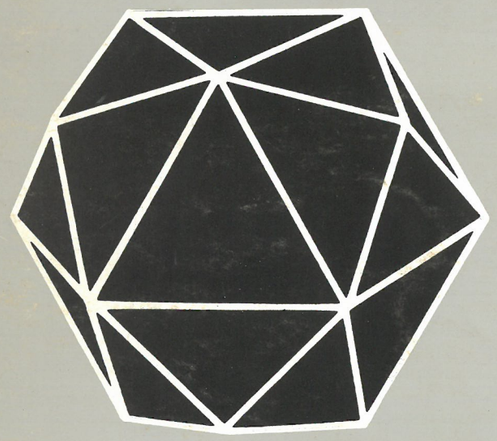
Spedizione in abbonamento postale, Gruppo IV, 2° semestre.
Pubblicità Inferiore al 70%.
N. 37, Settembre - Dicembre 1989

TAXE PERÇUE
P.P. Tassa riscossa PT Piacenza F. Italia

Quaderni di informatica

Rassegna di tecnologia ed applicazioni degli elaboratori elettronici

a cura del Centro Studi Informatica e Automazione Bull Italia



37
Anno XV - Numero 3 - 1989

FPDM-44

Worldwide
Information
Systems

Bull

Quaderni di informatica

Rassegna di tecnologia ed applicazioni degli elaboratori elettronici
Anno XV - Numero 3 - 1989

Sommario

Direttore Responsabile
Franco Filippazzi

Comitato di Redazione
Antonio Cicu, Carlo Falcetti, Paolo Lupo,
Mario Nobile, Giulio Occhini,
Fulvia Sala

Segreteria di Redazione
Elena Pighin

Redazione
Bull Italia
Centro Studi Informatica e Automazione
Via G.M. Vida, 11 - 20127 Milano

Stampa
tipolito Maggioni srl

Autorizzazione del Tribunale di Milano
n. 93 del 20 Marzo 1974

Pirateria del software: tecniche di protezione fisica

EUGENIO ORLANDI

ACEA
Roma

1. Introduzione

Il problema della protezione del software dall'uso illegale può essere affrontato con tecniche diverse appartenenti alle sfere del diritto e delle tecnologie informatiche.

Queste ultime, e in particolare le tecniche di protezione fisica, costituiscono l'oggetto della presente memoria.

2. La protezione del software

Innanzi tutto si deve definire cosa si intende per protezione del software: non tanto la sicurezza dei dati quanto quella dei programmi esposti ad appropriazione, riproduzione e uso non autorizzato, la cosidetta pirateria.

Per poter parlare di pirateria, i programmi, applicativi o di sistema, devono essere utilizzati illegalmente secondo le modalità per cui sono stati scritti e devono avere un valore di mercato riconosciuto.

I motivi oggettivi che rendono appetibile la pirateria sono i costi crescenti di sviluppo del software e la facilità con cui i programmi possono essere copiati.

La fig. 1 rappresenta il circuito attraverso cui il software raggiunge gli utilizzatori. I programmi acquisiti legalmente possono essere rivenduti illegalmente o distribuiti gratuitamente ad altri utilizzatori. In questo modo i distributori disonesti possono vendere prodotti per cui non hanno sostenuto i costi di sviluppo. Per spezzare questo ciclo vizioso sono state sviluppate le cosidette tecniche di autorizzazione - protezione della copia, validazione e crittografia - implementate con strumenti software e/o dispositivi hardware.

Le esigenze di protezione sono diverse per i grandi elaboratori e i personal computer: nel primo caso il rischio del fornitore, anche se elevato, è circoscritto: si tratta di impedire che il software di base e le utility siano utilizzati su macchine compatibili. Nel secondo caso si può avere una sensibile contrazione del mercato.

In seguito si farà riferimento a prodotti disponibili. Se l'offerta - almeno a conoscenza di chi scrive - è in regime di monopolio verrà descritto un determinato prodotto. In altri casi, pur non essendo l'offerta in condizioni di monopolio, la completezza della trattazione potrà risentire della comprensibile difficoltà a reperire informazioni.

Sono possibili diversi livelli di protezione a seconda che il pirata sia un utilizzatore o un'organizzazione dotata di ingenti risorse umane e finanziarie, ad esempio una software house o un rivenditore. La protezione ottimale consiste nell'avvicinare il costo sostenuto per disporre di una copia illegale eseguibile a quello di sviluppo del prodotto. Un livello così spinto di protezione può essere ottenuto solo col ricorso alla crittografia. A fronte di questa sicurezza, la crittografia presenta diversi svantaggi in termini di costi e di difficoltà gestionali. Per prodotti a basso valore è possibile servirsi di altre tecniche - protezione dalle copie e validazione - che consentono di ottenere buoni rapporti protezione/prezzo, soprattutto se concepiti come difesa da pirati non professionali.

Una prima forma di protezione consiste nell'adozione di tecniche che limitano la facoltà di eseguire copie del programma.

In altri casi viene usata la matricola della macchina, o una combinazione di matricola e informazioni ottenute dal software memorizzato su ROM, da inserire nel computer prima che il programma vada in esecuzione. Questi sistemi presentano due forti limitazioni: da un lato sono insufficienti contro i pirati professionali, dall'altro si traducono in un onere per gli utenti: limitano la possibilità di copie di salvataggio, richiedono un overhead eccessivo, ostacolano la distribuzione e la manutenzione delle copie autorizzate.

La seconda tecnica, la validazione, consiste nel fare in modo che il software protetto sia in grado di controllare il diritto di essere eseguito da un utente, impedendo così che copie illegali possano essere eseguite su sistemi non autorizzati.

La fig. 2 mostra le combinazioni possibili di tecniche di autorizzazione e dispositivi hardware mentre la fig. 3 costituisce una tassonomia dei sistemi di protezione del software.

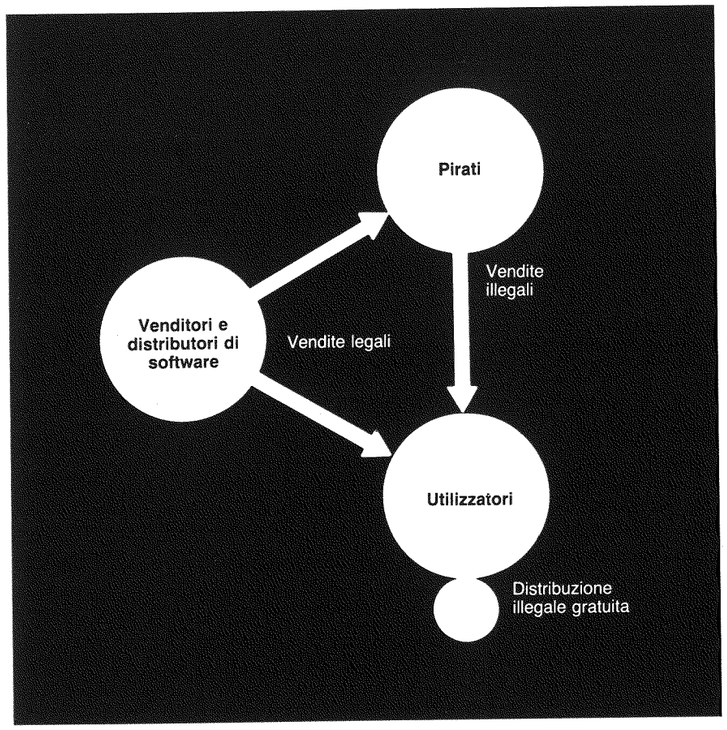


Fig. 1 - Circuito di distribuzione del software.

DISPOSITIVI / TECNICHE	Protezione delle copie	Validazione	Crittografia
Floppy disk	SI	SI	
Dispositivi removibili		SI	
Dispositivi sulle di porte I/O		SI	SI
Dispositivi sul bus		SI	SI
Criptoprocessori			SI

Fig. 2 - Combinazione tra tecniche di autorizzazione e dispositivi hardware.

Dispositivi hardware	Codice	Tecniche di autorizzazione	Codice
Disco	D	Protezione delle copie	C
Dispositivi removibili	X	Validazione	V
Dispositivi sulle porte di I/O	P	Crittografia	E
Dispositivi sul bus interno	B		
Criptoprocessori	C		

Fig. 3 - Tassonomia dei sistemi di protezione del software.

3. Le protezioni hardware per i piccoli sistemi

Per comprensibili motivi, è disponibile una maggiore gamma di prodotti per i piccoli sistemi: è stato stimato che a fronte di ogni copia legalmente venduta sono in circolazione anche sei copie illegali (Double Software Gold, 1986). Per limitare questo fenomeno sono state studiate diverse contromisure: psicologiche, logiche e fisiche. La protezione ideale deve essere poco costosa, facile da installare, trasparente per l'utente, compatibile con gli altri programmi non protetti, imprevedibile, multipla e difficilmente individuabile.

Le protezioni logiche (fig. 4) possono attuarsi mediante modifiche al sistema operativo - ad esempio non facendo comparire col comando "DIR" i file che si desidera proteggere - o mediante l'inserimento nel codice oggetto di trappole e bombe logiche.

Questa ultime sono costituite da istruzioni che, attivate da un evento esterno (una data, il numero di volte in cui è stato utilizzato il programma, ecc.) bloccano il programma e richiedono l'intervento del venditore.

Le trappole fisiche sono indipendenti dai programmi e interessano i supporti e l'hardware vero e proprio. Tra queste, hanno riscosso maggior successo:

1. le protezioni legate al supporto, tipicamente i floppy disk;
2. le protezioni removibili inserite sulle porte di I/O;
3. i criptoprocessori.

3.1. Le protezioni legate al supporto

La prima forma di protezione che viene in mente - e la prima ad essere stata adottata nel 1980 per il programma VisiCalc dell'Apple - consiste nel rendere difficile la duplicazione dei supporti. La fig. 5 riporta un elenco di protezioni legate al supporto, che operano a livello di tracce, settori e tempi. Fondamentalmente si tratta di ricorrere a espedienti basati:

a. sulla conoscenza di come operano i programmi standard di copia, ad esempio:
   - a1. inserendo segmenti fasulli ("dummy") con CRC (Cyclic Redundancy Check) o codici di controllo errati, ininfluenti in fase di esecuzione del programma ma letti come errati;
   - a2. inserendo bit deboli che vengono letti talvolta come 1 e talvolta come 0;
b. sull'uso di formati non standard: tracce e settori extra, allineamento perfettamente definito dei settori, tracce a spirale (e non circolari) o più grandi, ecc.

Questi tipi di protezione presentano diversi svantaggi: in alcuni casi è necessario addirittura un hardware speciale, ad esempio lettori di disco capaci di leggere incrementi di 1/2 traccia o settori extra.

Nel caso di interventi limitati al solo supporto, la protezione può essere aggirata utilizzando dei programmi copiatori che eseguono una copia fisica, bit per bit, del contenuto senza interpretare il CRC, i bit di protezione e le altre informazioni di controllo. In alcuni casi i copiatori sono persino in grado di rimuovere la protezione generando copie perfettamente duplicabili.

Un altro sistema per aggirare la protezione consiste nell'eseguire il programma sotto un debugger, interrompere l'esecuzione e copiare l'immagine della memoria su un disco non protetto.

3.2. I dispositivi fissi e removibili

Diversi dispositivi, fissi e removibili, sono utilizzati per la validazione (fig. 4) Nei sistemi protetti con la tecnica della validazione, il software controlla la presenza di una chiave e, se non la trova, interrompe l'esecuzione. La chiave può essere un numero memorizzato sul disco dove risiede il programma (in una locazione normalmente non copiata), in un registro a sola lettura, in un dispositivo hardware fisso o removibile alloggiato sul bus o sulla porta di I/O. I sistemi a validazione possono essere aggirati venendo a conoscenza della chiave (ad esempio usando un debugger o un disassembler), duplicandola, simulandone la presenza. Di contro, hanno un buon rapporto protezione/prezzo.

Un altro sistema di protezione consiste nel distribuire il software su ROM, anzichè su floppy. Si tratta di una soluzione indubbiamente efficace che non ha avuto un grande successo in quanto richiede un intervento a livello hardware. Ugualmente poco usati sono i dispositivi fissi sul bus che riducono il numero di slot disponibili.

Infine, è possibile legare la protezione a dispositivi removibili da inserire sulle porte di I/O, sulla tastiera o sullo schermo.

3.2.1. La chiave del software

Un esempio di dispositivo da inserire sulla porta di I/O è costituito dalla chiave di protezione del software, introdotta nel 1983 dalla società francese Microphar e commercializza in Italia dalla Siosistemi. Il Software viene venduto con la chiave - che può essere rimossa ma non duplicata - e non può essere utilizzato senza di essa.

La protezione funziona nel modo seguente: in ogni serie di chiavi approntata per un determinato prodotto software vengono inseriti, mediante un cablaggio personalizzato, il codice del produttore del software e il codice della serie. Il codice del produttore è utilizzato sia per distinguere le chiavi di ciascun produttore sia per elaborare la routine di integrazione (crittografata), che deve essere inclusa nel programma che si vuole proteggere. In fase di esecuzione, il programma cerca il codice e, se non lo trova, può decidere il livello di utilizzazione: bloccarsi o dare un'ulteriore possibilità segnalando la mancanza della chiave.

Le routine crittografate, fornite dalla Microphar, richiedono meno di mille byte nel programma e vanno inserite prima della compilazione seguendo precise istruzioni. L'attivazione avviene in fase di esecuzione.

La chiave è composta da un circuito integrato, componenti attivi (transistor e diodi) e passivi. La tecnologia di fabbricazione, "circuito elettroni-

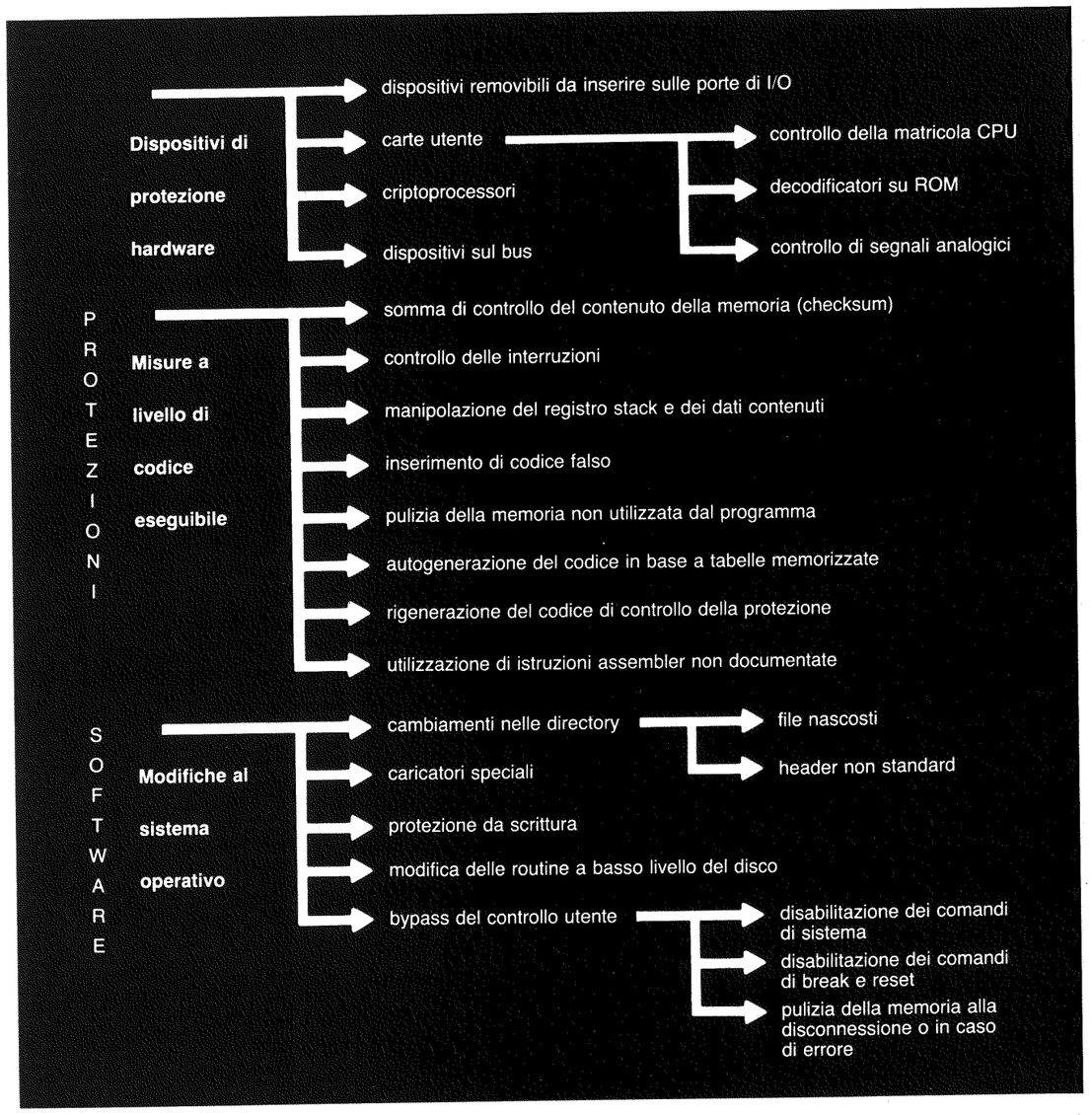


Fig. 4 - Metodi di protezione del software non legati al supporto.

co immerso in una resina dura" ne rende difficile la neutralizzazione.

Dal punto di vista dell'ambiente operativo, è disponibile in diversi linguaggi (Basic, Pascal, C, Fortran, Cobol, Assembler,dBase III, APL) e sistemi operativi (MS-DOS, Unix, Prologue).

Rispetto ad altri sistemi di protezione l'utilizzatore, che può usare fino a quattro chiavi una sull'altra senza compromettere il funzionamento della stampante, va incontro a disagi minimi, non necessita di dispositivi hardware complementari - in particolare non rischia di commettere gli errori possibili nel caso di dispositivi che richiedono l'immissione di dati da tastiera - e gode di diversi vantaggi: non è limitato nel numero di copie di salvataggio, può trasportare l'applicazione (e la chiave) da una macchina all'altra, non deve dedicare uno slot.

Il primo modello è stato prodotto in oltre 65000 unità - di cui 5000 solo in Italia - ed è utilizzato da più di 400 tra software house e aziende industriali. A favore della chiave si può portare il buon rapporto protezione/prezzo: da 210 a 450 FF in funzione dei modelli e delle quantità.

Una variante è costituita dalla "chiave intelligente", concepita per chi deve proteggere programmi applicativi che si collocano in una fascia di costo più alta e utilizzabile anche per altre funzioni: nel controllo dell'accesso e, in caso di affitto del software, per stabilire un limite di tempo o il tempo o il numero di volte in cui un programma può essere utilizzato. Identica esteriormente alla chiave tradizionale, dispone di una memoria di 64 byte che amplia la gamma di controlli che possono essere introdotti dal programmatore, consentendo la modifica delle caratteristiche interne della chiave e una notevole libertà sul livello di utilizzazione del programma. La memorizzazione è garantita per 10 anni.

La chiave è costituita da componenti elettronici; in fase di cablaggio viene inserito un codice attribuito ad ogni produttore cui sono fornite specifiche e univoche routine per leggere o scrivere i 64 byte.

4. La protezione del software nei grandi sistemi

Il problema della protezione è meno sentito nell'ambito dei grandi sistemi dove la distribuzione del software è preceduta dalla firma di un contratto di licenza d'uso. Il software di base richiede un supporto continuo del fornitore sotto forma di aggiornamenti e cambiamenti di release, update e upgrade.

Nei grandi sistemi - ma anche nei piccoli e nei medi - la protezione avviene mediante la personalizzazione del software, legato alla CPU dell'utente, attuata inserendo in diversi punti del programma un codice che la identifica. Anche in questo caso sono possibili diverse varianti. La Digital combina il numero di contratto di licenza con la matricola della CPU e, in base ad un algoritmo, fornisce una chiave che l'utente inserisce nel programma. Ogni volta che il software gira controlla che la CPU sia quella prevista dalla chiave di accesso. Per una maggiore sicurezza si possono usare tecniche di checksum, di cui il CRC e il controllo di parità sono due esempi (fig. 4, controlli sull'integrità della memoria).

Una checksum è un numero, una lettera o una combinazione di numeri e lettere, prodotta da un algoritmo che è sensibile ad ogni cambiamento di tutto o parte del codice sorgente o oggetto. Non ha valore in sé ma è solo l'identificatore di una configurazione di dati. L'algoritmo usato, ad esempio la somma di prodotti, deve essere sensibile all'aggiunta e distruzione, modifica, trasporto e riorganizzazione. Se utilizzata correttamente, la tecnica della checksum costituisce per il software l'analogo delle impronte digitali per gli esseri umani.

4.1. Il controllo dell'accesso

Un aspetto importante della protezione è il controllo dell'accesso realizzabile via software, mediante pacchetti add-on, o via hardware. In seguito si accennerà brevemente alle sole protezioni hardware. Le funzioni di controllo interessano gli utenti, la sicurezza degli archivi di programmi e dati.

Nell'architettura VME High Security Option dell'ICL, dove è implementata una politica rigida di tipo mandatory (livello "B1" della scala adottata dal Dipartimento della Difesa USA), il problema del controllo dell'accesso è visto in uno spazio bidimensionale. La protezione orizzontale è ottenuta impedendo l'accesso tra macchine virtuali. La protezione verticale è gestita dal registro di controllo degli accessi, che può essere abilitato solo da una chiamata di sistema in presenza di un setting valido del registro stesso. Un utente non può leggere la memoria centrale né il programma di un altro utente perchè si trovano in spazi di indirizzi disgiunti.

