UNIVERSIDAD NACIONAL AUTÓNOMA DE MÉXICO

FACULTAD DE CIENCIAS

ADMINISTRACIÓN DEL SISTEMA DE CORREO ELECTRÓNICO DE LOS ALUMNOS DE LA UNAM

REPORTE DE TRABAJO PROFESIONAL

Q U E P A R A O B T E N E R E L T Í T U L O DE

LICENCIADO EN CIENCIAS DE LA COMPUTACIÓN

P R E S E N T A :

RODRÍGUEZ GÓMEZ LUIS MIGUEL

TUTOR: MAT. FACUNDO RUÍZ DONCEL

CIUDAD UNIVERSITARIA JULIO 2007

1. Datos del alumno
    - Rodríguez
    - Gómez
    - Luis Miguel
    - Universidad Nacional Autónoma de México
    - Facultad de Ciencias
    - Ciencias de la Computación
    - 095325079
2. Datos del tutor
    - Mat.
    - Facundo
    - Ruíz
    - Doncel
3. Datos del sinodal 1
    - M. en C.
    - María Guadalupe Elena
    - Ibargüengoitia
    - González
4. Datos del sinodal 2
    - Dra.
    - Amparo
    - López
    - Gaona
5. Datos del sinodal 3
    - Dra.
    - Hanna
    - Oktaba
6. Datos del sinodal 4
    - Mat.
    - Salvador
    - López
    - Mendoza
7. Datos del trabajo escrito
    - Administración del Sistema de Correo Electrónico de los Alumnos de la UNAM
    - 127 p.
    - 2007

A mi padre:

Por su apoyo en el desarrollo de mi carrera.

A mi madre y a mi hermana.

A Ingrid.

Agradecimientos

- Al Mat. Facundo Ruíz Doncel por haber posibilitado el presente trabajo, por sus aportes sustantivos tanto en el desarrollo del proyecto como en la elaboración de este documento.

- A Paulita Martínez por sus comentarios a la revisión ortográfica de este documento.

- A la M. en C. María Guadalupe Elena Ibargüengoitia González por sus aportaciones en la elaboración de este documento.

- A la Dra. Amparo López Gaona, a la Dra. Hanna Oktaba y al Mat. Salvador López Mendoza por sus comentarios a la revisión del presente documento.

Índice general

Índice de cuadros

Índice de figuras

Introducción

El Proyecto de Sistema de Correo Electrónico para los Alumnos de la UNAM se estableció en varias etapas. La primera consistió en la propuesta, justificación, análisis de alternativas, estudio costo-beneficio y plan de trabajo para el desarrollo de la aplicación y su implementación, con el fin de lograr la meta inicial, que fue instalar y gestionar los buzones para los 70,000 alumnos de ingreso, generación 2005.

La descripción del proyecto y los resultados de esta primera etapa, se encuentran documentados en la tesis "Correo Electrónico para los Alumnos de la UNAM" de Enrique Reyes Castillo, Facultad de Ciencias 2005.

Del éxito de la implementación y funcionamiento del sistema en esta primera etapa, determinaba el continuar con las siguientes etapas y alcanzar los objetivos del proyecto, consistentes en ofrecer los servicios de correo electrónico a los alumnos activos de la UNAM, con los recursos para dar atención de calidad y de funcionalidad permanente a las necesidades académicas, sociales, culturales y de servicio en la Institución.

El presente trabajo describe las etapas subsecuentes del proyecto, considerando las adecuaciones tanto al equipo de cómputo como al software de aplicación, la problemática que se presentaba en la operación y las opciones de solución, así como la incorporación de las cuentas de correo para los alumnos de las generaciones 2006, 2007, y el complemento, llegando a los cerca de 300,000 buzones de correo, de los alumnos activos por ciclo escolar, además, las tareas de administración del sistema para su funcionalidad continua, con servicios de calidad.

El trabajo se divide en cinco capítulos a saber: Conceptos Generales, Sistema de Correo, Funcionalidad, Líneas de Desarrollo y Administración del Sistema.

En el primer capítulo se introducen los conceptos técnicos base del proyecto. En el segundo capítulo, Sistema de Correo, se enuncian las características del sistema, el equipo de cómputo y las reglas de operación como originalmente fue implementado. El capítulo Funcionalidad describe cómo el sistema atiende los servicios, la problemática en la operación y las necesidades de crecimiento para atender a toda la población escolar. En

el capítulo Líneas de Desarrollo se establecen las estrategias de crecimiento, adaptación a las nuevas necesidades y la solución a los problemas detectados. Finalmente, en el capítulo de Administración del Sistema se describe el trabajo cotidiano para mantener el sistema funcionando y ofrecer los servicios.

Este documento es el resultado del trabajo realizado por más de dos años, en adecuar el sistema, organizar e instalar el equipo de cómputo, administrar y atender los servicios de correo electrónico, así como el cumplir las metas del proyecto.

Capítulo 1

Conceptos Generales

Este capítulo contiene en su primera sección la definición general del sistema de correo electrónico en términos de los lineamientos para su funcionamiento y las decisiones que se tomaron para determinar el *hardware* y *software* para inicializar el sistema.

Además contiene la descripción de los principales paquetes de *software* utilizados para establecer el sistema y las herramientas de desarrollo para ofrecer un servicio continuo y de calidad.

1.1. Definición del sistema

El Sistema de Correo Electrónico de los Alumnos de la UNAM lo constituyen equipo de cómputo, aplicaciones de *software libre* adaptado, comunicación a través de Red UNAM, procesos, servicios y normas operativas, que facilitan el envío y recepción de mensajes.

Para definir el sistema, se presentará el fundamento sobre el cual se constituye el sistema de correo, acompañado de la descripción del *hardware* y *software* que el sistema utiliza.

1.1.1. Hardware

Los recursos de cómputo destinados para el correo electrónico, inicialmente se definieron para atender a unos 70,000 alumnos de ingreso al ciclo escolar 2004-2005 y que inició su operación en julio del 2004.

Para esto, se calculó una infraestructura de cómputo con 9 equipos; 5 para el servicio de correo, uno para la gestión del servicio, uno para la administración y respaldos, además de un equipo Sun Fire V250 para el servicio Web del correo y una PC como sustituto en caso de fallos del servicio Web. Por otra parte se consideraron dos UPS's, un switch con puertos de red 10/100/1000, una unidad de cinta y un rack para poner los equipos (ver figura 1.1).

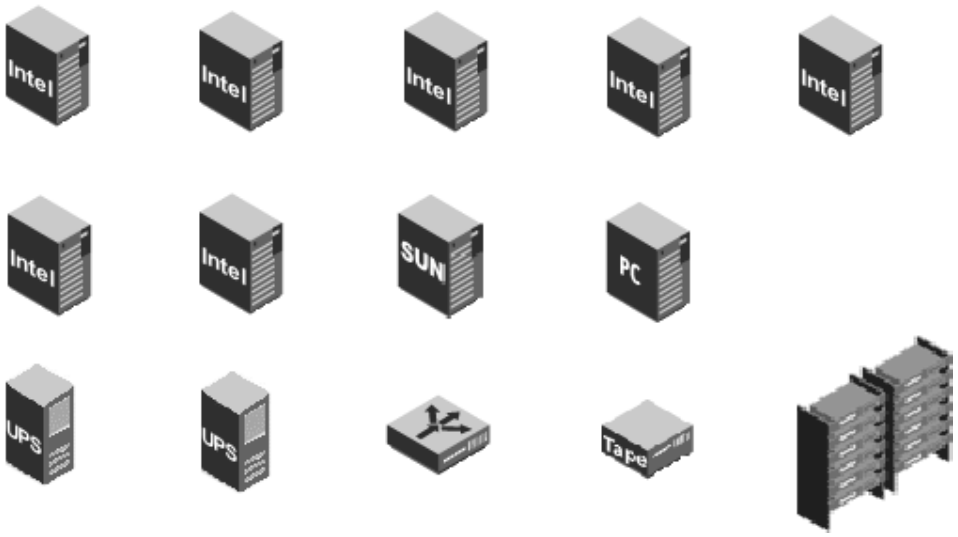


Figura 1.1: Equipos del sistema de correo.

El cuadro 1.3 muestra las características técnicas de los equipos utilizados para montar el sistema de correo.

Icono	Características	Cantidad	Función
	2 Pentium Xeon a 3.06 GHz duales, 2 GB de memoria, 2 discos de 73 GB y fuente redundante.	7	5 servidores de correo 1 gestión del servicio 1 administración y respaldo
	Sun Fire V250, 2 procesadores UltraSPARC[tm] IIIi a 1.28GHz, 2 GB de memoria, 2 discos de 80 GB y fuente redundante	1	Servicio Web
	PC armada, procesador Intel Pentium 4 a 1.28GHz, 2 GB de memoria y 1 disco de 80 GB	1	Respaldo del servicio Web
	UPS de 3KVA's en 2u	2	Fuente y regulación de energía
	Switch 10/100/1000 de 16 puertos	1	Conectividad ethernet
	Unidad de cinta externa DAT (DDS-4) 20GB / 40GB	1	Respaldo en cintas
	Rack 22u	1	Montaje de equipo

Cuadro 1.1: Características del equipo que conforma el sistema de correo.

Posteriormente debería considerarse la incorporación de los servicios de correo a las generaciones de ingreso 2006 y 2007, requiriendo el incrementar la infraestructura de *hardware* que hasta el momento se tenía, para cubrir las nuevas necesidades de almacenamiento y procesamiento de correo, tomando en cuenta el disponer de 20 Megabytes por cada usuario.

Igualmente, en el plan de trabajo futuro, se planteó incorporar al sistema a todos los alumnos de la Universidad que hasta el momento no contaran con el servicio, lo que supone que los cambios que se propongan, permitirán incorporar paulatinamente a nuevos usuarios, sin mayores ajustes.

Además de esto, como se verá en el capítulo 3, se realizó un análisis de las problemáticas técnicas y operativas que el sistema presentaba, llegando a conclusiones que implicaban la ampliación de la infraestructura de *hardware* y la reestructuración del *cluster*.

1.1.2. Software

Dadas las posibilidades que se tenían para implementar el sistema de correo, se optó por la opción que consideraba a *qmail-ldap* [1], como servidor de correo, el cual ha permitido darle continuidad al proyecto hasta la actualidad debido a las ventajas que ofrece en cuanto a seguridad, fiabilidad, flexibilidad, rendimiento, velocidad, bajo consumo de recursos, soporte para *cluster* y administración sencilla. Pero además, por ser *software libre*, que permite instalarlo, modificarlo y adecuarlo a las necesidades propias.

La conclusión a la que se llegó se basó en que *qmail-ldap* ofrece las opciones y posibilidades para el proyecto, cumple con los aspectos técnicos principales, no tiene costo y sólo se requiere hacer adaptaciones a los requerimientos específicos de la aplicación. Pero además permite disponer del código para realizar los cambios que se presenten en cualquier momento.

A diferencia de las otras posibilidades que se consideraron, *qmail-ldap* no proporciona interfaz gráfica, por lo que se tuvo que adaptar Horde-IMP[2] a las exigencias del proyecto. Esta integración aportó una interfaz gráfica Web, que afortunadamente llenó las expectativas que se tenían para brindar el servicio.

Con respecto a las herramientas de desarrollo que se pudieran llegar a necesitar, para administrar o para mantener el servicio, Qmail cuenta con muchas opciones tanto de *software libre*, como de propietario que se pueden adaptar e integrar. Esta es otra de las ventajas que se tienen, y para muestra sólo hay que dar un vistazo a la sección 1.5 (Herramientas de Desarrollo).

[1] En las siguientes dos secciones se describirán los paquetes Qmail y LDAP. En la sección 1.4 se describe a detalle ¿qué es y cómo funciona qmail-ldap?

[2] El Proyecto Horde se compone de unas librerías (el mencionado Horde Framework) que proporcionan funcionalidades básicas (autenticación, gestión de preferencias, interfaz gráfica, etc.) y que funciona como nexo de unión entre distintas aplicaciones de usuario, que son gestionadas como sub-proyectos independientes. IMP (sub-proyecto de Horde) es un sistema *webmail* o cliente de correo electrónico que provee una interfaz web.

1.2. Qmail

1.2.1. ¿Qué es Qmail?

Qmail es un Agente de Transporte de Correo (MTA, Mail Transport Agent en inglés) para sistemas operativos tipo UNIX. Se trata de un sustituto completo para el sistema sendmail que se suministra con los sistemas operativos UNIX. Qmail utiliza el Protocolo Simple de Transferencia de Correo (SMTP, Simple Mail Transfer Protocol en inglés) para intercambiar mensajes con los MTA's de otros sistemas. [1]

1.2.2. Razones para elegir Qmail

Lo que a continuación se presenta (hasta el final de la sección Qmail) está contenido en la página del Grupo de Usuarios de Linux de Toulouse [2].

Sendmail y menos aún su pequeño sustituto smail no se destacan precisamente por su rendimiento, por no hablar de los innumerables agujeros de seguridad[3] (que son menos numerosos en el caso de smail) debido tanto a su arquitectura como a la complejidad de la tarea que han de realizar. Además, la configuración de Sendmail no es intuitiva[4].

Los detractores de Qmail le reprochan que es menos "todo terreno" que su gran rival Sendmail, lo que según algunos autores es cierto, al menos por el momento. Sendmail es además un estándar de hecho, muy extendido y muy documentado, lo que dice algo en su favor. A ciertas personas no les gusta la gran cantidad de ejecutables y de archivos de configuración de Qmail, pero otras piensan que la filosofía de Qmail de dividirse por funcionalidades permite tenerlo todo mucho más claro.

Qmail es un programa con grandes ventajas, y aquellos que lo han utilizado lo encuentran muy seguro, fácilmente configurable, bien documentado, de gran rendimiento y muy orientado a los usuarios, mientras que reprochan a Sendmail sus agujeros de seguridad, su pesadez y lo complicado de su configuración. Pero por supuesto, como todo programa, Qmail tiene sus límites y es obra humana.

La competencia entre agentes de transporte de correo (MTA) bajo Unix no se limita a una confrontación entre Sendmail y Qmail. A menudo, puede resultar suficiente el más modesto smail. También se oye hablar cada vez más de vmailer y de exim.

[3] En las referencias [3] y [4], se encuentra información acerca de vulnerabilidades que se han encontrado en Sendmail.

[4] En la referencia [5], se describe como configurar Sendmail.

1.2.3. Organización de Qmail

A diferencia de Sendmail, Qmail no es monolítico. El sistema Qmail se compone de varios programas que se ejecutan bajo UID/GID diferentes y casi todos non nuls haciendo difícil toda tentativa de invasión. Además, la gestión de cadenas de caracteres en Qmail se ha tratado con mucho cuidado para evitar los problemas de desbordamiento, a menudo origen de agujeros de seguridad. Un grupo de usuarios ofrece incluso un premio de $1000 dólares USA a la persona que encuentre un agujero de seguridad en Qmail !.

Ningún binario lleva por nombre *qmail*. Por el contrario, muchos binarios tienen un nombre que comienza por *qmail-*.

Qmail es de hecho un conjunto de binarios, situados todos por defecto en */var/qmail/bin/*, y que están compuestas por una parte de MTA con su subprograma SMTP que es *qmail-smtpd* y QMTP que es qmail-qmtpd, *qmail-inject* y *mailsubj* para la inyección en la cola de espera, más una parte MDA *qmail-local* así como un pequeño paquete sustituto (wrapper) para sendmail.

Los binarios del sistema Qmail tienen todos su página de manual. La ayuda comprende igualmente las páginas de manual suplementarias que describen ciertos formatos de archivos. He aquí algunos de ellos: *qmail, qmail-limits, qmail-upgrade, qmail-header, addresses, envelopes*.

El protocolo POP3 está disponible gracias a la triada *qmail-popup, checkpassword y qmail-pop3d*. Existen parches para los gestores IMAP y POP3 habituales que permiten utilizarlos con Qmail, (véase la página web de Qmail *http://www.qmail.org/* para más detalles). Los mensajes se colocan en la cola de espera mediante *qmail-queue*. La gestión de la cola de espera la efectúan diferentes programas, *qmail-lspawn, qmail-clean y qmail-rspawn*, todos hijos del demonio *qmail-send*. El registro vía *syslog* puede efectuarse mediante *splogger*. He aquí un esquema que muestra la estructura de Qmail cuando se carga en memoria:

Usuario	Programa	Función
qmails	qmail-send	gestión salida cola de espera
qmaill	__ splogger qmail	registro vía syslog
qmailq	__ qmail-clean	limpieza de la cola de espera
qmailr	__ qmail-rspawn	gestor de mensajes remotos
root	__ qmail-lspawn ./Mailbox	gestor de mensajes locales

La opción *./Mailbox* pasada como parámetro a *qmail-lspawn* es la opción de entrega por defecto. Puede incluso ser una tubería (pipe) hacia un agente

de entrega de correo (MDA) clásico como procmail. Éste es por ejemplo el caso en Debian GNU/Linux.

He aquí un esquema, figura 1.2 que contiene la estructura de la cola de espera de Qmail y su gestión.

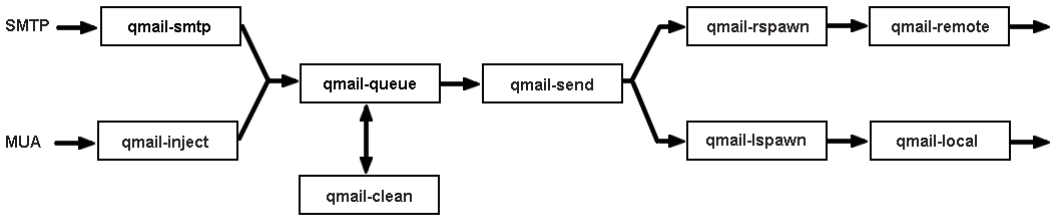


Figura 1.2: Colas de espera de Qmail y su gestión.

qmail-send toma los mensajes que *qmail-queue* ha colocado en la cola de espera. Luego llama a *qmail-lspawn* que a su vez invoca a *qmail-local* y llama a *qmail-rspawn* que a su vez invoca a *qmail-remote*. Véase las respectivas páginas del manual para los detalles. El comando "man qmail" proporciona una presentación global de Qmail.

Claro está, estas páginas del manual no estarán disponibles hasta que Qmail no esté instalado. Hasta entonces, existe una versión HTML de estas páginas [6].

Antes de seguir adelante, he aquí la estructura estándar del árbol de directorios de Qmail */var/qmail/*

Direcorio	Función
/var/qmail/	directorio principal
/var/qmail/alias/	aliases para root, administrador y mailer-daemon
/var/qmail/bin/	binarios de distribución de qmail
/var/qmail/control/	archivos de configuración
/var/qmail/man/	páginas del manual
/var/qmail/queue/	cola de espera

Cuadro 1.2: Estructura del árbol de directorios Qmail.

En la página de internet “the Qmail big picture” [7], la cual proporciona una visión en síntesis de la arquitectura de Qmail.

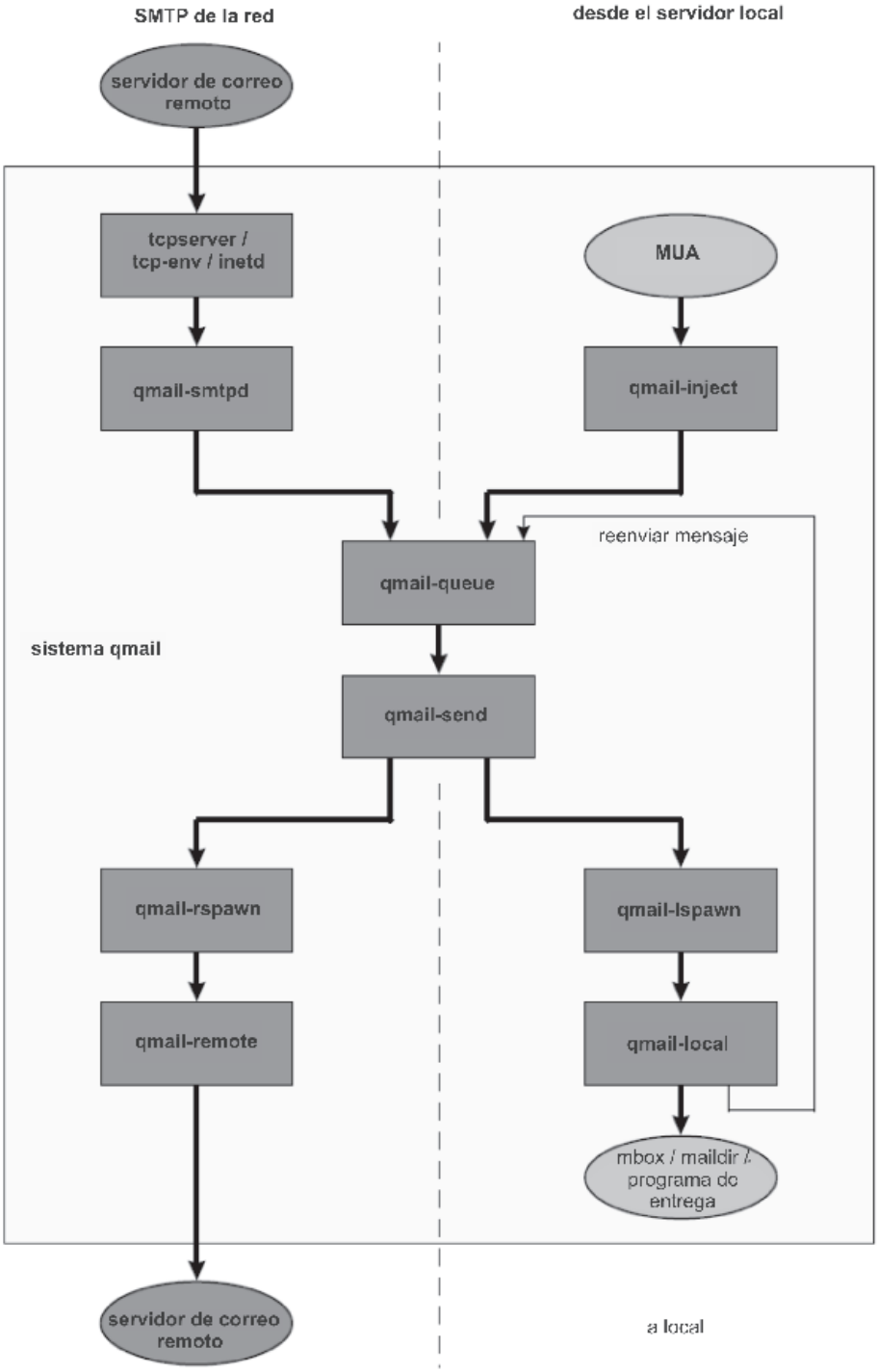


Figura 1.3: Arquitectura de Qmail.

1.3. LDAP

Lo que en esta sección se presenta se encuentra en las referencias [8], [9] para las subsecciones: ¿qué es el Servicio de Directorio?, ¿qué es LDAP?, ¿cómo funciona LDAP? y terminología LDAP, y en la referencia [10] para las subsecciones: archivos de configuración OpenLDAP, y demonios y utilidades OpenLDAP. Además es importante mencionar que en este caso se instaló el servicio LDAP con la distribución OpenLDAP (compatible con la versión 2), por lo que, todo lo que se presenta en esta sección tiene vínculo a esto.

1.3.1. ¿Qué es el Servicio de Directorio?

Para comprender qué es LDAP, es necesario primero saber qué es un directorio en este contexto.

Así como un Sistema de Gestión de Base de Datos (DBMS por sus siglas en inglés) como Sybase, Oracle, Informix, MySQL o Postgres se utilizan para procesar consultas y actualizaciones a una base de datos relacional, un servidor LDAP es utilizado para procesar consultas y actualizaciones a un directorio de información. En otras palabras, un directorio de información es un tipo de base de datos, pero no es una base de datos relacional. Y a diferencia de una base de datos relacional que está diseñada para procesar cientos o miles de cambios por minuto, los directorios están fuertemente optimizados para el rendimiento en lectura.

Como consecuencia de estar optimizado para lecturas, un directorio no implementa sistemas de transacciones[5], para "aplicar cambios" (*commit*) o "cancelar cambios" (*rollback*) sobre un conjunto de operaciones sobre el directorio.

Un directorio LDAP almacena sus datos en un árbol jerárquico, de manera muy parecida a como se estructura un directorio de archivos en UNIX. Un registro, o *nombre distinguido* en un directorio LDAP, esta formado por una serie de *entradas individuales* (hojas o nodos del árbol) que se extienden hasta la *base* del directorio (raíz del árbol). El *nombre distinguido* se utiliza para referirse a una registro sin ambigüedades. Más información sobre esto en la sección 1.3.4, Terminología LDAP.

[5]Una transacción es una unidad lógica de trabajo que permite a un usuario agrupar una o una secuencia de operaciones (consultas/actualizaciones) y mediante la cual un estado consistente de la base de datos se transforma en otro estado consistente. Esto es, si por algún motivo la secuencia de operaciones de una unidad de trabajo no se puede completar en su totalidad, entonces, ninguna de las operaciones de la secuencia tiene efecto. También asegura que distintas secuencias de operaciones (unidades de trabajo) se ejecuten sin interferir entre ellas. Definición basada en [11]

Por otra parte, la información puede estar distribuida y replicada aumentando su disponibilidad y fiabilidad. Puede existir un mismo árbol del directorio LDAP, con algunas de sus ramas repartidas en distintos servidores LDAP. También se pueden tener réplicas de un árbol o ramas del directorio LDAP en uno varios servidores LDAP esclavos. Cuando se replica la información del directorio, pueden aceptarse inconsistencias temporales entre la información que hay en las réplicas, siempre que finalmente exista una sincronización.

Una de las mejores características de LDAP es la capacidad de almacenar cuentas de usuario con sus contraseñas. Esto proporciona la base para centralizar la autenticación de usuarios para una o varias aplicaciones. Para esto se tienen que configurar las aplicaciones para que se enlacen con el servidor LDAP y luego realicen la autenticación con el directorio. De esta manera un solo acceso de usuario puede funcionar como entrada a diferentes aplicaciones.

Una vez que sabemos qué es un directorio, podemos explicar en qué consiste LDAP.

1.3.2. ¿Qué es LDAP?

LDAP es el Protocolo de Acceso Ligero a Directorio (Lightweight Directory Access Protocol en inglés), surge en 1993 en la Universidad de Michigan y define una serie de operaciones para la consulta y actualización de información almacenada en un directorio por red.

LDAP no es una base de datos en absoluto, sino un protocolo utilizado para acceder a la información almacenada en un directorio (también conocido como un directorio LDAP).

LDAP se deriva de DAP o Protocolo de Acceso a Directorio (definido en los estándares X.500 [6]) el cual accede a un directorio pero sobre el modelo OSI, de modo que resulta lento y poco eficiente. A diferencia de X.500 LDAP soporta el protocolo TCP/IP, el cual es significativamente más simple, pequeño y rápido.

[6] X.500 es un conjunto de estándares de redes de computadoras sobre servicios de directorio. Los protocolos definidos por X.500 incluyen, protocolo de acceso al directorio (DAP), el protocolo de sistema de directorio, el protocolo de ocultación de información de directorio, y el protocolo de gestión de enlaces operativos de directorio.

1.3.3. ¿Cómo funciona LDAP?

El servicio de directorio LDAP se basa en un modelo cliente-servidor. Uno o más servidores LDAP contienen los datos que conforman el árbol del directorio LDAP; el cliente LDAP se conecta con el servidor LDAP y le hace una consulta. El servidor contesta con la respuesta correspondiente, o bien con una indicación de dónde puede el cliente encontrar más información (normalmente otro servidor LDAP).

1.3.4. Terminología LDAP

El nivel superior de un directorio LDAP es la base, conocido como el "*dn base*". Un "*dn base*", generalmente toma la forma "*dc=escolar, dc=unam, dc=mx*", este formato está separado en "componentes de dominio" (*dc* - domain component, en inglés). Por ejemplo el dominio *escolar.unam.mx* puede deducirse de "*dc=escolar, dc=unam, dc=mx*". Es conveniente usar el nombre de dominio de la empresa u organización como base del directorio, por comodidad y para evitar duplicidades.

Debajo de la base del directorio, se pueden crear contenedores que separen lógicamente los datos. Por razones históricas, la mayoría de los directorios configuran estas separaciones lógicas por "unidades organizacionales" (*ou* - organizational units, en inglés), que en X.500 eran utilizadas para indicar la organización funcional dentro de la empresa: ventas, finanzas, etc. Actualmente las implementaciones de LDAP han mantenido la convención del nombre *ou=*, pero separa las cosas por categorías amplias como *ou=gente (ou=people), ou=grupos (ou=groups), ou=dispositivos (ou=devices)*, y demás. Se pueden usar niveles directamente inferiores a *ou*, para separar por subcategorías. Por ejemplo, un árbol de directorio LDAP (sin incluir entradas individuales) podría parecerse a esto:

```
dc=escolar, dc=unam, dc=mx
      ou=personas
            ou=academicos
            ou=alumnos
            ou=empleados
      ou=dispositivos
            ou=computadoras
            ou=impresoras
      ou=planteles
```

Todas las entradas almacenadas en un directorio LDAP tienen un único "nombre distinguido" (*dn* - distinguished name, en inglés). El *dn* para cada entrada está compuesto de dos partes: el Nombre Relativo Distinguido (RDN - Relative Distinguished Name, por sus siglas en inglés) y la localización dentro del directorio LDAP donde está el registro.

El RDN es la porción del *dn* que no está relacionada con la estructura del árbol de directorio. Todos los elementos que se almacenan en un directorio LDAP tienen un nombre o un identificador como RDN. El nombre es almacenado en el atributo "nombre común" (*cn* - common name, en inglés) y el identificador es almacenado en el atributo "id. de usuario" (*uid* - user id., en inglés).

De esta manera y siguiendo el ejemplo, el *dn* de un usuario se puede ver como sigue:

```
dn: uid=alumno1,ou=personas,ou=alumnos,dc=escolar,dc=unam,dc=mx
```

Cada entrada tiene atributos, los atributos son fragmentos de información directamente asociados con la entrada. Por ejemplo, un alumno podría ser una entrada LDAP y los atributos asociados al alumno podrían ser el nombre, el plantel, la carrera, etc. Los dispositivos podrían ser otra entrada del directorio LDAP y los atributos comunes a los dispositivos pueden ser la marca, el modelo, su dirección ip y su número de inventario.

Un *objectclass* define los atributos que cada entrada contendrá y además discrimina los atributos necesarios de los que son opcionales. La definición de los *objectclass* de LDAP se encuentran en el directorio */usr/local/etc/openldap/schema*.

El Formato de Intercambio de Datos (LDIF, LDAP Data Interchange Format en inglés) es un formato que se utiliza para la importación y exportación de datos independientemente del servidor LDAP que se esté utilizando.

El formato LDIF es simplemente un formato de texto ASCII para entradas LDAP, que tiene la siguiente forma:

```
dn: <nombre distinguido>
<nombre_atributo>: <valor>
<nombre_atributo>: <valor>
```

En un archivo LDIF puede haber más de una entrada definida, cada entrada se separa de las demás por una línea en blanco. A su vez, cada entrada puede tener una cantidad arbitraria de pares <nombre_atributo>: <valor>.

Este formato es útil tanto para realizar copias de seguridad de los datos de un servidor LDAP, como para importar pequeños cambios que se necesiten realizar manualmente en los datos, siempre manteniendo la independencia de la implementación LDAP y de la plataforma donde esté instalada.

A continuación podemos observar un ejemplo de una entrada en formato LDIF para representar una cuenta de usuario:

```
dn: uid=test,dc=escolar,dc=unam,dc=mx
cn: test
sn: test
givenName: test
registeredAddress: 0
telephoneNumber: 00000000
description: 00
postalCode: 00000
title: 0
objectClass: top
objectClass: person
objectClass: organizationalPerson
objectClass: inetOrgPerson
objectClass: qmailuser
mail: test@escolar.unam.mx
mailMessageStore: /var/qmail/maildirs/test
mailQuotaSize: 20000000
mailHost: bernoulli.dgae.unam.mx
uid: test
userPassword:: cG93ZXI=
```

Por otra parte, existe el Lenguaje de Marcas de Servicios de Directorio (DSML, Directory Services Markup Language) la cual es una implementación XML que proporciona un formato común para describir y compartir información de servicios de directorio entre distintos sistemas de directorios.

La diferencia entre LDIF y DSML es que esta última cuenta con herramientas para validar la integridad de un archivo DSML contra el esquema del directorio. Como ejemplo de estas herramientas se tiene *DSML Tools* bajo licencia GPL [7], ver referencia [12].

1.3.5. Archivos de configuración OpenLDAP

Los archivos de configuración OpenLDAP están instalados en el directorio */usr/local/etc/openldap*. Si se da un ls en este directorio se verá algo parecido a:

```
ldap.conf        ldapsearchprefs.conf  schema
ldapfilter.conf  ldaptemplates.conf    slapd.conf
```

[7] La GNU GPL (General Public License o licencia pública general) es una licencia creada por la Free Software Foundation a mediados de los 80, y está orientada principalmente a los términos de distribución, modificación y uso de *software*. Su propósito es declarar que el *software* cubierto por esta licencia es *software libre*.

