



DIPLOMADO EN LINUX

NIVEL 4. ADMINISTRACIÓN DE REDES LINUX

**CREAD JOSÉ ACEVEDO Y GÓMEZ
Miguel Ángel López Cacho, Juan Carlos Reyes García
BOGOTÁ D.C. 2002**

ADMINISTRACIÓN DE REDES LINUX

1. INTRODUCCIÓN

Los conceptos básicos son siempre un elemento fundamental en el entendimiento, desarrollo y aplicación de nuevas tecnologías, inclusive en procesos empíricos y no formales que a menudo son los que orientan para dar un satisfactorio final a determinado propósito.

A continuación presentaremos una serie de conceptos básicos que orientarán en los campos de configuración básica de la máquina, correo electrónico, manejo de Internet, Sendmail y los tan necesarios niveles de seguridad (Firewall).

Teniendo en cuenta que uno de los propósitos esenciales de éste texto es el de dar a conocer a la comunidad informática en general (docentes, estudiantes, personal administrativo, usuarios varios ,etc...) el exponencial crecimiento que ha tenido Linux en los campos empresariales y de la educación, lo encaminamos a ser una herramienta orientadora para aquellos que buscan una mejor capacitación.

2. OBJETIVOS

2.1 OBJETIVO GENERAL

El propósito de este nivel es dar a conocer los conceptos básicos necesarios sobre configuración básica, correo electrónico, sendmail y firewall.

2.1.1 Objetivos Específicos

Algunas configuraciones siempre se deben hacer independiente de la maquina que este en uso y la finalidad de la misma. Presentaremos donde y como insertar la información de nombre de la máquina, dominio e información de red

El correo electrónico es un método rápido de debatir ideas y encontrar soluciones para problemas; orientaremos como configurar el Sendmail para que la máquina pueda actuar como un servidor de correo electrónico.

La seguridad de una red es siempre una preocupación constante para quien la administra, presentaremos algunos conceptos relativos a protección de su red como lo son: tipos de firewall y enmascaramiento.

3. CONFIGURACIÓN BÁSICA DE LA MÁQUINA

Cada máquina *Linux* debe tener un nombre y generalmente hace parte de un dominio (un conjunto de máquinas con una administración común). Además de eso, cada computador con una tarjeta de red, pertenece a una red IP. Ésta es la sección que define la mayoría de las configuraciones necesarias para que su máquina se comunique en la red.

3.1 Consejo

Si usted:

- tiene una tarjeta de red,
- pretende colocar Linux en red con otras máquinas Unix/Linux,
- desea aprender sobre redes TCP/IP,
- y nadie más a su alrededor sabe sobre redes TCP/IP,

¡lea esta ayuda, rellene los datos de esta pantalla, y ponga su máquina en la red!

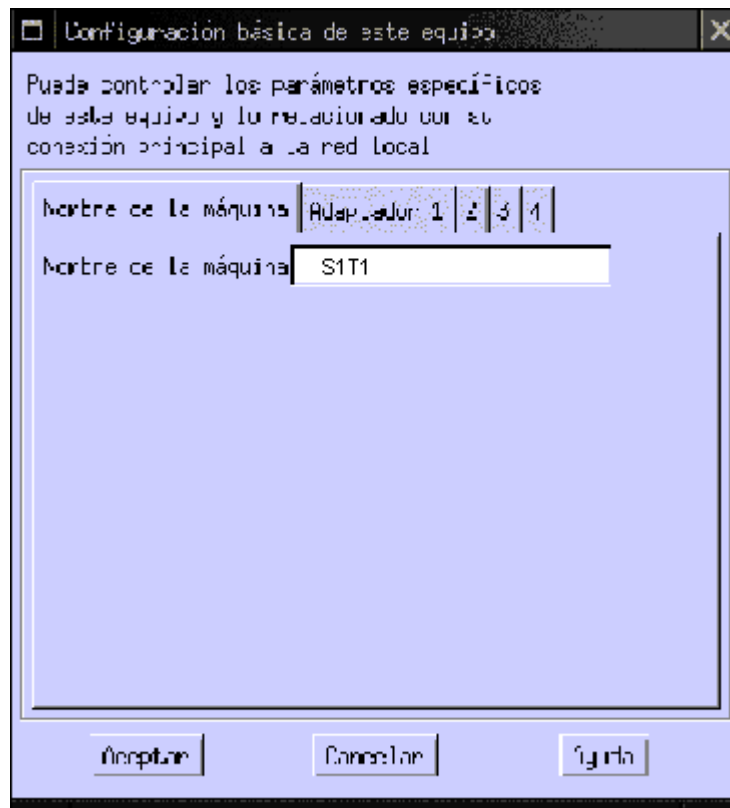


Figura: Configuración básica de la máquina

Las opciones:

Se presentan las siguientes opciones:

Primer nombre + dominio

Es el nombre de la máquina, que puede tener varios, pero hay uno que es el más importante (el oficial). Este nombre también contiene el nombre del dominio y puede estar compuesto de letras y dígitos, no habiendo la diferenciación entre letras mayúsculas y minúsculas. No es recomendable iniciar el nombre con un dígito, pues algunos programas pueden equivocadamente interpretarlo como un número IP.

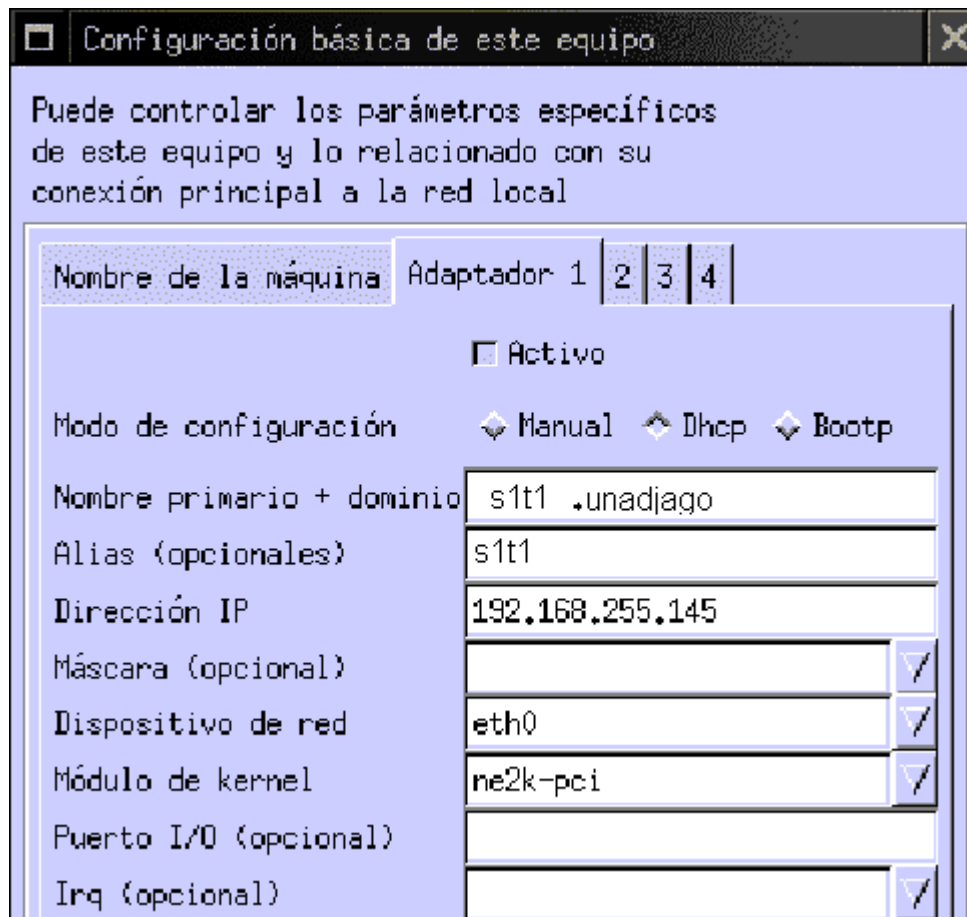


Figura: Configuración básica de la máquina – adaptador

La máquina es parte de un conjunto de máquinas con una administración común. Este nombre de dominio se usa generalmente para e-mails. Si usted no sabe qué escribir aquí, entonces use el nombre del dominio de su proveedor de acceso a Internet. Si usted no sabe qué poner aquí, coloque apenas una palabra (nombre sin dominio).

ej: mi_máquina.mi_dominio.com

3.1.1 Alias

La máquina puede tener alias para propósitos diferentes. Usted puede especificar varios, separados por espacios en blanco. El *Configurador Linux* va a adicionar automáticamente el alias ``loghost'', para que cada máquina administrada por el Configurador Linux pueda ser referenciada por él, permitiendo una mayor agilidad para crear procedimientos genéricos.

3.1.2 Dirección IP

Si su máquina está en una red (probablemente Ethernet), usted debe informar su número IP. Si actualmente no hay una red IP instalada en su red local, deje este campo vacío.

✓ Usted no tiene tarjeta de red

Usted puede definir la dirección IP de la primera tarjeta Ethernet para 127.0.0.1. De cualquier manera, el *Configurador Linux* le preguntará esto en caso de que usted deje este campo vacío.

Escogiendo ese número, usted podrá operar normalmente en ambiente de red aunque no tenga tarjeta de red. Usted puede querer hacerlo por las siguientes razones:

Quiere aprender sobre redes. *Linux* tiene un ambiente de red muy general y poderoso, pudiendo ser su propio servidor (usted puede hacer un telnet para su propia máquina, montar su propio disco como si él fuera un volumen NFS y así sucesivamente).

No tiene una tarjeta de red ``permanente'', pero usted hace conexiones PPP o SLIP.

✓ Usted tiene una tarjeta PCMCIA

Usted puede definir su red con el mismo número IP usado por la tarjeta Ethernet PCMCIA. Simplemente seleccione dummy0 (primer dispositivo simulado) en vez de eth0 como el

dispositivo de red. Así se puede tener un número IP inclusive cuando no está conectado a la red (el dispositivo PCMCIA está desconectado).

