INFORME DE EVALUACIÓN DE LA SUITE DE COMUNICACIÓN Y COLABORACIÓN EXPRESSO V3 PARA EL SISTEMA DE CORREO ELECTRÓNICO DEL ESTADO VENEZOLANO

Resumen Ejecutivo

A continuación se presenta un informe de evaluación de la Suite de Comunicación y Colaboración Expresso V3 para el Sistema de Correo Electrónico del estado Venezolano. El informe se estructura de la siguiente manera: en primer lugar se presentan los diversos subsistemas que implican la implantación de un sistema de correo electrónico para el estado venezolano. En segundo lugar, se hace referencia a la selección de tecnologías con miras a la implantación en el corto plazo de un sistema de correo electrónico para el estado venezolano. En tercer lugar, se hace una evaluación de las diversas aplicaciones para el manejo de la interfaz web que es la capa donde propiamente entra Expresso V3 y en la que se realiza la evaluación. Por último, se consideran los asuntos relacionados con la capacidad necesaria de hardware para el sistema piloto de corto plazo.

En la evaluación realizada para la interfaz web Expresso V3 y RoundCube aparecen como las mejores alternativas. Se presentan también algunas consideraciones a tomar en cuenta con respecto al contrato de Licencia que utiliza el sistema Postfix (utilizado por Expresso V3 en la interfaz SMTP, POP e IMAP) y a las vulnerabilidades documentadas del sistema.

INFORME DE EVALUACIÓN DE LA SUITE DE COMUNICACIÓN Y COLABORACIÓN EXPRESSO V3 PARA EL SISTEMA DE CORREO ELECTRÓNICO DEL ESTADO VENEZOLANO

El correo electrónico actualmente es una herramienta inherente a todos los procesos internos de los entes públicos o privados, para la interacción de las personas con las instituciones del Estado y como servicio público de comunicación entre personas. Por esta razón, es una responsabilidad del Estado Venezolano garantizar que sus instituciones cuenten con el talento humano, el conocimiento pleno de las tecnologías involucradas y las plataformas informáticas y de telecomunicaciones que hacen posible la implementación de sistemas de correo electrónico que proporcionen los más altos niveles de seguridad informática, disponibilidad, desempeño y usabilidad, manteniendo como prioridad el aseguramiento de la soberanía tecnológica. En resumen, se trata de un esfuerzo coordinado de desarrollo, investigación y principalmente de apropiación de tecnologías libres en el área de sistemas de correo electrónico de alta demanda.

Se ha planteado entonces una estrategia de implementación en tres etapas: *despliegue piloto, despliegue en toda la APN y despliegue para público en general.* Las premisas fundamentales se pueden resumir de la siguiente manera:

1. Es necesario implementar sistemas informáticos que permitan lograr soberanía tecnológica y seguridad informática en el intercambio de mensajes de datos.
2. Esta es una muy importante oportunidad para proporcionar un impulso significativo al despliegue y apropiación de tecnologías libres en diversos ámbitos.
3. Son fundamentales los procesos de formación en administración de infraestructura informática y telecomunicaciones mediante tecnologías libres.

Como propuesta de producto para el sistema de correo electrónico se nos ha presentado la alternativa ExpressoV3 que es una suite de comunicación y colaboración que surgió como una derivación del proyecto eGroupWare en el año 2004. El propósito de este informe es hacerle una evaluación al sistema de correo ExpressoV3 en comparación con otras alternativas de manera de mostrar sus ventajas, sus desventajas, sus oportunidades y posibles amenazas.

Antes de comenzar nuestra evaluación debemos comenzar por mostrar los subsistemas involucrados en el sistema de correo.

Subsistemas Involucrados

Un sistema de correo electrónico de las características planteadas requiere una plataforma informática que se puede esquematizar en los siguientes niveles, en los que se encuentran subsistemas con su propia complejidad interna:

1. Nivel Infraestructura base:
   - Sistemas de Bases de Datos redundantes, escalable, con alta disponibilidad.
   - Sistemas de archivo para aglomerados de servidores, con capacidad para almacenamiento masivo de datos.
   - Sistemas de respaldo masivo de datos.

2. Nivel almacenamiento y configuraciones:
   - Sistemas para almacenamiento estructurado de buzones de correo electrónico y demás archivos del usuario.
   - Sistemas para almacenamiento de datos de configuración del entorno personalizado de las cuentas de usuario en las aplicaciones.

