenta la figura 1.2, que presenta un diagrama en el que se puede apreciar como se encontraba físicamente la topología de la infraestructura utilizada para la capa de autenticación y autorización.

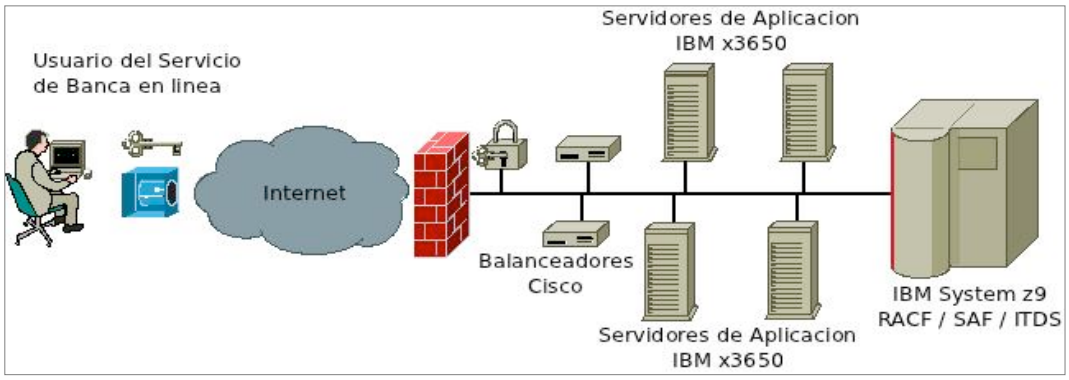

Figura 1.2 Diseño físico de la solución original

Esta arquitectura estaba considerada como altamente disponible, gracias a los componentes redundantes que incluía en sus diferentes capas. Sin embargo, para finales de 2010 el banco ya ofrecía una amplia gama de servicios por Internet, los cuales se habían venido construyendo con el paso del tiempo y ahora estaban empaquetados dentro de un solo servicio integral llamado *"servicio de banca en línea por Internet"*, que en lo sucesivo será referido únicamente como *BNET,* impactando directamente en los niveles de desempeño, ya que esta situación evidentemente generaba más carga transaccional en el sistema de *BNET* y como se ha venido

mencionando repetidamente, este servicio cobraba gran importancia, ya que muchos clientes podían efectuar sus operaciones bancarias de forma remota, sin necesidad de trasladarse físicamente a una sucursal.

1.3.- Niveles de servicio y disponibilidad

La disponibilidad de este servicio tomaba un papel sumamente importante y bajo ninguna circunstancia debía estar fuera de línea. Para ello previamente el cliente había definido que el nivel de disponibilidad hacía los usuarios del servicio debía ser de al menos 99.999%, considerando que la disponibilidad de un sistema es una medida de tiempo durante la cuál un servicio opera de manera normal, o visto de otra forma, es una medida de tiempo requerida por un proceso de recuperación después de que un sistema ha fallado (Office of Government Commerce, 2007b). Bajo este contexto cabe señalar que existen 2 factores que intervienen en la definición de disponibilidad:

- *Mean Time Between Failure* (*MTBF*).- Es el tiempo durante el cuál un sistema opera de manera ininterrumpida.
- *Mean Time To Repair* (*MTTR*).- Es el tiempo requerido para reparar un sistema, después de que ha ocurrido una falla.

A partir de los conceptos anteriores, la disponibilidad se define como sigue a continuación:

$$Disponibilidad = MTBF / (MTBF + MTTR)$$

De la expresión anterior se deduce que es deseable que el valor de *MTBF* tenga un valor tan grande como sea posible, ya que cuanto mayor sea el valor de *MTBF* se minimiza el impacto de *MTTR* y el nivel de disponibilidad se acercar más al 100%.

Para poder obtener el 99.999% de disponibilidad en el servicio final hacía el usuario era necesario que la suma de todos y cada una de los componentes de solución, dividida entre el número de componentes, resultara 99.999% de disponibilidad en el servicio, lo cuál significa que cada componente de la solución debía otorgar al menos un 99.999%, es decir:

Sea D = Disponibilidad
N = Número de Componentes

$\Rightarrow$ *D = (Disponibilidad Firewall + Disponibilidad Balanceadores + Disponibilidad Servidor de Apps + Disponibilidad Mainframe + Disponibilidad RACF + Disponibilidad SAF + Disponibilidad ITDS + Disponibilidad Almacenamiento + Disponibilidad Redes) / 9*

En seguida se presenta la figura 1.3 que muestra el nivel de disponibilidad que debía tener cada uno de los componentes del servicio de la *BNET*:

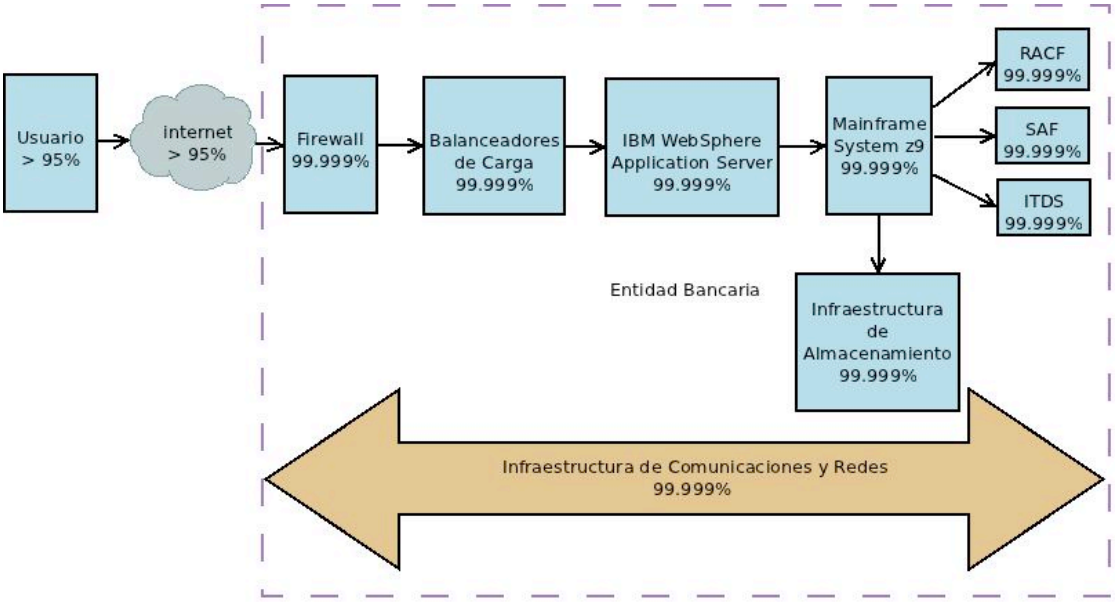

Figura 1.3 Nivel de servicio acordado para *BNET*

De lo anterior se deduce que la solución de autenticación y autorización propuesta debía tener un nivel de disponibilidad no menor 99.999% para conservar los niveles de disponibilidad comprometidos para el servicio completo.

A continuación se muestran algunos de los principales servicios contenidos y empaquetados dentro del catalogo de servicios de *BNET*:

- Servicio de consulta de saldos.
- Servicio de pago de tarjeta de crédito.
- Servicio de transferencias entre diferentes cuentas.
- Servicio de transferencias interbancarias.
- Servicio de recargas telefónicas.
- Servicio de pago de impuestos.
- Servicio de pago de otros servicios como agua, energía eléctrica, gas, telefonía, y cable.

Cada uno de estos servicios individuales ya contaba con cierta madurez en su funcionalidad, incluyendo rutinas y procedimientos. Los flujos de los procesos asociados a dichos servicios estaban perfectamente probados y habían sido sometidos a varias fases de aseguramiento de calidad de manera continua, lo que le daba cierta solidez y confiabilidad a la arquitectura que se encontraba funcionando en aquél momento. Evidentemente, el servicio de autenticación y autorización gozaba de los mismos beneficios, al ser parte integral y fundamental del servicio de la banca en línea, sin embargo el gran problema no era la funcionalidad en sí misma, sino más bien el desempeño de los componentes que conformaban la arquitectura que brindaba el servicio. La manera en como se obtenían los indicadores de desempeño era a través de algunas herramientas propietarias, como es *Compuware Gomez Real-User Monitoring*[2], que es un *appliance* de monitoreo de aplicaciones, el cuál tiene capacidad de generar estadísticas a partir de datos extraídos, referentes a los niveles de utilización.

[2] *Compuware Gomez Real-User Monitoring es uno de los líderes en el mercado de monitoreo de aplicaciones en tiempo real. Para mayor información consultar la siguiente liga de internet http://www.compuware.com/application-performance-management.*

Originalmente el proceso de autenticación y autorización se ejecutaba a través del módulo de la aplicación web, siendo este el principal actor, en conjunto con el sistema *mainframe*. Dicho módulo de autenticación tuvo un buen desempeño durante sus primeros años, sin embargo en los últimos tiempos había perdido viabilidad debido a que fue construido con una capacidad máxima para atender un volumen aproximado de 1,700 usuarios concurrentes, sin sufrir degradación en su desempeño. Por otro lado, se observó que durante 2009 y 2010 el número de usuarios del servicio de la banca en línea se había incrementado de forma exponencial y el registro indicaba que particularmente en el mes de Octubre de 2010 el universo total ascendía a 3,500,000 usuarios, considerando que en promedio se tenía una concurrencia de 1,800 conexiones y un tiempo de respuesta promedio de 3.17 segundos durante los días normales de operación en un horario de 7:00 am a 9:00 pm, pero evidentemente existían días en que la concurrencia se incrementaba demasiado, teniendo picos de casi el doble de sesiones, lo cuál significaba cerca de 3,600 usuarios conectados de manera simultanea y con tiempos de respuesta de hasta 10 segundos, dicha concurrencia fue el límite máximo que se logró registrar, ya que cuando se alcanzaba esta cifra, posteriormente el sistema comenzaba a presentar mensajes de error en las pantallas de los usuarios, mandando *ERROR 404 – NOT FOUND*[3] o de *timeout,* entre otros, y en seguida se colapsaba, congelando todas las sesiones que se tenían en ese instante. Lógicamente los tiempos de respuesta en el servicio de autenticación y autorización, y de las aplicaciones en general fueron severamente afectados, causando molestia en los usuarios finales del servicio y generando una alta incidencia en las llamadas al centro de atención de usuarios del servicio, mejor conocido como "primera línea", la cuál básicamente funcionaba como mesa de ayuda. Este escenario generaba gran tensión debido a la posible e inminente pérdida de usuarios disgustados.

Estaba claro que los niveles de servicio acordados, mejor conocido como *SLA*, previamente definidos por el cliente con un valor de 99.999%, no se estaban cumpliendo. La tolerancia máxima permitida de acuerdo a dicho *SLA* era de 5 minutos con 15.3 segundos por año con el servicio fuera de línea. Desafortunadamente debido a la presión ejercida por diferentes áreas de negocio se provocó a su vez que las diferentes áreas del departamento informática sometieran sus componentes a una revisión exhaustiva con el apoyo de los fabricantes de los diferentes productos que integraban la solución original, e incluso se hicieran reconfiguraciones, y algunos cambios de acuerdo a las mejores prácticas que dictaba cada fabricante para cada componente. Esta situación inevitablemente implicaba tener ventanas de tiempo para la ejecución de las actividades de mantenimiento, que en su mayoría se realizaron durante las noches y en fines de semana. Solamente durante el mes de Octubre de 2010 se tuvo un tiempo aproximado de 4 horas totales con el servicio no disponible, resultado de la suma de tiempos dedicados a eventos de mantenimiento, intentando optimizar la plataforma con el afán de mejorar los niveles de servicio y lograr estabilidad en el mismo. Desafortunadamente todos estos esfuerzos concluyeron sin éxito, al no lograr la mejora deseada en la calidad del servicio.

A continuación, en la siguiente página se presenta la tabla 1.1, que muestra una relación con las ventanas de tiempo utilizadas:

[3] *El mensaje ERROR 404 es un código de respuesta estándar que indica que el navegador ha sido capaz de comunicarse con el servidor, pero no existe el recurso solicitado.*

Octubre 2010				
Día	**Inicio**	**Término**	**Duración**	**Actividad**
Sab 2	00:30 hr	1:30 hr	1 hora	Afinación a Sistemas Operativos
Dom 9	01:00 hr	2:30 hr	1.5 horas	Afinación y Aplicación de parches a los componentes del *Mainframe*
Dom 23	01:30 hr	3:00 hr	1.5 horas	Afinación y re-parametrización de las instancias del *IBM WebSphere Application Server*

Tabla 1.1 Ventanas de tiempo para optimizar el rendimiento de *BNET*

Con la intención de clarificar el origen del valor para el *SLA* del servicio de *BNET*, a continuación se muestra una expresión a partir de la cuál se obtiene el valor de tolerancia máxima permitida:

Sabemos que: 1 Año = 365 días = 8760 horas, lo que equivale al 100% del tiempo por año.

Así que: $100\% / 8760 \text{ horas} = 99.999\% / \chi$

$\Rightarrow$ $\chi = ((99.999\%)*(8760 \text{ horas})) / 100\%$

$\chi = 8759.91 \text{ horas}$

Tiempo Fuera = 8760 horas – 8759.91 horas

Tiempo Fuera en minutos = 0.876(60 minutos) = 5.256

Por otro lado: *Segundos = 0.256 (60)*

$\Rightarrow$ ***Tiempo Fuera Permitido = 5 minutos 15.3 segundos***

A continuación se presenta la figura 1.4 que muestra una gráfica poligonal con los niveles de desempeño medidos en unidades de tiempo durante el mes de Octubre de 2010. Se presume que este comportamiento puede ser extrapolado al desempeño observado durante el penúltimo trimestre de 2010.

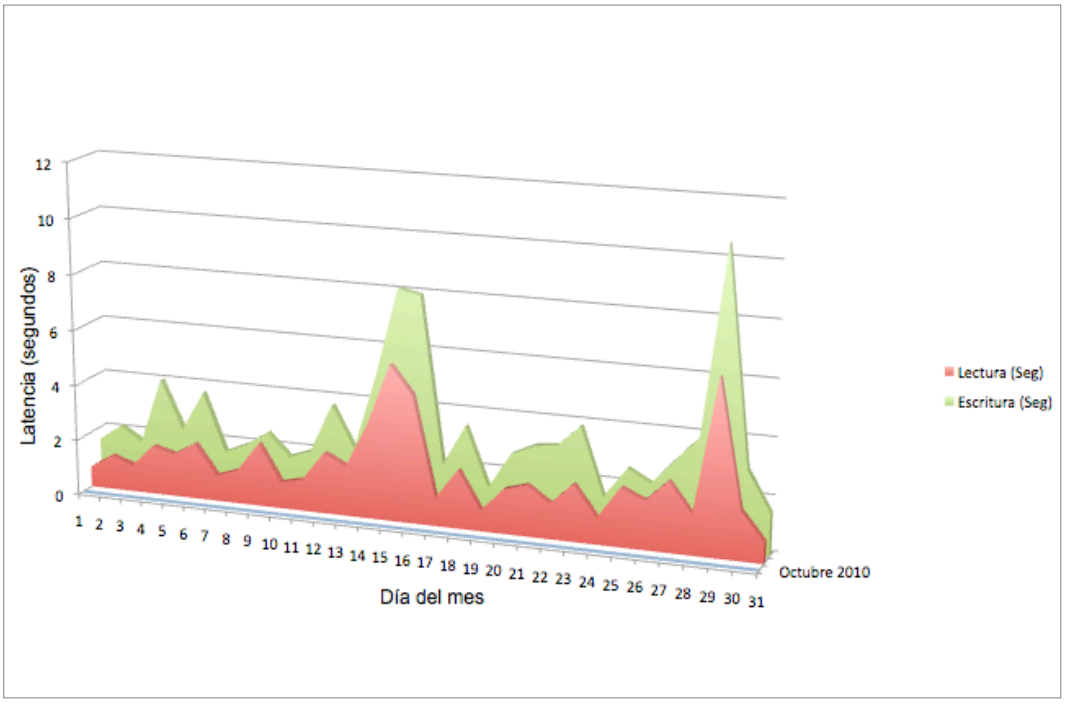

Figura 1.4 Gráfica de indicadores clave de desempeño de la solución original

La gráfica anterior se generó a partir de los reportes obtenidos desde la consola de monitoreo *Compuware Gomez Real-User Monitoring*, también conocido como *Agentless*. En dicha gráfica se pueden observar 2 series de datos representando los flujos de rendimiento con diferentes comportamientos que corresponden a los tiempos de respuesta expresados en segundos para las operaciones de lectura y escritura respectivamente dentro de la capa de autenticación y autorización. A continuación se muestra la tabla 1.2 que contiene las series de datos correspondientes a los tiempos de respuesta observados para las operaciones de lectura y escritura, a partir de las cuales se construyó la gráfica de la figura 1.4.

Fecha	Lectura (Seg)	Escritura (Seg)
1-Octubre-2010	0.75	1.56
2-Octubre-2010	1.23	2.12
3-Octubre-2010	0.96	1.61
4-Octubre-2010	1.75	3.98
5-Octubre-2010	1.54	2.24
6-Octubre-2010	2.01	3.66
7-Octubre-2010	0.94	1.58
8-Octubre-2010	1.21	1.92
9-Octubre-2010	2.27	2.45
10-Octubre-2010	0.95	1.65
11-Octubre-2010	1.14	1.95
12-Octubre-2010	2.19	3.7
13-Octubre-2010	1.78	2.2
14-Octubre-2010	3.43	4.87
15-Octubre-2010	5.58	8.06
16-Octubre-2010	4.57	7.89
17-Octubre-2010	1.01	1.98
18-Octubre-2010	2.1	3.44
19-Octubre-2010	0.75	1.33
20-Octubre-2010	1.59	2.6
21-Octubre-2010	1.83	2.99
22-Octubre-2010	1.26	3.08
23-Octubre-2010	2.05	3.84
24-Octubre-2010	0.95	1.45
25-Octubre-2010	2.1	2.53
26-Octubre-2010	1.73	2.09
27-Octubre-2010	2.51	3
28-Octubre-2010	1.44	3.86
29-Octubre-2010	6.25	10.49
30-Octubre-2010	1.76	2.92
31-Octubre-2010	0.75	1.5

Tabla 1.2 Tiempos de respuesta previos a la implementación

La tabla anterior sirve como línea de base para calcular el promedio de los tiempos de respuesta para las operaciones de lectura resultando una cifra igual a 1.947742 segundos, mientras que para las operaciones de escritura se obtiene un promedio de 3.1787097 segundos.

Es importante aclarar que las operaciones de lectura y escritura dentro del ámbito de los flujos de autenticación y autorización para el entorno del servicio de la banca en línea, objeto de estudio del presente documento, se definen como sigue a continuación:

- Operación de Lectura.- Se refiere a una operación que se presenta cada vez que un usuario desea autenticarse a través de sus credenciales, como son nombre de usuario y contraseña, ya que estos valores son leídos y buscados dentro de los repositorios de usuarios del sistema. Una vez que un usuario ha sido autenticado exitosamente, se buscan los perfiles que tiene asignado dicho usuario, para posteriormente otorgar los respectivos privilegios dentro del sistema.
- Escritura.- Es una operación que se genera bajo 5 posibles circunstancias:
    - Cuando un usuario se registra en el servicio de *BNET*, este ingresa sus datos, tales como nombre de usuario, contraseña, pregunta secreta, fecha de nacimiento, etc. y posteriormente se insertan en la base de datos que conforma el repositorio de usuarios.
    - Cuando un usuario decide cambiar voluntariamente su contraseña o su pregunta secreta para evitar el envejecimiento de credenciales, automáticamente se ejecuta una modificación dentro del repositorio de usuarios.
    - Cuando un usuario olvida su contraseña o número de tarjeta y el sistema lo obliga automáticamente a actualizar sus datos.
    - Cuando el sistema bloquea automáticamente una cuenta como resultado de varios intentos equivocados de contraseña.
    - Cuando un usuario administrador da de baja o elimina un usuario del servicio de *BNET.*

Las operaciones de escritura resultan más demandantes en recursos de E/S y requieren más tiempo para ejecutarse. Así que lógicamente para las operaciones de escritura se tienen tiempos de respuesta más elevados que para las operaciones de lectura, como se puede apreciar claramente en la gráfica de la figura 1.4.

Como se ha venido mencionando, el número promedio de sesiones de usuarios concurrentes para el mes de Octubre de 2010 fue de 1,800, tomando en cuenta que los picos máximos fueron de 3,500. En seguida se muestra la figura 1.5. donde se puede observar el comportamiento con el nivel de variación que se presentó:

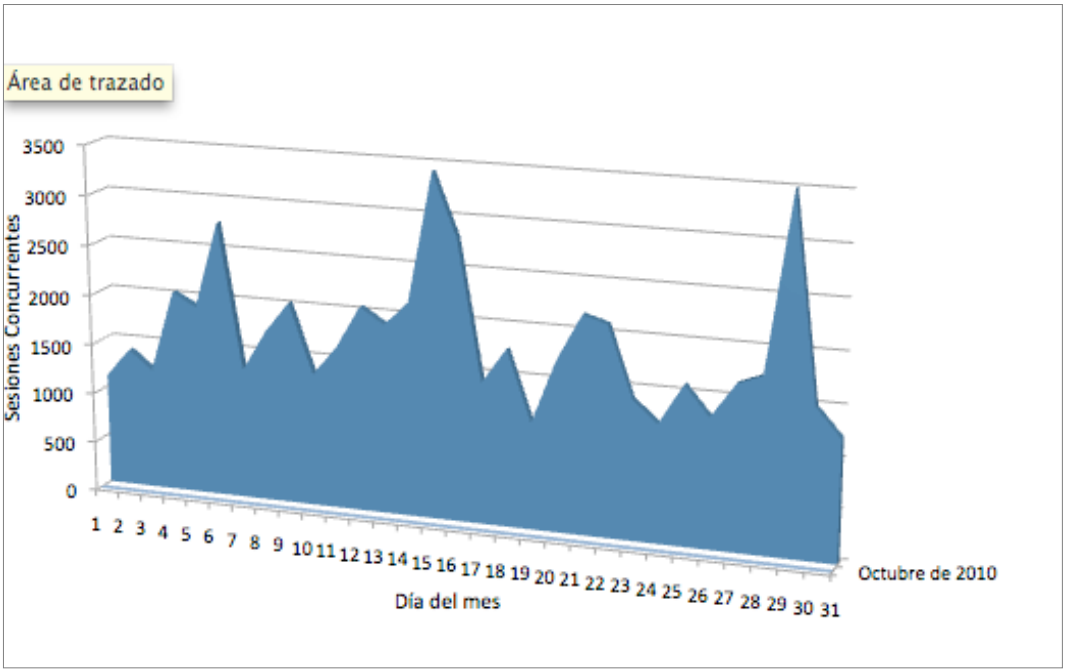

Figura 1.5 Gráfica de concurrencia de usuarios sobre la solución original

A partir de la gráfica anterior se puede determinar que los días con mayor actividad eran los días en que justamente los tiempos de respuesta eran más elevados, llegando al límite de sesiones concurrentes registradas.

Debido a los niveles de desempeño que se observaban en el servicio de la banca en línea en aquél momento, constantemente los usuarios llamaban al número telefónico de servicio en el Centro de Atención a Usuarios (CAU), también conocido como "Primera Línea" para reportar incidentes y problemas, que en su mayoría estaban asociados a la capa de autenticación y autorización.

Esta situación, como se menciono anteriormente, ya había generado incomodidad y tensión entre las diferentes áreas pertenecientes al departamento de informática, además de que el personal responsable de la administración y operación de la plataforma del servicio de la *BNET* estaba sufriendo demasiado desgaste físico a causa de constantes desveladas y jornadas de trabajo de más de 15 horas, incluyendo algunos fines de semana, todo ello con el afán de resolver los problemas que se presentaban, así que finalmente se extrajo un reporte de la información generada a través del sistema de mesa de ayuda y gestión de incidentes y problemas, *BMC Remedy Action Request System*[4], para ser analizado y determinar que alrededor del 65% de la totalidad de los tickets generados durante Octubre de 2010, eran *tickets* relacionados con la capa de autenticación y autorización, directamente asociados a fallas y errores atribuibles a los componentes de la misma. Así mismo se logró observar que el porcentaje promedio de utilización de *CPU* era de un 78%, mientras que para la memoria *RAM* era del 85%. Una vez que el reporte había sido revisado y analizado por el área de seguridad lógica, se determinó que esta situación era alarmante y estaba costando al banco mucho esfuerzo en tiempo y dinero, por lo que se convocó a los gerentes responsables de las diferentes áreas del departamento de informática para solicitar su apoyo e impulsar internamente la solicitud de recursos financieros para la ejecución de un nuevo proyecto, cuyo alcance sería la sustitución de la infraestructura de la capa de autenticación y autorización.

1.4.- Solicitud de propuesta

Después de 2 semanas de haber analizado un reporte con los resultados del rendimiento durante el mes de Octubre de 2010, se redacto un documento con la descripción del impacto negativo generado para el negocio. Dicho documento sería integrado como parte del caso de negocio para la justificación del proyecto, y una vez que se tenía completo, este fue presentado ante el comité de adquisiciones para posteriormente generar un documento formal de solicitud de propuesta, mejor conocido como *RFP,* en el que se solicitaba formalmente una propuesta para el proyecto del dominio seguro, describiendo el alcance del mismo. Es importante señalar que gracias a la urgencia por iniciar un proyecto para atender el problema, afortunadamente no fue necesario que el cliente lanzara una convocatoria para recibir propuestas de diferentes proveedores y someterlas a un proceso de evaluación, ya que esto probablemente hubiera retrasado el arranque del proyecto.

Finalmente el cliente estaba listo para liberar el *RFP,* pero antes contactó a un área

[4] *Es un sistema que provee una plataforma consolidada para la administración del servicio que automatiza y administra los procesos de negocio de la propia gestión del servicio. Para mayor información, consultar la siguiente liga de internet http://www.bmc.com/products/product-listing/remedy-action-request-system.html*

llamada Centro Corporativo Regional, mejor como *CCR*, perteneciente al área de Tecnologías de la Información dentro de la filial con base en México, cuya función principal en aquél momento era proporcionar servicios de soporte técnico y asesoría en materia de TI a sus áreas homólogas en todos los países de Latinoamérica dónde el grupo financiero tenía presencia. Es importante mencionar que desde algunos años atrás hasta ese momento, México había sido la filial más grande del grupo en todos los ámbitos de negocio, el de mayor volumen de usuarios, y el que iba a la vanguardia en tecnologías de información (sólo después de su filial en España), razón por la cuál, el personal del cliente, quién aún contaba con muy poca experiencia en proyectos de esta naturaleza, solicitó el apoyo incondicional a su homólogo en México para adquirir el liderazgo y la responsabilidad necesarios para la revisión detallada de todos y cada uno de los puntos expresados en el documento de *RFP*. Así mismo el personal de *CCR* estaría fungiendo como una instancia intermediaria para la negociación y evaluación del proyecto en sus diferentes fases. Esto significaba que *CCR* jugaría un papel como autoridad tomadora de decisiones en un segundo nivel, cuando la situación lo ameritara, pero al mismo tiempo permitiendo al cliente conservar la autonomía necesaria para administrar el desarrollo del ciclo del proyecto.

Posteriormente *CCR* se puso en contacto con la empresa para la cuál el autor de esta memoria trabajaba en aquél momento, que en lo sucesivo será referida únicamente como la consultoría, misma que ya le había prestado servicios profesionales en materia de seguridad informática con anterioridad, y en la cuál el grupo financiero en México y *CCR* depositaban su confianza plenamente gracias a la experiencia y a los antecedentes satisfactorios en otros proyectos, pero principalmente gracias a la relación entre la consultoría y la filial del grupo financiero en España, en donde había un gran vínculo de confianza y una serie de proyectos exitosos. Particularmente el personal de *CCR* convocó a una reunión cuyo propósito principal era explicar la manera en como se debían contestar las diferentes secciones del documento de respuesta al *RFP* y disipar cualquier duda relacionada con el formato de la propuesta de solución.

1.5.- Estructura organizacional del cliente

Con la finalidad de mostrar la estructura jerárquica de las áreas involucradas por parte del cliente dentro del proyecto, es importante mencionar que existen 2 grandes entidades consideradas como consumidores de los servicios que provee el área de seguridad lógica, misma que fue la responsable de la ejecución del proyecto de dominio seguro por parte del personal del mismo cliente. El primer consumidor de servicios es el universo exterior de usuarios del servicio de la *BNET*, considerado como el más importante por ser una fuente natural de ingresos para el cliente, mientras que la segunda entidad consumidora de servicios es propiamente el departamento de informática del cliente, la cuál por el hecho de ser un consumidor interno se encuentra asociada a los costos de operación, sin embargo no deja de tener importancia y criticidad. En la figura 1.6 se presenta un diagrama que muestra la relación de las entidades que interactúan con el área de seguridad lógica:

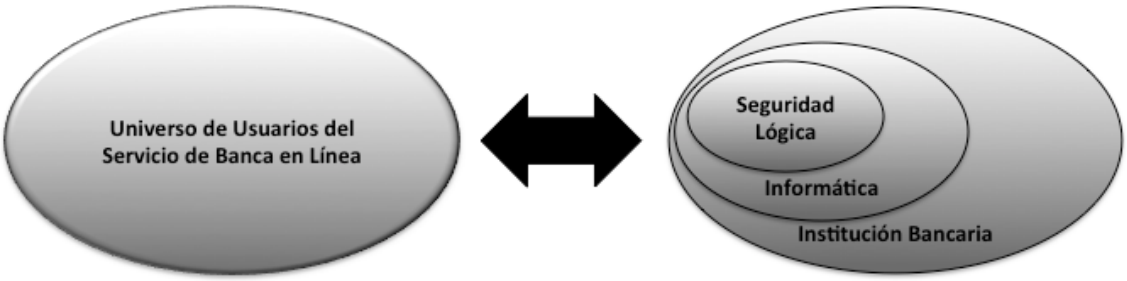

Figura 1.6 Relación de entidades con el área de seguridad lógica

El conjunto etiquetado como el universo total de usuarios, se refiere a todos aquellos usuarios que hacen uso del servicio de la *BNET*, incluyendo personas físicas y empresas que ingresan al portal desde cualquier lugar en cualquier momento.

El departamento de informática se encuentra integrado de forma general por 4 grandes unidades de operación, las cuales se describen a continuación:

- Seguridad lógica.- Responsable de la administración y operación de toda la infraestructura de seguridad informática, incluyendo la gestión e implantación de nuevos proyectos relacionados con temas de seguridad de la información en sus diferentes ámbitos.
- Infraestructura.- Responsable de la administración, operación, soporte y adquisición de la infraestructura de hardware, incluyendo equipamiento de procesamiento, almacenamiento, redes, y respaldos, así como también algunos elementos de software, como son sistemas operativos, servidores de aplicaciones, servidores web, motores de base de datos, etc.
- Diseño y Desarrollo.- Responsable del diseño, desarrollo, liberación, y soporte de aplicaciones, incluyendo la gestión y operación para todos los aplicativos desarrollados por personal del banco.
- Redes.- Responsable de la diseño, implantación, administración, operación, soporte, y monitoreo de las redes del banco en todos sus niveles, desde la capa física hasta la capa de aplicación dentro del modelo de *TCP/IP.*

En la figura 1.7 se muestra la jerarquía de las 4 unidades de operación que conforman el departamento de informática dentro de la estructura organizacional del cliente:

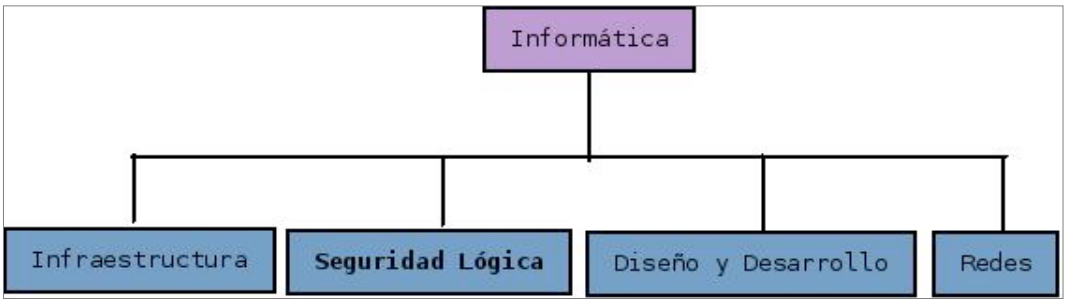

Figura 1.7 Estructura organizacional del departamento de informática

Es importante señalar que la estructura jerárquica del departamento de informática, al igual que las funciones que desempeñaba cada una de sus diferentes áreas, e incluso el uso de algunos productos y desarrollos propios, se encontraban homologados en todos los países en donde el grupo financiero tenía presencia, tomando siempre como marco de referencia los nuevos lineamientos dictaminados e implementados en la filial

de España, por ser este el país dónde surge el corporativo.

El equipo de trabajo que participó en el desarrollo del proyecto en sus diferentes etapas, estaba conformado por personal de la consultoría, y que prácticamente constaba de un servidor y un gerente de proyecto, ya que al inicio se contrató al autor de esta memoria con el objeto de estar asignado el 100% de su tiempo a proyectos con el cliente. Por otro lado, también había personal del área de TI por parte del cliente, mismo que contaba con diferentes especialidades, estos fungían como apoyo, al proporcionar todo lo requerido para la correcta consecución de actividades, al mismo tiempo que se encargarían de la gestión general del proyecto. A continuación se presenta la tabla 1.3 que contiene una matriz general de responsabilidades y muestra al equipo de trabajo involucrado desde las primeras fases del proyecto, desde su concepción:

Actividad	La Consultoría	Personal del Cliente			
		Redes	Infraestructura	Diseño y Desarrollo	Seguridad Lógica
Recopilación de información referente a la plataforma original y análisis de utilización de recursos.	X	X	X	X	X
Dimensionamiento de capacidad requerida para la nueva infraestructura de hardware y validación de compatibilidades.	X				
Proceso de adquisición de infraestructura de hardware y licencias software			X		X
Instalación, configuración y afinación de infraestructura de hardware, y sistema operativo.			X		
Instalación, configuración, y afinación de los componentes de software que conforman el dominio seguro.	X				
Endurecimiento del sistema operativo y la red de comunicaciones para todos los nodos que conforman el dominio seguro.	X				X
Ejecución de pruebas unitarias, pruebas integrales, y liberación a producción.	X	X	X	X	X

Tabla 1.3 Matriz general de responsabilidades

1.6.- Metodologías

Todos los procesos de TI dentro de la operación del cliente se encontraban alineados con las directrices y políticas corporativas del grupo financiero, los cuales a su vez estaban alineados con las buenas practicas que dicta la metodología para la administración del servicio, *ITIL v3.0*, y por lo tanto el desarrollo del proyecto objeto de estudio del presente trabajo, se integra de forma nativa con dicha metodología, la cuál, en lo sucesivo será referida únicamente como *ITIL*. Es importante mencionar, que a pesar de que todos los procesos de TI dentro del banco van de la mano con *ITIL*, para finales de 2010 aún existía falta de madurez en algunos de ellos, y día con día el cliente había venido trabajando de forma permanente y continua para mejorarlos y actualizarlos bajo el modelo de mejora continua del servicio o *CSI*. Cabe mencionar

que, a pesar de que el desarrollo de este proyecto en todo momento guarda una estrecha relación con *ITIL*, las últimas 2 fases del modelo están más orientadas a formar parte de las responsabilidades del personal del cliente, quienes se encargan de la operación del día a día y se encuentran inmersos en el proceso de mejora continua. A continuación se describe cada una de las fases del modelo de *ITIL* (Office of Government Commerce, 2007e) con el objeto de clarificar el detalle de mi participación dentro del proyecto:

- Estrategia.- Como centro y punto de origen del ciclo de vida de un servicio, se proveen las directrices sobre como clarificar y priorizar las inversiones en servicios de TI. De forma general se enfoca en ayudar a las organizaciones de TI a analizar las posibles mejoras a los servicios ya existentes, así como desarrollar nuevos servicios con una visión a largo plazo.
- Diseño.- Una vez que se ha identificado un posible servicio de TI, se debe analizar su viabilidad. Se debe tener en cuenta que cuando se habla de diseño dentro de *ITIL*, se refiere a todos los elementos relacionados con la prestación de un servicio de tecnología, y va mucho más allá del diseño exclusivo de la tecnología en si misma, algunos de los elementos a considerar pueden ser el personal capacitado, un plan de recuperación de desastres, la infraestructura disponible, etc.
- Transición.- Esta fase se relaciona directamente con el despliegue de servicios de TI requeridos por un negocio en estado operacional. Antes de poner en marcha un servicio, se deben realizar pruebas, respaldos necesarios de la información, así como revisar y valorar el estado de la infraestructura, ya que todos ellos son utilizados para la creación de un plan de retorno, que consiste en regresar al estado original que se tenía antes del punto de partida de implementación o cambio de un servicio.
- Operación.- En este punto se monitorea activa y pasivamente el funcionamiento del servicio con el objeto de asegurar la entrega de los niveles de servicio acordados hacía los usuarios. El monitoreo ayuda a la detección de eventos e incidentes, los cuales a su vez ayudan a prevenir y atender posibles problemas, al mismo tiempo que permiten hacer un balance entre la confiabilidad del servicio y sus costos asociados.
- Mejora Continua del Servicio.- El objetivo principal de esta fase consiste en alinear y realinear los servicios de TI, según cambien las necesidades de negocio, identificando e implementando mejoras a los servicios de TI que soportan los procesos de negocio. Para esta fase se utilizan herramientas de medición que ayudan a documentar la información referente al funcionamiento del servicio. Los resultados obtenidos a partir de las mediciones nos brindan retroalimentación y sirven para determinar las posibles causas a problemas ocasionados.

A continuación se presenta la figura 1.8 que muestra un representación gráfica del ciclo de vida del servicio de TI según el marco de trabajo de *ITIL*, donde se aprecia claramente la relación entre cada una de sus fases.

Figura 1.8 Ciclo de vida del servicio de *ITIL v3.0* [5]

Por otro lado, es importante mencionar que durante la ejecución del proyecto algunos de los procesos se mantienen al mismo tiempo alineados con el marco de trabajo *COBIT v4.1,* el cuál consta de 5 áreas de controles internos, todas ellas con un gran énfasis en la gobernanza de TI. A continuación se presenta una breve descripción de las mismas (IT Governance Institute, 2007) y como se relacionan con el desarrollo del proyecto, objeto de estudio del presente documento

- Alineación Estratégica.- Se deben hacer los esfuerzos necesarios para alinear las operaciones y las actividades de TI con todas las operaciones de la empresa a nivel general. Esto incluye asegurar la existencia de enlaces entre el negocio empresarial y los planes de TI, así como definir, mantener, y validar la calidad y el valor de las relaciones entre ellos. En este caso uno de los principales objetivos del cliente era brindar los servicios financieros que como institución bancaria le corresponde, todo ello a través de herramientas tecnológicas que le asegurarían otorgar la agilidad necesaria para los negocios.
- Entrega de Valor.- Se deben establecer procesos para asegurar que las unidades de TI, y otras unidades operativas otorgan los beneficios prometidos a través del ciclo de entrega y favorece una estrategia que optimiza costos, enfatizando los valores intrínsecos de TI y sus actividades relacionadas. El valor principal que entrega un servicio como el de la *BNET* radica en el ahorro de costos en tiempo y dinero para el usuario del servicio, como para la institución bancaria que se ahorra los costos asociados a la apertura de nuevas sucursales.
- Administración de Recursos.- Se debería tener una inversión óptima, y una administración propia de los recursos críticos de TI, aplicaciones, información, infraestructura, y gente. Una gobernanza efectiva de TI depende de la optimización del conocimiento y de la infraestructura. Justamente uno de los

[5] *Office of Government Commerce (2007e: p.19)*

principales activos que poseía la propuesta para el cliente, era una curva de aprendizaje verdaderamente aguda, reduciendo los tiempos de capacitación.

- Administración de Riesgos.- La administración en todos sus niveles debería tener un claro entendimiento del apetito de las empresas por el riesgo, requerimientos de conformidad, y el impacto de riesgos significativos. Las áreas de TI en conjunto con otras unidades de operaciones tienen sus propias responsabilidades compartidas de administración de riesgos que pueden impactar individualmente o en conjunto a una empresa en su totalidad. Dentro de este punto en particular, es importante mencionar que el desarrollo del proyecto del dominio seguro contaba con una serie de procesos orientados a la administración del riesgo y sus procesos de retorno. Principalmente durante la fase de transición es donde se hace particular énfasis en la gestión del riesgo.
- Medición del Desempeño.- Deberían existir procesos para rastrear y monitorear la implementación de la estrategia, la terminación de los proyectos, la utilización de recursos, el desempeño de los procesos, y la entrega del servicio. Los mecanismos de gobernanza de TI deberían ser capaces de traducir las estrategias de implementación en acciones y medidas para lograr estos objetivos. El desarrollo del proyecto tenía como base la medición del desempeño principalmente, ya que evidentemente uno de los principales indicadores clave de desempeño, mejor conocidos como *KPI,* para el servicio de la *BNET* era precisamente el rendimiento de la capa de autenticación y autorización, y por esa misma razón es que surge la necesidad de ejecutar el proyecto del dominio seguro.

En seguida se muestra la figura 1.9 que representa el esquema general del marco de trabajo de *COBIT v4.1*:

Figura 1.9 Marco de trabajo de *COBIT v4.1*[6]

Adicionalmente, es importante mencionar que en todo momento se siguieron los

[6] *IT Governance Institute (2007: p.6)*

lineamientos, las guías y las buenas prácticas que dicta cada uno de los fabricantes de los productos software que forman parte de nuestra solución de dominio seguro basado en *IBM Tivoli Access Manager* y *Sun Java System Directory Server Enterprise Edition 7.0* basado en el estándar de *Lightweight Directory Access Protocol,* que en lo sucesivo será referido únicamente como *LDAP,* que es como conoce comúnmente, todo ello con el objeto de mitigar los posibles riesgos y proveer una solución con los niveles de desempeño esperados, para con esto lograr la entera satisfacción del cliente. Por otro lado, cabe señalar que debido a que el cliente representa una filial que forma parte del grupo financiero, este se encuentra alineado con los estándares más estrictos en materia de seguridad de TI, como son *PCI,* y *FIPS*. Sin embargo la consultoría no estuvo involucrada directamente con la aplicación de dichos estándares y estos temas quedan fuera de alcance, ya que todos los componentes de software que conforman la solución del dominio seguro, al ser de clase empresarial cumplen de forma automática con el 100% de lo que dictan dichas normas, y además se integran completamente a la infraestructura de redes y comunicaciones que en su momento tenía el cliente.

1.7.- Servicio de banca en línea o *BNET*

El enfoque principal que se le da al proyecto va orientado a un esquema de servicios, ya que en realidad todos los procesos asociados al servicio de la *BNET* contaban con características propias de la definición de un servicio (Office of Government Commerce, 2007d). Sin embargo estos servicios que giraban alrededor del servicio de la *BNET,* no tenían la capacidad de entregar valor por si solos, ya que su razón de ser era precisamente brindar las funcionalidades requeridas por el servicio principal, que era *BNET.* Uno de los principales servicios encargado de soportar parte de la seguridad de *BNET* era el servicio de autenticación y autorización, y a continuación sus principales características:

- Medible.- Todos los procesos asociados podían ser medidos fácilmente en términos de su desempeño, básicamente el indicador clave de desempeño más importante era el tiempo que tardaba un usuario en autenticarse dentro del portal del banco. Existen otros indicadores, como es el tiempo en que tarda un usuario en darse de alta dentro del servicio, o el tiempo que tarda para cambiar sus contraseñas, aunque en realidad la frecuencia de estos suele ser mucho más baja.
- Orientado a resultados específicos.- El resultado del proceso de autenticación y autorización puede ser fácilmente identificable y cuantificable. Basta con observar el número de usuarios autenticados y/o rechazados, los cuales debieron seguir de forma correcta los flujos dentro del sistema de *BNET*. El resultado final solo puede tener 2 estados, que son exitoso o no exitoso.
- Orientado al cliente.- Cada proceso entrega resultados específicos de valor para el usuario

El propósito principal del proyecto fue la sustitución de la infraestructura encargada de proporcionar el servicio de autenticación y autorización para los usuarios de *BNET*. Dicha sustitución no podía ser conceptualizada de forma aislada, ya que dentro del entorno completo se consideraba como un cambio que afectaría el servicio de la *BNET* de forma general, impactando directamente en la capa de negocio, por lo que existía un riesgo alto al manipular cualquier componente dentro de la plataforma del servicio.

En la figura 1.10 se muestra un diagrama, cuyo objetivo principal es dar a conocer el alcance del servicio de autenticación y autorización, y la manera en como se relaciona

con otros servicios y procesos dentro del modelo de servicios del cliente implantado en aquél momento. Se puede apreciar con claridad la importancia y criticidad de dicho componente dentro del entorno completo, por ser el punto de control de acceso para posteriormente poder otorgar los privilegios requeridos para accionar cada uno de los servicios financieros. Cabe mencionar que en este diagrama, como en los anteriores no se intenta establecer una notación o patrón específico asociado a alguna metodología de diagramación en particular.

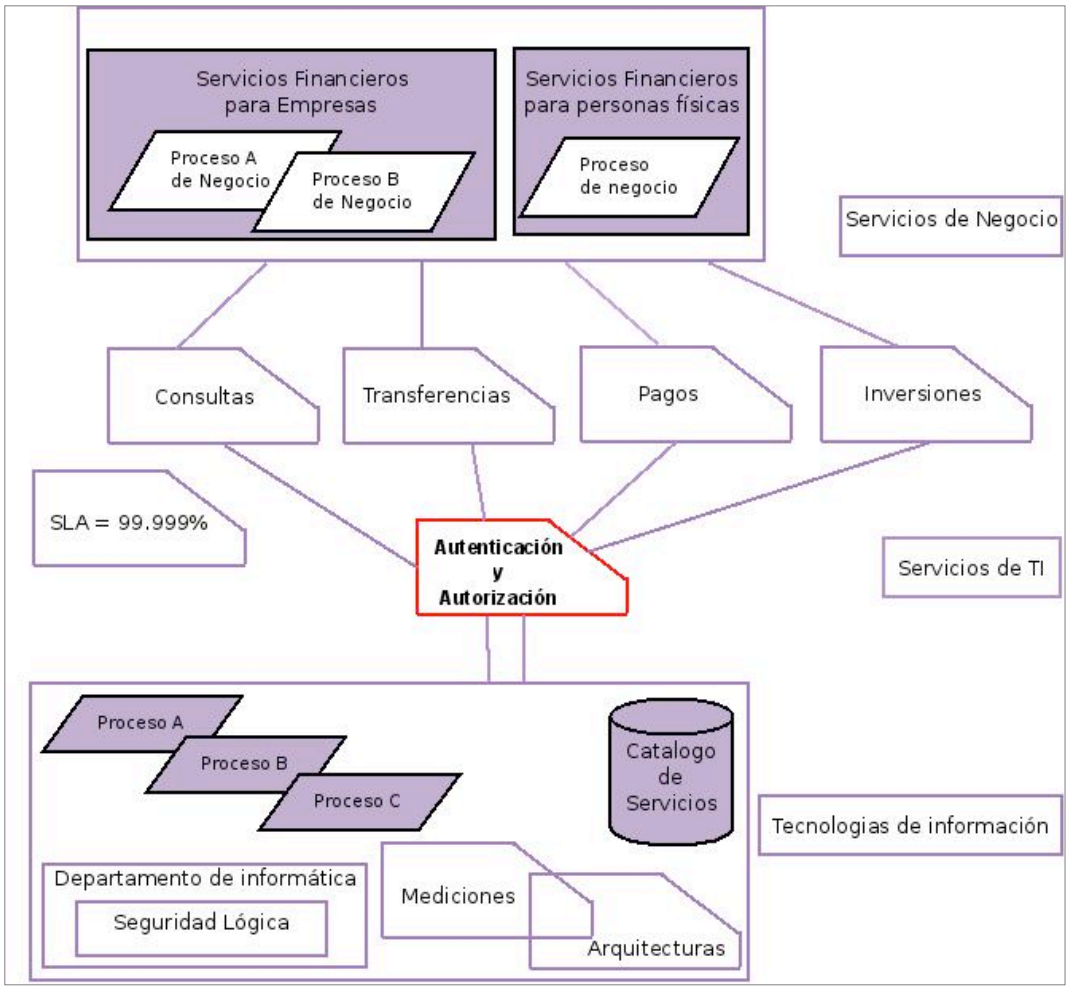

Figura 1.10 Modelo de servicios de *BNET*

La concepción de la capa de autenticación y autorización como un servicio significaba otorgar el valor suficiente a los usuarios de la *BNET* proporcionando la seguridad necesaria para que realizaran sus operaciones bancarias con plena confianza y sin temor de sufrir alguna afectación, sin costos adicionales, ni riesgos específicos asociados al uso de herramientas tecnológicas. Finalmente el valor agregado que se obtiene a partir del uso de este servicio garantiza cierto nivel de seguridad y se suma a la cadena de valor que otorga el servicio de la *BNET* para sus usuarios.

1.8.- Gestión financiera

Resultaba muy claro que el hecho de mantener la infraestructura anterior representaba costos muy elevados asociados a los recursos necesarios para brindar el servicio de

autenticación y autorización requerido por la *BNET.* Lo anterior cobraba importancia, ya que incrementaba considerablemente el Costo Total de la Propiedad, mejor conocido como *TCO.*

Es importante mencionar que el Costo Total de la Propiedad es una estimación financiera, cuyo propósito principal es ayudar a determinar los costos directos e indirectos de un producto o sistema (Office of Government Commerce, 2007a) y bajo este orden de ideas, cabe señalar que el cliente había incurrido en un gasto inicial para la adquisición de los componentes de hardware y software de la solución, más los costos asociados a los servicios de implantación de dicha solución. Sin embargo, el gasto más significativo era la sumatoria de los costos recurrentes de operación y mantenimiento a partir de que dicha solución había sido liberada a producción y por esta razón, la idea de evaluar una posible sustitución, resultaba muy atractiva en términos de costos.

Durante el mes de Octubre de 2010 se calcularon los costos de operación y mantenimiento con la intención de obtener un costo recurrente mensual y poder contar con los elementos suficientes para comparar los posibles escenarios de reemplazo. A continuación se presenta la tabla 1.4 que contiene los rubros más relevantes (sin mencionar los costos) que fueron detectados al momento de hacer los cálculos y comparativos que impulsaron con gran fuerza la aceptación de la propuesta:

OCTUBRE 2010			
No	**Centro de Costos - Departamento de Informática**	**Descripción**	**Costo Aproximado en dólares americanos**
1	Área de Infraestructura	Mantenimiento de infraestructura de hardware y software dedicada a la capa de autenticación y autorización. Dicha infraestructura tenía entre 4 y 5 años de obsolescencia, con lo cuál el costo asociado al mantenimiento se incrementaba considerablemente.	Oculto
2	Área de Seguridad Lógica	Contratación de servicios profesionales del fabricante *IBM* para la afinación de los componentes de software de la capa de autenticación y autorización.	Oculto
3	Área de Seguridad Lógica	Contratación de 2 ingenieros adicionales, con el objeto de poder atender en sitio el número creciente de incidentes reportados a un segundo nivel por el Centro de Atención de Usuarios, en un horario de atención de lunes a viernes de 11:00 pm a 7:00 am y sábados y domingos de 9:00 am a 6:00 pm.	Oculto
4	Área de Redes	Número total de horas hombre dedicadas a la reconfiguración y afinación de componentes de red, con el propósito de mejorar los tiempos de respuesta. Al mismo tiempo el personal de redes era constantemente involucrado en la atención a incidentes y cambios en la *BNET.*	Oculto
5	Área de Diseño y Desarrollo	Número total de horas hombre dedicadas a la optimización de código con el propósito de mejorar los tiempos de respuesta. Al mismo tiempo el personal de diseño y desarrollo era constantemente involucrado en la atención a incidentes de *BNET.*	Oculto
6	Área de Infraestructura	Número total de horas hombre dedicadas a la reconfiguración y optimización de sistemas operativos con el propósito de mejorar los tiempos de respuesta. Al mismo tiempo el personal de infraestructura era constantemente involucrado en la atención a incidentes y cambios en la *BNET.*	Oculto
COSTO TOTAL			Oculto

Tabla 1.4 Costos de operación previos a la implementación

Por otro lado, como resultado de los problemas de rendimiento experimentados por los usuarios durante dicho mes, se sabía que muchos clientes del banco habían retirado sus inversiones, provocando una ligera disminución en las utilidades, así como en los flujos de efectivo generados durante el mes y evidentemente esta situación generó pánico en diferentes áreas de negocio al interior de la organización del cliente y se convirtió en un catalizador, acelerando la iniciativa para un posible cambio.

Obviamente por razones de confidencialidad de la información, no se presenta el análisis de *TCO* dentro de este documento, ya que la información financiera del cliente es clasificada como altamente sensible, estratégica, y de carácter altamente confidencial y restringida. De hecho en su momento fue necesario firmar un acuerdo de confidencialidad, mejor conocido cómo *NDA*, entre el cliente y la consultoría.

De igual manera fue necesario elaborar un caso de negocio para determinar el Retorno De Inversión, entre otras cosas, mejor conocido como *ROI,* cuyo propósito principal es cuantificar el valor de la inversión, aunque no siempre resulta del todo preciso (Office of Government Commerce, 2007e). Para el *CIO,* por parte del cliente, el *ROI* representaba las herramientas necesarias para la justificación del proyecto ante el *CFO,* que además era la persona encargada de gestionar la asignación de presupuestos dentro de la estructura organizacional del cliente. Lo que se pretendía principalmente era demostrar la viabilidad del proyecto teniendo como base una serie de datos monetarios, con los cuales poder justificar de una forma muy evidente la toma de una decisión en particular. A continuación se muestran los principales objetivos de negocio de tipo financiero que fueron considerados para el análisis del caso de negocio elaborado:

- Disminuir los costos asociados a la operación y mantenimiento de la plataforma que proporciona el servicio de *BNET.*
- Aumentar los ingresos totales generados por el uso del servicio de *BNET*.
- Incrementar los márgenes de ganancia.

Adicionalmente fueron definidos algunos objetivos de tipo estratégico y de industria como parte del caso de negocio, ya que los argumentos de justificación presentados no solo se basaron en análisis de costos. Sin embargo no se profundiza más sobre el tema por quedar fuera del alcance de la presente memoria.

1.9.- Objetivos

1.9.1.- Objetivo general

Garantizar los niveles de seguridad en el servicio de la *BNET* a través de una capa de autenticación y autorización que cuente con los controles de acceso suficientes para determinar y restringir los privilegios otorgados a cada uno de los usuarios del servicio.

1.9.2.- Objetivos específicos

1.- Incrementar los niveles de disponibilidad del servicio de la *BNET* por medio de una solución para la capa de autenticación y autorización suficientemente ágil y robusta.

2.- Reducir los tiempos de acceso de los usuarios del servicio de la *BNET* y

particularmente en la capa de autenticación y autorización.

3.- Incrementar la capacidad de los componentes de hardware y software que conforman la solución de la capa de autenticación y autorización.

4.- Reforzar los controles de seguridad en la capa de autenticación y autorización del servicio de la *BNET* por medio de un dominio seguro.

5.- Disminuir los costos inherentes a la operación de la capa de autenticación y autorización de la *BNET.*

6.- Mejorar la satisfacción y la experiencia de los usuarios del servicio de la *BNET.*

Los objetivos del proyecto estaban perfectamente definidos gracias a que previamente habían sido alineados a los objetivos de negocio del cliente. Aparentemente el objetivo general estaba más orientado a una cuestión operativa, pero en realidad engloba objetivos de negocio de tipo financiero y estratégico, ya que al garantizar los niveles de seguridad requeridos, se genera una mayor confianza en el usuario del servicio, y esto eventualmente puede atraer un mayor número de usuarios y/o cuentahabientes, incrementando los ingresos del cliente. Por otro lado, el proveer un servicio estable y confiable puede significar un diferencial ante la competencia, lo cuál a su vez podía tener un impacto positivo para el negocio, posicionando al cliente como uno de los lideres de la industria bancaria.

2.- SOLUCION DEL PROBLEMA

2.1.- Componentes principales de la solución de dominio seguro

Es importante mencionar que la solución de dominio seguro propuesta inicialmente surge a partir de una arquitectura de referencia para la capa de autenticación y autorización basada en productos de la marca *Sun Microsystems*. Además de que en su momento los integrantes de la consultoría, incluyendo el autor de esta memoria, eran ex empleados de la empresa *Sun Microsystems de México,* subsidiaria de *Sun Microsystems Inc,* misma que posteriormente, en el año 2010 fue adquirida por la empresa *Oracle Corporation*[7]. Dicho antecedente evidentemente tuvo una gran influencia al momento de proponer la arquitectura de solución, ya que particularmente el autor de la presente memoria tenía el conocimiento suficiente para diseñar e implementar una solución basada 100% en una plataforma de la marca *Sun Microsystems,* por el hecho de haber trabajado casi 7 años para la compañía, durante los cuales participó en muchos proyectos de seguridad para la industria bancaria. Sin embargo por algunos temas de carácter comercial relacionados con el cumplimiento de normas y políticas de competencia al interior de la organización del cliente y situaciones ajenas a nuestro alcance, se tuvieron que sustituir algunos componentes de la solución propuesta por los de la marca I*BM,* mismos que se describen posteriormente.

Así que una vez que el autor de la presente memoria fue asignado oficialmente como consultor líder de proyecto, responsable de la correcta ejecución de todas las actividades y tareas en tiempo y forma por parte la consultoría, el siguiente paso fue agendar una reunión con el personal del cliente, con el propósito de comunicarles quién estaría a cargo del proyecto, y mencionarles la manera en como funcionaría la solución.

