

# Pasarelas de mensajería SMS

**Andrés Seco Hernández**  
Proyecto Alamin

**AndresSH@alamin.org**

*Conexión de aplicaciones servidoras a pasarelas de mensajería SMS para permitir su uso desde teléfonos móviles mediante mensajes cortos.*

En la actualidad, el uso de los mensajes cortos de los teléfonos celulares (SMS) está siendo utilizado de forma muy extensa. La facilidad de acceso al servicio, las posibilidades en la comunicación con otras personas y la sencillez de su concepción están haciendo que este medio de comunicación continúe creciendo tras muchos años de encontrarse disponibles para el usuario final.

Pero los SMS no sólo pueden utilizarse en la comunicación entre personas. Puede obtenerse un gran servicio de su utilización para comunicar con agentes automáticos que respondan a mensajes enviados automática o manualmente.

En este documento se presentan opciones para la puesta en marcha de este tipo de servicios mediante proyectos Open Source mostrando su arquitectura y algunas de sus posibilidades, centrándose de forma más detallada en la pasarela Alamin.

## 1. Copyright

v1.4, 29 de Octubre de 2001.

Copyright Andrés Seco Hernández. Se otorga permiso para copiar, distribuir y/o modificar este documento bajo los términos de la Licencia de Documentación Libre GNU, Versión 1.1 o cualquier otra versión posterior publicada por la Free Software

Foundation. Puede consultar una copia de la licencia en:

<http://www.gnu.org/copyleft/fdl.html>

La figura 2, "Arquitectura de kannel" es Copyright (c) 1998 WAPIT OY LTD. All rights reserved. This product includes software developed by WAPIT OY LTD.

## **2. Introducción**

Con el paso del tiempo los ordenadores han ido haciéndose más y más pequeños desde aquellos gigantes de acero y válvulas hasta los ordenadores personales de la actualidad, y estamos asistiendo ahora a la aparición de pequeños dispositivos del tamaño de la palma de la mano de potencia creciente día a día en un afán de alcanzar la posibilidad de comparación con los ordenadores de sobremesa.

A la vez, las comunicaciones se han ido haciendo cada vez más importantes, por no decir imprescindibles. Muchas personas, ya no conciben un ordenador sin conexión a otros. El ordenador aislado poco a poco va convirtiéndose en una reliquia del pasado.

Por otra parte, en los últimos años, el acercamiento de la telefonía móvil a todo tipo de gente mediante precios más o menos asequibles ha propiciado que nuestra conciencia de "estar conectados" aumente, convirtiéndonos así en "animales con necesidad de conexión", y si esta es permanente, on-line, mucho mejor.

Todas estas situaciones han propiciado que estar conectado con un teléfono móvil y poseer un ordenador personal con conexión a internet no sea extraño a comienzos del siglo XXI.

El papel de las compañías operadoras de telefonía ha sido muy importante en esta expansión de las comunicaciones, sin duda, y han tratado y tratan de amortizar sus inversiones mediante la oferta de diferentes servicios que inciten al usuario a hacer uso de estas comunicaciones.

Uno de los servicios ofertados por estas operadoras en los teléfonos celulares de tecnología GSM es el envío y recepción de mensajes cortos, los conocidos SMS. Este servicio, que ha tenido un éxito desigual en diferentes partes del mundo, en Europa ha causado furor, contándose por cientos de millones los mensajes enviados cada mes en la mayoría de los países de nuestro entorno.

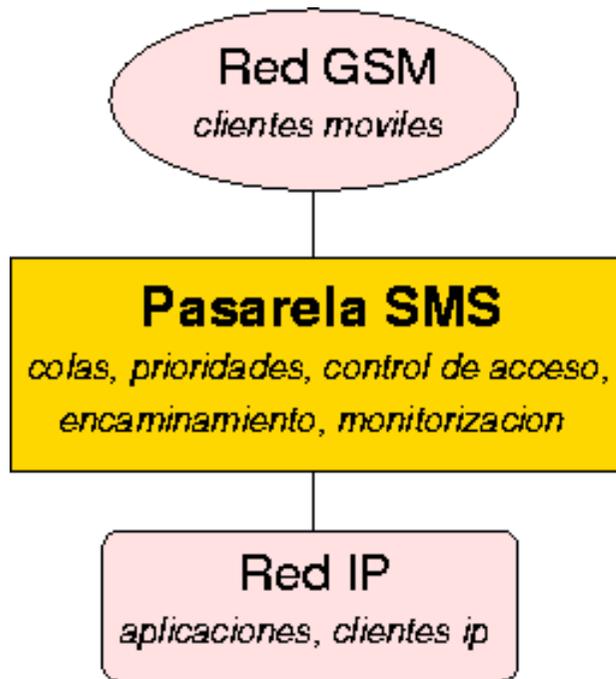
Los mensajes SMS son herederos directos de los mensajes de los equipos localizadores de personas, los llamados "buscas", pero extendiendo su funcionalidad para permitir que desde cualquier dispositivo GSM se pueda realizar un envío a otro equipo sin mediar comunicación vocal con un teleoperador.

El éxito de este servicio, el SMS, parece provenir de la sencillez y facilidad de manejo, por un lado, y de que "hay alguien al otro lado" con quien realizar el acto de la comunicación. Estos dos factores han provocado dicho éxito, aún teniendo en contra el precio del servicio en algunos casos y las limitadas características de esta comunicación.

Las posibilidades de comunicación mediante "mensajes cortos GSM" (SMS) son muchas y muy variadas, pero siempre limitadas por las características de estos mensajes, 160 caracteres, muy baja velocidad (en comparación con las líneas telefónicas convencionales), duración limitada (24 ó 48 horas normalmente, si no se entregan antes son cancelados), no es un servicio garantizado (el mensaje suele llegar pero no hay garantía de ello, ni que lleguen en el orden en que se han enviado) y posibilidad de comunicación sólo entre teléfonos celulares GSM entre los que haya "visibilidad" (que los operadores de los dos teléfonos, emisor y receptor, tengan convenio de intercambio de mensajes). Existen muchas especificaciones de formato de mensaje para los servicios prestados a través de SMS que les dotan de gran potencia y complejidad. Pero es en el uso básico con un sistema de enlace sencillo donde se están obteniendo los mejores resultados, tanto de cantidad de mensajes enviados como de servicios que se están utilizando.