Modificar el archivo slapd.conf

El archivo *slapd.conf*, localizado en */usr/local/etc/openldap*, contiene la información de la configuración necesaria para el servidor *slapd* LDAP. Se necesita modificar este archivo para hacerlo específico a cada dominio y servidor.

La línea suffix asigna el dominio para que el servidor LDAP proporcione información. La línea suffix debe ser configurada para que refleje el nombre de dominio. Por ejemplo:

```
suffix          "dc=escolar, dc=unam, dc=mx"
```

La entrada *rootdn* es el *dn* para un usuario que no está limitado por el control de acceso o por los parámetros administrativos limitados para las operaciones en el directorio LDAP. El usuario *rootdn* puede ser visto como el usuario root para el directorio LDAP. La línea *rootdn* debe parecerse a algo como sigue:

```
rootdn          "cn=adminitrador, dc=escolar, dc=unam, dc=mx"
```

La línea *rootpw* define un password para el usuario especificado por el *rootdn*:

```
rootpw          {crypt}algocifrado
o
rootpw          {MD5}algocifradoMD5
```

En este ejemplo, se usa una contraseña de root cifrada, mejor solución que usar contraseñas de root en texto plano en el archivo *slapd.conf*. Para hacer esta cadena encriptada, se puede usar Perl:

```
perl -e "print crypt('password','a_salt_string');"
o
perl -e "print md5('password');"
```

El directorio schema

A partir de la versión 2 de OpenLDAP, el directorio */usr/local/etc/openldap/schema* contiene las especificaciones de los esquemas que se utilizarán en el directorio LDAP. En versiones anteriores de OpenLDAP estas definiciones se encontraban en los archivos *slapd.at.conf* y *slapd.oc.conf*.

Cada archivo en el directorio *schema* contiene las definiciones de sus *objectclass* y de sus aributos correspondientes. Los archivos están referenciados en */usr/local/etc/openldap/slapd.conf* y usando la directiva *include*, como se muestran a continuación:

```
include         /usr/local/etc/openldap/schema/core.schema
include         /usr/local/etc/openldap/schema/cosine.schema
include         /usr/local/etc/openldap/schema/inetorgperson.schema
include         /usr/local/etc/openldap/schema/nis.schema
include         /usr/local/etc/openldap/schema/qmail.schema
```

1.3.6. Demonios y utilidades OpenLDAP

El paquete OpenLDAP incluye dos demonios: *slapd* y *slurpd*.

El demonio *slapd* es el demonio estándar de LDAP, que se necesita para ejecutar LDAP.

El demonio *slurpd* controla la réplica de los directorios LDAP en una red. *slurpd* envía los cambios del directorio maestro LDAP al directorio esclavo LDAP. No se necesita usar *slurpd* a no ser que se tenga más de un servidor LDAP en la red. Si se tiene más de un servidor LDAP, se tiene que usar *slurpd* para tener el directorio LDAP sincronizado.

OpenLDAP también incluye algunas utilidades para añadir, modificar y borrar las entradas en un directorio LDAP:

La utilidad *ldapmodify* se usa para modificar las entradas en una base de datos LDAP, aceptando la entrada mediante un archivo o entrada estándar.

La utilidad *ldapadd* se usa para añadir entradas al directorio (*ldapadd* es un enlace fijo para *ldapmodify -a*).

ldapsearch se usa para buscar entradas en el directorio LDAP usando el intérprete de comandos del *shell*.

ldapdelete borra las entradas del directorio LDAP, aceptando entradas a través de un archivo o del intérprete de comandos del *shell*.

A excepción de *ldapsearch*, cada una de estas utilidades es mucho más fácil de usar refiriéndose a un archivo con los cambios por realizar que a escribir los comandos uno tras otro. Cada una de las páginas de manual respectivas cubre la sintaxis de estos archivos.

Para importar o exportar bloques de información con un directorio *slapd* o para ejecutar tareas administrativas similares se requieren diferentes utilidades, localizadas en */usr/sbin*.

slapadd añade entradas desde un archivo LDIF a un directorio LDAP. Por ejemplo, el comando */usr/sbin/slapadd -l ldif* donde ldif es el nombre del archivo LDIF que contiene nuevas entradas.

slapcat extrae del directorio LDAP las entradas y las guarda en el archivo LDIF. Por ejemplo, el comando */usr/sbin/slapcat -l ldif* donde ldif es el nombre de un archivo LDIF que contiene la entrada de un directorio LDAP.

slapindex construye un índice de la base de datos *slapd* basada en el contenido de la base de datos actual. Ejecutar */usr/sbin/slapindex* para construir el índice.

slappasswd genera un valor de contraseña de usuario para usar con el valor *ldapmodify* o *rootpw* en */usr/local/etc/openldap/slapd.conf*. Ejecutar */usr/sbin/slappasswd* para crear una contraseña.

1.4. qmail-ldap

La gran parte de lo que se presentará en esta sección está traducido del contenido de la referencia [13].

El paquete *qmail-ldap* es una extensión a Qmail 1.03 la cual da el soporte para recuperar todos los datos del usuario de un directorio LDAP que están guardados en archivos en el disco duro. Esto permite una administración más sencilla, especialmente en ambientes distribuidos. También hay soporte para desarrollo en *cluster* mediante *qmail-ldap*, el cual ha tenido un desempeño óptimo para las instalaciones grandes de correo en ISPs (Proveedores de Servicios de Internet).

De hecho, hay una buena cantidad de sitios que usan Qmail, ver referencia [14], que contiene una lista de algunos sitios que lo usan dentro de los cuales destacan: Yahoo! Mail, Yahoo! Groups, Ohio State (la Universidad más grande de E.U., según la referencia citada), etc.

El desempeño de un *cluster* de correo con *qmail-ldap*, depende de un enorme número de factores, que van desde los más obvios como el tipo de infraestructura de cómputo utilizado (*hardware*), velocidad de transmisión de las interfaces y dispositivos de red, hasta factores sofisticados como lo pueden ser la configuración de los índices del directorio LDAP, el tipo de sistema de archivos donde se almacenarán los mensajes de correo[8], etc.

Para poder cargar las cuentas de usuario Qmail en un servidor de LDAP es necesario agregar el *qmailuser schema* al directorio LDAP. Así los usuarios pueden tener el *objectclass qmailuser* el cual define atributos como *mail, mailHost, mailQuotaSize*, entre otros.

qmail-ldap puede ser configurado de modo que todos los servidores de correo en una organización puedan compartir los mismos datos de la cuenta.

[8]Hay reportes en foros de Internet[15] donde se discute el desempeño de diferentes sistemas de archivos que contienen grandes volúmenes de archivos o directotios (más de 10,000). Personalmete he observado diferencia en el desempeño de manejo de datos en diferentes sistemas de archivos con arriba de 30,000 directorios (con los mensajes de los usuarios).

qmail-ldap soporta el reenvío de los mensajes al equipo al que lo debe de recibir, especificado en cada entrada de la cuenta de los usuarios, incluso cuando todas las direcciones internas de la compañía sean diferentes, es decir, *qmail-ldap* soporta dominios y no es necesario poner en la dirección electrónica del usuario en qué equipo del *cluster* está almacenado su correo.

1.4.1. Componentes de qmail-ldap

Lo que aquí se presenta está traducido de la parte "Components of qmail-ldap and how they fit together" (Componentes qmail-ldap y cómo se acoplan) de la referencia que se mencionó al inicio de esta sección.

qmail-queue

El componente *qmail-queue* toma los mensajes y los pone en la cola de salida de Qmail.

qmail-send

El componente *qmail-send* toma los mensajes de la cola de correos de salida y usa a *qmail-lspawn* para entregas locales y *qmail-rspawn* para entregas remotas. *qmail-send* planifica las entregas de todos los mensajes.

qmail-todo

El componente *qmail-todo* preprocesa correo para reducir la carga de trabajo del componente *qmail-send*. Con *qmail-todo* el desempeño total del servidor de correo es mucho mejor debido a que le quita carga al componente *qmail-send*.

qmail-lspawn

El componente *qmail-lspawn* recibe de *qmail-send* los mensajes identificados como locales, del *cluster*, para luego buscar en el directorio LDAP el usuario al que va dirigido el mensaje y realizar con *qmail-qmqpc* la entrega del correo a otro equipo del *cluster* o almacenarlo localmente con *qmail-local*.

qmail-local

El componente *qmail-local* realiza la entrega de mensajes localmente, en donde, ya no es necesario retransmitir el mensaje a otro equipo del *cluster*.

qmail-rspawn

El componente *qmail-rspawn* invoca a *qmail-remote* para realizar entregas remotas.

qmail-remote

El componente *qmail-remote* envía un correo a un equipo remoto vía SMTP.

qmail-inject

El componente *qmail-inject* lee un mensaje de la entrada estándar, agrega encabezados e invoca *qmail-queue* para que lo agregue a la cola.

qmail-smtpd

El componente *qmail-smtpd* recibe un mensaje de un equipo remoto vía SMTP (Simple Mail Transfer Protocol, Protocolo Simple de Transferencia de Mensajes). Una vez que recibe el mensaje lo pasa a *qmail-queue*.

qmail-qmqpd

El componente *qmail-qmqpd* recibe los mensajes de equipos remotos vía QMQP (Quick Message Queuing Protocol, Protocolo Rápido de Encolado de Mensajes). Este siempre reenvía todos los mensajes, así que se tiene que tener cuidado para que sólo los equipos autorizados puedan conectarse. QMQP es utilizado para las entregas dentro del *cluster*, por lo que si se quiere usar *qmail-ldap* con soporte para *cluster* se debe de instalar *qmail-qmqpd*.

qmail-imapd

El componente *qmail-imapd* es invocado desde *imaplogin* y maneja las sesiones IMAP.

auth_imap

Es normalmente invocado desde *imaplogin* para autenticar usuarios. También es responsable de las sesiones IMAP, direccionándolas dentro del *cluster qmail-ldap*.

qmail-ldaplookup

Es una herramienta para verificar que la instalación de LDAP está correcta.

1.4.2. Archivos de control de qmail-ldap

Los archivos de control de *qmail-ldap* los cuales se encuentran en el directorio */var/qmail/control*, son necesarios para el correcto funcionamiento y configuración de Qmail. A continuación se dará una breve descripción de los archivos de control de Qmail que considero más importantes basándome en la referencia [16].

Archivo de control	Descripción
control/me	nombre del servidor
control/locals	lista de los dominios locales de los cuales se aceptará correo
control/rcpthosts	lista de los dominios remotos a los cuales el *relay* (relevo) es permitido
control/defaultdomain	dominio
control/plusdomain	dominio
control/defaultdelivery	tipo de entrega "./Maildir/"
control/ldapmessagestore	directorio donde se almacenan los buzones
control/ldauid	id del usuario al cual los usuarios serán mapeados
control/ldapgid	id del grupo al cual los usuarios serán mapeados
control/defaultquotasize	cantidad máxima de espacio en disco que tendrán los usuarios por default (en bytes)
control/quotawarning	texto de alerta de acercamiento al límite de la cuota
control/ldapcluster	"1" habilitar soporte para cluster, "0" para deshabilitarlo
control/ldapserver	servidor o servidores (uno por línea), donde se encuentra el servicio LDAP o alguna de sus réplicas
control/ldapbasedn	base donde se realizarán las búsquedas LDAP
control/ldaplogin	usuario autorizado para realizar operaciones LDAP
control/ldappassword	la contraseña del usuario que realizará operaciones LDAP
control/ldaptimeout	tiempo de espera para una operación LDAP

Cuadro 1.3: Archivos de control de *qmail-ldap*.

1.4.3. Arquitectura de Qmail con LDAP

En la figura 1.4 se puede observar la arquitectura del funcionamiento de *qmail-ldap* (fue obtenida y traducida de [17]). Observar la diferencia con la figura 1.3, que muestra la arquitectura de Qmail sin soporte LDAP.

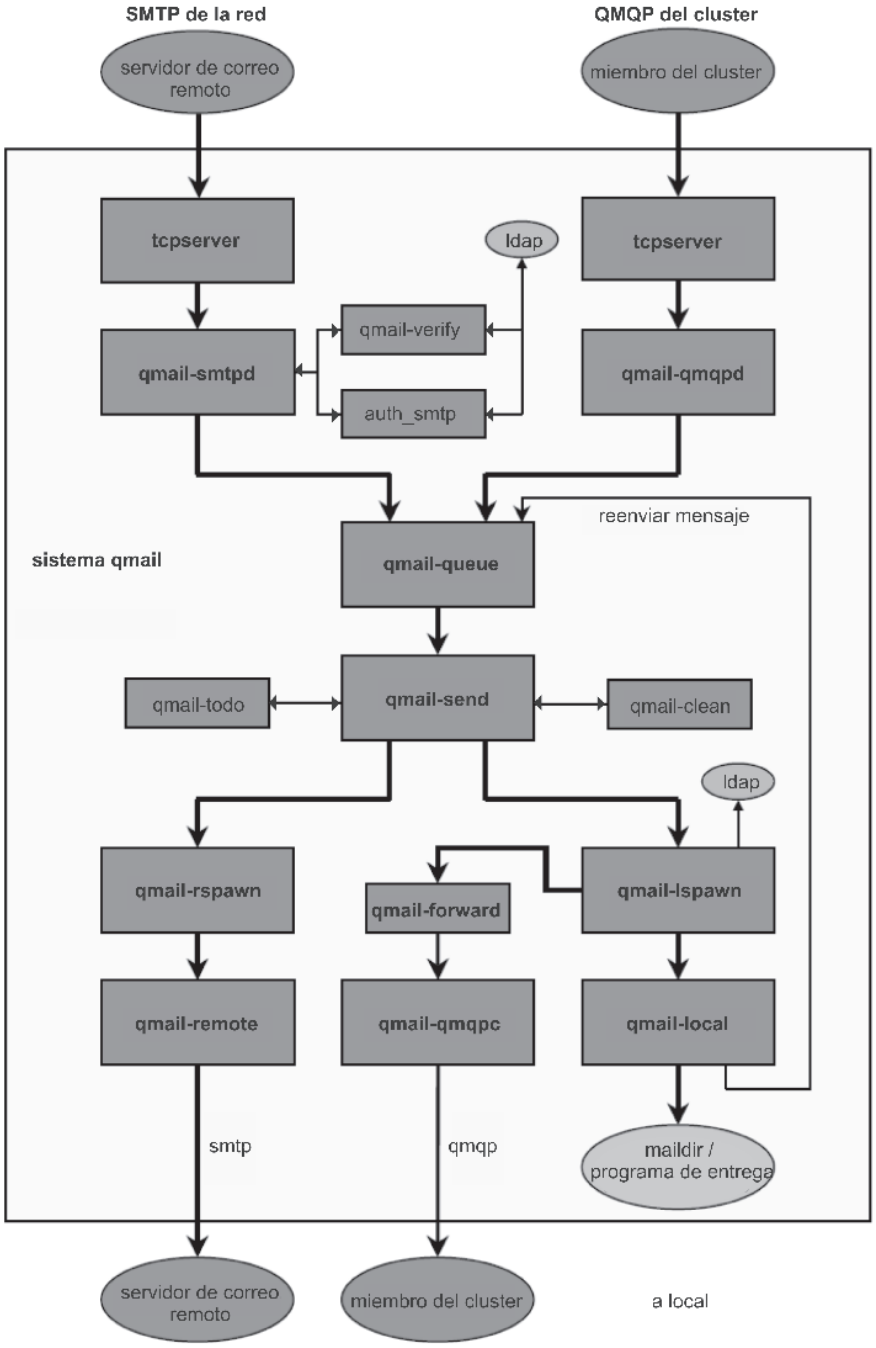


Figura 1.4: Arquitectura de Qmail con LDAP.

1.5. Herramientas de Desarrollo

Desde el punto de vista de la administración del sistema, se debe procurar contar con las herramientas que ayuden a monitorear, mantener y fortalecer el sistema de correo. Los sistemas de correo se han tenido que adaptar a un ambiente de alta desconfianza, por lo que es prioritario para el sistema de correo de los alumnos de la UNAM contar con *software* confiable que ayude a mejorar el servicio. A continuación se presentarán las herramientas: SpamAssassin, ClamAV, Qmail-Scanner, Qmqtool, SNMP y RRDtool.

1.5.1. SpamAssassin

SpamAssassin es una herramienta que surge como una opción para el análisis de correo electrónico, y su principal función es determinar si un correo se trata o no de correo *spam*. Consiste en un mecanismo heurístico basado en reglas en donde se comparan distintas partes del mensaje con cada una de las reglas definidas, las cuales son el corazón del sistema de detección, en donde cada regla suma y resta puntos de un marcador. Un mensaje con un marcador suficientemente grande es reportado como *spam*.

Esta herramienta fue originalmente registrada por DeerSoft, posteriormente adquirido por Network Associates, actualmente opera bajo la licencia Apache Software Fundation [18] [19].

Ventajas

SpamAssassin es un sistema ampliamente usado para la administración de Sistemas de Correo debido a que ofrece las siguientes ventajas:

- Modularidad
- Escalabilidad
- Puede ser usado como servidor o como cliente.
- Se puede instalar en diferentes Sistemas Operativos.
- Filtra correo de entrada, así como de salida.
- Se pueden implantar un rango amplio de políticas antispam.
- Se puede integrar a distintos Agentes de Transmisión de Correo MTAs

Además cabe mencionar que SpamAssassin forma la base de numerosos productos antispam comerciales disponibles en el mercado hoy en día.

¿Cómo trabaja SpamAssassin?

Hay varias formas en las que SpamAssassin procesa un mensaje:

- Comprobando que los encabezados del mensaje sean consistentes y se apeguen a los estándares (ejemplo, verificar que la fecha contenga el formato correcto).
- Comprobando que los encabezados y el cuerpo contengan frases o mensajes comúnmente encontrados en correo *spam* (ejemplo, "haz dinero rápido!!!" en varios idiomas)
- Buscando la suma de comprobación en las distintas bases de datos en línea con sumas de comprobación de correo *spam* conocido y que coincida con la del mensaje analizado.
- La dirección ip del sistema que envía se puede buscar en varias listas en línea de los sitios que han sido usados por *spammers* o que de alguna otra manera son sospechosos.
- Direcciones, hosts, o dominios pueden ser agregados en listas blancas o negras. Una lista negra puede ser construida automáticamente basada en el historial de reportes de los mensajes pasados.
- SpamAssassin puede ser entrenado para reconocer los tipos de *spam* que se reciben aprendiéndolos de un conjunto de mensajes que se consideran *spam* y un conjunto de mensajes que no se consideran *spam*.
- La dirección ip del sistema que envía se puede comparar con el nombre de dominio del remitente usando el protocolo Sender Policy Framework (SPF) para determinar si ese sistema tiene permisos para enviar mensajes de usuarios en ese dominio. Esta característica requiere SpamAssassin 3.0

Usando spamc/spamd

Si se está filtrando mucho correo de entrada, el tiempo de procesamiento requerido para invocar el *script* SpamAssassin (e iniciar el interprete de Perl) para cada mensaje puede llegar a ser insostenible. Una de las alternativas que se propone es correr SpamAssassin como demonio. El demonio *spamd* es iniciado cuando arranca el sistema y en este proceso carga los módulos de Perl necesarios para realizar la búsqueda de *spam*. En lugar de correr el *script* SpamAssassin en cada mensaje, los mensajes son canalizados al programa *spamc*. *spamc* es un cliente ligero, escrito en C compilado a un ejecutable que simplemente toma el mensaje, lo retransmite a *spamd*, y regresa los resultados.

Verdaderos Negativos (HAM)

Son los mensajes que el usuario y SpamAssassin no consideran *spam*. SpamAssassin adiciona al encabezado del mensaje las etiquetas "X-Spam-Status" con la leyenda "NO" y "X-Spam-Checker-Versión" con la versión utilizada por SpamAssassin.

Verdaderos Positivos (SPAM)

Son aquellos mensajes en que el usuario y SpamAssassin están de acuerdo en que son *spam*. SpamAssassin adiciona al encabezado del mensaje las etiquetas "X-Spam-Level, X-Spam-Status, y X-Spam-Flag". Si se habilita la opción *rewrite_subject* se agrega al asunto del mensaje la etiqueta "***SPAM*****".

Otras características

SpamAssassin también adiciona la posibilidad de aprender a clasificar mensajes en base a un grupo de carpetas en donde el usuario previamente han incluido sus mensajes basura (*spam*) y mensajes validos (*ham*). Esta operación le permite a SpamAssassin "aprender" a identificar cada correo.

1.5.2. ClamAV

ClamAV es uno de los antivirus más completos con licencia GPL [9] para UNIX. El propósito principal de este *software* es lograr una fácil integración con los servidores de correo, para brindar la funcionalidad de exploración de virus en los mensajes [20].

El paquete proporciona un demonio multi-hilo flexible y escalable, un explorador por línea de comandos, y una herramienta para la puesta al día automática vía Internet.

Uno de los puntos importantes de este *software*, es la base de datos de virus conseguida hasta la fecha.

[9] Esta licencia ya fue descrita en la sección 1.3, en la parte de "Terminología LDAP", en un pie de página.

Características de ClamAV

- Distribuido bajo los términos de la Licencia Pública General GNU.
- Explorador por línea de comandos.
- Demonio multi-hilo, de alta velocidad.
- Cumple con las especificaciones de familia de estándares POSIX (Portable Operating System Interface for UNIX o interfaz portable de sistema operativo para Unix).
- Capacidad para examinar contenido de archivos comprimidos en los formatos [10] : ZIP, RAR, Tar, Gzip, Bzip2, MS OLE2, MS Cabinet, MS CHM y MS. SZDD.
- Soporte para explorar archivos comprimidos con UPX[11], FSG[12] y Petite[13].
- Avanzada herramienta de actualización con soporte para firmas digitales y consultas basadas sobre DNS.

1.5.3. Qmail-Scanner

Teniendo instalados y configurados SpamAssassin y ClamAV, es primordial cubrir la forma en que se integrarán al MTA (Mail Transfer Agent). Cada MTA en sí mismo tiene varios métodos para realizar dicha integración.

Con respecto a Qmail, las opciones son pocas pero la forma más sencilla de hacerlo es usando Qmail-Scanner como integrador. Ver ver referencia [21].

Como ya se ha expuesto, Qmail incluye una cantidad compleja de componentes por lo que se tienen que realizar ajustes para indicar a Qmail que en la entrega y recepción de mensajes, se tiene que realizar una exploración intermedia de los mensajes para asegurarse que el mensaje no representa ningún riesgo para los usuarios ni para el sistema. A continuación se presenta cuál es este ajuste.

[10] Los formatos que tienen el sufijo "MS" son usados pos sistemas operativos tipo Windows, mientras que los restantes son formatos de compresión ampliemente utilizados en sistemas tipo Windows, Linux, UNIX, etc.

[11] UPX es un empaquetador de ejecutables gratuito, portable y de alto rendimiento. UPX soporta diferentes formatos de ejecutables, incluyendo programas de Windows 95/98/ME/NT/2000/XP y DLLs, programas de DOS, Linux, etc.

[12] FSG son las siglas de Fast, Small y Good. Es un compresor de ejecutables pensando e ideado originalmente para ejecutables Windows.

[13] Compresor de archivos ejecutables Windows. Los archivos comprimidos pueden ser ejecutados como si no lo estuviesen.

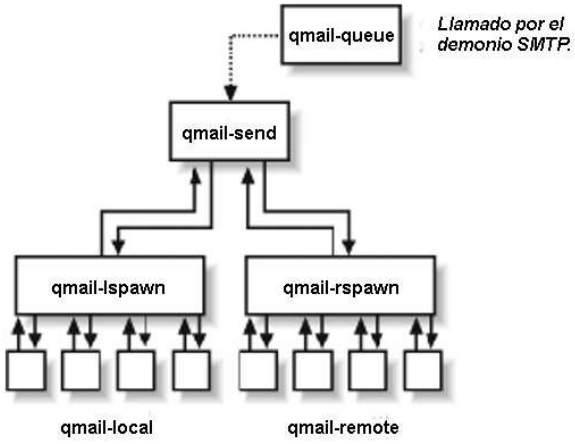


Figura 1.5: Arquitectura de Qmail cuando recibe un mensaje.

A cada componente de Qmail le corresponden un rol distinto en la recepción de mensajes de Internet. Los mensajes de Internet son típicamente incorporados vía el demonio *qmail-smtpd*, el cual escucha el puerto 25 y conduce las transacciones del SMTP con el remitente remoto. *qmail-smtpd* pasa el mensaje al programa *qmail-queue*, almacenándolo en una cola de salida para procesarlo posteriormente.

El demonio *qmail-send* lee los mensajes en la cola de salida e intenta entregarlos utilizando el demonio *qmail-lspawn* (el cual los pasa a *qmail-local* para la entrega local) o el demonio *qmail-rspawn* (el cual los pasa al programa *qmail-remote* para retransmitirlos a los servidores remotos).

Para realizar la integración propuesta es necesario contar con la versión de Qmail con el parche QMAILQUEUE (de Bruce Guenter's), la cual permite por medio de la variable de ambiente QMAILQUEUE invocar un programa *qmail-queue* distinto que el que viene con Qmail (ver figura 1.6).

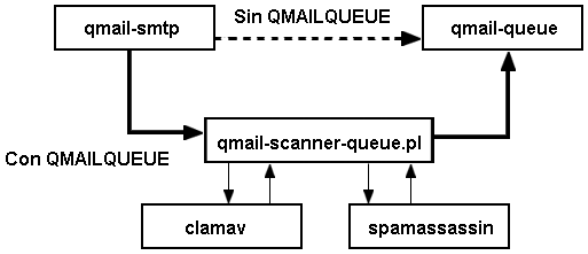


Figura 1.6: Qmailqueue redirigiendo el correo hacia el antispam y el antivirus.

De esta manera el script de perl qmail-scanner-queue.pl es empleado en lugar del binario de Qmail *qmail-queue*. Después de que qmail-scanner-queue.pl procesa el mensaje pasándolo por el antivirus y el antispam, este llama al binario original *qmail-queue* para que lo reintegre al sistema de correo.

Características de Qmail-Scanner

- Usa casi cualquier explorador de virus externo vía línea de comandos
- Puede llamar más de un explorador de virus para cada mensaje
- El explorador interno puede también ser usado para bloquear correo basado en el tipo de archivo adjunto, o en los encabezados de mensajes
- El motor interno busca mensajes mal formados que es sabido que son usados por troyanos para infectar clientes
- Puede integrarse con SpamAssassin para proporcionar etiquetado de correo *spam*

Requerimientos

- Qmail 1.03
- Una cuenta bajo la cual correrá Qmail-Scanner (default "qscand"). Para seguridad extra crearla con un directorio normal (ejemplo "/home/qscand"), pero con un shell falso (ejemplo "/bin/false")
- Reformime de Maildrop[14] (versión 1.3.8 o superior).
- Perl 5.005_03+
- Modulo Time::HiRes de perl
- Modulo DB_File de perl
- Modulo Sys::Syslog de perl

La distribución de este paquete trae un *script* que busca las dependencias necesarias que no están instaladas en el sistema. Las dependencias que falten se puede instalar con el comando:

```
perl -MCPAN -e 'install Sys::Syslog'
```

[14]Maildrop es un filtro de correo electrónico y un programa de entrega de correos locales. Fue escrito por Sam Varshavchik autor de courier-imap. El paquete Maildrop incluye el programa reformime, el cual es usado por qmail-scanner para trabajar con mensajes encapsulados MIME (Extensiones de Correo de Internet Multipropósito). La información fue obtenida de la página http://www.qmailinfo.org/index.php/Maildrop

1.5.4. Qmqtool

Qmqtool es un programa para manipular la cola de correos de Qmail, con funcionalidades que permiten ver y modificar de forma segura, el contenido de una cola de correos Qmail.

Qmqtool puede ser copiado y distribuido bajo los términos encontrados en el Perl (Artistic License[15]). Una copia de la licencia puede ser encontrada en la distribución estándar de Perl.

Qmqtool fue diseñado por Jeremy Kister con el *software* "qmHandle" en mente, de cualquier manera ningún código fuente de este *software* fue usado dentro de Qmqtool [22].

Este programa hace uso de varias utilidades del *shell*, por lo que hay que asegurarse que la sintaxis que se usa en el programa concuerdan con el *shell* que se está utilizando.

Qmqtool soporta muchos argumentos, y cada uno debe de ser usado separadamente, a menos que sea expresamente permitido. Toda la sintaxis es descrita con el comando: *qmqtool -h*.

1.5.5. SNMP

Existen muchos libros acerca de SNMP, en particular, el libro de Douglas Mauro y Kevin Schmidt [23] es una buena referencia acerca del tema y desde la cual se recopiló la información de esta sección.

El SNMP (Simple Network Management Protocol, Protocolo Simple de Administración de Redes) es un protocolo estándar para administrar dispositivos en redes IP.

Muchas clases de dispositivos soportan el SNMP, incluyendo los routers, switches, servidores, estaciones de trabajo, impresoras, módems, y las fuentes de alimentación continuas (UPSs). Es bastante simple utilizar el SNMP para supervisar el estado de los routers, servidores, y otros equipos de la red, pero también puede ser usada para controlarlos, e incluso para enviar las páginas de alertas o para tomar alguna acción automática en caso de que se presenten problemas. La información que se puede supervisar se extiende de temas relativamente simples y estandarizados, como la cantidad de tráfico que fluye en o de una interfaz, a temas más recónditos específicos del vendedor del *hardware*, como temperatura del aire dentro del equipo, entre otros.

[15]La Artistic License es una licencia utilizada para ciertos paquetes de *software libre*; los más destacables son la implementación del estándar Perl, y la mayor parte de los módulos de CPAN de Perl, que tienen doble licencia bajo la Artistic License y la GNU General Public License (GLP). Fue escrita por Larry Wall. Fuente http://en.wikipedia.org/wiki/Artistic_License .

Administración y monitoreo de la red

El corazón de SNMP es un conjunto simple de operaciones y la información que estas operaciones recolectan, que dan al administrador la posibilidad de registrar los cambios de estado que experimentan los distintos equipos de la red.

También SNMP proporciona la habilidad de cambiar el estado de algún dispositivo SNMP. Por ejemplo se puede usar SNMP para detener una interfaz de red de un router o checar la velocidad a la cual alguna interfaz Ethernet está operando.

SNMP es usualmente asociado con la administración de equipos de red, pero también puede ser usado para administrar sistemas Unix, sistemas Windows, impresoras y más. Cualquier dispositivo corriendo *software* que permita la recuperación de información SNMP puede ser administrado. Esto incluye no sólo dispositivos físicos, sino también *software*, tal como servidores de Web y bases de datos.

Gestores y agentes

En SNMP hay dos clases de entidades: gestores y agentes. Un gestor es un servidor corriendo un tipo de *software* que puede administrar tareas de una red. Los gestores son también conocidos como Network Management Stations (NMSs). Un NMS es responsable de pedir información a los otros dispositivos de la red (agentes) y recibir notificaciones de estos.

Pedir información en el contexto de la SNMP, es el acto de buscar en un agente (router, switch, servidor Unix, etc.) datos. Esta información después puede ser usada para determinar si alguna clase de evento catastrófico ha ocurrido.

Un *trap* es la forma en que un agente le dice al NMS que algo ha ocurrido. Los *trap's* son enviados asíncronamente, es decir, no en repuesta a búsquedas desde el NMS. El NMS es responsable de cualquier manera de realizar una acción basado en la información que recibe del agente.

La segunda entidad, el agente, es un programa que corre en los dispositivos de la red que se necesitan administrar o monitorear. Por medio de este programa el agente provee de información administrativa al NMS, manteniendo su atención en los aspectos operacionales del dispositivo.

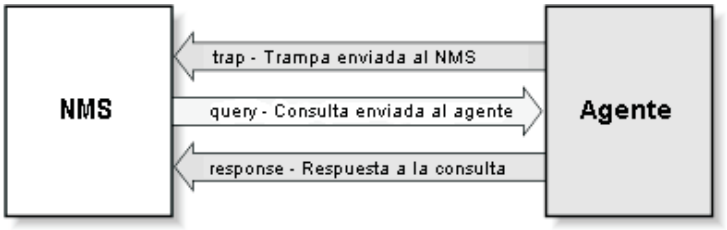


Figura 1.7: Relación entre un NMS y un agente.

Operaciones SNMP

Las tres operaciones básicas en SNMP son *snmpget, snmpset* y *snmpwalk*. Los nombres de las operaciones explican por sí mismas qué es lo que hacen: *snmpget* lee un valor de un dispositivo, *snmpset* pone un valor en un dispositivo, y *snmpwalk* lee una porción de el árbol MIB de un dispositivo.

¿Qué es un árbol MIB?

La *Managment Information Base*, MIB, puede ser pensada como una colección de información organizada jerárquicamente (ver figura 1.8).

Un objeto administrado (algunas veces llamado objeto MIB, objeto, o MIB) es uno de cualquier número de características específicas de un dispositivo administrado.

