R. Roman, J. Lopez, and J. Zhou, "Protección contra el Spam Utilizando Desafíos a Priori", *V Jornadas de Ingeniería Telemática (JITEL05)*, pp. 375-382, 2005.
NICS Lab. Publications: https://www.nics.uma.es/publications

Protección contra el Spam utilizando Desafíos "a priori"

Rodrigo Román[1], Javier López[1], y Jianying Zhou[2]
[1] E.T.S. Ingeniería Informática, Universidad de Málaga, 29071, Málaga, España
[2] Institute for Infocomm Research, 21 Heng Mui Keng Terrace, Singapore 119613
roman@lcc.uma.es, jlm@lcc.uma.es, jyzhou@i2r.a-star.edu.sg

Abstract*. Spam is considered to be one of the biggest problems in messaging systems. In the area of email Spam, A high number of anti-spam schemes have been proposed and deployed, but the problem has yet been well addressed. In this paper, we introduce a new scheme, called pre-challenge scheme, which avoids problems that exists in other schemes such as delay of service and denial of service. Some new mechanisms are employed to reach a good balance between security against Spam and convenience to email users. In addition, our scheme can be used for protecting other types of messaging systems, such as Instant Messaging (IM) and Blogs, against Spam.*

1 Introducción

El *Spam* (Mensajes electrónicos no solicitados) es considerado como uno de los mayores problemas de los mecanismos de mensajería sobre Internet. Mediante una inversión mínima, es posible tanto inundar con propaganda no deseada a cualquier usuario como aumentar la visibilidad de una página web en motores de búsqueda de forma fraudulenta.

El correo electrónico es el sistema más afectado por este problema. El volumen de *Spam* ("correo basura" en este contexto) recibido por cualquier usuario puede ser tal que el tiempo que se necesita para separar los correos importantes de los correos basura llega a ser prohibitivo. Es por esto por lo que existen multitud de soluciones que tratan de evitar, con mayor o menor éxito, que los correos basura alcancen los buzones de los usuarios.

Específicamente, existe una solución denominada "Desafío/Respuesta" ("Challenge/Response"), en la que se obliga a los emisores a resolver un desafío antes de poder acceder al buzón del receptor. Esta solución posee varios problemas que dificultan su uso, tales como la introducción de un tiempo de espera antes de que el receptor reciba realmente el mensaje (ya que el desafío se envía al emisor cuando el sistema que maneja los mensajes del receptor recibe su correo), la posibilidad de realizar ataques de denegación de servicio (si la dirección del emisor de un mensaje está falsificada), y otros problemas como el manejo de listas de correo y de mensajes de error.

En este artículo proponemos una nueva solución, denominada desafíos "a priori", la cual posee todos los beneficios de los mecanismos de "desafío/respuesta" aplicados al correo electrónico sin ninguno de sus inconvenientes, y que además puede aplicarse a otros problemas de mensajes no solicitados, tales como la mensajería instantánea y los Blogs. Esta solución también incluye mecanismos que pueden aplicarse de forma independiente a los gestores de correo actuales, como el mecanismo de manejo de mensajes de error o la "lista de alerta".

Este artículo se organiza de la siguiente forma: En la sección 2, se introducen cuales son los principales problemas en los sistemas de mensajería. En la sección 3 se analiza el trabajo previamente realizado para proteger a la infraestructura del correo electrónico contra el *Spam*. En la sección 4 se introduce la nueva técnica del desafío "a priori", y en la sección 5 se discuten sus propiedades y aplicaciones en otros sistemas de mensajería. Finalmente, en la sección 6 se concluye el artículo.

2. Problemas en los Sistemas de Mensajería

En los sistemas de mensajería, los principales problemas y al mismo tiempo causas de la aparición de *Spam* son la facilidad de acceso a las direcciones de los usuarios y la falta de autenticación de origen. Es muy sencillo obtener la dirección de un determinado usuario de forma automática, utilizando para ello "robots" (agentes pseudo-inteligentes) que o se encarguen de filtrar páginas web en busca de estas direcciones de contacto, o pregunten a servicios de localización de los propios servicios web. Una vez obtenidas las direcciones, en la mayoría de los casos se puede enviar un mensaje que no contenga información fiable sobre su procedencia real, ya que no existen mecanismos para poder autenticar al origen.

Todos estos problemas pueden encontrarse en los sistemas de correo electrónico, y más concretamente en su protocolo principal, SMTP. El protocolo SMTP fue introducido en 1982 [1], una época en la que mantener la seguridad de la red no suponía ningún problema ya que Internet estaba compuesta únicamente por miles de hosts. Actualmente el

contexto es muy diferente, pero el protocolo sigue siendo el mismo (con ligeras modificaciones [2]).