✓ **Usted no tiene ni idea**

Si usted está definiendo una red local aislada nueva (no conectada a *Internet*), la red IP 192.168.1.0 es una buena elección, pues es un número de red especialmente reservado para redes privadas. Así, si su red se conecta más tarde a *Internet* (a través de un firewall por ejemplo), su número de red no entrará en conflicto con una red ya existente, entonces el rutador no se confundirá.

Escoja un número IP para su máquina entre 192.168.1.1 y 192.168.1.254. La dirección de red será 192.168.1.0.

✓ **Dirección de red**

Su número IP puede ser dividido en dos componentes: la parte de la red y la identificación de la máquina. Por ejemplo, una máquina con el número IP 192.168.1.10 pertenece a la red 192.168.1.0 y tiene la identificación 10. Es una práctica común expresar una dirección de red usando cuatro números, donde el que corresponde a la identificación de la máquina es 0. Esa situación sólo se alterará en caso de que haya una subred en su red local.

✓ **Información adicional**

El Configurador Linux calcula muchos de los parámetros además del número IP que usted colocó. En algunas redes (que usan subredes), usted tiene que suministrar más informaciones. Generalmente solamente es necesaria la máscara. Aquí le presentamos una lista que muestra lo que el Configurador Linux puede calcular:

Usted suministra apenas el número IP

El Configurador Linux calcula la máscara, el número de la red y la dirección de broadcast.

Usted suministra el número IP y la máscara

El Configurador Linux calcula el número de la red y la dirección de broadcast.

Usted suministra el número IP, la máscara y el número de la red

El Configurador Linux calcula la dirección de broadcast.

✓ Máscara

A menos que tenga subredes, puede dejar este campo vacío. Su administrador le informará una secuencia mágica como 255.255.255.192 en caso de que haya subredes en su red local.

Una lista de ayuda está disponible en este campo, listando las máscaras de red más usadas. Esta lista es útil en caso de que usted no esté muy seguro de la máscara exacta que debe usar (que le fue suministrada por el administrador por el teléfono...).

Aquí está la lista que muestra una red x.y.z y su configuración. Esta lista fue producida por el comando ``/usr/lib/linuxconf/lib/calcmask x.y.z''

Máscara	IP de la red	Empieza	Termina
255.255.255.128	x.y.z.0	x.y.z.1	x.y.z.126
	x.y.z.128	x.y.z.129	x.y.z.254
255.255.255.192	x.y.z.0	x.y.z.1	x.y.z.62
	x.y.z.64	x.y.z.65	x.y.z.126
	x.y.z.128	x.y.z.129	x.y.z.190
	x.y.z.192	x.y.z.193	x.y.z.254
255.255.255.224	x.y.z.0	x.y.z.1	x.y.z.30
	x.y.z.32	x.y.z.33	x.y.z.62
	x.y.z.64	x.y.z.65	x.y.z.94
	x.y.z.96	x.y.z.97	x.y.z.126
	x.y.z.128	x.y.z.129	x.y.z.158
	x.y.z.160	x.y.z.161	x.y.z.190
	x.y.z.192	x.y.z.193	x.y.z.222
	x.y.z.224	x.y.z.225	x.y.z.254
255.255.255.240	x.y.z.0	x.y.z.1	x.y.z.14
	x.y.z.16	x.y.z.17	x.y.z.30

x.y.z.32	x.y.z.33	x.y.z.46
x.y.z.48	x.y.z.49	x.y.z.62
x.y.z.64	x.y.z.65	x.y.z.78
x.y.z.80	x.y.z.81	x.y.z.94
x.y.z.96	x.y.z.97	x.y.z.110
x.y.z.112	x.y.z.113	x.y.z.126
x.y.z.128	x.y.z.129	x.y.z.142
x.y.z.144	x.y.z.145	x.y.z.158
x.y.z.160	x.y.z.161	x.y.z.174
x.y.z.176	x.y.z.177	x.y.z.190
x.y.z.192	x.y.z.193	x.y.z.206
x.y.z.208	x.y.z.209	x.y.z.222
x.y.z.224	x.y.z.225	x.y.z.238
x.y.z.240	x.y.z.241	x.y.z.254

255.255.255.248

x.y.z.0	x.y.z.1	x.y.z.6
x.y.z.8	x.y.z.9	x.y.z.14
x.y.z.16	x.y.z.17	x.y.z.22
x.y.z.24	x.y.z.25	x.y.z.30
x.y.z.32	x.y.z.33	x.y.z.38
x.y.z.40	x.y.z.41	x.y.z.46
x.y.z.48	x.y.z.49	x.y.z.54
x.y.z.56	x.y.z.57	x.y.z.62
x.y.z.64	x.y.z.65	x.y.z.70
x.y.z.72	x.y.z.73	x.y.z.78
x.y.z.80	x.y.z.81	x.y.z.86
x.y.z.88	x.y.z.89	x.y.z.94
x.y.z.96	x.y.z.97	x.y.z.102
x.y.z.104	x.y.z.105	x.y.z.110
x.y.z.112	x.y.z.113	x.y.z.118
x.y.z.120	x.y.z.121	x.y.z.126
x.y.z.128	x.y.z.129	x.y.z.134
x.y.z.136	x.y.z.137	x.y.z.142

x.y.z.144	x.y.z.145	x.y.z.150
x.y.z.152	x.y.z.153	x.y.z.158
x.y.z.160	x.y.z.161	x.y.z.166
x.y.z.168	x.y.z.169	x.y.z.174
x.y.z.176	x.y.z.177	x.y.z.182
x.y.z.184	x.y.z.185	x.y.z.190
x.y.z.192	x.y.z.193	x.y.z.198
x.y.z.200	x.y.z.201	x.y.z.206
x.y.z.208	x.y.z.209	x.y.z.214
x.y.z.216	x.y.z.217	x.y.z.222
x.y.z.224	x.y.z.225	x.y.z.230
x.y.z.232	x.y.z.233	x.y.z.238
x.y.z.240	x.y.z.241	x.y.z.246
x.y.z.248	x.y.z.249	x.y.z.254

255.255.255.252

x.y.z.0	x.y.z.1	x.y.z.2
x.y.z.4	x.y.z.5	x.y.z.6
x.y.z.8	x.y.z.9	x.y.z.10
x.y.z.12	x.y.z.13	x.y.z.14
x.y.z.16	x.y.z.17	x.y.z.18
x.y.z.20	x.y.z.21	x.y.z.22
x.y.z.24	x.y.z.25	x.y.z.26
x.y.z.28	x.y.z.29	x.y.z.30
x.y.z.32	x.y.z.33	x.y.z.34
x.y.z.36	x.y.z.37	x.y.z.38
x.y.z.40	x.y.z.41	x.y.z.42
x.y.z.44	x.y.z.45	x.y.z.46
x.y.z.48	x.y.z.49	x.y.z.50
x.y.z.52	x.y.z.53	x.y.z.54
x.y.z.56	x.y.z.57	x.y.z.58
x.y.z.60	x.y.z.61	x.y.z.62
x.y.z.64	x.y.z.65	x.y.z.66
x.y.z.68	x.y.z.69	x.y.z.70

x.y.z.72	x.y.z.73	x.y.z.74
x.y.z.76	x.y.z.77	x.y.z.78
x.y.z.80	x.y.z.81	x.y.z.82
x.y.z.84	x.y.z.85	x.y.z.86
x.y.z.88	x.y.z.89	x.y.z.90
x.y.z.92	x.y.z.93	x.y.z.94
x.y.z.96	x.y.z.97	x.y.z.98
x.y.z.100	x.y.z.101	x.y.z.102
x.y.z.104	x.y.z.105	x.y.z.106
x.y.z.108	x.y.z.109	x.y.z.110
x.y.z.112	x.y.z.113	x.y.z.114
x.y.z.116	x.y.z.117	x.y.z.118
x.y.z.120	x.y.z.121	x.y.z.122
x.y.z.124	x.y.z.125	x.y.z.126
x.y.z.128	x.y.z.129	x.y.z.130
x.y.z.132	x.y.z.133	x.y.z.134
x.y.z.136	x.y.z.137	x.y.z.138
x.y.z.140	x.y.z.141	x.y.z.142
x.y.z.144	x.y.z.145	x.y.z.146
x.y.z.148	x.y.z.149	x.y.z.150
x.y.z.152	x.y.z.153	x.y.z.154
x.y.z.156	x.y.z.157	x.y.z.158
x.y.z.160	x.y.z.161	x.y.z.162
x.y.z.164	x.y.z.165	x.y.z.166
x.y.z.168	x.y.z.169	x.y.z.170
x.y.z.172	x.y.z.173	x.y.z.174
x.y.z.176	x.y.z.177	x.y.z.178
x.y.z.180	x.y.z.181	x.y.z.182
x.y.z.184	x.y.z.185	x.y.z.186
x.y.z.188	x.y.z.189	x.y.z.190
x.y.z.192	x.y.z.193	x.y.z.194
x.y.z.196	x.y.z.197	x.y.z.198
x.y.z.200	x.y.z.201	x.y.z.202

x.y.z.204	x.y.z.205	x.y.z.206
x.y.z.208	x.y.z.209	x.y.z.210
x.y.z.212	x.y.z.213	x.y.z.214
x.y.z.216	x.y.z.217	x.y.z.218
x.y.z.220	x.y.z.221	x.y.z.222
x.y.z.224	x.y.z.225	x.y.z.226
x.y.z.228	x.y.z.229	x.y.z.230
x.y.z.232	x.y.z.233	x.y.z.234
x.y.z.236	x.y.z.237	x.y.z.238
x.y.z.240	x.y.z.241	x.y.z.242
x.y.z.244	x.y.z.245	x.y.z.246
x.y.z.248	x.y.z.249	x.y.z.250
x.y.z.252	x.y.z.253	x.y.z.254

✓ **Dirección de red**

Usted puede colocar la dirección de red directamente. El Configurador Linux la calculará correctamente a partir de la máscara, pero para configuraciones especiales, puede sobrescribir ese valor calculado por el Configurador Linux.