Cualquier clase de estado o información estadística que pueda ser accesada por el NMS es definida en una MIB. La *Structure of Management Information* (SMI) provee una manera de definir objetos administrados, mientras la MIB es la definición (usando la sintaxis SMI) de los objetos en sí mismos. En otras palabras, un MIB define un nombre textual para un objeto administrado y explica su significado.

Este estándar define variables tales como velocidad de la interfaz, MTU, octetos envíados por una interfaz, bytes recibidos por una interfaz, etc.

¿Qué es un OID?

Los objetos manejados están organizados en una jerarquía tipo árbol. Esta estructura es la base del esquema de nombres SNMP. Un ID de un objeto (OID) está conformado de una serie de enteros basados en los nodos del árbol, separados por puntos(.), ejemplo *.1.3.6.1.2.1.1.2*. Aunque hay una forma más sencilla de expresar esto, que está formada por una serie de nombres separados por puntos, cada uno representando un nodo en el árbol, ejemplo: *.iso.org.dod.internet.mgmt.mib-2.system.sysDescr*

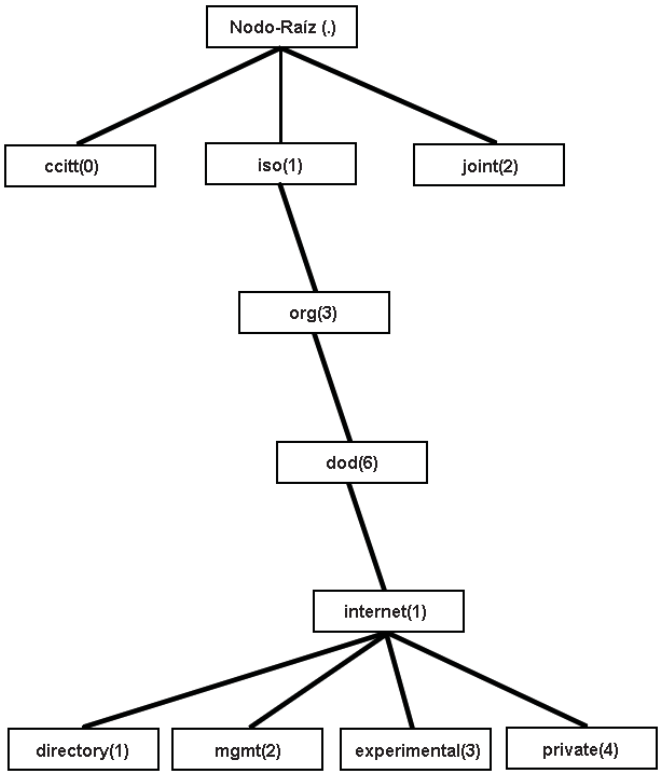


Figura 1.8: Ejemplo de árbol MIB.

Hay varias formas de ver los identificadores de objetos (OIDs) en Unix, pero la más cómoda es la que se presenta al final del siguiente ejemplo, dado que muestra las dos formas simultáneamente.

```
# snmptranslate -To | head
.1.3
.1.3.6
.1.3.6.1
.1.3.6.1.1
.1.3.6.1.2
.1.3.6.1.2.1
.1.3.6.1.2.1.1
.1.3.6.1.2.1.1.1
.1.3.6.1.2.1.1.2
#
# snmptranslate -Ts | head
.iso.org
.iso.org.dod
.iso.org.dod.internet
.iso.org.dod.internet.directory
.iso.org.dod.internet.mgmt
.iso.org.dod.internet.mgmt.mib-2
.iso.org.dod.internet.mgmt.mib-2.system
.iso.org.dod.internet.mgmt.mib-2.system.sysDescr
.iso.org.dod.internet.mgmt.mib-2.system.sysObjectID
#
```

```
# snmptranslate -Tl | head
.iso(1).org(3)
.iso(1).org(3).dod(6)
.iso(1).org(3).dod(6).internet(1)
.iso(1).org(3).dod(6).internet(1).directory(1)
.iso(1).org(3).dod(6).internet(1).mgmt(2)
.iso(1).org(3).dod(6).internet(1).mgmt(2).mib-2(1)
.iso(1).org(3).dod(6).internet(1).mgmt(2).mib-2(1).system(1)
.iso(1).org(3).dod(6).internet(1).mgmt(2).mib-2(1).system(1).sysDescr(1)
.iso(1).org(3).dod(6).internet(1).mgmt(2).mib-2(1).system(1).sysObjectID(2)
```

De lo anterior se puede concluir, por ejemplo, que el OID para verificar el espacio utilizado por un disco, se puede ver de las siguientes formas:

```
.iso(1).org(3).dod(6).internet(1).private(4).enterprises(1).ucdavis(2021).\
dskTable(9).dskEntry(1).dskUsed(8)
.iso.org.dod.internet.private.enterprises.ucdavis.dskTable.dskEntry.dskUsed
.1.3.6.1.4.1.2021.9.1.8
```

En sí un OID se podría ver cómo un objeto (nodo) de un árbol MIB, y que puede representar la cantidad de bytes enviados por una interfaz de red, o el espacio en disco de una partición linux, etc. Para encontrar el OID de algún objeto administrativo hay que consultar la documentación o buscarlo con ayuda del comando *snmptranslate*.

¿Cómo realizar consultas SNMP?

La sintaxis que debe utilizar un gestor para realizar una consulta SNMP hacia un agente es: *snmpget [OPTIONS] AGENT OID [OID]...*

Ejemplo:

```
# snmpget -u usuario -l authNoPriv -a MD5 -A password agente.com.mx \
.1.3.6.1.4.1.2021.8.1.100.1
UCD-SNMP-MIB::extResult.1 = INTEGER: 20
```

Esta es la base para realizar la recolección de datos para el monitoreo del sistema, donde es importante saber qué OID, de la información que se quiere consultar, se debe de utilizar.

Seguridad en SNMP

La seguridad ha sido el dolor de cabeza más grande de SNMP desde el principio. La autenticación en las versiones de SNMP 1 y 2 se reduce a una contraseña enviada en texto plano entre el gestor y el agente. Las contraseñas en texto plano por la red son muy fáciles de interceptar, por lo que representa un gran riesgo usar estas versiones.

El SNMPv3 resuelve los problemas de seguridad que plagaron a SNMPv1 y a SNMPv2. Por fortuna, la seguridad es el único cambio a SNMPv3; no hay cambios al protocolo.

1.5.6. RRDtool

RRDtool (Round Robin Database tool, herramienta para Base de Datos Round Robin) es un *software* bajo licencia GNU desarrollado por Tobias Oetiker, encargado de sistemas en el Instituto Federal de Tecnología Suizo [24]. Aunque es una base de datos, hay diferencias entre las bases de datos de RRDtool y otras bases de datos, las cuales serán listadas a continuación:

- El conjunto de comandos de RRDtool permiten almacenar datos y crear gráficas en base a estos datos. Otras bases de datos sólo almacenan datos pero no pueden crear gráficas.
- Se trata de una herramienta que trabaja con una base de datos que maneja planificación según Round-Robin. Esta técnica trabaja con una cantidad fija de datos, definida en el momento de crear la base de datos, y un puntero al elemento actual. La forma de almacenar es la siguiente: se trata la base de datos como si fuese un círculo, sobrescribiendo los datos almacenados con anterioridad una vez alcanzada la capacidad máxima de la misma. Esta capacidad máxima dependerá de la cantidad de información que se quiera conservar como historial.
- Otras bases de datos almacenan valores tal y como se les proporciona. RRDtool puede ser configurado para calcular la razón de cambio entre el valor anterior y el actual, y almacenarlo en su lugar.
- Una base de datos RRDtool está estructurada de tal forma que necesita datos a intervalos de tiempo predefinidos. Si no obtiene un nuevo valor durante el intervalo, almacena el valor UNKNOWN para ese intervalo. Así, RRDtool necesita el uso de *script's* que se ejecuten en intervalos regulares para asegurar un flujo constante de datos.

¿Para qué usar esta herramienta?

En este caso usamos RRDtool para guardar y procesar datos conseguidos a través de SNMP. Los datos que nos interesan son la cantidad de bytes (o bits) transferidos desde y hacia una red, uso del procesador, promedio de carga del procesador, espacio en las distintas particiones y uso de memoria.

Esta es un forma simple de realizar distintas tareas de administración en el *cluster* de correo, permitiéndo monitorizar, hacer estudios de rendimiento, disponibilidad, estadísticas de uso, alertas, etc.

1.5.7. Cacti

Cacti [16] es una interfaz gráfica para manejar RRDtool, que almacena toda la información necesaria para crear gráficos y generar los *script's* que necesita RRDtool para mantener la base de datos poblada de datos. La interfaz gráfica está programada en PHP.

Además de poder mantener gráficos, fuentes de datos, y archivos Round Robin en una base de datos, Cacti maneja la recolección de datos vía SNMP.

Fuentes de datos

Para manejar la recolección de datos, se puede alimentar a Cacti de rutas hacia algún comando/script externo junto con algún dato que el usuario necesite llenar. Luego Cacti recolectará estos datos con un crontab y poblará la base de datos Round Robin.

Las fuentes de datos pueden también ser creadas, correspondiendo a los datos en la gráfica. Por ejemplo, si se quisiera graficar los intervalos de tiempo que el comando *ping* regresa, se podría crear una fuente de datos usando un *script* con el comando *ping* que le regresará el valor en milisegundos. Después de definir las opciones de RRDtool tal como la forma en que se almacenarán los datos, se podrá definir información adicional que las fuentes de entrada de datos requieren, tal como el equipo al cuál se harán los *ping's* en este caso. Una vez que la fuente de datos ha sido creada, es mantenida automáticamente a intervalos de 5 minutos.

Gráficas

Una vez que han sido definidas las fuentes de datos, RRDtool con su comando graph puede crear las gráficas usando los datos. Cacti permite crear casi cualquier gráfica RRDtool usando todo tipos estándar de gráficas RRDtool.

No sólo se pueden crear gráficas en Cacti, también hay muchas formas de desplegarlas. Además de la "vista en forma de lista" y el "modo de vista rápida", Cacti permite crear la "vista en forma de árbol", la cual pone las gráficas en un árbol jerárquico para propósitos administrativos.

[16] La fuente de lo que se presenta en esta sección, esta en la referecia [25]

1.6. Modelando la distribución con UML

La naturaleza del sistema de correo que se está describiendo es evidentemente distribuido. El sistema requiere de la integración de un servidor Web, un servidor LDAP y los servidores de correo. Es claro que para esto es necesario contar con distintos componentes de *software* ejecutándose sobre distintos nodos de *hardware*.

Cualquiera que sea la razón, la naturaleza distribuida de la solución debe de ser documentada. UML [17] proporciona un modelo para mostrar la relación entre las plataformas de *hardware* y los componentes de *software*, en un diagrama llamado "Deployment Diagram" (Diagrama de Distribución), que a continuación describiré basándome en la fuente [26].

1.6.1. Descripción de los Diagramas de Distribución

Los Diagramas de Distribución de UML son modelos que representan la red de elementos de procesamiento de recursos y la configuración de los componentes de *software* en cada elemento físico.

Un Diagrama de Distribución está compuesto de nodos de *hardware*, componentes de *software*, dependencias de *software* y relaciones de comunicación. A continuación se presenta cómo se modela cada uno de ellos.

Nodo

Un nodo es un objeto físico que representa un recurso de procesamiento. El Diagrama de Distribución usa una vista de una caja tri-dimensional para representar cada plataforma o nodo.

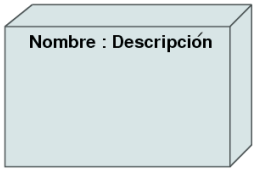


Figura 1.9: Modelando un nodo en UML.

[17] UML es el Lenguaje de Modelado Unificado (Unifie Modeling Language, en inglés) el cual provee un lengueje de modelado visual que es usado para especificar, visualizar, contruir y y documentar la "maquinaria" de un sistema de *software*.

Componente

Es una parte física reemplazable de un sistema que empaqueta su implementación y está adaptada a un conjunto de interfaces a las que proporciona su implementación.

Algunos componentes tienen identidad y pueden poseer entidades físicas, que incluyen objetos en tiempo de ejecución, documentos, bases de datos, etc. Los componentes existentes en el dominio de la implementación son unidades físicas en las computadoras que se pueden conectar con otros componentes, sustituir, trasladar, almacenar, etc.

Los componentes tienen dos características: Pueden mostrar un conjunto de componentes disponibles (una biblioteca de componentes) con sus dependencias. Pueden también mostrar un sistema configurado, con la selección de componentes (fuera de la biblioteca) usados para construirla.

La notación UML para un componente es un rectángulo grande con dos rectángulos pequeños sobre su costado izquierdo.

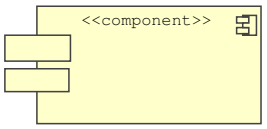


Figura 1.10: Modelando un componente en UML.

Relación de comunicación

Una relación de comunicación es una conexión física entre dos nodos o dos componentes ejecutables. Se puede definir esta relación por un nombre y un estereotipo. El nombre es el identificador de la misma y el estereotipo indica el protocolo de comunicaciones. Esta relación se representa en UML por una línea continua entre dos nodos o dos componentes.

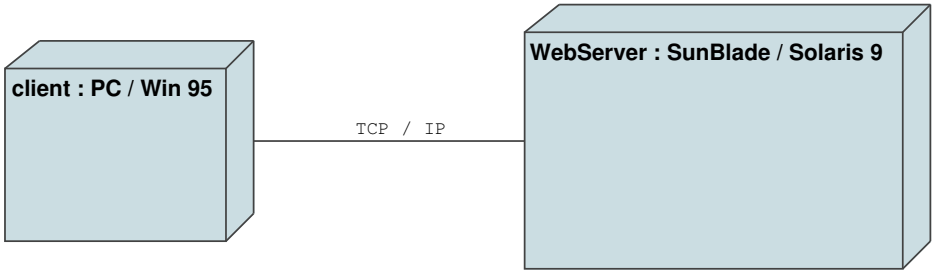


Figura 1.11: Modelando la relación de comunicación entre dos nodos en UML.

Dependencia

Las relaciones de dependencia se utilizan en los Diagramas de Distribución para indicar que un componente utiliza los servicios ofrecidos por otro componente. Esta relación es representada en UML por una línea punteada.

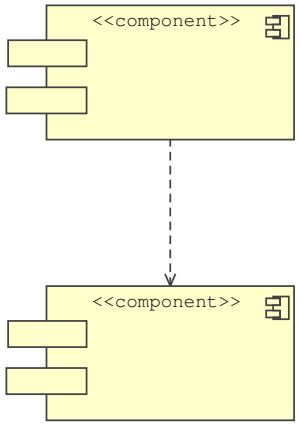


Figura 1.12: Modelando la dependencia entre dos componentes con UML.

En las figuras 2.7 y 4.8 de las subsecuentes capítulos se mostrará el Diagrama de Distribución del sistema de correo en dos etapas y configuraciones distintas.

Capítulo 2

El Sistema de Correo

En este capítulo se describe al sistema de correo en términos del proyecto y del producto final, como resultante del trabajo de la primera etapa y de las subsecuentes, para cumplir con las metas establecidas.

2.1. El Proyecto

Para contribuir al cumplimiento de las funciones sustantivas de la Universidad, particularmente en la formación de profesionistas y técnicos útiles a la sociedad, así como el ofrecer los servicios escolares que la institución requiere, la Dirección General de Administración Escolar (DGAE)estableció el proyecto de correo electrónico a los alumnos de la UNAM, con base a lo siguiente:

Disponer de los recursos de cómputo, *software* y procesos operativos que permitan el envío-recepción de mensajes a los alumnos con fines académicos, sociales y culturales, tendiente a fortalecer el conocimiento y aprovechamiento de las tecnologías de la información y las comunicaciones.

Los alcances del proyecto fueron:

- Ofrecer y administrar los servicios de correo electrónico a los alumnos activos de la Universidad.
- Disponer de una herramienta de apoyo a la docencia, para los alumnos y profesores.
- Proporcionar a las áreas académicas y de servicios, mecanismos de comunicación directa a la población escolar.
- Incidir en una mejor cultura informática de los alumnos.

2.2. El Sistema

El sistema de correo electrónico de los alumnos de la UNAM, se desarrolló y se implementó con la población de ingreso de la generación 2005, utilizando *software libre*, en equipo de cómputo dedicado y cuya aplicación considera los siguientes lineamientos:

- Los alumnos de la UNAM deben contar con el servicio de correo electrónico.
- La definición del correo es automática al ingreso del alumno.
- La duración de una cuenta de correo está sujeta al tiempo en que el alumno esté activo(inscrito) a un ciclo escolar.
- Si un alumno no se inscribe a un ciclo escolar, la cuenta de correo será suspendida, reactivándose cuando vuelva a inscribirse.
- Al ser informados de su cuenta ingresarán con su fecha de nacimiento como contraseña, pero inmediatamente la deben cambiar, por el que consideren apropiado y para hacer uso del correo, además de escoger y responder una pregunta secreta para poder cambiar su contraseña en caso de olvidarla. Deberán aceptar las reglas del servicio que se les ofrece.
- Se ofrecerá el servicio para cambiar contraseña, en caso de que el alumno la olvide, en el enlace ¿olvidaste tu passsword? donde primero se pedirá que el alumno se identifique y luego responda adecuadamente a su pregunta secreta.
- Las direcciones de correo electrónico de los alumnos están formadas de la siguiente manera: *No.-de-cuenta@escolar.unam.mx* .
- El alumno es responsable del manejo y uso de su cuenta de correo.
- Los alumnos acceden a los servicios de correo a través de la página *www.escolar.unam.mx*.
- La conexión de cualquier alumno al servicio de correo es exclusivamente bajo una conexión segura, al cual representa **https**.
- Los servicios de correo son los normales, como, enviar, recibir, contestar, reenviar, entre otros.
- Cada cuenta de correo dispone de 20 MB, sin excepción.
- Al llegar a la capacidad de uso de 18 MB (90 %) se les solicita que hagan la depuración de sus cuentas. Si no lo hacen y se aproximan al 100 % de uso de su capacidad, se hará un respaldo de las cuentas y se eliminarán los correos más antiguos, conservando el respaldo hasta un mes para que recuperen los correos eliminados.

- La página de acceso a los servicios de correo contendrá un enlace al concepto de preguntas frecuentes, un enlace de ayuda, para familiarizarse a las funciones del correo y un enlace a los instructivos y a las reglas del servicio.

2.3. Características

Las principales características del correo, a mi consideración, son:

- Acceso ininterrumpido al correo desde cualquier lugar por medio de un navegador, a través de la interfaz Horde.
- Interfaz con soporte para los idiomas Español e Inglés[1].
- Buzones maildir, es decir, cada mensaje se almacena como un archivo del sistema[2].
- Contactos personales.
- Acceso a través de clientes IMAP [3] (Protocolo de Internet de Acceso a Mensages).
- Qmail como MTA (Agente de Transferencia de Correo).
- Protección anti-virus y anti-spam proporcionada por *qmail-scanner* sin el uso de ningún explorador externo.
- Usuarios y dominios virtuales en una base de datos LDAP.
- Soporte para desarrollo en *cluster* en los servidores de correo.
- Soporta cuotas de espacio en los buzones de los usuarios.
- Soporte para SHA[4], SSHA[5], MD5[6], SMD5[7], MD4[8] y RIPE-MD160[9].

[1]También se puede habilitar el soporte para otros idiomas, entre los cuales están el Italiano, Alemán y Francés, pero no se habilitan debido a que si se hace una modificación a la aplicación, no hay personal por el momento que pueda hacer las traducciones para estas modificaciones.

[2]A diferencia de los buzones mbox, donde los mensajes se almacenan en un solo archivo.

[3]IMAP es un protocolo de red de acceso a mensajes electrónicos almacenados en un servidor.

[4]SHA (Secure Hash Algorithm, Algoritmo de Hash Seguro.)

[5]SSHA (Slated Secured Hashing Algorithm , Algoritmo de Hash Seguro Mejorado).

[6]MD5 (Message-Digest Algorithm 5, Algoritmo de Resumen del Mensaje 5).

[7]SMD5 (Message-Digest Algorithm 5, Algoritmo de Resumen del Mensaje 5 Mejorado).

[8]MD4 Anterior al MD5.

[9]RIPE-MD160 (RACE Integrity Primitives Evaluation Message Digest, primitivas de integridad del resumen del mensaje).

2.4. El cluster

De acuerdo al cuadro 2.1, se puede observar que el cluster está constituido por nueve equipos, en donde, para que puedan operar en conjunto, se necesita que todos los equipos tengan instalado el *software qmail-ldap*.

Núm. de equipo	Sistema operativo	Nombre del equipo	Función
1	Solaris 9	escolar	Servidor Web(HTTPS) y Qmail(SMTP)
2	Redhat 9	barajas	Servidor LDAP
3	Redhat 9	boole	Servidor Qmail, almacena correo
4	Redhat 9	babbage	Servidor Qmail, almacena correo
5	Redhat 9	bernoulli	Servidor Qmail, almacena correo
6	Redhat 9	banach	Servidor Qmail, almacena correo
7	Redhat 9	bussey	Respuesta rápida a incidentes
8	Redhat 9	escolarr	Respuesta rápida a incidentes (Web)
9	Redhat 9	bolzano	Respaldos

Cuadro 2.1: Equipos que conforman el *cluster*.

Dentro de estos nueve equipos se tiene uno dedicado a brindar el servicio Web, es decir, el equipo que brindará la funcionalidad para permitir que el cliente utilice el servicio mediante una interfaz. Este equipo en particular será el que reciba todas las peticiones de Internet de los servicios Web y SMTP por lo que se considera un elemento crítico para el servicio.

Por otra parte, se tiene un equipo dedicado a mantener y operar el directorio LDAP, que es donde se encuentra almacenada la información de los usuarios dados de alta en el sistema. Este directorio proporcionará soporte para permitir que los componentes de Qmail se puedan comunicar de un equipo a otro, y de esta manera, hacer posible la entrega de mensajes y la autenticación de usuarios distribuidamente en el sistema.

En este otro caso, dado que el servicio LDAP es básico para realizar casi todas las tareas del *cluster*, es considerado también un servicio crítico para el servicio.

La decisión para poner los servicios, Web y LDAP en dos equipos separados, obedece únicamente a que se intentan distribuir los servicios que se consideran más críticos.

Asimismo, se tienen cuatro equipos para almacenar los buzones de los usuarios.

Por último se tiene un equipo para almacenar semanalmente los respaldos de los demás equipos y hacer pruebas, además de otros dos equipos para respuesta rápida a incidentes.

2.5. Distribución y Comunicación

En esta sección se mostrará cómo se comunican los equipos dentro del *cluster*, al momento de enviar o recibir correo, al momento de autenticar a un usuario, así como a la hora de recolectar los mensajes de un buzón para mostrarlos en un navegador.

Para esto lo primero que se necesita es tener en los DNS's de la UNAM, dado de alta el servidor Web, *escolar.unam.mx*, ya que, este equipo será el único que tenga contacto directo con los usuarios.

Por su parte, los otros equipos del *cluster* no deben de ser visibles mediante un DNS, debido a que para los usuarios, es transparente la manera en que se almacenan los correos distribuidamente en los distintos equipos del *cluster*. En otras palabras, para los usuarios es un solo equipo el que realiza toda la operación.

Aunque los equipos no se encuentran en una red privada, están protegidos individualmente, para evitar recibir conexiones de equipos y servicios no autorizados. Además los equipos están configurados, de manera que sólo tienen instalados los servicios necesarios, y nada más.

Para esquematizar ésto, se presentarán una serie de diagramas, donde se describirá cada etapa en la comunicación de los equipos, asociando un número entre la explicación de lo que pasa y las líneas (continuas o punteadas) en los diagramas.

A continuación se presenta la figura 2.1 junto con los siguientes puntos que se tienen que considerar:

1. El servidor Web recibe todas las peticiones de Internet.
2. El servidor Web se comunica con los equipos del *cluster* para proporcionar las funcionalidades, tales como, autenticar a un usuario, descargar el buzón de un usuario, entregar un mensaje, enviar un mensaje o cambiar la contraseña de un usuario.
3. Los equipos del *cluster* se comunican y se transfieren mensajes de correo mediante la intervención del servidor LDAP, dado que éste brinda la información necesaria para permitir la comunicación de los distintos componentes de *software* dentro del *cluster*.
4. El servidor Web responde a las peticiones de los usuarios, una vez que éstas ya han sido procesadas por el *cluster*.
5. Cada servidor de correo realiza el regreso de los mensajes que por alguna razón no se pudieron entregar (bounces), enmascarando el mensaje para que el receptor lo reciba como si el mensaje fuera enviado por el servidor Web. El regreso de correo hacia otros dominios, es la única actividad donde se permite que los equipos del *cluster*, exceptuando el servidor Web, se comuniquen con servidores remotos.

Otras consideraciones importantes son:

- Cada equipo está protegido individualmente, mediante un *firewall de host*, no permiendo conexiones no autorizadas.
- Los servicios Web y LDAP son críticos para el funcionamiento del *cluster*, por lo que se encuentran distribuidos en distintos equipos.
- Los equipos 3, 4, 5 y 6 mostrados en la figura 2.1, contienen los buzones de los usuarios, en los directorios del sistema */var/qmail/maildirs/<usuario>*.
- Hay dos equipos para respuesta rápida a incidentes y uno más para respaldos, los cuales están predispuestos para sustituir a cualquier equipo en caso de falla.
- Un equipo de respuesta rápida a incidentes, puede sustituir al servidor LDAP copiando una versión reciente de la base datos LDAP, o a cualquier servidor de correo (equipos 3, 4, 5, 6) copiando los buzones de los usuarios.
- El otro equipo de respuesta rápida a incidentes , puede sustituir al servidor Web, copiando la base de datos que contiene los contactos de los usuarios.

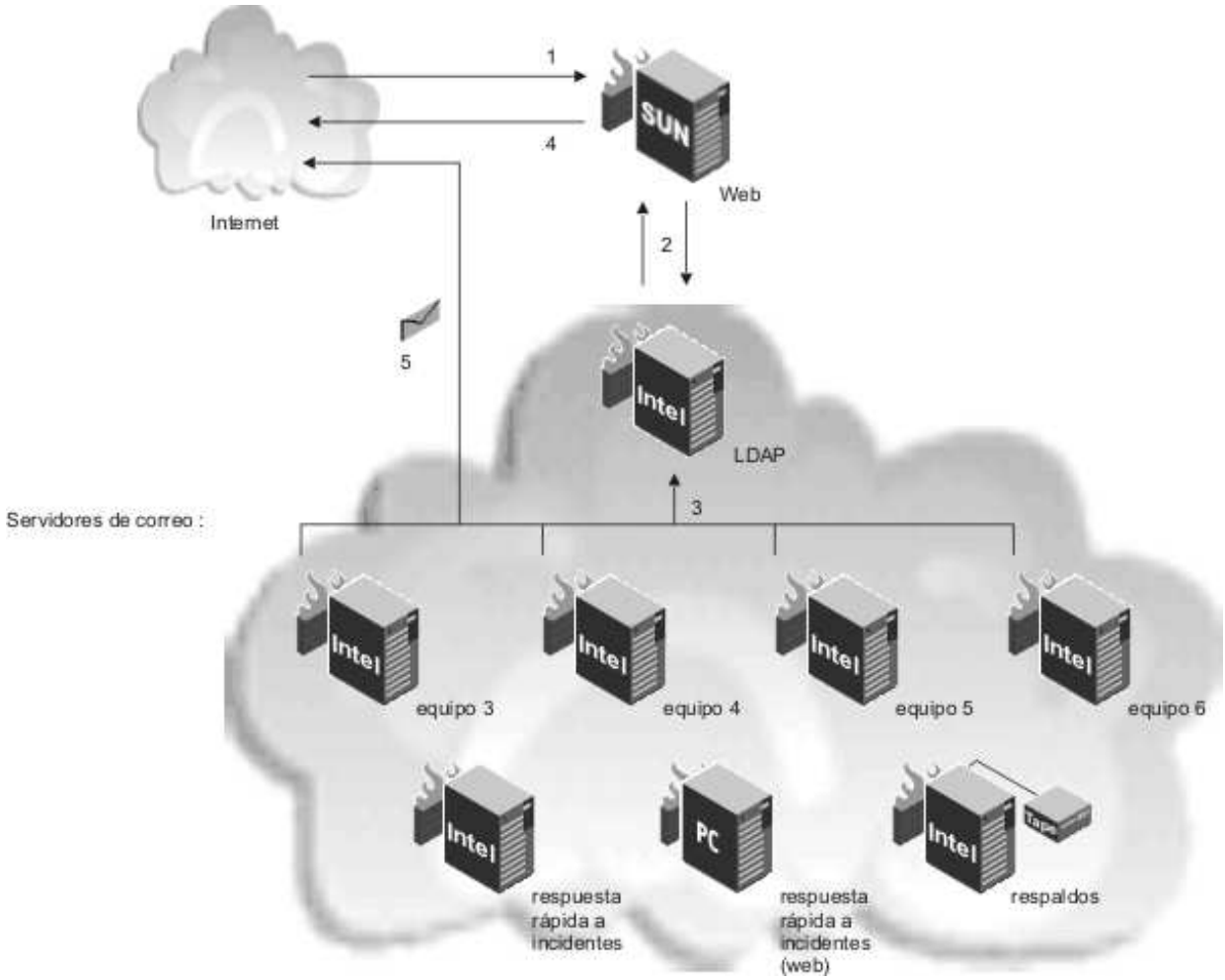


Figura 2.1: *Cluster* de correo en Internet.

A continuación, en las siguientes secciones, se muestra por cada funcionalidad como trabaja el *cluster*.

2.5.1. Autenticación de un usuario

Para autenticar un usuario el sistema utiliza el protocolo IMAP, en específico su módulo *auth_imap* integrado a Qmail, para comparar la información dada por los clientes contra la información almacenada en el directorio LDAP.

Además el protocolo IMAP también tiene la funcionalidad de recolectar el contenido de un buzón de un usuario para mostrarlo en el navegador.

Cuando el soporte para *cluster* está habilitado en *qmail-ldap*, *auth_imap* puede recolectar mensajes de los clientes IMAP, aún si los mensajes están almacenados en otro miembro del *cluster*, en donde, *auth_imap*, en favor del cliente IMAP, puede conectarse al otro equipo del *cluster* y obtener los mensajes, regresándolos al cliente IMAP. Esto es llamado "retransmisión de sesión"[10].

Para entender cómo funciona la autenticación *qmail-imap* y la retransmisión de sesión IMAP, este trabajo contiene la figura 2.2 que se puede explicar de la siguiente manera:

1. El usuario introduce desde la página de inicio de la aplicación su "nombre de usuario" y "contraseña", para luego hacer la petición de autenticación presionando el botón "Entrar". Una vez realizado esto la información es cifrada y enviada al servidor Web.

2. Ya que el servidor Web recibió la información, dispone del programa *auth_imap* para enviar el "nombre de usuario" y "contraseña" al servidor LDAP, para realizar la operación de búsqueda.

3. El programa *auth_imap* en el servidor LDAP busca en el directorio la entrada que contiene el atributo "nombre de usuario". Cuando encuentra la entrada, el programa verifica que la contraseña proporcionada por el usuario coincida con el campo "contraseña" que el usuario tiene en el directorio LDAP. Si coincide y si la cuenta está activa, entonces el programa recolecta los datos del directorio en cuestión, tal como, *uid, mailHost, mailAddress, maildir*, etc. y se los envía al servidor Web.

4. El servidor Web obtiene el campo *mailHost* (equipo k) de la información que recibe, y la compara con el archivo de control de Qmail, `~control/me`[11], que en este caso corresponde al nombre del servidor Web. No coinciden, dado que el usuario no tiene su cuenta en

[10] El "redirecionamiento de sesión", *auth_imap*, permite recolectar mensajes de los clientes IMAP, cuando los mensajes están almacenados en otro miembro del *cluster*, en donde, *auth_imap* puede conectarse a otro equipo del *cluster* para obtener los mensajes y regresarlos al cliente IMAP.

[11] Archivo de control de Qmail que contiene el nombre del servidor

este servidor, por lo que *auth_imap* redirecciona la conexión al servidor indicado por el campo *mailHost* (equipo k) donde se encuentra el buzón del usuario indicado.

5. El programa *auth_imap* en el *equipo k* intenta autenticarse tal y como se hizo en los dos pasos anteriores, es decir, realiza la operación de búsqueda LDAP, recibe la información del directorio, obtiene el *mailHost* del usuario y lo compara con el archivo de control `~control/me` (equipo k), que en este caso deben coincidir.

6. Dado lo anterior el programa *qmail-imapd* establece una sesión IMAP, entre el *equipo k* y el cliente, retransmitiendo la sesión a través del servidor Web. Esto implica además la recolección de los mensajes del buzón del usuario para enviarlos al cliente de nuevo retransmitiéndolos através del servidor Web.