En el protocolo SMTP, un mensaje consiste simplemente en una cadena de texto que contiene la siguiente información: origen, destino, servidores atravesados, mensaje, y cabeceras extras. El procedimiento para enviar de un mensaje de correo es sencillo: Un servidor de correo (cliente MTA) que maneja los correos del origen, contacta con el servidor de correo destino (servidor MTA) y le envía el mensaje. Es posible que un mensaje tenga que atravesar varios servidores MTA si el destinatario no es directamente accesible.

Sin embargo, un servidor MTA no puede averiguar quién le envió realmente el mensaje, debido a que un usuario malicioso puede tanto falsificar las cabeceras que indican quién fue el origen, como controlar o manipular un servidor de correo para que oculte quién fue el cliente MTA que envió el mensaje inicialmente. Como resultado, un Spammer (quien envía el *Spam*) puede enviar una cantidad ilimitada de *Spam* a cualquier usuario, y éste no podrá defenderse contra este ataque ya que no dispondrá de la información necesaria para poder bloquear el acceso de *Spam* a su cuenta de correo.

3. Trabajo Previo

El protocolo SMTP es un estándar que sirve como pilar a la infraestructura de correo electrónico de Internet. Como resultado, sería necesario planear una migración lenta y controlada (como está ocurriendo con IPv6) en caso de que el protocolo fuese cambiado. Por lo tanto, la mayoría de las investigaciones en el área de la lucha contra el *Spam* se centran en utilizar la información contenida en los mensajes (por ejemplo cabeceras) o en desarrollar aplicaciones que funcionen sin modificar el estándar.

Una cabecera capaz de proporcionar información útil es "Received:". Esta cabecera ofrece una lista de los clientes y servidores MTA que han reenviado el mensaje a través de Internet, por lo que se puede comprobar si uno de esos servidores es una fuente de *Spam*. Existen algunos proyectos que tratan de clasificar este tipo de servidores [3,4]. No obstante, es posible bloquear clientes y/o servidores MTA inocentes.

Otra cabecera cuya información puede aprovecharse es la dirección del destinatario. Ésta dirección puede ampliarse con políticas de acceso o passwords. En sistemas basados en políticas de acceso [5], una política se codifica dentro de la dirección del destinatario, y si esta política no se cumple al llegar al servidor MTA, el mensaje de desecha. En los sistemas basados en passwords [6 – 8], la dirección del destinatario se amplía con una secuencia de caracteres que actúan como una password, la cual solo puede ser obtenida mediante una prueba de un gasto computacional [11]. Estas soluciones funcionan bien en algunos escenarios (por ejemplo cuando se utiliza una dirección de correo en entornos automatizados como foros de discusión), pero las direcciones de correo ampliadas son muy complejas, y son difíciles de recordar y utilizar para un ser humano.

Existen varios trabajos que se centran en analizar el contenido de un mensaje utilizando técnicas de inteligencia artificial (IA) y de análisis estadístico [9,10]. Como resultado de estos análisis se asigna una "puntuación" que distingue si un mensaje de correo proviene de un usuario legítimo o de un spammer. Sin embargo, estas técnicas pueden ocasionar falsos positivos (cuando un correos real es tratado como *Spam*) y falsos negativos (cuando un spammer modifica el formato de sus correos y éstos ya no se consideran *Spam*).

Otras soluciones existentes son las técnicas de micropago (micropayment), "desafío/respuesta" (Challenge/Response) y ofuscación. Los esquemas de micropago [11 – 14] evitan que los spammers envíen millones de correos basura, al ralentizar a los clientes MTA pidiéndoles calcular una función matemática compleja para poder comunicarse con el servidor MTA. Con todo, esta técnica es difícil de aplicar a dispositivos con recursos reducidos (como teléfonos móviles).

En las técnicas de "desafío/respuesta" [15, 16], siempre que un servidor MTA recibe un correo de un origen desconocido, éste responde automáticamente con un desafío. Una vez se responde al desafío, los correos procedentes de ese origen podrán alcanzar al destinatario. Sin embargo, estas técnicas introducen nuevos problemas (no solucionados hasta ahora), tales como la introducción de un tiempo de espera en la recepción de los mensajes o la posibilidad de ataques de denegación de servicio.

En los esquemas de ofuscación, las direcciones de correo se muestran al público de forma ofuscada (p. ej. nombre GUION apellido ARROBA servidor PUNTO dominio), y los usuarios que deseen utilizar esa dirección de correo deben "traducirla" primero. Un problema en esta solución es que las combinaciones para ofuscar una dirección de correo son limitadas, y una vez que la dirección esta capturada, el spammer puede enviar correos basura al destinatario sin mayor problema.