3. Nivel autenticación
   - Manejo de los procesos de autenticación de las usuarias y usuarios.

4. Nivel acceso a datos
   - Sistemas para acceso controlado a los datos de las usuarias y usuarios.

5. Nivel protocolos específicos.
   - Sistemas para manejo de protocolos SMTP, POP e IMAP, todos ellos sobre SSL.
   - Sistemas para identificación y filtrado de mensajes de correo no deseado.
   - Sistemas para detección y eliminación de virus y demás software maligno de los mensajes de correo.

6. Nivel de acceso al sistema:
   - Servidores para hospedaje de aplicaciones que conformen la interfaz web del sistema.
   - Servidores para proporcionar acceso al contenido estático de las aplicaciones que conformen la interfaz web del sistema.

7. Nivel de fachada y aseguramiento de calidad:
   - Servicios y sistemas para balance de carga y alta disponibilidad.
   - Manejo de las conexiones SSL hacia las aplicaciones web.

8. Nivel de servicios de direccionamiento
   - Sistema para manejo seguro de protocolos para DNS.

9. Telecomunicaciones:
   - Enlaces con las redes de datos para servicio a usuarios.

- Enlaces entre nodos del sistema.

En este último nivel es importante tomar en cuenta que cualquier conexión no cifrada es susceptible de ser interceptada, sin importar en donde se encuentre. Por lo tanto, todos los enlaces deben usar cifrado de datos, aún cuando se trate de conexiones internas de un nodo o entre nodos del sistema.

En el presente informe el énfasis se encontrará en las primeras dos etapas del proyecto. Un resumen del esquema de implementación, centrado en la parte de la aplicación web, se describe a continuación. En ese esquema se tiene la premisa que la demanda puede estar en el orden de las 10000 conexiones por segundo y debe garantizarse el balance de carga, escalabilidad y alta disponibilidad. Ello conduce a un despliegue en varias capas:

1. DNS: cada nombre de dominio del sistema registrado con por lo menos tres IP públicas. Es deseable que esas IP estén en redes distintas y lo ideal es que estén manejadas por centros de datos ubicados en diferentes zonas geográficas y que las respuestas del sistema DNS sean optimizadas según la zona geográfica de origen de las peticiones. Con el orden de publicación configurado como cíclico o aleatorio se obtiene un primer nivel, bastante grueso, de balance de carga. Los servidores de nombres que alojen la zona correspondiente al dominio del sistema deberán tener alta disponibilidad y ser escalables.

2. Cortafuegos: Configurado para sólo permitir a las IP del sistema conexiones entrantes de tipo http (y https si fuera necesario). En esas IP sólo estaría publicado el acceso a las interfaces con usuarias y usuarios, para otras tareas del sistema que requieran conexión externa se usarían otras IP. Si se deseara habilitar acceso administrativo remoto, ello debería realizarse a través de una VPN con acceso resguardado por algún protocolo de golpeteo de puertos mediante reseña criptográfica. Esta capa deberá tener alta disponibilidad mediante linux-ha o algún sistema equivalente.

3. Enrutamiento - Traducción de direcciones: en esta capa las conexiones entrantes a los puertos http de las IP registradas para el dominio del sistema son dirigidas mediante DNAT a los servidores de la capa de fachada. La razón para incorporar esta capa en el sistema es el facilitar el mantenimiento y la escalabilidad de la capa de fachada descrita más adelante. Los servicios de enrutamiento y traducción de direcciones permiten que al momento que se haga necesario sacar del sistema uno de los servidores de la capa de fachada, se pueda detener la admisión de conexiones nuevas hacia él y entonces sólo retirarlo cuando haya respondido las peticiones que tenga en proceso. De esa manera, el retirar un servidor de la capa de fachada no generaría errores en las capas superiores o a nivel de usuario. Por otro lado, mediante el DNAT también es posible un segundo nivel un poco más refinado de balance de carga. Deberá cuidarse que los enrutadores tengan suficiente disponibilidad de CPU y RAM para manejar el volumen de conexiones esperado, en especial debido a que el uso de DNAT requiere el mantener una tabla de conexiones activas. Esta capa también deberá tener alta disponibilidad mediante linux-ha o algún sistema equivalente.