La primer tarea fue determinar los productos de software que conformarían la solución del dominio seguro, partiendo de la arquitectura de referencia, los cuales básicamente son los 3 que se describen brevemente a continuación:

- *Componente de Acceso Web*.- Es un componente que controla el flujo de autenticación a través de credenciales conformadas por un nombre de usuario y una contraseña, además de que autoriza los privilegios para cada usuario por medio de perfiles y roles, permitiendo que cada usuario que ha sido previamente autenticado de manera exitosa, posteriormente le sean asignados de forma única y exclusiva los privilegios que su organización le otorga de acuerdo a la jerarquía y a sus funciones dentro de la misma. El producto que originalmente se tenia pensado utilizar para esta capa era *Sun Java System Access Manager*, cuya última versión oficial al momento de escribir esta memoria, es la 7.1, liberada el 1 de Marzo de 2007 y que ha sido utilizado ampliamente para control de accesos hacia las aplicaciones web, Sin embargo debido a que el banco solicito explícitamente que para esta capa de la solución

[7] *La empresa Oracle adquirió a la empresa Sun Microsystems en el año 2010, tal y como se indica en el sitio oficial de Oracle Corporation http://www.oracle.com/us/sun/index.htm*

deseaban utilizar la suite de *TAM,* conocida oficialmente en el mercado como *IBM Tivoli Access Manager,* cuya funcionalidad dentro del entorno completo era exactamente la misma que la del *Sun Java System Access Manager*, con la particularidad de que en su estructura interna se dividía en 2 grandes componentes:

- *IBM WebSEAL*.- *IBM Tivoli Access Manager WebSEAL* es un componente fundamental de la Suite de *IBM Tivoli Access Manager*, y básicamente es un administrador de recursos, responsable de proteger y administrar la información y los recursos basados en web. Es decir, se define como un servidor web de alto rendimiento, multi-hilado y capaz de aplicar políticas de seguridad con una elevada granularidad sobre un espacio protegido de objetos web. *WebSEAL* ofrece una solución de Single *Sign-On a través de Tivoli Identity Manager o TIM,* e incorpora recursos de un servidor de aplicaciones dentro de su política de seguridad. *WebSEAL* normalmente funge como un *web Proxy* reverso, recibiendo todas las peticiones *HTTP/HTTPS* que hace un navegador y posteriormente entrega el contenido desde su propio servidor web o desde un servidor de aplicaciones que generalmente se encuentra unido por la parte de atrás. Todas las peticiones son pasadas a través del *WebSEAL* y son evaluadas por el servicio de autorización dentro del *IBM Tivoli Access Manager* para determinar si el recurso solicitado es autorizado para el usuario que hace la solicitud. Para mayor información respecto a este componente, consultar el anexo C.
- *IBM Tivoli Access Manager PD*.- Es una solución completa de autorización y administración de políticas de seguridad en la red que provee protección a los recursos web sobre intranets o extranets dispersas sobre cualquier región geográfica. Dicho elemento también es conocido como *Policy Director*. Para mayor información respecto a este componente, consultar el anexo C.

- *Componente de Servidor de Directorio o LDAP*.- Es un componente de software utilizado principalmente para almacenar la información de los usuarios dentro de registros organizados bajo una estructura jerárquica, el cuál se encuentra dentro de los estándares que señala el protocolo *Lightweight Directory Access Protocol*, mejor conocido como *LDAP*. Dentro de la arquitectura de solución propuesta este componente fungía como repositorio de usuarios. El producto elegido para esta capa de la solución fue *Sun Java System Directory Server Enterprise Edition,* en su versión 7.0*,* que en aquél momento era la evolución de la suite *Netscape Directory Server*. Para mayor información respecto a este componente, consultar el anexo C.

- *Componente de Servidor Proxy de Directorio*.- Era un componente de software, cuyo rol era fortalecer la seguridad dentro del repositorio de *LDAP*. Este producto no tendría ningún sentido de existir sin la presencia del producto anterior, y por esta razón es considerado como un complemento del producto *Sun Java System Directory Server Enterprise Edition*, cuyo nombre especifico es *Sun Java System Directory Proxy Server.* En realidad funciona como un tipo especial de servidor de directorio, el cuál típicamente se encarga de atender las peticiones hechas hacía el servidor de directorio ya que se ubica lógicamente entre el componente que hace la solicitud y los servidores de directorio, y su función principal es proveer un enrutamiento y filtrado basado en políticas para las peticiones que llegan al servidor de directorio, y con esto lograr un esquema de seguridad en el que solo aquellos objetos con los privilegios adecuados pueden acceder al servidor de directorio y además genera la percepción en la que aparentemente solo existe un servidor de

directorio, aunque en realidad pueden existir más de uno. Este componente de la solución propuesta también se encarga de hacer un balanceo de cargas entre los 2 servidores de directorio que se tienen considerados. Para mayor información respecto a este componente, consultar el anexo C.

Para el despliegue de estos componentes de software se propuso inicialmente el sistema operativo *Solaris 10*, ya que al momento de escribir esta memoria era una plataforma *UNIX* suficientemente robusta, estable, líder en el mercado, y con los niveles de seguridad requeridos. Así mismo, se propuso una plataforma de procesamiento basada en tecnología *RISC*, que básicamente eran servidores con procesadores *SPARC multicore y multithread.* La razón principal de dicha elección era que en aquel momento, a pesar de que el sistema operativo *Solaris* 10 se encontraba disponible para arquitecturas de procesador SPARC y x86, en el caso específico de la capa de directorio se requerían equipos de rango medio que soportaran particiones físicas con el afán de obtener mejores niveles de disponibilidad, además de que la plataforma de procesador *SPARC* presentaba mejor acoplamiento con el sistema operativo, y gracias a ello se podían explotar funcionalidades como *Fault Management Architecture* o *FMA,* la cuál no estaba disponible para la plataforma de procesador x86*,* debido a que el desarrollo de *Solaris* surge inicialmente solo para plataforma de procesador *SPARC* y a lo largo del tiempo se fueron agregando módulos y funcionalidades que explotaban de manera nativa las bondades de dicha arquitectura, además de que la solución era considerada de misión crítica, por el hecho de requerir 5 9s de disponibilidad o dicho de otra forma, 99.999%, y para tal efecto los servidores basados en este tipo de tecnología resultaban idóneos, gracias a sus características, como componentes redundantes, tolerancia a fallas, y facilidad para reemplazar algunos componentes, como fuentes de poder, ventiladores, y discos duros, en línea y sin necesidad de apagar el equipo. Por otro lado, derivado de la inclusión de la suite *IBM Tivoli Access Manager* y siguiendo con la preferencia del cliente de tener los productos de software sobre hardware del mismo fabricante, se determinó en conjunto con personal del cliente que los servidores ideales para alojar los componentes de dicha suite serían de la marca *IBM*, así que se propusieron servidores basados en tecnología de procesador x86, con sistema operativo Linux. Para mayor información acerca de los sistemas operativos utilizados en la solución, véase anexo C.

La primera aproximación de la solución propuesta desde un punto de vista general se representa en el diagrama de despliegue de *UML* de la figura 2.1.

En dicha figura se puede apreciar claramente que cada una de las 3 capas de la solución propuesta se encuentra en instancias de sistema operativo redundantes, lo cuál permite que ante cualquier falla en alguno de los elementos de software y/o hardware no haya una interrupción en el servicio. Adicionalmente cada uno de los servidores propuestos cuenta con redundancia en algunos componentes tales como fuentes de poder, ventiladores, y discos duros internos, todos ellos con tecnología *hot-swap*, la cuál permite que dichos componentes sean intercambiados en caliente, sin necesidad de hacer un reinicio en el sistema operativo ni apagar el equipo.

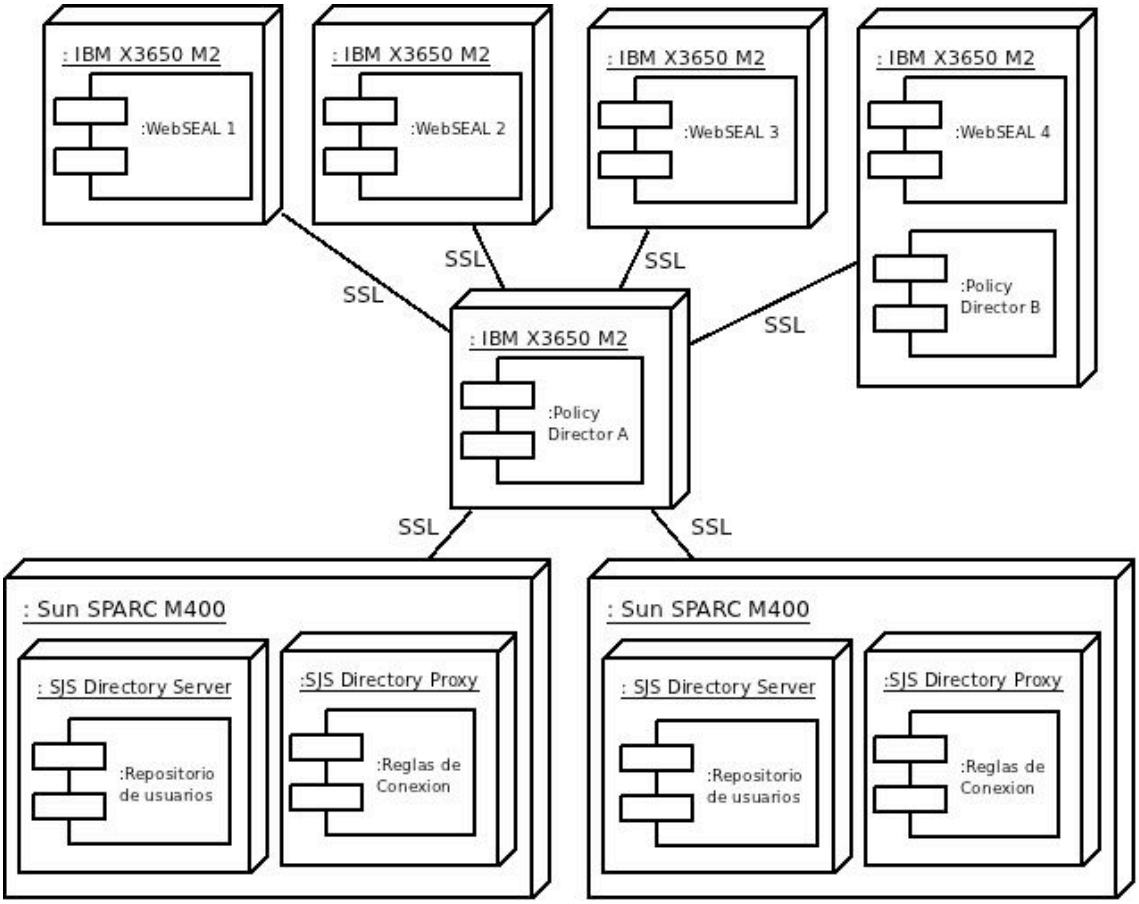

Figura 2.1 Diagrama de despliegue de la solución

En el caso particular de los servidores *Sun SPARC Enterprise M4000*, se obtenía automáticamente un gran nivel de confiabilidad en los mismos, gracias al reducido número de componentes físicos internos, lo cuál reducía a su vez la complejidad de los equipos y las probabilidades de algún fallo en los componentes individuales, dando como resultado un periodo de tiempo entre fallas, mejor conocido como *MTBF*, bastante favorable y alineado a los acuerdos de niveles de servicio acordado o *SLA*. Debido a que el servicio de directorio era el más importante y crítico por el hecho de contener la base de datos con todos los usuarios de la banca en línea, resultaba ideal montarlo sobre los servidores *Sun SPARC Enterprise M4000*. Para mayor información sobre dicho servidor véase anexo C.

Para el componente de acceso web se propusieron servidores menos robustos que los requeridos para el servidor de directorio. Sin embargo se propuso una mayor cantidad, con el objeto de dividir el 100% de los usuarios concurrentes entre un mayor número de servidores físicos, y con esto, obtener porciones con un porcentaje menor de usuarios, de tal forma, que ante una falla de software y/o hardware en alguno de los servidores que implicara la pérdida del mismo, se lograra la redistribución de las cargas de trabajo y/o usuarios entre los servidores restantes de forma equitativa y con una degradación mínima en la calidad del servicio. Para mayor información sobre los servidores propuestos del modelo *IBM System X3650 M2*, para alojar la capa de acceso web, véase el anexo C.

Adicionalmente se recomendó ampliamente al personal del cliente que negociara un contrato de soporte con el nivel más alto existente al interior de cada una de las 2 empresas, proveedores de los productos de software y hardware de la solución de

dominio seguro, es decir, *Oracle* e *IBM*. Esto con la idea de reducir el tiempo de recuperación ante la presencia de alguna falla de software y/o hardware que pudiera impedir parcial o totalmente la entrega del servicio. Dicho tiempo de recuperación es mejor conocido como *Mean Time To Recovery* o *MTTR*.

2.2.- Componentes complementarios de la solución de dominio seguro

Es importante señalar que existen otros elementos que no forman parte fundamental del núcleo de la solución de dominio seguro de manera directa y por lo tanto no se profundizará sobre los mismos, ya que quedan fuera del alcance de la presente memoria, sin embargo, resultan esenciales y tienen cierto grado de participación de manera indirecta dentro del entorno completo. A continuación se mencionan brevemente los más importantes:

Red de Almacenamiento.- Mejor conocida como *Storage Area Network* o *SAN*. Cabe mencionar que en ese momento el cliente tenía la necesidad de adquirir una red de almacenamiento, con el objeto de proveer mayor capacidad de espacio a todas sus aplicaciones, bases de datos, y servicios, entre los cuales se encontraba el servicio de la banca en línea, incluyendo todos sus componentes. Este proceso de adquisición se disparó de forma paralela en tiempo junto con el proyecto del dominio seguro, así que alrededor de 5 TB de la infraestructura de almacenamiento de la marca *EMC®*[2] serían asignados para ser utilizados por la solución del dominio seguro. El sistema de almacenamiento constaba de un arreglo de la familia *VNX 5700*[8].

Firewalls.- Como parte de la propuesta integral del dominio seguro se requería contar con un par de dispositivos para proveer la seguridad necesaria en el perímetro, haciendo un filtrado de puertos para todas las conexiones, basándose en reglas y políticas, dichos dispositivos son mejor conocidos como cortafuegos o *firewalls* (Cheswick, Bellovin & Rubin, 2003), cuya marca era libre elección por parte del cliente, siempre y cuando se otorgara el rendimiento y ancho de banda requeridos de acuerdo con el ancho de banda de los enlaces, así que finalmente el área de seguridad lógica determinó que conforme a los lineamientos y estándares de la institución, se debían adquirir equipos de la marca *Check Point®.* Todo el proceso de instalación, configuración y afinación de estos dispositivos fue ejecutado directamente por personal del fabricante de manera conjunta con personal del cliente, así que la participación de la consultoría para este rubro fue prácticamente nula.

Balanceadores de Carga.- Como parte del proceso de ajustes y modificaciones que se hicieron a la arquitectura original, en Agosto de 2010 fueron adquiridos un par de balanceadores de carga para protocolos *http/https* de la marca *Cisco.* El objetivo principal de la adquisición de estos dispositivos en su momento fue distribuir de forma equilibrada la carga de conexiones concurrentes hacía cada uno de los nodos responsables del control de acceso hacia el portal del servicio de la banca en línea, y con ello suponer que disminuirían los altos tiempos de respuesta. Estos dispositivos mantuvieron la permanencia dentro de la solución del dominio seguro propuesta.

2.3.- Administración de la capacidad y de la disponibilidad

[8] *Los detalles de la familia VNX 5700 están disponibles en la siguiente liga de internet http://www.emc.com/storage/vnx/vnx-series.htm*

Una vez que se tenían definidos lo productos de la solución de manera general, en términos de componentes lógicos y físicos, el siguiente paso era determinar las capacidades requeridas en cada uno de los nodos físicos donde se alojarían los componentes de software, teniendo como indicadores, la concurrencia de usuarios reportada en Octubre de 2010, así como el universo total de usuarios y su crecimiento estimado en volumen para los siguientes 3 años.
Cabe mencionar que no existe una regla mágica automatizada que permita determinar las capacidades requeridas para una solución de dominio seguro. Sin embargo existen algunas consideraciones importantes que fueron tomadas en cuenta al momento de definir las capacidades de hardware. A continuación se presentan los más importantes:

- Alta disponibilidad.- Para lograr un esquema de alta disponibilidad en el servicio de autenticación y autorización se debían tener al menos 2 servidores para cada capa de la solución.
- Capacidad de *CPU*.- Es importante mencionar que como parte de los indicadores de desempeño, quedó establecido un límite máximo permitido del 70% para los niveles de utilización del *CPU* y memoria *RAM* ante condiciones normales de operación, esto debido a su relación tan cercana con el *SLA,* ya que ante una posible falla en uno de los nodos de la solución, se debía contar con la capacidad para redistribuir las cargas de trabajo entre los nodos funcionales restantes, sin comprometer los niveles de servicio acordados, esta situación nos llevo a considerar un porcentaje de holgura en cada uno de los servidores que formaban parte de la solución propuesta.
- Ancho de banda.- Se debía tener presente que ante la posible falla de uno de los servidores, se requería contar con la capacidad para reacomodar las cargas de trabajo entre los equipos restantes en un tiempo mínimo que permitiera conservar el *SLA* a través de canales de comunicación 1GB y 10GB redundantes.
- Volumen de datos.- El volumen de datos que sería procesado por los equipos tendría un alto impacto en los propios niveles de desempeño, así que para lograr niveles óptimos de rendimiento, se debía contemplar que los servidores de *LDAP* contaran con suficiente memoria RAM para alojar todos los datos de los usuarios. El cálculo para determinar el volumen de datos se hizo con base en el número de entradas o usuarios del servicio de *BNET*, así como el tamaño de cada una de las entradas, considerando todos sus atributos, y el número de índices existentes.
- Calidad del Servicio.- También conocido como *QoS*. Antes de iniciar propiamente las actividades de dimensionamiento de la capacidad, se tuvo que observar el tiempo promedio por operación, y lo más importante, el tiempo de respuesta hacia el usuario, para determinar la cantidad de memoria RAM y los requerimientos de E/S, que básicamente son, el tipo de almacenamiento ideal para alojar los datos, y la capacidad de ancho de banda para las tarjetas de red.
- Tolerancia a fallos.- El hardware a considerar como parte de la solución de dominio seguro propuesta debía poseer componentes redundantes, tales como discos duros, fuentes de poder, y ventiladores. Dichos componentes redundantes debían permitir el “intercambio en caliente” de los mismos sin necesidad de hacer un reinicio del sistema.

Dentro de la fase del dimensionamiento de las capacidades de hardware requeridas, inicialmente se realizaron las actividades referentes al producto *Sun Java System Directory Server Enterprise Edition,* para posteriormente hacer lo propio con el producto *IBM Tivoli Access Manager.*

2.3.1.- Dimensionamiento de los servidores de directorio

Considerando los elementos necesarios para dimensionamiento, descritos anteriormente, se obtuvo la tabla 2.1 con valores específicos para cada una de las variables necesarias para determinar la capacidad de la infraestructura de hardware para albergar el componente de software *Sun Java System Directory Server Enterprise Edition*:

Sun Java System Directory Server Enterprise Edition	
Métricas	**Volumen**
Número de entradas totales	3,500,000
Tamaño Promedio de entrada	4KB
Número de atributos	10
Número de índices Personalizados	5
Máxima utilización de CPU	70%
SearchRate (Regresar los 10 atributos)	21K srch/seg, etime=0.2ms
AuthRate (Srch + Bind)	9K auth/seg,auth-etime=0.4ms

Tabla 2.1 Parámetros requeridos para el dimensionamiento de los servidores de directorio

En las siguientes secciones se muestra una serie de expresiones utilizadas para el cálculo de la cantidad de recursos mínima requerida, las cuales han servido para hacer un dimensionamiento más atinado y con menor margen de error.

2.3.1.1.- Memoria *RAM*

El dimensionamiento de la capacidad de memoria *RAM* resultaba esencial para lograr un óptimo rendimiento del producto *Sun Java System Directory Server Enterprise Edition,* ya que el factor principal que permitía ejecutar una lectura ágil y veloz, era justamente hacer que todos los datos residieran sobre un *filesystem* de tipo *cache*, de forma estructurada, y dentro de una base de datos del sistema de directorio conocida como *dbcache*. Dicha aceleración se presenta gracias a que al momento de subir los datos a un área de memoria *RAM*, la lectura de los mismos, solo requiere ir a una dirección física y traer el dato, lo cuál resulta mucho más rápido y eficiente, comparado con hacerlo desde un *filesystem* convencional basado en disco mecánico.
A continuación se muestra una expresión que fue utilizada para determinar la cantidad aproximada de memoria *RAM* que sería requerida, expresada en *MBytes*:

*2 * (nb_of_entries * avg_entry_size) / 1024 + 2*1024*

Donde:
nb_of_entries es el número de entradas que serán alojadas en el directorio
avg_entry_size es el tamaño promedio de una entrada en el directorio expresado en Kbytes

Además se considera un factor de crecimiento estimado del 30% simple para los 3 años siguientes, expresado en la cantidad de entradas (o usuarios) en el directorio. Finalmente el cálculo queda como se presenta a continuación:

*RAM Requerida = 2 * ((3,500,000 + 1,050,000) * 4KB) / 1024 + 2 * 1024*
RAM Requerida = 37,594.8MB

RAM Requerida = 36.7GB

Bajo la premisa de que se debían considerar al menos 2 servidores para otorgar alta disponibilidad en el servicio y contar con tolerancia a fallos, así como la limitante en los niveles de utilización máxima del 70%, se debía considerar el doble de la cantidad de memoria *RAM* requerida, obtenida a partir de la expresión anterior, y un 30% adicional de holgura para poder operar ante el posible escenario de que uno de los 2 servidores colapsara debido a una falla ocasionada por problemas eléctricos, hardware, sistema operativo, software, etc. Lo anterior significaba que un solo servidor debía tener la capacidad de memoria para soportar las transacciones que residirían en ambos equipos, sin sobrepasar el 70% de utilización permitida. Finalmente se determina que el doble de la memoria es igual a 73.4GB, al cuál se agrega el 30% adicional, resultando un total de 95.5GB, que redondeado en términos de memoria *RAM*, se cierra a 96GB, distribuida equitativamente entre los 2 equipos.

2.3.1.2.- CPU

Con el objeto de satisfacer las expectativas de rendimiento esperado, se dimensionó el tipo, y la cantidad de *CPUs* expresada en términos de núcleos de procesador o *cores*, considerando el desempeño de las lecturas y escrituras, así como la limitante de utilización máxima del 70% más el 30% de crecimiento adicional.

El producto *Sun Java System Directory Server Enterprise Edition* tiene la particularidad de que su escalamiento es lineal, es decir, la relación entre el número de *cores* de procesador y el número de operaciones por unidades de tiempo es directamente proporcional (Sun Microsystems Inc, 2009), donde el punto de partida es el número de operaciones que puede soportar un solo *core*.

Número de Cores	Búsquedas por Segundo	Autenticaciones por Segundo	Operaciones Totales por Segundo
1	2,850	1,196	4,046
2	5,850	2,400	8,250
4	11,400	4,762	16,162
8	21,150	9,525	30,675

Tabla 2.2 Relación entre el número de operaciones del directorio y el número de *cores*

La relación entre las cifras mostradas en la tabla anterior permanecen sin cambio alguno, sin importar si existen 2 millones o 10 millones de entradas, siempre y cuando residan en un área de memoria *RAM*. Se puede observar claramente que el número de búsquedas soportado por *core* es mayor con respecto al número de autenticaciones, y este comportamiento es natural, ya que el proceso de autenticación contiene dentro de si mismo el proceso de búsqueda, y además de que evidentemente existen usuarios que no se logran autenticar de forma exitosa por errores al momento de digitar sus nombres de usuario y/o contraseñas, además de todos aquellos ataques a los que es propenso un sitio de banca electrónica expuesto en Internet.
Por otro lado, se sabe que el número de sesiones concurrentes máximo en los días pico era alrededor de 3,500 y adicionalmente se requería considerar su respectivo crecimiento del 30%, lo cuál daba como resultado el siguiente universo de sesiones concurrentes:

Sesiones Concurrentes Máximo = (3,500) + 30%

Sesiones Concurrentes Máximo = 4,550

Cada sesión concurrente equivalía a 4 operaciones por segundo dentro del servidor de *directorio o LDAP*, lo cuál significaba que el número de operaciones se obtenía a partir de la siguiente expresión:

*OTPS = (SCM) *(OIPS)*
Donde:
OTPS.- Operaciones Totales Por Segundo
SCM.- Número Máximo de Sesiones Concurrentes
OIPS.- Operaciones Individuales Por Segundo

Entonces:

*OTPS = (4,550) * (4)*
OTPS = 18,200

Retomando la limitante de utilización máxima del 70%, significa que se debe aumentar automáticamente un 30% adicional de capacidad de procesamiento otorgada.

*OTPS = 18,200 * 1.3*
OTPS = 23,660

La cifra anterior representa el total de transacciones que debe soportar la infraestructura de procesamiento, considerando el escenario de un posible colapso en uno de los 2 equipos. Dicha valoración nos dio la pauta para referirse a la tabla de rendimiento y determinar que la cantidad de *cores* de procesador requerida en cada uno de los equipos era de 8, ya que de haber elegido únicamente 4 *cores* de procesador por servidor, solo podíamos llegar a un máximo de 16,162 operaciones por segundo, lo cuál resultaba insuficiente. Así que el número total de *cores* requerido era de 16, contemplando los 2 equipos para conservar el esquema de alta disponibilidad.

2.3.1.3.- Espacio de almacenamiento

Dentro de esta sección se contemplaron los requerimientos del espacio en disco y f*ilesystem* para que el producto *Sun Java System Directory Server Enterprise Edition* funcionara eficientemente, y para ello fueron considerados los siguientes componentes:

- La base de datos del Directorio (incluyendo todos los datos más los índices)
- La base de datos de la bitácora de cambios de replicación.
- Bitácoras de Acceso
- Respaldos de la instancia del directorio.

El espacio requerido fue la suma de los tamaños de cada uno de estos componentes más el porcentaje de crecimiento del 30% para los próximos 3 años, contemplando siempre un nivel de utilización máximo del 70%.

2.3.1.3.1.- Dimensionamiento de espacio para la base de datos

El dimensionamiento de la base de datos fue realizado considerando el número total de entradas, el tamaño promedio de cada entrada, así como el número de índices. Con base en estas 3 variables se aplica la siguiente expresión para realizar el cálculo:

*Directory_DB_Size = (nb_of_entries * avg_entry_size) + Directory_Indexes_Size*
Donde:
Directory_DB_Size.- Es el tamaño de la base de datos del directorio.
nb_of_entries.- Es el número de entradas en el directorio con un 30% adicional de crecimiento.
avg_entry_size.- Es el tamaño promedio de una entrada en el directorio expresado en KB.
Directory_indexes_Size.- Es el tamaño total de los índices del directorio.

Por otro lado:
*Directory_Indexes_Size = (nb_of_indexes * avg_index_size)*
Donde:
nb_of_indexes.- Es el número de índices en el directorio.
avg_index_size.- Es el tamaño promedio de un índice en el directorio expresado en KB.
Además:
avg_index_size = f(nb_of_entries, avg_attr_size)
Donde:
avg_attr_size.- Es el tamaño promedio de cada atributo.

Simplificando el cálculo, básicamente se estima la variable nb_of_indexes con el número de índices personalizados. Para el caso particular del cliente, habían 8 índices personalizados por lo que el cálculo final queda como se muestra a continuación:

*Directory_DB_Size = ((3,500,000 + 30%) * 4KB) + (8 * (3,500,000 +30%) * (0.04KB))*
*Directory_DB_Size = ((3,500,000 + 1,050,000) * 4KB) + (8 * (3,500,000 + 1,050,000) * (0.04KB))*
*Directory_DB_Size = (4,550,000 * 4KB) + ((8 * 4,550,000) * 0.04KB)*
Directory_DB_Size = (18,200,000KB) + (1,456,000KB)
Directory_DB_Size = 19,656,000KB
Directory_DB_Size = 18.74GB

2.3.1.3.2.- Dimensionamiento de espacio para bitácora de cambios de replicación

Al momento de realizar el cálculo de espacio requerido para la bitácora de cambios en el proceso de replicación, se consideró principalmente la cantidad de cambios que debían ser retenidos directamente en disco, así como el periodo de retención durante el cuál dichas bitácoras debían permanecer almacenadas, esto con el propósito de estar alineados con el cumplimiento de la ley *Sarbanes-Oxley*[9], bajo la cuál se rigen muchas instituciones bancarias alrededor de todo el mundo (Welytok, 2006) y para tal efecto, el número de cambios fue estimado calculando el número máximo de

[9] *Ley promulgada en los Estados Unidos de América en el año 2002, también conocida como SOX que de acuerdo a la regla llamada Retention of Records Relevant to Audits and Reviews, determina la retención de información. Esta regla se encuentra disponible para consulta en el sitio oficial de la U.S. Securities and Exchange Commission, a través de la siguiente liga de internet http://www.sec.gov/rules/final/33-8180.htm*

operaciones de escritura que eran ejecutadas durante un día y se considero el número de días durante los cuales aquellos cambios debían ser guardados directamente en disco, antes de ser migrados a través de un sistema de respaldos hacía un medio magnético, como la cinta, en donde los datos debían permanecer al menos por un periodo de 10 años antes de ser desechados. El periodo de retención en *filesystem* fue definido en conjunto con personal del cliente. Cabe mencionar que las políticas de retención para todos los cambios realizados en el servicio de directorio no necesariamente están alineadas con la política de retención de los datos bancarios de los cuentahabientes, ya que como se ha mencionado en repetidas ocasiones, la bitácora de cambios del servidor de directorio básicamente guarda las altas y bajas de usuarios, así como los cambios de contraseñas, dentro del repositorio de usuarios.

El número de días básicamente dependía de la estrategia de respaldo, en la cuál para este caso particular, se ejecutarían respaldos incrementales de los datos diariamente, y como el periodo de retención es de N+1 días, donde N es el intervalo entre los días que se ejecutarían los respaldos completos, entonces el periodo era igual a 7+1, es decir, 8 días. Considerando que el tamaño promedio de cada entrada en la base de datos del *LDAP* era de 4KBytes, la fórmula para determinar los requerimientos de almacenamiento expresados en KBytes para la base de datos de la bitácora de cambios es como se muestra a continuación:

*Esp1 = ((Nb_of_modifies_per_day * 0.44) + (Nb_of_adds_per_day * 0.45)) * Nb_of_days * 2*
Donde:
Nb_of_modifies_per_day.- Es el número de modificaciones por día.
Nb_of_adds_per_day.- Es el número de altas por día.
Nb_of_days.- Es el periodo de retención expresado en días.

Con base en los reportes obtenidos a partir de las actividades de monitoreo durante el mes de Octubre de 2010, se observó que del total de operaciones de escritura en el directorio, el 30% se constituía por altas de usuarios, y el 70% correspondía a modificaciones de al menos 1 atributo, incluyendo la contraseña, en usuarios existentes. Así que finalmente el cálculo se hizo como se presenta a continuación:

*Esp1 = ((15,120 * 0.44) + (6,480 * 0.45) * 8 *2)*
Esp1 = 6,652.8 + 46,656
Esp1 = 53,308.8KB = 52.0MB, pero se aproximó a 100MB por ser un valor pequeño

2.3.1.3.3.- Dimensionamiento de espacio para bitácoras de acceso y auditoría

Cabe mencionar que las bitácoras de acceso constituyen un aspecto muy importante a considerar en el dimensionamiento de la capacidad de almacenamiento, ya que en ellas se guarda toda la evidencia de acceso por parte de los usuarios. El comportamiento por defecto para el producto *Sun Java System Directory Server Enterprise Edtition* es retener hasta un máximo 1 GByte en las bitácoras de acceso, lo cuál en términos de tiempo puede representar desde unos cuantos minutos hasta varias semanas o incluso meses, dependiendo de la concurrencia y actividad de los usuarios. Para efectos prácticos de dimensionamiento en el caso particular del ambiente del cliente, el tamaño requerido para estas bitácoras se obtuvo a partir del número de operaciones totales ejecutadas durante 1 día, incluyendo accesos, considerando que el tamaño promedio de una operación es de 0.5 KBytes.
La fórmula para obtener el tamaño estimado requerido para las bitácoras de acceso y

auditoría era la siguiente:

*Esp2 = ((Nb_of_modifies_per_day * 0.5) + (Nb_of_adds_per_day * Average_entry_size)) * Nb_of_days*
Donde:
Nb_of_modifies_per_day.- Es el número total de modificaciones por día.
Nb_of_adds_per_day.- Es el número total de altas por día.
Average_entry_size.- Es el tamaño promedio de una entrada en el directorio.
Nb_of_days.- Es el número de días durante los cuales se desea retener la información.

Entonces:
*Esp2 = (15,120 * 0.5) + (6,480 * 4KB) * 30*
*Esp2 = (7,560) + (25,920) * 30*
Esp2 = (7,560) + (777,600)
Esp2 = 785,160KB
Esp2 = 0.74GB, pero se aproximó a 1.0GB

La capacidad total de espacio en almacenamiento requerido para albergar todos los datos referentes a la capa de directorio es la suma del espacio requerido por las bitácoras, y el de las base de datos. A continuación se muestra la expresión para calcular el espacio total:

EspTotal = Directory_DB_Size + Esp1 + Esp2
EspTotal = 18.74GB + 0.1GB + 1.0GB
EspTotal = 19.84GB

Finalmente se agrega el 30% adicional al valor obtenido anteriormente con el objeto de mantener los niveles de utilización máxima permitidos por debajo del 70%, es decir:

*EspTotal = 19.84GB * 1.3*
EspTotal = 25.79GB

Dicha capacidad de almacenamiento era necesaria de manera independiente en cada uno de los 2 servidores que conformarían la capa de directorio o *LDAP.*

Después de haber finalizado todo el proceso de dimensionamiento de capacidades de hardware requeridas por el componente de software *Sun Java System Directory Server Enterprise Edition,* dentro de la solución de dominio seguro, es importante tomar en cuenta que en todo momento fue considerada una solución capaz de otorgar alta disponibilidad, lo cuál significaba contar con al menos 2 equipos en cada una de las capas de la solución propuesta. Un punto importante a considerar es que se requerirían 2 equipos adicionales para el componente de software *Sun Java System Directory Proxy Server.* Dicho componente forma parte de la capa de directorio o *LDAP,* y básicamente se propusieron equipos con las mismas características que los servidores de *LDAP.* Cabe señalar que para el caso particular de la solución propuesta para el cliente, solo se utilizaron 2 servidores físicos, los cuales permitían la configuración de particiones eléctricamente aisladas, mejor conocidos como dominios físicos, cada uno con recursos independientes, esto permitió consolidar la infraestructura de la solución utilizando menor espacio físico y menores consumos eléctricos. El modelo de los servidores propuestos fue el *Sun SPARC Enterprise M4000*, para el cuál en el anexo C se presenta una ficha técnica con las especificaciones detalladas del servidor. A continuación se presenta la tabla 2.3, que muestra la configuración de los servidores donde sería instalado el software de la capa

de directorio o *LDAP.*

Servidor	Dominio	Procesador	Memoria RAM	Almacenamiento Interno (Boot)	Puertos de Red	Puertos FC
Sun SPARC Enterprise M4000	Dominio A: LDAP Server	2 CPU SPARC64 VII+ @2.66GHz QC	48GB	300GB	2	2
	Dominio B: LDAP Proxy	2 CPU SPARC64 VII+ @2.66GHz QC	48GB	300GB	2	2
Sun SPARC Enterprise M4000	Dominio A: LDAP Server	2 CPU SPARC64 VII+ @2.66GHz QC	48GB	300GB	2	2
	Dominio B: LDAP Proxy	2 CPU SPARC64 VII+ @2.66GHz QC	48GB	300GB	2	2

Tabla 2.3 Especificaciones de los servidores de directorio

Así mismo, a continuación se presenta una tabla donde se muestra el espacio requerido en la *SAN* de almacenamiento para alojar todos los datos del directorio.

Servidor	Dominio	Espacio en SAN (GB)
Sun SPARC Enterprise M4000	Dominio A: Servidor de LDAP	25.8
	Dominio B: Proxy de LDAP	No requerido
Sun SPARC Enterprise M4000	Dominio A: Servidor de LDAP	25.8
	Dominio B: Proxy de LDAP	No requerido

Tabla 2.4 Espacio en *SAN* requerido por los servidores de directorio

2.3.2.- Dimensionamiento de los servidores del componente de acceso web

Dentro de la solución propuesta, la capa correspondiente al producto de software *IBM Tivoli Access Manager,* se integra de 2 grandes componentes, uno es el *IBM Tivoli Access Manager WebSEAL* y el otro es el *IBM Tivoli Access Manager Policy Director.* El primer paso al momento de hacer el diseño de la solución, fue atender el dimensionamiento o planeación de la capacidad para *WebSEAL,* ya que las capacidades de hardware requeridas por el *Policy Director* son mínimas, al ser una base de datos que solo guarda autorizaciones, y el número de transacciones realizadas dentro de si misma es mínimo, ya que en realidad todas las operaciones se ejecutan en la capa de directorio o *LDAP*. El *WebSEAL* sería desplegado sobre varios equipos con el objeto de preservar el esquema de alta disponibilidad solicitado. El proceso de dimensionamiento se ejecuta tomando en consideración 2 parámetros, que son, la frecuencia de las transacciones y de la concurrencia de usuarios, contemplando su respectivo crecimiento del 30%, para lo cuál se utilizó la información real de los ambientes de producción anteriores con el objeto de mitigar los márgenes de error y variación al momento de determinar el número de equipos, así como las características individuales de procesamiento y memoria. Básicamente lo que se debía utilizar era la cantidad de usuarios que hacían uso del sistema durante un periodo de tiempo determinado, así como la cantidad de transacciones dentro de la capa de *TAM,* que ejecutaba cada usuario durante una sesión abierta, y con ello obtener una aproximación del promedio de las cargas de trabajo que se colocarían al sistema de autenticación y autorización.

Cabe señalar que cuando se habla de transacciones en este contexto, se refiere a operaciones de tipo *Java,* originadas por los cambios de pantalla que solicita un usuario desde el navegador, cifra que fue estimada en 45 transacciones promedio por cada sesión abierta. La expresión usada para el cálculo del número de transacciones totales por segundo se muestra a continuación:

*TTPS = (SCM * TPU)/DS*

Donde:
TTPS.- Es el número total de transacciones totales por segundo.
SCM.- Es el número de sesiones concurrentes máximo registrado, considerando el 30% adicional de crecimiento.
TPU.- Es el número de transacciones por cada sesión abierta por parte de un usuario.
DS.- Es la duración promedio de cada sesión expresada en segundos.

Así que:
*TTPS = ((3,500 * 1.3) * 45)/105*
*TTPS = (4,550 * 45)/105*
TTPS = (204,750)/105
TTPS = 1,950

Una vez que se obtuvo el número estimado de transacciones totales por segundo para el entorno del dominio seguro, el siguiente paso era elegir el hardware adecuado para soportar dicha carga de transacciones. Como ya fue citado previamente, desde el inicio del proyecto fue acordado ofertar una solución de autenticación y autorización funcional que permitiera al cliente contar con tecnología de última generación, así que en esta ocasión era recomendado considerar servidores que garantizaran un buen rendimiento para la capa de *IBM Tivoli Access Manager*. Siguiendo con la directriz de contar con una solución basada en bloques, ahora la parte crítica del proceso de arquitectura era definir la configuración de cada uno de los equipos, contemplando la cantidad, familia y velocidad de los procesadores, así como la cantidad de memoria *RAM*, y espacio de almacenamiento en discos internos.

La manera más rápida y acertada para encontrar el equipamiento idóneo donde residiría el componente de control de accesos *TAM*, fue tomar como referencia reportes o certificados de rendimiento generados por un organismo internacional sin ánimo de lucro y completamente imparcial como lo es *Standard Performance Evaluation Corporation* o *SPEC*, cuyo objetivo principal es producir, establecer, mantener, y avalar *benchmarks* de rendimiento para equipo de cómputo.

El *benchmark* que fue utilizado es el *SPECjEnterprise2010,* el cuál básicamente es una prueba multi-capa para medir el rendimiento de servidores de aplicaciones basados en tecnología *Java 2 Enterprise Edition.* La métrica utilizada esta basada en el número de operaciones *Java* por segundo, mejor conocidas como *EjOPS*, las cuales, en realidad son transacciones.

El modelo de servidor que fue elegido es el *IBM System X3650 M2,* basado en el certificado para dicho equipo, el cuál tiene la capacidad de soportar hasta un máximo de 1,013.4 *EjOPS*[10].

[10] *Al momento de escribir esta memoria, dicho reporte se encontraba disponible a través de la siguiente URL accesible desde internet: http://www.spec.org/jEnterprise2010/results/res2010q1/jEnterprise2010-20091207-00002.html*

Con base en el número de *EjOPS* que otorga un servidor *IBM System X3650 M2*, sabemos que con 2 servidores se podía cubrir el 100% del número máximo de transacciones por segundo con casi un 100% de utilización de cada uno de los equipos. Sin embargo, tomando en consideración los niveles de disponibilidad y utilización requeridos, así como el porcentaje de crecimiento esperado, se decidió proponer una solución basada en 4 servidores para albergar los componentes de *TAM.* Lo anterior significaba que el número de *EjOPS* debía ser dividido entre los 4 servidores para obtener una estimación del número de *EjOPS* promedio que tendría cada uno de los servidores. Es decir, la carga que tendría cada equipo por segundo se obtiene a partir de la cantidad total de transacciones por segundo dividido entre 4. A continuación se muestra una expresión para obtener el valor de dicha carga:

QPSI = TPS / n
Donde:
QPSI.- Carga de transacciones por segundo sobre un equipo en particular.
TPS.- Cantidad total máxima de transacciones por segundo.
n.- Cantidad de servidores para albergar TAM.
Luego entonces:
QPSI = 1,950 / 4
QPSI = 487.5

Por otro lado, de acuerdo al reporte de *SPEC* sabemos que cada servidor tiene la capacidad de soportar hasta un máximo de 1,013.4 transacciones por segundo, así que esto significaba que cuando los 4 equipos se encontraran funcionando correctamente, la carga en cada uno de ellos sería de menos del 50%, y en el supuesto caso de que uno de los equipos presentara una falla e interrumpiera el servicio, entonces la carga que contenía dicho equipo al momento de la falla, eventualmente tendría que ser redistribuida entre los 3 equipos restantes, ya que se encontrarían en una configuración modo cluster. Al redistribuir la carga entre 3 equipos, significaba que ahora la cantidad total de transacciones por segundo tendría que ser dividido entre 3, resultando un total de 650, lo cuál se mantenía por debajo del 70% de utilización.

A continuación se muestra la tabla 2.5 con las especificaciones más importantes del modelo de servidor que fue propuesto formalmente, tal y como aparecen en el reporte y en su ficha técnica.

Servidor	Procesador	Memoria RAM	Almacenamiento Interno (Boot)	Puertos de Red	Sistema Operativo
IBM System x3650 M2	8 cores, 2 chips, 4 cores/chip Intel Xeon X5570 @2.9Ghz	16GB	146GB 10K SAS	4	SUSE Linux Enterprise Server 10

Tabla 2.5 Especificaciones de los servidores del componente de acceso *web*

2.4.- Validación de la compatibilidad de componentes

Una parte importante dentro del proceso de diseño era justamente validar la compatibilidad de todos los componentes involucrados, ya que de existir alguna incompatibilidad, era el momento preciso de tomar la decisión adecuada para cambiar todos aquellos productos y/o versiones que representaran un riesgo. Básicamente, la idea era validar la compatibilidad entre los servidores, los sistemas operativos

propuestos, y los componentes de software del dominio seguro, para lo cuál fue necesario recurrir a la documentación oficial de los productos. En realidad, el primer paso era validar que los productos específicos que conformaban la solución del dominio seguro estaban soportados por los respectivos sistemas operativos, y posteriormente validar que los sistemas operativos estaban soportados por los servidores correspondientes. En seguida se presenta la tabla 2.6, que muestra una matriz de compatibilidad, indicándose con un "si" todos aquellos que son compatibles:

Sistema Operativo	Servidores		Componentes de Dominio Seguro	
	Sun SPARC Enterprise M4000	IBM System x3650 M2	Sun Java System Directory Server 7.0	IBM Tivoli Access Manager 6.0
Solaris 10	Si		Si	
SUSE Linux Enterprise Server 10		Si		Si

Tabla 2.6 Matriz de compatibilidad de componentes[11]

2.5.- Arquitectura física de la solución

Una vez que se tenía definida la arquitectura de la solución se programó una reunión con personal del cliente para presentar dicha arquitectura, y validar si representaba valor para ellos, así como recibir retroalimentación con respecto a temas de compatibilidad con algunos componentes del entorno, de los cuales no se tuviera visibilidad por parte de la consultoría. El diagrama de arquitectura física donde se muestran los equipos que conforman la solución se presenta en la figura 2.2:

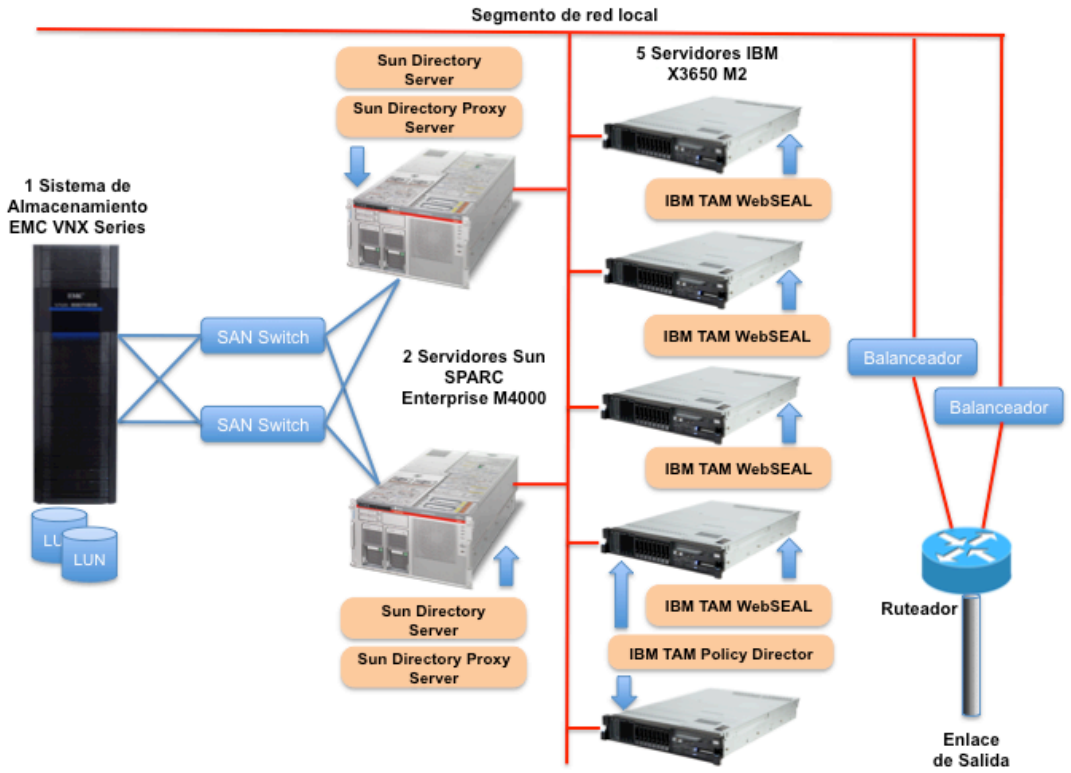

Figura 2.2 Diagrama físico de la solución

[11] *Para mayor información visite en internet los sitios oficiales de los fabricantes, Oracle e IBM, que son http://www.oracle.com y http://www.ibm.com respectivamente.*

2.5.1.- Arquitectura ambiental

Cabe señalar que el centro de datos donde serían alojados los servidores que conformaban la solución de dominio seguro ya contaba previamente con todas las capacidades físicas y eléctricas necesarias para la correcta operación del único gabinete, sobre el cuál estaría montado todo el hardware propuesto, tal y como se muestra en la figura 2.3:

Figura 2.3 Vista frontal del gabinete de la solución

En realidad, como parte del diseño de la solución se incluyo la información necesaria referente a los requerimientos de consumos eléctricos y físicos para cada uno de los servidores, con el objeto de que el cliente instalara los circuitos eléctricos con el amperaje requerido, así como los receptáculos donde sería conectado el gabinete. A continuación se presenta la tabla 2.7, la cuál indica dichos consumos eléctricos y físicos.

Servidor	Cantidad	Watts		BTUs/hr		Unidades de Rack		Peso (kg)	
		Unitario	Total	Unitario	Total	Unitario	Total	Unitario	Total
Sun SPARC Enterprise M4000	2	2,350 W	4,700	8,020	16,040	6	12	84	168
IBM X3650 M2	5	675 W	3,375	2,262	11,310	2	10	15.6	78
GRAN TOTAL	7		**8,075**		**27,350**		**22**		**246**

Tabla 2.7 Consumos eléctricos y físicos de la infraestructura propuesta

A partir de la tabla anterior se determinó que se requería un gabinete con 2 unidades de distribución de potencia, mejor conocidas como *PDU,* cada uno de 15 kVA 1 fase, para lo cuál era necesario que el centro de datos contara con 2 circuitos eléctricos independientes, cada uno de 24 A, así como 6 receptáculos tipo *NEMA L6-30R.* Para mayor información sobre el gabinete ofertado véase el anexo C.

2.6.- Administración de la continuidad del servicio

Se debe tener en cuenta que la solución de dominio seguro es una capa encargada de proveer exclusivamente servicios de seguridad a través de diversos controles, cuya función primordial era prevenir incidentes de seguridad que pudieran comprometer el servicio de la banca en línea, y bajo esta perspectiva, en realidad la razón de ser de la solución de dominio seguro radicaba en proveer la seguridad necesaria para los usuarios del servicio de la banca en línea, y como tal estaría sujeto a los niveles de servicio acordados para el servicio de la banca en línea, anteriormente mencionados y referidos en un 99.999%. La solución del dominio seguro debía ser considerada y valorada dentro de la Estrategia de la Continuidad de Negocio, mejor conocida como *BCS,* para lo cuál se requería definir los procesos necesarios para la continuidad del servicio del dominio seguro y posteriormente integrarlos a los planes de continuidad del servicio de la banca en línea y a los planes de recuperación ante la posibilidad de un evento que pudiera derivar en un desastre que a su vez ocasionara la interrupción del servicio. Desafortunadamente no existía un Plan de Recuperación de Desastres, mejor conocido como *DRP* o *Disaster Recovery Plan,* debido a que solo existía un centro datos donde se alojaba toda la infraestructura de TI, mismo donde sería alojada la infraestructura de la solución de dominio seguro propuesta, y ante el escenario de un desastre que ocasionara la destrucción del centro de datos, prácticamente sería imposible continuar con la prestación del servicio de la *BNET*. Por tal motivo no hubo participación directa de la consultoría en la definición de los procesos requeridos para la continuidad del servicio. Sin embargo, con el objeto de mitigar el riesgo de probables interrupciones en el servicio, todos y cada uno de los componentes de la solución se encontrarían bajo un esquema redundante otorgando alta disponibilidad y eliminando puntos únicos de falla en cada una de sus capas correspondientes, tal y como se ilustra en la tabla 2.8.

Componente	Redundancia	Cantidad de Instancias	Consecuencia por Falla
Servidor de Directorio	Si	2	Degradación de Servicio
Proxy de Directorio	Si	2	Degradación de Servicio
Servidor de Políticas	Si	2	Degradación de Servicio
Servidor de Acceso Web	Si	4	Degradación de Servicio

Tabla 2.8 Redundancia de instancias y consecuencias por posible falla

2.7.- Administración de la seguridad de la información

Es importante mencionar que el diseño de la solución del dominio seguro debía estar completamente alineado con los 3 aspectos fundamentales que definen la seguridad de la información, siendo estos, disponibilidad, integridad y confidencialidad, tal y como lo marcan los estándares de *ISO27K*[12], al igual que los lineamientos de la metodología de *ITIL v3* para la fase de diseño (Office of Government Commerce, 2007b: p.141).

Los servidores de directorio se convertían en elementos de alta criticidad por el hecho de resguardar información personal de todos los usuarios y fungir como fuente

[12]*Es un estándar que particularmente en la Sección 2.19 de ISO/IEC 27000/2009 [ISO27000] habla sobre las técnicas de seguridad, así como de los sistemas de administración de la seguridad de la información. Se encuentra disponible para consulta pública en el sitio oficial de ISO a través de la siguiente liga de internet http://www.iso.org/iso/catalogue_detail?csnumber=41933*

autoritativa para los servicios de autenticación y autorización para el servicio de la banca en línea, así que se establecieron una serie de controles alrededor de la solución del dominio seguro.

2.7.1.- Política de contraseñas

Una de las funciones principales del servidor de directorio es almacenar las contraseñas de los usuarios, por lo tanto resulta fundamental contar con un mecanismo de cifrado, para que dichas contraseñas no se guarden en texto claro y puedan ser observadas por entidades no autorizadas. Para ello, debía ser habilitado el esquema de almacenamiento de contraseñas del servidor de directorio, basado en *SSHA* que significa *Salted SHA-1,* el cuál genera cadenas de *hash* no reversibles, y además impide ataques basados en diccionarios, al no permitir que 2 contraseñas iguales generen la misma cadena de *hash*, como es el caso de los algoritmos *SHA-1* y *MD5.*

Como parte del proceso de definición de la política de contraseñas, fue establecido a 3 el número máximo de intentos de acceso no exitosos, antes de que una cuenta de usuario se bloqueara automáticamente por el servidor de directorio. Así que una vez que alguna cuenta ha sido bloqueada, la única forma de desbloquearla sería a través de un servicio de mesa de ayuda o directamente en alguna de las sucursales del cliente.

Finalmente fueron definidas las reglas de composición de contraseñas, al igual que la caducidad de las mismas y la cantidad de caracteres repetidos permitidos al efectuar el cambio de una contraseña a otra. Sin embargo por acuerdo de confidencialidad entre el cliente y la consultoría, no está permitido revelar dicha política de contraseñas.

2.7.2.- Control de accesos

El control de accesos es un elemento muy importante dentro del ámbito de la seguridad de la información, y particularmente en nuestra solución de dominio seguro resultaba un elemento crucial, ya que precisamente uno de los objetivos de la solución propuesta era reforzar el control de accesos del servicio de la banca en línea.

Cabe señalar que la mayoría de los servidores de directorio cuentan con algún tipo de lenguaje para el control de accesos, aunque en realidad la forma exacta de lograrlo puede variar de un producto a otro. Sin embargo los sistemas de control de accesos se relacionan con la misma serie de conceptos, a pesar de que los términos que se usen para referirse a dichos conceptos pudieran ser diferentes de una plataforma a otra. A continuación se presenta una breve definición para cada uno de los conceptos que se utilizan dentro del diseño del modelo de datos (Sun Microsystems Inc, 2009) para el control de accesos de la solución del dominio seguro[13]:

- Sujeto.- Es la persona o entidad que solicita el acceso. Este puede estar definido por el *DN* de un usuario, el *DN* de un grupo, o incluso por una regla o filtro dentro del servidor de directorio, cuyo resultado haga referencia a una serie determinada de sujetos o entradas dentro del *LDAP.*

[13] *Sun Microsystems Inc (2009).*

- Objeto.- Es aquella cosa que esta siendo accedida. Los objetos incluyen entradas completas, atributos particulares, y posiblemente valores específicos para dichos atributos.
- Acceso.- Se refiere al tipo de acceso que tiene un sujeto en particular. El tipo de acceso afecta de manera directa las operaciones que pueden ser ejecutadas dentro del servidor de directorio, tales como:
    - Agregar una entrada
    - Borrar una entrada
    - Acceder una entrada (no necesariamente se tiene acceso a los atributos)
    - Leer un atributo
    - Modificar un atributo
    - Buscar un atributo
- ACI.- Es un elemento de control de acceso, referido así por sus siglas en inglés, cuyo significado es *Access Control Item*, el cuál básicamente es una regla que define el nivel de acceso que tiene un sujeto sobre ciertos objeto(s).
- ACL.- Es una lista de control de accesos, referido así por sus siglas en inglés, cuyo significado es *Access Control List.* Dicho elemento esta conformado por un grupo de *ACIs.*

Antes de establecer una política de acceso basada en listas de control de accesos, era necesario definir el árbol de directorios, mejor conocido como *DIT,* para lo cuál se utilizaron una serie de patrones de arquitectura que a continuación se mencionan:

- Colocar las *ACLs* en el menor número posible de entradas, que para el caso particular del servicio de la *BNET* se tendrían definidos únicamente 2 tipos de usuarios, y por lo tanto el control de acceso al servicio de la banca en línea se lograría por medio de 2 grupos, donde cada uno de ellos cuenta con una serie de características únicas que lo hacen diferente del otro y por tal motivo tienen asociada una lista de control de acceso. Los 2 grupos en que se clasificó el universo total de usuarios son los siguientes:

    1. Grupo A.- Consta de personas físicas, cuyo tipo de cuenta o usuario requiere tener los privilegios básicos dentro del servicio de la banca en línea.
    2. Grupo B.- Consta de personas morales o empresas, cuyo tipo de cuenta o usuario requiere tener privilegios extendidos dentro del servicio de la banca en línea.
- Las *ACLs* se asocian directamente con los 2 grupos definidos, lo cuál permite a los administradores controlar la membrecía de los usuarios dentro de cada grupo, sin necesidad de asociar *ACLs* con cuentas de usuario de manera individual.
- Habilitar la opción de grupos dinámicos, ya que de esta manera es posible agrupar a los usuarios o entradas bajo un criterio de búsqueda a través de un filtro, lo cuál sería requerido para fines de agilizar y facilitar la administración y operación de la solución del dominio seguro.
- Definir las reglas de contenido para el *DIT,* con el objeto de indicar cuales clases auxiliares pueden ser utilizadas dentro de cada clase estructural, así como la creación de las listas de *MUST* y *MAY* , las cuales controlan los atributos que deben existir, y los que pueden o no existir dentro de una clase, respectivamente.
- Establecer los límites de búsqueda para evitar ataques basados en denegación de servicio, ocasionados por un número excesivo de operaciones por conexión, o incluso por un número excesivo de conexiones por grupo, lo cuál se refuerza con la utilización de los servidores de *proxy* de directorio.

2.7.2.1.- Diseño de esquema

Aprovechando que el árbol de directorios estaría designado para dar servicio única y exclusivamente a los usuarios del servicio de la *BNET*, se tomó la decisión de que todas las entradas de tipo *person*, mismas que hacen referencia a los usuarios, se colocarían bajo una misma ubicación, ya que finalmente la política de control de accesos estaría basada en la membrecía de los 2 grupos mencionados, así como algunos atributos que serían utilizados como filtros en el caso de los grupos dinámicos.