4. Propuesta de una nueva técnica: Desafíos "a priori"

4.1 Introducción

La técnica de desafios "a priori" se basa en los mecanismos de "desafío/respuesta", en el sentido de que ambas imponen al usuario que envía el mensaje un desafío que debe ser resuelto antes de acceder al correo del usuario destino. La particularidad de la

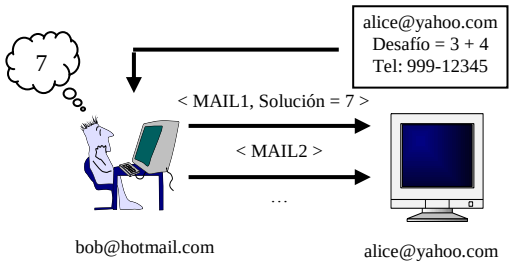

Fig. 1: Esquema básico del Desafío "a priori"

técnica de desafíos "a priori" se encuentra en *cuando* se accede al desafío: Cuando un usuario desee enviar un correo, éste recogerá tanto la dirección de correo del destinatario como un desafío asociado a esa dirección de correo, al mismo tiempo. Una vez se resuelva el desafío, éste se incluirá dentro del mensaje (ver Fig. 1).

Cuando un correo que proviene de un remitente desconocido alcanza un buzón protegido, el servidor de correo comprobará que el mensaje incluye la solución al desafío. Si es así, el mensaje es admitido dentro del buzón, y el remitente se añade a una "*lista blanca*" (White-List) para que sus correos no necesiten incluir más el desafío resuelto (aún en el caso de que ese desafío sea cambiado).

El objetivo de esta técnica es el de comprobar que el remitente es realmente un ser humano. Esto es así debido a que los spammers utilizan programas automáticos tanto para obtener las direcciones de correo contenidas en páginas web o servidores de correo como para enviar el *Spam*. Sin embargo, es difícil para estos programas recoger un desafío asociado a una dirección de correo determinada, y les es aún más difícil tener el conocimiento semántico suficiente para resolver el desafío una vez obtenido. Por lo tanto, cuando un correo basura llegue a un buzón protegido, éste será descartado al no incluir la solución al desafío asociado a ese buzón.

En comparación con los esquemas de "desafío/respuesta", la técnica del desafío "a priori" conserva sus beneficios sin incluir sus defectos:

- En la técnica del "desafío/respuesta", existe un tiempo de espera para que el destinatario de un mensaje obtenga la solución a su desafío. Por otro lado, en el desafío "a priori", el desafío se encuentra disponible junto a la dirección del receptor, por lo que el remitente puede resolver el desafío y enviar su mensaje al receptor directamente.

- Si un spammer falsifica la dirección del remitente en sus correos, ese remitente recibirá los desafíos en caso de que el buzón del receptor esté protegido con la técnica del "desafío/respuesta", sufriendo un ataque de denegación de servicio [17]. Esto no ocurriría en el desafío "a priori", ya que los mensajes inválidos no provocan respuesta alguna.

- Un sistema de "desafío/respuesta" sólo puede funcionar con listas de correo si se incluyen algunas reglas específicas para cada lista de forma manual. En cambio, un sistema de desafío "a priori" puede manejar listas de correo y procesar mensajes de error sin ningún problema.

Otro beneficio del desafío "a priori" es la protección continuada que ofrece ante "robots" recolectores de direcciones. Una dirección de correo capturada por uno de estos programas no tiene utilidad a menos que se resuelva el desafío asociado a esa dirección. Y aún en el caso de que el desafío fuera resuelto, y la dirección fuese vendida (p. ej. en colecciones de CD), el dueño de la dirección puede modificar su desafío, haciendo que la combinación <email, solución> sea inútil.

La técnica del desafío "a priori" se integra de forma sencilla dentro de la infraestructura de correo actual, porque no obliga a cambiar ninguno de los protocolos de correo existentes (como POP3, IMAP, y SMTP). Puede incorporarse como un "plugin" dentro de cualquier servidor de correo, cuyas tareas serían las de proveer y mantener una serie de listas y reglas (secciones 4.3, 4.4 y 4.5) y las de interactuar con los dueños de los buzones de correo para tareas de mantenimiento (actualizar desafío y solución, modificar manualmente ciertas listas).

4.2 Obtención del Desafío

Cada cuenta de correo tiene un desafío asociado, y son los propietarios de esas cuentas quienes crean sus propios desafíos. Cada desafío puede ser actualizado en cualquier momento y tantas veces como su dueño desee. El grado de complejidad de los desafíos puede oscilar entre palabras o preguntas sencillas hasta sistemas complejos que solo un humano podría resolver [18].

En la mayoría de los casos un desafío se encuentra justo al lado de su dirección de correo asociada, de tal forma que cuando un posible remitente accede a la dirección de correo también puede recoger y resolver el desafío de forma inmediata. Sin embargo, en ciertos casos, puede que éste no se encuentre disponible de forma directa. En esos casos debe incluirse una URI que apunte a donde podría obtenerse ese desafío.