Un livello più alto, la protezione di tipo "A1" secondo il DoD, è ottenibile col ricorso ai "nuclei di sicurezza", noti come security kernel o reference monitor. Il nucleo è un sottosistema che contiene tutte le funzioni di sicurezza di un sistema operativo commerciale. Caratteristiche essenziali del nucleo sono:

1. la completezza: vengono controllati tutti gli accessi a tutti gli oggetti da parte di tutti i soggetti;
2. l'inviolabilità: è protetto da modifiche o interferenze da qualsiasi altro software;
3. la correttezza: esegue solo le funzioni per cui è stato predisposto e non altre.

Un esempio commerciale è SCOMP, Security Communication Processor della Bull. Recentemente, il nucleo è stato trasferito completamente in hardware (Security Ada Target).

4.2. La protezione nei sistemi aperti

Un secondo aspetto rilevante nei grandi sistemi è costituito dalle reti dove i programmi memorizzati su un nodo possono essere trasferiti ed eseguiti su altri nodi. Nelle reti il problema del controllo dell'accesso risulta più delicato. In questo caso sono disponibili:

1. dispositivi di call-back da inserire tra la rete telefonica ed il nodo, per intercettare le chiamate, verificarne la validità, disconnettendo il chiamante per richiamarlo all'indirizzo previsto nelle liste di autorizzazione;
2. sistemi di identificazione, autenticazione e autorizzazione;
3. la crittografia link-by-link o end-to-end sulle linee di trasmissione dati.

5. La protezione del software attraverso la crittografia

Com'è noto, crittografia è il processo che trasforma l'informazione (testo in chiaro) in forma non intelleggibile (testo cifrato), così che può essere inviata su un canale non sicuro o può essere memorizzata su file non sicuri. Con riferimento al problema in esame, l'idea base consiste nel crittografare il software da proteggere in modo che possa essere decifrato ed eseguito solo con una chiave opportuna. In questo modo il problema della protezione del software è ricondotto a quello della gestione delle chiavi: generazione, distribuzione e protezione. La semplice vendita di software crittografato non risolve il problema in quanto si limita a spostare l'oggetto della pirateria dal software alla chiave. Per superare questa limitazione, Albert e Morse (1984) hanno proposto un metodo che consente di proteggere la chiave crittografandola e rendendone possibile la decrittazione solo

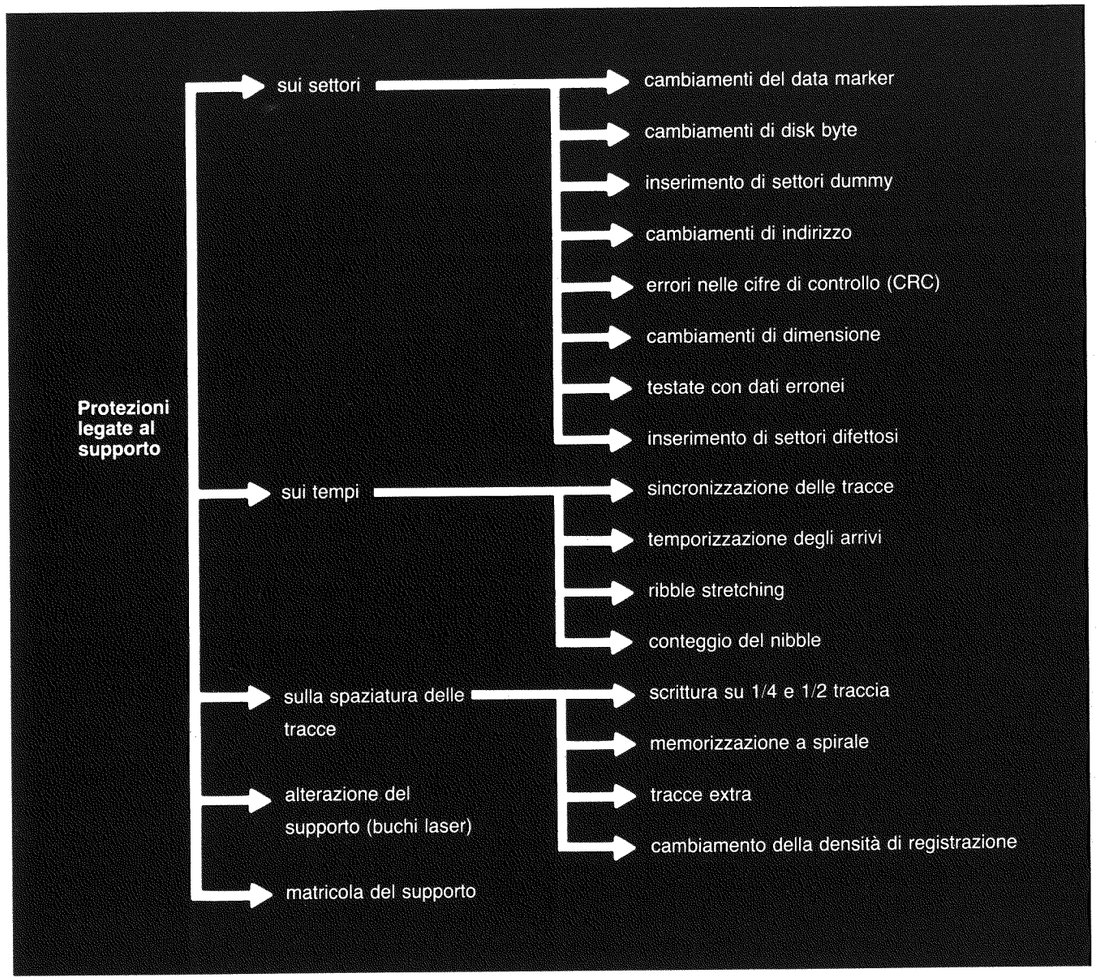


Fig. 5 - Protezioni legate al supporto.

nell'hardware abilitato a far girare il software in oggetto. In questo modo il problema è ricondotto alla protezione hardware della chiave di protezione.

Ciascun processore ha incorporata una chiave di decrittazione che viene memorizzata in una zona protetta della memoria. Il fornitore del software conosce la chiave di codifica, ma non quella di decodifica: se la conoscesse per un particolare processore potrebbe ordinare il software di un suo competitore per quel processore e decodificarlo ottenendo così il testo in chiaro.

La chiave di decodifica non deve essere derivabile da quella di codifica, vincolo che richiede il ricorso alla crittografia a chiave pubblica.

La proposta di Albert e Morse si basa su un hardware speciale per la conservazione delle chiavi mentre il software cifrato può essere distribuito su un supporto qualsiasi (floppy, nastro, ROM) o inviato su rete di trasmissione dati. Nel corso dell'esposizione si farà riferimento al costruttore dell'hardware, al fornitore del software a all'utilizzatore anche se il modello può tener conto di altre figure quali fornitori OEM e distributori di software. La logica del funzionamento è illustrata nella fig. 6.

Il costruttore dell'hardware memorizza la matricola del processore e la chiave per la decrittazione kH in una zona di memoria non volatile protetta. Il fornitore del software P, a sua volta, passa il testo in chiaro attraverso un traduttore ottenendo un testo cifrato con una chiave di esecuzione kS. Quando il cliente acquista il software crittografato con la chiave di esecuzione kS, invia la matricola del processore di decrittazione al fornitore P. Questi, a sua volta, manda la matricola e la chiave esecutiva al costruttore dell'hardware per la cifratura o procede direttamente alla cifratura se viene adottata la crittografia a chiave pubblica.

L'utente esegue un programma di installazione che, inserendo la matricola e la kS in una lista di chiavi,

rende eseguibile sul processore il software protetto. Le prime istruzioni del programma di installazione sono in chiaro, caricano un registro speciale - il registro di esecuzione - con la chiave esecutiva e predispongono il processore a lavorare secondo la modalità esecutiva. L'ultima istruzione logica rimuove questa modalità. La sicurezza è affidata alla chiave esecutiva e alle coppie matricola-chiave di decrittazione e matricola-chiave di crittografia.

Il modello ammette diverse varianti che utilizzano per la conservazione della chiave di decrittazione kH e/o la chiave esecutiva kS carte a microprocessore, smartcard o supersmartcard con tastiera e display. La decrittazione può essere effettuata dalla CPU all'atto del caricamento del programma a runtime da un criptoprocessore o da un dispositivo collegato al bus o alla porta di I/O. In questo caso non compare mai il testo in chiaro. Le difficoltà nascono dal fatto che la crittografia è tanto più sicura quanto più laborioso è il processo di decrittazione, fatto questo che può portare ad un degrado intollerabile delle prestazioni del sistema.

A seconda delle esigenze, può essere sufficiente cifrare solo alcuni moduli del software, ad esempio eventuali algoritmi esclusivi. A causa della natura di auto-autenticazione della cifratura, questo metodo di protezione può anche essere utilizzato per prevenire modifiche non autorizzate.

I sistemi crittografici possono essere aggirati identificando la chiave, ad esempio usando un debugger per copiare il programma dopo che è stato decifrato o partendo da una copia non crittografata. Un pirata può cercare di violare l'algoritmo o può ricorrere a metodi fisici per estrarre la chiave di decrittazione dal processore su ROM o PROM da leggere con un microscopio ottico. Un maggior livello di sicurezza si può ottenere con una memorizzazione su EPROM.

5.1. L'hardware per la crittografia

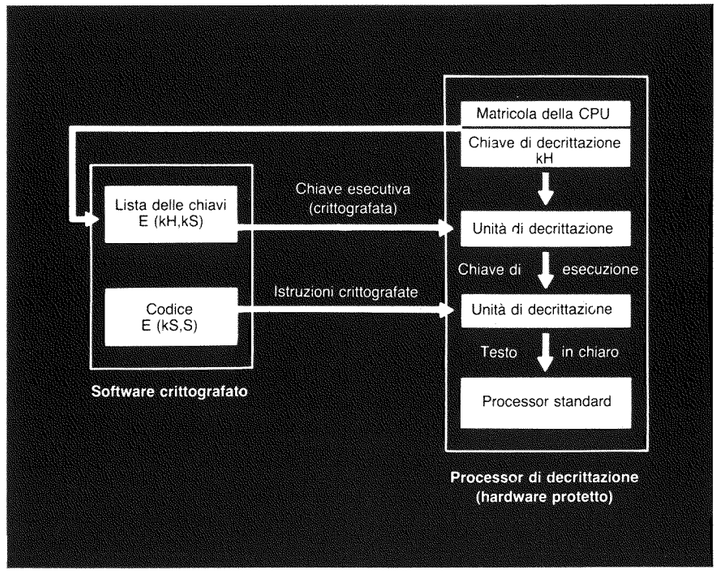
Com'è noto, la crittografia può essere effettuata via hardware o via software. In questo caso, sono disponibili pacchetti con facility per la checksum e l'autenticazione, ad esempio: Datafase, Super Encryptor II, Cript Master 24, Fast-Crypt, Padlock, For Your Eyes Only e Pack & Cript.

Tra i prodotti hardware risultano particolarmente efficaci i processori di sicurezza che leggono le istruzioni crittografate, le decodificano e le eseguono. In queste macchine il codice non appare mai in chiaro anche se il processore è in grado di lavorare in entrambi i modi - chiaro e cifrato - operando in base a istruzioni privilegiate. Si tratta di schede per PC IBM XT/AT e compatibili, in grado di crittografare il sistema operativo, i programmi e i dati su floppy o su disco fisso. In genere contengono un'implementazione veloce del DES (Data Encryption Standard) su circuito integrato, la logica di interfaccia col bus, una ROM con i kernel driver per implementare le funzioni DES e offrono la possibilità di memorizzare la chiave su ROM. Un software opportuno - utility e programmi applicativi - consente ad un utente non professionale di utilizzare le funzioni crittografiche. Oltre alle funzionalità di base, si possono avere funzioni di autenticazione di messaggi (ad esempio secondo lo standard ANSI X9.9 MAC, Message Authentication Code), memorie CMOS protette accessibili dal solo processore di sicurezza per la conservazione delle chiavi (in alcuni modelli) o di un numero limitato di dati. Le chiavi usate per la crittografia sono combinate con le chiavi personalizzate cablate sulla scheda prima di essere caricate nel processore DES.

I sistemi di protezione basati sulla crittografia presentano diversi svantaggi:

- è necessario aprire il cabinet per installare la scheda;

Fig. 6 - Protezione del software con tecniche crittografiche.
Legenda: kH = chiave del costruttore hardware;
E = funzione di cifratura;
kS = chiave del fornitore software.



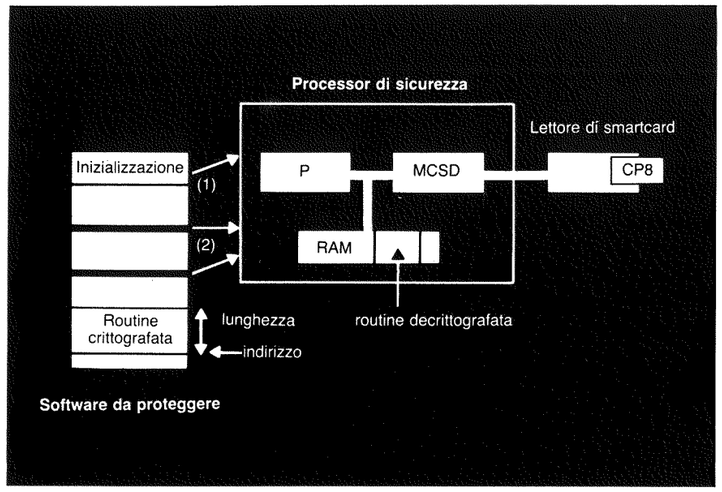


Fig. 7 - Protezione del software con la carta a microprocessore
(1) Trasmissione della lunghezza e dell'indirizzo della routine crittografica
(2) Chiamata al processore di sicurezza per l'esecuzione della routine crittografata

TECNICHE DI AUTORIZZAZIONE	VANTAGGI	SVANTAGGI
Protezione delle copie	Non necessità di hardware addizionale È totalmente trasparente	Può essere aggirata con programmi copiatori o debugger
Validazione	Basso costo	Può essere aggirata scoprendo il codice di validazione Richiede hardware addizionale Causa overhead
Crittografia	Massima sicurezza potenziale	Richiede hardware addizionale Allunga i tempi di esecuzione Comporta problemi di gestione delle chiavi

Fig. 8 - Vantaggi e svantaggi delle tecniche di autorizzazione.

- se il processor si guasta, e deve essere sostituito, il nuovo processore deve avere la stessa matricola o deve essere ripetuta la procedura di cifratura del software;
- la decrittazione riduce le prestazioni del sistema;
- la sicurezza è proporzionale alla lunghezza del blocco di dati crittografato.

Esempi di schede sono PC Encryptor della Eracom e Secad (Security Adapter) della IDO. Quest'ultimo è un coprocessore su scheda che può essere integrato con la Smart Card CP8.

Tutte le parti sensitive sono fisicamente protette. Altre facility sono:

a. la gestione delle chiavi secondo lo standard IBM;
b. l'uso della crittografia a chiave pubblica per l'autenticazione;
c. l'identificazione e l'accounting dell'utente;
d. la memorizzazione della chiave;
e. la memorizzazione removibile della chiave su carta a microprocessore;
f. la memorizzazione removibile su carta a microprocessore di parti di programma.

La crittografia può essere selettiva - limitata ad alcuni file e non all'intero disco - come nel caso di Fileencryption della IDO, che utilizza per la memorizzazione della chiave la smart card CP8. In un altro prodotto della IDO, Multikey, basato sulla carta a microprocessore, viene crittografato tutto il software meno l'MS-DOS. Se durante l'esecuzione del programma la carta a microprocessore viene rimossa, si genera un interrupt hardware che ordina al Multikey di distruggere il codice crittografato e decrittografato e l'algoritmo in memoria.

5.2. La protezione con le carte a microprocessore

Le carte a microprocessore, di cui la CP8 della Bull è stata l'antesignana, possono essere usate sia per il controllo dell'accesso sia per la protezione del software di mini e micro. Inoltre le smart card possono essere utilizzate con componenti (RAM, ROM, processori) che consentono altre funzioni di sicurezza quali l'identificazione e la certificazione. In fase di verifica si possono utilizzare:

Fig. 9 - Confronto tra i diversi dispositivi hardware.

DISPOSITIVI HARDWARE	VANTAGGI	INSTALLAZIONE
Floppy disk	È totalmente trasparente	Non richiede operazioni speciali
Dispositivi removibili	Richiede l'intervento dell'utente ad ogni esecuzione	Non richiede operazioni speciali
Dispositivi sulle porte	Operazione trasparente Richiede spazio sul retro del computer Può interferire con i dispositivi di I/O	Relativamente facile: si tratta di inserire un connettore
Dispositivi sul bus	Operazione trasparente	Non immediata: può richiedere l'apertura del cabinet
Criptoprocessori	Operatività trasparente	Difficile: si deve aprire il cabinet per inserire un chip o una scheda

1. metodi passivi, per controllare se il software è stato alterato;
2. metodi attivi, per proteggere l'uso del software, prevenendo utilizzazione, copia e modifiche illegali.

La fig. 7 mostra un esempio di protezione di parte del software basato sulla crittografia, studiato e sviluppato in Francia presso l'Ecole Nationale Supérieure des Telecommunications.

L'hardware comprende un processore di sicurezza, costituito da un microprocessore Motorola 6809, e un componente specializzato per il colloquio col lettore di carta CP8, l'MCSD, un Motorola MC68705P5, dotato di memoria e di un meccanismo di accoppiamento col bus del processore.

All'atto dell'inizializzazione del software che si desidera proteggere viene inviato al processore di sicurezza l'indirizzo della parte da crittografare.

Questo copia il software nella sua memoria e, attraverso il colloquio attivato dall'MCSD con la carta a microprocessore, ottiene la chiave di decrittazione. Una volta decifrato, il software resta nella memoria del processore di sicurezza, che lo esegue su richiesta del sistema operativo.

Conclusioni

Sono oggi disponibili per la protezione del software diverse tecniche: fisiche, logiche e legali. Nel corso dell'esposizione è stata utilizzata una tassonomia delle protezioni basata sulla definizione delle tecniche di autorizzazione - protezione della copia, validazione e crittografia - e sulla descrizione di dispositivi che possano implementarle. Nella fig. 8 vantaggi e svantaggi delle tecniche di autorizzazione sono messi a confronto, mentre nella fig. 9 sono descritti vantaggi e problemi di implementazione dei diversi dispositivi.

Com'è noto a chi si occupa di sicurezza, nessun sistema è sicuro al cento per cento: è solo possibile far crescere il costo necessario per violarlo oltre il limite dell'economicità. La soluzione ottimale non esiste: è necessario scegliere di volta in volta la difesa più opportuna ricordando che, se le singole misure - fisiche, logiche e legali - sono indubbiamente efficaci, dalla combinazione di più misure è possibile ottenere significative sinergie.

Bibliografia

[1] ALBERT D. and MORSE S.; *Combatting Software Piracy by Encryption and Key Management,* IEEE Computer, vol. 17, n. 4 68-73, April 1984.

[2] BASSANETTI A.; *C'è un antifurto nel mio programma,* Zerouno, Dic 83, n. 23, pp. 42-46.

[3] BOEBERT et al.; *Elaborazione a prova di intruso: come neutralizzare i pirati informatici,* Quaderni di Informatica, Anno XII, N. 1, 5-22 (1986).

[4] BROWN B.J.; *Checksum Methodology as a Configuration Management Tool,* Journal of Systems and Software, n. 7 141-143 (1987).

[5] CHRISTOFFERSON et al.; *Crypto User's Handbook,* North-Holland, (1988).

[6] DUPUY M. et al.; *About Software Security With CP8 Card,* in: Information Security: The Challenge, A. Grissonnanche (Ed) IFIP/Sec '86, Monte-Carlo, (1986).

[7] HIGHLAND H.J.; *Data Protection in a Microcomputer Environment;* in: Computer Security: A Global Challenge, J.H. Finch and E.G. Dougall (Eds), IFIP/Sec '84, North-Holland, 517-531, (1984).

[8] ORLANDI E.; *Ingegneria del rischio,* Angeli, 1989.

[9] SUHLER P.A. et al.; *Software Authorization Systems,* IEEE Software, 34-41, Sept. 1986.

I virus del computer

C. CREMONESI, G.C. MARTELLA

Dipartimento di Scienze dell'Informazione
Università degli Studi di Milano

1. Introduzione

Il problema del virus del computer è di grande attualità grazie all'attenzione che la stampa - anche quella non specializzata - ha dedicato all'argomento.

Purtroppo, spesso accade che le informazioni fornite non siano del tutto attendibili o siano espresse poco chiaramente; ciò è dovuto soprattutto al fatto che coloro che subiscono l'esperienza di un virus nei propri programmi sono spesso restii a comunicare ad altri le proprie procedure di comportamento e persino ad ammetterlo per paura degli effetti di una pubblicità sfavorevole.

Sarebbe utile avere notizie direttamente dalle fonti ma ciò, per i motivi sopraesposti, è difficile.

Le informazioni più complete sono giunte dagli ambienti universitari dove il virus viene visto come un fenomeno da studiare piuttosto che come un problema da nascondere.

L'interesse suscitato dai virus è strettamente legato ai danni potenziali e reali da essi provocati. Un programma in grado di autoreplicarsi non fa notizia, ma la paura che il sistema informativo aziendale possa venire paralizzato da tale programma sì.

Il virus è diventato un problema quando le organizzazioni si sono trovate di fronte all'impossibilità di lavorare, alla perdita di giornate-uomo (se non di mesi) per ripristinare le condizioni necessarie alla ripresa dell'attività e a tutti gli altri danni correlati ad una sospensione non prevista e/o alla perdita di informazione.

