KIS JÁNOS — SZEGEDI IMRE

Új víruslélektan

ALAPLAP KÖNYVEK 3

Kis János – Szegedi Imre

ÚJ VÍRUSLÉLEKTAN

Kis János – Szegedi Imre

ÚJ VÍRUSLÉLEKTAN

CÉDRUS KIADÓ

SOSZ : 76
MC 125.292
1991

ISSN 0866–434X 7429 00x
ISBN 963 7429 026

© Kis János és Szegedi Imre, 1991
Szerkesztette: Faklen Pál és Lukács Erzsébet
Sorozatszerkesztés és borítóterv: Faklen Pál

Felelős kiadó: Sebestyén Ilona
Felelős vezető: Vékony Tamás
Cédrus Informatikai Részvénytársaság

Nyomdai előkészítés: **PRINTSELF** Kft.
Felelős vezető: Dr. Kassay Árpád

Nyomtatás: Sprint^{R} – Hungaroprint
Felelős vezető: Tóth Éva
Készült a Stoffel & Tsai. gondozásában

ELÖLJÁRÓBAN
avagy
NIL NOCERE

Ki gondolta volna egy–másfél évvel ezelőtt, hogy a magyar számítástechnikai szakirodalom egyik legnépszerűbb kiadványa a számítógépes vírusok „lélektanával" foglalkozó könyv lesz?! Hamarosan jelentkezett az igény a második kiadásra, de közben annyira felduzzadt a téma, hogy a szerzők közül ketten elhatároztuk: inkább átdolgozzuk a művet. Terjedelmi okokból az anyagot két kötetre kellett bontanunk. Az *Új víruslélektan* című könyvbe a vírusok általános jellemzése, történeti és etikai bemutatása és néhány programozástechnikai rész került, míg a vírusok részletes leírásával, felismerésével és irtásával a *Vírushatározó* című könyvben foglalkozunk.

Hallottunk olyan véleményeket, miszerint sokan azért vették meg a *Víruslélektan* könyvet, hogy megtanuljanak belőle vírust írni. Akadtak olyanok is, akik megkísérelték megszerezni Szegedi Imre doktori disszertációjának víruskódokat is tartalmazó, zárolt példányát. A nemzetközi tapasztalat azonban idejében figyelmeztetett bennünket, hogy széles körben hozzáférhető anyagban víruskódokat közölni nem szabad, mert etikai érzék hiányában sokan átbarkácsolják a hozzáférhető víruskódokat, újabb, a korábbiaknál is ártalmasabb vírusokat hozva létre.

Az USA-ban meggondolatlanul meghirdetett, vírusfejlesztésre kiírt Pentagon-pályázat szintén felkeltette a terrorista hajlamú szakemberek és az etikátlan programozók figyelmét. Szabotázs és hadviselés céljaira megindult a számítógépes vírusok rohamléptékű fejlesztése. Ha hinni lehet a szakmai pletykának: volt olyan programozói csoport, amelyik vírusterméket a vírusok nemzetközi feketekereskedelmének csatornáin útjára bocsátva csinos bevételre tett szert.

A korábbi jóslat is beigazolódott: a vírusírás súlypontja áttevődött a Szovjetunióba, Bulgáriába és Romániába. Hamburg környékén is dolgozik egy csoport, amely a Fish és a Whale vírusok formájában bocsátotta az informatikai társadalomra legújabb generációs, egymással intelligens kapcsolatban lévő, evolúciós és mutációs képességeket is felmutató első vírusgenerációit.

Egy év igen nagy idő. Ezalatt változott a vírusok világa, de új, a ko-

rábbinál hatékonyabb vagy éppen sokkal rosszabb védekezési eljárások is megjelentek. Abszolút védelmet eddig csak egyetlen védőeszköz kínált, amellyel Hannoverben, az 1991-es Cebit kiállításon a Memorex mágneslemezeket gyártó cég standján találkoztunk. Az általuk javasolt betét a mágneslemezes egységekben kulccsal rögzíthető, s ekkor a rendszer floppyval nem használható. Igaz, ez a megoldás az adatátviteli vonalakon érkező vírusok ellen mit sem használ.

Az emberi orvoslás klasszikusa, Hippokratész etikai elve »nil nocere« (ne árts) volt. Azaz ha a betegen segíteni nem tudunk, legalább ne ártsunk neki. Nos, mi is ezt valljuk. Különösen most, amikor a forgalmazók egy része éppúgy híján van az elemi etikai érzéknek, mint a programozók vagy programbarkácsolók elég jelentős hányada. A vírustechnológia alapjában a „piszkos trükkök" közé tartozik, éppúgy, mint a programok élettartamát, másolhatóságát korlátozó eljárások. És az emberek nem „Grál-lovagok", mindig vannak köztük ártó szándékúak, akiktől meg kell(ene) védeni a társadalmat. A tudás hatalom. De visszaélni vele annál kevésbé lehet, minél szélesebb körben ismerik a korábbi titkokat. Az emberek józanságában és etikai érzékében bízva invitáljuk az olvasót a számítástechnika eme rejtett boszorkánykonyhájába.

A korábbi változathoz képest most jóval több alapinformációt kívánunk adni. Segíteni szeretnénk a felhasználókat abban, hogy idejében felismerjék a vírus jelenlétét, s ha tudják, előzzék meg a fertőzést. Ha pedig a baj már bekövetkezett, mérsékeljék a vírusok okozta kárt.

Napjainkra a vírusírás a vele rokon másolásvédelem-írással ugyanúgy üzletté vált, mint a vírusirtás. S mint minden ilyen tevékenységnél, a tisztességes szakemberek mellett jelen vannak a piacon a kóklerek és a gengszterek is. A nem eléggé tájékozott felhasználó kétségbeesésében–zavarában megveszi az útjába eső első – hangzatos névvel kínált – terméket, később pedig csodálkozik, ha elvesznek adatai, programjai. Mások a másolásvédett szoftver remekléseit tapasztalhatják hosszabb-rövidebb idő elteltével.

Sajnos, sok nyugat-európai forgalmazó valamiféle „vadkeletnek" tekintette Magyarországot. A hazai forgalmazók rémhíreinek és félretájékoztatásának eredményeképpen maguk is követték az itt látottakat. Hosszabb távon azonban hazánk sem szigetelheti el magát a számítástechnika világában és az üzleti életben külföldön végbemenő folyamatoktól. A tapasztalat egyre inkább igazolja, hogy a tisztesség, a bizalom – és a megfelelő jogi garancia – üzletileg mind a forgalmazóknak, mind a felhasználóknak kifizetődik.

Könyvünkben megpróbálunk a hazai törvényes szabályozás adta lehetőségeken belül segíteni, az eligazodáshoz minél több információt ad-

ni. A tájékoztatás annál is inkább érdekünk, mivel tudjuk, a legnagyobb érték nem is annyira a program vagy a számítógép, sokkal inkább a benne tárolt adat, adatbázis. Azokat pedig, ha megsérülnek – ismervén a hazai spórolási gyakorlatot –, a rendszeres mentés hiányában sokszor még pótolni sem lehet.

Köszönetünket nyilvánítjuk a BBS-ek fenntartóinak szerte a világon, hogy összeállíthattuk ezt az anyagot. Különösen sokat használtunk fel dr. Solomon's, valamint Patricia M. Hoffman dokumentációjából, amely John McAfee VIRNET rendszerében szerepelt. Az adatbiztonsággal és a számítógépes terrorizmus elhárításával foglalkozó csoportok rendszeres publikációikkal és adatszolgáltatásukkal szintén hatásosan segítik a nemzetközi fellépést az adatgyilkosok ellen. Közülük a legfrekventáltabb helyen talán Yasrael Radai csoportja tevékenykedik a Héber Egyetemen, Izraelben. Hollandiában Jan Tepstra, Németországban pedig a Hamburgi Egyetem Vírustesztelő Központja publikál rendszeresen adatokat a programvírusokról.

Szeretnénk ezúton is, ismételten köszönetet mondani a névtelen közreműködőknek, akik segítettek megismerni a piszkos trükkök hazai és külhoni fejlesztő laboratóriumaiban a bevétel tisztességtelen növelésére kifőzött programokat. Ők lelkiismeretük parancsának engedelmeskedve sok esetben még forráskódok kiszolgáltatásával is hozzájárultak olyan víruskárok elkerüléséhez, amelyek ezen információk nélkül feltétlenül bekövetkeztek volna.

Bár könyvünkben az IBM-kompatibilis gépek vírusaival foglalkozunk, a leírtak jó része – főleg az általános terjedési elvekkel és részben a védekezési stratégiával kapcsolatban – érvényes más rendszerkörnyezetre is. A C-64-es, az Atari, az Amiga és a Macintosh gépeknek és a a nagygépes operációs rendszereknek szintén megvan a maguk vírustenyészete. Hogy hazánkat a nagygépes vírusok pusztítása eddig elkerülte, arra nem igazán lehetünk büszkék – ez pusztán csapnivaló adatátviteli rendszerünknek és telefonhálózatunknak köszönhető. A gépek sokfélesége és elszigeteltsége ez esetben előnyt jelentett, de ez remélhetőleg már nem tart sokáig... A politikai változásokkal együtt megnyílt annak a lehetősége, hogy a korábbiaknál nagyobb teljesítményű berendezések kerüljenek be teljesen legálisan az országba. A DEC VAX és az IBM AS400 gépeinek megjelenésével már megkaptuk az első Decnet vírust, amely a DEC-gépek lokális halózati rendszereiben fejti ki pusztító hatását. Ugyanakkor igen erélyes vírusfejlesztések indultak meg a Novell alapú hálózati rendszerek ellen. Ezek veszélye – különösen ha 1991 végén megkezdi működését a Magyar Távközlési Vállalat-

nál az első, valóban üzemképes országos csomagkapcsolt kommunikációs adathálózat – a későbbiek során jelentősen megnő.

Nagyon sok múlik azonban azon, hogy a forgalmazók a különben igen drága berendezésekhez képesek lesznek-e megfizethető áron megfelelő szoftvereket is biztosítani. Mert ha nem, akkor a szoftverkommuna ezekre a nagy berendezésekre is kialakul, annak minden veszélyével együtt. A gépek méretéből és hálózati kiépítéséből következően az okozott kár azonban itt jóval nagyobb lehet.

Ne dugjuk a homokba a fejünket, inkább készüljünk fel szakmailag és etikailag is az új korszakra! Ehhez természetesen hozzátartozik a megfelelő törvények megalkotása és elfogadása. A forgalmazóknak viszont az eddigi árkalkulált feladva, tisztességes áron, tisztességes szoftvereket kell árusítaniuk, hogy ne legyen érdemes szoftvert lopni... és a szoftvert meg az adatokat tönkretenni. A védelem nélküli Quattro új verziójának vagy a PC Tools eredeti változatának üzleti sikere megmutatta, hogy lehet Magyarországon kifizetődő üzleteket tisztességesen is kötni. Az így vásárolt szoftvereket ugyanis általában nem adják tovább megvásárlóik, hanem becsületbeli ügynek tekintik a szerződés betartását. Most a forgalmazókon múlik, hogy ehhez megfelelő ármegállapítással és védelem nélkül kapható programokkal megfelelő alapot nyújtsanak. A felhasználók nem kívánnak mindenáron szoftvert lopni.

Hippokratész, az orvostudomány atyja, a jó orvos alapvető tulajdonságának tartotta a szaktudás mellett a fejlett etikai érzéket. A számítástechnikai berendezésekkel foglalkozó programozók, kereskedők ugyanabban a helyzetben vannak, mint az orvos a beteg mellett: pusztítani és gyógyítani egyaránt tudnak, mások számára szinte észrevétlen módon. Mi a magunk részéről Hippokratész felfogását valljuk, *nil nocere*. Arra törekszünk, hogy amikor használni nem tudunk, legalább ne ártsunk a számítástechnika ügyének. Ezért olvasóink is hiába keresnének víruskód-forráslistákat a könyvben. Az élet rákényszerít bennünket arra, hogy néhány dologról hallgassunk. Mindannyiunk érdekében.

Budapest, 1991. szeptember 15.

Kis János – dr. Szegedi Imre

MÁSOLÁSVÉDELEM ÉS BIZTONSÁG

Amikor a pandúrból rabló lesz

Ebben a fejezetben részben az első kiadás *Jótékony köd, mely ápol s eltakar* című anyaga szerepel, ami miatt azóta rengeteg támadás ért bennünket, hogy tisztességes szoftverforgalmazás helyett mi a szabadrablást, a szabad programlopást szeretnénk megvalósítani. Igyekszünk tehát álláspontunkat most bővebben és világosabban leírni. Ezzel a témával egyébként az *Alaplap* 1991. januári száma részletesen foglalkozott, és abban az ellentábor egyik prominens képviselője kifejtette a miénkkel gyökeresen ellentétes nézeteit a felhasználókról, forgalmazókról és a vírusokról, azt állítván, hogy a programozóknak, forgalmazóknak (egyéb szabályozás hiányában) egyedüli védekezési eszköze az önbíráskodás. Szerintünk a szoftvernek az árával és a szolgáltatásaival kell eladnia magát, nem pedig egyéb praktikákkal. A tisztességes forgalmazás az előfeltétele annak, hogy a felhasználók is tisztességesek legyenek!

Úgy látjuk, hogy a felhasználók is kezdenek felébredni tespedtségükből. Egyelőre még messze vannak attól, hogy egyesületbe szerveződve védjék érdekeiket, de a védelmek íróihoz hasonló informális csatornákon sokat tesznek a tisztességes szoftverpiac kialakítása érdekében. A KeyProject keretében például rövid idő alatt elkészülnek a másolásvédelemmel forgalmazott szoftverekhez biztonságosan alkalmazható védelemfeltörő programok. A forgalmazók többsége vagy nem vesz tudomást róluk, vagy pedig mindent megtesz a programok íróinak kiderítésére és lehetetlenné tételére. Eddig egyetlen olyan cég volt, amely másként reagált. A Rolitron Bioelektronikai Rt. 1990 végén hardlock védelemmel hozta forgalomba a Rosytext szövegszerkesztő normál billentyűzetes változatát. Ennek feltörése a KeyProject keretében néhány nap alatt elkészült, amire a cég visszavonta a hardlockos verziót, és egyelőre csak a normál billentyűzetes verziót forgalmazza. Tóth Bélát, a termék menedzserét, a Rosykey program gyors megjelenése nem lepte meg, csak azt fájlalta, hogy a kísérő dokumentációban a feltörő program szerzői arra utalnak: hardlock nélkül a program kárt okoz. Szerinte némi kellemetlenséget okoz ugyan, de kárt semmilyen körülmények között sem...

Minél alacsonyabb egy országban az alkalmazástechnika kultúrája,

annál jobban be lehet csapni a felhasználót. Ilyen környezetben még olyan nagyvállalatok, szoftverházak is képesek megélni, amelyek a tisztességtelenséget a cég üzletpolitikájának rangjára emelik. Nagy értékű szoftverrendszerekbe beépítenek például olyan időzítő rutinokat, amelyek hatására a szoftver nem sokkal a garanciális idő letelte után tönkremegy. Mások másolásvédelem ürügyén okoznak kárt a felhasználó gépében, adatbázisaiban.

A Clippert gyártó és értékesítő neves amerikai cég legelső programverziói bizonyos számú futtatás után megölték magukat, s ha a gépen véletlenül volt Clipperrel fordított adatbázis-kezelő rendszer, akkor azt is megsemmisítették. Miután ebbe a trükkbe majdnem belebuktak, a későbbi verziókat már korrektül készítették. Magyarországon a forgalmazó maga próbálta meg másolásvédelemmel ellátni a Clipper hazai változatát. A felhasználók azonban inkább a védelem nélküli kalózpéldányokat használták, így végül hazánkban is az eredeti, védelem nélküli szoftvercsomag került forgalomba...

A Unix alapú rendszerek esetében ott vagyunk, mint valamikor a PC korszakának kezdetén. PC-s programokat az USA-ban ma már szinte csak másolásvédelem nélkül forgalmaznak, a Unix alatti programokat azonban annál inkább védik. A hardlockot szemérmesen átnevezték „security block"-ra. Szerepe és részben áramköre is azonos a PC-s programokat „védő" hardlockokéval. Ugyanakkor a Unix kínál még néhány, a felhasználó pénztárcáját alaposan megcsapoló lehetőséget. Például az egygépes licenccel eladott program csak egy adott sorszámú gépen futtatható. Itt is licence válogatja, hogy egy vagy több feladat indítható egyszerre. A következő fokozat a sétáló hálózati licenc, mellyel a hálózaton belül bárhonnan futtatható a program, de egyszerre csak egy példány létezhet. És végül van a józan ésszel normálisnak ítélt lehetőség, amikor egy hálózaton belül bármennyi példányban futhat és létezhet egyszerre a program. (Ilyen szintű programmal szinte csak bétatesztes amerikai verzióban találkozhatunk.)

Az elmúlt évek nagy számítógépes rendszerösszeomlásai, adatkatasztrófái megmutatták, hogy a számítógépes terrorizmus nagyobb veszélyeket rejt, mint azt korábban gondolták. A legnagyobb gondot az érzékeny, esetlegesen háborút is elindító rendszerek megpiszkálása és a kizárólag számítógépen tárolt, kulcsfontosságú adatok elvesztése jelenti. A nyugaton kirobbant KGB-s adatlopási botrány, az USA Arpanet tudományos-katonai adathálózata ellen elkövetett szabotázs bizonyos etikai kódex kialakítására kényszerítette a számítógépes hálózatok kalózait is. A német és amerikai számítógépes betörők nagy része – bár szorgosan folytatja az adathálózatok feltörését – a megtalált adatokat nem

módosítja, a „kapott" üzleti információkat csak saját célra használja fel. Ha pedig egy érzékenyebb rendszerbe hatol be, akkor jelzi az üzemeltetőnek, hogy védelmi rendszerében valami hiba van. (Hisz' aki számítógépes programot ír, ugyanúgy hibázhat, mint mások...)

A programok és adathálózatok feltörésével korábban önfitogtatási, károkozási céllal foglalkozó, ezáltal a szakma fenegyerekeinek számító fiatalok (a hackerek vagy crackerek) közül egyre többen döbbentek rá: az informatikai társadalmakban tudásuk olyan fegyverré is válhat, mint az orvosé, a mikrobiológusé vagy az atomtudósé.

1989 végén az egyik hackertalálkozón elfogadtak néhány olyan alapszabályt, amely lehetővé teszi, hogy ne kelljen mindenki által megvetve a társadalmon kívül, szakmai illegalitásban élniük. Németország állambiztonsági szervei bár nem engedélyezik, de már megtűrik a korábban csak illegálisan kiadott *Bayerische Hackerpost, Datenschleuder* vagy a *Hackerbibel* című hackerkiadványok rendszeres megjelenését, és elnézik azok könyvesbolti terjesztését is. Érdemes e kiadványok alapján összefoglalni etikai kódexüket, amelynek néhány pontjával nem értünk ugyan egyet, de annak elfogadása mindenképpen konszolidáltabbá tette az addigi kaotikus és kiszámíthatatlan veszélyekkel járó helyzetet.

*Csillaggal megjelölve hozzávettünk egy-két kifejezetten hazai vonatkozású elvet, melyek az egyre szorosabbra fűződő kapcsolatok révén Németországban is kezdenek terjedni.

1. Vírust ne írj, víruskódot ne adj oda kívülállónak, ne terjessz, mert következményei beláthatatlanok!
2. A kommunikációs hálózatokat használd ki, derítsd fel! Szolgáltatást lopni nem bűn. De ne tedd tönkre ezeket a rendszereket, amelyek a te kényelmedet is szolgálják.
3. A fizető adatbankokat ingyenesen nyisd meg magad és a mások számára! Információt lophatsz, de azokat a rendszerekben módosítani és törölni tilos.
4. Ha egy érzékeny (katonai, nemzetbiztonsági stb.) rendszerbe sikerült bejutnod, akkor a hackertársadalom megbecsült tagjainak segítségével és a rendszerben elhelyezett információval hívd fel a gazdák figyelmét arra, hogy lyukas a védelmük. Ne feledd, nemcsak hecc az adatlopás, hanem hatalmi téboly és terrorista őrület is fel tudja használni ezeket az adatokat, s téged is kényszeríthetnek tudásod kiadására. Egy esetleges atom- vagy biológiai háború kirobbantása neked sem lehetne érdeked, mert azt te sem élnéd túl.
5. Noha a banki, pénzügyi rendszerek nem érzékenyek, azok módo-

sítása éppen olyan, mintha fegyverrel rabolnál bankot. A következményei is ugyanazok!

6. Amit megtudtál az egyes számítógépes rendszerekről, az nem lehet üzleti alku tárgya. Csak egymást közt adható tovább, mert különben terroristák, ipari vagy politikai kémek célpontjává válsz!
7. Vírus és másolásvédelem írásához sem pénzért, sem pedig szívességből senkinek ne nyújtsd segítséget! Ha valakit ilyesmin kapsz, tegyél meg mindent az általa okozott kár következményeinek enyhítésére.
8. Lépj fel minden olyan jelenséggel vagy cselekedettel szemben, amely az informatikai társadalom stabilitását veszélyezteti! Ne rombolj, hanem járulj hozzá konstruktívan. A magánszféra számítógépes ellenőrzését azonban saját eszközeiddel minden módon akadályozd meg!

*9. Másolásvédett program védelem nélküli verzióját ingyen add oda annak, aki kéri, hogy minél kevesebb programot tudjanak védetten eladni.

*10. Ha védett program feltörését kérik, tedd meg, ha tudod. Ha nem megy, keresd meg azt, aki képes rá. A felhasználót se vágd meg, mert nem illik komoly hasznot húzni olyan dologból, amit magad is elítélsz!

*11. Másolásvédelem nélkül, elérhető áron forgalmazott programot olyannak odaadni, aki azt nyugodtan megvehetné, illetlenség. Mit szólnál hozzá, ha az általad készített olcsó programokból minimális bevételed sem lenne? A védelmeket leszedő programokat viszont mindig ingyenesen add tovább!

*12. Egy program rendszerüzeneteit átírhatod, de a szerzői jog jelzését átírni tiszteletlenség. Különösen erkölcstelen dolog az ilyen programot sajátként árusítani.

A magyar jogrendszer még nem készült fel ilyen informatikai kihívásokra. Sokáig egyes személyek – sőt neves szoftverforgalmazó cégek – azt is megakadályozták, hogy valóban részletes információk jelenjenek meg a számítógépes vírusprogramokról. Érveik sorában hivatalosan az állt első helyen, hogy mi lenne, ha mindenki elkezdene vírusokat írni. De a háttérben az a félsz bujkált, hogy saját trükkjeik is lelepleződhetnek és extraprofitjuk csökkenne. Korszerű jogrendszerben nem tartható fenn az információnak ez a monopóliuma. Az általánosságok mellett konkrétumokról is kell írni, hogy minden számítógép-alkalmazó megismerkedhessen az őt is fenyegető láthatatlan veszedelemmel, és tenni tudjon valamit ellene.

A másik hivatalos indoklás az volt, hogy ezekkel az ismeretekkel visszaélhetnek a terroristák, inkább ne terjesszük. Ebben sajnos van valami igazság. Egyes nyilvánosságra került víruskódok (Töltögető, Jerusalem, Reboot), valamint oktatási célra készült ártalmatlan, demonstrációs vírusok kódjainak átírása a vírusok szinte beláthatatlan mennyiségű változatára adott ötletet. Ezért mellőzzük mi is – a vírustalanítók nemzetközileg elfogadott gyakorlatához alkalmazkodva – a víruskódok, valamint a nem tájékoztatásként, hanem a gyakorlati életben az egyes verziók azonosítására használt másodlagos vírusazonosítók közzétételét. Hasonló okokból vírust nem adunk ki senkinek, csak a nemzetközi víruscserében részt vevő, megfelelő jogi és emberi garanciákkal rendelkező intézeteknek, cégeknek és programozóknak. Ezek száma a világon igen kicsi, nem éri el a hatvanat...

Évekkel ezelőtt azt hirdettük, hogy a számítógépvírusok és más kártevő szoftverek Magyarországon nem fejthetik ki tevékenységüket. Most pedig már nemcsak a külföldön elterjedt vírusprogramok ütötték fel fejüket, hanem saját, hazai tenyésztésű vírusváltozatok és trójai programok is felbukkantak. Legtöbbjük másolásvédelem ürügyén. Magyarországon találták fel az automatikus vírusgenerálást is, amikor is egy szoftverrendszer az ismert víruskódot adott helyeken szétvágja és NOP (azaz üresutasítás) parancsokkal spékeli meg. Ez megváltoztatja a vírus hosszát, és a vírusellenes programok ugyan felismerik, de rosszul irtják, tönkretéve a szoftvert. Ezzel a technológiával a vírusazonosító, azaz a keresési szekvencia is alaposan megváltoztatható. Sajnos az eljárás teljesen automatikusan, a kód visszafejtése nélkül is alkalmazható. Ennek „gyümölcse" a *kitolós potyogós,* amelyet a hagyományos programok rosszul irtanak...

Az egészségügyi járványtani számításokkal foglalkozók már az ötvenes években felfigyeltek arra, hogy egy fertőző gócból kiinduló járvány terjedése nagyon jól modellezhető. Amennyiben ráadásul egy gyógyíthatatlan kórról van szó, akkor a megfertőzhető népesség mintegy kétharmadának kihalása után a fertőzés önmagától már nem terjed tovább, majd teljesen megszűnik. Ezt a teóriát a középkor nagy pestisjárványai a gyakorlatban igazolták. S lám: hasonló törvényszerűségek vonatkoznak a vírusprogramok terjedésére is. A számítógépvírus az élő anyag működőképes modellje!

E tárgyban az első komoly publikáció egy ilyen járványmatematikával foglalkozó szakember tollából látott napvilágot még 1957-ben! (N. T. J. Baily: *The Mathematical Theory of Epydemics.* Hafner, 1957.) Természetesen senki sem hitt a szerzőnek, tanulmánya eltűnt a hasonlóan száraz anyagok süllyesztőjében. Maga a programvírus, illetve vírus-

program fogalom is jóval később, 1974-ben bukkant fel a szakmai publikációkban. (A két elnevezés sokáig mindenféle jelentésbeli különbség nélkül, vegyesen volt használatos. Az utóbbi időben már kezd kialakulni, hogy a vírusprogram az átfogó kategória, a programvírus pedig azon belül a programokat megtámadó vírusféléket jelöli, elhatárolva például a bootvírusoktól.) Az ACM-nek a *Use of Virus Functions to Provide a Virtual APL Interpreter under User Control* című tanulmányában találhattunk rá először erre a meghatározásra, egy B. Gunn nevű szerző munkájában. A következő publikáció az 1982-ben megjelent *The Worm Programs – Early Experience with a Distributed Computation* című, ma már beszerezhetetlen tanulmánykötet.

Európában a Dortmundi Egyetemen ugyanezzel a témakörrel foglalkozott 1980/81-ben J. Kraus. A kutatás ekkor még a legteljesebb titoktartás mellett folyt. Egyes katonai körök úgy látták, hogy a vírusprogramok alkalmasak érzékeny technológiák és szoftverek le- és elloprásának megakadályozására és az ellenséges hatalmak számítógéprendszerének totális megbénítására. A bomba 1984-ben robbant az NSZK-ban: a Der Spiegel hírmagazin (1984. 47. szám) *Verborgener Befehl – Bericht über Cohens Arbeit* címmel rövid cikkben számolt be az önreprodukáló programok létéről, felhíván a figyelmet a számítógépes technikai kultúrára leselkedő veszélyekre is.

Milyen különös véletlen! Magyarországon is akkor bukkant fel az első vírusprogram, igaz, még Commodore 64-en. Ennek hordozója néhány népszerű játékprogram volt. Terjesztője, egy a C–64 javításával foglalkozó „szakember" így akart nagyobb forgalomra, könnyű kereseti lehetőségre szert tenni. Programvírusa ugyanis olyan külső pályára vitte a lemezmeghajtó olvasófejét, ahonnan csak kézzel, a lemezegység szétszedésével lehetett visszavezényelni. Mellesleg hazánkban ezt a vírust használták először másolásvédelemre egy kereskedelmi forgalomban árusított Commodore könyvelőprogramban.

Az információt egy idő után már nem lehetett visszatartani. Botorság volt azt hinni, hogy a szakirodalom kirekesztésével nem terjed tovább ennek az új programozási lehetőségnek az ismeretanyaga. A témával konkrétan foglalkozó elméleti publikációk sorát az amerikai Friedrich Cohen *Computer Viruses, Theory and Experiment* (University of Southern California, 1983) gyakorlati tanulmánya követte. Cohen végzett először valós kísérleteket az egyetem VAX 11/750 típusú gépén Unix többfelhasználós rendszerben, amit megismételt egy VMS-VM/370 operációs rendszerű hálózati környezetben is. Az eredmény több mint megdöbbentő volt. A többfelhasználós (multitask) környezetben a vírus elindítása után gyakorlatilag nulla időpillanatban mind a 33 rend-

szerállományt és az adminisztrátor programállományát megfertőzte, majd az elindítás utáni 18. másodpercben a négy felhasználó állományai is fertőzöttek voltak. A hálózatos rendszerben a hatszázadik másodpercben vált teljessé a fertőzés.

Hangsúlyozni kell azonban, hogy ezek a rendszerek annak idején semmilyen külön beépített védelemmel sem rendelkeztek a vírusok ellen. Ennek akkor még nem sokan mérték fel a veszélyeit. Megjelentek a vírusprogramot kibocsátó, tehát az illegális programmásolókat megbüntető (önbíráskodó) másolásvédelmi programrendszerek.

Az első nem elméletieskedő publikáció a német hackerek *Bayerische Hackerpost* lapjában jelent meg (Adalbertstr. 41/B, D-8000 München 40). Ez a kiadvány a világ azon kevés technikai szamizdatjai közé tartozik, amelynek ismerete elengedhetetlen a számítógéprendszerek biztonságtechnikájával foglalkozó szakemberek számára. (Mellesleg a titkosszolgálatoknak kedvenc olvasmányai közé tartozik.) Hasonlóan érdekes információkkal szolgál egy másik hackerkiadvány, a Chaos Computer Club viszonylag rendszeresen megjelenő periodikája, a *Datenschleuder* is. Ez adta közre az első teljes és valóban futtatható vírus-forráskódot, 1986. évi 12. számában. (B. Fix Virussource Rush-Hour. Chaos Computer Club e. V. D-2000 Hamburg 20, Schwenkenstr. 85.)

1989 februárjától Magyarországon is végigsöpört a vírusjárvány. Az első, komoly károkozásra képes vírusprogram a rendszer folyamatos újraindítását eredményező Reset vagy Reboot vírus volt. Hála az előzetes felvilágosító munkának és a szabadszoftverként is terjesztett vírusölő programoknak, viszonylag kevés kárt okozott. Ennél többet pusztított 1989 őszén a Péntek 13 vírus, amelynek első példányait a Szovjetunióból egy program hurcolta be floppyn az egyik bányavállalatunk számítóközpontjába, ahonnan azután továbbterjedt. A vírus eredetije a Magyar Postánál, valamint a Budapesti Műszaki Egyetem számítógépes rendszerében programok és adatok végleges elvesztését és a rendszer leállását okozta, de a helyzet még mindig nem volt annyira katasztrofális.

A szovjet vírusváltozat egy izraeli eredetű vírus, az Israeli #2, más néven Jerusalem-B „megpatkolt" verziója. Az eredeti onnan kapta nevét, hogy a palesztinok 1987 decemberében az izraeli Hebrew University (Jerusalem) számítógép-hálózatába juttatták be azzal a céllal, hogy azon a napon, amikor 13-ika péntekre esik, törölje az általa megfertőzött állományokat. Ezt a vírust a Szovjetunióban teljesen visszafejtették, majd némileg módosítva újrafordították. Sajnos egyre több Magyarországon készült átírása is megjelent, amelyet a hagyományos killerek és detektorok nem érzékelnek. Például felbukkant egy olyan változat is,

amely az elkövetkezendő első keddre, 1990. május elsejére volt programozva, hogy elpusztítsa a fertőzött adatállományokat. Hasonló sorsra jutott a magyar eredetű Töltögető is, amelyet a Szovjetunióban átbarkácsoltak, s azóta (rendszerüzenetére utalva) Gorbacsov néven járja a világ számítástechnikai rendszereit.

Az 1990-es és 1991-es esztendő hazai slágere a Stoned/Marijuana és a Vacsina/Yankee Doodle vírus volt. Immár tömegesen jelentek meg fertőzött gyári szoftverek, ezek mindegyike Tajvanból származott. Az 1990-es év legnagyobb kárt okozó leállását is egy ilyen tajvani „állat" okozta a Margaréta Csomagküldő Szolgálat rendszerében. Mivel a fertőzés elhárításában csoportunk vett részt, sikerült kinyomozni a forrást is: egy Tajvanból érkezett laptop gép merevlemeze és segédprogramjai hozták a vírust. Ugyanez a fertőzés ugyanilyen forrásból egy másik számítástechnikai rendszeren is fellépett. Megjegyzendő: a Margaréta Csomagküldő Szolgálat volt az első olyan vírusáldozat, amely nem tagadta le a nyilvánosság előtt a fertőzést, hanem a rádió, televízió és a szaksajtó nyilvánosságát vállalva korrektül tájékoztatta ügyfeleit és a szakmai közvéleményt.

A vírusprogramok ellen a világ minden jogszerűségre törekvő országában hivatalosan fellépnek. A vírusok íróit és tudatos terjesztőit vagy az adatvédelmi törvény keretében (mint az USA-ban), vagy a polgári törvénykönyvben a károkozással kapcsolatosan (mint Németországban), vagy pedig a terrorizmus elleni harccal és az állambiztonsággal kapcsolatosan büntetik (mint például Izraelben). Az Amerikai Egyesült Államokban az államigazgatásban, a hadseregben nem alkalmazható egyetlen olyan program sem, amely bármiféle másolásvédelemmel van ellátva. Sőt az USA majdnem minden tagállamában tilos az ilyen programok kereskedelmi forgalomba bocsátása is.

A törvényi szankciók az európai országok nagy részében sajnos nem igazán hatékonyak, és a jogszabályok is kétértelműek, ezért rendelkezéseik egyelőre még kijátszhatók. A szaklapokban nem is egy másolásvédelmi eszköz (program és hardverlock) hirdetését olvashatjuk. Márpedig ha hirdetik, akkor valószínűleg vevő is akad rá. (Az 1991-es Cebit szakvásáron ismét találkozhattunk ilyen termékkel.) Súlyosbítja a helyzetet, hogy a kevéssé vagy egyáltalán nem hozzáértő vevőnek bemesélik: a másolásvédelem kifejezetten vírusvédelmet szolgál, vagy éppenséggel megakadályozza a drága pénzen megvásárolt programjuk eltulajdonítását. Sajnos a CAD/CAM szoftverek forgalmazói is ilyen trükkökkel élnek, holott itt különösen fontos lenne a gyors telepíthetőség, a megbízhatóság, s hogy a felhasználó a programot saját vállalatának határain belül úgy használhassa, ahogy az munkájához a legmegfelelőbb.

(Pedig vannak nem károkozó, tisztességes lopásvédelmi eljárások! Ilyen például, hogy a programot „beégetett" sorszámmal vagy a felhasználó nevére szóló dedikációval látják el, s így nyomon követhető az illegálisan forgalmazott példányok eredete.)

A másolásvédelem fogalmát sokan összekeverik – nem egy esetben tudatosan! – a hozzáférési jogok, illetve a szelektív hozzáférés biztosításával, pedig két különböző dologról van szó.

A másolásvédelem célja a program futásképességének megakadályozása vagy tönkretétele, ha a programot nem a forgalmazó által készített eredeti lemezről indítják (kulcslemezes futtatás), illetve ha nem építenek be vagy nem csatlakoztatnak a géphez szintén az eladó által rendelkezésre bocsátott, nem szabványos hardvereszközt (kulcskártyát, hardverlockot). Amennyiben a program nem talál megfelelő környezetet, jó esetben csak öngyilkos lesz, rossz esetben aránytalanul nagy kárt is okoz a felhasználónak, önkényesen megbüntetve őt az illegális másolásért. Rafináltabb módszerek esetén ez a rutin később – egy látszólag zavartalan működési periódus után – aktivizálódik, ezáltal a kártétel még nagyobb lesz. Ha a lappangási idő alatt nem szabadít ki magából szaporodásképes kórokozókat, akkor „trójai funkcióban" működik, ha pedig kiszabadít, akkor vírusprogramként lép akcióba. Mindenképpen illetéktelen beavatkozás a felhasználó munkájába. (Ki venne például olyan kalapácsot, amely csak egy bizonyos üzem, bizonyos színűre festett szobájában, egy meghatározott gyári számú munkaasztalon, az eladó által rendelkezésre bocsátott és sorszámozott satuban megfogott munkadarabhoz lenne használható?! S ha ezen feltételek valamelyike hiányozna, akkor a kalapács felrobbanna, elpusztítva az üzemet és használóját. Hogy ez az analógia azért túlzás? Igen, de csak kis túlzás!)

A másolásvédelemmel ellentétben a hozzáférés-védelem a számítástechnika egyik legtermészetesebb eszköze. Azt határozza meg, hogy milyen jogkörű felhasználó mely adatállományokat vagy azok mely részeit és milyen jogosítvánnyal, beavatkozási lehetőséggel (írás, olvasás, írás és olvasás, keresés, másolás stb.) használhatja. Ezt vagy az adathálózat biztosítja a hálózati vagy normál operációs rendszer alapszolgáltatásaként (pl. Novell, Unix), vagy pedig szoftveres titkosítással oldják meg. Ilyen titkosítás esetén csak a kulcsszó (kulcsszavak) ismeretében végezhetők el meghatározott műveletek a meghatározott állományokban, de kárt akkor sem okoznak, ha nem ismerjük a kulcsot és illetéktelenül próbálkozunk belenézni a számunkra nem hozzáférhetővé tett „aktákba". A rendszer ilyenkor sem töröl állományokat, legfeljebb megtévesztő információkkal traktál bennünket. Például a PKARC, a PKXARC, a PKPAK, a PKZIP rendszer rossz jelszó megadásakor „szemetet" csoma-

gol ki magából, vagy pedig hibaüzenettel utal arra, hogy az állomány sérült. Kárt azoban soha nem okoz!

Hazánkban a vírusok járványszerű terjedése 1990-ben felgyorsult. Ennek több oka is van. Egyes forgalmazók például egy-egy program néhány eladott példányával szeretnének meggazdagodni. A főkönyvi rendszerek tucatjaiban találhatunk olyan másolásvédelemnek álcázott időzített aknákat, amelyek a forgalmazók bevételét növelik, s teremtenek számukra folyamatos piacot. Ezek a programok sok esetben a megadott idő letelte után maguk is vírussá válnak, megrongálják az adatállományokat, sőt egy részük terjedni is képes.

Ismét számolnunk kell azzal a jelenséggel, amivel a Commodore 64 megjelenésekor találkoztunk, hogy a javításra szoruló gépek száma mesterségesen is növelhető. Láttunk olyan vírusirtó programot is, mely ugyan levette a vírust, de biztos ami biztos alapon feltett egy másikat, amelyik ellen nem volt hatásos. Most pedig esély van arra is, hogy egyesek maguk írnak olyan vírust, melyet aztán komoly tarifáért kiirtanak. Nem véletlen, hogy Nyugat-Európában a számítógépvírusok elleni programokat vagy önköltségi áron, vagy ingyenesen biztosították a felhasználóknak.

A vírusirtó szoftverek olyanok, mint az elsősegély. Egy adott problémát oldanak meg. Viszont a hangsúlynak a fertőzés megelőzésén kellene lennie. Az erre a célra forgalmazott szoftverek árából legalább a fejlesztési költségeknek és a témával foglalkozók bérének, közterheinek meg kell térülniük, emiatt ezek a vírusmegelőző védelmi programrendszerek igencsak drágák. Ráadásul a piacon kapható termékek egyre nagyobb hányada alapoz az emberek félelmének megvámolására. Szerencsére az olcsó közprogramok, a freeware-ek, shareware-ek között is találhatunk valóban jó programokat – ezeknek a komoly szellemi befektetést tartalmazó szoftvereknek az ára csak töredéke a hasonló tudású kereskedelmi szoftverekének. A vírusellenes törekvéseket, saját érdekükben, a nagy forgalmazó cégek is támogatják fejlesztési és reklámpénzükből. Erre különösen akkor figyelhetünk fel, amikor náluk is felüti a fejét valamilyen alattomos vírusjárvány.

