

di sistema PCdisk che permette lo scambio dei file e attiva finalmente la Janus. Questa effettuerà il suo boot normalmente, avendo l'accortezza di inserire nel Config.sys

```
DEVICE=JDISK.SYS
```

e nell'Autoexec.bat le ultime quattro righe del listato mostrato in figura 2.

Le altre caratteristiche

Dicevamo che la utility presentata non fa SOLO l'aggiornamento della data e dell'ora della Janus, ma anche altro. Innanzitutto è necessario scorporare dalla startup-sequence del vostro disco di boot tutta la parte relativa all'attivazione della Bridge Board. Infatti la scheda non deve partire prima che Amiga non abbia provveduto a scrivere i due filetti di cui sopra. Dunque quello che fa il mio programma è di montare correttamente l'hard disk JHO:. Oltre a questo potremo inserire in coda quante altre attivazioni possibili, come ad esempio il lancio in background di un virus killer proprio in questi tempi tanto «infettosi». Ancora, ho previsto una password per l'attivazione dell'hard disk in modo da tenere lontani curiosi non troppo craker. Infatti la protezione è abbastanza sempliciotta, ma comunque è più che sufficiente per scoraggiare attacchi di ordinaria amministrazione. Certo se un esperto Amiga decidesse di mettere mani sui vostri dati, non ci sarebbero affatto password che tengano: bypassando completamente il programma si riesce comunque ad effettuare una attivazione a mano. Ma bisogna sapere dove mettere le mani... Personalmente sono anche andato oltre ma non posso dirvi pubblicamente il successivo livello di protezione da me implementato altrimenti che segreto sarebbe? I più svegli comunque non impiegheranno molto a capirlo...

Sempre in merito alla password, onde evitare attacchi a colpi di file-zap nel sorgente, ho effettuato una piccola (e banale) crittazione in modo da non lasciare visibile nel compilato il nome adoperato. Nel programma presentato la PW è stata fissata a "pippo.3" (minuscolo) e per cambiarla sarà sufficiente smanettare nei codici ASCII lì usati. Per inserirla alla richiesta «Enter Password» non bisognerà terminare con un classico [Return] ma con il CTRL- \ che in Amiga equivale all'EOF.

Commento al listato

Il programma Auto Date & Time è scritto in un C abbastanza elementare. Tutte le operazioni, infatti, sono effettuate senza ricorrere a sporchi trucchetti ottimizza-

```

/*****
 *
 *      Auto   Date & Time
 *
 *      -----
 *      (C) 1989   ADP Software
 *      -----
 *****/

#include "exec/types.h"
#include "libraries/dosextens.h"
#include "libraries/dos.h"
#include "stdio.h"

extern struct FileHandle *Open();

main()
{
    /* definizioni variabili e array */

    int i,j,k,c;
    struct FileHandle *infh;
    struct FileHandle *outfh;
    char buffin[BUFSIZ],buffout[BUFSIZ],p[10];

    /* richiesta e controllo password */

    c=0;
    printf("\nEnter password: ");
    while ((p[c] = getchar())!=EOF && c<10) c++;
    for (i=0; i<c; i++) printf("\b\b");
    printf("\n");
    c=5;
    if (p[0] == 80+32    /* sta per 'p' */
    && p[1] == 73+32    /* sta per 'i' */
    && p[2] == 70+10+32 /* sta per 'p' */
    && p[3] == 85-c+32  /* sta per 'p' */
    && p[4] == 74+c+32  /* sta per 'o' */
    && p[5] == '.'      /* sta per '.' */
    && p[6] == 51)      /* sta per '3' */

    {
        /* password giusta: attivazione Janus */

        Execute( "Binddrivers",0,0);
        Execute( "setclock opt load",0,0);

        /* scarica in un file temporaneo
           date & time Amiga */

        Execute( "date > TimeAmiga",0,0);

        /* conversione di formato dell'ora
           (sostituisce ai ':' un '.') */

        infh = Open("TimeAmiga",MODE_OLDFILE);
        outfh = Open("TimePC",MODE_NEWFILE);
        if (infh == 0 || outfh == 0) Exit(0);
        k=0;
        Read(infh,buffin,BUFSIZ);
        for (i=12; buffin[i]!=' '; i++);
        for (j=i+1; j<i+9; j++)
            {
                if (buffin[j]!=':') buffin[j]='.';
                buffout[k]=buffin[j];
                k=k+1;
            }
        buffout[k]='\r';
        Write(outfh,buffout,k+1);
    }
}