Da allora, questo argomento ha occupato le pagine di riviste tecniche e quotidiani, ed i pensieri dei manager EDP.

D'altra parte il virus non è altro che l'ultima forma di sabotaggio di cui sono oggetto i sistemi informatici, da sempre nel mirino della criminalità informatica. (Fig. 1).

Nell'era antevirus i sabotaggi venivano compiuti in altro modo, ovvero: mettendo una bomba; provocando un incendio; provocando un allagamento; provocando un cortocircuito; distruggendo fisicamente l'elaboratore; distruggendo fisicamente i supporti di memorizzazione; distruggendo fisicamente altre apparecchiature connesse all'elaboratore; contaminando la superficie dei supporti di memorizzazione; provocando una alterazione dei dati o alterazione dei programmi (cavalli di Troia, bombe logiche); distruggendo le linee di comunicazione.

È bene pertanto sottolineare che la protezione dai virus va vista nel quadro delle misure di sicurezza fisica, logica ed organizzativa che sempre dovrebbero essere presenti nelle organizzazioni e nei sistemi EDP.

Questo lavoro è così organizzato.

Nella sezione 2 sono presentate le principali caratteristiche di un virus. Nella sezione 3 sono analizzate le varie forme di contagio e nella sezione 4 i danni provocati. Nella sezione 5 sono descritti i virus storicamente più conosciuti. Infine, nella sezione 6, sono presentate alcune misure per la prevenzione, l'individuazione ed il trattamento dei virus.

2. Caratteristiche dei virus

Un "computer virus" è un programma che è in grado di "infettare" altri programmi modificandoli in modo che includano una versione evoluta di se stesso. Ogni programma infetto agisce, a sua volta, come un virus ed è in questo modo che il contagio si diffonde in un sistema computazionale o in una rete. (Fig. 2).

F. Cohen, nell'articolo "Computer Viruses" introduce il seguente esempio di virus in pseudo-linguaggio:

```
program virus :=
begin
  1234567;
  subroutine infect-executable :=
    begin
      loop: file = random-executable;
      if first-line-of-file = 1234567
        then goto loop;
      prepend virus to file;
    end
  subroutine do-damage :=
    begin
      whatever damage is desired
    end
  subroutine trigger-pulled :=
    begin
      return true on desired conditions
    end
  main-program :=
    begin
      infect executable;
      if trigger-pulled
        then do-damage;
      goto next;
    end
  next:
end
```

Il virus dell'esempio ricerca un file eseguibile non infetto.

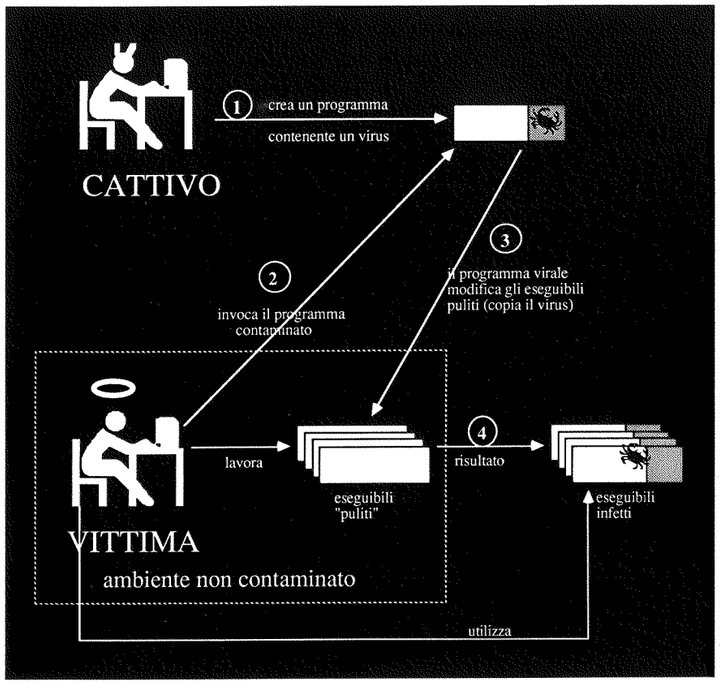


Fig. 1 - Il processo di infezione.

Tutti i file già contaminati contengono come prima linea la stringa "1234567" che è la condizione che identifica un virus; quindi si suppone che un file senza tale stringa di apertura sia "sano".

Trovato il file bersaglio, il virus copia se stesso all'inizio del file e controlla se una condizione di innesco è verificata.

In caso affermativo compie il danno per cui è stato scritto e di seguito esegue il programma contenuto nel file infettato.

Riguardo ai virus sono opportune alcune osservazioni.

Innanzitutto un virus - inteso come programma in grado di autoreplicarsi - non è necessariamente dannoso: esistono programmi virus che chiedono all'utente se desidera che il virus entri in funzione e svolgono una funzione utile al sistema ma abitualmente trascurata dall'utente (ad esempio, la compressione dei file). Questo tipo di virus infetta solo su richiesta ed esegue una funzione "buona" invece di procurare danno. Questo non è il tipo di virus di cui si parla in questo lavoro.

Un virus non è necessariamente un programma software: in teoria potrebbe essere realizzato in hardware o in firmware.

Un virus non è necessariamente un "cavallo di Troia" (trojan horse). Un trojan horse è un programma (sw,hw o firmware) di cui l'utente ignora l'esistenza e che, in generale, svolge funzioni dannose. Trojan horse e virus sono oggetti distinti anche se spesso un virus è anche un trojan horse perchè i virus vengono inseriti nel sistema senza che l'utente ne sia a conoscenza. Un trojan horse, tuttavia, non è in grado di replicarsi.

Esistono molte forme di virus; le varie forme possono essere raggruppate nelle seguenti tre classi principali (dipendenti dalla localizzazione del virus nei sistemi infetti):

- "boot infector"
- "system infector"
- "generic application infector"

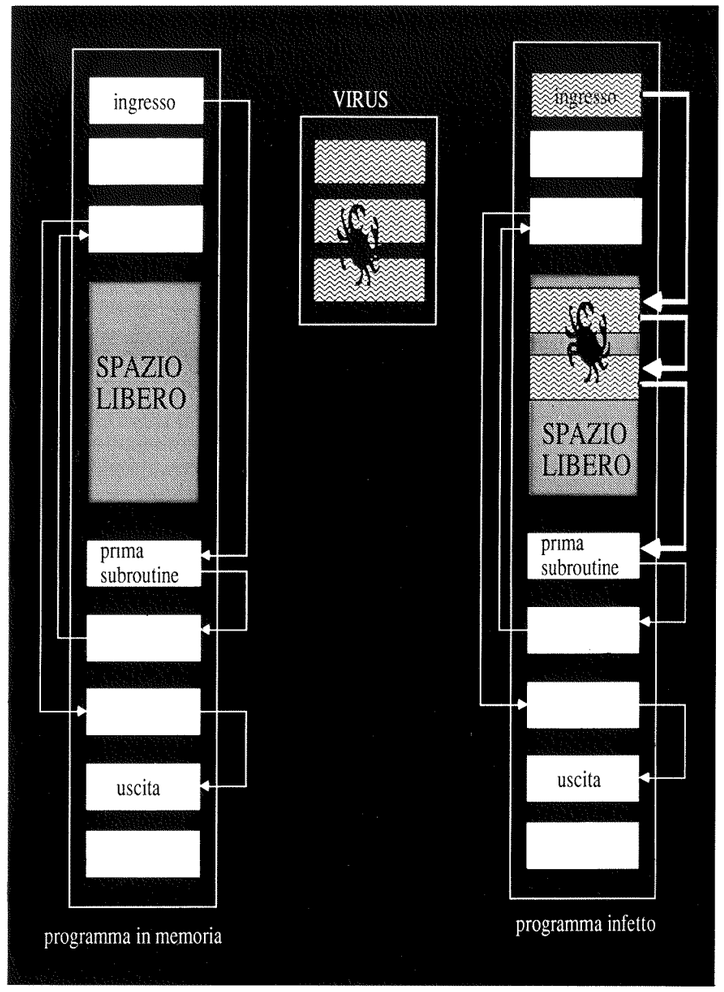
I virus "boot infector" attaccano i settori di boot dei floppy e degli hard disk. In questo modo, ottengono il controllo del sistema quando viene effettuato il boot e possono controllare l'intera attività del sistema, verificando se vengono inseriti nuovi dischetti. Quando uno di essi viene inserito e vi si accede per la prima volta, il virus si copia nel settore 0 del floppy.

Infetterà qualsiasi sistema il cui boot venga effettuato da tale dischetto. Avviare il sistema da un disco infetto è l'unico modo che questa classe di virus ha per infettare nuovi elaboratori.

I virus "system infector" attaccano l'ultimo modulo del sistema operativo o il device driver di sistema. Possono contaminare programmi interpreti, routine di I/O, driver speciali. Ottengono il controllo dopo il boot di sistema durante le operazioni di inizializzazione e attendono l'inserimento di un dischetto di sistema in uno dei drive. Quando ciò accade il virus si riproduce sui file di sistema.

I virus "generic application infector" sono in grado di infettare qualsiasi applicazione. Ottengono il controllo quando viene eseguita una applicazione infetta e ricercano altri host, altri dischi fissi o dischetti da contaminare. Se trovano una di queste

Fig. 2 - Il contagio.



unità sana, la infettano e restituiscono il controllo all'applicazione che continua la sua esecuzione. I virus appartenenti a questa classe sono i più diffusi.

3. Le forme di contagio

Si può venire contagiati:

- da un dischetto infetto da una fonte esterna
- attraverso l'acquisizione o il movimento di elaboratori infetti
- attraverso i sistemi di comunicazione
- tramite l'accesso a software pubblico.

Nell'ambiente dei personal computer, tuttavia, il contagio avviene più frequentemente tramite lo scambio di dischetti infetti.

I virus contaminano attaccandosi a un pezzo di codice o sostituendo un pezzo di codice. Alcuni virus della classe "boot infector", ad esempio, rimpiazzano interamente il settore di boot con una copia di se stessi: in pratica, diventano il nuovo programma di boot del sistema.

Un virus si attacca esternamente ad un programma oppure internamente ad esso. Se si posiziona esternamente al programma, generalmente la dimensione del programma cambia, facilitando l'individuazione del virus tramite il comando che elenca i file e lo spazio occupato. Se si posiziona internamente deve trovare elaboratori con spazio libero sufficiente a contenerlo.

In generale, i virus si attaccano esternamente rimpiazzando l'istruzione del programma dell'host con nuove istruzioni che puntano al codice principale del virus: il codice è di solito posizionato alla fine dell'applicazione.

Si possono individuare i vari stadi dell'infezione:

- contaminazione della memoria centrale
- contaminazione della memoria su disco locale
- contaminazione del file-system condiviso

- contaminazione dei mezzi rimovibili di tutto il sistema (floppy, hard disk rimovibile, WORM (Write-Once-Read-Many), nastri di back-up)

Il virus, quindi, si compone generalmente di due parti:

- una che svolge la funzione di duplicazione, che è sempre presente
- una che svolge la funzione di danneggiamento, che può non essere presente.

La complessità della funzione di danneggiamento è lasciata alla fantasia del progettista del virus.

Il danno, quindi, può essere più o meno grave in relazione alla funzione che lo produce.

4. I danni provocati dai virus

Le azioni dannose più comuni sono:

- distruggere la "file allocation table" che tiene traccia della catena logica dei segmenti in cui il file viene suddiviso per motivi di ottimizzazione dello spazio su disco, rendendo inutilizzabile il contenuto del file, in quanto l'utente si trova di fronte ad una specie di rompicapo composto da tutti i pezzi dei suoi file senza alcuna informazione su come fare a ricomporli;
- cambiare l'assegnamento del disco in modo che le informazioni vengano scritte sul disco sbagliato: questa azione è particolarmente dannosa quando il disco viene sostituito con l'entità "disco" corrispondente alla memoria RAM poiché provoca la definitiva perdita dell'informazione se l'elaboratore viene spento;
- cancellare specifici programmi eseguibili e/o file su hard disk e/o floppy disk;
- alterare i dati nei file di dati;
- impedire l'esecuzione di alcuni programmi residenti nella RAM;
- creare "bad sector" su un disco, a volte distruggendo parte di file di programmi o di dati;
- diminuire lo spazio disponibile su disco facendo copie di altri file già presenti nel sistema non interferendo con l'elaborazione dei programmi;
- scrivere una etichetta di volume su un disco se non ne esiste nessuna;
- formattare tracce specifiche di un disco o l'intero disco;
- riscrivere sulla directory del disco;
- condizionare il sistema in modo che non risponda più ad alcuna richiesta da tastiera.

Anche i virus che non possiedono alcuna parte dannosa ed il cui intento è solo quello di duplicarsi senza espletare alcuna funzione possono arrecare danno.

Infatti il loro codice non è stato certamente sottoposto a controlli di qualità: accade, quindi, che in presenza di situazioni particolari causi - anche se involontariamente - un crash del sistema provocando la perdita dei dati elaborati in quel momento.

5. Alcuni esempi di virus

Passiamo, ora, ad esaminare alcuni esempi di computer virus.

Mac Intosh Virus

Il virus era contenuto in un disco insieme al pacchetto grafico "Freehand" della Aldus Corporation di Seattle. Due specialisti contagiati avevano acquistato il programma in due negozi diversi. Fortunatamente il virus non era letale: visualizzava un messaggio universale di pace tratto da "MacMag" (una rivista canadese per il Mac Intosh) dopo di che tentava di duplicarsi su un altro disco. Il virus si autodistruggeva dopo che il messaggio era apparso sullo schermo.

Tuttavia, se qualsiasi disco che era stato contagiato veniva utilizzato dopo che l'infezione aveva avuto luogo, sullo schermo compariva comunque il messaggio che si cancellava poco dopo. Il virus, in generale, non causava danni al disco.

Lo stesso virus apparve su due servizi di bulletin board on-line: Compuserve a Colombus (OH) e alla tavola rotonda Mac Intosh su "Genie" della GE Information Services di Rockville (MD).

Lehigh Virus

Il virus appartiene alla classe "system infector" e colpisce i Personal Computer IBM e IBM compatibili.

Fu scoperto nel 1987 alla Lehigh University a Bethlehem (PA).

Questo era un virus letale che non solo danneggiò diverse centinaia di dischi all'università e provocò un crash dei microcomputer con hard disk nel laboratorio dell'università, ma infettò anche innumerevoli dischi di proprietà dei singoli studenti o dei membri dell'università.

Il virus era nascosto all'interno dello stack del file COMMAND.COM: tale parte del file occupava 300 byte ed era normalmente piena di "0".

Il fatto che il virus fosse nascosto nello stack faceva sì che la dimensione su disco del file non risultasse modificata.

Era stata alterata la prima istruzione del file che, di solito, consiste in un JMP ad una parte del codice del COMMAND.COM; la nuova istruzione era un JMP al codice di installazione del virus.

Quando il COMMAND.COM veniva eseguito, il virus si spostava in un'altra locazione di memoria e poi il controllo veniva restituito al vero codice del COMMAND.COM con una istruzione di JMP ad una locazione uguale a quella che era originariamente contenuta nella prima istruzione.

Quindi, mentre su disco un COMMAND.COM infetto ed uno sano occupavano lo stesso spazio, in memoria il COMMAND.COM infetto occupava più spazio di quello sano.

Quando il virus era in memoria intercettava gli interrupt INT 21H (che controllano fra l'altro l'I/O su schermo e su disco).

Ogniqualvolta una delle funzioni

- "find next file"
- "execute file"

dell'INT 21H veniva eseguita, prima di permettere la normale esecuzione, veniva eseguito il virus.

Queste due funzioni sono usate frequentemente da comandi come DIR, TYPE, COPY e ogni volta che un programma viene eseguito da disco.

Un errore di progettazione del virus faceva in modo che esso si installasse in memoria ogni volta che il COMMAND.COM veniva caricato: ciò portava a conflitti di memoria anche in presenza di un numero ridotto di programmi, causando inaspettati crash di sistema.

Quando il virus andava in esecuzione cercava un disco diverso da quello di default; se lo trovava, controllava se era bootable e se lo era si copiava sul disco ed incrementava un contatore.

In un sistema senza hard disk il contatore veniva memorizzato in memoria centrale, in un sistema con hard disk il contatore veniva memorizzato su di esso.

Quando il contatore era uguale a 4, il virus usava l'interrupt 26H per riempire di "0" i primi 50 settori del disco. In questo modo venivano rimossi anche i primi dodici settori del disco indispensabili per il suo funzionamento. Se il disco era "bootable", venivano cancellati il driver IBMMIO o MSBIO e parte del driver IBMDOS o MSDOS. Se il disco non era "bootable", venivano persi 38 settori contenenti dati.

In pseudo-codice il virus può essere rappresentato come segue:

```
begin
  if another_disk_is_being_accessed then
    if  (the_other_disk_is_not_infected and
         the_other_disk_is_bootable) then
           copy_virus
           increase(counter)
           if hard_disk then
              store_counter_on_disk
           if counter >= 4 then
              destroy_original
end
```

Questo virus fu scoperto osservando che al file COMMAND.COM corrispondeva una data recente.

Pakistani Virus

Questo virus appartiene alla categoria dei "boot infector" e colpisce i Personal Computer IBM e IBM compatibili.

Venne dapprima osservato all'università del Delaware e, in seguito, in una forma lievemente diversa, anche all'Università di Pittsburgh, alla George Washington University, alla Università della Pennsylvania e alla Georgetown University.

Il virus si annidava all'interno del settore di boot nella DPT (disk parameter table) che contiene informazioni sul numero di facce del disco che sono state formattate, sul numero delle tracce, sul numero dei settori per traccia, sul numero dei byte per settore e così via.

Se il floppy disk era senza etichetta di volume, il virus glie ne assegnava una: "(c)Brain".

A causa di questa etichetta, il virus stesso divenne noto come "Brain". Altre versioni del virus hanno scelto come etichetta "Bufued" o "Ashar" (Università della Pennsylvania).

Dopo aver creato una etichetta di volume,il virus creava tre "bad sectors" e alcuni file nascosti sul floppy disk. I "bad sectors" erano contigui e occupavano in totale 3.072 bytes. In realtà, i settori non venivano rovinati e i dati in essi contenuti si potevano leggere usando un programma speciale.

Il codice del virus era collocato nei file nascosti o nei tre settori apparentemente rovinati.

Usando la "disk view/edit utility" di "PC Tools" è possibile visualizzare i contenuti di un settore in codice esadecimale e i corrispondenti valori ASCII.

Il testo chiaro del settore 0 contagiato contiene il seguente messaggio:

"Welcome to the Dungeon (c) 1986 Basit & Amjad (pvt) Ltd. BRAIN COMPUTER SERVICES 730 Nizam Block Allama Igbal Town Lehore, Pakistan. Phone: 430791, 443238, 2800530. Beware of this VIRUS. Contact us for vaccination".

I fratelli Alvi (Basit e Amjad) vendevano Personal Computer in un negozio di Lehore. Essi furono contattati da un giornalista e Basit Alvi ammise di aver scritto il virus nel 1986, di averlo inserito in un disco per divertimento e di averne dato una copia ad un altro studente suo amico. Nessuno dei due fratelli, tuttavia, fu in grado di spiegare in che modo il virus avesse potuto "emigrare" negli U.S.A.

Si pensa che il virus sia stato portato negli U.S.A. su dischetti contenenti programmi famosi acquistati, a causa del prezzo vantaggioso, da cittadini statunitensi in viaggio in Pakistan.

Una mappa del disco infetto mostra molti file nascosti oltre ai file di sistema BIOS e DOS (che lo sono normalmente) e i tre "bad sectors".

Con lo stesso programma utilizzato per visualizzare il settore 0, si è in grado di osservare il contenuto dei "bad sectors": in essi è stato trovato parte del messaggio di avvertimento di cui è stato appena riportato il testo.

Alla Università del Delaware, il virus infettò diverse centinaia di dischi, ne rese completamente inusabili circa l'1% e distrusse almeno una tesi di uno studente.

All'università di Pittsburgh, il lavoro di centinaia di studenti fu intaccato dal virus.

Hebrew University Virus

Questo virus appartiene alla categoria dei "general application infector" e colpisce i Personal Computer IBM o IBM compatibili.

Scoperto alla Hebrew University in Israele, era stato progettato in modo che cancellasse tutti i file ogni venerdì 13 dal 1988 in poi. La prima data in cui il fenomeno avrebbe avuto luogo era il 13 maggio 1988. Oltre ai 1000 elaboratori che componevano il sistema dell'università, il virus avrebbe colpito anche centinaia di dischetti di proprietà degli studenti e dei membri della facoltà. Come è riferito da Israel Radai, membro dello staff del centro di calcolo dell'università, il virus sarebbe stato assai diffuso a Gerusalemme e anche nella zona di Haifa (esistono stime che parlano di 10.000 o 20.000 dischetti infetti).

Il virus venne scoperto prima che riuscisse nel suo intento catastrofico. A causa di un errore intrinseco nella sua costruzione che provocava il contagio anche degli elementi già infetti, il virus rallentava a tal punto le attività da indurre un'analisi di ciò che stava succedendo.

Si notò così che ogni volta che un file. EXE veniva eseguito la sua dimensione aumentava di 1808 bytes. Anche i file .COM erano soggetti allo stesso fenomeno ma l'accrescimento si verificava una volta sola.

Amiga Virus

Questo virus apparve quasi simultaneamente sia in Inghilterra sia in Australia e sembra sia stato introdotto su un disco fornito dai distributori Amiga (il microcomputer Amiga è prodotto dalla Commodore Corporation).