Ya que el desafío no se encuentra limitado a ofuscar una dirección de correo, la cual tiene una estructura fija (nombre, dominio), el usuario posee una mayor libertad en su creación. Cuando se almacena dentro de una página web, el desafío puede aprovechar el contenido que lo rodea (información personal, aspecto visual de la web). En entornos estáticos (p. ej. una tarjeta de visita) la solución al desafío puede

incluirse directamente, ya que no hay peligro de que un spammer acceda a esa solución.

Finalmente, otra solución para obtener el desafío es la utilización de un servicio "majordomo" [21], donde un posible remitente pide a un servidor de correo cual es o donde está localizado el desafío de un usuario determinado. Eso sí, para prevenir que los spammers utilicen este servicio para recolectar direcciones de correo, el servicio debería devolver un desafío automáticamente generado para cada usuario no existente.

4.3 Estructuras de Datos

El desafío "a priori" requiere de ciertas estructuras de datos para poder funcionar. Las dos estructuras más importantes son el desafío "per se" (o una URI donde pudiera encontrarse) y la solución a ese desafío. Utilizando estas estructuras sería posible proporcionar el desafío actual a quienes lo necesiten y comprobar si la solución a un desafío es la correcta. Aparte, deben almacenarse las soluciones de antiguos desafíos.

Otras estructuras necesarias son la "*lista blanca*" (white-list), la "*lista de respuesta*" (reply-list) y la "*lista de alerta"* ("warning-list", específicamente diseñada para el desafío "a priori"). Cada una de esas estructuras contiene una lista de direcciones de correo y, adicionalmente, una fecha ("timestamp") para guardar el tiempo que una dirección de correo puede estar dentro de la lista.

"Lista Blanca". Los mensajes procedentes de remitentes incluidos en la *lista blanca* son inmediatamente admitidos dentro del buzón del usuario protegido, sin necesidad de comprobar la solución al desafío. Algunos remitentes pueden ser incluidos dentro de esta lista de forma manual si el usuario ya los conoce, evitando de esa forma que esos remitentes deban responder un desafío si el usuario ya confía en ellos.

"Lista de Respuesta". Esta lista contiene las direcciones de correo de aquellos usuarios a los que el propietario del buzón protegido ha enviado un correo, y aún no ha recibido respuesta. Es uso de esta lista se justifica de la siguiente forma: Si el propietario del buzón desea establecer una comunicación con otro usuario, sería innecesario requerirle una solución a un desafío.

"Lista de Alerta". La *lista de alerta* contiene las direcciones de correo de aquellos remitentes que han enviado un mensaje incluyendo una solución a un desafío antiguo. Debido a que es posible que un remitente solo haya tenido acceso a un desafío antiguo, la técnica del desafío "a priori" envía una respuesta automática a estos remitentes incluyendo el desafío actual. Cuando se incluye un remitente en esta lista, se indica que no debe recibir ninguna respuesta automática más en el futuro. En caso de cambiar el desafío actual, esta lista se vacía completamente.

4.4 Niveles de Seguridad

El desafío "a priori" puede ser configurado para trabajar en dos niveles de seguridad, *nivel alto* y *nivel bajo*. La diferencia entre ambos niveles de seguridad se encuentra en la forma de consultar la *lista de respuesta*.

El desafío "a priori" empieza trabajando en el nivel alto de seguridad. Este nivel implica que todas las consultas a la *lista de respuesta* se realizan buscando un par <usuario, dominio>, y que todas las coincidencias serán eliminadas. Por ejemplo, cuando se recibe un correo de bob@hotmail.com, se comprueban los campos "De:" y "Responder A:" del mensaje, y la *lista de respuesta* será consultada con el par <bob, hotmail.com>.

Por otro lado, en el nivel bajo de seguridad todas las consultas a la *lista de respuesta* se realizarán mediante el par <*, dominio>. De esta forma, si se recibe un correo de bob@hotmail.com, la *lista de respuesta* será consultada con el par <*, hotmail.com>.

La presencia de estos niveles de seguridad se debe a la existencia de cuentas de correo cuyas direcciones son distintas para enviar el correo y para recibir el correo. Esto suele ocurrir con las listas de correo, como se verá en la sección 5.1.

4.5 Funcionamiento del Desafío "a priori"

Ahora se procede a la explicación del funcionamiento de la arquitectura del desafío "a priori". Supondremos que existe un usuario B (remitente) que quiere enviar un mensaje al usuario A (destinatario), asumiendo (para simplificar la explicación) que el usuario A está utilizando el desafío "a priori" y que B no lo utiliza.

1. El servidor de A comprueba que la dirección de B se encuentre dentro de la *lista blanca*. En ese caso, el mensaje alcanza el buzón de A, y B recibe una confirmación si es el primer mensaje que envía a A.