4. Fachada: Consta de por lo menos dos balanceadores de carga en cada centro de datos implementados

con, por ejemplo, linux virtual server. En esta capa se realizaría de forma más refinada el balance de carga, incorporando supervisión de los equipos de la capa inferior y ajuste dinámico de la carga enviada a cada uno. En esta capa el procedimiento contemplado en el balanceador de carga para el retiro de un nodo de servicio permite el realizar ese tipo de tareas sin generar errores a nivel de usuario. Esta capa también deberá tener alta disponibilidad mediante linux-ha o algún sistema equivalente. Obsérvese que tanto esta capa como las superiores a ella no están circunscritas a la aplicación web publicada por el sistema ni a los protocolos http ó https. Por lo tanto es razonable esperar que todas estas capas formen parte de la infraestructura básica disponible en cada centro de datos para los distintos sistemas alojados en él.

5. Proxy reverso: Consta de un conjunto de servidores que implementan el nginx configurado como proxy reverso. La alta disponibilidad del sistema está atendida en las capas superiores y por tanto no es necesario incorporarla en ésta. Buena parte de la escalabilidad del sistema está asociada a la cantidad de proxys reversos disponibles, en especial si está contemplado el uso de protocolo https debido a que la carga computacional de las conexiones cifradas debe ser atendida en esta capa y esa carga es bastante significativa cuando se trata de escenarios de alta demanda. Para un desempeño óptimo, cada uno de los proxys estaría configurado para distribuir las conexiones hacia todos los nodos de la capa inferior. La razón para incorporar esta capa al sistema reside en la necesidad de optimizar el manejo del alto volumen de concurrencia esperado en lo concerniente a los protocolos http y https, descargando de esa tarea a las capas inferiores y aprovechando los avances tecnológicos del nginx como duende http de nueva generación. Obsérvese que en esta capa el único elemento asociado directamente al sistema es la configuración de los proxy reversos y los certificados SSL, por lo tanto es razonable esperar que esta capa también esté disponible como parte de la infraestructura informática básica de cada centro de datos.

6. HTTP: En esta capa, también implementada con nginx, se manejaría lo concerniente a las redirecciones, reescrituras, publicación de contenido estático y demás asuntos HTTP que pudieran necesitarse y sean muy específicos de la aplicación web. Al esperar que el servicio de proxy reverso sea parte de la infraestructura informática básica de los centro de datos, y por tanto esté compartida por diversos sistemas, se hace necesario el colocar el manejo http específico a la aplicación web en una capa separada, la cual estaría dedicada al sistema. La alta disponibilidad está atendida en las capas superiores y por tanto no es necesario incorporarla en ésta. Para efectos de sacar del sistema a uno de los servidores de esta capa, primero se desactivaría en la configuración de la capa de proxy reverso y luego, cuando sea completada la atención a las peticiones activas, se puede retirar el servidor sin generar errores en las capas superiores o a nivel de usuario. Para un desempeño óptimo, cada uno de los servidores http estaría configurado para distribuir las conexiones hacia todos los nodos de la capa inferior. El contenido estático estaría alojado en la capa de almacenamiento de archivos descrita más adelante.

7. Aplicación: En esta capa se encuentra el conjunto de servidores de aplicación que responden a las peticiones que llegan de la capa HTTP a través de protocolos livianos tales como FastCGI o WSGI. La alta disponibilidad del sistema está atendida en las capas superiores y por tanto no es necesario

incorporarla en ésta. Parte importante de la escalabilidad del sistema está asociada a la cantidad de servidores de aplicación disponibles en esta capa. Para efectos de sacar del sistema a uno de los servidores, primero se desactiva en la configuración de la capa de http y, una vez haya terminado de atender las peticiones abiertas, puede completarse su salida del sistema sin que ello genere errores a nivel de las capas superiores o del usuario. Es importante que en la aplicación la información de estado resida completamente en el sistema de base de datos ó el sistema de archivos compartidos proporcionado por las capas inferiores, de manera que la incorporación o desincorporación de servidores de aplicación pueda realizarse sin afectar de manera alguna las conexiones ya establecidas.