Sotto questo punto di vista, è simile al Macintosh Virus.

Lo stesso virus giunse negli U.S.A., e precisamente in Florida, dove un gruppo di utenti venne contagiato. Voci non ufficiali parlano anche di infezione in altre parti degli Stati Uniti e in Canada.

Il virus agiva trasferendosi da un disco infetto nella memoria RAM e contaminando tutti i dischi che venivano usati nella stessa sessione. Risiedendo nella memoria RAM il virus poteva esser eliminato spegnendo l'elaboratore. Quando veniva utilizzato un disco infetto, appariva un messaggio sullo schermo: "Something wonderful has happened: your machine has come alive".

A questo punto l'utente non poteva più eseguire alcun file del disco.

Alcuni distributori fornirono gratuitamente rivelatori di virus agli utenti colpiti.

Flu-Shot 4 Virus

Questo virus apparve ai primi di marzo del 1988 sui "bulletin boards" spacciandosi per una versione aggiornata di "Flu-Shot 3", un programma per la scoperta dei virus. Le videate presentate all'utente durante la fase di collegamento e di illustrazione dlle finalità erano identiche a quelle del programma originale per la rilevazione di virus.

All'utente veniva, quindi, fornita la possibilità di installare tale programma nel suo sistema.

Il virus era in grado di infettare anche coloro che avevano avuto accesso solamente alla documentazione. Una volta che il virus era stato attivato, cancellava diversi settori di vitale importanza per il funzionamento dell'hard disk - se il sistema ne aveva uno - e danneggiava la DPT (disk parameter table) nel settore 0 di tutti i floppy disk presenti nel sistema.

Se un disco infettato veniva usato su un sistema protetto da programmi per la rilevazione dei virus che si basavano sul controllo dell'attività degli interrupt 13H e 26H, tali programmi non riuscivano ad impedire che il virus penetrasse nel sistema.

IBM Christmas Tree Virus

Questo virus apparve nel Dicembre 1987 sulla rete di comunicazione interna dell'IBM. L'effetto del programma era la visualizzazione di un messaggio di auguri natalizi e di un disegno di un albero di Natale.

Si duplicò moltissime volte saturando la rete con copie di sè stesso. Ogni volta che una macchina si rivolgeva al sistema diveniva infetta e contagiava a sua volta tutti i suoi dischi.

Secondo i rapporti IBM, il virus rallentò le operazioni delle reti e fu rimosso prima che potesse raggiungere gli elaboratori dei clienti.

Secondo Frank Bodes, un portavoce della IBM, in questo sistema i programmi eseguibili non possono essere trasmessi da un computer ad un altro. Tuttavia, il Christmas Tree Virus si diffuse attraverso il sistema. Sebbene non esista su questo punto alcuna informazione ufficiale da parte dell'IBM, H.J. Highland, autore dell'articolo "Computer Viruses - A Post Mortem" (Computer & Security, Aprile 1989), sostiene che il virus debba essersi inserito nei file di dati per poter essere trasmesso.

Questa affermazione avvalora la tesi di alcuni specialisti secondo i quali il pericolo di infezione non viene solo dal trasferimento di programmi eseguibili. Sembra, quindi, che un virus possa nascondersi anche all'interno di un file di dati.

Se questo fosse vero sarebbe una importante novità: soprattutto se si pensa che la maggior parte dei rivelatori di virus esamina solamente codice eseguibile.

Virus della pallina

L'effetto di questo virus del Personal Computer è la visualizzazione di una pallina che appare improvvisamente sul video e si mette a rimbalzare da un lato all'altro dello schermo.

La comparsa di questo fenomeno blocca l'elaborazione e per ripristinare il normale funzionamento del calcolatore è necessario spegnerlo: così facendo, tuttavia, si possono perdere dati preziosi contenuti in memoria centrale.

L'unico vettore di contagio è il floppy disk. L'infezione si divide in due fasi: il contagio della memoria dell'elaboratore e il contagio di un altro floppy disk.

La contaminazione della memoria di un Personal Computer può avvenire solo durante il caricamento del sistema operativo. Al momento dell'accensione dell'elaboratore, vengono dapprima eseguite le routine di diagnostica facenti parte del BIOS che valutano le risorse disponibili (numero di unità disco presenti, memoria RAM, espansioni ed interfacce varie) e provvedono a testarne le funzionalità. A questo punto viene letto dal floppy disk e posto in memoria RAM il programma "caricatore" in grado di caricare in memoria il resto del sistema operativo. Il "caricatore" risiede in una zona del disco chiamata Boot-Record.

In un disco infetto il "caricatore" è più lungo del normale: in esso è contenuta la parte del virus che si occupa di contagiare i dischi che verranno in contatto con il sistema. La seconda parte del virus si trova nel primo settore di un cluster qualsiasi che fosse libero al momento del contagio. Nel secondo settore del medesimo cluster è racchiusa la copia del Boot-record, necessaria al virus per portare a termine l'operazione di caricamento dopo essersi installato in memoria centrale.

Questo tipo di virus occupa 2 Kb: il programma "caricatore" alterato si occupa, quindi, di assegnare tale spazio in memoria e di predisporre una specie di filtro in modo che il virus possa intercettare tutte le chiamate al sistema operativo riguardanti le unità disco.

È importante sottolineare che il virus è presente all'interno dell'elaboratore prima del sistema operativo stesso: non è possibile, quindi, che il sistema operativo venga in alcun modo a conoscenza della avvenuta contaminazione.

In un sistema infetto, il virus può agire in due modi: duplicarsi o manifestarsi.

Questo dipende da eventi quali la richiesta di accesso a un disco da parte di un utente e il trovarsi in un determinato periodo di tempo.

Esaminiamo in maggior dettaglio: se un utente richiede un accesso a disco, il filtro creato dal programma "caricatore" alterato intercetta la richiesta e attiva il virus il quale controlla se i floppy disk e l'hard disk presenti nel sistema sono infetti. Se non lo sono, provvede a contagiarli immediatamente: cerca un cluster libero sul disco da infettare e in esso copia la porzione di virus che controlla l'effetto della pallina e, di seguito, il Boot-Record originale; marca il cluster come "bad" in modo che il sistema operativo lo ignori da questo momento in poi; copia al posto del Boot-Record originale la prima parte del virus.

A questo punto, il controllo viene nuovamente ceduto al sistema operativo che provvede a soddisfare la richiesta di accesso a disco dell'utente.

Tuttavia, se la richiesta di accesso avviene in un particolare momento, questo rende attiva la seconda parte del virus che si occupa dell'effetto "pallina". Infatti, per 18 millisecondi ogni 30 minuti il virus è in attesa che si verifichi un accesso a disco: se ciò accade, il controllo passa alla seconda parte del virus.

L'unica soluzione, allora, è spegnere l'elaboratore: questo elimina la pallina dallo schermo, ma non la possibilità di contaminazione (in particolare se il bootstrap avviene da hard disk infetto).

Esistono metodi per rilevare la presenza del virus sia nella memoria del computer, sia nei floppy disk.

Con il comando CHKDSK si può controllare quanta memoria RAM è libera; effettuando il comando immediatamente dopo il caricamento del sistema operativo e confrontando tale valore con quello di un sistema sicuramente libero da virus sul quale sia stato posto lo stesso tipo di sistema operativo, se l'elaboratore in esame presenta alla voce "Byte Total Memory" 2 Kb di RAM in meno rispetto all'elaboratore "sano", possiamo dire che nel calcolatore che stiamo considerando è presente il virus.

Per sapere se un floppy disk è stato contagiato o meno, si possono usare le Norton Utilities: si controlla se nel Boot-record sono presenti i messaggi di errore che compaiono nel caso in cui il bootstrap non possa essere portato a termine per qualsiasi ragione. Se non ci sono, è perchè al loro posto è stato inserito il "caricatore" alterato contenente il virus.

Scores Virus

Questo virus ha avuto origine alla Electronic Data Systems; appartiene alla classe "generic application infector" e colpisce gli elaboratori Macintosh.

Contamina qualsiasi applicazione aumentandone la dimensione di 7.000 bytes. Cerca un nuovo host ad intervalli di tre minuti e mezzo. Crea file dell'archivio appunti e del blocco note nella cartella di sistema e crea file invisibili. Cerca file specifici per distruggerli. Il sistema viene rallentato e si possono avere problemi di stampa. La dimensione dei file aumenta e si verifica un'alterazione delle icone dell'archivio appunti e del blocco note (le icone vengono sostituite da altre tipo testo con un angolo rivoltato).

Il virus può provocare crash di sistema con conseguente perdita di informazioni.

nVIR Virus

Questo virus è nato in Germania ad Amburgo, appartiene alla classe "generic application infector" e colpisce gli elaboratori Macintosh.

Esistono molte versioni del virus e ciò è dovuto al fatto che ne è stato pubblicato il codice sorgente. Molti individui si sono divertiti a modificarne alcune parti lasciando inalterata la proprietà infettiva.

I sintomi che possono far sospettare la presenza del virus nVIR variano, quindi, a secondo della versione considerata: tuttavia, crash di sistema, un beep quando viene avviata un'applicazione, il messaggio "don't panic" e la sparizione di file sono tutti indizi favorevoli alla presenza dell'infezione.

Alameda Virus

Questo virus ha avuto origine al Merritt College di Oakland in California, appartiene alla categoria "boot infector" e colpisce i Personal Computer IBM e IBM compatibili.

Esso rimpiazza il settore di boot originale spostandolo nel primo settore libero.

L'infezione si propaga attraverso la sequenza software di reboot.

Non pone nessun flag particolare sul settore di boot. Si può venire contaminati effettuando il boot da un dischetto di cui non conosciamo l'origine o inserendo un dischetto di boot sano in un sistema infetto.

I principali sintomi sono i crash di sistema, boot lenti e perdita di dati.

6. Le tecniche di difesa

Da quando i virus sono apparsi sullo scenario dell'informatica si sono moltiplicati gli sforzi tesi a difendersi da essi.

La strategia alla base di tali sforzi è insita nel nome stesso dato a questo tipo di attacco. Infatti, così come un virus di natura biologica, il virus del computer ha le seguenti caratteristiche:

- il virus del computer produce dei sintomi di infezione nel sistema (tipicamente riduzione delle prestazioni attese)

- esistono i portatori del virus (dischetti infetti o sistemi infetti)
- esiste il vettore del virus (file infetto)
- c'è una porta di ingresso del virus (drive del dischetto o lettore di rete)
- esiste la predisposizione (virus che si sviluppano su PC non si sviluppano in ambienti diversi)
- esistono i portatori sani (non si è a conoscenza del virus - ovvero non si hanno sintomi - ma esso è ugualmente presente nel sistema)
- esiste un periodo di incubazione (in cui il virus infetta altre entità ma non si manifesta)

Si può proseguire il parallelo con il virus biologico anche quando si parla di contromisure.

È possibile classificarle in tre classi, a ciascuna delle quali corrispondono sistemi antivirus (prodotti software in commercio) e misure a carattere organizzativo:

- misure di igiene e profilassi (prevenzione del virus)
- misure diagnostiche (rilevazione ed identificazione del virus)
- misure terapeutiche (eliminazione del virus).

6.1 La prevenzione dei virus

Le misure di igiene e profilassi cercano di prevenire la possibile infezione. In pratica si tratta di una serie di consigli di comportamento o di programmi speciali che cercano di evitare che avvenga il contagio.

In generale, le principali procedure da seguire (non si ha la pretesa della esaustività) affinchè sia più difficile contrarre un virus sono:

1. ogni programma presente nel sistema dovrebbe essere in forma sorgente: in questo modo un'eventuale verifica risulta facilitata;
2. se una parte del codice è in forma di codice oggetto o di codice macchina, dovrebbe provenire da fonti affidabili e ragionevolmente protette dalle infezioni;
3. bisognerebbe rendere consapevoli gli utenti del danno che un computer virus è in grado di causare;
4. sarebbe bene proibire lo scambio di dischetti con l'esterno, ossia il trasporto all'esterno delle unità interne (es. lavoro a casa) e viceversa (es. nuovo videogioco o programma provati in ufficio);
5. sarebbe bene utilizzare le protezioni "read-only" laddove è possibile;
6. si dovrebbero effettuare periodici controlli tra software sano e software esistente sull'elaboratore;
7. non bisognerebbe ricevere la posta elettronica su qualsiasi elaboratore ma usarne uno ad hoc;
8. sarebbe bene proibire l'esecuzione di qualsiasi programma ricevuto tramite posta elettronica;
9. sarebbe necessario esaminare la struttura dei file oggetto con un debugger per verificare se esistono spazi disponibili all'eventuale inserimento di un virus;
10. non si dovrebbe mai effettuare boot da dischetti diversi dagli originali protetti in scrittura (o dalle apposite copie protette in scrittura);
11. bisognerebbe avere un dischetto di boot per ogni sistema ed esso dovrebbe essere chiaramente marcato come il dischetto per quel sistema;
12. se si ha a disposizione un hard disk, non si dovrebbe mai effettuare il boot da dischetto (con l'eccezione ovvia del caso in cui si stia disinfestando il sistema);
13. sarebbe bene creare etichette di volume significative sui dischi al momento della formattazione;
14. se si opera in un ambiente distribuito bisognerebbe non usare il nodo di file server come una workstation;
15. se si usano emulatori 3270 connessi a mainframe sarebbe necessario tenere tutto il software relativo in una sottodirectory a parte e non porre codice eseguibile non necessario nella sottodirectory.

Esistono programmi antivirali ad azione preventiva. Essi rimangono sempre in memoria centrale e controllano tutta l'attività del sistema alla caccia di operazioni che possano nascondere un tentativo di duplicazione di un eventuale virus, filtrando i tentativi di accesso ai file e le richieste di servizi del sistema operativo da parte di altri programmi, controllando i programmi che vengono caricati e scaricati in memoria, le tabelle e le strutture di controllo del sistema. (Fig. 3).

Quando questi programmi si accorgono che un virus sta tentando di contaminare il sistema, interrompono le attività congelando il sistema e mandando un messaggio all'utente.

Gli aspetti negativi di questo genere di contromisura sono:

- l'elevato numero di falsi allarmi
- l'impossibilità di impedire le infezioni dei virus della classe "boot infector".

6.2 La rilevazione dei virus

Le misure diagnostiche cercano di evitare l'ulteriore diffusione del virus qualora sia riuscito a penetrare nel sistema cercando le tracce lasciate dall'infezione nel sistema.

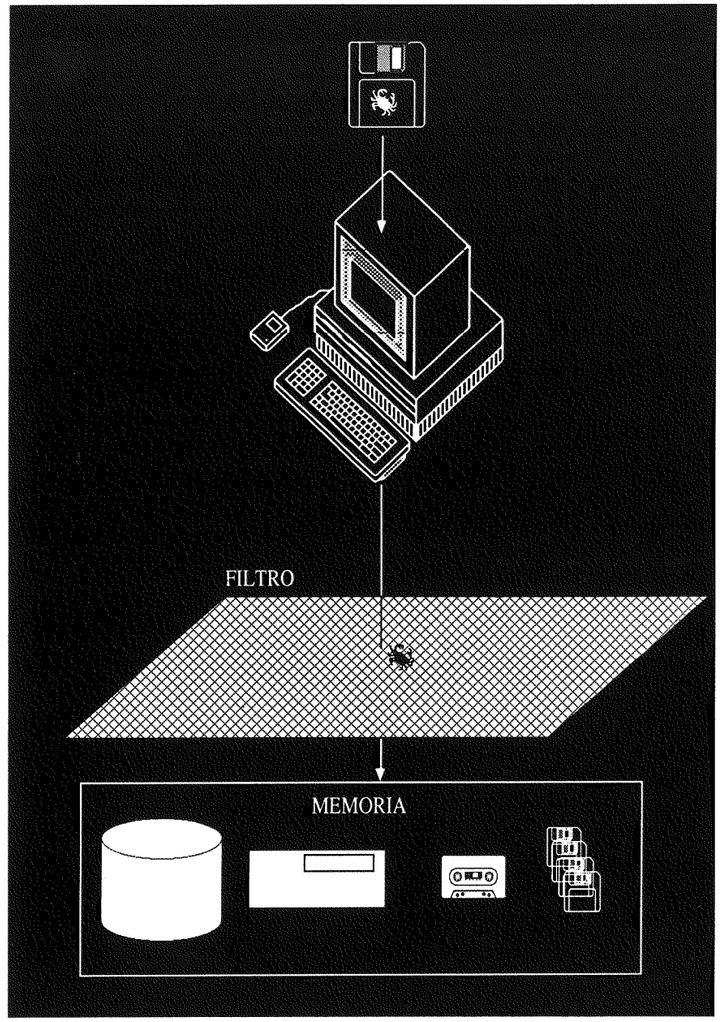
Queste tracce sono le alterazioni compiute dal virus nel sistema durante la contaminazione.

Anche in questo caso possono essere effettuate operazioni di controllo oppure ci si può affidare ad appositi programmi.

Le procedure da seguire potrebbero opportunamente comprendere:

1. il controllo dei programmi originali con copie sane e protette
2. il controllo della lunghezza di un programma
3. la verifica della data e dell'ora (ultima modifica)

Fig. 3 - La prevenzione.



4. la verifica dei totali di controllo (un totale di controllo viene eseguito e confrontato con il totale di controllo del programma originale)
5. la crittografia (tutti i file sono crittografati e vengono decrittati al momento dell'esecuzione; se un virus altera il contenuto del file l'esecuzione viene sospesa): è consigliabile un crittosistema a chiave pubblica.

   Questa contromisura non è efficace se il virus esiste nel file prima che questo venga crittografato.
6. il controllo delle etichette di volume
7. il controllo (non preciso, ma lasciato alla sensibilità dell'utente) del tempo di esecuzione.

I programmi antivirus ad azione diagnostica agiscono principalmente secondo due tecniche:

- la tecnica della vaccinazione
- la tecnica del confronto

La tecnica della vaccinazione include nei programmi un meccanismo di test che verifica per mezzo di un algoritmo (ad es. checksum) se la sequenza delle istruzioni è stata alterata. Questo test va in funzione ogni volta che il programma viene eseguito e controlla che non ci siano cambiamenti dall'ultima esecuzione. Se c'è una qualsiasi differenza, si presume la presenza di un virus e viene inviato un messaggio. Purtroppo la vaccinazione non funziona per il segmento di boot: infatti, nella maggior parte dei casi, il virus rimpiazza interamente il settore e quindi il settore di boot vaccinato non andrà mai in esecuzione e anche se ciò accadesse non si creerebbe alcuna anomalia perchè il settore di boot non è stato alterato ma solo spostato.

Inoltre, se un virus riesce ad infettare un programma vaccinato, sicuramente viene eseguito prima del meccanismo di controllo; quindi, può replicarsi indisturbato e disabilitare il meccanismo stesso.

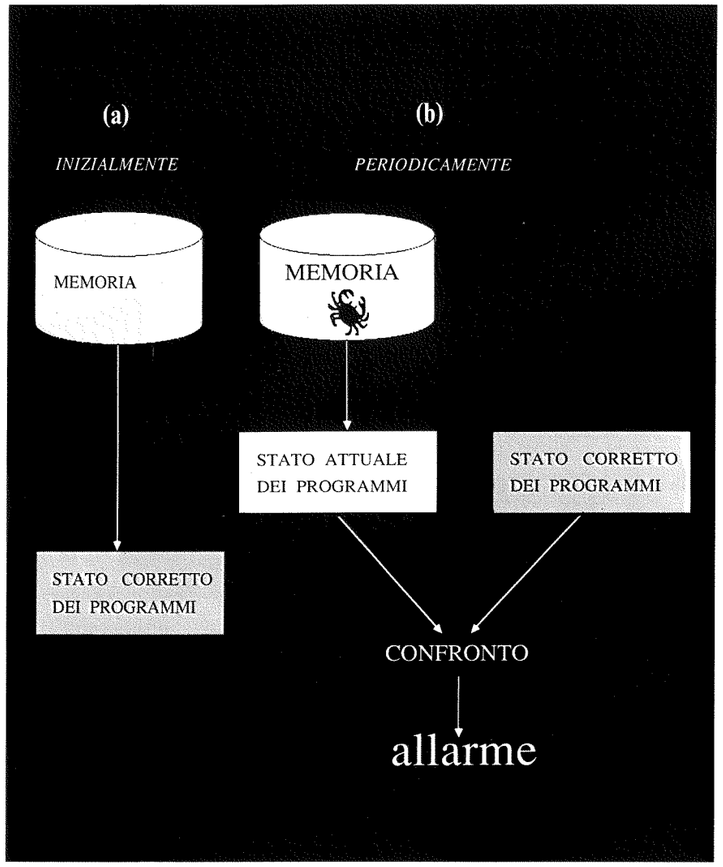


Fig. 4 - Scoperta per confronto:
(a) descrizione dello stato dei programmi (inizialmente);
(b) analisi dello stato attuale dei programmi (periodicamente).