✓ **Broadcast**

La dirección de broadcast también se calcula a partir de la máscara. En algunas redes, la dirección de broadcast es configurada de forma diferente, entonces usted tendrá que informarla manualmente.

✓ **Dispositivo de red**

Cada configuración de adaptador está asociada a un dispositivo de red específico en el kernel. Por ejemplo, el eth0 corresponde a la primera tarjeta de red Ethernet, y el eth1 a la segunda. En una red token ring, o tr0 corresponde al primer adaptador token ring. Este campo lo deja escoger el dispositivo apropiado, y hay una lista de ayuda que le muestra los que hay disponibles.

✓ **Configuración del módulo del kernel**

Usted puede manipular el controlador que será cargado para cada adaptador.

✓ **Módulo del kernel**

Cada dispositivo de red está asociado a un módulo del kernel. Este módulo o es construido en el kernel o debe cargarse en tiempo de ejecución. En caso de que sea necesario el Configurador Linux lo cargará, pero usted tiene que informarle qué tipo de adaptador de red tiene usted. Use la lista de ayuda para encontrar los nombres de los diversos módulos del kernel disponibles en Linux. Los módulos con el prefijo ``(inst)" están instalados corrientemente en su máquina. Si el módulo que usted necesita no está instalado, tendrá que recompilar el kernel y seleccionar (usando el ``make menuconfig") el módulo del kernel apropiado.

✓ **Puerta E/S**

Muchos módulos del kernel tienen la habilidad de autodetectar su configuración, especialmente los PCI. Algunos necesitan que usted les informe su configuración básica, que es la dirección de E/S (i/o) y la IRQ (interrupción). Especialmente para tarjetas NE2000 (desde Linux 2.0, las tarjetas NE2000 PCI son autodetectables, pero antes del 2.0.30, usted tenía que informar el E/S y la IRQ aunque la tarjeta fuera PCI). Usted colocará la dirección de E/S usando el formato 0xNNN donde NNN es un número hexadecimal, como lo presenta el utilitario de configuración de la tarjeta.

✓ **Irq**

Como en el E/S, ocasionalmente usted necesitará informar un valor para la IRQ, en caso que el controlador no pueda detectarlo (especialmente la NE2000). Puede usar la lista de ayuda para escoger la IRQ correcta, y ver cuáles ya están reservadas para los dispositivos de su computador.

✓ **Un módulo del kernel soportando varios adaptadores**

Cuando usted tiene varias tarjetas de red en la misma máquina, debe informar la puerta de E/S y la IRQ para todas ellas de una sola vez, en la misma línea (separado por comas) en la configuración del primer adaptador.

Puerta E/S (opc.) : 0x300,0x320

Irq (opc.) : 10,11

✓ **Varios adaptadores**

Un PC *Linux* puede tener más de cuatro tarjetas Ethernet (puede tener más, pero éste es el límite configurado en el kernel).

Cada tarjeta Ethernet debe tener un número IP y opcionalmente, un nombre.

4. CORREO ELECTRÓNICO

4.1 Introducción

Los modernos sistemas operativos tipo son muy buenos a la hora de hablar con otras máquinas y funcionar en red. Dos máquinas diferentes pueden intercambiar información de muchas, muchas maneras distintas.

Uno de los usos más comunes de las redes informáticas desde sus orígenes ha sido el correo electrónico. Empezó siendo un simple servicio que copiaba un fichero de una máquina a otra, y lo añadía al fichero *mailbox* (buzón de correo) del destinatario. Básicamente, en esto sigue consistiendo el *email* (correo electrónico), aunque una red en continuo crecimiento con sus complejas necesidades de encaminado y su volumen de mensajes siempre en aumento ha hecho necesario un esquema más elaborado.

Una gran cantidad de programas para transporte de correo han sido implementados para sistemas. Uno de los más conocidos es el *sendmail*, de la Universidad de *Berkeley*, que se usa en diversas plataformas. El autor original fue *Eric Allman*, que está trabajando activamente en el equipo *sendmail* de nuevo. Hay dos adaptaciones para del *sendmail-5.56c* disponibles. La versión de *sendmail* actualmente en desarrollo es la 8.6.5.

El agente de correo de uso más común con es el *smail-3.1.28*, escrito por *Curt Landon Noll* y *Ronald S. Karr* y con *Copyright* de los mismos autores. Éste es el que se incluye en la mayoría de las distribuciones de Linux. En lo sucesivo nos referiremos a él simplemente como *smail*, aunque hay otras versiones del mismo programa que son totalmente diferentes, y que no describiremos aquí.

Comparado con *sendmail*, *smail* es bastante joven aún. Si se ocupan del correo de un nodo pequeño sin necesidad de direccionamiento complicadas, sus capacidades son muy

parecidas. Para nodos grandes, sin embargo, *sendmail* siempre gana, porque su método de configuración es mucho más flexible.

Ambos *smail* y *sendmail* admiten un conjunto de ficheros de configuración que deben ser adaptados a cada caso particular. Aparte de la información que se necesita para hacer funcionar el subsistema de correo (como puede ser el nombre del ordenador local), hay muchos más parámetros que pueden ajustarse. El fichero principal de configuración de *sendmail* es muy difícil de entender la primera vez. Los ficheros de configuración de *smail* están más estructurados y son más fáciles de entender que los del *sendmail*, pero no dan al usuario tanto poder a la hora de ajustar el comportamiento del gestor de correo. De todos modos, para nodos pequeños de UUCP o Internet, el trabajo que se necesita para poner a punto cualquiera de ellos es prácticamente el mismo.

En esta guía trataremos sobre qué es el *e-mail* y qué temas tendrá que abordar usted como administrador del sistema. La información que se suministra debe bastar para poner en marcha pequeños nodos, pero hay muchas más opciones y usted podrá pasar muchas horas felices frente a su ordenador configurando las características más superficiales.

4.2 Definición de mensaje de correo

Un mensaje de correo consta de un contenido (*body*), que es el texto que ha escrito el remitente, y datos especiales que especifican el destinatario o destinatarios, el medio de transporte, etc., de manera similar a lo que aparece en el sobre de una carta ordinaria. Estos datos administrativos se clasifican en dos categorías; en la primera categoría están los datos que son específicos del medio de transporte, como son las direcciones del remitente y del destinatario. A esto se le llama *el sobre (envelope)*. Puede ser modificado por el software de transporte a medida que el mensaje es transmitido.

La segunda variedad es cualquier dato necesario para la manipulación del mensaje, que no es propio de ningún mecanismo de transporte, como es la línea del encabezado en la que indicamos el tema del mensaje (*Subject*), la lista de todos los destinatarios, y la fecha

en la que se envió el mensaje. En muchas redes, se ha convertido en un estándar incluir estos datos al comienzo del mensaje, formando lo que se denomina encabezado del mensaje (*mail header*). Se separa del contenido del mensaje (*mail body*) por una línea en blanco.

En todos esos estándares, el encabezado consiste en varias líneas, separadas por caracteres de retorno de carro. Cada línea consiste en un nombre de campo, que comienza en la columna uno, y el campo en sí, separados por dos puntos (:) o un espacio. El formato y la semántica de cada campo varía dependiendo del nombre del mismo. Un campo del encabezado se puede continuar más allá de una línea, si la línea siguiente comienza con un TAB. Los campos pueden aparecer en cualquier orden.

Usualmente, todos los campos del encabezado necesarios son generados por la interfaz con el servidor de correo que usted use, como *elm*, *pine*, *mush*, o *mailx*. Algunos, sin embargo, son opcionales, y pueden ser añadidos por el usuario. *elm*, por ejemplo, permite editar parte del encabezado del mensaje. Otros campos son añadidos por el software de transporte de correo. Una lista de campos de encabezado comunes y su significado se da a continuación:

✓ ***From:***

Contiene la dirección de correo electrónico del remitente, y posiblemente el "nombre real". Un verdadero zoológico de formatos distintos se usa aquí.

✓ ***To:***

Ésta es la dirección de email del destinatario.

✓ ***Subject:***

Describe el contenido del mensaje en pocas palabras. Al menos eso es lo que *debería* hacer.

✓ ***Date:***

La fecha en la que el mensaje fue enviado.

✓ **Reply-To:**

Especifica la dirección a la que el remitente desea que el destinatario le conteste. Esto puede ser útil si se tienen varias direcciones, pero si desea recibir la mayor parte del correo sólo en aquélla que se usa más a menudo. Este campo es opcional.

✓ **Organization:**

La organización que posee la máquina desde la que se ha enviado el mensaje. Si la máquina usada es la suya propia no incluya este campo, o bien indique ``privado" o cualquier trivialidad sin sentido. Este campo es opcional.

✓ **Message-ID:**

Una cadena generada por el transporte de correo en el sistema remitente. Es única para cada mensaje.

✓ **Received:**

Cada nodo que procesa su correo (incluyendo las máquinas del remitente y el destinatario) inserta este campo en el encabezado, dando el nombre del nodo, una identificación de mensaje, hora y fecha a la que lo recibieron, de qué nodo procede, y qué software de transporte ha sido usado. Esto se hace así para que usted pueda conocer la ruta que su mensaje ha seguido, y pueda protestar a la persona responsable si algo ha ido mal.

✓ **X-cualquier-cosa:**

Ningún programa relacionado con el correo debe protestar sobre cualquier encabezado que comience con X-. Esto se usa para implementar características adicionales que aún no han sido incluidas en un RFC, o que no lo serán nunca. Esto se usa, por ejemplo, en la lista de correo de los Activistas de Linux, donde el canal a usar se selecciona con el campo de encabezado *X-Mn-Key*..

4.2.1 Distribución del correo electrónico.

Generalmente, usted escribirá su correo usando una interfaz de correo como mail o *mailx*; u otros más sofisticados como *elm*, *mush*, o *pine*. Estos programas se denominan agentes de usuario de correo (*mail user agents*), o MUAs para abreviar. Si usted envía un mensaje de correo, el programa interface en la mayoría de los casos se lo pasará a otro programa para que lo transmita. Este programa se denomina el agente de transporte de correo (*mail transport agent*), o MTA. En algunos sistemas hay agentes de transporte de correo distintos para envíos locales o lejanos; en otros hay sólo un MTA. El comando para envíos lejanos se denomina usualmente *rmail*, el otro se denomina *lmail*, si existe.