```

```

Close(outfh);
Close(infh);

/* conversione di formato della data
( '.' al posto di '-' e traduzione
del mese in numero) */

outfh = Open("DatePC",MODE_NEWFILE);

for (i=1; buffin[i]!=' '; i++);
buffout[0]=buffin[i+1];
buffout[1]=buffin[i+2];
buffout[2]=buffout[5]='.';
buffout[3]='0';
buffout[6]=buffin[i+8];
buffout[7]=buffin[i+9];
switch (buffin[i+6]) {
  case 'n': if (buffin[i+5] == 'a') buffout[4] = '1';
            else buffout[4] = '6';
  case 'b': buffout[4] = '2';
  case 'r': if (buffin[i+5] == 'a') buffout[4] = '3';
            else buffout[4] = '4';

  case 'y': buffout[4]='5';
  case 'l': buffout[4]='7';
  case 'g': buffout[4]='8';
  case 'p': buffout[4]='9';
  case 't': buffout[3]='1';
            buffout[4]='0';
  case 'v': buffout[3]='1';
            buffout[4]='1';
  case 'c': buffout[3]='1';
            buffout[4]='2';
}

buffout[8]='\r';

Write(outfh,buffout,9);
Close(outfh);

Execute( "runback :PC/PCDisk",0,0);

/* operazioni relative all'attivazione di JH0: */

Delay(800);
Execute( "DJMount",0,0);
Execute( "Assign WB: JH0:",0,0);
Execute( "AddBuffers WB: 30",0,0);
Execute( "CD WB:",0,0);
Execute( "Assign c: WB:c",0,0);
Execute( "Assign s: WB:s",0,0);
Execute( "Assign t: WB:t",0,0);
Execute( "Assign l: WB:l",0,0);
Execute( "Assign libs: WB:libs",0,0);
Execute( "Assign devs: WB:devs",0,0);
Execute( "Assign fonts: WB:fonts",0,0);
Execute( "Assign sys: WB:",0,0);
Execute( "Assign System: WB:System",0,0);
Execute( "Path add System",0,0);
Execute( "Path add Utilities",0,0);
Execute( "Path add WB:PC",0,0);
Execute( "assign c-devel: WB:c-devel",0,0);
Execute( "path WB:c-devel add",0,0);
Execute( "Assign WB:",0,0);

/* solo per i possessori di virusx... */

Execute( "runback virusx",0,0);
}

```

tori che certamente avrebbero reso il listato illeggibile. Dopo le dichiarazioni delle variabili e degli array utilizzati dal programma, si passa alla fase di richiesta e controllo password. Utilizzando per l'input da tastiera la funzione getchar() è necessario, come detto, terminare la PW in fase di inserimento dal CTRL-\ . Per il controllo di questa, la crittazione effettuata per renderla illeggibile dopo la compilazione si realizza semplicemente non facendo comparire mai i caratteri ma i codici ASCII opportunamente truccati sommando o sottraendo costanti. Ad esempio non troviamo nel sorgente la «o» (di "pippo.3") né direttamente il suo codice ASCII 111 ma l'espressione 74+c+32 che dà lo stesso valore a Run Time dal momento che «c» è fissata per l'occasione pari a 5. Posto di aver inserito la PW giusta, l'esecuzione continua eseguendo alcuni comandi CLI con la funzione Execute(). Il cuore del programma è la traduzione della data fornita dall'orologio di Amiga in data e time compatibile con i corrispondenti comandi MS-DOS. Infatti per Amiga «otto dicembre '88 ore 18 e 26» si scrive:

Thursday 08-Dec-88 18:26:00

e così il sistema ce lo restituirà, in MS-DOS occorre tradurlo nelle due stringhe:

18.26.00

e

08.12.88.

Terminate le due traduzioni, il programma salva le due stringhe ottenute in due file «TimePC» e «DatePC». Il resto del listato non è altro che la trasposizione in C dei rimanenti comandi CLI da eseguire per completare le operazioni di inizializzazione dell'HD JH0:.

Utilizzazione

Dopo aver copiato fedelmente il programma in un vostro file è naturalmente necessario compilarlo e linkarlo opportunamente. Una volta in possesso del codice eseguibile consiglio di appiccicarvi un'icona in modo da tenerlo disponibile nella finestra del vostro disco di Boot. Inutile ricordarvi che dovrete ripulire la startup-sequence da tutti i comandi di attivazione dell'HD (ci penserà il programma compilato) e che questa utility deve naturalmente giacere sullo stesso dischetto sul quale troviamo i comandi DJMOUNT, BINDDRIVERS, la Janus.library e tutto ciò che serve per lanciare la Janus. E non dimenticate, mi raccomando, di modificare Config.sys e Autoexec.bat prima di dare il via al programmino. OK?