2. En otro caso, si B se encuentra en la *lista de respuesta*, el correo alcanza el buzón de A y B es añadido a la *lista blanca*. Aquí hay que puntualizar que la consulta a la *lista de respuesta* es distinta dependiendo del nivel de seguridad al que esté funcionando el sistema (ver sección 4.4). En el caso de que el nivel de seguridad sea alto, la dirección de B se borra de la *lista de respuesta* (ya que A recibió la respuesta que esperaba de B).

3. En otro caso, se comprueba si el mensaje incluye la solución al desafío actual. Si es así, el correo alcanza el buzón de A, y B es

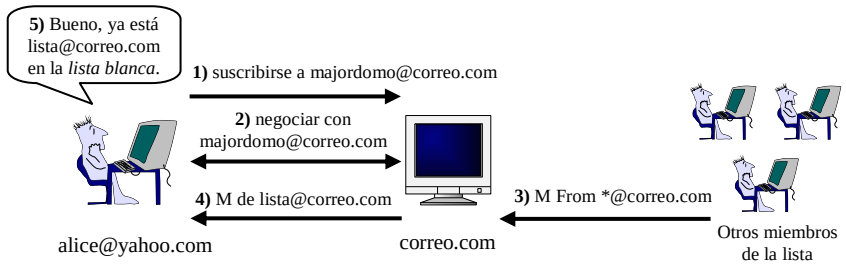

Fig. 2: Proceso de suscripción a una lista de correo.

añadido a la *lista blanca*. Adicionalmente, B recibe un correo de confirmación.

4. En otro caso, si el mensaje incluye una solución a un desafío antiguo, se responde a B con un correo que incluya el desafío actual, pero solo si la dirección de B no se encuentra en la *lista de alerta*. Si se ha enviado la respuesta con el desafío actual, la dirección de B se añade a la *lista de alerta*.

5. En otro caso, el correo se desecha, sin enviar ningún tipo de respuesta a B. El problema de desechar un correo proveniente de un usuario legítimo se discute en la sección 5.3.

En este punto, cabe anotar que desechar un correo no significa que el usuario del buzón no pueda acceder a él. La técnica del desafío "a priori" puede configurarse para incluir una "puntuación" a los correos basura, de tal forma que el usuario pueda acceder a esos correos a través de su cliente de correo si así lo desea.

4.7 Escenarios

Existen dos posibles escenarios en el caso de que un spammer quiera enviar su correo basura a un usuario que esta protegido por la técnica del desafío "a priori".

Escenario 1. El spammer solo dispone de la dirección de correo del destinatario, pero no de su desafío. En ese caso, todo el correo basura enviado a un buzón protegido será automáticamente desechado, debido a la falta de una solución (tanto antigua como moderna) dentro del mensaje.

Escenario 2. El spammer solo dispone de la dirección de correo del destinatario, pero se identifica como un usuario existente dentro de la *lista blanca*, debido a los problemas de autenticación existentes en la infraestructura mundial de correo electrónico. Todas las técnicas que utilizan listas blancas comparten este inconveniente, pero no es un gran problema dado que un spammer debería encontrar una dirección de correo "valida" por cada una de las direcciones a las que quiere enviar *Spam*. Y para millones de direcciones, esto no es rentable.

Eso sí, podría parecer que un spammer solo necesita de un pequeño esfuerzo (resolver un desafío) para enviar todo el correo basura que desee a una dirección de correo en particular. También podría ocurrir que un grupo de spammers intercambiasen las soluciones de los desafíos que conocen para simplificar su tarea. Sin embargo, lo que los spammers persiguen es enviar millones de mensajes a millones de destinatarios. Y los desafíos son distintos por cada destinatario, y solo pueden ser resueltos por un ser humano. De esta forma, la tarea de espiar la red o utilizar mano de obra barata para obtener las soluciones a los desafíos no es rentable.

5. Discusiones

En esta sección se discuten como la técnica del desafío "a priori" funciona para usuarios de listas de correo, y también los problemas existentes en el acceso a un desafío. También se discute sobre como manejar adecuadamente mensajes de error, y de cómo aplicar nuestra técnica a sistemas como mensajería instantánea (IM) o blogs.

5.1 Manejo de Listas de Correo

Todas las listas de correo [19 – 21] poseen un mecanismo de registro similar: cuando un usuario desea registrarse dentro de una lista, ésta le envía un desafío para comprobar que quien ha enviado el mensaje es un ser humano. Este comportamiento hace imposible el manejo automático de listas de correo en sistemas de "desafío/respuesta".

Afortunadamente, existe una solución a este problema para la técnica del desafío "a priori", en la forma de los niveles de seguridad. Ya que todos los correos procedentes de una lista de correo pertenecen a un mismo dominio, es posible utilizar el nivel de seguridad bajo (ver sección 4.4) en el momento de empezar el registro dentro de la lista. De esta forma, todos los mensajes que se reciban durante el proceso de registro (desafíos incluidos) y que tengan una coincidencia en la *lista de respuesta* serán admitidos e incluidos dentro de la *lista blanca*. Finalmente, una vez que se reciba el primer correo de la lista, el usuario puede volver al nivel de seguridad alto (ver Fig. 2).