En todo caso, estas posibilidades resultan suficientes aprovechadas de forma adecuada, y una de esas formas es tener uno de los lados de la comunicación gobernado por un servicio automático que se encargue de responder a las peticiones recibidas desde múltiples teléfonos celulares. La automatización de la recepción de los mensajes SMS, su procesado y posterior respuesta es lo que conforma la funcionalidad de una pasarela SMS.

Figura 1. Esquema general de una pasarela SMS



El acceso a la red GSM se puede obtener de diferentes formas. El método más sencillo es utilizando directamente un terminal GSM conectado al ordenador que actúa de pasarela. En realidad este terminal puede ser un teléfono GSM normal con su kit de conexión a PC (cable y software) o un modem GSM (igual que los modems convencionales de red telefónica básica -RTB- pero su medio de transmisión es la red GSM, no el par de hilos telefónicos).

La comunicación entre el ordenador y el terminal se suele realizar por un puerto de comunicaciones serie. Casi todos los teléfonos móviles actuales incluyen un modem GSM en su interior, de manera que la forma de comunicar ordenador y teléfono/modem GSM es la misma que con un modem de RTB convencional, es decir, comandos AT. Si el teléfono no incluye un modem GSM en su interior, es necesario comunicar con el teléfono utilizando las especificaciones del protocolo que el fabricante haya utilizado (normalmente se trata de protocolos propietarios, aunque cada vez menos). Este es el caso de algunos modelos de teléfonos Nokia como el 5110 y 6110. En GNU/Linux podemos utilizar el proyecto Gnokii, que trata de implementar el paquete de software Nokia Data Suite para la comunicación con estos teléfonos que implementan un protocolo propietario de Nokia.

El otro método más habitual para acceder a la red GSM es contactar directamente con un centro servidor de mensajes (SMSC) del operador de telefonía. Los SMSC de cada fabricante incorporan también protocolos propietarios, por lo que es necesario realizar en cada caso un dialogo diferente con cada tipo de SMSC, además de que el medio de conexión también puede variar de unos a otros (IP, Frame Relay, X.25, RDSI, ...).

Una vez comprendido el funcionamiento básico de una pasarela de mensajería SMS, nos centraremos en la unión de las redes GSM, en un lado de la pasarela, e IP, en el otro, dada su gran extensión y su uso común en todos los tipos de sistemas informáticos.

Existen, o pueden existir, otras pasarelas SMS entre las redes de telefonía celular y otras redes de ordenadores, o incluso otras redes celulares de diferente tecnología, pero no serán objeto de estudio en este documento.

### **3. Aplicaciones de una pasarela de mensajería SMS**

Los posibles usos que se pueden dar a una pasarela SMS entre las redes GSM e IP son extensísimas.

Sirva la siguiente lista como ejemplo de actividades que se realizan en la actualidad con este tipo de pasarelas de mensajería:

- *Correo electrónico.* La pasarela convierte un mensaje de correo en SMS y un mensaje SMS en mensaje de correo, con las consiguientes generaciones/eliminaciones de cabeceras de mensaje.
- *Distribución de mensajes SMS.* Al igual que funcionan las listas de correo electrónico, en las que un mensaje es reenviado a los suscriptores de la lista, en la lista de distribución de mensajes SMS la pasarela permite el mantenimiento (alta/baja/consulta) de suscriptores y envía al resto de suscriptores de la lista los mensajes que no son comandos de actuación sobre la propia pasarela.
- *Recepción de alarmas de los sistemas de monitorización de servicios.* Aplicaciones como "mon", "heartbeat", agentes "snmp", etc. generan avisos cuando se alcanzan ciertos eventos. Estos avisos pueden ser encaminados mediante mensaje SMS dependiendo de su importancia, para que ciertas personas sean avisadas inmediatamente.

- *Transporte de contenidos web.* El SMS es utilizado como paquete de transporte para hacer llegar desde el móvil al servidor la petición de una página web y desde el servidor al móvil el contenido de dicha página una vez "filtrada" para eliminar imágenes, tags html, cabeceras de página, etc.
- *Concursos de preguntas y respuestas.* Ante una solicitud desde el teléfono móvil para comenzar el concurso, la pasarela envía mensajes con preguntas, recibe respuestas y mantiene un contador de resultados para cada participante, de manera que se pueden generar clasificaciones.
- *Sistemas de seguimiento de flotas de vehículos.* Un teléfono celular unido a un módulo GPS permite enviar información acerca de la posición exacta del portador del teléfono, de manera que para flotas de vehículos se reciben sus notificaciones de posición en la pasarela y ésta actualiza una base de datos consultable por otras aplicaciones que pueden mostrar la situación de cada elemento de la flota.
- *Notificación de estado de dispositivos aislados.* Máquinas de vending, dispositivos de control de temperatura y presencia, etc. que se encuentren aislados y sin comunicación con una red mediante enlace físico pueden hacer uso de los mensajes SMS para recibir ordenes y para notificar de su estado (queda poca bebida, la temperatura ha superado los 45 grados, etc.). Normalmente esta comunicación se realiza sin intervención manual, por lo que realmente se conectan equipos automáticos en ambos lados. Es lo que Nokia denomina M2M (machine to machine).

Además, de inmediato, a cada persona le surgen nuevas aplicaciones, orientadas a su área de conocimiento:

- *Consulta de bases de datos.*
- *Mantenimiento de sistemas.* Consultas de estado de servicios, reinicio de los mismos, reinicio de equipos.
- *Notificación de informaciones académicas.* Notas, convocatorias de examen, etc.
- *Banca GSM.* Servicios financieros y alarmas para el seguimiento de operaciones de valores.
- *Cualquier otro tipo de alarmas* debido a caídas de tensión, detección de intrusos en firewalls, etc.
- *Domótica.* Control y consulta de dispositivos a distancia.
- *Avisos de intervención* para equipos médicos, mantenimiento, rescate, etc.

Dadas las características de la red GSM que permiten la movilidad de los terminales (teléfonos) en su zona de cobertura, se pueden imaginar aplicaciones que aprovechen la posibilidad de localización de un teléfono, en base a la estación base de la red que en ese momento tiene conexión con él. Sin embargo, esta información no es directamente accesible desde el exterior de la red del operador de telefonía, por lo que, salvo aplicaciones fuertemente integradas con la red del operador, no es posible su utilización.