Állítólag a piaci helyzet kényszeríti arra a forgalmazókat, hogy Magyarországon másolásvédett szoftvereket hozzanak forgalomba. Valójában ez segíti az irreálisan magas ár fenntartását, és fokozza a vevő kiszolgáltatottságát az eladóval szemben. Hazánkban a másolásvédelem szabályozása várhatóan akkor lép igazán életbe, amikor a biztosítási üzletágban megjelenik az adatok, adatállományok biztosítása. Ez törvényszerű, hiszen az adatbázisokba foglalt információk és a programok rendkívüli értéket képviselnek.

A különösen érzékeny technológiákat nyugaton inkább speciális számítógépekkel védik. Ennek tipikus példája az USA minisztériumaiban általánosan használt, Alfa Micro névre hallgató, igen nagy teljesítményű megamikro gép. Az embargó enyhülésével a jó üzlet reményében ezek a gépek hazánkban is megjelentek. Csúfos vereségüket a csakis önmagukkal való kompatibilitásnak tudhatjuk be. Egy gazdag ország azonban megengedheti magának, hogy ekképp védje adatait.

A legnagyobb probléma az adatvédelem és a másolásvédelem esetenként tudatos összekeverése. Pedig a másolásvédelmek készítői, a vírusírók és az informatikai terroristák ugyanazt teszik. A vírusíró és a terrorista válogatás nélkül rombol, a másolásvédelmek írója pedig „üzleti feltételekhez" köti a károkozást.

A másolásvédelmek egyik legősibb csoportja a lemezekkel manipulált. Hazánkban a lézerlyukas módszert alkalmazták: a lemezen lézerrel fizikai hibát okoztak. A program megpróbált erre a területre írni, s ha sikerült, akkor a szoftver nem futott. Néhány felhasználó összefogásával bírósági úton sikerült igazolni, hogy a lyuk károsítja a lemezolvasó fejet, így erről a módszerről viszonylag hamar lemondtak.

A következő kategóriába a lemezre író védelmi rendszerek tartoznak. Ezek napjainkban élik virágkorukat. Az ilyen védelem legtöbbször kihasználja a BIOS-on keresztüli lemezírás lehetőségét: az operációs rendszer feje felett átnyúlva olyan formátumokat ír a lemezre, amelyeket csak ugyanígy lehet olvasni.

A másolásvédelmek mindenképpen beavatkozást jelentenek a szoftverek használóinak tevékenységébe. Nem véletlen a felhasználók elemi erejű, ösztönös tiltakozása a másolásvédelmek alkalmazása ellen. Ez jelentkezik egyrészt e programtermékek megvásárlásának bojkottjában, másrészt abban, hogy megvásárolt programjaikat igyekeznek megszabadítani a védelemtől. Nagyobb egyetemi városainkban mintegy hetvenen jutnak egy kis rendszeres kereseti lehetőséghez a védelmek „levakarásával". Németországban is jól képzett szoftveres csoportok dolgoznak azon, hogy a legális felhasználók programját megtisztítsák a szoftveres és hardveres másolásvédelmektől. Ugyanez az igény szülte a sorszámátírós Autocad 9.0 változat megjelenését: a felhasználók a saját sorszámukkal szerették volna látni a gépen a védelem nélküli amerikai változatot.

Vírustalanító tevékenységünk során igen sok felhasználóval kerültünk kapcsolatba. Furcsa módon azt tapasztaltuk, hogy bár legtöbbjüknek birtokában volt a jogos szoftverpéldány, mégsem azt használta, hanem a védelemtől megfosztott változatot vagy egy védelem nélküli amerikai példányt. Ezért nem volt értelme azoknak az akcióknak, amikor

„szoftveramnesztia" keretében védett példányra lehetett kicserélni a védelem nélkülit.

Különösen nagy a veszélye annak, ha az operációs rendszert látják el másolásvédelemmel. Ilyenkor teljes adatvesztést kockáztat a vásárló. Ha bármilyen probléma van az eredeti lemezzel (pl. a magyar DR DOS esetében), az eredeti rendszer kívülről nem állítható fel.

A hardverkulcsos megoldásnál (állítólag így fejlesztik ki a Nexos programrendszer vagy a PageMaker magyarul tudó, kelet-európai verzióját) sincs kizárva annak a lehetősége, hogy géphiba esetén teljesen elveszítsük adatainkat. Tapasztalataink szerint a legtöbb hardverkulcsos rendszer nem olyan ártatlan, mint amilyennek feltüntetik. Fejlettebb változataikban külön rutinok védik a hardverkulcsot ellenőrző programrészeket, s ezek sérülését úgy értelmezik, hogy valaki a programot akarta feltörni, amire a legváltozatosabb megtorlási módokat alkalmazzák.

Kecskeméten egy társaság olyan hardverkulcsot forgalmaz, amelynek használatával alaposan megfejhető a szerencsétlen programvásárló. Ugyanezt a trükköt alkalmazták az egyik legsikeresebb számítástechnikai pénzszerző akcióban a Trojan AIDS Information program szerzői és terjesztői is (lásd könyvünk 3. fejezetében). Kecskeméten korábban egy másik cégtől egy másik, hasonlóan tisztességtelen védelem is megjelent. Ha ezt a Clipperbe befordítható „védelmi rutint" egy másik gépre átvitték, adott számú futtatás után leformázta a merevlemezt. (S a szerző még büszke is volt arra, hogy ő a forgalmazók érdekeit minden eszközzel megvédi!)

Most lássuk, hogyan büntethet a másolásvédelem. Éppen úgy, mint egy vírus! A legegyszerűbbek csak magát a programot teszik futásképtelenné. Az újabbak – mondván, bírósággal semmire sem lehet menni – az önbíráskodást tekintik céljuknak. Minél értékesebb egy programrendszer, annál durvábban ütnek vissza a felhasználóra.

Megrongálhatják annak a könyvtárnak az adatait, ahol a program található. Ennek könyvelési programoknál fenomenális hatása lehet, elveszhet egy cég teljes adminisztrációja akár egy évre visszamenőleg, még abban az esetben is, amikor a lehető legjobb indulattal, géphiba vagy karbantartás miatt az egész rendszert áttelepítik egy másik gépre.

Ennél egy fokkal durvábbak azok a büntető másolásvédelmek, amelyek a merevlemez egész tartalmát teszik tönkre. Ezt a módszert a másolásvédelmek forgalmazói hivatalból is tagadják. Hasonlóképpen tagadják, hogy a hardverkulcsos rendszerek kárt okoznának. Pedig van olyan hazai forgalmazású hardlockos proram, amely a soros–párhuzamos kártyát teszi tönkre, mások a lézernyomtatóra vagy magára a gépre jelentenek veszélyforrást.

A forgalmazók azzal a látszólag jogos érvvel védekeznek, hogy meg kell védeniük érdekeiket a kalózokkal szemben. Nos, ez lehetséges, csak éppen kicsit többet kellene foglalkozni a programmal eladás után! Nem úgy, mint manapság szokás, amikor az ősi „becsali (vándor)kereskedő" elve uralkodik. („Add a pénzt, itt az áru, de most aztán többet ne is lássalak.")

Tanulságos volta miatt az alábbiakban ismertetjük egy amerikai szoftverkereskedő ezzel kapcsolatos nézeteit. Kínálatában megtalálható a nálunk hardlockkal forgalmazott Oracle adatbázis-kezelő, az Autocad, valamint a PCAD védelem nélküli verziója. S nem fél attól, hogy nem tudja eladni ezeket a drága programokat. Amit elmondott, a jó kereskedő filozófiáját tükrözi:

Minden amerikai kereskedő pánikba esett, amikor a Novell kijött a kulcskártya nélküli új hálózati szoftverével. Legnagyobb megdöbbenésünkre nem csökkentek az eladások. Az Autocad és az Oracle védelem nélküli verziójánál sem visszaesést, inkább a forgalom növekedését tapasztaltuk! Ennek oka az, hogy csak a bejegyzett felhasználó kap vevőszolgálati tanácsadást, kedvezményes árú új verziót. Ezek a szoftverek nagyok, bonyolultak. Ládányi dokumentáció kell használatukhoz. Sokszor többe kerül csak a leírást lemásolni, mint több példányban megvenni a programot! Így bár egy rendszer korlátlan számban telepíthető, egy közepes cég is legalább három-négy darabot rendel belőle.

Akik másolják a programokat, azok általában szerény anyagi lehetőségekkel rendelkezők, mint például a diákok. A nagy cégek oktatási célra jelképes áron biztosítják a programokat teljes szolgáltatással. Másrészt az így okozott bevételkiesés jelentéktelen, ugyanis nem végeznek vele termelő munkát. Esetleg csak tanulmányozni akarják a rendszer képességeit. Lehet, hogy meg sem tetszik nekik, így nem is potenciális vevők. A korábbi verziók szinte szabadszoftverként való terjesztése pedig végképp nem zavaró. Ez inkább növeli a piaci lehetőségeket. Hiszen aki egy ilyen programon megismerte egyik vagy másik rendszert, az később munkahelyén döntési helyzetbe kerülve épp ennek a beszerzését fogja szorgalmazni. Azért tartunk jó kapcsolatot a sajtóval is, rendszeresen küldünk tiszteletpéldányokat, hogy a tesztekhez ne kelljen ellopniuk a programokat.

Arra a kérdésre, hogy Amerikában nincs-e másolásvédelem, a következőket válaszolta: *Olyan programoknál fel-felbukkan, amelyeket az abszolút kisembernek szánnak. Olyannak, akinek semmi esélye sincs arra, hogy fellépjen a forgalmazóval szemben. Ezek a programok szinte mind játékprogramok, minimális mértékben igen olcsó könyvelési programok. De ezek is nagyon gyorsan eladhatatlanokká válnak. Ezeknek a rétegek-*

nek az igényeit egyre jobban kielégítik a szabadszoftverek, a használók önkéntes befizetéseiből finanszírozott (user supported) programok, melyek ugyan rontják a mi üzletünket, de amíg nem jelennek meg közöttük valóban mamutcégek, addig nincs félnivalónk tőlük. Komolyabb vevők számára másolásvédett programot eladni sem nálunk, sem Kanadában nem lehet. A védelem nélküli programrendszerek főleg Kelet-Európába való félig legális szállítása egyre nagyobb üzlet az amerikai és angol szoftverforgalmazó cégeknek. Így a kelet-európai cégek vagy követik az USA forgalmazási stratégiáját, vagy csak a mi pénztárcánk fog vastagodni. Már többen megoldottuk azt is, hogy ezeknek az amerikai kópiáknak a vásárlói is megkaphassák a kedvezményes frissítés (upgrade) és vevőszolgálat lehetőségét is.

Jogos érdekeiket a forgalmazók Magyarországon is védhetik tisztességes módon! Igazolja ezt a Cédrus Rt. is, amely védelem nélkül forgalmazott programokkal érte el eddigi eredményeit. Csak figyelembe kell venni azt, hogy a vevő tisztességes áron, tisztességes árut, tisztességes szolgáltatásokat kíván.

A *tisztességes ár* fogalmát a nyugati országokban nagyon jól behatárolták: arányban áll annak a személynek a jövedelmével, munkabérével, akinek a munkáját a program helyettesíteni hivatott. Ezt az árat nevezik „lopáshatárnak" is. Ugyanakkor az oktatási intézmények nem termelő célra ennek az üzleti forgalmi árnak a töredékéért kapnak jogos példányokat. Vagy például a szaksajtó képviselői, a szaklapok szerkesztőségei is kaphatnak ingyenes, de jogosított és működő, azaz nem demonstrációra lebutított programokat. Ilyesmi hazánkban ritkaságszámba megy, nyugati cégek készségesebben adnak tesztpéldányokat.

A programrendszerek illegális forgalmazás elleni védelmének egyik legfőbb eleme tehát a feladathoz és a potenciális vevő pénztárcájához arányított ár. Ha ez megfelelő, utána még tisztességesen át kell segíteni a felhasználót az új program bevezetésének gondjain is. Ilyenkor szinte természetes, hogy bármilyen problémájával hozzánk fog fordulni.

A program viszont a saját illetékességi körén belüli használat során semmiben nem korlátozhatja a felhasználót. Így a programról korlátlan számban egyszerű eszközökkel készíthessen másolatot és azt a saját területén belül egyszerre annyi gépre tehesse fel, amennyire kell. Erre az amerikai kereskedők bevezették azt az átalánydíjat, amikor nem gépre, hanem egy cég meghatározott telephelyére adják el a használati jogot. Az ár mérsékelten az ottani gépszámtól is függ.

Három olyan eszköz létezik a gyakorlatban, amellyel a felhasználók tiltakozása nélkül a kereskedők is ellenőrizhetik az eladott példányok

jogosságát. Ebből egyik a beégetett kópiaszám. A másik lehetőség, hogy a program az első installálás után csak a külön kártyán megadott jelszó begépelése után indul el, s minden újrainstallálás vagy áttelepítés után kéri a jelszót. Végül harmadik lehetőség a névre és cégre szóló dedikáció, amelyet vagy az első installáláskor ír vissza a lemezre, vagy pedig a kereskedő eladáskor így jogosítja a programot.

A közhiedelemmel ellentétben a magyar helyzet jobb, mint például az osztrák vagy a németországi, ahol pedig fejlettebb a számítógépes kultúra. A védelem nélküli programoknak a mienknél nagyobb a feketepiaci forgalma, és vannak olyanok is, akik ezeket a programokat pénzért árulják. Ugyanakkor nagyobb a védetten forgalmazott szoftverek aránya is, különösen a német nyelvű változatoké. Ezeket külön erre a célra szakosodott cégek szabadítják meg (szoftverinstallálás címén) a védelemtől. Illetve aki nem ragaszkodik a német verzióhoz, minden nagyobb szoftveres céggel meghozathatja az amerikai kópiát védelem nélkül.

A számítástechnikai fejlődés, az adatbiztonság előtérbe kerülése egyre inkább visszaszorítja a másolásvédelem alkalmazását. Ez ellen a felhasználók nyílt vagy hallgatólagos szövetségekbe tömörülnek, még olyan áron is, hogy békét kötnek a programok feltörésével hivatásszerűen foglalkozó csoportokkal. Egyúttal egyre inkább felismerik, hogy a másolásvédelem és a hozzáférés-védelem (azaz az adatokhoz való jogosulatlan hozzáférés elleni védelem) nem ugyanaz a terület. Míg a másolásvédelem a vírusíráshoz és a számítógépes szabotázshoz hasonlóan a számítógépes terrorizmus egyik válfaja, a másik terület tökéletesítése a számítástechnika fejlődésének elemi érdeke.

HADIBACIK

Célpont a vezérlőrendszer

Az arab-öböli háború megélénkítette a számítógépes vírusok piacát. A vírusírás súlypontja a háború alatt ismét a Közel-Keletre tolódott át. Megjelent a Szaddam vírussorozat, amely az eddigi ismeretek alapján a Stupid víruscsalád alapelemeiből építkezik. A sajtóban nyíltan felvetődött, hogy a szövetségesek miért nem indítják el az iraki számítástechnikai rendszerekbe beépített időzített szoftveres pokolgépeket.

Már régen tudunk hadviselési vagy éppen technológia-ellenőrzési céllal folyó víruskutatásokról. E téren természetesen csak nyomokból, utalásokból és szakmai pletykákból lehet információhoz jutni. Olvastunk egy *Israeli defence* nevű, kifejezetten hadműveleti célra kifejlesztett vírusról, amely eddig szerencsére egy-két szakirodalmi utaláson kívül nem bukkant fel. Hamburgban viszont kiszabadult a Fish-6 (Hal) és a Whale (Bálna) vírus, amelyek folyamatos beépített mutációs lehetőségeikkel és a hagyományos módszerekkel észrevehetetlen terjedésükkel ideális hadviselési vírusok.

Az USA-ban már az ötvenes évek végén, később Németországban a Bundeswehr keretén belül is foglalkoztak olyan vírusok előállításával, amelyek megbéníthatják az ellenfél gépét. Ezeknek a vírusoknak a fejlesztői a nagygépes rendszerek, illetve a nagyon bonyolult vezérlőrendszerek ellophatatlanságát, valamint az ellenség rendszerének megbénítását tűzték ki célul.

Arról, hogy a volt szocialista országokban folytak volna ilyen kutatások, egészen 1990 októberéig nem sokat tudtunk. Akkor járta be a CWI információs hálózatán a hír a világot: Bulgáriában több mint ötven igen fejlett számítógépvírust engedtek szabadon. Egy Szovjetunióban végzett, jelenleg már Magyarországon élő programozó pedig arról számolt be, hogy Leningrádban a flotta egyik számítástechnikai intézete kifejezetten hardver tönkretételére alkalmas vírusok fejlesztésén dolgozott. Olyan, nem DOS-formátumú lemezeket készítettek, amelyek a merevlemezt és a gép elektronikáját 10–20 másodperc alatt helyreállíthatatlanul tönkreteszik. (Eredetileg a fenyegetett helyzetben lévő berendezések gyors „önpusztítására" dolgozták ki.) A szabotázs céljaira alkalmas

vírusok fejlesztését valószínűsíti, hogy a Szovjetunióból egyre több, normál környezetben terjedő, igen fejlett vírus indult el útjára.

A szovjetek, románok, bolgárok és mongolok számítógépei jelenleg is hemzsegnek a programvírusoktól. Hasonló magyar tervekről csak nehezen kontrollálható információink vannak. Nyilván másolásvédelem ürügyén napjainkig folynak ilyen kutatások; magányos farkasok vagy intézeti kutatócsoportok foglalkoznak vírusok fejlesztésével, átírásával. Ezek azonban egyes cégek vagy egyének magánkezdeményezései, államilag nem támogatják őket.

Főleg azok élesztik a vírusírás tüzét, akik a konkurencia termékeit és saját munkájuk korábbi verzióit akarják ezzel a módszerrel kiiktatni. A másolásvédelembe rejtett vírus vagy a trójai romboló funkció legális kereskedelmi forgalmát a felhasználók érdekszövetségei egyre inkább visszaszorítják.

Szinte elképzelhetetlen, mennyi rosszat tud szülni mindaz, amit az amatőr és hivatásos laboratóriumok a vírusírásban eddig is produkáltak. Emiatt zárolják mindenütt a vírusokra vonatkozó érdemi információkat: teljes víruskód nem jelenik meg. Ha valahogy mégis kitudódik egy-egy vírus forráskódja, akkor felelőtlen programozók hamarosan tucatszám szállítják annak átbarkácsolt változatait, s ezeket a megszokott antivírus-programok nem ismerik.

Az izraeli Héber Egyetem számítógépl-aboratóriumának vezető munkatársa, Ysrael Radai tudna legtöbbet beszélni arról, hogy mennyi vírust írnak és próbálnak gépeikbe terrorista céllal bejuttatni. Például a vírustalanítók között a *Jerusalem* vagy *Suriv* sorozatnak ismert vírusok is bizonyíthatóan számítógépes szabotázs eredményeként születtek.

A számítógépes vírusok ugyanolyan veszélyesek, mint a biológiai fegyverként alkalmazott vírusok. Az amerikai *Computer Professionals for Social Responsibility* (hivatásos számítógép-szakértők a társadalmi felelősségért) társaság ezért foglalt állást a számítógépvírusok hadrafogása ellen.

Franz Feldmann ismert matematikus és filozófus, aki az USA-ban több adabiztonsággal foglalkozó kormányszerv tanácsadója, szintén kifejtette, hogy nem hisz a vírusfegyver használhatóságában. Katonai vezetők is egyre inkább hajlanak arra, hogy teljesen mellőzzék, és csakis a passzív védekezésre szorítsák a kutatásokat, mivel senki nem tudhatja, hogy egy ilyen fegyver a végén kit és hogyan fog eltalálni.

A számítógépes rendszerek ellenőrzése már korán olyan technikák kidolgozását is jelentette, amelyekkel az eladott berendezések működése vezérelhető, esetleg megbénítható. A korszerű számítógép alkalmas arra, hogy olyan többlet alkatrészeket helyezzenek el benne, amelyek ré-

vén a gép helyzete műholdról lekérdezhető, illetve a berendezéseknek nagy távolságból is utasítás adható. Elgondolkodtató volt a Magyar Tudományos Akadémia CDC4400 gépének esete, melyet a kinti ellenőrök a vételi szerződés alapján bizonyos időközönként felkerestek és kivallattak. (Azóta külön „szakterület" lett az ilyen felesleges alkatrészek kibányászása.)

A nagyteljesítményű amerikai célszámítógépekhez, közöttük az IBM nagygépeihez valószínűleg már elég korán létezett ilyen helymeghatározó eszköz. (Bizonyítva a szakirodalmi utalások alapján ez csak a Cray kategóriájú berendezéseknél látszik). Hasonló gyanú érte a szakma részéről az amerikai államigazgatásban alapgépként használt Alfa Micro megamini gépet, melyet a programlopás és a szabotázs megakadályozására tudatosan inkompatibilissá fejlesztettek ki, mind programjaiban, mind operációs rendszerében. (Ezt ugyan a cég soha nem ismerte be...) Nyugat-európai hacker körökben a *Medusa* tervezőrendszert hardverkulcson keresztül kézben tarthatónak vélik. Nem véletlenül indultak meg a kutatások igen jelentős tőkével olyan integrált áramköri tokok kialakítására, amelyek fizikai sérülés esetén vagy egyéb utasításra tönkreteszik a bennük lévő áramköri lapkát.

A hatvanas években több fejlődő ország jutott hozzá olyan számítástechnikai rendszerhez, amelynek operációs rendszere egy adott időpont után, illetve telefonvonalon vagy rádión kapott jel hatására megszüntette működését. A valóban nagy teljesítményű amerikai számítóközpontokat pedig eleve úgy építik, hogy a berendezések válságszituációban vagy illegális behatolási kísérlet gyanúja esetén önmagukat „megcsonkítják".

Ezek a hírek keletre is eljutottak. S bár a keleti tömbben is működtek korszerű nyugati berendezések, mindegyik ország saját számítástechnikai rendszer kialakítását tűzte ki célul. Így jött létre a lidérces ESZR-program, de ennek köszönhettük az NDK–s Robotron művek és az akkor hozzá kapcsolódó katonai csoport több látványos ipari kémbotrányát és egyéb csínytevéseit is. Idővel az is kiderült: alig-alig átírva használták a lopott operációs rendszert.

Magunk is tapasztalhattuk sok amerikai eredetű integrált áramkör, többek között modem IC-k megfejtésekor, hogy ezek a csipek sok olyan – látszólag értelmetlen – funkciót tudtak, amit a nyilvánosan elérhető adatlapok nem közöltek. A jelenlegi számítógépek alkatrészeinél is előfordul ilyesmi, ugyanis egyazon áramköri lapkát több típusszámon, eltérő specifikációkkal forgalmazzák, vagy éppen kettős hasznosítású áramkörként alkalmazzák. Ilyenkor szorgos mérések szükségesek a minden bizonnyal használható többletképességek kiderítésére. Bizo-

nyos AM (modemben alkalmazható) IC-k a fenntartott (unused vagy reserved) lábakat megfelelően bekötve ismerik a katonai rendszerekben szokásos hibajavító kódolást, ami nagyon jól jöhet például a magyar vonalakon is.

A szocialista országokban előírás volt, hogy bizalmas információk titkosítására csak olyan rendszerek alkalmazhatók, amelyeket az utolsó „szögig" ezekben az országokban fejlesztettek ki. Az Amerikai Nemzetbiztonsági Hivatal (NSA) ugyanis, amely büszke volt arra, hogy minden elektronikus kommunikációt lehallgat, nem engedélyezte olyan technológiák és készülékek forgalmazását, amelyeket nem tudott rövid időn belül megfejteni. Ez rejlik a DES titkosítási algoritmussal dolgozó szoftverek exporttilalma mögött, s nekünk is ezért kell egy könnyen megfejthető PC-Secure algoritmusú nemzetközi PC Tools verzióval megelégednünk. Amikor megjelent a Motorola egy viszonylag könnyen elérhető csipje, amely hasonló hardveres titkosítási algoritmussal dolgozott, éltünk a gyanúval, hogy ez a csip valahogy a titkosítás megfejtéséhez szükséges kulcsot is kiadja.

Érzékeny rendszerek védelmére több gyakorlat született. A múlt évben védte meg hadtudományi egyetemi doktori disszertációját a Zrínyi Miklós Katonai Akadémián *Kötél Gyula* adjunktus, aki a strukturált programozásban, az egyes részfeladatok más-más programozó csoportoknak való kiadásában látja az illegális funkciók elkerülésének lehetőségét. Nyugaton például banki rendszert csak dokumentált forrásállományokkal vásárolnak. Ezt azután független cégekkel ellenőriztetik, és a kereskedelemben kapható fordítóprogramok véletlenszerűen kiválasztott példányaival fordítják. Ha ezek a verziók azonosak, akkor a kereskedelmi szoftver sem volt manipulált. Máshol adatállományaiban és felhasználói interfészében teljesen azonos programokat dolgoztatnak ki egymásról mit sem tudó csoportokkal, és utána azokat véletlenszerűen cserélgetik.

A számítógépek elleni szabotázsok, (különösen ha fejlesztésüket államilag is támogatják) roppant veszélyesek az informatikai társadalmakra. Hát még ha ebbe a szoftverházak és a programozók is bekapcsolódnak. Gondoljunk csak bele! Egy sikerprogram, mondjuk egy operációs rendszer kódjában elrejtenek egy alvó bombát, egy becsomagolt, önmagát titkosító vírust. Ez szép lassan elterjed az egész világon. S ha elérkezik (akár több esztendő múltán) a megadott nap, akkor az addig szendergő rész felébred és a vírusok elindulnak romboló útjukra. És sajnos nem is tudhatjuk, hogy hány ilyen bomba ketyeg.

Előbb-utóbb ezt a kérdést is olyan egyezményekkel kell szabályozni, mint amilyenek jelenleg a vegyifegyverekkel kapcsolatos kísérleteket, a

biológiai fegyverek előállítását, a génmanipulációkat tiltják. Ez a világ érdeke is.

Meghökkentő meglepetésben lehetett része annak a számítástechnikai szakembernek, aki 1990-ben elolvasta a Pentagon *Small Business and Innovation Research Program* kiadványát. Ez a hivatalos, minden évben megjelenő könyv azokat a katonai tématerveket ismerteti, amelyeket kisebb magáncégeknek fejlesztésre vagy koncepció kidolgozására kiadnak. Az amerikai minisztérium bürokratái nem gondolták volna, hogy az ebben megjelent egyik kiírás mekkora vihart és felháborodást fog kelteni számítástechnikai körökben.

A kiadvány 45. oldalán található *Computer Virus Electronic Counter Measure* pályázat ötvenezer dollárt ígér annak a cégnek vagy magányos fejlesztőnek, aki hadi célokra alkalmas vírussal vagy annak fejlesztési koncepciójával jelentkezik a minisztérium illetékeseinél. Ha megnyerte a stratégák tetszését, akkor félmillió dolláros megrendelés vár rá, hogy sorozatérettre fejlessze pusztító termékét.

Az ötlet háttere kézenfekvő. A fegyverrendszereket világszerte egységes irányítási rendszerbe kötött számítógépek vezérlik. Ez így van az USA-ban, Magyarországon, Szovjetunióban, Irakban, de kisebb méretekben ugyanezt a technikát alkalmazzák a terroristák is. A cél az lenne, hogy az ellenség kommunikációs rendszerébe bekapcsolódva a rádiós adatátviteli vonalakon keresztül csempésszék be akár egyetlen helyre is a vírust, amely ott elszaporodik és tönkreteszi a rendszert. (Ha már előre beépítették, akkor hasonló módon rádión keresztül lehet aktivizálni.) Ehhez tehát elsősorban olyan vírusokat kellene kifejleszteni, amelyek előbb manipulálják az ellenfél adatait, így a rakéták, tankok közömbös területekre mennének – s ehhez operatív hírszerzési eszközökkel az ellenfél számítógépének még csak a közelébe sem kellene kerülni.

Csakhogy... A vírus nem szelektál, hogy ellenséges vagy saját gépbe, katonai vagy polgári célú a berendezésbe jutott be, hanem mindegyiket tönkreteszi. Így valóságos számítástechnikai katasztrófát indíthatnak el az egész világon. Mi történne, ha az atomerőművek reaktorainak, az egyes országok energetikai rendszereinek vezérlőgépei egyik pillanatról a másikra megőrülnének?! Igen nagy a bumeráng-effektus veszélye. Egy kellően jól elkészített vírus észrevétlenül elszaporodhat az őt előállító cégnél, majd a vele összekötött saját hálózatban is. S amikor ezek a vírusok elkezdenek dolgozni, szinte semmi esély a számítástechnikai rendszer megmentésére. Már több olyan vírust ismerünk (a Whale-Fish-6 rendszer, a magyar Phantom és Polimer, néhány orosz és bolgár „csoda"), ami ennek a feltételnek némi barkácsolás után megfelel.

A vírusok ellen küzdő szakértői csoporokat tehát mélyen felháborította a Pentagon által kiírt terv. Aláaknázza eddigi erőfeszítéseiket. De a felhasználókat is megfélemlítheti, teljes bizonytalanságban tarthatja a jelenlegi programforgalmazási rendszerek és használati szokások következtében. Hiszen már vannak olyan, kifejezetten zsaroló forgalmazási módszerek, vírusok, trójai programok, amelyeket inkább szabotázsprogramnak nevezhetünk.

A vírusok elleni küzdelem komoly erőforrásokat köt le a víruselhárító cégeknél és a ráutalt vállalatoknál. A New Yorkban székelő *Forst & Sullivan* csak a kisebb amerikai vállalatok vírus elleni védekezésre fordított pénzét egymillió dollárra becsüli, és szinte minden cég szenvedett már kisebb-nagyobb kárt a vírusoktól.

A pályázati kiírásra a számítástechnikai biztonságtechnikával foglalkozó egyik amerikai cég, az *Information System Security* biztonságtechnikai tanácsadója, William Murray szkeptikusan jelentette ki: „Az amerikai hadsereg használja hadi célokra a világon a legtöbb számítógépet. Fordítanák inkább arra a katonák a félmilliót, hogy kideríttsék, milyen vírusaik vannak már most is, tudtukon kívül, a rendszerben! Ha nagyon ugrálnak ezután is, hogy vírus kell nekik, hát menjenek be egy floppyval egy kisebb szoftverboltba, s aztán örüljenek az azon talált leleteiknek..."

Jürgen Ettelkofer, a Neubibergben működő *Bundeswehrhochschule* számítástechnikai intézetnek a vezetője szerint az amerikaiaknak kell tudniuk, mit csinálnak, őket ez nem érdekli. A vírus kereskedelmi cikké vált. Hozzáállása a tipikus katonai mentalitás: majd vigyáznak arra, hogy ne legyen baj belőle! Erre rácáfol a hadi célú vírusok minden kritériumának megfelelő Whale–Fish-6 vírusrendszer elszabadulása, amely világszerte egyre több gondot okoz.

A német Bundestag SPD frakcióvezetője, Michael Catehausen, pártpolitikus volta ellenére tisztán látja a vírusfejlesztő stratégia veszélyeit. Szerinte az USA védelmi minisztériumának támogatásával kifejlesztett számítógépes vírusfegyver mindenki számára világossá tenné: egy ilyen szabotázsprogram nemcsak a katonai, de a gazdasági háborúk megvívására is ideális fegyver lenne. Ha az emberek ennek hatására fokozzák a biztonsági intézkedéseket, akkor a fenyegetés elérte célját. Viszont tart tőle, hogy elsősorban a terroristák és a bűnözők ismerik fel és alkalmazzák mind többen a vírusok fegyver jellegét, ami a vírusvédelem, vírusírás és a másolásvédelem további eszkalációját jelentené. Az eddig marginális problémát okozó vírusok rövid időn belül a számítástechnika létének vagy nemlétének kulcsává válnak.

Werner Schmidt, a német Alkotmányvédő Hivatal mellett működő

szövetségi adatvédelmi biztos információtechnikai referense a jelenlegi vírusgondokat elsősorban a programok cseréjében látja. Ugyanakkor felhívja a figyelmet, hogy a katonák vírusprogramozásra való ösztönzése számtalan tehetséges „fekete bárányt" fog oda vonzani. Esetleg egy másik titkosszolgálat tagjait, akik az amerikai rendszereket veszik célba. De akadhat olyan programozó is, aki végigcsinálván a programot, a végén kijelenti: nem jutott eredményre. Ez ugyebár nem ellenőrizhető. S utána munkájának eredményét fejben kiviheti és eladhatja bármilyen terrorista célra.

Egy biztos: jelen tudásunkkal és technikai eszközeinkkel a Pentagon pályázatban leírt feltételeknek megfelelő vírus elkészíthető. Sőt ez a technika és technológia – bizonyos jelek szerint – mintegy 10–15 éve létezik is. Csupán azt nem lehetett megvalósítani, hogy ezek a vírusok szelektívek legyenek. (Egy német antivírus-szakértő szerint emiatt fordult el az informatikai szakemberek nagy része az SDI fejlesztő programoktól.)

A vírusokkal vívott bármilyen akciónak egyetlen hatalmas stratégiai előnye van a terroristák és a programforgalmazók szemében. A vírus írója, terjesztője nem bizonyítható. Sok esetben még a rombolás után is csak valószínűsíthető. A világban kialakult forgalmazási rendszer és gyakorlat kedvez a szelektív célbajuttatási módszereknek, segítségével elérhető, hogy egy vírusrendszer *kiindulásul* egy adott régiót fertőzzön meg. Az már egészen más kérdés, hogy utána a vírus nagyon gyorsan és kontrollálatlanul elterjed az egész világon.

Sokan azt hiszik, hogy a legálisan kapható, jogos szoftverek minden országban egyformák. Valójában igen jelentős eltérések vannak a különböző régiókban forgalmazásra szánt programok egyazon verziói között. Ennek csak egyik oka a technológiai transzfer korlátozása. A PC Tools és a Norton Utilities amerikai és európai verziója a DES titkosító algoritmusra vonatkozó exporttilalom miatt eltérő. Az már inkább piacpolitikai okokkal magyarázható, hogy az amerikai verzió az, amelyik programozástechnikailag a legnagyobb teljesítményre képes! (A Norton Commander amerikai kiadásában például egy könyvtárban nem 256 a maximális állománymennyiség, hanem ennek kétszerese, amit az újabb DOS-verziók is mind ismernek.) A többi korlátozást a maximális extraprofit elérése éppúgy motiválhatja, mint a katonai meggondolás.

Ismerkedjünk meg most azzal, hogy milyen változatok léteznek egy nemzetközi érdeklődésre számot tartó program megjelenésekor!

Az USA-ban kerül forgalomba az *„inside in the USA & Canada"* (csak amerikai-kanadai használatra) változat. Ez az alapverzió a legjobb teljesítményű. Csak sorszám vagy névre dedikáló jelzés, esetleg jelszavas indítás van rajta, egyéb hozzáférés-védelem vagy másolásvédelem az

esetek 95%-ában nincs. Maximális képességei mellett az egyetlen korlát: betűkészlete szigorúan ékezetmentes, képernyő- és billentyűzetmeghajtói (általában) nem tudják kezelni a kiterjesztett ASCII-kódkészletet. Így biztosítják, hogy az angolszász nyelvterületen kívül még a kalózmásolatokat se lehessen korrektül használni. A szoftvereket a kibocsátó cég az USA és Kanada területén kívül nem ismeri el jogosnak, szürke kópiának tekinti azokat, így vevőszolgálatot és frissítést sem ad hozzá. Néhány esetben ráfizetéssel hajlandó kicserélni a nemzetközi verzióra, amit a szoftver dobozán lévő *„international"* felirat jelez.

A nemzetközi verzió a forgalmazó által bármely országban jogosnak elismert példány. Másolásvédelmet tartalmazhat, de ezek a védelmek a műszaki hibáktól eltekintve csak igen ritkán büntetnek. E prgramok kevés kivételtől eltekintve (a kivételek közé tartozik az Autocad) egészükben és részeikben kompatibilisek az amerikai kiadással. Eltérések lehetnek abban, hogy az amerikai kiadás egyes funkciói hiányoznak (például a DES algoritmusos titkosítás a PKZIP-ből és a Norton Utilitiesből), korlátozták a hálózat adatátviteli sebességét vagy a tartozék meghajtóprogramok képességeit („kék" Novell), vagy a képernyőmeghajtó és a billentyűzetmeghajtó által kezelhető karakterkészlet tér el annyiban, hogy ezek a teljes extended ASCII karakterkészletet használhatják. (Ebben van eltérés például a PageMaker 4.0 amerikai és nemzetközi verziója között.)

Vannak ezután a szoftvereknek regionális változatai. Ezeket például a kelet-európai országokban való forgalmazásra (és esetleg Ausztriába) szánják. Itt a legfőbb eltérés a másolásvédelem szokott lenni (például a Carmel Software TNT antivírus programjánál). Az itt alkalmazott másolásvédelmeknél már nem törődnek azzal, hogy ne ártsanak a felhasználónak. Az alkalmazott védelem attól függ, mennyire ítéli bebizonyíthatónak csibészségeit a forgalmazó cég.

Végül a nemzeti változatok esetében gyakorlatilag mindent szabad, amit a forgalmazó szükségesnek lát. Néhány esetben ez csak a másolásvédelemben tér el a nem védett nemzetközi verziótól, néha viszont még adatszinten sem kompatibilis vele, ha a forgalmazó vélt piaci érdekei úgy kívánják. Ilyenkor a jogos példányt a (szokványosan angol parancsnyelvű) nemzetközi verzió megvásárlásával lehet beszerezni, mert arra nem szokás értékesítési monopoljogot adni.

Ez a szoftverforgalmazási szokásjog és stratégia kedvezően kihasználható a vírusháborúk céjaira is. Elegendő egy adott régió kópiáiba csak vírus egyik felét beépíteni, míg az aktiváláshoz szükséges másik részt például egy népszerűségre számot tartó szabadszoftver hordozhatja. Az eredményt könnyen elképzelhetjük.

Ilyen vírushadviselés jeleit már láthattuk is a nemzetközi szoftverforgalomban. Mi másnak, ha nem ennek lehet értékelni a tajvani gépekkel és gyári szoftverekkel terjesztett Invader vírust? Az Autocad ellen írt Anticad vírus pedig tipikus visszavágás az Autodesknek, amiért fellépett a nemzetközi verzió korlátozásait kijátszó, másolásvédelem nélküli, 27 dolláros illegális Autocad-verzió forgalmazói ellen.

Ez a differenciált forgalmazás az esetleges gyártási gondatlanságok kiküszöbölésénél is jól jöhet. A WordPerfect egy évvel ezelőtt véletlenül vírusfertőzött sorozatot hozott forgalomba Ausztráliában. A csere és a károk elhárítása sokba került, a cég viszont nemcsak megmenekült a csődtől, de Európába és az USA-ba a hír olyan későn jutott el, hogy az már nem csökkentette piaci forgalmukat.

AIDS-TÁJÉKOZTATÓ LEMEZ

Informatikai merénylet trójai módra

1989 decemberében, néhány napos eltéréssel többezer „informatikai bombát" postáztak mágneslemezen. A küldemény az *AIDS Information Disk* volt, s a címzetteknek mint AIDS szakértői segédeszközt kínálták. A CW Communication, a szétküldési címlista jóhiszemű szolgáltatója csak utólag tudta meg, milyen akciónak tették részesévé, akkor viszont azonnal vállalta Jim Bates szakértő munkájának finanszírozását, hogy minél előbb kidolgozzák az ellenanyagot... – Ugyanezen sorokkal vezettük be első víruskönyvünkben a számítógépes rendszerek ellen intézett különös támadás leírását. Azóta kézre került a merénylet tettese, sőt sikerült egy példányt beszereznünk az inkriminált lemezből, így ellenőrizhettük a Jim Bates dokumentációjára alapozott korábbi állításainkat. A történetet az időközben immár pontosított adatokkal helyesbítve tesszük közzé.