8. Fachada de Base de Datos y almacenamiento de archivos: Esta capa proporciona una interfaz sencilla y unificada de los servicios de base de datos y almacenamiento de archivos a las capas superiores y se encarga de manejar los requerimientos de alta disponibilidad, escalabilidad y diversidad geográfica del despliegue. También forman parte de esta capa el subsistema de respaldo automático de datos. Para el caso que el motor de base de datos sea postgresql en el software a considerar para la implementar esta capa pueden estar: pgpool para el balance de carga en el cluster de cada centro de datos, linux-ha para proporcionar alta disponibilidad y bucardo para la replicación síncrona ó asíncrona entre los cluster de base de datos ubicados en distintas locaciones geográficas. En cuanto al almacenamiento de archivos se plantea el uso de un sistema de archivos basado en Lustre. La escalabilidad de esta capa puede manejarse mediante un esquema sencillo donde sus servicios están disponibles a través de varios servidores, que estarían coordinados entre sí para mantener la integridad de los datos; en escenarios de mayor demanda se puede considerar incorporar balance de carga mediante el uso de linux-virtual-server. Es razonable esperar que los servicios de esta capa, exceptuando quizás la diversidad geográfica, puedan estar disponibles como parte de la infraestructura informática básica de los centro de datos donde sea alojado el sistema.

9. Nodos de Base de datos y de almacenamiento de archivos: En esta capa se maneja la información de estado del sistema. La alta disponibilidad es proporcionada por la capa inmediatamente superior y por tanto no es necesario incorporarla en ésta. Similarmente, el proceso de incorporación o retiro de nodos de bases de datos o almacenamiento de archivos se realizaría según lo establecido en esa misma capa. Si el tamaño de la base de datos y los archivos lo permite, es recomendable que la dotación de RAM de estos nodos sea suficiente como para mantener la totalidad de los datos en el caché del sistema de archivos. De esa manera se eliminan las latencias asociadas a la inevitable lentitud del almacenamiento secundario, comparado con la capacidad de respuesta de la RAM. La escalabilidad de esta capa consiste en sencillamente agregar mayor cantidad de nodos a los cluster de base de datos y almacenamiento de archivos. El desempeño general de todo el sistema está influido en buena medida por la capacidad de respuesta de esta capa.

Para la implementación del esquema descrito es recomendable utilizar tecnologías de virtualización de servidores basada en Xen, por supuesto asegurándose que los nodos de respaldo de aquellas partes del sistema que tengan implementada alta disponibilidad estén alojados en servidores físicos distintos a los nodos principales. Por otro lado, tratándose de un sistema informático que deberá atender un alto volumen de conexiones, es recomendable que se utilice la tecnología de servidores de hojilla para de

esa manera conseguir alta densidad en el uso de espacio físico en los centros de datos donde sea alojado el sistema.

Sobre la selección de tecnologías para el escenario del corto plazo

Es importante advertir que en este escenario de corto plazo no tiene factibilidad la implementación de un sistema de archivos para aglomerados de servidores. La complejidad técnica de ese sistema y su rol crítico dentro de la arquitectura hace que sea indispensable un mayor plazo para lograr una implementación que cumpla rigurosamente con los mayores estándares de seguridad, integridad de datos, desempeño, disponibilidad y escalabilidad.

Esta circunstancia obliga a que en el escenario de corto plazo el almacenamiento de datos se encuentre centralizado en un solo servidor para cada dominio a atender, lo cual aún cuando sea implementado siguiendo los estándares más rigurosos no deja de constituirse en un punto de falla. Debido a que este servidor es responsable de asegurar la integridad de gran parte de los datos de las usuarias y usuarios, colocar a este servidor en un subsistema de alta disponibilidad tiene una complejidad no muy distinta a la de la implementación de un sistema de archivos para aglomerados y por lo tanto tampoco se ve muy factible en el corto plazo.

Con esa advertencia presente, en base a la investigación realizada se propone el siguiente esquema de implementación:

1. Nivel Infraestructura base:
    - Sistemas de Bases de Datos basado en PostgreSql, pgPool, Bucardo, MongoDB, linux-ha, Linux Virtual Server.
    - Dispositivos de almacenamiento no estructurado de datos basados en arreglos RAID 60 publicados en red mediante iSCSI, usando Linux MDRAID e IET.
    - Sistemas de respaldo masivo de datos usando rdiff-backup y similares.

2. Nivel almacenamiento y configuraciones:
    - Almacenamiento estructurado de buzones de correo electrónico y demás archivos del usuario basado en sistema de archivos btrfs o ext4, implementado sobre volúmenes lógicos conectados a dispositivos iSCSI.
    - Sistemas para almacenamiento de datos de configuración del entorno personalizado de las cuentas de usuario en las aplicaciones basado en directorios implementados con OpenLDAP.