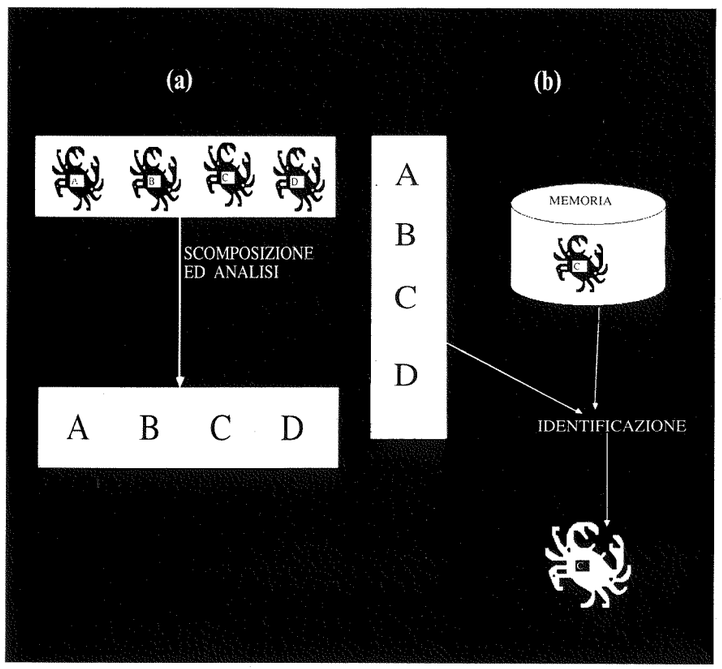


Fig. 5 - Identificazione:
(a) analisi dei virus e creazione di modelli descrittivi;
(b) esame del sistema ed identificazione del virus.

La tecnica del confronto (Fig. 4) consiste nella memorizzazione di tutte le informazioni critiche del sistema al momento dell'installazione iniziale. In seguito, viene eseguita una routine di confronto tra la situazione attuale e quella iniziale.

Se vengono scoperte tracce di infezione, viene identificata l'area interessata e viene informato l'utente.

Con questa tecnica possono essere controllate tutte le aree del sistema (settore di boot, aree riservate al sistema operativo, ecc.) ed inoltre il file con la fotografia del sistema può essere tenuto off-line (quindi, al sicuro da eventuali contaminazioni).

Lo svantaggio di questa tecnica, in grado di riconoscere i virus della classe "boot infector", è la considerevole quantità di tempo che impiega per effettuare la diagnosi. C'è una bassa probabilità che il virus possa essere attivo prima che il programma di diagnosi venga eseguito per la verifica (ad esempio, se il virus contiene una bomba logica a tempo).

6.3 L'eliminazione dei virus

Le misure terapeutiche si occupano di riportare il sistema nelle condizioni in cui si trovava prima che la contaminazione avesse luogo.

Anche in questo caso si possono seguire procedure oppure eseguire programmi specifici.

Le procedure da adottare potrebbero essere le seguenti.

Anzitutto determinare l'estensione dell'infezione.

Se il virus appartiene alla classe "system infector" o "general application infector" e se ha colpito il disco fisso:

1. spegnere il sistema infetto;
2. reperire il dischetto originale, proteggerlo dalla scrittura, metterlo nel drive e avviare il sistema dal dischetto;
3. assicurarsi che il sistema sia partito correttamente;
4. effettuare il backup di tutti i file non eseguibili su dischetti o nastri appena formattati. Se le copie di backup vengono memorizzate su un altro disco fisso, assicurarsi che non sia infetto. (Se si hanno dubbi in proposito, assumere che sia contagiato). Non usare il programma di backup del disco fisso ma quello del dischetto originale. (In nessun punto di questa procedura si devono usare programmi tratti dal disco fisso infetto);
5. elencare tutti i file batch del disco infetto. Se una linea qualsiasi di questi file sembra strana non effettuare il backup. In caso contrario eseguire il backup;
6. effettuare una formattazione a basso livello del disco infetto;
7. ripristinare il sistema operativo sul disco fisso;
8. ristrutturare le directory;
9. ripristinare gli altri programmi eseguibili dai pacchetti originali;
10. ripristinare i file in cui era stato fatto il backup;
11. raccogliere tutti i dischetti che possono essere stati inseriti nel sistema negli ultimi anni (in genere da quando sono comparsi i primi virus sulla scena mondiale);
12. a discrezione, distruggere tutti i dischetti o eseguire il backup di tutti i file non eseguibili su floppy sani (formattati);
13. formattare i dischetti sospetti.

Se il virus appartiene alla classe "boot infector":

1. spegnere il sistema,
2. effettuare il boot da dischetti originali protetti in scrittura,
3. rimpiazzare il sistema operativo ed il settore di boot su tutte le entità infette.

I programmi antivirali (Fig. 5) ad azione terapeutica funzionano cercando le caratteristiche specifiche di un virus: un particolare pezzo di codice, un flag, un'etichetta, ecc.

Quando viene trovato un elemento che identifichi con certezza il virus, questo viene individuato e tolto.

Questi programmi esistono solo dopo che il virus è stato attentamente studiato, esaminato, sezionato e si è trovato un antidoto: un processo del genere può impiegare svariati anni per arrivare a compimento.

7. Osservazioni conclusive

In questo lavoro sono state sommariamente esposte alcune considerazioni generali sui virus del calcolatore, su come combatterli e su come difendersi. Le tecniche di difesa dai virus vanno in ogni caso viste nell'ambito del sistema di sicurezza informatica di cui sono parte integrante. Va osservato che spesso nelle organizzazioni i sistemi di sicurezza sono inadeguati alle esigenze di protezione.

Certamente fra le cause di inadeguatezza dei sistemi di sicurezza vanno annoverate: la scarsa sensibilità al problema da parte dell'alta direzione, la quale spesso è fuorviata da rapporti del manager EDP tendenti a non presentare i problemi di sicurezza ma tendenti ad un efficacia ed efficienza dei sistemi informativi sempre più spinte; la cronica carenza dei vari specialisti della sicurezza; una domanda che non giustifica forti investimenti da parte dei fornitori; un'attività di ricerca non sufficientemente distribuita e finanziata; la confusione fra sistema di sicurezza e assicurazione contro i crimini informatici (spesso a torto la polizza di assicurazione, comunque opportuna in quanto nessun sistema di sicurezza offre una garanzia assoluta, viene intesa come sostitutiva e non integrativa del sistema di sicurezza); una scarsa opera di sensibilizzazione verso gli utenti (si pensi alla incuria con cui spesso viene gestita la tessera Bancomat da parte del legittimo possessore).

Da questa analisi necessariamente sommaria possono derivare alcune linee di intervento.

La costituzione di banche di dati sui virus e sui crimini informatici; forme di sensibilizzazione (convegni e seminari) rivolti al management e agli utenti; corsi specialistici a carattere tecnico; una legislazione più

adeguata; una attività di ricerca nelle università e nelle industrie che veda impegnate risorse adeguate alla rilevanza del problema; un impegno da parte dei progettisti dei sistemi informativi a considerare i problemi di sicurezza fin dalle fasi iniziali dei progetti; una continua revisione dei propri sistemi di sicurezza, in modo da reinizializzare continuamente la curva di apprendimento degli attaccanti, sono tutte attività che possono notevolmente contribuire ad affrontare nel giusto verso un problema che è destinato ad assumere sempre più rilevanza.

Bibliografia

[1] RADAI Y., *"The Israeli PC Virus"*, Computer & Security, 1989;

[2] A. BORRIONE, M. MEZZALAMA, F. MONTI, *"Il virus del personal computer"*;

[3] BRUNO A., *"Sistemi di protezione contro i crimini informatici: valutazioni e proposte"*, DSI - Tesi di Laurea - 1989;

[4] CREMONESI C., *"Un modello per l'analisi e la classificazione dei crimini informatici"*, DSI - Tesi di Laurea - 1989;

[5] FERRARI M., *"I virus Macintosh"*, DSI - Rapporto interno - 1989;

[6] HIGHLAND H. J., *"Anatomy of a virus attack"*, Computer & Security, 1988;

[7] KLUNDER D., *"Virus update"*, Datamation, 1989;

[8] Mc AFEE J., *"Managing the virus threat"*, Computerworld, 1989;

[9] Mc AFEE J., *"The virus cure"*, Datamation, 1989;

[10] MURRAY W.H., *"The application of epidemiology to computer viruses"*, Computer & Security, 1988;

[11] VAN WYK K.R., *"The Lehigh Virus"*, Computer & Security, 1989;

L'Interscambio Elettronico dei Dati (EDI)

LEONARDO FELICIAN

Dipart. di Ingegneria Elettronica e Informatica
Università di Trieste

1. Definizione del problema

Poche tecnologie sono altrettanto pervasive quanto l'informatica, che offre strumenti e tecniche per elevare gli obiettivi delle aziende, dalla ricerca dell'efficienza al conseguimento dell'efficacia, spingendo la velocità di reazione e di innovazione, essenziale per ottenere vantaggi competitivi sui mercati.

Nonostante gli ingenti investimenti in sistemi di elaborazione dati, finora il computer è stato usato come un mezzo per automatizzare procedure concepite nell'era preinformatica e ritagliate sul modo di lavorare degli uomini, senza mai abbandonare il riferimento cartaceo, anche a causa di vincoli di natura giuridica.

I veri ritorni delle procedure di automazione si vedono però nell'integrazione del trattamento a ciclo completo di un'unità di lavoro, e ciò richiede al giorno d'oggi - per la natura stessa delle relazioni internazionali - una sempre maggior attenzione al problema delle comunicazioni. In questo modo si accorciano le distanze geografiche e si contribuisce a spianare i solchi storici che hanno diviso da sempre le aree a forte sviluppo da quelle più arretrate del nostro continente.

La frontiera delle telecomunicazioni in Europa e in Italia si chiama oggi Interscambio Elettronico dei Dati (*Electronic Data Interchange, EDI*): con questo termine, spesso abusato, ci si riferisce al trasferimento a mezzo elettronico di messaggi strutturati (ordini, fatture, polizze, trasferimenti di fondi, comunicazioni in genere) in formato conforme a standard predefiniti.

Questa definizione mette in chiaro la differenza saliente tra l'Interscambio Elettronico dei Dati e l'altra emergente tecnologia della posta elettronica: l'interesse principale dell'EDI non è tanto nella trasmissione, quanto nella preventiva standardizzazione dei dati, che dovranno essere automaticamente riconoscibili ed immediatamente elaborabili una volta giunti a destinazione.

Le dimensioni del problema non sono trascurabili: un recente studio delle Nazioni Unite ha calcolato che nel mondo vengono spesi più di 60 mila miliardi di lire all'anno per documentare "in modo cartaceo" i normali rapporti commerciali. Gran parte delle aziende coinvolte in questo scambio di informazioni dispone però oggi di un elaboratore, costretto a stampare enormi volumi di carta che, tramite il canale postale, vengono recapitati ad altri enti, obbligati a loro volta a immettere i dati nei propri sistemi informativi. Questo processo di comunicazione, ideato ai tempi in cui la corrispondenza veniva prodotta, archiviata ed elaborata manualmente, è molto poco efficiente nella società informatizzata.

Per comprendere i limiti dei tradizionali supporti di comunicazione e la necessità di standardizzare basta prendere in esame i risultati di uno studio riguardante le usuali transazioni commerciali tra aziende:

- negli Stati Uniti il 70% degli output elaborati dai computer viene oggi emesso su supporto cartaceo: considerando la situazione italiana questa percentuale è probabilmente molto sottostimata, perchè è ancora assai limitato tanto lo scambio diretto di supporti magnetici, quanto la possibilità di interrogare a video dati residenti nel sistema informativo di un'altra azienda;
- il 90% dei dati spediti su supporto cartaceo viene reimmesso in altri computer con pesanti attività di *data entry*, per le quali si incominciano a studiare soluzioni semiautomatiche (lettori ottici) ai fini di ridurre tempi, costi e margini di errore;
- qualunque protezione e procedura di controllo delle immissioni di dati sia messa in funzione, l'immissione manuale di dati in un sistema comporta un margine di errori stimato fino al 25%;
- tenendo conto delle attività necessarie per il controllo e la correzione degli errori, il costo della pura operazione di immissione dati può raggiungere il 25% del costo totale di elaborazione di una transazione; a queste considerazioni si deve aggiungere il ritardo di trasmissione delle informazioni introdotto dal canale postale: si stima che il ritardo medio dovuto a questo sistema di spedizione tra due città europee sia di sette giorni lavorativi, ma purtroppo in Italia l'intervallo è spesso maggiore;

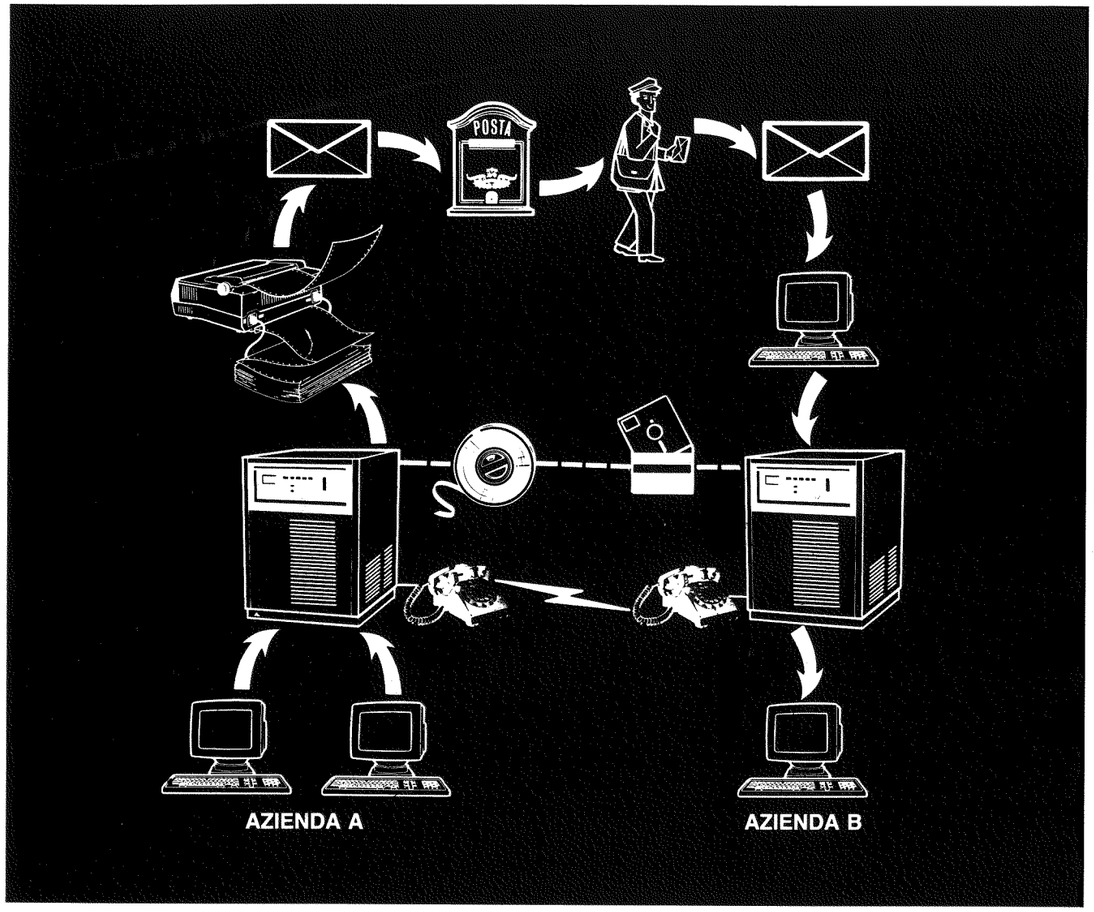


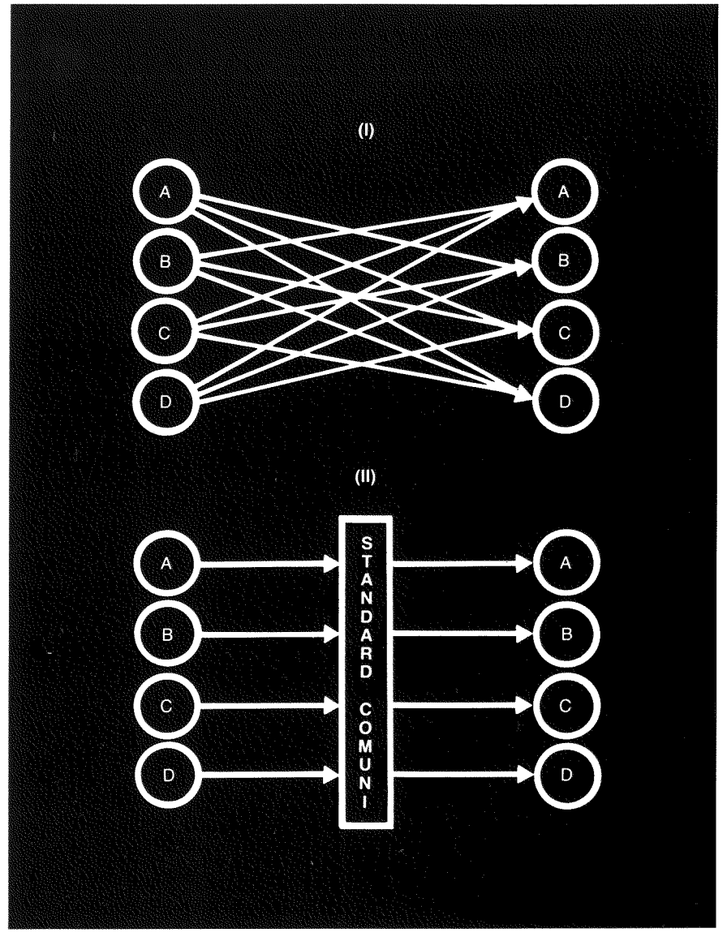
Fig. 1 - EDI (Electronic Data Interchange) significa lo scambio di documenti (ordini, fatture, polizze, ecc.) direttamente tra elaboratori tramite mezzi di telecomunicazione.
Esso sostituisce, con grandi vantaggi, lo scambio mediante i tradizionali documenti cartacei o attraverso supporti magnetici.

- il costo di trasferimento dei dati su carta è stimato in circa 300 miliardi di dollari all'anno a livello mondiale.

Come in molti altri casi, anche di fronte a questo problema di vaste proporzioni la mancanza della capacità di innovare conduce a costi crescenti con risultati modesti. La soluzione più naturale, che interviene a monte nel processo, e quindi "taglia" costi, ritardi ed errori consiste nel mettere direttamente in contatto gli elaboratori delle aziende interessate ad una certa transazione (fig. 1).

Le tecniche EDI interessano dunque trasversalmente una grande varietà di settori di attività: industria, distribuzione, banche, trasporti, servizi in generale. Non basta però un accordo privato tra due o più partner interessati ad una transazione per dare il via ad una applicazione EDI vera e propria: mentre infatti per lo scambio elettronico di informazioni tra due corrispondenti può essere utilizzato qualunque metodo concordato, l'aumento combinatorio delle possibilità di comunicazione (fig. 2) fa divergere i costi di questa soluzione in "lingue" diverse all'aumentare del numero dei partner coinvolti. In presenza di uno standard, infatti, sarà sufficiente che ciascuno degli *n* corrispondenti predisponga un protocollo di spedizione ed uno di ricezione, per un totale di *2n* interfacce da realizzare con diversi sistemi informativi, mentre se ciascuno stabilisce standard diversi con ciascun partner le interfacce da realizzare e successivamente da mantenere sono in numero enorme, n^2-n.
Da queste considerazioni e dal fatto che il numero delle organizzazioni che usano l'EDI sta raddoppiando

Fig. 2 - Numero di connessioni tra unità diverse.
Senza standard comuni (I), tale numero è dato da n^2-n; con standard comuni (II) da 2xn.
(Per n=100 si hanno quindi 9.900 connessioni nel primo caso, contro 200 nel secondo).



ogni anno, segue la necessità di procedere alla definizione di standard che siano pubblici e non privati. La standardizzazione procede pertanto per gradini successivi, con standard proprietari, industriali, nazionali ed internazionali. Gli standard *de facto* precedono molto spesso quelli *de iure* e dipendono dalla forza e dalla capacità di diffusione di chi li propone nella pratica.

L'EDI si propone di soddisfare standard di:

- autenticità
- integrità
- confidenzialità
- continuità

Per raggiungere questi obiettivi è necessario innanzitutto codificare i dati aziendali in formato elettronico, con contenuti e sintassi definiti e concordati tra tutti i possibili partner (presenti e futuri) interessati all'interscambio di un certo tipo di dati.

In particolare va effettuata un'accurata analisi dei documenti cartacei, per estrarne

- la sintassi (*syntax*), cioè l'insieme di regole che controllano la struttura del messaggio (con questo termine, per certi versi improprio, ci si riferisce a qualcosa di simile alla grammatica di un linguaggio umano);
- le informazioni elementari (*data elements*), che vanno a costituire un *dizionario dati* e sono in un certo senso equivalenti alle parole del linguaggio umano;
- i codici standard (*standardized codes*), cioè i codici usati per rappresentare ogni informazione elementare;
- i segmenti (*segments*), insiemi logici di informazioni elementari, che raggruppano dati omogenei;
- i messaggi (*messages*), aggregati di segmenti secondo le regole della sintassi.

Da un punto di vista informatico, le transazioni EDI possono infine essere di lunghezza fissa o variabile. Le prime applicazioni sono state caratterizzate da messaggi di lunghezza fissa (ciò vale ad es. per quasi tutte le banche americane che utilizzano l'EDI per le loro compensazioni). Benchè richiedano una sintassi più sofisticata, i formati di lunghezza variabile sono più efficienti (fino al 70%), per evidenti ragioni di compattamento nella trasmissione.

2. Storia dell'EDI e standardizzazioni internazionali

Le attività di standardizzazione, che per loro natura si protraggono nel tempo poichè necessitano sempre della ricerca di un largo consenso, sono state avviate per diversi settori merceologici ed hanno già prodotto i primi risultati concreti.

La storia dell'Interscambio Elettronico dei Dati ha origini lontane, negli Stati Uniti [4]. Nel 1968 venne infatti costituito il *Transportation Data Coordinating Committee* per uno studio preliminare sui problemi connessi allo scambio internazionale di documenti e di merci. Dopo sette anni di lavoro, i vari gruppi del comitato americano arrivarono alla pubblicazione di una documentazione degli standard approvati. Successivamente, nel 1980, vennero definiti i primi standard per settore merceologico, che soddisfacevano tutti le regole dello standard generale ANSI ASC X12 ed utilizzavano un dizionario base di nomenclatura emanato dall'ONU: il TDED (*Trade Data Element Directory*).