2.7.2.2.- Diseño del árbol de directorios

La base del directorio sería el nodo raíz, cuyo objeto se refiere a la organización, para lo cuál el nombre propuesto fue `banco`. Por el hecho de no requerir nuevas definiciones de objetos y/o atributos, se facilitaba bastante el diseño del árbol, ya que el contenido de este nodo al igual que para toda la estructura del árbol de directorios estaría basado en clases de objetos estándar, tal y como se indica de acuerdo a la definición de la última versión del protocolo *LDAP*, publicada en el documento *RFC 4517*[14]. A continuación se muestra la definición del nodo raíz:

```
dn:  o=InstitucionBancaria
aci: Definición de ACIs
objectClass: top
objectClass: organization
o: InstitucionBancaria
```

Cabe señalar que por motivos de confidencialidad no se tiene permitido publicar la definición explícita de las *ACIs* que se encuentran alojadas en la base del directorio.

A partir del nodo raíz se empieza a definir toda la estructura de datos que conformaría el *DIT,* así que en un siguiente nivel estaría ubicado el nodo correspondiente al país en cuestión, que en este caso es Venezuela, para lo cuál se utilizaría el prefijo '`ve`'. A continuación de presenta la definición del nodo país:

```
dn: c=ve, o=InstitucionBancaria
objectClass: top
objectClass: country
c=ve
```

A partir del nodo país se desprenden dos nodos adicionales, el primero de ellos fue creado para contener la definición de los 2 grupos mencionados anteriormente, mientras que el segundo sirve para contener la unidad organizacional, denominada como `ou`, la cuál hace referencia a '`banco`'. A continuación se muestra la definición del nodo unidad organizacional:

```
dn: ou=banco, c=ve, o=InstitucionBancaria
objectClass: top
objectClass: organizationalUnit
```

[14] *El RFC 4517 define la especificación del protocolo LDAP y se encuentra disponible en el sitio oficial de la IETF a través de la siguiente liga de internet http://www.ietf.org/rfc/rfc4517.txt*

```
ou: banco
```

Por otro lado, ahora se muestra la definición del nodo grupo:

```
dn: cn=bnetXXX, c=ve, o=InstitucionBancaria
member: secAuthority=default
objectClass: groupOfNames
objectClass: top
cn: bnetXXX
```

A partir del nodo de la unidad organizacional se derivan los nodos en donde reside propiamente el contenido de los usuarios del servicio de la banca en línea, alojados bajo un nombre común o `'cn'`. A continuación se muestra la definición del nodo de nombre común de usuario:

```
dn: cn= username, ou=banco, c=ve, o=InstitucionBancaria
nsUniqueId: X
uid: Y
userPassword: Z
sn: apellido
objectClass: top
objectClass: organizationalPerson
objectClass: inetOrgPerson
objectClass: person
objectClass: ePerson
cn: username
```

Adicionalmente fue necesaria la creación de un nodo llamado `operation`, el cuál debía contener una clave de operaciones, necesaria para hacer operaciones interbancarias, como transferencias y pagos, desde el servicio de la *BNET*. A continuación se muestra la definición de dicho nodo:

```
dn:    cn=operation,    cn=username,    ou=banco,    c=ve,    \
o=InstitucionBancaria
nsUniqueId: X
personalTitle: n
userPassword: Y
sn: operation
objectClass: top
objectClass: inetOrgPerson
objectClass: organizationalPerson
objectClass: ePerson
objectClass: person
cn: operation
```

Finalmente era necesario definir dentro del servidor de directorio un nodo necesario para la correcta operación del componente de control de accesos *Tivoli Access Manager*, denominado como `secAuthority`. Dicho objeto debía residir dentro de cada usuario, ya que es el elemento que se utilizó para generar el vínculo con los 2 grupos definidos previamente, así que estaría contenido en el nodo de nombre común de usuario, y a continuación se muestra la definición del mismo:

```
dn:  secAuthority=default,  cn=username,  ou=banco,  c=ve,  \
o=InstitucionBancaria
nsUniqueId: X
```

```
secPwdLastChanged: Y
secUUID: Z
secHasPolicy: boleano
secAcctValid: boleano
objectClass: top
objectClass: secUser
objectClass: eUser
objectClass: cimManagedElement
principalName: username
secPwdValid: boleano
secLoginType: Default:LDAP
secAuthority: default
```

Derivado de la definición del nodo anterior fue definido un nodo hijo adicional denominado como `PolicyData`, sobre el que se guardaría información relacionada con la política de seguridad propiamente, incluyendo el control de fechas en que se cambian las contraseñas del usuario y a continuación se presenta la definición del mismo:

```
dn: cn=PolicyData, secAuthority=default, cn=username, ou=banco,\
c=ve, o=InstitucionBancaria
nsUniqueId: X
modifyTimestamp: fecha de última modificación
modifiersName: cn=root
secPwdLastChanged: fecha de último cambio
objectClass: top
cn: PolicyData
creatorsName: cn=root
createTimestamp: fecha de creación
```

2.7.2.3.- Administrador del servidor de directorio

Por otro lado era importante establecer la política de control de acceso para el usuario administrador del dominio seguro, para lo cuál se creo una cuenta de usuario administrador con la siguiente definición dentro del árbol de directorios:

```
dn: cn=Directory Administrators, o=InstitucionBancaria
nsUniqueId: X
objectClass: top
objectClass: groupofuniquenames
cn: Directory Administrators
```

Es importante aclarar que en las definiciones de algunos de los nodos anteriores se mencionan en repetidas ocasiones las variables X, Y, y Z para representar los diferentes valores que pueden tener los atributos de los mismos objetos, a través de cadenas de caracteres que no necesariamente tienen relación entre la definición de un objeto y la de otro. Estas cadenas pueden contener números, alfanuméricos, y meta caracteres. Así que finalmente, después de haber definido todos y cada uno de los nodos que conformarían el árbol de directorios, a continuación se muestra de manera gráfica la estructura de datos propuesta en la figura 2.4:

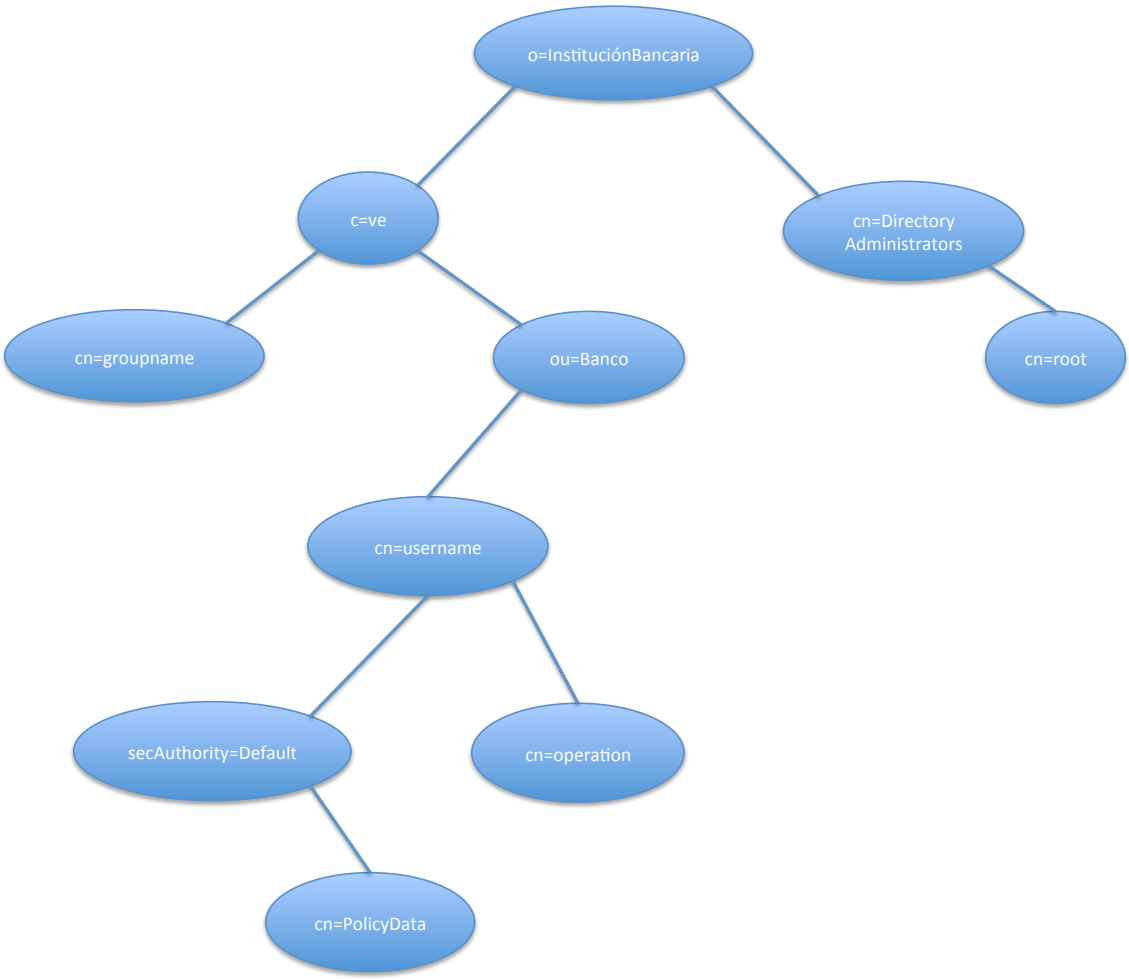

Figura 2.4 Estructura del árbol de directorios

2.7.3.- Uso de memoria cache

Como parte de la solución se propuso utilizar dentro del servidor de directorio la funcionalidad de *cache,* con lo cuál se logra que toda la información contenida en el servidor de directorio se mueva a un área de memoria *RAM,* y con ello se incrementa el rendimiento y se agilizan los tiempos de respuesta para las peticiones realizadas por los clientes. Cada una de las 2 instancias de directorio tendría su propia área de *cache,* en donde residirían principalmente páginas que a su vez contienen índices de la base de datos del directorio, y no necesariamente cada página es un índice, sino mas bien, cada página corresponde a una porción del universo total de índices.
Al habilitar esta función, el servidor de directorio estaría moviendo estas páginas entre el grupo de archivos de la base de datos y la *cache* basándose en la frecuencia de acceso para cada registro, es decir, que las páginas residentes en memoria siempre contendrían los registros que fueron accedidos recientemente, mientras que el resto residiría directamente en los archivos de las bases de datos. La cantidad de memoria designada en cada servidor físico sería de 48 GB, de acuerdo a lo establecido en la sección 2.3.1.1.

En realidad al momento de inicializar el servidor de directorio, todo el contenido de los sufijos de donde se derivan los usuarios y los grupos sería cargado en un área de

disco de donde podían ser tomados fácilmente hacia el área de *cache*. Cabe mencionar que todos los datos leídos desde disco serían automáticamente movidos a memoria para futuras peticiones.

2.7.4.- Uso de índices

Como parte de la solución se propuso que el servidor de directorio usara índices para acelerar las operaciones de búsqueda, ya que de lo contrario cada vez que se iniciara un proceso de autenticación se tendría que comparar cada una de las entradas de cada uno de los sufijos para encontrar el usuario solicitado. Los índices serían creados y administrados de manera independiente para cada sufijo a pesar de que todos serían almacenados en los archivos de la misma base de datos. Cada archivo de índices correspondería a todos los índices para un sufijo definido para un atributo especifico dentro del árbol de directorios. Aparte de los índices creados por defecto para cada uno de los sufijos, se crearían índices para cada uno de los directorios definidos a partir del nodo denominado como `cn=username`.

2.7.5.- Replicación

La replicación es una técnica esencial que contribuye directamente a incrementar la disponibilidad de la información, así que se propuso un esquema de replicación en modo multimaestro entre los 2 servidores de directorio, basado en una topología donde ambos se encontrarían activos atendiendo solicitudes de diferentes clientes al mismo tiempo, de tal forma que las modificaciones hechas a los datos estarían propagándose recíprocamente entre sí a través de vectores de actualización de replicas, mejor conocidos como *RUV*. Cabe señalar que la replicación multimaestro permite la resolución y conciliación de conflictos generados por los cambios realizados directamente en los datos de los usuarios de manera concurrente.

Por otro lado, uno de los grandes beneficios que otorga la configuración multimaestro, es que en el preciso instante en que se presente una falla que ocasione la pérdida del servicio en uno de los servidores de directorio, el nodo restante toma el control de las operaciones que se estaban ejecutando en el nodo perdido al momento de colapsar. Dicho proceso es comúnmente conocido como *failover.*

La parte más crítica dentro del proceso de replicación entre los 2 servidores de directorio, era definir los acuerdos de replicación, mismos que a su vez definen la relación entre los 2 servidores bajo el esquema de proveedor – consumidor, en el cuál ambos servidores son proveedores y consumidores al mismo tiempo. Los sufijos que se estarían replicando entre los 2 servidores de directorio eran, `cn=groupname`, y `cn=username`, respectivamente, bajo un esquema en el que el primer servidor es proveedor del sufijo `cn=username` y consumidor del sufijo `cn=groupname`, y el segundo servidor es proveedor del sufijo `cn=groupname` y consumidor del sufijo `cn=username`.

El intervalo de tiempo para la propagación de cambios entre cada replica fue establecido a 3 minutos, con lo cuál se generaría una alta carga de tráfico de paquetes en la red, por lo que se decidió asignar una red dedicada completamente a la replicación, en la que todo el tráfico sería cifrado por medio del protocolo *TLS* para evitar que el contenido pudiera ser visto por entidades no permitidas. A continuación se presenta la figura 2.5 en la que se muestra de manera gráfica la configuración de la

replicación:

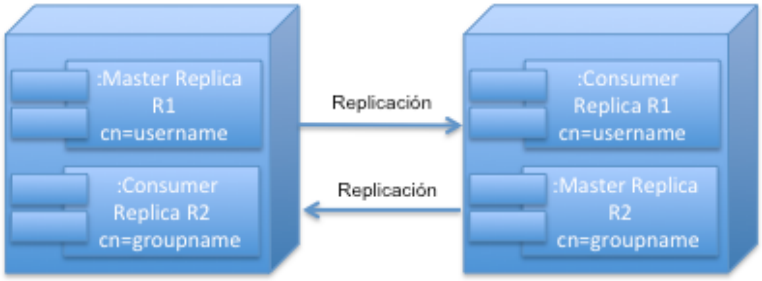

Figura 2.5 Esquema de replicación multimaestro

2.7.6.- Autenticación y cifrado de datos en la red

El proceso de autenticación y cifrado entre las diferentes instancias de sistema operativo dentro de la solución de dominio seguro se realizaría por medio de el estándar *SASL* con la idea de reducir la probabilidad de que cualquier dato que viajara a través de los canales de comunicación pudiera ser expuesto ante un tercero y comprometer la seguridad del sistema. El mecanismo seleccionado para habilitar la autenticación por medio de *SASL* en los servidores de directorio sería *GSSAPI,* el cuál estaba disponible únicamente para el sistema operativo Solaris. Con esto el servidor del componente de acceso web tendría que proveer sus propias credenciales para validar su identidad ante el servidor de directorio.

Debido a que todos los datos que circularían entre los componentes de la solución de dominio seguro eran altamente sensibles por el hecho de ser información sobre los tarjetahabientes, se propuso que en todos los canales de comunicación hacia los servidores de directorio se habilitara el protocolo *TLS* junto con el uso de certificados *X.509* emitidos por una entidad certificadora o *CA*, bajo el estándar de la *ITU-T* para *PKI*, en donde el servicio estaría vinculado hacía el puerto *TCP* 636, usado para publicar el protocolo *LDAP* sobre *TLS/SSL,* en lugar del puerto *TCP* 389, que es el puerto estándar por definición para el protocolo *LDAP*[15] de acuerdo al documento publicado por la Autoridad de Números Asignados en Internet, mejor conocida como *IANA*[16], que es una entidad global encargada de administrar los símbolos y números relacionados con los diferentes protocolos de internet, entre otras cosas. Dicha entidad es operada por la Corporación de Internet para los Números y Nombres Asignados, mejor conocida como *ICANN*[17].

2.7.7.- Controles de seguridad para el despliegue de los componentes

Como parte de la definición de controles de seguridad fue establecida una política de búsqueda dentro del dominio seguro en la que fue definido el límite del tamaño de los resultados de una búsqueda, ya que evidentemente un resultado con gran tamaño

[15] *La definición de los puertos TCP estándar es publicada en internet a través de la URL http://www.iana.org/assignments/service-names-port-numbers/service-names-port-numbers.xml*
[16] *El sitio oficial de la IANA se encuentra disponible para consulta pública en internet a través de la siguiente liga http://www.iana.org/*
[17] *El sitio oficial de ICANN se encuentra disponible para consulta pública en internet a través de la siguiente liga http://www.icann.org/*

puede consumir cantidades significativas de memoria, pudiendo provocar un ataque basado en la denegación del servicio, así que el tamaño máximo se configuró igual a 2, ya que el dominio seguro sería usado exclusivamente para autenticar y autorizar usuarios, y no tendría razón alguna para permitir devolución de resultados de tamaño mayor. Particularmente el atributo que fue configurado en el servidor de directorio fue *nsSizeLimit.*
Adicionalmente para el atributo *nsTimeLimit* fue definido un valor igual a 60*,* el cuál indica el tiempo máximo que un servidor puede utilizar para procesar una operación de búsqueda.
Para el servidor del componente de acceso web se configuró el parámetro *ignore-suffix* con un valor igual a `cn=groupname, c=ve, o=InstitucionBancaria`, con la intención de evitar que un usuario envíe búsquedas directamente sobre el nodo que contiene los grupos.
Por otro lado, con el objeto de tener pleno control sobre las peticiones realizadas hacia los servidores de directorio, se propuso que los servidores de Proxy seleccionarían con base en una política hacia cual de las 2 instancias de directorio propuestas enviaría la solicitud. Dicha política estaría basada en 3 parámetros relacionados entre sí, el primero de ellos es conocido como el manejador de la conexión y es identificado como *connection-handler*, el cuál otorga las reglas necesarias para la toma de decisiones cuando una operación es solicitada a través de esta conexión, de tal forma que cuando un cliente solicita realizar un cambio de contraseña, el servidor de Proxy canaliza la solicitud hacia el servidor de directorio que esta designado para realizar las escrituras sobre el sufijo dentro del árbol de directorios, basándose en la ruta sobre la cuál se encuentra el objeto de la operación, identificándola por el *DN* o nombre distinguido, identificado dentro de los servidores de Proxy como *data-view.* El tercer parámetro se refiere a la instancia específica de directorio, pudiendo ser cualquiera de las 2 propuestas, y es identificado como *data-source*, el cuál cuenta con un peso que sirve para determinar el orden en el que son lanzadas las operaciones.
Un beneficio adicional al configurar servidores de Proxy dentro de la solución del dominio seguro, era que automáticamente se obtendría un esquema de balanceo de cargas, y para ello fue seleccionado un algoritmo conocido como *Failover,* cuyo mecanismo está basado en la distribución de peticiones hacia un *data-source* en particular de acuerdo a su peso, y ante una posible falla en el mismo, todas las peticiones son migradas hacia la otra instancia de directorio a través de su *data-source* asociado.

2.7.8.- Controles de seguridad dentro de los sistemas operativos

Antes de definir una configuración para los sistemas operativos de los servidores que alojarían los componentes de software de la solución de dominio seguro, era necesario asegurarse de que dichos sistemas operativos contaran con los últimos niveles de parches requeridos por cada uno de los componentes de la solución. Así que para el caso de los servidores de directorio era necesario contar con al menos la actualización *5* de la versión 10 de *Solaris*, tal y como se indica en las notas de instalación del producto publicadas en el sitio oficial del fabricante, Oracle[18]*.* De igual manera, para el caso de los servidores que alojarían el componente de acceso web, se hizo la respectiva validación para la suite de *IBM Tivoli Access Manager* en su versión 6.1.0, resultando que el sistema operativo mínimo requerido era *SuSE Linux Enterprise Server 10,* sin necesidad de algún *Service Pack* adicional siguiente, tal y como se

[18]*Dicha información está disponible para su consulta a través de la URL http://docs.oracle.com/cd/E19424-01/820-4805/install-notes/index.html*

indica en el documento publicado en el sitio oficial del fabricante IBM[19].

Una vez que fueron especificadas con exactitud las versiones de sistemas operativos requeridos, el siguiente paso fue elaborar una lista de pre-requisitos de instalación que se debían cumplir para el sistema operativo de cada uno de los ambientes con la finalidad de agilizar el proceso de instalación y configuración de cada componente de software del dominio seguro. Es importante mencionar que a pesar de que la instalación, configuración, y afinación del sistema operativo para los diferentes entornos era propiamente responsabilidad de los fabricantes del hardware, el autor de esta memoria era el responsable de indicar los parámetros de cómo debía ser configurado el sistema operativo de acuerdo a los requerimientos de los componentes que conformarían la solución del dominio seguro.

Básicamente la lista de pre-requisitos de instalación para cada uno de los componentes de software se enfocaba en 3 grandes rubros:

a) *Layout de disco*.- Se refiere a la distribución de particiones de los discos, así como el espacio requerido en cada una de ellas y el nombre del directorio que sirve como punto de montaje para el *filesystem*. Esta información se obtuvo derivado del análisis de dimensionamiento de la capacidad.
b) *Parámetros de Kernel*.- Se refiere a todos aquellos parámetros y semáforos que debían ser configurados dentro del núcleo del sistema operativo con valores específicos que permitieran obtener el máximo rendimiento del producto y de la solución completa.
c) *Usuarios y Grupos.-* Se refiere a todos aquellos usuarios y grupos *UNIX* y/o Linux que debían existir previo al proceso de instalación, y que serían utilizados para configuración y administración de cada componente, considerando los niveles de seguridad requeridos por el cliente.

Lo anterior resultaba de suma importancia, debido a que en el momento en el que el equipamiento de hardware llegara a las instalaciones del cliente, los fabricantes de dicha infraestructura, eran los responsables del montaje y colocación de los equipos, así como la instalación y configuración del sistema operativo. De modo que la consultoría tuvo la necesidad de trabajar directamente en equipo con el personal del cliente para definir la lista de pre-requisitos, con el objeto de cumplir con los estándares internos de nomenclatura, así como la definición de valores para los parámetros y el tamaño de las particiones requeridas que permitieran contar con cierta holgura por los siguientes 3 años. En realidad se generó una tabla con la lista de pre-requisitos para cada componente de la solución de dominio seguro, resultando un total de 3 tablas, las cuales se muestran a continuación:

- Servidores para capa de Directorio o *LDAP*.- En este caso el sistema operativo en cuestión es *Solaris 10 Update 5*.

- Servidores de *LDAP Proxy*.- En este caso el sistema operativo en cuestión también es *Solaris 10 Update 5.*

- Servidores de Tivoli Access Manager.- En este caso el sistema operativo en cuestión es *SuSE Linux Enterprise Server 10*.

[19] *Dicha información está disponible para su consulta a través de la URL http://www-01.ibm.com/support/docview.wss?uid=swg24027380*

Servidores para Directorio o LDAP					
	Dispositivo	**Nivel de RAID**	**Tipo de FS**	**Montaje**	**Tamaño**
Layout de Disco	/dev/md/dsk/d0	RAID 1	ufs	/	20GB
	/dev/md/dsk/d1	RAID 1	swap	swap	96GB
	/dev/md/dsk/d3	RAID 1	ufs	/usr	10GB
	swap	RAID 1	swap	/dbcache	80GB
	/dev/md/dsk/d4	RAID 1	ufs	/export/home	7GB
	ldapcores	RAID 5	ufs	/cores	82GB
	ldapbackup	RAID 5	ufs	/var/opt/backup	42GB
	ldapdb	RAID 5	ufs	/var/opt/db	82GB
	ldaplogs	RAID 5	ufs	/var/opt/logs	110GB
	ldapmps	RAID 5	ufs	/var/opt/mps	6GB
	ldapwsBackup	RAID 5	ufs	/wsBackup	6GB
		Parámetro		**Valor**	
Parámetros de Kernel		tcp_conn_req_max_q		1024	
		tcp_keepalive_interval		600000	
		tcp_ip_abort_cinterval		10000	
		tcp_ip_abort_interval		60000	
		tcp_strong_iss		2	
		tcp_co_min		1500	
		tcp_xmit_hiwat		65536	
		tcp_recv_hiwat		65536	
		rlim_fd_max		4096	
		Nombre de Usuario		**Identificador**	
Usuarios y Grupos		ldapmanager		UID=1001	
		Nombre de Grupo		**Identificador**	
		ldap		GID=1001	

Tabla 2.9 Prerrequisitos de configuración en *Solaris 10* para servidor de directorio

Servidores para Directorio o LDAP Proxy					
	Dispositivo	**Nivel de RAID**	**Tipo de FS**	**Montaje**	**Tamaño**
Layout de Disco	/dev/md/dsk/d0	RAID 1	ufs	/	20GB
	/dev/md/dsk/d1	RAID 1	swap	swap	96GB
	/dev/md/dsk/d3	RAID 1	ufs	/var	10GB
	/dev/md/dsk/d4	RAID 1	ufs	/export/home	7GB
	/dev/md/dsk/d5	RAID 1	ufs	/opt	16GB
	ldapproxy	RAID 5	ufs	/var/opt/proxy	8GB
		Parámetro		**Valor**	
Parámetros de Kernel		tcp_conn_req_max_q		1024	
		tcp_keepalive_interval		600000	
		tcp_ip_abort_cinterval		10000	
		tcp_ip_abort_interval		60000	
		tcp_strong_iss		2	
		tcp_co_min		1500	
		tcp_xmit_hiwat		65536	
		tcp_recv_hiwat		65536	
		rlim_fd_max		4096	
		Nombre de Usuario		**Identificador**	
Usuarios y Grupos		ldapmanager		UID=1001	
		Nombre de Grupo		**Identificador**	
		ldap		GID=1001	

Tabla 2.10 Prerrequisitos de configuración en *Solaris 10* para servidor de *proxy*

Servidores para Tivoli Access Manager					
	Dispositivo	**Nivel de RAID**	**Tipo de FS**	**Montaje**	**Tamaño**
Layout de Disco	/dev/sda1	RAID 1	ext3	/	20GB
	/dev/sda2	RAID 1	swap	swap	32GB
	/dev/sda3	RAID 1	ext3	/home	10GB
	/dev/sdb1	RAID 1	ext3	/opt/pdweb	3GB
	/dev/sdb2	RAID 1	ext3	/var	5GB
	/dev/sdb3	RAID 1	ext3	/var/pdweb	16GB
		Parámetro		**Valor**	
Parámetros de Kernel		file-max soft		1024	
		file-max hard		65535	
		Nombre de Usuario		**Identificador**	
Usuarios y Grupos		ivmgr		UID=1002	
		Nombre de Grupo		**Identificador**	
		ivmgr		GID=1002	

Tabla 2.11 Prerrequisitos de configuración en *SLES 10* para servidor de acceso *web*

Dentro de la configuración del sistema operativo, parte de la responsabilidad del autor de esta memoria, involucraba la participación conjunta para la definición de una configuración de seguridad apegada en todo momento a la política de seguridad del cliente, aplicando para las diferentes instancias donde se alojarían los componentes del dominio seguro. Para ello fue necesario establecer de manera conjunta con el *Chief Information Security Officer*, mejor conocido como *CISO*, una serie de controles de seguridad que tendrían por objeto el endurecimiento de los ambientes y con ello reducir la probabilidad de una explotación de posibles vulnerabilidades. Después de haber realizado un análisis para las plataformas de sistema operativo involucradas en la solución de dominio seguro, *Solaris 10 Update 5*, y *SUSE Linux Enterprise Server 10*, se determinó que el proceso para robustecer la seguridad sería dividido en 3 aspectos generales, cada uno incluyendo diferentes puntos, cuya descripción a continuación se menciona de forma muy breve, ya que la descripción de un proceso de endurecimiento de sistema operativo no es el propósito principal de la presente memoria:

- Seguridad en los usuarios del sistema
  - Integridad de cuentas.- Se refiere a la identificación de cada una de las cuentas de usuarios *UNIX* que serían creadas en cada uno de los ambientes, bajo la premisa de que solo sería requerida una cuenta dedicada a la administración del componente en cuestión, alojado en cada instancia de sistema operativo, adicional a la cuenta del usuario administrador *root*. Por otro lado, se acordó restringir los permisos y el tipo de acceso a los recursos del sistema para cada uno de los usuarios y grupos propios del sistema, requeridos para ejecutar algunos procesos y demonios dentro de cada instancia de sistema operativo.
  - Archivos de usuarios.- Se refiere a la revisión y control de los archivos donde reside el perfil de cada usuario *UNIX*. Dichos archivos, por lo general son requeridos al momento en que un usuario ingresa exitosamente al sistema. La idea de la implementación de este control era precisamente evitar cualquier envenenamiento de ruta en la cuenta de un usuario que pudiera derivar en una escalada de privilegios.
  - Acceso de archivos.- Se refiere al control que sería establecido para autorizar o denegar el acceso a archivos y directorios, basado en la definición de permisos *UNIX* estándares. El objetivo principal de este punto era asegurar que cualquier archivo o directorio únicamente pudiera ser leído o manipulado por aquellos usuarios y grupos con los privilegios necesarios.

- Atributos de archivos.- Se refiere a la búsqueda, el análisis, y la modificación de todos aquellos atributos de archivos y/o directorios que probablemente representaran una vulnerabilidad que podría comprometer la instancia de sistema operativo.
- Búsqueda de archivos.- Se refiere a la identificación y modificación de todos aquellos archivos y directorios que tuvieran permisos especiales basados en atributos extendidos, como pueden ser *setuid*, y *setgid* que pudieran llegar a comprometer la seguridad del sistema, así como la configuración del permiso *sticky bit* para los directorios.
- Parámetros de *login*.- Se refiere a la definición de los valores para todos aquellos parámetros que intervienen al momento en que un usuario inicia una sesión basada en terminal *UNIX* a través de un canal de comunicación.
- Política de contraseñas.- Se refiere al ajuste de valores para todos aquellos parámetros relacionados con las reglas de composición y expiración de contraseñas, con el objeto de reducir las probabilidades de que una contraseña débil y vulnerable pudiera llegar a comprometer el sistema

• Seguridad en los servicios de red
  - Integridad de la red.- Se refiere al proceso de revisión y análisis de todos aquellos puertos *TCP* y *UDP* que se encontraban abiertos sin justificación alguna, para posteriormente cerrar todos aquellos que publican algún servicio innecesariamente.
  - Sistema de correo.- Se refiere básicamente al proceso para deshabilitar el sistema de correo *sendmail* que viene integrado por defecto en ambos sistemas operativos usados en nuestra solución de dominio seguro
• Seguridad en *filesystems* y archivos del sistema
  - Archivos de arranque.- Se refiere a la búsqueda, análisis, y modificación de todos aquellos archivos de arranque, cuyo propósito principal es iniciar algún servicio, con el objeto de mitigar al máximo posible cualquier riesgo inherente a la naturaleza de dichos archivos de arranque, por el hecho de ser pequeños programas escritos en *Shell script.*
  - Visibilidad de archivos.- Se refiere a la revisión de todos los archivos del sistema con el objeto de identificar posibles archivos con contenido malicioso capaz de ocasionar daño en el sistema.
  - Auditoría del sistema.- Se refiere al proceso para habilitar el sistema de auditorías en el sistema operativo, con el propósito de establecer una serie de controles sobre todas las acciones realizadas dentro del sistema a través de registros que guardan todas las modificaciones hechas al sistema.

En el caso de la seguridad física, cabe mencionar que el centro de datos donde sería alojada la infraestructura de la solución de dominio seguro ya contaba con sistemas de control de acceso basados en el uso de dispositivos biométricos, así como cámaras de video vigilancia.

El procedimiento detallado de endurecimiento que fue ejecutado para los 2 sistemas operativos que conformaban la solución de dominio seguro se muestra con todo detalle en el anexo B. Dicho procedimiento fue parte fundamental del proyecto, ya que de no haber pasado las pruebas de detección de vulnerabilidades, probablemente se hubiesen retrasado los tiempos de liberación. Afortunadamente el autor de esta memoria contaba con una amplia experiencia sobre temas relacionados con la detección de vulnerabilidades, y esto permitió reducir drásticamente los tiempos

dedicados para el endurecimiento de los sistemas operativos. En realidad el proceso detallado en el anexo B fue automatizado a través de algunos *Shell scripts*.

Es importante mencionar que una vez que fue ejecutado el procedimiento referido en el párrafo anterior, cada uno de los sistemas fue sometido a una revisión exhaustiva por parte del personal del área de seguridad lógica del cliente, utilizando para ello el producto de software, *Symantec Enterprise Security Manager 6.5*[20]*,* el cuál básicamente es una herramienta de detección de vulnerabilidades que funciona a través de un agente que lee y recolecta toda la información requerida para hacer un análisis de vulnerabilidades y posteriormente generar un reporte sobre el mismo. Evidentemente la herramienta de detección de vulnerabilidades detectó y alertó sobre un listado de puertos y servicios abiertos, así como usuarios y grupos relacionados propiamente con los componentes de software que eran parte del dominio seguro, y para ello fue necesario elaborar un documento de justificación que debía contener los siguientes datos:

- Nombre del servicio y puertos abiertos requeridos
- Nombre de los usuarios y grupos requeridos
- Nombre de los directorios y *filesystems* requeridos

El plan de trabajo, el alcance, y la documentación de los procedimientos descritos en el anexo B fue elaborada por el autor de esta memoria, para posteriormente ser ejecutado por el mismo, teniendo el respaldo de los fabricantes de tecnología involucrados durante la fase de transición del servicio.

2.8.- Aceptación de la propuesta

Después de que el cliente finalmente dio el visto bueno y aceptó de manera ex profesa la propuesta de solución, el siguiente paso fue elaborar una matriz donde se plasmara de manera general la descripción de las actividades a ejecutar, incluyendo el dueño o responsable para cada una de ellas durante la fase de transición o implementación.

2.9.- Estrategia de transición

Como parte de la estrategia de transición del servicio resultaba muy importante delimitar el alcance de cada una de las actividades que formarían parte del contrato en donde los servicios asociados a la solución propuesta cobraban un valor crítico, luego entonces era necesario establecer los criterios de aceptación para cada una de las tareas que formarían parte del plan de trabajo, y la más importante que era el resultado final de la implementación en la que se ejecutaría una serie de pruebas funcionales previas a la liberación a producción, con el objeto de evaluar la solución completa y lograr la entera satisfacción del cliente. El plan de trabajo fue dividido principalmente en 2 grandes partes, una para la parte correspondiente a la capa de los servidores de directorio y la capa del componente de acceso web, y otra para el proceso de endurecimiento de sistema operativo.

Los criterios de aceptación estaban de alguna manera amarrados al plan de trabajo,

[20] *Las especificaciones técnicas del producto se encuentran en la ficha técnica disponible para consulta pública a través de la siguiente liga http://eval.symantec.com/mktginfo/enterprise/fact_sheets/ent-factsheet_enterprise_security_manager_6.5_06-2005.en-us.pdf*

en donde el elemento principal para decidir si el proyecto sería aceptado o no, era el programa de pruebas finales, así como las 2 semanas posteriores a la liberación de la plataforma hacia la fase de operación, en la que la consultoría colocaría un recurso en sitio diariamente para atender y resolver cualquier incidente que se presentara causando algún tipo de degradación o incluso interrupción parcial o total en el servicio. Cabe señalar que durante la fase de transición existiría un administrador del proyecto encargado de llevar un control de posibles desviaciones, así como atenderlas en el cumplimiento de las tareas, y asegurar la calidad en la entrega del proyecto.

2.9.1.- El contrato y la declaración de trabajo

La declaración del trabajo a realizar, refiriéndose particularmente a los servicios de implementación de la solución, se integraría de manera formal al contrato a celebrar entre el cliente y la consultoría, de tal forma que específicamente esta declaración de trabajo contara con su propia sección de firmas, en donde el cliente ratificaría la aceptación de los términos y condiciones de la propuesta. Es importante mencionar que el plazo de vigencia para la aceptación de los términos y condiciones que regirían la declaración de trabajo, así como también para los precios estipulados, sería de 30 días naturales a partir de la entrega de la propuesta.

En caso de que no se lograra obtener la firma de aceptación en el documento de la declaración de trabajo por parte del cliente dentro del plazo de vigencia de la propuesta, entonces esta perdería toda validez legal y comercial, con lo cuál la consultoría quedaría libre de toda obligación para prestar los servicios de implementación asociados a la solución de dominio seguro descritos, y en otro ámbito, los precios correspondientes contenidos en la propuesta económica quedarían obsoletos y podían ser susceptibles a incrementos, debiendo en todo caso remitirse a la solicitud de una nueva cotización.

Resulta relevante señalar que el cliente tenía la libertad de cancelar los servicios sin cargo alguno, siempre y cuando se emitiera un aviso de cancelación por lo menos 15 días hábiles antes de que dicha cancelación tuviera efecto, en cualquier otro caso, al momento de la cancelación el cliente debía pagar todos los servicios suministrados, incluyendo entregables y/o configuraciones hasta la fecha de terminación y el cliente recibiría todo el trabajo en progreso por el que pagó.

2.9.2.- Entidades involucradas en el proyecto

Es importante señalar que en el momento en que el cliente aceptó de manera formal la propuesta para la ejecución del proyecto completo, se obtuvieron las firmas correspondientes para la celebración del contrato mencionado. Los compromisos establecidos en el mismo aplicaban únicamente para 2 partes o entidades involucradas, siendo una de ellas el cliente y la otra, la consultoría, ya que los fabricantes de las tecnologías de hardware y software eran únicamente los proveedores de la infraestructura, así que para la adquisición de la misma se elaboro un contrato por separado entre el cliente y cada uno de los fabricantes en cuestión.

2.9.3.- Inicio y planeación del proyecto

Como parte del proceso de planeación de la transición se incluyó la generación de una matriz general de responsabilidades específicamente para la fase de implementación o puesta en marcha de la solución en donde se especificaban y delimitaban las responsabilidades de cada una de las partes dentro del proceso completo de implementación. A continuación se presenta la tabla 2.12, donde se muestra dicha matriz general de responsabilidades:

Actividad	Consultoría	Área del Cliente	Fabricantes de Hardware
Instalación física de la infraestructura de hardware.			IBM y Sun
Instalación, y configuración de Sistemas Operativos			IBM y Sun
Afinación de Sistemas Operativos	X	Infraestructura	IBM y Sun
Instalación y configuración de los productos de software del dominio seguro	X		
Afinación de los productos de software del dominio seguro	X		
Ejecución de pruebas Unitarias	X	Infraestructura	
Carga de usuarios	X	Seguridad Lógica	
Endurecimiento de Sistemas Operativos	X	Seguridad Lógica	
Ejecución de pruebas de vulnerabilidades	X	Seguridad Lógica	
Ejecución de pruebas integrales	X	Seguridad Lógica Infraestructura Redes Diseño y Desarrollo	
Liberación a producción	X	Seguridad Lógica Infraestructura Redes Diseño y Desarrollo	
Soporte a la solución después de haber sido liberada	X		

Tabla 2.12 Matriz general de responsabilidades durante la fase de transición

2.9.4.- Descripción de las actividades

Para poder concluir exitosamente con la implementación de la solución del dominio seguro, se requería la instalación, configuración, y afinación de los diferentes productos referidos anteriormente en la sección 2.1, los cuales se encargarían de funcionar como una entidad única cuya misión sería otorgar el servicio de autenticación y autorización para ser consumido por el servicio de la banca en línea. Así que para facilitar la administración del proyecto desde el inicio hasta su término se dividieron las actividades en dos grupos. Particularmente al referirse a los componentes de la solución de dominio seguro se observo claramente que la estrategia sería clasificar las actividades de acuerdo a las tareas asociadas a cada componente especifico, es decir, un grupo de actividades para el componente de acceso web y para el componente de directorio. Así mismo se debía elaborar el listado de actividades relacionadas con el endurecimiento de los sistemas operativos involucrados. Basado en dicha segmentación de actividades, el objetivo sería ejecutar un listado de pruebas unitarias al final de la implementación de cada componente para validar el buen funcionamiento del mismo de manera previa a la ejecución de las

pruebas integrales y de aceptación. En realidad, los scripts de pruebas unitarias debían estar basados en los casos de uso cotidianos que se presentarían durante la fase de operación.

Por otro lado, con respecto a las pruebas necesarias para validar la efectividad del proceso de endurecimiento de sistemas operativos, se debería seguir el procedimiento regular correspondiente para la detección de vulnerabilidades basado en la herramienta de seguridad referida anteriormente en la sección 2.7.8. Toda vez que dicha herramienta emitiera un diagnostico libre de vulnerabilidades, entonces las pruebas de aceptación para este punto en particular serían calificadas como exitosas, obteniendo la aceptación por parte del cliente.

Prácticamente el marco de actividades a realizar por parte de la consultoría define de manera general el alcance real de los servicios inherentes al proyecto, objeto de estudio de esta memoria. A continuación se muestra el listado de actividades para la implantación del dominio seguro respetando la interdependencia de las mismas:

- Reunión de inicio de proyecto
- Revisión de ambientes donde se alojaría el componente de directorio
- Instalación del componente de directorio
- Configuración del componente de directorio
- Afinación del componente de directorio
- Instalación del componente de proxy de directorio
- Configuración del componente de proxy de directorio
- Afinación del componente de proxy de directorio
- Ejecución de pruebas unitarias para el componente de directorio y proxy de directorio
- Revisión de ambientes donde se alojaría el componente de acceso web
- Instalación del componente de acceso web
- Configuración del componente de acceso web
- Afinación del componente de acceso web
- Ejecución de pruebas unitarias para el componente de acceso web
- Carga de usuarios en servidores de *LDAP* a través de un archivo *LDIF* generado a partir de *RACF*
- Configuración de endurecimiento para los sistemas operativos
- Ejecución de pruebas de aceptación para la detección de vulnerabilidades en los sistemas operativos
- Ejecución de pruebas integrales y de aceptación
- Transferencia de conocimiento hacia los operadores / administradores de la solución
- Acompañamiento al cliente durante 15 días posteriores a la liberación con la intención de brindar soporte a la solución.

2.9.5.- Requisitos

Basado en el marco de actividades a ejecutar fue definida una serie de requisitos necesarios para la correcta y completa ejecución de todas y cada una de las actividades en cuestión. A continuación se menciona el listado de dichos requisitos:

- Acceso remoto o local hacia todos y cada uno de los servidores dedicados a la solución del dominio seguro.
- Contraseña temporal del usuario administrador del sistema, conocido como

root, en cada una de las instancias de sistema operativo que forman parte del dominio seguro.

- Sistemas operativos que debían estar configurados de acuerdo a las tablas 2.9, 2.10, y 2.11, que contienen los pre requisitos de instalación y configuración para cada producto. Además de que cada una de las instancias debía tener instalados los últimos parches liberados por el fabricante.
- Conectividad hacia la red *SAN* donde residirían los volúmenes requeridos por parte de los servidores de directorio para alojar las bases de datos.
- Conectividad a través de la red *LAN* hacia cada uno de los segmentos de red requeridos.
- Conectividad sobre un mismo segmento de red hacia los balanceadores de carga web.
- Conectividad sobre un segmento de red hacia los servidores de aplicaciones.
- Contar con el archivo *LDIF* extraído a partir del servidor de *RACF.*
- Acceso físico a las instalaciones del cliente donde debía ser provisto un lugar de trabajo, que incluyera al menos un asiento, una mesa para colocar una laptop, una toma eléctrica 100/240 VAC, y una línea telefónica.
- Todas las reuniones de trabajo para dar seguimiento al progreso del proyecto debían ser en las instalaciones del cliente. Así mismo debía otorgar el consentimiento para que el desarrollo de los entregables se pudiera llevar a cabo fuera de sus instalaciones.
- Periodos de tiempo disponibles sobre los cuales se ejecutarían las pruebas integrales y de aceptación, así como la transición a la fase de operación, estos periodos son mejor conocidos como ventanas de tiempo.
- Infraestructura de monitoreo para la plataforma de dominio seguro una vez que haya sido liberada a producción, con el objeto de obtener indicadores de desempeño, mejor conocidos como *KPI*s, y con ello determinar los múltiplos de ganancia a partir del antes y el después.
- Usuario de *VPN* que permita contar con la conectividad remota a través de un túnel cifrado para acceder a los servidores cuando sea requerido durante la fase inicial de la operación y brindar el soporte acordado por un periodo de 15 días posteriores a la liberación.
- Disponibilidad de personal de diferentes áreas dentro del departamento de informática por parte del cliente para apoyos eventuales en la realización de los trabajos, dicho personal debía contar con los conocimientos mínimos requeridos para el cumplimiento de las actividades correspondientes a cada área, referidas en la tabla 2.12, solicitando particularmente los siguientes perfiles:
    - Administrador de Seguridad.- Este pertenece al área de seguridad lógica, y eventualmente sería el responsable de ejecutar la herramienta de detección de vulnerabilidades *Symantec ESM.*
    - Administrador de Redes.- Este pertenece al área de redes.
    - Administrador de Sistemas.- Este pertenece al área de infraestructura.
    - Administrador de Almacenamiento.- Este pertenece al área de infraestructura.
    - Administrador de Cambios.- Esta figura pertenece al área de gestión de cambios, la cuál se encontraba parcialmente fuera del área de informática.
    - Administrador de Proyecto.- Esta figura no pertenece directamente al área de informática, ya que en realidad depende del departamento de proyectos, que es una unidad de negocio independiente.

2.9.6.- Personal del proyecto

Una vez que la infraestructura de hardware fue instalada hasta el nivel de sistema operativo, fue definido un grupo de trabajo que estaría formado por recursos por parte de la consultoría, así como un grupo de recursos por parte del cliente, ambos designados para la ejecución del proyecto durante su fase de transición. Cada persona tenía responsabilidades muy específicas dentro del proyecto, el autor de esta memoria personalmente debía trabajar muy de cerca con el personal del cliente, perteneciente a las diferentes áreas mencionadas en la matriz de responsabilidades, con el objeto de que la configuración de la solución estuviera alineada a los requerimientos del negocio en todos sus ámbitos. Es importante resaltar que la consultoría contaba con la capacidad para ajustar los niveles de personal según fuera apropiado para cumplir con las obligaciones adquiridas en el contrato, y con base en ello a continuación se presenta la tabla 2.13, que muestra las responsabilidades y habilidades requeridas para cada rol dentro de la consultoría:

Rol	Responsabilidades
Gerente de Proyecto de la consultoría	Proveer la dirección y controlar las actividades del proyecto, así como presentar los entregables, y ejercitar los procedimientos de escalamiento según sea apropiado.
Consultor (Jaime Varela)	1. Instalar, Configurar, y afinar los productos de software que conforman la solución del dominio seguro, así como la ejecución de pruebas unitarias e integrales. 2. Ejecutar el procedimiento de endurecimiento de sistemas operativos para las instancias que conforman la solución del dominio seguro. 3. Compartir el conocimiento hacia el personal del cliente. 4. Elaborar las memorias técnicas. 5. Brindar soporte en sitio durante 15 días posteriores a la liberación de la plataforma.

Tabla 2.13 Responsabilidades adquiridas y habilidades requeridas para cada rol por parte de la consultoría

Como parte del personal del cliente, se debían proveer los recursos definidos en la siguiente tabla:

Rol	Responsabilidades
Gerente del proyecto	1. Proveer dirección y guía al personal interno por parte del cliente. 2. Proveer la información y recursos requeridos descritos como parte de las responsabilidades del cliente. 3. Estar disponible y en sitio por si se requiere resolver problemas y responder por las responsabilidades del cliente. 4. Recibir cualquier entregable creado para el cliente. 5. Cuenta con la autoridad necesaria para aceptar o no los entregables definidos en el punto 2.9.14. 6. Llevar a cabo los procedimientos de escalamiento apropiados para mantener el proyecto dentro del plan de trabajo.
Personal Técnico	Personal con conocimientos técnicos, que cuenta con la capacidad para resolver dudas al consultor responsable de implementar la solución de dominio seguro. Básicamente los perfiles requeridos son los mencionados en la sección 2.9.5

Tabla 2.14 Responsabilidades adquiridas y habilidades requeridas para cada rol por parte del cliente

2.9.7.- Transferencia de conocimiento

Durante la reunión inicial de arranque del proyecto se acordó que durante la etapa de implementación habría una persona por parte del cliente dedicada a observar todo el proceso completo, teniendo el derecho de que todas sus dudas expuestas debían ser respondidas adecuadamente y a entera satisfacción de dicha persona. En caso de no obtener la respuesta esperada, la situación se podía escalar hasta la obtención de una respuesta que lograra clarificar por completo la duda.

Evidentemente era necesario que el conocimiento lograra permear particularmente al personal encargado de la operación y administración de la solución, todo ello enfocado a la utilización futura de dicho conocimiento para la atención de incidentes y resolución de problemas.

Por otro lado, existía el compromiso de elaborar una memoria técnica describiendo todo el proceso de instalación, configuración, y afinación de los componentes del dominio seguro, misma que debía ser entregada y revisada por el cliente al final del proyecto. En la memoria técnica debían venir plasmados todos los flujos de tareas, de tal modo que el cliente tuviera el conocimiento necesario para entender la ruta necesaria para llegar a la configuración final, así como para atender requerimientos que implicaran cambios posteriores a la configuración inicial de la solución.

2.9.7.1.- Talleres para reforzar la gestión del conocimiento

Previamente durante la fase de negociación había sido acordado con el cliente que de manera complementaria se impartiría un taller con el propósito de reforzar la transferencia de conocimiento. Las condiciones para poder llevar a cabo dicho taller básicamente dependían del comportamiento de la plataforma recién liberada, ya que de no lograr la estabilidad requerida, el tiempo sería consumido en otras actividades, pero afortunadamente los 2 últimos días de la estancia en sitio con el cliente fueron destinados a la impartición del taller, teniendo una duración de 8 horas efectivas diarias, y abarcando temas relacionados directamente con el componente de acceso web, y con el componente de directorio.

El taller estuvo orientado a los operadores de la plataforma con la intención de proveer el conocimiento necesario para identificar, atender, y resolver problemas relacionados con el servicio de autenticación y autorización de la *BNET,* además de volver más eficiente la administración y operación de la misma, ya que precisamente la capacidad para proveer un buen servicio radicaría en las habilidades y conocimientos de aquellas personas responsables de la toma de decisiones relacionadas con la plataforma. La idea era que dicho conocimiento basado principalmente en experiencias, ideas, y juicios, se lograra transformar en un núcleo de sabiduría para el personal del cliente encargado de la gestión del servicio de la *BNET.*

El conocimiento transferido hacia el personal del cliente eventualmente sería registrado en los Sistemas de Administración del Conocimiento, mejor conocidos como *SKMS,* en los cuales se guarda la información referente a la experiencia y el nivel de habilidades del personal (Office of Government Commerce, 2007d).

2.9.8.- Administración de cambios

La “solicitud de un cambio” u “orden de cambio” significa un acuerdo de cambio o la modificación de los entregables con respecto a lo que inicialmente fue definido en el documento de declaración de los trabajos, así que cualquier solicitud por parte del cliente para los pedidos de cambio estarían sujetos a los procedimientos establecidos en la siguiente sección y siempre debían hacerse por escrito.

- Los cambios podían ser solicitados por cualquiera de las 2 partes.
- Todos los cambios debían solicitarse por escrito, debían ser aceptados y firmados mutuamente por cada una de las partes.
- Los cambios solicitados podían representar una tarifa para el cliente, dependiendo del esfuerzo requerido para la ejecución del mismo, así como el impacto dentro del proyecto. Esto después de haber sido evaluado y analizado.
- Las ordenes de cambio serían procesadas tan pronto como fuera comercialmente razonable.
- Las ordenes de cambio debían incluir lo siguiente:
    - Una descripción de los servicios adicionales que serían proporcionados.
    - Una declaración sobre el impacto de los servicios adicionales o de los cambios relativos a los servicios inicialmente definidos.
    - La agenda estimada para la realización de los servicios especificados en el pedido de cambio, y el impacto sobre la determinación de los precios y pagos, en caso de existir.
    - Identificar cuando corresponda las responsabilidades y los roles específicos afectados por la orden de cambio.
    - La documentación que debía ser modificada o proporcionada como parte de los servicios adicionales.

Es importante señalar que dentro del contrato existía una gran cantidad de clausulas de carácter comercial asociadas a la administración de cambios, que al igual que muchas otras no se mencionan en esta memoria por el hecho de estar fuera del alcance de la misma.

2.9.9.- Proceso de escalamiento

La resolución oportuna de todos los problemas que se llegaran a presentar durante la fase de transición resultaría esencial para mantener el control del proyecto y obtener la entera satisfacción del cliente. El propósito del proceso de escalamiento era precisamente ayudar a asegurar que todos los problemas fueran identificados y resueltos de manera oportuna. Dicho proceso de escalamiento proporcionaría un mecanismo para alertar a los gerentes del proyecto y al personal de niveles jerárquicos superiores sobre problemas que no fueran resueltos dentro de los tiempos previstos y que representaran un impacto y un riesgo mayor dentro del proyecto. El flujo de escalamiento estaría sujeto a ciertas consideraciones que a continuación se mencionan:

- Informar del problema inicialmente al administrador o líder del proyecto .
- Si el problema no logra ser resuelto en el presente nivel, entonces se notificará al siguiente nivel a través de un informe sobre la situación del problema.
- Si el problema no se logra resolver dentro de un plazo de 30 días naturales después de haber sido notificado, entonces debe ser comunicado al comité directivo del proyecto.
- Problemas internos de la consultoría podrían ser escalados hacia el director de la misma con el propósito de lograr su resolución.

2.9.10.- Plan de trabajo

El plan de trabajo correspondiente a la implementación de la solución del dominio seguro fue presentado ante el cliente para obtener su retroalimentación con respecto a los tiempos estipulados y al alcance de dichas actividades por si mismas, en dicho plan de trabajo claramente se describían de manera puntual todas las actividades y tareas que debían ser realizadas para concluir en tiempo y forma, como parte del compromiso adquirido por parte de la consultoría.

Como se ha mencionado anteriormente, el plan de trabajo fue dividido en 2 grandes grupos, uno de los cuales contenía todas las actividades relacionadas con la instalación de la infraestructura de hardware y sistema operativo, en el cuál la participación del autor de esta memoria fue muy limitada, ya que la realización de dichas tareas era responsabilidad de los diferentes fabricantes. Esta primera fase que de acuerdo al cronograma iniciaría el día 27 de Abril, era vista como prerrequisito para la siguiente fase, cuya fecha de inicio era el día 5 de Mayo, y que correspondía propiamente a la implementación de los 2 componentes del dominio seguro, incluyendo la carga de usuarios, y las tareas relacionadas con el endurecimiento de los sistemas operativos, todas estas consideradas como responsabilidad de la consultoría.

Cabe mencionar que el tiempo dedicado para el proceso de endurecimiento constaba de solo 4 días, ya que prácticamente debía ser ejecutado en 2 días para posteriormente dedicar 2 días más a la ejecución de las pruebas de vulnerabilidades. Aparentemente resultaba corto el tiempo dedicado para esta actividad, sin embargo, el proceso estaba controlado, de tal forma que solo se tenía que ejecutar una serie de comandos de manera secuencial, como se puede observar en el anexo B.

Básicamente las actividades debían comenzar el día en que llegara el equipamiento de hardware, por lo que el cliente debía tener listas las adecuaciones eléctricas, así como la infraestructura de comunicaciones requerida. Evidentemente, como se puede apreciar en la figura 2.6, el tiempo solicitado para la implantación de la solución era reducido, ya que desafortunadamente no fue posible negociar con el cliente una extensión debido a que los directivos del área de informática dentro de la organización del cliente habían hecho el compromiso de liberar la solución a una fase de producción al menos una semana antes de que iniciara el mes de Junio, así que los tiempos definidos y acordados no daban lugar para cometer errores que retrasaran la ejecución del proyecto más allá de 1 ó 2 días, colocando mucha presión en todos los participantes.

Como se mencionó anteriormente la fecha tentativa de inicio era el día 27 de Abril de 2011, teniendo como fecha estimada de término el día 22 de Mayo del mismo año, que sería la fecha en que se ejecutaría el cambio de plataforma. A continuación se presenta la figura 2.6, que contiene el cronograma que indica las fechas y las actividades a nivel macro del plan de trabajo propuesto al cliente:

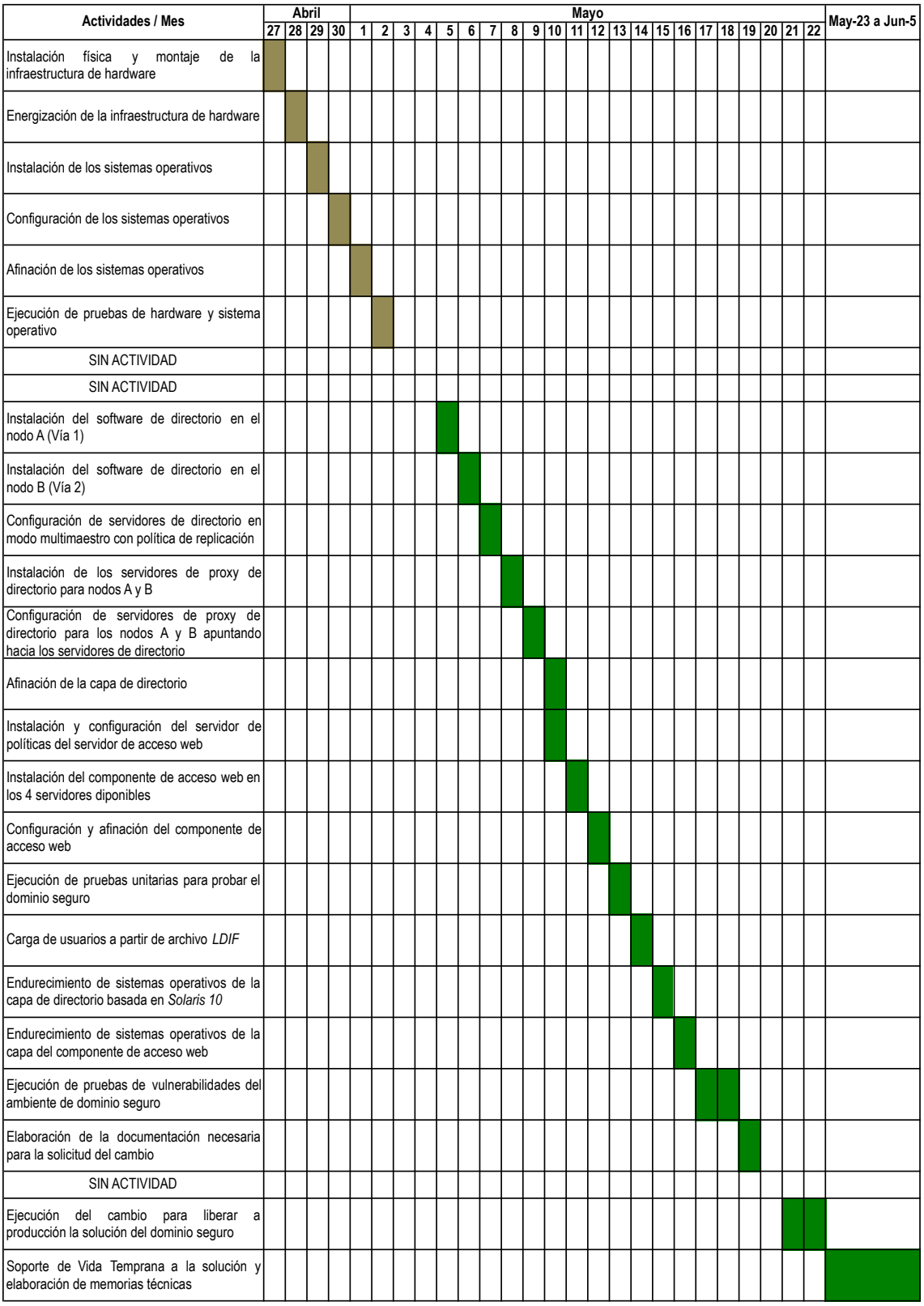

Figura 2.6 Cronograma de actividades

2.9.11.- Procedimiento de implementación

Con base en el cronograma mostrado anteriormente, se puede apreciar que todo el proceso de implementación de los componentes del dominio seguro se ejecutaría de

manera controlada, bajo la premisa de que ninguna actividad podría ser ejecutada si su antecesora no concluyó con éxito, dando como resultado una interdependencia secuencial entre las mismas.
De manera independiente al plan de trabajo, el autor de esta memoria tuvo la responsabilidad de elaborar un documento donde se debía describir paso a paso el procedimiento de instalación, configuración, y afinación de todos los componentes del dominio seguro. Dicha información se encuentra disponible en el anexo A, y cabe señalar que por motivos de confidencialidad, algunos datos como son direcciones *IP*, nombres de *host,* y nombres de usuarios fueron cambiados, y no necesariamente hacen referencia a un dato real. Sin embargo esto no altera el contenido del procedimiento mostrado.