Podríamos imaginar una aplicación que permitiera emitir mensajes SMS a teléfonos móviles entre las 2 y las 3 de la tarde en la zona de cobertura de una estación base situada junto a un restaurante que contratara los servicios de publicidad que un operador pudiera ofrecer, para que todos los que por allí cerca pasaran supieran donde está dicho restaurante. Al margen de la aplicación, podrían surgir problemas con la utilización de la posición de los terminales para operaciones no solicitadas por el propietario del terminal, ya que al fin y al cabo, la situación de cada terminal es información privada del propietario.

De cualquier manera, la pasarela de mensajería sólo trata de servir de intermediario y facilitar la labor de desarrollo de las extensiones móviles para una aplicación dada.

La pasarela de mensajería SMS/IP trata de ser como un servidor web, realiza sus tareas de cambio de formato y ajuste del mensaje, pero precisa de contenidos que realmente le den una utilidad, aunque en este caso los contenidos son pequeñas o grandes aplicaciones que permiten interactuar al usuario móvil con otro programa.

También tiene un comportamiento similar al de un Agente de Transferencia de Mensajes de correo (MTA) ya que, de alguna forma, ésta también es su tarea: el encolado, conmutado y entrega de mensajes.

Para finalizar con las utilidades de las pasarelas de mensajería SMS/IP, añadir que no hay un estándar para que las aplicaciones comuniquen con las pasarelas, en general, sino que cada una define su propio interfaz, que es diferente en todos los casos. En este aspecto queda mucho camino por recorrer para, quizás, definir un wrapper, un interfaz intermedio, normalizado, que permita la utilización de diferentes pasarelas SMS sin necesidad de realizar cambios en la aplicación.

## **4. Algunos proyectos de fuente abierto**

### **4.1. Kannel**

Kannel utiliza la licencia BSD.

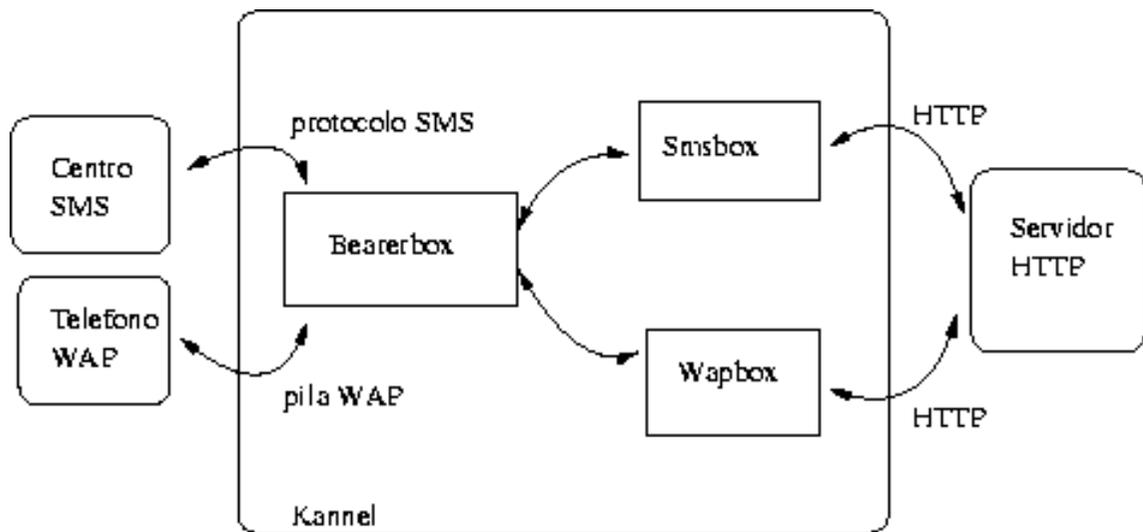
Kannel fue en primer lugar una pasarela WAP, pero amplió sus funcionalidades posteriormente para dar soporte a WAP usando SMS como transporte de los datos y para obtener páginas web.

El código ahora es bastante estable, estando reconocido y certificado por el WapForum como una pasarela que se ajusta a los estándares WAP. Detrás de Kannel hay un numeroso grupo de desarrolladores y colaboradores. La empresa que comenzó a desarrollar Kannel se llamaba WapIt. Recientemente tuvo problemas económicos y cerró. Desde entonces, se está tratando de crear una fundación que dé apoyo al mantenimiento del desarrollo de Kannel, pero las últimas noticias que aparecen en el sitio web de Kannel son del mes de Junio de este año...

Kannel es un proyecto grande, con una estructura detallada que le convierte en "demasiado complejo" para algunas instalaciones. Sin embargo, su objetivo principal es la escalabilidad y alta disponibilidad desde su planteamiento inicial.

Básicamente, en una instalación mínima, aparece lo que en Kannel denominan "BearerBox", el componente que comunica con el centro servidor de mensajes (SMSC) o un teléfono WAP que puede sustituirlo en parte. Con este BearerBox comunican los otros componentes, "SMSBox" y "WAPBox", que se encargan de acceder a los servidores HTTP que tienen el contenido que se desea acceder, y que realizan las conversiones/compresiones que establece el protocolo WAP para el envío de los mismos a los teléfonos.

Figura 2. Arquitectura de Kannel



En una instalación orientada a SMS, desde el teléfono celular se envía un mensaje que es recibido por el SMSC u otro teléfono móvil, que recibe el mensaje para pasárselo al BearerBox; el mensaje es encaminado al SMSBox, que accede a una página web (puede ser un cgi que genere la página dinámicamente, para ejecutar comandos o lanzar programas, por ejemplo) y el resultado es devuelto al BearerBox de nuevo para ser devuelto al teléfono.

Kannel está orientado principalmente al trabajo directo con centros servidores de mensajes (SMSC) e implementa muchos protocolos de acceso a SMSCs de diferentes fabricantes. Es en la conexión directa con un SMSC cuando se obtienen los mejores resultados de rendimiento; sin embargo, no siempre se tiene disponible una conexión de este tipo con la red GSM, por lo que también implementa conexión mediante modem GSM o teléfono móvil con modem incorporado.

## 4.2. Alamin

Alamin utiliza la licencia GPL.