Il passo seguente avviene nel 1983, con la partenza del progetto COST306, orientato al settore dei trasporti.

Finalmente, due anni dopo, nel 1985, incomincia a muoversi qualcosa anche in Europa nel settore manifatturiero: l'industria automobilistica europea lancia ODETTE, un proprio progetto orientato al settore manifatturiero. Negli anni successivi vengono definiti rispettivamente EDIFICE, un progetto di standard per l'industria elettronica europea, CEFIC, orientato all'industria chimica, e CADDIA per il commercio.

Nel 1987 nasce in Europa lo standard *EDIFACT (Electronic Data Interchange for Administration, Commerce and Transport)* orientato all'applicazione in settori merceologici diversi, basato comunque sullo standard ASC X12 e sullo stesso dizionario dati.

L'armonizzazione dello standard americano con quello europeo diventa a questo punto un passo obbligato per procedere insieme verso una più vasta diffusione dell'Interscambio Elettronico dei Dati: grazie anche al deciso appoggio della Comunità Economica Europea, che vara il progetto TEDIS (*Trade Electronic Data Interchange Systems*) [3], questo importante obiettivo, incoraggiato anche dalle Nazioni Unite, dà finalmente origine nel 1988 allo standard unificato UN/EDIFACT. Da questo momento le modalità di Interscambio Elettronico dei Dati si presentano come uno strumento sicuro, universale e pronto per una rapida diffusione.

Esiste una struttura ufficiale dell'organizzazione per gli standard in campo EDI a livello di Nazioni Unite.

Informazioni generali sullo stato di avanzamento dei lavori possono essere ottenute al seguente indirizzo: EDIFACT Secretariat, Commission of the European Communities, DG XIII/D-5, 200, Rue de la Loi, B-1049 Bruxelles.

A livello italiano non manca attenzione ed interesse per il tema della standardizzazione: patrocinata da Confcommercio, Confindustria, Unioncamere e SIP, si è tenuta a Roma nel novembre 1987 la prima conferenza italiana sull'Interscambio Elettronico dei Dati, con l'obiettivo di creare un sistema che consenta di trasmettere da un computer all'altro ordini, bollette, fatture, pagamenti, messaggi, informazioni. È sufficiente citare poi l'attività di promozione svolta da Ediforum Italia, un'emanazione del Forum Telematico Italiano, associazione guidata dal prof. Gesualdo Le Moli del Politecnico di Milano, cui partecipa una molteplicità di aziende di vari settori, e la presenza dell'ANIA (Associazione Nazionale tra le Imprese di Assicurazione) all'interno della sottocommissione EDIFACT che sta definendo la struttura standard delle transazioni assicurative.

3. Esempi di applicazione

L'Interscambio Elettronico dei Dati in Italia non è però fermo al palo di partenza, ma è già in grado di proporre realizzazioni funzionanti di tutto rispetto.

Tra le prime in ordine di tempo va ricordato il prototipo nel campo della logistica funzionante presso la Benetton: lo stabilimento di Ponzano Veneto (Treviso) è un sistema complesso, che coordina l'attività di circa 200 fornitori di materie prime e semilavorati distribuiti in tutto il mondo e di 500 laboratori esterni all'azienda che producono i capi di abbigliamento per Benetton. La rete di comunicazioni dell'azienda veneta si appoggia sui servizi della General Electric, collegando non solo i fornitori, ma anche 5000 punti di vendita in 70 diversi Paesi, con una forte presenza negli Stati Uniti, in Belgio, Olanda, Germania e Francia. Il legame teleinformatico diretto tra il sistema dei clienti e quello dei fornitori permette alla Benetton di risparmiare circa il 20% sui tempi di consegna e di smistamento della merce verso gli Stati Uniti e il 15% sulle consegne in Gran Bretagna. Considerando che il volume delle spedizioni negli Stati Uniti è di circa 6 milioni di capi all'anno, i vantaggi conseguiti con il varo di questa iniziativa nel campo dell'EDI sono tangibili e giustificano gli investimenti effettuati.

Il 1989 rappresenta un anno importante per l'EDI nel gruppo Fiat Auto, con il progetto di connettere gradualmente 400 grandi fornitori per l'invio di oltre 1 milione e 300 mila fatture all'anno. Il risparmio in termini di minor lavoro amministrativo sarà certamente notevole: la General Motors, che ha già esperienza negli Stati Uniti di soluzioni consimili, ha dichiarato un risparmio di circa 200 dollari per automobile prodotta. In questo settore la FIAT ha varato fin dal 1987 una joint venture destinata ad offrire servizi di rete agli operatori di diversi settori merceologici.

Anche la Montefluos, azienda del gruppo Montedison, si è mossa verso soluzioni EDI, con una riduzione dichiarata del 30% in termini di ridondanza organizzativa.

Tralasciando in questa sede la rete interbancaria SWIFT, costituita già nel 1977, e passando invece al settore assicurativo, si è costituita a Bruxelles nel settembre 1988 una sottocommissione in seno al gruppo EDIFACT dedicata allo studio di uno standard in campo assicurativo. Alla riunione d'avvio hanno partecipato una quarantina di delegati, provenienti per la maggior parte da associazioni di categoria, da organismi già all'epoca impegnati nella costruzione di reti e da compagnie di assicurazione. Sono stati formati due separati gruppi di lavoro, dedicati rispettivamente all'assicurazione diretta e alla riassicurazione. I risultati crescenti di queste attività di standardizzazione si vedranno in due importanti progetti attualmente in fase di realizzazione: in ambito europeo è infatti in fase di avanzata definizione la rete RINET (*Reinsurance and Insurance Network*) dedicata alle transazioni di riassicurazione, che per loro natura avvengono su scala internazionale; in campo italiano, invece, è nata a fine 1988 la Rete Italiana Teleinformatica per le Assicurazioni (RITA), società consortile a responsabilità limitata, che ha il compito in un primo tempo di connettere le Direzioni delle Compagnie con la propria periferia.

La natura stessa del lavoro assicurativo mette infatti in luce come in Italia le transazioni siano originate non all'interno dell'impresa di assicurazione, ma in una realtà esterna, e cioè in agenzia: è in quella sede che avviene il contatto con il cliente e l'origine della transazione (emissione o variazione di polizza, denuncia di sinistro, etc.), che viene successivamente trasmessa in Direzione. Nel lavoro assicurativo si manifesta perciò il tipico problema della transazione tra più imprese, con tutti i risvolti economici, normativi e gestionali connessi.

Per rimanere nel campo di questi ultimi, visto il grande volume di transazioni nel settore delle polizze individuali, non va sottovalutato il problema della trasmissione fisica dei dati dalle agenzie - dove essi sono originati - alle Direzioni, dove vengono conservati ed elaborati. Dalla tempestività e dall'affidabilità di queste comunicazioni dipendono in buona parte i costi di gestione dell'attività, non meno che l'immagine di puntualità e di precisione che viene data al cliente.

L'obiettivo di connessione è però soltanto il primo passo per RITA, nel quale la rete si limita a fare da vettore di informazioni strutturate in maniera proprietaria da ciascuna Compagnia di assicurazione: per poter pensare alle applicazioni interassicurative l'EDI è un passo obbligato, con la definizione preventiva di un "gergo" comune: c'è già un esempio in Belgio, dove la rete nazionale Assurnet è decollata proprio grazie al poderoso lavoro di standardizzazione di Telebib, una specie di Bibbia delle telecomunicazioni che fissa nel dettaglio il significato e le caratteristiche infomatiche di tutte le informazioni assicurative.

4. Validità giuridica del documento elettronico

È chiaro che l'applicazione estensiva di modalità di Interscambio Elettronico dei Dati ha una dimensione economica tutt'altro che trascurabile, che alcuni studi quantificano in una diminuzione di costi intorno a 5000 miliardi di lire all'anno in Italia, pari al 10% del costo dei prodotti manufatti. Gran parte di questi risparmi sono però *potenziali*, non necessariamente destinati a tradursi in realtà, ma semplicemente riassorbiti dalle spese per l'impianto di sistemi di trasmissione di dati, a meno che non si possa agire in maniera completa, eliminando veramente la carta dal ciclo di trattamento e realizzando quel sogno di *"paperless trade"* che l'informatica promette da tempo, ma che la legislazione non accenna a recepire.

L'Interscambio Elettronico dei Dati richiede dunque di riconoscere una validità giuridica per i documenti elettronici: quest'ultimo tema è stato oggetto di discussione nel 1988 nel IV Congresso Internazionale su "Informatica e Regolamentazioni Giuridiche" organizzato a Roma dalla Suprema Corte di Cassazione [2], nonchè al Circolo della Stampa di Milano durante il convegno dell'Associazione Internazionale di Diritto Assicurativo (AIDA) su "Forma e assicurazione".

La via per attribuire validità giuridica al documento elettronico passa attraverso tre fondamentali requisiti del documento:

- la leggibilità
- l'inalterabilità
- la non disconoscibilità

È soprattutto quest'ultimo punto, e cioè la prova certa dell'assenso del soggetto interessato, solitamente espressa con la firma autografa sul supporto cartaceo, ad essere in discussione, giacchè il giurista è chiamato a garantire senza eccezioni la parte più debole nell'effettuazione di una transazione.

Nonostante ciò, è già oggi possibile immaginare soluzioni informatiche atte a garantire il rispetto di questi requisiti, tramite l'impiego di particolari tecniche di compressione di dati e la conservazione in luogo separato delle "chiavi" degli algoritmi di cifratura, come ampiamente discusso in [1]. Il problema si sposta allora sul piano normativo, dove tra l'altro l'estrema lentezza del nostro sistema parlamentare non facilita certo il compito all'Italia rispetto ad altre Nazioni, e in particolare a Paesi come la Gran Bretagna basati sulla forma giuridica della *Common Law:* nonostante questa posizione di svantaggio, anche all'interno della nostra normativa è possibile incominciare ad intravedere qualche possibilità di cambiamento.

Una delle attività principali del già citato Ediforum consiste appunto nella sensibilizzazione dei nostri legislatori verso il problema normativo, che rischierebbe di penalizzare in maniera eccessiva in questo settore le imprese italiane in vista della competizione sul mercato unico europeo.

5. Conclusioni e prospettive

Da quanto visto, risulta chiaro che l'Interscambio Elettronico dei Dati ha sostanzialmente due aspetti: uno burocratico, legato alla diffusione e all'accettazione di certi standard, l'altro tecnologico, che si concretizza nell'imponente lavoro di adeguamento dei sistemi informativi aziendali per interfacciare le modalità di scambio concordate.

È difficile fare previsioni adeguate sulle potenzialità di penetrazione di queste tecniche perchè non esiste campo di applicazione che non possa venir toccato in maniera profonda dalla disponibilità di un mezzo di trasmissione immediato, preciso e soprattutto automaticamente elaborabile. Al di là degli esempi già citati, si pensi ad esempio al sistema dei trasporti e della distribuzione, alla movimentazione di merci ed in particolare di derrate deperibili o - come nel caso della Benetton - di articoli di moda, al settore del turismo, etc.

Una considerazione che deve far riflettere riguarda il fatto che le applicazioni più importanti dell'EDI si ritrovano oggi presso le aziende manifatturiere, dove il movimento della carta (e con essa dell'informazione) non costituisce l'oggetto dell'attività lavorativa, anche se ne forma un supporto indispensabile. Paradossalmente le aziende che si muovono nel mondo dei servizi sono meno attente al problema, o ne considerano spesso una porzione limitata, non standardizzata e orientata piuttosto verso la posta elettronica, per garantire la tempestività, che è vitale nel terziario, piuttosto che l'elaborabilità propria dell'EDI.

Per un'azienda di servizi, invece, l'EDI dovrebbe essere un po' quello che è il CIM per le aziende manifatturiere: una metodologia di produzione in grado di cambiare radicalmente il posizionamento di un'impresa sul mercato. Le prospettive aperte da un'introduzione massiccia di tecniche EDI nel settore dei servizi non sono dunque meno rivoluzionarie (e traumatiche in termini di contenimento e riqualificazione dei posti di lavoro) dell'introduzione delle catene di montaggio automatizzate e dei robot nelle fabbriche.

È proprio questo aspetto del problema, legato all'accettazione delle nuove tecnologie all'interno di un modo di lavorare consolidato da decenni, che potrebbe costituire un freno alla diffusione dell'Interscambio Elettronico dei Dati, come messo tra l'altro in evidenza dall'esperienza della Montefluos, che ha scelto una strategia di introduzione graduale proprio per vincere eventuali resistenze psicologiche, sempre presenti quando l'integrazione del trattamento automatico dell'informazione compie un deciso balzo in avanti. Questa tattica non è da sottovalutare: meglio standardizzare ed utilizzare poche e semplici transazioni piuttosto che discutere per decenni per creare un monumento alla burocrazia che non verrà mai messo in pratica.

Riassumendo, i benefici dell'introduzione di un sistema EDI per un'azienda sono:

- risparmio di tempo nel trattamento delle transazioni;
- riduzione di errori ed eliminazione dei costi conseguenti alle azioni di correzione;
- riduzione del personale addetto ai lavori amministrativi;
- miglior controllo dei magazzini e possibilità di diminuire le giacenze medie;
- possibilità di utilizzare tecniche di produzione o di distribuzione *just-in-time.*

Dall'affermazione di questo nuovo esperanto nelle comunicazioni tra computer deriverà dunque una serie di benefici, di cui senza dubbio il più importante in prospettiva sarà la riduzione dei compiti umani di trattamento e controllo di quanto un elaboratore può trattare e controllare da solo, con conseguente riposizionamento della soglia di separazione uomo-computer.

Bibliografia

[1] G. ABIUSO, L. FELICIAN:*Società dell'informazione, documento e assicurazione*, Atti Convegno Forma e assicurazione, AIDA, Milano, 1988.

[2] R. BORGINI, L. FELICIAN: *Validità giuridica del documento elettronico*, Corte Suprema di Cassazione, Centro Elettronico di Documentazione, Atti del 4° Congresso Internazionale su Informatica e regolamentazioni giuridiche, Roma, 1988.

[3] Commission of the European Communities (ed.): *TEDIS: a community programme for cooperation*, Directorate General XIII, Telecommunications, Information, Industries and Innovation, Bruxelles, 1988.

[4] Ediforum Italia (ed.): *EDI: Electronic Data Interchange*, Milano, 1989.

La tecnologia di montaggio superficiale

GIUSEPPE BARONI

Bull Italia
Centro di ricerca e progettazione
Pregnana Milanese

1. Introduzione

In questi ultimi anni il packaging elettronico e le tecnologie di interconnessione hanno subito una grande evoluzione al fine di ottenere una ulteriore miniaturizzazione delle apparecchiature elettroniche.

Questo cambiamento è stato determinato da varie esigenze, in particolare dalla necessità di ridurre peso e dimensioni e di ottenere una riduzione di costi a parità di prestazioni.

La tecnologia di montaggio superficiale, SMT (*Surface Mount Technology*), si inserisce in questo trend evolutivo e il suo utilizzo rappresenta una importante opportunità per ottenere prodotti elettronici con dimensioni contenute, a più alta affidabilità e con migliori prestazioni circuitali.

Inoltre, in prospettiva, questa tecnologia offre la possibilità di avere significative riduzioni di costo, soprattutto in applicazioni con alto volume produttivo.

2. Che cosa si intende per SMT

Nei circuiti stampati tradizionali i componenti con terminali sono inseriti nei fori praticati nel circuito stesso e successivamente vengono saldati facendo passare lo stampato su un'onda di stagno-piombo fuso che bagna i terminali dei componenti.

Il montaggio superficiale consente invece di posizionare i componenti, dotati di un appropriato package, appoggiandoli direttamente sulla piastra e saldandoli poi con un particolare processo di rifusione.

In figura 1 sono mostrati due tipici componenti SMT: un integrato di tipo SOP *(Small Outline Package)* ed uno di tipo PLCC *(Plastic Leaded Chip Carrier).*

Quindi, mentre la tecnologia tradizionale consente di utilizzare un solo lato del circuito stampato per il posizionamento dei componenti, il montaggio superficiale permette di posizionare i componenti su entrambe le facce.

I componenti usati per il montaggio superficiale, denominati SMD (*Surface Mounted Device*), avendo dimensioni inferiori a quelle dei componenti tradizionali, necessitano per il posizionamento sulla piastra di macchine automatiche.

Essi vengono fissati al circuito stampato tramite una pasta a saldare ed eventualmente con un adeguato adesivo.

La saldatura viene effettuata con tecniche di rifusione; nel caso di componenti passivi, essi vengono montati sul lato inferiore della piastra, attraverso un passaggio su un'onda di stagno-piombo fuso.

3. Da dove viene questa tecnologia

I primi SMD e quindi la tecnologia di montaggio superficiale fu sviluppata in Europa alla fine degli anni '60 per applicazioni nell'industria svizzera degli orologi, dove la necessità di miniaturizzazione era già profondamente sentita.

Nello stesso periodo, negli USA furono sviluppati particolari SMD ad alto numero di terminali (*Flat Packs*) utilizzati nei computer per usi militari.

Nell'industria giapponese la SMT fu adottata a partire dall'inizio degli anni '70.

Le applicazioni furono indirizzate nell'ambito dei prodotti "consumer" come gli orologi, le calcolatrici tascabili, gli apparecchi radio e TV.

Quindi, accanto ad una esigenza di manutenzione, si pose subito un obiettivo di riduzione costi, ottenibili con una produzione in alti volumi.

Attualmente la SMT può considerarsi una tecnologia consolidata.

Infatti essa dispone di componenti attivi e passivi appositamente progettati, di macchine di montaggio altamente sofisticate e di un'ottima possibilità di automazione.

Inoltre i fabbricanti di componenti si sono orientati verso una standardizzazione meccanica dei package ormai definitiva per quasi tutti i tipi.

Nella figura 2 è riportata un'indagine di mercato che confronta l'impiego di circuiti integrati tradizionali rispetto agli SMT.

È da rilevare che accanto ad una sostanziale stabilità del numero di componenti tradizionali fa riscontro un aumento di oltre dieci volte della quantità degli SMD.

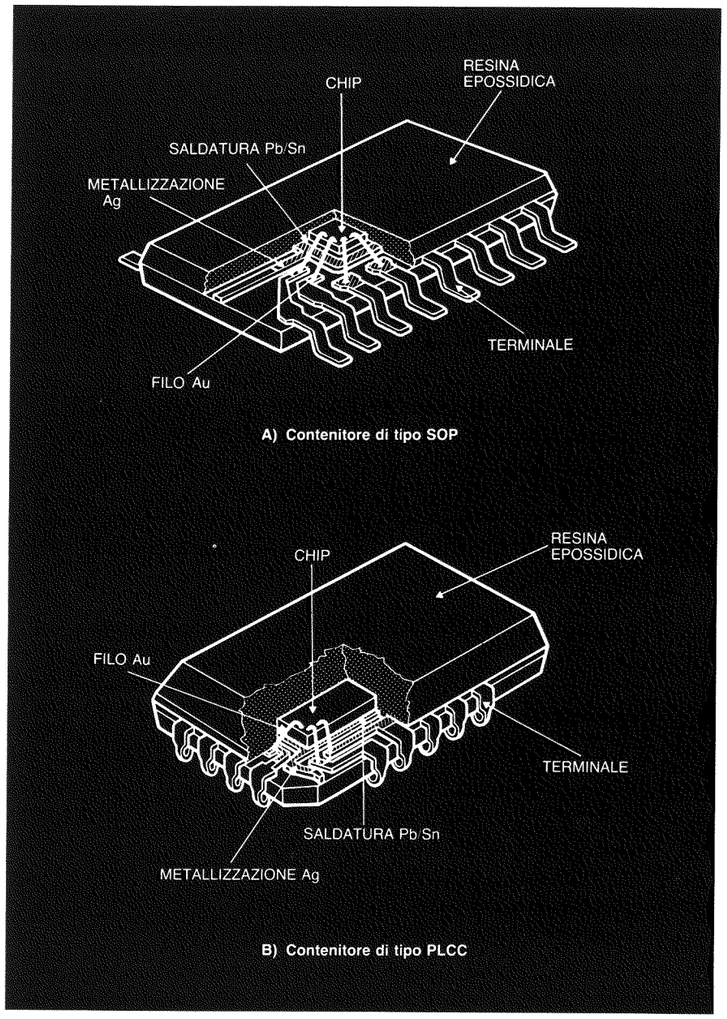


Fig. 1 - Esempi di componenti per montaggio superficiale:
a) contenitore di tipo SOP (Small Outline Package);
b) contenitore di tipo PLCC (Plastic Leaded Chip Carrier).

TIPO DI COMPONENTE	1984	1989
PTH DIP PLASTICI ALTRI PTH	 17000 2600	 16000 1500
TOTALE PTH	19600	17500
SMD SOP ALTRI SMD	 250 150	 2600 900
TOTALE SMD	400	3500
TOTALE COMPONENTI (PTH + SMD)	20000	21000

Fig. 2 - Mercato mondiale dei circuiti integrati, in milioni di pezzi (PTH = Plated Throught Hole; SMD = Surface Mount Device). Fonte AT&T.

4. Vantaggi del montaggio superficiale

Fondamentalmente i vantaggi del montaggio superficiale possono essere individuati nelle seguenti aree:

- componenti
- circuiti stampati
- montaggio
- qualità

Esaminiamo in dettaglio i singoli punti.

Componenti.