Az USA szövetségi nyomozóirodájának, az angol Scotland Yardnak és jó néhány titkosszolgálatnak köszönhetően került elő 1990. február 2-án az az ember, aki e nagyszabású akció kitervelője és egyik programozója volt: Joseph L. Popp (akkor 39 éves). Felsőfokú tanulmányait az Ohio State University zoológia szakának elvégzése után a Harvard Egyetemen folytatta. Ezután dolgozott az UNICEF-nél, majd egy másik ENSZ-szervezetnél, a WHO-nál. 1989 januárjában kezdte meg vírusforgalmazó tevékenységének előkészítését, bejelentette Panamában a PC Cyborg céget. Majdnem egy évig dolgozott tervén és programrendszerén.

Az informatikai merénylet kivitelezése profi munkára vall. Az AIDS-kutatással foglalkozó intézetekhez Angliában vagy Panamában feladott mágneslemezt szállított a posta. A kísérőlevél szerint a csomag egy AIDS szakértői rendszer demonstrációs változata. Ez azonban csak az álca, a *faló* volt. Ez a trójai program csak akkor nem veszélyes a felhasználóra, ha merevlemez nélküli gépen, floppyról futtatja.

A rendőrségi nyomozócsoport adatai szerint ezt a trójai programot tartalmazó floppyt legalább 26 ezer(!) cégnek küldték el 1989 decemberében és 1990 januárjában. A címlistát a magazinok és újságok címlistáit készítő és címkéket nyomtató *CW Communications* cégtől szerezték be. A *PC Business Word* sajtószolgálatának nevében többek közt azok-

nak a listáját kérték, akik jelen voltak a WHO 1988-as stockholmi AIDS-konferenciáján. A címlistákat és a címkéket Londonba, a Bond Streetre küldették, arra hivatkozva, hogy az ottani *Ketema and Associates* cég Nigériából származó kereskedelmi szoftvercsomagot szeretne postázni. Mindezért fizettek is, a Chase Manhattan bankon keresztül, 158 ezer dollárt!

A vírusriadót a skandináv AIDS-kutatási centrumként is üzemelő Roalagstull kórház indította el, ahol a nyilvántartás adatainak nagy része megsemmisült. Franciaországban a küldemények leginkább a WHO párizsi központjának adatrendszerében pusztítottak. 1989 decemberében a Londoni Tőzsdén, az új-zélandi, ausztráliai bankokban és a brit Honvédelmi Minisztériumban okozott hatalmas károkat. Több tucat lemez érkezett a WHO genfi központjába is, különböző kutatók nevére. Az Európában azonosított küldemények közös vonása, hogy feladóként a PC Cyborg panamai cég szerepelt, 1989. december 8. és 12. közötti bélyegzéssel, London SW1, SW7, W1 körzeteinek postahivatalaiból. A lista szerint hazánkba legalább három lemezt küldtek, a Haematológiai Intézetbe, a János Kórházba és a KFKI-ba címezve. Ezek sorsáról nincs hír, vagy eltűntek – ezúttal szerencsére – a postán, vagy leformázva nyerslemezként használták fel azokat. Így teljesen veszélytelenek.

Sok ellentmondó hír kering ennek a trójai programnak a terjedéséről, természetéről és az általa okozott károkról. Mindenesetre szakszerűen, nagy tudással előkészített terrorista akcióról volt szó.

Először is, maga a floppylemez nem tartalmazza a vírust, így nincs is mit keresni rajta. Amikor az installáló program elindít egy számlálót, létrehoz egy látszólag véletlen számot a rendszerindítások számából, majd aktivizálódik a trójai program, és szétroncsolja az adatokat, valamint a programokat a merevlemezen. Hogy megismerhessük a valódi veszélyt, tisztán kell látni, hogyan élesíti be magát ez a rendszer.

Miután a Trojan AIDS program installálta magát és kényelmesen elhelyezkedett a merevlemez méhében (az egyszerű felhasználó számára elérhetetlenül), a program figyelőállásba helyezkedik. Az első fázisban egy számlálórutinnal figyeli, hányadik alkalommal indítjuk újra a rendszert. Amikor letelt a számunkra engedélyezett *kilencven* újraindítás, aktivizálódik a második, a romboló rutin: egy sajátos algoritmus alapján titkosítja a merevlemezen az állományokat és a könyvtárakat, hihetetlen káoszt okozva.

Ha a program befurakodott a merevlemezre és nincs megfelelő killerünk, a kezdeti időben adatainkat egy ügyes fogással még hiánytalanul visszanyerhetjük. Ha ugyanis *nem* a fertőzött merevlemezről indítunk rendszert, akkor nem aktivizálódik a vírusrutin. Ekkor adatainkat za-

vartalanul kimenthetjük, utána pedig a merevlemezen alacsony szintű formázást kell végeznünk. Az így letisztított lemezre már installálhatunk egy tiszta rendszert, s végül visszatölthetjük adatainkat.

Általános érvényű óvintézkedés, hogy amikor megveszünk egy programot, mindig két másolatot készítsünk róla, amelyek közül az egyiket használjuk, míg az eredetit és a másik másolatot mint könyvtári példányt elzárjuk. Vírusok támadása vagy bármilyen más állománypusztulás esetén így mindig van honnan hibátlanul visszatölteni a gépbe programjainkat. Bár így egy kicsit több floppyt fogunk felhasználni, de megéri. Másolásvédett szoftvert már csak ezért sem szabad vásárolnunk. A demólemezeket és az újonnan vásárolt vagy szerzett szoftvereket először olyan gépen kell futtatni, amelyen nincsenek kulcsfontosságú adatok, s csak a sikeres vizsga után engedjük azokat az éles számítástechnikai rendszerbe. A bootvírusok esetében adatmentő lehet a tiszta tartalék DOS rendszer, hiszen legalább az adatállományokat ki tudjuk másolni a merevlemezről.

A Trojan AIDS generáló rendszere is a programozástechnikában szokatlan, érdekes megoldásokat tartalmaz. A programot egy installációs rutin teszi fel a gépre. Kihasználja, hogy a könyvtárak és a futtatható programok neve a képernyőn számunkra láthatatlan karakter vagy DOS-terminátor is lehet, ha azt ügyesen alkalmazzák. A Cyborg tagjai a 255-ös kódú karaktert nevezték ki erre a célra. Ez a „hi space" a „kemény szóköz" standard DOS eljárásban az állománynév végének és a kiterjesztés kezdetének a jele, így önmagában hagyományosan sem állománynévként, sem pedig könyvtárnévként nem használható. Itt mégis használják.

Most lássuk a folyamatot!

1. Az installáló program létrehoz egy rejtett AUTOEXEC.BAT állományt, ami maga is szokatlan. Hát még a tartalma!

```
CD \<ALT 255>

REM<ALT 255>
```

Az eredeti AUTOEXEC.BAT állományt átnevezi AUTO.BAT-ra.

2. A program létrehoz egy rejtett <ALT 255> karakterrel megnevezett alkönyvtárat, benne egy REM<ALT 255>.EXE nevű programmal.

Amikor ezt a rendszert indítjuk, csendesen számolja a rendszerindításokat. A következő fázisba csak 90 indítás után lép át a program. Ezek a rejtett alkönyvtárak persze törölhetők, és ezáltal meg tudjuk bénítani a programot, de ilyenkor még nem biztos az eredmény, ezért néhány hónapon át ellenőrzéseket szükséges végezni a lemezen,

nagyon figyelve minden szokatlanra, rejtett állományra és alkönyvtárra. Ezt azonban csak az első időszakban lehet megtenni!

Eddig senkinek sem sikerült visszafejtenie a kódot. Az installációs lemezen két, összesen mintegy 320 kilobájtos állomány tartalmazza a programrendszert, INSTALL.EXE és AIDS.EXE nevű állományok formájában. Az elfogott programozó vallomása megerősítette a visszafejtéssel foglalkozók sejtését, hogy az egész rendszert assemblyvel megspékelt QuickBasic programnyelven írták. A programrendszer csak installálás után tud működni. Minden más forrás és saját tapasztalatunk is egy szövegállományokkal tűzdelt tömör kódot jelez, míg a Vogel Verlag kiadásában megjelenő Viren Telex (90/2. szám) egy primitív „if then goto" technikával készült programra céloz. (A kettő, legalábbis részben, nem zárja ki egymást.)

A trójai funkciók nyomonkövetése hihetetlenül nehéz, mert a program sok-sok lépésben építi fel végleges struktúráját, egyéb rejtett és irracionális nevekkel számos furcsa alkönyvtárat és állományt hozva létre. A károk mérséklésében érdekelt szakemberek programozási eljárásokat próbáltak kidolgozni a folyamat követésére, mert a trójai programrendszer DOS-SHELL rutinokkal és magukat sokszor átíró és módosító segédprogramokkal teszi teljesen követhetetlenné a folyamatot.

A továbbiakban ismertetjük a rendszer működését, mint a számítógépes terrorizmus egyik „gyöngyszemét". Természetesen nem forráskód szinten, csak annyira, hogy megérthessük: sok olyan hely van egy gépen, ahova el lehet dugni egyet s mást.

Nyomdatechnikai okokból, illetve a nem láthatók láthatóvá tétele érdekében néhány számítógépes karakter helyett a könyvtári struktúrák jelzésére az alábbi jelzést fogjuk alkalmazni:

– — az ALT 255 (másképpen HEX FF) karakter helyett, amely a monitoron szóköznek mutatkozik, de valójában DOS-karakterként funkcionál.

@ — az aláhúzás karakter (ALT 95 vagy másképpen HEX 5F) helyett.

s — a szóköz karakter (ALT 32) helyett az állomány- és alkönyvtárnévben.

A trójai vírus mindegyik elkapott program esetében 90 rendszerindítás után lépett fel, de csak akkor, ha az INSTALL program szabályosan lefutott. Ami érthetetlen: hogyan biztosítja a program a hamis ellenőrzést (dozen verify), mielőtt a kód akár egyszer is végrehajtódott volna? Mert ezt még a program aktivizálódása, azaz az első rendszerindítás előtt megteszi.

A folyamat teljes lefutásához egy normál AT gépen mintegy 90 másodperc szükséges. A teljes installációs folyamat alatt egy referenciaszám látható a képernyőn. A leírás szerint, ha regisztrálja a programot, ez lesz a referenciaszáma. Vajon miért aktivizálódik számos kópia esetében a figyelmeztetés, előre felhívva a figyelmet a károkozásra, mielőtt még visszafordíthatatlanul elindulna a destruktív rendszer? Talán hiba van az install vagy a számláló rutinban? Amikor a program destruktív folyamata beindul, ismételten megmutatja a monitoron a referenciaszámot, világos utalással a számmal kapcsolatos összefüggésekre. Esetleg titkosítási és dekódolási folyamat során van szükség erre a betű-szám kombinációra. Például ilyen 12 jegyű referenciaszám volt az egyik példányon: A9738–1655603. Ez nem hozott létre átadó lemezt. Egy másik vizsgált példány kódja A935759–1048965, ez pedig létrehozott. Már tudjuk, valóban több változat került forgalomba a lemezből. Ezek dekódolási eljárása, ha nem fix kóddal, hanem a merevlemezről vett kóddal dolgozik a szoftver (Jim Bates programja pedig ilyen), mindegyik verziót levakarja a merevlemezről.

Amikor a küldeményt gyanútlan tulajdonosa behelyezi a gépébe, először is el kell indítania az installálási folyamatot. Ekkor jön létre a 255-ös ASCII karakter (azaz kemény szóköz) elnevezésű rejtett alkönyvtára, mégpedig a C: merevlemez gyökérkönyvtárából nyílóan. Az eredeti AUTOEXEC.BAT állományt átmásolja egy AUTO.BAT nevű állományba. Ennek első sorába elhelyez egy megjegyzést:

```
REM Use this file in place of AUTOEXEC.BAT for con-
venience
```

Azaz: az AUTOEXEC.BAT helyett „kényelmi okokból" ajánlja ezt az állományt használni. S itt van az első csavar. Ugyanis a program ezen álcázó állomány mellett létrehoz egy rejtett (hidden) AUTOEXEC.BAT állományt is. A gép pedig azt fogja végrehajtani!

```
echo off
C:
cd\#
rem# PLEASE USE THE auto.bat FILE INSTEAD OF
autoexec.bat FOR CONVENIENCE
auto.bat
```

A CD után, még ha a felhasználó megleli is ezt az állományt, egy olyan karaktert talál, amely nem jelenik meg a monitorán. Így valójában ez az ártalmatlannak tűnő program a következőket hajtja végre:

Belép a rejtett szóköz (hi space) karakterrel megnevezett alkönyvtárba, és ott lefuttatja a REM#.EXE állományt. Ennek az a feladata, hogy visszaszámoljon a beállított rendszerindításokból, s a számláló nulla állásánál indítsa a tönkretétel második fázisát. Amikor a véletlen számláló eléri a nullát, kitakarítja a merevlemezt, mégpedig alaposan. Ez már a vég, de előtte még sok minden történik. Az eredeti AUTOEXEC.BAT végrehajtódik, miután a program megfelelő része lefutott. Hiába keressük hagyományos módon, mert Hidden, System, Read-only attribútumot kapott. Létrehoz egy meglehetősen szokatlan könyvtári struktúrát is a C: meghajtó főkönyvtárából kiindulva. Említett jelölésrendszerünkkel ez így néz ki:

```
C:\###s###

C:\###s###\##s####

C:\###s###\##s####\####s##

C:\###s###\##s####\####s##\ERROR IN THE
```

Az Error nem rejtett könyvtár, de jól el van dugva szem elől. Utána építi fel saját állományait is, amelyek hasonlóan ötletesen vannak elnevezve:

```
@.ss@

@.s@

@.s@

@@@.ss@

@@@@.s@
```

Ezekből képződnek azok a nagyméretű összekutyult állományok, amelyek majd teljesen megtöltik a merevlemezt, amikor a pokolgép „felrobban". Az installálás során először megvizsgálja, hogy nem írásvédett rendszerrel van-e dolga. Az A: meghajtón létrehoz egy TESTZZT.P állományt, amit utána eltüntet. Egyszerű ötlet, de működik! Ha a floppy írásvédett, akkor nem működik a rendszer.

Kilencven újraindítás után a képernyő közepén a következő üzenet jelenik meg:

The software lease for this computer has expired. If you wish to use this computer, you must renew the software lease. For further information turn on the printer and press Return.

A szoftverbérleti szerződés erre a számítógépre lejárt. Amennyiben még szeretné használni ezt a számítógépet, meg kell újítania a bérleti szerződést. További információkért kapcsolja be a nyomtatót és nyomja meg az Enter billentyűt.

Ha a felhasználó eleget tesz az utasításnak, akkor a program a printere a következőket írja ki:

„If you are reading this message, then your software lease from PC Cyborg Corporation has expired. Renew the software lease before using this computer again. Warning: do not attempt to use this computer until you have renewed your software lease. Use the information below for renewal.
Dear Customer!
It is time to pay for your software lease from PC Cyborg Corporation. Complete the INVOICE and attach payment for the lease option of your choice. If you don't use the printed INVOICE, then be sure to refer to the important reference numbers below in all correspondence. In return you will receive:
- a renewal software package with easy-to-follow, complete instructions;
- an automatic, self-installing diskette that anyone can apply in minutes.
Important reference numbers: A9738-1655603-
The price of 365 user applications is US$189. The price of a lease for the lifetime of your hard disk is US$378. You must enclose a bankers draft, cashier's check or international money order payable to PC CYBORG CORPORATION for the full amount of $189 or $378 with your order. Include your name, company, address, city, state, country, zip or postal code. Mail your order to PC Cyborg Corporation, P.O. Box 87-17-44, Panama 7, Panama."

Ha Ön ezt az üzenetet olvassa, akkor szoftverének bérlete a PC Cyborg Corporationnél lejárt. Újítsa meg szoftverbérletét, mielőtt számítógépét ismét használná. Figyelmeztetés: ne is próbálkozzon a számítógép használatával mindaddig, amíg bérletét meg nem

újította. Használja a megújításhoz alábbi információinkat.
Kedves Ügyfelünk!
Itt az ideje, hogy kifizesse a bérleti díjat a PC Cyborg Corporationnek. Töltse ki a SZÁMLÁT és csatolja befizetését az Önnek megfelelő bérleti konstrukcióra. Ha nem a kinyomtatott SZÁMLÁT használja, akkor minden levelezésében hivatkozzon a lent közölt referenciaszámra. Válaszul megküldjük Önnek: megújított szoftvercsomagunkat, könnyen követhető, komplett leírással; önmagát automatikusan installáló lemezünket, melyet percek alatt bárki használni tud.
Fontos referenciaszámok: A9738-1655603-
A 365 felhasználói alkalmazás ára 189 dollár. Merevlemezének teljes élettartamára a bérleti díj 378 dollár. Megrendelése mellé csatolnia kell a PC CYBORG CORPORATION részére szóló bankátutalást, csekket vagy egyéb nemzetközi fizetőeszközt, a 189, illetve 378 dolláros teljes összegre vonatkozóan. Tüntesse fel a nevet, a céget, a címet, a várost, az államot, az országot, az irányítószámot. Megrendelését küldje az alábbi címre: PC Cyborg Corporation, P.O. Box 87-17-44, Panama 7, Panama.

Miután ezt a levelet a gép kinyomtatta, a következő üzenetet jeleníti meg a monitoron:

„Please wait thirty minutes during this operation. Do not turn off the computer since this will damage your system. You will be given instruction later. A flashing hard disk access light means WAIT!!!!!"

Kérem, várjon harminc percet, amíg tart ez a művelet. Ne kapcsolja ki a számítógépét, mert tönkreteszi a rendszerét. Később újabb utasításokat fog kapni. A merevlemez kijelzőjén villogó fény azt jelenti, VÁRJON!!!!!

Ez az üzenet mintegy óra hosszat marad a képernyőn, fél óráig pedig a merevlemez nagy intenzitással dolgozik.

Felbukkant ennek a trójai programrendszernek egy másik verziója is, amely tovább variálja a felhasználó és a gép kínzását.

Ha újra akarjuk indítani a gépet a Ctrl-Alt-Del billentyűkkel, akkor a gép hangszórója a rendőrségi sziréna hangját hallatja, a monitorra pedig kiírja az alábbi üzenetet:

„Sorry for the long delay ... still processing ... please wait."

Elnézést a hosszú várakozásért ... tart a feldolgozás ... kérem, várjon.

Az újraindítási kísérlet során nemcsak a rendőrsziréna hangját hallatja a rendszer, hanem egy újabb üzenettel is meglep bennünket.

WARNING - if you interrupt the program you will destroy the files on drive C.

FIGYELMEZTETÉS – ha megszakítja a programot, tönkreteszi állományait a C: meghajtón.

Hosszabb idő eltelte után – pl. 40 perc volt 20 Mbájtos XT-n, 4,77 MHz órajellel – a megjelenő angol nyelvű üzenet felszólítja a felhasználót, hogy tegyen be egy formázott üres lemezt. Az Enter gomb lenyomása után a lemezre generál egy SHARE.EXE nevű programot, valamint a teszt során egy MMM247HU.CPA nevű állományt. Ez utóbbi neve a tapasztalatok szerint kópiánként változik! Utána ismét üzenget nekünk:

„The SHARE DISKETTE in drive A is now ready for use. Please remove it. Take it to another computer. Turn on that computer in the usual way. After the computer has been booted, insert the SHARE DISKETTE into its drive A. Then at the C prompt type A:SHARE and then press ENTER. A short routine will follow on that computer. Afterwards, return the diskette to this computer."

Az ÁTADÓ LEMEZ az A: meghajtóban használatra készen. Vegye ki. Vigye át egy másik számítógéphez. Kapcsolja be azt a szokásos módon. Rendszerindítás után tegye be a ÁTADÓ LEMEZT az A: meghajtóba. A C bejelentkezőhöz írja be, hogy A:SHARE és nyomja le az Enter billentyűt. Egy rövid rutin következik azon a gépen. Utána hozza vissza a lemezt ehhez a számítógéphez.

Ha a gyanútlan felhasználó mindezt megteszi, tönkretette a másik gépet is. Ezzel installálja ugyanis a lemezt felismerhetetlenül összekeverő trójai programot. Ha újra beteszi a gépbe ugyanezt a lemezt, akkor egy új átadó lemezt követel a gép, mert különben törli az adatokat...

Rövid idővel utána összeáll a rendszer, és attól a pillanattól kezdve a gép használhatatlanná válik. Innen már a forgalomban lévő két trójai verzió ugyanúgy működik.

A trójai program elleni védőprogramot író Jim Bates, valamint a szétküldés alapjául szolgáló címlistát összeállító PC Business World szerkesztője, Mike Magee 1989. december 20-án a fentiekről számítógépes körlevelében tájékoztatta az elektronikus levelesládák használóit.

Amikor vége a merevlemez megmunkálásának, a főkönyvtárban egy új állomány jön létre a C: meghajtón, a CYBORG.DOC. Ebben az állományban megismétlik a kinyomtatott levélben található utasításokat a program „regisztráltatására". Utána azon a lemezen 0 bájtos szabad helyet találunk, amikor a tartalomjegyzéket szeretnénk kilistázni. A DOS operációs rendszerre ráül egy sajátos héj rutin, a shell, ami ezután megakadályozza, hogy rendeltetésszerűen használjuk a gépet. Ezt a rutint CYBORG.EXE-nek nevezik, és természetesen csak olvasható rejtett (hidden read-only) attribútumai vannak, hogy a normál listázás során ne lehessen észrevenni. Ez azután nem enged semmilyen DOS funkciót

meghívni vagy futtatni, helyette csökönyösen egyetlen rendszerüzenetet ismétel.

„WARNING: You risk destroying all of the files on drive C. The lease for a key software package has expired. Renew the lease before you attempt any further file manipulations or other use of this computer. Do not ignore this message."

FIGYELMEZTETÉS: Ön a C: meghajtón lévő összes állomány tönkretételét kockáztatja. Egy kulcsfontosságú szoftvercsomag bérlete lejárt. Újítsa meg a bérletet, mielőtt megkísérelné, hogy további állományműveleteket végezzen vagy más módon használja a számítógépet. Ne hagyja figyelmen kívül ezt az üzenetet.

Ha mégis akarunk más műveletet végezni, akkor a számítógép illegális utasításra vagy állománynévre hivatkozó üzenettel nem hajtja végre. Ha pedig kikapcsoljuk a gépet, és egy tiszta rendszerlemezről indítunk, akkor azt látjuk, hogy az egész merevlemezen egyetlen állomány, a CYBORG.DOC jelentkezik, és 0 bájt szabad hely maradt. Természetesen minden korábbi állomány rajta van a lemezen, de titkosítva, átkódolva és hidden attribútummal. A kísérletek során megfertőztek egy 20 Mbájtos merevlemezt. 90 rendszerindítást végezve elindították a trójai programot, majd miután az elvégezte a fentebb említett műveleteket, a hidden attribútum levételével listázhatóvá tették a tartalomjegyzéket. Az eredmény magáért beszél.

```
Volume in drive C has no label
Directory of C:\
#UCU#R      AK          1007          13-07-85      1:43p
#UC@R&      AK         27760           3-07-85      1:43p
COMMAND     COM        23717          13-07-85      1:43p
#1!8_68@    AU           587           3-19-89      9:11a
6#1N        AK            32           2-27-89     12:33p
KF{0U       AK           853          13-12-89      4:07p
}G6R        AG            98           1-04-80     12:01a
AUTOEXEC    BAT          108           1-04-80     12:01a
AUTOEXEC    BAK           17           1-04-80     12:01a
}#@@&       AU        172562           8-07-89     10:40a
&_}1        AU         46912          12-07-89     11:58a
!}          AU          7294           3-01-87      4:00p
1G          AU        102383           3-01-87      4:00p
```

```
H8C          AU       146188          1-04-80      12:11a
CYBORG       DOC        1326          1-04-80      12:05a
CYBORG       EXE         642          1-04-80      12:05a
AUTO         BAT         117          1-04-80      12:06a
17 File(s)    0 bytes free
```

Ezekhez még számos rejtett alkönyvtár is hozzáadódik. Azokban egy indexelt szekvenciális adatbázis részeit találhatjuk, amelynek mezőit a 20h-val töltötték fel. Ez az adatbázis foglalja el a lemez szabad területeit. Amennyiben a rendszer tápfeszültségét kikapcsoljuk, a merevlemez már nem bootol. Ha az AUTOEXEC.BAT állományt akár csak egyszer is végrehajtotta a rendszer, a Ctrl-Alt-Del billentyűkombinációval történt minden újraindítás után kiadott DIR parancs vagy DOS utasítás végrehajtásakor láthatóvá válik a figyelmeztetés. Amennyiben a Norton Utilities vagy más segédprogram segítségével belenézünk a CYBORG.EXE állományba, a következő érdekes jelenséggel állunk szemben. Ezt a szöveget az 560 offset címen találjuk az állományban:

```
<false end-file-marker> <The Norton Utilities cannot
read this file because the FAT has been locked> BORG
EXE
```

És kódot természetesen nem találunk semmilyen megszokott segédprogrammal. Amennyiben direkt szektorolvasással próbálkozunk, rájöhetünk a trükkre, hogy a CYBORG.EXE programkód egyes részleteit szétszórva a legkülönbözőbb offset címeken találhatjuk meg. A rendszer bennünket megelőzve a szövegeket és a merevlemez teljes könyvtári struktúráját alaposan átkódolta. Az előbbiekben bemutatott 20 Mbájtos merevlemez korrekt főkönyvtári listája a következő volt.

```
Volume in drive C has no label
Directory of  C:\
IBMBIO       COM       10071         13-07-85      1:43p
IBMDOS       COM       27760          3-07-85      1:43p
COMMAND      COM       23717         13-07-85      1:43p
INFECTED     EXE         587          3-19-89      9:11a
TINY         COM          32          2-27-89     12:33p
W13_B        COM         853         13-12-89      4:07p
AUTO         BAT          98          1-04-80     12:01a
```

```
AUTOEXEC   BAT        108          1-04-80     12:01a
AUTOEXEC   BAK         17          1-04-80     12:01a
AIDS       EXE     172562          8-07-89     10:40a
SCAN       EXE      46912         12-07-89     11:58a
FA         EXE       7294          3-01-87      4:00p
NU         EXE     102383          3-01-87      4:00p
REM        EXE     146188          1-04-80     12:11a
14 File(s)   15872000 bytes free
```

Megállapítható, hogy a titkosítás során az egyes betűket vonalak vagy egyéb jelek helyettesítik. A rendszer két kódtáblát alkalmazott. Az egyik a fájlkiterjesztéseket titkosította a következő táblázat alapján:

Eredeti	Kódolt	Eredeti	Kódolt
Nincs kiterj.	AB	APP	AC
BAK	AD	BAS	AF
BAT	AG	CAT	AH
CMP	AI	CNF	AJ
COM	AK	DAT	AL
DB	AM	DBF	AN
DCT	AO	DEM	AP
DIR	AQ	DOC	AR
DVC	AS	DYN	AT
EXE	AU	FIL	AV
FNT	AW	FRM	AX
GLY	AZ	HLP	BA
INP	BC	LBR	BD
LOC	BF	INI	BB
MDF	BG	?MF	BH
MNU	BI	MSG	BJ
NDX	BK	OUT	BL
OVL	BM	OVR	BN
PGM	BO	PIF	BP
PRD	BQ	PRG	BR

TBL	CA	TXT	CB
WK1	CC	WK2	CD
WKS	CE	XLT	CF
XQT	CG	ZBA	CH
DRV	CI	LRN	CJ
CAL	CK	FON	CL
SPL	CM	MAC	CN
TST	CO	LGO	CP
GRB	CQ	GRA	CR
DTA	CS	$$$	CT
VC	CU	TMP	CV
PAS	CW	OBJ	CX
MAP	CY	LST	CZ
LIB	DA	ASM	DB
BLD	DC	COB	DD
COD	DE	FOR	DF
FMT	DG	DIF	DH
DRW	DI	FLB	DJ
PIC	DK	PAT	DL
VFN	DM	GEM	DN
REN	DO	IMG	DP
RSC	DQ	MEM	DR

Az egyes állományok tartalmát szintén igen ötletesen egy másik kódtáblát használva tünteti el:

Eredeti	Kódolt	Eredeti	Kódolt
!	F #	I	
$	' (apostrophe)	%	)
&	S (apostrophe)	G	
(	#)	7	

-	(minus)	9	0 (zero)	_	(underscore)		
1		N	2	-	(minus)		
3		$	4	{			
5		}	6	T			
7		&	8	E			
9		0	(zero)	@			D
A		K	B	(			
C		M	D	J			
E		5	F	1			
G		U	H	R			
I		Z	J	4			
K		W	L	@			
M		8	N	Y			
O	(letter O)		V	P			L
Q		H	R	O	(letter O)		
S		!	T	6			
U		B	V	X			
W		%	X	P			
Y		2	Z	Q			
^	(caret)	~	(tilde)	_	(underscore)		C
{		3	}	A			
~	(tilde)	^	(caret)				

A rendszer adatait és a rendszerállományokat érintetlenül hagyja, és nem is kódolja át. A partíciós tábla és a boot-szektor érintetlen marad. Bootvírus a rendszerből sok-sok winchester tönkretétele után sem vált le. A rendszerállományokra viszont ráépül egy sajátos héj, ami vezérli a disznóságokat, és a felhasználó meglévő eszközeivel nem tud ennek a sajátos operációs rendszernek a mélyére hatolni. A program a rendszerállományokat és egyéb fontos rendszerelemeket egy titkosított alkönyvtárban rejti el, ahová a FAT-ban lévő belépési pontot is elkódolja. A kapuőr és a rendszer karmestere a CYBORG.EXE állomány, amely programozástechnikai csúcsteljesítmény. Eddig egyetlen használható ellenprogram készült, az amerikai Jim Bates által írt Clearaid, amely az Ázsió-Vikinél szükség esetén hozzáférhető. Ezzel vissza lehet állítani a rendszer eredeti állapotát.

Dr. Solomon's jelezte, hogy a rendszer által létrehozott SHARE.EXE nevű állomány futtatása eredeti formában 30 szabad újraindítást engedélyez, mielőtt tönkretenné a merevlemez adatállományát, amennyiben az instrukcióknak megfelelően cselekedtek. Megjegyzendő, hogy más forrásokból származó Cyborg lemezek a próbák során nem hozták létre sem a SHARE állományt, sem pedig a hozzájuk tartozó dokumentációt. A SHARE-t létrehozó példányokat a jelek szerint az USA-ban terjesztették.

Végezetül, mit tanácsol Jim Bates, ha valaki ilyen programrendszerrel találkozik:

1. Először tiszta rendszerlemezről indítsuk el a gépet.
2. Valamilyen segédprogrammal szedjük le a hidden és hystem attribútumot a rejtélyes állományokról és könyvtárakról.
3. A fejezetben említett összes állományt törölni kell, beleértve az AUTOEXEC.BAT-ot is.
4. A Trojan AIDS esetében vissza kell nevezni az AUTO.BAT-ot AUTOEXEC.BAT-ra.
5. Normálisan újra kell indítani a gépet.

Mindezt azért írtuk le ugyanolyan részletességgel, mint az első kiadásban, mert azóta Bács-Kiskun megyében is kidolgoztak egy olyan hardverkulcsot, amely szinte minden funkcióját megvalósítja a fentebb ismertetett trójai programnak. A két eset jogilag is analóg.

A forgalmazók általában gondosan titkolják, milyen másolásvédelmet alkalmaznak programjaikon, és az mit csinál a különböző helyzetekben. Éppen ezért unikum az a *Szoftvervédelem hardverkulcsokkal* című brosúra, amelyet egy kecskeméti cég bocsátott ki saját hardverkulcsának reklámozására, s amelyet valaki felháborodva küldött be az *Alaplap* szerkesztőségébe. Eddig szerencsére még nem találkoztunk olyan programtermékkel, amelyet ezzel az eljárással védtek volna.

A brosúra szerzői szerint a kulcs felnyitás ellen védett, felületszerelt technológiával készült, így szétszedése esetén áramkörei megsemmisítik magukat, helyrehozhatatlanul roncsolódnak. Megjegyzendő, hogy profik az ilyen kulcsok szétszedését meg sem kísérlik, helyette úgynevezett árakör-analizátorokkal derítik fel belső szerkezetét. Vannak nagyon egyszerűen alkalmazható technikák, amelyekkel minden kulcs válasza lehallgatható, és megkereshető a szoftverben az ezt kiváltó, illetve erre reagáló rutin. Ehhez csak egyszerre két vagy három gép és sok idő szükséges. Ha a kulcs bonyolult, akkor nem is a kulcs szimulálására gondol a programot megtisztító szakember, hanem arra, hogy az ellenőrzést végző rutinokat a védett programból kitakarítsa. Utána még meg kell szüntetni a programon belül ezeknek az átírandó részeknek az el-

sődleges és másodlagos védelmét (CRC vagy egyéb ellenőrzőösszeg), illetve ezeket a rutinokat is ki kell irtani. A programkód egy-egy részének különleges védelme árulkodik, hogy ott valamilyen hardverkulcs-ellenőrzési folyamat lehet. Szerencsére a szoftvereknél használatos legtöbb hardlock keresőrutinja tartalmaz egy kikapcsoló bitet, s ha ennek értékét átállítjuk, nem ellenőrzi a lockot. Ezt a fejlesztők maguknak tartogatták. Ilyenkor csak ezt kell megtalálni, és viszonylag egyszerűen lepucolhatóak a programok.

Az ismertető szerint a kulcs ellenőrzésének módját a fejlesztő maga határozza meg. (Minő nagyvonalú ajánlat! Minden felelősség elhárítva, tegyen meg a fejlesztő mindent, amit csak tud.) Ehhez a kulcs szállítója díjmentesen ad programokat a visszafejtés lehetetlenné tételére és példaprogramokat az ellenőrzésre. A kulcsellenőrző rutinok szinte minden programnyelvet támogatnak, a prospektus szerint legalább harminc-negyven félét. S most nézzük magát a hardverkulcsot!

A védelem lehet egyszerű védőkulcs. Ilyenkor a kulcsot alkotó komponensek huzalozása egyedi, és az ún. megrendelői kódot adja vissza. A szoftverbe beépített rutin ebből előállít egy ellenőrzőszámot, és a programozó dönt arról, hogy továbbengedi-e a programot, s ha nem, akkor hogyan bünteti meg a felhasználót a védelem hiánya miatt. Ez a „Biztonsági blokk nincs installálva" rendszerüzenettől a merevlemez és a gép tönkretételéig csak a programozó fantáziáján múlik.

A másik, a többregiszteres memóriakulcs felkínál minden olyan funkciót, amely a Trojan AIDS Informationben rejlő ötletek gyakorlati megvalósításához szükséges. Az egyszerű kulccsal ellentétben, amely gyártás közben rögzített, tehát módosíthatatlan információt tárol, ez 31 darab 16 bites regiszterben a futás közben képződő információk tárolására is képes. Minden megrendelő egy rá jellemző rutin segítségével – amelyet a 32. nem publikus regiszter tartalmaz – írhatja és olvashatja ezeket a regisztereket.

Ezek a lehetőségek igen komoly fegyvert adnak a fejlesztő kezébe – például a véletlenszámokkal való védelem lehetőségét. A program egyik pontján előállított véletlenszámot egy másik ponton kiolvassa, és ha a kettő nem egyezik, elindítja a védelmi rutinokat. Lehet a kód egy másik regiszter címének a kulcsban való tárolása és onnan beolvasása is.

Az igazán tisztességtelen lehetőségek még ezután jönnek. S erre a kulcsot reklámozó brosúra külön felhívja a programozók figyelmét. Például megvalósítható a „szoftverbérlet", amikor a kulcs meghatározza azt, hogy a szoftver meddig futásképes. A program élettartamának megváltása pedig új kiegészítő program vagy újabb jelszó megvásárlásával történik. Hasonló eljárással korlátozható, hogy hányszor lehessen indí-

tani a programot, akár egy napon, akár egy adott időszakon belül. Ilyenkor diszkréten újabb élettartam vagy futásszám megvásárlására adhat ajánlatot a program írója. Hasonlóképpen lehet ezzel az eljárással fokozatosan „lebutuló" vagy éppen fokozatosan okosodó programot írni. Kizárólag a programozó fantáziáján múlik minden.

Hangsúlyozzuk, hogy joghézagos szabályozásunk szerint a hardverkulcs árusítója nem követett el semmilyen törvénybe ütközőt. (Mint ahogy jelenlegi törvényeink szerint akár vírust is árulhatna valaki.) Éppen ezért nem is neveztük meg írásunkban. Egyszerűen etikátlanul, a programozók erőfölénye tudatában cselekedett.

AZ ADATFELDOLGOZÁS SZABADSÁGA

Ki kit és mi mit véd?

„Államunk továbbfejlődésének alapja, hogy a lakosság jól informált legyen, ne korlátozzák az állampolgárokat számítógép-birtoklási és használati jogaiban."
(Az Egyesült Államok alkotmánykiegészítési javaslata)

Még jócskán van mit tenni Magyarországon azért, hogy a számítástechnikában is jogállamiság legyen. Ehhez mindenekelőtt meg kellene változnia a jogi szabályozásnak, szigorúan megbüntetve a számítógépes programokkal kárt okozókat, korlátozva az erőfölénnyel való visszaélés lehetőségét. Példaként lebeghet szemünk előtt az USA szabályozása, amely a számítógépek vírusainak tenyésztését és terjesztését, a számítógépes programokkal való károkozást, továbbá az adatlopást bűncselekménynek tekinti (data crime), és elég szigorúan bünteti. Programozástechnikai kérdésekben azonban a hallgatás fala még ott is csak helyenként omladozik.

Az 1989-ben Londonban rendezett, adatbiztonsággal foglalkozó konferencián az ottani szakemberek bírálták a kormány halogató taktikáját, amellyel késlelteti a számítógépes kalózkodásnak, az adatok és gépek megrongálásának büntetését célzó törvény életbeléptetését. *A Lordok Háza még mindig úgy véli, hogy az effajta kalózkodás nem több rosszindulatú viccnél, ami legfeljebb némi anyagi hátrányt okoz* – jelentette ki az egyik alsóházi tag (CWI-Számítástechnika, 1989/49). A személyi jogok védelméről szóló törvénytervezetet első olvasatban még 1988 tavaszán tárgyalta az angol kormány, a büntető szankciókra vonatkozó ajánlásokat pedig 1989 novemberében az angol törvényhozás jogi bizottsága. Az október végén közzétett szöveg három informatikai bűncselekményt különböztet meg.

– Alapbüntetésként három hónapig terjedő szabadságvesztés kiszabását javasolja, ha valaki engedély nélkül hatol be egy információs rendszerbe. A bűntett jogi meghatározásánál a magánlaksértést tekintették analóg esetnek.

– Azok a számítógépes kalózok, akik tisztességtelen céllal avatkoznak be egy rendszer működésébe (mindegy, hogy ezt milyen módon teszik!), öt évig tartó büntetéssel sújthatók.

– Ugyancsak ötesztendei börtönnel büntethetnék azokat is, akik számítógépvírus segítségével rongálnak vagy semmisítenek meg valamilyen informatikai rendszert, illetőleg abban tárolt adatokat.

Ha ehhez hasonló törvények Magyarországon is lennének, vajon egyes számítástechnikai cégek vezetői és munkatársai közül hányan kerülnének kényelmetlen helyzetbe?

Még szigorúbb Svédország jogi szabályozása. Itt a büntetési tétel felső határa az életfogytig terjedő fegyház.

Az adatrendszerek megzavarása néha emberéletet is követel. Az USA-ban valaki behatolt egy kórházi gyógyszertár számítógépes rendszerébe, és az ennek következtében túladagolt gyógyszer egy ember életét kioltotta. Jóval többen halhattak volna meg Franciaországban, ahol egy gyógyszergyár központi számítógépében módosítottak információkat, kifejezetten terrorista szándékkal.