7. El servidor Web retransmite la sesión IMAP y muestra el buzón del cliente en su navegador.

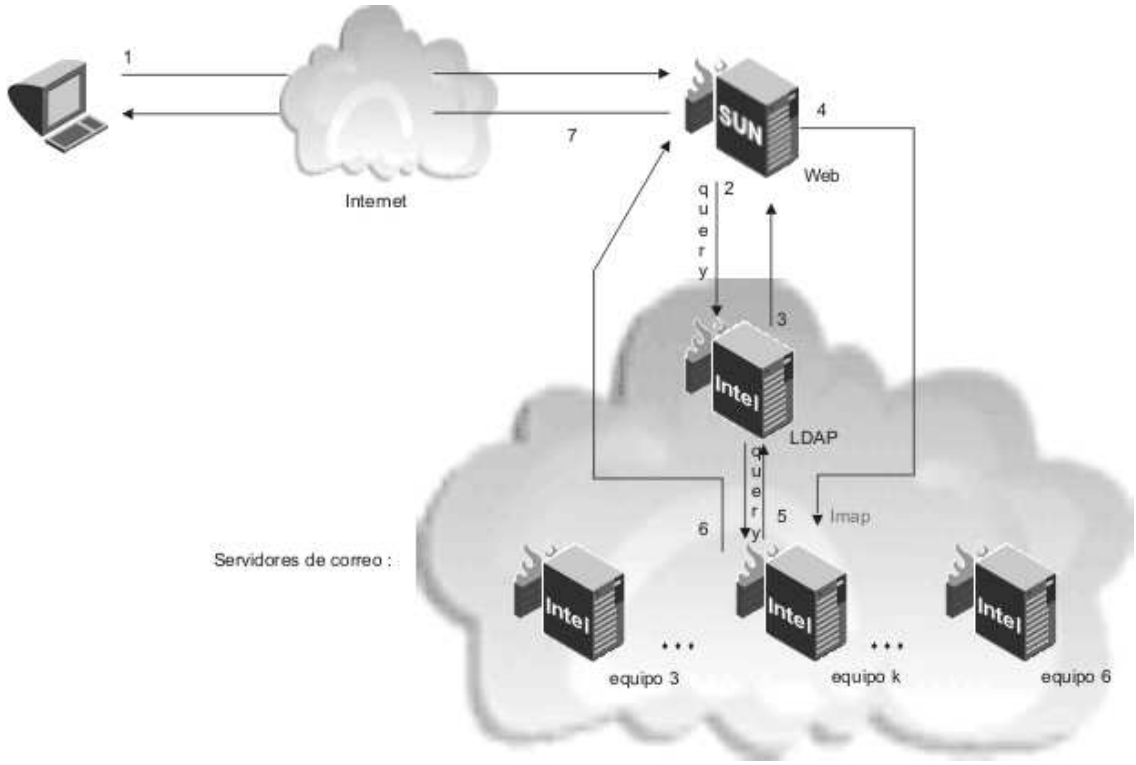


Figura 2.2: Funcionamiento del *cluster* al autenticar un usuario.

2.5.2. Envío y recepción de correo local

En esta sección se mostrará cómo se procesa correo de forma local, es decir, el cómo se procesan los mensajes que parten de un servidor del *cluster* y están dirigidos a otro o al mismo servidor, dentro del *cluster*.

Como se muestra en la figura 2.3 el sistema de correo procesa correo local de la siguiente manera:

1. Desde la aplicación Web para enviar mensajes de correo, el usuario llena los campos: destinatario, asunto, etc. (partes del encabezado del mensaje), escribe la parte principal del mensaje y en caso de requerirlo, agrega los archivos adjuntos. Hecho esto, el usuario debe notificar al servidor que comience a procesar el correo dando la instrucción "enviar".

2. El servidor Web recibe la petición, formatea el mensaje y se lo da a Qmail para que sea procesado por varios de sus subprogramas:

   - *qmail-inject* interpreta el mensaje e incorpora información extra al encabezado del mensaje, para que pueda ser entregado;
   - *qmail-queue* pone el mensaje en la cola de correos de salida de Qmail;
   - *qmail-send* toma el mensaje de la cola de correos de salida, lo identifica como correo local y lo pasa a *qmail-lspawn*;
   - *qmail-lspawn* extrae el "uid" del destinatario del mensaje y lanza la operación de búsqueda LDAP, enviando el "uid" del destinatario.

3. El servidor LDAP recibe la petición y busca en el directorio la entrada que contiene el "uid" del destinatario. Una vez que localiza la entrada, extrae los datos del destinatario (*mailHost, maildir*, etc.) y lo obtenido lo regresa al servidor Web.

4. El programa *qmail-lspawn* en el equipo Web recibe la respuesta, extrae el campo *mailHost* del destinatario del mensaje y luego por medio de *qmail-qmqpc* envía el mensaje al servidor indicado por el campo *mailHost*, (*equipo k* en la figura 2.3).

5. El *equipo k* recibe el mensaje y por medio de Qmail realiza un proceso similar a los puntos 2, 3 y 4:

   - *qmail-queue* pone el mensaje en la cola de correos de salida de Qmail;
   - *qmail-send* toma el mensaje de la cola de correos de salida, lo identifica como correo local y lo pasa a *qmail-lspawn*;

- *qmail-lspawn* extrae el "uid" del destinatario del mensaje y lanza la operación de búsqueda LDAP, enviando el "uid" de este;
- *qmail-lspawn* recibe la respuesta del servidor LDAP, extrae el *mailHost* del destinatario, y se percata que el mensaje debe de ser entregado a un buzón que está en el mismo servidor.

6. Dado lo anterior el mensaje es pasado a *qmail-local* para ser almacenado en el buzón de destino.

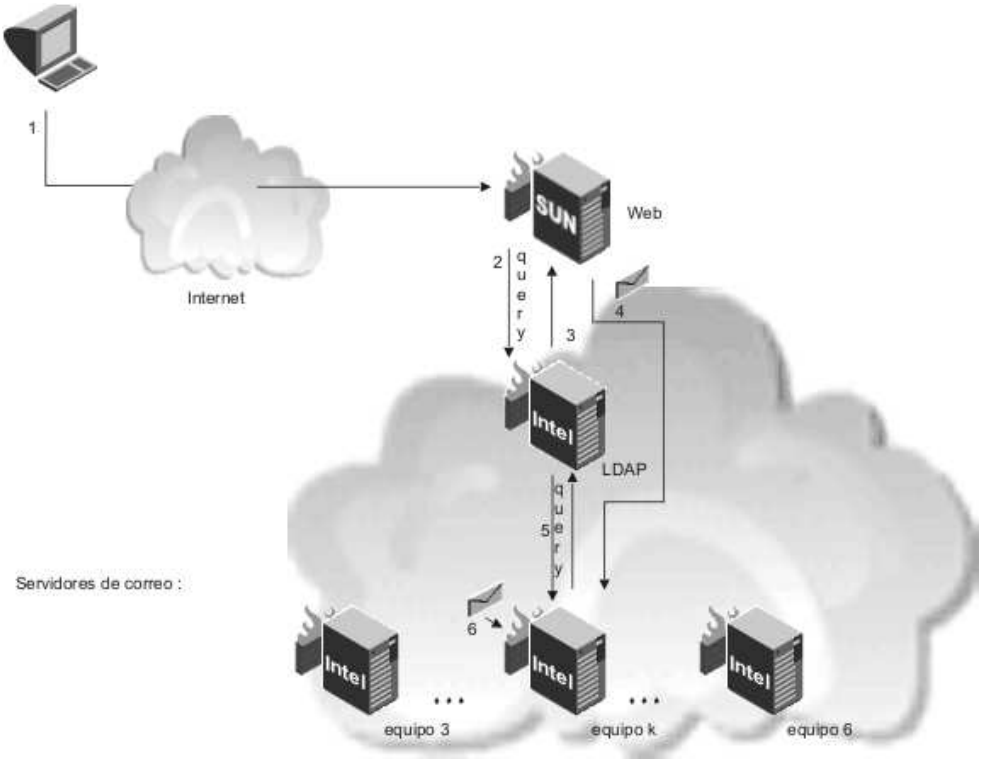


Figura 2.3: Funcionamiento del *cluster* al enviar y recibir correo local.

2.5.3. Envío de correo remoto

Para el envío de mensajes de correo remoto, como se puede apreciar en la figura 2.4, el funcionamiento del *cluster* es como sigue:

1. Desde la aplicación Web para enviar mensajes de correo, el usuario llena los campos: destinatario, asunto, etc. (partes del encabezado del correo), escribe la parte principal del mensaje y en caso de requerirlo, agrega los archivos adjuntos. Hecho esto, el usuario debe notificar al servidor que comience a procesar el correo dando la instrucción "enviar".

2. El servidor Web recibe la petición, formatea el mensaje y se lo da a Qmail para que sea procesado por varios de sus subprogramas:

   - *qmail-inject* interpreta el mensaje e incorpora información extra al encabezado del mensaje, para que pueda ser entregado;
   - *qmail-queue* pone el mensaje en la cola de correos de salida de Qmail;
   - *qmail-send* toma el mensaje de la cola de correos de salida, lo identifica como correo remoto y lo pasa a *qmail-rspawn*;
   - *qmail-rspawn* planifica la entrega y transfiere el mensaje a *qmail-remote*.

3. Finalmente Qmail por medio del subprograma *qmail-remote* envía el mensaje al servidor de correo remoto.

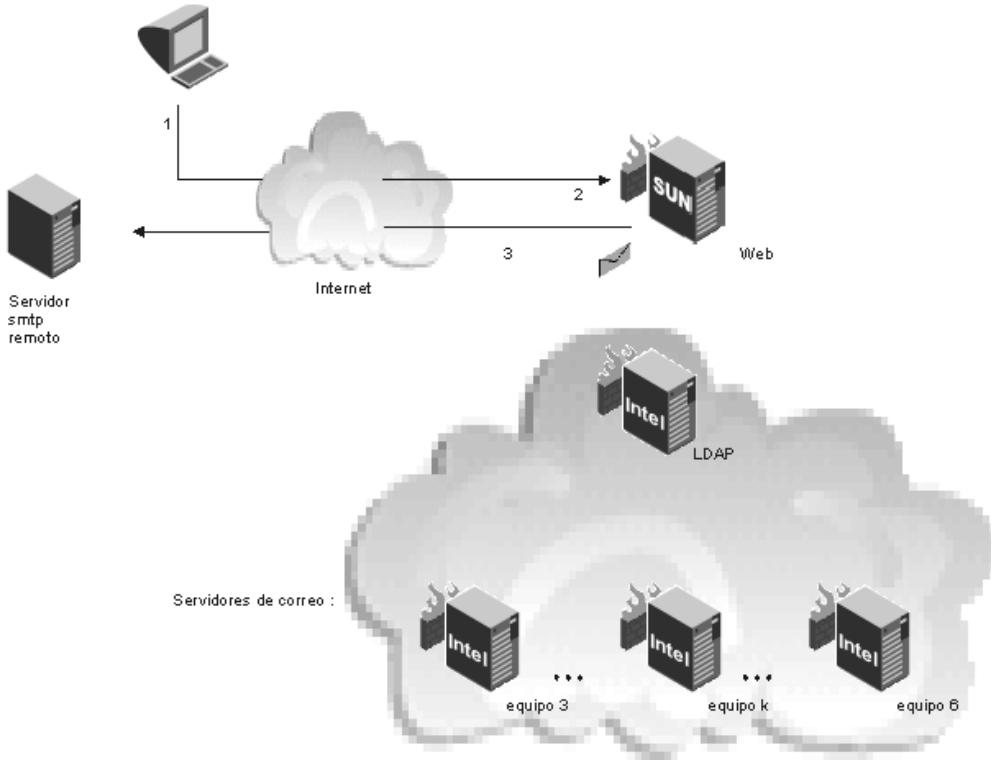


Figura 2.4: Funcionamiento del *cluster* al enviar correo.

2.5.4. Recepción de correo remoto

1. El programa *qmail-smtpd* en el servidor Web, recibe el mensaje de correo del servidor remoto vía SMTP[12].

2. Luego Qmail en el servidor Web, comienza a procesar el mensaje de la siguiente manera:

   - *qmail-smtpd* transfiere el mensaje a *qmail-queue*;
   - *qmail-queue* pone el mensaje en la cola de correos de salida de qmail;
   - *qmail-send* toma el mensaje de la cola de correos de salida, lo identifica como correo local y lo pasa a *qmail-lspawn*;
   - *qmail-lspawn*, lanza la operación de búsqueda LDAP enviando el "uid" del destinatario de este.

3. El servidor LDAP recibe la petición y busca en el directorio la entrada que contiene el "uid" del destinatario. Una vez que localiza la entrada, extrae los datos del destinatario (*mailHost, maildir*, etc.) y lo obtenido lo regresa al servidor Web.

4. El programa *qmail-lspawn* en el equipo Web recibe la respuesta, extrae el campo *mailHost* del destinatario del mensaje y por medio *qmail-qmqpc* lo envía el mensaje al servidor indicado por el campo *mailHost*, (*equipo k* en la figura 2.5).

5. El *equipo k* recibe el mensaje y por medio de Qmail realiza un proceso similar a los puntos 2, 3 y 4:

   - *qmail-queue* lo pone en la cola de correos de salida de Qmail;
   - *qmail-send* toma el mensaje de la cola de correo de salida, lo identifica como correo local y lo pasa a *qmail-lspawn*;
   - *qmail-lspawn* extrae el "uid" del destinatario del mensaje y lanza la operación de búsqueda LDAP, enviando el "uid" de este;
   - *qmail-lspawn* recibe la respuesta del servidor LDAP, extrae el *mailHost* del destinatario, y se percata que el mensaje debe de ser entregado a un buzón que está en el mismo servidor.

6. Dado lo anterior el mensaje es pasado a *qmail-local* para ser almacenado en el buzón de destino.

[12]SMTP Simple Mail Transfer Protocol

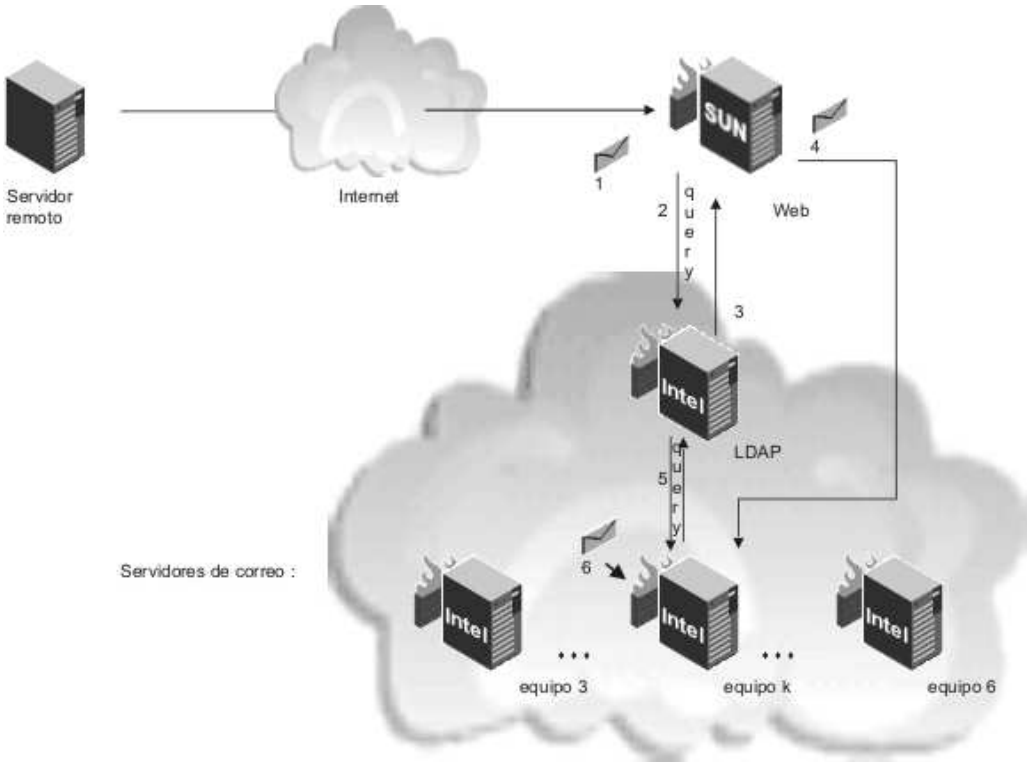


Figura 2.5: Funcionamiento del *cluster* al recibir correo.

2.5.5. Cambio de contraseña

El sistema proporciona dos formas de cambiar contraseña. Una de ellas, caso (a) se proporciona cuando el usuario está ya autenticado y desea desde el menú de la aplicación Web cambiarla. La otra forma, caso (b) se dispone en caso de que el usuario haya olvidado su contraseña, para lo cual desde la página de inicio se tiene el acceso "olvidaste tu contraseña", donde el usuario debe de proporcionar su número de cuenta y otros datos personales, para que el sistema le asigne una nueva contraseña.

Ahora bien, para mostrar cómo funciona el *cluster* para cambiar contraseñas se tiene la figura 2.6 y se realiza de la siguiente manera:

1. El usuario entra a cualquiera de las dos aplicaciones Web para cambiar contraseña, dependiendo de lo que necesite, posteriormente proporciona sus datos, en el caso (a) es antigüa contraseña y nueva contraseña, en el caso (b) es número de cuenta y otros datos. Se envían los datos al servidor LDAP para verificar que la información que se proporcionó es correcta.

2. El servidor LDAP busca la información, la compara con lo que tiene en su directorio y envía la respuesta.

3. El servidor Web recibe la respuesta y si ésta confirma que la información que dio el usuario es correcta, entonces se continúa con el siguiente paso.

4. Una vez que se ha confirmado la información, entonces se envía la nueva contraseña al servidor LDAP para actualizar la información.

5. El servidor LDAP hace la actualización de la contraseña en el directorio del usuario indicado y envía la notificación al servidor Web.

6. A su vez, el servidor Web envía una notificación al usuario para indicarle que su contraseña ha sido actualizada.

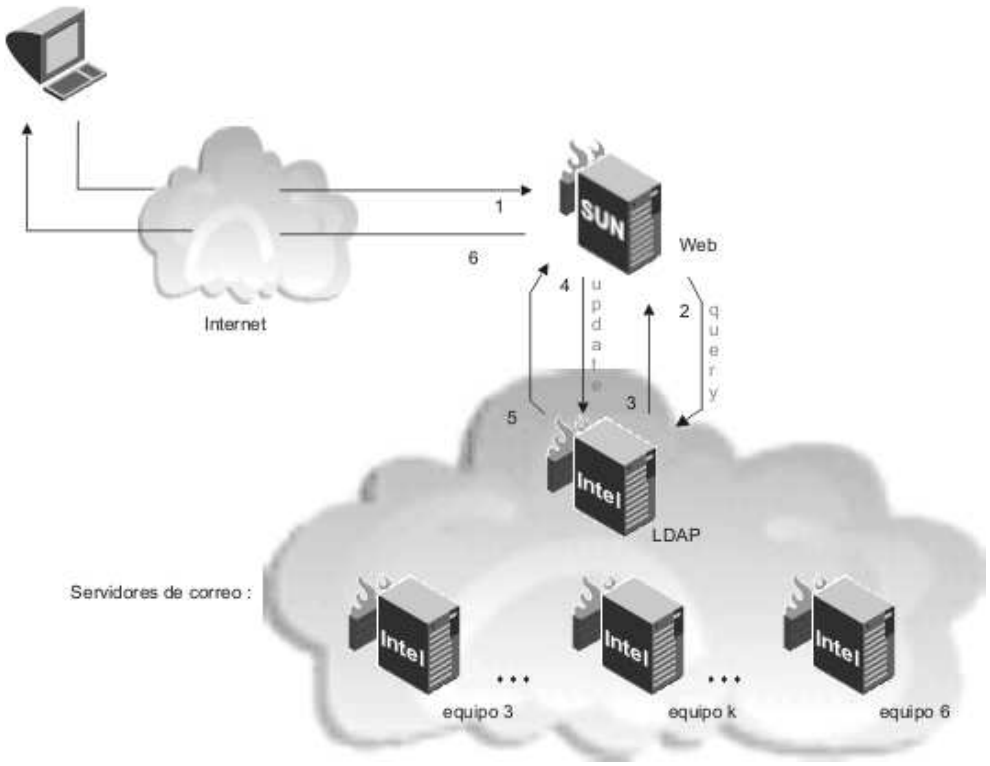


Figura 2.6: Funcionamiento del *cluster* al cambiar el contraseña de un usuario.

2.5.6. Diagrama de Distribución UML, del Sistema de Correo

Aquí se presenta el diagrama de distribución del sistema de correo, de acuardo a lo que se presentó en el primer capítulo en la sección 1.6.

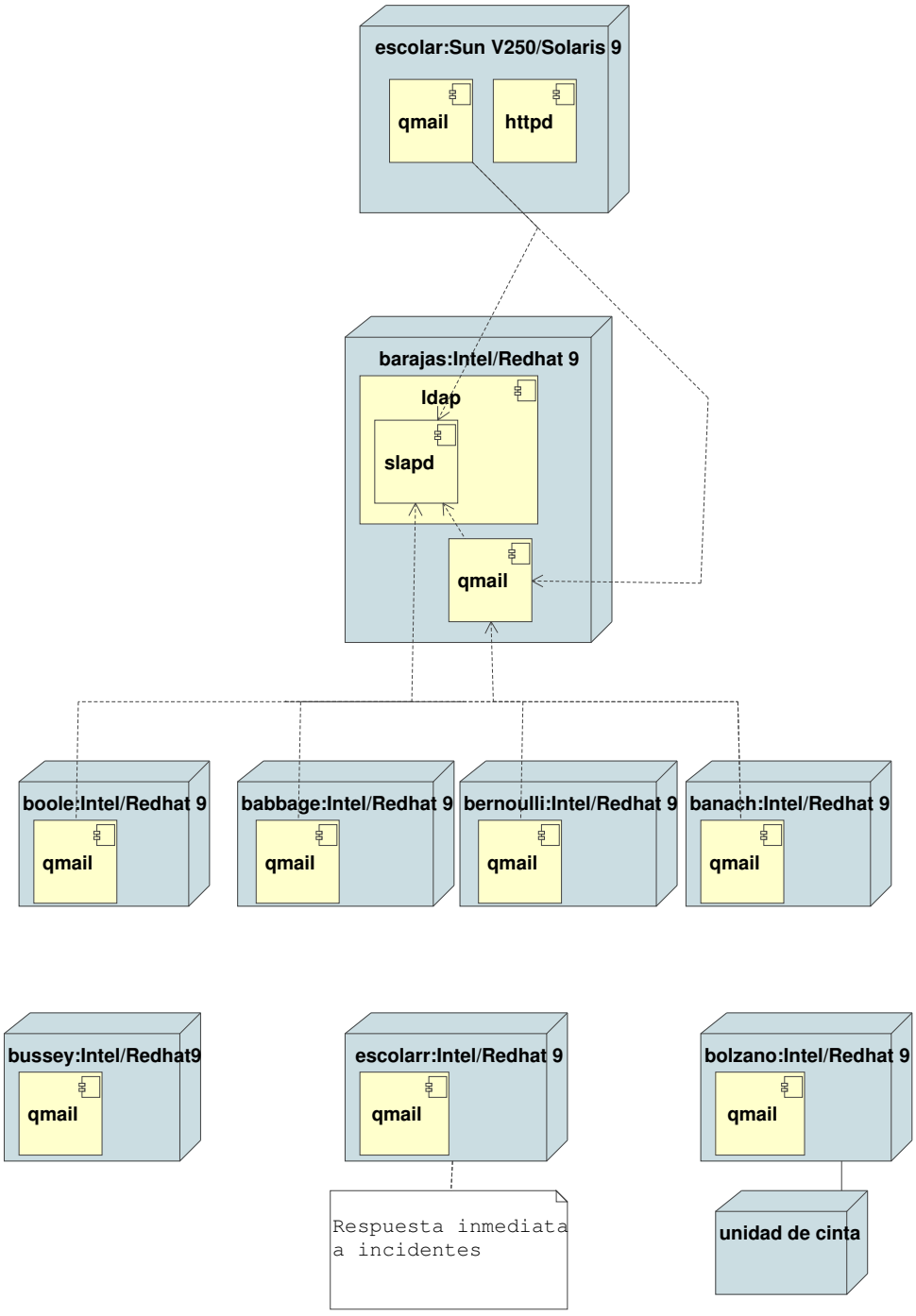


Figura 2.7: Distribución inicial del sistema de correo.

2.6. Situación del proyecto

El reto fundamental del desarrollo del proyecto fue lograr que la aplicación de los servicios de correo para los alumnos de la UNAM, se hiciera con *software libre*. El utilizar equipo de cómputo armado, el ofrecer los servicios con una buena administración del sistema y el mantener la operación continua, aumentaron los desafíos. Los retos del futuro establecieron la consolidación del sistema, que todos los alumnos utilicen el recurso, que realmente sea una herramienta para la actividad docente y que las diversas instancias universitarias aprovechen las facilidades de comunicación directa con los alumnos por esta vía.

En la primera etapa del proyecto, que implicaba 70,000 cuentas de correo[13], lo relevante fue que se logró concretar la funcionalidad del sistema que aquí se ha descrito, lo que implicaba contar con los recursos de cómputo, *software* y procesos operativos para el envío y recepción de mensajes con fines académicos.

El programa de trabajo, previó continuar con el proyecto, incorporar a los alumnos de las generaciones 2006 y 2007 para llegar a administrar 210,000 cuentas de correo [14], posteriormente, hacer una validación del sistema y realizar los ajustes necesarios para incorporar al resto de alumnos, con el fin de llegar a las cerca de 300,000 cuentas de usuarios [15] activos, por ciclo escolar.

[13]Esta cifra fue proporcionada por personal autorizado de la DGAE-UNAM.

[14]Esta cifra contempla dos generaciones de de alumnos. Fuente: personal autorizado de la DGAE-UNAM.

[15]Esta cifra contempla a todos los alumnos de la UNAM. Fuente: personal autorizado de la DGAE-UNAM.

Capítulo 3

Funcionalidad

El presente capítulo describe las observaciones que se hicieron al poner en funcionamiento el sistema, en su primera etapa.

3.1. Puesta en producción

El sistema de correo implementado en julio de 2004, está basado en el *software libre* Qmail, con la interfaz gráfica Horde, un servidor Web con apache (*http://escolar.unam.mx*) y ocho equipos de servidores para rack, funcionando en un modelo de *cluster*, uno de ellos como servidor de autenticación de usuarios, cuatro equipos para la administración de correos (CU, Unidades multidisciplinarias, ENP, CCH y posgrado), dos equipos para la administración de los servicios y respuesta en caso de fallos y un equipo para respaldo de información.

El desarrollo del proyecto en la primera etapa, consistió en armar los equipos de cómputo, instalarlos y ponerlos en funcionamiento para los servicios previstos, probar y hacer las adaptaciones al *software libre*, de acuerdo a las estrategias de servicio y operación e implementar la aplicación, dar de alta a los usuarios, monitoreo e incorporar nuevas facilidades operativas.

3.2. Uso del Sistema

El objetivo de esta sección es evaluar la eficacia del Sistema de Correo, una vez que se seleccionó su configuración y se puso en marcha. Esto supone establecer un conjunto de disposiciones que den solidez al proyecto desde su inicio, para garantizar que los objetivos actuales y futuros sean cumplidos.

Partiendo de una extensa revisión del funcionamiento de los diferentes aspectos del sistema, se intentará determinar el nivel de eficacia de éste, mediante un estudio de las fallas que reporten usuarios o que mediante el monitoreo cotidiano se detecten.

Con esto se pretende dar respuesta a cuestionamientos como ¿qué parámetros se deben de tomar en cuenta para medir la eficacia del sistema?, ¿cuál es el nivel de eficacia del sistema en comparación con otros sistemas de correo?, ¿qué tan útil es el sistema de correo para los alumnos?, ¿cómo mejorar el servicio?.

Una de las conclusiones a las que se llegaron fue que para medir la eficacia del sistema, había que procurar que el desempeño de ciertos aspectos del sistema estuvieran en niveles aceptables. Dentro de estos aspectos se encuentran:

- Rapidez del servicio
- Continuidad del servicio
- Seguridad del servicio
- Consistencia de la información
- Monitoreo del sistema

Pero a qué se refieren cada uno de estos aspectos y en qué se relacionan con el sistema de correo, es lo que verá en los siguientes párrafos.

Rapidez del servicio

Se debe de suponer que por la cantidad de usuarios que maneja el sistema, se tengan que procesar cantidades importantes de sesiones Web, sesiones imap y correo, cosa que puede llegar a comprometer el tiempo de respuesta del servicio.

Debido a lo anterior se debe de verificar que la rapidez del sistema sea la adecuada para el uso que se tiene considerado. En este caso se detectaron factores internos del *cluster* que afectaban el rendimiento del sistema, los cueles se podrían enumerar en orden de importancia como sigue:

1. El tiempo de respuesta del servidor LDAP
2. Carga del servicio Web (cantidad de sesiones abiertas)
3. Carga del servicio SMTP (cantidad de correos que se reciben en un lapso)
4. El estado de la cola de correos de Qmail en los miembros del *cluster*

Continuidad del servicio

Debido a la cantidad de equipos involucrados en el proyecto, existe la posibilidad de que algún equipo o servidor(demonio) deje de funcionar, mientras todo lo demás continúa funcionando adecuadamente.

Por lo que se observó, en el tiempo que el sistema ha estado funcionando, hay casos donde la falla de un equipo o un servicio sólo afectó a cierta cantidad de usuarios o a una funcionalidad en especial.

Dado lo anterior se debe de tener cuidado a la hora de verificar el estado del funcionamiento del *cluster*, ya que aparentemente el *cluster* puede funcionar correctamente, pero puede ocurrir que un módulo del servicio no. Esto puede comprometer la integridad de la información del sistema.

También se dieron casos donde la falla de un servicio, provocó que el *cluster* dejara de funcionar en su totalidad. Un ejemplo de esto es el directorio LDAP. Cuando este servicio deja de funcionar todo el servicio falla, desde la autenticación de usuarios hasta la entrega y recepción de mensajes.

Dado esto, se encontraron factores que había que tomar en cuenta para garantizar la continuidad del servicio en su totalidad y para todos sus usuarios. Estos factores se pueden enumerar en orden de importancia como sigue:

1. Funcionamiento correcto del servicio LDAP
2. El conjunto de componentes de Qmail (*qmail-send, qmail-imapd, qmail-smtp, qmail-queue*), en todos los equipos estén arriba y funcionando correctamente
3. Estado correcto de la cola de Qmail en todos los equipos

Seguridad del servicio

Hoy en día el correo electrónico es una de las herramientas más utilizadas por estudiantes y profesores para el intercambio de información. Pero actualmente el servicio de correo es usado de forma masiva en un ambiente agresivo y de enorme desconfianza, donde muchos desarrolladores y saboteadores de Internet han hallado diversas formas de mal utilizarlo. Con el objetivo de mejorar el servicio de correo electrónico se deben de considerar las bases para establecer una plataforma, que por medio de herramientas, logre brindar el servicio de forma segura.

Para esto es necesario contar con un antivirus y un antispam que proporcione la protección adecuada y que de la posibilidad de integrarse al sistema de correo.

Como ya se ha comentado en este documento, el antivirus y antispam que el sistema utiliza es perlscanner, integrado al sistema por medio de *qmail-scanner*, sin el uso de ningún explorador externo.

Para garantizar la eficiencia de los sistemas de filtrado de mensajes, se deben de revisar los siguientes puntos.

1. Verificar que el sistema no permite el *relay*, es decir, no permite que el sistema sea utilizado como relevo para el envío de correo *spam*.
2. Evitar en medida de lo posible la entrada de correo *spam*.
3. Verificar que el sistema detiene los mensajes con virus.

Consistencia de la información

Para garantizar la consistencia de la información de los usuarios, se deben de establecer criterios para determinar que la información que un usuario está recibiendo o modificando en la aplicación, sea actualizada y modificada correctamente en los distintos componentes del *cluster*. Además se debe de garantizar que la información que el usuario modificó, permanezca así hasta que el usuario haga otras modificaciones.

Durante la puesta en marcha del sistema y debido a ciertas fallas que presentaba el directorio LDAP, ocurría que el *cluster* dejaba de funcionar, para ponerlo de nuevo en marcha se tenía que restablecer el directorio LDAP de un respaldo y esto ocasionaba que la consistencia de las contraseñas de los usuarios a veces se comprometía.

Se dedujo, después de observar el comportamiento del *cluster*, que para garantizar la consistencia de los datos de los usuarios, se tenían que revisar los siguientes aspectos:

1. Vigilar que el servicio de directorios LDAP esté funcionando correctamente.
2. Vigilar que la base de datos de los contactos de los usuarios está disponible y funcionando correctamente.
3. Vigilar que todos los equipos del *cluster* están almacenando y transfiriendo mensajes de correo correctamente.

Monitoreo del sistema

Este es un aspecto fundamental para garantizar el buen funcionamiento de todos los componentes. Para esto se deben definir la forma y los componentes que deben de ser verificados para coadyuvar al buen funcionamiento del sistema.