Alamin nació inicialmente con el objetivo de poder lanzar sencillas líneas de comando desde un teléfono móvil. En Marzo de 1998 se implementó un servicio que monitorizaba un buzón de correo y recibía sus mensajes, interpretaba su contenido y

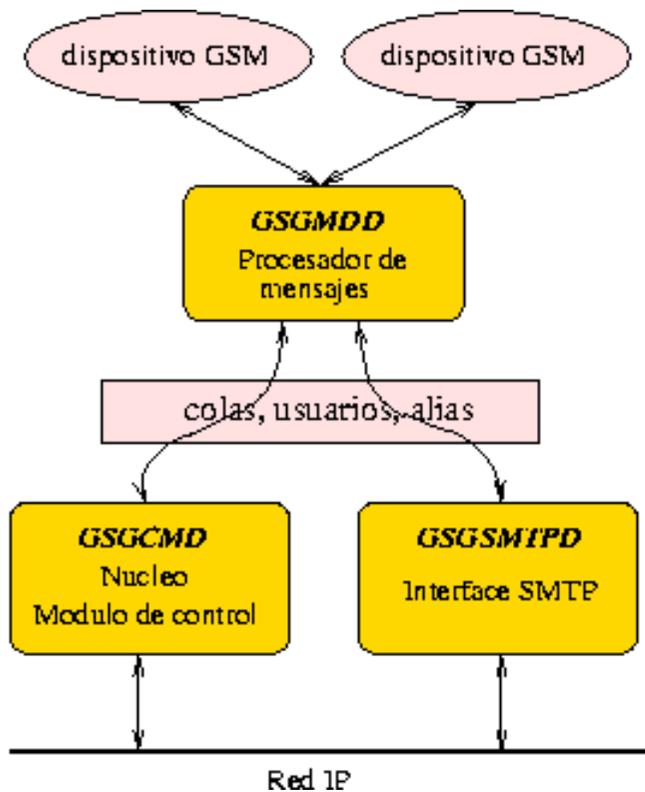
realizaba alguna acción, devolviendo un resultado también por correo electrónico. El acceso al teléfono móvil se realizaba mediante una pasarela sms/email. Esta pequeña pasarela estaba realizada en Perl y sus especificaciones se pueden encontrar en <http://andressh.alamin.org/SSJP/>. Este servicio no llegó a ser puesto en marcha en ninguna instalación real.

Posteriormente, a mediados de 1999, el proyecto Gnokii permitió acceso directo a la red GSM. Ya existían los modems GSM, pero estaban muy poco extendidos, eran caros y difíciles de conseguir.

Alamin utilizó primero Gnokii como medio de acceso a la red GSM, y, al evitar usar otra pasarela a más bajo nivel, se ganó en fiabilidad. En mayo de 2000 se pusieron en marcha las primeras instalaciones en explotación de Alamin.

Alamin se compone principalmente de dos componentes: el manejador de mensajes (gsgmdd, Message Dispatcher) y el núcleo que gestiona las peticiones desde la red IP (gsgcmd, Core Module).

**Figura 3. Arquitectura de Alamin**



El Core Module es el que centra la gestión del servicio, el control de acceso de las peticiones desde la red IP, despliega los alias en los números que corresponda y prepara las solicitudes para ser enviadas. La comunicación entre los clientes de la red IP y el Core Module se realiza mediante el protocolo smsqp, que utiliza por defecto el puerto 11201/tcp para su comunicación. Este protocolo no está terminado de especificar, ya que se está desarrollando con Alamin.

El Message Dispatcher se encarga de la entrega de los mensajes desde las colas de prioridades al dispositivo GSM, comprueba la existencia de mensajes entrantes y lanza programas externos para el tratamiento de los mensajes que han llegado. Inicialmente, el Message Dispatcher utilizaba Gnokii para realizar estas tareas, pero ahora ya usa comunicación directa con modems GSM mediante comandos AT. En el momento actual de desarrollo, sólo se permite un dispositivo GSM por cada Message Dispatcher. En sucesivas versiones se podrá utilizar múltiples dispositivos GSM, tablas de rutas para decidir por dónde se envía un mensaje, y su arquitectura será cambiada para facilitar que se puedan añadir de forma sencilla nuevos "controladores" para otros dispositivos GSM o para utilizar otras pasarelas en cascada (http, accesos directos a SMSC, protocolos de otras pasarelas, etc.)

Adicionalmente a estos dos componentes van a ser añadidos otros que implementan diferentes protocolos para permitir que otras aplicaciones, directamente, sin cambios, accedan a la red GSM. Uno de ellos, que ya está incluido en la distribución de la pasarela y que implementa el protocolo SMTP, es gsgsmtpd (SMTP interface). De esta forma, la pasarela traduce directamente mensajes de correo a SMS.

Están pendientes de implementación otros protocolos, como POP, que permitirá que desde cualquier cliente de correo se puedan recibir mensajes SMS convertidos a mensajes de correo electrónico. También está pendiente de implementar el modo PDU en el envío de mensajes para modems GSM. Sólo se soporta, temporalmente, el modo texto. El modo PDU en el envío de mensajes SMS mediante comandos AT permite enviar caracteres no imprimibles, es decir, no solo letras, números y algunos caracteres especiales, si no que cualquier valor desde 0 a 255 puede ser utilizado. Es necesario para el envío de mensajes en formatos avanzados o el envío de melodías y logotipos.

El cliente de Alamin es un sencillo programa en Perl de línea de comando que puede tener sus opciones de configuración en un fichero o indicarlas en la propia llamada al programa:

```
andres@pruebas:~$ gsgc --help
Usage: gsgc [--version|--help]
        gsgc [--configfile config_file_name]
```

```

[--debug|--nodebug]
[--verbose|--noverbose]
[--copyright|--nocopyright]
[--port port_number]
[--host host_name]
[--syslog facility|--nosyslog]
[--priority 1-9]
[--smsc short_message_service_center_number]
[--user username]
[--password secret_password]
[[--send dest_number[,more_dests]... message_content]
]
andres@pruebas:~$ gsgc --user andres --password clave \
--host msgate.dominio.com --priority 5 \
--send 612345678 "mensaje de prueba"
Message sent.