A számítógép az alkalmazott szoftverekkel együtt határtalanul kiterjeszti az emberek lehetőségeit, de egyúttal növeli függőségüket is: kiszolgáltatottjai vagyunk a programozóknak, a szoftverházaknak, a forgalmazóknak, mert a program, amit megveszünk, „fekete doboz". Nem tudjuk pontosan, mi van benne, de amíg nincs baj, addig ez senkit sem érdekel. Ha viszont bajba kerülünk, akkor kénytelenek vagyunk eleget tenni a számítástechnikai szakemberek, a forgalmazók diktátumának. Erről a szituációról önkéntelenül a klasszikus római jog *oroszlánszerződés* tényállása jut eszünkbe. A forgalmazó és a programozó olyan ismeretanyag birtokában van, amelyet korlátlan erőfölénnyel érvényesíthet a másik féllel, a számítógép (egyszerű) felhasználójával szemben.

A történelem előző korszakaiban a fegyverekben testesült meg a hatalom. Ma annak a kezében van korszerű fegyver, aki jól tudja használni a számítógépet. A fejlett informatikai társadalmakban szinte írástudatlannak számít az, aki nem tud bánni a géppel.

Az információ hatalmi tényezővé vált. Személyi számítógépe segítségével a pénzügyi szakember megszerezheti a legfrissebb tőzsdei információkat, tisztában lehet vállalatának és konkurenseinek helyzetével is. Amikor például a nyers adatokat egy számítógépes rendszeren átfuttatva minimalizálhatja adóját és maximalizálhatja nyereségét, akkor már jelentős előnyre tehet szert mind az állammal, mind a hagyományosan gondolkodó és dolgozó versenytársakkal szemben. A kicsit tehetősebb szakember pedig már otthon is olyan berendezéssel dolgozik, mint egy vállalat. Olyan történelmi korban élünk, amikor az egyes em-

bereknek megvan a lehetőségük, hogy akár saját maguk is adatfeldolgozó központként dolgozhassanak.

Az állampolgárok és a kormányok érdekei természetesen nem teljesen esnek egybe. Angliában a törvény minden állampolgárt kötelez arra, hogy ha rendelkezik adatbázissal, közölje az állammal, milyen adatokat tart nyilván benne, és honnan szerzi be azokat. A törvény indoklása szerint ezáltal a magáncégek nem kérdezhetik le jogosulatlanul a másokra vonatkozó adatokat. (Ez természetesen nem vonatkozik a kormányzatra.) Hasonló jellegű szabályozásokat tartalmaz a többször elfektetett magyar informatikai törvény tervezete is. Igaz, ebben az is benne foglaltatik, hogy mindenkinek joga van a rá vonatkozó adatok megtekintésére és szükség szerinti helyesbítésére – ez alól kivételt képeznek a belügyi és honvédelmi célú nyilvántartások...

Az Egyesült Államokban a jobboldali, bigott vallásos szervezetek profi módon nyomoznak. Olyan területeket céloznak meg, mint például a „pornográf" szoftver, a számítógépes társkereső szolgálatok kiszimatolják a postázási listákról, hogy kik olvasnak „veszélyes" könyveket, kik kölcsönöznek „erkölcstelen" videokazettákat. Más országokban ezt megtetézhetik azzal, hogy nemzetiségre, világnézetre vagy bármi egyébre kiterjesztik ezeket a (titokban szervezett) adatbázisokat. Már az egyes alapnyilvántartások összekapcsolása is nagy veszélyt jelent az egyén szabadságára.

Az Egyesült Államok nagy szakszervezetei és kormánytestületei azon vitatkoznak, milyen jellegű „számítógépes bűnözési" törvényt kellene náluk életbe léptetni. A kérdés az, hogy milyen legyen a törvény és mikortól legyen érvényes, nem pedig az, hogy kell-e egyáltalán ilyen törvény. 1979-ben hozták Michigan állam számítástechnikai törvényét, amely bűnnek tekinti, ha valaki „jogosulatlanul megkísérel hozzáférni ... bármely számítógépes rendszerhez ... illetve szoftverhez". A *jogosultság* fogalmát azonban sehol sem definiálták. A későbbiekben ez esetleg azt is jelentheti, hogy *a kormány engedélyével.*

A nyugati szakszervezetek is fenyegetést jelentenek a személyi számítógépek használatára. A 80-as években például Olaszországban aktívan támadták mindazokat, akik otthonukban női ruhákat készítettek eladásra. Olyan nagykereskedők véleménye szerint, akik követték ezt a folyamatot, ez csak az első lépcső. A szakszervezetek támadása megindulhat az összes otthon végzett üzleti tevékenység ellen, ide sorolhatják majd a számítógépes szoftverfejlesztést is. Az amerikai szakújságírók szakszervezeti vezetői, akik igen nagy jelentőséget tulajdonítanak az otthon fejlesztett szoftvereknek, úgy ítélték meg, hogy ezzel lényegében a munkához való jogot korlátoznák. A számítógép hibája ezeknek a szer-

vezeteknek a szemében, hogy könnyen hordozható, jól rejthető, s természetesen igen termelékeny.

A *PC World* 1986. januári száma Kevin Jenkinst, a *Hercules Computer Technology* cég vezetőjét idézte. Jenkins véleménye szerint az, hogy a számítógép kiterjeszti az ember szabadságát és új területeket nyit meg az emberi alkotóképesség előtt, csak egy „újhullámos" ostobaság. (A Jenkins által cáfolt megállapítások Steve Jobs-tól, az Apple Computers cég egyik alapítójától származnak.) Jenkins láthatólag a kormány és a szakszervezetek oldalán áll, velük karöltve szeretné ellenőrzése alá vonni az állampolgárokat és számítógépeiket.

A kormányok félelme a számítástechnikától a keleti blokk országaiban korábban volt jellemző. Az általános tiltásból már többnyire csak az adatkommunikáció monopolizálása, a betarthatatlan rendeletekkel való szabályozás maradt meg, amihez gazdasági oldalról még hozzájárul az egyéni import diszkriminatív vámokkal való visszaszorítása és a magas forgalmi adó is.

1985 nyarán Michael Brown, a Central Point termékmenedzsere a CopyIIPC program eladott példányszáma után a *Software Publishers Association* kitüntetésre pályázott. A könyveit megvizsgáló független csoport a 100 000 feletti értékesítés alapján megítélte az aranyozott emlékérmet, a *Software Publishing Association* azonban vissza akarta vonni. Szerintük ugyanis a CopyIIPC másolóprogram egyik alapfunkciója a másolásvédelem feltörése, amivel lehetőséget adnak a felhasználóknak arra, hogy valami kis előnyre tegyenek szert a forgalmazókkal szemben – akár több millió dolláros eladáskieseést is okozva. Azt nem vették figyelembe, hogy számos esetben szükség van a védelmek feltörésére, mert csak így tudnak biztonsági másolatokat készíteni lemezeikről, így tudják megelőzni azokat a károkat, amelyek vírustámadás vagy hardverhiba esetén érnék a felhasználókat az egyszer már megvásárolt programok ismételt megvásárlásával.

Michael Brown bíróság elé vitte az ügyet. A védelmi költségekre ő, aki akkor egy szabadpiaci fejlesztő volt, 180 000 dollárt szerzett. Az SPA csak 50 000 dollárt, annak ellenére, hogy „közösségi tragédiának" nevezte az eseményt. Az SPA kénytelen volt a kormányhoz fordulni, hogy megőrizze egyeduralmát a szoftverkiadás területén.

A nyomás a Central Pointot arra kényszerítette, hogy rontsa a CopyIIPC programjának képességeit. Így sem ez a program, sem hardveres megvalósítása nem tudja korrekten másolni a sávok közötti üres helyeket kihasználó, arra író védelmeket. (A keletkezett piaci űrbe viszont azonnal betört néhány távol-keleti termék.) Csak azt nem értjük, miért írja a szoftver a különben nem használt utolsó utáni floppysávra

a szoftververzió számát, nevét, és azt, hogy hányadik másolatot készítettük vele. Pedig Michael Brown maga állította, hogy cége a kalózakciók ellen a szoftver gyakori továbbfejlesztésével, javításával védekezik, és csak a legális vásárlókhoz juttatja el az aktualizált változatokat.

Az Ashton-Tate, a Microsoft és az ADAPSO (Association of Data Processing Organizations) 1986 szeptemberében közösen bejelentette, hogy a jövőben *nem alkalmaznak* másolásvédelmet az Egyesült Államokban és Kanadában forgalmazott szoftvertermékeire. Bill Gates, a Microsoft cég elnöke lakonikusan csak ennyit mondott: a vásárlók győztek.

Hazánkban azonban még mindig nem a forgalmazói diktátumok érvényesülnek. Érdekességként megemlítjük, hogy az 1990-es budapesti Compfair-en az egyik hazai nagy szoftverforgalmazó cég vezető menedzsere megkereste a kiállítókat egy szoftverrendőrség felállításának koncepciójával. Ennek a rendőrségnek felhatalmazást szeretett volna adni ahhoz, hogy bárkinek a gépét bármikor megvizsgálhassa, és ha ott találnak nem jogos szoftvert, vagy jogos szoftverből jogtalanul módosított verziót, akkor eljárhassanak az illetővel szemben. Eredményt azonban – szerencsére – még a forgalmazók között sem tudott elérni, megmosolyogták elképzelését.

Mindig vannak, akik korlátozni akarják számítástechnikai tevékenységünket. Készen kell állnunk arra, hogy megvédjük és lehetőség szerint ki is terjesszük jogainkat. Nemcsak a másolásvédelmek és a vírusok támadásával szemben, hanem a „kíváncsiskodókkal" szemben is védekezni kell, hogy adatbázisainkhoz ne juthassanak hozzá illetéktelenek.

Adataink védelmének számtalan módja van. Bemutatunk rá egy példát. Olyan bejelentkező rutint írunk, amely rákérdez a hét napjára, s ha a megfelelő nappal (vagy a hét bármely napjával) válaszolunk, nem enged belépni, leblokkolja a gépet. A helyes válasz ugyanis például a nagynénénk lánykori neve lett volna. Ezt senki illetéktelen nem tudná kitalálni. Így tudunk védekezni bármiféle jogsértő lekérdezés, kémkedés ellen. Létrehozhatunk biztonságos bulletin-board rendszert is (BBS), amelyhez mindazok hozzáférhetnek, akikkel meg akarjuk osztani adatainkat. Információk széles skáláján közösködhetünk így olyanokkal is, akikkel személyesen esetleg soha nem is találkozunk. Ezek a rendszerek nálunk még csak most szerveződnek, Nyugaton viszont már évtizedes hagyományuk van. Ezekben a rendszerekben békésen egymás mellett élhetnek a közkincs számba menő nyilvános részek és a személyes, védett adatbázisok, amelyekhez csak néhány arra jogosult férhet hozzá.

A gyors és biztonságos kommunikáció kulcsa lehet a bulletin-board szoftverekhez való hozzáférés. A szabad hozzáférésű elektronikus adat-

banki szoftver lehetővé teszi, hogy otthoni számítógépünk üzenetközpontként működjék. Az olyan shareware kommunikációs csomagok, mint például az RBBS-PC és annak jóval használhatóbb utódprogramjai, de még az intelligens kommunikációs programcsomagok, mint a TELIX is, kiválóak gép–gép közötti közvetlen üzenetküldésre, míg a Hayes SmartCom II ideális a személyek és a gépek közötti kapcsolathoz. Nincs igazán jelentősége annak, melyik szoftvert választjuk. A fő, hogy dolgozzunk vele.

E szoftverek segítségével – például ilyen a FIDONET – olyan önszerveződő rendszerek alakíthatók ki, amelyek véletlenszerű kapcsolati úton érik el a címzetthez legközelebbi adatbankot. Ezek a rendszerek a hivatalos kommunikációs rendszereket ellenőrző hivatalok és egyéb szervek számára gyakorlatilag megfoghatatlanok. Hatékonyságukat és a hivatalos postai szolgáltatásoknál nagyobb megbízhatóságukat, adattitkosságukat csak növeli az AX25 amatőr rádiós műholdas csomagkapcsolt adathálózat, amelyet sok esetben minden kormányzati és postai tilalom ellenére összekötnek ezekkel az önszervező hálózatokkal.

Jogaink védelmében minden adatállományunkat titkosítsuk! Olyan sok algoritmus közül választhatunk, hogy szinte mindegy, melyiket alkalmazzuk. Az információ kódolásának szükséges és elégséges mértékét az információ érzékenysége határozza meg. Olyan algoritmussal, bonyolultsággal kell kódolni adatainkat, hogy optimális esetben is tovább tartson megfejteni, mint ameddig az információ kényes, bizalmas információnak minősül. Például a másnapi tőzsdei előrejelzéseket viszonylag egyszerű algoritmussal is elegendő kódolni. Bonyolultabb kódolási algoritmust igényelnek a bankszámlák s valamivel még bonyolultabbat az érzékeny pénzátutalási rendszerek. A diplomáciai és nemzetbiztonsági információknál már szigorúbb követelmények érvényesülnek.

Tudnunk kell azonban, hogy nagyon könnyű megfejteni mindazokat a titkosító rendszereket, amelyek ismert, bevált módszereken alapulnak. Az amerikai Stanford Egyetem titkosító rendszerének kulcsát egyetlen személy, Adi Shamir teljesen egyedül fejtette meg, kiváló matematikai meglátásai alapján. Adi Shamir Izraelben a Weizmann Intézetben folytatta a nyilvánosságra hozott kulcsú titkosító rendszerekkel kapcsolatos kutatásait, amit az MIT-n Rivest és Adlemann kollégáival kezdett meg. A kaliforniai Santa Barbarában a titkosítási rendszerekről tartott konferencián, Adlemann egy Apple II gépen mutatta be Shamir eredményeit.

A Stanford munkatársai ezek után a DEA adattitkosító algoritmust fejlesztették ki, s azt elfogadásra ajánlották az Amerikai Szabványügyi

Hivatalnak. Ezt az algoritmust lehetne használni az összes bankban és központi intézményben. Az amerikai kormányzat azonban nem engedélyezi a DEA használatát a hadügy legalacsonyabb szintjén sem.

A Stanford munkacsoport 5 millió dollárt tűzött ki arra, hogy annyi párhuzamos processzorral működő gépet fejlesszenek ki, amely egy napon belül meg tudja fejteni a DEA algoritmussal lebonyolított adatátviteleket. Korlátlan idő és korlátlan eszköz birtokában (amivel senki nem rendelkezik) bármelyik kód, illetve titkosítás megfejthető. Saját tapasztalatunkból tudjuk, hogy a Sysdoki program titkosítási kulcsát arra vállalkozó személyek viszonylag gyorsan visszafejtették. Itt azonban mi is elkövettünk egy hibát: a másodlagos titkosítás kulcsa, Farmosi István kollégánk személyi száma, kódolás nélkül ott volt a programban.

Gyakorlatilag megfejthetetlen kódolási eljárás azonban csak egyetlen van. Algoritmusa teljesen ismert és publikált, de nehezen kezelhető, ezért nem terjedhetett el. Ez a *becsapódó ajtó* nevű kódolás, amelynek alapja két elegendő hosszúságú (minimálisan 20 számjegyű) prímszám szorzata. Itt a kódolás és a dekódolás külön-külön eljárást igényel, és még a megfelelő nagyságú prímszám előállítása is nagyon sok időt vesz igénybe. A kód megfejtése a mai legfejlettebb számítástechnikával is több száz évig tartana. Tudomásunk szerint csak az USA Nemzetbiztonsági Hivatala, valamint a nemzetközi bankszakma alkalmazza... Ők általában negyven jegy feletti prímszámokkal dolgoznak.

Saját használatunkra is találhatunk azonban olyan módszereket, amelyek egyszerűek, könnyen kezelhetők és nehezen fejthetők meg. Ilyen például a *szótárkód*. A biztonságos kommunikáció résztvevői kiválasztanak egy olyan könyvet, amelyet mindannyian megvesznek. A Biblia, az Oxford English Dictionary és hasonlók nagyon alkalmasak erre. Választhatunk azonban kódszótárnak akár geológiai cikkgyűjteményt, mesekönyvet vagy verskötetet is. Tudományos kutatók, akik nagy távolságról kommunikálnak egymással, szemeljenek ki inkább valamilyen szakmai szöveget tartalmazó könyvet, hiszen a Bibliában a molekula, ion, oszcilloszkóp és hasonló szavak nem találhatók meg. Programozhatjuk úgy személyi számítógépünket, hogy olyan új szótárakat hozzon létre, amelyekben a kulcsszavak (pl. tartárjárás, tőzsdei árfolyam, hidroxil ion) állandóak, és a kódok (szám- vagy betűcsoportok) változnak nagy gyakorisággal.

Az így titkosított kommunikáció visszafejtése évekbe, akár évtizedekbe is telhet, nem éri meg a feltörésbe dollármilliókat fektetni. Még ma is vannak második világháborúból származó olyan üzenetek, amelyeket nem sikerült megfejteni. Ezekkel természetesen már nem is érdemes foglalkozni – a titkosítás elérte célját, biztonságos volt az információküldés.

A kibernetikai forradalom olyan új joggyakorlatot kíván, amely azt juttatja kifejezésre, hogy míg a számítógépek a régi kapitalista értelemben magántulajdont képeznek, az adatokra már új törvények vonatkoznak, amelyek jobban illeszkednek az adatok természetéhez. Kevés kivételtől eltekintve minden ország joggyakorlata a szerzői jogokkal analóg módon kezeli a számítástechnikai szoftverrendszerek és adatbázisok kiadói és terjesztői jogait. Ugyanakkor a forgalmazók a fegyverekre jellemző technológiai eljárásokkal a felhasználót igen alárendelt helyzetbe hozó joggyakorlatot folytatnak.

Az adatokat „megfoghatatlan" vagyontárgynak tekinthetjük. Korlátlan mennyiségben előállíthatók, másolhatók. A másolás nem teszi azokat tönkre. Az adatok felhasználhatósága nem univerzális. Egy adat csak saját környezetében, idejében és közösségében képvisel értéket. Számos olyan értékes információ van, amely sokak számára semmiféle értékkel nem rendelkezik. Mit kezdünk például a tőzsdei számhalmazokkal, ha nem értjük és nem tudjuk mire használni azokat, mert nem vagyunk a tőzsdei akciókban érdekeltek?

E. F. Hutton közgazdász 1982 végén a Federal Reserve System számítógépének régi jelszavával hozzájutott a pénzkibocsátással kapcsolatos adatokhoz. Ezeket az adatokat egyébként hetente nyilvánosságra hozzák, a forgalomban levő pénz mennyisége pedig közvetlenül befolyásolja a keresletet. Ha viszont az adatokat olyan szaktudású ember kapja meg valamivel korábban, mint Hutton, akkor „csodákat" művelhet, amit meg is tett. A *Time* 1983. január 13-i számában ezt *adatlopásnak* minősítette. Pedig nem történt lopás, hiszen az adatok továbbra is változatlanul ott voltak a gépben, régi helyükön. Mind a kormány, mind a *Time* alapvetően azt kifogásolta, hogy egy magánszemély miért nem várja ki, amíg a hivatalos képviselő sajtótájékoztatón ismerteti az adatokat.

Michael Crichton *Electronic Life* című könyvében leszögezi, hogy a számítástechnika és a videotechnológia elképzelhetetlen az információmásolás és az információátvitel nélkül. John Brunner *The Shockwave Rider* könyvében pedig egyenesen azt olvashatjuk, hogy a kormány összes adatállományát nyílttá kellene tenni. Igaz, ez sok ember számára az erkölcsi véget jelentené. Nem véletlen, hogy túlbuzgó vagy éppen nagyon is józan tisztviselők minden elérhető adatra rá akarják aggatni az államtitok, banktitok vagy éppen a nemzetbiztonsági titok címkéit. Milyen pánikot okozott, amikor bizonyos listák létét világgá repítette a sajtó! Pedig csak névsorról volt szó. Olyanról, amilyet egy telefonkönyvből is lehetne generálni. Érzékennyé az az egy mondat tette, amely a lista címe volt...

1986 novemberében a Reagan-kormány közvetlen támadást intézett az adatfeldolgozási jogok ellen. Az akkori biztonsági tanácsadó, John Poindexter elmondta, hogy a kormány keresi a módját annak, hogyan korlátozhatná az egyéni adatbázis-szolgáltatásokhoz való hozzáférést. Ugyanilyen értelemben nyilatkozott Diane Fountain, a Nemzetbiztonsági Hivatal szóvivője is. Poindexter arra hivatkozott, hogy a FED meg akarja szüntetni a hozzáférést a nyilvános adatbankokban található kereskedelmi és gazdasági információkhoz, amelyek *érzékeny, de nem titkos* (sensitive but unclassified) minősítésűek. Fountain azt jelölte meg célul, hogy korlátozzák a hozzáférést olyan nyilvános adatbázis-szolgáltatásokhoz, mint a Nexis, a Dialog és a Delphi. A minisztérium olyan törvény kibocsátását sürgette, amelynek alapján megjelölhetnék azokat a személyeket, akik túl sok kérdést tesznek fel például a hitech témakörben (ide tartoznak például a lézerek!), és ezekről a megjelölt személyekről a minisztérium listát kapjon.

Ez az egyik legveszélyesebb tendencia. Felér az Orwell által megjósolt gondolati bűncselekmény fogalmával. Milyen jogon vesznek nyilvántartásba egy bizonyos témakör iránti érdeklődés miatt? A megoldás: ha az ember nem egy konkrét téma, hanem statisztika iránt érdeklődik, akkor az még gond nélkül megkapható. Így elegendő azt figyelni, hogy ezekben a rendszerekben mikor csökken a hozzáférhető hitech információk darabszáma. Akkor azután holtbiztos, hogy abban a témakörben egy szigorúan titkos katonai program indult. Ez tapasztalható volt a Manhattan project esetében is, a maghasadási témájú közlemények csökkenésekor, ma pedig egyes kémiai lézerek vagy a nagybonyolultságú dekódolási algoritmusok területén. A társadalom önvédelmének kell kimondania a végső szót, hogy ne váljunk a gépek vagy az információ szolgájává.

A szoftverfejlesztők és -terjesztők számára igen fontos a megfelelő intézkedések meghozatala és foganatosítása az olyan szellemi termékek jogvédelmével kapcsolatban, mint a számítógépes programok. Ezzel egyszerre számos eredményt érnek el:

a) Elveszik a számítógépes kalózkodással foglalkozók kedvét attól, hogy termékeiket lemásolják.

b) Támogatják a programozói és fejlesztői tevékenységet, mivel megakadályozzák azt, hogy más cégek minimális ráfordítással elérhessék ugyanezeket az eredményeket.

c) Megnövelik a fejlesztő cég és a vásárlók közötti bizalmat. A vásárló ugyanis biztos lehet abban, hogy nem olcsó, koppintott szoftvermásolatot kap.

d) Biztosítják a folyamatos árbevételt az egyes termékek után.

e) Megtartják a tisztességes versenyszellemet az üzleti riválisokkal.

A szoftver védelmével kapcsolatos szellemi tulajdonjogok három alaptípusa a szerzői jog (copyright), a szabadalom és a kereskedelmi titok. A copyright a szerző munkájának megjelenési formáját és tartalmát védi – ilyen a legtöbb országban létezik. Hazánkban a szoftvereket a szerzői jog eszközeivel védik. Ugyanakkor alkalmazzák a kereskedelmi titok fogalomkörébe eső eljárásokat is. Mivel a szerzői joggal kapcsolatosan igen sok kérdés tisztázatlan, lehetőség van például az installációs kulcsszót mint kereskedelmi titkot védeni. A hazai jogszabályok mellett a legjobban talán a jelszó ipari titokként való kezelése és a felhasználóval ilyen értelmű kötelezvény aláíratása mint polgári jogi aktus vezet eredményre az egyes szoftverek jogainak védelmében.

A szabadalom olyan jogokat jelent, amelyet a kormány garantál a feltalálónak. A feltaláló egy adott időtartamra kizárhat másokat a találmány elkészítéséből, használatából, forgalmazásából, majd az adott időszak elteltével a találmány „közprédává" válik. (Magyarországon a szoftver szabadalommal nem védhető, de bizonyos jogi furfanggal egy-egy algoritmus igen. Ezeket azonban célszerűbb az ipari titok körének megfelelő módon védeni.)

A kereskedelmi titok olyan know-how, amely egy adott üzletágban titkosan kezelendő, ez előnyt jelenthet a versenyben, mivel az iparban általában még ismeretlen. A kereskedelmi titok védelme igen sok mindenre kiterjedhet, beleérthetők előállítási módszerek, tervek, tervrajzok, képletek és üzleti információk és a számítógépes programok is.

Minden országnak megvan a maga sajátos jogrendszere, amely szabályozza az adott országban a szellemi termékekre vonatkozó védelmet. Ezek a törvények–szabályok megadják a szellemi termékek védelmének formáját:

– a szellemi termékekkel kapcsolatos jogvédelem hatáskörét;

– annak a módját, ahogy hozzájuthatunk ehhez a jogvédelemhez;

– azokat az intézkedéseket, amelyek a szellemi termékek jogvédelmét biztosítják.

Számos két- és többoldalú egyezmény is született a szellemi termékek tulajdonjogának nemzetközi szabályozására.

A szoftverfejlesztőknek és -terjesztőknek ismerniük kell azokat a problémákat, amelyek gyakran merülnek fel, amikor a nemzetközi piacra akarják kiterjeszteni termékeik jogvédelmét. Általában három alapforma valamelyikéről, esetleg ezek kombinációjáról van szó.

A szoftver jogi védelme a legtöbb országban a szerzői jog körébe tartozik. Mivel a szerzői jogi védelem kiterjesztése a szoftver területére általában egészen újkeletű, az ilyen jellegű védelemmel kapcsolatosan bizonyos kérdések még megoldatlanok maradtak. (Ide tartozik a védelem

érvényességi köre, és az, hogy milyen típusú szoftverekre érvényesek a szabályok.) Egyes országokban a szoftver rövidebb ideig élvez szerzői jogi védelmet, mint az egyéb szellemi termékek; máshol a törvény csak az adott ország állampolgáraira érvényes. Más országok pedig (főként a fejlődő országok) nem sorolják a szoftvert a copyrighttal védhető termékek csoportjába.

További nehézségeket okoz, hogy a copyright-védelem megszerzéséhez szükséges formalitások országonként változóak. Vannak például olyan országok, ahol letétbe kell helyezni egy referenciapéldányt (ez a *letéti védelem* a francia jogrendszerre jellemző), ugyanez másutt nem szükséges. Gyakran előfordul, hogy a szoftvercég, miután megkapta egy idegen országban a copyright-védelmet, bosszankodva veszi észre, hogy a törvénysértőkre kiszabott büntetés összege olyan alacsony, hogy ez bizony nem fogja elrettenteni a szoftverkalózokat a további működéstől.

Számos csapdát kell elkerülniük a szoftvercégeknek akkor is, amikor szabadalom jellegű védelmet kívánnak biztosítani termékeikre külföldön. Ennek fő oka az, hogy nem a szabadalom jellegű védelem az, amit szoftverviszonylatban nemzetközileg preferálnának. Vannak olyan országok is, ahol a szoftver nem tartozik a szabadalmaztatható termékek csoportjába. Még az olyan országokban is, mint például az Egyesült Államok, ahol a szabadalmi védelem kiterjed a számítógépes programok területére, csak bizonyos jellegű szoftverek szabadalmaztathatók. Sőt: ilyen jellegű védelmet csak azokkal a szoftverekkel kapcsolatban engedélyeznek, amelyek szoros kapcsolatban állnak a hardverrel, illetve bizonyos rendszerjellemzőkkel.

A szabadalmak védelmi időtartama is országonként változik. Az Egyesült Államokban például egy segédprogram (utility) szabadalmi időtartama az engedélyezéstől számított 17 év. Más országokban ugyanerre a szabadalomra hosszabb időtartamú a védelem, és a benyújtás időpontjától számítják (gyakori a benyújtástól számított 20 év). Számos fejlődő országban pedig jóval rövidebb ideig érvényes a szabadalmak védettsége.

Az egyes országok más-más prioritású szabvánnyal dolgoznak, amikor a szoftvertermékeknek megadják a szabadalmi védettséget. A legtöbb országban az kapja meg a szabadalmi jogot, aki elsőként jelezte szabadalmaztatási szándékát. Az Egyesült Államokban viszont az kapja a szabadalmi jogot, aki elsőként találta fel az adott terméket vagy folyamatot. Viszonylag gyakori, hogy nem az nyújtja be elsőként a szabadalmaztatási kérést, aki elsőként találta fel, illetve fejlesztette ki a szóban forgó terméket.

Vannak olyan országok, ahol az átlagosnál sokkal kevésbé veszik szigorúan a szabadalmi jogok betartását. Így a szoftvercégek olyan jelenséggel is találkozhatnak, hogy az adott országban ugyan szabadalom védi egy bizonyos termékük, ennek ellenére semmilyen intézkedést sem hoznak a jogsértőkkel szemben. (A szoftvercégeket per esetén a szakma legteljesebb megvetése is sújtja. Philip Katz ugyan elvesztette a pert a PKARC program eredeti forgalmazójával szemben, de erkölcsileg győztesen került ki a peres eljárásból. A felperes, pernyertes SeaWare pedig azóta állandó eladási nehézségekkel küzd. A tapasztalat azt mutatja, hogy a peren kívüli, esetleg kölcsönös engedményekkel járó egyezség a célravezetőbb a számítástechnikai szoftverek vitás ügyeiben.)

A fentiekhez nagyon hasonló az az eset, amikor értékes számítógépes programunkat kereskedelmi titok kategóriába akarjuk soroltatni a különböző országokban. Védelmet élvezhetnek ezek a programok mint kereskedelmi titkok hivatalosan, de a gyakorlatban országonként igen eltérő ennek a kategóriának a kezelése. Az Egyesült Államokban például az egyes államok helyi törvényei rendelkeznek erről, ezek érvényesülnek az országos hatáskörű törvénnyel szemben, gyakori a harmadik fél kizárása. Számos országban (ilyen például Japán) a kereskedelmi titok védelme csak arra terjed ki, hogy ha valaki kiadta a titkot, akkor azzal szemben fellépjenek.

A fejlődő országokban nem nagyon szeretik az olyan jellegű megállapodásokat, amelyek az üzleti információk (beleértve a számítógépes programokat is) bizalmas kezelésésére irányulnak. A bizalmas megegyezések gyakran ellentmondásba kerülnek azokkal a törvényekkel, amelyek a külföldi állampolgárok és a fejlődő ország állampolgárai közötti technológiai transzfert korlátozzák.

Vannak országok, amelyek korlátozzák a megegyezések időtartamát, így a kereskedelmi titok hamarabb nyilvánosságra hozható. Ennek következménye, hogy a szoftvercégek egyszer csak azon veszik észre magukat, hogy eddig megszokott egyezményeik, amelyeknek az volt a céljuk, hogy megakadályozzák egy számítógépes program nyilvánossá tételét, nem fognak megfelelően működni, ha a programot kereskedelmi titokként akarják védeni.

A szoftvercégeknek azt is tudniuk kell, hogy milyen korlátai vannak az egyes országok szellemi termékek védelmére vonatkozó törvényeinek. Komoly problémát okoz, hogy a copyright, szabadalom és kereskedelmi titok védelmét szabályozó törvények csakis az adott országban és a hozzá tartozó területeken érvényesek, semmiféle hatásuk nincs az országhatárokon kívül. A szoftvercégek számára ezért ajánlatos, hogy kétoldalú megállapodásokat kössenek a szellemi tulajdonjogokkal

kapcsolatban, és ezeket a megállapodásokat aláírják azok az országok, amelyekkel az adott szoftvercég üzleti kapcsolatban áll.

Az egyik legrégibb és egyben leginkább elfogadott nemzetközi copyright-egyezmény a *Berne Convention for the Protection of Literary and Artistic Works,* melyhez nyolcvannál több ország csatlakozott. Ez a Berni Egyezmény kötelezi az aláíró országokat arra, hogy a tagországok bármelyikéből beérkező szerzői jogi igényeket a saját törvényeik alapján kezeljék. Ezek az országok tehát a szellemi termékekre ugyanolyan jogvédelmet kötelesek adni a külföldi, de a Berni Egyezmény tagországai állampolgárainak, mint a saját országuk állampolgárainak. A megállapodás tartalmaz ezenkívül olyan minimum követelményeket is, amelyeket minden aláíró ország köteles az összes többi tagország állampolgárainak nyújtani. A Berni Egyezmény megtiltja továbbá, hogy az ide tartozó országok bármely más tagországból szerzői jogi védelemért folyamodó állampolgárral szemben akadályozó formalitásokat foganatosítsanak (kivéve természetesen a jogsegéllyel kapcsolatos formalitásokat).

A szabadalmi jogokkal kapcsolatban a legfontosabb nemzetközi egyezmény a *Paris Convention,* melyhez mintegy kilencven ország csatlakozott. A Berni Egyezményhez hasonlóan a Párizsi Egyezmény is megkívánja a tagországoktól, hogy a külföldiekre ugyanazok a rendelkezések vonatkozzanak szabadalmi jogok viszonylatában, mint az adott ország állampolgáraira. Emellett a Párizsi Egyezmény még egy, az előzőnél talán még fontosabb kikötést is tartalmaz. Ez a szabadalombenyújtás prioritását szabályozza. Azok a szakemberek, akik bármely tagországban benyújtották szabadalmazási igényüket, ezt a benyújtási dátumot használhatják az összes többi tagországban is, ha az eredeti időpont óta még nem telt el 12 hónap. Ez azt jelenti, hogy a szoftvercégeknek, miután valamelyik tagországban benyújtottak egy szabadalmaztatási kérvényt, egy évük van arra, hogy kihasználják elsőbbségüket.

Számos további nemzetközi egyezmény ismeretes, a szoftvercégeknek ezeket kell kihasználniuk, hogy termékeiknek védelmet tudjanak biztosítani a nemzetközi piacon is. Ilyen jelentős nemzetközi megállapodás például a *USA Copyright Convention* vagy a *Patent Cooperation Treaty.*

A szoftver hatékony védelme hazai pályán kezdődik, itt kell először biztosítanunk a copyright, szabadalom, illetve kereskedelmi titok jellegű védettséget. Ezután azokban az országokban kell megszereznünk a szellemi tulajdon védettségét, amelyekkel üzleti kapcsolatba kerülhetünk, ahol a legkomolyabb riválisaink jelen vannak, és ahol a kalózkodás nagyon elterjedt.

Ki kell használnunk azokat a nemzetközi megállapodásokat, amelyek

a szellemi tulajdonjogok védelmére szolgálnak (a Berni Egyezmény által kínált lehetőségeket és a Párizsi Egyezményben garantált elsőbbségi jogainkat). Ezenkívül hatékony módszert kell kidolgoznunk a kereskedelmi titkok megőrzésére is. Ennek magában kell foglalnia a „fekete doboz" elvét és a bizalmas megegyezéseket is. Így védekezhetünk az ellen, hogy értékes számítógépes programjaink tartalmát esetleg az előzetes licenctárgyalások és disztribútori megbeszélések során nyilvánosságra hozzák.

Olyan jogi tanácsadót alkalmazzunk azokban az országokban, ahová eladjuk programjainkat, aki teljes mélységében átlátja az adott ország szellemi termékek jogvédelmével kapcsolatos törvényeit. Lehetőség szerint ezek a tanácsadók ne álljanak semmiféle kapcsolatban azzal a külföldi céggel, amellyel licencmegállapodást kötünk. Mivel az országok többségében a szerzői jogi védelem a legelterjedtebb a szoftvertermékekkel kapcsolatban, előfordulhat, hogy egyáltalán nem tudunk szabadalmi védelmet szerezni termékeinkre, vagy ez a szabadalmi védelem csak nagyon korlátozott mértékű lesz.

Miután megtörtént az összes jogvédelmi intézkedés, gondoskodnunk kell a jogvédelmi előírások betartásának ellenőrzéséről. Fontos, hogy folyamatosan figyeljük az importot és a hazai eladásokat, s ha lopott szoftver behozataláról értesülünk, az azonnali intézkedések minimalizálhatják veszteségeinket.

Mindig jogi és sohasem szoftveres vagy hardveres eszközökkel biztosítsuk jogaink védelmét, ha azt akarjuk, hogy potenciális és valós vevőink ne ellenfeleink, hanem kooperatív és hasznot hozó partnereink legyenek.

VÍRUSTIPOLÓGIA

A családfa változásai

Könyvünk első kiadása óta jelentős változásokat tapasztalhattunk a vírusprogramozás elveiben és gyakorlatában. Nyilvánosságra kerültek részben azok a kutatások, amelyeket hadviselési célokkal folytattak, és a vírusok megismerésével fény derült több, addig nem publikált eredményre is. Megjelentek olyan vírusok, amelyek egyik vagy másik főcsoport színeiben – tehát boot- és fájlvírusként – egyaránt felléphetnek, és azok is, amelyek két vagy több komponens együttes jelenléte esetén aktivizálódnak vagy épülnek fel. A korábbi kiadásban ismertetett tipológia így csak részben alkalmazható, helyette fel kellett állítanunk egy jobb osztályozási rendszert.

Milyen is egy vírusprogram? Semmiképpen sem olyan, mint az élő szervezet, azaz nem látható mikroszkóp alatt, mint egy baktérium. Bár volt egy orvosnő, aki a sajtóban közölt egyik első víruscikk megjelenése után felhívta a szerkesztőséget, és kérte, mutassunk meg neki egy ilyen vírust mikroszkóp alatt! Igaz, a számítógépes vírusokhoz is van „mikroszkóp", a nagy hatásfokú disassembler program. Mi az AFD-t, az FSD-t, a Sourcer különböző verzióit és a Periscope szoftveres verzióját alkalmaztuk, s azokat az utóbbi fél évben néhány olyannal egészítettük ki, amelyek sok debug-ellenes trükköt ki tudnak védeni. Szerencsére ezek a programok bekerültek a hazai legális szoftverkínálatba is. (Már csak egy jó „disclipper" és „disfox" programra vágyunk, hogy belenézhessünk a könyvelési és anyaggazdálkodási rendszerek másolásvédelmeinek disznóságaiba, mert meggyőződésünk, hogy a vírusok jelenlétével nem igazolható rejtélyes adatvesztések okai sokszor az ezekben a programokban elrejtett „piszkos trükkök".)

A vírusprogramokra évekkel ezelőtt nagyon találó meghatározást adott Buruzs Tamás, a Kandó Kálmán Villamosipari Műszaki Főiskola informatika szakos hallgatója, amikor a MTESZ egyik rendezvényén Éltető Lászlóval, a másolásvédelmek jól ismert szerzőjével folytatott vitájában a következőket mondta: „A vírusprogram intelligencia és mesterséges értelem, de erkölcs és érzelem nélkül. Intelligenciáját a programozójától kapta, és annyira lehet erkölcstelen, amennyire a program írója is az. Már ma is lehetséges olyan programot írni, amely belátható

időn belül tönkreteheti egy teljes számítógép-generáció működését. Például egy vagy két esztendőn belül lehetetlenné tehető az MS-DOS alatt futó programok alkalmazása. A vírusprogram valójában az élő anyag működését utánzó életképes modell. Olyan, mint a biológiai fegyver, mert miután kiengedték a laboratóriumból, még maga az alkotója is elveszíti az ellenőrzést felette."

Vírusfejlesztő készlettel később sem találkoztunk (hacsak néhány vírus teljes forráskódban terjedő verzióját nem vesszük annak), viszont automatikusan generálódó vírusváltozattal már igen. Többen is kifejlesztettek olyan programot, amely a víruskód szétvágásával, az egyes részek közé üres utasítások beszúrásával a hagyományos programok számára felismerhetetlen kódot állít elő. A kód csak a megszokott, szekvenciakeresésen alapuló programok számára felismerhetetlen. Ezzel az eljárással vírusok szinte végtelen számú változata generálható anélkül, hogy a szerzőnek a legcsekélyebb mértékben vissza kellene fejtenie a kódot. Az első ilyen produktumra 1991 februárjában bukkantunk: a Potyogós vírus átirata, „Kitolós Potyogós" néven vált ismertté a szakmában. Hosszabb ugyan, de azonosítóját a programozó „művész" változatlanul hagyta, így a hagyományos vírusölők felismerik, viszont garantáltan rosszul irtják, mert közben tönkreteszik magát a megtámadott programot is.