Un envío local de correo es, algo más que añadir el mensaje al buzón del destinatario. Usualmente el MTA local entenderá como usar alias (definir direcciones locales de destinatarios que dirigen a otras direcciones), y como usar redirecciones (en inglés *forwarding*, dirigir el correo de un usuario a otra dirección). También, los mensajes que no pudieron ser enviados deben ser usualmente devueltos (*bounced*), al remitente junto con algún mensaje de error.

Para envíos lejanos, el software de transporte usado depende del tipo de enlace. Si el correo debe enviarse a través de una red que usa TCP/IP, se usará normalmente SMTP. SMTP son las siglas de *Simple Mail Transfer Protocol*, o Protocolo Simple de Transferencia de Correo que se define en el RFC 788 y RFC 821. SMTP usualmente conecta con la máquina del destinatario directamente, negociando la transferencia del mensaje con el demonio SMTP del otro lado.

4.2.2 Direcciones de correo electrónico

Para el correo electrónico, una dirección consiste en, al menos, el nombre de la máquina que maneja el correo del destinatario, y una identificación de usuario reconocida por ese sistema. Puede ser el nombre de acceso del destinatario, pero puede ser también cualquier otra cosa. Otros esquemas de direcciones, como el X.400, usan un conjunto más general de "atributos" que se utilizan para buscar la máquina del destinatario en un servidor de directorio X.500.

La forma en que se interpreta un nombre de máquina, es decir, a qué nodo va a llegar finalmente nuestro mensaje, y cómo combinar este nombre con el nombre de usuario del destinatario depende enormemente de la red en la que nos encontremos.

Los nodos en la Internet siguen el estándar RFC 822, que requiere una notación usuario@máquina.dominio, donde máquina.dominio es el nombre de dominio totalmente cualificado (*Fully Qualified Domain Name, o FQDN*) de la máquina. El objeto que aparece entre medias se denomina signo "at". (N. del T. de la preposición inglesa "at", que significa "en"). Dado que esta notación no indica la ruta hasta la máquina de destino, sino que da el nombre (único) de dicha máquina, a esto se le suele llamar una dirección absoluta.

Estos dos tipos de direcciones no se mezclan muy bien. Supongamos una dirección máquinaA!usuario@máquinaB. No queda claro si el signo '@' tiene precedencia sobre la ruta o viceversa: Hemos de enviar el mensaje a máquinaB, que lo enviará a máquinaA!usuario, o debe ser enviado máquinaA, que lo redirigirá a usuario@máquinaB? Las direcciones que mezclan diferentes tipos de operadores de dirección se denominan direcciones híbridas. El más notorio es el ejemplo anterior. Se resuelve usualmente dándole precedencia al signo '@' sobre la ruta. En el ejemplo anterior, esto significa enviar el mensaje a máquinaB primero.

4.2.3 Encaminado de correo electrónico en Internet .

En la Internet, depende enteramente del nodo de destino que se realice algún encaminado específico de correo. El comportamiento por defecto consiste en enviar el

mensaje al nodo de destino buscando su dirección IP, y dejando el encaminado en sí de los datos a la capa IP de transporte.

La mayoría de los nodos usualmente querrán que todo el correo entrante se dirija a un servidor de correo fácilmente accesible que sea capaz de procesar todo ese tráfico, y que distribuirá ese correo localmente. Para anunciar ese servicio, el nodo publica el llamado campo MX para su dominio local en la base de datos DNS. MX significa *Mail Exchanger* (*Intercambiador de correo*) y básicamente quiere decir que el servidor va a actuar como un redistribuidor de correo para todas las máquinas de este dominio. Los campos MX también pueden usarse para manipular el tráfico dirigido a máquinas que no están ellas mismas conectadas a la Internet, como redes UUCP o redes corporativas que contienen información confidencial.

Los campos MX también tienen una preferencia asociada. Es un entero positivo. Si existen varios intercambiadores de correo para una máquina, el agente de transporte de correo intentará enviar el mensaje al intercambiador con menor valor de preferencia, y sólo si este falla probará uno con mayor valor. Si el nodo local es él mismo un intercambiador de correo para la dirección de destino, debe no redirigir los mensajes a cualquier máquina MX que tenga un valor de preferencia mayor que el suyo propio: ésta es una forma segura de evitar bucles de correo.

4.2.4 Formatos de Fichero Mapa y Alias de Ruta

Escribir un fichero de alias de ruta es aceptable sólo cuando se está manejando un nodo que no necesita hacer demasiados encaminados. Si se tienen que hacer encaminados hacia un gran número de máquinas, una manera mejor es usar el comando *pathalias* para crear el fichero a partir de los ficheros de mapa. Los mapas se pueden mantener con más facilidad, porque sólo hay que añadir o quitar un sistema editando la entrada de dicho sistema en el mapa, y volver a crear el fichero de mapa.

Un fichero de mapa consiste principalmente en una lista de nodos, junto con los nodos que cada sistema registra o lo registran. El nombre del sistema comienza en la columna

uno, y sigue una lista de enlaces separados por comas. La lista se puede continuar en varias líneas, comenzando cada nueva línea con un tab. Cada enlace consiste en el nombre del nodo enlazado, seguido del coste, entre corchetes. El coste es una expresión aritmética, consistente en números y costes simbólicos. Las líneas que comienzan con un signo hash (#) se ignoran.

Usando la información de dichos ficheros de mapa, *pathalias* puede calcular rutas óptimas para cualquier destino que aparezca en el fichero de rutas, y producir una base de datos de alias de ruta a partir de éste, que se puede usar para encaminar hacia esos nodos.

pathalias proporciona varias posibilidades más, como esconder nodos (es decir, hacerlos accesibles sólo a través de una pasarela), etc. Para obtener más detalles, además de una lista completa de costes de enlace.

4.3 Configurando *“electronic mail”* ó *elm*

Significa correo electrónico en español y es una de las utilidades más razonablemente bautizadas. Proporciona una interfaz de pantalla completa con una buena utilidad de ayuda. No vamos a explicar aquí como se usa *elm*, sólo nos detendremos en sus opciones de configuración.

Cuando comienza, *elm* lee un conjunto de variables de configuración desde el fichero *elmrc* en */usr/lib/elm*. Entonces, intentará leer el fichero *elm/elmrc* en nuestro directorio personal. Normalmente este fichero no se genera a mano. Se crea cuando se escoge *“save options”* (grabar opciones) desde el menú de opciones de *elm*.

4.3.1 Cómo configurar y poner en marcha *smail*

En esta guía se hace una breve introducción a la forma de configurar *smail* y además dará una idea general de la funcionalidad que este programa provee. Aunque *smail* es muy similar en comportamiento a *sendmail*, sus archivos de configuración son totalmente diferentes.

El archivo de configuración principal es `/usr/lib/smail/config`. Este archivo es el que se debe editar para ajustar los valores específicos al sistema que se está configurando. Si únicamente es un ordenador terminal de UUCP, serán relativamente pocas las opciones a cambiar. Hay además otros archivos que configuran las opciones de encaminamiento y transporte que se pueden modificar; se hablará brevemente sobre la forma de hacerlo.

La forma de operación normal de *smail* hace que procese y entregue todo el correo de entrada inmediatamente. Si se tiene un tráfico relativamente alto, se puede preferir que *smail* guarde todos los mensajes en una *cola*, y los procese a intervalos regulares.

Cuando se trabaja con correo dentro de una red TCP/IP, es frecuente que *smail* funcione como demonio: en el momento de iniciar el ordenador, se invoca desde el archivo `rc.inet2`, y se coloca en segundo plano, desde donde espera que haya una conexión de entrada TCP que entre por el puerto SMTP (el puerto 25 es lo normal). Este esquema es muy bueno cuando se espera una gran cantidad de tráfico, pues *smail* no lanza por separado para cada conexión que ingresa. La alternativa es usar a *inetd* como el administrador del puerto SMTP, y lanzar una copia de *smail* cada vez que haya una conexión en este puerto.

El programa *smail* tiene muchas opciones que se usan para controlar su comportamiento; describirlas una por una en detalle no es de gran utilidad. Afortunadamente *smail* tiene varios modos estándar de operación que se habilitan cuando es invocado con un nombre específico tal como `rmail` o `smtpd`. Es común que estos nombres específicos sean enlaces simbólicos al binario de *smail*. Se verán más de éstas cuando se discutan algunas otras características de *smail*.

Hay dos enlaces a *smail* que deben estar siempre:

/usr/bin/rmail y */usr/sbin/sendmail*.

Cuando se crea y se envía un mensaje de correo con un agente de usuario tal como *elm*, el mensaje se pasará a *rmail* para su entrega, con la lista de destinatarios dada en la línea de instrucciones. Lo mismo sucede con el correo que entra vía UUCP. Algunas versiones de *rmail*, sin embargo, invocan a */usr/sbin/sendmail* en vez de a *rmail*, por lo que son necesarios ambos enlaces. Por ejemplo, si *smail* está en */usr/local/bin*, se debe escribir lo siguiente en la línea de comandos:

```
# ln -s /usr/local/bin/smail /usr/bin/rmail
```

```
# ln -s /usr/local/bin/smail /usr/sbin/sendmail
```

4.3.2 Configuración para una red local

Si está funcionando una instalación con dos o más nodos conectados por medio de una red local, es necesario designar a uno de ellos para que maneje la conexión UUCP con el mundo exterior. Entre las máquinas de la red local, es muy probable que se quiera intercambiar correo con SMTP sobre TCP/IP.

En un entorno de red, es preferible mantener todos los archivos con el correo de los usuarios en un solo sistema de archivos, que puede ser montado con NFS desde todas las demás máquinas. Esto permite a los usuarios desplazarse de máquina en máquina sin tener que mover su correo por todos lados. Así mismo, es deseable hacer que las direcciones de los usuarios sean independientes del ordenador en la cual el correo se almacena. Es una práctica común utilizar el nombre del dominio como la dirección de quien envía el correo en vez de utilizar el nombre de la máquina servidora del correo. A continuación se explicará cómo hacer que el servidor reconozca el nombre del dominio como un nombre válido para su instalación.