I componenti SMT hanno dimensioni ridotte rispetto a quelli tradizionali e quindi permettono di ottenere una maggiore densità di impaccamento sulla piastra.

In figura 3 è mostrato un diagramma che confronta l'area occupata sulla piastra, in funzione del numero di terminali, di componenti tradizionali (DIP) con quella di componenti SMT (SOP, PLCC, QFP).

Ne consegue anche una riduzione dello spazio di immagazzinamento necessario per il loro stoccaggio.

Hanno un minor peso e quindi permettono di realizzare apparecchiature più semplici ed affidabili, il che è estremamente importante negli impieghi mobili.

Non è necessaria alcuna preparazione dei terminali dei componenti ed inoltre vengono drasticamente ridotte le induttanze e capacità parassite, il che rappresenta un notevole vantaggio per gli impieghi in alta frequenza.

La compattezza e la miniaturizzazione del package SMT consente un notevole aumento del numero dei collegamenti di ingresso/uscita.

Attualmente lo standard JEDEC prevede componenti di tipo PQFP (*Plastic Quad Flat Pack*) con 196 terminali a passo 0,65 mm (vedere figura 4).

Circuiti stampati

Le dimensioni ridotte dei componenti SMT e la possibilità di montaggio su entrambe le facce della piastra consentono di ridurre le dimensioni del circuito stampato.

Nella figura 5 sono riportate alcune valutazioni di riduzione dell'area in funzione dei componenti usati, sia per piastre di unità centrale che di stampante.

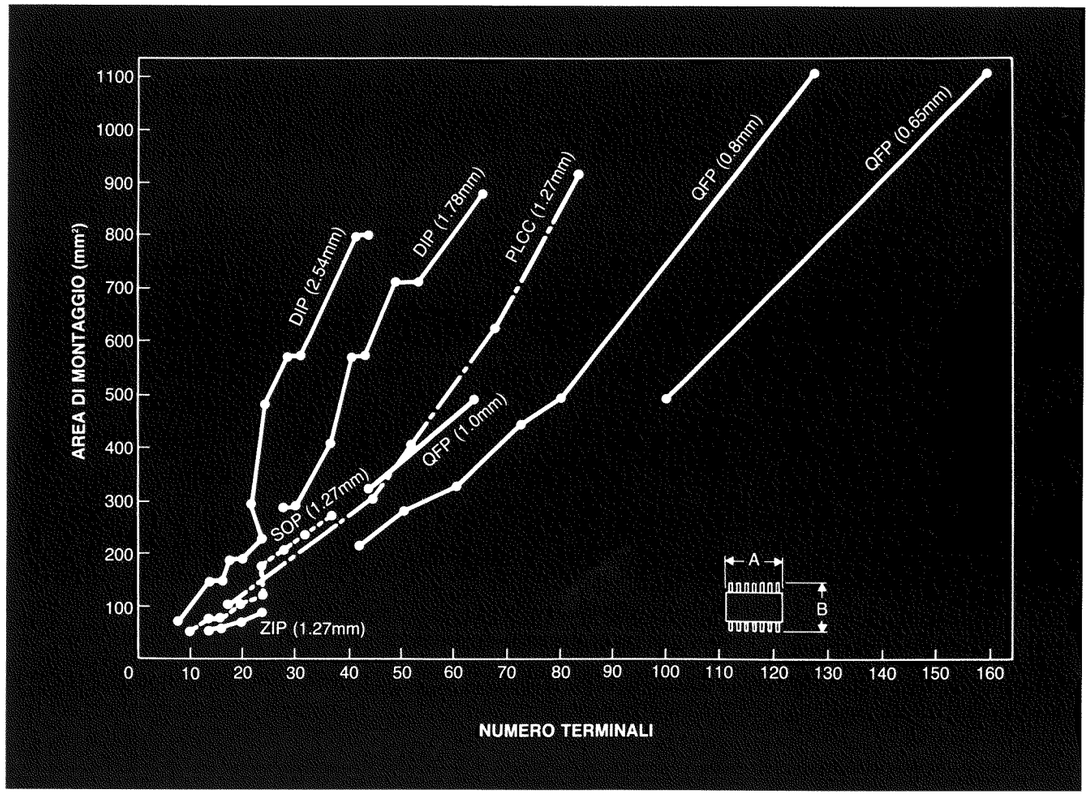
D'altro lato è possibile, mantenendo invariate le dimensioni del circuito stampato ed aumentando la densità di impaccamento dei componenti, aggiungere funzionalità elettriche/logiche.

Il minore ingombro dei componenti e le dimensioni di piastra più compatte permettono anche di ridurre la lunghezza delle piste di collegamento elettrico con indubbio vantaggio nelle applicazioni in alta frequenza.

L'esperienza fatta in Bull Italia indica che la riduzione della lunghezza media delle connessioni elettriche può essere stimata fra il 10% e il 20%.

Il tipo di materiale utilizzato continua ad essere la classica fibra di vetro con resina epossidica (FR4) che consente di avere ottime rese costruttive con costi contenuti.

Fig. 3 - Area di montaggio di componenti SMT in funzione del numero di terminali. L'area è data da AxB. I numeri tra parentesi indicano il passo dei terminali.



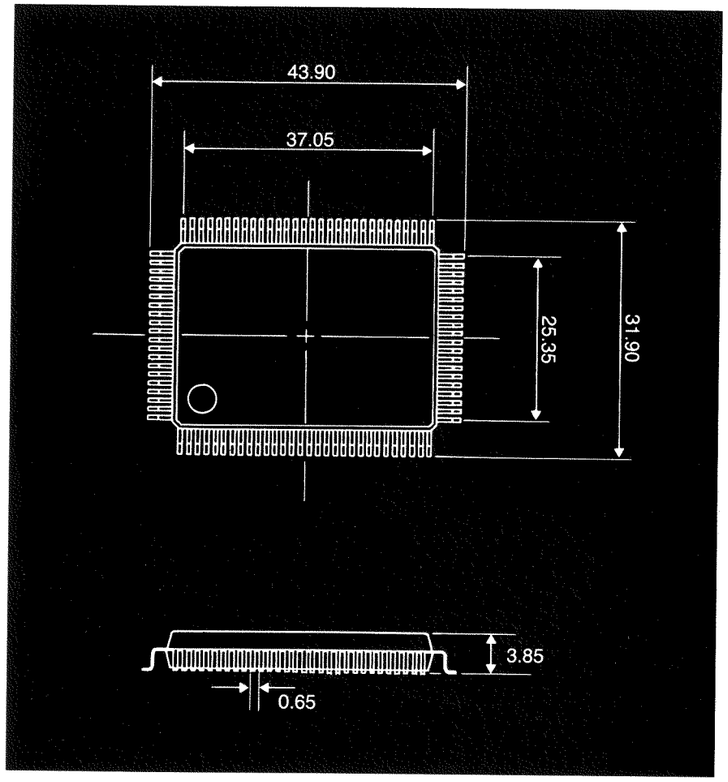


Fig. 4 - Plastic Quad Flat Pack (PQFP). Le dimensioni (in mm) si riferiscono ad un componente con 196 terminali

	PIASTRA DI UNITÀ CENTRALE			PIASTRA DI STAMPANTE		
	solo condens.	componenti discreti	discreti + integrati	solo condens.	componenti discreti	discreti + integrati
PIASTRA TIPO 1	10-15%	15-20%	–	5-10%	20-30%	–
PIASTRA TIPO 2	–	–	30-40%	–	–	25-35%
PIASTRA TIPO 3	–	–	40-50%	–	–	35-45%

Fig. 5 - Riduzione dell'area di piastra attraverso l'utilizzo di componenti SMT. Si fa riferimento a diverse combinazioni di componenti (condensatori, discreti, integrati), e per gli integrati a diversi valori del passo.

Con queste scelte possono essere utilizzati tutti i componenti SMT in contenitore plastico montati sia sul lato superiore che su quello inferiore del circuito stampato.

I componenti ceramici non vengono generalmente utilizzati a causa della differenza di coefficiente di dilatazione termica tra il componente stesso (6-7ppm/C) e il substrato FR4 (13-17ppm/C).

Montaggio

L'introduzione di questa nuova tecnica permette di ridurre il numero delle macchine automatiche di montaggio.

Generalmente con componenti tradizionali sono necessari almeno tre tipi di macchine:

- inseritrice per componenti radiali
- inseritrice per componenti assiali
- inseritrice per integrati *"dual in line"* (DIP)

La versatilità delle macchine automatiche di montaggio degli SMD consente di montare una notevole varietà di componenti con prestazioni estremamente elevate (decine di migliaia di componenti all'ora).

Inoltre i costruttori di queste macchine automatiche ne affermano la elevata affidabilità, con tassi di guasto inferiori a 100ppm.

Qualità

In generale, si può affermare che i componenti SMT non presentano tassi di guasto superiori a quelli dei componenti tradizionali.

L'accorciamento dei terminali, la riduzione di peso e la compattezza del package presentano vantaggi soprattutto in relazione agli stress meccanici (vibrazioni, urti, flessioni).

Inoltre, la riduzione delle dimensioni del circuito stampato comporta solitamente anche una riduzione delle dimensioni dell'apparecchiatura con conseguenti benefici a livello meccanico generale.

È noto che la qualità di un prodotto industriale è inversamente proporzionale al numero di interventi manuali avvenuti nella sua costruzione.

Risulta evidente che il montaggio superficiale, correlato strettamente all'utilizzo di macchine automatiche, riduce in modo drastico la manualità del processo aumentando, quindi, il livello di qualità.

Fig. 6 - Tipi di componenti integrati in funzione del numero di terminali.

TIPO DI COMPONENTE	NUMERO TERMINALI	COMMENTI
DIP plastico/ceramico	8-64	~ 80% del mercato dei circuiti integrati • passo terminali ≥ 1.78 mm
SOP	8-32	• bassa/media integrazione • passo terminali = 1.27 mm
CHIP CARRIER plastico ceramico	18-124 18-320	• per altissima velocità
QUAD FLAT PACK (QFP) plastico/ceramico	40-360	
PIN GRID ARRAY (PGA)	81-500	• per altissima velocità
CHIP ON BOARD (COB)	→ 804	• passo terminali → 0.1 mm

5. Limiti del montaggio superficiale

Allo stato attuale della tecnologia esistono alcune aree critiche che devono essere attentamente valutate per un corretto utilizzo della SMT. Esse vengono qui di seguito elencate in modo sintetico.

- L'esigenza di avere sulla piastra una densità di componenti maggiore e l'uso di circuiti integrati con parecchi terminali può richiedere l'utilizzazione di circuiti stampati con caratteristiche più evolute (diminuzione dell'insolamento tra i conduttori, aumento di strati, uso di tecniche di microforatura).

- L'elevata densità comporta anche un aumento della potenza dissipata per unità di area, dato che il contenuto di silicio non muta rispetto ai componenti tradizionali. Ciò può significare una maggiore difficoltà nell'eliminazione del calore prodotto a livello di piastra.

- Come ricordato precedentemente, la differenza di coefficiente di dilatazione termica tra substrato tradizionale e componenti ceramici sconsiglia l'impiego diffuso di questi ultimi.

- La standardizzazione delle dimensioni meccaniche dei componenti è stata fino a qualche tempo fa uno degli ostacoli maggiori per l'utilizzazione di questa tecnologia.

  È stato perciò compiuto un grosso sforzo da parte degli utilizzatori e dei produttori di componenti per ottenere una standardizzazione diffusa.

  Attualmente la quasi totalità della componentistica SMT ha dimensioni meccaniche ben definite.

  Rimangono alcuni problemi sui componenti nuovi e più sofisticati (memorie di alta capacità, integrati con elevato numero di terminazioni).

- Il controllo visivo dei punti di saldatura, considerate le dimensioni ridotte e il maggior impaccamento, se effettuato in modo convenzionale comporta notevoli difficoltà e tempi di realizzazione molto lunghi.

  Si stanno perciò sperimentando nuovi metodi, a laser o con telecamere, che finora non hanno però conseguito i risultati sperati.

  Il controllo tutt'oggi più utilizzato è ancora quello visivo dell'operatore.

- Analogamente, per le caratteristiche tecnologiche sopra ricordate, la riparazione dei circuiti in SMT risulta più lenta e generalmente più costosa di quella attuale.

6. Le ulteriori prospettive

La tecnologia di montaggio superficiale è stata una prima concreta risposta, nel campo del packaging, alle esigenze di miniaturizzazione e di integrazione che sempre più caratterizzano lo sviluppo dell'elettronica, sia in campo civile che militare.

Componenti con maggior numero di terminali sono richiesti per soddisfare le esigenze di packaging della nuova generazione di circuiti integrati (Very Large Scale Integration).

Nella figura 6 sono riportati alcuni tipi di contenitori che permettono di soddisfare le esigenze sopra menzionate.

Un primo modo per risolvere il problema è quello di ridurre ulteriormente il passo tra i terminali. Con componenti di tipo ceramico (CQFP) si può arrivare a passi di circa 0.25 mm.

Un'altra strada è quella di minimizzare il più possibile il contenitore del chip di silicio, in modo da ottenere, in linea di principio, il montaggio del chip stesso direttamente sul circuito stampato.

In pratica, il chip di silicio, dotato di terminali opportunamente sagomati, viene protetto con resina speciale per garantirne l'ermeticità e la possibilità di essere maneggiato nelle fasi di montaggio.

Questa tecnologia è denominata "*chip-on-board*"; al momento viene utilizzata per applicazioni dove le esigenze di impaccamento sono estremamente critiche (calcolatrici tascabili con alte prestazioni, apparecchiature militari, computer di fascia medio-alta).

7. Conclusioni

Da questa rapida analisi emerge che le problematiche di packaging ai diversi livelli (componente, piastra, sistema) hanno assunto un'enorme importanza nello sviluppo dei prodotti elettronici.

La tecnologia di montaggio superficiale rappresenta una risposta alle esigenze di integrazione ed impaccamento connesso.

È altresì evidente che questa nuova tecnologia comporta notevoli modifiche sia nel progetto che nei processi produttivi. Si rendono quindi necessari investimenti per nuove macchine di produzione e, nel contempo, una preparazione specifica del personale tecnico che deve affrontare queste nuove problematiche.

Macchine intelligenti tra utopia e realtà

FRANCO FILIPPAZZI,
GIULIO OCCHINI

Bull Italia
Centro Studi Informatica e Automazione
Milano

"Temo più la stupidità dell'uomo che l'intelligenza della macchina"

John Mc Carthy

1. Una aspirazione faustiana

Un antico desiderio dell'uomo è costruire macchine capaci di emulare alcune delle sue precipue caratteristiche, in particolare quelle attinenti la sfera delle attività mentali. Alla radice di questo desiderio si può scorgere l'ambizione faustiana di dar vita e intelligenza alla materia, in concorrenza con la divinità. Rientrano in questo quadro il mito del Golem e gli automi realizzati nel "secolo dei lumi" con complicati meccanismi di orologeria, che, nei casi più ambiziosi, come il famoso giocatore di scacchi di Von Kempelen, si rivelano poi delle mistificazioni.

Per dare corpo a questo antico sogno, l'umanità ha dovuto attendere la comparsa del computer. Il computer è infatti una macchina che si differenzia da tutte le altre inventate dall'uomo perchè opera su entità immateriali, quali sono l'informazione e la conoscenza.

Il termine "Intelligenza Artificiale" fu coniato negli anni '50, nell'atmosfera di grande entusiasmo creata dall'avvento del computer. In effetti, alla euforia iniziale subentrò via via la consapevolezza della grande difficoltà di emulare la mente dell'uomo. Per molti anni l'Intelligenza Artificiale rimase oggetto di ricerca in ambito accademico, fornendo contributi importanti all'informatica teorica, ma senza approdare a significativi risultati concreti.

Un clamoroso rilancio della Intelligenza Artificiale si ebbe nel 1981, quando i giapponesi annunciarono il piano per arrivare negli anni '90 alla cosiddetta quinta generazione di elaboratori, caratterizzati appunto dal fatto di incorporare "intelligenza". Questa nuova generazione di macchine dovrebbe costituire una ulteriore tappa nella evoluzione della "species informatica", la cui comparsa risale a neanche mezzo secolo fa.

Alla base di questo rapido succedersi di generazioni di computer c'è lo straordinario progresso della tecnologia elettronica che, nell'arco di tempo menzionato, è passata dagli ingombranti e inefficienti tubi termoionici agli attuali microcircuiti, che, in pochi millimetri quadrati di silicio, realizzano ormai l'equivalente di milioni dei vecchi tubi.

In quanto segue si cercherà di sintetizzare l'evoluzione del computer verso forme più intelligenti, considerando dapprima l'aspetto strutturale e poi quello più generalmente paradigmatico.

Il materiale qui presentato rispecchia una conversazione degli Autori al Rotary Club Milano Sud, nell'ambito di un ciclo di dibattiti sulla tecnologia, promosso da Umberto Pellegrini.

2. L'evoluzione strutturale della "species informatica"

Il modello di struttura concepito originariamente negli anni '40 dal matematico Von Neuman, e tuttora largamente in uso, prevede che l'elaborazione avvenga a piccoli passi concatenati, l'uno successivo all'altro.

A questo modello "seriale" si contrappone la concezione "parallela", in cui cioè l'elaborazione avviene, per quanto possibile, con operazioni sovrapposte nel tempo. Chiaramente le strutture parallele aumentano in misura anche molto grande la velocità del computer. Ed è in questa direzione che si orientano i nuovi modelli computazionali. Occorre però dire che le maggiori prestazioni delle strutture parallele non si ottengono in modo semplice né gratuito. Per rendercene conto si può ricorrere ad un paragone con la costruzione di un edificio.

Nell'approccio seriale si impiega un solo operaio che, in sequenza, esegue tutte le operazioni richieste: erigere i muri, posare le condutture idrauliche, stendere l'impianto elettrico, mettere le piastrelle e gli infissi, imbiancare le pareti e via di seguito.

Nell'approccio parallelo, invece, più operai lavorano contemporaneamente a piani diversi dell'edificio con compiti differenti. È evidente che in questo caso non solo occorrono più uomini, ma che la gestione dell'attività è molto più complessa. Occorre infatti coordinare e sincronizzare il lavoro di tante persone, facendo tra l'altro in modo che non si intralcino nell'uso di risorse comuni, quali possono essere il magazzi-

no dei materiali, i mezzi di lavoro o anche i percorsi nel cantiere.

Fuori di metafora, un computer con architettura parallela richiede più risorse "hardware" (una pluralità di unità di elaborazione, anziché una sola) e implica un "software" di gestione del sistema di gran lunga più complesso che nel caso seriale.

Sono queste, in sostanza, le ragioni per cui finora hanno prevalso le strutture seriali. Ora le cose stanno cambiando e si registra un crescendo di attività nelle soluzioni parallele. Ciò è dovuto a due ordini di motivi: da un lato, la richiesta di maggiori velocità non ottenibili con le strutture tradizionali, dall'altro, il fatto che la microelettronica consente oggi soluzioni fino a ieri fisicamente ed economicamente irrealizzabili.

L'elaborazione parallela costituisce ormai un grande capitolo della "computer science" e abbraccia una quantità di soluzioni diverse su cui non possiamo qui addentrarci. Ci limiteremo a darne una classificazione generale basata sulla "granularità" del sistema (fig. 1).

Un'architettura parallela si dice "a grana grossa" se costituita da poche unità di elaborazione di elevata potenza, "a grana fine" se invece ve ne sono molte di modesta portata. La scelta della granularità dipende dalla natura del problema e dalla sua modellizzazione. Un esempio può aiutare a capire il concetto.

Si consideri l'elaborazione delle previsioni meteorologiche. Nell'approccio "a grana grossa", l'atmosfera viene suddivisa in un numero limitato di volumi di grande dimensione, migliaia di chilometri cubi, ad ognuno dei quali viene associata una unità di elaborazione. È ragionevole ritenere che il tempo meteorologico in ciascuna zona sia determinato prevalentemente dalle variabili interne alla zona stessa. L'elaborazione implica quindi un grosso carico di lavoro per le singole unità di elaborazione, mentre la loro interazione sarà modesta, essendo limitata alle interfacce tra volumi adiacenti.

Nell'approccio "a grana fine", l'atmosfera è invece suddivisa in un gran numero di piccoli volumi, ciascuno di pochi chilometri cubi, ad ognuno dei quali viene ancora associata una unità di elaborazione. È evidente che, in questo caso, si ha una situazione opposta alla precedente: ogni unità di elaborazione può essere molto piccola perché ci sono limitate necessità di calcolo, però si richiede un intenso scambio di informazioni tra le varie unità durante tutta l'elaborazione.

In sostanza, nei sistemi "a grana grossa" il fattore critico risiede nelle unità di elaborazione, mentre in quelli "a grana fine" è nelle comunicazioni tra di esse.

Il concetto di elaborazione parallela si può ulteriormente spingere verso quelli che possiamo definire sistemi "a grana finissima", costituiti cioè da un grandissimo numero di elementi, ciascuno dei quali estremamente semplice. Sistemi di questo tipo appartengono ad un'area avanzata e ancora poco sviluppata della ricerca e sono meglio noti come elaboratori "neuronali". Come indica il nome, essi si rifanno alla struttura del cervello, costituito per l'appunto da una miriade di cellule nervose, i neuroni, tra loro fittamente interconnesse.

3. I paradigmi di elaborazione

Parallelamente alla evoluzione della struttura del computer, si vanno modificando i paradigmi di elaborazione, ossia il modus operandi della macchina. Sotto questo profilo, si possono individuare tre diversi paradigmi: quello tradizionale, quello della Intelligenza Artificiale "classica" e quello "connessionista" (fig. 2).