Könyvünk egyik célja, hogy leleplezzük a vírusírók által használt programozástechnikai trükköket, s bemutassuk azt az eszköztárat, amelynek segítségével még egy ismeretlen vírus ellen is viszonylag rövid idő alatt megírható a specifikus killer (azaz a vírusprogramot kitakarító és az eredeti állományt a lehetőségekhez képest eredeti formájában visszaállító), illetve a detektor (azaz a vírus jelenlétét, a vírus által végzett rendszerműveleteket érzékelő) program. Az elsődleges cél a felhasználók adatainak védelme!

Elmúlt már az az idő, amikor egy-egy vírus ellen megírt programmal le lehetett tarolni a piacot. A felhasználó nagyobb biztonságra vágyik, és azt részesíti előnyben, amelyik a károk megelőzésére komplex védelmet tud nyújtani. A világon ma még nem sok cég szakosodott erre a feladatra, s azok is folyamatos információ- és programcserét folytatnak egymással. Most folyik annak a rendszernek a kidolgozása, amely szigorú garanciák mellett teszi lehetővé a vírusok cseréjét. Vírusok elleni programot könyvből ugyanis nem lehet írni, mert szükséges hozzá magának a vírusnak a birtoklása és visszafejtése. A garancia pedig azért kell, mert túl nagy a kísértés arra, hogy miután valaki kidolgozta az ellenanyagot, kiszabadítsa a vírust a „palackból", hogy ezzel saját víruselleneS programjának keresletét növelje.

A számítógépes vírusokat, a romboló programokat igyekszünk lelep-

lezni, mielőtt azok még kárt okozhatnának. A számítógépes vírusprogramok íróinak egyik célja, hogy a számítógéppel dolgozókat bosszantsák, és megakadályozzák a program vagy a számítógép rendeltetésszerű használatát. Tehát hogy romboljanak, kárt okozzanak. A vírusok ott szaporodnak a legjobban, ahol egy számítógépet egymástól függetlenül többen használnak: egyetemeken, iskolákban, számítástechnikai klubokban. Ha ilyen helyekről kerül elő egy vírusprogram, az általában még nem jelenti azt, hogy azt ott is fejlesztették ki. Nagy szoftverforgalmuk miatt ezek a számítógéplaborok sajátos „légyfogóként" működnek, és az itt felbukkanó vírusok mintegy előre jelzik az országos járványokat is.

A számítógépes vírusprogramok a biológiai vírusokhoz hasonlóan az egészséges szervezetet (programot, gépet) megtámadva szaporodnak. A számítógép operációs rendszerét felhasználva fertőzik meg a programokat, ritkábban magát a gépet, a hardvert, a bootvírusok pedig az üres floppylemezt. A vírusok olyan „ördögi kóddal" toldják meg a programokat, amelyek másik számítógépen is reprodukálni tudják önmagukat. (A hordozó programot – miként a biológiai hadviselés szakirodalmában – a szakírók „vektornak" nevezik.)

Nagy meglepetésnek számított, hogy a könyv első kiadása után viszonylag rövid időn belül felbukkantak a változó (mutáló) kóddal, illetve változó terjedési algoritmussal működő vírusok; jóval korábban, mint ahogy azokat előre jeleztük. A magukat álcázó, „lopakodó" programozástechnikát alkalmazó vírusok gyors hódítása is várható volt, hiszen az elmúlt egy évben sorra jelentek meg az MS-DOS operációs rendszer „átverésével" a gép és a merevlemez kapacitását megtöbbszöröző segédprogramok. Például a gép sebességét felgyorsító programok a RAM felfrissítési idejével variálnak. A magyar Moxla is ennek a lehetőségnek a kihasználásával okoz vagy szimulál RAM paritási hibákat. Az amerikai Stacker programcsomag memóriában történő tömörítő on-line kódolással és a merevlemez nem használt területeinek bekapcsolásával növeli a lemez tárolási kapacitását. Ez a program teljesen legális célra használja fel a dBase, a Trojan AIDS Information és a Virus2000 kódolási és memóriakezelési trükkjeit. A szerzőcsoport így tudta elérni, hogy e program memóriában maradó része mindössze 30 kilobájt.

A hordozó program ügyes megválasztásától függ, hogy a fertőzés mekkora adatállományban és programállományban lép fel egyidejűleg. Vegyünk például egy olyan programot, amely egyszerűen és gyorsan készít másolatot lemezünkről. Ezt nyilvánvalóan mindenki szívesen használja, különösen ha szabadszoftverként került a piacra. Csábító lehetőség tehát beleépíteni olyan vírusprogramot, amely például 100 másolás után aktivizálódik. De ha a vírus programozója nem tudja megoldani a

másolás figyelését (ami programozástechnikailag valóban nehéz feladat), akkor választhatja például azt, hogy a vírus egy év lappangási idő alatt nem csinál semmit. Addig pedig, amíg életre nem kel, csak terjednie, észrevétlenül szaporodnia kell, azaz bemásolnia magát más programokba. Amikorra kellőképpen elterjedt, mintegy varázsütésre megindul a károkozás. A pokolgép robban. Ha a felhasználó még a terjedési szakaszban észreveszi a vírus jelenlétét, akkor természetesen megakadályozhatja a pusztítást.

A vírusok leggyakrabban a COMMAND.COM, az IBMBIO.COM és az IBMSYS.COM (illetve .SYS) programokat fertőzik meg az MS-DOS™, valamint az IBM PC-DOS™ operációs rendszerek esetén, mivel ezek minden DOS rendszerlemezen megtalálhatók. A UNIX™, a XENIX™ operációs rendszerrel, valamint a NOVELL™ hálózati rendszerrel már nehezebb a vírusok dolga, mert ezek részben védettek a külső, illetéktelen programmódosítások ellen. De ezek is kicselezhetőek. Az operációs és hálózati rendszerállományok méretének figyelése minden esetben célszerű. Természetesen ha már bejutott a vírus, az a programok elindításával azonnal tovább is terjedhet.

Az egyes önreprodukáló és kártékony programokat a szakirodalom különbözőképpen nevezi. Célszerű tehát tisztázni, hogy mi melyik elnevezésen mit értünk, beleértve néhány biztonságtechnikai fogalmat is. Összeállítottuk ezért a magunk kis értelmező szótárát, nem betűrend, hanem inkább témakörök szerint.

Biztonság

Számítógépes rendszerekben biztonságon a számítástechnikai feladatok jogosult, megfelelő idő alatt történő és helyes végrehajtását értjük. Magában foglalja a megbízhatóságot, az integritást és a rendelkezésre állást is.

Megbízhatóság

Az a biztonsági tényező, amely arról gondoskodik, hogy az információ csak azokhoz juthasson el, akik erre fel vannak jogosítva. A megbízhatóság hiánya esetén illetéktelenek olyan adatbázisokhoz férhetnek hozzá, olyan állományokat nyomtathatnak ki, amelyekre nincs jogosultságuk. A megbízhatóság az adatvédelem alapkérdése.

Integritás

Az a biztonsági tényező, amely az információ és az információfeldolgozás helyességéért felelős. Csorbát szenved az integritás, ha fontos ál-

lományokat törlünk vagy károsítunk meg, ha tévesen változnak meg az adatbázisok elemei stb. A programok különböző eljárásokkal ellenőrzik önmaguk vagy egyes rutinjaik sértetlenségét. Ha ilyesmiben változást észlelnek, működésbe lépnek különféle mechanizmusok. A PC-SCAN esetében a sérült program újragenerálja magát eredeti formájában. A McAfee féle programok szöveggel figyelmeztetnek, hogy megváltoztak és fertőzésgyanús körülményeket tapasztaltak. A magyar CHKVIR program figyelmeztetés kíséretében öngyilkos lesz, így hívja fel a veszélyre a figyelmet. A másolásvédelmek pedig olyan mechanizmussal vannak ellátva, hogy ha bizonyos részeknél változást észlelnek, azonnal „ütnek, mint a bolondóra".

Rendelkezésre állás

Azt a biztonsági tényezőt jelenti, amely azért felelős, hogy az információk és a szolgáltatások idejében eljussanak a felhasználókhoz. Ez a funkció sérül hálózati problémák esetén, rendszerek és hozzáférési jogok keveredésekor stb.

Kiskapu, hátsó ajtó, vészki(be)járat

Olyan lehetőség, amelyet a program tervezője épít bele a programba. Célja az, hogy speciális lehetőségeket nyújtson a tervezőnek, de ezekkel a program normál felhasználói nem élhetnek. Egy bejelentkező program kiskapuja például lehetővé teszi a tervező számára, hogy akkor is bejelentkezhessen, ha számára a rendszerben nincs kijelölt jogosultság.

Programférgek (worms)

Olyan programok, amelyek nem szaporodnak, hanem belépve egy rendszerbe keresztülrágják magukat annak védelmi mechanizmusán. Feladatuk legtöbbször az, hogy behatoljanak az operációs rendszer magjába (a kernelbe), és onnan kihozzanak bizonyos információkat, például jelszótáblákat. Ennek elvégzése után általában csendesen kimúlnak. Újabban férgeknek nevezik azokat a vírusokat is, amelyek pusztítása nem látványos, hanem lassan, fokozatosan munkálkodva teszik tönkre az operációs rendszert.

Trójai programok

Kissé körülményesen „trójai faló típusú programoknak" is nevezik őket. Lényegük: maguk a programok csak álcázásra szolgálnak, mást (vagy mást is) tesznek, mint amit ígérnek. Például megveszünk egy könyvelői vagy DTP rendszert, majd egy év és néhány nap elteltével gé-

pünket bekapcsolva azt vesszük észre, hogy programunk is, adatállományunk is tönkrement. Tessék megvenni újra! A trójai programok általában nem viselkednek „tisztességes" vírushoz illően. Legtöbbjük aktivizálódása után azonnal „üt", olyannyira, hogy nincs is ideje szaporodásra, legfeljebb a gépen belül.

Vírusprogramok

Vírusprogramoknak azokat a programrendszereket nevezzük, amelyek önmagukat reprodukálni képesek. Más szoftverek megfertőzésével, floppyval (bootvírusok), magával a géppel (hardvervírusok) vagy számítógépes adatátviteli hálózaton keresztül terjednek. A fertőzés, a támadási felület, a károkozás és a terjedés módja szerint további osztályokba sorolhatók. Vegyük sorra most mi is, hogy milyen típusú vírusok keseríthetik meg életünket.

Többen többféleképpen csoportosítják a vírusokat, például a vírus támaszpontja vagy a vírus által végzett tevékenység szerint is.

A vírusok tevékenység szerinti csoportosításának alapját az irodalomjegyzékünkben is idézett IBM-tanulmányban fektették le. Bár nem egyezik meg egyik nálunk megszokott és a Vírushatározóban megtalálható felosztással sem, érdemes megismerkedni vele, mert a szakirodalomban sokszor találkozhatunk ezekkel az elnevezésekkel.

Logikai bomba (logical bomb): olyan trójai program, amely azért maradt a számítógépes rendszerben, hogy bizonyos körülmények között aktivizálódjon. Ezt előidézheti például egy állomány változása (megtörtént eset: a tervező nevének törlése a cég alkalmazottainak listájáról), a programnak adott bizonyos input sorozat, egy speciális időpont vagy dátum stb.

Baci *(szakzsargon):* olyan önálló program, amely végrehajtásakor saját maga másolatát átküldve terjed más felhasználókra vagy rendszerekre. Mivel nem fertőz meg más programokat, inkább rendszervírusnak minősül. Nem törekszik a rendszerforrások kimerítésére.

Patkány vagy **rat** *(szakzsargon):* a rendszer bizonyos erőforrásait (CPU-idő, lemezterület, spool-terület stb.) teljesen kimeríti azáltal, hogy korlátlanul megsokszorozza önmagát. Abban különbözik a baciktól, hogy ezeket kifejezetten az erőforrások kimerítésére tervezték. A vírusoktól pedig abban, hogy a patkányok önmagukban is teljes programok, nem kellenek hozzá más hordozóprogramok.

Programpestis: inkább csak a külföldi újságszövegekben használják ezt a terminológiát, gyakorlatilag minden olyan programra, amely más programokat, illetve adatokat veszélyeztet vagy megkísérli a rend-

szer biztonságának áttörését. Így tehát beleértendők az olyan rosszindulatú képződmények, mint a pokolgépek (trójai programok), logikai bombák, vírusok stb.

Időzített bomba vagy **time bomb:** olyan logikai bomba, amely egy adott időpontban vagy adott napon aktivizálódik.

Pokolgép vagy **trójai program:** minden olyan program, amelyet úgy terveztek, hogy olyan tevékenységet is végrehajtson, amelyet a felhasználó nem akart elvégeztetni. Jó példa erre egy olyan program, amely a gép bejelentkezési rendszerét szimulálja, és ahelyett hogy beléptetné a felhasználót, csak egy állományban rögzíti azonosítóját és jelszavát egy későbbi kigyűjtés céljából. Ahelyett tehát, hogy bejelentkeztetné a felhasználót (ez az, amit a felhasználó szeretett volna), ellopja a jelszavát (amit a felhasználó egyáltalán nem akart), ezáltal a trójai program tervezője bejelentkezhet a továbbiakban úgy, mintha ő lenne ez a felhasználó (ezt ismét nem akarhatta az, aki éppen megkísérelte a bejelentkezést).

Vírus: olyan program, amely más programokat módosít úgy, hogy ezek a továbbiakban tartalmazzák a vírusprogram másolatát. Nem kell feltétlenül rosszindulatú műveleteket végrehajtania egy programnak ahhoz, hogy vírusnak minősüljön, a lényeg, hogy megfertőz más programokat. Számos ismert vírus azonban jócskán végez rosszindulatú tevékenységet is.

Programféreg vagy **féregprogram:** olyan program, amely önmaga másolatait terjeszti a hálózatba kapcsolt számítógépekben. Vannak olyan programférgek, amelyek a biztonsági rendszert veszélyeztetik, ugyanis a rendszertulajdonos akarata ellenére arra használják a hálózatot, hogy önmagukat terjesszék, így a túlterheléssel tönkretehetik a teljes rendszert. (Nem azonos a szintén programféreg névre hallgató és például a jelszót kihozó programocskákkal. A korábbi kiadásban ezeket a programokat „memóriaszemét" vagy „kuka" vírusoknak neveztük.)

A programkódot módosító vírusok

A legismertebb és leggyakoribb víruscsalád. Az eredeti programmal sohasem találkozunk, hacsak magunk nem írunk ilyet. Viszont az általa módosított programmal könnyen köthetünk nem kívánatos ismeretséget. Igaz, ez a kód már magát a vírust is tartalmazza. Tágabb értelemben ide sorolhatjuk a floppy és a merevlemez formátumát megzavaró „vendégeket" is. Szintén nagyon sok altípusa létezik, támadáspontja és terjedési módja szerint.

Veszélyességük miatt ezekkel fogunk könyvünkben legtöbbet foglal-

kozni, hiszen a szoftvermásolás és az adatátvitel során főképpen ezek a vírusok terjednek. Ebben a csoportban tartják számon a szakemberek az etikátlan másolásvédelmet, valamint egyes hasznos, a vírusok ellen használható szoftvertípusokat is. Megjegyzendő, hogy a másolásvédelemként alkalmazott vírusjellegű programok egy része nem szaporodik, csak a kijelölt célprogramot fertőzi meg a „védelem felrakásának" nevezett folyamat során.

Lopakodó és alakváltó vírusok

Most van kialakulóban a vírusok olyan új nemzedéke, amely nem sorolható az eddig ismert egyetlen kategóriába sem (jóllehet, egy következő kiadásban már önálló családot is képezhetnek). Ezek szintén módosítják a programkódot, de ha jelen vannak is a memóriában, nem láthatóak (stealth programozástechnika). Változtatják terjedési algoritmusukat (Typo boot és Com verzió), fertőzési hosszukat (a magyar Phantom vírus) vagy éppen saját víruskódjukat (Whale). Van a kétlépéses fertőzéssel dolgozó vírus is (a Vacsina néhány változata, amely először az EXE>COM konverziót végzi el, majd később teszi bele a víruskódot a programba) és a egymással szimbiózisban élő vírusrendszerek (Whale–Fish-6).

Hardvervírusok

Az eddig említett vírusok fertőzéshordozója mindig valamilyen program, amely adatátviteli csatornán vagy floppyn kerül a rendszerbe. Fertőzési forrás, „bacilusgazda" azonban lehet maga a hardver is, pontosabban az abban gyárilag vagy egyéb úton elhelyezett szoftver (firmware) is.

A régi PC-kben és XT-kben nagyon kevés olyan rejtett zug volt, ahova ilyen csapdát be lehetett volna építeni. Ehhez ki kellett volna cserélni a gép ROM BIOS-át vagy a lemezmeghajtó kontrollerének ROM-ját. Az AT az első olyan gép, amely szinte tálcán kínálta a vírusnak a lehetőséget: az órakörnek vannak olyan nem publikált EEPROM regiszterei, ahova beírható egy rövid, de üzemképes víruskód. Azért hallunk viszonylag ritkán ezekről, mert egyetlen gyártó cég sem engedheti meg magának, hogy fertőzött terméket bocsásson ki. Komolyabb hardvervírust pedig szinte kizárólag csak a gyártó helyezhet el egy rendszerben.

Hardvermódosító vírusok

Bár általában ezeket is a hardvervírusok közé sorolják, inkább a vírusprogramok speciális nemzedékének tekinthetők. Az elektronikai

szakmérnökök körében sokáig tartotta magát az a tévhit, hogy pusztán programokkal nem tehető tönkre egy áramkör vagy nem válhat ócskavassá egy egész berendezés. Az élet rácáfolt erre. Az egyre intelligensebb építőelemeknek egyre több a sebezhető pontjuk, hiszen a gyártás tipizált, és sok áramköri lapkáról csak a mikroprogramok beégetése során dől el, hogy valójában milyen feladatokra szánják azokat. S amit így utólag „beégetnek" (tehát nem az áramkörbe huzaloznak be), azt megfelelő eljárással módosítani is lehet.

(Szórványosan hazánkban is felbukkantak a 80386-os processzor mikroprogramját módosító vírusprogramok. Ezek a processzorban a gyártás során betöltött, elektromosan írható és törölhető regiszterekben lévő mikroprogramokat írják át. A szükséges utasításokat a gyártók természetesen nem publikálták, de ügyes programozók, kiderítették azokat.)

Álvírusok

Bár nem vírus, de ebben a témakörben feltétlenül említésre érdemes a hardverben előforduló „selejt", ami működési zavarokat okoz. Az új processzorok első sorozatai például eddig rendszerint hibásnak bizonyultak.

Hasonlóan kellemetlen a szoftverben felejtett programhiba, a „bogár" (bug), ami trójai programokhoz hasonló viselkedést eredményezhet. Például a Norton Disk Doctor első kiadása volt ilyen (NDD.EXE a Norton Advanced Utilities programcsomagból). Ha ugyanis nem DOS-ból, hanem például a Disk Manager szoftverrel formáztuk merevlemezünket, az NDD közölte, hogy a lemez hibás (ugyanis nem DOS), és a C: partíció kivételével – sajnos rákérdezés nélkül – „helyreállította", vagyis tönkretette azt. (Az újabb verziókban előbb kérdez, tehát a hibát kijavították.)

Magunk is előidézhetünk a trójai programokéhoz hasonló következményeket, ha meggondolatlanul átnyúlunk az operációs rendszer feje felett, például megfelelő mélységű szakismeretek nélkül belenyúlva a FAT-ba (az állományelhelyezési táblába). A programokba mindig csak olyan helyeken turkáljunk bele, ahol értjük is, mi történik, ha valamit megváltoztatunk!

MIT TEGYÜNK, HA JÖN A VÍRUS...?

...És hogy ne jöjjön!

A vírusszakértők eddig több mint háromszáz vírus ezer feletti változatát regisztrálták. Ez óriási potenciális fenyegetést jelent a számítógépes rendszerek biztonsága, integritása szempontjából. A védekezés szempontjából legfontosabb a „szabályos" vírusok viselkedésének ismerete, a számítógépes rendszerekbe való bejutásának megakadályozása, az esetleges fertőzések felderítése és a betolakodóknak a kitakarítása. Ebben a fejezetben főleg ezekkel a témákkal foglalkozunk.

Mindenekelőtt fontos, hogy tisztában legyünk a vírusok működésének lényegével. A vírusok alapelemei olyan utasítássorozatok, amelyek végrehajtásuk után más programokra, állományokra is átterjednek. Egy tipikus számítógépes vírus két műveletet végez el. Először bemásolja magát a még fertőzetlen programokba, állományokba. Másodszor (adott számú futtatás után vagy megadott dátum elérésekor) végrehajtja azokat az utasításokat, amelyeket a vírus szerzője előre beprogramozott. Az aktivizálódás feltételrendszerét nevezi a szakirodalom trigger (kiváltó) feltételnek. A vírusszerző szándékától függően ezeknek az utasításoknak gyakorlatilag korlátlan kihatásuk lehet a rendszerre. Szerencsés esetben csak üzenetek jelennek meg, máskor viszont állományok törlése vagy adatok megváltozása is bekövetkezik. Sajnos egyre ritkábbak azok az esetek, amikor a vírus nem tartalmaz pusztító hatású utasításokat. Bár még a csupán önmagát megsokszorozó vírus is sok kellemetlenséget okozhat: lemezterületet foglal, CPU-időt igényel vagy a hálózatot terheli.

„Apu, hod' med' be..."

Vegyük azt az esetet, amikor egy cég külső dolgozónak ad ki munkát, a feladat egy részét azonban helyben kell elvégeznie, például ki kell javítania a behozott anyagot. Magával is hozza kedvenc szövegszerkesztőjét, ami az ő tudta nélkül is vírusfertőzött lehet. Ha ezt a szövegszerkesztőt a cég bármely gépén elkezdi használni, a vírus már át is terjed a gépen lévő valamelyik programra, esetleg egy táblázatkezelő állományba. Az is előfordulhat, ha floppyját bootvírussal fertőzött gépen

formázta, s ilyenkor az is elegendő, hogy véletlenül ráindítja a rendszert (hiszen az A: meghajtóban felejtett floppyval történő indítás az egyik leggyakoribb tévesztésünk), s mire a hibaüzenet megjelenik, a bootszektorban vagy partíciós táblában elhelyezkedő vírus már fenn is csücsül a cég gépének merevlemezén.

Ha ezek után bárki elkezd dolgozni egy megfertőzött programmal, a vírus azonnal átterjed újabb programokra. Ha az editor ugyanazon a hajlékonylemezen van, amelyen a táblázatkezelő adatállományok, akkor fertőződik a listázó program is, a fertőzés pedig tovább terjedhet mindazokra a gépekre, amelyekbe ezt a floppyt beteszik. Ha a gép a cég hálózatára is rá van kapcsolva, akkor más kollégáknak is átküldi a fertőzött segédprogramot.

A vírus minden egyes példánya másolatokat készít önmagáról, és megfertőzheti az összes olyan programot, amellyel kapcsolatba kerül. Hatása viszonylag rövid idő alatt kiterjed az egész szervezetre. Az összes fertőzött gép minden egyes fertőzött programja végrehajthatja azokat az utasításokat, amelyeket a szerző a vírusba beírt. Ha ezek az utasítások romboló hatásúak, a vírus igen széles körben okozhat felmérhetetlen károkat.

A víruskockázat csökkentése

Viszonylag nagy biztonságot csak teljesen zárt rendszerek esetén érhetünk el, ahol a bemenő adatokat kizárólag billentyűzetről gépelik be, a rendszer kimenete pedig sornyomtató. De itt is ki vagyunk szolgáltatva egyéb programhibáknak.

Amennyiben vonalon bejelentkező személyek küldenek egymásnak anyagokat, ha más azonosítási eszköz nem áll rendelkezésre, érdemes a visszahívásos módszert választani. Ekkor a bejelentkezés után a rendszer visszahívja a memóriájában tárolt számon a modemet, és így győződik meg a hívó azonosságáról. Nálunk azonban ez csak korlátozottan vagy egyáltalán nem alkalmazható, gyermekcipőben topogó kommunikációs kultúránk miatt a legfőbb veszélyteny ező továbbra is az ember marad.

A vírus nemcsak reprodukálja önmagát, hanem általában tartalmaz egy másik, romboló hatású részt is. Ez lehet egy *időzített bomba*, mely adott napon megjelenít egy bizonyos üzenetet, lehet *logikai bomba*, mely beír valahová a lemezre, ezáltal tönkreteszi az állománystruktúrát, s lehet bármi egyéb. Minden a vírusszerző fantáziájától függ. A lehetséges hatások széles skálája okozza azt, hogy sokan nem is fogják fel igazán, mi is a vírus. A *vírus* terminológiát gyakran használják rosszul.

A vírust az különbözteti meg minden mástól, hogy szaporodik, terjed és fertőz.

Igen fontos, hogy igyekezzünk korlátozni a céghez bekerülő vírusok számát, de a fertőzés lehetőségét semmiképpen sem zárhatjuk ki teljesen. Még nem ismeretes olyan megoldás, amely egy számítógépes rendszert teljesen immunissá tenne a vírusfertőzéssel szemben, de néhány megelőző intézkedéssel csökkenthetjük a fertőzésveszélyt és az esetleg keletkező kár mértékét.

Vírusvédelmi óvintézkedések

1. A kritikus adatokról és programokról tároljunk jó másolatokat.
2. Alkalmazzunk hozzáférés-védelmet, így az egyes felhasználók csak a munkájukkal kapcsolatos adatállományokat kezelhetik.
3. Számítástechnikai munkahelyeken mindig legyen egy „beléptetési pont", egy közepes kapacitású (80 MB merevlemez) színes VGA-s gép. Ide kell bemásolni a kívülről kapott anyagokat, ezen mindenki kísérletezhet, játszhat, programokat próbálgathat. Így nem veszélyeztetik az éles rendszereket vírusok behurcolásával. Persze ezen a gépen is rendszeres vírusellenőrzést kell tartani.
4. Bizonyos időközönként végezzünk teljes körű ellenőrzést, így megállapíthatjuk a gyenge pontokat, s az esetleg mégis bejutott vírusokat is felfedezhetjük.
5. Ne használjunk és ne is vásároljunk másolásvédett programterméket! Erről egyrészt nem tudunk biztonsági másolatot készíteni, másrészt az rejthet olyan trójai funkciót, amely maga is adatvesztést okozhat, tönkreteheti a rendszert.
6. Ne lépjünk külső céggel kapcsolatba hálózaton keresztül vagy hajlékonylemezes programcserével mindaddig, amíg nem ismerjük kölcsönösen egymás biztonsági intézkedéseit, gyakorlatát! (Ennek a szabálynak a figyelmen kívül hagyása miatt jött be Pécsre a Frodo programvírus, egy NDK-ból baráti alapon kapott tervezőrendszer lemezein.)
7. Az elektronikus postai kommunikációt korlátozzuk a nem végrehajtható állományokra. Válasszuk külön azokat a folyamatokat, amelyek során végrehajtható állományok érkeznek be, ezeket elkülönítve kezeljük, teszteljük. Csak megfelelő eredmények után integráljuk rendszerünkbe.
8. A biztonságtechnikai oktatás legyen a számítógépes munka alapfeltétele.
9. Alakítsunk ki szakértői csoportot, melynek tagjai a vírusokkal kap-

csolatos problémákat kezelni tudják. Lehet ez a csoport a cégnek formálisan is különálló szervezeti egysége, vagy állhat olyan hozzáértő kollégákból is, akik a cég különböző osztályain dolgoznak. Szakmai szempontból legjobb megoldás a készenléti szerződés egy vírus- vagy adatvédelemre szakosodott céggel.

10. Győződjünk meg arról, hogy a cég minden egyes alkalmazottja tudja-e teendőit, ha vírusfertőzés gyanúja merül fel. Ehhez megfelelő tervet kell kidolgozni.
11. Alakítsunk ki olyan környezetet, hogy a vírusokat gyorsan fel tudjuk ismerni. Dolgozzuk ki azt a folyamatot, amire szükségünk lesz akkor, amikor felfedezünk egy vírust a rendszerben. Legyünk tisztában azzal, hogyan lehet felélesztení (tiszta lemezről indítani) a rendszert, ha már megállapítottuk, hogy megfertőződött.

A fertőzés technikája

Számos módon fertőződhet meg egy rendszer. Fontos, hogy tisztában legyünk ezekkel a lehetőségekkel.

1. A vírus bekerülhet külső munkatársak fertőzött szoftverével, ha az kapcsolatba kerül a cég rendszerével.
2. Saját kollégánk szoftverén is behurcolható a vírus, ha otthoni gépe fertőzött. (Nem is kell, hogy tudjon róla.)
3. A szoftverkereskedőtől vásárolt program akkor okozhat vírusfertőzést, ha fertőzöttek a gyártóberendezések, ha a program másolásvédett és örökké ír a kulcslemezre vagy éppen maga a védelem tartalmaz vírus jellegű rutinokat.
4. Az információközvetítő adatbázisokból (BBS-ekből) átvett szoftverek is lehetnek fertőzöttek, ha ott továbbadás előtt nem ellenőrzik a programokat. (De a trójai programok és a logikai bombák ellen a leggondosabb ellenőrzés sem mindig nyújt védelmet!)
5. A rosszindulatú vagy sértett munkatárs készakarva, bosszúból is hozhat fertőzést a rendszerbe.
6. A vírusfertőzés tettese lehet a konkurencia vagy a gépek folyamatos javításában érdekelt szakember is. Ezért jó az átalánydíj, mert akkor a javító a minél kevesebb kiszállásban és javításban érdekelt. (Gondoljunk a kínai császár orvosának esetére, aki csak akkor kapott pénzt, amikor gazdája egészséges volt...)

Munkahelyi alapszabályok

Nagyon fontos a számítógépet kezelők és az őket első menetben segítő munkatársak alapos felkészítése. Az egyes szintek tevékenységének

összehangolásával érhetjük csak el, hogy a vírusfertőzés észlelését gyorsan kövessék az elhárító intézkedések. Ugyanakkor van néhány olyan intézkedés, amely minden számítógéppel foglalkozó munkahely számára elengedhetetlen.

1. BIZTONSÁGI MÁSOLAT

A számítógépes rendszerek biztonságos működése elképzelhetetlen a másolatok folyamatos készítése nélkül. Programok, illetve adatállományok elvesztése, tönkremenetele esetén napok – vagy nemritkán –, hónapok munkáját takaríthatjuk meg, ha megvannak a másolatok. A vírusfertőzés veszélye még inkább indokolja a mentések rendszeresítését.

Miután megállítottunk egy fertőzési folyamatot, kiirtottuk a vírust az összes szoftverből, elkezdjük az elveszett vagy sérült állományokat a másolati példányokból pótolni. Legyünk nagyon óvatosak, nehogy éppen ekkor hozzuk vissza a vírust a rendszerbe, mert a másolatok is potenciális vírusrejtekhelyek! Az adott vírus viselkedésével pontosan tisztában kell lennünk, mielőtt megkezdenénk bármely mentett anyag visszatöltését. Előbb ellenőriznünk kell az összes mentett állományt, nem tartalmaznak-e vírust. Ne használjunk mentett állományt addig, amíg nem bizonyosodtunk meg arról, hogy vírusmentes. Akkor tehetünk csak kivételt, ha az adott állományt semmilyen más módon nem tudjuk visszatölteni a rendszerbe. Legyen legalább két eltérő időpontban készült mentésünk. Ezek összehasonlítása is sokat segít a vírustalanítási munkában!

2. VÍRUSKAPUK ELLENŐRZÉSE

A vírusok terjedhetnek egy rendszeren belül a felhasználók közt vagy egyik rendszerről a másikra. Általában külső forrásból származnak a vírusok, mert Magyarországon nagyon ritka eset, hogy a vírust cégen belül írják és indítják útjára. Vírust tévedésből írni nem lehet, kívülről viszont behozhatjuk akaratlanul is (miközben fogalmunk sincs arról, hogy a program vírusgazda). Hasonlóan bootvírust is gyanútlanul behozhatunk a rendszerbe: máshol formázott floppyval.

Ha egy cégnek nagy a szoftvermozgása, szinte elképzelhetetlen megakadályozni egy kívülről jövő fertőzést, hiszen a számítástechnikában a programok cseréje mindennapos. Előfordulhat például, hogy az egyik dolgozó munkáját az otthoni (fertőzött) gépén folytatja, a módosított változatot másnap visszaviszi, s ezzel már be is jutott a vírus a cég rendszerébe!

A vírusvédelem szempontjából legnagyobb biztonságot jelentő megoldások sajnos alaposan korlátozzák a rendszerek működését, és ilyenkor

éppen azt nem aknázhatjuk ki, ami a számítástechnikai rendszerek egyik nagy előnye: az univerzalitást. (Azt, hogy ugyanaz a gép az egyik pillanatban valamelyik programmal rajzokat tervez, később szövegszerkesztő lesz belőle, majd pedig kalkulációs programot futtat.)

Első zsákutca: elszigetelt rendszerek használata

Mivel a vírusok csakis információcsere, illetve kommunikáció útján kerülnek be az egyes rendszerekbe, kirekeszthetjük őket, ha teljesen elszigeteljük a rendszereket, a felhasználókat. Hálózatba kapcsolt rendszerek esetén a hálózaton keresztül is terjedhet a vírus. (Különösen az olyan rendszerekben, amelyek tervezésekor még nem vették figyelembe a vírusokkal kapcsolatos óvintézkedéseket.)

Ha leválasztjuk rendszerünket a hálózatról, kiküszöböljük a hálózaton keresztüli fertőzésveszélyt. Ha egy cég hálózaton keresztül kapcsolatban áll más cégekkel (egyetemekkel, intézetekkel), a vírus bejöhet a céghez a hálózaton át. Ha lekapcsolódunk erről a külső hálózatról, megszűnik az egyik veszélyforrás. (Egyelőre sajnos nem nagyon kell félni a kiterjedt adatkapcsolatoktól, mert az egykori szocialista országokban ezeket a lehetőségeket tudatosan igen szűkre szabták, s a telefonvonali rendszer sem igazán volt alkalmas adatkommunikációra.)

E hálózatról történő leválasztás főleg olyan rendszereknél ajánlott, ahol érzékenyek a programok és az adatok. Sok esetben azonban értelmetlen, mert a cég hatékonyságát növelő kényelmes program- és adatcserétől fosztjuk meg vele magunkat.

Felmerül az ötlet, hogy korlátozzuk a külső hozzáférést a cég rendszeréhez – őket általában kevésbé érdekli a mi rendszerünk biztonsága. Jobb, ha nem engedélyezzük számukra végrehajtható állományok küldését, illetve a belső rendszerek teljes interaktív hozzáférését. Ez is magától adódó rendszabály, de így teljesen elveszíthetjük a szolgáltatásaink iránti külső érdeklődést. Ugyanakkor például az országos pénzügyi rendszereknél vagy a banki, bankpénztári helyi rendszereknél már az adatbiztonság miatt is ez a célravezető megoldás.

A vírusok terjedésének fő „útvonala" Magyarországon az, hogy programokat viszünk át floppyn egyik személyi számítógépről a másikra. Kiadhatunk tehát olyan utasítást, hogy tilos programok és bootolható lemezek hazavitele, illetve behozatala a munkahelyre. (Abban is biztosak lehetünk azonban, hogy ezt a szabályt garantáltan megszegik!) Kötelezhetjük viszont alkalmazottainkat, hogy otthon is használják ugyanazt a vírusfelismerőt, amit a cégnél. Ez a megoldás bátran ajánlható mindenkinek. Olcsóbb beszerezni a beosztottak magángépére az antivírus programokat, mint egy céget a kiterjedt vírusfertőzéstől utólag kitakaríttatni!

Ne feledjük: kidolgozhatunk elméletileg bármilyen jó szabályzatot, fenyegethetjük az alkalmazottainkat „lenyakazással", ha a szabályok a gyakorlatban betarthatatlanok vagy kényelmetlenek, akkor garantáltan megszegik azokat.

Második zsákutca: lebutított rendszerek használata

Ahhoz, hogy terjedni tudjanak, a vírusoknak meg kell fertőzniük más végrehajtható rendszerkomponenseket. Erre alapozva terjedésüket úgy is megakadályozhatjuk, hogy kiszűrjük ezt a lehetőséget a rendszerből. Bizonyos esetekben korlátozhatjuk vagy le is tilthatjuk a programok cseréjét, módosítását. Fejlesztő rendszerek esetén természetesen lehetetlenség, hogy ne készítsünk vagy ne vegyünk át új programokat. Itt ez a módszer nem használható.

Számos cég rendszerében találhatók olyan munkaállomások, személyi számítógépek, amelyeket soha nem használnak programfejlesztésre. Ha megakadályozzuk új programok bevitelét, ezzel egy víruskaput le is zártunk. Erre a célra néhány szemfüles cég az 1991-es hannoveri Cebit kiállításon kulcsra zárható *floppyajtókat* kínált. (Akik ugyanis korábban hasonló megfontolások alapján floppy nélküli munkaállomást vásároltak, előbb-utóbb megbánták ezt a lépést, és utólag építették be a floppyt.)

Ez kicsit goromba eljárás, de hatékony. Ahol lehetséges, ne engedélyezzük a gép használóinak új programok bevitelét. Ezt csak az arra kijelölt munkatársak tehessék meg, ők is csak úgy, hogy installálás előtt minden rendelkezésre álló módszerrel ellenőrizzék az új termék vírusmentességét. Hasonló módon célravezető engedélyezni, hogy még a kedvenc lopott játékprogramok és szövegszerkesztők is bekerülhessenek a rendszerbe, ha átestek a hivatalos vírusvizsgálati eljáráson. Ezáltal ugyanis nem kényszerítjük a dolgozót a biztonsági szabályok megszegésére. Sajnos a munkahelyi vezetők nagyon kevés helyen értik meg ezt a szempontot.

A munkaállomások kialakítása arra ösztönöz, hogy a saját igényeit legjobban ismerő felhasználó a céljainak leginkább megfelelő programokat önállóan fejlessze ki. Hosszú távon tehát nem érdemes túl sok rendszerre kiterjeszteni a fenti korlátozásokat. Össze kell mérnünk az esetleges fertőzés által okozott kárt azzal a veszteséggel, ami a felhasználó egyéni kezdeményezéseinek letöréséből származik.

Egyirányú utca: a közösködés szükséges, de veszélytényező

Közös programbankoknak azokat a helyeket tekintjük, ahol olyan programok találhatók, amelyeket többen is használnak. Ez lehet például a központi gép tárolóegysége vagy a hálózati szerver merevlemeze,

amelyhez sokan hozzáférhetnek. Ezek a közös területek sokkal veszélyesebbek a vírusok terjedése szempontjából, mint az egyéni programok tárolási helyei. Ha egy ilyen, többek által használt program megfertőződik, annak beláthatatlan következményei lehetnek.

Közös használatú új programokat igen alapos ellenőrzés után legyen szabad csak bevinni a rendszerbe. Érdemes megvizsgálni ezeket a programokat minden olyan eszközzel, amelyet az eddig ismert vírusok felismerésére fejlesztettek ki.

Ennél is szigorúbb ellenőrzést ajánlatos bevezetni akkor, ha a közös programokkal dolgozó felhasználók különlegesen érzékeny más programokat, adatokat kezelnek. A beviendő új programokat különálló rendszerben érdemes először tesztelni, itt is észrevesszük, hogy gyanúsan viselkednek-e.

A közös területek hasznosak is lehetnek a vírusok felismerésekor és kezelésekor. Tekintsük azt az esetet, amikor a géphasználók többsége a központi szerverről futtatja a szoftvert, a szerverhez pedig nincs írásjogosultsága. Mivel ezt a szoftvert nem aktualizálják rendszeresen, a vírusok nem fognak gyorsan terjedni. Ha az egyik közös program megfertőződik, viszonylag egyszerű kicserélni egy vírusmentes változatra (vagy teljesen kiirtani). Sokkal bonyolultabb lenne a programot különkülön minden rendszerben, egyenként megvizsgálni.

Amikor nincs visszaút: a szoftver-előállítás és -sokszorosítás

Ebbe a kategóriába azokat a rendszereket soroljuk, amelyek a belső fejlesztésű programokat előkészítik a cégen belüli szétosztásra, illetve a külső megrendelőkhöz kiszállításra. Ha egy ilyen rendszerbe vírus kerül be, az átterjed a cégen belül és a cég vevőinél használt programokra. Cégen belül mindent fertőz, kifelé még a cég hírnevét is tönkreteszi.