2.9.12.- Exclusiones del alcance

A continuación se presentan algunas actividades que quedaban fuera del alcance y de toda responsabilidad por parte de la consultoría con respecto al proyecto:

- Adecuaciones eléctricas para la instalación de la infraestructura de hardware.
- Instalación física de la infraestructura requerida para la solución de dominio seguro.
- Instalación y configuración de infraestructura de red, como son cableado estructurado, *switches, routers*, etc.
- Generación de respaldos de la configuración de la solución de autenticación y autorización previa a la implantación de nuestra solución propuesta.
- Análisis de arquitectura y/o migración de aplicaciones.
- Instalación de sistemas operativos en los servidores que formarían parte del dominio seguro o en cualquier otro ambiente.
- Generación de archivo *LDIF* a partir de la solución de *RACF.*
- Instalación de cualquier software o agente aparte de los 2 componentes de la solución de dominio seguro indicados expresamente dentro del documento de la declaración del trabajo, siendo estos el software para la capa de directorio, *Sun Java System Directory Server Enterprise Edition 7.0* y el software para la capa de control de accesos, *Tivoli Access Manager 6.1.*
- Reconfiguración en los balanceadores de carga web para direccionar el tráfico hacia los nuevos servidores de control de acceso web.
- Reconfiguración en los servidores de aplicaciones web del servicio de la banca en línea o de cualquier otro ambiente.
- Codificación y/o recompilación de aplicaciones.
- Cualquier otra actividad no explícitamente indicada dentro del documento de declaración de trabajo, en el cuál únicamente se describían las actividades de implementación de los componentes de la solución de dominio seguro.

2.9.13.- Factores críticos de éxito

Para la realización de la propuesta se consideraron los siguientes supuestos y factores críticos de éxito:

- El cliente se encargaría de proveer todos los requisitos descritos en el apartado 2.9.5, subtitulado como “requisitos”.
- El personal técnico de las diferentes áreas involucradas por parte del cliente

debía contar con el conocimiento necesario para brindar el apoyo requerido cuando fuera necesario atender incidentes.

- Debía existir un líder de proyecto por parte del cliente con las habilidades necesarias para poder gestionar todos los ciclos dentro de la fase de transición del proyecto.
- Se debía contar con el apoyo y la información referente a todos los componentes del servicio de la banca en línea cuando se le requiriera a los distintos responsables.
- Se debían realizar juntas semanales en las instalaciones del cliente para revisar los avances del proyecto. Los días propuestos para estas juntas fueron los lunes.
- Se debía elaborar un documento de acuerdo de confidencialidad con el afán de establecer a través de una figura legal el marco de referencia para no compartir con terceros la información relacionada con el proyecto.

2.9.14.- Entregables

Los entregables se refieren principalmente a todos aquellos documentos que debían ser entregados al cliente como parte del alcance del proyecto. Dichos documentos debían ser entregados en formato impreso, acompañados de una copia en formato electrónico. Para la fase de transición básicamente fueron definidos los siguientes entregables:

- Propuesta de Solución.- La propuesta de la solución se tuvo que revisar de manera conjunta con personal del cliente, y el enfoque estaba muy orientado a la arquitectura de la solución, incluyendo la administración de la capacidad y de la disponibilidad.
- Cronograma del Plan de Trabajo.- El cronograma del plan de trabajo fue entregado al líder de proyecto por parte del cliente para obtener su visto bueno y con ello poder comenzar con las actividades.
- Reportes de Avances.- Estos reportes debían ser entregados al líder de proyecto por parte del cliente con una frecuencia semanal con el objeto de contar con suficiente visibilidad que permitiera la detección de cualquier desviación.
- Memoria Técnica.- En este documento se debía dejar plasmado todo el proceso de instalación y configuración de los productos de la solución del dominio seguro, al igual que para el proceso de endurecimiento de los sistemas operativos. Se debía indicar sobre el estado de la configuración final para que cualquier administrador del área de seguridad lógica por parte del cliente contara con el conocimiento necesario para efectuar las tareas de operación y mantenimiento que eventualmente se debían realizar durante la fase de operación. En realidad se elaboraron 2 documentos de memoria técnica, uno para los componentes del dominio seguro, y otro para el endurecimiento de los sistemas operativos.
- Script de pruebas unitarias.- Dentro de este documento se debía especificar el conjunto de pruebas que se ejecutarían para probar aisladamente la funcionalidad de cada uno de los componentes de la solución de dominio seguro por separado, basado en los casos de uso definidos. Es importante aclarar que este documento formaría parte del certificado de aceptación final por parte del cliente en donde se indicaría explícitamente que aceptaba la liberación de la plataforma para su operación.
- Script de pruebas integrales.- Dentro de este documento se debía especificar el

conjunto de pruebas a realizar para probar la solución completa de dominio seguro desde el punto de vista funcional. Cabe mencionar que adicionalmente como parte de las pruebas funcionales, el cliente ejecutó una serie de scripts de simulación de carga a través de la herramienta *Loadrunner*[21] de la marca *HP,* para lo cuál la participación de un servidor fue prácticamente nula, ya que dicho proceso fue completamente administrado y ejecutado por personal del cliente como parte de sus procedimientos de aseguramiento de calidad y sin dar a conocer el resultado de las pruebas de carga, solo fue notificado que se podía continuar con las siguientes actividades.

- Reporte de operaciones.- Este documento incluye todos los incidentes reportados durante los 15 días posteriores a la liberación de la plataforma, así como algunas gráficas de rendimiento durante el mismo periodo de tiempo.

Con base en la descripción anterior para cada uno de los entregables, a continuación se presenta la tabla 2.15 que contiene la relación de los mismos, especificando claramente que todos ellos requieren aceptación por parte del cliente:

Número	Descripción	¿Requiere Aceptación?
1	Propuesta de Solución	Si
2	Plan de Trabajo	Si
3	Reportes de Avances Semanales	Si
4	Memoria Técnica	Si
5	Script de Pruebas Unitarias	Si
6	Script de Pruebas Integrales	Si
7	Reporte de Operación por 15 días posteriores a la liberación	Si

Tabla 2.15 Relación de entregables

2.9.15.- Criterios de aceptación

Los entregables debían ser formalmente aceptados por medio de un certificado de aceptación en donde el cliente firmaba de conformidad. El término "certificado de aceptación" fue utilizado para definir una serie de documentos especializados usados para aceptar entregables especializados. En realidad se entregaría un certificado de aceptación para cada entregable referido en la tabla 2.15.
Los criterios de aceptación dentro de la fase de transición se tornaban vitales debido a que el cumplimiento de los objetivos del proyecto prácticamente dependería de los factores críticos de éxito, que en realidad son identificados como requisitos dentro de la fase de transición. Así que a continuación se presentan los principales criterios de aceptación mediante los cuales se regiría la aceptación del proyecto:

- Se aceptaba la propuesta de solución, toda vez que se explicaban perfectamente los niveles de soporte, así como las características técnicas de la solución indicando las capacidades de la misma.
- Se aceptaba el plan de trabajo siempre y cuando estuviera alineado a los tiempos esperados por el cliente.
- Se aceptaban los reportes de avances, siempre y cuando estuvieran firmados

[21] *Las especificaciones técnicas del software HP LoadRunner se encuentran disponibles para consulta pública en el sitio oficial del fabricante (HP) a través de la siguiente liga http://www8.hp.com/mx/es/software-solutions/software.html?compURI=1175451#.Udo0PnB2lvA*

por el administrador del proyecto por parte de la consultoría, ya que nos indicaban el progreso de las actividades del proyecto.

- Se aceptaba el documento de memoria técnica, ya que se plasmaba todo el contenido técnico necesario y suficiente para el entendimiento de la solución. El contenido debía haber sido revisado por al menos 3 elementos del personal del cliente.
- Se aceptaban los scripts de pruebas unitarias, ya que cumplían con la expectativa del cliente, respecto a probar la funcionalidad de los diferentes productos involucrados. El resultado de todas y cada una de las pruebas debía ser exitoso.
- Se aceptaba el script de pruebas integrales, toda vez que cubrían perfectamente los flujos correspondientes a los casos de uso del servicio de la banca en línea, previo a ejecutar las pruebas de simulación de carga directamente por parte del cliente y sin intervención por parte de la consultoría. El resultado de cada uno de los puntos del script debía ser exitoso.
- Se aceptaba el reporte diario de horas de soporte en sitio dedicadas a la atención de incidentes durante la fase de operación, siempre y cuando tuviera el visto bueno del gerente del área de seguridad lógica, esto durante los 15 días posteriores a la liberación de la plataforma, mencionando dentro de dicho reporte algunos indicadores clave de desempeño.

2.9.16.- Pruebas unitarias

Una vez que se concluyeron las tareas referentes a los procesos de instalación, configuración, y afinación de los componentes del dominio seguro, se prosiguió con la ejecución de pruebas independientes para cada uno de dichos componentes. El objetivo principal de este tipo de pruebas era justamente probar los componentes de la solución de manera aislada, y con ello tener la capacidad de poder generar un mejor diagnóstico ante una posible falla durante las pruebas integrales o incluso durante las fases de transición y operación.

Es importante señalar que el diseño del plan de pruebas, así como la ejecución del mismo era una responsabilidad compartida entre la consultoría y el cliente, aunque en realidad la responsabilidad de que dichas pruebas resultaran exitosas era una responsabilidad que recaía completamente en la consultoría, ya que cualquier prueba que fallara, debía ser revisada, analizada, y corregida, hasta que la prueba resultara exitosa en un siguiente intento. Por ello mismo, la consultoría tomó el liderazgo para todas las actividades a realizar como parte de dichas pruebas y un aspecto importante fue acotar el alcance de cada punto a probar, para posteriormente tener la capacidad para comparar el resultado deseado contra el resultado arrojado sobre un marco de referencia y con esto evitar la generación de falsas expectativas.

Particularmente para esta sección, referente a las pruebas unitarias, afortunadamente el trabajo en equipo resultó muy eficiente y se logró establecer un acuerdo sobre los casos de uso a probar, basándose en los más comunes que se presentarían durante la fase de operación de la solución. Obviamente las pruebas fueron separadas en 2 grandes bloques, un plan de pruebas para la capa de directorio, incluyendo los servidores de *proxy,* y otro plan para la capa del componente de control de accesos *web*, ya que posteriormente durante la fase de pruebas integrales se probarían los mismos casos de uso sobre la solución completa, incluyendo todos sus componentes.

Inicialmente se implementó la capa de directorio, así que se construyó el escenario para las pruebas resultando la siguiente tabla que contiene la descripción y el archivo requerido para cada uno de los casos de uso a probar, así como los comandos correspondientes a ejecutar desde la consola.

Caso de Uso	Comandos a ejecutar	Archivo Requerido	Resultado Obtenido
Alta de un Usuario	# ldapsearch -h HOSTNAME –p PORT –D cn=root, cn="Directory \ Administrators", o=InstitucionBancaria –w PASSWORD –b ou=banco cn=dummy # ldapadd –h HOSTNAME –p PORT –D cn=root, cn="Directory \ Administrators", o=InstitucionBancaria –w PASSWORD –f add.ldif	add.ldif	Exito
Modificación de un atributo de Usuario	# ldapsearch -h HOSTNAME –p PORT –D cn=root, cn="Directory \ Administrators", o= InstitucionBancaria –w PASSWORD –b ou=banco cn=dummy # ldapmodify –h HOSTNAME –p PORT –D cn=root, cn="Directory \ Administrators", o= InstitucionBancaria –w PASSWORD –f modify.ldif # ldapsearch -h HOSTNAME –p PORT –D cn=root, cn="Directory \ Administrators", o= InstitucionBancaria –w PASSWORD –b ou=banco cn=dummy	modify.ldif	Exito
Cambio de contraseña de un Usuario	# ldapsearch -h HOSTNAME –p PORT –D cn=root, cn="Directory \ Administrators", o= InstitucionBancaria –w PASSWORD –b ou=banco cn=dummy # ldapmodify –h HOSTNAME –p PORT –D cn=root, cn="Directory \ Administrators", o= InstitucionBancaria –w PASSWORD –f passwd.ldif # ldapsearch -h HOSTNAME –p PORT –D cn=root, cn="Directory \ Administrators", o= InstitucionBancaria –w PASSWORD –b ou=banco cn=dummy	passwd.ldif	Exito
Borrado de un Usuario	# ldapsearch -h HOSTNAME –p PORT –D cn=root, cn="Directory \ Administrators", o= InstitucionBancaria –w PASSWORD –b ou=banco cn=dummy # ldapdelete –h HOSTNAME –p PORT –D cn=root, cn="Directory \ Administrators", o= InstitucionBancaria –w PASSWORD "cn=dummy, ou=banco, c=ve, o= InstitucionBancaria"	N/A	Éxito

Tabla 2.16 Casos de uso para servidor de directorio

En seguida se muestra la tabla 2.17 con el contenido de cada uno de los archivos *LDIF* referenciados en los casos de uso anteriores:

Nombre del archivo	Contenido del archivo
add.ldif	dn: cn=dummy, ou=banco, c=ve, o= InstitucionBancaria objectclass: inetOrgPerson cn: dummy sn: dummy givenName: dummy
modify.ldif	dn: cn=dummy, ou=banco, c=ve, o= InstitucionBancaria changetype: modify add: telephoneNumber telephoneNumber: 5555-555555
passwd.ldif	dn: cn=dummy, ou=banco, c=ve, o= InstitucionBancaria changetype: modify replace: userPassword userPassword: XXXXXX

Tabla 2.17 Archivos requeridos por casos de uso para servidor de directorio

Posteriormente al concluir la fase de implementación para el componente de acceso

web, se ejecutó nuevamente un script de pruebas correspondiente a la capa del control de accesos, cuyo alcance básicamente era el mismo, ya que la idea era probar las altas, bajas, cambios, y consultas. A continuación se muestra la tabla 2.18 que contiene los casos de uso para las pruebas, así como los comandos a ejecutar desde la consola de administración basada en línea de comando, conocida como *pdadmin*:

Caso de Uso	Comandos a Ejecutar	Resultado Obtenido
Alta de un Usuario	pdadmin sec_master> user list *dummy* pdadmin sec_master> user create dummy cn=dummy, ou=banco, \ c=ve, o= InstitucionBancaria dummy dummy PASSWORD pdadmin sec_master> user list *dummy*	Exitoso
Modificación de un atributo de un Usuario	pdadmin sec_master> user list *dummy* pdadmin sec_master> user modify dummy account-valid yes pdadmin sec_master> user list *dummy*	Exitoso
Cambio de contraseña de un Usuario	pdadmin sec_master> user list *dummy* pdadmin sec_master> user modify dummy password PASSWORD pdadmin sec_master> user modify dummy password-valid yes pdadmin sec_master> user list *dummy*	Exitoso
Borrado de un Usuario	pdadmin sec_master> user list *dummy* pdadmin sec_master> user delete –registry dummy pdadmin sec_master> user list *dummy*	Exitoso

Tabla 2.18 Casos de uso para servidor de acceso *web*

Es importante aclarar que todos los comandos mostrados en la tabla anterior fueron ejecutados con el usuario administrador, conocido como *sec_master* para el producto *Tivoli Access Manager.*

2.9.17.- Pruebas integrales

En el caso de las pruebas integrales, el cliente pidió que se utilizara su propio *script* de pruebas para validar el buen funcionamiento de la solución recién construida. El resultado de las pruebas integrales era el punto de control más importante antes de comenzar a elaborar la solitud del cambio, esto resultaba crucial para disparar posibles cambios en la configuración, de tal manera que se permitiera asegurar que el resultado esperado fuera realmente lo que se entregaría.
Las pruebas integrales permitirían minimizar la probabilidad de que se presentaran fallas durante la etapa de operación, y al mismo tiempo documentar las soluciones para complementar el conocimiento e incrementar la capacidad para generar diagnósticos.
Una parte muy importante dentro de esta fase de pruebas integrales fueron las pruebas de carga o estrés, para las cuales, fueron emuladas cientos de conexiones que probarían el flujo completo de inicio a fin para el proceso de autenticación y autorización y con ello observar el rendimiento de la solución ante distintos niveles de concurrencia. Para estas pruebas en particular el cliente ejecutó una serie de *scripts* a través de una herramienta propietaria, llamada *HP Loadrunner* en su versión 9.5 y en realidad la participación del autor de esta memoria fue muy limitada ya que solo estuvo en calidad de observador y en algunos momentos tenía que detener o iniciar los servicios del dominio seguro.
Al concluir las pruebas, el cliente generó un reporte con los resultados de las mismas y se informó que afortunadamente la solución había soportado una cantidad máxima de

7,200 usuarios concurrentes antes de que el rendimiento fuera degradado considerablemente y por lo tanto las pruebas fueron calificadas como exitosas y se obtuvo la aprobación por parte de las diferentes áreas al interior de la organización del cliente para seguir adelante con las siguientes fases del proyecto.

2.9.18.- Administración de la liberación de la plataforma

Después de haber ejecutado todo el proceso de instalación y configuración de componentes del dominio seguro, así como la realización de pruebas unitarias y las pruebas integrales, se programó una reunión con personal de las distintas áreas involucradas por parte del cliente y se acordó una fecha tentativa para ejecutar el proceso de liberación a producción en el que prácticamente todo el tráfico de los balanceadores de carga web por donde llegaban los usuarios, sería direccionado hacia los servidores del componente de acceso web, encargados de fungir como la primera capa dentro de la solución de dominio seguro, antes de transferir la solicitud hacia un servidor de directorio en particular. Antes de realizar el cambio previsto, se solicitó ejecutar un respaldo de la configuración de cada uno de los elementos de la solución de autenticación y autorización que operaba en ese momento, al igual que para los balanceadores de carga web.

Los respaldos generados resultarían muy importantes, ya que ante la inminente posibilidad de una falla que pudiera provocar la interrupción del servicio al momento de hacer el cambio, se debía contar con un plan de remediación que retornara al punto de origen para posteriormente hacer un análisis de los errores cometidos y reprogramar nuevamente el cambio para una siguiente fecha.

Por disposición del Comité Administrador de Cambios al interior del cliente se determinó que la fecha tentativa para realizar el cambio no debía estar a menos de 7 días de las fechas en las que se procesaban las nóminas y además se tenía que solicitar una ventana de tiempo durante un fin de semana, preferentemente de sábado para domingo, ya que era cuando se encontraba más baja la actividad. Para ello el gerente del proyecto elaboró una solicitud de cambio, mejor conocida como *RFC*, en la que se explicaba técnicamente en que consistía el cambio y cuales eran los posibles riesgos, así como la propuesta de remediación, y el flujo de actividades para el proceso del cambio. A continuación los principales datos que fueron incluidos en el *RFC:*

- Número Identificador
- Descripción del proceso del cambio
- Caso de negocio o justificación del cambio
- Efecto al no implementar el cambio
- Hora y Fecha para ejecutar el cambio propuesto
- Categoría del cambio propuesto
- Plan de remediación
- Evaluación del riesgo
- Evaluación del impacto
- Firma de autorización
- Fecha y hora de autorización
- Ubicación para la liberación
- Cierre

Posteriormente se elaboró un documento que contenía un análisis del impacto al

negocio, mejor conocido como *BIA*, ya que se podría provocar una eventual falla del cambio dentro del servicio completo de la *BNET*, para lo cuál fue necesario elaborar una matriz en la que se categorizaba el impacto y el riesgo asociados al cambio propuesto. Para llegar a una valoración real de dicho impacto se inició una serie de planteamientos de manera conjunta con el Comité Consejero de Cambios, mejor conocido como *CAB*, a partir de los cuales se concluyó que el impacto a la operación del negocio era calificado como alto, ya que podría presentarse una afectación hacia miles de usuarios del servicio de la banca en línea inhabilitando por completo la capacidad de hacer operaciones bancarias y comprometiendo los niveles de servicio acordados. Además de que existía la probabilidad de afectar incluso la infraestructura de red y comunicaciones al tener que ser reconfigurada para liberar a producción la solución del dominio seguro, pudiendo impactar en otros servicios al interior del cliente. A continuación se presenta la tabla 2.19 que contiene la matriz de categorización de impacto y riesgo que fue utilizada para el cambio solicitado:

	Matriz de Categorización del Impacto y Riesgo del Cambio	
Impacto del Cambio	Alto impacto Baja Probabilidad Categoría 2 de Riesgo	Alto Impacto Alta Probabilidad Categoría 1 de Riesgo
	Bajo Impacto Baja Probabilidad Categoría 4 de Riesgo	Bajo Impacto Alta Probabilidad Categoría 3 de Riesgo
	Probabilidad	

Tabla 2.19 Categorización del impacto y riesgo del cambio

Como parte del análisis de impacto al negocio, se realizó una valoración de riesgo enfocado a identificar los factores que podrían ser disruptivos para el negocio del cliente e impedir la entrega del servicio, provocando un impacto corporativo en los objetivos y políticas. Con base en la matriz anterior, el cambio propuesto fue clasificado como categoría 2, ya que tendría un alto impacto, pero una baja probabilidad de falla, ya que después de haber realizado las pruebas unitarias e integrales, la probabilidad de éxito se estimaba muy cercana al 98%, esto debido a que del 100% de pruebas realizadas, solo alrededor de un 2% había presentado pequeños problemas relacionados con el uso de algunos caracteres especiales en la composición de contraseñas.

Una vez que fue realizado el análisis de riesgos, el siguiente paso por parte del *CAB,* fue evaluar la viabilidad del cambio y para ello tomaron como base, la categoría del riesgo, comparado contra los beneficios potenciales del cambio, así como la urgencia, y los costos asociados propiamente al mismo. Evidentemente el beneficio potencial que el cambio podía otorgar hacia el negocio se relacionaba directamente con el grado de afectación y costos que representaba para el negocio el continuar con la solución de autenticación y autorización anterior que estaba presentando serios problemas de rendimiento.

Por otro lado, tomando en cuenta que la prioridad del cambio se deriva a partir del impacto y la urgencia (Office of Government Commerce, 2007d: p.55), y que ambos atributos tenían valores altos, a continuación se presenta la tabla 2.20, que indica la prioridad estimada para el cambio propuesto, basada en la propia clasificación del cambio:

Prioridad	Cambio Correctivo	Cambio de Mejora
Alta	1.- Corregir la afectación severa para una gran cantidad de usuarios del servicio de la banca en línea, ocasionada por problemas de rendimiento.	1.- Contar con la capacidad para atender la demanda del servicio durante los próximos 3 años. 2.- Reforzar el esquema de seguridad del servicio de la banca en línea.

Tabla 2.20 Prioridad del cambio

A partir de tabla anterior se observa que el cambio propuesto era de tipo correctivo ya que pretendía solucionar un problema existente, pero al mismo tiempo otorgaría beneficios adicionales, por lo cuál también era catalogado como un cambio de mejora.

Una semana después de que el gerente del proyecto hizo la solicitud del cambio, el comité de cambios aprobó formalmente la ejecución del mismo. Cabe mencionar que la aprobación fue escalada a un nivel de dirección de negocio, ya que en ese nivel recaía la responsabilidad de mantener la operación del servicio de la banca en línea. Como parte del proceso de aprobación se sostuvo una reunión de trabajo con el comité de cambios para acordar la fecha definitiva en que se realizaría el cambio propuesto.

A continuación se presenta la figura 2.7, que ilustra claramente la secuencia de actividades sobre la línea del tiempo, indicando el horario de inicio y el horario de término para cada una de ellas:

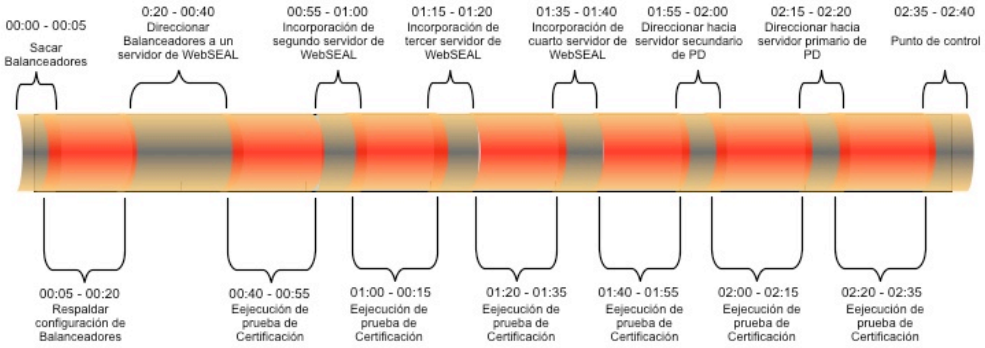

Figura 2.7 Línea del tiempo para la ejecución del cambio programado en *BNET*

A continuación se muestra la tabla 2.21, misma también contiene la secuencia de actividades que describen el proceso para habilitar el cambio y que se presentó al equipo de trabajo del *CAB* para documentar formalmente la solicitud y la aprobación del cambio:

ÁREA: Seguridad Informática			CONTACTO: José A. Ramírez 75681			
SISTEMA: BNET			RESPONSABLE: José A. Alvarez - 74086			
ACCIONES A SEGUIR POR USUARIOS						
ACCIONES TÉCNICAS A REALIZAR PARA EFECTIVIDAD DEL PROCESO DE IMPLANTACION						
Nº	**ÁREA**	**TAREA**	**RESPONSABLE**	**EXT**	**DIA PREV. P/ EJECUCION**	**HORA**
1	Centro de Mando	MENSAJE DE INICIO CONTROL DE CAMBIO	Centro de Mando	75588	22-May	12:00 AM
2	Seguridad Informática	Sacar de Balanceo Balanceador 1 y Balanceador 2	Soporte Primera Línea,	75681	22-May	12:05 AM
3	Seguridad Informática	Respaldar configuración de Balanceadores	José A. Ramírez	75681	22-May	12:20 AM
4	Seguridad Informática	Redireccionar Balanceadores a primer Servidor de WebSEAL	José A. Ramírez	75681	22-May	12:40 AM
5	Seguridad Informática	Ejecución de prueba de certificación interna	José A. Ramírez	75681	22-May	12:55 AM
6	Seguridad Informática	Incorporar al Balanceo un segundo servidor de Webseal	Soporte Primera Línea,	75681	22-May	1:00 AM
7	Seguridad Informática	Ejecución de prueba de certificación interna	José A. Ramírez	75681	22-May	1:15 AM
8	Seguridad Informática	Incorporar al Balanceo un tercer servidor de Webseal	José A. Ramírez	75681	22-May	1:20 AM
9	Seguridad Informática	Ejecución de prueba de certificación interna	José A. Ramírez	75681	22-May	1:35 AM
10	Seguridad Informática	Incorporar al Balanceo un cuarto servidor de Webseal	Soporte Primera Línea,	75681	22-May	1:40 AM
11	Seguridad Informática	Ejecución de prueba de certificación interna	José A. Ramírez	75681	22-May	1:55 AM
12	Seguridad Informática	Redireccionar hacia servidor de PD secundario	José A. Ramírez	75681	22-May	2:00 AM
13	Seguridad Informática	Ejecución de prueba de certificación interna	José A. Ramírez	75681	22-May	2:15 AM
14	Seguridad Informática	Redireccionar hacia servidor de PD primario	José A. Ramírez	75681	22-May	2:20 AM
15	Seguridad Informática	Ejecución de prueba de certificación interna	José A. Ramírez	75681	22-May	2:35 AM
16	Gestión de cambios	**Punto de Control** (Continuar / Roll-Back)	Personal de Guardia	75303	22-May	2:40 AM
17	Centro de Mando	MENSAJE DE FIN CONTROL DE CAMBIO	Centro de Mando	75588	22-May	2:45 AM

Tabla 2.21 Secuencia de actividades del cambio

Parte del material presentado al *CAB* incluía también un plan de remediación que contenía la secuencia de actividades correspondientes al proceso de retorno en caso de que el cambio no tuviera éxito. Afortunadamente el proceso del cambio consistía en sacar de producción una plataforma de hardware y software obsoleta y reemplazarla por una nueva, lo cuál permitía conservar intactos los servidores anteriores, dándole al cambio la capacidad de ser reversible y en cualquier momento poder regresar a su estado original antes de haber iniciado el cambio.

Bajo la premisa de que no existe cambio alguno que logre mitigar el riesgo al 100%, fue definido un flujo de actividades dentro de la estrategia de remediación que contemplaba la posibilidad de que en algún punto se presentara una falla. A continuación se presenta la figura 2.8 que muestra un diagrama de flujo con la

secuencia de acciones a seguir en caso de una falla.

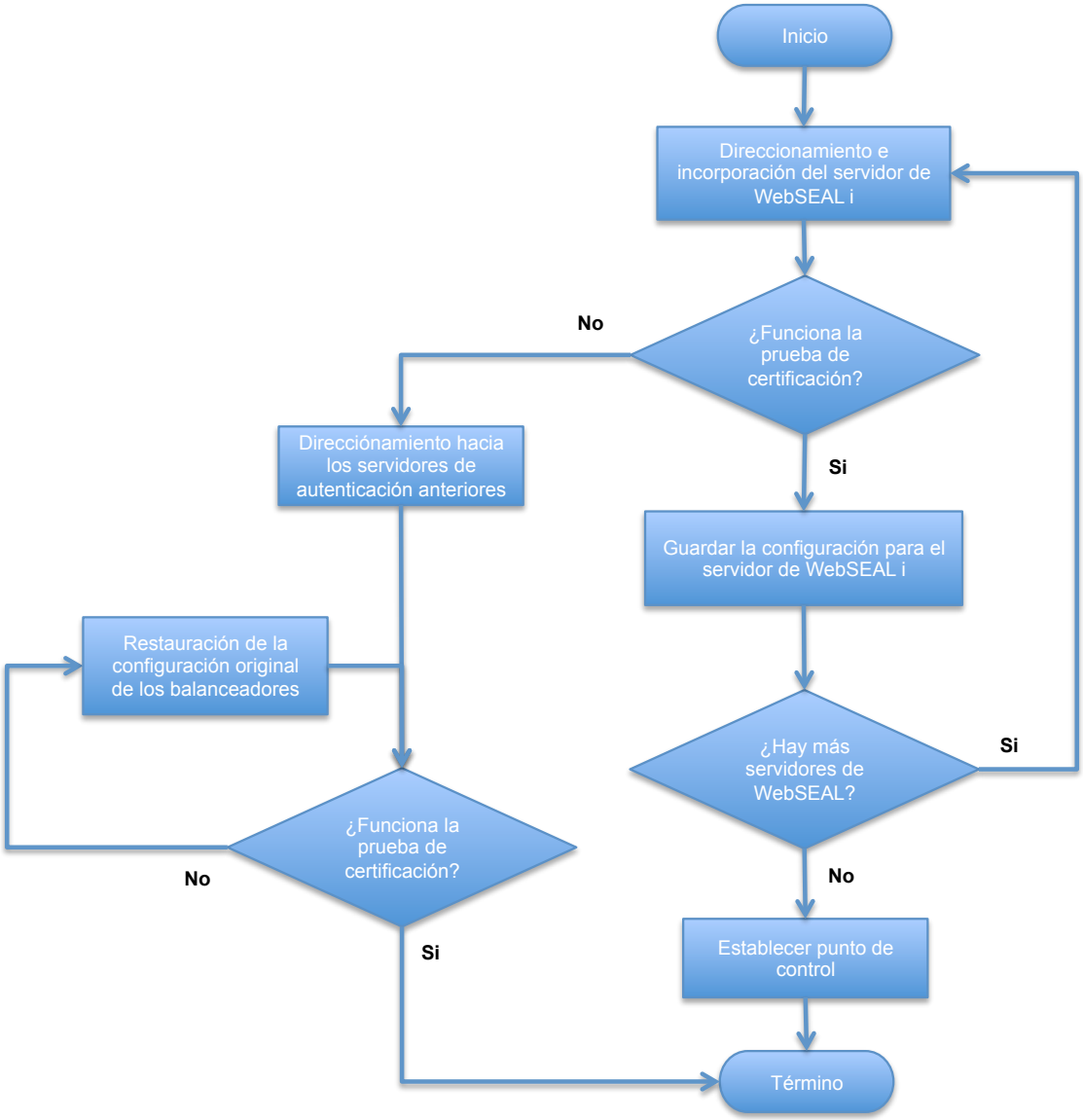

Figura 2.8 Diagrama de flujo de acciones en el cambio

2.10.- Ejecución del cambio

Finalmente llegó el día en que se debía ejecutar el cambio. Todo estaba listo para ejecutar de manera controlada el procedimiento en el que prácticamente se debían apuntar los balanceadores hacía los servidores de *WebSEAL* y gradualmente ir incorporando uno a uno todos ellos. El horario acordado para el inicio de las actividades fue exactamente a las 00:00 horas del día domingo 22 de Mayo de 2011, para lo cuál, como primer paso se realizaría una llamada al administrador de cambios para que le notificara el cambio al personal de la mesa de servicio conocida como *primera línea.* Una vez que se dio la indicación de que se podía iniciar con las actividades, entonces el personal por parte del cliente, encargado de administrar la red, procedió con la ejecución del cambio de direcciones *IP* dentro de los equipos balanceadores de carga web, dando de alta las direcciones que tenían configuradas los servidores de *WebSEAL.*

Iniciado el cambio, prácticamente se siguió al pie de la letra cada una de las actividades dentro de los tiempos estimados, concluyendo con éxito el cambio dentro de la ventana de tiempo otorgada por el cliente. Al término de las actividades fue establecido un punto de control en donde el personal de primera línea certificó y otorgó el visto bueno de que el cambio fue concluido con éxito.

Durante las siguientes horas posteriores al cambio fue necesario que el autor de esta memoria permaneciera en sitio con el propósito de atender cualquier incidente o mal funcionamiento que se pudiera presentar. De antemano se sabía que la carga de operaciones y la concurrencia de usuarios intensa se presentaría a partir de las primeras horas del día lunes siguiente, así que se debía estar completamente alertas, monitoreando el rendimiento de la plataforma. Para ello, todos los componentes de la solución fueron dados de alta dentro de los sistemas de administración de la configuración, mejor conocidos como *CMS*, mismos que se encontraban conectados hacia los sistemas de monitoreo y mesa de ayuda.

Afortunadamente el comportamiento de la plataforma en términos de rendimiento siempre se mantuvo estable. En la figura 2.9 se muestra una salida de los *Shell scripts* desarrollados con el propósito de monitorear constantemente las conexiones concurrentes de los usuarios del servicio de la banca en línea, *BNET* y se puede apreciar un total de 433 hilos de ejecución activos, lo que significaba que existían 866 usuarios conectados, ya que un hilo de ejecución equivalía a 2 sesiones concurrentes.

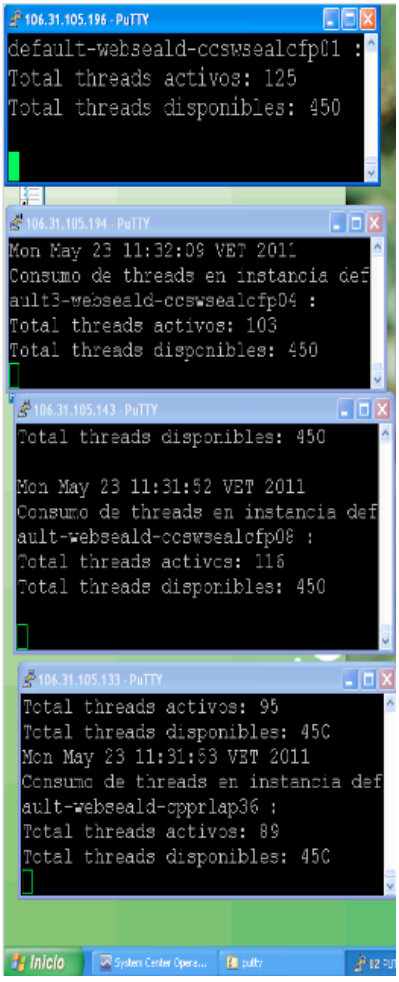

Figura 2.9 Salida de los *shell scripts* de monitoreo

2.11.- Soporte a la solución

La siguiente etapa fue definir una serie de actividades como parte del soporte de vida temprana a la solución, mejor conocido como *ELS,* el cuál brindaría la oportunidad de transferir la operación del nuevo servicio a la gente que lo estaría operando de una manera controlada. La idea era implementar mejoras y resolver problemas, todo ello con la intención de mantener estable el servicio. A continuación se presenta una serie de actividades que fueron definidas como apoyo para el personal encargado del servicio de la banca en línea y particularmente para el dominio seguro:

- Brindar apoyo en la operación cotidiana del servicio
- Recolectar datos referentes al rendimiento del servicio.
- Reportar el rendimiento de los niveles de servicio logrados
- Verificar constantemente la estabilidad del servicio
- Diagnosticar problemas en caso de ser necesario
- Identificar mejoras y procedimientos de mitigación de riesgo en caso de ser necesario
- Documentar los posibles cambios
- Transferir el conocimiento hacia la gente operativa de la solución

Evidentemente, en caso de que se presentara un problema que no se pudiera resolver y que impidiera la entrega del servicio de manera parcial o total, entonces se retomaría nuevamente el plan de retorno para regresar a la configuración inicial que se tenía hasta antes de la ejecución del cambio, y el autor de esta memoria tenía la responsabilidad de solucionar el problema sin importar si esto implicaba permanecer en sitio más tiempo de lo que se había acordado con el cliente. De hecho, dentro del contrato se mencionaba que si por algún motivo el proyecto generaba alguna desviación en tiempo, en la cuál la responsabilidad fuera imputable a la consultoría, entonces todos los gastos derivados de ello tendrían que ser cubiertos por la misma.

El periodo de tiempo oficial durante el cuál se debía permanecer en sitio para brindar soporte abarcaba únicamente las 2 semanas posteriores al cambio. Sin embargo, cabe señalar que posteriormente se tenía el compromiso de monitorear la plataforma y brindar soporte técnico vía telefónica desde México, así que fue necesario conservar vigente el usuario de red *VPN,* así como también fue necesario comprar un número telefónico con código de área de la ciudad de Caracas, por medio de una cuenta de *Skype*[22]*,* ya que resultaba una opción económica para la consultoría y al mismo tiempo brindaba al cliente la comodidad de llamar a un número local para contactar a un especialista de soporte. Ambas vías de comunicación permitirían contar con la capacidad suficiente para atender eficientemente cualquier posible incidente.

Afortunadamente todos los incidentes que se presentaron durante la estancia del autor de esta memoria, fueron eventos aislados relacionados principalmente con bloqueos de contraseñas por parte de los usuarios. Esto permitió que durante ese periodo se pudiera dedicar tiempo suficiente para generar la documentación requerida, que consistía principalmente de las memorias técnicas de la solución. Dichas memorias técnicas describían claramente todo el proceso de implementación con el objeto de que la gente que estaría operando la solución contara con el conocimiento y las herramientas necesarias para hacer cambios en la configuración, según fuese requerido, además de poder atender de manera más eficiente los incidentes que

[22] *Las especificaciones técnicas del software Skype se encuentran disponibles para consulta pública en el sitio oficial del producto, a través de la siguiente liga http://www.skype.com/en/*

pudieran llegar a presentarse.

En la figura 2.10 se muestra una gráfica con el número de incidentes registrados, relacionados con la capa de autenticación y autorización del servicio de *BNET,* en la que se puede apreciar el comportamiento durante las 10 semanas posteriores a la liberación de la nueva solución de dominio seguro.

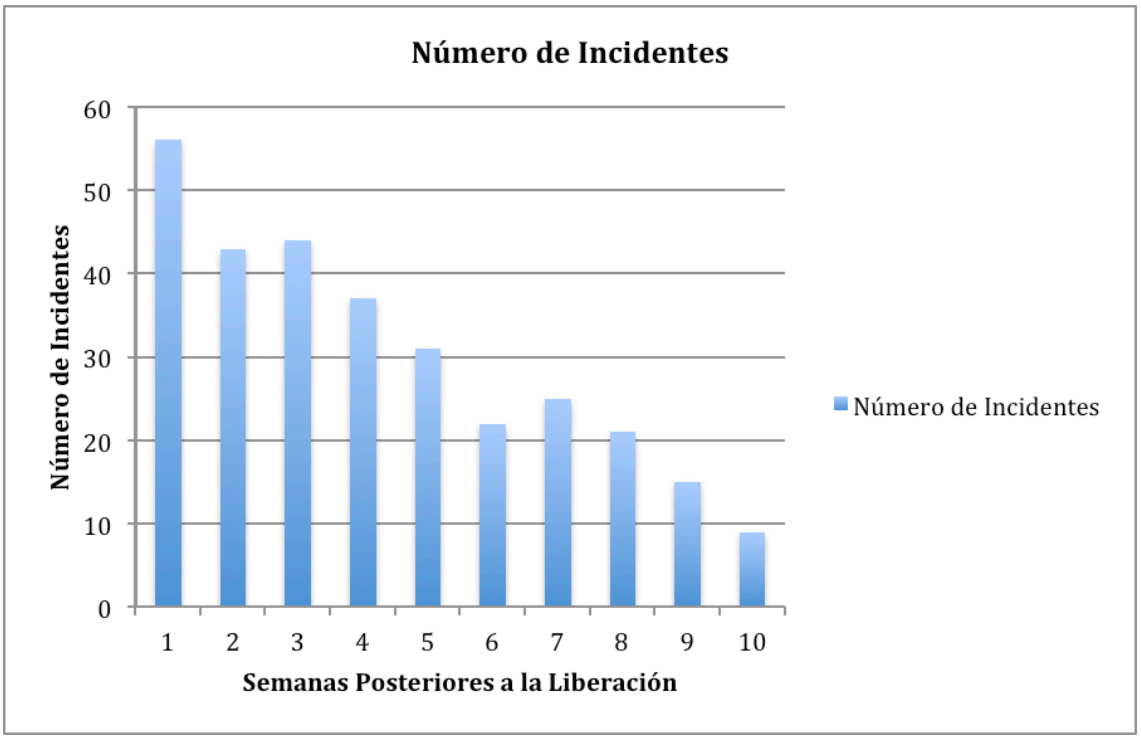

Figura 2.10 Incidentes reportados durante 10 semanas posteriores al cambio

En la figura 2.10 se puede apreciar claramente como la cantidad de incidentes reportados durante 10 semanas, referentes a la solución recién liberada, iba disminuyendo conforme los usuarios se lograban adaptar a la nueva plataforma, mientras esta se estabilizaba.

2.11.1.- Indicadores clave de rendimiento

Los indicadores clave de rendimiento utilizados durante esta fase de soporte de vida temprana fueron los siguientes:

- El factor de variación en el rendimiento del servicio requerido por el cliente.
- El número de incidentes reportados contra el servicio de la *BNET*.
- El incremento en la satisfacción del usuario / cliente gracias a la calidad del servicio entregado.
- La disminución en la cantidad de quejas derivadas de la percepción negativa sobre la calidad del servicio de *BNET.*

Cabe señalar que en realidad el indicador clave de rendimiento más importante para el cliente era precisamente el rendimiento de la solución, en términos de tiempos de respuesta, influenciado por diferentes variables.

2.12.- Certificados de aceptación

Finalmente concluyó el periodo de las 2 semanas de soporte en sitio y aparentemente la solución recientemente liberada mostraba un rendimiento aceptable, como se podrá observar en las gráficas mostradas en el siguiente capitulo, sin haber presentado problemas de funcionalidad, así que casi de manera inmediata se solicitó al cliente que firmara los certificados de aceptación, en los que se mencionaba explícitamente que el proyecto de implementación de la solución de dominio seguro había concluido con éxito, y con ello se lograba alcanzar el objetivo del proyecto de acuerdo a lo estipulado en el plan de trabajo, que formaba parte del contrato. Básicamente eran 2 certificados de aceptación finales, en el primero de ellos se indicaba de manera explicita que la solución había sido probada previamente a la liberación a producción, obteniendo resultados satisfactorios sobre dichas pruebas, y en el segundo se mencionaba que el comportamiento de la solución durante los 15 días posteriores a la ejecución del cambio, era positivo y de acuerdo a lo esperado por parte del cliente. En ambos documentos se mencionaba expresamente que la solución era aceptada por el cliente a su entera satisfacción. Adicionalmente, una vez que el cliente aceptó el contenido y el formato de las memorias técnicas, también fue necesario elaborar un certificado de aceptación en el cuál se daba por aceptado y concluido dicho compendio de memorias técnicas.

3.- ANALISIS DE RESULTADOS Y COMENTARIOS

3.1.- Análisis de resultados

El propósito principal de este capitulo es cuantificar los beneficios generados para el cliente, de manera que se pueda observar claramente cuales fueron los resultados obtenidos, expresados principalmente en términos de rendimiento, y que estos puedan ser medidos. Para ello fue necesario tomar lectura del comportamiento de la plataforma del dominio seguro durante las 4 semanas posteriores a la liberación, iniciando el día lunes 23 de Mayo y finalizando el domingo 19 de Junio, y a partir de los datos recolectados fueron generados los indicadores clave de desempeño, referidos anteriormente como *KPI,* para ser analizados y poder compararlos con los que fueron generados en su momento sobre la plataforma de autenticación y autorización anterior.

Para la recolección de los datos relativos al rendimiento, fue necesario recurrir a las herramientas de monitoreo que ya tenía implementadas el cliente, y en este caso, hablamos particularmente del *appliance Compuware Gomez Real-User Monitoring* referido anteriormente, del cuál se extrajeron los datos estadísticos relacionados con la concurrencia, para posteriormente presentarlos en un formato de gráfica, tal y como se muestra en la figura 3.1

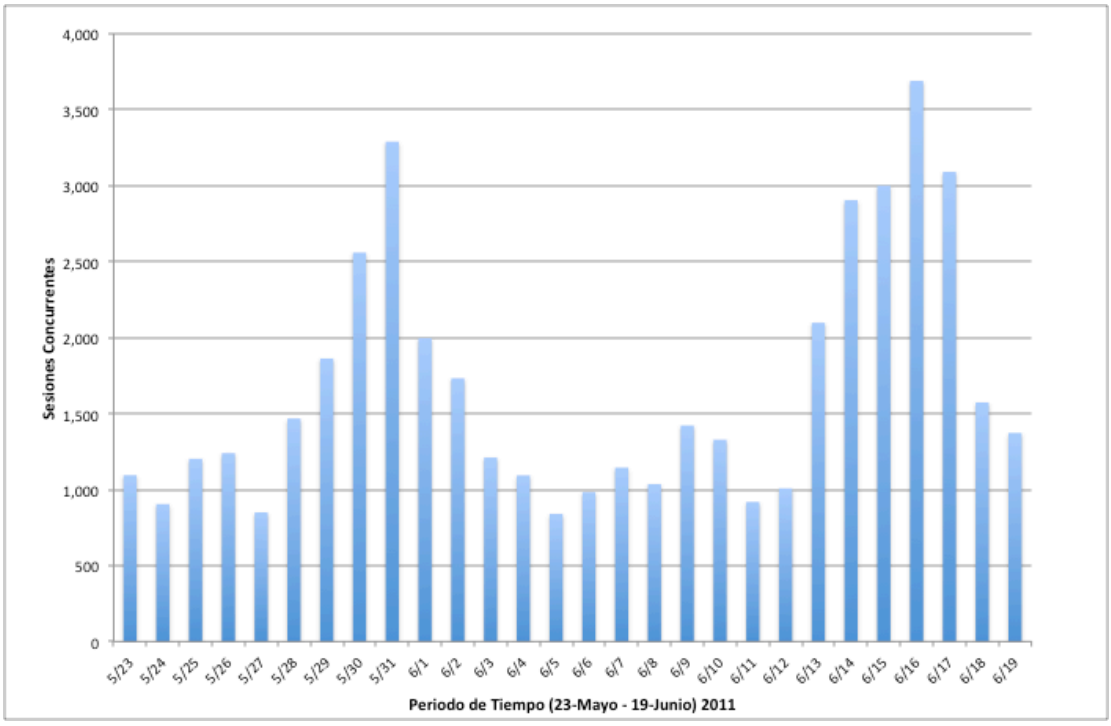

Figura 3.1 Concurrencia de usuarios sobre la *BNET*

A partir de la gráfica anterior se puede determinar claramente que la concurrencia de

usuarios del servicio de *BNET* era muy variable, dependiendo de los periodos de nóminas. En realidad, el promedio de la cantidad de sesiones concurrentes era ligeramente menor a la que se tenía de manera regular 8 meses atrás, cuando se iniciaron las actividades correspondientes al análisis de la plataforma anterior, en Octubre de 2010. Se sostiene la teoría de que esto se debió principalmente a que en el mes de Enero de 2011, el entonces presidente de Venezuela, Hugo Chávez amenazo públicamente a los inversionistas del grupo financiero con expropiar la filial por un aparente incumplimiento de los decretos presidenciales, provocando así que muchos cuentahabientes cancelaran sus cuentas bancarias y transfirieran sus capitales hacia otras instituciones bancarias. El hecho es que en la primer lectura que se había realizado en Octubre de 2010, el promedio de usuarios concurrentes se ubicaba alrededor de 1,800, mientras que de acuerdo con el conjunto de datos que se utilizó para generar la gráfica de la figura 3.1, el promedio de los mismos se ubicaba en 1,675, generando una diferencia de 125 usuarios. Sin embargo, para efecto de poder visualizar con claridad el factor de ganancia en los tiempos de respuesta, esta diferencia en la cantidad de sesiones concurrentes fue despreciada.

3.1.1.- Medición de indicadores clave de desempeño

Para poder representar numéricamente el beneficio generado para el cliente, fue necesario definir un indicador clave de rendimiento en función de los tiempos de respuesta en el servicio de la solución de dominio seguro. De esta manera resultaba muy sencilla la medición de la eficiencia del proceso de administración del servicio. Por otro lado también era importante definir un indicador clave de rendimiento más orientado a la parte de calidad, así que para ello fue necesario tomar como parámetro de medición la cantidad de incidentes relacionados con problemas de autenticación y autorización, reportados a través del servicio de la mesa de ayuda, y con esto tener la capacidad para saber que tan bien se estaba entregando el servicio, más allá de solo conocer la velocidad con la que operaba la solución.

Ambos indicadores clave de rendimiento serían utilizados para medir constantemente y de manera permanente el rendimiento de la solución, y con ello tener la capacidad para determinar cuando era necesario ejecutar algún cambio correctivo o de mejora, como parte de la fase de mejora continua del servicio o *CSI*.

Por medio de la misma herramienta de monitoreo con la que se extrajo el promedio de la concurrencia de usuarios, referida anteriormente, fue generada una traza, a través de la cuál se podían observar los flujos completos de transacciones dentro del servicio de *BNET*. A partir de la traza fueron desagregados cada uno de los procesos ejecutados, de tal manera que fue posible extraer los tiempos de respuesta para cada una de las capas dentro del servicio de *BNET.* Posteriormente fue generado un reporte que contenía una serie de datos, representando específicamente los tiempos de respuesta observados durante el mismo periodo de tiempo en el que fue medida la concurrencia. En la siguiente tabla se muestra un conjunto de valores obtenidos para las operaciones de lectura y escritura con los cuales posteriormente se construyó la gráfica que se presenta en la figura 3.2.

Fecha	Lectura (Seg)	Escritura (Seg)
23-Mayo-2011	0.04	0.06
24-Mayo-2011	0.05	0.08
25-Mayo-2011	0.07	0.09
26-Mayo-2011	0.05	0.08
27-Mayo-2011	0.05	0.07
28-Mayo-2011	0.03	0.05
29-Mayo-2011	0.06	0.11
30-Mayo-2011	0.07	0.17
31-Mayo-2011	0.13	0.21
01-Junio-2011	0.04	0.06
02-Junio-2011	0.02	0.06
03-Junio-2011	0.07	0.09
04-Junio-2011	0.01	0.02
05-Junio-2011	0.01	0.03
06-Junio-2011	0.04	0.05
07-Junio-2011	0.04	0.07
08-Junio-2011	0.06	0.1
09-Junio-2011	0.04	0.07
10-Junio-2011	0.05	0.06
11-Junio-2011	0.02	0.05
12-Junio-2011	0.01	0.04
13-Junio-2011	0.05	0.13
14-Junio-2011	0.08	0.17
15-Junio-2011	0.14	0.21
16-Junio-2011	0.03	0.25
17-Junio-2011	0.02	0.19
18-Junio-2011	0.05	0.09
19-Junio-2011	0.03	0.07

Tabla 3.1 Tiempos de respuesta posteriores a la implementación

Se puede apreciar claramente que durante este periodo de tiempo que comprende un total de 28 días naturales, en ningún momento se presentaron tiempos de respuesta superiores a 0.25 segundos, lo cuál afortunadamente generó de manera inmediata una gran satisfacción y confianza por parte del cliente. En realidad, era una reducción sustancial en los tiempos de respuesta y el cliente aún deseaba observar el comportamiento de la solución durante las siguientes semanas para poder estar plenamente convencido de los beneficios que otorgaba la nueva plataforma y con ello poder comunicar el éxito del proyecto al interior de la organización. En seguida se presenta la figura 3.2 que muestra una gráfica poligonal, representando los tiempos de respuesta observados para las operaciones de lectura y escritura durante el periodo de tiempo referido anteriormente en la tabla 3.1.

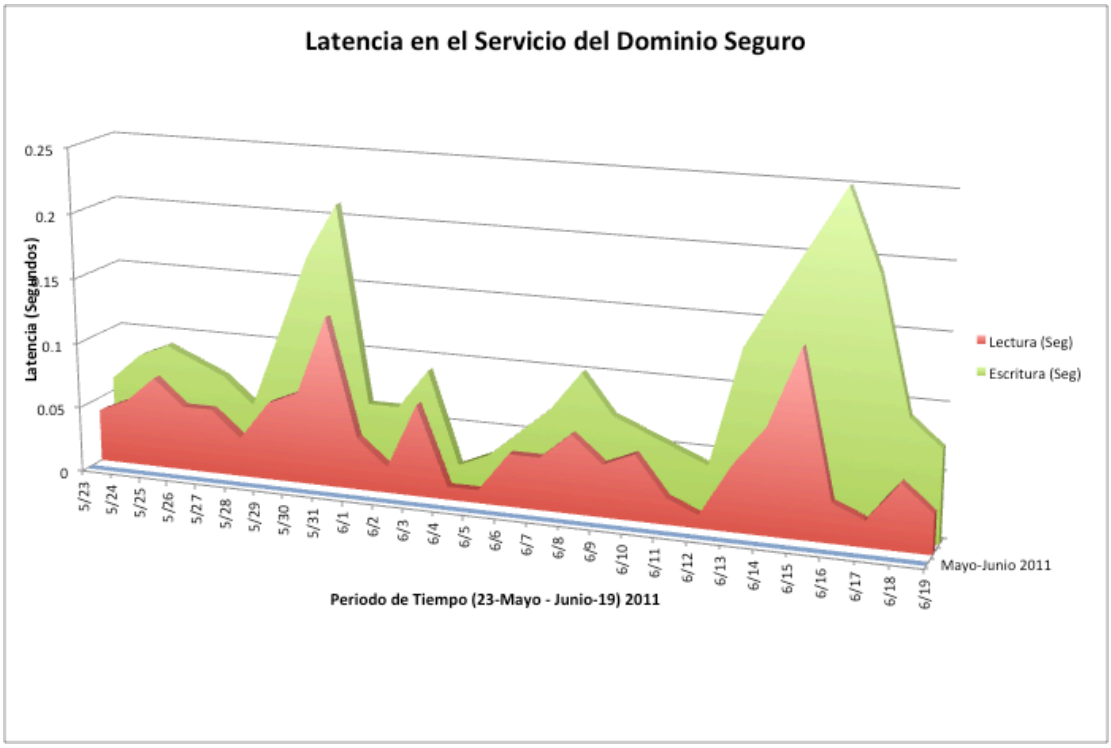

Figura 3.2 Gráfica de tiempos de respuesta posteriores a la implementación

A partir de la tabla 3.1, básicamente se obtienen 2 valores independientes, uno referente al tiempo de respuesta promedio de lectura, y el otro referente al tiempo de respuesta promedio de escritura. En el caso del primero, resulta una cifra igual a 0.048571 segundos, y en el caso del segundo resulta una cifra igual a 0.0975 segundos. Así que inicialmente se realizan 2 comparativos de manera independiente, es decir, un comparativo para la lectura y otro más para la escritura de datos. La forma de calcular el factor de ganancia se hace por medio de la siguiente expresión:

$$G = TR1/TR2$$

Donde:
G representa el factor de ganancia en el tiempo de respuesta promedio.
TR1 es el promedio de los tiempos de respuesta de la plataforma anterior para las operaciones de lectura expresado en segundos.
TR2 es el promedio de los tiempos de respuesta de la plataforma nueva para las operaciones de lectura expresado en segundos.