```

En el ejemplo, *--user andres --password clave* se usan para identificarse de cara a la pasarela. Si no se especifican, se realiza una conexión anónima. Según esté configurada la pasarela, puede permitirse/denegarse el envío a determinados números dependiendo el usuario que trate de realizarlo. El parámetro *--host msgate.dominio.com* especifica el nombre del equipo en el que está alojada la pasarela. Finalmente, *--priority 5* indica que debe ser utilizada la cola de prioridad 5. Estas colas permiten, por ejemplo, enviar masivamente mensajes por una prioridad baja e intercalar posteriormente mensajes más urgentes por una cola de mayor prioridad, adelantando estos mensajes a los anteriores en la salida hacia la red GSM. El parámetro *--send 612345678 "mensaje de prueba"* solicita el envío de la cadena que aparece entre comillas al número de teléfono indicado.

Mientras, en el servidor el mensaje es recibido, tras comprobar que la conexión se realiza desde un equipo permitido, con un usuario y contraseña válidos (la contraseña no viaja por la red, de forma similar a como CHAP lo hace), el mensaje se coloca en la cola correspondiente y, finalmente, es enviado al dispositivo GSM. En caso de que haya algún problema durante el envío, el mensaje se reintentará enviar según se haya indicado en la tabla de reintentos, en el fichero de configuración.

En el siguiente extracto de syslog se registra el diálogo entre el cliente (gsgc) y el Core Module (gsgcmd), hasta que el mensaje queda encolado. Las líneas *Received* indican el mensaje que la pasarela envía al cliente, y las *Sending* las que el cliente envía a la pasarela:

```
Sep 15 18:48:26 pruebas gsgc[13]: Starting Alamin GSM SMS
```

```
Gateway - Client
Sep 15 18:48:26 pruebas gsgc[13]: Received: OK Alamin GSM
SMS Gateway
Sep 15 18:48:26 pruebas gsgc[13]: Received: READY
Sep 15 18:48:26 pruebas gsgc[13]: Sending: send 5 - default <list>
Sep 15 18:48:26 pruebas gsgc[13]: Sending: 612345678
Sep 15 18:48:26 pruebas gsgc[13]: Sending: <EOL>
Sep 15 18:48:26 pruebas gsgc[13]: Sending: mensaje de prueba<EOM>
into queue 5 message 00004891-14660.sms.txt
Sep 15 18:48:26 pruebas gsgc[13]: Received: WAIT Sending
to 612345678
Sep 15 18:48:26 pruebas gsgc[13]: Received: OK Message to
612345678 queued
Sep 15 18:48:26 pruebas gsgc[13]: Received: READY
Sep 15 18:48:26 pruebas gsgc[13]: Sending: CLOSE
Sep 15 18:48:26 pruebas gsgc[13]: Received: OK Closing
Sep 15 18:48:26 pruebas gsgc[13]: Received: BYE
```

El cliente, al estar escrito en Perl, puede correr en cualquier plataforma con intérprete de perl y sockets. Está siendo desarrollado un applet Java para permitir su uso directamente desde clientes que dispongan de navegador de internet y máquina virtual Java.

### **4.3. SMSLink**

SMSLink utiliza la licencia GPL.

SMSLink es similar en muchos aspectos a Alamin, pero tiene ciertas diferencias de arquitectura.

En principio, sólo hay un proceso corriendo, y lanza nuevos hilos para cada petición de envío de mensaje recibido de la red IP. El acceso concurrente al dispositivo GSM se gestiona mediante un sistema de bloqueos y semáforos.

Permite el uso de múltiples dispositivos GSM desde su comienzo. Para la recepción de mensajes, existe la opción de almacenar los mensajes entrantes o pasárselos a un programa que los procese.

Los dispositivos GSM soportados son todos los que soporten comandos AT extendidos, esto es, modems GSM y teléfonos con modem incorporado. Tanto el modo PDU como el modo texto están disponibles para el envío de mensajes.

En este momento no incorpora todavía gestión de colas, prioridades, usuarios y permisos por usuario. Estas son las deficiencias con respecto a Alamin, sin embargo, Alamin solo puede manejar un dispositivo GSM, y en caso de ser un modem, no puede usar el modo PDU para envío de mensajes con melodías y logotipos.

## **4.4. SMSTools**

SMSTools utiliza la licencia GPL.

La filosofía general de la aplicación es la misma que Alamin y SMSLink. Tan solo hay diferencias en detalles de implementación.

Soporta hasta 32 dispositivos GSM simultáneos. Gestiona una cola por cada dispositivo y se pueden establecer rutas de manera que ciertos mensajes se envíen solo por un dispositivo determinado.

También permite establecer qué programa debe ser lanzado cada vez que se recibe o se envía un mensaje.

No resulta muy flexible en redes con problemas de transmisión o en momentos de mucha carga de la red, ya que la configuración de reintentos no resulta flexible.

Tampoco, como SMSLink, incorpora todavía gestión de colas, prioridades, usuarios y permisos por usuario.

## **4.5. Otros programas que pueden utilizarse como pasarelas**

Existen multitud de programas que permiten el envío de mensajes cortos, de uno en uno, sin gestión de colas, prioridades, etc. y que no incluyen gestión de la contención en el acceso al dispositivo GSM ni reintentos en caso de fallo en el envío. Su función principal es enviar un mensaje accediendo físicamente al dispositivo GSM (Chat, Gnokii) o conectar con otra pasarela de mensajería (Esms, Smsgsend, Smsgclient), por lo que podríamos considerarlos solo la parte cliente de una pasarela.

Muchos de estos programas se usan como "interfaz de acceso a la red GSM" desde otras pasarelas más complejas, como es el caso de Gnokii y Chat, que son utilizados por Alamin.

Pero en ocasiones, las necesidades de comunicación no requieren demasiada complejidad y cualquiera de estos sencillos programas son suficientes para notificar de

un evento ocurrido en una red o recibir un comando simple que sea pasado a bash desde un SMS como "/etc/init.d/apache start".