3. Nivel autenticación
    - Procesos de autenticación de las usuarias y usuarios basada en directorios implementados con OpenLDAP.

4. Nivel acceso a datos
   - Sistemas para acceso controlado a los datos de las usuarias y usuarios basado en Dovecot.

5. Nivel protocolos específicos.
   - Sistemas para manejo de protocolos SMTP, POP e IMAP, todos ellos sobre SSL, implementado preferiblemente mediante EXIM 4.
   - Sistemas para identificación y filtrado de mensajes de correo no deseado basado en SpamAssassin.
   - Sistemas para detección y eliminación de virus y demás software maligno de los mensajes de correo basado en ClamAV o Amavisd.

6. Nivel de acceso al sistema:
   - Servidores para hospedaje de aplicaciones que conformen la interfaz web del sistema, basados en uwsgi. La selección de la aplicación web específica es materia de la siguiente sección:
   - Servidores para proporcionar acceso al contenido estático de las aplicaciones que conformen la interfaz web del sistema, basados en nginx.

7. Nivel de fachada y aseguramiento de calidad:
   - Servicios y sistemas para balance de carga y alta disponibilidad, basados en Linux-HA y Linux Virtual Server.
   - Manejo de las conexiones SSL hacia las aplicaciones web, basado en nginx.

8. Nivel de servicios de direccionamiento
   - Sistema para manejo seguro de protocolos para DNS, basado en Bind 9.

9. En telecomunicaciones, implementación de cortafuegos, enrutamiento y traducción de direcciones basada en Linux-HA y Shorewall.

Selección de la aplicación web

El componente de acceso al sistema de correos mediante una interfaz web es solo una parte del sistema completo, la cual sin embargo puede requerir bastantes recursos informáticos. Por supuesto, su implementación debe cumplir los más altos estándares de seguridad, confiabilidad y usabilidad.

Es a este nivel que se propuesto la utilización de la alternativa ExpressoV3. De esta manera, podemos decir que es en este nivel que basaremos propiamente la evaluación. Dicha evaluación se realizará utilizando criterios de Seguridad Informática, Fortaleza Técnica, Eficiencia de Desarrollo, Usabilidad y Formación de Comunidad.

La Seguridad Informática es un elemento clave que está determinado por un adecuado diseño interno del software y el cuidado con el que maneja recursos de acceso privilegiado. El empleo de paradigmas consolidados como Modelo-Vista-Controlador y una organización robusta del código fuente también influyen en este criterio. Un indicador de la seguridad informática también es el historial de vulnerabilidades conocidas que pueda tener el software.

En cuanto a la Fortaleza Técnica, ella también está determinada por el diseño del software junto a la selección de lenguajes de programación, el desempeño y eficiencia en uso de los recursos de hardware. Una amplia y apropiada documentación para desarrollo, administración y uso son también componentes importantes de este criterio.

En la Eficiencia de Desarrollo se evalúa lo concerniente a la complejidad técnica de la incorporación de nuevas funcionalidades o la realización de mejoras. Ella está determinada por las herramientas de desarrollo, la cantidad de conocimientos y experiencia necesarios para participar efectivamente en ese desarrollo. La complejidad intrínseca en la arquitectura del software es determinante aquí.

La Usabilidad por su parte evalúa los conceptos o metáforas empleadas en la interfaz con el usuario en lo concerniente a facilitar la interacción con la aplicación mediante un diseño amigable e intuitivo.

La Formación de Comunidad por su lado es uno de los criterios más importantes en lo concerniente a tecnologías libres. Consiste en la valoración de las condiciones y contexto en el que esté planteada la dinámica de desarrollo del software y su coherencia con los paradigmas del conocimiento como bien público y la no neutralidad de la tecnología. Al tener como prioridad el fortalecimiento de la soberanía tecnológica, para este criterio es importante la cercanía ideológica de los equipos de desarrollo involucrados con la aplicación con estos paradigmas, manifestada en la selección de licencias, el tipo de planteamientos a los que se da importancia en sus publicaciones más relevantes (páginas web, artículos u otros documentos), la búsqueda de la excelencia sin menoscabo de los fines sociales y demás elementos de contexto.