A partir de la tabla 1.2, se sabe que el valor de TR1 = 1.947742 s

Así que, en el caso de las operaciones de lectura, sustituyendo los valores, resulta la siguiente ecuación:

$$G = 1.947742s/0.048571s$$

Así que finalmente: $G = 40.1$

Evidentemente el factor de ganancia obtenido para las operaciones de lectura resulta impresionantemente alto, significando que la solución recién implantada era prácticamente 40 veces más rápida para leer con respecto a la anterior.

Por otro lado, es necesario realizar el mismo procedimiento para las operaciones de escritura, así que a continuación se presenta nuevamente la expresión para el cálculo del factor de ganancia:

G = TR1/TR2

Donde:
G representa el factor de ganancia en el tiempo de respuesta promedio.
TR1 es el promedio de los tiempos de respuesta de la plataforma anterior para las operaciones de escritura expresado en segundos.
TR2 es el promedio de los tiempos de respuesta de la plataforma nueva para las operaciones de escritura expresado en segundos.

A partir de la tabla 1.2, se sabe que el valor de TR1 = 3.1787097 s

Así que, en el caso de las operaciones de escritura, sustituyendo los valores, resulta la siguiente ecuación:

G = 3.1787097s/0.0975s
Así que finalmente: G = 32.60

Evidentemente el factor de ganancia para las operaciones de escritura resulta un poco menor que para las operaciones de lectura, pero de cualquier modo sigue siendo considerablemente alto, ya que esto significa que la solución de dominio seguro recién implantada es 32 veces más rápida para hacer escrituras con respecto a la anterior.

El factor de ganancia total prácticamente es la suma del factor de ganancia sobre las operaciones de lectura más el factor de ganancia sobre las operaciones de escritura, dividido entre 2, así que a continuación se muestra la expresión utilizada para la obtención de dicho valor:

FGT = (FGL + FGE)/2

Donde:
FGT es el factor de ganancia total
FGL es el factor de ganancia para las operaciones de lectura
FGE es el factor de ganancia para las operaciones de escritura

Sustituyendo los valores conocidos, se obtiene la siguiente ecuación:

FGT = (40.1 + 32.60)/2
Así que finalmente: FGT = 36.35

Afortunadamente el factor de ganancia total es un dato duro sumamente importante que muestra el beneficio real cuantificado de la solución en términos de tiempo de respuesta para los usuarios del servicio de *BNET,* con el cuál se pudo demostrar a directivos de diferentes áreas de negocio, que el haber tomado la decisión de ejecutar el proyecto del dominio seguro fue un éxito indiscutible y contundente. Cabe señalar que el comparativo puede llegar a presentar un margen de error debido a que se hizo la comparación entre datos capturados en el mes de Octubre contra datos capturados en los meses de Mayo y Junio, sin haber tomado en cuenta que los bancos regularmente están sujetos a estacionalidad.

Respecto al indicador clave de desempeño de calidad, también fue necesario definir un parámetro de medición, que obviamente debía ser numérico, así que de primera instancia se consideró la cantidad de incidentes reportados, relacionados con la capa de autenticación y autorización, como ya fue referido anteriormente.

Durante las primeras semanas posteriores a la liberación de la solución, se llevó un registro minucioso de los incidentes reportados con el propósito de detectar y corregir

posibles anomalías en la funcionalidad y a pesar de que en el transcurso de la semana 1 se presentaron un total de 55 incidentes, tal y como se puede observar en la figura 2.9, la frecuencia de los mismos fue disminuyendo considerablemente a lo largo de las siguientes semanas, hasta llegar a la semana 10, en la que únicamente fueron reportados 9 incidentes. Cabe señalar que en realidad para un universo total de alrededor de 3,500,000 usuarios del servicio de la *BNET*, un rango de 9 - 55 incidentes por semana resultaba bastante aceptable, además de que del total de incidentes reportados, aproximadamente el 80% de los mismos estaban relacionados con errores humanos, en los que derivado de la solicitud de un cambio de contraseña, el propio usuario provocaba el bloqueo de su cuenta. Es importante aclarar que desafortunadamente no fue posible que el cliente indicara con precisión la cantidad de incidentes reportados sobre la plataforma anterior durante las últimas semanas previas a la ejecución del cambio, lo cuál limitaba la capacidad para poder realizar un comparativo real y con ello obtener un posible factor de ganancia estimado. Sin embargo a partir de la cantidad máxima de incidentes reportados, podemos inferir lo siguiente:

Se sabe que IR = 55, donde IR se refiere a los incidentes reportados durante la semana 1

Por otro lado, también sabemos que:
1 semana = 7 días
1 día = 24 horas
1 hora = 3,600 segundos

Esto significa que:
1 semana = (7) (24) (3,600 segundos)
1 semana = 604,800 segundos
Luego entonces:
55 incidentes / 604,800 segundos = 0.00009094 incidentes / segundo
O visto de otra forma:
7.85 incidentes / día

Bajo la premisa de medir el tiempo en segundos, se puede percibir que se tenía una frecuencia mínima de incidentes relacionados con la capa de autenticación y autorización, que prácticamente presentaba un comportamiento con tendencia a cero. Básicamente la idea era obtener un parámetro con el cuál intentar medir la satisfacción del cliente durante las primeras semanas posteriores a la liberación, teniendo como antecedente que para la plataforma anterior, tampoco se contaba con información formal referente a dicho nivel de satisfacción de usuario.

Por otro lado, un factor determinante al momento de estudiar el comportamiento de la plataforma recién liberada a producción, era justamente analizar el nivel de utilización de recursos que presentaba dicha plataforma del dominio seguro. La manera más ágil, rápida, y confiable para poder determinar el porcentaje de utilización de recursos fue a través de las herramientas y utilerías propias de cada sistema operativo, ya que en realidad lo que se deseaba conocer era básicamente el nivel de utilización de *CPU* y memoria *RAM*. De cualquier modo, el agente de monitoreo utilizado por el cliente generaba sus reportes de estadísticas a partir de las salidas que arrojan las utilerías de sistema operativo *UNIX* o *Linux,* según fuera el caso.

A continuación se presenta la figura 3.3 que muestra una gráfica poligonal generada a partir del promedio de los datos arrojados por la utilería *top* en los servidores con *SLES 10* y por las utilerías *mpstat* y *sar*, en los servidores con *Solaris 10*, con los que se obtiene el porcentaje de consumo de *CPU* durante el mismo periodo de estudio que

se ha utilizado en la presente sección, y que básicamente comprende del 23 de Mayo al 19 de Junio.

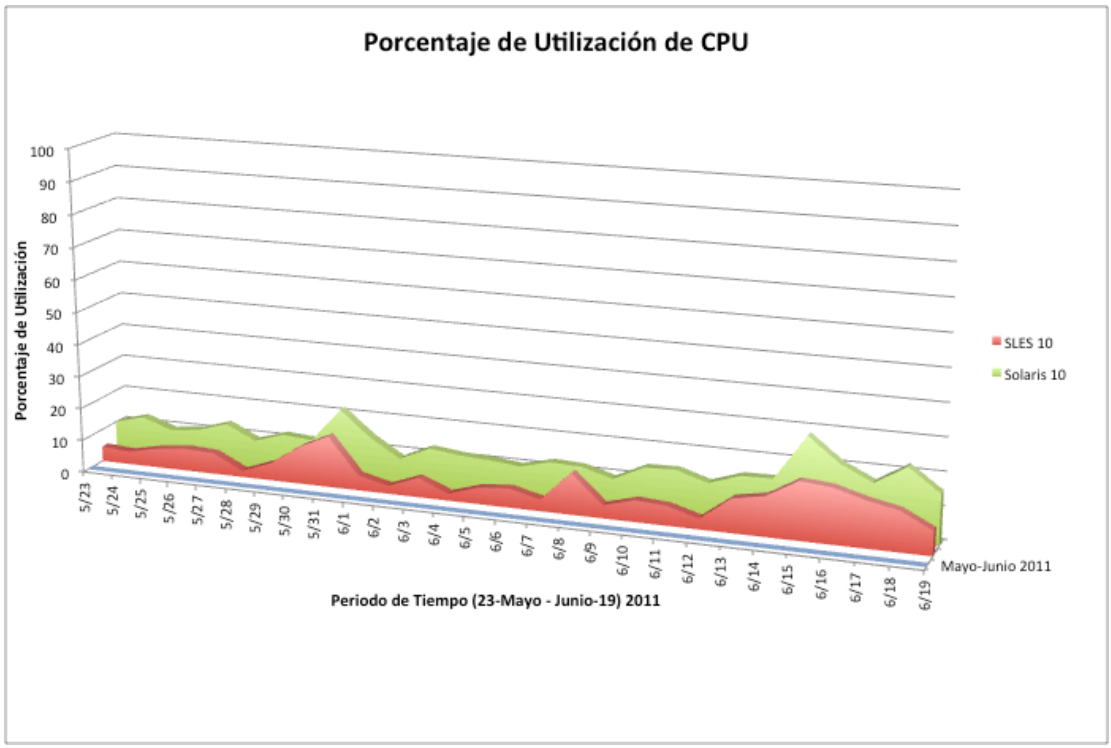

Figura 3.3 Gráfica de porcentaje de utilización de *CPU*

En la figura anterior se puede apreciar claramente que el porcentaje de utilización de recursos de *CPU* en cada una de las 2 principales capas de la solución se mantuvo en un rango que va desde un 2% hasta un 28.2%, así que jamás superó siquiera el 30% de utilización, con lo cuál se demostraba que aún durante los días de mayor actividad en el servicio de *BNET,* los niveles de utilización de procesador se mantenían muy por debajo del límite máximo permitido, establecido en 70%. Esto significaba, que aún en los periodos de mayor utilización de recursos de procesamiento, conocidos como "picos", se tenía disponible alrededor de un 40% de capacidad adicional, antes de llegar a dicho límite máximo permitido.

El porcentaje promedio de utilización de *CPU* en la capa de directorio era de 13.53%, mientras que en la capa del componente de acceso *web* era ligeramente menor, teniendo 7.71%. Así que el porcentaje promedio total de utilización de *CPU* se obtiene a partir de la suma de ambos porcentajes dividido entre 2, dando como resultado 10.62%.

Por otro lado, con respecto a la memoria *RAM,* que era el segundo recurso más importante de monitorear aparte del procesador, se lograron extraer los datos necesarios para la construcción de la gráfica 3.4

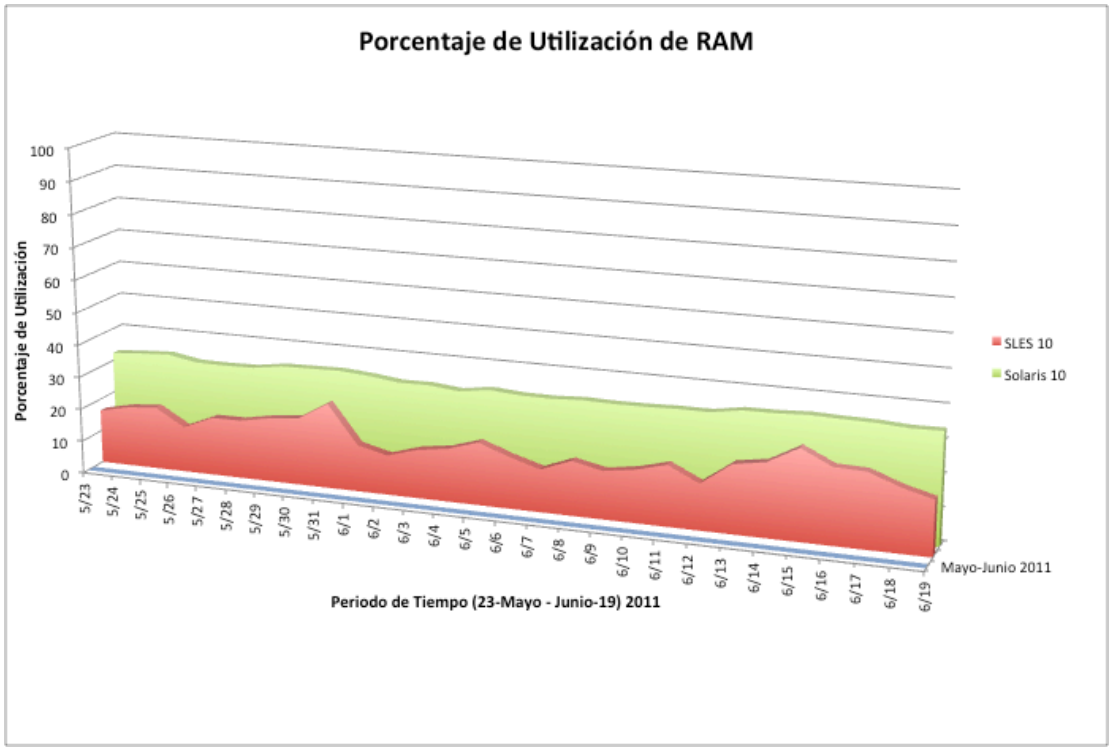

Figura 3.4 Gráfica de porcentaje de utilización de memoria *RAM*

De acuerdo a la gráfica anterior, el comportamiento en los niveles de utilización de memoria *RAM* resultaba muy distinto con respecto a los de *CPU,* ya que en el caso de los servidores de directorio, el consumo era casi lineal debido a que al momento de inicializar los servicios de directorio, todos los datos de los usuarios son alojados directamente en un área de memoria *RAM* designada como *cache,* provocando así que una vez inicializados dichos servicios, se presentara una variación mínima, dentro de un rango que iba desde 31.2% hasta un 34.5%. En el caso de los servidores del componente de acceso *web,* el comportamiento es diferente debido a que los objetos de las sesiones de usuario se crean y se destruyen de manera dinámica generando una variación dentro de un rango más amplio en los niveles de utilización de la memoria *RAM,* teniendo como valor mínimo un 12.1% y como valor máximo un 27.5%. El porcentaje promedio de utilización en los servidores de directorio era de 32.83%, mientras que para los servidores de acceso web era de 18.12%. Así que el porcentaje promedio total de utilización de memoria se obtiene a partir de la suma de ambos valores dividida entre 2, dando como resultado 25.475%.

Evidentemente, los datos reportados durante estas primeras semanas de operación, generaron en el cliente un nivel alto de satisfacción y fueron un factor fundamental y decisivo al momento de firmar los certificados de aceptación, en los que de manera explícita se aceptaba la terminación del proyecto de acuerdo a lo estipulado en el contrato.

Al momento de redactar la presente memoria, primer semestre de 2013, se sabía que los servicios de la *BNET* eran monitorizados los 365 días del año y prácticamente a poco más de 2 años de distancia de haber sido implementada la solución del dominio seguro, y sin haberse reportado incidentes y/o problemas graves que implicaran un involucramiento directo por parte de la consultoría para atenderlos como parte de la responsabilidad adquirida a través del contrato de soporte a la solución de dominio seguro, se puede afirmar de manera indiscutible que el haber decidido implantar dicha solución fue una excelente decisión y que el proyecto había resultado un éxito, ya que

sobre todo, los tiempos de respuesta asociados a la capa de autenticación y autorización de usuarios del servicio de la *BNET* se lograron mejorar en un factor redondeado, absoluto de 36 veces, obtenido a partir de los indicadores clave de desempeño referidos anteriormente. Cabe resaltar que los incidentes reportados a lo largo de los 2 primeros años de vida de la solución, estaban relacionados principalmente con el bloqueo de usuarios, los cuales eran considerados como eventos aislados causados por errores humanos al momento de hacer cambios de contraseñas, y no precisamente asociados a un mal funcionamiento de la solución.

Aunado a lo anterior, se sabe que el número de incidencias reportadas a través del servicio de mesa de ayuda, había disminuido con respecto a lo señalado durante las primeras semanas posteriores a la liberación.

A continuación se presenta la tabla 3.2 que muestra un resumen de los indicadores clave de rendimiento utilizados de acuerdo a lo establecido durante la fase de diseño.

Indicador Clave de Rendimiento	Valor Promedio Anterior	Valor Promedio Actual	Aceptado (Si/No)
Tiempo de Respuesta de Lectura en segundos	1.947742	0.048571	Si
Tiempo de Respuesta de Escritura en segundos	3.1787097	0.09750	Si
Porcentaje de Utilización de CPU	78.00	10.620	Si
Porcentaje de Utilización de RAM	85.00	25.475	Si

Tabla 3.2 Resumen de indicadores de desempeño

3.1.2.- Errores y aciertos

Es importante señalar que la manera en como fue ejecutado el proyecto, logró el cumplimiento de los objetivos planteados, y se debe hacer especial énfasis en las actividades y tareas que fueron bien ejecutadas, y que son calificadas como aciertos por los buenos resultados que produjeron, así como también todas aquellas actividades y tareas que tienen puntos de mejora significativos y que son calificadas como errores.

A continuación se presenta una lista de los principales aciertos durante el ciclo de vida del proyecto:

- Se logró establecer vínculos eficientes de comunicación con todos los integrantes del equipo de trabajo, transmitiendo mensajes claros y concisos en los momentos en que fue requerido. Esto eliminó la necesidad de programar reuniones repetitivas con una duración prolongada de tiempo.
- Se realizó una excelente planeación del cambio, incluyendo el plan de retorno que sería utilizado en caso de presentarse alguna falla, todo ello con el apoyo de los miembros del equipo de trabajo.
- Se logró transmitir hacia el personal del cliente el conocimiento necesario para poder atender y resolver algunos incidentes posteriores a la fase de soporte de vida temprana a la solución.

- Hubo una gran actitud de colaboración y servicio por parte de cada uno de los integrantes del equipo de trabajo formado por personal de la consultoría y por el cliente. Esto redujo el impacto de las diferencias de opinión.
- Todos los requisitos fueron atendidos oportunamente en tiempo y forma, con lo cuál se pudo ejecutar cada una de las actividades sin retrasos ni desviaciones que implicaran un ajuste en el cronograma.

Por otro lado, a continuación se presenta una lista de los principales errores cometidos durante el ciclo de vida del proyecto:

- No involucrarse de una manera más activa en el desarrollo de las pruebas de aceptación controladas y ejecutadas por el cliente.
- No se logró iniciar con el desarrollo de la memoria técnica antes de la conclusión del proyecto. Esto hubiera permitido agilizar la entrega de la misma sobre la última fase del proyecto.
- No fue posible permanecer más tiempo directamente en sitio para poder atender cualquier situación de emergencia.
- El personal del área de infraestructura por parte del cliente contaba con las contraseñas de los usuarios *root,* cuando aún no se firmaban los certificados de aceptación. Esto pudo haber ocasionado algún problema al ceder parcialmente el control de la infraestructura al personal del cliente. La única manera de evitar esto era haber configurado controles de seguridad basados en roles para determinadas tareas de administración de sistema operativo.

En realidad, durante todo el ciclo de vida del proyecto se presentaron eventos y tareas que no siempre arrojaban los resultados esperados. Sin embargo al ser tareas muy granulares que formaban parte de otras tareas más complejas, el impacto de las primeras era mínimo dentro del resultado global, y por tal motivo no son categorizados como errores.

3.1.3.- Experiencia profesional adquirida

Finalmente, como parte del análisis de resultados, es de suma importancia señalar el impacto que se produjo en el perfil profesional del autor de esta memoria, durante y después del desarrollo del ciclo de vida del proyecto del dominio seguro, ya que se generó una serie de eventos y experiencias que enriquecieron en gran medida la carrera profesional, al estimular y fortalecer algunas habilidades, como son la capacidad establecer y mantener en todo momento los vínculos necesarios entre la fase de diseño y la fase de implantación dentro del desarrollo de un proyecto, así como la capacidad para administrar adecuadamente y de manera óptima el tiempo como un recurso finito a lo largo del ciclo de vida del proyecto, ambas habilidades ampliamente requeridas en roles relacionados con la gestión de proyectos, además de que son habilidades críticas en la formación que tiene un Ingeniero en Computación, ya que generalmente son utilizadas directa o indirectamente dentro de las actividades rutinarias. La experiencia profesional que se tenía previamente a la realización del proyecto, objeto de estudio de la presente memoria, era más limitada, ya que a pesar de haber participado en diferentes proyectos ejecutando roles como analista, diseñador, arquitecto de soluciones, ingeniero de implantación, y de soporte, en realidad nunca antes había participado en un solo proyecto que involucrara todos y cada uno de los diferentes roles al mismo tiempo y que abarcara eventualmente diferentes fases de un proyecto, desde que se concibe hasta que se entrega y se soporta.

A continuación se enlistan las principales habilidades adquiridas y/o mejoradas a lo largo del ciclo de vida del proyecto:

- Se mejoró la capacidad para interpretar los requerimientos de negocio planteados por el cliente, con el propósito de poder ofrecer una solución tecnológica acorde a sus necesidades, de la manera más atinada posible y con el menor porcentaje de riesgo posible, acotando siempre el alcance de la misma.
- Se desarrolló la comprensión y el entendimiento de las buenas prácticas y metodologías sugeridas para la realización de proyectos de tecnologías de información. Particularmente las metodologías utilizadas para este proyecto, que fueron *ITIL v3.0 y COBIT v4.1.*
- Se desarrolló el entendimiento y la comprensión sobre la importancia de los diferentes aspectos de la seguridad dentro de los sistemas informáticos bancarios.
- Se desarrolló y se reforzó la capacidad para trabajar bajo un entorno de presión muy fuerte, en donde la mejor herramienta fue el dominio técnico de los procedimientos a ejecutar dentro de los servidores durante la fase de implantación.
- Se mejoró la capacidad de entendimiento sobre las fronteras o límites de responsabilidad que tiene cada participante dentro de un proyecto complejo de tecnologías de información, considerando las dependencias entre todos y cada uno de ellos, dependiendo de los roles y perfiles requeridos.
- Se mejoró el entendimiento sobre la importancia y autoridad que representa un líder técnico dentro de la ejecución de un proyecto de tecnologías de información, así como la manera en como ayudarlo a lograr los objetivos establecidos previamente.
- Se mejoró la capacidad para establecer una relación de confianza sólida con todos los integrantes de un equipo de trabajo completo.
- Se mejoró la capacidad para identificar el momento y el lugar preciso para transmitir los mensajes necesarios, haciéndolo de una manera clara y concisa.
- Se desarrolló la habilidad para manejar situaciones de conflicto humano en donde la principal herramienta fue el análisis de posibles resultados sobre 2 o mas alternativas de solución a un problema en particular.

Adicionalmente, resulta relevante comentar que se incrementó sustancialmente el conocimiento técnico relacionado con los productos utilizados para la solución de dominio seguro, al igual que lo fue con las metodologías utilizadas para el desarrollo del proyecto, referidas anteriormente, ya que evidentemente, más allá de conocer teóricamente cada una de las fases, resultó de alto impacto el hecho de tener la posibilidad de aplicar toda la teoría dentro de un proyecto que presentaba un grado de complejidad importante.

3.2.- Comentarios finales

La identidad digital de los usuarios de servicios financieros que ofrecen las instituciones bancarias resulta de vital importancia en nuestros tiempos, ya que de no existir una plataforma robusta que brinde servicios de autenticación de identidad, se puede originar una brecha de seguridad que permitiría a usuarios maliciosos robar la identidad de cuentahabientes y/o tarjetahabientes, derivando en operaciones fraudulentas y pérdidas de capital. Ante la reciente oleada de ataques cibernéticos que han sufrido los bancos más importantes del mundo durante los últimos años, resulta

altamente recomendable para las áreas de tecnología al interior de las diferentes instituciones bancarias, el implementar mecanismos que permitan protegerse de los accesos no autorizados, así como restringir los privilegios de cada identidad, para lo cuál la presente memoria puede aportar ideas que complementen un diseño preliminar o incluso puede sugerir el uso de algún componente en particular. Es importante señalar que en ningún momento se pretende influenciar al lector para que utilice algún producto de una marca en particular, metodología, o patrón de arquitectura, más bien, simplemente se exponen de forma general los beneficios de la solución completa, vista en su totalidad, en donde cada componente contribuye con su parte correspondiente, lo cuál no significa que se pueda garantizar el comportamiento de dicho componente bajo un escenario diferente. El lector interesado en el proceso de definición y selección de componentes para una solución de dominio seguro, deberá realizar una investigación mas profunda sobre la compatibilidad de productos que se encuentren disponibles al momento de diseñar una solución, ya que es muy probable que los productos y versiones referidas en esta memoria, sean descontinuados o reemplazados.

Por otro lado, con respecto al lector que no necesariamente se encuentra ante la necesidad inmediata de involucrarse en un proyecto de esta naturaleza, pero que se relacione directa o indirectamente con las áreas de tecnología, y particularmente de seguridad informática al interior de las instituciones bancarias, resulta importante contar con un panorama general de lo que representa una plataforma de dominio seguro, resaltando los beneficios y las implicaciones inherentes a la misma, que se tienen en la operación cotidiana.

El objetivo fundamental de esta memoria es mostrar la manera en como fue resuelto un problema de negocio, que en su momento estaba generando pérdidas monetarias importantes al cliente, en donde el principal afectado era el usuario de servicios de la banca. Todos los procesos de negocio relacionados con el servicio de la banca en línea estaban siendo afectados directamente por un mal funcionamiento de las tecnologías de información y específicamente por una plataforma que brindaba servicios de autenticación y autorización de una manera deficiente.

Así pues, la aportación principal de esta memoria consiste en mostrar de forma clara la manera en como se utilizó la información recopilada a partir de los reportes estadísticos de utilización de recursos, para posteriormente iniciar con la fase de diseño, haciendo especial énfasis en el proceso de dimensionamiento de la capacidad requerida, con base en la concurrencia y la cantidad de transacciones, considerando a solicitud del cliente, una proyección de crecimiento del 10% lineal para cada uno de los 3 años siguientes, resultando en un 30% total. Por otro lado se muestra el desarrollo de la fase de implementación, resaltando la planificación y control de actividades, incluyendo el proceso de endurecimiento de sistemas operativos sobre los que descansa la solución del dominio seguro, y finalizando el proyecto con el soporte de vida temprana a la solución recién implantada. Todo ello con el afán de despertar en el lector un interés por estudiar y comprender el proceso que se siguió para dar solución a un problema muy específico de seguridad informática. Como fue mencionado anteriormente, la intención es que el lector pueda utilizar única y exclusivamente la información que requiera, ya que la información cuenta con un alto grado de especificidad, lo que permite resolver casos muy particulares o por el contrario se puede tomar el caso como una arquitectura de referencia en la que a partir de cierta cantidad de usuarios se pueda proponer una plataforma con una capacidad de procesamiento y almacenamiento mayor o menor, según sea el caso, ya que las soluciones de autenticación y autorización pueden presentarse en una gran variedad de posibles combinaciones de productos y tamaños de la infraestructura, por lo que se recomienda que en caso de presentarse una problemática de características similares,

la presente memoria sea usada únicamente como una guía de referencia para resolver dicha situación, sin perder de vista que de cualquier forma para cada caso particular será necesario realizar un análisis minucioso del requerimiento. Una parte muy importante para el lector es que se hayan documentado las experiencias profesionales obtenidas a partir de la aplicación de las metodologías *ITIL v3.0 y COBIT v4.1* al análisis, diseño e implementación de una solución de dominio seguro.

Los anexos de esta memoria sirven para preservar una estructura de orden al contenido, ya que existen descripciones, procedimientos, y definiciones muy especificas que conviene mostrar de manera independiente al cuerpo principal del trabajo con el propósito de hacer más fácil la búsqueda de la información dentro del documento, y evitar así la confusión que podría derivar en la pérdida del significado en las ideas principales del contenido.

Glosario de Términos

ACI.- Por sus siglas en inglés significa *Access Control Item,* y es un atributo del servidor de directorio que determina quién puede tener un tipo de acceso especifico a los datos del directorio.
ACL.- Por sus siglas en inglés significa *Access Control List,* y se refiere a un grupo de directivas de acceso definidas. Estas directivas garantizan los niveles de acceso a datos específicos para ciertos clientes o grupos específicos.
Agentless.- Es un término usado en el ámbito de las tecnologías de información que se refiere la recolección de datos de un grupo de computadoras sin la necesidad de instalación de agentes sobre las mismas.
API.- Por sus siglas en inglés significa *Application Programming Interface* y es una especificación que se compone de rutinas, protocolos, y herramientas, e indica la manera en como interactúan diversos componentes de software.
Appliance.- Es un término que se refiere generalmente a un dispositivo de *hardware* que ha sido diseñado para cubrir un propósito específico y que tiene poca flexibilidad para ser reconfigurado.
BCS.- Por sus siglas en inglés significa *Business Continuity Strategy* y se refiere a la estrategia definida por una organización asegurar la recuperación y la continuidad del negocio ante un eventual desastre o un corte de energía eléctrica de magnitudes superiores. Este incluye planes y metodologías que son determinados por la estrategia de la organización.
Benchmark.- Se refiere a un tipo de prueba que es usada para comparar el rendimiento de una infraestructura de *hardware* y *software.*
BIA.- Por sus siglas en inglés significa *Business Impact Analysis* y se refiere al proceso diseñado para priorizar las funciones del negocio por medio de una valoración del impacto potencial cualitativo y cuantitativo que podría resultar si una organización experimenta un evento en la continuidad del negocio.
BTU.- Por sus siglas en inglés significa *British Thermal Unit* y se refiere a la cantidad de calor requerido para incrementar en un grado *Fahrenheit* la temperatura de una libra de agua.
CA.- Por sus siglas en inglés significa *Certificate Authority* y se refiere a una entidad autorizada para emitir certificados digitales que sirven para establecer relaciones de confianza al comprobar la autenticidad del que posee una llave pública.
CAB.- Por sus siglas en inglés significa *Change Advisory Board* y se refiere a un grupo de personas que evalúa las solicitudes de cambio, basado en la necesidad, la prioridad, costo / beneficio, y el impacto potencial hacia otros sistemas o procesos.
Cache.- Dentro del ámbito de la solución del dominio seguro, el término se refiere a una porción de memoria *RAM* y/o de disco de alto rendimiento donde son alojados los datos para lograr un acceso rápido a los mismos.
CFO.- Por sus siglas en inglés significa *Chief Financial Officer* y es un término para designar al ejecutivo corporativo que cuenta con la autoridad financiera dentro de una organización.
Checksum.- Se refiere a un esquema de detección de errores, en el cuál cada mensaje transmitido es acompañado por un valor numérico basado en el conjunto de bits del mensaje.
CIO.- Por sus siglas en inglés significa *Chief Information Officer* y se refiere a la persona que se encuentra a cargo del departamento de sistemas de la información dentro de una organización.
Cluster.- Se refiere a un grupo de servidores o nodos que funcionan como un solo

servidor para lograr un objetivo común.
CMS.- Por sus siglas en inglés significa *Configuration Management System* y se refiere a una serie de herramientas y bases de datos que son usadas para gestionar los datos de configuración de un proveedor de servicios.
CN.- Es un atributo dentro de la definición del directorio, conocido también como *CommonName* que tiene como opción un subtipo.
COBIT.- Por sus siglas en inglés significa *Control Objectives for Information and Related Technology* y se refiere a un marco de trabajo creado por un organismo llamado ISACA. Básicamente consta de un conjunto de herramientas que permiten eliminar la brecha entre los requerimientos de control, las cuestiones técnicas, y los riesgos de negocio.
Core.- Se refiere a un núcleo de un procesador que puede estar contenido dentro de un procesador de varios núcleos.
CPU.- Por sus siglas en inglés significa *Control Processing Unit* y se refiere a la parte de una computadora o servidor que interpreta y ejecuta instrucciones.
CSI.- Por sus siglas en inglés significa *Continual Service Improvement* y es un concepto que se refiere al proceso de identificar y actuar sobre posibles ajustes para corregir o mejorar un servicio que ya se encuentra en operación.
CISO.- Por sus siglas en inglés significa *Chief Information Security Officer* y es un término utilizado para designar a la persona responsable de la seguridad relacionada directamente con las tecnologías de información.
DIT.- Por sus siglas en inglés significa *Directory Information Tree* y se refiere a la estructura jerárquica en forma de árbol en contiene todos los DN de las entradas en el directorio.
DN.- Por sus siglas en inglés significa *Distinguished Name* y se refiere al nombre único de una entrada en el directorio. Este comprende todos los nombres individuales de las entradas padre hasta llegar a la raíz.
DRP.-Por sus siglas en inglés significa *Disaster Recovery Plan* y se refiere al plan definido para dar continuidad al negocio ante un eventual desastre que destruye parte o incluso todos los recursos necesarios para el negocio, incluyendo la infraestructura de TI.
EAR.- Por sus siglas en inglés significa *Enterprise Archive* y en realidad es un archivo en formato *Java,* que usado para empaquetar módulos como si fueran un solo archivo con el propósito de asegurar la coherencia en el despliegue de los diferentes módulos dentro de un servidor de aplicaciones.
ELS.- Por sus siglas en inglés significa *Early Life Support* y se refiere al soporte que se da un servicio recién liberado a producción.
Failover.- Este término se refiere a un proceso de conmutación en el que automáticamente se mueven las cargas de trabajo de un servidor hacia otro. Normalmente dicho proceso se ejecuta entre los nodos que pertenecen a un *cluster.*
Filesystem.- Se refiere a un sistema de archivos que tiene una estructura jerárquica en forma de árbol y que especifica la convención para la nomenclatura de archivos y directorios.
FIPS.- Por sus siglas en inglés significa *Federal Information Processing Standards* y hace referencia al conjunto de estándares que describen el procesamiento de documentos, algoritmos de cifrado, y otros estándares de tecnologías de información para ser usados principalmente por agencias de gobierno de tipo no militar en los Estados Unidos de América, al igual que los fabricantes que trabajan con dichas agencias.
Firewall.- Se refiere a un dispositivo que es usado para prevenir accesos no autorizados desde o hacia una red de datos privada. Los firewalls pueden estar basados en *hardware*, *software* o una combinación de ambos.
FMA.- Por sus siglas en inglés significa *Fault Management Architecture* y se refiere a una arquitectura construida dentro del sistema operativo *Solaris*, cuyo propósito es detectar y prevenir fallas en los servidores.

GSSAPI.- Por sus siglas en inglés significa *Generic Security Services API* y se refiere a la especificación de una interfaz de programación de aplicaciones para programas específicos que requieren acceso a servicios de seguridad, definida por el *Internet Engineering Task Force (IETF).*
Hash.- Se refiere a un valor generado por medio de una función criptográfica, también es conocido como *message digest.* Dicho valor es generado a partir de una cadena de texto que alimenta la función criptográfica.
HBA.- Por sus siglas en inglés significa *Host Bus Adapter* y se refiere a una tarjeta de circuitos que provee el procesamiento de entrada y salida, así como la conectividad física entre un servidor y un dispositivo de almacenamiento.
Host.- Es un término que se utiliza para denominar a un servidor que se encuentra conectado a una red para la cuál provee servicios, generalmente éste es accedido desde una terminal por usuarios localizados en una ubicación remota.
Hot-Swap.- Es un término que se utiliza para referirse a un componente o dispositivo de computadora que puede ser fácilmente removido y reemplazado sin necesidad de apagar el sistema.
HP LoadRunner.- Es un producto de la marca HP que es utilizado para automatizar pruebas de rendimiento y examinar el comportamiento de los sistemas, mientras genera carga para los mismos.
HTML.- Por sus siglas en inglés significa *HyperText Markup Language* y se refiere a un conjunto de estándares usados para etiquetar un documento de hipertexto. Es un protocolo estándar usado para formatear y desplegar documentos en la *World Wide Web.*
HTTP.- Por sus siglas en inglés significa *HyperText Transfer Protocol* y se refiere a un protocolo usado para establecer comunicación y transferencia de datos a través de la *World Wide Web*.
HTTPS.- Por sus siglas en inglés significa *HyperText Transfer Protocol Secure* y se refiere al protocolo HTTP cifrado por medio del protocolo SSL.
DB2.- Es una contracción que significa *Database 2,* y se refiere a un sistema de administración de bases de datos relacionales de la marca IBM.
ICANN.- Por sus siglas en inglés significa *Internet Corporation for Assigned Names and Numbers* y se refiere a una corporación organizada internacionalmente y sin ánimo de lucro, que tiene entre otras responsabilidades, la del alojamiento del espacio de direcciones del protocolo de internet o IP.
IETF.- Por sus siglas en inglés significa *Internet Engineering Task Force* y se refiere a una comunidad abierta internacional de diseñadores, operadores, fabricantes, e investigadores de redes, dedicados al estudio de la evolución de la arquitectura y la operación de internet.
ISO.- Por sus siglas en inglés significa *International Standards Organization* que básicamente se refiere a una organización internacional para la estandarización que promulga los estándares para productos comerciales, e industriales, así como procesos.
ITDS.- Por sus siglas en inglés significa *IBM Tivoli Directory Server,* básicamente es una implementación del protocolo LDAP de la marca IBM.
ITU-T.- Por sus siglas en inglés significa *International Telecommunication Union - Telecommunication* y se refiere a un sector particular dentro de una agencia encargada de desarrollar estándares internacionales en el área de telecomunicaciones que surgen a partir de recomendaciones alrededor del mundo.
IP.- Por sus siglas en inglés significa *Internet Protocol* y se refiere a uno de los protocolos principales de comunicaciones, encargado entregar paquetes denominados como datagramas a través de diferentes redes gracias a que cuenta con funciones de enrutamiento.
ISACA.- Por sus siglas en inglés significa *Information Systems Audit and Control Association* y se refiere a una asociación profesional de ámbito internacional, enfocado en el estudio de la gobernanza de Tecnologías de la Información.

ITIL.- Por sus siglas en inglés significa *Information Technology Infrastructure Library* y se refiere a un conjunto de prácticas para la administración de un servicio de TI, va enfocado principalmente en alinear los servicios de TI con las necesidades del negocio.
Java EE.- Por sus siglas en inglés significa *Java Enterprise Edition* y se refiere a una plataforma de desarrollo empresarial basada en *Java,* la cuál provee una API y un ambiente de ejecución.
Kernel.- Se refiere al módulo central del sistema operativo, siendo el primer componente que es alojado en la memoria principal del sistema al momento de arrancar.
KPI.- Por sus siglas en inglés significa *Key Performance Indicator* y se refiere a un conjunto de mediciones cuantificables que se utilizan para comparar el rendimiento con respecto a los objetivos estratégicos y operacionales, y así determinar el cumplimiento de los factores críticos de éxito.
LAN.- Por sus siglas en inglés significa *Local Area Network* y se refiere a una red de área local que abarca un grupo de computadoras que generalmente se encuentran en una proximidad cercana.
Layout.- Es un término que bajo el contexto del presente trabajo se refiere a la forma en como queda la tabla de particiones de un disco duro.
LDAP.- Por sus siglas en inglés significa *Lightweight Directory Access Protocol* y se refiere a un protocolo estándar y extensible de acceso a directorio que es usado por los clientes y los servidores de LDAP para comunicarse entre si.
LDIF.- Por sus siglas en inglés significa *LDAP Data Interchange Format* y se refiere a un conjunto de estándares para formatear un archivo de entrada requerido por las utilerías de línea de comando de LDAP.
Linux.- Es un sistema operativo de código abierto basado en *UNIX* escrito en lenguaje de programación C.
Mainframe.- Se refiere a una computadora de alto rendimiento diseñada para soportar cientos o hasta miles de usuarios de manera simultanea, al igual que la ejecución de varios programas al mismo tiempo.
MD5.- Por sus siglas en inglés significa *Message Digest 5* y se refiere a un *checksum* basado en un algoritmo de *hash* que verifica que la información ha permanecido sin alteraciones desde su última modificación.
MTTR.- Por sus siglas en inglés significa *Mean Time To Repair* y se refiere al tiempo que toma reparar un componente que pueda afectar parcial o totalmente la entrega de un servicio.
Multicore.- Es un término que generalmente se utiliza para referirse a un procesador que se encuentra conformado por varios *cores.*
Multithread.- Es un término que generalmente se utiliza para referir que un procesador o *core* cuenta con varios hilos de ejecución.
MTBF.- Por sus siglas en inglés significa *Mean Time Between Failure* y se refiere al tiempo transcurrido entre 2 fallas consecutivas ocurridas a un componente que afecte parcial o totalmente la entrega de un servicio
NDA.- Por sus siglas en inglés significa *Non Disclosure Agreement* y se refiere un acuerdo legal de confidencialidad en el que 2 o más partes deciden no revelar la información compartida entre ellos hacia entidades externas, a menos que haya una aprobación previa.
NSA.- Por sus siglas en inglés significa *National Security Agency* y se refiere a una agencia encargada de producir y administrar señales de inteligencia para los Estados Unidos de América, y opera bajo la jurisdicción del departamento de defensa.
OU.- Por sus siglas en inglés significa *Organization Unit* y se refiere al nombre único de una entrada en el directorio. Este comprende una unidad de negocio dentro de una organización.
PCI.- Por siglas en inglés significa *Payment Card Industry* y se refiere a un conjunto de estándares específicos de seguridad que fueron desarrollados para proteger la

información de las tarjetas durante y después de una transacción financiera.
PDU.- Por sus siglas en inglés significa *Power Distribution Unit* y se refiere a un dispositivo utilizado en los centros de datos para distribuir potencia eléctrica en corriente alterna requerida por la infraestructura de *hardware.*
PKI.- Por sus siglas en inglés significa *Public Key Infrastructure* y básicamente se refiere a la infraestructura necesaria para permitir que los usuarios de una red pública insegura puedan cifrar el contenido de los datos por medio de un par de llaves obtenidas y compartidas a través de una autoridad confiable.
Proxy.- Se utiliza para referirse a un servidor que se encuentra colocado entre una aplicación cliente y un servidor real. El *proxy* intercepta todas las solicitudes dirigidas hacia el servidor real para determinar si pueden o no ser completadas.
RACF.- Por sus siglas en inglés significa *Resource Access Control Facility* y se refiere a un producto de la marca *IBM,* usado para la administración de la seguridad principalmente sobre el sistema operativo de *mainframe, OS/390 (MVS).*
RAM.- Por sus siglas en inglés significa *Random Access Memory* y se refiere a un tipo de memoria volátil que puede ser accedida de manera aleatoria. Esto es, cualquier *byte* de memoria puede ser accedido sin tocar los *bytes* precedentes.
RBAC.- Por sus siglas en inglés significa *Role Based Access Control* y se refiere a un sistema de control de acceso a los recursos de una computadora, basado en el rol del usuario.
RFC.- Por sus siglas en inglés significa *Request For Change* y se refiere a un documento formal, resultado de un comité y que posteriormente será revisado por las partes interesadas.
RFP.- Por sus siglas en inglés significa *Request For Proposal* y se refiere a un tipo de solicitud de oferta en la que una organización anuncia que tiene fondos disponibles para un proyecto en particular.
RISC.- Por sus siglas en inglés significa *Reduced Instruction Set Computer* y se refiere a una arquitectura de procesador diseñada para soportar una lista corta de instrucciones, haciéndolo más simple y rápido.
ROI.- Por sus siglas en inglés significa *Return On Investment* y se refiere a una métrica de rendimiento utilizada para evaluar la eficiencia de una inversión. Para calcularlo es necesario dividir el beneficio entre el costo de inversión y el resultado siempre es expresado como un porcentaje.
root.- Se refiere a una cuenta de usuario usada para la administración del sistema operativo, que principalmente es UNIX / Linux.
Round-Robin.- Se refiere a un algoritmo para procesar peticiones que demandan algún tipo de recurso. Dicho algoritmo generalmente asigna fracciones de tiempo de igual proporción y en orden circular a cada uno de los procesos, sin ningún tipo de prioridad.
Router.- Se refiere a un dispositivo que tiene la capacidad para reenviar paquetes de datos entre diferentes redes de computadoras.
RUV.- Por sus siglas en inglés significa *Replication Update Vector* y se refiere a un vector de réplica que identifica el estado de cada réplica dentro de una topología de alta disponibilidad del servidor de directorio.
SAF.- Por sus siglas en inglés significa *System Authorization Facility* y se refiere a una interfaz que trabaja en conjunto con RACF y permite a distintos programas usar servicios de autorización para controlar el acceso a los recursos.
Salted.- Es un término que se usa para hacer referencia a una contraseña que ha sido generada a partir de una función criptográfica de manera aleatoria.
SAN.- Por sus siglas en inglés significa *Storage Area Network* y se refiere a una red de alto rendimiento, cuyo propósito principal es habilitar dispositivos de almacenamiento para comunicarse con sistemas de procesamiento, así como entre ellos mismos.
SASL.- Por sus siglas en inglés significa *Simple Authentication Security Layer* y se refiere a un marco de trabajo que provee servicios de autenticación y seguridad en los datos mediante mecanismos basados en protocolos orientados a la conexión.
SHA.- Por sus siglas en inglés significa *Secure Hash Algorithm* y se refiere a un

algoritmo desarrollado como estándar por la *National Security Agency o NSA* para el uso de firmas digitales.
Shell Script.- Es un *script* desarrollado para ser ejecutado por el interprete de la línea de comando de un sistema operativo.
Skype.- Es una aplicación propietaria que brinda el servicio de voz sobre IP *(VoIP)*, cuyo fabricante es *Microsoft.*
SKMS.- Por sus siglas en inglés significa *Service Knowledge Management System* y se refiere a un conjunto de herramientas y bases de datos que son usadas para administrar el conocimiento y la información, incluyendo el *Configuration Management System*.
SLA.- Por sus siglas en inglés significa *Service Level Agreement* y se refiere básicamente al nivel de servicio acordado como parte del contrato celebrado entre el proveedor de un servicio y un cliente. Este es expresado como el porcentaje que los servicios están disponibles.
SPARC.- Por sus siglas en inglés significa *Scalable Processor ARChitecture* y se refiere a una arquitectura de procesador de tipo RISC que fue creada en el año 1985 por la compañía *Sun Microsystems Inc.*
SPEC.- Por sus siglas en inglés significa *Standard Performance Evaluation Corporation,* y básicamente se refiere a una organización sin ánimo de lucro que produce, establece, mantiene, y ratifica un conjunto de pruebas de rendimiento para computadoras de diferentes fabricantes.
SSL.- Por sus siglas en inglés significa *Secure Sockets Layer* y básicamente se refiere un protocolo desarrollado por *Netscape* para transmitir documentos privados vía internet utilizando un sistema criptográfico que utiliza una llave pública para cifrar los datos y una llave privada para descifrarlos.
Switch.- Es un dispositivo de red que se utiliza ampliamente para enlazar diferentes segmentos de red o incluso varios dispositivos de red.
TAM.- Por sus siglas en inglés significa *Tivoli Access Manager* y se refiere a una solución de autenticación y autorización de la marca IBM, utilizada principalmente para proteger servicios web corporativos.
TCO.- Por sus siglas en inglés significa *Total Cost of Ownership* y se refiere a un cálculo financiero que se utiliza para estimar los costos totales directos e indirectos por poseer un bien. En esta ocasión va enfocado a un componente de TI.
TCP.- Por sus siglas en inglés significa *Transport Control Protocol* y se refiere a un protocolo conformado por un conjunto de reglas usadas para enviar y recibir datos en forma de mensajes entre computadoras. Dicho protocolo controla y registra las unidades de datos enviadas y/o recibidas.
TIM.- Por sus siglas en inglés significa *Tivoli Identity Manager* y se refiere a un producto de la marca IBM, el cual provee una plataforma de administración centralizada del ciclo de vida de las identidades.
Timeout.- Se refiere a una señal de cancelación después de haber esperado un periodo específico de tiempo sin haber tenido respuesta.
TLS.- Por sus siglas en inglés significa *Transport Layer Security* que se refiere al protocolo que asegura la privacidad de los datos que viajan por la red y este supercede y es una extensión del protocolo SSL.
Token.- Se refiere a un dispositivo basado en *hardware* o en algunos casos basado en *software,* cuyo propósito principal es probar la identidad electrónica de una persona, y regularmente es usado para proveer un servicio de autenticación multifactor.
TPM.- Por sus siglas en inglés significa *Trusted Platform Module*, y se refiere a una especificación detallada de un cripto-procesador seguro que cuenta con la capacidad para almacenar las llaves necesarias para proteger la información.
UDP.- Por sus siglas en inglés significa *User Datagram Protocol* y se refiere a un protocolo no orientado a la conexión que sirve principalmente para transmitir información en una sola dirección desde la fuente al destino sin verificar la lectura y/o el estado del mensaje en el lado del receptor. Dicho protocolo forma parte de la suite

de *TCP/IP.*
UNIX.- Es un sistema operativo multi-usuario y multi-tarea desarrollado a principios de los 70s, escrito en lenguaje de programación C.
URL.- Por sus siglas en inglés significa *Uniform Resource Locator* y es una dirección regida por una convención estandarizada usada para identificar un recurso en la red.
UML.- Por sus siglas en inglés significa *Unified Modeling Language* y se refiere a una notación estándar para el modelado de los objetos del mundo real y que consiste de una serie de símbolos y conectores que pueden ser usados para crear diagramas.
VPN.- Por sus siglas en inglés significa *Virtual Private Network* y se refiere a una red que usa infraestructura pública de telecomunicaciones para proveer un acceso seguro a los usuarios de una red específica dentro de una organización.
VAC.- Por sus siglas en inglés significa *Voltage in Alternating Current,* y se refiere al voltaje requerido por un componente eléctrico.
XML.- Por sus siglas en inglés significa *Extensible Markup Language* que se refiere a una especificación desarrollada y regulada por el *World Wide Web Consortium (W3C)* que se utiliza principalmente para la definición, transmisión, validación, e interpretación de datos entre diferentes aplicaciones.

Referencias

- Allspaw J. (2008). *The Art of Capacity Planning 1st Edition*. Sebastopol, CA 95472: O'Reilly Media, Inc. ISBN: 9780596518578.

- Andress, J. (2011). *The Basics of Information Security: Understanding The Fundamentals of InfoSec in Theory and Practice*. Syngress, Waltham, MA 02451, USA. ISBN: 978-1-59749-653-7.

- Bialasky, T. & Haines, M. (2001). *Solaris and LDAP Naming Services: Deploying LDAP in the Enterprise*. Prentice Hall PTR. ISBN-13: 978-0130306784.

- BMC Software Inc. (2012). *BMC Remedy Action Request System [online]* disponible en: http://www.bmc.com/products/product-listing/remedy-action-request-system.html [Consultado el 20 de Diciembre de 2012]

- Buecker, A., Lay, S., Rahnenfuehrer, D., & Sommer, F. (2009). *Deployment Guide Series: IBM Tivoli Access Manager for Enterprise Single Sign-On 8.0.* International Business Machines Corp. ISBN: 0738432954.

- Carter, G. (2003). *LDAP System Administration*. Sebastopol, CA 95472: O'Reilly & Associates, Inc. ISBN: 1-56592-491-6.

- Cheswick, W. R., Bellovin S. M., & Rubin, A. D. (2003), *Firewalls and Internet Security*, Second Edition, Addison-Wesley, ISBN: 0-201-63466-X.

- Compuware Corporation. (2012). *Application Monitoring [online]* disponible en: http://www.compuware.com/application-performance-management [Consultado el 25 de Noviembre de 2012]

- Davies, S., d´Amico, M., Rettore, E., Witterick, D., de Valence, F., & Young, T. (2005). *Distributed Security and High Availability with Tivoli Access Manager and WebSphere Application Server for z/OS.* International Business Machines Corp. ISBN: 0738490083.

- Hausman, K., Weiss, M., & Barrett, D. (2011). *CompTIA® Security+™ SY0-301 Exam Cram*. Pearson Certification. ISBN-13: 978-0-7897-4829-4.

- International Organization for Standardization. (s.a.) *ISO/IEC 27000:2009 [online]* disponible en: http://www.iso.org/iso/catalogue_detail?csnumber=41933 [Consultado el 18 de Enero de 2011]

- Internet Assigned Numbers Authority. (2013), *Service Name and Transport Protocol Port Number Registry [online]* disponible en: http://www.iana.org/assignments/service-names-port-numbers/service-names-port-numbers.xhtml [Consultado el 16 de Junio de 2013]

- IT Governance Institute. (2007). *COBIT® 4.1*, ISBN: 1-933284-72-2.

- Office of Government Commerce. (2007a), *ITIL Continual Service Improvement*, Norwich, Crown Copyright. ISBN: 9780113310494.

- Office of Government Commerce. (2007b), *ITIL Service Design*, Norwich, Crown Copyright. ISBN: 9780113310470.

- Office of Government Commerce. (2007c), *ITIL Service Operation*, Norwich, Crown Copyright. ISBN: 9780113310463.

- Office of Government Commerce. (2007d), *ITIL Service Transition*, Norwich, Crown Copyright. ISBN: 9780113310487.

- Office of Government Commerce. (2007e), *The Official Introduction to the ITIL Service Lifecycle*, Norwich, Crown Copyright. ISBN: 9780113310616.

- Oracle Corporation. (2010), *Sun Directory Server Enterprise Edition 7.0 Release Notes [online]* disponible en: http://docs.oracle.com/cd/E19424-01/820-4805/install-notes/index.html [Consultado el 3 de Enero de 2011]

- Oracle Corporation. (2011). *Oracle and Sun [online]* disponible en: http://www.oracle.com/us/sun/index.htm [Consultado el 13 de Enero de 2013]

- Peltier, T. R., Peltier, J., & Blackley, J. (2005), *Information Security Fundamentals*, Auerbach Publications ISBN: 0-8493-1957-9.

- Standard Performance Evaluation Corporation. (2010), *SPECjEnterprise®2010 Result [online]* disponible en: http://www.spec.org/jEnterprise2010/results/res2010q1/jEnterprise2010-20091207-00002.html [Consultado el 08 de Diciembre de 2010]

- Sun Microsystems Inc. (2009), *Sun Directory Server Enterprise Edition 7.0 Reference*, Network Circle, Santa Clara, CA 95054 U.S.A.

- Symantec Corporation. (2005), *Data Sheet: Security Management [online]* disponible en: http://eval.symantec.com/mktginfo/enterprise/fact_sheets/ent-factsheet_enterprise_security_manager_6.5_06-2005.en-us.pdf [Consultado el 19 de Julio de 2013]

- The Internet Engineering Task Force. (2006) *Lightweight Directory Access Protocol (LDAP) : Syntaxes and Matching Rules [online]* disponible en: http://www.ietf.org/rfc/rfc4517.txt [Consultado el 15 de Enero de 2011]

- U.S. Securities and Exchange Commission. (2003), *Retention of Records Relevant to Audits and Reviews [online]* disponible en: http://www.sec.gov/rules/final/33-8180.htm [Consultado el 15 de Mayo de 2013]

- Welytok J. G., (2006). *Sarbanes-Oxley For Dummies*, Wiley Publishing Inc. ISBN-13: 978-0-471-76846-3.

ANEXO A.- PROCEDIMIENTO DE IMPLEMENTACION

Como se sabe, el servicio denominado como *BNET* pertenece al grupo financiero referido anteriormente y en Venezuela, a la filial del cliente, por medio de dicho servicio se ofrece a millones de clientes el acceso al banco a cualquier hora, desde cualquier lugar y de la manera más conveniente, a través de una de las redes de distribución más extensa del país, una novedosa red de atención telefónica y acceso a servicios de *Home Banking* vía internet. Debido a la importancia de este servicio, el cliente se ha preocupado por mantener su plataforma actualizada y con las últimas versiones en todos sus componentes involucrados.
Una parte crítica de dicho servicio es la plataforma de autenticación web basada en el dominio seguro, la cuál consta básicamente de *IBM Tivoli Access Manager for E-Business 6.1.1* y *Sun Directory Server 7.0* como repositorio de usuarios.
La solución del dominio seguro fue implementada de acuerdo a los tiempos estipulados en el plan de trabajo.
La primera parte de la implementación se refiere a la capa de directorio y básicamente se utilizaron 4 instancias de *Solaris 10*, de las cuales 2 de ellas fueron dedicadas para los servidores de directorio y 2 más para los servidores de *proxy*.
A continuación se describe el proceso que fue ejecutado paso a paso de manera secuencial en cada uno de los nodos. Es importante señalar que con el afán de eliminar la redundancia de los comandos ejecutados y las salidas reportadas, solo se incluye la secuencia para uno de los nodos en cada capa, teniendo en cuenta que en el segundo nodo se teclearon exactamente los mismos comandos.