El riesgo de que un spammer entre en la *lista blanca* de un usuario mientras éste se encuentra en el nivel de seguridad bajo es pequeño, ya que la dirección de correo del spammer debe tener el mismo dominio que

la de la lista de correo, y además un usuario se suele suscribir a muy pocas listas de correo al año.

Además, el usuario puede configurar el sistema para incluir las direcciones válidas en una *lista blanca* temporal cuando se funcione en el nivel de seguridad bajo, de tal forma que cuando el sistema pase al nivel de seguridad alto el usuario decida que direcciones de correo deben añadirse (manualmente) a la *lista blanca*.

5.2 Acceso al Desafío

Es evidente que existe un problema de disponibilidad si el desafío no se publica junto a su dirección de correo asociada. Si un usuario no puede obtener el desafío de otro usuario, sea porque acceder al desafío o al lugar que contiene el desafío no sea posible (p. ej. el remitente no puede acceder a Internet, o la página web que contiene el desafío esta bajo un ataque de denegación de servicio), es imposible que sus correos puedan alcanzar ese buzón protegido (sin ser marcados como *Spam*).

Debido a esa razón, es conveniente proporcionar tanto el desafío como una URI que apunte a donde ese desafío pueda obtenerse. De esta forma, si la URI no funciona, la solución al desafío, aunque éste no sea el actual, puede utilizarse para enviar un mensaje al destinatario (Si el desafío no es el actual el remitente recibirá un mensaje con el desafío que se está utilizando actualmente).

Finalmente, existe un problema de disponibilidad que es común tanto para los sistemas de desafíos "a priori" como para los sistemas de "desafío/respuesta". Un desafío que sea sencillo para un usuario concreto puede ser imposible de resolver para otro tipo de usuarios (por ejemplo, un usuario ciego no será capaz de resolver un desafío basado en imágenes).

5.3 Manejo de Falsos Positivos

Uno de los mayores problemas existentes en el desafío "a priori" ocurre cuando los correos de un usuario humano son desechados (sin enviar respuesta alguna) por el servidor de correo del destinatario protegido, al no incluir la solución a un desafío. Esto evita tanto que suba el tráfico en Internet como los ataques DoS causados por respuestas a los mensajes de spammers, pero a su vez un usuario que no sepa que un destinatario está protegido por un sistema de desafío "a priori" no será capaz de saber si sus mensajes han llegado a su destino o no.

Una posible solución consiste en definir un prefijo estándar para direcciones de correo protegidas por el mecanismo de desafíos "a priori". De esta forma, un remitente sabría que debe resolver un desafío para acceder al buzón del destinatario, y que si su primer mensaje es aceptado recibirá una confirmación.

Existe una solución alternativa en caso de que el desafío "a priori" se encuentre implementado en los servidores de correo. En esta solución, el usuario que envíe un mensaje no valido a un buzón protegido podrá recibir un mensaje de error gracias al protocolo de negociación de SMTP, sin que eso signifique un coste adicional para el servidor MTA que recibe el mensaje. Este protocolo funciona como sigue:

1. El cliente MTA del lado del remitente contacta con el servidor MTA del destinatario. Después de intercambiar mensajes de control, el servidor MTA permite al cliente MTA enviar el contenido del mensaje.

2. El cliente MTA envía el contenido del mensaje terminando con una simple ".". Después, el servidor MTA comprueba si el mensaje debe ser aceptado o rechazado. Si es rechazado, el cliente MTA recibe el mensaje "*554 Transaction failed* " (Transacción falllida).

3. Si la negociación fracasa, el cliente MTA genera un correo que incluya el mensaje original y el error enviado por el servidor MTA. Ese correo se envía al remitente original, en el caso de que este cliente MTA no maneje sus mensajes.

Cuando el servidor MTA comprueba si el correo es válido (paso 2), puede inspeccionar las cabeceras o contenidos del mensaje en busca de la solución al desafío del destinatario, ya que en este punto dispone de toda la información necesaria para realizar ese chequeo (origen, mensaje, destino). Si no hay solución al desafío, el servidor MTA puede devolver *"554 Transaction Failed: Solución al desafío errónea"* (indicando donde encontrar el desafío actual), y el cliente MTA generará un correo de error que incluirá automáticamente en el buzón del remitente.

5.4 Manejo de Mensajes de Error

Durante el curso del protocolo de negociación de SMTP, si un mensaje no puede llegar a su destinatario el cliente MTA debe enviar al remitente un correo que incluya las causas del error. Esos errores pueden ocurrir tanto por problemas de la cuenta destino (p. ej. cuota excedida) o por problemas administrativos o de seguridad (p. ej. solución de desafío no incluida).