Teniendo en cuenta estos criterios podemos realizar un cuadro comparativo entre distintas herramientas de interfaz que se puede resumir en lo siguiente:

Criterio / Herramientas	Seguridad	Fortaleza Técnica	Eficiencia de Desarrollo	Usabilidad	Formación de Comunidad
Nivel de Interfaz					
Sogo	Alta	Alta	Media	Media	Media
Expresso	Media	Media	Media	Media	Alta
Round Cube	Media	Media	Alta	Alta	Alta
Horde Project	Media	Media	Media	Media	Media

Al comparar el Expresso con otras alternativas vemos lo siguiente:

- En los campos de “Seguridad” y “Fortaleza Técnica” aparece sólo rebasado por Sogo que es un proyecto mucho más reciente lo cual le brinda ciertas ventajas, pero que por ser un proyecto más reciente no queda bien en otros criterios.

- En los campos de “Eficiencia de Desarrollo” y “Usabilidad” Expresso aparece solo rebasado por Round Cube que es uno de los proyectos más consolidados en el campo de Software Libre para Correo Electrónico y herramientas organizativas.En el campo de “Formación de Comunidad”, Expresso aparece punteando junto a Round Cube dado que en ambos casos se ha dado un proceso de conformación de comunidad interesante. En el caso de Round Cube es un proyecto colaborativo mundial. En el caso de Expresso, de manera distinta, la comunidad se está conformando más hacia Suramérica y, en particular, Brasil. Es también importante decir que Expresso proviene del proyecto eGroupWare y que este proyecto en los actuales momentos no cuenta con una nutrida comunidad de desarrolladores. Se estima que esta comunidad (la de eGroupWare) es de solo 9 desarrolladores. En el repositorio de ExpressoLivre aparecen 6 desarrolladores inscritos.

En este sentido vemos que las dos mejores opciones aparecen como RoundCube y ExpressoV3.

Ahora bien, es importante tomar en cuenta que la opción ExpressoV3 recomienda utilizar en la capa de manejo de protocolos SMTP, POP e IMAP el software **Postfix** en vez de la opción de **Exim** que es la recomendada en este informe. Nuestra recomendación de **Exim** no se basa exclusivamente en los criterios técnicos sino principalmente en los criterios legales.

Postfix es un software liberado bajo licencia **IBM Public License** la cual no es considerada como libre por varios autores[1]. Algunas de las consideraciones que creemos conveniente revisar se presentan en la tabla siguiente.

1 Ver Guerra, Nayareth (2013) Licencias Incompatibles de Software Libre. *Revista Chilena de Derecho y Tecnología.* Vol. 2, N° 1**.** Santiago.

Datos generales	Derechos que otorga	Rasgos distintivos	Características de la cesión de la licencia	Obligaciones de la licencia	Compatibilidad con las licencias GPL
.-Licencia IBM Public License 1.0 (IPL 1.0) .-Creada por la compañía Internacional Business Machines (IBM) .- Aprobada por la Open Source Inniciative (OSI). .- De carácter más restrictivo en relación a la GPL	.- Ejecutar y reproducir libremente el software Modificar libremente el programa. .- Crear cualquier tipo de desarrollo e integración derivada del mismo. .- Distribuir libremente el software licenciado, a través de cualquier modalidad, soporte o formato. .- Poner el software a disposición de una pluralidad de personas sin previa distribución de ejemplares físicos a cada una de ellas sublicenciar libremente el software en código fuente o código objeto (binario).	.- Licencia con copyleft débil mixto. .- Sus destinatarios son usuarios y desarrolladores. .- Establece la ausencia de garantía del software en relación a su calidad y rendimiento, salvo que se establezca expresamente en la licencia. .- Establece un régimen de exclusión de responsabilidad tanto del autor de una modificación del software como del destinatario del mismo, por cualquier daño que el software cause, salvo en caso de que se establezca expresamente en la licencia. .- El Licenciatario que distribuya el software comercialmente podrá introducir en la licencia cláusulas de responsabilidad, que únicamente le afecten a él mismo y no a anteriores Licenciantes. .- Se aplica expresamente la legislación del estado de Nueva York y la legislación de copyright de Estados Unidos de América.	.-El Licenciante puede licenciar a diferentes usuarios. .- De ámbito mundial .- Gratuita	.- Distribución del software a través del código objeto: respetando las condiciones de la IPL e incluyendo cláusulas de exclusión de responsabilidad del licenciatario y de ausencia de garantía del software, además de informar a los destinatarios de la distribución de la forma de obtener el código fuente. .- Distribución del software a través del código fuente: bajo las condiciones de la licencia IPL y adjuntando una copia de la misma. .- Debe mantenerse el aviso o nota de copyright de IBM en cada copia del software distribuida.	.- La licencia IPL 1.O es incompatible con las licencias GPL. .- La IPL concede derechos necesarios para la utilización de una patente que se derive de un software solo para determinadas combinaciones de éste. .- La GPL permite la libre utilización de cualquier patente derivada de un software sujeto a la misma)*.