Para el primer servidor de directorio:

```
root@cpprssc01.banco.corp # prtdiag

System Configuration: Sun Microsystems sun4u Sun SPARC Enterprise M4000
Server
System clock frequency: 1012 MHz
Memory size: 49152 Megabytes

================== CPUs ===================================
CPU CPU Run L2$ CPU CPU
LSB Chip ID MHz MB Impl. Mask
--- ---- ------------------------------------ ---- --- ----- ----
00 0 0, 1, 2, 3, 4, 5, 6, 7 2400 5.0 7 145
00 1 8, 9, 10, 11, 12, 13, 14, 15 2400 5.0 7 145
00 2 16, 17, 18, 19, 20, 21, 22, 23 2400 5.0 7 145
00 3 24, 25, 26, 27, 28, 29, 30, 31 2400 5.0 7 145

================== Memory Configuration ===========================
Memory Available Memory DIMM # of Mirror Interleave
LSB Group Size Status Size DIMMs Mode Factor
--- ------ ------------------ ------- ------ ----- ------- ----------
00 A 16384MB okay 2048MB 8 no 4-way
00 B 16384MB okay 2048MB 8 no 4-way

========================= IO Cards =========================
LSB Name Model
--- -------------- ------------
00 scsi LSI,1064
00 network N/A
00 network N/A
00 SUNW,qlc QLE2462
00 SUNW,qlc QLE2462
00 network SUNW,pcie-qg
00 network SUNW,pcie-qg
00 network SUNW,pcie-qg
00 network SUNW,pcie-qg
00 SUNW,qlc QLE2462
00 SUNW,qlc QLE2462

==================== Hardware Revisions ==================== ======
```

```
System PROM revisions:
----------------------
OBP 4.24.12 2009/10/30 06:59

================== Environmental Status ================== =======
Mode switch is in UNLOCK mode

================== System Processor Mode ================== ======
SPARC64-VII mode
```

La versión del sistema operativo instalado fue verificada con el siguiente comando:

```
root@cpprssc01.banco.corp # more /etc/release

Solaris 10 10/09 s10s_u8wos_08a SPARC
Copyright 2009 Sun Microsystems, Inc. All Rights Reserved.
Use is subject to license terms.
Assembled 16 September 2009
```

Dicha instancia no debía contar con zonas no globales, ni contenedores, así que:

```
root@cpprssc01.banco.corp # zoneadm list -ivc
ID NAME STATUS PATH BRAND IP
0 global running / native shared
```

Dentro de dicho servidor de directorio se verificaron los siguientes filesystems que habían sido creados previamente de acuerdo a la lista de requerimientos:

Filesystem	Tamaño	Usado	Disponible	Capacidad	Montaje
/dev/md/dsk/d0	20G	4.9G	15G	25.00%	/
/devices	0K	0K	0K	0.00%	/devices
ctfs	0K	0K	0K	0.00%	/system/contract
proc	0K	0K	0K	0.00%	/proc
mnttab	0K	0K	0K	0.00%	/etc/mnttab
swap	95G	1.7M	95G	1.00%	/etc/svc/volatile
objfs	0K	0K	0K	0.00%	/system/object
sharefs	0K	0K	0K	0.00%	/etc/dfs/sharetab
fd	0K	0K	0K	0.00%	/dev/fd
/dev/md/dsk/d3	9.9G	3.4G	6.4G	35.00%	/usr
swap	96G	861M	95G	1.00%	/tmp
swap	95G	88K	95G	1.00%	/var/run
swap	80G	7.5G	73G	10.00%	/dbcache
/dev/md/dsk/d4	6.9G	14M	6.8G	1.00%	/export/home
ldapcores	82G	21K	82G	1.00%	/cores
ldapseguridad	3.9G	3.0G	961M	76.00%	/seguridad
ldapbackup	41G	31K	41G	1.00%	/var/opt/backup

ldapdb	82G	12G	71G	14.00%	/var/opt/db
ldaplogs	109G	9.1G	100G	9.00%	/var/opt/logs
ldpapmps	5.9G	450M	5.4G	8.00%	/var/opt/mps
ldpapwsBackup	5.9G	21K	5.9G	1.00%	/wsBackup

Tabla A.1 Distribución de *filesystems* para la capa de directorio

Todos los filesystems donde residen los archivos relacionados con el servidor de directorio se encuentran en un arreglo de discos externo *EMC VNX 5700* bajo las rutas emcpower0a - emcpower9a gestionadas a través del software de alta disponibilidad *EMC Powerpath*. Dichos filesystems se encuentran configurados con el manejador de volumenes *ZFS*. A continuación se presenta la forma en como se verificó la configuración de dichos volúmenes:

```
root@cpprssc01.banco.corp # zfs list

NAME                 USED         AVAIL        REFER        MOUNTPOINT
ldapbackup           444K         41.1G        31K          /var/opt/backup
ldapcores            82.5K        82.2G        21K          /cores
ldapdb               11.5G        70.7G        11.5G        /var/opt/db
ldaplogs             9.08G        100G         9.08G        /var/opt/logs
ldapseguridad        2.97G        961M         2.97G        /seguridad
ldpapmps             451M         5.44G        450M         /var/opt/mps
ldpapwsBackup        84K          5.88G        21K          /wsBackup
```

Todos los volúmenes mostrados anteriormente y que se encuentran configurados bajo el control de *ZFS,* actualmente son vistos a través de las controladoras C1 y C3, como se presenta a continuación la salida del comando utilizado para comprobar que los canales de acceso eran redundantes:

```
root@cpprssc01.banco.corp # cfgadm -al

Ap_Id         Type         Receptacle   Occupant           Condition
SB0    System_Brd          connected    configured         ok
SB0::cpu0     cpu           connected    configured         ok
SB0::cpu1     cpu           connected    configured         ok
SB0::cpu2     cpu           connected    configured         ok
SB0::cpu3     cpu           connected    configured         ok
SB0::memory   memory        connected    configured         ok
SB0::pci0     io            connected    configured         ok
SB0::pci1     io            connected    configured         ok
SB0::pci2     io            connected    configured         ok
SB0::pci3     io            connected    configured         ok
SB0::pci8     io            connected    configured         ok
SB1                         disconnected unconfigured       unknown
SB2                         disconnected unconfigured       unknown
SB3                         disconnected unconfigured       unknown
SB4                         disconnected unconfigured       unknown
SB5                         disconnected unconfigured       unknown
SB6                         disconnected unconfigured       unknown
SB7                         disconnected unconfigured       unknown
SB8                         disconnected unconfigured       unknown
SB9                         disconnected unconfigured       unknown
SB10                        disconnected unconfigured       unknown
SB11                        disconnected unconfigured       unknown
SB12                        disconnected unconfigured       unknown
SB13                        disconnected unconfigured       unknown
SB14                        disconnected unconfigured       unknown
SB15                        disconnected unconfigured       unknown
c0            scsi-sata     connected    configured         unknown
c0::dsk/c0t0d0      disk    connected    configured         unknown
```

```
c0::dsk/c0t1d0      disk   connected    configured          unknown
c0::dsk/c0t3d0      CD-ROM connected    configured          unknown
c0::rmt/0           tape   connected    configured          unknown
c1           fc-fabric     connected    configured          unknown
c1::5006048452a5cf99 disk connected     configured          unknown
c2           fc            connected    unconfigured        unknown
c3           fc-fabric     connected    configured          unknown
c3::5006048452a5cf96 disk connected     configured          unknown
c4           fc            connected    unconfigured        unknown
```

Con respecto a la configuración de las interfaces de red, se verificaron las direcciones *IP*, máscaras de red, direcciones de *broadcast*, y direcciones *ethernet* (48 bits), a través de diferentes comandos de red. Sin embargo por motivos de confidencialidad unicamente se muestra el comando a ejecutar

```
root@cpprssc01.banco.corp # ifconfig -a
```

Posteriormente se validaron las tablas de ruteo que tenian declaradas los servidores a través del siguiente comando:

```
root@cpprssc01.banco.corp # netstat -nr
```

El siguiente paso fue desplegar una lista de los controles de recursos definidos en el sistema operativo *Solaris 10*:

```
root@cpprssc01.banco.corp # rctladm -l
```

```
process.max-port-events syslog=off [ deny count ]
process.max-msg-messages syslog=off [ deny count ]
process.max-msg-qbytes syslog=off [ deny bytes ]
process.max-sem-ops syslog=off [ deny count ]
process.max-sem-nsems syslog=off [ deny count ]
process.max-address-space syslog=off [ lowerable deny no-signal bytes ]
process.max-file-descriptor syslog=off [ lowerable deny count ]
process.max-core-size syslog=off [ lowerable deny no-signal bytes ]
process.max-stack-size syslog=off [ lowerable deny no-signal bytes ]
process.max-data-size syslog=off [ lowerable deny no-signal bytes ]
process.max-file-size syslog=off [ lowerable deny file-size bytes ]
process.max-cpu-time syslog=off [ lowerable no-deny cpu-time inf seconds ]
task.max-cpu-time syslog=off [ no-deny cpu-time no-obs inf seconds ]
task.max-lwps syslog=off [ count ]
project.max-contracts syslog=off [ no-basic deny count ]
project.max-device-locked-memory syslog=off [ no-basic deny bytes ]
project.max-locked-memory syslog=off [ no-basic deny bytes ]
project.max-port-ids syslog=off [ no-basic deny count ]
project.max-shm-memory syslog=off [ no-basic deny bytes ]
project.max-shm-ids syslog=off [ no-basic deny count ]
project.max-msg-ids syslog=off [ no-basic deny count ]
project.max-sem-ids syslog=off [ no-basic deny count ]
project.max-crypto-memory syslog=off [ no-basic deny bytes ]
project.max-tasks syslog=off [ no-basic count ]
project.max-lwps syslog=off [ no-basic count ]
project.cpu-cap syslog=off [ no-basic deny no-signal inf count ]
project.cpu-shares syslog=n/a [ no-basic no-deny no-signal no-syslog count ]
zone.max-swap syslog=off [ no-basic deny bytes ]
zone.max-locked-memory syslog=off [ no-basic deny bytes ]
zone.max-shm-memory syslog=off [ no-basic deny bytes ]
zone.max-shm-ids syslog=off [ no-basic deny count ]
zone.max-sem-ids syslog=off [ no-basic deny count ]
zone.max-msg-ids syslog=off [ no-basic deny count ]
zone.max-lwps syslog=off [ no-basic count ]
zone.cpu-cap syslog=off [ no-basic deny no-signal inf count ]
zone.cpu-shares syslog=n/a [ no-basic no-deny no-signal no-syslog count ]
```

Antes de ejecutar el procedimiento de migración, se respaldo debidamente el sufijo

declarado en *RACF*, denominado como el sufijo de negocio (o=InstitucionBancaria). Posteriormente el archivo generado por el respaldo fue transferido hacia uno de los nodos de directorio, y en seguida se tecleo el siguiente comando para verificarlo:

```
root@cpprssc01.banco.corp # ls -ltrha /var/opt/backup/export-May18/

total 8971510
drwxr-xr-x 2 root root 4 Jan 18 22:49 .
-rw------- 1 root root 3.7G Jan 18 23:35 userRoot.ldif
```

Finalmente se validaron los siguientes parámetros de kernel dentro del run control script *rc3.d* para asegurar que el stack de *TCP/IP* de *Solaris 10* funcione de manera correcta:

```
root@cpprssc01.banco.corp # cd /etc/rc3.d

root@cpprssc01.banco.corp # more S68inettune

#Seteo de parametros tcp
ndd -set /dev/tcp tcp_co_min 1500
ndd -set /dev/tcp tcp_compression_enabled 1
ndd -set /dev/tcp tcp_conn_grace_period 0
ndd -set /dev/tcp tcp_conn_req_max_q 10240
ndd -set /dev/tcp tcp_conn_req_max_q0 10240
ndd -set /dev/tcp tcp_conn_req_min 1
ndd -set /dev/tcp tcp_debug 0
ndd -set /dev/tcp tcp_deferred_ack_interval 100
ndd -set /dev/tcp tcp_fin_wait_2_flush_interval 67500
ndd -set /dev/tcp tcp_ip_abort_cinterval 180000
ndd -set /dev/tcp tcp_ip_abort_interval 60000
ndd -set /dev/tcp tcp_ip_abort_linterval 180000
ndd -set /dev/tcp tcp_keepalive_interval 900000
ndd -set /dev/tcp tcp_naglim_def 1
ndd -set /dev/tcp tcp_recv_hiwat 400000
ndd -set /dev/tcp tcp_time_wait_interval 30000
ndd -set /dev/tcp tcp_xmit_hiwat 400000
root@cpprssc01.banco.corp # pwd
/etc/rc3.d
```

Iniciando con el proceso de instalación del software de directorio, se copian todos los paquetes a un area del sistema operativo *Solaris 10*, y a continuación se detalla el proceso de instalación para dichos paquetes:

```
root@cpprssc01.banco.corp # cd /ruta-de-instalacion/
root@cpprssc01.banco.corp # gunzip sun-dsee7.tar.gz
root@cpprssc01.banco.corp # tar –cvf sun-dsee7.tar
```

Posteriormente se procedió a la creación y configuración de la instancia de directorio por medio de los siguientes comandos:

```
root@cpprssc01.banco.corp # dsadm create –h hostname –p 389 –P 636
/ruta/dsBanco

Choose the Directory Manager password: password
Confirm the Directory Manager password: password
Use 'dsadm start /local/dsBanco' to start the instance
```

El siguiente paso fué inicializar los procesos asociados a la instancia:

```
root@cpprssc01.banco.corp # dsadm start /ruta/dsBanco
Server started: pid=2845
```

Una vez inicializados los servicios se procedió con la creación de los sufijos dentro de la estructura del árbol de directorios:

```
root@cpprssc01.banco.corp # dsconf create-suffix –h hostname –p 389 –e
o=InstitucionBancaria
```

El siguiente paso fue importar todos los datos contenidos en el archivo *LDIF* que me fue proporcionado previamente:

```
root@cpprssc01.banco.corp # dsconf import –h hostname –p 389 –e
/ruta/del/archivo/ldif o=InstitucionBancaria
Enter "cn=root" password:
New data will override existing data of the suffix "o=InstitucionBancaria".
Initialization will have to be performed on replicated suffixes.
...
...
## Closing files...
## Import complete. Processed 160 entries in 4 seconds. (40.00 entries/sec)

Task completed (slapd exit code: 0).
```

El código de salida igual a cero nos indica que la carga de datos dentro del sufijo fué exitosa.

Posteriormente se prosiguió con la instalación y configuración de los servidores de *proxy*, para lo cuál también se hizo una serie de validaciones, y como primer paso se verificó la versión del sistema operativo instalado por medio del siguiente comando:

```
root@CPPRSAPRXY # more /etc/release

Solaris 10 10/09 s10s_u8wos_08a SPARC
Copyright 2009 Sun Microsystems, Inc. All Rights Reserved.
Use is subject to license terms.
Assembled 16 September 2009
root@CPPRSAPRXY # uname -a

SunOS CPPRSAP3X 5.10 Generic_142900-13 sun4u sparc SUNW,SPARC-Enterprise
```

La plataforma de hardware para cada uno de los 2 servidores de *Proxy* es idéntica, ya que cada instancia se ejecuta sobre un dominio de *Sun SPARC Enterprise M4000*. A continuación se presenta el procedimiento que muestra la salida que detalla la configuración para dichos dominios:

```
root@CPPRSAPRXY # prtdiag|more

System Configuration: Sun Microsystems sun4u Sun SPARC Enterprise M4000
Server
System clock frequency: 1012 MHz
Memory size: 49152 Megabytes

==================================== CPUs ================================
CPU CPU Run L2$ CPU CPU
LSB Chip ID MHz MB Impl. Mask
--- ---- ------------------------------------- ---- --- ----- ----
01 1 40, 41, 42, 43, 44, 45, 46, 47 2530 5.5 7 161

============================ Memory Configuration
============================
Memory Available Memory DIMM # of Mirror Interleave
LSB Group Size Status Size DIMMs Mode Factor
--- ------ ------------------ ------- ------ ----- ------- ----------
01 A 49152MB okay 2048MB 4 no 2-way

======================== IO Cards ========================
LSB Name Model
--- ------------- ------------
01 LSILogic,sas LSI,1068E
```

```
01 SUNW,qlc QLE2462
01 SUNW,qlc QLE2462
01 SUNW,qlc QLE2462
01 SUNW,qlc QLE2462
01 network SUNW,pcie-qg
01 network SUNW,pcie-qg
01 network SUNW,pcie-qg
01 network SUNW,pcie-qg

==================== Hardware Revisions ====================
System PROM revisions:
----------------------
OBP 4.24.14 2010/07/02 13:21

=================== Environmental Status ===================
Mode switch is in LOCK mode

=================== System Processor Mode ===================
SPARC64-VII+ mode
```

En seguida se ejecutó un comando para validar la cantidad y tipo de los procesadores con que contaba cada dominio de *LDAP Proxy*:

```
root@CPPRSAPRXY # psrinfo

40 on-line since 10/28/2010 17:02:11
41 on-line since 10/28/2010 17:02:12
42 on-line since 10/28/2010 17:02:12
43 on-line since 10/28/2010 17:02:12
44 on-line since 10/28/2010 17:02:12
45 on-line since 10/28/2010 17:02:12
46 on-line since 10/28/2010 17:02:12
47 on-line since 10/28/2010 17:02:12
```

Después de observar la salida anterior se pudo apreciar que existía un total de 8 *cores* en cada dominio, ya que en realidad habían 2 procesadores, cada uno con 4 *cores*. Por otro lado, la configuración de las *HBAs*, por medio de las cuales se tiene acceso a los volúmenes externos publicados en la *SAN* se presenta a continuación:

```
root@CPPRSAPRXY # cfgadm -al

Ap_Id         Type         Receptacle          Occupant           Condition
SB0           disconnected                     unconfigured       unknown
SB1           System_Brd   connected           configured         ok
SB1::cpu1     cpu          connected           configured         ok
SB1::         memory       memory connected    configured         ok
SB1::pci2     io           connected           configured         ok
SB1::pci3     io           connected           configured         ok
SB1::pci8     io           connected           configured         ok
SB2                        disconnected        unconfigured       unknown
SB3                        disconnected        unconfigured       unknown
SB4                        disconnected        unconfigured       unknown
SB5                        disconnected        unconfigured       unknown
SB6                        disconnected        unconfigured       unknown
SB7                        disconnected        unconfigured       unknown
SB8                        disconnected        unconfigured       unknown
SB9                        disconnected        unconfigured       unknown
SB10                       disconnected        unconfigured       unknown
SB11                       disconnected        unconfigured       unknown
SB12                       disconnected        unconfigured       unknown
SB13                       disconnected        unconfigured       unknown
SB14                       disconnected        unconfigured       unknown
SB15                       disconnected        unconfigured       unknown
c0            scsi-bus     connected           unconfigured       unknown
c1            fc           connected           unconfigured       unknown
```

```
c2            fc            connected           unconfigured        unknown
c3            fc            connected           unconfigured        unknown
c4            fc            connected           unconfigured        unknown
iou#0-pci#3   scsi/hp       connected           configured          ok
iou#0-pci#4   pci-pci/hp    connected           configured          ok
iou#0-pci#4: iobE07ET.pcie1      unknown             empty unconfigured
      unknown
iou#0-pci#4:iobE07ET.pcie2 fibre/hp connected        configured   ok
iou#0-pci#4:iobE07ET.pcie3 unknown      empty unconfigured  unknown
iou#0-pci#4:iobE07ET.pcie4 fibre/hp connected        configured   ok
iou#0-pci#4:iobE07ET.pcie5 unknown      empty unconfigured  unknown
iou#0-pci#4:iobE07ET.pcie6 etherne/hp connected      configured   ok
```

Ambos servidores de *LDAP Proxy* quedaron configurados con *filesystems* de tipo *ZFS*, donde residen las instancias, datos, configuración y bitácoras. A continuación se muestra el comando ejecutado y la salida para verificar dichos *filesystems*:

```
root@CPPRSAPRXY # zfs list

NAME                 USED   AVAIL         REFER         MOUNTPOINT

pool_CPPRSAPRXY      14.5G  387G          21K           /pool_CPPRSAP33
pool_CPPRSAPRXY /1   3.92G  6.08G         3.92G         /var/opt/mps/dps
pool_CPPRSAPRXY/2    7.31G  42.7G         7.31G         /var/opt/dps/logs
pool_CPPRSAPRXY/3    21K    20.0G         21K           /cores/dps
pool_CPPRSAPRXY/4    21K    20.0G         21K           /var/opt/dps/backups
pool_CPPRSAPRXY/5    3.23G  1.77G         3.23G         /seguridad
```

El siguiente paso fue comprobar que realmente los grupos de discos estaban configurados con los volúmenes de la *SAN*.

```
root@CPPRSAPRXY # format

Searching for disks...done
AVAILABLE DISK SELECTIONS:
0. c5t600A0B800068967D0000046B4C987804d0 <SUN-LCSM100_S-0735-410.00GB>
/scsi_vhci/disk@g600a0b800068967d0000046b4c987804
1. c5t600A0B800068967D000004684C9876FAd0 <SUN-LCSM100_S-0735 cyl 46078 alt 2
hd 128 sec 64>
/scsi_vhci/disk@g600a0b800068967d000004684c9876fa
Specify disk (enter its number): ^C
```

Una vez que se habían validado las capacidades de procesamiento y almacenamiento dentro de cada uno de los ambientes de *LDAP Proxy*, en seguida se ejecutaron algunos comandos para validar la configuración a nivel de red, como direcciones *IP*, máscaras de red, direcciones de *broadcast,* así como tablas de ruteo. En seguida se muestran únicamente los comandos sin las salidas, esto por motivos de confidencialidad.

```
root@CPPRSAPRXY # ifconfig -a

root@CPPRSAPRXY # netstat -nr
```

Finalmente como parte del proceso de verificación en los ambientes de *LDAP Proxy* a continuación se muestra una lista de comandos ejecutados con los valores que debían ser configurados para los parámetros del stack de *TCP/IP* dentro del kernel del sistema operativo *Solaris 10*:

```
root@CPPRSAPRXY # ndd -set /dev/tcp tcp_conn_req_max_q 1024
root@CPPRSAPRXY # ndd -set /dev/tcp tcp_ip_abort_cinterval 10000
root@CPPRSAPRXY # ndd -set /dev/tcp tcp_ip_abort_interval 60000
root@CPPRSAPRXY # ndd -set /dev/tcp tcp_strong_iss 2
root@CPPRSAPRXY # ndd -set /dev/tcp tcp_smallest_anon_port 8192
```

```
root@CPPRSAPRXY # ndd -set /dev/tcp tcp_naglim_def 1
root@CPPRSAPRXY # ndd -set /dev/tcp tcp_keepalive_interval 180000
root@CPPRSAPRXY # ndd -set /dev/tcp tcp_time_wait_interval 30000
root@CPPRSAPRXY # ndd -set /dev/tcp tcp_co_min 1500
root@CPPRSAPRXY # ndd -set /dev/tcp tcp_fin_wait_2_flush_interval 67500
root@CPPRSAPRXY # ndd -set /dev/tcp tcp_maxpsz_multiplier 10
root@CPPRSAPRXY # ndd -set /dev/tcp tcp_xmit_hiwat 65536
root@CPPRSAPRXY # ndd -set /dev/tcp tcp_recv_hiwat 65536
```

Antes de ejecutar el procedimiento de instalación y configuración de esta capa, se respaldó debidamente la configuración de los 2 servidores de directorio, así como los datos de los 2 sufijos. Así que se hizo un *login* en uno de los servidores de directorio y se ejecuto el siguiente comando para validar los sufijos existentes:

```
cpprssc01.banco.corp # dsconf list-suffixes -h  -p 389 -D cn=root -w
/tmp/.pwd

o=InstitucionBancaria
```

Iniciando con el proceso de instalación y configuración del software de *LDAP Proxy*, se copio el paquete a los servidores de *LDAP Proxy*, y a continuación se descomprime, como se muestra en seguida:

```
root@CPPRSAPRXY # unzip sun-dsee7.zip -d /var/opt/mps/dps/dsee7.0

root@CPPRSAPRXY # ./dpadm create –p 389 /var/opt/mps/dps/dsproxy-banco-prov

root@CPPRSAPRXY # ./dpadm info /var/opt/mps/dps/dsproxy-banco-prov
Instance Path: /var/opt/mps/dps/dsproxy-banco-prov
Install Path: /var/opt/mps/dps/dps7
Owner: root(root)
Non-secure port: 389
Secure port: 636
State: stopped
DSCC url: -
SMF application: -
Instance tag: P-A00
Java command: /var/opt/mps/dps/jre/bin/java

root@CPPRSAPRXY # ./dpadm get-flags /var/opt/mps/dps/dsproxy-banco-prov
cert-pwd-prompt:off
jvm-args:-Xmx4096M -Xms4096M
jvm-path:
umask:0
```

El siguiente paso fue inicializar los procesos de la instancia de *proxy* de directorio a través del siguiente comando:

```
root@CPPRSAPRXY # ./dpadm start /var/opt/mps/dps/dsproxy-banco-prov

Directory Proxy Server instance '/var/opt/mps/dps/dsproxy-banco-prov'
started: pid=23400

root@CPPRSAPRXY # pgrep -lf dps

23400 /var/opt/mps/dps/dsee7.0/dsee7/jre/bin/sparcv9/java -Xmx4096M -
Xms4096M –server
```

Posteriormente se verificó la versión del producto recién instalado que estaba corriendo mediante los siguientes comandos:

```
root@CPPRSAPRXY # ./dpadm -V
[dpadm]
dpadm : 11.1.1.3.0 B2010.0630.2203 ZIP
```

```
Copyright (c) 2010, Oracle and/or its affiliates. All rights reserved.

[DPS]
Sun Microsystems, Inc.
Sun-Directory-Proxy-Server/11.1.1.3.0 B2010.0630.2146
```

Cabe mencionar que en la salida anterior se puede apreciar el mensaje donde se indicaba que la distribución usada en esta versión estaba en la modalidad de *ZIP Distribution*, para la cuál no fue necesario instalar los paquetes uno a uno. El procedimiento consiste únicamente en descomprimir el paquete y empezar a ejecutar los comandos de configuración y administración.

```
root@CPPRSAPRXY # ./dpconf -V
[dpconf]
dpconf : 11.1.1.3.0 B2010.0630.2145

Copyright (c) 2010, Oracle and/or its affiliates. All rights reserved.
```

En seguida se verificó que no hubiera mensajes referentes a errores en el archivo *errors*, y se observó la siguiente salida:

```
[01/Jun/2011:23:53:51 -0430] - STARTUP - INFO - Logging Service configured
[01/Jun/2011:23:53:51 -0430] - STARTUP - INFO - Java Version: 1.6.0_20 (Java
Home: /var/opt/mps/dps/dsee7.0/dsee7/jre)
[01/Jun/2011:23:53:51 -0430] - STARTUP - INFO - Java(TM) SE Runtime
Environment (build 1.6.0_20-b02)
[01/Jun/2011:23:53:51 -0430] - STARTUP - INFO - Java HotSpot(TM) 64-Bit
Server VM (build 16.3-b01, mixed mode)
[01/Jun/2011:23:53:51 -0430] - STARTUP - INFO - Java Heap Space: Total
Memory (-Xms) = 3929MB, Max Memory (-Xmx) = 3929MB
[01/Jun/2011:23:53:51 -0430] - STARTUP - INFO - Operating System:
SunOS/sparcv9 5.10
[01/Jun/2011:23:53:51 -0430] - STARTUP - INFO - Initializing LDAP server
cn=dsve1,cn=data sources,cn=config
[01/Jun/2011:23:53:51 -0430] - STARTUP - INFO - Initializing LDAP server
cn=dsve2,cn=data sources,cn=config
[01/Jun/2011:23:53:51 -0430] - STARTUP - INFO - SSL initialization succeeded.
[01/Jun/2011:23:53:52 -0430] - CONFIG - WARN - Attribute
certMappingDataViewPolicy in entry cn=LDAPS Listener,cn=Client
Listeners,cn=config missing. Using ALL_DATA
_VIEW
[01/Jun/2011:23:53:52 -0430] - STARTUP - INFO - Creating 50 worker threads.
[01/Jun/2011:23:53:52 -0430] - STARTUP - INFO - Sun-Directory-Proxy-
Server/11.1.1.3.0 B2010.0630.2146 started on host UNKNOWN in directory
/var/opt/mps/dps/dsproxy-
```

La salida anterior muestra que en efecto no hubo errores relacionados con el proceso de instalación.

Posteriormente se prosiguió con el proceso de creación y configuración de los 3 *connection handlers*, uno para el balanceador, uno para los servidores de *WAS*, mejor conocido como *Websphere Application Server* y uno para los servidores de *WebSEAL*, como se muestra a continuación:

```
root@CPPRSAPRXY # ./dpconf create-connection-handler –h hostname –p 389
connection-handler-name
```

Una vez que fueron creados los *connection handlers,* se validó el estado de los mismos mediante el siguiente comando:

```
root@CPPRSAPRXY # ./dpconf list-connection-handlers -h hostname -p 389 -v -D
cn=root -w /tmp/.pwd

NAME                is-enabled   priority                  description
```

```
---------------------------------- ----------   --------                    ---
------------------------------
anonymous                    false        99           unauthenticated
connections
cnhandlerBAL                 true         2            -
cnhandlerVE                  true         3            -
cnhandlerWSVE                true         4            -
default connection handler true           100          default connection
handler
directory services administrators true    1            Administrators
connection handler
```

El siguiente paso fue ajustar las propiedades de cada uno de los *Connection Handler* creados, para lo cuál se usaron los siguientes comandos:

```
root@CPPRSAPRXY # ./dpconf get-connection-handler-prop -h hostname -p 389 -v
-D cn=root -w /tmp/.pwd cnhandlerBAL

aci-source                                : none
allowed-auth-methods                      : anonymous
allowed-auth-methods                      : sasl
allowed-auth-methods                      : simple
allowed-ldap-ports                        : ldap
allowed-ldap-ports                        : ldaps
bind-dn-filters                           : any
close-client-connection                   : false
data-view-routing-custom-list             : none
data-view-routing-policy                  : all-routable
data-view-use-internal-client-identity    : false
description                               : -
domain-name-filters                       : any
enable-data-view-affinity                 : false
group-dn-filters                          : any
group-search-bind-dn                      : any
group-search-bind-pwd                     : none
ip-address-filters                        : 10.10.10.1
is-enabled                                : true
is-ssl-mandatory                          : false
priority                                  : 2
request-filtering-policy                  : RFPcnhandlerBAL
require-data-view-availability            : true
resource-limits-policy                    : RLPcnhandlerBAL
schema-check-enabled                      : false
user-filter                               : any
```

The "get-connection-handler-prop" operation succeeded on "172.10.10.1:389".

```
root@CPPRSAPRXY # ./dpconf get-connection-handler-prop -h hostname -p 389 -v
-D cn=root -w /tmp/.pwd cnhandlerVE

aci-source                                : none
allowed-auth-methods                      : anonymous
allowed-auth-methods                      : sasl
allowed-auth-methods                      : simple
allowed-ldap-ports                        : ldap
allowed-ldap-ports                        : ldaps
bind-dn-filters                           : ou=banco, c=ve,
o=InstitucionBancaria
close-client-connection                   : false
data-view-routing-custom-list             : vwAdmin
data-view-routing-custom-list             : vwVEsecAuth
data-view-routing-custom-list             : vwVEuserRoot
data-view-routing-policy                  : custom
data-view-use-internal-client-identity    : false
description                               : -
domain-name-filters                       : any
enable-data-view-affinity                 : false
group-dn-filters                          : any
```

```
group-search-bind-dn                    : any
group-search-bind-pwd                   : none
ip-address-filters                      : 172.16.25.3
ip-address-filters                      : 172.16.25.4
ip-address-filters                      : 172.16.25.5
ip-address-filters                      : 172.16.25.6
ip-address-filters                      : 172.16.25.8
is-enabled                              : true
is-ssl-mandatory                        : false
priority                                : 3
request-filtering-policy                : no-filtering
require-data-view-availability          : true
resource-limits-policy                  : RLPcnhandlerVE
schema-check-enabled                    : false
user-filter                             : any

The "get-connection-handler-prop" operation succeeded on "172.10.10.2:389".
```

```
root@CPPRSAPRXY # ./dpconf get-connection-handler-prop -h hostname -p 389 -v
-D cn=root -w /tmp/.pwd cnhandlerWSVE
```

```
aci-source                              : none
allowed-auth-methods                    : anonymous
allowed-auth-methods                    : sasl
allowed-auth-methods                    : simple
allowed-ldap-ports                      : ldap
allowed-ldap-ports                      : ldaps
bind-dn-filters                         : cn=.*,secAuthority=Default
close-client-connection                 : false
data-view-routing-custom-list           : vwVEsecAuth
data-view-routing-custom-list           : vwVEuserRoot
data-view-routing-policy                : custom
data-view-use-internal-client-identity  : false
description                             : -
domain-name-filters                     : any
enable-data-view-affinity               : false
group-dn-filters                        : any
group-search-bind-dn                    : any
group-search-bind-pwd                   : none
ip-address-filters                      : 10.10.15.21
ip-address-filters                      : 10.10.15.22
ip-address-filters                      : 10.10.15.24
ip-address-filters                      : 10.10.15.26
ip-address-filters                      : 10.10.17.10
ip-address-filters                      : 10.10.17.11
ip-address-filters                      : 10.10.17.12
ip-address-filters                      : 10.10.17.13
ip-address-filters                      : 172.16.25.17
ip-address-filters                      : 172.16.25.18
ip-address-filters                      : 172.16.25.76
is-enabled                              : true
is-ssl-mandatory                        : false
priority                                : 4
request-filtering-policy                : no-filtering
require-data-view-availability          : true
resource-limits-policy                  : no-limits
schema-check-enabled                    : false
user-filter                             : any

The "get-connection-handler-prop" operation succeeded on "172.10.10.3:389".
```

Posteriormente se validó que los servidores de directorio estuvieran dados de alta como fuentes de datos legítimas, mediante la siguiente instrucción:

```
root@CPPRSAPRXY # ./dpconf list-ldap-data-sources -h hostname -p 389 -v -D
cn=root -w /tmp/.pwd
```

```
SRC_NAME     is-enabled   ldap-address  ldap-port    ldaps-port description
--------     ----------   ------------  --------- ----------      ---------
```

```
dsve1        true         172.16.25.75 ldap         ldaps        -
dsve2        true         172.16.25.73 ldap         ldaps        -

The "list-ldap-data-sources" operation succeeded on "172.16.25.76:389"
```

Posteriormente se verificaron las propiedades de cada una de las 2 fuentes de datos, mediante los siguientes comandos:

```
root@CPPRSAPRXY # ./dpconf get-ldap-data-source-prop -h 172.16.25.76 -p 389
-v -D cn=root -w /tmp/.pwd dsve1

bind-dn :                          none
bind-pwd :                         none
client-cred-mode :                 use-client-identity
connect-timeout :                  10s
description:                       -
down-monitoring-interval :         inherited
is-enabled :                       true
is-read-only :                     false
ldap-address :                     172.16.25.75
ldap-port :                        ldap
ldaps-port :                       ldaps
monitoring-bind-dn :               none
monitoring-bind-pwd :              none
monitoring-bind-timeout :          5s
monitoring-entry-dn :              ""
monitoring-entry-timeout :         5s
monitoring-inactivity-timeout :    2m
monitoring-interval :              30s
monitoring-mode :                  proactive
monitoring-retry-count :           3
monitoring-search-filter :         (|(objectClass=*)(objectClass=ldapSubEntry))
num-bind-incr :                    10
num-bind-init :                    2
num-bind-limit :                   unlimited
num-read-incr :                    10
num-read-init :                    2
num-read-limit :                   unlimited
num-write-incr :                   10
num-write-init :                   2
num-write-limit :                  unlimited
proxied-auth-use-v1 :              false
ssl-policy :                       never
use-read-connections-for-writes :  false
use-tcp-keep-alive :               true
use-tcp-no-delay :                 true

The "get-ldap-data-source-prop" operation succeeded on "172.16.25.76:389".
```

```
root@CPPRSAPRXY # ./dpconf get-ldap-data-source-prop -h 172.16.25.76 -p 389
-v -D cn=root -w /tmp/.pwd dsve2

bind-dn :                          none
bind-pwd :                         none
client-cred-mode :                 use-client-identity
connect-timeout :                  10s
description :                      -
down-monitoring-interval :         inherited
is-enabled :                       true
is-read-only :                     false
ldap-address :                     172.16.25.73
ldap-port :                        ldap
ldaps-port :                       ldaps
monitoring-bind-dn :               none
monitoring-bind-pwd :              none
monitoring-bind-timeout :          5s
monitoring-entry-dn :              ""
monitoring-entry-timeout :         5s
monitoring-inactivity-timeout :    2m
```

```
monitoring-interval :            30s
monitoring-mode :                proactive
monitoring-retry-count :         3
monitoring-search-filter :       (|(objectClass=*)(objectClass=ldapSubEntry))
num-bind-incr :                  10
num-bind-init :                  2
num-bind-limit :                 unlimited
num-read-incr :                  10
num-read-init :                  2
num-read-limit :                 unlimited
num-write-incr :                 10
num-write-init :                 2
num-write-limit :                unlimited
proxied-auth-use-v1 :            false
ssl-policy :                     never
use-read-connections-for-writes :false
use-tcp-keep-alive :             true
use-tcp-no-delay :               true

The "get-ldap-data-source-prop" operation succeeded on "172.16.25.76:389"
```

El siguiente paso fue verificar los "*data source pools"* mediante la siguiente instrucción:

```
root@CPPRSAPRXY # ./dpconf list-ldap-data-source-pools -h 172.16.25.76 -p
389 -v -D cn=root -w /tmp/.pwd

NAME                     load-balancing-algorithm         description
----------------------   ------------------------         ----------------
defaultDataSourcePool    proportional                     data source pool
example
poolVE                   failover                         -

The "list-ldap-data-source-pools" operation succeeded on "172.16.25.76:389".
```

De la salida anterior se observa claramente que el "*data source pool*" que se encontraba activo para los 2 servidores de *LDAP Proxy* es el *PoolVE*. Y en seguida se listan sus propiedades que fueron verificadas:

```
root@CPPRSAPRXY # ./dpconf get-ldap-data-source-pool-prop -h 172.16.25.76 -p
389 -v -D cn=root -w /tmp/.pwd poolVE

client-affinity-bind-dn-filters :            any
client-affinity-criteria :                   connection
client-affinity-ip-address-filters :         any
client-affinity-policy :                     read-write-affinity-after-
write
client-affinity-timeout :                    20s
description :                                -
enable-client-affinity :                     true
load-balancing-algorithm :                   failover
```

Como parte del proceso de implementación del dominio seguro el siguiente paso fue proceder con la instalación y configuración de los componentes de acceso web, *Tivoli Access Manager,* lo cuál implicaba una actualización del esquema para soportar los nuevos *objectclass* que vienen definidos en *TAM 6.1.1*. Para hacerlo se utilizó la siguiente instrucción:

```
root@cpprssc01.banco.corp # ./ivrgy_tool -d -h 172.16.25.52 -p 389 -D
cn=root -w xxxxxxxxxxxxxx schema
ivrgy_tool: Attempting to add schema.
ivrgy_tool: IRA interface reports result (x'0'):
Request was successful.
```

A continuación el proceso de configuración para el *Runtime Environment*:

```
root@pd-banco-prov # pdconfig

Tivoli Access Manager Setup Menu

1. Configure Package
2. Unconfigure Package
3. Display Configuration Status
x. Exit

Select the menu item [x]: 1

Tivoli Access Manager Configuration Menu


1. Access Manager Runtime Configuration
2. Access Manager Policy Server Configuration
3. Access Manager Authorization Server Configuration
4. Access Manager Runtime for Java Configuration
x. Return to the Tivoli Access Manager Setup Menu


Select the menu item [x]: 1
Tivoli Common Directory logging is not configured.
This scheme provides a common location for log files
for Tivoli products instead of separate locations
determined by each application.



Do you want to use Tivoli Common Directory logging (y/n) [No]?

Log files for this application will be created in directory:
/var/PolicyDirector/log


1. LDAP

Registry [1]:

LDAP server host name: 172.16.25.52

LDAP server port [389]:

The package has been configured successfully.


Press Enter to continue.


Tivoli Access Manager Configuration Menu


1. Access Manager Policy Server Configuration
2. Access Manager Authorization Server Configuration
3. Access Manager Runtime for Java Configuration
x. Return to the Tivoli Access Manager Setup Menu


Select the menu item [x]:
```

A continuación se configuró el *Policy Director server*:

```
root@pd-banco-prov # pdconfig

Tivoli Access Manager Setup Menu

1. Configure Package
2. Unconfigure Package
```

```
3. Display Configuration Status
x. Exit

Select the menu item [x]: 1

Tivoli Access Manager Configuration Menu


1. Access Manager Policy Server Configuration
2. Access Manager Authorization Server Configuration
3. Access Manager Runtime for Java Configuration
x. Return to the Tivoli Access Manager Setup Menu


Select the menu item [x]: 1

LDAP administrator ID [cn=root]:
LDAP administrator password:

Management domain name [Default]:


The LDAP management domain location DN is the location in the LDAP
server where the management domain information is stored. If the
LDAP management domain location DN is not specified, the
management domain information is stored in its own suffix by
default. Whether the DN is specified or the default is used,
the location must already exist in the LDAP server.


LDAP management domain location DN []:

Enable SSL between the Tivoli Access Manager policy server
and the LDAP server (y/n) [Yes]? n


A policy server is already configured to this LDAP server. A second
policy server may be configured for migration or standby purposes ONLY!
Would you like to configure a second policy server to this
LDAP server (y/n) [No]? y

Use this policy server for standby (y/n) [No]:
Specify the directory where the extracted pdbackup files reside:
/opt/PolicyDirector

* Configuring the server.

The SSL configuration of Access Control Runtime has completed successfully.
Tivoli Access Manager policy server domain name: Default
Tivoli Access Manager policy server host name: cpprssc01
Tivoli Access Manager policy server listening port: 7135


* Starting the server.


The server has been started.

The package has been configured successfully.


Press Enter to continue.
```

El proceso anterior se había ejecutado de manera exitosa, así que en seguida se validó el estado de los procesos de la instancia de *Policy Director* y se hizo una prueba para verificar que se podían listar las listas de control de acceso, usuarios, y grupos.

```
root@pd-banco-prov # pd_start status

Tivoli Access Manager servers

Server        Enabled      Running

------------------------------------------

pdmgrd        yes          no
pdacld        yes          no
pdmgrproxyd   no           no


root@pd-banco-prov # pdadmin -a sec_master

Enter Password:
pdadmin sec_master> acl list
default-webseal
BNET_VE
default-management-proxy
default-management
Privado_ve
default-root
RECURSOS_PUBLICOS
default-gso
default-policy
favicon
default-config
default-domain
Publico
default-replica
pdadmin sec_master> exit
```

El siguiente paso como acción preventiva fue copiar los archivos de la base de datos de la *master authorization* hacia un equipo remoto antes de proceder con la instalación de *WebSEAL*, así que en seguida se muestra el proceso que se siguio para ejecutar dicha actividad:

```
root@pd-banco-prov # ls -l /var/PolicyDirector/db/

total 2272
-rw------- 1 ivmgr ivmgr 573440 May 19 00:45 ivacld.db
-rw------- 1 ivmgr ivmgr 573440 May 19 00:45 master_authzn.db

root@pd-banco-prov # scp *.db root@172.16.25.52:/var/PolicyDirector/db
```

Es importante mencionar que con la ejecución del punto anterior, el nuevo servidor de *Policy Director* estaba completamente instalado y configurado y todos los servidores de *WebSEAL* serían apuntados a dicho *PD*.

Como primer paso antes de proceder con la instalación del componente de acceso web en cada uno de los 4 servidores considerados para dicha finalidad, se validó que la versión de *Linux SuSE Enterprise Edition* se encontrara en la versión mínima requerida.

```
<servidor>:~ # more /etc/SuSE-release

SUSE LINUX Enterprise Server 10 (i586)
VERSION = 10
PATCHLEVEL = 1

<servidor>:~ # uname -a

Linux server-name 2.6.5-7.244-smp #1 SMP Mon Dec 12 18:32:25 UTC 2005 i686
i686 i386 GNU/Linux
```

Posteriormente se verificó el valor de los parámetros de sistema operativo que se relacionan la cantidad de archivos que se podían abrir de manera simultánea mediante el siguiente comando:

```
<servidor>:~ # ulimit -a

core file size          (blocks, -c) 0
data seg size           (kbytes, -d) unlimited
file size               (blocks, -f) unlimited
max locked memory       (kbytes, -l) unlimited
max memory size         (kbytes, -m) unlimited
open files              (-n)         1024
pipe size               (512 bytes, -p)     8
stack size              (kbytes, -s) unlimited
cpu time                (seconds, -t)unlimited
max user processes      (-u)         16381
virtual memory          (kbytes, -v) unlimited
```

La distribución de los *filesystems* fue la misma para los 4 servidores de *WebSEAL*, y a continuación se presenta una tabla con la distribución de *filesystems*, tipo *ext3*:

Filesystem	Tamaño	Usado	Disponible	Capacidad	Montaje
/dev/sda1	20G	8.2G	11G	44.00%	/
/dev/sda2	32G				swap
tmpfs	1013M	8.0K	1013M	1.00%	/dev/shm
/dev/sda3	9.9G	35M	9.4G	1.00%	/home
/dev/sdb1	3.0G	46M	2.8G	2.00%	/opt/pdweb
/dev/sdb2	5.0G	318M	4.4G	7.00%	/var
/dev/sdb3	16G	1.9G	14G	13.00%	/var/pdweb

Tabla A.2 Distribución de *filesystems* para la capa del componente de control de accesos

A continuación se presenta la salida que se utilizó para verificar el *layout* de los discos físicos internos de cada uno de los 4 servidores.

```
<servidor>:/etc # fdisk -l

Disk /dev/sda: 36.4 GB, 36401479680 bytes
255 heads, 63 sectors/track, 4425 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes

Device Boot   Start  End    Blocks Id     System
/dev/sda1 *   1      2611   20972826 83   Linux
/dev/sda2     2612   3003   3148740         82  Linux swap
/dev/sda3     3004   4309   10490445 83   Linux

Disk /dev/sdb: 36.4 GB, 36401479680 bytes
255 heads, 63 sectors/track, 4425 cylinders
Units = cylinders of 16065 * 512 = 8225280 bytes

Device Boot   Start  End    Blocks Id     System
/dev/sdb1     1      392    3148708+ 83   Linux
/dev/sdb2     393    1045   5245222+ 83   Linux
/dev/sdb3     1046   3134   16779892+ 83  Linux
```

El siguiente paso fue verificar toda la configuración de sistema operativo a nivel de red, con el objeto de asegurar que la plataforma que sería utilizada para la capa de

WebSEAL cumpliera con toda la conectividad requerida. Particularmente para esta parte relacionada con la red, por motivos de confidencialidad unicamente se mostraran los comandos ejecutados sin la respectiva salida de cada uno de ellos. El primer punto fué validar las direcciones *IP*, así como máscaras de red, y direcciones de *broadcast* que tenía asignadas cada equipo, y se hizo a través del siguiente comando:

```
<servidor>:/etc # ifconfig -a
```

En seguida se validaron las rutas estáticas definidas en cada uno de los servidores de *WebSEAL,* como se presenta a continuación:

```
<servidor>:~ # netstat -nr
```

Como último paso de verificación del ambiente destinado para *WebSEAL*, se revisó la configuración de los parámetros del stack de *TCP/IP* para los 4 servidores considerados, tal y como se muestra a continuación:

```
<servidor>:~ # sysctl -a|grep -i ipv4 |more

error: "Invalid argument" reading key "net.ipv6.route.flush"
error: "Invalid argument" reading key "net.ipv4.route.flush"
net.ipv4.conf.eth0.force_igmp_version = 0
net.ipv4.conf.eth0.disable_policy = 0
net.ipv4.conf.eth0.disable_xfrm = 0
net.ipv4.conf.eth0.arp_ignore = 0
net.ipv4.conf.eth0.arp_announce = 0
net.ipv4.conf.eth0.arp_filter = 0
net.ipv4.conf.eth0.tag = 0
net.ipv4.conf.eth0.log_martians = 0
net.ipv4.conf.eth0.bootp_relay = 0
net.ipv4.conf.eth0.medium_id = 0
net.ipv4.conf.eth0.proxy_arp = 0
net.ipv4.conf.eth0.accept_source_route = 1
net.ipv4.conf.eth0.send_redirects = 1
net.ipv4.conf.eth0.rp_filter = 0
net.ipv4.conf.eth0.shared_media = 1
net.ipv4.conf.eth0.secure_redirects = 1
net.ipv4.conf.eth0.accept_redirects = 1
net.ipv4.conf.eth0.mc_forwarding = 0
net.ipv4.conf.eth0.forwarding = 0
net.ipv4.conf.eth1.force_igmp_version = 0
net.ipv4.conf.eth1.disable_policy = 0
net.ipv4.conf.eth1.disable_xfrm = 0
net.ipv4.conf.eth1.arp_ignore = 0
net.ipv4.conf.eth1.arp_announce = 0
net.ipv4.conf.eth1.arp_filter = 0
net.ipv4.conf.eth1.tag = 0
net.ipv4.conf.eth1.log_martians = 0
net.ipv4.conf.eth1.bootp_relay = 0
net.ipv4.conf.eth1.medium_id = 0
net.ipv4.conf.eth1.proxy_arp = 0
net.ipv4.conf.eth1.accept_source_route = 1
net.ipv4.conf.eth1.send_redirects = 1
net.ipv4.conf.eth1.rp_filter = 0
net.ipv4.conf.eth1.shared_media = 1
net.ipv4.conf.eth1.secure_redirects = 1
net.ipv4.conf.eth1.accept_redirects = 1
net.ipv4.conf.eth1.mc_forwarding = 0
net.ipv4.conf.eth1.forwarding = 0
net.ipv4.conf.lo.force_igmp_version = 0
net.ipv4.conf.lo.disable_policy = 0
net.ipv4.conf.lo.disable_xfrm = 0
net.ipv4.conf.lo.arp_ignore = 0
net.ipv4.conf.lo.arp_announce = 0
```

```
net.ipv4.conf.lo.arp_filter = 0
net.ipv4.conf.lo.tag = 0
net.ipv4.conf.lo.log_martians = 0
net.ipv4.conf.lo.bootp_relay = 0
net.ipv4.conf.lo.medium_id = 0
net.ipv4.conf.lo.proxy_arp = 0
net.ipv4.conf.lo.accept_source_route = 1
net.ipv4.conf.lo.send_redirects = 1
net.ipv4.conf.lo.rp_filter = 0
net.ipv4.conf.lo.shared_media = 1
net.ipv4.conf.lo.secure_redirects = 1
net.ipv4.conf.lo.accept_redirects = 1
net.ipv4.conf.lo.mc_forwarding = 0
net.ipv4.conf.lo.forwarding = 0
net.ipv4.conf.default.force_igmp_version = 0
net.ipv4.conf.default.disable_policy = 0
net.ipv4.conf.default.disable_xfrm = 0
net.ipv4.conf.default.arp_ignore = 0
net.ipv4.conf.default.arp_announce = 0
net.ipv4.conf.default.arp_filter = 0
net.ipv4.conf.default.tag = 0
net.ipv4.conf.default.log_martians = 0
net.ipv4.conf.default.bootp_relay = 0
net.ipv4.conf.default.medium_id = 0
net.ipv4.conf.default.proxy_arp = 0
net.ipv4.conf.default.accept_source_route = 1
net.ipv4.conf.default.send_redirects = 1
net.ipv4.conf.default.rp_filter = 0
net.ipv4.conf.default.shared_media = 1
net.ipv4.conf.default.secure_redirects = 1
net.ipv4.conf.default.accept_redirects = 1
net.ipv4.conf.default.mc_forwarding = 0
net.ipv4.conf.default.forwarding = 0
net.ipv4.conf.all.force_igmp_version = 0
net.ipv4.conf.all.disable_policy = 0
net.ipv4.conf.all.disable_xfrm = 0
net.ipv4.conf.all.arp_ignore = 0
net.ipv4.conf.all.arp_announce = 0
net.ipv4.conf.all.arp_filter = 0
net.ipv4.conf.all.tag = 0
net.ipv4.conf.all.log_martians = 1
net.ipv4.conf.all.bootp_relay = 0
net.ipv4.conf.all.medium_id = 0
net.ipv4.conf.all.proxy_arp = 0
net.ipv4.conf.all.accept_source_route = 0
net.ipv4.conf.all.send_redirects = 1
net.ipv4.conf.all.rp_filter = 1
net.ipv4.conf.all.shared_media = 1
net.ipv4.conf.all.secure_redirects = 1
net.ipv4.conf.all.accept_redirects = 1
net.ipv4.conf.all.mc_forwarding = 0
net.ipv4.conf.all.forwarding = 0
net.ipv4.neigh.eth0.locktime = 100
net.ipv4.neigh.eth0.proxy_delay = 80
net.ipv4.neigh.eth0.anycast_delay = 100
net.ipv4.neigh.eth0.proxy_qlen = 64
net.ipv4.neigh.eth0.unres_qlen = 3
net.ipv4.neigh.eth0.gc_stale_time = 60
net.ipv4.neigh.eth0.delay_first_probe_time = 5
net.ipv4.neigh.eth0.base_reachable_time = 30
net.ipv4.neigh.eth0.retrans_time = 100
net.ipv4.neigh.eth0.app_solicit = 0
net.ipv4.neigh.eth0.ucast_solicit = 3
net.ipv4.neigh.eth0.mcast_solicit = 3
net.ipv4.neigh.eth1.locktime = 100
net.ipv4.neigh.eth1.proxy_delay = 80
net.ipv4.neigh.eth1.anycast_delay = 100
net.ipv4.neigh.eth1.proxy_qlen = 64
net.ipv4.neigh.eth1.unres_qlen = 3
net.ipv4.neigh.eth1.gc_stale_time = 60
```

```
net.ipv4.neigh.eth1.delay_first_probe_time = 5
net.ipv4.neigh.eth1.base_reachable_time = 30
net.ipv4.neigh.eth1.retrans_time = 100
net.ipv4.neigh.eth1.app_solicit = 0
net.ipv4.neigh.eth1.ucast_solicit = 3
net.ipv4.neigh.eth1.mcast_solicit = 3
net.ipv4.neigh.lo.locktime = 100
net.ipv4.neigh.lo.proxy_delay = 80
net.ipv4.neigh.lo.anycast_delay = 100
net.ipv4.neigh.lo.proxy_qlen = 64
net.ipv4.neigh.lo.unres_qlen = 3
net.ipv4.neigh.lo.gc_stale_time = 60
net.ipv4.neigh.lo.delay_first_probe_time = 5
net.ipv4.neigh.lo.base_reachable_time = 30
net.ipv4.neigh.lo.retrans_time = 100
net.ipv4.neigh.lo.app_solicit = 0
net.ipv4.neigh.lo.ucast_solicit = 3
net.ipv4.neigh.lo.mcast_solicit = 3
net.ipv4.neigh.default.gc_thresh3 = 1024
net.ipv4.neigh.default.gc_thresh2 = 512
net.ipv4.neigh.default.gc_thresh1 = 128
net.ipv4.neigh.default.gc_interval = 30
net.ipv4.neigh.default.locktime = 100
net.ipv4.neigh.default.proxy_delay = 80
net.ipv4.neigh.default.anycast_delay = 100
net.ipv4.neigh.default.proxy_qlen = 64
net.ipv4.neigh.default.unres_qlen = 3
net.ipv4.neigh.default.gc_stale_time = 60
net.ipv4.neigh.default.delay_first_probe_time = 5
net.ipv4.neigh.default.base_reachable_time = 30
net.ipv4.neigh.default.retrans_time = 100
net.ipv4.neigh.default.app_solicit = 0
net.ipv4.neigh.default.ucast_solicit = 3
net.ipv4.neigh.default.mcast_solicit = 3
net.ipv4.tcp_westwood = 0
net.ipv4.ipfrag_max_dist = 64
net.ipv4.ipfrag_secret_interval = 600
net.ipv4.tcp_low_latency = 0
net.ipv4.tcp_frto = 0
net.ipv4.tcp_tw_reuse = 0
net.ipv4.icmp_ratemask = 6168
net.ipv4.icmp_ratelimit = 1000
net.ipv4.tcp_adv_win_scale = 2
net.ipv4.tcp_app_win = 31
net.ipv4.tcp_rmem = 4096 87380 174760
net.ipv4.tcp_wmem = 4096 16384 131072
net.ipv4.tcp_mem = 195584 196096 196608
net.ipv4.tcp_dsack = 1
net.ipv4.tcp_ecn = 0
net.ipv4.tcp_reordering = 3
net.ipv4.tcp_fack = 1
net.ipv4.tcp_orphan_retries = 0
net.ipv4.inet_peer_gc_maxtime = 120
net.ipv4.inet_peer_gc_mintime = 10
net.ipv4.inet_peer_maxttl = 600
net.ipv4.inet_peer_minttl = 120
net.ipv4.inet_peer_threshold = 65664
net.ipv4.igmp_max_msf = 10
net.ipv4.igmp_max_memberships = 20
net.ipv4.route.secret_interval = 600
net.ipv4.route.min_adv_mss = 256
net.ipv4.route.min_pmtu = 552
net.ipv4.route.mtu_expires = 600
net.ipv4.route.gc_elasticity = 8
net.ipv4.route.error_burst = 5000
net.ipv4.route.error_cost = 1000
net.ipv4.route.redirect_silence = 20480
net.ipv4.route.redirect_number = 9
net.ipv4.route.redirect_load = 20
net.ipv4.route.gc_interval = 60
```

```
net.ipv4.route.gc_timeout = 300
net.ipv4.route.gc_min_interval = 0
net.ipv4.route.max_size = 262144
net.ipv4.route.gc_thresh = 16384
net.ipv4.route.max_delay = 10
net.ipv4.route.min_delay = 2
net.ipv4.icmp_ignore_bogus_error_responses = 1
net.ipv4.icmp_echo_ignore_broadcasts = 1
net.ipv4.icmp_echo_ignore_all = 1
net.ipv4.ip_local_port_range = 32768 61000
net.ipv4.tcp_max_syn_backlog = 1024
net.ipv4.tcp_rfc1337 = 0
net.ipv4.tcp_stdurg = 0
net.ipv4.tcp_abort_on_overflow = 0
net.ipv4.tcp_tw_recycle = 0
net.ipv4.tcp_syncookies = 1
net.ipv4.tcp_fin_timeout = 60
net.ipv4.tcp_retries2 = 15
net.ipv4.tcp_retries1 = 3
net.ipv4.tcp_keepalive_intvl = 75
net.ipv4.tcp_keepalive_probes = 9
net.ipv4.tcp_keepalive_time = 7200
net.ipv4.ipfrag_time = 30
net.ipv4.ip_dynaddr = 0
net.ipv4.ipfrag_low_thresh = 196608
net.ipv4.ipfrag_high_thresh = 262144
net.ipv4.tcp_max_tw_buckets = 180000
net.ipv4.tcp_max_orphans = 32768
net.ipv4.tcp_synack_retries = 5
net.ipv4.tcp_syn_retries = 5
net.ipv4.ip_nonlocal_bind = 0
net.ipv4.ip_no_pmtu_disc = 0
net.ipv4.ip_autoconfig = 0
net.ipv4.ip_default_ttl = 64
net.ipv4.ip_forward = 0
net.ipv4.tcp_retrans_collapse = 1
net.ipv4.tcp_sack = 1
net.ipv4.tcp_window_scaling = 1
net.ipv4.tcp_timestamps = 1
```

En seguida se aseguró que el archivo */etc/hosts* contenía la entrada de los nombres de servidores de aplicaciones *Websphere Application Server,* mejor conocidos como *WAS* donde se alojaban las aplicaciones *web* a las que harían referencia las *junctions* definidas dentro de las instancias de *WebSEAL*.