Előfordul, hogy egy cég fertőzött programterméket szállít megrendelőinek. Például évekkel ezelőtt, amikor egy NSZK-ból származó Disk Manager gyári programmal Disk Killer programot terjesztettek – tudtukon kívül. Másfél éve Győrben szállított ki megrendelőinek egy szoftverfejlesztő cég Stoned/Marijuana vírussal fertőzött terméket. A cég korrekt módon, szakemberekkel végigjárta megrendelőit és kitakarította a vírusokat, szerencsére a könyvelői program terjesztése során vírusfertőzés nem következett be. A fertőzés oka egy írásvédelem nélkül használt gyári szoftverlemez volt, amelyet valahol bemutatóra is installáltak egy ismeretlen gépen, s onnan hurcolták be a fertőzést. Utóbb ennek megszüntetése jelentős veszteséget okozott.

Az ilyen rendszerek védelmének megkülönböztetett figyelmet kell szentelnünk, hiszen az itt fellépő fertőzések igen komoly következmé-

nyekkel járnak. A változtatások szigorított ellenőrzése jelentheti itt a megoldást.

Használjuk gyakran a vírusellenőrző programokat! Multitasking rendszerekben a háttérben állandóan futhat egy vírusdetektor. Ajánlatos a vírusdetektort legalább egyszer kötelezően lefuttatni, mielőtt új állományokat hoznánk be a rendszerbe, illetve minden olyan esetben is, amikor kiviszünk állományt a rendszerből.

Ezeket a rendszereket szigorúan el kell különíteni az összes többi nem gyártó–sokszorosító rendszertől, és csak a legszigorúbban ellenőrzött állományok kerülhessenek be a rendszerbe. Tárgykódok helyett installáljuk inkább a forráskódot, és ezt egy megbízható fordítóval fordítsuk le helyben. Ahol a legsúlyosabb következményekkel járna a vírusfertőzés, ajánlatos fordítás előtt a forrásszöveget is ellenőrizni, nehogy innen kerüljön be egy vírus. Ha elkerülhetetlen, hogy tárgykódot installáljunk, ennek eredetét előzőleg meg kell vizsgálni. Személyi számítógépre például csak olyan eredeti, írásvédett lemezről szabad installálni, amelyet vírusmentesnek találtunk.

Végül még egy tanács a programokat kibocsátó cégeknek: ne használjanak a kész, ellenőrzött termék sokszorosítására számítógépet! Kaphatók olyan lemezmásolók, amelyek operációs rendszer nélkül működnek, tehát pusztán fizikai másolással végzik a lemezek sokszorosítását, ezért a másolat csak akkor lehet vírusos, ha a mintalemez is az volt. Az *Alaplap* olvasói közül mindig akad néhány, aki felháborodottan küldi vissza a mágneslemez mellékletet, hogy vírus van rajta. Sajnos nehéz őket meggyőzni arról, hogy a fertőzés akkor történt, amikor a lemezt saját számítógépükön elindították, mert ha a mi eredeti mintalemezünkön vírus lett volna, akkor annak mind a 11 000 példányra rá kellett volna kerülnie. Sokszorosításs közben a fertőződés amúgy is fizikai lehetetlenség.

Hogyan ismerjük fel a vírusfertőzést?

Az eddig ismertetett módszerek a teljes elszigetelés kivételével csak bizonyos mértékben szűrik ki a vírusfertőzés lehetőségét, a megelőző intézkedések ellenére is bekerülhetnek vírusok a rendszerbe. Nincs tökéletes védelem. Egy katonai biztonságtechnikai szakértő egyszer bemutatta „szuperbiztos" rendszerét, amelynek bemenete kizárólag a billentyűzet, tehát csak begépelni lehet minden információt és programot, kimenetei pedig a nyomtatók. Szerinte ide soha nem juthat be vírus. Arra a kérdésre, hogy mi van akkor, ha egy elégedetlen munkatársa begépeli és a rendszeren belül fordítóprogrammal lefordítja a víruskódot, a válasz csak hümmögés volt... Abszolút biztonság tényleg nem létezik!

Tudatában kell lenni a veszélynek és idejében felismerni a vírus jelenlétét. A vírusfertőzésnek ugyanis vannak gyanús jelei. Ezek bármelyikét tapasztalva, ki kell derítenünk az okot. Ha az vírusnak bizonyul, meg kell tennünk az elhárító lépéseket. Vannak olyan tünetek, amelyek csak egyetlenegyszer, a vírus betelepedésekor lépnek fel, de vannak olyanok is, amelyek a vírus terjedését jelzik. Fontos azonban, hogy tudatában legyünk annak is, hogy a jövőben megjelenő új vírusok esetleg nem produkálják ugyanazokat a tüneteket. A felhasználóknak mindig a szokásostól eltérő jelenségeket kell vadászniuk, és ezekről kell a vírusokkal foglalkozó biztonságtechnikai szakembereket sürgősen értesíteniük.

1. Szokatlan viselkedés

A szokásostól eltérő viselkedés általában nem vírusfertőzés következménye. Sokkal mindennapibbak az okok: szoftverproblémák, felhasználói hibák, hardverhibák és hasonlók. Lehetőleg kerüljük el a vakriasztásokat. Valóságos fertőzés észlelésekor azonban szükséges a gyors intézkedés.

2. Nőnek az állományok, megváltoznak a könyvtárbejegyzések

Váratlan, indokolatlannak tűnő változások a végrehajtható állományok hosszában és időbejegyzésében. Esetleg egyik pillanatról a másikra új állományok jelennek meg, amelyek lehetnek hidden, system vagy read-only attribútumúak is. Ez utóbbiakat azonban nem szabad összetéveszteni az egyes programok elszállása után maradó szeméttel. (Az Autocad például hidden és system állományokat hagy maga után, ha rendszerhiba miatt leáll.)

3. Lassabban indulnak, hosszabb ideig futnak a programok

Néhány vírus, például a Potyogósnak és a Péntek 13 egyes változatainak fellépésekor a gép betöltési ideje, a programok futási ideje jelentősen meghosszabbodik. A Moxla vírus pedig tisztázatlan eredetűnek látszó paritáshibát szimulál a memóriafelfrissítés manipulálásával.

4. A programok írásvédett eszközökre akarnak írni

A vírusok korábbi nemzedékei még nem voltak elég intelligensek, és ha írásvédett floppyval találkoztak, akkor is makacsul megpróbáltak „rámászni".

5. Memória csökkenése, hibás lemezfelületek szaporodása

Egyes vírusok úgy rejtik el magukat, hogy a FAT-ban egyes szektorokat hibásnak jeleznek. Különösen gyanús, ha egymás után több leme-

zen tapasztaljuk a rossznak elkönyvelt felületek növekedését. Ha ezek a területek többszörösei egymásnak, akkor biztosan vírus van a háttérben.

6. Végrehajtható állományok hirtelen eltűnése

7. A rendszer automatikus újraindulása

A számítógép teljesen váratlanul – véletlenszerűen vagy viszonylag állandó időközönként – újra betölti a rendszert, megszakítva az előzőleg jól működő programok futását.

8. Szokatlan dolgok jelennek meg a képernyőn

Például a kép egyes részei elkezdenek futni, táncoló labdák, furcsa üzenetek és jelek láthatók, esetleg a kép eltűnik (Phantom).

9. Lemezek címkéje (label) megváltozik

10. Hálózati rendellenességek

A lokális hálózatokon vagy más összeköttetéseken keresztül szokatlan dolgok érkeznek. Különösen gyanús, ha ugyanannak az adatnak több másolata jön egyszerre.

11. A végrehajtható programelemek megváltoznak

12. Hibátlan programok lemerevedése és kilépése hibaüzenettel

Az egyszerű vírusok felismerésére hatékony módszer lehet, ha figyeltetjük a rendszerben lévő állományok hosszát. Vannak ugyanis az operációs rendszernek fontos állományai, amelyek csak ritkán vagy egyáltalán nem szoktak módosulni. Ha ezek megváltozását észleljük, azonnal értesítsük vírusszakértőnket, mert ilyen esetekben az idő nagyon fontos tényező. (A tisztességes forgalmazó felhívja a szoftver használójának figyelmét arra, ha programja setup műveletkor önmagába ír bele. Ezt ugyanis a változásdetektorok jelzik, bár vírusnak nyoma sincs.)

A többfelhasználós nagy rendszerek ilyen jellegű ellenőrzése központi rendszerprogramozó csoport feladata. Ők megadott időközönként lefuttatják a programot, így ellenőrzik, hogy történt-e változás a közös operációs rendszerben vagy annak segédprogramjaiban. A vírusok gyakran éppen a privilegizált használókra terjednek át, ezáltal pontosan azokat a rendszerelemeket veszélyeztetik, amelyek csak a jogosultságvédelem legmagasabb szintjéről érhetők el. Az Állami Számvevőszék Magyarországon az elsők között hozott létre saját vírusriadó csoportot a cég szá-

mítóközpontjaiban előforduló problémák tisztázására. Hasonló riadótervet készített az Állami Népességnyilvántartó Hivatal. Más cégek a vírusmentesség garantálására gyakran külső vírusszakértőket bíznak meg átalánydíjas szerződéssel.

A vírusfelismerő programok lefuttatásának gyakorisága függhet attól, hogy milyen a végrehajtható állományok mozgása és a a rendszerek közti információforgalom. A hálózat nélküli számítógépekre floppyval felkerülő vírusok terjedési ideje nehezen prognosztizálható, néha napokba, hetekbe, sőt hónapokba is telhet, amíg megfertőz egy nagyobb szervezetet, más esetekben pedig a kór terjedése robbanásszerű. Ha az alkalmazási szoftver kibírja, legcélszerűbb rezidens vírusvédő programokat vagy hardveres vírusvédelmi kártyát alkalmazni.

MILYEN A JÓ VÍRUSVÉDELEM?

Minden kaput becsukni

Az alábbiakban a vírusok észlelésének és kiszűrésének egyik megoldási módját, egyik eljárását ismertetjük. Ez csak egy elképzelt minta, amitől eltérő struktúrák ugyanilyen jók vagy akár sokkal jobbak is lehetnek. A tesztelési fejezetekben néhány ismert vírusvédő programot konkrétan is bemutatunk, minősítünk.

Alapfeladatok

1. A programnak tartalmaznia kell az ellenőrizendő állományok listáját. MS-DOS, PC-DOS esetén például ide kell sorolni az összes .COM és .EXE kiterjesztésű állományt, továbbá azokat az állományokat, amelyek a CONFIG.SYS-ben mint meghajtók szerepelnek, a CONFIG.SYS-t magát, valamint minden egyebet, ami végrehajtható (batch állományokat, GEM .APP programokat stb.).
2. A programnak tartalmaznia kell a kimentett boot-szektor és partíciós tábla adatait. A program írásakor különösen kell ügyelni arra, hogy nem a megszokott kiosztású partíciós táblával dolgozó számos DOS-verzió használatos (Tandon, Compaq, HP-Vectra, DR DOS, MS–DOS béta 5.0). Ugyancsak illik elmenteni a SETUP RAM állapotát is.
   Képzeletbeli ideális programunk ezekkel az állományokkal a következőket végzi el:
   — Indításkor összehasonlítja a kimentett állapotot a setup RAM, a boot és a partíciós tábla állapotával. Ha változást talál, riaszt, és kérésre az eredeti állapotot helyreállítja. (Éppen ezért nem alkalmazható másolásvédelem a vírusvédett gépeken, de a hozzáférésvédelemnek is megvannak a maga illemszabályai.)
   — Meghatározza az időpont- és dátumadatokat, illetve az állományhosszt az operációs rendszerből. Utána az operációs rendszer szintje alá nyúlva megkísérli ismét meghatározni a hosszúságot (counting byte). Erre azért van szükség, mert ha bizonyos új típusú vírusok jelen vannak a memóriában, akkor a DOS szabályos kérdezéskor az eredeti hosszt adja vissza. Ugyancsak más méretet kapunk, ha valamilyen on-line tömörítőprogramot alkalma-

zunk (pl. Stackert). Itt azonban ez már nem hiba. A vírusokat eddig még nem készítették fel e programok átverésére, az SDIR itt sok esetben ad vissza valós értéket.

— Ha szükségesnek tűnik, végigolvassa a teljes állományt és ellenőrző összeget képez. Az állományok fontosságától és a program sebességétől függ, hogy hány állományt vizsgálunk meg ilyen alaposan. Befolyásolja természetesen az is, hogy a felhasználó mennyi időt szán erre az ellenőrzésre.

— Az állomány jellemzőit (időpont, dátum, hossz, esetenként CRC vagy más ellenőrző kód) összehasonlítja azzal az adatbázissal, amelyet a legutóbbi programfutás generált. Ha az utolsó programfutás idején nem volt még ilyen állomány, illetve ha az adatbázisban talált információ különbözik a most feljegyzettektől, a program megjegyzi, hogy az állomány új, illetve módosult.

3. A kijelölt állományok ellenőrzése után végigolvassa a rendszer további végrehajtható elemeit, CRC vagy más ellenőrző kódjukat összehasonlítja az adatbázisban található értékekkel és feljegyzi az esetleges változásokat.
4. Miután minden egyes elem ellenőrzése befejeződött, a program aktualizálja az adatbázist, a legközelebbi alkalommal ezekkel az értékekkel fog majd összehasonlítást tenni, a felhasználónak pedig átadja a futási eredményeket.
5. Ezen információk birtokában a felhasználó már el tudja dönteni, hogy a változások között vannak-e gyanúsak, amelyekről a szakértőket értesítenie kell.

Sajnos ez a módszer sem nyújt teljes biztonságot, mert a rafináltabb vírusok úgy változtatják meg a végrehajtható elemeket, hogy az időpont, a dátum, a hosszúság, illetve a CRC adatai mind ugyanazok maradnak. Viszont csak olyan elemeket tudnak megváltoztatni, amelyeket nemrégiben legálisan is megváltoztattunk. Közvetlenül kifejthetik hatásukat az adatbázisra is, ekkor pedig nem mutathatók ki. Szerencsére ilyen vírusellenes program ellen fellépő vírust még nem fejlesztettek ki, de a lehetőséggel számolni kell.

Érdemes több vírusfelismerő programot is lefuttatni. Ha a vírust nem ismeri fel az egyik, a következő esetleg megtalálja. Természetesen csak eltérő vírusismerettel és eljárással dolgozó programok futtatásának van értelme. Például lefuttatunk egy adatbázissal dolgozó változásfigyelő rendszert, majd utána a TBSCAN és SCAN vírusazonosítóra kereső programot, és végül a nemzeti specialitásokat ismerő programmal zárjuk a sort. A felhasználóknak állandóan készenlétben kell lenniük. Fertőzés esetén pedig kész intézkedési terv alapján kell a károkat minimálisra szorítani.

Ha jön a vírus...

Igen fontos, hogy amikor detektáló programunk vírust jelez, el tudjuk dönteni, hogy valóban vírussal állunk-e szemben. A vakriasztások száma jelentősen csökkenthető az alábbi szempontok figyelembevételével:

a) Ismerni kell az általunk használt szoftverek korlátait. Például aki Above EMS meghajtót alkalmaz, annak az izraeli TNT programcsomag Bootsafe programja ismeretlen vírus jelenlétét jelzi a memóriában, pedig az nem vírus, hanem az Above meghajtóprogramja. Hasonlóképpen a McAfee-féle SCAN a DOS 3.2x verzió memóriavizsgálatakor az 1701/1704 programvírust, DOS 3.3x esetén pedig a Yankee Doodle jelenlétét észleli. Ha nem találjuk meg a programban is ezeket a vírusokat, akkor a jelzés csak vaklárma.

b) A vírusjelzés csak ott tekinthető reálisnak, ahol a vírus a természetének megfelelő környezetben van. Például ha vírusfelismerő programunk .BAT vagy .TXT állományban „talált" vírust, az ott nyilvánvalóan nem lehet. Nem találhatunk továbbá bootvírust az állományokban és fordítva: programvírust a boot-szektorban. Ehhez McAfee aktualizált víruslistája és *Vírushatározó* kötetünk ad részletes támpontokat.

c) Ha víruskereső programjainkkal nem találtunk vírust, akkor sem jelenthetjük ki, hogy rendszerünk vírusmentes. Mindössze annyit mondhatunk, hogy ismert vírust nagy valószínűséggel nem tartalmaz. A vírusmegelőző programok használata nem mentesít a gyanús jelek állandó figyelésétől és a szokatlan jelenségek okainak kivizsgálásától.

A szabadjára engedett vírusok igen gyorsan képesek megfertőzni a programokat. Éppen ezért az újabb generációs vírusokba már bizonyos fékeket építettek be, hogy gyors terjedésük ne leplezze le őket túl korán. (Például nem minden futtatás után fertőznek, vagy időnként változtatják terjedési algoritmusukat.)

A számítógépet használók között mindenütt vannak „sztárok", akik a többieknél aránytalanul nagyobb információcserét bonyolítanak le. Általában ők tesztelik az új szoftvereket, mielőtt azokat mások is elkezdenék használni, ők azok, akik az új dolgokat kipróbálják. A többfelhasználós rendszerekben is általában nekik van privilegizált joguk, minden állományhoz hozzáférhetnek, s azokat módosíthatják. Ha hozzájuk jut el a vírus, az különösen gyorsan terjedhet. Éppen ezért célszerű őket kiképezni a helyi víruscsoportban való közreműködésre.

A vírusvédelmi terv

A vírus felismerését követően minden másodperc, amelyet azzal töltünk, hogy eldöntsük, mi is a teendőnk, újabb terjedési lehetőséget je-

lenthet a vírusnak. Fontos tehát, hogy már a fertőzés megjelenése előtt összeállítsuk a vírusvédelmi terveket. Ezekben figyelmünknek a következőkre kell kiterjednie:

1. A kríziscsoport tagjainak kijelölése.
2. A kríziscsoport tagjainak felkészítése, szakirodalommal, programokkal való ellátása.
3. Az intézkedési hatáskör kijelölése. Például az, hogy ha magukban nem birkóznak meg a feladattal, külső szakembereket vonhassanak be a vírusveszély elhárításába és az adatmentésbe.
4. Az informatikai szükségállapot elrendelése, a fertőzött rendszerek izolálása.
5. A fertőzött rendszerekkel kapcsolatban annak kiderítése, hogy honnan és hogyan jutott be a vírus.
6. A fertőzésben érintett partnerek tájékoztatása, hogy mit kell tenniük a vírus terjedésének megállítására.
7. A vírusok azonosításának és a fertőzés megszüntetésének módja, megjelölve az ehhez szükséges speciális szoftvereket is.
8. A vírusfertőzéskor és utána megtett ellenintézkedések hatásának regisztrálása.

A 7. pontot természetesen rugalmasan kell kezelni. Egy nagyhírű amerikai cég magyar leányvállalata például ragaszkodott a Dr. Solomon's Antivirus Toolkit kizárólagos használatához. Ez azonban a Magyarországon és Kelet-Európában honos bacik ellen édeskeveset ér...

Miután a terv alapján felismertünk és azonosítottunk egy vírust, és megtettük a megfelelő intézkedéseket, hogy megállítsuk továbbterjedését, fertőtleníteni kell a rendszert, s utána ismét a megszokott módon dolgozhatunk tovább. Ehhez el kell érnünk, hogy az egész rendszerben sehol ne maradjanak fertőzött elemek, amelyek később aktivizálódhatnak. Fertőzött elem lehet minden olyan számítástechnikai elem, futtatható program, boot-rekord, partíciós tábla, rejtett lemezpálya, sőt hardver is, amelyben a vírus meg tud bújni és ahonnan újra elszaporodhat.

A mentesítés alapszabályai

— A rendszer minden egyes fertőzött elemét fertőzetlennel kell helyettesítenünk.
— Minden olyan részt helyre kell állítanunk, amelyet a vírus megrongált.
— A mentesítési munkák során meg kell akadályozni a vírus aktivizálódását és továbbhurcolását.
— El kell kerülni a vírus újbóli beléptetését a rendszerbe.

A vírusirtó programok

Ha alaposan megismertünk egy vírust, magunk is írhatunk olyan programokat, amelyek alkalmasak eltávolítására, bár kényelmesebb beszerezni kész programokat. Az ilyen jellegű programok egyik csoportja azt ellenőrzi, hogy van-e vírus a végrehajtható elemekben. Egy másik programcsoport pedig megpróbálja úgy megszüntetni a fertőzést, hogy az adott részt korábbi, fertőzetlen formájában állítja vissza.

Az *eltávolító* programok hasznosságát e könyv szerzői nem ugyanúgy ítélik meg, mint az USA szoftveres cégei. Mi úgy látjuk, hogy a fertőzést, ha lehetséges, ki kell „piszkálni" a gépen lévő elemekből, és az eredeti állapotot víruseltávolító programok segítségével helyre kell állítani. Az amerikai felfogás pedig az, hogy az elmentett fertőzetlen állományokkal kell felülírni a fertőzötteket, s csak akkor kell a helyreállító programokhoz fordulni, ha ilyenünk nincs. Ennek egyik oka inkább érzelmi színezetű. Ugyanis az USA-ban azt mondják, hogy a helyreállított program szinte soha nem lesz bitszintig azonos az eredetivel, a gyártó garanciái pedig csak az eredeti állapotra vonatkoznak. A másik indoklást az IBM és néhány nagy szoftvergyártó cég szokta felhozni, hogy a programok megváltoztatása sérti az eredeti forgalmazó szerzői jogait, miként a programok átkódolása is, amilyet a Philip Katz-féle Pklite csinál. Szerintük a vírus eltávolítása egy végrehajtható állományból nagyon bonyolult és nehéz feladat, s még jól megírt programok esetén is fennáll a veszély, hogy a helyreállítás nem lesz tökéletes.

Miután készen van a helyreállítás, és az összes fertőzött vagy módosult elem ismét eredeti, fertőzetlen állapotában található, egy ideig még résen kell lennünk. Újrafertőzést előidéző gócok lehetnek még az ellenőrzés alól kicsúszott, például fiókban rejtegetett „maszek" floppyk, a munkatársak megfertőződött otthoni gépe stb. A veszély főleg akkor áll fenn, ha nem sikerült felderítenünk a fertőzés forrását.

Néha érdemes egyetlen vírust felismerő és kitakarító programot is beszerezni (vagy írni), amely a tömeges fertőzés ellen gyors és hatásos. Az egyedi detektorokat és killereket lehetőleg szabadszoftverként terjesszük, mert mindenkinek fontos, hogy a vele adatkapcsolatban állók is vírusmentes környezetben dolgozzanak. Egy időre be is építhetjük ezeket a programokat a mentési rendszerbe, így meg tudunk bizonyosodni a mentendő állományok fertőzetlenségéről. Elhelyezhetjük ezeket a programokat a hálózat megfelelő helyein és a közös állományokat használó rendszerekben is. Mivel az ilyen ellenőrzés folyamatos pluszmunkát jelent, kompromisszumot kell kötnünk a rendszer hatékonysága és fokozott biztonsága között. Magunknak kell eldöntenünk, hogy

meddig maradjanak benn a rendszerben azok a speciális ellenőrző programok, amelyek mellékhatása, hogy lassítják annak működését.

Víruskapuk

Hol támadhatja meg számítógépes rendszerünket a vírus? Mi történik akkor, amikor egy szoftvert elindítunk? Melyek azok a részek, amelyeket nem ellenőriznénk?

Nézzük át röviden azokat a rendszerfolyamatokat, amelyek MS-DOS, PC-DOS, Tandon MS-DOS, Compaq DOS esetén egy vírus bejutásához vezetnek.

1. Kapcsoljuk be a gépet, azután pedig...
2. Floppyról indítunk! De vírusmentes-e a lemez? Ha újra indítjuk, akkor az ellenőrzött rendszerlemezt használjunk, vagy egy másikat?
3. Winchesterről indítunk. Milyen kód töltődik be? Nem vírusos-e a partíciós tábla és a bootrekord? Vírusosak-e a winchesteren lévő programok?
3. Programot futtatunk. Ez olvas a lemezről. Mit olvas most éppen? És előtte mit olvasott? (Dilemmáink ugyanazok, mint fent!)
4. Mikor néztünk rá utoljára a CONFIG.SYS-re? És az AUTOEXEC.BAT-ra? Esetleg éppen előttünk szórakozott valaki a gépen... Nem írta-e át? (Merevlemezes gépek esetén a Disk Manager vagy a Stacker segédállományainak kihagyása a CONFIG.SYS-ből és az AUTOEXEC.BAT-ból frenetikus hatású lehet és a víruspusztítással vetekedő adatvesztést okozhat.)
5. Most vásároltunk egy vadonatúj programot. Hazavittük a boltból. Mit tudunk erről a programról? És a cégről, amely kifejlesztette? És arról, hogy mit művel a másolásvédelme?
6. Épp most vettünk át egy új programot a fejlesztőjétől. Mit tudunk erről a programról? Milyen meglepetéseket rejtett el benne a fejlesztő, hogy további jövedelmét biztosítsa? Vírusmentes környezetben dolgozott-e? És mit csinál a másolásvédelme?

Listánk természetesen nem lehet teljes. Csak mintaként szolgál, hogy kell tételesen végiggondolnunk azokat a veszélyforrásokat, amelyek ránk leselkednek. A ? mindenhol arra figyelmeztet, hogy ez olyan pont, ahol fennáll a vírus belépésének veszélye.

Ami a gép bekapcsolásakor lejátszódik

Gondolkoztunk-e már azon, mi történik akkor, amikor az IBM PC-t bekapcsoljuk? A továbbiakban (??) jelöli a baci támadáspontjait.

Feszültség alá helyeztük a rendszert. Ekkor a gépen a ROM BIOS-ból

egy POST (Power On Self Test) nevű kódrész kezd futni. Ez ellenőrzi a memóriát és a gép többi elemét, a perifériákat, majd átadja a vezérlést valamelyik „kijelölt" ROM-nak az I/O csatornában. Ha ezek a részek problémamentesen lefutnak, a vezérlés visszakerül a POST-hoz. Miután a POST befejezte munkáját, keresi a floppyt a floppymeghajtóban. Ha nem találja, akkor a merevlemezen keresi a törzsrekordot. Ha ezt sem találja, akkor elindítja a ROM-jában található Basic interpretert. Ha ilyen nincs, márpedig a nem IBM gépekében nincs, akkor kéri a bootlemezt a rendszerrel, és ha azt betettük, megnyomhatunk neki egy gombot. Ha a ROM Basic helyén egy RAM-ba égetett operációs rendszer van, akkor megkönnyebbülhetünk, hogy legalább ez nem fertőződik!

Az első olyan hely, ahol programokat olvasunk be a rendszer RAM tárába, a törzsrekord (master boot), illetve a floppy. Egészen addig minden kód, amely futott, a ROM-ból jött. Mivel ezek a ROM-ok megbízható forrásból származnak, feltételezhetjük, hogy nem vírusfertőzötten installálták őket. A ROM-ok természetüknél fogva csak speciális készülékekkel írhatók, ilyen pedig nincs a PC-nkben. Ezek csak úgy módosíthatók, ha tudtunkon kívül kicserélik azokat.

Amikor a rendszer merevlemezről indul, két kódállományt használ. Az első a betöltő törzsrekord, a *master boot* rekord(??), mely arra vonatkozó információt tartalmaz, hogy melyik rendszerbetöltő *system boot* rekordot(??) kell beolvasni és futtatni. Ez az aktív partíció. (Több rendszerbetöltő rekord is van, minden partícióhoz tartozik egy, törzsrekord azonban összesen csak egy lehet a rendszerben.)

A merevlemezek bootrekordjai különbözőek, általában azonban egy teljes sávot elfoglalnak, ami elég nagy terület, és annak legnagyobb részét nem is használjuk. Ez pedig kapóra jöhet a vírusok elrejtőzéséhez. A vírus, ha ügyes, ráteheti magát az általa *bad*-nek nyilvánított, valójában jó clusterre vagy a tartalék- és a parkolópályára is.

Floppylemezeknél a bootrekord egy pontosan kijelölt szektor, a 0. fej 0. sávjának 1. szektorán(??). Ha a gép megfelelő jelzést talál, átveszi azokat a bájtokat, amelyek ebben a szektorban vannak, és elkezdi őket végrehajtani. Ez a kód általában nagyon rövid. Először többnyire csak azt mondja meg a gépnek, hogy hol találja azokat a szektorokat, amelyek a teljes bootprogramot tartalmazzák.

A vírusok egyes részeiket a merevlemezen és a hajlékonylemezen is el tudják rejteni, olyan szektorokban, amelyeket ők maguk rossznak(??) minősítenek. Ezek a szektorok csak speciális utasításokkal olvashatók, annak megakadályozására, hogy a kódokat valaki véletlenül felülírja. A rossz szektorok szaporodásakor tehát gyanakodhatunk, hogy valami megtévesztő trükk is lehet mögötte. (A CHKDSK segédprogram meg-

mutatja a rossz szektorokat). A jó szektorok lefoglalását és rossznak nyilvánítását az új típusú másolásvédelmek is alkalmazzák.

Miután betöltöttük, az operációs rendszer integrált része lesz a számítógép működésének, feladata például az állományok írása–olvasása(??), memória-hozzárendelés(??), más rendszererőforrások hozzárendelése(??). Igen kevés az olyan alkalmazás, amelyik ne használná az operációs rendszert az erőforrások eléréséhez. Egyes vírusok azonban módosítják a COMMAND.COM állományt, vagy akadályozzák a rendszer erőforrásaihoz való hozzáférést.

Két szinten is zavaróak ezek a tevékenységek. Először is: olyan helyre kerül a vírus, ami közös, gyakran használatos. Így rengeteg lehetősége adódik a továbbterjedésre. A másik kellemetlenség: keresztezi az erőforrás-igényléseket és meg is változtatja ezeket. Fizikai tünete ennek az, hogy a gép szörnyen lassan dolgozik.

Milyen kódot futtatunk, amikor programot futtatunk? (??)

Az itt következő felsorolás távolról sem teljes. Arra kívánunk mindössze rámutatni, hogy bármely kapcsolat potenciális támadási pontot jelenthet a vírusok számára. A vírusnak az a célja, hogy gyakran legyen futtatva, és nagy gyakoriságot érhet el, ha az alábbi területek valamelyikére sikerül betelepednie.

A DOS elfogadja a leütött billentyűket

BIOS INT 9h, INT 16h, INT 15h, INT 1Bh DOS INT 21h tasztatúraműveletek, INT 28h

Bármely billentyűzetmeghajtó vagy TSR (Terminate and Stay Resident) program.

A DOS betölti a programunkat

BIOS INT 13h, INT 40h, INT 15h

DOS INT 21h állománykereső műveletek, memória-hozzárendelés, DTA-beállítás, lemezről való olvasás, Ctrl-Break ellenőrzése stb.

Bármely DOS bővítőmeghajtó vagy TSR program.

Általános háttérműveletek

BIOS INT 8 (timer), INT 0Fh (printer), INT 1Ch (timer)

Bármely rendszermeghajtó vagy TSR program.

Program futtatásakor minden egyes alkalommal lezajlanak a fenti események, ezenkívül természetesen még sok más is. Amikor csak elindítjuk a rendszert, a CONFIG.SYS(??) és az AUTOEXEC.BAT(??) utasítja őt, hogy számos állományt töltsön be, még mielőtt elkezdenénk dolgozni a számítógéppel. Ha egy vírus ezen állományok valamelyikének

egyik utasítássorát célozza meg, az azt eredményezheti, hogy a programmal mindennap betöltődik. Újra megjelentek a .BAT vírusok! Mikor vizsgáltuk meg utoljára a CONFIG.SYS-t vagy az AUTOEXEC.BAT-ot? Emlékszünk-e még minden egyes sor jelentésére a két állományban?

Miután az adott vírust olyan szempontból kiismertük, hogy milyen típusú elemekre terjed tovább, a rendszer helyreállításakor háromféle kategóriába sorolhatjuk az elemeket.

1. Olyan típusú elemek, amelyeket a vírus megfertőz. Ezeket csak olyan korábbi mentésekből szabad visszaállítanunk, amelyeket egyenként alaposan megvizsgáltunk és fertőzetlennek bizonyultak. Ha lehetséges, használjunk minél „megbízhatóbb" forrást a visszatöltéskor. Ilyenek például a gyártó cégektől származó eredeti, írásvédett példányok, vagy ilyen módszer az, amikor a forráskódot újra lefordítjuk. Még ezekben az esetekben is ajánlatos a helyreállítás után egy ismételt ellenőrzés.
2. Olyan elemtípusok, amelyeket a vírus ugyan nem fertőz meg, de tevékenysége során tönkretesz vagy módosít. Ezeket általában biztonságosan visszaállíthatjuk a mentésekből, ennek ellenére itt is ajánlatos a mentett állományok előzetes ellenőrzése. Ha ugyanis a vírus elég hosszú ideig tartózkodott a rendszerben, károsíthatott olyan elemeket is, amelyek később lettek kimentve.
3. Olyan elemtípusok, amelyeket a vírus nem fertőz és nem is károsít meg. Ha egészen bizonyosak vagyunk abban, hogy bizonyos állományfajtákat a vírus érintetlenül hagyott, akkor ezekkel nem kell foglalkoznunk a helyreállítás során (például az ASCII szövegállományok).

A másolásvédelem csapdái

A másolásvédelem a program futásának fizikai vagy szoftveres eszközökkel való megakadályozása arra az esetre, ha az installálás vagy a futtatás nem az eredeti (vásárolt) lemezekről történik. Ez azonban adat- és üzembiztonsági szempontokba ütközik. Ha eladónk mindenáron másolásvédelemmel akarja ellátni munkáját, akkor meg kell vele értetni, hogy legfeljebb hozzáférés-védelmet alkalmazhat, installálási korlátozás nélkül. Ennek egyik oka, hogy ha egy nemzetközileg elfogadott programdiagnosztikai rendszer vírusfunkcióra utaló jelet mutat, akkor könnyen feltételezhető, hogy az a program mást is csinál. A másik ok *Éltető László* cikkében (Alaplap, 1991/3.) olvasható. Az eredeti programlemeznek mindig írásvédettnek kell maradnia, hogy véletlenül se következhessen be a vírusfertőzés. Ilyen „uninstall–deinstall" programokkal

vittek át egyik gépről a másikra még a közelmúltban is komoly károka okozó fertőzést. (A Lotus 1-2-3 R3 védelme is ilyen.)

Nem írhat a program olyan illegális helyre, hogy vírusfunkcióra adjon gyanúokot. Ugyanis a nemzetközileg bevett antivírusrendszerek a bootszektort, a partíciós táblát, valamint a merevlemez rejtett pályáit ellenőrzik. Ugyancsak nem írhat a setup ROM-ba, mert vírusfunkció kapcsán ezt is ellenőrzik a vírusvédő hardvereszközök, és visszatöltik tartalmát, ha változás van benne. Hasonlóképpen nem alkalmazható a rejtett állományok és könyvtárak létrehozása, illetve a merevlemez egyes szektorainak logikai BAD-re állítása sem. Nem jelölhetőek meg a rendszerállományok sem.

Hozzáférés-védelmen a program illetéktelen használatának logikai-szoftveres eszközökkel való megakadályozását tekintjük. Ilyenkor az eredeti lemez gyárilag írásvédett, és korlátlan számú biztonsági másolat készíthető róla. Ez azonban csak a diskcopy parancsra és egyéb, teljes lemezt másoló segédprogramokra kell hogy igaz legyen, a DOS copy *.* parancsára nem vonatkozik, azaz a floppy lemezcímkéje tartalmazhat információt. A gépről backup eljárással vagy streamerkazettával lementett verzió ugyanarra a gépre feltéve fut, más esetben feltűnő módon közli demonstrációs voltát, illetve az illegális másolat tényét, és kéri azt az azonosító kódot, amellyel ismét teljes értékű példánnyá változtatja magát.

MIT SZABAD ÉS MIT NEM?

a) Megengedhető a debug-ellenes kód, de csak akkor, ha nem akályozza más rezidens programok (Sidekick, hálózati, billentyűkezelő, antivírus programok stb.) használatát. Ha debug jelenlétét érzékeli, akkor figyelmeztető üzenettel lépjen ki, de kárt ne okozzon!

b) Megengedhető a lemezcímke állományként való kezelése, de ezzel vigyázni kell, mert elég közismert, és más programok is alkalmazzák.

c) Megengedhető az alkönyvtár dátum és időpont adatainak figyelése (századmásodperces érték van, de a DOS nem mutatja, s a hagyományos programokkal nem módosítható). Nem megengedhető ugyanakkor a különleges karaktereket (például a 255-ös karaktert) tartalmazó könyvtárnév alkalmazása.

d) Nem megengedhető a saját könyvtáron kívüli ellenőrző állomány létrehozása. Saját könyvtáron belül tetszőleges ellenőrző állomány hozható létre, de az sem lehet rejtett, rendszer vagy csak olvasható, és a név csak a DOS-ban szokásos normál ASCII karaktereket tartalmazhatja.

e) Megengedhető, hogy a program önmaga vagy valamelyik segédállománya regisztrálásakor átíródjék, átkódolódjon. Ha ez .COM, .EXE, .APP vagy overlay, akkor erre figyelmeztető feliratnak (pl. a konfiguráció felírásának) kell utalnia.

f) Megengedhető a konfigurációs információkban a program helyhezkötése céljából a gép jellemző alapparamétereinek tárolása. Ezekből az egyediség érdekében legalább három-négy megoldást kell egyidejűleg alkalmazni:

— A BIOS gyártója, típusszáma, sorszáma.
— Tetszőleges helyről vett BIOS-minta.
— A kontroller BIOS-ából vett minta.
— A merevlemez gyári hibatáblája és mérete.
— Az operációs rendszer gyártóját jelző azonosító string vagy annak részlete.
— A video BIOS-ból vett minta.
— A memória SETUP-ból vett mérete (csak extended vagy expanded).
— A merevlemez SETUP-típusa.
— Egyes hardverportok, DMA-k címe.

g) Megengedhető a program könyvtárhoz és meghajtóhoz kötése, de csak akkor, ha a jelszó ismeretében ez megváltoztatható.

A fentiek mellett a gép szinte minden konfigurációhoz kötött adata állandó. Ha a program változást észlel, kérnie kell a felhasználói azonosítót (USER ID) és a jelszót. Utóbbi minimum 20, maximum 25 karakter lehet, kis- és nagybetű között nem szabad különbséget tenni, mert elátkoznak.

A jelszót egyszer, az installálás során kell beadni a programnak, hogy a környezet megváltozásáig azt ne is kérdezze. A programba beépíthető olyan rutin, mely a felhasználói azonosítóból állítja elő a jelszót, így a sokszorosítás során csak az ID-t kell beleépíteni. A kártyát az ID alapján a forgalmazó egy saját, nem publikus programjával bármikor előállíthatja és pótolhatja.

A regisztrálás elősegítésére bevezethető a kétlépcsős forma is. Ennek első lépcsőjét az ügyfél a programmal kapja kézhez. Ekkor programja az installálástól kezdve mondjuk egy hónapig vagy adott futásszámig üzemel. Bejelentkezéskor mindig üzenettel figyelmezteti az ügyfelet, hogy a korlátlan futáshoz szükséges a végleges regisztrálás. Ha a forgalmazó megkapta a vevő kártyáját, akkor elküldi a végleges jelszót, melyet az ID ismeretében tud generálni.

SZÁMHÁBORÚ VAKLÁRMÁVAL

A szükséges és elégséges védelem

Az izraeli *Carmel Engineering* Turbo Antivirus Toolkit terméke állítólag 149-féle vírust tud irtani. De valóban annyit, vagy csak reklámból írták rá ezt a számot? A vírusok többségének van .COM és .EXE állományokba beépülő kódja. Tehát ha ezeket különállóaknak tekintjük, akkor egy vírusból máris kettő lesz, mert a két változat irtása természetesen más algoritmust igényel. De folytassuk tovább. Az egyes vírusok kódjában a kezdő vírusgyártók átírják a karakteres azonosítókat. Például a Disk Killernek és a Stoned/Marijuanának is van ilyen magyar változata. Akinek nincs gátlása, ezeket is külön vírusoknak tüntetheti fel, annak ellenére, hogy ha jó a felismerő algoritmus, az eredeti eljárással is lehet őket irtani. Vagyis nem külön vírusok. Azután vannak vírusok, amelyeknek a kódjába egy kicsit belenyúltak, így például a magyar Kedd 1 csak a Péntek 13 B változata. Ha azonban az eredeti vírus kódját alaposan átírják, azokat már inkább tekinthetjük önállóaknak.