No aparecen aquí todos los existentes, pero sí algunos:

- *Chat*. El programa que muchos conocemos para el marcado del número en un modem. Al igual que existen modems para Red Telefónica Básica (RTB) existen modems GSM, que utilizan la red GSM para realizar su transmisión. Dado que la red GSM es digital y dispone de servicios adicionales, los comandos AT con los que se gobierna un modem convencional de RTB han sido extendidos para los modems GSM y existen nuevos comandos (normalmente identificados por un caracter de más "+" tras la cadena de atención, "AT"), que nos permiten enviar un mensaje, leer los mensajes recibidos, borrar un mensaje, establecer opciones como caducidad, modo de transferencia, etc. En resumen, las mismas funciones que podemos realizar desde el teclado de un teléfono móvil. La siguiente línea de comando para chat elige el modo texto para los mensajes SMS y comprueba los nuevos mensajes que hayan llegado.

El siguiente ejemplo muestra una llamada a Chat en la que se establece un tiempo de 20 segundos de espera a la respuesta del modem y varios pares de parametros en los que el primer elemento del par es el texto que Chat espera recibir y al que responde con el segundo par:

```
/usr/sbin/chat -v -e TIMEOUT 20 " '^EAT+CMGF=1' \  
OK 'AT+CMGL="ALL"' OK " </dev/ttyS0 >/dev/ttyS1
```

El resultado en syslog sería el siguiente:

```
Sep 17 16:48:25 pruebas chat[11767]: timeout set to 20 seconds  
Sep 17 16:48:25 pruebas chat[11767]: send (^EAT+CMGF=1^M)  
Sep 17 16:48:26 pruebas chat[11767]: expect (OK)  
Sep 17 16:48:26 pruebas chat[11767]: ^EAT+CMGF=1^M^M  
Sep 17 16:48:26 pruebas chat[11767]: OK  
Sep 17 16:48:26 pruebas chat[11767]: -- got it  
Sep 17 16:48:26 pruebas chat[11767]: send (AT+CMGL="ALL"^M)  
Sep 17 16:48:26 pruebas chat[11767]: expect (OK)  
Sep 17 16:48:26 pruebas chat[11767]: ^M  
Sep 17 16:48:26 pruebas chat[11767]: AT+CMGL="ALL"^M^M  
Sep 17 16:48:26 pruebas chat[11767]: OK  
Sep 17 16:48:26 pruebas chat[11767]: -- got it  
Sep 17 16:48:26 pruebas chat[11767]: send (^M)
```

Y para enviar un mensaje, el siguiente script de chat en el que igualmente se ve que sin esperar nada, será enviada la cadena *AT+CMGF=1*, cuando se reciba la cadena *OK* será enviada *AT+CSCA="+34609090909",145* (que indica cual es el centro servidor de mensajes a utilizar), cuando vuelva a ser recibido *OK* se responderá con *AT+CMGS="612345678"* (solicitud de envío de un mensaje al número indicado), y cuando se reciba el caracter "menor que" se enviará el texto de mensaje finalizando con el caracter "ctrl+Z":

```
TIMEOUT 20
" 'AT+CMGF=1'
OK 'AT+CSCA="+34609090909",145'
OK 'AT+CMGS="612345678"'
'>' 'texto del mensaje^Z'
OK "
```

genera este resultado en syslog:

```
Sep 17 17:00:17 pruebas chat[12]: timeout set to 20 seconds
Sep 17 17:00:17 pruebas chat[12]: send (AT+CMGF=1^M)
Sep 17 17:00:17 pruebas chat[12]: expect (OK)
Sep 17 17:00:17 pruebas chat[12]: AT+CMGF=1^M^M
Sep 17 17:00:17 pruebas chat[12]: OK
Sep 17 17:00:17 pruebas chat[12]: -- got it
Sep 17 17:00:17 pruebas chat[12]: send (AT+CSCA="+34609090909",
145^M)
Sep 17 17:00:18 pruebas chat[12]: expect (OK)
Sep 17 17:00:18 pruebas chat[12]: ^M
Sep 17 17:00:18 pruebas chat[12]: AT+CSCA="+34609090909",145^M^M
Sep 17 17:00:18 pruebas chat[12]: OK
Sep 17 17:00:18 pruebas chat[12]: -- got it
Sep 17 17:00:18 pruebas chat[12]: send (AT+CMGS="612345678"^M)
Sep 17 17:00:18 pruebas chat[12]: expect (>)
Sep 17 17:00:18 pruebas chat[12]: ^M
Sep 17 17:00:18 pruebas chat[12]: AT+CMGS="612345678"^M^M
Sep 17 17:00:18 pruebas chat[12]: >
Sep 17 17:00:18 pruebas chat[12]: -- got it
Sep 17 17:00:18 pruebas chat[12]: send (texto del mensaje^Z^M)
Sep 17 17:00:19 pruebas chat[12]: expect (OK)
Sep 17 17:00:23 pruebas chat[12]: texto del mensaje^Z^M^M
Sep 17 17:00:23 pruebas chat[12]: +CMGS: 28^M
Sep 17 17:00:23 pruebas chat[12]: ^M
Sep 17 17:00:23 pruebas chat[12]: OK
Sep 17 17:00:23 pruebas chat[12]: -- got it
Sep 17 17:00:23 pruebas chat[12]: send (^M)
```

- *Gnokii*. Requiere teléfono móvil compatible, habitualmente Nokia, aunque recientemente han sido soportados modelos de otros fabricantes. Permite tanto envío como recepción de mensajes.

```
#!/bin/bash
echo Enviando un mensaje...
echo "texto del mensaje" | gnokii --sendsms 616333444
echo Comprobando mensajes nuevos en las posiciones 1 a 14...
gnokii --getsms ME 1 14
```

- *Esms*. Esms es un pequeño programa que envía mensajes SMS a teléfonos móviles de España a través de Internet. Puede enviar mensajes a móviles de Airtel, Amena y Movistar.

El envío de los mensajes SMS puede realizarse gratuitamente a través de las páginas de los operadores y portales, pero esms presenta una serie de ventajas, como son que no tiene límite de envío, permite enviar tantos mensajes como se desee, que los mensajes largos se fragmentan en varios trozos que se envían uno a uno en orden, que permite almacenar los números de teléfono, y que acepta los mensajes de la entrada estándar. El uso canónico sería:

```
fortune -s love | esms mi_novia
```

Necesita algunos módulos Perl, como LWP, HTML::Parser y HTML::TreeBuilder.

- *SMSSend*. Este programa es un "kit" que permite utilizar múltiples y diferentes pasarelas SMS externas. Realmente el no envía el mensaje, si no que se lo pasa a otra pasarela. Podríamos llamarlo "multicliente", ya que permite actuar de cliente de casi cualquier otra pasarela SMS.