Otra forma de mantener todos los apartados postales en un anfitrión central es utilizar POP o IMAP. POP quiere decir, por sus siglas en inglés *Post Office Protocol*, es decir, Protocolo de Oficina Postal y permite a los usuarios tener acceso a sus archivos de correo a través de una conexión TCP/IP. IMAP, o Protocolo de Acceso Interactivo al Correo por su sigla en inglés de *Interactive Mail Access Protocol*, es similar a POP, excepto que es más general. Ambos clientes y servidores para IMAP y POP han sido transportados a Linux, y están disponibles en *sunsite.unc.edu* bajo el directorio */pub/Linux/system/Network*.

4.3.3 Cómo ejecutar *smail*

La primera cosa que se debe hacer es decidir si se ejecutará *smail* como un demonio independiente, o si se permitirá que *inetd* administre el puerto SMTP e invoque a *smail* cuando un cliente solicite una conexión SMTP. Normalmente es preferible la operación como un demonio independiente en el servidor de correo, debido a que esto carga la computadora menos que lanzar una copia nueva de *smail* cada vez que se solicita una conexión individual. Cuando un servidor de correo entrega casi todo el correo que recibe directamente a los usuarios, es preferible optar por la operación con *inetd*.

Independientemente del modo de operación que se haya elegido para cada anfitrión individual, es importante asegurarse que se tiene la siguiente línea en el archivo */etc/services*:

```
smtp          25/tcp          #Simple Mail Transfer Protocol
```

Esto define el número del puerto TCP que *smail* utilizará para las conexiones SMTP. 25 es el puerto estándar definido por el RFC de Números de Puerto Asignados.

Cuando se ejecuta como demonio, *smail* se coloca a sí mismo en segundo plano, y espera a que ocurra una conexión en el puerto SMTP. Cuando haya una conexión, lanza un proceso y conduce una conversación SMTP en dicho puerto. El demonio *smail* se lanza normalmente invocándolo desde el *script* *rc.inet2* con la siguiente instrucción:

```
/usr/local/bin/smmail -bd -q15m
```

El modificador *-bd* indica que funcionará como demonio, y *-q15m* hace que se procesen los mensajes acumulados en la cola cada 15 minutos.

Si en cambio, se quiere utilizar *inetd*, el archivo */etc/inetd.conf* deberá contener una línea como la siguiente:

```
smtp stream tcp nowait root /usr/sbin/smtpd
```

smtpd debe ser un enlace simbólico al binario de *smmail*. Recuerde que tiene que forzar a que *inetd* relea *inetd.conf* enviándole una señal HUP después de hacer estos cambios.

El modo demonio y el modo *inetd* son mutuamente excluyentes. Si ejecuta *smmail* como demonio, asegúrese de que esté comentada cualquier línea en *inetd.conf* para el servicio *smtp*. De manera similar, cuando tenga a *inetd* como administrador de *smmail*, asegúrese de que *rc.inet2* no lanza al demonio *smmail*.

4.4 Modos de entrega de correo

Como se mencionó anteriormente, *smmail* es capaz de entregar los mensajes inmediatamente o encolarlos para un proceso posterior. Si se decide encolar los mensajes, *smmail* guardará todo el correo en el directorio *messages* debajo de */var/spool/smmail*. No se procesarán hasta que se le indique explícitamente que lo haga (a este proceso se le conoce como "ejecutar la cola").

Se pueden seleccionar uno de los tres modos de entrega ajustando el atributo *delivery_mode* en el archivo *config* para que esté como *foreground*, *background*, o *queued*. Es decir, proceso normal, proceso en segundo plano, o proceso en cola. Estas opciones seleccionan la entrega normal (procesamiento inmediato de los mensajes de entrada), en segundo plano, (los mensajes son entregados por medio de un hijo del

proceso receptor.) El proceso padre muere inmediatamente después de la creación del hijo), y el encolado. El correo de entrada siempre será encolado independientemente de esta opción si la variable booleana `queue_only` está puesta en el archivo `config`.

Si se activa el modo de cola, se debe asegurar que las colas se revisen regularmente; probablemente cada 10 o 15 minutos. Si se ejecuta *smail* como demonio, se debe agregar la opción `-q10m` en la línea de comandos para procesar la cola cada 10 minutos. De forma alternativa, se puede invocar *run* desde el *cron* en esos intervalos de tiempo. *runq* deberá ser un enlace a *smail*.

Aún cuando no se utilice el modo de cola, *smail* pondrá de forma ocasional los mensajes en la cola cuando falle la entrega inmediata por una razón transitoria. Para las conexiones SMTP, esto puede ser debido a que el nodo siguiente sea un inalcanzable; pero los mensajes pueden también ser pospuestos cuando el sistema de archivos del receptor esté lleno. En cualquier caso, debe poner una cola que se revise, por ejemplo, cada hora, (utilizando *runq*), porque si no, cualquier mensaje pospuesto se quedará encolado indefinidamente.

4.4.1 Encaminamiento de mensajes y entrega

smail divide la entrega del correo en tres partes, la ruta, el módulo de entrega local y el módulo de transporte.

El módulo de encaminamiento resuelve todas las direcciones remotas, determinando el nodo al que el mensaje será enviado y el transporte que será utilizado. Dependiendo de la naturaleza del enlace, se utilizarán transportes diferentes tales como UUCP o SMTP.

Las direcciones locales se dan al módulo de entrega local que resuelve cualquier reenvío o alias. Por ejemplo, la dirección podría ser un alias o una lista de correo, o el usuario podría querer reenviar su correo a otra dirección. Si la dirección resultante es remota, se maneja de nuevo en el módulo de encaminamiento, de otra forma se asigna a un transporte para su entrega local. Normalmente, la acción a realizar será entregar a un archivo de correo, pero los mensajes también pueden ser pasados a la entrada de un

comando (por ejemplo, un filtro de correo que el usuario quiera establecer) o agregados a un archivo arbitrario cualquiera.

El módulo de transporte, finalmente, es el responsable de la entrega independientemente del método que se haya escogido. Intenta entregar el mensaje y en caso de problemas, puede devolver un mensaje al remitente o posponer la entrega para intentarlo de nuevo más tarde.

Con *smail* se tiene mucha flexibilidad para configurar estas tareas. Para cada una de ellas, hay varios controladores disponibles, de los cuales se puede elegir el más adecuado. Se debe indicar a *smail* la elección a través de los siguientes archivos: *routers*, *directors*, y *transports* que se encuentran en */usr/lib/smail*. Si estos archivos no existen, se toman valores por defecto razonables que funcionan en la mayor parte de los sistemas que utilizan SMTP o UUCP como transporte. Si se quiere cambiar la política de encaminamiento de *smail*, o modificar un transporte, es conveniente obtener los archivos ejemplo que vienen con la distribución de los programas fuente de *smail*, copiar los archivos ejemplo a */usr/lib/smail*, y modificarlos de acuerdo con sus necesidades

4.4.2 La base de datos de trayectorias *paths*

El programa *smail* espera encontrar una base de datos de alias de trayectorias en el archivo *paths* en el subdirectorio */usr/lib/smail*. Este archivo es opcional, por lo que si no se quiere hacer ningún encaminamiento por medio de alias de trayectorias, simplemente se borra el archivo *paths*.

El archivo *paths* debe ser un archivo ASCII ordenado que contenga líneas que mapeen los nombres de los nodos destino a trayectorias UUCP con signos de admiración. El archivo tiene que estar ordenado debido a que *smail* utiliza búsqueda binaria para encontrar un sitio. No se permiten comentarios en este archivo, y el nombre del sitio debe estar separado de la trayectoria utilizando un carácter de tabulación.

Si su nodo es simplemente terminal, no será necesario tener un archivo *paths*: basta con ajustar los atributos de nodo de relevo en su archivo *config*, y dejarle todo el trabajo de encaminamiento que su correo genere.

4.4.3 Cómo entregar mensajes a las direcciones locales

Es común que una dirección local de correo sea sólo el nombre del usuario, en cuyo caso el mensaje se entrega en su archivo de correo, */var/spool/mail/user*. En otros casos se incluyen alias y nombres de la lista de correo, y correo redirigido por el usuario. En estos casos, la dirección local se expande a una nueva lista de direcciones que pueden ser locales o remotas.

Independientemente de estas direcciones "normales", *smail* puede manejar otros tipos de destino para los mensajes locales tales como nombres de archivos o comandos (que reciben el mensaje por su entrada estándar). Éstas no son propiamente direcciones, de tal forma que no se puede mandar correo, por ejemplo a */etc/passwd@dominio.com*; sólo son válidas si se han tomado de un archivo alias o de redireccionamiento.

Un *nombre de archivo* es cualquier cosa que comience con una diagonal (*/*) o una tilde (*~*). El segundo se refiere al directorio inicial del usuario, y es posible sólo si el nombre del archivo ha sido tomado de *.forward* o una línea de redirección del archivo de correo (ver más abajo). Cuando *smail* manda el mensaje a un archivo, lo añade al final del archivo y, de ser necesario, lo puede también crear.

Una *instrucción por tubería* puede ser cualquier comando precedido por el símbolo (*|*). Esto hace que *smail* envíe el comando al *shell* junto con sus argumentos, pero sin el *`|* que lo encabeza; pasando el mensaje a la entrada estándar del comando.

Si la invocación contiene un espacio en blanco, se debe encerrar en comillas dobles. Debido a los problemas de seguridad que pueden ser ocasionados aquí, es importante cuidar que no se ejecute el comando si la dirección ha sido obtenida de alguna forma dudosa (por ejemplo, si el archivo de alias del cual la dirección se ha obtenido ha sido escrito por cualquiera).

4.4.4 Usuarios locales

El caso más común para una dirección local es mostrar el archivo de correo del usuario. Este apartado postal está en `/var/spool/mail` y tiene el nombre del usuario. También es propiedad de éste, con grupo mail y tiene el modo 660. Si no existe, *smail* lo crea.