Consideraciones finales de la licencia IPL 1.0

1. Posee una naturaleza bastante comercial y ello puede notarse cuando asume términos como distribuidor o colaborador comercial. Haciendo una pequeña comparación con lo establecido en GPL V2, ésta permite al licenciante recibir una contraprestación económica por la distribución de copias, pero en IPL puede tratarse de individuos o empresas que comercializan con otros productos y quieren usar el software licenciado como parte de un producto para la venta.
2. Hace referencia a la reserva total de derechos cuando indica que el colaborador debe colocar la siguiente frase: Copyright (c) 1997,1998,1999, International Business Machines Corporation and others. All Rights Reserved (http://linorg.usp.br/postfix/release/LICENSE). Ello es mi criterio vulnera nuestro enfoque de conocimiento libre, pues establece la necesaria subordinación a los parámetros que IBM posee o puede generar a futuro.
3. Posee una cláusula de jurisdicción asociada a las leyes de Nueva York y de Propiedad Intelectual de los Estados Unidos, establece un plazo de un año para accionar y además la renuncia explícita a una modalidad de juicio: "This Agreement is governed by the laws of the State of New York and the intellectual property laws of the United States of America. No party to this Agreement will bring a legal action under this Agreement more than one year after the cause of action arose. Each party waives its rights to a jury trial in any resulting litigation" (http://linorg.usp.br/postfix/release/LICENSE).

* GNU en un trabajo sobre licencias realizado destaca muy brevemente que esta licencia es incompatible con GPL por la cláusula legal que posee "This is a free software license. Unfortunately, it has a choice of law clause which makes it incompatible with the GNU GPL"
(http://www.gnu.org/licenses/license-list.es.html#IBMPL)

Recomendamos evaluar la posibilidad de utilizar el Expresso V3 con Exim y no con Postfix.

Por último es importante hacer mención a algunos asuntos relacionados con el proyecto eGroupWare del cual deriva ExpressoV3. Expresso es una modificación de eGroupware realizada por una empresa brasilera. En la revisión del código fuente se observa que esa tiene un alcance relativamente pequeño. Sin embargo, se observa que uno de los componentes agregados está orientados a generación o almacenamientos de certificados digitales de usuarios del lado del servidor, lo cual implica riesgos de seguridad muy grandes que no deben ser ignorados. Ese componente no debe estar presente si se desea seguridad informática. Por otro lado, en las modificaciones a eGroupware fueron incorporados componentes en lenguaje Java, lo cual también introduce riesgos significativos debido al demostrado comportamiento agresivo de la corporación Oracle, la cual controla la propiedad intelectual e industrial de ese lenguaje y tecnologías asociadas. Historial de seguridad con un número importante de vulnerabilidades: http://www.cvedetails.com/vulnerability-list/vendor_id-2373/Egroupware.html

Sistema piloto para el corto plazo

Especificaciones de demanda:

- 50.000 usuarias o usuarios, conexiones de 1024Kbps.
- 7 GB espacio almacenamiento para cada cuenta.
- Nivel de concurrencia: 80% de usuarios conectados, 10% con dos sesiones o más. 50.000 sesiones abiertas.
- Nivel de concurrencia: 5 peticiones/min/usuaria(o). 4166,66 peticiones/seg.

Estimación de requerimientos base:

- Sistema de almacenamiento de datos en red con capacidad neta de 350 TB
- Ancho de banda requerido: 4266,66 Mbps (533.33 MB/s)
- Ancho de banda de E/S del sistema de almacenamiento en red: 5000 Mbps (625 MB/s)
- Capacidad en operaciones de E/S acceso aleatorio (4k /transacción) del sistema de almacenamiento en red, a razón de 10 operaciones E/S al almacén de correo por petición: 41666,66 IOPS

Implementación de sistema de almacenamiento de datos::

Depende en gran medida de los servidores disponibles. Una arquitectura podría ser la siguiente:

- Gabinetes de discos SAS, para unidades de 3.5 pulgadas.