A continuación, el siguiente paso fue proceder con la instalación y configuración de paquetes como se muestra a continuación:

```
<Server-name>:~ # rpm -Uvh gsk7bas-7.0-4.28.i386.rpm
Preparing... ########################################### [100%]
1:gsk7bas ########################################### [100%]

<Server-name>:~ # rpm -ivh idsldap-cltbase61-6.1.0-6.i386.rpm
Preparing... ########################################### [100%]
1:idsldap-cltbase61########################################### [100%]

<Server-name>:~ # rpm -ivh idsldap-clt32bit61-6.1.0-6.i386.rpm
Preparing... ########################################### [100%]
1:idsldap-clt32bit61 ########################################### [100%]

<Server-name>:~ # rpm -ivh TivSecUtl-TivSec-6.1.1-0.i386.rpm
Preparing... ########################################### [100%]
1:TivSecUtl-TivSec ########################################### [100%]

<Server-name>:~ # rpm -ivh Pdlic-PD-6.1.1-0.i386.rpm
Preparing... ########################################### [100%]
1:PDlic ########################################### [100%]
```

```
<Server-name>:~ # rpm -ivh PDRTE-PD-6.1.1-0.i386.rpm
Preparing... ########################################### [100%]
1:PDRTE-PD ########################################### [100%]

<Server-name>:~ # rpm -ivh --replacefiles PDWebRTE-PD-6.1.1-0.i386.rpm
Preparing... ########################################### [100%]
1:PDWebRTE-PD ########################################### [100%]

<Server-name>:~ # rpm -Uvh --replacefiles PDWeb-PD-6.1.1-0.i386.rpm
Preparing... ########################################### [100%]
1:PDWeb-PD ########################################### [100%]
```

Una vez que se encuentran instalados los paquetes del software de control de accesos, mejor conocido como *TAM WebSEAL*, se procede a ejecutar el comando que inicia el proceso "*webseald*" en modo *foreground*, tal y como se muestra a continuación:

```
<Server-name>:~ # /opt/pdweb/bin/webseald -config etc/webseald-default.conf
–foreground
Ctrl + D
```

Una vez que los procesos de *WebSEAL* se encontraban en ejecución se cambiaron algunos parámetros directamente en el archivo de configuración de cada una de las instancias, como se muestra en la siguiente tabla donde se indica el valor original y el nuevo valor que fue colocado:

Valor Original	Valor Configurado
dynurl-map=lib/dynurl.conf	# dynurl-map=lib/dynurl.conf
file=cfg(server::dynurl-map)	# file=cfg(server::dynurl-map)
# eai-session-id-header=am-eai-session-id	eai-session-id-header=am-eai-session-id

Tabla A.3 Parámetros requeridos en la configuración del componente de acceso

Una vez hecho lo anterior, se procedió a inicializar los procesos de las instancias de *WebSEAL* por medio del comando que se presenta a continuacion:

```
<Server-name>:~ # pdweb start;pdweb status
```

Finalmente fueron definidas cada una de las *junctions* que serían configuradas para cada una de las instancias de *WebSEAL*. A continuación se presenta la sintáxis del comando, así como los valores definidos para efectuar dicha configuración:

server task default-webseald-<server> create -f -t ssl -c all -h 172.16.25.35 -p 6010 /banco-prov-net1
server task default-webseald-<server> create -f -t ssl -c all -h 172.16.25.35 -p 6025 /banco-prov-net2
server task default-webseald-<server> create -f -t ssl -c all -h 172.16.25.35 -p 6015 /banco-prov-net3
server task default-webseald-<server> create -f -t ssl -c all -h 172.16.25.34 -p 6020 /banco-prov-net4
server task default-webseald-<server> create -f -t ssl -c all -h 172.16.25.35 -p 6033 /banco-prov-net5

Tabla A.4 Definición de *junctions* para cada instancia del componente de acceso de *BNET*

Posteriormente se ejecutó el siguiente comando para verificar el número de *threads* totales que podrían ser ejecutados de forma simultánea en cada una de las instancias de *WebSEAL*:

```
pdadmin sec_master> s t default-webseald-<hostname> stats get pdweb.threads
active :            1
total :             900
'default' active :  1
'default' total :   900
```

La definición de *junctions* se configuró exactamente igual en los 4 servidores de *WebSEAL*. Después de haber ejecutado la secuencia de comandos sobre las instancias de *WebSEAL v6.1.1* para la creación de las respectivas *junctions*, se verificó manualmente desde la consola de administración el listado de propiedades para cada una de las 5 *junctions* definidas como se muestra de manera secuencial a continuación.

Definición de /banco-prov-net1:

```
pdadmin sec_master> s t default-webseald-servidor s /banco-prov-net1
Junction point:                  /banco-prov-net1
Type:                            SSL
Junction hard limit:             0 - using global value
Junction soft limit:             0 - using global value
Active worker threads:           0
Basic authentication mode:       filter
Forms based SSO:                 disabled
TFIM junction SSO:               no
Authentication HTTP header:      insert - iv_user iv_groups iv_creds
Remote Address HTTP header:      do not insert
Stateful junction:               no
Boolean Rule Header:             no
Scripting support:               no
Preserve cookie names:           no
Cookie names include path:       no
Transparent Path junction:       no
Delegation support:              no
Mutually authenticated:          no
Insert WebSphere LTPA cookies:   no
Insert WebSEAL session cookies:  no
Request Encoding:                UTF-8, URI Encoded
Server 1:
ID:                              10af252e-f359-11df-bd17-0011259b6524
Server State:                    running
Operational State:               Online
Hostname:                        172.16.25.35
Port:                            6010
Virtual hostname:                172.16.25.35
Server DN:
local IP address:
Query_contents URL:              /cgi-bin/query_contents
Query-contents:                  unknown
Case insensitive URLs:           no
Allow Windows-style URLs:        yes
Current requests :               0
Total requests :                 1
```

Definición de /banco-prov-net2:

```
pdadmin sec_master> s t default-webseald-servidor s /banco-prov-net2
Junction point:                  /banco-prov-net2
Type:                            SSL
Junction hard limit:             0 - using global value
Junction soft limit:             0 - using global value
Active worker threads:           0
```

```
Basic authentication mode:       filter
Forms based SSO:                 disabled
TFIM junction SSO:               no
Authentication HTTP header:      insert - iv_user iv_groups iv_creds
Remote Address HTTP header:      do not insert
Stateful junction:               no
Boolean Rule Header:             no
Scripting support:               no
Preserve cookie names:           no
Cookie names include path:       no
Transparent Path junction:       no
Delegation support:              no
Mutually authenticated:          no
Insert WebSphere LTPA cookies:   no
Insert WebSEAL session cookies:  no
Request Encoding:                UTF-8, URI Encoded
Server 1:
ID:                              21fda44a-f359-11df-bd17-0011259b6524
Server State:                    running
Operational State:               Online
Hostname:                        172.16.25.35
Port:                            6025
Virtual hostname:                172.16.25.35
Server DN:
local IP address:
Query_contents URL:              /cgi-bin/query_contents
Query-contents:                  unknown
Case insensitive URLs:           no
Allow Windows-style URLs:        yes
Current requests :               0
Total requests :                 1
```

Definición de /banco-prov-net3:

```
pdadmin sec_master> s t default-webseald-servidor s /banco-prov-net3
Junction point:                  /banco-prov-net3
Type:                            SSL
Junction hard limit:             0 - using global value
Junction soft limit:             0 - using global value
Active worker threads:           0
Basic authentication mode:       filter
Forms based SSO:                 disabled
TFIM junction SSO:               no
Authentication HTTP header:      insert - iv_user iv_groups iv_creds
Remote Address HTTP header:      do not insert
Stateful junction:               no
Boolean Rule Header:             no
Scripting support:               no
Preserve cookie names:           no
Cookie names include path:       no
Transparent Path junction:       no
Delegation support:              no
Mutually authenticated:          no
Insert WebSphere LTPA cookies:   no
Insert WebSEAL session cookies:  no
Request Encoding:                UTF-8, URI Encoded
Server 1:
ID:                              2689eece-f359-11df-bd17-0011259b6524
Server State:                    running
Operational State:               Online
Hostname:                        172.16.25.35
Port:                            6015
Virtual hostname:                172.16.25.35
Server DN:
local IP address:
Query_contents URL:              /cgi-bin/query_contents
Query-contents:                  unknown
Case insensitive URLs:           no
Allow Windows-style URLs:        yes
```

```
Current requests :              0
Total requests :                1
```

Definición de /banco-prov-net4:

```
pdadmin sec_master> s t default-webseald-servidor s /banco-prov-net4
Junction point:                 /banco-prov-net4
Type:                           SSL
Junction hard limit:            0 - using global value
Junction soft limit:            0 - using global value
Active worker threads:          0
Basic authentication mode:      filter
Forms based SSO:                disabled
TFIM junction SSO:              no
Authentication HTTP header:     insert - iv_user iv_groups iv_creds
Remote Address HTTP header:     do not insert
Stateful junction:              no
Boolean Rule Header:            no
Scripting support:              no
Preserve cookie names:          no
Cookie names include path:      no
Transparent Path junction:      no
Delegation support:             no
Mutually authenticated:         no
Insert WebSphere LTPA cookies:  no
Insert WebSEAL session cookies: no
Request Encoding:               UTF-8, URI Encoded
Server 1:
ID:                             2b403cfc-f359-11df-bd17-0011259b6524
Server State:                   running
Operational State:              Online
Hostname:                       172.16.25.35
Port:                           6020
Virtual hostname:               172.16.25.35
Server DN:
local IP address:
Query_contents URL:             /cgi-bin/query_contents
Query-contents:                 not found
Case insensitive URLs:          no
Allow Windows-style URLs:       yes
Current requests :              0
Total requests :                12
```

Definición de /banco-prov-net5:

```
pdadmin sec_master> s t default-webseald-servidor s /banco-prov-net5
Junction point:                 /banco-prov-net5
Type:                           SSL
Junction hard limit:            0 - using global value
Junction soft limit:            0 - using global value
Active worker threads:          0
Basic authentication mode:      filter
Forms based SSO:                disabled
TFIM junction SSO:              no
Authentication HTTP header:     insert - iv_user iv_groups iv_creds
Remote Address HTTP header:     do not insert
Stateful junction:              no
Boolean Rule Header:            no
Scripting support:              no
Preserve cookie names:          no
Cookie names include path:      no
Transparent Path junction:      no
Delegation support:             no
Mutually authenticated:         no
Insert WebSphere LTPA cookies:  no
Insert WebSEAL session cookies: no
Request Encoding:               UTF-8, URI Encoded
Server 1:
ID:                             b501832e-f426-11df-be2f-0011259b6524
```

```
Server State:                   running
Operational State:              Online
Hostname:                       172.16.25.35
Port:                           6033
Virtual hostname:               172.16.25.35
Server DN:
local IP address:
Query_contents URL:             /cgi-bin/query_contents
Query-contents:                 unknown
Case insensitive URLs:          no
Allow Windows-style URLs:       yes
Current requests :              0
Total requests :                2
```

En seguida fué verificada la configuración del espacio de objetos protegido, incluyendo *ACLs* que aplican para cada uno de los servidores de *WebSEAL*.

Como primer paso se listaron los objetos existentes:

```
pdadmin sec_master> object list /WebSEAL/<servidor>-default
/WebSEAL/<servidor>-default/banco-prov-net1
/WebSEAL/<servidor>-default/banco-prov-net2
/WebSEAL/<servidor>-default/banco-prov-net3
/WebSEAL/<servidor>-default/banco-prov-net4
/WebSEAL/<servidor>-default/banco-prov-net5
/WebSEAL/<servidor>-default/favicon.ico
```

Posteriormente se listan las propiedades de cada objeto de forma secuencial mediante el siguiente comando:

```
pdadmin sec_master> object show /WebSEAL/<servidor>-default/banco-prov-net1
Name:                       /WebSEAL/<servidor>-default/banco-prov-net1
Description:                Object from host <servidor>.
Type:                       16 (Management Object)
Is Policy Attachable:       Yes
Extended Attributes:
Attached ACL:               Publico
Attached POP:
Attached AuthzRule:
Effective Extended Attributes:
Effective ACL:              Publico
Effective POP:
Effective AuthzRule:

pdadmin sec_master> object show /WebSEAL/<servidor>-default/banco-prov-net2
Name:                       /WebSEAL/<servidor>-default/banco-prov-net2
Description:                Object from host <servidor>.
Type:                       16 (Management Object)
Is Policy Attachable:       Yes
Extended Attributes:
Attached ACL:               Privado_ve
Attached POP:
Attached AuthzRule:
Effective Extended Attributes:
Effective ACL:              Privado_ve
Effective POP:
Effective AuthzRule:

pdadmin sec_master> object show /WebSEAL/<servidor>-default/banco-prov-net3
Name:                       /WebSEAL/<servidor>-default/banco-prov-net3
Description:                Object from host <servidor>.
Type:                       16 (Management Object)
Is Policy Attachable:       Yes
Extended Attributes:
Attached ACL:               Publico
Attached POP:
Attached AuthzRule:
```

```
Effective Extended Attributes:
Effective ACL:            Publico
Effective POP:
Effective AuthzRule:

pdadmin sec_master> object show /WebSEAL/<servidor>-default/banco-prov-net4
Name:                     /WebSEAL/<servidor>-default/banco-prov-net4
Description:              Object from host <servidor>.
Type:                     16 (Management Object)
Is Policy Attachable:     Yes
Extended Attributes:
Attached ACL:             Privado_ve
Attached POP:
Attached AuthzRule:
Effective Extended Attributes:
Effective ACL:            Privado_ve
Effective POP:
Effective AuthzRule:

pdadmin sec_master> object show /WebSEAL/<servidor>-default/banco-prov-net5
Name:                     /WebSEAL/<servidor>-default/banco-prov-net5
Description:              Object from host <servidor>.
Type:                     16 (Management Object)
Is Policy Attachable:     Yes
Extended Attributes:
Attached ACL:             RECURSOS_PUBLICOS
Attached POP:
Attached AuthzRule:
Effective Extended Attributes:
Effective ACL:            RECURSOS_PUBLICOS
Effective POP:
Effective AuthzRule:

pdadmin sec_master> object show /WebSEAL/<servidor>-default/favicon.ico
Name:                     /WebSEAL/<servidor>-default/favicon.ico
Description:              Object from host <servidor>.
Type:                     16 (Management Object)
Is Policy Attachable:     Yes
Extended Attributes:
Attached ACL:             Publico
Attached POP:
Attached AuthzRule:
Effective Extended Attributes:
Effective ACL:            Publico
Effective POP:
Effective AuthzRule:

pdadmin sec_master> acl show Publico
ACL Name:                 Publico
Description:
Entries:
User sec_master TcmdbsvaBRl
Any-other Tr
Unauthenticated Tr

pdadmin sec_master> acl show RECURSOS_PUBLICOS
ACL Name:                 RECURSOS_PUBLICOS
Description:
Entries:
User sec_master TcmdbsvaBRl
Any-other Tr
Unauthenticated Tr

pdadmin sec_master> acl show Privado_ve
ACL Name:                 Privado_ve
Description:
Entries:
User sec_master TcmdbsvaBRl
Group bnetve01 Tr
Group bnetve02 Tr
```

En realidad los 2 grupos que aparecen en la salida anterior poseen los mismos permisos, ya que ambos son grupos de usuarios calificados como personas físicas y estaban asociados a la misma *ACL*. Sin embargo, el producto de directorio *Sun Directory Server* tenía un límite en el número de miembros que puede contener un grupo, y por esta razón fue particionado el universo total de los usuarios. A continuación se muestran las propiedades que fueron configuradas para cada grupo.

```
pdadmin sec_master> group show bnetve01
Group ID:           bnetve01
LDAP DN:            cn=bnetve01,c=ve,o=InstitucionBancaria
Description:
LDAP CN:            bnetve01
Is SecGroup:        Yes
pdadmin sec_master> group show bnetve02
Group ID:           bnetve02
LDAP DN:            cn=bnetve02,c=ve,o=InstitucionBancaria
Description:
LDAP CN:            bnetve02
Is SecGroup:        Yes
```

Como parte de la configuración de las instancias de *WebSEAL* del servicio de la banca en línea, conocido como *Nueva BNET* se integraba un servicio de portal de pagos, para el cuál fue necesario configurar en uno de los 4 servidores, una instancia de *WebSEAL* llamada "*webseald-PortalPagosP*" apuntando al mismo servidor de *PD*. Para este servicio la definición de las *junction* es distinta y a continuación se presenta la sintáxis usada para la creación de dichas *junction*:

server task PortalPagosP-webseald-ccswsealcfp10 create -f -t ssl -h 172.16.25.72 -p 6033 -c iv_user,iv_groups -j -s -r -e lcp_bin /DFAUTH
server task PortalPagosP-webseald-ccswsealcfp10 create -f -t ssl -h 172.16.25.72 -p 6039 -c iv_user,iv_groups -j -s -r -e lcp_bin /BANCO_VE_WEB
server task PortalPagosP-webseald-ccswsealcfp10 create -f -t ssl -h 172.16.25.72 -p 6040 -c iv_user,iv_groups -j -s -r -e lcp_bin /POPA_VE_WEB

Tabla A.5 Definición de *junctions* para el servicio de Portal de Pagos

Después de haber ejecutado la secuencia de comandos anterior, fue verificado el listado de propiedades para cada una de las *junction* de dicha instancia. A continuación se muestra el comando ejecutado para cada una de las 3 *junctions*, considerando que las propiedades eran exactamente iguales:

```
s t PortalPagosP-webseald-ccswsealcfp10 s /Junction-name
Junction point:                 /Junction-name
Type:                           SSL
Junction hard limit:            0 - using global value
Junction soft limit:            0 - using global value
Active worker threads:          0
Basic authentication mode:      filter
Forms based SSO:                disabled
TFIM junction SSO:              no
Authentication HTTP header:     insert - iv_user iv_groups
Remote Address HTTP header:     insert
Stateful junction:              yes
Boolean Rule Header:            no
Scripting support:              yes
Preserve cookie names:          no
Cookie names include path:      no
Transparent Path junction:      no
Delegation support:             no
Mutually authenticated:         no
Insert WebSphere LTPA cookies:  no
```

```
Insert WebSEAL session cookies:  no
Request Encoding:                Local Code Page, Binary
Server 1:
ID:                              ceaf378e-8419-11e0-b08c-00215e2ffaa0
Server State:                    running
Operational State:               Online
Hostname:                        172.16.25.72
Port:                            6040
Virtual hostname:                172.16.25.72:6040
Server DN:
local IP address:
Query_contents URL:              /cgi-bin/query_contents
Query-contents:                  unknown
Case insensitive URLs:           no
Allow Windows-style URLs:        yes
Current requests :               0
Total requests :                 3
```

A continuación se muestra el espacio de objetos protegidos que fueron configurados para dicha instancia "*webseald-PortalPagosP*", así como el detalle de cada *junction* y el detalle de las *ACLs* asociadas:

```
pdadmin sec_master> object list /WebSEAL/ccswsealcfp10-PortalPagosP
/WebSEAL/ccswsealcfp10-PortalPagosP/BANCO_VE_WEB
/WebSEAL/ccswsealcfp10-PortalPagosP/DFAUTH
/WebSEAL/ccswsealcfp10-PortalPagosP/POPA_VE_WEB
/WebSEAL/ccswsealcfp10-PortalPagosP/cgi-bin
/WebSEAL/ccswsealcfp10-PortalPagosP/favicon.ico
/WebSEAL/ccswsealcfp10-PortalPagosP/icons
/WebSEAL/ccswsealcfp10-PortalPagosP/index.html
/WebSEAL/ccswsealcfp10-PortalPagosP/pics

pdadmin sec_master> object show /WebSEAL/ccswsealcfp10-PortalPagosP/DFAUTH
Name:                      /WebSEAL/ccswsealcfp10-PortalPagosP/DFAUTH
Description:               Object from host ccswsealcfp10.
Type:                      16 (Management Object)
Is Policy Attachable:      Yes
Extended Attributes:
Attached ACL:              Publico
Attached POP:
Attached AuthzRule:
Effective Extended Attributes:
Effective ACL:             Publico
Effective POP:
Effective AuthzRule:

pdadmin sec_master> object show /WebSEAL/ccswsealcfp10 \
PortalPagosP/BANCO_VE_WEB
Name:                      /WebSEAL/ccswsealcfp10-PortalPagosP/BANCO_VE_WEB
Description:               Object from host ccswsealcfp10.
Type:                      16 (Management Object)
Is Policy Attachable:      Yes
Extended Attributes:
Attached ACL:              Privado_ve
Attached POP:
Attached AuthzRule:
Effective Extended Attributes:
Effective ACL:             Privado_ve
Effective POP:
Effective AuthzRule:
```

ANEXO B.- PROCEDIMIENTO DE ENDURECIMIENTO PARA LOS SISTEMAS OPERATIVOS

La descripción de los servicios de endurecimiento de los sistemas operativos *Solaris 10* para la plataforma *Sun SPARC Enterprise M4000*, y *SUSE Linux Enterprise Server 10* que en lo sucesivo será denominado como *SLES 10* para la plataforma *IBM System x3650 M2*, que formaron parte de la solución propuesta, y que se encuentran alineados a un modelo de confianza y una arquitectura de seguridad, deben cumplir con las siguientes metas.

- Elaborar y proveer el contexto para otros componentes que forman parte de una arquitectura de seguridad.
- Determinar y formalizar los puntos que representan un riesgo.
- Soporte al proceso de análisis de riesgo utilizado dentro de la arquitectura de seguridad para las instancias de la solución del dominio seguro.
- Atenuar los riesgos descubiertos.

1.- Seguridad en los Usuarios del Sistema

Una vez que se habían definido los objetivos y el alcance, se procedió a ejecutar un plan se endurecimiento de sistema operativo relacionado con los usuarios del sistema y los usuarios de administración.
Solaris 10 maneja un esquema de administración de usuarios, los cuales pueden tomar diferentes roles y cuentan con distintos privilegios dentro del sistema, ofreciendo la ventaja de que un rol no puede autenticarse a través de la red, ni tampoco puede tener una consola al inicio de una sesión. Con respecto a la plataforma basada en *SLES*, los mecanismos de implementación de los controles de seguridad pueden variar debido a que es un sistema operativo diferente. En caso de ser necesario, debía considerarse la configuración del *RBAC (Role Based Access Configuration)* para *Solaris,* así como el control de accesos basado en privilegios a través de la utilería *sudo* para *Linux.* A continuación se presentan los pasos que se deben seguir para el proceso de endurecimiento:

- Verificación de usuarios
- Verificación de privilegios de usuarios.
- Asignación de los derechos para auditar privilegios de perfil a los usuarios.
- Creación de nuevos roles.
- Asignación de operaciones restringidas a los nuevos roles.
- Verificación de configuración de roles.
- Verificación de acciones de los usuarios.

Dentro de este segmento de seguridad para los sistemas operativos *Solaris 10* y *SuSE Linux Enterprise Server* se tenían que ejecutar las siguientes actividades en cada una de las particiones físicas y/o lógicas de los servidores de la solución de dominio seguro, *Sun SPARC Enterprise M4000*, e *IBM System x3650 M2*, utilizando la siguiente nomenclatura para los controles de seguridad:

Nombre del control de seguridad

- Plataforma
- Tiempo requerido aproximado para la implantación en cada ambiente
- Pasos de preparación
- Riesgos previstos
- Proceso de activación del control

- Comprobación
- Actividades en caso de fallo
- Criterios de aceptación

1.1.- Robustecer la seguridad para acceso al servidor

Plataforma
Solaris 10 y SLES 10

Tiempo requerido
25 minutos

Pasos de preparación en Solaris 10
• Respaldar el archivo /etc/default/login
```
#cp /etc/login/default/login /etc/default/login.<dia>
```

Pasos de preparación en SLES 10
• Respaldar el archivo /etc/default/login
```
#cp /etc/pam.d/common-auth /etc/pam.d/common-auth.<dia>
```

Riesgos previstos
• Que los usuarios se bloquen por no “recordar” su contraseña.
• Aplicar la política de bloqueo a cuentas que no las debiera tener.

Proceso de activación en Solaris 10
• Editar el archivo /etc/default/login
• Modificar los siguientes valores
```
SYSLOG=YES
SLEEPTIME=4
RETRIES=3
SYSLOG_FAILED_LOGINS=3
```

Proceso de activación en SLES
• Editar el archivo /etc/pam.d/common-auth
• Agregar las siguientes líneas
```
auth        required    pam_tally.so      onerr=fail  no_magic_root
account     required    pam_tally.so      per_user    deny=5
no_magic_root reset
```

Comprobación
Verificar que en el archivo /etc/default/login y verificar que los parámetros tengan los valores definidos.

Actividades en caso de fallo para Solaris 10
Regresar el archivo /etc/login:
```
#cp /etc/login.<dia> /etc/login
```

Actividades en caso de fallo para SLES 10
Regresar el archivo /etc/pam.d/common-auth:
```
#cp /etc/pam.d/common-auth.<dia> /etc/pam.d
```

Criterios de aceptación

• Se realicen los cambios planeados
• Una vez aplicados los cambios, el servidor continué operando correctamente como lo hacia antes de realizar las actividades
• Una vez aplicados los cambios, las aplicaciones continúen operando correctamente como lo hacian antes de realizar las actividades
• Que las pruebas de comprobación del control sean exitosas.

1.2.- Limitar cron's

Plataforma
Solaris 10 y SLES 10

Tiempo requerido
10 minutos

Pasos de preparación en Solaris 10
• Obtener la lista de usuarios permitidos a ejecutar cron's
• Respaldar el archivo `/etc/cron.d/cron.allow`

```
#cp /etc/cron.d/cron.allow /etc/cron.d/cron.allow.<dia>
```

Pasos de preparación en SLES 10
• Obtener la lista de usuarios permitidos a ejecutar cron's
• Respaldar el archivo `/etc/cron.allow`

```
#cp /etc/cron.allow /etc/cron.allow.<dia>
```

Riesgos previstos
• Quitarle permisos a un usuario valido.
La corrección es inmediata

Proceso para limitar calendarización de tareas en Solaris 10
Crear el archivo `/etc/cron.d/cron.deny` agregando a los usuarios que no necesiten generar tareas programadas.
Ejemplo:

```
#vi /etc/cron.d/cron.deny
daemon
bin
nuucp
listen
nobody
```

Si existiera algún otro usuario no permitido para usar cron agregarlo al final del archivo.
Por otro lado se recomienda mejor crear y utilizar el archivo /etc/cron.d/cron.allow, en el que van todos los usuarios que si tienen permitido realizar tareas programadas.

Proceso para limitar calendarización de tareas en SLES 10
Crear el archivo /etc/cron.deny agregando a los usuarios que no necesiten generar tareas programadas. Por otro lado se recomienda mejor crear y utilizar el archivo /etc/cron.d/cron.allow, en el que van todos los usuarios que si tienen permitido realizar tareas programadas.

Comprobación
Verificar la existencia de los archivos cron.allow y cron.deny y su empleo, entrando a las rutas correspondienrtes.

La trayectoria en donde se localiza el registro de los “archivos cron” es: /var/cron/log

Actividades en caso de fallo para Solaris 10
Regresar el archivo /etc/cron.d/cron.allow
```
#cp /etc/cron.d/cron.allow.<dia> /etc/cron.d/cron.allow
```

Actividades en caso de fallo para SLES 10
Regresar el archivo /etc/cron.d/cron.allow
```
#cp /etc/cron.allow.<dia> /etc/cron.allow
```

Criterios de aceptación
• Se realicen los cambios planeados
• Una vez aplicados los cambios, el servidor continué operando correctamente como lo hacia antes de realizar las actividades
• Una vez aplicados los cambios, las aplicaciones continúen operando correctamente como lo hacian antes de realizar las actividades
• Que las pruebas de comprobación del control sean exitosas

1.3.- Restricción de root en perfiles de usuario

Plataforma
Solaris 10 y SLES 10

Tiempo requerido
30 minutos

Riesgos previstos
• Quitarle permisos a un usuario valido.
La corrección es inmediata

Pasos de preparación
• Obtener la lista de usuarios permitidos a tener uid de administrador
• Respaldar el archivo /etc/passwd
```
#cp /etc/passwd /etc/passwd.<dia>
```

Proceso para limitar perfil de root
• Validar que solo root tenga ID = 0
o Ejecutar la siguiente instrucción para verificar que solo root tenga ID=0
```
#nawk –F: '{if ($3==0 || $4==3) print $1,$3,$4}' /etc/passwd
```
• Para cambiar el ID de los usuarios reportados ejecutar el siguiente comando:
```
#usermod -u <num_nvo_ID> <nom_user>
```
• El perfil de root maneja como UID el 0 y como GID el 1.

Comprobación
Para verificar el número de usuarios pueden utilizar el siguiente comando:
```
#nawk '{FS=":"};{if ($3==0 || $4==1) print $1,$3,$4}' /etc/passwd
root 0 1
daemon 1 1
```
Por regla general, solo debería haber estos usuarios. Si existen más hay que verificar si son absolutamente necesarios.

Actividades en caso de fallo

Regresar el archivo /etc/passwd original:
```
#cp /etc/passwd.<dia> /etc/passwd
```

Criterios de aceptación
• Se realicen los cambios planeados
• Una vez aplicados los cambios, el servidor continué operando correctamente como lo hacia antes de realizar las actividades.
• Una vez aplicado los cambios, las aplicaciones continúen operando correctamente como lo hacian antes de realizar las actividades
• Que las pruebas de comprobación del control sean exitosas

1.4.- Limitar perfil de root en grupos

Plataforma
Solaris 10 y SLES 10

Tiempo requerido
5 minutos

Pasos de preparación
• Obtener la lista de usuarios permitidos a pertenecer al grupo administrador
• Respaldar el archivo /etc/group
```
#cp /etc/group /etc/group.<dia>
```

Riesgos previstos
• Quitarle permisos a un usuario valido.
La corrección es inmediata

Proceso para limitar el perfil de root
Para evitar usuarios “ocultos“ en /etc/group se debe verificar que no existan usuarios dentro del grupo de root con la siguiente instrucción:
```
#nawk –F: '{if ($3==0) print $1,$3,$4}' /etc/group
```

Comprobación
• Ejecutar el siguiente comando
```
#nawk '{FS=":"};{if ($3==0) print $1,$3,$4}' /etc/group
```
La salida debe ser:
root 0 root

Actividades en caso de fallo
Regresar el archivo /etc/group original:
```
#cp /etc/group.<dia> /etc/group
```

Criterios de aceptación
• Se realicen los cambios planeados
• Una vez aplicados los cambios, el servidor continué operando correctamente como lo hacia antes de realizar las actividades
• Una vez aplicados los cambios, las aplicaciones continúen operando correctamente como lo hacia antes de realizar las actividades
• Que las pruebas de comprobación del control sean exitosas

1.5.- Política de contraseñas

Plataforma
Solaris 10 y SLES 10

Para definir la política de contraseñas se deben realizar 2 tareas:
• Definir la longitud mínima de 8 caracteres
• Definir la vigencia de 30 días

Tiempo requerido
A partir de tener la lista de las cuentas a las que se les va a aplicar la política de contraseñas, 30 minutos.

Pasos de preparación en Solaris 10
• Respaldar el archivo /etc/default/passwd
```
#cp /etc/default/passwd /etc/default/passwd.<dia>
```

Pasos de preparación en SLES 10
• Respaldar el archivo /etc/pam.d/common-password
```
#cp /etc/default/passwd /etc/pam.d/common-password.<dia>
```

Riesgos previstos
• Aplicarle la política a una cuenta aplicativa
La corrección es inmediata

Proceso de activación en Solaris 10
Para modificar la longitud de la contraseña de los usuarios y obligarlos a que esta sea de 8 caracteres, hay que cambiar el parámetro de PASSLENGHT del archivo /etc/default/passwd por el valor 8 como se muestra a continuación:
```
#vi /etc/default/passwd
MAXWEEKS=
MINWEEKS=
PASSLENGTH=8
```
En donde:
MAXWEEKS Es el máximo número de días en las que la contraseña es válida.
MINWEEKS Es el mínimo número de días entre cambios de contraseña
PASSLENGTH Es la longitud de la contraseña Solo los primeros ocho caracteres son los significativos.

Proceso de activación en SLES 10
Para modificar la longitud de contraseña de los usuarios y obligarlos a que esta sea de 8 caracteres, se debe ejecutar el siguiente comando:
```
#pam-config –cracklib-minlen=8
```

Comprobación
Teclear un more /etc/default/passwd y verificar que el valor del parámetro de PASSLENGTH sea el deseado “8”.
Considere instalar Npasswd para llevar un control de seguridad de las contraseñas más confiable.

Actividades en caso de fallo para Solaris 10
Regresar el archivo /etc/default/passwd original:

```
#cp /etc/default/passwd.<dia> /etc/default/passwd
```

Actividades en caso de fallo para SLES 10
Regresar el archivo /etc/pam.d/common-password original:
```
#cp /etc/pam.d/common-password.<dia> /etc/pam.d/common-password
```

Criterios de aceptación
• Se realicen los cambios planeados
• Una vez aplicados los cambios, el servidor continué operando correctamente como lo hacia antes de realizar las actividades
• Una vez aplicados los cambios, las aplicaciones continúen operando correctamente como lo hacian antes de realizar las actividades
• Que las pruebas de comprobación del control sean exitosas

1.6.- Vigencia de las contraseñas

Tiempo requerido
A partir de tener la lista de las cuentas a las que se les va a aplicar la política de contraseñas, 30 minutos.

Pasos de preparación en Solaris 10
• Respaldar el archivo /etc/shadow
```
#cp /etc/shadow /etc/shadow.<dia>
```

Pasos de preparación en SLES 10
• Respaldar el archivo /etc/login.defs
```
#cp /etc/login.defs /etc/login.defs.<dia>
```

Riesgos previstos
• Aplicarle la política a una cuenta aplicativa
La corrección es inmediata

Proceso de activación en Solaris 10
Una parte de la administración de las contraseñas es la de aplicar reglas de expiración de los mismos para los usuarios. La forma de indicarle al usuario que debe de cambiar su contraseña cada 30 días es la siguiente:
Ejecutar:
```
#passwd -x 30 <usuario>
```
en donde:
–x indica el máximo número de días que deberá tener esa contraseña para validar la aplicación de la regla.

Proceso de activación en SLES 10
La vigencia de la contraseña también es establecida a 30 días y para ello se ejecuta el siguiente comando:
```
#chage –M 30 <usuario>
```

Comprobación para Solaris 10
Para validar que la aplicación de la regla se cumpla y ver el status de los usuarios con su límite de expiración de contraseña se utiliza el siguiente comando:
```
#passwd –sa
```
El cual despliega algo similar a:

vi616xy PS 08/09/05 0 30
La primera columna se refiere al usuario
La segunda al estatus de la contraseña, la cual puede ser:
PS para cuentas que tienen contraseña y están activas
LK para cuentas bloqueadas
NP para cuentas que no tienen contraseña
La fecha mm/dd/yy es de la ultima vez que cambio del contraseñas
La columna 4 es el # de días requeridos entre cambio de contraseñas
La columna 5 es para el numero de días en el que la contraseña esta vigente

Comprobación para SLES 10
Para validar que la aplicación de la regla se cumpla y ver el status de los usuarios con su límite de expiración se ejecuta el comando:
#chage –l <usuario>

Actividades en caso de fallo para Solaris 10
Regresar el archivo /etc/shadow original:
#cp /etc/shadow.<dia> /etc/shadow

Actividades en caso de fallo para SLES 10
Regresar el archivo /etc/shadow original:
#cp /etc/login.defs.<dia> /etc/login.defs

Criterios de aceptación
• Se realicen los cambios planeados
• Una vez aplicados los cambios, el servidor continué operando correctamente como lo hacia antes de realizar las actividades
• Una vez aplicados los cambios, las aplicaciones continúen operando correctamente como lo hacian antes de realizar las actividades
• Que las pruebas de comprobación del control sean exitosas

1.7.- Algoritmo de cifrado de contraseñas

En este procedimiento, nos basamos en el algoritmo *MD5*, el cual es usado como algoritmo por defecto que es usado cuando los usuarios cambian sus contraseñas. Este algoritmo es adecuado para redes mixtas de maquinas que corren *Solaris, BSD, y Linux*. En esta ocasión los algoritmos de configuración de cifrado aseguran que el algoritmo más debil *crypt_unix* nunca será usado.

Tiempo Requerido
20 minutos

Pasos de preparación en Solaris 10
Respaldar el archivo ***/etc/security/policy.conf*** mediante el siguiente comando:

```
# cp /etc/security/policy.conf /etc/security/old.policy
```

Pasos de preparación en SLES 10
Respaldar el archivo ***/etc/default/passwd*** mediante el siguiente comando:

```
# cp /etc/default/passwd /etc/default/passwd.old
```

Riesgos Previstos
Es importante contemplar que se tiene evitar cualquier error dentro del archivo de configuración de los algoritmos de encripción para evitar un mal funcionamiento

Proceso de activación en Solaris 10
Para activar el algoritmo de cifrado que requerimos, hay ejecutar las siguientes tareas de manera secuencial:

```
# vi /etc/security/policy.conf
....
CRYPT_ALGORITHMS_ALLOW=1,2a,md5

#Use the version of MD5 that works with linux
#CRYPT_DEFAULT=__unix__

CRYPT_DEFAULT=1
```

Proceso de activación en SLES 10
Para activar el algoritmo de cifrado que requerimos, hay ejecutar las siguientes tareas de manera secuencial:

```
# vi /etc/default/passwd
CRYPT = des

#Use the version of MD5 that works with linux
#CRYPT = des

CRYPT = md5
```

Actividades en caso de fallo para Solaris 10
En caso de presentarse un mal funcionamiento después de haber efectuado dichas modificaciones en el archivo de configuración, se tiene que corregir nuevamente restaurando el archivo original:

```
# cp /etc/security/old.policy /etc/security/policy.conf
```

Actividades en caso de fallo para SLES 10
En caso de presentarse un mal funcionamiento después de haber efectuado dichas modificaciones en el archivo de configuración, se tiene que corregir nuevamente restaurando el archivo original:

```
# cp /etc/default/passwd.old /etc/default/passwd
```

Criterios de aceptación
• Se debe cambiar la contraseña de un usuario para validar que dentro del archivo */etc/*shadow en el campo de la contraseña que corresponde a dicho usuario, existe una cadena mayor a 13 caractéres.

1.8.- Establecer máscara de creación de archivos

Tiempo requerido
A partir de tener la lista de las cuentas a las que se les va a aplicar la mascara, 30 minutos.

Pasos de preparación en Solaris 10
• Respaldar el archivo /etc/default/login
#cp /etc/default/login /etc/default/login.*<dia>*

Pasos de preparación en SLES 10
• Respaldar el archivo /etc/login.defs
#cp /etc/default/login.defs /etc/login.defs.*<dia>*

Riesgos previstos
• Afectarle a una cuenta aplicativa
La corrección es inmediata, se debe agregar la mascara deseada a su archivo .profile

Proceso para establecer la máscara de creación de archivos en Solaris 10
Para que se cambie el default de la asignación de permisos se debe colocar en el archivo profile del usuario el comando umask, es por ello que es preferible asignarle a los usuarios una mascara más segura de forma global.
Editar el archivo ***/etc/default/login y*** descomentar la línea umask y ponerle el valor 077
UMASK=077
Nota: Validar el no impacto sobre usuarios de aplicación.
Los permisos que le asigna son de lectura y escritura para el usuario y no asigna permisos ni para el grupo ni para el resto.

Proceso para establecer la máscara de creación de archivos en SLES 10
Editar el archivo **/etc/login.defs y** colocar la línea umask con el valor 0077
UMASK=0077

Comprobación
Para verificar que la mascara sea la adecuada hay que teclear el comando **umask** y ver el dato que arroja, si no equivale al que se pide **"007"** cambiarlo.

Actividades en caso de fallo para Solaris 10
Regresar el archivo /etc/default/login original:
#cp /etc/default/login.<dia> /etc/default/login

Actividades en caso de fallo para SLES 10
Regresar el archivo /etc/login.defs original:
#cp /etc/login.defs.<dia> /etc/login.defs

Criterios de aceptación
• Se realicen los cambios planeados
• Una vez aplicados los cambios, el servidor continué operando correctamente como lo hacia antes de realizar las actividades
• Una vez aplicados los cambios, las aplicaciones continúen operando correctamente como lo hacian antes de realizar las actividades
• Que las pruebas de comprobación del control sean exitosas

1.9.- Limitar conexiones confiables

Tiempo requerido
15 minutos

Pasos de preparación
• Obtener la lista de relaciones de confianza utilizadas y requeridas

Riesgos previstos
• Quitarle permisos a una cuenta que requiere el servicio.
La corrección es inmediata

Proceso para limitar conexiones confiables en Solaris 10 y SLES 10
• Correr el siguiente comando:
#find / \(-name .rhosts –o -name .shosts -o -name .netrc \) -ls -ok mv {} .rhost_respaldo \; > rhosts.log
Si encuentra un archivo que permite el acceso remoto lo renombra y guarda una bitácora de todos los archivos que hallo.

• Renombrar el archivo */etc/hosts.equiv*
```
# mv /etc/hosts.equiv /etc/host.equiv.orig
```

Comprobación
• Correr el comando:
#find / \(-name .rhosts –o -name .shosts -o -name .netrc \) -ls
No debe aparecer ningún archivo
• Correr el comando:
#cat /etc/hosts.equiv
No debe encontrar el archivo

Actividades en caso de fallo
Regresar el archivo .rhost del usuario afectado
```
#cp .rhost_respaldo .rhost
```

Criterios de aceptación
• Se realicen los cambios planeados
• Una vez aplicado los cambios, el servidor continué operando correctamente como lo hacia antes de realizar las actividades
• Una vez aplicado los cambios, las aplicaciones continúen operando correctamente como lo hacia antes de realizar las actividades
• Que las pruebas de comprobación del control sean exitosas

1.10.- Definir shell's validos

Tiempo requerido
25 minutos

Pasos de preparación
• Respaldar el archivo /etc/shells

```
#cp /etc/shells /etc/shells.<dia>
```

Riesgos previstos
No hay impacto previsto

Proceso de definición de shell's validos en Solaris 10 y SLES 10
• Correr el siguiente comando:

```
#vi /etc/shells
Agregar los siguientes shells
/sbin/sh
/nosuchshell
/bin/ksh
/bin/csh
/usr/sbin/ksh
/bin/false
/bin/rksh
/usr/bin/ksh
/usr/bin/bash
```
• Ponerle permisos de solo lectura
```
#chmod 444 /etc/shells
```

Comprobación
Teclear un more /etc/shells y verificar que shells están permitidos

Actividades en caso de fallo
Regresar el archivo /etc/shells original:
```
#cp /etc/shells.<dia> /etc/shells
```

Criterios de aceptación
• Se realicen los cambios planeados
• Una vez aplicados los cambios, el servidor continué operando correctamente como lo hacia antes de realizar las actividades
• Una vez aplicados los cambios, las aplicaciones continúen operando correctamente como lo hacian antes de realizar las actividades
• Que las pruebas de comprobación del control sean exitosas

1.11.- Bloqueo de cuentas del sistema sin utilizar

Tiempo requerido
25 minutos

Pasos de preparación
• Respaldar el archivo /etc/passwd
```
#cp /etc/passwd /etc/passwd.<dia>
```
• Respaldar el archivo /etc/shadow
```
#cp /etc/shadow /etc/shadow.<dia>
```

Riesgos previstos
No hay impacto previsto

Proceso de activación en Solaris 10 y SLES 10
• Ejecutar los siguientes comandos:
```
#for user in uucp nuucp sys adm daemon bin lp nobody4 nobody noaccess
smmsp listen
do
passwd –l "$user"
/usr/sbin/usermod -s /bin/false "$user"
done
```
Lo que realizan las líneas anteriores, es que para cada usuario mencionado bloqueara su passwd en el archivo /etc/shadow y asignara un shell falso a cada uno.

Comprobación
Verificar si existe en cada usuario mencionado la bandera de *LK* en el archivo /etc/shadow y en el /etc/passwd un shell falso (/bin/false) al final de la línea de cada usuario:

```
/etc/shadow:
lp:*LK*:13390::::::
listen:*LK*:::::::
/etc/passwd
lp:x:71:8:Line Printer Admin:/usr/spool/lp:/bin/false
listen:x:37:4:Network Admin:/usr/net/nls: :/bin/false
```

Actividades en caso de fallo
Regresar los archivos originales:

```
#cp /etc/passwd.<dia> /etc/passwd
#cp /etc/shadow.<dia> /etc/shadow
```

Criterios de aceptación
• Se realicen los cambios planeados
• Una vez aplicados los cambios, el servidor continué operando correctamente como lo hacia antes de realizar las actividades
• Una vez aplicados los cambios, las aplicaciones continúen operando correctamente como lo hacian antes de realizar las actividades
• Que las pruebas de comprobación del control sean exitosas

1.12.- Validación de consistencia de archivos de usuarios, grupos y contraseñas

Tiempo requerido
10 minutos

Pasos de preparación
• Respaldar el archivo /etc/passwd

```
#cp /etc/passwd /etc/passwd.<dia>
```

• Respaldar el archivo /etc/group

```
#cp /etc/group /etc/group.<dia>
```

• Respaldar el archivo /etc/shadow

```
#cp /etc/shadow /etc/shadow.<dia>
```

Riesgos previstos
No hay impacto previsto

Proceso de activación en Solaris 10 y SLES 10
• Ejecutar los siguientes comandos:

```
#pwconv
```

• Sincroniza o se actualizan el /etc/shadow con la información del /etc/passwd.
• Si el /etc/shadow no existe, este comando lo va a crear con la información del /etc/passwd
• Para verificar que no exista la bandera de + en el passswd y/o en el group realizar el siguiente comando:

```
#grep '^+:' /etc/passwd /etc/group
```

• Comandos para encontrar inconsistencias en el /etc/passwd y en /etc/group
• pwck

• grpck

Comprobación
Basta con ejecutar nuevamente los comandos y no debe de arrojar nada a la salida, de no ser así realizar el procedimiento.

Actividades en caso de fallo
Regresar los archivos originales:
```
#cp /etc/passwd.<dia> /etc/passwd
#cp /etc/group.<dia> /etc/group
#cp /etc/shadow.<dia> /etc/shadow
```

Criterios de aceptación
• Se realicen los cambios planeados
• Una vez aplicados los cambios, el servidor continué operando correctamente como lo hacia antes de realizar las actividades
• Una vez aplicados los cambios, las aplicaciones continúen operando correctamente como lo hacian antes de realizar las actividades
• Que las pruebas de comprobación del control sean exitosas

1.13.- Detección de archivos sin dueño/grupo

Tiempo requerido
Variable

Pasos de preparación
Correr el comando
```
#find / \( -nouser –o –nogroup \) –ls > archivos_sin_dueno.txt
```

Riesgos previstos
Impactar a alguna aplicación o proceso.

Proceso de activación en Solaris 10 y SLES 10
Es necesario saber si existen archivos sin dueño y sin grupo. Para ello ejecutar el siguiente comando:
```
#find / \( -nouser -o -nogroup \) -print -ls
```
La salida del find puede ser algo similar a lo siguiente:
```
-rw-r--r--  1  15178  20  12522  Jun  21  2002  /var/sadm/patch/111554-
09/README.111554-09
```
En dado caso de encontrar este tipo de archivos se tendrán que analizar debido a que pueden pertenecer a algún tipo de aplicación. Por otro lado estos archivos se le pueden asignar al usuario y grupo de root para su análisis.

Comprobación
• Correr los siguiente comando:
```
#find / \( -nouser –o –nogroup \) –ls
```
El comando no debe arrojar ningún resultado.

Actividades en caso de fallo
No aplica

Criterios de aceptación

• Se realicen los cambios planeados
• Una vez aplicados los cambios, el servidor continué operando correctamente como lo hacia antes de realizar las actividades
• Una vez aplicados los cambios, las aplicaciones continúen operando correctamente como lo hacian antes de realizar las actividades
• Que las pruebas de comprobación del control sean exitosas

2.- Seguridad en los Servicios de Red

Ambos sistemas operativos, Sun Solaris 10, y SuSE Linux Enterprise Server contaban con un modulo de administración de servicios que nos facilitaban la administración y control de los mismos. En el caso del sistema operativo Sun Solaris 10 es posible delegar el acceso a las funciones de administración del nucleo de servicios basados en el concepto del menor privilegio, en el que si un usuario ó servicio no requiere tener estrictamente un determinado grado de privilegios, entonces dichos privilegios no son otorgados.

En versiones anteriores de Solaris los servicios eran administrados y arrancados por medio de un run-control script (inetd), con el cuál solo el usuario root podía cambiar la configuración de los servicios mediante algún mecanismo. Además de que el acceso a root era estrictamente requerido para remover, habilitar, deshabilitar, reiniciar, detener, ó arrancar un servicio.

En el caso de Solaris 10 se debe determinar la propiedad “tipo de grupo” como se muestra en el siguiente ejemplo:

```
# svccfg -s cron listpg
usr              dependency
ns               dependency
general          framework
dependents       framework
startd           framework
start            method
stop             method
tm_common_name   template
tm_man_cron      template
tm_man_crontab   template
```

Note que la granularidad de estas autorizaciones esta en un nivel de propiedad “tipo de grupo”. Sin embargo esto no restringe que el servicio pueda ser modificado. Como resultado de garantizar una de estas autorizaciones se permitiría a un usuario ó rol, agregar, cambiar ó remover alguna propiedad de un servicio, siempre y cuando el grupo de propiedades sea manipulado por un miembro del grupo.

Cierre de servicios
Para el cierre de servicios se deben realizar 2 tareas:
- Cierre de servicios sobre demanda
- Cierre de servicios de inicio

2.1.- Servicios sobre demanda

Para cerrar los puertos y servicios dependientes de inetd.conf se debe de comentar cada servicio anteponiendo el signo de # al inicio de la línea en el archivo /etc/inet/inetd.conf. Se recomienda comentar todos los servicios que vienen por default con la instalación:
Posiblemente existan algunos otros servicios dados de alta que no se requieran utilizar por la aplicación que se tiene y por lo tanto debemos también comentarlos.

Tiempo requerido
25 minutos

Pasos de preparación en Solaris 10
• Obtener la lista de los servicios que se deseen deshabilitar
• Respaldar el archivo /etc/inetd.conf
`#cp /etc/inetd.conf /etc/inetd.conf.<dia>`

Pasos de preparación en SLES 10
Obtener la lista de los servicios que se deseen deshabilitar
• Respaldar el archivo /etc/xinetd.conf
`#cp /etc/xinetd.conf /etc/xinetd.conf.<dia>`

Riesgos previstos
Cerrar algún servicio que se este ocupando.
En caso de retorno es inmediato.

Proceso de cierre de servicios para Solaris 10 y SLES 10
• Editar el archivo /etc/inetd.conf y /etc/xinetd.conf respectivamente
• Desactivar los servicios, colocando al inicio de la línea el carácter **#**, por ejemplo:
`#imap stream tcp nowait root /etc/uva/tcp_wrapper/tcpd Usr/local/etc/imapd4 imapd`
• Una vez desactivados todos los servicios deseados, salvar el archivo
• Refrescar el daemon para que tome las modificaciones, con el comando :
`#kill –HUP <proceso de inetd>`

Comprobación
Teclear more /etc/inetd.conf o /etc/xinetd.conf, según sea el caso y verificar que estén comentados cada uno de los servicios deseados y posteriormente ejecutar un netstat –a para confirmar que ya no estén corriendo esos servicios.

Actividades en caso de fallo para Solaris 10
Regresar el archivo /etc/inetd.conf
`#cp /etc/inetd.conf.<dia> /etc/inetd.conf`

Actividades en caso de fallo para SLES 10
Regresar el archivo /etc/xinetd.conf
`#cp /etc/xinetd.conf.<dia> /etc/xinetd.conf`

Criterios de aceptación
• Se realicen los cambios planeados
• Una vez aplicados los cambios, el servidor continué operando correctamente como lo hacia antes de realizar las actividades
• Una vez aplicados los cambios, las aplicaciones continúen operando correctamente como lo hacian antes de realizar las actividades

• Que las pruebas de comprobación del control sean exitosas

2.2.- Servicios de inicio ó de boot

Tiempo requerido
25 minutos por servicio a deshabilitar

Pasos de preparación
• Obtener la lista de los servicios que se deseen deshabilitar

Riesgos previstos
Cerrar algún servicio que se este ocupando.
En caso de retorno 5 minutos por servicio que se desee retornar

Proceso de activación en Solaris 10
Deshabilitar los servicios no requeridos. En caso de no ser necesarios, se recomienda deshabilitar los siguientes servicios y configurar para que no se levanten automáticamente cuando se reinicie el servidor
S88sendmail .- *Servicio de sendmail*
S73nfs.client .- *NFS*
S74autofs .- *NFS*
S84appserver .- *Servidor Administrativo de SUN ONE*

Para deshabilitar los servicios:
• Ejecutar los siguientes comandos de Unix:

```
#cd /etc/rc2.d
#for var in S70uucp S71rpc S73nfs.client S74autofs S76nscd S80lp
S88sendmail S90wbem S99dtlogin S73cachefs.daemon S95networker
do
./$var stop
mv $var .NO_$var
done
```

Posteriormente reiniciar el servidor.

Proceso de activación en SLES 10
Deshabilitar los servicios no requeridos. En caso de no ser necesarios, se recomienda deshabilitar los servicios que se encuentren bajo el control de xinetd y configurarlos para que no se levanten automáticamente cuando se reinicie el servidor. Para ello es necesario ejecutar los siguiente comandos:

```
#vi /etc/xinetd.d/<servicio>
```

Colocar el valor yes a la variable disable, como se muestra en seguida:

```
disable = yes
```

Posteriormente reiniciar el servidor.

Comprobación
• Ejecutar un netstat –a para confirmar que ya no estén corriendo esos servicios

Actividades en caso de fallo para Solaris 10
Regresar el script de servicio que se desee reactivar, por ejemplo:

```
#mv .NO_S88sendmail S88sendmail
```

Correr el script para levantar el servicio, por ejemplo:

```
#./S88sendmail
```

Actividades en caso de fallo para SLES 10
Editar el archivo del servicio que corresponda bajo la ruta /etc/xinetd.d y colocar lo siguiente:
`disable = no`

Criterios de aceptación
• Se realicen los cambios planeados
• Una vez aplicados los cambios, el servidor continué operando correctamente como lo hacia antes de realizar las actividades
• Una vez aplicados los cambios, las aplicaciones continúen operando correctamente como lo hacian antes de realizar las actividades
• Que las pruebas de comprobación del control sean exitosas

2.3.- Bitácora de ftp

Tiempo requerido
25 minutos

Pasos de preparación en Solaris 10
• Respaldar el archivo /etc/inetd.conf

```
#cp /etc/inetd.conf /etc/inetd.conf.<dia>
```

Pasos de preparación en SLES 10
• Respaldar el archivo /etc/vsftpd.conf

```
#cp /etc/vsftpd.conf /etc/vsftpd.conf.<dia>
```

Riesgos previstos
No hay impacto previsto

Proceso de activación en Solaris 10
Se debe habilitar el log de ftp, como indica el documento 16725 de Sunsolve.
Document ID Synopsis Date
16725 How to turn on FTP logging to log all FTP commands. 20 Mar 1998
Problem Description
User is having problems logging FTP connections.

Proceso de activación en SLES 10
• Activar el parámetro xferlog_enable en el archivo /etc/vsftpd.conf

```
xferlog_enable = yes
```

Comprobación
Ejecutar un ftp a la máquina deseada, posteriormente revisar con un grep ftp /var/adm/messages o /var/vsftpd.log según sea el caso, para comprobar si existen registros del ftp ejecutado.

Actividades en caso de fallo para Solaris 10
Regresar el archivo /etc/inetd.conf original:

```
#cp /etc/inetd.conf.<dia> /etc/inetd.conf
```

Actividades en caso de fallo para SLES 10
Regresar el archivo /etc/vsftpd.conf original:
```
#cp /etc/vsftpd.conf.<dia> /etc/vsftpd.conf
```

Criterios de aceptación
• Se realicen los cambios planeados
• Una vez aplicado los cambios, el servidor continué operando correctamente como lo hacia antes de realizar las actividades
• Una vez aplicado los cambios, las aplicaciones continúen operando correctamente como lo hacia antes de realizar las actividades
• Que las pruebas de comprobación del control sean exitosas

2.4.- Ftpusers

Tiempo requerido
25 minutos

Pasos de preparación para Solaris 10 y SLES 10
• Obtener la lista de usuarios permitidos a realizar FTP
• Respaldar el archivo /etc/ftpusers o /etc/ftp-users, según corresponda
```
#cp /etc/ftpusers /etc/ftpusers.<dia>
#cp /etc/ftp-users /etc/ftp-users.<dia>
```

Riesgos previstos
Quitarle permisos de ftp a un usuario que si puede ocupar el servicio
La corrección de este problema es inmediata

Proceso de activación para Solaris 10
• Ejecutar los siguientes comandos sobre el archibvo ftpusers o ftp-users, según corresponda:
```
#if [[ "$(uname -r)" = 5.9 ]]; then
ftpusers=/etc/ftpd/ftpusers
else
ftpusers=/etc/ftpusers
fi
for name in root daemon bin sys adm lp uucp nuucp nobody
do
```

Proceso de activación para SLES 10
• Ejecutar los siguientes comandos sobre el archibvo ftpusers o ftp-users, según corresponda:
```
#if [[ "$(uname -r)" = 5.9 ]]; then
ftpusers=/etc/ftpd/ftp-users
else
ftpusers=/etc/ftp-users
fi
for name in root daemon bin sys adm lp uucp nuucp nobody
do
```

Comprobación

Realizar un ftp hacia la máquina deseada e intentar acceder con algún usuario que se encuentran en el /etc/ftpusers creado.