Nel primo caso la macchina è un puro esecutore di soluzioni (algoritmi), definite in ogni dettaglio dall'uomo. È questo il modello originario dell'elaborazione automatica, tuttavia valido e di gran lunga il più usato.

Il secondo paradigma contempla macchine in grado di "ragionare", ossia di trarre delle conclusioni da premesse assegnate. In altri termini, la macchina è capace di trovare da sè l'algoritmo risolutore del problema. Una macchina di questo tipo si rifà ad un modello del funzionamento della mente umana, che implicitamente già si trova nella logica aristotelica. In tale modello si individuano due componenti fondamentali del ragionamento: da un lato, la "conoscenza" del dominio cui il problema si riferisce, dall'altro, i meccanismi di "inferenza" logica. Questi ultimi sono, per loro natura, di tipo universale, ossia indipendenti dal problema. Secondo la definizione del matematico Robinson, "Logic deals with what follows from what".

La differenza sostanziale tra i due paradigmi esaminati è che, mentre nel primo la soluzione del problema (l'algoritmo) viene fornita dall'uomo, nel secondo caso l'uomo si limita a descrivere il problema e la macchina trova autonomamente una possibile soluzione.

Va notato che ambedue i paradigmi sono accomunati dal fatto di operare su una "rappresentazione del mondo" di tipo simbolico. In sostanza, in ambedue i casi, oggetto del processo elaborativo sono sequenze di segni alfanumerici.

In effetti, i problemi che l'uomo è chiamato a risolvere non sono rappresentati soltanto mediante segni convenzionali, come le lettere dell'alfabeto o i numeri, ma anche, in molti casi, da configurazioni ("pattern") di elementi, percepite attraverso i sensi: immagini, odori, sensazioni tattili.

In tutti questi casi, l'approccio analitico dei paradigmi sopra citati risulta largamente inefficiente.

Si pensi, ad esempio, alla difficoltà di descrivere un volto, servendosi solo di espressioni verbali, in modo sufficientemente dettagliato da poterlo riconoscere tra milioni di altri. Eppure, questa è una operazione che il cervello dell'uomo compie, senza apparente difficoltà, in una frazione di secondo.

È da queste considerazioni che nasce l'idea di un terzo paradigma elaborativo, che ha come riferimento, non una schematizzazione dei processi mentali di ragionamento, ma la struttura fisica del cervello.

Come noto, quest'ultimo è costituito da particolari cellule biologiche, i neuroni. Ognuno di questi è un'unità elementare di elaborazione avente numerosissime ramificazioni di ingresso (i dendriti) e un tronco di uscita (l'assone). Ogni ingresso ha un "peso" (valore della sinapsi). Allorché la somma pesata dei segnali di ingresso supera un determinato valore di soglia, il neurone scatta ed emette un impulso in uscita.
In sostanza, nelle reti cerebrali i neuroni svolgono la funzione di elaborazione e le sinapsi quella di memorizzazione. La memoria non costituisce quindi un blocco a sé stante ma è diffusa nelle connessioni.

Una caratteristica fondamentale delle reti neuroniche è che il valore delle sinapsi, ossia il peso delle connessioni, non è fisso, ma può modificarsi durante il funzionamento del cervello. In altri termini, una connessione può rafforzarsi, indebolirsi o addirittura sparire a seconda di come è esercitata.

La proprietà descritta ha enorme importanza, poiché è alla base della nostra capacità di apprendere, adattarci, evolvere.

Un'altra caratteristica fondamentale del cervello è il suo altissimo parallelismo operativo. Ciò è dovuto al numero dei neuroni, che è dell'ordine di cento miliardi, ma soprattutto all'enorme sviluppo delle loro interconnessioni.
Basti pensare che l'uscita di ogni neurone si collega mediamente a diecimila altri neuroni. Per cui l'attivazione di un singolo neurone può coinvolgere in cascata, in una frazione di secondo, intere zone del cervello e l'elaborazione sembra avvenire dappertutto. Va notato che i neuroni sono in sè molto lenti, almeno se paragonati ai dispositivi impiegati nei computer. Infatti, il neurone ha un tempo medio di attivazione dell'ordine di un millise-

Fig. 1 - Tipologia dei sistemi di elaborazione parallela.

ELABORAZIONE PARALLELA

A GRANA GROSSA	• POCHI PROCESSORI MOLTO COMPLESSI • MEMORIA SEPARATA
A GRANA FINE	• MOLTI PROCESSORI POCO COMPLESSI • MOLTE MEMORIE LOCALI
A GRANA FINISSIMA (reti neuronali)	• MOLTISSIMI PROCESSORI MOLTO SEMPLICI • MEMORIA DISTRIBUITA NELLE CONNESSIONI

Fig. 2 - I paradigmi di elaborazione: tradizionale, della Intelligenza Artificiale "classica" e connessionista (reti neuronali). I tre paradigmi vanno visti non in alternativa, ma tra loro complementari.

PARADIGMI ELABORATIVI			
	tradizionale **MACCHINE CHE ESEGUONO**	I.A. "classica" **MACCHINE CHE RAGIONANO**	connessionista **MACCHINE CHE SI AUTOPROGRAMMANO**
L'UOMO	fornisce la soluzione dei problemi	fornisce la descrizione dei problemi	insegna a risolvere i problemi
LA MACCHINA	opera su simboli, mediante processi algoritmici	opera su simboli, mediante processi logico-deduttivi	opera su pattern, mediante processi associativi
IL MODELLO	non ha particolari riferimenti umani	è la mente (modello funzionale dell'intelligenza)	è il cervello (modello fisico dell'intelligenza)

condo, ossia è circa un milione di volte più lento dei nostri circuiti elettronici. È il grande parallelismo della struttura che controbilancia la relativa lentezza dei neuroni e fa del cervello quella straordinaria macchina che conosciamo.

Gli elementi che, per quanto ne sappiamo, stanno alla base del funzionamento del cervello costituiscono il modello cui si ispirano le ricerche sul computer neuronale. Usando l'equivalente elettronico del neurone, si possono costruire reti neuroniche artificiali che presentano similitudini funzionali con quelle biologiche.

In particolare, queste reti artificiali presentano la capacità di automodificare le proprie connessioni attraverso l'interazione col mondo esterno; in altre parole, sono in grado di apprendere dall'esperienza. In questo ambito sono stati di recente realizzati esperimenti che, se pur ancora poco significativi in termini pratici, sono rilevanti dal punto di vista concettuale. Ad esempio una rete neuronale che impara a pronunciare delle parole scritte, riuscendo a passare da un borbottio inintelligibile ad una dizione accettabile, confrontandosi via via con la voce umana. In un altro esempio, una rete neuronale impara a guidare un veicolo sulla pista di un videogioco, seguendo il comportamento di un istruttore che guida manualmente il veicolo mediante un joystick.

L'apprendimento avviene, in sostanza, utilizzando l'errore - ossia la differenza tra l'output fornito dalla rete e l'output corretto - per modificare i pesi delle connessioni. Attraverso un processo iterativo, l'errore via via si riduce tendendo ad annullarsi.

Il modello "neuronale" (o "connessionista"), pur essendo stato proposto già all'inizio dell'era del computer, solo di recente è stato ripreso concretamente in considerazione. Siamo attualmente in una fase ancora pionieristica con prospettive di sviluppo difficilmente anticipabili.

Va comunque precisato che il modello connessionista non si pone in alternativa a quello della Intelligenza Artificiale "classica", ma è ad esso complementare. L'emulazione dei processi intelligenti dovrebbe comportare l'uso integrato dei due paradigmi, in analogia con quanto sembra avvenire tra i due emisferi del cervello umano.

Da molte indagini neurologiche risulta, infatti, che i due emisferi cerebrali hanno funzionalità diverse: quello di sinistra appare prevalentemente coinvolto nei processi di tipo razionale e analitico, quello di destra, invece, nei processi di tipo intuitivo e sintetico. In altri termini, l'emisfero di sinistra starebbe a quello di destra come il computer tradizionale a quello neuronale.

Può essere interessante, infine, ricordare che, oltre agli studi sulle reti neuroniche artificiali, sono in atto ricerche dirette a realizzare dispositivi su scala molecolare, basati su materiali organici. In entrambi i casi, si fa riferimento a modelli propri del mondo biologico.

4. Potenziali impatti

L'Intelligenza Artificiale è destinata, per sua natura, a suscitare reazioni contrastanti, a seconda che se ne enfatizzino i vantaggi o i rischi potenziali. Non si vuole qui affrontare il tema in tutta la sua portata, ma solo presentare qualche riflessione su alcuni dei potenziali impatti delle macchine intelligenti.

Macchine per decidere

Si può prevedere che le tecniche e gli strumenti della Intelligenza Artificiale avranno un'estesa applicazione nell'area del supporto alla decisione (IDSS: Intelligent Decision Support Systems). Sistemi di questo tipo trovano impiego potenziale in tutti gli ambiti in cui l'uomo è chiamato a prendere decisioni: dal settore aziendale a quello sanitario, dallo studio professionale fino alle attività di governo. È superfluo sottolineare quali implicazioni sociali possono avere gli IDSS. Si può, comunque, affermare che, da un lato, la loro diffusione costituisce un processo ineluttabile a causa della crescente complessità dei sistemi con cui l'umanità ha a che fare, e, dall'altro, che particolare attenzione va posta ai rischi connessi con la delega di compiti alle macchine.

Per fare uno tra i vari esempi possibili, si può ricordare il cosiddetto "ottobre nero" di Wall Street, dove gli automatismi di compra-vendita computerizzata dei titoli si ritiene abbiano avuto un ruolo non trascurabile. Il rischio, con le macchine "intelligenti", è che la delega di compiti si traduca, in effetti, in un trasferimento di responsabilità dall'uomo alla macchina.

Informatica e occupazione

Un'altra preoccupazione nei riguardi del diffondersi dell'informatica in generale e delle "macchine intelligenti" in particolare, è relativa alla disoccupazione di cui esse sarebbero la causa.

È indubbio che la innovazione informatica impatti sul mercato del lavoro, ma questo rapporto è assai meno diretto di quanto troppo spesso non si sostenga anche da parte di autorevoli commentatori.

Il punto chiave è la produttività in senso lato. In quanto la macchina riesce a migliorare la produttività, essa viene impiegata e in quanto la produttività riguarda prima di tutto le risorse umane, essa può agire negativamente sulla occupazione.

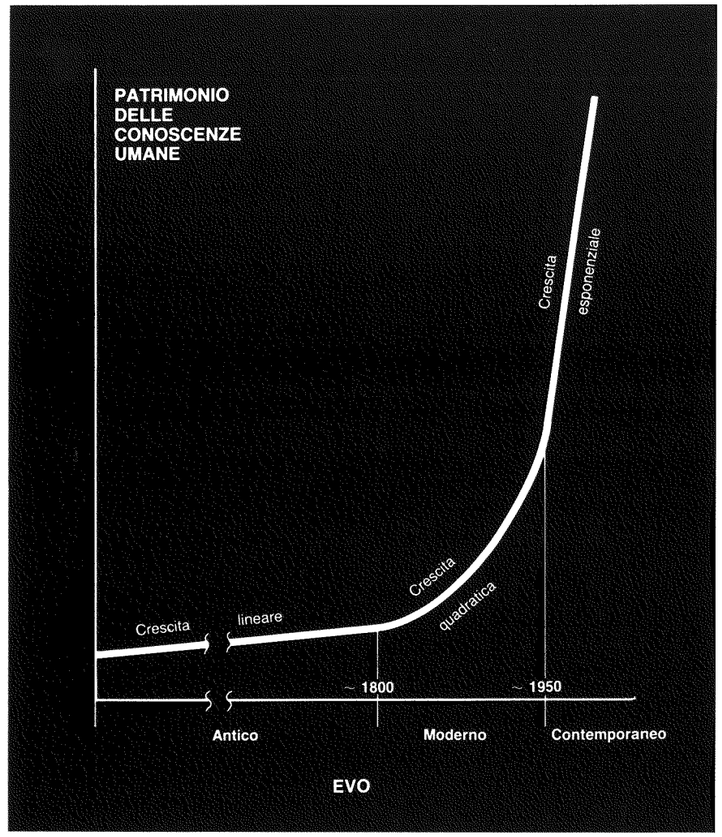
La produttività è però anche fattore condizionante lo sviluppo della domanda di mercato.

Di conseguenza, applicare l'innovazione informatica significa anche aumentare la produzione di beni e servizi e quindi agire positivamente sulla occupazione.

Bisogna cioè innescare quel "circuito virtuoso" dell'economia per cui la produttività cresce in misura minore o uguale della produzione. In questo caso l'innovazione tecnologica non solo non crea disoccupazione ma è anzi la via maestra per riassorbirla.

Che sia possibile mantenere un tale circuito virtuoso è ampiamente dimostrato dal Giappone, pur con un sistema sociale differente dal nostro, ma anche dagli Stati Uniti, che, negli ultimi 8 anni hanno saputo

Fig. 3 - Il patrimonio delle conoscenze umane si accresce ormai a ritmo esponenziale. Secondo alcune valutazioni, attualmente esso raddoppia circa ogni 8 anni. Un ruolo fondamentale delle macchine intelligenti è di gestire questo fenomeno.



creare 15 milioni di nuovi posti di lavoro.

In realtà, le macchine informatiche non creano disoccupazione ma provocano piuttosto una redistribuzione delle risorse sia con riferimento alle nuove leve di lavoratori che a quelle già attive.

Per quanto riguarda le nuove leve di lavoratori, il rimedio sta in una migliore sintonizzazione tra mondo della scuola e mondo del lavoro per ridurre al minimo la discrasia oggi esistente in alcune aree.

Per quanto riguarda invece gli occupati che devono cambiare la loro attività sotto la spinta della innovazione informatica, gli uomini, a differenza delle macchine, sono in grado di imparare.

Si tratta di far leva su questa loro capacità per metterli in condizioni di affrontare nuovi mestieri non emulabili dalle macchine; si tratta, cioè, di investire nella formazione e, in particolare, in quella di tipo permanente che dura, cioè, tutta la vita e di cui tanto si parla nell'ambito della cosidetta società dell'informazione.

Ed è proprio nella formazione, cioè nei processi di trasferimento della conoscenza, che le macchine intelligenti sono destinate a fornire un contributo, che se è ancora scarsamente percepito, è destinato tuttavia ad essere di portata epocale.

Il trasferimento del sapere

Nella sua storia evolutiva, l'umanità si è avvalsa di modalità di trasmissione del sapere sempre più articolate.

L'insegnamento si è all'inizio fondato sull'esempio e il concetto di "bottega artistica" del rinascimento ben rispecchia questo modello.

Oggi ancora il laboratorio artigiano vi fa riferimento.

Si svilupparono successivamente le modalità di trasmissione orale attraverso la lezione e il dialogo e, più di recente, quelle basate sul testo scritto.

Oggi, insieme con la comunicazione verbale e il libro, disponiamo degli audiovisivi che ci danno l'immagine statica o in movimento.

Va sottolineato che, con l'eccezione del dialogo, gli altri veicoli di trasmissione del sapere sono di tipo passivo nel senso che il discente non ha modo di interagire con essi per cambiarli e per ottenere risposte a domande non previste.

Per questa loro limitazione, la scrittura e la stampa (e oggi la televisione) sono state oggetto di contestazione.

Si ricorda, a riguardo, la polemica contro la scrittura che Platone mette in bocca a Socrate, nel dialogo intitolato a Fedro.

Socrate giudica la scrittura disumana perchè tenta di ricreare al di fuori della mente il simulacro di ciò che

in realtà esiste solo al suo interno.

Una vera realtà del pensiero, sostiene Socrate (o meglio, glielo fa sostenere Platone, che ne scrive senza evidentemente avvertire il paradosso) è dentro l'uomo e non potrà mai essere resa nella sua interezza e nella sua completezza da un manufatto come la scrittura.

Usarla significa impoverire la realtà umana, dandone una rappresentazione artefatta e riduttiva.

Inoltre la scrittura sarebbe destinata a distruggere la memoria poichè l'uomo non sarà più obbligato a ricordare in quanto potrà contare su risorse esterne a scapito dello sviluppo di quelle interne.

E la memoria, secondo Socrate, è alla base dello sviluppo delle capacità mentali.

Un timore sostanzialmente analogo si è ripresentato, due millenni dopo, nei confronti della stampa, cioè della riproduzione meccanica della scrittura.

Per tornare al tema, con le macchine intelligenti il processo di trasferimento del sapere sta imboccando oggi una nuova e originale direzione.

Una macchina di questo genere, infatti, è in grado di acquisire, accumulare e rendere fruibile, in modo interattivo, il sapere sia formale che euristico, proveniente dall'esperienza che via via gli esperti accumulano in un determinato settore.

I "sistemi esperti" che già oggi si diffondono nelle aree più disparate della medicina, della ingegneria e della finanza, sono macchine in grado di diventare, pur di approntare l'opportuna rete di acquisizione, il più potente strumento di accumulo e di distribuzione delle conoscenze umane su un determinato argomento.

Il sapere di un sistema esperto è aggiornabile e fruibile in modo interattivo e il suo sviluppo può avvenire seguendo in tempo reale quello dell'umanità, che si accresce ormai con un ritmo esponenziale (fig. 3).

In altre parole, la macchina può assorbire conoscenza dai vari esperti esistenti a livello mondiale, man mano che essa si crea, per restituirla poi elaborata ai suoi utenti in un processo di autoalimentazione che non presenta alcun limite superiore di natura concettuale.

Questa "crescita all'infinito" è certamente un fenomeno senza precedenti nella storia dell'uomo e può costituire un formidabile acceleratore del suo sviluppo intellettuale.

Quanto meno è destinata a rimettere in discussione metodi e istituzioni didattiche in vigore da secoli che oggi, nell'ambito della formazione permanente, risultano impari al compito.

5. Verso una discontinuità evolutiva?

Tutto ciò che è stato detto in precedenza rientra nell'ambito di sviluppi già oggi prevedibili. A più lungo termine, ci si può porre il problema filosofico della intelligenza delle macchine nel futuro della specie umana.

A questo riguardo si può affermare che, mentre le capacità del cervello umano si possono considerare limitate, nel senso che la struttura di tale organo evolve su archi di tempo lunghissimi, la capacità delle macchine è, in confronto, virtualmente senza limiti di crescita.

La simbiosi uomo-macchina, che si va instaurando, introduce quindi una discontinuità nella evoluzione della specie umana, che non ha riscontro nella storia della vita sulla Terra (fig. 4).

La domanda che a questo punto ci si può porre non è se ciò avverrà o meno, perchè si tratta di un processo ineluttabile, ma piuttosto quale sia l'atteggiamento più opportuno per affrontare questa mutazione.

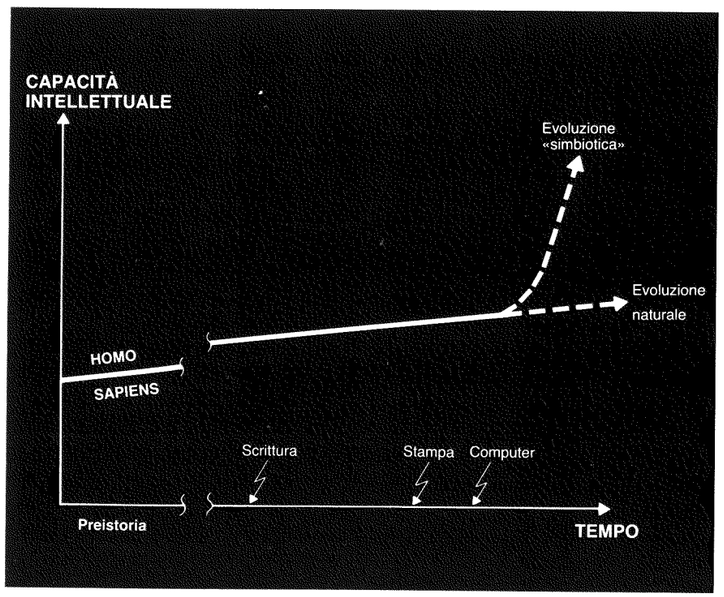
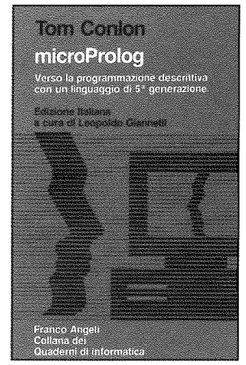
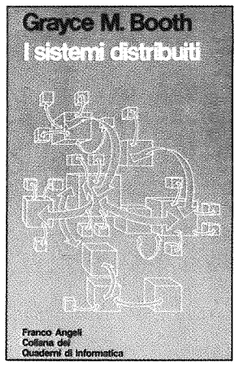
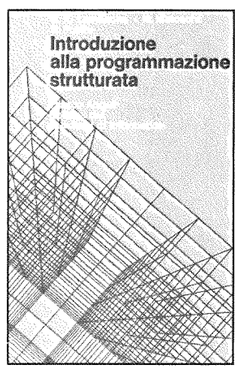


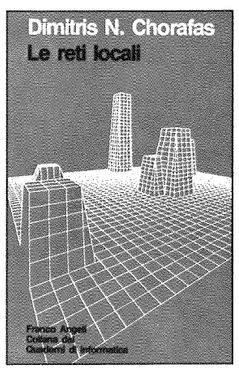
Fig. 4 - La capacità del cervello umano evolve con ritmi biologici, ossia su archi di tempo lunghissimi. Il contrario si può dire, invece, dei sistemi artificiali di elaborazione. L'amplificazione della mente rappresentata dal computer introduce, quindi, una discontinuità nella evoluzione della specie umana, che non ha riscontro nella storia della vita sulla Terra.

I TESTI DEI QUADERNI DI INFORMATICA









I TESTI DEI
QUADERNI DI INFORMATICA