Resulta sencillo ampliar sus posibilidades de utilización de otras pasarelas inicialmente no previstas, con scripts que se ajusten a las necesidades de conexión de dichas pasarelas. Incluye definiciones de alias de teléfonos (como una libreta de direcciones de correo) para no tener que recordar los números.

- *SMSClient*. Muy similar al anterior, básicamente puede establecer comunicación con centros de envío de mensajes para móviles GSM y buscapersnas automáticos, así como pasarelas de correo a SMS de las que ofrecen los operadores de telefonía.

## **5. Seguridad y tolerancia a fallos**

En principio, la red GSM dota de una cierta seguridad a sus comunicaciones e identifica a los terminales móviles mediante un número, de manera que para la mayoría de las aplicaciones resultan suficientes las garantías que la red provee.

Sin embargo, en ocasiones, esta seguridad requiere ser reforzada con sistemas de autenticación que obliguen al usuario a teclear una clave cada vez que vaya a utilizar el servicio, y que esta clave no viaje por la línea de transmisión "en texto claro", siempre considerando que la red GSM no envía los datos directamente así, aunque, a efectos prácticos, al ser relativamente sencillo decodificar esta señal, podemos considerarlo insuficiente.

Los SIM Toolkits permiten insertar una aplicación en la tarjeta SIM, que puede llegar a ser relativamente compleja, capaz de cifrar los datos que vayamos a enviar, actuando como un filtro de nuestro mensaje. En la parte servidora se requiere entonces también un filtro para descifrar en el instante de la recepción, así como para cifrar en el siguiente envío al número del remitente. No todos los teléfonos móviles pueden utilizar aplicaciones incluidas en la tarjeta SIM.

En todo caso, la solución definitiva en cuanto a seguridad para garantizar confidencialidad absoluta, identificación del origen y destino de la comunicación, y no repudio de la actuación, puede no estar lo suficientemente madura para sistemas como el comercio electrónico o los sistemas de pago mediante el móvil.

En cuanto a la tolerancia a fallos debido a la caída de los sistemas donde se alojan los servicios, en ocasiones basta con replicar el servicio entre varios equipos en diferentes redes IP y con diferentes células GSM, de manera que los clientes (IP o GSM) envíen los mensajes a uno de varios equipos, que se encargará de su entrega y procesado.

Algunas pasarelas ya incluyen y tienen en cuenta esta funcionalidad de tolerancia a fallos, mientras que otras mantienen un "punto crítico" que puede hacer que no resulten excesivamente tolerantes.

## **6. Rendimiento**

Para pequeños volúmenes de envío de mensajes, del orden de hasta, por ejemplo, unos 2000 diarios, no es necesario apenas hacer pruebas de rendimiento.

En el peor de los casos, un dispositivo GSM (un teléfono móvil o un modem GSM) puede enviar sin mucho esfuerzo hasta 4 mensajes por minuto, lo cual implica que, teóricamente, podría alcanzar hasta 5760 mensajes diarios.

Para mayores volúmenes, puede ser suficiente con elegir el dispositivo GSM adecuado o incrementar el número de dispositivos GSM. Con un modem GSM que acepte comandos AT, y dependiendo de la red GSM, se pueden alcanzar hasta entre 8 y 10 mensajes por minuto, lo que incrementa la cifra diaria hasta entre 11520 y 14400.

Repartiendo la carga entre varios dispositivos GSM, el volumen aumenta linealmente. El cuello de botella en la entrega de mensajes es la propia red GSM, no el procesador del equipo que actúa de pasarela ni ninguno de sus componentes, por lo que el aumento de dispositivos GSM se refleja en un aumento de rendimiento lineal aún estando conectados todos los dispositivos en el mismo equipo pasarela.

Para mayores volúmenes, o para mayor velocidad en la entrega para volúmenes menores pero urgentes, sería necesario una conexión directa con el centro servidor de mensajes (SMSC), lo cual evita un primer trasiego del mensaje por la red GSM. Los operadores de telefonía móvil suelen ofrecer conexiones directas a sus SMSC para grandes volúmenes de mensajes.

## **7. Coste del envío de mensajes y mensajes gratuitos**

En este apartado hablaré en primera persona, ya que su contenido es fruto de mi experiencia directa.

En el proyecto Alamin hemos recibido muchos mensajes preguntando acerca de la posibilidad de usar esta pasarela para enviar mensajes gratuitos. Sin ánimo de desanimar, la situación no es muy alentadora ahora en España. Los mensajes con garantías grandes de llegar son los que pagas. Los que no pagas, suelen llegar, pero no siempre. Basta con hacer la prueba. Unas cuantas visitas por las páginas de envío de mensajes gratuitos, unos cuantos envíos a un mismo número de teléfono y algunos de ellos no llegarán. Nadie sabrá por qué.

Por hacer un poco de historia, al principio, los mensajes cortos en España fueron gratuitos. Después pasaron a ser un servicio de pago. Más tarde, una operadora de telefonía móvil ofreció, con coste mensual fijo, la primera pasarela email-sms, que funcionó muy bien, esto hay que decirlo, hasta que otra operadora ofreció el mismo servicio de forma gratuita. En este momento ambas dejaron de funcionar correctamente. ¿Qué es considerado aquí correcto? Que TODOS los mensajes que se envían lleguen en un tiempo razonable, y razonable no son 3 o 6 horas, ni mucho menos que no lleguen.

Y esta es la situación actual. El envío de un SMS desde un teléfono móvil en España llega siempre, siempre, salvo problemas puntuales en la red GSM por tareas de mantenimiento o células con configuración incorrecta. El envío de un SMS utilizando una pasarela email-sms o http (web) de las que ofrecen sin coste los operadores de telefonía móvil suele llegar habitualmente, pero no siempre. Si el servicio que utiliza este medio de envío es importante, esta no es la opción adecuada, al margen de las maniobras que ha habido que ir mejorando con el paso de las semanas para que estas pasarelas http puedan seguir funcionando desde programas utilizables desde scripts.

Inicialmente, las páginas web que permitían el envío de un mensaje SMS eran un simple programa cgi que lanzaba el mensaje a través de otro dispositivo GSM o directamente a un centro servidor de mensajes (SMSC).