Ha tehát egy forgalmazó elhatározza, hogy minden vírus minden változatát külön típusnak veszi, akkor ugyanaz a program 10 helyett rögtön 22 eltérő vírust tud irtani, anélkül hogy közben akár egyetlen bitet is átírtak volna benne. És még csak nem is hazudik senki.

Sajnos a komoly szakemberek is kénytelenek bekapcsolódni a számháborúba. Már McAfee is azt írja dokumentációjában új Scan verziójáról, hogy 501 programvírust detektál. A verziószámban azonban csak a tisztességesen elkülöníthető főcsoportokat jelöli meg, könyvünk írásakor 76-ot. Hasonló bűvészmutatványokkal találkozhatunk az izraeli Turbo Anti Virus Toolkit dokumentációjában és néhány magyar szoftvernél.

Zavaróak lehetnek a vakriasztások is. Ha egy vírusjelző tévesen fúj riadót, annak legtöbbször az az oka, hogy a vizsgált program tartalmazza a vírus azonosítóját. Ezek általában első generációs vírusdetektorok, mint a Prgdoki v.2.xxE sorozata, az osztrák Ikarus program rezidens és főmodulja, a Look nevű víruskereső, hogy csak a legismertebbeket említsük. Komolyabb a gond akkor, ha egy jónak tartott szoftver téved, s ráadásul gyári programlemezt gyanúsít vírussal. Ez történt 1990 júniusában, nem kis pánikot okozva egyik tervezővállalatunknál. Itt a

CHKSeq hazai szabadszoftverként forgalmazott víruskereső a (c)Brain vírust jelezte egy építészeti tervezőrendszer kulcslemezén. De csak vaklárma volt. A standardnak tartott Scan program is okozhat vakriasztást. Például a 6.3V72 jelzésű verzió a Turbo Pascal 6.0-val fordított programokban és magában a Turbo Pascalban egy rosszul megválasztott azonosító szekvencia miatt Kamikazi vírust jelzett. Ugyanez a szoftver a Sysdoki terjesztésére használt lemezeken Stoned-2 vírust talált. Éppen ezért igen nagy a felelőssége annak, aki a vírusazonosítókat kiemeli a vírus testéből. Sajnos a Scan is viszonylag rövid azonosítókkal dolgozik, holott éppen a több helyről vett viszonylag hosszú azonosító lenne a biztos feltétele annak, hogy ne legyen sok vakriadó.

Sokan felteszik a kérdést, hogy érdemes-e hazai programot írni könyvekből és más szakirodalomból összeszedett vírusazonosítók felhasználásával, vagy jobb átvenni egy olyan szoftvert, amely már bevált és ingyenes standard programként használják szerte a világon. Nem inkább a birtokunkban lévő magyar specialitások biztos detektálására és irtására kellene koncentrálni erőinket? Az így írt programok a standard programokkal együtt alkalmazva nagyobb biztonságot adnak, mint a számháború jegyében hevenyészve megírt programok. A Cebiten bemutatott vírusellenes programokból hamar kiderül: a jövő a komplex védelmi rendszereké, az egyetlen termékbe integrált megelőző, kereső, irtó és helyreállító programrendszereké.

A helyreállító algoritmusok területén hazánk is eredményes kutatásokat folytat. Jó lenne, ha ilyesmivel, nem pedig vírusírással állítanánk ki erkölcsi bizonyítványt a magyar számítástechnikáról. Ennek a szaklapokban többször is megismételt kérésnek sajnos nincs foganatja. Már legalább tizenöt eredeti magyar fejlesztésű vírussal „gazdagodott" a számítástechnika.

Az idő és a vírusok fejlődése túlfutott a Prgdokit felváltó Sysdoki koncepcióján, emiatt dolgoztuk ki az új detektáló–kereső rendszert, a PC–SCAN és a PC–CLEAN programokat. Elvetettük a korábbi immun koncepciót, mely igen sok gondot okozott az eltérő, nem szabványos DOS-verziók és az integritásvédelemmel ellátott programok miatt.

A gyakorlat és az újabb vírusok megjelenése arra kényszeríti a vírusok ellen küzdő szakembereket, hogy olyan új technológiákat dolgozzanak ki, amelyek minimális mértékben zavarják a gépek futását. Ezért kezdődött el a vírusellenes kártyák fejlesztése. Ezek nem ismeretlenek a magyar közönség előtt. A holland Thunderbyte és az izraeli fejlesztésű, de tajvani továbbfejlesztésű Virus Guardian nemcsak sok helyet foglal el a memóriából, de használóik életét is megkeseríti. Ugyanakkor sok vírus könnyedén elfut mellettük. Ez adta az ötletet, hogy elkerülve a ko-

rábbi vírusvédő kártyák hibáit, elkészítsünk egy sokkal megbízhatóbb vírusvédő rendszert. Ennek algoritmusa az eddigi legkorszerűbb, és semmi helyet nem foglal le a gép memóriájából, mert a szoftver a kártya RAM-jában fut.

Szoftverei kidolgozásakor McAfee különböző védelmi szinteket (level) állított fel. Logikáját követve saját kutatásaink során további fokozatokat határoztunk meg. Ezek megismerése mindenkit segíthet az érzékeny számítástechnikai rendszerek adat- és vírusvédelmi koncepciójának kialakításakor is.

LEVEL 1 — Primitív védelem

Kizárólag elméleti kritériumok alapján próbál védőmechanizmust kialakítani. Például megakadályozza a vírus rezidenssé válását, vagy azt, hogy futtatható programba írjon egy másik program. Egyre több baci simán átbújik alatta. Alkalmazása ennek ellenére szükséges, mert a primitívebb vírusok ellen védelmet nyújt.

LEVEL 2 — Célzott keresés

Betáplált vírusismerete alapján az elterjedt vírusokat célzottan keresi. Lehet hagyományos szekvenciális keresés, mint a Scané, de lehet ezzel kombinálva a vírusjelenségekre való figyelés (például a dátum megváltozása). Védőképessége korlátozott, mert csak az *ismert* vírusok ellen hatásos. Állandó korszerűsítésre szorul.

LEVEL 3 — Változásérzékelés

A programok elváltozására hívja fel a figyelmet, amiből vírusok jelenlétére lehet következtetni. Ebbe a típusba tartozik a program végére tett ellenőrzőkód (McAfee Scan program /CA opciója) vagy a program végére illesztett védekező „farok". Ilyen például a Sysdoki immun opciója, a Buruzs Tamás-féle Self Protection System (SPS) koncepció. A lopakodó vírusok egy része azonban, ha jelen van a memóriában, ezen a védelmi bástyán is képes áthatolni.

LEVEL 4 — Kombinált védelem

Csak hardvereszközökkel oldható meg. Egyesíti magában az előző szintek minden ismeretanyagát. Azzal bővül, hogy indításkor a gép boot és setup rendszerét is nem felülírható tárolóból töltjük be (a ROM-ba égetett DOS operációs rendszer a saját állományaira nézve ilyen). Vírusismerete alapján automatikusan kitakarítja az ismert vírusokat, az illegális műveletek megakadályozásával, illetve a figyelem felhívásával

az ismeretlen vírusok ellen részben vagy teljes mértékben védelmet nyújt.

A fentiek megvalósítási kísérletei a következő programoknál hoztak több-kevesebb sikert:

Leitold Ferenc és Tábor Csaba CHKVIR programja a /d opcióval az első száz utasítás lefuttatásával próbált vírusgyanús programrészekre bukkanni. Nem ad folyamatos védelmet.

Az izraeli Iris szoftverház és az Emsys közös terméke az Antivirus Plus öntanuló vírusvédő program. Folyamatos védelmet ad, vírusismerettel és általános elveken nyugvó vírusvédelemmel rendelkezik. Nem szíveli a különleges DOS-t, ha ehhez nagy partícióméret is társul. Az eddigi legjobb rezidens szoftveres megoldás. Egy baja van: nagy a tárban maradó része és sok program „összevész" vele.

A holland Thunderbyte vírusvédő kártya a tárrezidens műveleteket és a direkt lemezírást tartja kontroll alatt. Néhány vírus sétagaloppban fertőz mellette, s a konkrét vírusismeret hiánya miatt helyreállítás sem történik. Minden apróságra reagál és minden döntést a felhasználóra bíz. Teljesen legálisan dolgozó szoftverek esetében (Norton Commander) is állandóan visít. Csak birkatürelmű, erős idegzetű embereknek ajánlott! Szoftverének rezidens része igen sok memóriát emészt fel.

Izraeli eredetű a tajvani gyártmányú Virus Guardian védőkártya. Szintén általános védelmi eljárást alkalmaz, vírusismerete teljesen hiányzik, bár jobban megoldották a booteljárás védelmét, mint a Thunderbyte esetében. Ha nem DOS-t, hanem PC Tools-t vagy Norton Commandert alkalmazunk, állandó sípolásával és a rendszer lemerevedésével alaposan próbára teszi türelmünket.

Mivel sok szoftvert és kártyát végigtesztelve sem sikerült megfelelő védelmet adó technológiat találni, profi szoftveres és hardveres háttér felhasználásával kifejlesztettük a saját koncepciónknak megfelelő vírusvédő kártyát, amelynek prototípusát a budapesti Ifabón, 1991 májusában sikerrel mutattuk be. Az előzetes tapasztalat alapján valóban képes negyedik szintű védelmet adni, és mind a vírusok által használt eljárásokra, mind vírusismeret szempontjából aktualizálható, alkalmas továbbá adatvédelmi–hozzáférés-védelmi rendszerekbe történő beépítésre.

A SCAN-CSALÁD

Semmi sem tökéletes

Hazánkban az alapszoftverek közé tartoznak ezek az USA-ban szabadszoftverként terjesztett programok. A Scan-t az adatbiztonsággal foglalkozó McAfee Associates készíti és terjeszti. Hálózatos rendszerekre csak a nem szabadszoftverként forgalmazott párja, a NetScan, valamint a komplex kereskedelmi termékként értékesített Pro-Scan alkalmas.

A Scan65 verzió soha nem készült el. Helyette egy kárt okozó trójai változat került forgalomba, éppen ezért a programot a dokumentációk alapján a Validate segédprogrammal előzetesen ellenőriznünk kell. Nálunk a 45-ös jelzésűt írták át trójaivá, illetve vírushordozóvá: a Yankee Doodle hazai átiratát, valamint a V2000 vírust kapcsolták hozzá titokzatos kezek. Néhol a számot is átírták 99-re vagy másik, az eredetinél nagyobb kétjegyű számra.

A Scant a korábbi Pkarc helyett most a Pkzip programmal tömörített formában terjesztik. Az átbarkácsolásokra való tekintettel használni kell a becsomagolt állomány ellenőrző opcióját. Akkor bízhatunk meg egy tömörített McAfee programban, ha kibontás után a következő rendszerüzenetet adja:

```
PKUNZIP (R) FAST! Extract Utility Version
1.1 03-15-90
Copr. 1989-1990 PKWARE Inc. All Rights
Reserved. PKUNZIP/h for help
PKUNZIP Reg. U.S. Pat. and Tm. Off.
Searching ZIP: SCANV76C.ZIP - ComNet
Luxembourg BBS (+352)22534 USR_HST/V32
Europe's finest ms-dos files collection !
Exploding: VIRLIST.TXT -AV
Authentic files Verified! # NWN405 Zip
Source: McAFEE ASSOCIATES
```

Első üzenete a .ZIP állomány forrását jelöli, amely vagy van, vagy nincs. Ez jelen esetben a COMNET hálózat. A kibontott állománynév mellett a -AV az eredetiségre utal. Az utolsó sor a McAfee cég számító-

gépes aláírása, amelyet csak az ő részére regisztrált ZIP verzióval tud elhelyezni a programban. A Scan 72-es sorozatától kezdve csak a kibontás után így megjelenő McAfee programokat tekinthetjük megbízhatóan hitelesnek.

A szoftver kibontás után mindig tartalmaz egy VIRLIST.TXT szövegállományt a felismerhető vírusok rövid táblázatával, valamint egy Validate nevű programot, amellyel ellenőrizni lehet az állomány sértetlenségét. A Validate algoritmusát — bár a vírusírás megnehezítése érdekében szakmai körökön kívül nem publikálják — sok országban szabványként fogadták el. Lényegében egy speciális módon képzett ellenőrző összeget vizsgál. A program dokumentációja azokat a CRC-értékeket tartalmazza, amelyeket ellenőrizni szeretnének vele.

A csomagból a legismertebb a SCAN.EXE program, jelenlegi változatában a 6.8V76-C. Az első (tizedespontos) szám a szoftververzió jelzése, a V betű utáni szám pedig azt mutatja, hogy hány vírusfőcsoportot tud azonosítani. Nagy biztonsággal felismeri a barkácsolt változatokat is, például a Yankee Doodle, a Péntek 13, a Vacsina-B variánsait. Újdonságszámba megy, hogy sok eredeti kelet-európai vírust is kifogástalanul azonosít. A Monxla, a Turbo Kukac, sőt a Phantom is szerepel már az általa felismerhető vírusok között, és már észleli az egyes Vacsina-átiratokat is. Sajnos két-három hónap is eltelik, mire egy-egy kelet-európai újdonság belekerül.

McAfee cége sem tévedhetetlen, s a nagyüzemivé vált kibocsátással a selejt is gyakoribb. Például nagy megrökönyödést keltett a Scan 6.8V74 verziója, amikor ugyanis 4.xx feletti MS-DOS vagy PC-DOS változattal, esetleg 31 MB-nál nagyobb winchesteren Tandon DOS alatt akarták megnézni a partíciós táblát, programfutás helyett a partíciós tábla nagyságát kifogásoló, következő épületes rendszerüzenetet kapták:

```
Sorry, the partition table of disk C is 1024
bytes long.
That's too big for me.
```

Ezt megelőzően a 6.3V72 verzió még jól használható volt, és az újabb változatokban már kijavították a hibát.

VIRUSCAN 7.2C76 — a védelem bástyája

Ez a program jelentősen eltér a korábban megszokott McAfee programoktól, ugyanis az 5.3-as főverzió megjelenésekor alakították ki az egységes védelmi rendszer koncepcióját, és a csomagot több, egymással együttműködni képes segédprogramból állították össze.

Az alapot jelentő detektorprogram a ViruScan (SCAN.EXE), valamint ennek Novell, Banyan és Token Ring hálózatokon is működőképes változata, a NetScan. Az észlelt vírusok eltávolítására, illetőleg a fertőzött állomány törlésére a Clean-Up (CLEAN.EXE) szolgál. Ehhez csatlakoznak a fertőzést megelőző programok.

A tárban maradó és az ismert vírusok ellen folyamatos védelmet nyújtó program az FSHIELD.EXE (a vírusernyő), a korábbi rezidens Scan-Res víruskereső utódprogramja (amit könyvünk első kiadása óta két VShield tárrezidens program is felváltott). Korábban a VCopy a másolást is folyamatosan ellenőrizte, ennek fejlesztése azonban valami ok miatt megállt.

A 7.2C76 verzióban kijavították a korábbi változatok hibáit, és új lehetőségekkel is bővítették a programokat. A manapság terjedőben lévő vírusok szinte mindegyikét ismeri. A korábbiakkal ellentében McAfeeék viharos gyorsasággal beépítettek jó pár szovjet, magyar és bolgár vírusváltozatot. Ennek következtében a program által nyújtott biztonság jelentősen megnőtt.

A SCAN.EXE jelen verziója már végez öntesztet. Fertőzött állapotára külön üzenettel hívja fel használója figyelmét. Hossza 58 467 bájt, dátuma 1991.04.09. Változatlanságának ellenőrzésére a vele egy állományba tömörített Validate program szolgál, amely egy speciális szabványosított algoritmus alapján két ellenőrző összeget képez. Ennek meg kell egyeznie a dokumentációban találhatóval. (Ugyanezt a szabványosított algoritmust alkalmazta a Sysdoki is.) Sok újabb generációhoz tartozó vírus, amennyiben jelen van a memóriában, alaposan átveri ezeket az eljárásokat.

A 76-os verzió kibocsátását néhány héten belül követték a javított változatok, mert a széles körben alkalmazott program számos hibája viszonylag gyorsan kiderült. Az A változatban probléma volt a merevlemez nagy partíciójának kezelésével, és benne hagytak néhány hibás, sok vakriadót okozó azonosítósorozatot. A B változatban a bootszektor és az írásvédett lemezek kezelése volt problematikus, amit azután a C verzióban javítottak ki. Az ellenőrző kód hozzáadása az ellenőrzött állományhoz viszont továbbra is sok gondot okozhat.

A jelenlegi változat 224 ismert és nemzetközileg elterjedt boot, partíciótábla és fájlvírus 501 változatát mintegy 95%-os biztonsággal azonosítja. Kivéd továbbá egy vírusterjesztő trükköt is! A vírusos program kódját gyakran tömörítették az LZH.EXE segédprogrammal, így a hagyományos keresők nem találták meg benne a víruskódot, mert az csak a tárban, a program kicsomagolása után vált felismerhetővé. Az új Scan már bele tud nézni ebbe az állománytípusba is.

A program kapcsoló-utasításai:

```
SCAN d1: ... d10: \ /A /AV /CV /D /E .xxx
.yyy .zzz /EXT d:állománynév /FR /MANY /NLZ
/NOBREAK /NOMEM /NOPAUSE /REPORT állománynév
/RV /X
```

d1–d10 — A vizsgálni kívánt meghajtók betűjele, maximum tíz, szabványosan leírva, szóközzel elválasztva (például C: D:).

\ — Csak a főkönyvtárat, a bootszektort és a partíciós táblát vizsgálja.

/A — Minden állományt vizsgáljon, ne csak a .COM és .EXE kiterjesztésűeket.

/AV — Kapcsoljon ellenőrző kódot a vizsgált állományhoz.

/CV — Ellenőző kód alapján vizsgáljon.

/D — Rákérdezés után töröljön minden fertőzött állományt.

/E .xxx .yyy — Az általunk beírt kiterjesztésű fájlokat vizsgálja.

/EXT d:*állománynév* — Az általunk készített vírusazonosító állomány alapján keressen.

/FR — Az üzeneteket francia nyelven írja ki.

/NLZ — Ne vizsgálja az LZEXE összenyomott állományok belsejét.

/M — A memóriában is keresse az általa ismert vírusokat.

/MANY — Folyamatosan kérje a további megvizsgálandó floppykat.

/NOBREAK — Víruskeresés alatt tiltsa le a Ctrl-C és a Ctrl-Break megszakítást. (Hivatali használatra .BAT állományból így ajánlott.)

/NOMEM — Hagyja el a memóriavizsgálatot.

/NOPAUSE — Ne írjon a monitorra.

/X — Kihaltnak tűnő (az utolsó 12 hónapban nem jelentkezett) vírusokat is keressen.

/REPORT d:*állománynév* — Vizsgálatának eredményét mentse el fájlba.

/RV — Szedje le az ellenőrző kódot.

A Scan 64-es verziójában jelent meg az egyes állományokhoz kapcsolható azonosítókód, annak használatakor azonban több program tönkre-

megy, mert saját önvédelme vagy olyan másolásvédelme van, amely nem tűri a beavatkozást. Ilyen például a Turbo Debugger, a Lotus 1-2-3 R3 másolásvédett verziója, a CHKVIR vírusellenes program. A korábbi verziókat alkalmazva az azonosítókód sem volt nyom nélkül eltávolítható. Ha egy önvédelemmel rendelkező program védelmi mechanizmusa működésbe lépett, akkor hiába szedtük le a kódot, a program többé nem működött.

Külön érdekesség az LZEXE programmal tömörített .EXE állományok vizsgálata. Miután nálunk is kiterjedten alkalmazzák ezt a tömörítő eljárást, feltétlenül megéri nem letiltani ezt a lehetőséget, annak ellenére, hogy a vizsgálat így mintegy 10%-kal lassul. A program azonban a PKZIP, PKPAK, ARJ, LHARC, LHA, PAK és ZOO programokkal tömörített állományokat még mindig nem tudja vizsgálni. Erre a célra a SHEZ program alkalmazható, amelyhez a jelenlegi SCAN-verzió installálható.

A /D kapcsolóval előbb felülírja, majd törli a fertőzött állományokat, minden esetben rákérdezve. Csak gyorssegélyként javasoljuk, mert a fertőzött program elvész!

A ritkán előforduló vírusok keresésére szolgál az /X opció. Enélkül ezeket a táblázatunkban csillaggal megjelölt vírusokat *nem* keresi, vagyis gyorsabban dolgozik. Mégis ajánlatos, hogy használjuk, mert azért időnként a nem gyakori vírusok is felbukkannak.

Az /M memóriaellenőrző opció sok vakriasztást okozhat, különösen a korábbi rezidens víruskereső programok miatt, amelyek nem kódolják azonosítóikat. Ha régebbi verziójú (v42 alatti) ScanRes vagy a Sysdoki után pásztázzuk a memóriát, a memóriaszemétben ott található a kibontott vírusazonosító karaktersorozat. Ha a program ezt észleli, hamis riadójelzést ad. Figyelmeztetése csak akkor helytálló, ha ugyanazt a vírust valamelyik állományban is megtalálja. A teszt viszonylag lassú, a gép sebességétől függően akár egy percig is eltarthat. Hasonlóan hibás jelzést kaphatunk MS–DOS 3.2xx esetén, mert annak egyes változataiban megvan az 1701/1704 vírusra jellemző kódrészlet, de ez nem vírus. Egyes Scan-verziók a DOS 3.3x és a 4.xx esetén a memóriában Vacsina vagy Yankee Doodle fertőzést látnak, mivel a DOS-ban ott a .COM/.EXE konverter 200 bájtja, ami a Vacsina-fertőzés első lépése is lehet. Ilyenkor, ha nem találunk az állományokban vírust, vakriadónak minősíthetjük jelzését. Ezek a tünetek eddig csak az /M opciónál fordultak elő, anélkül nem. Éppen ezért a módosított eljárás már célzottan keresi a következő vírusokat, amelyek sok DOS-on keresztül elérhető paramétert elmaszkolnak, tehát detektálásuk nagyon fontos. Kiemelt vírusok a memóriában:

1554, 1971, 1253, 2100, 3445-Stealth, 4096, 512, Anthrax, Brain, Dark Avenger, Disk Killer, Doom-2, EDV, Fish-6, Form, Invader, Joshi, Microbes, Mirror, Murphy, Nomenclature, Phantom, Plastique, Polish-2, P1R (Phoenix), Taiwan-3, Whale, Zero-Hunt.

Az /E kapcsolót akkor kell megadni, ha a rendszerkörnyezetben az alapértelmezéstől eltérő overlay-állományok is vannak, és nem a minden állomány megvizsgálását kérő /A opciót alkalmazzuk. A Scan rendszer overlay-kiterjesztésének alapértelmezése OVL, OVG, OV1, OV2, OVR, SYS, BIN és PIF. Rendszeres számítógép-használat esetén célszerű naponta egyszer vírusvizsgálatot tartani az /A opcióval. Ez akár tízhúsz percig is elhúzódhat, ha nagy a merevlemez és sok az állomány. Ha figyelmetlenségből mindkét opciót megadjuk, az /A automatikusan elnyomja az /E utasítást.

A még ismeretlen vírusok felderítését szolgálja a változás-ellenőrzés. A program az /AV opcióval egy ellenőrző összeget hoz létre, amelyet binárisan kódolva az állomány végére tapaszt. Ennek hossza 10 bájt. A későbbi futtatások alkalmával ezt ismételten kiszámítja, s ha eltérést talál, jelez. A saját integritás-ellenőrzéssel működő programok és másolásvédelmek azonban nem szívelik ezt a manipulációt. A másolásvédett program ilyenkor büntet, az integritás-ellenőrzéssel ellátott program pedig nem indul el, vagy önmagát kiirtja. Ha tehát ilyen opciót adunk meg, előtte próbáljuk ki a gyanús programok másolatán, másolásvédett programokon pedig ne alkalmazzuk. Azt is tudni kell, hogy ezt az ellenőrző kódot a többi változásfigyelő program is jelezni fogja, de ilyenkor természetesen szó sincs vírusról. A kódtól az állományok az /RV opcióval szabadíthatók meg. A /CV opcióval csak ellenőrzőkód alapján vizsgálja azokat az állományokat, amelyeken ilyet talál.

Néhány nem standard DOS-nál a program örökké a boot megváltozását jelzi. Ilyenek például a Hewlett Packard valamint a Zenith operációs rendszerei, ahol ezzel a jelzéssel (de csak ott!) nem kell törődni.

Az /X opció egyes verziókban megvan, másokban nincs. Mindenesetre mindegyik elfogadja. Célja, hogy a régi, kihaltnak tűnő bacikat is megkereshessük.

Programozási hiba volt a korábbi verziókban, hogy a /Report opcióra a megadott néven 0 bájt hosszú állományt hozott létre. Ezt a mostani verzióban már kijavították, de a hiba megmarad akkor, ha a program futását menet közben megszakítjuk. Ilyenkor kilépéskor illene lezárni a megnyitott állományt!

A /NOBREAK letiltja a Ctrl-C és Ctrl-Break megszakítást, ha például

az AUTOEXEC.BAT-ból futtatjuk. Ilyenkor a program mindenképpen lefut, hacsak nem hajtunk végre új rendszerindítást.

Szintén ebben a verzióban jelent meg a /NOPAUSE opció és ugyancsak .BAT állományból való futtatáskor alkalmazható. Nem ír ki zavaró rendszerüzeneteket, csak akkor jelez, ha baj van. Nem kéri, hogy hibalistáknál és felsorolásoknál néhány sor kiírása után mindig megnyomjunk egy gombot a továbblépésre. Végre!

A program .BAT állományból vagy másik programból is indítható, ha legalább 320 kbájt áll a rendelkezésére. Ilyenkor az összes McAfee program a DOS-tól lekérdezhető hibakóddal tér vissza, ami meghatározhatja további tevékenységünket. Az errorlevel kódok a következők:

0 — Nincs vírus, minden normálisan lefutott.

1 — Vírust talált.

2 — Futáshiba vagy erőszakos megszakítás.

A külső szignatúraállományt az /EXT d:*állománynév* kapcsolóval tölthetjük be. Ekkor a beépített szignatúrák és a mi állományunkban foglalt adatok alapján egyaránt keresi a vírusokat.

Ha a programot vírusfertőzés éri, integritásvédelme a következő üzenettel figyelmeztet bennünket a bajra:

```
SCAN 7.2C76 Copyright 1989-91 by McAfee Associates.
(408) 988-3832
Warning: The file "C:\TEMP\SCAN.EXE" has been
damaged!
Scanning for known viruses.
Scanning 576K RAM
Sorry, I cannot find "C:scan".
SCAN 7.2C76 Copyright 1989-91 by McAfee Associates.
(408) 988-3832
This program may not be used in a business,
corporation, organization, government or agency
environment without a negotiated site license.
```

Hogyan írjunk /EXT adatállományt? A külső vírus-adatállomány (External Virus Datafile) megírásához vezérlőkódoktól mentes, tiszta ASCII szövegállományt produkáló szövegszerkesztők használhatók. Az egyes sorokat kocsivissza–soremelés karaktereknek kell lezárniuk. (Ilyen szövegszerkesztő a Personal Editor, a Norton Editor, a Kedit stb.) Ha /EXT álomány írására szánjuk el magunkat, fontos meghatározni, milyen szekvenciára akarunk keresni, hiszen ennek jó vagy rossz megválasztásától függ, hogy meg tudjuk-e találni a vírust, vagy csak vakriadókkal szerzünk kellemetlenséget magunknak és másoknak. Segítsé-

gével viszont a Jan Terpsta-féle TBSCAN.DAT szignatúraállományok közül átírhatjuk Scan alá azokat, amelyeket az aktuális SCAN verzió még nem ismer. Az állományba írtakat az adott jelfüzér (string) végigmaszkolásával keresi a merevlemez, illetve a floppy állományaiban. A fájl az alábbi formátumot használja:

```
#Kommentár a Virus_1-hez
"aabbccddeeff..." Virus_1_Név

#Kommentár a Virus_2-höz
"gghhiijjkkll..." Virus_2_Név

...

"uuvvwwxxyyzz..." Virus_n_Név
```

Az aa, bb, cc stb. helyén hexadecimális számok állnak, tartalmazva azokat az azonosítókat, amelyekkel a vírusokat keressük. Minden sor egy-egy vírusra vonatkozik. A vírus neve 25 karakter hosszú lehet. A "macskaköröm" (az Alt 34 karakter) a hexadecimális string kezdetét és végét jelzi. A Scan az így beadott azonosító sorozatot automatikusan keresi a memóriában, a partíciós táblában, a bootszektorban, a rejtett rendszerállományokban, minden .COM és .EXE állományban, valamint a .BIN, .OV?, .PGM, .PIF, .PRG, .SYS, .XTP overlay-állományokban. Ennyiben butább a hasonló programoknál, hiszen nem tudjuk megmondani neki, hogy melyik bacit hol keresse, pedig úgy csökkenthető lenne a vakriadó valószínűsége.

A keresési sorozatot megadhatjuk helyettesítő, dzsóker-karakterekkel is; így sok változat is megtalálható. A dzsókereket kétféleképpen lehet alkalmazni.

a) Állandó helyzetű dzsóker jele a kérdőjel. Ha egy keresési szekvenciába ?-et írunk, akkor tökéletesen mindegy, hogy azon az egy helyen milyen karaktert lát az állományban, találatot jelez. Például:

```
"E9 7C 00 10 ? 37 CB"
```

b) A változó hosszúságú dzsóker szintén sok segítséget jelenthet a keresés során. Megadásához *-ot, majd ezt követően egy gömbölyű zárójelbe tett arab számot kell írni, amelynek értéke minimum 1, maximum 99 lehet. Ilyenkor a program minden olyan esetben találatot jelez, amikor a dzsókerben megadott pozíció után folytatódik a keresési szekvencia (a változó dzsóker helyén 1-től a dzsókerben megadott pozícióig lehet eltérés). Bizonyos fokig használható a NOP-utasítások beszúrásával operáló vírusgyártó technika ellen.

Például:

```
"E9 7C *(4) 37 CB"
```

Egy keresési szekvenciában maximálisan 10 különböző dzsóker alkalmazható, akár vegyesen is a kétféle típusú. Megjegyzéseket is lehet a külső vírus adatállományokban elhelyezni. Akármennyi megjegyzés sor lehet, de mindegyik elején ott kell lennie a # numerikus jelnek (hashmark). Ide bármi beírható, ezt természetesen nem veszi figyelembe a keresésben.

```
#Új magyar .COM csodavírus: Kétfejű Lajcsika
#Izolálva: Lopware Kft. 1991.03.15.
"53 48 45 45 50" Lajcsika_[Laj-1]
```

A Scan program nem alkalmas hálózatra, de egy kis csellel, azaz ha nem vizsgáltatjuk vele a szervergép bootrekordját, hanem csak könyvtárakat adunk meg, hálózatban is alkalmazható.

Sajnálatos, hogy a vírusdefiníciós állományokra még nem alakult ki szabványos formátum, pedig azt nem védi semmilyen jogi korlátozás. Az IBM SCAN formátumából kinőtt, holland eredetű Thunderbyte Scan (azaz TBscan) formátuma tűnik olyannak, amelyik egyszerűen alkalmazható más programokban is. Ráadásul a VIRSCAN.DAT-ot Jan Terpsta rendszeresen aktualizálja, s az hazánkban is gyorsan terjed.

NETSCAN 74 — a hálózati szoftver

Nem szabadszoftver! Ennek ellenére, ha nem is a legfrissebb, de az azt megelőző verzióhoz általában hozzá lehet jutni szabadprogramként is, egy verziószámmal elmaradva a Scan programtól, melyhez a magyar felhasználó hozzájut. A NetScan ugyanazokat a vírustörzseket detektálja, mint a neki megfelelő Scan, viszont nem okoz zavarokat a halózat működésében. Ajánlatos, hogy a rendszergazda (supervisor) használja. Ez a hatáskör, tehát minden állománynak és könyvtárnak olvasási és írási joggal való elérése a program hálózati futtatásához alapfeltétel. Egy komolyabb Novell rendszer ellenőrzése, ha minden térképen végigmegy (egy könyvtárat ilyenkor esetleg többször is végigfuttat), akár fél óráig is eltarthat. Az esti rendszerleállítás előtt vagy reggel indításkor célszerű a műveletet elvégezni. Eltérés a normál verzióhoz képest, hogy bootvírusokat nem keres, hiszen a hálózati szerver merevlemezének bootrekordja általában nem is DOS formátumú. Az egyes opciókat a normál Scannek megfelelően kell megadni.

A Novell rendszereken való használata során, ha nem DOS, hanem

Novell rendszerállományhoz jut, olvasási hibát jelez. Ekkor a figyelmen kívül hagyáshoz szükséges I betűt (ignore) lenyomva az állomány átlépésével kell tovább futtatni a programot. Ha egy állományt a hálózatban más is használ, akkor hibaüzenetet kapunk. Ugyanígy egyes speciális Novell attribútummal rendelkező állományoknál is jelez (például a NET$.OS esetében), s ilyenkor tovább kell léptetni a programot. Használata:

```
NETSCAN d1: d2: ... dn: [/M /D /A /NOMEM]
```

Ahol d1: ... dn: az összes olyan meghajtó meghatározása, amelyet vizsgálunk. Ezek között lehetnek logikai meghajtók is! Az opciókra hasonlóképpen viselkedik, mint a normál Scan, és inkompatibilitásai is ugyanazok. Lehetséges egyetlen állomány vagy könyvtár vizsgálata is. Ezt a következőképpen adjuk meg:

```
NETSCAN L:\DIRECT\PROGRAM.EXE
```

Ha valami miatt nem tudna olvasni egy állományt, pl. Novell esetében, akkor kérdésére az „ignore" (I) válasszal kell kihagyatni azt az állományt. Ez a Novell egyes rendszerállományai esetében fordulhat elő. Más esetekben ilyen megállások után a „fail" válasznak megfelelő F-fel lökhetjük tovább az ellenőrzés folyamatát.

MDISK — a bootvírusok eltávolítója

Az MDISK programcsomag általános program a partíciós tábla vírustól való megtisztítására, verziószáma nem lényeges. Nem is szükséges gyakran cserélni, hiszen itt egy megadott eredeti és jól definiált struktúra visszaállításáról van szó. A legtöbb problémát mégis ez a McAfee-program okozza.

A programcsomag kizárólag szabványos, PC-DOS és MS-DOS formátumú bootszektort és partíciós táblát állít helyre. Ha nem szabványos, például HP, DR, Tandon vagy Compaq DOS-szal találkozik, azokat kérdezés nélkül tönkreteszi! Hatása felér egy jól megírt vírusával. Érthető tehát, hogy nem szívesen terjesztjük Magyarországon. A floppyk bootvírustól való mentesítésére viszont a tapasztalatok szerint kitűnően és nagy biztonsággal használható.

Az egyes DOS-verziókhoz eltérő programot kell alkalmaznunk, amelyek mindegyikét tartalmazza a tömörített .ZIP állomány:

```
MD40.EXE – MS/PC DOS 4.0/4.01
MD33.EXE – MS/PC DOS 3.3
```

```
MD32.EXE - MS/PC DOS 3.2
MD30.EXE - MS/PC DOS 3.0/3.1
```

MDISKxx — használatának legfontosabb tudnivalója, hogy ne indítgassuk el próbaképpen. Alapértelmezésben a merevlemez bootszektorát és partíciós tábláját vizsgálja, és ha azokat nem találja szabványosnak, helyreállítás címén rögtön tönkre is teszi. (Ugyanaz a hiba, mint amit a Norton Disk Doctor okozott a Disk Managerrel formázott winchesterekné!) Sok esetben a boot helyreállítása a DOS SYS parancsával veszélytelenebb és problémamentesebb!

MDxx F — a floppy bootszektorát takarítja le, de kizárólag az első, az A: meghajtóban. Ha a B: formátumának megfelelő floppval van baj, akkor más programot kell használni. (Nem véletlen, hogy az M–Disk rendszer alig-alig elérhető nálunk. McAfee rosszabbul sikerült termékei közé tartozik.)

MDxx P — a merevlemez bootszektorát figyelmen kívül hagyva a partíciós táblát vizsgálja és állítja helyre. Problémái ugyanazok, mint amiket előbb vázoltunk.

CLEAN-UP 6.8 74-B — a legismertebb vírusölő

Ha bekövetkezett a fertőzés, akkor segít a McAfee programrendszer víruseltávolító programja, a Clean-Up Virus Remover, közismert nevén Clean. Ám sokszor csak annyira juthatunk vele, mint a Scan /D opciójával: törölhetjük a fertőzött állományt. Az elsősegélycsomagban ott a helye, de használatával vigyázni kell, mert kárt is okozhatunk. Könyvünk előző kiadásának megjelenése óta sokat változtattak, javítottak rajta, a megfelelő Scanhez tartozó verzió megjelenése mégis várat magára. Jelenleg a 6.8V74-B verzió a hivatalos, de mire e könyv megjelenik, várhatóan már közforgalomban lesz a V76-os változat.

A Clean nem vírusdetektor! A „felderítésre" a Scan-t használhatjuk, amelynek újabb változatai már megadják a McAfee-féle azonosítót, a vírus szögletes zárójelbe írt nevét, amellyel a Clean programot utasíthatjuk a vírus kitakarítására. Az azonosítókat az aktuális VIRLIST.TXT állomány is közli. (Mi a *Vírushatározó* című könyv végén szintén közreadjuk.)

A tapasztalat azt mutatja, hogy a betolakodó eltávolítására a Clean sok olyan esetben is a fertőzött állomány törlését javasolja, amikor például a Sysdoki vagy a CHKvir programrendszer kifogástalanul „levakarja" a vírust.

A Clean nem képes minden esetben helyreállítani a floppyt a Stoned/Marijuana fertőzése után, de ez nem az ő hibája. Olyankor áll elő,

ha a lemezen nincs alkönyvtár, és a sok apró állomány miatt a főkönyvtári bejegyzés helye majdnem telített. A Stoned ugyanis a floppyn a főkönyvtár utolsó szektorába menti az eredeti bootszekort, és így felülírja az ottani bejegyzéseket. Ilyenkor egyes állományok elvesznek, és a könyvtári bejegyzésnél mindenféle „csillag–halálfej" jelsorozatokat olvashatunk. Ez a floppy azonban már nem fertőz, és az állományok nagy része ennek ellenére lementhető róla.

A program dokumentációja a tesztekben tapasztaltaknak megfelelően korrekt. Kifogástalanul irtja az alábbi vírusokat, annak változataival együtt, de más vírusok esetén is megkísérli a helyreállítást, vagy törlést ajánl fel. Ilyenkor ne fogadjuk el ezt a lehetőséget, hanem próbáljuk meg másik programmal elvégezni a takarítást.

A Clean 6.8V64 verzió által irtott vírusok

1260, 1591, 1701, 1704, 4096, Alabama, Alameda, Ashar, Bloody, Dark Avenger, DataLock, Disk Killer, EDV, Fish, Flip, Invader, Jerusalem A, Jerusalem B, Jerusalem E, Joshi, KeyPress, Liberty, Music Bug, New Jerusalem, Pakistan Brain, PayDay, Ping Pong B, Plastique, Slow, Stoned, SunDay, Suriv03, Taiwan 3, Taiwan 4, V800, VacSina, Vienna, Violator, Whale, Yankee Doodle, ZeroBug.

A felhasználók számára kulcsfontosságú, hogy e program irtja azokat az új generációs vírusokat is (Whale, Fish, 4096, Invader, Plastique), amelyek mostanában terjednek, s amelyek kitakarítására nagyon kevesen tudnának saját programot írni, mert az rendkívüli rendszerismeretet igényel.