Observe que aunque `/var/spool/mail` es el lugar estándar para poner los archivos de correo, algunas aplicaciones tienen diferentes trayectorias compiladas en ellos, por ejemplo `/usr/spool/mail`. Si la entrega a los usuarios en su sistema falla constantemente, se puede intentar hacer un enlace simbólico a `/var/spool/mail` para ver si esta situación mejora.

Hay dos direcciones que *smail* necesita para funcionar: *MAILER-DAEMON* y *Postmaster*. Cuando se devuelve un mensaje de informe debido a un correo que no pudo ser entregado, se envía una copia a la cuenta del administrador postal (*el postmaster*) para su revisión (en el caso de que este mensaje se deba a un problema de configuración). El usuario *MAILER-DAEMON* se utiliza como la dirección del remitente del mensaje devuelto.

Si estas direcciones no tienen nombres de cuentas válidas en su sistema, *smail* mapea implícitamente *MAILER-DAEMON* a *postmaster*, y *postmaster* a *root*. Es conveniente cambiar esto dándole un alias *postmaster* al responsable del mantenimiento de los programas de correo.

4.5 Transportes basados en SMTP *Simple Mail Transfer Protocol* .

El *smail* soporta actualmente un controlador de SMTP para entregar el correo sobre conexiones TCP. Es capaz de entregar un mensaje a cualquier número de direcciones de una máquina, con el nombre de la misma especificado como nombre de dominio totalmente calificado que puede ser resuelto por el software de red, o con la notación de cuarteto de puntos encerrados entre corchetes. En general, las direcciones se resuelven con los controladores de encaminamiento del *BIND*, *gethostbyname(3)*, o *gethostbyaddr(3)* que lo entregarán al transporte SMTP.

El manejador de SMTP intentará conectarse al sistema remoto inmediatamente a través del puerto smtp como está listado en */etc/services*. Si no puede ser alcanzado, o expira el tiempo máximo de espera, la entrega del correo se reintentará posteriormente.

5. SENDMAIL

5.1 Introducción

Configuración básica del *Sendmail*

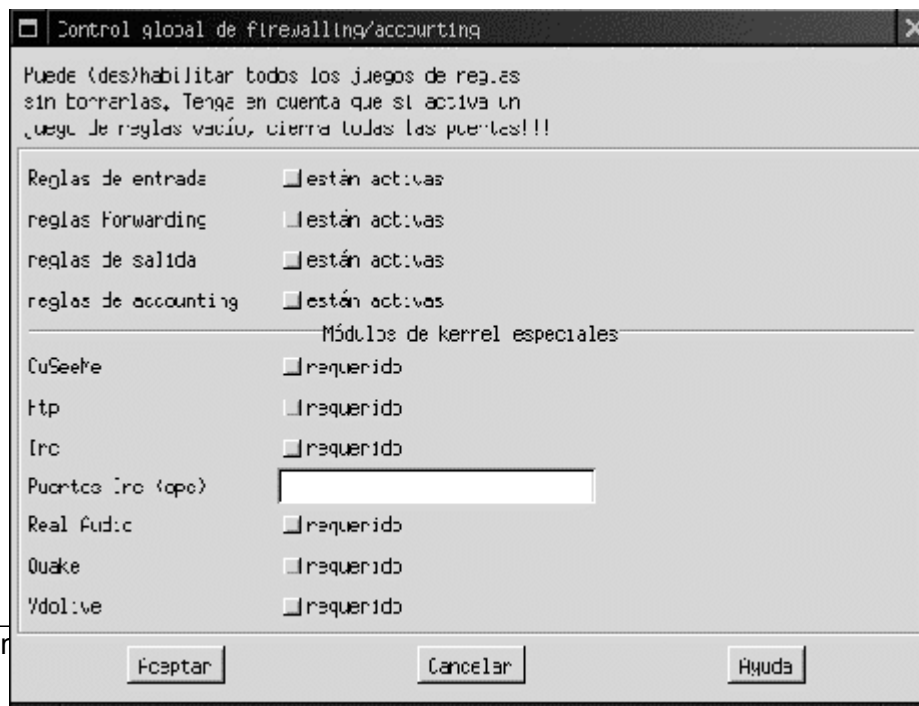


Figura: Configuración básica del servidor de email

Para muchos sistemas, ésta es probablemente la única pantalla que usted tendrá que rellenar para que los *e-mails* sean mandados. Esta configuración generalmente es diferente para servidores de *e-mail* y máquinas cliente.

5.1.1 Presentar su sistema como

Aunque el *e-mail* venga de máquinas diferentes de su empresa, usted puede querer esconder ese hecho y presentar los *e-mails* como si hubieran venido de una sola máquina o dominio.

Este campo se usa normalmente en la mayoría de las configuraciones y sencillamente contiene el nombre oficial de su dominio.

5.1.2 Aceptar *e-mail* para ``su dominio''

Si usted está configurando el servidor principal de una empresa, debe seleccionar esa opción. Normalmente, un servidor *sendmail* aceptará apenas *e-mails* enviados a su nombre@dominio exacto. Si éste es el servidor principal, él recibirá también *e-mails* dirigidos al dominio, y no al nombre del servidor. Seleccionando esa opción, el servidor aceptará ambos.

Por ejemplo, si su dominio es domy.com y su servidor es mail.domy.com, entonces un *e-mail* para juan@mail.domy.com serán aceptados y guardados localmente en este servidor (en caso del que el usuario juan exista)

5.2 Servidor de *e-mail*

En muchas empresas, hay una máquina que guarda los *e-mails* de todos los usuarios, incluso si el *e-mail* es originario de máquinas diferentes. Las máquinas *Unix* son multiusuario. Es posible que un usuario mande un *e-mail* para otro usuario de la misma máquina. A veces eso es inapropiado. Las máquinas *Unix* además de ser multiusuario, son orientadas a la red. Ese usuario (o destinatario) puede tener cuentas en otras

máquinas de la red (inclusive en todas las máquinas) y puede escoger otra máquina para leer sus *e-mails*. Tener esos mensajes de *e-mail* esparcidos por todas las máquinas de la red no es muy deseable.

Usted especifica aquí la dirección completa del servidor de la empresa y todos los mensajes de *e-mail* serán enviados para él.

Muchas máquinas en una red no tienen conectividad suficiente para repetir *e-mails* apropiadamente. Usted puede especificar el nombre de la máquina que tiene. En muchas empresas, generalmente es la misma máquina donde está el servidor de *e-mail*.

A diferencia del servidor de *e-mail*, el rutador de *e-mail* puede ser accesado por varios protocolos de envío diferentes como el ESMTP y el UUCP.

Usted debe especificar cómo acceder el rutador de *e-mail*. Si usted escoge el ESMTP, entonces debe especificar un nombre completo para el rutador de *e-mail*.

Si usted escoge UUCP, entonces debe especificar un nombre UUCP para el rutador de *e-mail*.

5.3 Funcionalidades

✓ **Activar control de envío (spammers)**

El control de envío lo deja decidir quién puede acceder su servidor SMTP (*sendmail*) para repetir mensajes. Los spammers suelen "prestar" servidores SMTP para esparcir un mensaje para una lista enorme de personas. Configurando el control de envío, usted puede limitar eso. Desafortunadamente, reglas de reenvío mal configuradas pueden impedir que algunos de sus usuarios manden *e-mails*.

Esta opción lo deja activar/desactivar las reglas. Un archivo *sendmail.cf* nuevo será generado, sin perder la configuración de las reglas.

✓ **No intentar enviar inmediatamente**

Esta funcionalidad está hecha para personas que no tienen una conexión permanente con Internet. Ella le dice al *sendmail* que grabe el *e-mail* sin revisarlo. Más tarde, usted puede disparar la entrega del *e-mail*. Generalmente se usa en un comando de postconexión (vea la sección de Discado PPP) como `"/usr/sbin/sendmail -q"`, que obliga al *sendmail* a procesar y enviar los mensajes.

✓ **Procesar consulta a cada(min.)**

De vez en cuando el *sendmail* se levanta e intenta enviar los *e-mails* que todavía le quedan. Usted puede ver esa fila de salida con el comando `mailq` sin argumentos.

Personas sin conectividad continua con Internet, pueden querer usar esa opción. Definiendo el intervalo para 0, usted desactiva esa opción. Si su conectividad es *por demanda*, usted puede querer colocar aquí un número bastante grande, como por ejemplo, 30 minutos. Cuando el *sendmail* se levante, él irá/deberá disparar la conexión.

Aquéllos que desactiven esa opción, pueden querer controlar el *sendmail* "a mano" (o vía *cron*) con el comando:

```
/usr/sbin/sendmail -q
```

✓ **Máximo tamaño de los mensajes**

Los mensajes pueden ser limitados; así, mensajes muy grandes serán rechazados. Definiendo el tamaño para 0, usted desactiva esa opción (sin límite).