En este caso se encontró que los siguientes aspectos resultaban críticos para brindar el servicio y por lo tanto merecen ser monitoreados

1. Estado del servicio LDAP
2. Estado del servicio Qmail en todos los equipos
3. Cantidad de correos en la cola Qmail de todos los equipos
4. Estado de dispositivos físicos tal como interfaces de red, procesadores, memoria, etc.

3.3. Problemática

Dada la experiencia adquirida en la puesta en marcha del proyecto y después de un análisis de los aspectos que determinaban la eficiencia del sistema, se lograron identificar una serie de problemáticas.

A continuación se presentará una lista de los componentes del sistema que presentaron problemas y además se intentará explicar cuál era la causa y el impacto que tenían sobre el servicio.

3.3.1. Problemática con el directorio LDAP

Descripción

LDAP es un servicio consistente en una base de datos, en este caso Berkeley DB, y contiene información útil para intercambiar mensajes de correo entre los equipos del *cluster* y para autenticar usuarios.

A veces esta base de datos puede responder muy lento y en casos extremos corromperse, al dañarse los archivos que contienen los índices de la base de datos. Esto ocurre cuando se encuentra con muchas actualizaciones y consultas en periodos cortos de tiempo.

Las consecuencias de esto resultaban catastróficas para el servicio, porque cargaba demasiado a los procesos LDAP, intentando consultar o actualizar información en una base de datos corrupta. Como consecuencia de esto el servicio se tardaba mucho en responder, y paulatinamente se detenían los procesos LDAP. Esto causaba que el *cluster* simplemente dejara de funcionar interrumpiendo el servicio en su totalidad.

Para lidiar con esto se implementó una solución temporal, la cual se basó en tener a la mano un respaldo reciente de la base de datos, con la cual cada vez que se paraba el servicio por la razón expuesta, se restablecía la base de datos permitiendo levantar de nuevo el servicio.

El problema con esto fue que se perdían los cambios de contraseña que se actualizaban entre el último respaldo y el momento del daño.

Además de los problemas descritos se necesitaba contar con un sistema de monitoreo para conocer el estado del servicio y realizar las acciones pertinentes en caso de contingencia.

Aspectos afectados por la problemática LDAP

- Continuidad del servicio
- Rapidez del servicio
- Consistencia de la información
- Monitoreo

3.3.2. Problemática con el esquema LDAP

Como ya se explicó en la sección 1.3.4, Terminología LDAP, un registro del directorio bajo el esquema original OpenLDAP, soporta los siguientes campos:

```
dn: uid=test,dc=escolar,dc=unam,dc=mx
cn: test
sn: test
givenName: test
registeredAddress: 0
telephoneNumber: 00000000
description: 00
postalCode: 00000
title: 0
objectClass: top
objectClass: person
objectClass: organizationalPerson
objectClass: inetOrgPerson
objectClass: qmailuser
mail: test@escolar.unam.mx
mailMessageStore: /var/qmail/maildirs/test
mailQuotaSize: 20000000
mailHost: bernoulli.dgae.unam.mx
uid: test
userPassword:: cG93ZXI=
```

El problema es que este esquema no contiene campos para almacenar datos útiles para la administración de cuentas, la asignación de contraseñas y la forma de cambiar las mismas en el contexto de la Administración Escolar.

El problema es, entonces, buscar la forma de extender el esquema LDAP para que soporte otros campos, por ejemplo:

- fecha de nacimiento del alumno
- lugar de nacimiento del alumno
- plantel del alumno

- carrera del alumno
- pregunta secreta para recuperar contraseña
- respuesta secreta para recuperar contraseña
- bandera para indicar si el alumno ya cambió su contraseña

Con datos como estos se tendrían elementos para saber si el usuario entra por primera vez a su cuenta. Si esto es verdadero, entonces se le obligaría al usuario a identificarse y a definir una contraseña. Otra cosa que se podría hacer es que en caso de que el usuario olvide su contraseña, exista la forma de autenticarlo para permitirle definir una nueva contraseña.

Otra aplicación de esto sería corroborar, con los datos plantel y carrera, que el buzón del alumno se encuentra en el equipo correcto.

3.3.3. Problemática con el servicio Web

Descripción

En el programa de trabajo se consideró incorporar paulatinamente a los alumnos de la Universidad que no cuenten con el servicio de correo.

El estimado fue que para el 2006 se tenía que llegar a un total aproximado de 300,000 cuentas de correo, lo que implicaba una carga fuerte sobre el servidor Web [1].

Uno de los principales problemas de un sitio Web en Internet es cómo gestionar las solicitudes de una gran cantidad de usuarios. Se trata de un problema de escalabilidad que surge con el continuo crecimiento de la cantidad de usuarios.

Aspectos afectados con la problemática del servicio Web

- Rapidez del servicio
- Continuidad del servicio

[1] Como se explica en el capítulo 5, Administración del Sistema, en la sección 6, Estadísticas de uso, se lleva la cuenta de la cantidad de conexiones que se hacen, por lo que se puede determinar, por lapsos, la carga de un servidor Web en relación a la cantidad de conexiones.

3.3.4. Problemática con el manejo de colas de Qmail

Descripción

Como ya se ha indicado, Qmail está compuesto de diversos subprogramas que realizan tareas específicas y que en conjunto constituyen el sistema de correo. Dentro de estos subprogramas se encuentra *qmail-queue* el cual se encarga de apilar correos en una complicada estructura de mensajes, en el directorio */var/qmail/queue*, para procesarlos y luego pasarlos a otros subprogramas Qmail para su entrega.

A veces este proceso se ve alterado, y súbitamente la estructura de mensajes se corrompe, las causas pueden ser desde un archivo corrupto en el sistema de archivos hasta permisos incorrectos[2]. Esto causa que la cola pierda la capacidad de desalojo de mensajes para su entrega, y por el contrario se dedica sólo a aceptar mensajes, por lo que la cola comienza a crecer.

El problema es entonces, dependiendo del equipo donde haya ocurrido el error, que se dejen de entregar y enviar correos de todos los usuarios o sólo de algunos de ellos.

Esto es si el error se produce donde se están recibiendo todos los correos (equipos Web) entonces todas las entregas y envíos quedan afectadas.

Por otro lado si el error se produce en uno de los equipos donde se encuentran almacenados ciertos usuarios, entonces sólo las entregas de estos usuarios quedan afectados.

Aspectos afectados a la problemática de las colas de Qmail

- Rapidez del servicio
- Continuidad del servicio
- Monitoreo del sistema
- Consistencia

[2]Se puede sospechar también, que al tratar, dos o más procesos (Qmail) modificar la misma estructura de mensajes, la corrompan. Pero aun no tenemos elementos para asegurar esto.

3.3.5. Problemática con el antivirus y antispam

Descripción

Para filtrar y explorar mensajes de correo, Qmail cuenta con el subprograma *qmail-scanner*, cuyo explorador de mensajes interno es *perlscanner*. La finalidad de *qmail-scanner* es permitir el bloqueo de mensajes de correo con ciertas características, tales como asuntos asociados a correo *spam*, o archivos adjuntos con ciertos nombres o tamaños; además hacer una exploración muy básica de virus en los archivos.

qmail-scanner es muy genérico, lo cual significa que puede filtrar vía expresiones fijas o vía expresiones regulares, lo cual implica que puede rechazar todos los mensajes que contengan el asunto "Viaja gratis a todo el mundo", o rechazar todos los mensajes cuyos archivos adjuntos contengan la extensión .mp3 o excedan algún tamaño.

El problema es que *qmail-scanner* no tiene un mecanismo para agregar reglas de filtrado por sí mismo. La forma de agregar reglas es editando un archivo de configuración y luego por medio de un comando actualizar la base de datos de filtrado para reflejar los nuevos cambios.

Administrativamente hablando es muy complicado mantener las reglas de filtrado de esta manera, dado que cada día aparecen nuevos virus y mensajes *spam*.

Lo que se necesita es un sistema que agregue las reglas automáticamente, ya sea vía Internet o vía algoritmos especializados para esto, con el objetivo de reducir la intervención de los administradores al mínimo.

Aspectos afectados con la problemática del antivirus y antispam

- Seguridad del servicio

3.3.6. Problemática con los respaldos

Descripción

Los respaldos consisten en guardar la información de un sistema en un medio extraíble, en este caso cintas magnéticas, para tener capacidad de recuperar los datos ante posibles pérdidas.

Para esto se pensó en dos tipos de respaldos, uno para respaldar toda la información de los discos duros, y otro sólo para respaldar información específica concerniente al servicio de correo electrónico, en este caso, la base de datos LDAP, los buzones de los usuarios, los contactos de los usuarios, las bitácoras del servicio de correo y Web, etc.

Dado lo anterior surge una pregunta, ¿por qué aparte de respaldar toda la información del *cluster*, también se respalda redundantemente información específica del servicio de correo?.

La respuesta es simple. En caso de que un equipo falle lo más probable es que se necesite restaurar en un equipo predispuesto para esto. Pero, no se trata de restaurar toda la información, sino sólo la necesaria para que el equipo de reemplazo, que ya cuenta con todos los paquetes instalados y configurados para entrar al *cluster*, sustituya al equipo en cuestión.

Esto se hace para reducir considerablemente el tiempo de respuesta a incidentes, dado que sólo es necesario restaurar información específica. Ejemplos de esto pueden ser, la base de datos LDAP en caso de falla del servidor LDAP, los buzones de correo en caso de que fallen los equipos de almacenamiento.

Por otra parte, se decidió que los respaldos se harían semanales y que se almacenarían en cuatro juegos de cintas. Para esto se considera una rotación mensual, siguiendo una secuencia circular. Esto nos da la posibilidad de restaurar información de hasta un mes de antigüedad.

Hasta aquí el planteamiento para hacer respaldos resultaba buena, pero el problema fue cuando esto se puso en marcha, dado que los respaldos se realizaban de forma manual y sólo se contaba con una unidad de cinta.

Debido a esto, el proceso requería de una exhaustiva supervisión de los administradores para hacer los respaldos, uno después de otro, lo que requería de mucho tiempo y resultaba un trabajo muy engorroso.

Esto creaba una nueva necesidad, la planificación de forma automática de los respaldos. Generalmente esto no es absolutamente posible, pero lo que si se puede lograr es simplificar sustancialmente el proceso.

Aspectos afectados a la problemática de los respaldos

- Consistencia de la información
- Continuidad del servicio

3.3.7. Problemática con el monitoreo del sistema

Descripción

La manera de monitorear el *cluster* es simplemente utilizando los comandos que proporciona el Sistema Operativo de los equipos (ver tabla 3.1).

Comando	Función
top	Ver actividad del CPU y memoria, en tiempo real
df	Reporte de espacio en disco utilizado por el sistema de archivos
netstat	Reporte de conexiones, actividad de interfaces de red, etc.
free	Memoria en uso

Cuadro 3.1: Comandos del sistema operativo para monitoreo.

Sin embargo, a pesar de que estos comandos producen reportes detallados y exactos de lo que ocurre en cada uno de los equipos, la verdad es que en términos prácticos no resulta conveniente monitorear el *cluster* así, puesto que todos los reportes que realizan son en tiempo real y no generan bitácora.

Para resolver ésto se necesita de *software* que tenga la capacidad de recolectar en un solo equipo toda la información necesaria de los demás equipos, almacenar la información en una base de datos y luego presentarla de forma clara para su lectura e interpretación.

Aspectos afectados por la problemática del monitoreo del sistema

- Continuidad del servicio

Capítulo 4

Líneas de Desarrollo

Este capítulo describe las líneas de desarrollo que derivaron en la solución de las problemáticas que se describieron en el capítulo anterior y que determinan una nueva etapa en el desarrollo del sistema.

4.1. Solución a la Problemática

Con base a la problemática presentada en el capítulo anterior, se describirán las líneas de desarrollo que les dieron solución.

Todas las aportaciones que se intentaron fueron estudiadas y analizadas por el equipo de administradores del sistema de correo y actualmente las que resultaron aplicables, se encuentran en producción.

Por las características del proyecto, donde el sistema de correo se basa en el *software libre qmail-ldap*, las alternativas de solución, tenían que proporcionar la característica de poder ser adaptables a éste.

4.1.1. Solución a la problemática del directorio LDAP

En ciertas configuraciones, un solo servidor slapd puede ser insuficiente para manejar la cantidad de clientes que requieren del servicio de directorio LDAP. En estos casos puede ser necesario correr más de un servidor slapd. Para esto se necesita configurar varios servidores slapd: uno maestro y uno o más esclavos. Este modelo de servidores maestro/esclavos provee una forma simple y efectiva de incrementar su capacidad y disponibilidad.

Para levantar una réplica del servidor slapd, se deben de seguir los siguientes pasos:

- Configurar el servidor slapd maestro
  - Agregar la directiva "*replica*" para cada réplica. El parámetro *binddn=* debe de coincidir con la opción *updatedn*, en el archivo de configuración del servidor slapd esclavo correspondiente.

    ```
    replica host=slave1.com.mx:389
           binddn="cn=usuario,dc=escolar,dc=unam,dc=mx"
           bindmethod=simple
           credentials=password
    ```

  - Agregar la directiva *replogfile*, la cual le indica a slapd dónde guardar los cambios. Este archivo será utilizado por *slurpd*.

    ```
    replogfile /var/log/slurpd.log
    ```

- Configurar los servidores *slapd* esclavos. Los archivos de configuración *slapd* esclavos tienen que ser idénticos al archivo de configuración *slapd* maestro, con las siguientes excepciones:
  - No incluir la directiva *replica*.
  - No incluir la directiva *replogfile*.
  - Incluir una directiva *updatedn*. El dn dado debe de coincidir en dn dado en la directiva *binddn=* del servidor maestro.
  - Incluir la directiva *updateref* para indicar el servidor desde el cual se harán las actualizaciones.

    ```
    updatedn "cn=usuario,dc=escolar,dc=unam,dc=mx"
    updateref ldap://master.com.mx
    ```

- Parar el servicio *slapd* en el servidor maestro con los comandos *ps* y *kill* (se debe tener un *script* que haga esto con el parámetro *stop*).
- Copiar la base de datos del servidor maestro al servidor esclavo. La base de datos se encuentra en el directorio especificado por la directiva *directory* del archivo de configuración *slapd*, en el servidor maestro.
- Reiniciar el servicio *slapd* maestro y arrancar el servicio *slapd* esclavo, con el comando:

  ```
  /usr/local/libexec/slapd
  ```

- Por último, arrancar el servicio *slurpd* en el sevidor maestro:

  ```
  /usr/local/libexec/slurpd
  ```

4.1.2. Solución a la problemática del esquema LDAP

Para poder extender el esquema utilizado por OpenLDAP para soportar tipos de atributos adicionales y clases de objetos, se pueden utilizar los archivos de esquema predefinidos por OpenLDAP como guía. Para hacerlo, se debe crear un archivo local.schema en el directorio */usr/local/etc/openldap/schema* que debe de contener lo siguiente:

```
# cat /usr/local/etc/openldap/schema/local.schema

# Attribute types are under 1.1.132.1
# Object classes are under 1.1.132.2

# Attribute Type Definitions

attributetype ( 1.1.132.1.1 NAME 'lugarNacimiento'
        DESC 'Luga de nacimiento en formato de dos numeros'
        EQUALITY numericStringMatch
        SYNTAX 1.3.6.1.4.1.1466.115.121.1.36{2} SINGLE-VALUE )

attributetype ( 1.1.132.1.2 NAME 'fechaNacimiento'
    DESC 'Fecha de nacimiento en formato DDMMAAAA'
        EQUALITY numericStringMatch
    SYNTAX 1.3.6.1.4.1.1466.115.121.1.36{8} SINGLE-VALUE )

attributetype ( 1.1.132.1.3 NAME 'genero'
        DESC 'Genero masculino (1), femenino (2)'
        EQUALITY numericStringMatch
        SYNTAX 1.3.6.1.4.1.1466.115.121.1.36{1} SINGLE-VALUE )

attributetype ( 1.1.132.1.4 NAME 'plantel'
        DESC 'Plantel del alumno'
        EQUALITY numericStringMatch
        SYNTAX 1.3.6.1.4.1.1466.115.121.1.36{3} SINGLE-VALUE )

attributetype ( 1.1.132.1.5 NAME 'carrera'
        DESC 'Carrera del alumno'
        EQUALITY numericStringMatch
        SYNTAX 1.3.6.1.4.1.1466.115.121.1.36{3} SINGLE-VALUE )

attributetype ( 1.1.132.1.6 NAME 'pregunta'
        DESC 'Pregunta, olvido su password'
        EQUALITY numericStringMatch
        SYNTAX 1.3.6.1.4.1.1466.115.121.1.36{2} SINGLE-VALUE )

attributetype ( 1.1.132.1.7 NAME 'respuesta'
        DESC 'Respuesta, olvido su password'
        SYNTAX 1.3.6.1.4.1.1466.115.121.1.15 SINGLE-VALUE )

attributetype ( 1.1.132.1.8 NAME 'cambioPass'
        DESC 'Indica si el usuario ya cambio su password'
        EQUALITY numericStringMatch
        SYNTAX 1.3.6.1.4.1.1466.115.121.1.36{1} SINGLE-VALUE )


# Object Class Definitions

objectclass ( 1.1.132.2.1 NAME 'local'
    DESC 'Clase local de alumnos'
        SUP top
        AUXILIARY
    MAY ( lugarNacimiento $ fechaNacimiento $ genero $ plantel $ carrera \
          $ pregunta $ respuesta $ cambioPass ) )
```

De esta manera se crea un nuevo *objectClass* junto con sus nuevos atributos, dando la siguiente estructura:

- objectClass: local
  - attribute: fechaNacimiento
  - attribute: lugarNacimiento
  - attribute: plantel
  - attribute: carrera
  - attribute: cambioPass
  - attribute: pregunta
  - attribute: respuesta

Lo que sigue es referenciar este nuevo esquema en el archivo de configuración *slapd.conf*, añadiendo la siguiente línea al final de las directivas include:

```
include /usr/local/etc/openldap/schema/local.schema
```

Si todo se hizo correctamente entonces todos los usuarios que utilicen este *objectClass = local* podrán incluir los nuevos atributos.

Para mayor información de cómo extender el esquema para reunir requisitos específicos, se tiene la página *http://www.openldap.org/doc/admin/schema.html*.

4.1.3. Solución a la problemática del servicio Web

Uno de los principales problemas de un sitio Web en Internet es como gestionar las solicitudes de una gran cantidad de usuarios. Se trata de un problema de escalabilidad que surge con el continuo crecimiento de la cantidad de usuarios del sistema.

Una solución para este tipo de problema es el balanceo de cargas. El balanceo de cargas es un concepto que se refiere a la técnica usada para compartir el trabajo a realizar entre varios procesos, equipos, discos u otros recursos.

Para resolver el problema se probaron varios balanceadores de carga y se encontró que uno de los que se adaptaron mejor al sistema fue el paquete de *software* Pen.

Pen es un balanceador de cargas para protocolos TCP simples tales como HTTP, HTTPS, SMTP, etc. Permite a muchos servidores aparecer

como uno solo al exterior y automáticamente detecta servidores que están fuera de servicio para distribuir las solicitudes entre los servidores que están disponibles. Esto da alta disponibilidad y desempeño escalable. Para más información ver la referencia [27].

Por lo anterior se decidió agregar al sistema un equipo para balanceo de carga, además de otro equipo más para el servicio web[1] . De esta manera se pudo balancear la carga de los servicios http y SMTP hacia los dos servidores Web que quedaron.

4.1.4. Solución a la problemática del manejo de colas de Qmail

La problemática con el manejo de la cola Qmail era que la cola se corrompía cuando recibía demasiado correo, lo que causaba que la cola comenzara a crecer, deteniendo la entrega de correo.

La solución consistió en conseguir la herramienta qmqtool descrita en el primer capítulo, y con el parámetro -c "check" comprobar el estado de la cola, que en caso de estar corrupta, entonces la herramienta cuenta con el parámetro -r "repair" para reparar la cola.

Esta solución resultó efectiva sin ningún problema, además de proporcionar varias funcionalidades para el manejo de las colas Qmail.

4.1.5. Solución a la problemática con el antivirus y antispam

En el caso del antivirus y antispam en la página oficial de Qmail *http://www.lifewithqmail.org/lwq.html*, se sugiere utilizar los paquetes clamAV y SpamAssassin, descritos en el capítulo primero, como exploradores externos de *qmail-scanner* para detectar virus y spam en los mensajes de correo.

Esta solución permitió que estos paquetes ya instalados y configurados, agreguen reglas de filtrado de forma automática, actualizando las bases de datos de virus y spam desde Internet o con otros algoritmos sofisticados para esto. Además estas herramientas cuentan con archivos de configuración para agregar reglas manualmente.

[1] En la sección 4.3, se describen estos y otros equipos que se agregaron al sistema.

4.1.6. Solución a la problemática con los respaldos

En el caso de los respaldos, la solución para pasar la información de todo el *cluster* a cintas magnéticas fue implementar con *script's* un procedimiento donde cada equipo produce con el comando *tar*, un archivo *tgz* con su información. Los archivos *tar* permiten restaurar directorios o archivos (ver manual de *tar*), por lo que una restauración puede hacerse por archivo, mensaje(archivo) o usuario(directorio). Estos archivos son enviados al momento de ser generados a un equipo predeterminado para esto. Cada equipo genera su archivo de respaldo a distintas horas, para evitar congestionamientos en la red.

El procedimiento es realizado un día a la semana, en la madrugada, con la intención de que durante el día se pase la información a cintas magnéticas.

El procedimiento para pasar los archivos de respaldo a cintas, es un *script* que incluye una verificación rápida de que se encuentren todos los respaldos y que estén bien formados.

Por lo tanto, es un solo procedimiento el que se requiere para pasar los respaldos a las cintas, el cual era el objetivo.

4.1.7. Solución a la problemática del monitoreo del sistema

Para monitorear el *cluster* se tuvo que realizar la integración entre los paquetes SNMP, RRDTools y Cacti los cuales ya fueron explicados en el primer capítulo.

El esquema bajo el cuál funcionan es el siguiente:

1. SNMP proporciona el soporte para la recolección de datos específicos de los aspectos que se necesitan monitorear
2. RRDTool proporciona el soporte para almacenar los datos recolectados por SNMP en bases de datos Round Robin.
3. Cacti es una interfaz final o front-end para RRDTool, la cual tiene capacidad de configurarse para poblar la base de datos circular, generar las gráficas y luego mostrarlas en un navegador.

De esta manera sólo hay que configurar correctamente el servicio SNMP en cada equipo y luego desde la interfaz Cacti agregar las gráficas correspondientes de lo que se quiera monitorear.

4.2. Características del sistema reconfigurado

Dados los cambios que se proponen como solución a los problemas funcionales, técnicos y operativos del sistema, se requiere de una reconfiguración del sistema que cambia las características de este. A continuación se presentan las nuevas características del sistema de correo.

- Acceso ininterrumpido al correo desde cualquier lugar por medio de un navegador, a través de la interfaz Horde.
- Interfaz con soporte para los idiomas Español e Inglés[2].
- Buzones maildir, es decir, cada mensaje se almacena como un archivo del sistema[3].
- Contactos personales (Horde con Turba).
- Acceso de clientes IMAP (Courier-IMAP).
- Qmail como MTA (Agente de Transferencia de Correo).
- Protección anti-virus y anti-spam proporcionada por *qmail-scanner* con el uso de los exploradores externos ClamAV y SpamAssassin en los dos servidores que manejarán la entrada de correo (ver figura 4.6).
- Usuarios y dominios virtuales en una base de datos LDAP.
- Soporte para clustereo nativo en los servidores de correo.
- Soporta cuotas de espacio en los buzones de los usuarios.
- Soporte para SHA[4], SSHA[5], MD5[6], SMD5[7], MD4[8] y RIPE-MD160[9].
- Dos servidores Web.
- Ocho equipos para almacenar correos.
- Dos equipos más para respaldos y contingencias.
- Un equipo para balancear todo el tráfico de entrada al *cluster*.
- Un servidor LDAP con dos réplicas del servicio para consultas al directorio.
- Un equipo para envío masivo de correos a los alumnos.

[2]También se puede habilitar el soporte para otros idiomas, entre los cuales están el Italiano, Alemán y Francés

[3]A diferencia de los buzones mbox, donde los mensajes se almacenan en un solo archivo.

[4]SHA (Secure Hash Algorithm, Algoritmo de Hash Seguro.)

[5]SSHA (Slated Secured Hashing Algorithm , Algoritmo de Hash Seguro Mejorado).

[6]MD5 (Message-Digest Algorithm 5, Algoritmo de Resumen del Mensaje 5).

[7]SMD5 (Message-Digest Algorithm 5, Algoritmo de Resumen del Mensaje 5 Mejorado).

[8]MD4 Anterior al MD5.

[9]RIPE-MD160 (RACE Integrity Primitives Evaluation Message Digest, primitivas de integridad del resumen del mensaje).

4.3. Nuevo Hardware

Para afrontar el nuevo reto que implica, el aumento masivo de cuentas de correo y la reestructuración del *cluster*, se decidió aumentar la infraestructura de *hardware* en seis equipos.

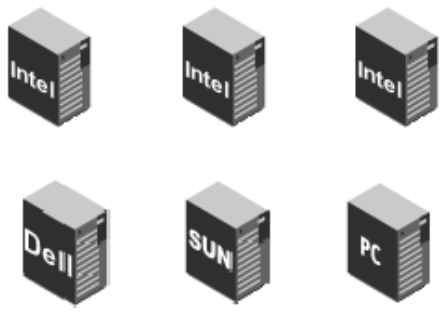


Figura 4.1: Equipo que se integrará al sistema de correo.

Para esto se contemplaron tres equipos Intel y un equipo Dell para almacenar correos y respaldos, un equipo Sun más para el servicio Web y una PC para enviar correo masivo a los alumnos. Sus características son:

Icono	Características	Cantidad	Función
	Pentium Xeon a 3.06 GHz duales, 2 GB de memoria, 2 discos de 73 GB y fuente redundante. (rack)	3	Servidores de correo
	Dell PowerEdge 2850 Pentium Xeon a 2.8 GHz, 1 GB de memoria y 1 disco SCSI de 73 GB	1	Servidor de Correo
	Sun Fire V240, 2 procesadores UltraSPARC[tm] IIIi a 1.28GHz, 2 GB de memoria, 2 discos de 80 GB y fuente redundante	1	Servicio Web
	PC armada, procesador Intel Pentium 4 a 1.28GHz, 2 GB de memoria y 1 disco de 80 GB	1	Envío masivo de correo

Cuadro 4.1: Características del equipo que se integró al sistema de correo.

4.4. La reestructuración del cluster

De acuerdo al cuadro 4.2, se puede observar que el *cluster*, después de la reestructuración, está constituido por quince equipos[10], los cuales tienen una funcionalidad específica dentro del *cluster*.

Dentro de estos quince equipos se tiene un equipo dedicado a balancear cargas de los servicios Web y SMTP hacia dos equipos.

Por otra parte, ahora el servidor LDAP tiene dos réplicas, con la intención de que las consultas al directorio LDAP se dividan en dos servidores y las actualizaciones se realicen directamente en el servidor LDAP. Bajo este esquema ahora el *cluster* funciona con el soporte de tres servidores LDAP, dos para consultas y uno para actualizaciones.

Ahora los servicios Web y LDAP, críticos para el funcionamiento del sistema, están distribuidos en cinco equipos.

Además ahora son ocho equipos los que almacenan correo, con el propósito de hospedar las nuevas cuentas incorporadas al sistema.

Por último se tiene un equipo para almacenar semanalmente los respaldos de todo el *cluster* y otro más para responder a incidentes.

Núm. de equipo	Sistema operativo	Nombre del equipo	Función
1	Solaris 9	escolar	Balanceo de cargas
2	Solaris 9	escolar0	Servidor Web(HTTPS) y Qmail(SMTP)
3	Solaris 9	escolar1	Servidor Web(HTTPS) y Qmail(SMTP)
4	Redhat 9	barajas	Servidor LDAP
5	Redhat 9	boole	Servidor Qmail, almacena correo
6	Redhat 9	babbage	Servidor Qmail, almacena correo
7	Redhat 9	bernoulli	Servidor Qmail, almacena correo
8	Redhat 9	banach	Servidor Qmail, almacena correo
9	Redhat 9	bussey	Réplica LDAP, servidor Qmail, almacena correo
10	Redhat 9	bolzano	Réplica LDAP, servidor Qmail, almacena correo
11	Fedora 2	briggs	Servidor Qmail, almacena correo
12	Fedora 2	bacon	Servidor Qmail, almacena correo
13	Fedora 2	blake	Respaldos
14	Fedora 2	bomberg	Respuesta rápida a incidentes
15	Redhat 9	bayes	Correo masivo

Cuadro 4.2: Equipos que conforman el *cluster* después de su reestructuración.

[10]Los nuevos equipos tienen un S.O. (Fedora 2) distinto al que se había venido utilizando (Redhat 9), por cuestiones de compatibilidad de *hardware*, aunque se tiene pensado cambiarlo por un S.O. más seguro y estable (no precisado).

4.5. Distribución y Comunicación

La reestructuración del *cluster* ocasionó cambios en la forma en que se realiza la comunicación de componentes de *software* en el sistema.

Por esta razón esta sección presentará el nuevo esquema bajo el cual funciona el *cluster*, además de mostrar la forma en que los equipos se comunican, al momento de autenticar un usuario, al momento de enviar o recibir correo, así como a la hora de recoger los mensajes de los buzones para mostrarlos en el navegador de un usuario. Esto mediante el uso de diagramas que contendrán líneas con números asociados a una descripción de lo que ocurre.

Para esto la figura 4.2 muestra el nuevo esquema, que se puede explicar como sigue:

1. Se tiene primero un equipo de balanceo de cargas que recibe todas la peticiones Web y SMTP.

2. Luego este equipo balancea cargas redireccionando las peticiones hacia cualquiera de los dos equipos Web.

3. Cada equipo Web tiene su propia réplica LDAP para realizar las consultas al directorio, para realizar el procesamiento de correo y autenticar usuarios.

4. Los dos equipos Web deben de realizar las actualizaciones, que en el caso del sistema sólo son los cambios de password, sobre el servidor LDAP maestro ya que en las réplicas no se pueden hacer actualizaciones.

5. Una vez que se ha realizado una actualización en el servidor maestro LDAP, este envía los cambios a las réplicas. De esta forma se mantienen sincronizados los equipos y se preserva la consistencia de la información en los tres servidores LDAP.

6. Los equipos que almacenan correo (equipos 5, 6, ..., 12) están divididos desde el punto de vista administrativo, para que la mitad de estos realice las consultas LDAP al servidor réplica 1, mientras los otros utilicen la réplica 2.

7. El equipo para envío masivo de correo electrónico hace las consultas LDAP a cualquiera de las réplicas, que para fines de ejemplo se presenta en la figura que lo hace a la réplica 1. Esto se puede ajustar para que la consulta se ralice hacia la réplica LDAP menos cargada.

8. Una vez que las peticiones han sido procesadas por el *cluster*, el servidor Web que maneja la sesión es el encargado de regresar la respuesta al usuario por medio del servidor de balanceo de cargas.