Actividades en caso de fallo para Solaris 10 o SLES 10
Regresar el archivo /etc/ftpusers original (verificar porque el archivo ftpusers puede estar bajo /etc/ftpd, dependiendo de la versión de Solaris y/o SLES):
```
#cp /etc/ftpusers.<dia> /etc/ftpuser
#cp /etc/ftp-users.<dia> /etc/ftp-users
```

Criterios de aceptación
• Se realicen los cambios planeados
• Una vez aplicados los cambios, el servidor continué operando correctamente como lo hacia antes de realizar las actividades
• Una vez aplicados los cambios, las aplicaciones continúen operando correctamente como lo hacian antes de realizar las actividades
• Que las pruebas de comprobación del control sean exitosas

2.5.- Habiliar TCP-wrappers

Tiempo requerido
30 minutos

Pasos de preparación en Solaris 10 y SLES 10
• Obtener la lista de direcciones Ip's permitidas a acceder al servidor
• Obtener el archivo de instalación de la herramienta TCP-WRAPPERS

Riesgos previstos
• No darle permisos de acceso a direcciones Ip's validas
Si se tiene acceso al servidor, la corrección es inmediata

Instalación y configuración para Solaris 10 y SLES 10
• Bajar el archivo tcp_wrappers-7.6.tar.gz para solaris o tcp_wrappers_7.6.tar.gz para SLES, descomprimirlo y desempaquetarlo.
• Desempaquetar e instalar el paquete que corresponda según sea el caso.
libiconv-1.8-sol9-sparc-local
• Entrar al directorio de tcp-wrappers y modificar el makefile
o Descomentar las siguientes líneas del Makefike
REAL_DAEMON_DIR=/usr/sbin/… #Para el caso de Solaris
STYLE = -DPROCESS_OPTIONS #Habilitar extensiones
• Cambiar la siguiente línea en el Makefile:
FACILITY=LOG_MAIL # Para registrar en syslog
por:
FACILITY=LOG_LOCAL0 # Enviar logs a otro archivo.
• Modificar el PATH de root.
```
#PATH=$PATH:/usr/ccs/bin:.
#export PATH
```
• Hacer la liga de gcc, la cual se debe borrar al final de la instalacion
```
#ln -s /usr/local/bin/gcc /usr/bin/cc
```
• Ejecutar make
```
#make
```
• Volver a ejecutar make o make install, según sea el caso
```
#make sunos5
```

• Copiar los archivos binarios que se han generado a /usr/sbin
#cp safe_finger tcpd tcpdchk tcpdmatch try-from /usr/sbin/
• Editar el archivo inetd.conf
```
telnet stream tcp nowait root /usr/sbin/tcpd /usr/sbin/in.telnetd
ftp stream tcp nowait root /usr/sbin/in.ftpd /usr/sbin/in.ftpd –a
ssh stream tcp nowait root /usr/sbin/tcpd /usr/local/sbin/sshd –i
```
Y en este punto cambiar tcp6 por tcp para trabajar con Ipv4. También cambiar las líneas de /usr/sbin/in.telnetd por /usr/local/bin/tcpd, /usr/sbin/in.ftpd por /usr/local/bin/tcpd y el de ssh por /usr/local/bin/tcpd.
• Modificar (o agregar al final las rutas) el archivo /etc/syslog.conf local0.info /var/adm/tcpd.log
mail.info /var/adm/tcpd.log

• Crear el archivo donde se van a mandar los logs
#touch /var/adm/tcpd.log
• Reiniciar los demonios de inetd y syslog.
#ps -fe | grep inetd
#kill -HUP proceso
#ps -fea | grep syslog
#kill -HUP proceso
Se recomienda aplicar la siguiente configuración
#echo "ALL:ALL" /etc/hosts.deny
• **Editar el archive /etc/hosts.allow. Agregar una línea por cada IP/segmento de red al que se desee dar acceso. La sintaxis de las líneas es:**
<servico>:<IP>
por ejemplo:
in.telnetd:192.168.80.

Comprobación
Verificar el log de Tpcwrappers tecleando **tail /var/adm/tcpd.log** para que despliegue las ultimas diez conexiones

Actividades en caso de fallo
Borrar o renombrar el archivo /etc/hosts.deny:
```
#mv /etc/hosts.deny /etc/hosts.deny.bak
```

Criterios de aceptación
• Se realicen los cambios planeados
• Una vez aplicados los cambios, el servidor continué operando correctamente como lo hacia antes de realizar las actividades
• Una vez aplicados los cambios, las aplicaciones continúen operando correctamente como lo hacian antes de realizar las actividades
• Que las pruebas de comprobación del control sean exitosas

2.6.- Configuración del servicio Secure Shell

Tiempo requerido
30 minutos

Pasos de preparación
• Obtener el archivo de instalación de la herramienta SSH

Riesgos previstos

No hay impacto previsto

Proceso de activación en Solaris 10

Si se va a instalar en un equipo con Solaris 10, antes de instalar se debe:

• Bajar el parche 112438 para corregir el modulo de /kernel/drv/random y corregir un problema en la generación de números aleatorios.

• Correr los siguientes commandos:

```
#patchadd 112438-01
#mknod /devices/pseudo/random at 0:random c 82 0
#mknod /devices/pseudo/random at 0:urandom c 82 1
#chgrp sys /devices/pseudo/random*
#cd /dev
#ln -s ../devices/pseudo/random at 0:random /dev/random
#ln -s ../devices/pseudo/random at 0:urandom /dev/urandom
#modload /kernel/drv/random
```

Para instalar el ssh:

• Para instalar SSH se requieren copiar al servidor los siguientes paquetes:
openssh-4.3p2-sol9-sparc-local.gz openssl-0.9.8c-sol9-sparc-local.gz tcp_wrappers-7.6-sol9-sparc-local.gz (opcional)
zlib-1.2.1-sol9-sparc-local.gz

• Descomprimir los paquetes, ejecutando los siguientes comandos

```
#gunzip openssh-4.3p2-sol9-sparc-local.gz
#gunzip openssl-0.9.8c-sol9-sparc-local.gz
#gunzip zlib-1.2.1-sol9-sparc-local.gz
#gunzip libgcc-3.3-sol9-sparc-local.gz
#gunzip tcp_wrappers-7.6-sol9-sparc-local.gz (opcional)
```

• Instalar los paquetes anteriores, ejecutando como root, los siguientes comandos:

```
#pkgadd -d openssh-4.3p2-sol9-sparc-local
#pkgadd -d openssl-0.9.8c-sol9-sparc-local
#pkgadd -d zlib-1.2.1-sol9-sparc-local
#pkgadd -d libgcc-3.3-sol8-sparc-local
#pkgadd -d tcp_wrappers-7.6-sol9-sparc-local (opcional)
```

• Crear el usuario y grupo aplicativo para el ssh. Para esto ejecutar los siguientes comandos:

```
#mkdir /var/empty
#chown root:sys /var/empty
#chmod 755 /var/empty
#groupadd sshd
#useradd -g sshd -c 'sshd privsep' -d /var/empty -s /bin/false sshd
```

• Modificar el archive /usr/local/etc/sshd_config

Cambiar la linea

Subsystem sftp /usr/libexec/sftp-server

por

Subsystem sftp /usr/local/libexec/sftp-server

• Si se tiene instalado tcp-wrappers, configurar el tcp-wrappers para soporte a ssh.

En el archivo /etc/hosts.deny, agregar la linea:

```
sshd: ALL
```

En el archivo /etc/hosts.allow , agregar la linea

```
sshd: <IPS>
```

donde <IPS> es una lista de IP's permitidas a conectarse por SSH, separadas por comas

• Configurar el ssh

```
#ssh-keygen -t rsa1 -f /usr/local/etc/ssh_host_key -N ""
```

```
#ssh-keygen -t dsa -f /usr/local/etc/ssh_host_dsa_key -N ""
#ssh-keygen -t rsa -f /usr/local/etc/ssh_host_rsa_key -N ""
```
• Crear un script de arranque para el servicio de SSH.
o Crear el archivo sshd, ejecutando el siguiente comando:
```
#vi /etc/init.d/sshd
#!/bin/sh
case "$1" in
'start')
if [ -x /usr/local/sbin/sshd ]; then
echo "Starting the secure shell daemon"
/usr/local/sbin/sshd &
fi
;;
'stop')
echo "Stopping the secure shell daemon "
pkill -TERM sshd
;;
*)
echo "Usage: /etc/init.d/sshd { start | stop }"
;;
esac
exit 0
```

o Correr los siguientes comandos
```
#chown root /etc/init.d/sshd #chgrp sys /etc/init.d/sshd #chmod 555
/etc/init.d/sshd #ln -s /etc/init.d/sshd /etc/rc2.d/S98sshd
#vi /etc/ssh/sshd_config
PermitRootLogin no
```
• Levantar el servicio
```
#/etc/rc2.d/S98sshd start
```

Proceso de activación en SLES 10
En el caso de SLES no es necesario descargar la última versión de ssh, ya que por defecto viene instalado, configurado, y listo para utilizarse.

Comprobación
Realizar una prueba de conexión a ese equipo vía ssh

Actividades en caso de fallo
No hay actividad a realizar. La herramienta puede vivir en el servidor aunque no se utilice.

Criterios de aceptación
• Se realicen los cambios planeados
• Una vez aplicados los cambios, el servidor continué operando correctamente como lo hacia antes de realizar las actividades
• Una vez aplicados los cambios, las aplicaciones continúen operando correctamente como lo hacian antes de realizar las actividades
• Que las pruebas de comprobación del control sean exitosas

2.7.- Incrementar seguridad en secuencia inicial de TCP

Tiempo requerido
25 minutos

Pasos de preparación en Solaris 10
• Respaldar el archivo /etc/default/inetinit
`#cp /etc/default/inetinit /etc/ default/inetinit.<dia>`

Riesgos previstos
No hay impacto previsto

Proceso de establecer seguridad en secuencia de TCP en Solaris 10
Cambiar el tipo de asignación de puertos para la secuencia de TCP en el archivo /etc/default/inetinit:
Cambia la línea:
`TCP_STRONG_ISS=1 por TCP_STRONG_ISS=2`

Comprobación
Solamente dar un more al archivo /etc/default/inetinit y buscar la línea del tcp. Hay que verificar que los valores estén cambiados como se indico, que el valor sea de "2" en vez de "1".

Actividades en caso de fallo para Solaris 10
Regresar el archivo original:
`#cp /etc/default/inetinit.<dia> /etc/default/inetinit`

2.8.- Protección de parámetros de red

Para esta parte vale la pena comentar que ambos sistemas operativos, tanto Sun Solaris 10, como SLES 10 han experimentado varias mejoras, equilibrando el balance entre la administración y la seguridad de los sistemas. Existe una clara división entre la seguridad de red en las capas de alto nivel, como NFS, NIS, NIS+, RPC, DNS, etc. y otros niveles de servicio. La aplicación de la mayoría de estos parámetros de seguridad para la red requieren planeación y pruebas.

En el caso de Solaris 10, la mayoría de los parámetros de red se asignan por medio del comando ***ndd***, así como también es usado para análizar y asignar los parámetros de kernel relacionados con los controladores de TCP/IP. La mayoría de los parámetros de kernel accesibles a través de ***ndd*** pueden se modificados sin reiniciar el sistema, para ver cuales parámetros estan disponibles, vea lo siguiente:

```
# ndd /dev/arp \?
# ndd /dev/icmp \?
# ndd /dev/ip \?
# ndd /dev/tcp \?
# ndd /dev/udp \?
```

Estos comandos listan los parámetros pára ARP, ICMP, IP, TCP, y UDP. Los parámetros de red que se asignan con el comando ***ndd*** se aplican a la instancia de manera inmediata, el único inconveniente es que los valores de estos parámetros no persisten tras los reinicios al sistema, una vez que el sistema es reiniciado toma los valores por defecto. Para proveer un metodo simple de asignar estos parámetros de red al momento del inicio, es necesaria la creación del script de sistema ***init***.

ARP
El Address Resolution Protocol es usado en direcciones Ipv4 de 32-bit para direccionar en el esquema de la capa de enlace. Particularmente los dispositivos de red de Sun usan una dirección de hardware system-wide, algunas veces referida como MAC (Media Access Control). En esta capa se pueden presentar varios ataques frecuentes como un DoS (Denial of Service).

ICMP
El protocolo Internet Control Message Procotol provee un mecanismo para reportar errores y hacer peticiones de cierta información. En esta capa se presentan diversos ataques basados en broadcast, el cuál basicamente funciona por medio de paquetes *icmp echo request y reply*. Todos los parámetros de configuración discutidos aquí son administrados por el controlador IP.

IP
El protocolo de Internet es un protocolo de un nivel mas bajo que provee el transporte de los datos, es un protocolo no orientado a conexión y no hace provisiones para una entrega confiable. Para este protocolo existen ataques tan letales, tales como habilitación dinámica de ruteo, provocando que se hagan saltos incorrectos. Los parámetros de configuración en cuestión son administrados y controlados por el controlador de IP.

TCP
El protocolo de Control de Transmisión es un protocolo orientado a la conexión, y provee una capa confiable de transporte de datos. Usa un protocolo de mas bajo nivel como IP para administrar la entrega de datagramas. Este maneja la conexión y la entrega de datos confiable y es susceptible a diversas formas de ataque como la inundación por paquetes SYN, en que de acuerdo a la conexión 3 vías, un cliente envia un segmento TCP a un servidor con una bandera SYN colocada en el encabezado, posteriormente el servidor regresa al cliente una bandera de SYN de acknowledgement ACK, y finalmente el cliente regresa un segmento con una bandera ACK. Esta comunicación esta basada en los números de sequencia, los cuales pueden ser robados y sustituidos.

Tiempo requerido
30 minutos, ya que requiere reiniciar el servidor

Riesgos previstos
No hay impacto previsto

Pasos de preparación en Solaris 10
• Guardar un respaldo de los valores originales. Para esto ejecutar los siguientes comandos:

```
#/usr/sbin/ndd /dev/ip ip_respond_to_echo_broadcast >valores_ndd.txt
#/usr/sbin/ndd /dev/ip ip_forward_directed_broadcasts >> valores_ndd.txt
#/usr/sbin/ndd /dev/ip ip_strict_dst_multihoming >> valores_ndd.txt
#/usr/sbin/ndd /dev/ip ip_ignore_redirect >> valores_ndd.txt
#/usr/sbin/ndd /dev/ip ip_forwarding >> valores_ndd.txt
#/usr/sbin/ndd /dev/ip ip_forward_src_routed >> valores_ndd.txt
#/usr/sbin/ndd /dev/ip ip_ire_arp_interval >> valores_ndd.txt
#/usr/sbin/ndd /dev/arp arp_cleanup_interval >> valores_ndd.txt
#/usr/sbin/ndd /dev/ip ip_respond_to_timestamp >> valores_ndd.txt
```

```
#/usr/sbin/ndd     /dev/ip     ip_respond_to_timestamp_broadcast     >>
valores_ndd.txt
```

Para cada interfaz de red se debe ejecutar los siguientes comandos:

```
# ndd -set /dev/hme instance 0
# ndd -set /dev/hme adv_100fdx_cap 1
# ndd -set /dev/hme adv_autoneg_cap 0
```

Pasos de preparación en SLES 10

• Guardar un respaldo de los valores originales. Para esto ejecutar los siguientes comandos:

```
#sysctl –a | grep –i net.ipv4.tcp_syncookies >> /tmp/valores_sysctl.txt
#sysctl –a | grep –i net.ipv4.conf.all.accept_source_route = 0 >>
/tmp/valores_sysctl.txt
#sysctl –a | grep –i net.ipv4.conf.all.accept_redirects = 0 >>
/tmp/valores_sysctl.txt
#sysctl –a | grep –i net.ipv4.conf.all.rp_filter = 1 >>
/tmp/valores_sysctl.txt
#sysctl –a | grep –i net.ipv4.icmp_echo_ignore_all = 1 >>
/tmp/valores_sysctl.txt
#sysctl –a | grep –i net.ipv4.icmp_echo_ignore_broadcasts = 1 >>
/tmp/valores_sysctl.txt
#sysctl –a | grep –i net.ipv4.icmp_ignore_bogus_error_responses = 1 >>
/tmp/valores_sysctl.txt
#sysctl –a | grep –i net.ipv4.conf.all.log_martians = 1 >>
/tmp/valores_sysctl.txt
```

Proceso para protección de parámetros de red en Solaris 10

Editar el archivo /etc/rc2.d/S99netconfig, agregar las siguientes líneas:

```
#!/bin/ksh
# Eliminar el bug por un ping broadcast
/usr/sbin/ndd –set /dev/ip ip_respond_to_echo_broadcast 0
# Bloquear los paquetes de broadcast hacia una red
/usr/sbin/ndd –set /dev/ip ip_forward_directed_broadcasts 0
# Prevenir Spoofing, DOS (redireccionar paquetes con error y solo de interfaces validas)
/usr/sbin/ndd –set /dev/ip ip_strict_dst_multihoming 1
/usr/sbin/ndd –set /dev/ip ip_ignore_redirect 1
# Prevenir un redireccionamiento de IP
/usr/sbin/ndd –set /dev/ip ip_forwarding 0
# Eliminar paquetes de enrutamiento
/usr/sbin/ndd –set /dev/ip ip_forward_src_routed 0
#Redefinir el tiempo de expiración de las tablas de ARP
/usr/sbin/ndd –set /dev/ip ip_ire_arp_interval 60000
/usr/sbin/ndd –set /dev/arp arp_cleanup_interval 60000
# Adicionales broadcast para sintonización de reloj y fecha (rdate) ICMP
/usr/sbin/ndd –set /dev/ip ip_respond_to_timestamp 0
/usr/sbin/ndd –set /dev/ip ip_respond_to_timestamp_broadcast 0
```

Comprobación en Solaris 10

• Tener privilegios de administrador.

• Teclear por cada uno de los servicios comentados ndd /dev/ip o ndd /dev/arp y el servicio.

Ej. ndd /dev/ip ip_respond_to_echo_broadcast en éste caso se pide que el valor que regrese debe arrojar es “0” o ndd /dev/arp arp_cleanup_interval en éste otro caso el

valor que se pide que regrese es “60”. En dado caso de no ser así ejecutar nuevamente el procedimiento para la línea especifica.

Proceso para protección de parámetros de red en SLES 10
El proceso consiste en editar el archivo /etc/sysctl.conf, agregar las siguientes líneas:

#Habilitar la protección de paquetes TCP SYN basados en cookies
net.ipv4.tcp_syncookies = 1
Deshabilitar el ruteo basado en dirección IP
net.ipv4.conf.all.accept_source_route = 0
#Deshabilitar la aceptación de paquetes ICMP Redirect
net.ipv4.conf.all.accept_redirects = 0
#Habilitar protección para evitar IP spoofing
net.ipv4.conf.all.rp_filter = 1
#Habilitar el parámetro para ignorar las peticiones ICMP Requests
net.ipv4.icmp_echo_ignore_all = 1
#Ignorar las peticiones de Broadcast Request
net.ipv4.icmp_echo_ignore_broadcasts = 1
#Habilitar la protección para mensajes de error
net.ipv4.icmp_ignore_bogus_error_responses = 1
#Habilitar el registro de paquetes
net.ipv4.conf.all.log_martians = 1

Comprobación en SLES 10
• Tener privilegios de administrador.
• Teclear el siguiente comando y fitrar cada uno de los parámetros comentados en el punto anterior.
Ej. `#sysctl –a | grep –i net.ipv4.tcp_syncookies`

Actividades en caso de fallo para Solaris 10
Regresar los valores originales, para esto ejecutar el siguiente comando para cada parámetro que se desee retornar:
`#/usr/sbin/ndd –set <dispositivo> <parámetro>=<valor>`
Por ejemplo:
`#/usr/sbin/ndd –set /dev/ip ip_respond_to_echo_broadcast 0`

Actividades en caso de fallo para SLES 10
Regresar los valores originales, para esto ejecutar el siguiente comando para cada parámetro que se desee retornar:
`#sysctl -w <parámetro>=<valor>`
Por ejemplo:
`#sysctl –w net.ipv4.tcp_syncookies = 0`

Criterios de aceptación
• Se realicen los cambios planeados
• Una vez aplicado los cambios, el servidor continué operando correctamente como lo hacia antes de realizar las actividades
• Una vez aplicado los cambios, las aplicaciones continúen operando correctamente como lo hacia antes de realizar las actividades
• Que las pruebas de comprobación del control sean exitosas

3.- Seguridad en Filesystems y Archivos del Sistema

Para esta sección se inicia trabajando con la parte referente a las auditorías. Dicha opción de auditoría provee una alta flexiilidad y un mecanismo extensible para evaluar el estado de un sistema.

Tipicamente la mayoría de los usuarios encuentran scripts de auditoría que se pueden personalizar para auditar sus ambientes. Ocasionalmente, algunos usuarios encuentran necesario agregar revisiones de funcionalidad de seguridad del sistema operativo.

Para los 2 sistemas operativos que conforman la solución del dominio seguro, se cuenta con un módulo de auditoría, que en el caso de SLES 10 se activa a través del sistema de control de autitoría y en el caso de Solaris, este provee el Basic Security Module (BSM) para auditar las acciones de los usuarios. Cabe mencionar que existe un pequeño impacto al rendimiento asociado con el uso de estos modulos.

3.1.- Configuración de Auditorías

Tiempo Requerido
45 minutos

Riesgos Previstos
Una ligera afectación al rendimiento de los sistemas operativos, Solaris 10 y SLES 10.

Pasos de preparación para Solaris 10
Ejecutar el archivo ***# /etc/security/bsmconv,*** el cuál agrega la siguiente entrada ***"set abort_enable = 0"*** al archivo /etc/system. Posteriormente ejecutar el siguiente comando:

```
# /usr/sbin/shutdown –i6 –g0 –y
```

Pasos de preparación para SLES 10
Respaldar los archivos ***# /etc/audit/auditd.conf y #/etc/sysconfig/auditd*** a través de los siguientes comandos:

```
# cp /etc/audit/auditd.conf /etc/audit/auditd.conf.<dia>
# cp /etc/sysconfig/auditd /etc/sysconfig/auditd.<dia>
```

Proceso de Activación en Solaris 10
Clases configuradas para eventos de log:

```
# vi /etc/security/audit_control
dir:/var/audit
flags:lo,ad,pc,fc,fd,fm
naflags:lo,ad
#
#   lo - login/logout events
#   ad - administrative actions: mount, exportfs, etc.
#   pc - process operations: fork, exec, exit, etc.
#   fc - file creation
#   fd - file deletion
#   fm - change of object attributes: chown, flock, etc.
```

Auditar todas las acciones hechas por root:

```
# vi /etc/security/audit_user
# log all of the commands that the root user runs
root:lo,ex:
```

Instalar un script de rotación de log.

```
# touch /etc/security/newauditlog.sh
# chmod 700 /etc/security/newauditlog.sh
# mkdir -p /var/audit/logs
# vi /etc/security/newauditlog.sh
#!/bin/ksh

# Solaris Basic Security Module (BSM) Log Rotation Script
# newauditlog.sh - Start a new audit file and expire the old logs
#
# Source: Solaris Security Guide
# Modifications: Added log compression and deletion with e-mail
# notification when the log directory grows past a certain size.
# - gtr
#
#****************************************************************
PATH=/usr/sbin:/usr/bin
AUDIT_EXPIRE=30
AUDIT_DIR=/var/audit
LOG_DIR=/var/audit/logs
# Rotate the audit log
/usr/sbin/audit -n
# Move log files to the archive directory and compress
for i in `/usr/bin/ls $AUDIT_DIR | grep -v not_terminated | grep -v
logs`
do
compress $AUDIT_DIR/$i
mv $AUDIT_DIR/$i.Z $LOG_DIR/$i.Z
done
# Delete old log files
cd $LOG_DIR # in case it is a link
/usr/bin/find . $LOG_DIR -type f -mtime +$AUDIT_EXPIRE -exec rm {} >
/dev/null 2>&1 \;
# Ensure that log files do not take up more than 250MB
# The next variable can be set for multiple addresses
# (i.e. jsmith@yahoo.com,jsmith@hotmail.com)
MAILADD=status
# The maximum size $OUTPUTDIR is allowed to reach before log files
# are deleted. (250000=250MB)
MAXSIZ=250000
LOGDU=`du -sk $LOG_DIR | awk '{ print $1 }`
if [ "$LOGDU" -gt "$MAXSIZ" ]; then
          find $LOG_DIR -mtime +21 -exec rm {} \;
          mail $MAILADD <<EOF
From: $0
To: $MAILADD
Subject: Security Audit Log Size on `uname -n`
$LOG_DIR was $LOGDU KB. $0 does not
allow more than 250 MB of log files in this directory.
Log files older than 21 days have been deleted.
The current size of $LOG_DIR is `du -sk $LOG_DIR | awk '{ print $1 }`
KB.
Thank you.
EOF
fi
exit 0
```

Proceso de Activación en SLES 10

Ejecutar el siguiente comando para editar uno de los archivos de configuración del sistema de auditorías:
#vi /etc/sysconfig/auditd
Colocar el siguiente valor:
AUDITD_DISABLE_CONTEXTS = no

Posteriormente ejecutar el siguiente comando para activar el servicio de auditoría:
 # auditctl –e 1

Comprobación para Solaris 10
Se deben generar los archivos de bitácora bajo la ruta ***/var/audit***, además de que se puede convertir el archivo de datos en formato binario a formato ASCII con el siguiente comando:

```
# praudit logfile
```

Comprobación para SLES 10
Se deben generar los archivos de bitácora bajo la ruta ***/var/log/audit***.

Actividades en caso de fallo
Regresar a la configuración inicial y deshabilitar el servicio

Criterios de aceptación
- En el momento en que se empiecen a generar las bitácoras y sean circulares.
- El demonio debe estar ejecutandose de manera continua.

3.2.- Configuración del Banner de login

Tiempo requerido
25 minutos

Riesgos previstos
No hay impacto previsto

Pasos de preparación para Solaris 10 y SLES 10
• Respaldar el archivo /etc/issue

Proceso de activación para Solaris 10 y SLES 10
• Se deben de crear banners de advertencia al ingresar al servidor para los servicios que utilizan Login

Ejemplo:
#vi /etc/issue
SISTEMA PRIVADO DEL BANCO. UNICAMENTE USUARIOS AUTORIZADOS
#chown root:root /etc/issue
#chmod 644 /etc/issue
y para sshd cambie la siguiente línea en el archivo /usr/local/etc/sshd_config
#Banner /some/file por Banner /etc/issue
En general los banners se encuentran en:
/etc/issue
/etc/motd
/etc/default/telnetd

Estos se pueden sustituir con con algún mensaje de advertencia:
```
BANNER="Acceso solo a usuarios autorizados del Banco. Los accesos son monitoreados \n"
```
Los permisos para el usuario y el grupo para este archivo deben ser root : sys con los permisos de solo lectura.

Comprobación
Abrir una nueva sesión de telnet al servidor. Se debe observar el texto deseado, antes del texto que solicita el login

Actividades en caso de fallo
Regresar el archivo /etc/issue:
```
#cp /etc/issue.<dia> /etc/issue
```

Criterios de aceptación
• Se realicen los cambios planeados
• Una vez aplicados los cambios, el servidor continué operando correctamente como lo hacia antes de realizar las actividades
• Una vez aplicados los cambios, las aplicaciones continúen operando correctamente como lo hacian antes de realizar las actividades
• Que las pruebas de comprobación del control sean exitosas

3.3.- Permisos de archivos de configuración

Tiempo requerido
25 minutos

Pasos de preparación en Solaris 10 y SLES 10
• Ejecutar el siguiente comando:
```
#ls -ld /etc/passwd /etc/group /etc/shadow > permisos.txt
```

Riesgos previstos
No hay impacto previsto

Proceso para establecer permisos de archivos de configuración en Solaris 10 y SLES 10
Se recomienda poner los permisos de solo lectura para el archivo /etc/passwd y /etc/group para todos y permisos de solo lectura para el archivo /etc/shadow para el administrador. Para cambiar los permisos se ejecutan los siguientes comandos:
Para el /etc/passwd y el /etc/group:
```
#chmod 444 /etc/passwd /etc/group
```
Para el /etc/shadow
```
#chmod 400 /etc/shadow;
```

Comprobación en Solaris 10 y SLES 10
Dar un ls –l a los archivos para verificar sus permisos:
```
#ls -l /etc/passwd /etc/group
-rw-r--r-- 1 root sys 441 Aug 25 11:41 /etc/group
-r--r--r-- 1 root other 781 Aug 30 11:26 /etc/passwd
#ls -l /etc/shadow
-r-------- 1 root root 505 Sep 5 09:14 /etc/shadow
```
En dado caso que no tenga los permisos permitidos cambiarlos.

Actividades en caso de fallo en Solaris 10 y SLES 10
Regresar los permisos originales con el siguiente comando
```
#chmod <permiso> <archivo>
```
Por ejemplo:
```
#chmod 444 /etc/passwd
```

Criterios de aceptación
• Se realicen los cambios planeados
• Una vez aplicados los cambios, el servidor continué operando correctamente como lo hacia antes de realizar las actividades
• Una vez aplicados los cambios, las aplicaciones continúen operando correctamente como lo hacian antes de realizar las actividades
• Que las pruebas de comprobación del control sean exitosas

3.4.- Protección de la Pila

Tiempo requerido
30 minutos, ya que requiere reiniciar el servidor

Pasos de preparación en Solaris 10
• Respaldar el archivo /etc/system
```
#cp /etc/system /etc/system.<dia>
```

Pasos de preparación en SLES 10
• Respaldar el archivo /etc/sysctl.conf
```
#cp /etc/sysctl.conf /etc/sysctl.conf.<dia>
```

Riesgos previstos
No hay impacto previsto

Proceso de protección del stack en Solaris 10
Esta protección es requerida para evitar problemas de desbordamiento de pila o buffer overflow. Para evitarlo tenemos que editar el /etc/system y debemos de agregar las siguientes líneas o en su defecto realizar el siguiente procedimiento desde el prompt de root.
Ejecutar:
```
#echo " " >> /etc/system
#echo "***** Parametros para evitar overflow *****" >> /etc/system
#echo "set noexec_user_stack=1 " >> /etc/system
#echo "set noexec_user_stack_log=1 " >> /etc/system
```

Proceso de protección del stack en SLES 10
Esta protección es requerida para evitar problemas de desbordamiento de pila o buffer overflow. Para evitarlo tenemos que editar el /etc/sysctl.conf y debemos de agregar la siguiente línea o en su defecto realizar el siguiente procedimiento desde el prompt de root.
Ejecutar:
```
#echo " " >> /etc/sysctl.conf
#echo "***** Parametros para evitar overflow *****" >> /etc/sysctl.conf
#echo "kernel.randomize_va_space = 0" >> /etc/sysctl.conf
```

Comprobación para Solaris 10
Para poder revisarlo solamente debemos de dar un more al archivo /etc/sytem o un grep stack /etc/system y verificar que los valores estén cambiados como se indico anteriormente.

Comprobación para SLES 10
Para poder revisarlo solamente debemos de dar un more al archivo /etc/sysctl.conf y verificar que los valores estén cambiados como se indico anteriormente.

Actividades en caso de fallo para Solaris 10
Regresar el archivo original:
```
#cp /etc/system.<dia> /etc/system
```

Actividades en caso de fallo para SLES 10
Regresar el archivo original:
```
#cp /etc/sysctl.conf.<dia> /etc/sysctl.conf
```

Criterios de aceptación
• Una vez aplicados los cambios, el servidor continué operando correctamente como lo hacia antes de realizar las actividades.
• Una vez aplicados los cambios, las aplicaciones continúen operando correctamente como lo hacian antes de realizar las actividades
• Que las pruebas de comprobación del control sean exitosas.

3.5.- Aseguramiento de los Filesystems

Tiempo Requerido
25 minutos, ya que es recomendable reiniciar el servidor

Pasos de preparación en Solaris 10
Respaldar el archivo /etc/vfstab
```
# cp /etc/vfstab /etc/vfstab.old
```

Pasos de preparación en SLES 10
Respaldar el archivo /etc/fstab
```
# cp /etc/fstab /etc/fstab.old
```

Riesgos Previstos
No hay impacto previsto

Proceso de activación en Solaris 10
Para activar un aseguramiento de los filesystems, es necesario modificar el archivo /etc/vfstab indicando que el filesystem /*usr* como solo lectura, quedando de la siguiente manera.

```
/dev/md/dsk/d3    /dev/md/rdsk/d3   /usr  ufs   1     no    ro
```

Proceso de activación en SLES 10

Para activar un aseguramiento de los filesystems, es necesario modificar el archivo /etc/fstab indicando que el filesystem */usr* como solo lectura, quedando de la siguiente manera.

```
# device name     mount point fs-type      options    dump-freq pass-num
LABEL=/usr        /usr        ext4        ro          1 1
```

Actividades en caso de fallo en Solaris 10
Regresar el archivo original
```
#cp /etc/vfstab.orig /etc/vfstab
```

Actividades en caso de fallo en SLES 10
Regresar el archivo original
```
#cp /etc/fstab.orig /etc/fstab
```

Criterios de aceptación
• Se debe ejecutar el comando *#mount*
• Se debe ejecutar el comando df -h para validar como ha montado cada uno de los filesystems al momento de boot.

3.6.- Aseguramiento de EEPROM - Hardware

Solo aplica para los servidores SPARC
Tiempo Requerido
10 minutos

Pasos de preparación
Validar la salida de la variable security-mode de la eeprom

Riesgos Previstos
Es importante considerar que ante la perdida de la contraseña que asignaremos a la eeprom, no existe forma alguna de recuperar el sistema, ya que la contraseña se almacena en un firmware que es imposible resetear.

Proceso de activación
Para activar un nivel de seguridad a nivel de eeprom unicamente es necesario dar el siguiente comando dentro de una consola de sistema operativo Solaris 10.
```
# eeprom security-mode=command
```

Después de esto el sistema nos pedira una contraseña, la cual será necesaria para ejecutar algunas tareas de administración a nivel de prompt de ok

Actividades en caso de fallo
Es importante especial atención a la contraseña ingresada, ya que es un **procedimiento irreversible**

Criterios de aceptación
• Se debe reiniciar el sistema y ejecutar un boot para ingresar la contraseña solicitada y todo debe funcionar normal

3.7.- Sequencia de Abort - Hardware

Solo aplica para los servidores SPARC
Tiempo Requerido
10 minutos

Pasos de preparación
No es necesaria ninguna tarea previa a esta actividad

Riesgos Previstos
No se contempla ningún impacto desfavorable

Proceso de activación
Para deshabilitar la sequencia de abort, la cuál es la configuración por default se tiene que editar el archivo ***/etc/default/kbd*** y cambiar una bandera de configuración como se muestra en seguida:

```
# vi /etc/default/kbd

#KEYBOARD_ABORT affects the default behavior ot the keyboard abort
#sequence, see kbd(1) for details. The default value is # “enable”. The
optional value is “disable”

#KEYBOARD_ABORT=enable
KEYBOARD_ABORT=disable
```

Posteriormente actualice el sistema:

```
# kbd –i
```

Actividades en caso de fallo
Regresar el valor de la variable al valor por defecto

Criterios de aceptación
• Se debe mandar una sequencia de abort, mediante la combinación de teclas “Stop” + A sin que se presente evento alguno

Notas

- Es importante aclarar que el alcance definido anteriormente se encontraba alineado con el plan de trabajo.
- Las actividades necesarias para cada uno de los ambientes no eran secuenciales, esto es, que podían ser ejecutadas de manera simultánea en 2 ó mas instancias de sistema operativo, y en un orden indistinto.
- El tiempo podía ser optimizado y reducido considerablemente **si se cumplían todos los supuestos y premisas mencionados en el cuerpo principal del trabajo en la sección correspondiente a requisitos.**

ANEXO C.- DESCRIPCION DE COMPONENTES DE LA SOLUCION DE DOMINIO SEGURO

1.- Componentes de Software

1.1.- Tivoli Access Manager for E-Business V6.1.1

IBM Tivoli Access Manager for e-business provee una plataforma de administración de seguridad integrada para servicios de autenticación, control de acceso o servicios de autorización, mapeo de identidad, *Web Single Sign-On,* autorización y derechos, y servicios de auditoria sobre los recursos empresariales. Dicha plataforma provee administración de la seguridad integrada y esta basada en políticas de clase empresarial, para que los clientes, socios de negocio, empleados, proveedores, y distribuidores accedan de forma segura a los recursos de un portal empresarial de una manera confiable.

En realidad *Tivoli Access Manager* es una solución de autenticación y autorización para aplicaciones web y cliente/servidor, ya que permite controlar el acceso de los usuarios hacia la información protegida y recursos propios de las aplicaciones. Debido a que *Tivoli Access Manager* provee una solución de control de accesos centralizada, flexible, y escalable es posible construir aplicaciones basadas en red seguras y fácil de administrar.

Tivoli Access Manager soporta autenticación, autorización, seguridad de datos, y capacidad de administración de recursos. Para ello, el producto se basa 2 *frameworks:*

- *Framework* de autenticación.- Es propiamente un servicio de autenticación que usa una amplia gama de métodos de autenticación integrados, al igual que soporta elementos de autenticación externos.
- *Framework* de autorización.- Es un propiamente un servicio de autorización accedido a través de una interfaz de programación de aplicaciones, mejor conocida como *API* por sus siglas en inglés, de autorización estándar, la cuál provee decisiones para permitir o denegar el acceso hacia otras aplicaciones.

El servicio de autorización, en conjunto con el administrador de recursos, provee un mecanismo de autorización estándar para diferentes aplicaciones.

El administrador de recursos que usa comúnmente *Tivoli Access Manager* es el *IBM Tivoli Access Manager WebSEAL*, el cuál se encarga de administrar y proteger la información y los recursos basados en *web.*

La autenticación es el primer paso que debe dar un usuario cuando hace una solicitud sobre un recurso que se encuentra protegido por *Tivoli Access Manager*. Durante el proceso de autenticación la identidad de un usuario es validada, y dicho proceso es usualmente dependiente de requerimientos específicos de la aplicación que provee el servicio. *Tivoli Access Manager* provee soporte integrado para el método de autenticación basado nombre de usuario y contraseña a través de la *API* de autorización, además de que las aplicaciones pueden utilizar cualquier mecanismo de autenticación personalizado que use dicha *API* de autorización.

Por otro lado el proceso de autorización refuerza la política de seguridad al determinar que objetos pueden ser accedidos por un usuario y que acciones puede tomar este sobre dichos objetos. *Tivoli Access Manager* maneja el proceso de autorización a través de los siguientes elementos:

- Servicio de autorización de Tivoli Access Manager.
- Listas de control de acceso, conocidas como *ACL.*
- Políticas de objetos protegidos, conocidos como *POP.*
- *APIs* de autorización basadas en estándares, tales como *aznAPI* para aplicaciones desarrolladas en lenguaje *C,* y *JAAS* para aplicaciones desarrolladas en lenguaje de *Java.*
- Capacidad de servicio de autorización externo.

Tivoli Access Manager provee algoritmos de cifrado que son proveídos por *GSKit* y *Java Secure Socket Extension,* además de que soporta nativamente los protocolos *Secure Sockets Layer (SSL)* y *Transport Layer Security (TLS)* para proveer integridad y privacidad de los datos. Por un lado *SSL,* se utiliza principalmente la llave pública para autenticar y la llave secreta para cifrar los datos que son transferidos sobre el túnel de conexión de *SSL,* y en el caso del protocolo *TLS* se cumple con el estándar conocido como *FIPS 140-2* que por sus siglas en inglés significa *Federal Information Processing Standard.* Específicamente las versiones utilizadas en cada uno de los protocolos son *SSLv3* y *TLSv1.* Cabe señalar que no es posible combinar ambos protocolos *SSL* y *TLS* dentro de un ambiente de *Tivoli Access Manager*.

El producto *Tivoli Access Manager* en su versión 6.1.1 cuenta con la escalabilidad requerida para responder al incremento de usuarios que acceden a los recursos dentro de un dominio seguro, y para ello utiliza las siguientes técnicas que proveen dicha escalabilidad:

- Servicios de Replicación
  - Servicios de Autenticación
  - Servicios de Autorización
  - Políticas de Seguridad
  - Servicios de Cifrado de Datos
  - Servicios de Auditoría
- Servicios de Replicación de *Front-End*
  - Recursos espejeados para alta disponibilidad
  - Balanceo de Cardas en las peticiones realizadas
- Servicios de Replicación de *Back-End*
  - Servidores de *Back-End*
  - Integración de Negocio con otros servidores de aplicaciones
  - Recursos espejeados, tal es el caso del espacio de objetos protegido
  - Contenido Adicional y recursos
  - Balanceo de cargas de peticiones entrantes
- Rendimiento Optimizado al permitir que los servicios de autenticación y los servicios de autorización se ejecuten en servidores separados
- Despliegue escalado de servicios sin incrementar la carga de administración

Tivoli Access Manager provee una gran gama de capacidades de registro y auditoría. Los archivos de bitácora capturan cualquier mensaje de advertencias y errores

generados por los servidores de *Tivoli Access Manager,* debido a la actividad del mismo.

Tivoli Access Manager cuenta con un sistema de administración centralizada, usando los siguientes métodos proporcionados la política de administración de la seguridad dentro de los servidores:

- Interface de línea de comando, llamada *pdadmin*
- Interface gráfica de usuario *(GUI),* llamada *Web Portal Manager*
- *API* de administración

La mayoría de las tareas de administración se pueden ejecutar a través de cualquiera de los tres métodos.

Con respecto a la política de seguridad para *Tivoli Access Manager*, el objetivo es proteger adecuadamente los recursos y activos del negocio con un mínimo de esfuerzo al momento de administrar dicha política, para lo cuál es importante definir que recursos específicos requieren ser protegidos, y estos podrían ser cualquier tipo de objeto, tales como archivos, directorios, servidores en la red, mensajes, bases de datos, o páginas *web.* Posteriormente se tiene que decidir que usuarios y/o grupos de usuarios deberían tener acceso a dichos recursos, y que tipo de acceso se les debe asignar a los mismos.

Para mayor detalle véase el sitio oficial del fabricante en la siguiente liga de internet, http://pic.dhe.ibm.com/infocenter/tivihelp/v2r1/index.jsp?topic=%2Fcom.ibm.itame.doc_6.1.1%2Fwelcome.htm.

1.2.- Sun Java System Directory Server Enterprise Edition 7.0

El producto *Sun Java System Directory Server* almacena los datos en bases de datos personalizadas de árboles binarios, lo cuál permite realizar búsquedas rápidas, incluso para grandes conjuntos de datos.
Los directorios son bases de datos orientadas a objetos. Los directorios organizan los datos de objetos, conocidos como entradas, en un árbol de información de directorios, a menudo llamado el *DIT*. Cada entrada es identificada por un nombre distinguido. El nombre distintivo identifica donde se encuentra localizada la entrada dentro del árbol de información de directorios.
Cada entrada de directorio tiene ciertos atributos. Para las entradas que hacen referencia a la gente, estos atributos pueden referirse al nombre, números de teléfono, y direcciones de correo electrónico, entre otros. Un atributo tiene al menos un nombre de tipo, que en realidad es el nombre del atributo. Por ejemplo, las entradas de las personas pueden tener un atributo de apellido, que también puede ser llamado por el nombre más corto o *sn*. Cabe señalar que los atributos también pueden tener uno o más valores.
Los directorios están diseñados para ser muy rápidos al momento de buscar entradas en función de los valores de sus atributos. Una consulta de ejemplo podría ser: "Encuentra todas las entradas bajo *dc = example, dc = com* con el apellido Jensen." Esta capacidad de búsqueda rápida hace que los directorios sean adecuados para aplicaciones en las que se almacena la información que debe ser leída con mayor frecuencia. Los directorios se posicionan como una muy buena opción para almacenar información de personas, así como también son buena opción para manejar

credenciales de autenticación, información de identidad, y datos de configuración de aplicaciones.
Sun Java System Directory Server también cuenta con un buen diseño para manejar actualizaciones de datos sobre las entradas del directorio a altas velocidades. Hoy en día, el tamaño de muchas implementaciones de directorio implica que las actualizaciones de manipulación de información pueden ser tan importantes como el manejo de búsquedas.
Sun Java System Directory Server soporta muchos estándares y *RFC* relacionados con directorio. *Sun Java System Directory Server* permite una rápida replicación de datos a través de la red de alta disponibilidad.
La lista de características de *Sun Java System Directory Server* es demasiado larga para ser cubierta en este breve anexo, para lo cuál se recomienda Sun Directory Server Enterprise Edition 7.0 Guía de evaluación incluye una lista más extensa. Los otros capítulos de esta parte de esta referencia ayudan a entender muchas de las características en detalle.
Para cada instalación del software de *Sun Java System Directory Server*, se pueden crear varias instancias de servidor, y además todas pueden ser alojadas en el mismo sistema de archivos donde se instala el software.
El software *Sun Java System Directory Server* incluye los archivos ejecutables, los datos de la plantilla, y los archivos de muestra necesarios para crear, ejecutar, y administrar los servidores actuales. A medida que el software se hace independiente de los servidores reales, se pueden aplicar parches o *service packs* al software sin cambiar los datos del servidor. Así que no es necesario aplicar parches en cada instancia de servidor, por el contrario, sólo se aplican a una instalación del software.
Una instancia de servidor de directorio mantiene los datos de configuración y los datos de los directorios necesarios para servir a las aplicaciones cliente del directorio. Aunque en los sistemas de producción se controla cuidadosamente la identidad del usuario en el servidor, normalmente se puede crear y ejecutar una instancia de *Sun Java System Directory Server* como un usuario cualquiera dentro del sistema. Debido a lo anterior, los datos del directorio pertenecen al usuario que creó la instancia.

Para mayor detalle véase el sitio oficial del fabricante a través de la siguiente liga de internet, http://www.oracle.com/technetwork/middleware/id-mgmt/overview/index-085178.html.

1.3.- Sistemas Operativos

1.3.1.- Sun Solaris 10

Oracle Solaris 10 ha sido la plataforma estratégica para las aplicaciones empresariales demandantes desde el año 2005, entregando desde entonces resultados probados en todo tipo de aplicaciones, desde bases de datos empresariales de misión crítica hasta granjas de servidores *Web* de alto rendimiento, abarcando desde las últimas generaciones de sistemas tanto *SPARC*, como *x86*. Para los clientes que enfrentan retos de negocio y requerimientos técnicos, tales como la reducción de costos, simplificar la administración de los sistemas y conservar altos niveles de servicio, *Oracle Solaris 10* es la elección ideal multiplataforma.
Oracle Solaris 10 ofrece la solución de virtualización eficaz e integrada más rentable de la industria que abarca una amplia gama de tecnologías, así como ofertas de gestión y

apoyo sin costos adicionales. Dichas tecnologías de virtualización incluyen tanto *Oracle VM Server* y *Oracle Solaris Containers*. *Oracle VM Server* sigue el modelo de la industria para la colocación de ambientes *guest* independientes sobre una arquitectura de hipervisor que soporta la migración en vivo de los ambientes *guest* entre los diferentes sistemas. *Oracle Solaris Containers* provee un enfoque diferente al proporcionar entornos virtuales independientes y aislados dentro de una única instancia del sistema operativo *Solaris 10*.
Oracle Solaris garantiza la compatibilidad binaria entre versiones y su arquitectura de interfaces de programación neutral que aseguran la compatibilidad del código fuente entre sistemas *SPARC* y *x86*. *Oracle Solaris Containers* cuentan con la compatibilidad requerida para ayudar a colocar aplicaciones heredadas que se ejecutan sobre *Solaris 8* y *Solaris 9* sobre *Oracle Solaris Containers* en *Oracle Solaris 10*. La ejecución de aplicaciones sobre *Oracle Solaris Containers* en conjunto con *Oracle VM* es la forma más eficaz de garantizar la compatibilidad de aplicaciones, al otorgar la capacidad para ejecutar aplicaciones más viejas en hardware más reciente. Al agregar *Oracle Solaris 11* es posible consolidar 4 generaciones de *Oracle Solaris* en una sola plataforma de hardware.
Las características de seguridad de *Oracle Solaris 10* están a la vanguardia para ayudar reducir el riesgo de intrusiones, asegurar los datos de las aplicaciones, asignar el conjunto mínimo de funciones y privilegios que necesitan los usuarios y las aplicaciones, y controlar el acceso a los datos en función de su etiqueta de sensibilidad. *Oracle Solaris 10* ha sido evaluado independientemente en *EAL4 +* sobre tres perfiles de protección, uno de los más altos niveles de las certificaciones *Common Criteria*. La administración de usuarios de *Oracle Solaris*, así como el proceso de gestión de derechos y la tecnología de *Oracle Solaris Containers,* en conjunto permiten que cientos de aplicaciones y múltiples clientes sean alojados en el mismo sistema. Además, *Oracle Solaris Trusted Extensions* proporciona una verdadera seguridad multinivel para el primer sistema operativo de calidad comercial, donde todas las aplicaciones existentes pueden funcionar sin modificación alguna.
Con el sistema operativo *Oracle Solaris 10,* se pueden obtener los siguientes beneficios relacionados con la seguridad:

• Reducir el riesgo mediante la concesión sólo de los privilegios necesarios para los usuarios y los procesos requeridos.
• Simplificar la administración y aumentar la privacidad y el rendimiento mediante la gestión de claves basada en estándares y *frameworks* de cifrado en *Oracle Solaris.*
• Asegurar el sistema usando perfiles de servicios dinámicos, incluyendo la reducción de exposición de perfiles de servicios en la red.
• Controlar el acceso a los datos en función de su nivel de sensibilidad mediante el uso de tecnologías de seguridad basadas en etiquetas en *Oracle Solaris Trusted Extensions*

Predictive Self Healing es una característica clave de *Oracle Solaris 10*, que ayuda a aumentar la disponibilidad del sistema y del servicio. Automáticamente detecta, diagnostica, y aísla las fallas del sistema y del software antes de que puedan provocar un tiempo de inactividad. Dicha funcionalidad controla todo el proceso, desde el diagnóstico hasta la recuperación de los sistemas *Oracle SPARC* y *x86*.
Oracle Solaris ZFS fue diseñado desde cero para proporcionar un sistema de archivos de propósito general, que incluyera un software administrador de volúmenes "sin costo adicional", el cuál provee completa protección de datos y una inmensa capacidad de escalabilidad. Los *ZFS snapshots* crean una baja carga de procesamiento, reduciendo el riesgo y disminuyendo el tiempo de inactividad para cualquier cambio en el software

de sistemas. La compresión de datos a través de *ZFS* reduce la cantidad de espacio en disco utilizado por el empleo de algoritmos de compresión seleccionables. Esto permite que el consumo de *CPU* requerido para la compresión resulte altamente eficiente.
La tecnología *Oracle Solaris Containers* permite consolidar múltiples cargas de trabajo en un solo sistema, *SPARC* o *x86*. Los contenedores son una tecnología de virtualización a nivel de sistema operativo integrado en *Oracle Solaris 10* que no representa un costo adicional. Usando límites flexibles definidos por software para aislar las aplicaciones y servicios. Cada ambiente tiene su propia identidad, completamente separada del hardware subyacente, de manera que funciona, como si se ejecutara en su propio sistema, creando una consolidación muy simple y segura. Las ventajas de los entornos virtuales que se ejecutan en un solo sistema operativo, es que se genera una carga de procesamiento casi nula, una distribución superior de los recursos, una mayor flexibilidad en la distribución de los recursos y una mayor flexibilidad en los límites del uso de recursos.
Los administradores de sistemas, integradores y desarrolladores pueden utilizar la instrumentación dinámica y las capacidades de rastreo de *Oracle Solaris* para ver lo que realmente está pasando en el sistema. *Oracle Solaris DTrace* es una tecnología que puede ser usada con plena seguridad en los sistemas productivos, sin modificar las aplicaciones. Es una herramienta de gran alcance con más de 30,000 pruebas dentro de *Oracle Solaris* que ofrece una amplia visibilidad de todo el sistema completo, desde el núcleo hasta las aplicaciones, incluso aquellos que se ejecutan en una *Java Virtual Machine*. Este nivel de penetración puede reducir el tiempo para el diagnóstico de problemas de días y semanas a minutos y horas, y en última instancia, reduce el tiempo necesario para solucionar los problemas.
Por otro lado, cabe mencionar que el crecimiento exponencial en la conectividad *Web*, servicios y aplicaciones está generando una necesidad crítica de un mayor rendimiento de la red. Con *Oracle Solaris 10*, se satisfacen los desafíos actuales y futuros al mejorar significativamente el rendimiento de la red sin requerir cambios en las aplicaciones existentes. *Oracle Solaris 10* acelera el rendimiento de las aplicaciones a través de la capa de red 7 Cache y del rendimiento mejorado de *TCP / IP* y de *UDP / IP*. Las últimas tecnologías de red, tales como 10 *Gigabit Ethernet* (GbE) y *hardware offloading*, están todas soportadas fuera de la caja.
Además, *Oracle Solaris 10* es compatible con las especificaciones actuales de *IPv6*, alta disponibilidad, *streaming,* y voz sobre IP (*VoIP*) a través de una red extendida de enrutamiento y soporte a protocolo de acuerdo a las necesidades del cliente.

Para mayor detalle véase el sitio oficial del fabricante a través de la siguiente liga de internet, http://www.oracle.com/us/products/servers-storage/solaris/overview/index.html.

1.3.2.- SUSE Linux Enterprise Server 10

SUSE Linux Enterprise Server es un sistema operativo altamente confiable, escalable, y seguro, construido para mejorar el rendimiento de las cargas de trabajo de misión crítica en entornos físicos, entornos virtuales, y aquellos basados en la nube. Debido a la fundación de código abierto, se puede otorgar interoperabilidad de una manera efectiva hacia los servicios del núcleo del negocio, permitiendo redes seguras y simplificando la infraestructura heterogénea, maximizando su eficiencia y su valor.
SUSE Linux Enterprise Server es un sistema operativo modular, de propósito general, que se ejecuta sobre cinco arquitecturas diferentes de procesador y es ideal para una amplia gama de cargas de trabajo. Además está optimizado para funcionar con los

principales *hypervisores* líderes del mercado y se permite la creación de un número ilimitado de máquinas virtuales huéspedes por sistema físico con una sola suscripción, convirtiéndose en una plataforma perfecta para el cómputo virtual.
SUSE Linux Enterprise Server ayuda a las organizaciones a maximizar la eficiencia y el valor del negocio, y al mismo tiempo a mitigar el riesgo, todo ello a través de los siguientes beneficios clave:

- Reducción de costos.- No sólo se reduce el costo asociado a la infraestructura de software en los servidores, sino también se ahorra dinero en hardware debido a que *SUSE Linux Enterprise Server* es compatible con los últimos estándares en la industria del hardware. Además de que se oferta bajo un modelo simplificado de subscripción, cuyo precio es "por servidor", con el cuál es posible escalar fácilmente bajo un enfoque de solo agregar capacidad de cómputo.
- Ahorro de tiempo.- *SUSE Linux Enterprise Server* incluye un paquete de administración, que permite actualizar los sistemas en cuestión de minutos en lugar de horas. Las herramientas de instalación intuitivas y los perfiles de pre-configuración permiten configurar rápidamente los servicios más populares, de manera local como remotamente. Las herramientas de administración eficiente ayudan a realizar un seguimiento y administración de los activos de software del sistema.
- Mitigar el riesgo.- Entrega confiable de una amplia variedad de servicios de misión crítica. *SUSE Linux Enterprise Server* es un sistema operativo de código abierto que incluye el código fuente y los binarios haciéndolo inherentemente más seguro que un sistema operativo propietario. Adicionalmente, la transparencia en los términos y condiciones de licenciamiento fomentan la innovación técnica, la libre elección, y la competencia en el mercado.
- Obtener lo que necesita.- *SUSE Linux Enterprise Server* cuenta con extensiones que proveen capacidades avanzadas, como son, habilitar un esquema de alta disponibilidad, *clustering*, funcionalidad para ejecutar aplicaciones *.NET* sobre *Linux*, y computación en tiempo real. El usuario paga solo por aquello que necesita, evitando extras innecesarios. Conforme la demanda crece o surge la necesidad de un cambio, es posible añadir fácilmente las capacidades adicionales.

En seguida se muestran las características más importantes para este sistema operativo:

- Sistemas de administración Integrados.- *SUSE Linux Enterprise Server* integra nativamente el subsistema de gestión de paquetes, llamado *ZYpp* dentro de la única herramienta de instalación, configuración y administración conocida como *YaST*® y *AutoYaST* que permiten una rápida instalación y configuración de los servicios, así como la aplicación de parches y actualizaciones del sistema.
- Fiabilidad, Disponibilidad y Facilidad de servicio.- *SUSE Linux Enterprise Server* admite memoria de tipo *swap* sobre *NFS* para aprovechar de mejor manera el almacenamiento remoto acorde a las necesidades del servidor local. También cuenta con el soporte para características de *RAS* sobre las últimas familias de procesadores *Intel* y *AMD*. También aprovecha las capacidades de *btrfs copy-on-write* y *snapshots* para mejorar la resistencia y la disponibilidad de los servicios

- Virtualización Cross-Platform.- *SUSE Linux Enterprise Server* incluye soporte comercial para las versiones más recientes de los *hipervisores Xen* y *KVM*, así mismo otorga un esquema de virtualización altamente eficiente y de bajo consumo de recursos.
- Interoperabilidad.- *SUSE Linux Enterprise Server* está diseñado para interoperar con *Windows* y otras plataformas operativas, por lo que es la opción ideal para ambientes heterogéneos.
- *Green IT*.- *SUSE Linux Enterprise Server* cuenta con características innovadoras para el ahorro de energía, como lo es *tickless idle* que permite obtener el máximo rendimiento por *watt* de consumo eléctrico, mediante el uso de perfiles granulares basados en unidades de consumo, *watt.*
- Seguridad Integral.- *SUSE Linux Enterprise Server* incluye un *framework* para aplicaciones de seguridad, conocido como *AppArmor* ®, así como otras características de seguridad esenciales, como *firewall*, *VPN*, y sistema de detección de intrusos. También incluye soporte para la especificación *Trusted Platform Module (TPM)* estándar.
- *Advanced Networking*.- *SUSE Linux Enterprise Server* soporta el protocolo IPv6, incluyendo *Open Fabrics Enterprise Distribution (OFED)* y soporte para *Fibre Channel over Ethernet (FCoE).* Así mismo se cuenta con el soporte para *Data Center Bridging (DCB)*, que se puede utilizar para ejecutar tráfico *SAN* y *LAN* sobre un mismo enlace.
- Cómputo de alto rendimiento.- Su gestión avanzada de memoria, y sus funciones de *multi-pathing* y E / S, en conjunto con el soporte de interconexiones de alta velocidad y sistemas de archivos *multi-threaded*, hacen de *SUSE Linux Enterprise Server* la opción preferida para el cómputo de alto rendimiento de hoy.

Para mayor detalle véase el sitio oficial del fabricante a través de la siguiente liga de internet, https://www.suse.com/products/server/.

2.- Componentes de Hardware

Es importante señalar que la parte correspondiente a la descripción de los componentes de hardware se hace directamente a través de las fichas técnicas de cada uno de los modelos de servidor utilizados, así como para el gabinete propuesto donde fueron montados los equipos. Adicionalmente, vale la pena mencionar que en los sitios de los fabricantes *Oracle* e *IBM*, se podrá obtener información con mayor detalle sobre dichos componentes.