Pero el envío de un mensaje SMS a través de http ha terminado siendo en algunos casos un largo proceso, comenzando con la solicitud de una página web, aceptación de una cookie, solicitud de una segunda página web y envío de los datos del mensaje, incluyendo una espera de algunos segundos entre la segunda página y el envío de los datos, ya que, de lo contrario, el servidor se percata de que "el usuario ha tardado demasiado poco en escribir un mensaje, puede que no sea una persona, sino un programa", y no procede al envío.

Para terminar, simplemente indicar que las operadoras de telefonía tienen precios especiales para los mensajes enviados a destinos dentro de su propia red y descuentos por volumen de mensajes. Esta suele ser la única forma de conseguir enviar muchos mensajes con cierta seguridad.

Y algo más aún, la conexión directa a un centro servidor de mensajes es quizá la forma más segura de enviar los mensajes, quizá a un coste superior, pero es difícil encontrar esta posibilidad en España. Todos los operadores la tienen, por uno u otro medio, no todos por IP, pero nadie quiere hablar de ellos. Si alguien consigue utilizar una conexión directa a un SMSC con cualquier operador en España, por favor, que se ponga en contacto conmigo. A mi también me gustaría, pero por más que lo he intentado, no he conseguido que ninguna compañía me ofrezca datos técnicos de cómo realizar la conexión. En la página del proyecto Kannel, que se indica en la Bibliografía al final del

documento, hay información de la manera de conectar a diferentes SMSC de diferentes fabricantes, cada uno con su propio protocolo de comunicaciones.

## 8. Demostración práctica

Para terminar la presentación de estos servicios, los asistentes podrán enviar mensajes cortos a un número de teléfono móvil de prueba, donde su mensaje será procesado y recibirán una respuesta también mediante mensaje corto.

Básicamente, el servicio de prueba ofrecido es un simulador de banca GSM, en la que se pueden realizar una serie de operaciones directamente desde el teléfono móvil, como consultar los últimos movimientos, activar y desactivar el seguimiento de operaciones, avisar acerca de la sustracción/pérdida de tarjetas para inhabilitarlas, etc.

Comandos válidos para la demo de banca-gsm:

```
AYUDA
PRUEBA
MOVIMIENTOS
SALDO
TRANS _cantidad_ _cta_destino_
CHEQUES
NOTIFICACION [ABONOS|CARGOS] [ACT|DES]
DENUNCIA [_numero_tarjeta_]
```

## 9. Conclusión

Finalmente, tras la revisión realizada de las pasarelas de mensajería SMS, es importante señalar que se puede obtener un gran beneficio de las posibilidades de expansión de las aplicaciones en el mundo móvil, sencillo y reducido de los mensajes cortos.

Se han analizado varios y diferentes programas para comunicar con la red GSM. Cada uno se ajusta a unas necesidades distintas, aunque en el análisis realmente podemos distinguir entre dos grandes grupos, los programas de acceso al medio (modem, móvil, SMS) y los agentes de transferencia de mensajes cortos, con sus diferencias en cuando a número de dispositivos móviles, colas, prioridades y objetivos).

Muchos programas, con poco esfuerzo, pueden ser extendidos más allá de sus teclados e interactuar con los clientes móviles allá donde quiera que estén.

Cierto es que no existe en la actualidad un estándar para realizar la conexión de las aplicaciones de alto nivel a las pasarelas de mensajería, quizá por la descoordinación de los grupos de trabajo y las diferentes aproximaciones que se han tomado, pero las facilidades de colaboración e intercambio de conocimientos que nos proporciona el Software Libre nos llevan a pensar que no tardando mucho si que existirá esta normalización, y desde la aplicación de alto nivel será sencillo conectar con "cualquier" pasarela, al igual que ahora cualquier aplicación manda un mensaje de correo sin preocuparse de la capa inferior en la que se mueven los mensajes.

Muchos de estos proyectos que aquí han sido presentados precisan ayuda para continuar su desarrollo. Si estás interesado, no dudes en ponerte en contacto con ellos. Recuerda que los desarrolladores que están detrás de estas pasarelas también necesitan "sentirse queridos". Tu experiencia con estos programas, tus informes de error, tus sugerencias y tus críticas no solo son bien recibidas, sino que son necesarias.

## 10. Agradecimientos

A todos los colaboradores y testadores de Alamin, Dan Sotomayor, Brian Russo, Abu Hudzaefah, José Luis Ranz, Jaime Gil, José Luis Hernández, de la Universidad Carlos III de Madrid, Lars Wirzenius por su ayuda con Kannel, Alfredo Martín, y a la gente de las listas por su ánimo.

Al proyecto GArtist por su nuevo logo para Alamin (gartist.sf.net), a SourceForge por sus servicios (www.sf.net), a la gente de Debian por sus enseñanzas (www.debian.org) y a Caja de Guadalajara por realizar pruebas y mantener su pasarela a mi disposición.

A todos los demás desarrolladores de programas libres y más concretamente de pasarelas de mensajería y aplicaciones de envío de mensajes.

A Hispalinux y su gente por la posibilidad de presentar esta ponencia, a Manuel Carro, José Luis Ranz y Pilar Guerrero por sus correcciones y sugerencias.

Gracias.

## Bibliografía

Andrés Seco Hernández, *Alamin GSM SMS Gateway*, Proyecto Alamin, disponible en <http://www.alamin.org> .

Lars Wirzenius and others, *Kannel*, WapIT and 3GLab, disponible en <http://www.kannel.3glab.org> .

Pavel Janik and others, *Gnokii*, Gnokii project, disponible en <http://www.gnokii.org> .

Philippe Andersson, *SMS Link*, CreoScitex Europe, S.A., disponible en <http://smslink.sourceforge.net> .

Mauricio Julio Fernández Pradier y otros, *ESMS*, disponible en <http://esms.sourceforge.net> .

Stefan Frings, *SMSTools*, disponible en <http://www.isis.de/members/~s.frings/smstools/>

Christophe Calmejane, *SMSSend*, disponible en [http://zекiller.skytech.org/smssend\\_menu\\_en.html](http://zекiller.skytech.org/smssend_menu_en.html) .

Angelo Masci, *SMSCClient*, disponible en <http://www.styx.demon.co.uk> .

*ETSI*, European Telecommunications Standards Institute, disponible en <http://www.etsi.org> .