9. Finalmente el servidor de balanceo de cargas reenvía la respuesta al usuario.

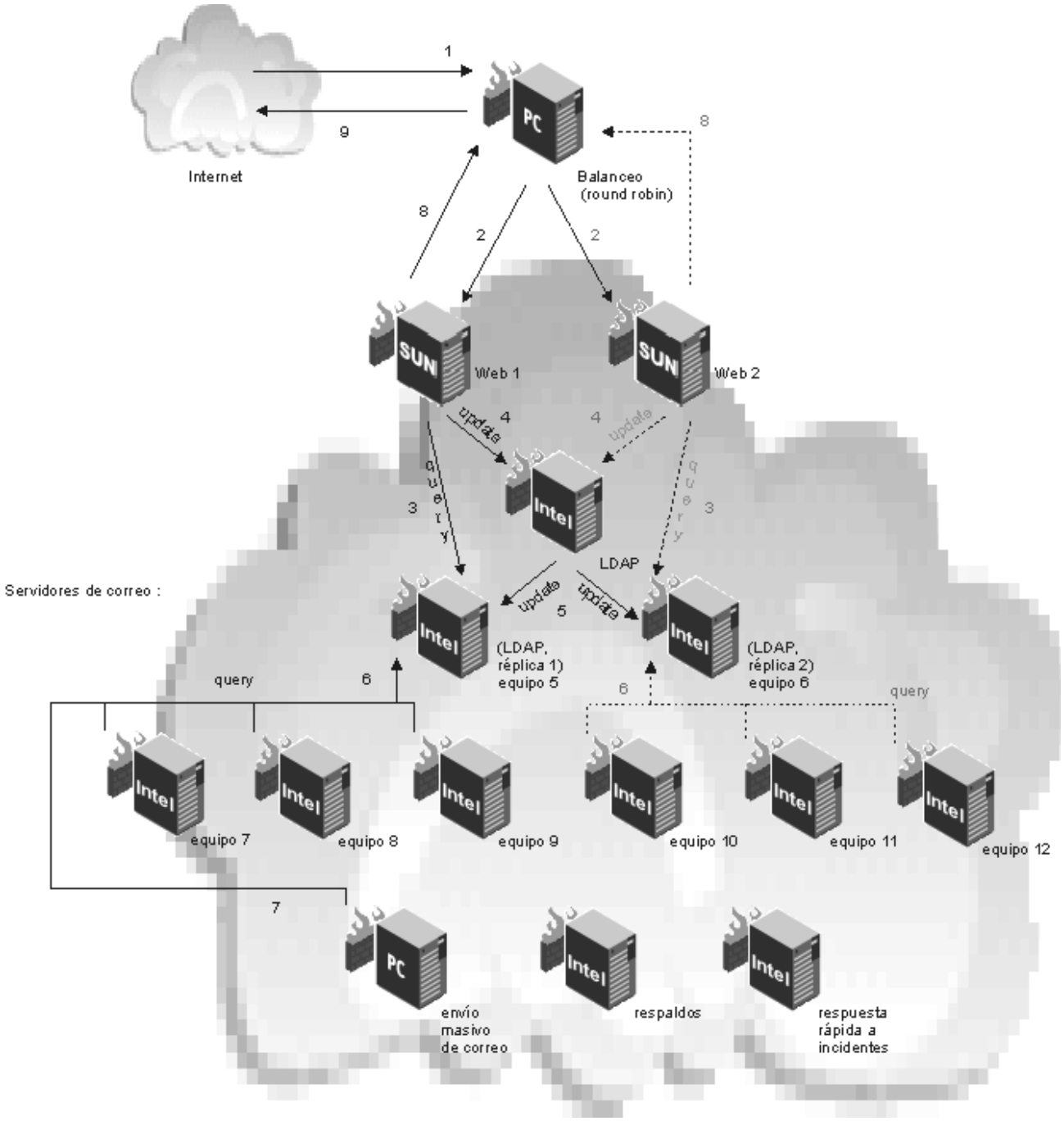


Figura 4.2: *Cluster* de correo en Internet.

Otras consideraciones importantes son:

- Cada equipo está protegido individualmente, mediante un *firewall de host*, no permitiendo conexiones no autorizadas.
- Los servicios Web y LDAP son críticos para el funcionamiento del *cluster*, por lo que se encuentran distribuidos por separado en varios equipos.
- Los equipos 5, 6, ..., 12 mostrados en la figura 4.2, contienen los buzones de los usuarios, en los directorios del sistema */var/qmail/maildirs/<usuario>*.
- Los equipos 5 y 6 de la figura 4.2 contienen buzones de usuarios, por lo que almacena correo y además son réplicas del servicio LDAP.
- Las líneas continuas y las líneas punteadas de las figuras que se presentan, representan las rutas alternativas de comunicación entre los equipos, dependiendo de hacia qué servidor Web fue redireccionada la petición Web o SMTP, o hacia dónde le corresponde a un equipo hacer las consultas LDAP.
- Existe un equipo para respuesta rápida a incidentes y uno más para respaldos, los cuales están dispuestos para sustituir a cualquier equipo en caso de falla.
- Bajo este esquema no se tiene un equipo sustituto para el servicio Web, ya que por las características que proporciona el servidor de balanceo de cargas, si falla un Web entonces todas las peticiones son redireccionadas al Web que esté disponible. De esta manera se logra tener un servicio continuo en caso de falla.
- En el caso de fallar un servidor LDAP, los equipos del *cluster* tienen capacidad para cambiar de servidor LDAP sin realizar ningún ajuste, lo que permite arreglar el problema sin interrumpir el servicio.

A continuación, en las siguientes secciones, se muestra por cada función cómo trabaja el *cluster*.

4.5.1. Autenticación de un usuario

Como ya se había comentado, el sistema utiliza el protocolo IMAP, para autenticar usuarios en el sistema, utilizando en específico la "retransmisión de sesión"[11], *auth_imap*, para comparar la información dada por un usuario, contra la información almacenada en el directorio LDAP, y así verificar la autenticidad del usuario.

El esquema bajo el cual funciona la autenticación de un usuario en el sistema, se muestra en el figura 4.3 y se puede explicar como sigue:

1. El usuario introduce el "nombre de usuario" y "contraseña" desde la página de inicio de la aplicación y luego envía la información cifrada al servidor Web *escolar.unam.mx* que en este caso es el servidor de balanceo de cargas.

2. El servidor de balanceo de cargas recibe la petición y mediante un algoritmo round robin, envía esta petición a uno de los dos servidores Web.

3. Una vez que cualquiera de los dos servidores Web recibe la información, dispone del programa *auth_imap* para enviar el "nombre de usuario" y "contraseña" al servidor LDAP que le corresponde (obsérvese que cada servidor Web tiene su propio servidor LDAP), para realizar la operación de búsqueda.

4. El programa *auth_imap* en el servidor LDAP correspondiente busca en el directorio la entrada que contiene el atributo "nombre de usuario". Cuando encuentra la entrada, el programa verifica que la contraseña proporcionada por el usuario coincida con el campo "contraseña" del directorio LDAP. Si coincide y si la cuenta está activa, entonces el programa recolecta los datos del directorio en cuestión, tal como, *uid, mailHost, mailAddress, maildir, quote*, etc. y se los envía al servidor Web.

5. El servidor Web en cuestión obtiene el campo *mailHost* (equipo k) del directorio correspondiente al usuario, para compararlo con el archivo de control Qmail, `~control/me`[12](el nombre del servidor Web). No coinciden, dado que el usuario no tiene su cuenta en este servidor, por lo que *auth_imap* retransmite la sesión al servidor indicado por el campo *mailHost* (equipo k), donde se encuentra el buzón del usuario indicado.

[11] El "redirecionamiento de sesión", *auth_imap*, permite recolectar mensajes de los clientes IMAP, cuando los mensajes están almacenados en otro miembro del *cluster*, en donde, *auth_imap* puede conectarse a otro equipo del *cluster* para obtener los mensajes y regresarlos al cliente IMAP.

[12] Archivo de control de Qmail que contiene el nombre del servidor

6. El programa *auth_imap* en el *equipo k* intenta autenticarse tal y como se hizo en los dos pasos anteriores, es decir, realiza la operación de búsqueda en la réplica LDAP correspondiente, recibe la información del directorio, obtiene el *mailHost* del usuario y lo compara con el archivo de control `~control/me` (equipo k), que en este caso deben coincidir.

7. Dado lo anterior el programa *qmail-imapd* establece una sesión IMAP, entre el *equipo k* y el cliente, retransmitiendo la sesión a través del servidor Web que inició la petición. Además recolecta los mensajes del buzón del usuario y los envía al servidor Web.

8. El servidor Web notifica al cliente por medio del balanceador de cargas que la sesión IMAP se ha establecido.

9. El cliente recibe la información del buzón solicitado y lo muestra en el navegador.

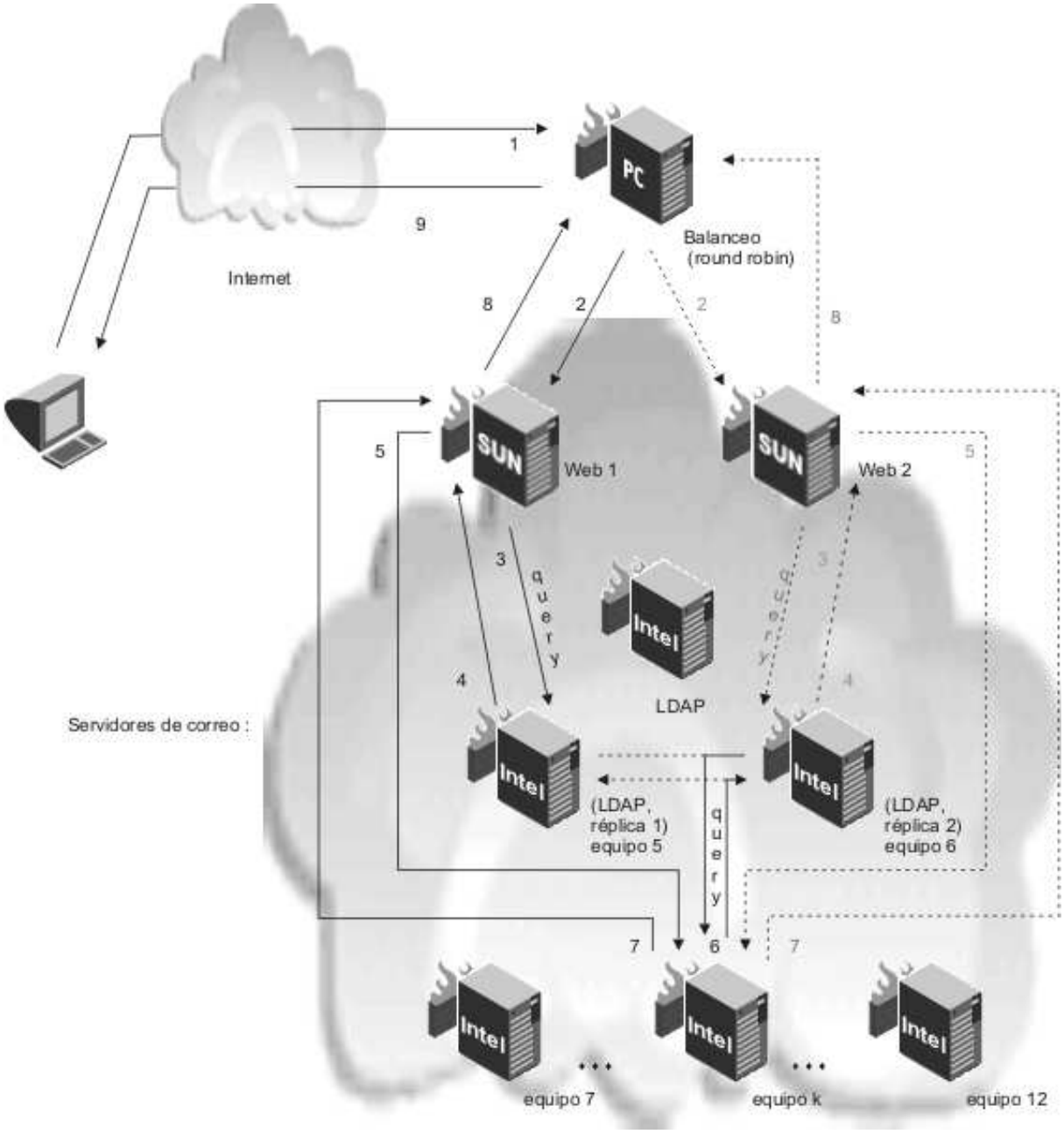


Figura 4.3: Funcionamiento del *cluster* al autenticar un usuario.

4.5.2. Envío y recepción de correo local

Esta sección expone cómo procesa mensajes de correo, el *cluster* reestructurado, dado un menaje que parte de un servidor del *cluster* y está dirigido a otro servidor dentro del *cluster*. Para esto se muestra la figura 4.4 que se puede explicar como sigue:

1. Desde la aplicación Web para enviar mensajes de correo, el usuario llena los campos, destinatario, asunto, etc. (partes del encabezado del correo), escribe la parte principal del mensaje y en caso de requerirlo, agrega los archivos adjuntos. Hecho esto, el usuario envía la petición al servidor de balanceo de cargas.

2. El servidor de balanceo de cargas redirecciona la petición a uno de los dos servidores Web.

3. El servidor Web recibe la petición, formatea el mensaje y se lo da a Qmail para que sea procesado por varios de sus subprogramas:

   - *qmail-inject* interpreta el mensaje e incorpora información extra al encabezado del mensaje para que pueda ser entregado.
   - *qmail-queue* lo pone en la cola de correos de salida de qmail;
   - *qmail-send* lo toma de la cola, lo identifica como correo local y lo pasa a *qmail-lspawn*
   - *qmail-lspawn*, lanza la operación de búsqueda LDAP enviando el "uid" del usuario al que va dirigido el mensaje.

4. Luego el servidor LDAP recibe la petición y busca en el directorio la entrada que contiene el atributo "uid" del destinatario del mensaje, para extraer los datos de su directorio (*mailHost, maildir*, etc.). Lo obtenido se lo manda al servidor Web.

5. El programa *qmail-lspawn* recibe la respuesta, extrae el *mailHost* del destinatario del mensaje y por medio de *qmail-qmqpc* lo envía al servidor con nombre *mailHost*, es decir al *equipo k*.

6. El *equipo k* recibe el mensaje y por medio de Qmail realiza un proceso similar a los puntos 3, 4 y 5:

   - *qmail-queue*, lo pone en la cola de correos de salida de Qmail;
   - *qmail-send* lo toma de la cola, lo identifica como correo local y lo pasa a *qmail-lspawn*
   - *qmail-lspawn* lanza la operación de búsqueda LDAP enviando el uid del usuario al que va dirigido el mensaje.
   - *qmail-lspawn* recibe el *mailHost* del destinatario, y se percata que el mensaje debe de ser canalizado a un buzón que está en el mismo servidor.

7. Dado lo anterior el mensaje es pasado a *qmail-local* para ser almacenado en el buzón de destino.

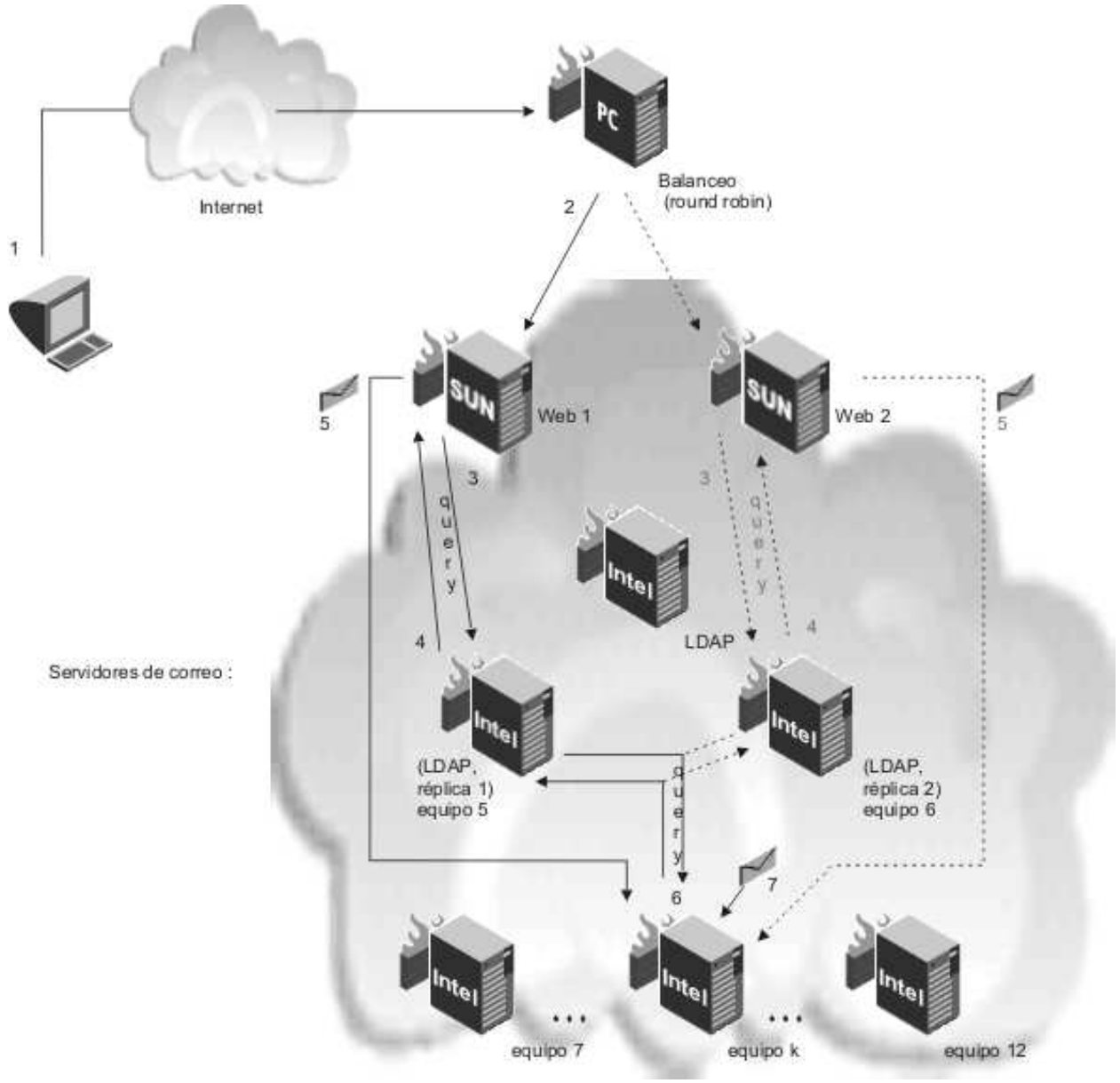


Figura 4.4: Funcionamiento del *cluster* al enviar y recibir correo local.

4.5.3. Envío de correo remoto

Para el envío de mensajes de correo remoto, como se puede apreciar en la figura 4.5 el funcionamiento del *cluster* es como sigue:

1. Desde la aplicación Web para enviar mensajes de correo, el usuario llena los campos: destinatario, asunto, etc. (partes del encabezado del correo), escribe la parte principal del mensaje y en caso de requerirlo, agrega los archivos adjuntos. Hecho esto, el usuario debe notificar al servidor que comience a procesar el correo dando la instrucción "enviar".

2. El servidor de balanceo de carga recibe la petición y la redirecciona a uno de los servidores Web.

3. El servidor Web recibe la petición, formatea el mensaje y se lo da a Qmail para que sea procesado por varios de sus subprogramas:

   - *qmail-inject* interpreta el mensaje e incorpora información extra al encabezado del mensaje para que pueda ser entregado;
   - *qmail-queue* pone el mensaje en la cola de correos de salida de Qmail;
   - *qmail-send* toma el mensaje de la cola de correos de salida, lo identifica como correo remoto y lo pasa a *qmail-rspawn*;
   - *qmail-rspawn* planifica la entrega y transfiere el mensaje a *qmail-remote*.
   - Finalmente Qmail por medio de *qmail-remote* envía el mensaje al servidor de correo remoto.

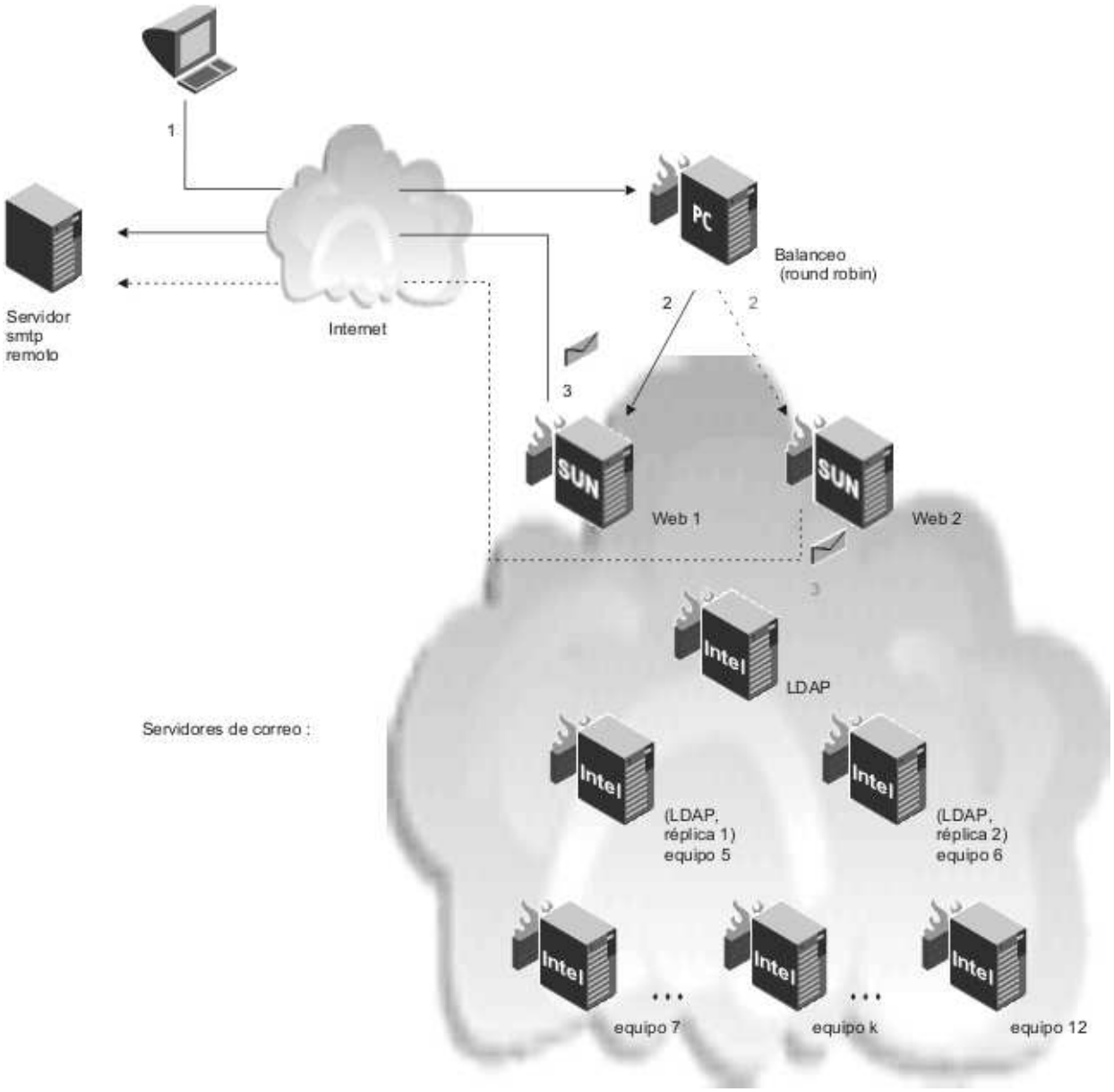


Figura 4.5: Funcionamiento del *cluster* al enviar correo.

4.5.4. Recepción de correo remoto

1. El servidor de balanceo de cargas recibe la petición SMTP, luego conforme a un algoritmo round robin redirecciona la petición a uno de los dos servidores Web, para que sea procesado el mensaje.

2. El programa *qmail-smtpd* en alguno de los servidores Web, recibe el mensaje de correo vía SMTP.

3. Después Qmail en el mismo equipo, comienza a procesar el mensaje de la siguiente manera:

   - *qmail-smtpd* transfiere el mensaje a *qmail-queue*;
   - *qmail-queue* pone el mensaje en la cola de correos de salida de Qmail;
   - *qmail-send* toma el mensaje de la cola, lo identifica como correo local y lo pasa a *qmail-lspawn*;
   - *qmail-lspawn* lanza la operación de búsqueda LDAP a la réplica que le corresponde, enviando el uid del usuario al que va dirigido el mensaje.

4. Luego el servidor LDAP recibe la petición y busca en el directorio la entrada que contiene el atributo "uid" del destinatario del mensaje, para extraer los datos de su directorio (*mailHost, maildir*, etc.). Lo obtenido se lo manda al servidor Web en cuestión.

5. El programa *qmail-lspawn* en el equipo indicado recibe la petición, extrae el *mailHost* del destinatario del mensaje y por medio *qmail-qmqpc* lo envía al servidor con nombre *mailHost*, es decir al *equipo k*.

6. El *equipo k* recibe el mensaje y por medio de Qmail realiza un proceso similar a los puntos 3, 4 y 5:

   - *qmail-queue* lo pone en la cola de correos de salida de Qmail;
   - *qmail-send* lo toma de la cola, lo identifica como correo local y lo pasa a qmail-lspawn;
   - *qmail-lspawn* lanza la operación de búsqueda LDAP, a la réplica que le corresponde, enviando el uid del usuario al que va dirigido el mensaje.
   - *qmail-lspawn* recibe el *mailHost* del destinatario, y se percata que el mensaje debe de ser canalizado a un buzón que está en el mismo servidor.

7. Dado lo anterior el mensaje es pasado a *qmail-local* que lo almacena en el buzón de destino.

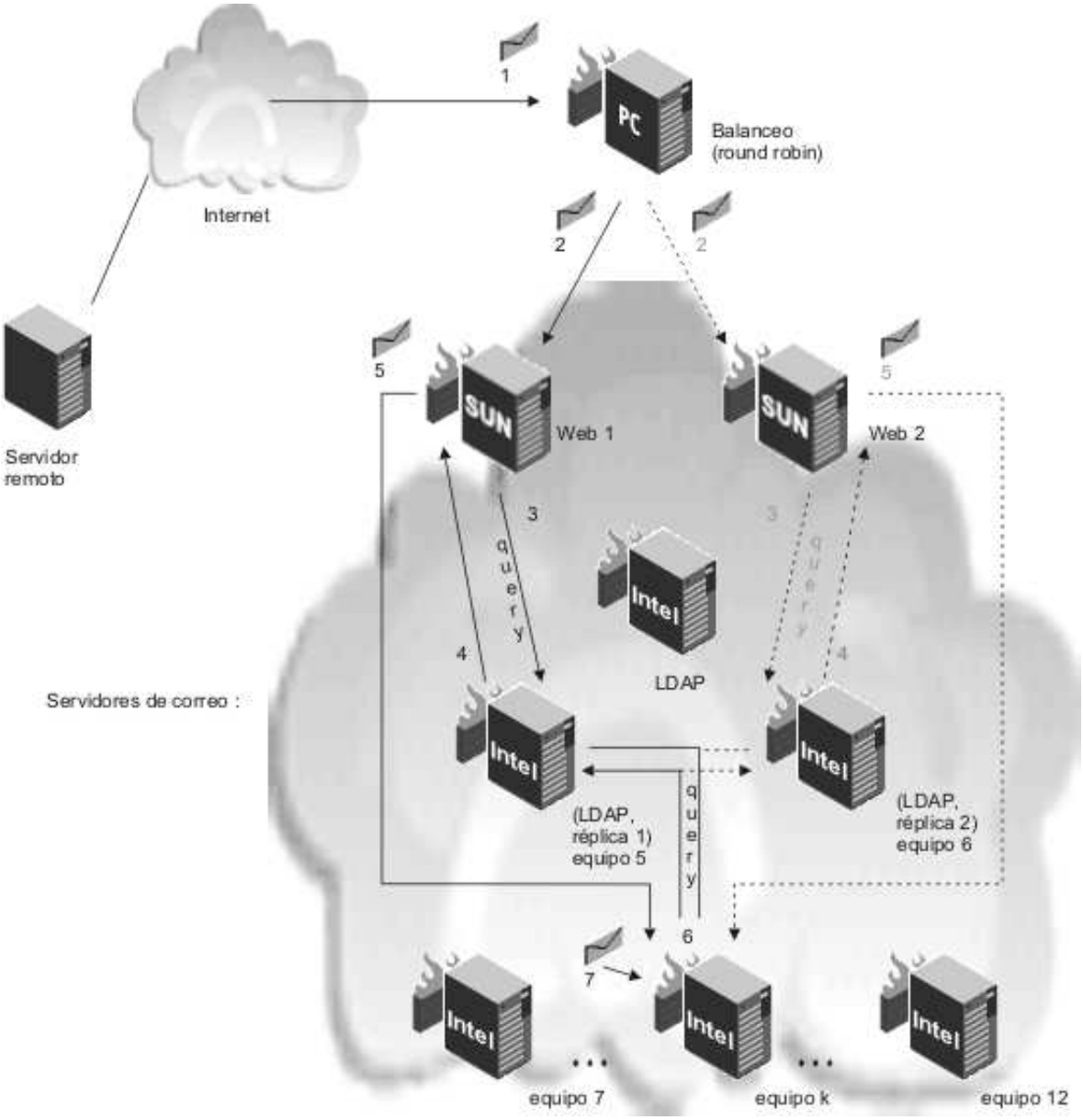


Figura 4.6: Funcionamiento del *cluster* al recibir correo.

4.5.5. Cambio de contraseña

El sistema proporciona dos formas de cambiar contraseña. Una de ellas, caso (a) se proporciona cuando el usuario está ya autenticado y desea desde el menú de la aplicación Web cambiarla. La otra forma, caso (b) se dispone en caso de que el usuario haya olvidado su contraseña, para lo cual desde la página de inicio se tiene el acceso "olvidaste tu contraseña", donde el usuario debe de proporcionar su número de cuenta y otros datos personales, para que el sistema le permita cambiar la contraseña.

Ahora bien, para mostrar cómo funciona el *cluster* para cambiar contraseñas se tiene la figura 4.7 y se realiza de la siguiente manera:

1. El usuario entra a cualquiera de las dos aplicaciones Web para cambiar contraseña, dependiendo de lo que necesite, posteriormente proporciona sus datos, en el caso (a) es antigüa contraseña y nueva contraseña, en el caso (b) es número de cuenta y datos personales. Se envían los datos al servidor de balanceo de cargas.

2. El servidor de balanceo de cargas redirecciona la petición al servidor Web que esté manejando la sesión del usuario.

3. El servidor Web en cuestión recibe la petición e inmediatamente después envía la petición a la réplica LDAP que le corresponde, para verificar que los datos que dio el usuario son correctos (algo así como una segunda autenticación).

4. El servidor LDAP busca la información, la compara con lo que tiene en su directorio y envía la confirmación en caso de que los datos sean correctos.

5. Si los datos son correctos, entonces, el servidor Web genera una contraseña o verifica la nueva contraseña que dio el usuario, dependiendo del tipo de cambio de contraseña que se esté llevando a cabo, para luego hacer la petición de actualización al servidor maestro LDAP.

6. El servidor LDAP hace la actualización de la contraseña en el directorio del usuario indicado y envía la misma información a las dos réplicas para sincronizar los directorios.

7. El servidor maestro LDAP, envía una notificación al servidor Web que le hizo la petición.

8. El servidor Web involucrado envía una notificación al usuario, por medio del balanceador de cargas, para confirmarle que su contraseña ha sido actualizada.

9. El usuario recibe la confirmación de cambio.

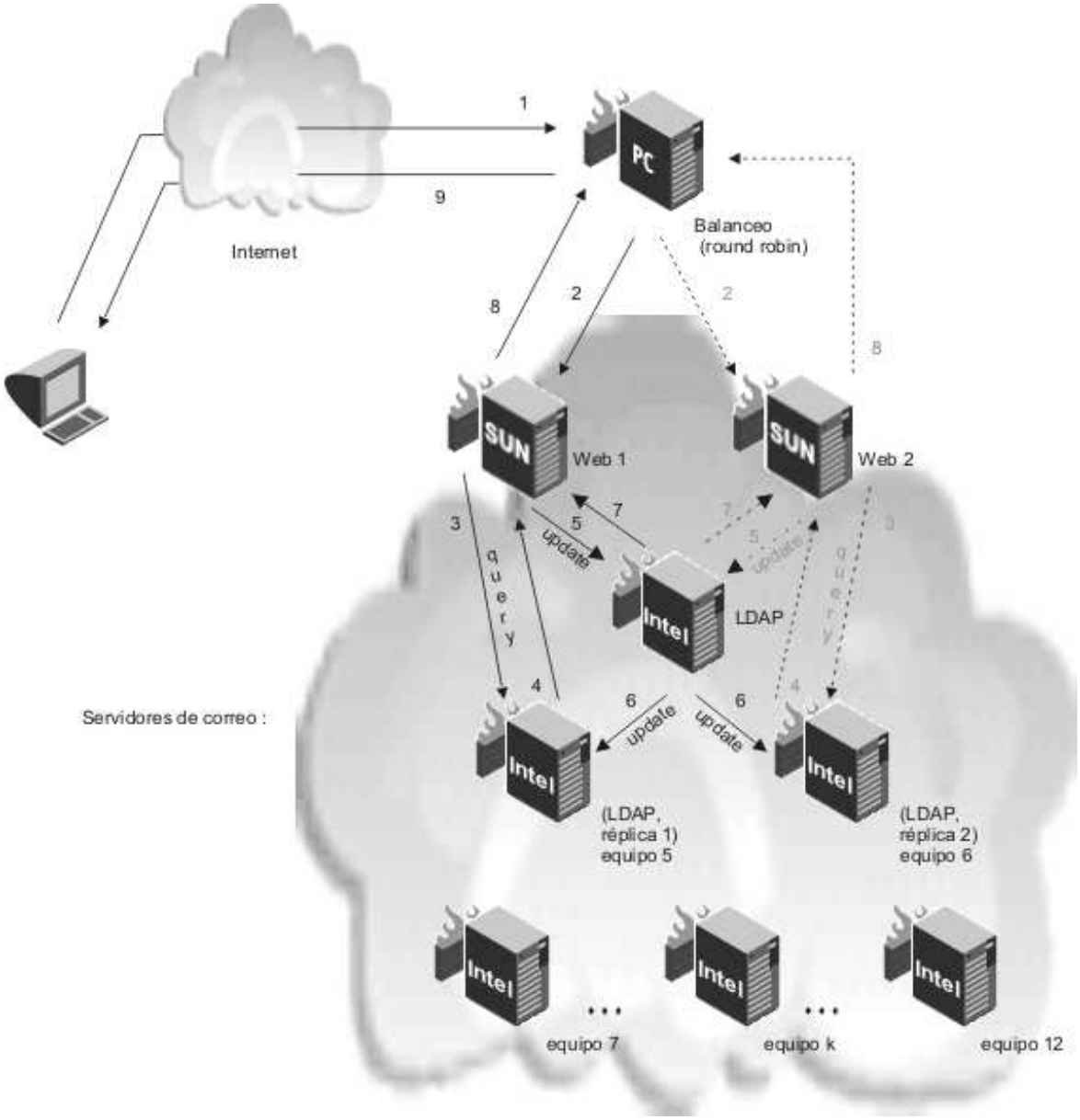


Figura 4.7: Funcionamiento del *cluster* al cambiar la contraseña de un usuario.