Si el correo que avisa del error es generado por el cliente MTA que implementa el mecanismo de desafío "a priori", esto no supone ningún problema, ya que ese correo se incluye automáticamente en el buzón del usuario. Sin embargo, existe un problema en el caso de que el mensaje de error sea enviado al remitente original a través de un servidor MTA.

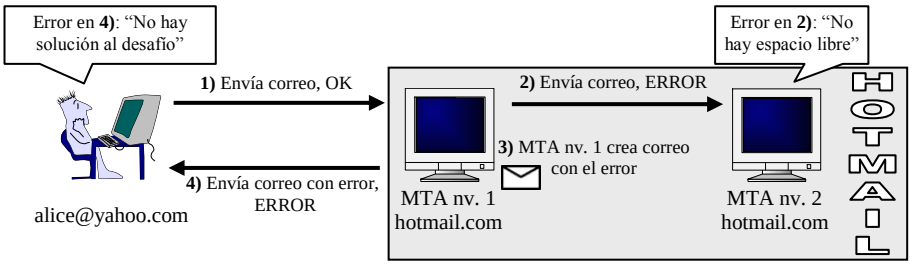

Fig. 3: Problemas en el manejo de mensajes de error

Un ejemplo de este problema puede verse en la Fig. 3. En el ejemplo, el error se produce en la MTA de nivel 2, por lo que la MTA de nivel 1 debe generar el mensaje de error. No obstante, el mensaje de error se envía a la MTA que maneja los mensajes del remitente original, y la MTA de nivel 1 no incluye la solución a ningún tipo de desafío – es, a fin de cuentas, una máquina.

Este problema puede resolverse gracias a dos principios: Primero, los mensajes de error pueden ser identificados gracias a la cabecera "message/delivery-status", e incluyen el mensaje original del remitente. Segundo, todos los correos tienen un número único que los identifica en la cabecera "message-ID".

De esta forma, cuando un mensaje de error llega a un buzón protegido, éste mensaje es aceptado si y solo si tanto la dirección del destinatario original como el número ID del mensaje original se encuentran en la *lista de respuesta*. Es por tanto necesario incluir el número ID de los mensajes dentro de la *lista de respuesta* si se desean manejar correctamente los mensajes de error. Dado que para obtener este número ID sería necesario interceptar un correo en su viaje hacia el servidor MTA, los spammers no pueden sacar provecho de este mecanismo.

Hay que hacer notar que un spammer podría realizar un ataque DoS a un buzón no protegido si enviara mensajes a un servidor con una dirección de origen falsificada, en el escenario expuesto en la Fig. 3. Esta situación se evita incorporando los mecanismos del desafío "a priori" en las MTAs de nivel 1.

5.5 Protección de Otros Sistemas de Mensajería

5.5.1 *Spam* de Mensajería Instantánea (IM)

Los sistemas de mensajería instantánea (IM) proporcionan servicios de comunicación simple (texto) o compleja (audio/video) entre dos extremos, y servicios de localización entre un grupo de usuarios denominados "lista de amigos".

Un usuario debe registrarse primero dentro de un servicio de mensajería instantánea para poder contactar con otros usuarios. Además, los usuarios tienen mecanismos que les permiten comprobar quién quiere comunicarse con ellos, y pueden prohibir el acceso a usuarios sospechosos. Por esa razón, el *Spam* no es un problema común en estos sistemas.

No obstante, existen ciertos servicios de IM que sufren el problema del *Spam*, como el servicio World-Wide Pager de ICQ [22]. Estos servicios permiten que usuarios anónimos envíen un mensaje instantáneo, utilizando un formulario HTML, a cualquier usuario. Como hay autenticación de origen, existen programas automáticos que permiten enviar *Spam* a usuarios de IM en tiempo real.

Dado que estos servicios de IM están incluidos en páginas web, la técnica del desafío "a priori" puede ser utilizada, permitiendo a los usuarios ofrecer un desafío a aquellos que quieran enviarles un mensaje. De esta forma, el *Spam* de mensajería instantánea no sería rentable, tal y como se ha explicado en este artículo.

5.5.2 Blog *Spam*

Los Weblogs (o simplemente Blogs) son un tipo de aplicación web en el que uno o más usuarios escriben información (no modificable) que más tarde podrá ser accedida por otros usuarios. Una de las características más interesantes de los blogs es que permite que los visitantes escriban comentarios sobre la información incluida en cualquier parte del blog.

Sin embargo, es posible que un blog reciba *Spam*, en forma de un comentario corto que incluye un enlace a una página web, la cual suele anunciar un producto fraudulento. El objetivo de este tipo de *Spam* es el de aumentar la importancia de esas páginas web en buscadores como google, y provocan que los usuarios legítimos tengan dificultades en leer comentarios que merezcan la pena.