5.3.1 Diversos

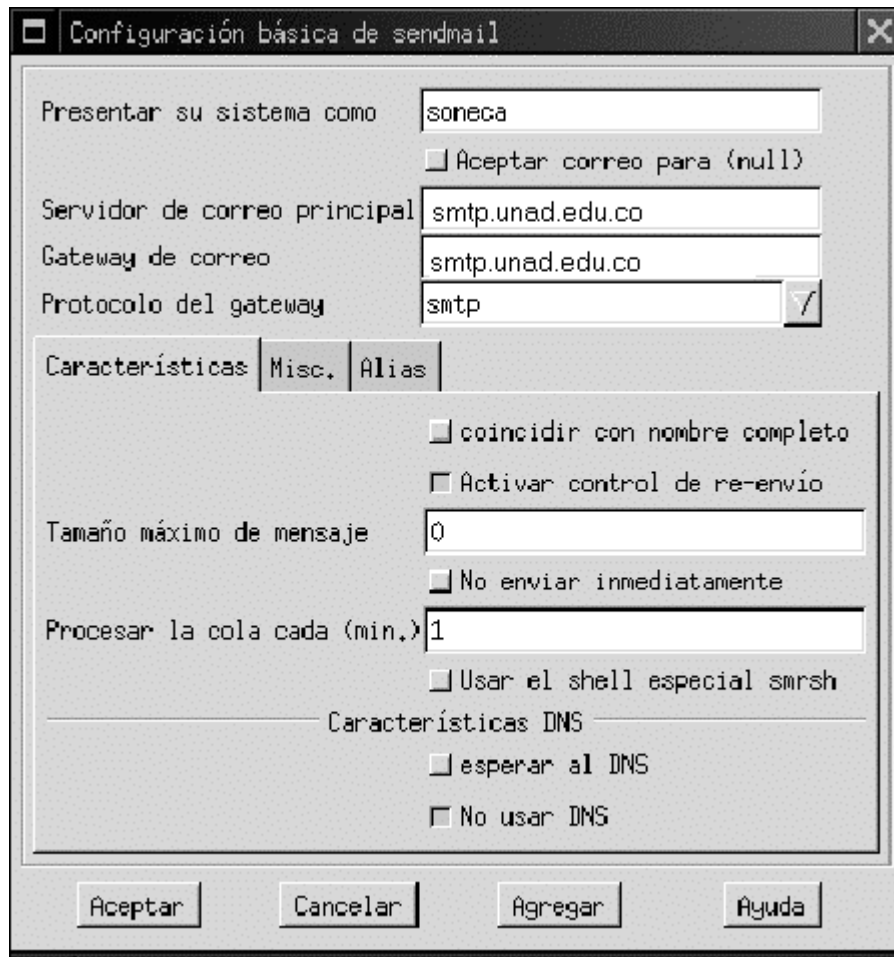


Figura: Configuración básica del servidor de email – diversos

✓ **No usar *batch* para *e-mail* UUCP**

Cuando se repite usando UUCP, los mensajes normalmente son requeridos y entregados en intervalos regulares. Usted puede querer procesarlos y enviarlos inmediatamente. Para un tráfico poco intenso, esto a veces es necesario, pues el *e-mail* es repetido inmediatamente y los usuarios quedan felices.

✓ **Entrega local para usuarios**

Coloque aquí el nombre de los usuarios para quienes usted quiere ignorar la definición de Servidor de *e-mail*. Los mensajes siempre serán guardados localmente para estos usuarios. Esta funcionalidad se usa normalmente para pseudo usuarios administrativos (root por ejemplo).

✓ **No enmascarar el origen de**

Usted puede querer ignorar la definición de "Presentar su sistema como" para algunos usuarios. Uno es el root. Eso le permite al destinatario diferenciar fácilmente en cuál sistema (¿cuál root?) el mensaje se origina.

✓ **Banco de rutados especiales**

Actívelo en caso de que usted pretenda usar rutado de dominio especial. Desafortunadamente, el Configurador Linux no puede revisar si la versión del *sendmail* que está instalada en su sistema soporta el banco de rutados especiales y en cuáles formatos. Esas son opciones de tiempo de compilación para el *sendmail* y ninguna opción de línea de comando del *sendmail* puede dar el soporte a ellas.

✓ **Formato del banco de rutados**

El *Sendmail* puede usar muchos formatos para algunas de sus bases de datos. El patrón generalmente es el formato *dbm*. Note que algunos *Sendmail* tienen soporte apenas a un formato (*dbm*), que es una opción de tiempo de compilación.

✓ **Agente de entrega local**

Utilitarios diferentes se usan para administrar *e-mails* enviados para usuarios locales. Esos utilitarios proveen varias extensiones que le permiten al usuario hacer una entrega personalizada de *e-mails*. El *procmil* por ejemplo soporta ese filtrado detallado. Algunos usan el *procmil* para esparcir los mensajes que llegan, en varias cajas y siguiendo varios criterios.

Usted puede dejar ese campo vacío y el Configurador Linux localizará el utilitario apropiado disponible en su sistema.

✓ **El *Sendmail* y el DNS**

El *Sendmail* hace un uso pesado del DNS. Máquinas sin conexión directa con la *Internet* (incluso atrás de un firewall) pueden querer desactivar el uso del DNS.

Por otro lado, máquinas con una conectividad normal del DNS deben forzar el uso del DNS.

✓ **Alias para su sistema**

Los mensajes de *e-mail* que son específicamente para su dominio son aceptados por el *Sendmail*. Otros serán retornados o repetidos.

Usted puede querer aceptar mensajes de *e-mail* para dominios diferentes. Una situación es cuando usted se registra a un proveedor de acceso y él le da uno de sus subdominios. Usted todavía quiere aceptar el antiguo, porque muchas personas lo conocen por aquella dirección.

Usted puede colocar aquí tantas líneas como considere necesarias. En caso de que quiera más líneas, use el botón Adicionar.

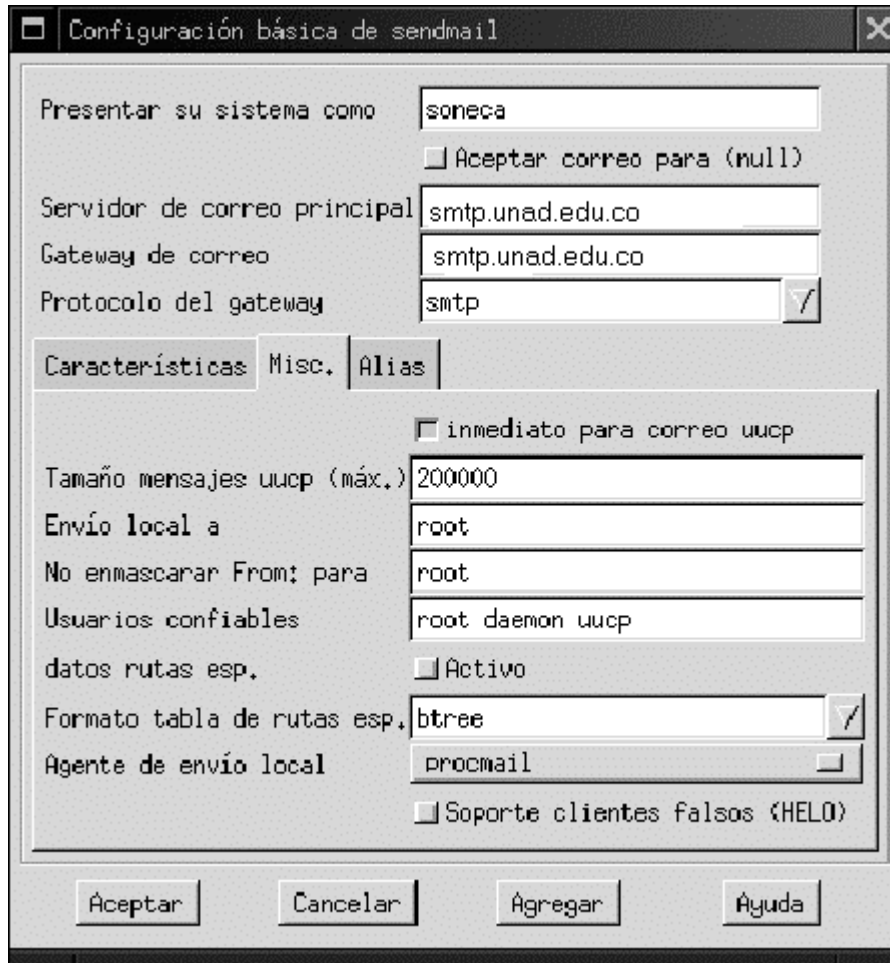


Figura: Configuración básica del servidor de email

✓ Rutado especial

El *Sendmail* normalmente transmitirá mensajes de *e-mail* directamente a su destino usando SMTP (standard *e-mail* protocol over TCP/IP, o, en español, protocolo estándar de *e-mail* sobre TCP/IP). A veces él es inapropiado. Usted puede querer intervenir "manualmente" en una decisión tomada por el *Sendmail*.

5.3.2 Diferentes Razones para hacer rutado especial

Hay muchas razones técnicas por las que alguien puede querer tener el control sobre el funcionamiento normal del *Sendmail*. He aquí algunas:

- La máquina de destino no es accesible desde su computador. Su computador está atrás de un *firewall*.
- El DNS de la máquina de destino está mal hecho o incompleto (Sin un registro MX válido).
- La máquina es accesible sólo por UUCP. Usted pasa por esa situación cuando una máquina tiene un dominio válido, pero no posee conexión directa con *Internet*. Su computador puede haber sido designado como el receptor de *e-mail* para ese dominio y como tal, él recibirá todos los *e-mails* para este dominio. Él necesitará de reglas especiales para repetir los mensajes apropiadamente.
- Usted tiene una ruta mejor (menos cara, rápida...) que la patrón para esta máquina.

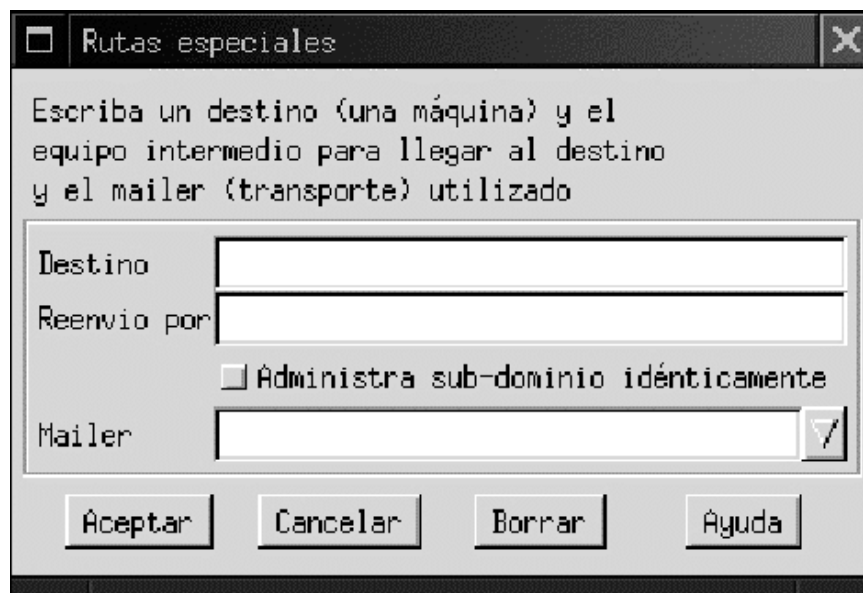


Figura: Rutas Especiales

5.3.3 Reglas de enmascaramiento

A veces tenemos un problema difícil de resolver: los mensajes de *e-mail* salen para Internet con dirección de retorno equivocada. Casi nunca existe algo que se pueda hacer, porque el sistema de origen de esos mensajes no puede ser arreglado, debido a muchas razones. Las reglas de enmascaramiento ofrecen una forma de arreglar eso en el rutador de *e-mail*.