4.5.6. Diagrama de Distribución UML, del Sistema de Correo reestructurado

Como ya se había mencionado en la sección 1.6, UML proprociona un modelo para mostrar la relación entre las plataformas de *hardware* y los componentes de *software*, en el diagrama conocido como "Diagrama de Distribución".

Dado lo cuál, sólo queda presentar el Diagrama de Distribución del sistema de correo reestructurado (ver figura 4.8)

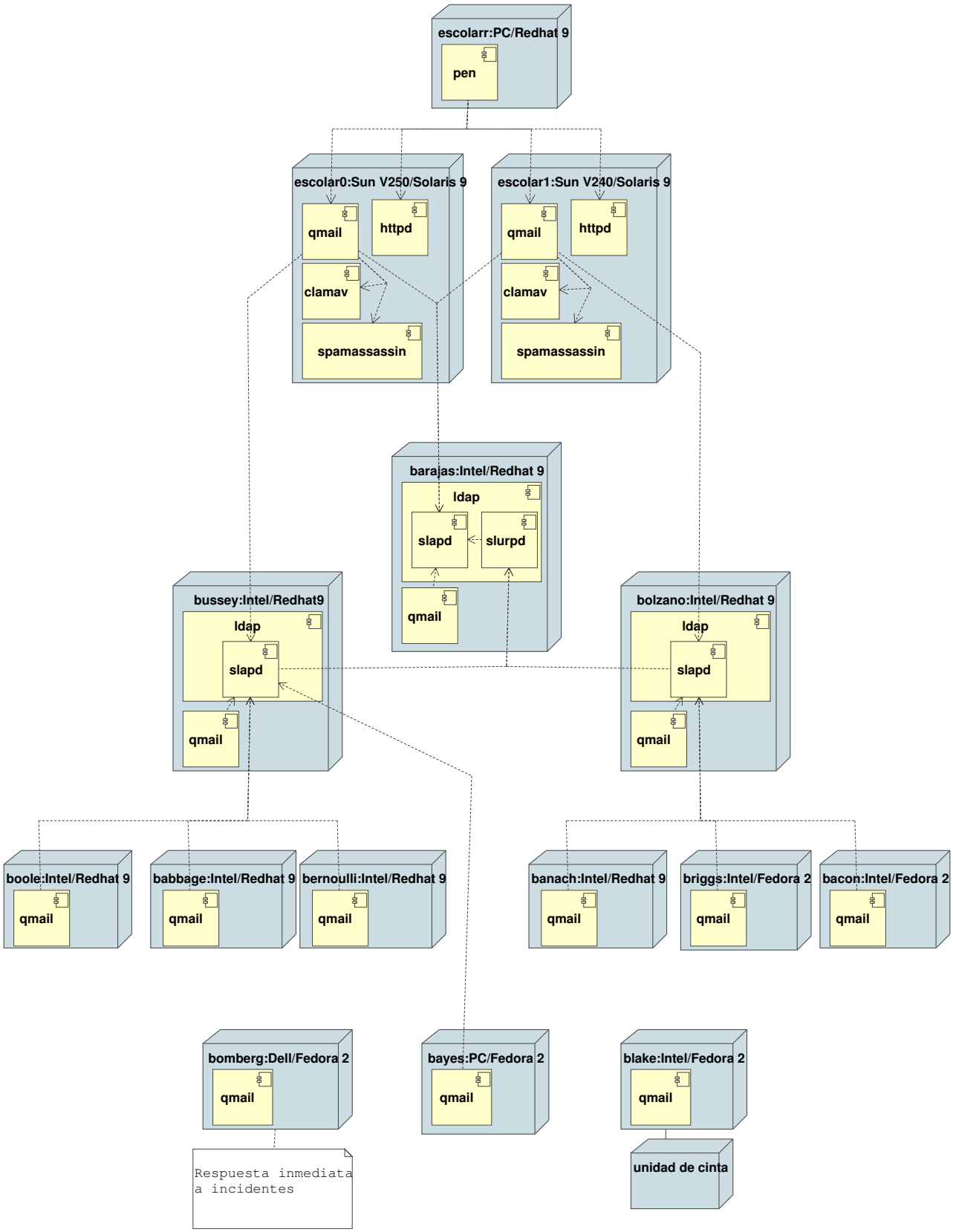


Figura 4.8: Distribución del sistema de correo reestructurado.

4.6. Situación actual del Sistema

Actualmente el sistema se encuentra funcionando con aproximadamente 300,000 cuentas de correo, bajo el esquema que se presentó en este capítulo.

Después de la reestructuración y reconfiguración del *cluster*, se puso en marcha por segunda ocasión y se pudo observar que el funcionamiento del sistema es eficiente y presenta tiempos de respuesta a solicitudes muy aceptables.

El reto fundamental de la reconfiguración del *cluster* fue resolver los problemas que aparecieron después de la primer puesta en marcha del sistema, aprovechando la flexibilidad que proporciona el *software* bajo el cual se decidió instalar el sistema.

En la actualidad, el sistema lo utilizan en todas las escuelas y facultades, quizá no para todos sus alumnos, sin embargo, gran parte de los profesores dirigen mensajes a sus alumnos vía este sistema de correo. En particular, los programas del Sistema Abierto y a Distancia es su herramienta principal de la actividad académica.

Así mismo, se ha logrado que las áreas de difusión y servicios interactúen constantemente con los alumnos, a través del correo, con la participación y coordinación de la Dirección General de Orientación Vocacional, que envía correos a población numerosa de alumnos.

Capítulo 5

Administración del Sistema

La administración del sistema de correo es una labor que involucra la constante actualización de la seguridad del mismo, la configuración inicial y posterior de programas, la administración de las cuentas de correo, la administración de los recursos, el monitoreo proactivo en intervalos de 5 minutos, la configuración de respaldos automáticos, la restauración de respaldos, así como la resolución de cualquier problema técnico que se presente.

Este capítulo servirá para describir los puntos anteriores que no se han detallado en el documento.

5.1. Administración del espacio en discos

El espacio de almacenamiento de un equipo es limitado. No se puede utilizar más espacio de disco del que se tiene disponible.

El espacio de almacenamiento es compartido por los programas. Un programa puede utilizar todo el espacio disponible; en ese caso, cualquier programa que necesite más espacio fallará. Un ejemplo de esto ocurría cuando se realizaban los respaldos automáticos y por descuido estos llenaban la partición */var*, donde se almacenan las bitácoras del sistema. Esto ocasionaba la baja del servicio Qmail entre otras cosas.

Para proteger a los programas y usuarios de que se queden sin espacio, es necesario contar con espacio en disco suficiente para cubrir las necesidades, realizar una adecuada partición del espacio en discos y contar con un sistema de monitoreo del espacio disponible por cada partición.

A continuación se presentará una tabla con la información de los discos, particiones y el cálculo del espacio necesario para ofrecer 20 MB de espacio para cada usuario del correo.

Equipo	No. Discos y Capacidad	No. Usuarios	20 MB x c/Usuario	Escuelas o Faculades Asignadas
Barajas	1 x **70 G**			
Boole	2 x 70 G 2 x 275 G **690 G**	58,505	1143 G	Contaduría, Ingeniería Medicina, CCH Sur
Babbage	3 x 70 G 2 x 275 G **760 G**	52,416	1024 G	Ciencias Políticas Derecho, Enfermería Filosofía, Prepa 5 y 6
Bernoulli	2 x 70 G 2 x 275 G **690 G**	45,156	882 G	Arquitectura, Ciencias, Artes Plásticas, Química, Trabajo Social, Música, Economía, Psicología, Veterinaria, Odontología.
Banach	2 x 70 G 2 x 275 G **690 G**	42,275	826 G	Prepas 1 a 4, Prepas 7 a 9
Bussey	3 x 70 G 2 x 275 G **760 G**	27,277	533 G	CCH Azcapotzalco, CCH Vallejo.
Bolzano	3 x 70 G 2 x 275 G **760 G**	25,188	492 G	CCH Naucalpan, CCH Oriente.
Briggs	2 x 70 G 2 x 275 G **690 G**	39,997	782 G	FES Acatlán, FES Aragón.
Bacon	3 x 70 G 2 x 275 G **760 G**	39,774	777 G	FES Cuautitlan, FES Iztacala, FES Zaragoza, Institutos, CUT, CUEC, Tlaxcala, Oaxaca.
Total	**5,800 G**	**330,588**	**6,459 G**	

Cuadro 5.1: Administración del espacio en disco en el *cluster*.

Nota: El número de usuarios está excedido por aproximadamente 30,000[1]. Esto debido a que se tienen que dar de baja cuentas, que por el momento no se han podido determinar con exactitud, debido a los movimientos de alumnos entre semestres. Se trabaja actualmente para buscar la forma de que esto no ocurra. La tabla refleja el balance de cuentas y espacio de almacenamiento para el periodo escolar 2007-2.

[1]Cifra proporcionada por personal autorizado de DGAE.

5.2. Administración de cuentas

Las cuentas de usuarios permiten definir qué usuarios tendrán acceso al sistema, y por lo tanto podrán utilizar el servicio de correo.

Para la administración de cuentas el sistema utiliza el servicio de directorio LDAP. OpenLDAP ofrece una serie de herramientas para la administración de datos en el directorio LDAP. Las cuatro herramientas más importantes para añadir, suprimir, buscar y modificar los datos almacenados se explican brevemente a continuación.

5.2.1. Creación de cuentas

Una vez que la configuración del servidor LDAP en la ruta */usr/local/etc/openldap/slapd.conf* sea correcta y esté lista (presenta las entradas apropiadas para *suffix, directory, rootdn, rootpw* e *index*), como se presentó en el la sección 1.3.5, lo que sigue es la introducción de registros. OpenLDAP cuenta con el comando *ldapadd* para esta tarea. LDAP es capaz de procesar el formato LDIF (formato de intercambio de datos de LDAP) visto en la sección 1.3.4.

Los primeros registros que se tiene que agregar son la raíz del directorio y el administrador:

```
# cat registros.ldif

dn: dc=escolar,dc=unam,dc=mx
objectclass: dcObject
objectclass: organization
o: UNAM
dc: escolar

dn: cn=administrador,dc=escolar,dc=unam,dc=mx
objectclass: organizationalRole
cn: administrador


# ldapadd -x -D "cn=administrador,dc=escolar,dc=unam,dc=mx" \
 -w password -c -S registros.skip -f registros.ldif

adding new entry "dc=escolar,dc=unam,dc=mx"
adding new entry "cn=administrador,dc=escolar,dc=unam,dc=mx"
```

Hecho esto se puede continuar con los registros de los usuarios, por ejemplo:

```
# cat registro_usuario.ldif

dn: uid=test,dc=escolar,dc=unam,dc=mx
cn: test
sn: test
givenName: test
registeredAddress: 0
telephoneNumber: 00000000
postalCode: 00000
objectClass: top
objectClass: person
objectClass: organizationalPerson
objectClass: inetOrgPerson
objectClass: qmailuser
objectClass: local
mail: test@escolar.unam.mx
mailMessageStore: /var/qmail/maildirs/test
mailQuotaSize: 20000000
uid: test
lugarNacimiento: 09
genero: 1
plantel: 003
carrera: 104
fechaNacimiento: 01011980
mailHost: boole.dgae.unam.mx
pregunta: 02
respuesta: 338c23e8e90aspoiiub404deac0bef4
userPassword:: {SSHA}e1NasdKHkjyuyyjkgliigOIticmJvPQ==
cambioPass: 1

...
...

# ldapadd -x -D "cn=administrador,dc=escolar,dc=unam,dc=mx" \
 -w password -c -S registro_usuario.skip -f registro_usuario.ldif

adding new entry "uid=test,dc=escolar,dc=unam,dc=mx"
```

Para que la cuenta pueda ser utilizada, además de agregar la entrada LDAP correspondiente, se debe de crear el maildir del usuario de la manera siguiente:

```
# mkdir /var/qmail1/maildirs1/test
# /var/qmail/bin/maildirmake /var/qmail1/maildirs1/test/Maildir
# chown -R vmail:vmail /var/qmail1/maildirs1/test
# ln -s /var/qmail1/maildirs1/test /var/qmail/maildirs/test
```

Nota: El ejemplo anterior presenta el caso especial en el cual, por limitaciones del S.O. o por limitaciones de espacio, no se pueda crear la nueva cuenta en el directorio */var/qmail/maildirs*, por lo que la opción que queda, es crear el maildir en un lugar alternativo */var/qmail1/maildirs1*, para luego crear la liga al directorio */var/qmail/maildirs*.

5.2.2. Modificación de cuentas

La herramienta *ldapmodify* sirve para modificar los registros almacenados. La manera más sencilla de hacerlo es mediante un archivo LDIF y, seguidamente, mandar el archivo modificado al servidor LDAP.

Por ejemplo si se quieren agregar a un registro el objectClass local, junto con sus campos fechaNacimiento, lugarNacimiento, género, plantel, carrera; pero además se quiere cambiar su mailHost a bernoulli.dgae.unam.mx y se quiere borrar el campo postalCode, del usuario, entonces se debe de realizar el siguiente procedimiento:

```
# vi modifyReg.ldif

dn: uid=test,dc=escolar,dc=unam,dc=mx
changetype: modify
add: objectClass
objectClass: local
-
add: fechaNacimiento
fechaNacimiento: 01011980
-
add: lugarNacimiento
lugarNacimiento: 01
-
add: genero
genero: 1
-
add: plantel
plantel: 005
-
add: carrera
carrera: 301
-
replace: mailHost
mailHost: bernoulli.dgae.unam.mx
-
delete: postalCode

...
...
EOF

# ldapmodify -D "cn=administrador,dc=escolar,dc=unam,dc=mx" \
 -w password -x -c -S modifyReg.skip -f modifyReg.ldif

modifying entry "uid=test,dc=escolar,dc=unam,dc=mx"
```

5.2.3. Búsqueda o lectura de datos

OpenLDAP ofrece, mediante el comando *ldapsearch*, una herramienta de línea de comando para buscar datos en un directorio LDAP y leer datos de él. Una consulta sencilla tendría la sintaxis siguiente:

```
# ldapsearch -LLL -D "cn=administrador,dc=escolar,dc=unam,dc=mx" \
 -w password -x -b "dc=escolar,dc=unam,dc=mx" "(uid=test)"
```

La opción -b determina la base de la búsqueda (la sección del árbol en la que debe realizarse la búsqueda). En el caso actual, es dc=escolar,dc=unam,dc=mx. La opción -x pide la activación de la autenticación simple. El filtro (objectClass=*) indica que deben leerse todos los objetos contenidos en el directorio. Se puede encontrar más información sobre el uso de *ldapsearch* en la página Man correspondiente (*ldapsearch(1)*).

5.2.4. Eliminación de cuentas

OpenLDAP cuenta con el comando *ldapdelete* para borrar registros. La sintaxis es similar a la de los comandos descritos anteriormente. Para suprimir, por ejemplo, el registro del usuario test, se debe de seguir el siguiente procedimiento:

```
# vi deleteReg.ldif

uid=test,dc=escolar,dc=unam,dc=mx
...
...

# ldapdelete -x -D cn=administrador,dc=escolar,dc=unam,dc=mx \
 -w password -c -f deleteReg.ldif
```

Y desde el servidor donde se encuentra el buzón:

```
# rm -rf /var/qmail/maildirs/test
```

5.3. Administración del filtrado de mensajes

El filtrado de mensajes de correo, en el *cluster*, se lleva acabo desde los servidores Web (escolar0 y escolar1). La decisión de hacerlo así se basa en que estos equipos son los que comienzan a procesar todos los mensajes de correo, tanto de entrada como de salida.

Esta sección describe brevemente la configuración necesaria para filtrar los mensajes de correo, con los paquetes ClamAV, SpamAssassin y *qmail-scanner*.

5.3.1. ClamAV

Configuración

La configuración de ClamAV se concentra en dos archivos : clamav.conf y freshclam.conf, ambos ubicados bajo el directorio */usr/local/clamav/etc/*. El primero de estos archivos contiene parámetros globales de ClamAV, como los serían generación de registros ("Logs") y límites de archivos a inspeccionar, mientras el segundo – freshclam.conf – incluye la configuración para consultar la Base de Datos ClamAV y así actualizar la información local referente a virus más recientes.

Al igual que SpamAssassin, los parámetros de ClamAV son extensos, sin embargo, sus valores predefinidos son razonables. Antes de proceder con la ejecución de esta configuración, es necesario editar un solo valor dentro del archivo clamav.conf, que simplemente elimine o comente el primer renglón que inicia con Example.

Ejecución

ClamAV está configurado para que inicie cuando arranca el sistema, pero en caso de que sea necesario levantarlo manualmente, se debe de usar el comando:

```
# /etc/init.d/clamd start
```

Verificamos que este funcionando

```
# ps -fea | grep clamav
    root   350     1  0   Jul 25 ?        0:00 /usr/local/clamav/sbin/clamd
```

5.3.2. SpamAssassin

Configuración

La configuración de SpamAssassin en el sistema de correo es llevada acabo de manera global, afectando todos los buzones/usuarios del sistema. Por el momento no se soporta la configuración individual, donde cada usuario define reglas de filtrado.

Los parámetros globales de SpamAssassin son definidos en un archivo llamado local.cf ubicado en el subdirectorio */etc/mail/spamassassin*, dicho archivo contiene las reglas que son aplicadas a cualquier buzón.

Cada regla en SpamAssassin posee un puntaje, valor que en caso de sobrepasar dicha norma, es considerado como spam. El valor promedio para que un correo electrónico sea considerado chatarra también es configurable como se describirá a continuación, finalmente, vale mencionar que para efectos prácticos, SpamAssassin posee puntajes predefinidos ("default") para todas sus reglas, mismas que pueden ser modificadas.

Aunque la nomenclatura utilizada para definir reglas es intuitiva, SpamAssassin posee un gran número de variantes, por lo que las siguientes reglas son sólo un ejemplo de cómo se filtra correo en el sistema de correo.

- Las siguientes reglas le indican a spamassassin que sobreescriba el subject del mensaje, en caso de detectarlo como spam, además de indicar el puntaje necesario para determinar si un mensaje es spam.

```
rewrite_subject 1
required_hits 5
rewrite_header Subject ****SPAM(_SCORE_)****
```

- La regla whitelist_from indica las direcciones de correo de remitentes, que nunca deben ser consideradas spam. En este caso se usan como excepción a la regla blacklist_from

```
whitelist_from *@escolar.unam.mx
whitelist_from *@bayes.dgae.unam.mx
whitelist_from becario2@servidor.unam.mx
```

- La regla whiteist_to indica las direcciones de correo de destinatario, que nunca deben ser consideradas spam. Al igual que en el ejemplo anterior se usan como excepción a la regla blacklist_to.

```
whitelist_to buzon@escolar.unam.mx
whitelist_to test*@escolar.unam.mx
whitelist_to *MAILER*@escolar.unam.mx
```

- Las siguientes reglas indican las direcciones de correo de remitentes y destinatarios que deben de ser consideradas spam. Con la siguiente configuración se lograron filtran los mensajes de dominios ya identificados como spamers.

```
blacklist_from *hinet.net *yyhmail.com *doramail.com *keromail.com
blacklist_to *hinet.net *yyhmail.com *doramail.com *keromail.com
blacklist_to *a*@escolar.unam.mx *e*@escolar.unam.mx *i*@escolar.unam.mx
blacklist_to *o*@escolar.unam.mx *u*@escolar.unam.mx
blacklist_to *A*@escolar.unam.mx *E*@escolar.unam.mx *I*@escolar.unam.mx
blacklist_to *O*@escolar.unam.mx *U*@escolar.unam.mx
```

En este caso, como las cuentas de correo están constituidas por la sintaxis [8-dígitos]@escolar.unam.mx, entonces, es fácil detectar spamers que intentan enviar correos a direcciones de correo inexistentes en el sistema, tales como, [letras]@escolar.unam.mx.

- Por último se agrega una regla para detectar mensajes que contenga algún contenido en el cuerpo del mensaje, además de asignarle una puntuación.

```
body DRUGS        /drugs/is
score DRUGS       10
```

Ejecución

Spamassassin está configurado para levantar sólo en el arranque del sistema, pero si es necesario levantarlo manualmente, se debe de usar el siguiente comando:

```
# /etc/init.d/spamd
```

Verificamos:

```
# ps -fea | grep spamd
   spamd   359   1    0   Jul 25 ?        0:07 /usr/local/bin/perl -T
/usr/local/bin/spamd -x -u spamd -H /export/home/spamd
   spamd   360   359  0   Jul 25 ?        0:07 /usr/local/bin/perl -T
/usr/local/bin/spamd -x -u spamd -H /export/home/spamd
   spamd   361   359  0   Jul 25 ?        0:07 /usr/local/bin/perl -T
/usr/local/bin/spamd -x -u spamd -H /export/home/spamd
   spamd   362   359  0   Jul 25 ?        0:07 /usr/local/bin/perl -T
/usr/local/bin/spamd -x -u spamd -H /export/home/spamd
```

5.3.3. Qmail-scanner

Configuración

La configuración de *qmail-scanner* se lleva acabo durante su instalación, por lo que para reconfigurarlo hay que cambiarse al directorio */usr/local/src/antivirus/qmail-scanner-1.25st* donde está el código fuente. Ahí hay que buscar el *script qms-config-script*, si no existe hay que crearlo y debe de contener todas las opciones:

```
# cat qms-config-script


if [ "$1" != "install" ]; then
INSTALL=
else
INSTALL="--install"
fi

./configure \
--domain escolar.unam.mx \
--admin test1 \
--scanners fast_spamassassin,clamscan \
--add-dscr-hdrs yes \
--dscr-hdrs-text X-Antivirus-escolar \
--sa-quarantine 0 \
--sa-delete 10 \
--sa-reject no \
--sa-subject ":SPAM:" \
--sa-delta 0 \
--sa-alt yes \
--sa-debug no \
--ignore-eol-check yes \
--unzip yes \
--lang es_ES \
--notify all \
--debug yes \
"$INSTALL"
```

Dentro de las opciones destacables de esta configuración se tienen:

- –sa-delete 10, la cual indica que si un mensaje obtiene una putuación de 10, entonces el mensaje debe de ser eliminado.

- –scanners fast_spamassassin,clamscan, indica a *qmail-scanner* que debe de usar a SpamAssassin y a ClamAV como fitros de correo.

Si se quiere más información

```
./configure -help
```

Teniendo el *script* de configuración, ya con las opciones adecuadas, entonces se instala *qmail-scanner* con el siguiente comando:

```
# ./qms-config-script install
```

Se deben de contestar las preguntas y se debe verificar que detecte correctamente a SpamAssassin y a ClamAV.

Finalmente se debe de agregar la variable QMAILQUEUE en el archivo de arranque */service/qmail-smtpd/run* y en el archivo de configuración */etc/tcp.smtp* (ver siguiente sección).

5.3.4. Antirelay

¿Qué es un "Open Relay"?

El ataque de Open Relay consta en usar el MTA como puente para correos (usualmente spam, aunque pueden ser muchas otras cosas) que de otra manera no podrían llegar a su destino, gracias a que los servidores bloquearon la dirección IP de origen.

De esta manera, la gente que manda spam de forma indiscriminada se ve obligada a usar otros servidores para esta tarea. Estos servidores que permiten que se envíe correos a través de ellos, se les denomina Open Relay.

Configuración para evitar ser un Open Relay

Para configurar y ejecutar Qmail teniendo en cuenta de no transformarlo en un generador de spam (Open Relay) se debe de editar el archivo de configuración */etc/tcp.smtp* como sigue:

```
# less /etc/tcp.smtp

# Utilizar qmailscannner para el correo procedente de 127.0.0.1
127.:allow,RELAYCLIENT="",QMAILQUEUE="/var/qmail/bin/qmail-scanner -queue.pl"
# [Qmailscanner y SpamAssassin se disparan por la presencia de la variable QMAILQUEUE]
<ip_balanceador_cargas>:allow,QMAILQUEUE="/var/qmail/bin/qmail-scanner-queue.pl"
# No permitir conexiones SMPT a nadie mas
:deny
```

Como se puede observar, al único servidor que se le permite ser Open Relay es al 127.*.*.*, es decir, al localhost.

Otra regla indica a Qmail que sólo se permitirá hacer peticiones SMTP, al servidor de balanceo de cargas. Esto debido a que la petición del cliente llega al balanceador de cargas y de aquí se toma la decisión de mandarlo a cualquiera de los dos Web's, por lo que esta configuración aplica para estos dos servidores.

Por último se debe dar el siguiente comando para que Qmail lea el archivo */etc/tcp.smtp* y aplique los cambios:

```
# qmailctl cdb
```

5.4. Copias de respaldo y recuperación

En esta sección se incluye la información sobre cómo hacer copias de respaldo del *cluster* y cómo recuperar archivos en caso de requerirse.

Respaldos

Por la forma en que se planificaron los respaldos, cada servidor tiene la responsabilidad de generar su propio archivo de respaldo para ser almacenado en el servidor de respaldos. Los tipos de respaldos que se generan son totales y no incrementales.

Debido a lo anterior se hicieron *script's* que se encuentran en cada equipo del *cluster*. Estos *script's* son como el que se muestra a continuación:

```
FECHA=‘date +%d_%B_%G‘
SERV=‘hostname | cut -d. -f1‘
ARCHIVO_TAR=/var/qmail/SO_$SERV"_"$FECHA.tgz

sudo tar -C / --exclude-from /home/admin/.administracion/exclude \
         -pzcf - / | ssh <ip> dd of=/var/respaldos/SO_$SERV"_"$FECHA.tgz
```

Este *script* puesto en un crontab sincronizado en todos los equipos, produce en el servidor de respaldos una lista de archivos con las copias de seguridad de todo el *cluster*:

```
# ls /var/respaldos
data-mysql_escolar0_11_August_2006.tar.gz  maildirs_boole_11_August_2006.tgz
log-apache_escolar0_11_August_2006.tar.gz  maildirs_briggs_11_August_2006.tgz
log-apache_escolar1_11_August_2006.tar.gz  maildirs_bussey_11_August_2006.tgz
log-horde_escolar0_11_August_2006.gz       openldap-data_barajas_11_August_2006.tgz
log-horde_escolar1_11_August_2006.gz       SO_babbage_11_August_2006.tgz
log-slapd_barajas_11_August_2006.gz        SO_bacon_11_August_2006.tgz
log-slapd_bolzano_11_August_2006.gz        SO_banach_11_August_2006.tgz
log-slapd_bussey_11_August_2006.gz         SO_barajas_11_August_2006.tgz
logs-pen_escolar_11_August_2006.tgz        SO_bernoulli_11_August_2006.tgz
maildirs_babbage_11_August_2006.tgz        SO_bolzano_11_August_2006.tgz
maildirs_bacon_11_August_2006.tgz          SO_boole_11_August_2006.tgz
maildirs_banach_11_August_2006.tgz         SO_briggs_11_August_2006.tgz
maildirs_barajas_11_August_2006.tgz        SO_bussey_11_August_2006.tgz
maildirs_bernoulli_11_August_2006.tgz      SO_escolarr_11_August_2006.tgz
maildirs_bolzano_11_August_2006.tgz
```

Como se puede observar, los nombres de los archivos contienen la información de lo que se está respaldando, a qué equipo pertenece y su fecha de creación.

Teniendo esto es fácil pasar esta información a una cinta magnética con un *script* que puede ser como el siguiente:

```
# cat respaldosCinta.sh
mt -f /dev/st0 rewind
tar -C /var/respaldos -zcf /dev/st0 SO_barajas_11_August_2006.tgz

mt -f /dev/nst0 fsf 1
tar -C /var/respaldos -zcf /dev/st0 SO_boole_11_August_2006.tgz

mt -f /dev/nst0 fsf 2
tar -C /var/respaldos -zcf /dev/st0 SO_banach_11_November_2004.tgz

...
```

Se puede notar que el comando *mt* lleva el control para posicionar la cinta en el lugar adecuado, y el comando tar realiza la copia del archivo en la cinta.

Recuperación de respaldos

Para recuperar los respaldos es necesario primero extraerlos de la cinta con el siguiente procedimiento:

```
# tar -tvf /dev/st0
-rw-rw-r-- user/user 1743697920 2006-08-11 02:42:39 SO_barajas_11_August_2006.tgz
-rw-rw-r-- user/user 948098233  2006-08-11 02:42:39 SO_boole_11_August_2006.tgz
-rw-rw-r-- user/user 1434232423 2006-08-11 02:42:39 SO_banach_11_August_2006.tgz
...
```

Una vez que se muestran los archivos de respaldos, en formato tar, se escoge el archivo que se quiere recuperar, por ejemplo el tercero, SO_banach_11_August_2006.tgz, se coloca la cinta en la pista correcta y se extrae:

```
# mt -f /dev/nst0 fsf 2
# tar -xvf SO_banach_11_August_2006.tgz
# ls -l SO_banach_11_August_2006.tg
-r--r--r--    1 user    user      1434232423   Aug 11 10:57 SO_banach_11_August_2006.tgz
```

El archivo extraído ya puede ser tratado como un archivo tar normal. Para más opciones ver el manual de tar.

5.5. Seguridad

Para proteger los servidores, lo primero que se debe verificar es que sólo los servicios necesarios estén activos, de manera que se evitan riesgos innecesarios. Para lograr esto no se instalan aplicaciones innecesarias, tal como lo es el sistema gráfico X, ni aplicaciones para tratamiento de gráficas, que además consumen espacio en disco.

Otro aspecto a considerar en la seguridad de los servidores es el uso de un filtro de paquetes, que permita habilitar sólo los puertos necesarios a los equipos autorizados, y por el contrario bloquear a todas las demás peticiones.

Por regla general las conexiones en el *cluster* se permiten sólo entre equipos del *cluster*, con la excepción del servidor de balanceo de cargas (por donde se reciben todas las peticiones de los usuarios), que permite las conexiones desde cualquier equipo a los puertos SMTP, http y https.

Otro aspecto que se debe de cuidar, es que el servicio de filtrado de paquetes levante, con todas sus reglas, al momento de arrancar el equipo. De esta manera se evita que por descuidos se comprometa la seguridad del servidor.

Finalmente es útil comprobar con alguna herramienta como nmap o portscan, que los puertos estén bloqueados adecuadamente.

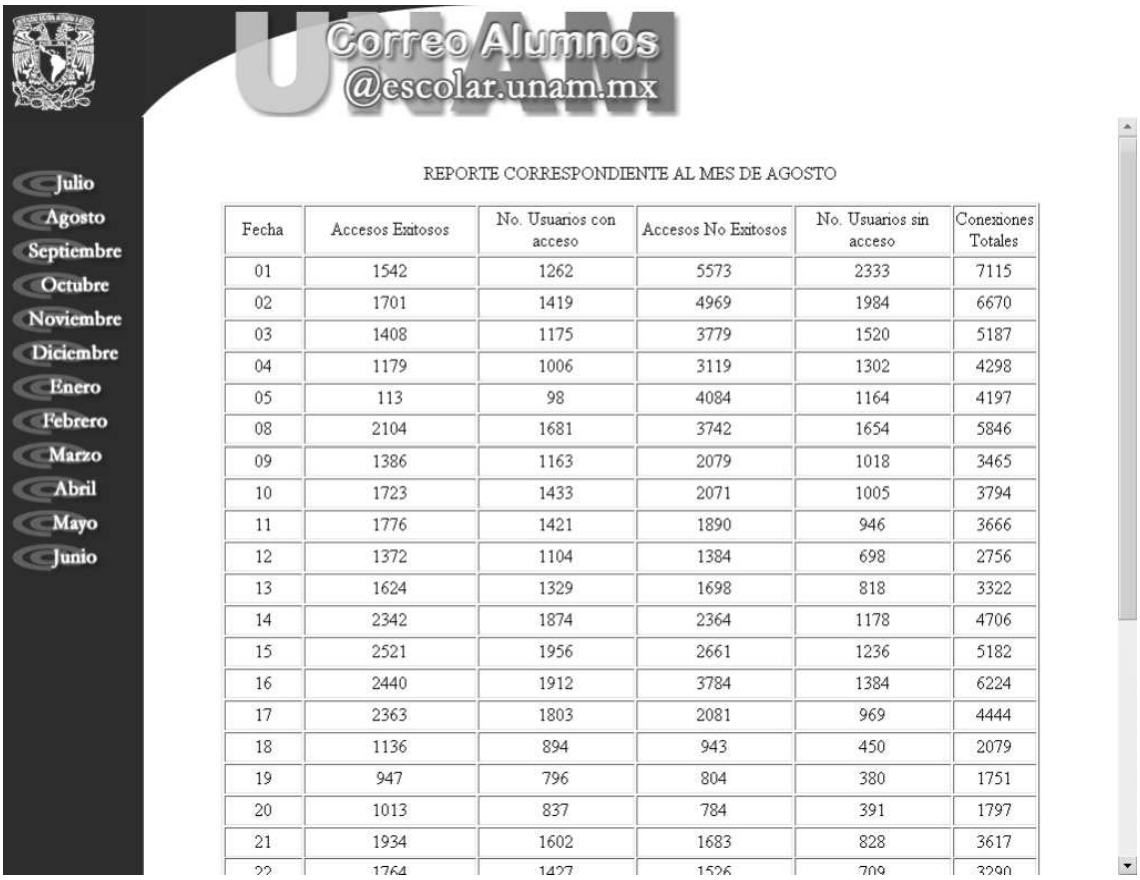
5.6. Estadísticas de uso

Para verificar el uso que se le da al sistema de correo se tiene que contabilizar el número de conexiones y el número de accesos exitosos y no exitosos que se tienen diariamente.