- Organizados como arreglos RAID 6+0 con grupos de 10 unidades en modo RAID6 agregados en modo RAID 0.
- Con unidades de 4TB serían en total 110 discos y capacidad neta de 352TB.
- Uso de 10 unidades de estado sólido de 400GB como cache de lectura-escritura del arreglo principal.
- Por lo menos dos servidores para implementar en software los arreglos RAID cada uno con: 4 CPU de 16 núcleos, 2.5GHz. 512GB de RAM.
- Exportación de segmentos del almacenamiento de datos mediante protocolo iSCSI.
- Conexión de los servidores controladores de almacenamiento con la red de alta velocidad mediante enlaces 10GB Ethernet, agregados mediante bonding.

Estimación de requerimientos de cómputo y memoria:

- Para cada dominio a atender, un servidor virtual para controlar el almacenamiento de los buzones e implementar protocolo IMAP. La dotación puede ser a razón de un cpu virtuales de 2.5GHz y 2GB de RAM para 500 usuarios, con un mínimo de 256MB de RAM y un cpu de 2.5GHz.
- Para cada dominio a atender, un servidor virtual para hospedar el sistema LDAP. La dotación puede ser a razón de un cpu virtuales de 2.5GHz y 1GB de RAM para 1000 usuarios, con un mínimo de 256MB de RAM y un cpu de 2.5GHz.
- Para cada dominio a atender, por lo menos dos servidores virtuales preferiblemente con exim4, para manejar el protocolo SMTP. Autenticación mediante un servicio LDAP. La dotación puede ser a razón de 1 cpu de 2.5GHz y 1GB de RAM para 400 usuarios, con un mínimo de 256MB de RAM y un cpu de 2.5GHz.
- En el caso de aplicación web basada en un software desarrollado en PHP, posiblemente con motor de aplicación supervisado por uwsgi:
    - Para cada dominio a atender, por lo menos dos servidores virtuales para hospedar la ejecución de la aplicación web. La dotación puede ser a razón de 10 núcleos de 2.5GHz y 6GB de RAM para 400 usuarios, con un mínimo de 256MB de RAM y un núcleo de 2.5GHz.
- Para cada dominio a atender, por lo menos dos servidores virtuales para hospedar la ejecución de la capa http (nginx). La dotación puede ser a razón de 1 núcleo de 2.5GHz y 256MB de RAM para 2000 usuarios.
- Para cada dominio a atender, por lo menos dos servidores virtuales para hospedar la ejecución de la capa proxy (nginx). La dotación puede ser a razón de 1 núcleo de 2.5GHz y 256MB de RAM para 1000 usuarios.

La virtualización de servidores será realizada mediante implementación del hipervisor de software libre Xen.

Los servidores físicos a utilizar deberán tener la mayor dotación de CPU y memoria RAM disponible. Por ejemplo, servidores de hojilla con 64 núcleos de 2.5GHz y 256GB de RAM.

El almacenamiento de datos local solo sería usado para el sistema operativo. El almacenamiento para los servidores virtuales provendrá del sistema de almacenamiento de datos en red.

Para el hospedaje de las máquinas virtuales para 50000 usuarios se estiman los siguientes requerimientos de cómputo y memoria:

proxy: 50k/1k * (1 núcleo + 256MB) = 50 núcleos + 12,50 GB
http: 50k/2k * (1 núcleo + 256MB) = 25 núcleos + 6,25 GB
app (php): 50k/400 * (10 núcleos + 6GB) = 1250 núcleos + 750,00 GB
exim: 50k/400 * (1 núcleo + 1GB) = 125 núcleos + 125,00 GB
dovecot:50k/500 * (1 núcleo + 2GB) = 100 núcleos + 200,00 GB
ldap: 50k/1k * (1 núcleo + 256MB) = 50 núcleos + 12,50 GB

Total (con php): 1600 núcleos + 1106,25 GB

Agregando un porcentaje para reserva y redundancia, se puede estimar un requerimiento de 2048 núcleos de 2,5GHz y 1280GB de RAM. **Es importante mencionar que estas estimaciones no incluyen los servidores del sistema de almacenamiento de datos, los balanceadores de carga, servidores para DNS, cortafuegos, y otros componentes más externos del sistema**.