Usted puede definir reglas de transformación sencillas que serán aplicadas a todos los *e-mails* enviados por el *sendmail*. En términos generales, usted especifica un *De:* y cómo éste *De:* debe ser convertido.

Usted tiene alguna flexibilidad aquí. Puede convertir de

- Un dominio a otro.
- Un usuario de un dominio a otro usuario de otro dominio.
- Cualquier usuario de un dominio a un usuario de otro dominio (extraño).

Sigue aquí una presentación de los campos de la caja de diálogo de las reglas.

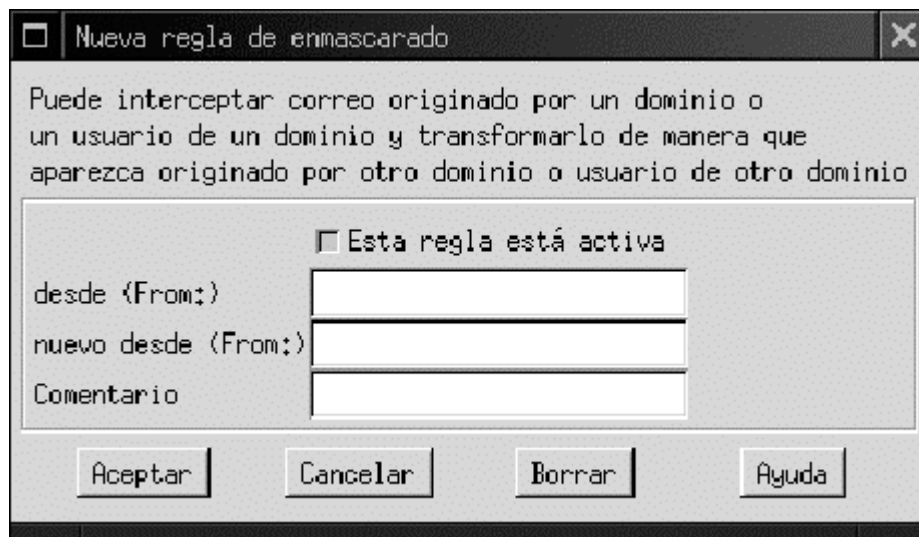


Figura: Nueva regla de enmascarado

6. FIREWALL

6.1 Firewall y contabilidad de paquetes

6.1.1 Tareas

Muchas redes privadas no están conectadas directamente con Internet por razones de seguridad. La máquina rutadora que hace de barrera entre la red pública y la red privada se llama firewall. Firewall es como si fuera un componente de un carro que protege al pasajero del fuego y de una posible explosión del motor.

Esa analogía viene del hecho de que *Internet* es un lugar "caliente", y en ciertas oportunidades muy caliente. Es una buena idea limitar el impacto de *Internet* en su red privada.

6.1.2 Filtrado de paquetes

El filtrado de paquetes hace que su computador acepte o descarte paquetes IP, basado en el siguiente criterio:

- Origen
- Destino
- Interfaz de recibimiento

Origen y destino son evaluados por:

- Número IP
- Número del puerto
- Protocolo

6.1.3 Reglas bidireccionales

Las reglas que usted está haciendo son bidireccionales. En una pantalla, usted evalúa los paquetes que vienen del origen para un destino y viceversa. Es por eso que usted puede especificar la interfaz dos veces.

6.1.4 Reglas de bloqueo

Usted puede controlar los paquetes que su máquina acepta cuando llegan. Las reglas de bloqueo lo dejan descartar paquetes en el momento de su llegada.

6.1.5 Reglas de reenvío

Usted puede controlar los paquetes que están para salir de su máquina. Usted puede descartar paquetes en el momento de su salida. Muchos firewalls pueden ser implementados utilizando reglas de bloqueo. Un caso especial de las reglas de reenvío son las reglas de enmascaramiento, que son administradas en la misma pantalla (usa opciones sencillas).

6.1.6 Activar las reglas global o selectivamente

Usted puede desactivar permanentemente una o todas las reglas sin excluirlas. Eso es útil para una primera prueba de las capacidades de un firewall y para establecer políticas "mutables". Cada regla tiene una opción para activarla.

En una pantalla es posible activar/desactivar todas las reglas de bloqueo y todas las de reenvío de una sola vez.

6.1.7 Cuando todo sale mal

El firewall en *Linux* puede ser fatal. Una mala configuración puede derribar la máquina. Un comando que puede salvarle la vida es:

```
netconf -resetfw
```

Este comando desactiva todas las reglas de firewall. Éste no afecta a la configuración, sólo a las reglas vigentes. Un

```
netconf -update
```

reactivará las reglas. Puede ser que necesite emplear estos dos comandos para revisar si un mal funcionamiento de su red tiene como causa o no las reglas de firewall.

6.2 La lógica del firewall con el Configurador Linux

El firewall en *Linux* es altamente flexible y puede llegar a ser muy complicado. El *Configurador Linux* propone una lógica que simplifica la composición de las reglas de firewall. A pesar de que esa lógica ayuda a crear y mantener un firewall básico, algunas configuraciones no pueden ser creadas con el Configurador Linux. Presentamos a continuación la lógica empleada:

6.2.1 Lógica positiva

Cuando usted activa uno de los 3 sistemas de firewall (entrada, reenvío, salida), el *Configurador Linux* definirá la política estándar para *bloquear*. Todas las reglas que usted coloque son brechas en el firewall. Si usted no coloca ninguna regla, su máquina quedará aislada, y puede que responda o no.

6.2.2 Una solución prefabricada de firewall

He aquí un ejemplo para una red sencilla. A lo sumo, usted tiene una eth0 apuntando para la red local. Ninguna persona de fuera debe acceder esa red. La eth1 apunta para un rutador en Internet.

6.2.3 Reglas de bloqueo: aceptando todos los paquetes ICMP

```
Protocolo      ICMP
-----De-----
Máquina o red 0.0.0.0
Máscara
Faja de puertos
Otros puertos
Interfaz      eth1
-----Para-----
Máquina o red eth1
Máscara
Faja de puertos
Otros puertos
Interfaz     Any
```

6.2.4 Reglas de bloqueo: aceptando sólo paquetes DNS

```
Protocolo      udp
-----De-----
Máquina o red 0.0.0.0
Máscara      0.0.0.0
Faja de puertos
Otros puertos
Interfaz      eth1
-----Para-----
Máquina o red eth1
Máscara
Faja de puertos
Otros puertos
Interfaz     Any
```

6.2.5 Estado del soporte al firewall en *Configurador Linux*

El firewall es un tópico caliente hoy en día y también es un arte que se considera aún como oscuro. El *Configurador Linux* está intentando volverlo un poco más accesible. El autor espera recibir comentarios referentes a este tópico.



Figura: Regla de Firewall para reenvío

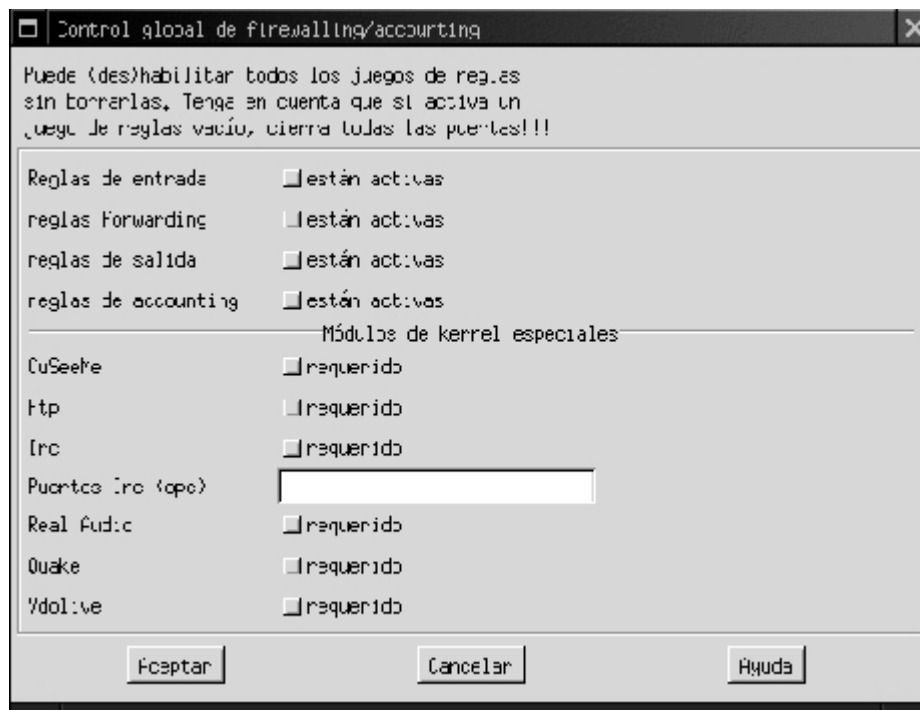


Figura: Control global de firewall/contabilidad

BIBLIOGRAFIA

OLAF Kirch, Guía del Administrador de Redes, Conectiva Linux, 1994

Conectiva Informatica y Otros Autores, Guía del Servidor, Conectiva Linux, 2000

David Bandel y Robert Napier, Linux 6a. Edición, Prentice Hall,,2001

Thomas Schenk et a, Administración de Red Hat Linux Al descubierto, Prentice Halll, 2001