Con esto se pueden generar gráficos y tablas donde se desglose, por ejemplo: el número de conexiones diarias, por plantel, que hicieron sus alumnos durente un mes. Las posibilidades para desglosar la información pueden ser muy variadas y depende de la información que se quiera comparar.

Para recolectar los datos se utilizan básicamente las bitácoras que genera el sistema de correo.

A continuación se presenta una tabla, donde se desglosa en número de conexiones que se tuvieron diariamente en el mes de agosto de 2006.



REPORTE CORRESPONDIENTE AL MES DE AGOSTO

Fecha	Accesos Exitosos	No. Usuarios con acceso	Accesos No Exitosos	No. Usuarios sin acceso	Conexiones Totales
01	1542	1262	5573	2333	7115
02	1701	1419	4969	1984	6670
03	1408	1175	3779	1520	5187
04	1179	1006	3119	1302	4298
05	113	98	4084	1164	4197
08	2104	1681	3742	1654	5846
09	1386	1163	2079	1018	3465
10	1723	1433	2071	1005	3794
11	1776	1421	1890	946	3666
12	1372	1104	1384	698	2756
13	1624	1329	1698	818	3322
14	2342	1874	2364	1178	4706
15	2521	1956	2661	1236	5182
16	2440	1912	3784	1384	6224
17	2363	1803	2081	969	4444
18	1136	894	943	450	2079
19	947	796	804	380	1751
20	1013	837	784	391	1797
21	1934	1602	1683	828	3617
22	1764	1427	1526	709	3290

Figura 5.1: Estadística mensual de accesos a la aplicación de correo.

Otro tipo de gráfica que se genera, es el desglose de información acerca del número de usuarios que han visto un correo específico, enviado masivamente por el personal del sistema de correo.

Para consultar esta información es necesario escoger de un menú el título del mensaje a consultar, para luego desplegar la información por plantel del número de alumnos que han visto dicho mensaje.

Para generar este tipo de gráfica no se utilizan bitácoras, sino más bien, un algoritmo que implica en cada equipo donde se almacenan los correos, hacer un barrido diario de todos los títulos de los mensajes. Luego se juntan estos títulos en un solo archivo y a partir de este archivo se realiza la cuenta de mensajes vistos y no vistos.

El resultado es una gráfica como la que se encuentra abajo.

	23may06	24may06	25may06	26may06	28may06	29may06	31may06	28j
Arquitectura	10	19	22	28	33	36	42	1
Artes Plasticas	6	10	12	18	22	116	28	
Ciencias	26	42	56	65	76	86	96	1
Ciencias Politicas	15	32	46	50	64	66	76	1
Quimica	28	52	72	82	102	113	128	3
Contaduria y Administracion	58	108	140	158	192	216	248	4
Derecho	60	104	140	165	188	197	229	5
Economia	456	462	467	472	477	479	482	5
Enfermeria y Obstreticia	6	11	15	16	22	135	32	
Filosofia y Letras	56	90	111	127	150	159	182	3
Ingenieria	43	71	89	111	131	145	161	3
Medicina	21	36	52	61	83	89	106	3
Nacional de Musica	4	5	6	6	6	51	7	

Figura 5.2: Estadística diaria de accesos por plantel a la aplicación de correo.

5.7. Monitoreo

El monitoreo cotidiano del sistema de correo es realizado por medio de la interfaz Cacti. Como ya se había mencionado Cacti es una interfaz final para la herramienta RRDTool. A su vez RRDTool necesita del soporte SNMP para recolectar la información de los dispositivos y guardarla en bases de datos round robin.

Para acceder a la interfaz Cacti es necesario teclear desde un navegador la dirección en la cual se encuentra la aplicación. Hecho esto el navegador mostrará la página de inicio de la aplicación, ver figura 5.3, donde se pide un usuario y una contraseña para entrar.

Figura 5.3: Pantalla de autenticación Cacti.

Una vez que el usuario fue autenticado por la aplicación, ésta le despliega una consola, tal y como se muestra en la figura 5.4, desde la cual se tiene el control para administrar y configurar: la forma en que se recolectarán los datos, la forma en que se desplegarán las gráficas, etc.

Figura 5.4: Consola de administración Cacti.

Ajustes

Para facilitar el trabajo, Cacti permite dar de alta parámetros por defecto, los cuales se estarán utilizando para administrar las gráficas de los dispositivos a monitorear. Por ejemplo, para dar de alta la configuración específica SNMP de una red, se debe de hacer lo siguiente:

- Desde la Consola ir al elemento "Configuration -> Settings".
- En la pestaña "General", configurar adecuadamente los parámetros por defecto para SNMP (versión, usuario, contraseña, puerto, etc.) y guardar los cambios.

Para continuar con el ejemplo, para dar de alta un nuevo dispositivo y generar sus gráficas de monitoreo se pueden seguir los siguientes pasos:

- Desde la Consola ir al elemento "Management -> Devices"
- Se desplegará una pantalla como la de la figura 5.5, donde se muestran los dispositivos que ya están siendo monitoreados.

Figura 5.5: Lista de todos los dispositivos monitoreados, desde la consola.

- Ir a la liga “Add” (esquina superior derecha, figura 5.5)
- Se desplegará otra pantalla donde se deberá ingresar el hostname o ip del dispositivo que se quiera agregar. Además en el campo “Host Template” se deberá seleccionar el elemento “ucd/net SNMP Host”, ya que en este caso la recolección de información es vía SNMP. Luego presionar el botón “Create”.
- Hecho esto, se desplegará una página con la información del dispositivo que se va a crear. De aquí se deberá ir a la liga “Create Graphs for this Host”
- Se desplegará una página donde se seleccionarán las gráficas que se quieran agregar. Finalmente presionar el botón “Create” y las gráficas se comenzarán a generar.

Las gráficas de monitoreo

La herramienta RRDTool además de recolectar datos y almacenarlos en bases de datos, puede crear gráficas para representar estos datos. Una gráfica RRDTool es como la mostrada en la figura 5.6.

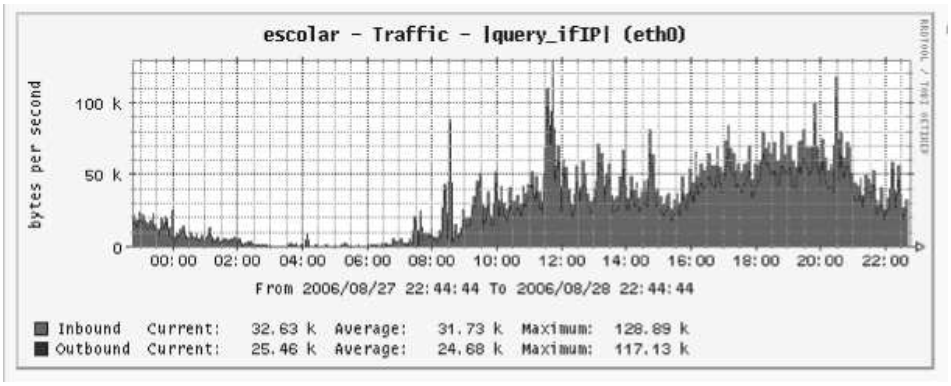


Figura 5.6: Gráfica de monitoreo.

Como se puede observar, la gráfica contiene el intervalo de tiempo en el cual se está haciendo la medición, además de los valores en texto de lo que está siendo representado. Por lo cuál resulta sencillo hacer la interpretación de la gráfica.

Visualización de las gráficas

Cacti provee de varias vistas configurables para desplegar conjuntos de gráficas.

Para monitorear del Sistema de Correo se decidió configurar la vista en forma de árbol, de manera que desplegará las gráficas de todos los equipos del *cluster*, por cada aspecto del sistema.

Esto significa que por cada nodo del árbol jerárquico mostrado a la derecha, se desplegará la información de todos los elementos del *cluster* respecto de ese nodo, que representa un aspecto a monitorear del sistema.

Por ejemplo si se sigue la liga
"Servicios -> correos encolados en Qmail" se mostrarán las gráficas de los correos encolados en todos los equipos en las últimas 24 hrs.

Así se pueden generar gráficas donde cada gráfica individual (rectángulos dentro de las figuras) corresponde a un equipo del *cluster*. Por ejemplo:

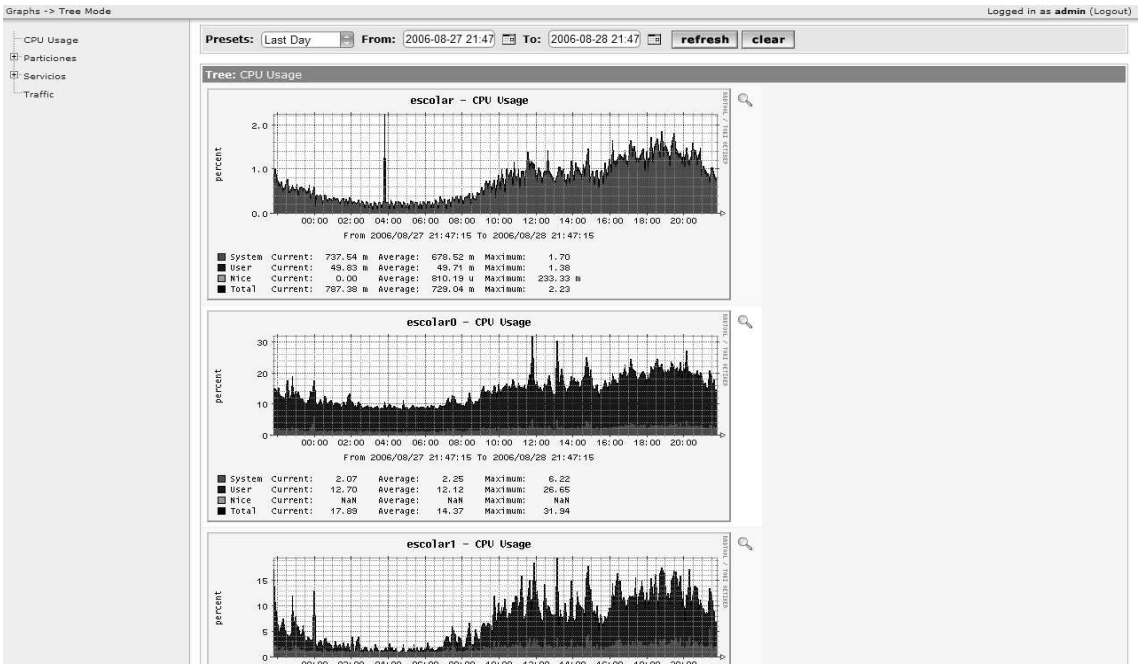


Figura 5.7: Actividad del CPU, 100 % equivale a capacidad máxima.

Figura 5.8: Porcentaje de partición llena (parte obscura), en este caso "/".

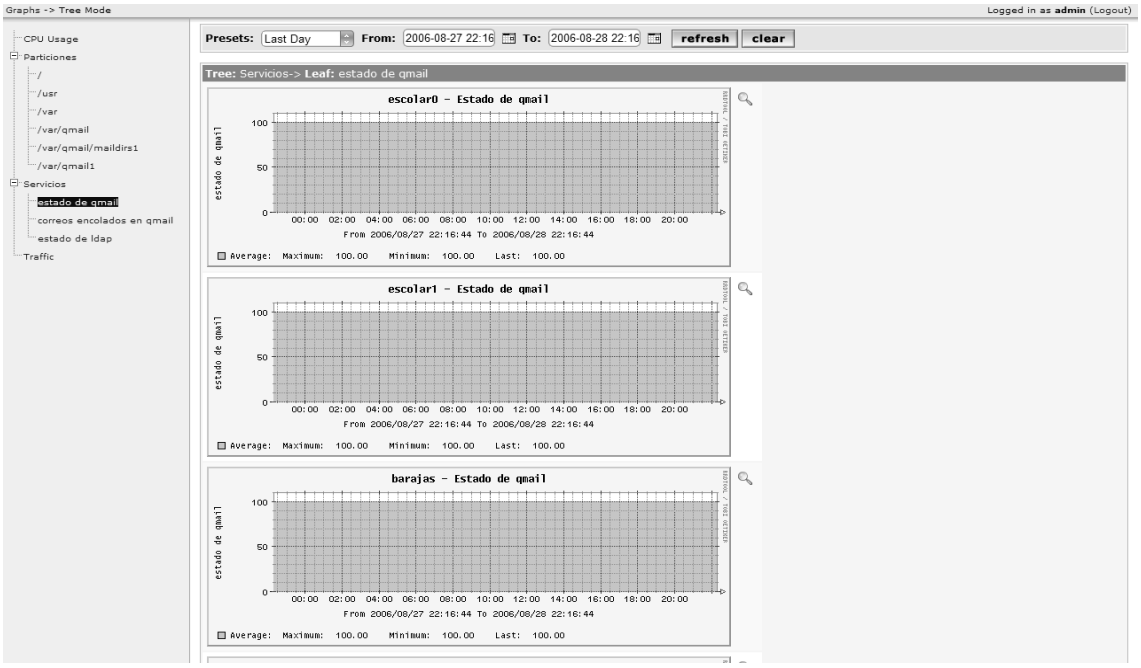


Figura 5.9: El estado del servicio Qmail. Esta gráfica funciona con un *script* en cada equipo que va sumando un porcentaje por cada subproceso de Qmail que funciona bien, hasta llegar al 100 %.

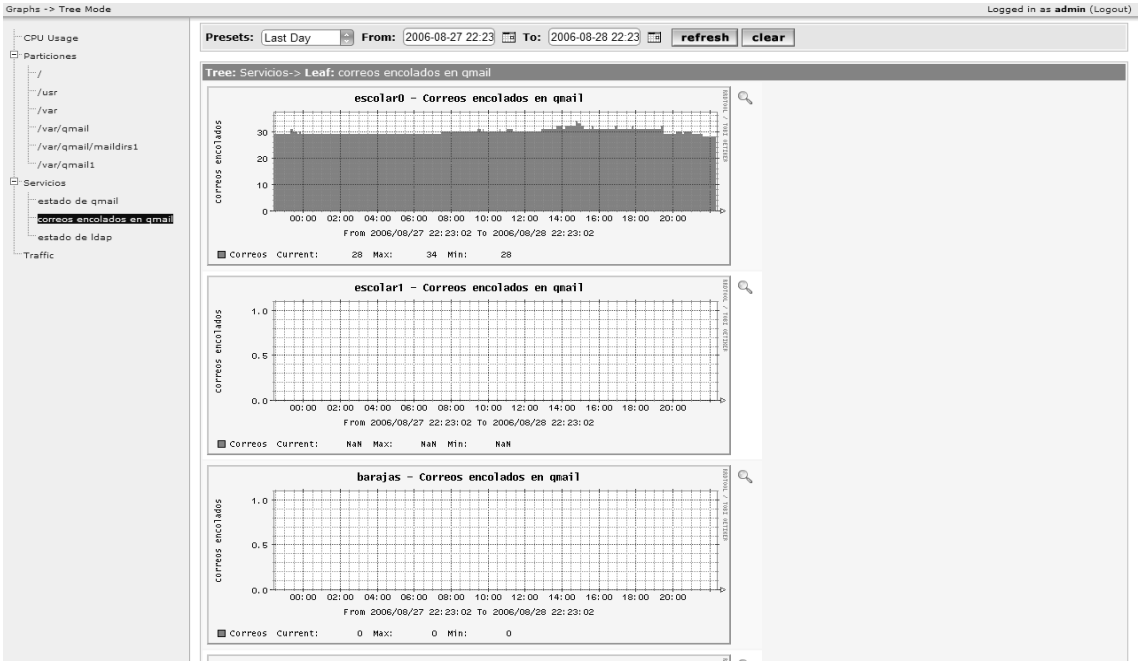


Figura 5.10: La cantidad de correos encolados en Qmail (leer los números para saber cantidad).

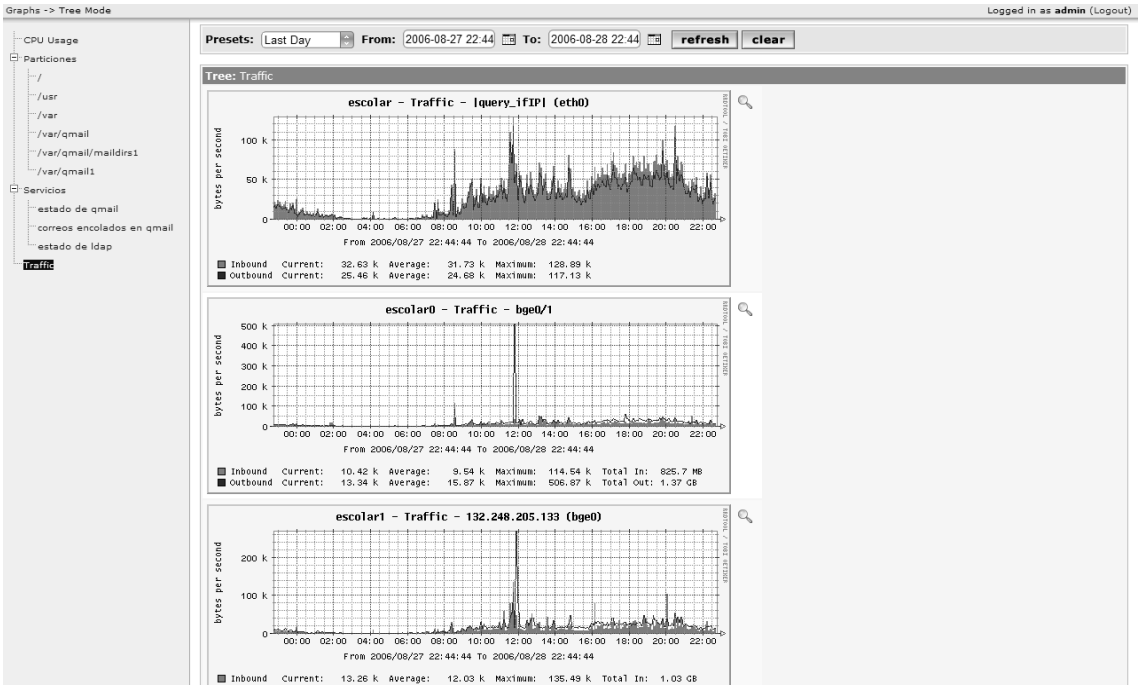


Figura 5.11: Tráfico entrante y saliente en las interfaces de red (en bytes). La parte rellena de la gráfica índica el tráfico entrante, y una ralla en otro color (no visible) índica el tráfico saliente.

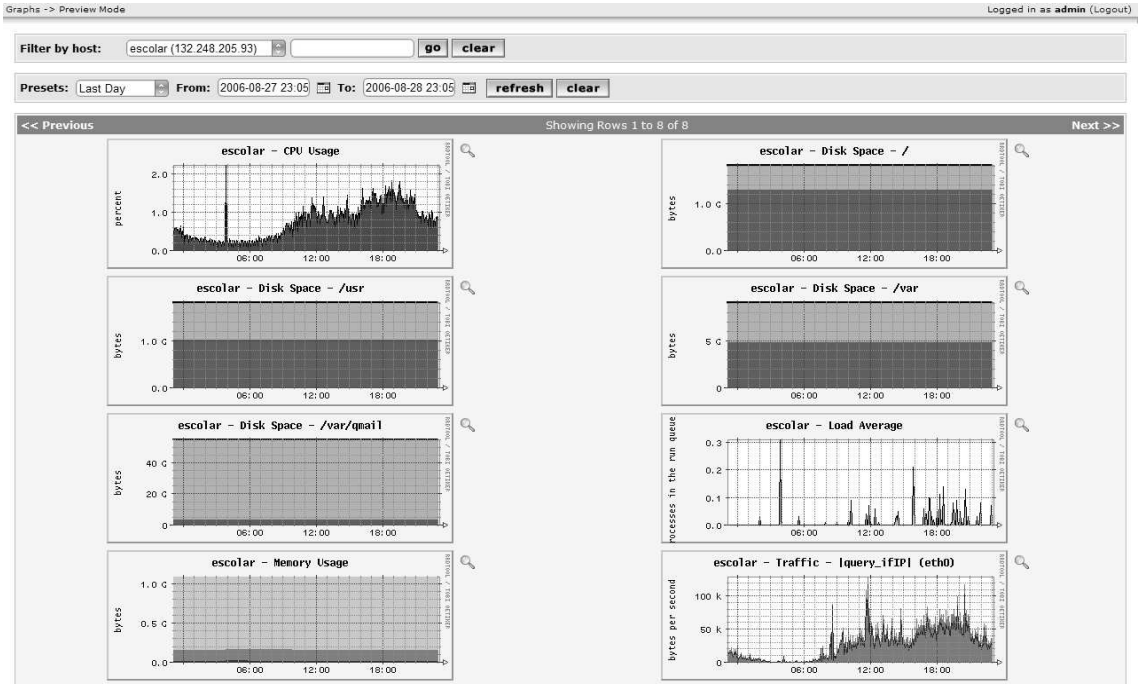


Figura 5.12: Esta vista agrupa todas las gráficas de un equipo del *cluster*.

5.8. Resolución de problemas

A continuación se presentará una lista de problemas que en ocasiones presenta el sistema de correo, junto con su solución:

- Qmail no levanta adecuadamente, ejemplo

```
# qmailctl stat
/service/qmail-send: up (pid 4910) 0 seconds
/service/qmail-send/log: up (pid 4911) 3332 seconds
/service/qmail-smtpd: up (pid 4912) 3332 seconds
/service/qmail-smtpd/log: up (pid 4913) 3332 seconds
/service/qmail-pop3d: up (pid 4914) 3332 seconds
/service/qmail-pop3d/log: up (pid 4915) 3332 seconds
messages in queue: 0
messages in queue but not yet preprocessed: 0
```

  Este ejemplo muestra cómo el servicio */service/qmail-send* no levantó adecuadamente. Pueden existir muchas razones para que esto ocurra, pero la causa mas común, es que otro proceso esté ocupando el puerto que está tratando de levantarse. Para asegurarse que esto no ocurra verificar con

```
# qmailctl stop
# netstat -nepa
```

  Buscar los puertos 25, 143, 628, si están activos, entonces buscar el proceso que lo levantó y matarlo con el comando kill.

  Por último levantar de nuevo Qmail y verificar que todos los servicios levanten correctamente.

```
# qmailctl start
# qmailctl stat
```

- La cola de correos de Qmail comienza a crecer, ejemplo

```
# qmailctl stat
/service/qmail-send: up (pid 4910) 3332 seconds
/service/qmail-send/log: up (pid 4911) 3332 seconds
/service/qmail-smtpd: up (pid 4912) 3332 seconds
/service/qmail-smtpd/log: up (pid 4913) 3332 seconds
/service/qmail-pop3d: up (pid 4914) 3332 seconds
/service/qmail-pop3d/log: up (pid 4915) 3332 seconds
messages in queue: 322
messages in queue but not yet preprocessed: 0
```

  Solución:

```
# qmailctl stop
# qmqtool -c
no rogue files found

//ó

# qmqtool -c
21/1534420 is rogue
removing 1534420 shrapnel
```

Si hay archivos corruptos en la cola de Qmail

```
# qmqtool -r
```

Finalmente levantar Qmail y verificar que se entreguen los correos encolados

```
# qmailctl start
# qmailctl stat
...
...
messages in queue: 0
messages in queue but not yet preprocessed: 0
```

En general cualquier problema con la cola de Qmail se puede solucionar con el comando *qmqtool* (consultar documentación, *qmqtool -h*)

- En caso de presentarse cualquier otro problema con Qmail, sus bitácoras muestran la información del error. Para monitorear las bitácoras de Qmail se debe de hacer lo siguiente

```
# cd /var/log/qmail
# tail -f current imapd/current smtpd/current \
          /var/qmail/service/qmqpd/main/log/current
...
```

- Otro problema que se puede presentar es que el servidor de balanceo de cargas no funcione adecuadamente. Para evitar esto se debe de verificar con el comando ps que los procesos *pen* estén activos y estén redireccionando peticiones de los protocolos SMTP, HTTP y HTTPS a los servidores correctos. En caso de que algo no funcione bien con pen se deben de levantar los procesos de la manera siguiente:

```
# pen -p /var/run/pen25.pid -l /var/log/pen25.log 25 \
  -r <ip_web1>:25 <ip_web2>:25
# pen -p /var/run/pen80.pid -l /var/log/pen.log80 80 \
  -r <ip_web1>:80 <ip_web2>:80
# pen -p /var/run/pen443.pid -l /var/log/pen.log443 443 \
  -h <ip_web1>:443 <ip_web2>:443
```

Como observación el parámetro *-r* es para indicar los servidores a seleccionar para retransmitir la conexión, bajo el esquema round robin, donde no importa a qué servidor fue redireccionado el cliente con anterioridad. Por el contrario el parámetro *-h*, indica los servidores a seleccionar para retransmitir la conexión, bajo el esquema hash, donde la retransmisión se intenta hacer al servidor ya asignado al cliente. Útil para el protocolo HTTPS.

Conclusiones

El propósito del trabajo realizado fue el de consolidar el sistema de correo electrónico, instalar y organizar los recursos de cómputo con la capacidad para atender los servicios de toda la población escolar de la Institución, mantener su funcionalidad y administrar la aplicación.

El modelo del sistema de correo electrónico abarca una gran cantidad de aspectos técnicos y operativos de mucha complejidad, que implican investigación, estudio, pruebas, ajustes e implementación, lo que nos ha permitido lograr una buena experiencia, fortaleciendo nuestro nivel académico, particularmente en sistemas distribuidos con alto volumen de transacciones, servicios a través del Web, seguridad, cluster con equipo rack y uso de software libre, así como la participación en proyecto Institucional, que aporta beneficios tanto a las actividades de docencia y de difusión como de aprovechamiento de las tecnologías de información y comunicación.

El sistema de correo electrónico para los alumnos de la UNAM, es un proyecto terminado, pero que mantenerlo en funcionamiento requerirá de una buena administración, la renovación de los recursos de cómputo, por falla u obsolescencia, crecimiento para dar más facilidades a los alumnos en el envío y recepción de correos, adecuaciones al software por necesidades operativas e instalación de nuevos elementos, resultado del avance tecnológico, por nuevos requerimientos, por renovación de las estrategias o por los alcances e impacto de su aplicación en la Universidad.

Es importante mencionar, que el proyecto de sistema de correo electrónico, ha sido desarrollado y se mantiene en operación, con la participación del equipo de trabajo del Departamento de Administración y Seguridad de Equipo, de la Subdirección de Diseño de Proyectos de la Dirección de Administración Escolar.

Referencias

[1] Dave Sill. Traducción de Iván Juanes Prieto. *Mi vida con Qmail*. http://www.es.qmail.org/documentacion/usuarios/lwq/html/mvq.html, 1999. [En línea; consultado el 31-Enero-2007].

[2] Djalil Chafaï. Traducción de Iván Juanes Prieto. *Introducción a Qmail*. http://qmail.free.fr/iaq-v0.4-sp/book1.html, 1998. [En línea; consultado el 31-Enero-2007].

[3] D. J. Bernstein. *Sendmail disasters - (Desastres en Sendmail)*. http://cr.yp.to/maildisasters/sendmail.html, 1997. [En línea; consultado el 31-Enero-2007].

[4] Inc. Sendmail. *Product Security, Security Advisories - (Seguridad del Producto, Reportes de Seguridad)*. http://www.sendmail.com/security/, 2006. [En línea; consultado el 31-Enero-2007].

[5] Eric Allman (Sendmail Consortium). *Sendmail Configuration Files - (Archivos de Configuración Sendmail)*. http://cr.yp.to/maildisasters/sendmail.html, 1997. [En línea; consultado el 31-Enero-2007].

[6] *Man Pages For Qmail 1.03 - (Páginas Man de Qmail 1.03)*. http://www.qmail.org/man/, 1998. [En línea; consultado el 31-Enero-2007].

[7] Andre Oppermann. *The big Qmail picture-(El diagrama Qmail)*. http://www.nrg4u.com/qmail/the-big-qmail-picture-103-p1.gif, 1998. [En línea; consultado el 31-Enero-2007].

[8] Michael Donelly. Traducción de Pere Benavent. *Introducción a LDAP*. http://www.ldapman.org/articles/sp_intro.html, Abril 2000. [En línea; consultado el 31-Enero-2007].

[9] Heinz Johner , Larry Brown y Johan Westman. *Understanding LDAP*. IBM, June 1998.

[10] Inc. Red Hat. *Lightweight Directory Access Protocol (LDAP).* http://www.europe.redhat.com/documentation/rhl7.1/rhl-rg-es-7.1/ch-ldap.php3, 2001. [En línea; consultado el 31-Enero-2007].

[11] Amparo López Gaona. *Introducción a la Base de Datos.* Vínculos Matemáticos, Facultad de Ciencias, UNAM, 2000.

[12] Gervase Markham. *dsmltools.org.* http://www.dsmltools.org/index.html, 1997. [En línea; consultado el 31-Enero-2007].

[13] Henning Brauer. *Life With qmail-ldap.* http://www.lifewithqmail.org/ldap/, 2004. [En línea; consultado el 31-Enero-2007].

[14] *The Qmail home page - (Página oficial de Qmail).* http://www.qmail.org/top.html, 2007. [En línea; consultado el 31-Enero-2007].

[15] *Blog qmail-ldap: how many users per directory is sane? - (Foro qmail-ldap: ¿cuantos usuarios por directorio es sano tener?* http://osdir.com/ml/mail.qmail.ldap/2005-01/msg00179.html, 2005. [En línea; consultado el 31-Enero-2007].

[16] Wiki qmail-ldap Team. *Qmail-LDAP Wiki - Control Files.* http://www.qmail-ldap.org/wiki/Control_Files, 2006. [En línea; consultado el 31-Enero-2007].

[17] Andre Oppermann. *The big Qmail-LDAP picture - (El diagrama de Qmail-LDAP).* http://www.nrg4u.com/qmail/the-big-qmail-ldap-picture-20031112.pdf, 2003. [En línea; consultado el 31-Enero-2007].

[18] The Apache SpamAssassin Project. *Página oficial de SpamAssassin.* http://spamassassin.apache.org/. [En línea; consultado el 31-Enero-2007].

[19] Alan Schwartz. *SpamAssassin.* O'Reilly, July 2005.

[20] Equipo ClamAV. *Página oficial de Clam AntiVirus.* http://www.clamav.net/, 2007. [En línea; consultado el 31-Enero-2007].

[21] Jason Haar y equipo Qmail-Scanner. *Qmail-Scanner: Content Scanner for Qmail - (Qmail-Scanner: explorador de contenidos para Qmail).* http://qmail-scanner.sourceforge.net/, 2007. [En línea; consultado el 31-Enero-2007].

[22] Jeremy Kister. *Página oficial de Qmqtool*.
`http://jeremy.kister.net/code/qmqtool/`.
[En línea; consultado el 31-Enero-2007].

[23] Douglas Mauro and Kevin Schmidt. *Essential SNMP, 2nd Edition*. O'Reilly, September 2005.

[24] Tobias Oetiker. *Página oficial de RRDTool*.
`http://oss.oetiker.ch/rrdtool/`, 2007.
[En línea; consultado el 31-Enero-2007].

[25] The Cacti Group. *Cacti: The Complete RRDtool Graphing Solution - (Cacti: la solución a la representación gráfica con RRDtools)*.
`http://cacti.net/`, 2007.
[En línea; consultado el 31-Enero-2007].

[26] James Rumbaugh, Ivar Jacobson y Grady Booch. *The Unified Modeling Language Reference Manual*. Addison-Wesley, 1999.

[27] Ulric Eriksson. *Página oficial de Pen*.
`http://siag.nu/pen/`, Julio 2004.
[En línea; consultado el 31-Enero-2007].