Una solución desarrollada por google [23] consiste en incorporar automáticamente a la etiqueta HREF de HTML la opción NOFOLLOW, de tal forma que los enlaces existentes dentro de un comentario no servirán a la hora de contar la prioridad de la página web enlazada. Sin embargo, es posible que esto no acabe con el *Spam* debido a la existencia de Blogs sin proteger y a los bajos conocimientos técnicos de los spammers.

La técnica del desafío "a priori" puede utilizarse también para proteger a los blogs del *Spam*. Si fuera

necesario responder a un desafío antes de poder enviar un comentario, los programas automáticos de envío de *Spam* dejarían de funcionar (como se ha discutido durante todo este articulo), y los comentarios se verían libres de *Spam*.

6. Conclusión

En este artículo, se ha presentado una técnica denominada desafio “a priori” para controlar el *Spam* del correo electrónico, basada en los mecanismos de “desafío/respuesta” pero sin ninguno de sus problemas, y capaz de proteger otros sistemas de mensajería como la Mensajería Instantánea y los Blogs.

Esta técnica también puede ser utilizada conjuntamente con otras soluciones contra el *Spam*. Así se deja la puerta abierta a otras soluciones como las de análisis de contenido. Al mismo tiempo, es posible integrar nuestra técnica con sistemas de autenticación de origen como DomainKeys [24] o IBE [25], evitando los problemas de autenticación que surgen en el manejo de la *lista blanca*.

Referencias

[1] J. Postel. *Simple Mail Transfer Protocol*. RFC 821, Internet Engineering Task Force, Agosto 1982.

[2] J. Klensin. *Simple Mail Transfer Protocol*. RFC 2821, Internet Engineering Task Force, Abril 2001.

[3] RBL. http://mail-abuse.org/rbl/.

[4] SBL. http://spamhaus.org/.

[5] J. Ioannidis. *Fighting Spam by Encapsulating Policy in Email Addresses*. In Proceedings of NDSS'03 (Network and Distributed System Security), Febrero 2003.

[6] E. Gabber, M. Jakobsson, Y. Matias, and A. Mayer. *Curbing Junk E-Mail via Secure Classification*. In Proceedings of FC'98 (Financial Cryptography), pages 198--213, Febrero 1998.

[7] R. J. Hall. *How to Avoid Unwanted Email*. Communications of the ACM, 41(3):88-95, Marzo 1998.

[8] L. F. Cranor and B. A. LaMacchia. *Spam!*. Communications of the ACM, 41(8):74--83, Agosto 1998.

[9] M. Sahami, S. Dumais, D. Heckerman, and E. Horvitz. *A Bayesian Approach to Filtering Junk Email*. In Proceedings of AAAI'98 Workshop on Learning for Text Categorization, Julio 1998.

[10] P. Cunningham, N. Nowlan, S. J. Delany, and M. Haahr. *A Case-Based Approach to Spam Filtering that Can Track Concept Drift*. In Proceedings of ICCBR'03 Workshop on Long-Lived CBR Systems, Junio 2003.

[11] C. Dwork and M. Naor. *Pricing via Processing or Combatting Junk Mail*. In Proceedings of Crypto'92, pages 139--147, Agosto 1992.

[12] C. Dwork, A. Goldberg, and M. Naor. *On Memory-Bound Functions for Fighting Spam*. In Proceedings of Crypto'03, pages 426--444, Agosto 2003.

[13] M. Abadi, A. Birrell, M. Burrows, F. Dabek, and T. Wobber. *Bankable Postage for Network Services*. Proceedings of the 8th Asian Computing Science Conference, Mumbai, India, Diciembre 2003.

[14] Penny Black Project, Microsoft Research. http://research.microsoft.com/research/sv/PennyBlack/.

[15] SpamArrest. http://spamarrest.com/faq/.

[16] SpamCap http://www.toyz.org/cgi-bin/wiki.cgi?SpamCap.

[17] J. Mirkovic, J. Martin, and P. Reiher. *A Taxonomy of DDoS Attacks and DDoS Defense Mechanisms*. University of California, Computer Science Department, Technical Report #020018.

[18] L. von Ahn, M. Blum, N. J. Hopper, and J. Langford. *CAPTCHA: Using Hard AI Problems for Security*. In Proceedings of Eurocrypt'03, pages 294--311, Mayo 2003.

[19] Ezmlm Mailing List. http://www.ezmlm.org/.

[20] Mailman Mailing List. http://www.list.org/.

[21] Majordomo Mailing List. http://www.greatcircle.com/majordomo/.

[22] ICQ Pager. http://www.icq.com/panels/messagepanel/.

[23] Google Blog. *Preventing Blog Spam (Enero 18,2005)*. http://www.google.com/googleblog/2005/01/preventing-comment-spam.html

[24] Yahoo DomainKeys. http://antispam.yahoo.com/domainkeys/.

[25] D. Boneh and M. Franklin. *Identity Based Encryption from the Weil Pairing*. Crypto'01, pages 213-229, Agosto 2001.