A program használata a korábbiaktól annyiban tér el, hogy nőtt a megadható opciók száma. A fejlesztés iránya láthatóan a Scan és a Clean használatának egységesítése felé tart.

```
CLEAN d1: ... dn: [vírusazonosító] /A /E .xxx
/FR /MANY /M /REPORT d:állománynév
```

A Scan programéval lényegében megegyezik az /A, az /E, az .xxx, az M/, a /MANY, a /REPORT kapcsoló. A többi:

dn: — A tisztítandó meghajtók. Számuk maximálisan 10 lehet. Hálózatnál kötelező az olvasást és írást lehetővé tevő felügyelői jogkör.

[*vírusazonosító*] — A Scan által megadott vagy táblázatból kikeresheto a vírusazonosító, amit szögletes zárójelbe kell beírni. Egyszerre csak egy vírusnév szerepelhet a parancsban.

A Clean program fejlesztésében a 64-es verzió volt a fordulópont. Ekkor építették bele az önmagukat titkosító vírusok kitakarítására szolgáló algoritmust, amely a program vázának lényeges átírását eredményezte. Ezáltal a szoftver minősége sokat javult.

Az újabban elterjedt magyarországi vírusok esetében legtöbbször a törlést javasolja. Sok esetben a rossz irtás oka nem szoftverhiba, hanem hogy nem az a vírusváltozat van a gépben, amelyiknek az azonosító alapján lennie kellene. Sajnos a program hosszának csökkentése érdekében McAfee rövid azonosító sorozatokat alkalmaz — rövidebbeket, mint amilyeneket mi közlünk a *Vírushatározóban* —, így a tévedés veszélye is nagyobb.

Előfordul, hogy számos program .EXE állománya „helyreállítás" után nem működik. Ennek oka a programok önvédelmi mechanizmusában rejlik, mert a program megváltozását feltörési vagy megváltoztási kísérletnek érzékelik. Ilyenkor az eredeti lemezről kell visszatölteni az állományokat.

A programok végén gyakran ott van néhány bináris nulla karakter (amelyek száma sehol nincs láthatóan tárolva) és az EOF karakter. A vírustalanító programok a nullákat nem veszik (nem is vehetik) figyelembe, és „méretre" vágják a programot. Ha sok állományt tesznek tönkre látszólag biztos felismerés után, akkor nagy a valószínűsége, hogy egy NOP-technikával spékelt vírusátirattal van dolgunk. Csak fohászkodhatunk, hogy ilyennel ne találkozzunk.

A vírustalanítás akkor tekinthető befejezettnek, ha a program lefutása után a Scan tisztának találja az állományt. Ha maga a Clean fertőzött, akkor a fertőzést az állományok megnyitása során továbbadhatja, és éppen ő válhat a fertőzések forrásává. Ennek kiküszöbölésére saját integritásvédelemmel rendelkezik, amely a következő üzenettel adja tudtul vírusfertőzött voltát:

```
CLEAN 6.8B74 Copyright 1989-91 by McAfee
Associates. (408) 988-3832
Warning: The file "C:\TEMP\CLEAN.EXE" has been
damaged!
Cleaning [170x]
No viruses found.
CLEAN 6.8B74 Copyright 1989-91 by McAfee
Associates. (408) 988-3832
This program may not be used in a business,
corporation, organization, government or agency
environment without a negotiated site license.
```

VCOPY 0.5V67 — a másoló ellenőr

A VCOPY a DOS 3.3 verziójának másoló parancsát helyettesíti. Olyan könyvtárban kell lennie, amely az elérési útvonalon (path) szerepel. Legcélszerűbb a DOS többi programja mellé tenni. Lehetőségei és kapcsolói is teljesen azonosak a DOS kézikönyveiben leírtakkal. A program mintegy 10-20%-kal lassabb, mint a megfelelő DOS parancs.

A program intelligens másolást végez. Ennek során a másolt állományokban megkeresi, hogy előfordul-e bennük vírusazonosító jelsorozat. Ha igen, akkor felhívja rá a figyelmünket, és arra az állományra megtagadja a másolást. Átnézi az LZEXE tömörítőprogrammal összepréselt programok belsejét is, hogy tömörítés előtt volt-e bennük vírus. Ha igen, akkor azt is jelzi, hogy milyen tömörítési móddal becsomagolt állományban lelte. Az ARC, ZIP, PAK és ZOO tömörítőkkel előállított állományokba nem tud belenézni.

A következő másolási parancs kiadása egyben megmutatja az egyetlen eltérést a DOS kapcsolóitól:

```
VCOPY A:*.COM D:\PROG\TEST /E
```

A /E kapcsoló a programot arra kéri, hogy minden átmásolandó állományt végigbogarásszon. Teljesen egyenértékű a Scan /A kapcsolójával.

Viszonylag ritkán találkozunk új verzióval, a NetScanhez hasonlóan nem szabadszoftver, csak olcsó program, így az elektronikus postán csak a régebbi változatok érhetők el.

VSHIELD 2.8V72 és VSHIELD1 0.1 — a rezidens vírusernyő

A programrendszer két tagból áll: a hagyományos VShield (vírusernyő) programból, valamint egy kisebb tárigényű változatból, a VShield1-ből.

A VShield 2.8V72 tárrezidens, célszerű már az AUTOEXEC.BAT-ból betölteni. Amikor valamelyik programunkat használni akarjuk vagy új lemezt helyezünk a gépbe, akkor az tényleges indítás előtt vírusellenőrzést végez. A lemezeken először a bootvírusokat keresi, majd megnézi, hogy a memóriába töltendő programra nem telepedett-e valamilyen fájlvírus.

Melegindítás (Ctrl–Alt–Del) esetén is a tárban marad, így megakadályozza a fertőzött floppyról való újraindítást is. Ellenőrzi a bootszektort, a partíciós táblát és a rejtett állományokat. Memóriaigénye és sebessége nem teszi lehetővé, hogy itt is olyan gyorsan cseréljék a verziókat, mint a többi programnál.

A VShield program használata:

```
VSHIELD /CV /F [útvonal] /LOCK /M /NB /NOMEM
/SWAP [útvonal] /X
```

A kapcsolók jelentése:

/CV — A Scan által rátett ellenőrzőkód szerint vizsgál (ha van ilyen). Ha nem talál ilyet, akkor a szignatúrák alapján keres.

/LOCK — Letiltja a /REMOVE opció alkalmazását, tehát a memóriából csak a főkapcsoló segítségével lehet kipucolni, ugyanakkor vírust észlelve lemerevíti a rendszert. (Ha éppen dBase feldolgozás folyik, hatása pusztító lehet. Inkább mellőzzük!)

/NOMEM — Kihagyja a memória-ellenőrzést.

/REMOVE — Kiveszi a memóriából a VShield programot.

/SWAP [*útvonal*] — A VShield programnak csak egy kis részét installálja a memóriaigény miatt. A többit az útvonallal megadott helyre teszi, és szükség esetén onnan tölti be. Alapértelmezése a főkönyvtár. Ha van RAM-diszk, oda kell irányítani, hogy gyorsabb legyen.

/F [*útvonal*] — Segítség a DOS 2.0 verzió használóinak. Itt meg kell adni a programhoz vezető útvonalat.

/X — Olyan vírusokat is keres, amelyek az USA-ban több mint egy éve nem fordultak elő. Használata Magyarországon nagyon ajánlatos.

/NB — Kihagyja a boot ellenőrzését. Ez akkor kell, ha a rendszer melegindításkor ütközik más programokkal. (Ilyenkor viszont nő a fertőzés veszélye.)

A VShield, mint minden tárban maradó program, hajlamos arra, hogy más programokkal „összevesszen", és ne legyen velük kompatibilis. Garantáltan nem tűri a másolásvédett programokat (debugot észlel), az MS–Windows 2.xx és 3.xx verzióit, a Ventura 2.0 és a 2.0 professzionális DTP-programot. Hálózatba kötött gépen szintén nem alkalmazható.

Mintegy 30 kilobájtot vesz el a rendszerből, ami 6 kilobájtra csökken, ha a SWAP opciót használjuk. Ugyanakkor — átlagos gyorsaságú gépen — minden lemezművelet 600 ms-mal hosszabb lesz. Ez még elviselhető.

Tapasztalat szerint jól együtt tud maradni a legtöbb szövegszerkesztővel, a clipperezett programokkal, a dBase III Plus, a FoxBase, a Quattro programmal. Mint általában a rezidens programok, ő is inkompatibilis azonban az MS Windows programmal. A Lotus 1-2-3 R3 védett

(magyar és európai) verziója a memória elégtelenségére való hivatkozással szintén nem tűri el a VShield jelenlétét.

A VSHIELD1 a forgalmazója szerint alapszintű védelmet nyújt. Szintén tárban maradó program, a memóriából 6 kilobájt helyet foglal el. Jelenlétekor a lemezműveletek idejéhez minden olvasáskor 1 másodpercet kell hozzáadni. (Ha igazi csigatempót akarunk elérni, akkor csak a buffers értéket kell levenni 8-ra, és kiadni a verify on DOS parancsot.)

A program egyetlen lehetséges kapcsolója:

```
VSHIELD1 /NB
```

/NB — Jelentése ugyanaz, mint a normál rezidens programnál, letiltja a bootszektor ellenőrzését.

A Vshield program közli a DOS-tól lekérdezhető hibakódokat.

SZEKVENCIÁLIS KERESŐPROGRAMOK

A vírusok különös ismertetőjele

TBSCAN V1.8 víruskereső segédprogram

A TBSCAN v1.8 a Thunderbyte vírusvédő kártya mellé adott külön víruskereső segédprogram, amely szabadon terjeszthető. Írója Frans Veldman holland programozó, aki az ESaSS B.V. támogatásával rendszeresen kibocsátja az új verziókat, és a Thunderbyte BBS-ről bárki lekérheti azokat. (Telefon: 00-31-85-212-395 8N1 300/1200/2400 baud MNP5) Amilyen jó ez a program, olyannyira nem tetszik a kártya működése (lásd erről a Kártyajátékok című fejezetet). Szoftvere könnyen kezelhető, igen gyors, klasszikus szekvenciális keresőprogram. Nem könnyű hasonló gyorsaságú és megbízhatóságú újabb programot írni.

A gyorsaságot többféle programozástechnikai trükkel érték el. Ez a szoftver is egy szignatúraállományból olvassa ki az azonosítókat, de kicsit másképp bánik velük, mint a hasonló programok. Felállít néhány logikai feltételt is, amelyek alapján eldönthető, hogy érdemes-e végigmaszkolni az egész programot.

Ha a vírus megtámad egy .EXE állományt, akkor legtöbbször annak végére épül be. Nagyon ritka az olyan program, amelyiknél az első ugrás a program vége előtti részre mutat. A TBSCAN megvizsgálja, hogy az ugrás hová irányul. Ha további utasítástömegbe, a program belsejébe, akkor program, ha egy rövid szakasz lefutása után a program elejére, akkor valószínűleg vírus. Ez szinte természetes, mert a víruskódnak a program előtt kell lefutnia, és azt így lehet egyszerűen megoldani.

A másik ilyen trükk, hogy a TBSCAN nem az egész állományt olvassa be, csupán az állomány elejéről és végéről meghatározott, mintegy 3 kilobájtos részletet. Hogy honnan, azt abból dönti el, hogy a vírusra .EXE típusú (programvégi) vagy .COM típusú (programeleji) beépülés a jellemző. A jelenlegi vírustechnológiák és vírusírási tendenciák mellett ez 4–10%-os hibahatárt jelent. A program némi rokonságot mutat a CHKVIR programmal (a /D opcióval használva a magyar program is valami hasonlót végez, mely a TBSCAN esetében megfelel az /A opciónak). Foglaljuk össze ennek az eljárásnak a lényegét!

1. *Keresés és felderítés.* Megkeresi a program belépési pontját. Ha ilyet talál, akkor a program valóban futtatható, stabil. Ebből (és nem a kiterjesztésből) eldönti, milyen formátumú állománnyal van dolga.
2. *Nyomkövetés.* Megnézi a program kilépését. Ha ott az ugróutasítások nem mutatnak a program elejére, akkor a program stabil. Egyben megvizsgálja, hogy ezen a részen történik-e rezidenssé válás. Ha igen, akkor a program valószínűleg fertőzött.
3. *Analizálás.* Ha a kód nem stabil, azaz vírusra gyanút adó jelet talál, akkor böngészi végig a teljes programot. Ekkor már az azonosítók alapján keres. A SYS állományok megvizsgálását hasonlóképpen végzi el. Ha a program relokálja magát (adapt CS/IP relation), akkor is automatikusan és korrektül vizsgálja a CS és az IP regiszter változásait.

A program kihasználja a DOS FCB-vel (file control block) végezhető gyors memóriaműveleteit. A Scan mellett ez a másik olyan szekvenciális víruskereső, amely a memóriában is vadászik vírusok után. Kifogástalanul fut MS–DOS 5.xx verziókkal is. (A DOS kiakadt a vírusra, mert nem volt képes magát fertőzötten a HIMEM-be feltölteni, a program viszont jól működött!)

Használata

```
TB_SCAN [meghajtó és útvonal] [állománynév]...
[kapcsolók]
```

Lehetséges opciói:

-h	— Help, az opciók és a szintaxis kiíratása.
-f [*állománynév*	— A program az általunk használt szignatúraállományt alkalmazza a VIRSIG.DAT helyett.
-q	— Háttérben dolgozik, nem ír ki állományneveket.
-v	— Kommentált üzemmód, kiírja a belépési címeit.
-m	— A hosszú listákat képernyőre tördeli (more prompt).
-d	— Közvetlen DOS és BIOS hívásokat használ.
-a	— Minden állományt elemez, nemcsak a futtathatókat.
-s	— Kihagyja a bootszektort és a partíciós táblát.
-r	— Kihagyja a rezidens vírusok vizsgálatát.
+r	— Minden vírust keres a memóriában.
-l [*állománynév*	— A megadott állományba naplóz.
+l [*állománynév*	— A megadott naplót kiegészíti.
-n	— Kihagyja az alkönyvtárakat.
+n	— Csak az alkönyvtárakban keres.

Indítsuk el a TBSCAN-t a –V opcióval. A következő lista fog végigfutni a monitoron:

```
C:\KK\V11\HTSCANtbscan.com C:\ -v
TBSCAN fast Virusscanner v1.8 (C) 1990, ESaSS,
Nijmegen, The Netherlands.
Signature file announcement:
Virus information file for TBSCAN and HTSCAN virus
scanners
(C) Copyright 1989-1991 by Jan Terpstra of FIDONET
2:512/10.0
P.O. Box 66, 1462 ZH, Beemster, The Netherlands
Revision: 910331 (yymmdd)
LOW MEMORY          analyzing  >                +++ OK
HIGH MEMORY         analyzing  >                 ++ OK
BOOTSECTOR C:       analyzing  >                  + OK
PART.TABLE          analyzing  >                  + OK
IBMBIO.COM          scanning   >      000000      + OK
IBMDOS.COM          tracing    >      006C11      + OK
COPYQM.COM          tracing    >      005FFE      + OK
CQCONFIG.EXE        scanning   >      001A92      + OK
CRCHECK.EXE         scanning   >      000BDC      + OK
HGCIBM.COM          tracing    >      001087      + OK
MASOLO.EXE          analyzing  >      000000     ++ OK
SLICE.COM           analyzing  >      000000      + OK
...
```

Láthatjuk, hogy ebben az esetben a program nemcsak vírusismerete alapján dolgozott, hanem alkalmazta az említett intelligens víruskeresési eljárást. Ismerkedjünk meg alaposabban azokkal az opciókkal, amelyeknek a jelentése nem kézenfekvő.

–Q — Alapértelmezésben a TBSCAN kiírja, hogy éppen milyen állományt vizsgál. Ezzel az opcióval letilthatjuk a sok-sok feliratot. Természetesen a vírust ekkor is kiírja!

–V — Kiírja azt a pozíciót, ahol megtalálta az első érdemi utasítást (hexadecimális címként). Innen veszi a 3 kilobájtos mintát az elemzéshez.

–M — A listák gyorsan elszaladnak a szem előtt, követésüket segíti, hogy ezzel az opcióval képernyőnyi részletekre lehet azokat tördelni.

-A — Alapértelmezésben a TBSCAN a beépített debugger-interpreter segítségével csak a futtatható állományokkal törődik. Ezzel az opcióval rá lehet venni, hogy mindegyik állományt így nézze végig, ezáltal az átnevezett állományok és az állománytöredékek is analizálhatók. Természetesen a rendszer jelentősen lelassul. Ha nem végrehajtható állományról van szó, akkor a keresés során figyelmen kívül hagyja a vírus típusát, és mindegyik szekvenciára keres, tekintet nélkül azok boot, .COM vagy .EXE előfordulási helyére.

-D — A TBSCAN a DOS operációs rendszerrel a 21h interrupt segítségével tartja a kapcsolatot, amit azonban egyes vírusok is ismernek. A -D opcióval kikerülhető, hogy a vírus rájöjjön arra, hogy figyelik. Ha ezt alkalmazzuk, akkor a TBSCAN megkeresi a BIOS belépési pontját, majd utána direkt BIOS-hívásokkal kommunikál a rendszerrel, így a rezidens programok nehezebben vehetik észre.

Ezt az opciót szigorúan tilos alkalmazni többfelhasználós rendszerekben és hálózatokban, mert akkor kikerüli a többszörös állománynyitást kizáró hálózati szoftverrutinokat is. Jó pár másolásvédett szoftver egyszerűen megbolondul, ha a TBSCAN-t -D opcióval használjuk. Megszállottan másolásvédő programozók ezt az átlépést szokták felhasználni arra, hogy figyeljék, nem debuggolja-e valaki védelmüket. Néhány védelem ennek hatására elszabadul, és azonnal üt, mint a bolondóra.

A többszörös állomány-hozzáférés kikerülése miatt nem alkalmazhatjuk ezzel az opcióval akkor sem, ha a gépben más rezidens program (például Sidekick) van betöltve, mert ilyenkor a többi nyitott állomány, például a jegyzetblokk tartalma károsodhat. A disk cache nem okoz problémát, attól nyugodtan használhatjuk.

-S — Ott célravezető alkalmazni, ahol a bootszektor nem DOS formátumú, például a Novell szerver merevlemezén.

-R — Nem keresgél memóriában a vírusok után.

+R — A memóriában a szignatúraállomány alapján keresi az összes vírust. Kiválóan alkalmazható a vírusbecsomagolós és memóriában kipakolós trükkök ellen.

-L — Ezzel a paraméterrel a program létrehoz egy naplóállományt, amelynek neve alapértelmezésben TBSCAN.LOG, helye az a kurrens könyvtár, ahonnan a programot indítottuk, de más név és útvonal is megadható. A fertőzött program nevét és elérési útvonalát rögzíti.

+L — Ugyanaz, mint az előző, de a meglévő naplót nem írja felül, csak hozzáfűzi megjegyzéseit.

A VIRSCAN.DAT és a TBSCAN.DAT szignatúraállomány formátuma

Érdemes megismerkedni ezzel az adatformátummal, amely szabvánnyá, illetve szokvánnyá vált a víruskeresők között. Eredete az első szekvenciális kereső adatformátumára nyúlik vissza, csak azóta lényegesen átalakult. Jelentősen eltér a McAfee Scan-je által használt külső adatállomány formátumától, programozó hajlamúak azonban kis fáradsággal írhatnának a két formátum között automatikusan konvertáló programokat. Megérné...

Az adatállomány megjegyzés sorait pontosvessző kezdi. Ha a pontosvesszőt % jel követi, akkor a program ezeket a sorokat változtatás nélkül, üzenetként megjeleníti a képernyőn. Egy adatállományban azonban maximálisan 15 ilyen megjelenítendő százalékjeles sor állhat. A .DAT állomány tiszta ASCII formátumú. Ugyanúgy szigorú szabályai vannak, mint egy programnyelvnek.

A vírusleíró rész három sorból áll. Az első sorban a vírus megjelenítendő nevét adjuk meg.

A második sor azt mondja meg, hogy a következő sorban megadott keresési szekvenciát hol kell keresni a gépen belül. Ebben a sorban csak a kulcsszavak állhatnak, külön-külön vagy együtt is. Az egyes szavakat elválaszthatja szóköz, tabulátor vagy vessző. A kulcsszavak a következők:

```
BOOT   SYS   EXE   COM   HIGH   LOW
```

BOOT — A keresett vírus a bootszektorban vagy/és a partíciós táblában található. Csak ott keresi, amennyiben az /A opcióval vagy a memórián belüli kereséssel nem utasítottuk másként a programot.

COM, EXE, SYS — A keresett vírus ezekben a típusú állományokban található. Fontos azonban tudni, hogy amennyiben a vírus overlay-állományokat fertőz meg, a program az .OV* kiterjesztésű állományokat veszi figyelembe, és azokban csak az .EXE típusú vírusokat vizsgálja. Ez a filozófia a GEM programrendszer egyes állományainak vizsgálatakor sok téves észlelésre, illetve bacik elnézésé-

re vezethet. A GEM ugyanis tele van SYS nevű, de .COM típusú programmal, illetve .EXE kiterjesztésű, de .COM viselkedésű állományokkal és 64 kbájt feletti .COM formátumú, de .EXE-ként viselkedő programokkal (.APP). Ilyenkor a programot a /A opcióval kell futtatni.

HIGH — Ezzel a paranccsal megmutatja a vírust akkor is, ha az a TBSCAN fölött helyezkedett el a memóriában. Megadása különösen akkor célszerű, ha tudjuk a vírusról, hogy képes manipulálni a DOS memóriavégét jelző „farokpointert".

LOW — Ez a parancs azt nézi, hogy a vírus beépült-e a memóriába a TBSCAN allokációs helye alatt.

A következő sor a vírus keresési szekvenciájának ASCII hexadecimális leírása. Ez lehet komplett keresési szekvencia, de lehet dzsókerekkel kiegészített, közös leírás is, ilyenkor egyetlen szekvenciával több, részben vagy adott százalékban, illetve fő részleteiben egyező szekvenciát egyetlen karaktersorozattal írunk le. (A dzsókerek használatának szabályait a HTSCAN rekurzív víruskereső programnál részletesen ismertetjük, mivel a két adatformátum egymásnak megfelel.)

A jobb érthetőség kedvéért most a program írójának, Frans Veldmannak a példáit ismertetjük.

```
A5E623CB??CD21??83FF3E
```

A kérdőjelek helyén, mint a DOS-ban, bármilyen hexa érték állhat, de mindegyik helyén csakis egy-egy.

Használható dzsókernek a csillag is. Szerepe egy kissé bonyolultabb, mint a DOS esetében (lásd a HTSCAN-nél).

```
A5E623CB*3CD2155??83FF3E
```

A fenti sorokkal leírt vírusazonosító:

```
A5E623CB142434CD21554583FF3E
```

A program egy .DAT adatállományon belül maximálisan 500 vírus szignatúraállományának kezelésére képes. Egy vírusszignatúra maximálisan nyolcvan karakter hosszú lehet. Az adatállomány maximálisan 64 kbájt lehet, a vírusneveket is 30 karakteres hosszban maximálta a szerző. 15 szintű könyvtárstruktúráig alkalmazható, és fut az MS–DOS 5.xx verzión is.

Az újabb TBSCAN.DAT állományok végén megjegyzésként látható néhány azonosító adata, amelynek képzési szabályait a szerző nem pub-

likálta. Arra szolgál, hogy a TBSCAN család programjainak futtatásakor az hiteles (authorized) azonosítókat használjuk. A rezidens scanner normál üzemmódban csak ezeket az állományokat fogadja el, ha tehát a sajátunkat akarjuk futtatni, akkor az –u kapcsolót kell használni.

```
;DO NOT DISTRIBUTE MODIFIED COPIES OF THIS FILE!
;This file carries a sanity check. Do NOT change
the checksum! Add your own.
;info BELOW the cheksum line at the bottom of this
file.
;TITCV C9C7
```

A fenti üzenetet soha ne módosítsuk, saját szekvenciáinkat utánaírhatjuk!

A program hibaüzenetei:

```
+ Not enough memory.
```

Nincs elég szabad memória. Legalább 128 KB kell!

```
+ Error in data line at line <number>.
```

Az adatállomány jelzett sorában hibát követtünk el.

```
+ Failed to find DOS entry point.
```

Nem találja a megfelelő DOS belépési pontot. Folytathatjuk, ha a /D kapcsoló nélkül újraindítjuk a programot. Például a DR DOS őrjíti meg a rendszert ekképpen.

```
+ Error reading bootsector.
```

Nem találja vagy nem tudja olvasni a bootrekordot.

```
+ Limit exceeded.
```

Az adatállomány túl hosszú ahhoz, hogy a program feldolgozza (64 kbájtnál nagyobb).

```
+ Data file not found.
```

Nem találja az adatállományt.

```
+ Command line error.
```

Elrontottuk a használható kapcsolók szintaxisát vagy elgépeltünk valamit.

```
+ No matching files found.
```

Vagy az elérési út, vagy a megadott állomány nem létezik, esetleg elgépeltünk valamit.

```
+ No matching executable files found.
```

A megadott útvonal nem létezik, üres, vagy az ott vizsgálni kívánt állományok nem végrehajtható programok, illetve 0 bájtosak.

Ha .BAT állományból futtattuk, akkor a program a következő hibaüzeneteket (DOS errorlevel) adja vissza:

0 — Nem talált vírust.

1 — Legalább egy állományban vírust talált.

2 — A parancssor vagy az útvonal megadása nem pontos.

A TBSCAN sorozat speciális, nem szekvenciális víruskereső algoritmusát alaposan teszteltük. A tapasztalat azt mutatja, hogy minden szép meggondolás ellenére minimális manipulációval alaposan átverhető. Például, ha nem egyszerre ugrik az állomány elejéről a vírusra, hanem apránként ugrál, akkor a program úgy átugorja a vírust, hogy meg sem találja. (Ami tiktak, azaz többszörös fertőzés esetén könnyen előfordul.) Ilyenkor csak az /A opció segít szekvenciális kereséssel. Egyes furcsa szerkezetű vírusokat, például egy hazánkban átírt potyogós változatot is, szó nélkül futni hagy, ha nincs benne a szekvencia. Eljárása tehát csak részben alkalmazható, egy nagy rendszer elemeként, és nem önmagában. A változatos vírusvilág miatt nincs egyedüli üdvözítő megoldás, eltérő elveken alapuló többféle eljárással kell dolgozni ahhoz, hogy a szükséges és elégséges biztonságot megkaphassuk.

TBSCANX 1.4 rezidens víruskereső programcsalád

A TBSCANX programcsalád is Frans Veldman műhelyéből került ki. A program által használt összes eljárás és az adatállomány formátuma teljesen azonos a nem rezidens változatéval. Ugyanakkor ez az a program, amelyet az eddig ismert rezidens programoknál alaposabban felkészítettek az eltérő számítástechnikai környezetekben való munkára. A rendszer vizsgálati eredményére egy másik programból közvetlenül rákérdezhetünk.

A TBSCANX családot felkészítették a multitaszk üzemre. Például Desqview alatt is mindig az aktuális videolapra, azaz valóban a monitor képernyőjére írja ki a vírusokra vonatkozó információit és egyéb üzeneteit.

A géptípus processzorkódjára optimalizált fordítás segítségével mindig a következő verziókat bocsátják ki:

TBSCANX.COM — Alapverzió, amely minden processzortípuson egyformán fut. Normál fordítással készül. Ismeri az MS-Windows 386 védett üzemmódját (386 enhanced mód, amikor a Windows-t a /3 opcióval, 386-os gépen indítjuk). Jóval több memóriát használ és lassabb, mint a géptípus-processzorkódra optimalizált változatok.

TBSCANX.286 — Átnevezendő .COM állományra. A programot a NEC-V20, NEC-V30, 80286, 80386, 80486 processzorok kódjára optimalizálták. Az általános verzióval ellentében nem fut az MS-Windows kiterjesztett üzemmódjában. Csak 100 bájtot vizsgál programonként, éppen ezért gyorsabb, de megbízhatatlanabb is.

TBSCANX.386 — Átnevezendő .COM állományra. A 80386 és 80486 típusú processzorkódra optimalizált változat. Az MS-Windows védett üzemmódjában természetesen működik. Gyorsabb, mint a normál verzió, de lassabb, mint a 286-os változat. Mindenesetre használható. A memóriavizsgálat a standard módú változatnál lassabb.

A program használatához a CONFIG.SYS-be meghajtóként beírandó:

```
device=TBSCANX.COM [kapcsolók]
```

Normál .COM programként is kiadható a parancs:

```
TBSCANX [kapcsolók]
```

-?	— Megjeleníti az opciókat (HELP).
-d	— Letiltja a TBSCANX működését, miközben a program a memóriában marad.
-e	— Engedélyezi a TBSCANX működését, miközben nem tölti be újra a memóriába.
-r	— Kipucolja a TBSCANX programot a memóriából.
-f *állománynév*	— Megadott szignatúraállományt használ.
-o	— Optimalizálja a szignatúraállományt, hogy kisebb helyet foglaljon el a memóriában.
-me	— Az expanded memóriát használja.
-mu	— A felső memóriatartományt használja.
-mh	— A Hercules-half féllapnyi memóriáját használja.
-mf	— A Hercules-full teljes memóriáját alkalmazza.
-mc	— CGA/EGA/VGA memóriát használ.
-u	— Más szignatúra (nemcsak a TBSCAN.DAT ellenőrzőösszegével ellátott adatállomány) használatát is megengedi.

A program alapverziója megdöbbentően kevés helyet foglal el a memóriából: a magára rátöltött és tömörített szignatúraállománnyal együtt 8 kbájtot, sőt extra memóriaspecifikáció (például EMS) esetén mindössze 0,8 kbájtot.

MS-Windows indítása előtt a víruskeresőt kell lefuttatni. Ha támogatja a Windows használatát, akkor annak indítását észlelve automatikusan elvégzi a szükséges memóriaműveleteket.

Néhány, nem önmagáért beszélő, opciója:

-F — Akkor kell megadni ezt a kapcsolót, ha device driverként töltjük be a programot a CONFIG.SYS állományból, vagy más nevű programot töltünk be, programként futtatva. A szignatúraállományt abban a könyvtárban keresi, amelyben a program elindításakor voltunk.

-O — Jó néhány százalékkal csökkenti a memóriában a szignatúraállomány helyfoglalását egy pofonegyszerű trükkel. Ha két szignatúrát veszünk, annak biztos van közös eleme.

```
1. szekvencia: CD2145A689BF452F1E77CBCD21

2. szekvencia: CD2111A689BF4A3F1E77CBCD21
```

Ezek 75%-ban megegyeznek. A nem rezidens TBSCAN és a HTSCAN adatállomány formátumánál (ld. ott) szerepel, hogyan lehet dzsókerekkel leírni az ilyen közös szekvenciákat. E szabályok alkalmazásával a fenti kettőből előáll a harmadik, egyetlen szekvencia:

```
3. szekvencia: CD21??A689BF????1E77CBCD21
```

Ezzel az eljárással a rossz felismerés, sőt a vírus elnézésének veszélye is jócskán megnő.

A programhoz tartozó Assembly nyelvi illesztés hasonló, mint ami az általunk kifejlesztett TVG vírusvédelmi kártyában van, de a komolyabb védelmi rendszerek előbb-utóbb a világon mindenütt ilyen programozói–alkalmazói felületeket fognak a fejlesztők rendelkezésére bocsátani.

Program interfész Assembly nyelvhez

Ez a program eddig az egyetlen, amelynek más programokból elérhető nyelvi intefésze van, és ezt a készítő közkinccsé is tette. Az interfész az Assembly nyelvet használja fel, s a rutinok bármelyik magasabb szintű nyelvbe beágyazhatóak.

Az illesztőfelület számos többszöri hívást tartalmaz (int 2Fh). Az AH regiszter tartalmazza a CAh értéket, míg az AL regiszterben találhatjuk a funkciót meghívó kódot (request number). Lehetőségeink:

```
AL=0          InstallationCheck

              Visszatérési érték (return value):
```

AL=0 TbScanX nincs a memóriában

AL=FFh TbScanX ott van a memóriában

Ha a BX regiszterben 'TB'-t találunk, akkor ez lecserélhető 'tb'-re.

AL=1 GetStatus

Visszatérési érték:

AH Verziószám: TbScanX in BCD. (CAh ha ez kisebb
mint 2.2)

AL=0 TbScanX munkája letiltva

AL=1 TbScanX műveletei engedélyezve

BX Segment swap area. Ha nulla, akkor nincs swap funkció.

CX A keresett szignatúra száma

DX EMS_Handle. Ha értéke -1, akkor nem használja az expanded memóriát.

AL=2 SetStatus

BL=0 TbScanX letiltva

BL=1 TbScanX engedélyezve

Visszatérési értéke NINCS!

NONE

AL=3 ScanBuffer

DS:DX A scannelésre használt puffer címe

CX A scannelésre használt puffer hossza

Visszatérési értékei:

No Carry flag set Éppen nincs szignatúra. a pufferben.

Carry: Szignatúrát talált a pufferben!

ES:BX ASCIIZ-vírus neve (null terminated)

Megváltozó regiszterek:

AX,BX,CX,DX,ES

A puffertartalom változatlan marad

AL=4 ScanFile

DS:DX Az éppen vizsgált állomány neve

Figyelem! Ez a funkció csak 4 KB szabad memória esetén hívható meg!

Visszatérési értékei:

No Carry flag set Nincs vírusszignatúra az állományban

Carry: Szignatúrát talált a pufferben!

ES:BX ASCIIZ-vírusnév (null terminated)

Megváltozó regiszterek:

AX,BX,CX,DX,ES

Vegyünk (lopjunk) egy példát Veldmantól:

```
mov ah,0CAh  ;Multiplex number
mov al,0
int 02Fh     ;Installáció ellenőrzés
cmp al,0FFh  ;ha AL=FFh TbScanX nem installált
jne notinstalled ;menj oda ha a TbScanX nincs a
             ;memóriában
lea dx,buffer  ;A puffer címe a DS:DX
             ;regiszterben
mov cx,512   ;Puffer hossza
mov ah,0CAh  ;Multiplex number
mov al,3
int 02Fh     ;ScanBuffer
jnc notinfected ;Nincs carry? Ha nem, akkor
             ;nincs vírus!
call print   ;Vírus van jelen. Írasd ki a
             ;nevét a ES:BX regiszterből
             ;ASCII formátumban
notinfected:
```

A szoftver viszonylag kevés programmal akad össze. A magyar Ventura régi és új védelemmel ellátott verziójával ütközik, az MS-Windows-t viszont kiválóan tűri. 386-os gépen, ha device driverként van betöltve,

és utána installáljuk a QEMM386-os Quaterdes memóriakezelő programot, akkor teljesen felmegy a HIMEM-be, és nem foglal el memóriát.

Sok esetben éppen a gyorsaság miatti kompromisszumok következtében elnézi a vírusokat. Így használata csak más programokkal együtt ad megfelelő biztonságot.

A HTScan programozható rekurzív víruskereső

Szerzőként Harry Thijssent jegyzik. A program nem éppen friss, 1990. 06. 05. dátumú, és azóta nem is jelent meg új verziója. A TBSCAN sorozat lesöpörte a prorondról.

A HTSCAN-nek hasonló a szabad memóriaigénye, mint a Scan sorozat tagjainak, minimálisan 256 kbájt szükséges a futásához. A merevlemez vagy a floppy egész területén vírusazonosító karaktersorozatokat keres, igen nagy sebességgel. A mintát egy szövegszerkesztővel készített ASCII állományból veszi, amelynek neve HTSCAN.DAT vagy VIRSCAN.DAT.

A HTScan programot igen sok azonosító string befogadására tervezték: a vírusazonosítók száma maximálisan 4000 lehet. A fejlesztő dokumentációja szerint a vírusazonosítók mennyisége nem befolyásolja a program futási idejét. A tesztek alapján tényleg nincs lényeges időkülönbség 1 vagy 100 vírusazonosító használata esetén. A HTScan a vírusazonosítókat tartalmazó állományt minden futtatás előtt külön állományból olvassa be. A vírusazonosító állományok tartalmát a felhasználó tetszőlegesen módosíthatja, illetve új vírus esetén bővítheti. Ezzel a módszerrel lehetővé tették a víruskereső programozhatóságát. Elindítása után a program a HTSCAN.DAT állományt keresi az aktuális könyvtárban. Ha nem talál ilyen nevűt, akkor a VIRSCAN.DAT állományt keresi. Ezek egyikének feltétlenül léteznie kell, vagy helyette más nevű állományt is kijelölhetünk szignatúraállományként.

A program használata:

```
HTScan <path> [<path>…] [opciók…]
```

Opciók:

/A — Az összes állomány ellenőrzése az összes vírusra.

/D — A fertőzött állomány törlése/átnevezése. (Mielőtt a HTScan program törölné/átnevezné az állományokat, kijelzi azokat.)

/I — A program copyright információjának megjelenítése.

/R — A fertőzött állományok átnevezése, előzetes rákérdezéssel.
/N — Nem ellenőrzi az alkönyvtárakat.
/O[=]<log.file> — A vírusellenőrzés eredményét a megadott állományba teszi.
/S — A bootszektor átlépése (hálózati használatban).
/V[=]<sign.list> — A megadott állományt használja vírusazonosító szignatúraállományként.
/X — Több floppylemez ellenőrzése.

Programkilépési kódok:
0 — Nem volt vírus.
1 — Egy vagy több vírus volt.
2 — Hibás programfutás.

A vírusazonosító állomány (HTSCAN.DAT, VIRSCAN.DAT) programozása

Vírus neve:
Bármilyen, 1-től 80 karakterig terjedő ASCII karaktersorozat.

A vírusazonosító string állomány hatásköre:
PART, BOOT, SYS, COM, EXE, OVL, BIN, PIF kiterjesztés bármelyike vagy akár mindegyike felsorolva, szóközzel elválasztva. Értelmezésük:
PART — Partíciós tábla, csak merevlemezre.
MAIN — A partíciós tábla szinonimája.
BOOT — Bootszektor.
SYS — *.SYS állományok.
COM — *.COM állományok.
EXE — *.EXE állományok.
OVL — *.OVL állományok.
OV* — Overlay-állományok.
BIN — *.BIN állományok.
PIF — *.PIF állományok.

A vírusszignatúrák leírásának általános szabályai

A vírusszignatúrák összefüggő, tetszőleges hexadecimális számok. A vírusazonosító minimális hossza 8, maximális hossza pedig 80 karakter

lehet. Az első karakter után a ?, * helyettesítő karakterek használata megengedett.

? — A kérdőjel helyén bármilyen (akár félbájtos) érték lehet.

*x — A következő x bájt értékét a program figyelmen kívül hagyja az ellenőrzés során. Az x hexa 1 és hexa F közötti értéket vehet fel.

A *x értékkel megadott bájt után, a következő bájt ismét tartalmazhat *x-et, de nem tartalmazhat ?-et. A maszkolt vírusazonosító string hossza nem lehet nagyobb mint, 128 bájt.

Lehetőség van a vírusban elhelyezett szöveg keresésére is. A keresendő szöveget dupla kérdőjelek között kell elhelyezni. (Ilyen szabályok szerint közöljük jó pár vírus azonosító karaktersorozatát.)

A Safe Kft-n keresztül forgalmazott, önköltségi áron terjesztett példányokhoz rendszeresen bocsátunk ki a magyar vírusokkal is aktualizált és tesztelt szignatúraállományokat, amelyek a magyar vírusfajtákkal és mutánsokkal együtt több mint 180 vírus felismerésére alkalmasak.

NORTON A VÍRUSSZÍNRE LÉP

Nem mind arany...

Peter Nortonnak az Apple programokkal hírnevet szerzett *Symantec* cégnél készített terméke a Norton AntiVirus 1.0 programcsomag (NAV), amely 1990 végén jelent meg bétatesztes formában, majd 1991-től kereskedelmi forgalomban. Bár a cég az Apple vírusok kitakarításában élen jár, a PC-világban még van mit tanulnia. Ez a program is csak másodlagos eszközként alkalmazható, elég sok megszorítással.

A probléma ott kezdődik, hogy Európában nem megoldott a programnak az újabb vírusok ismeretével való felvértezése. Ezt az USA-ban úgy oldják meg, hogy egy BBS-ről letölthetőek a friss vírusfelismerő és vírusirtó definíciós állományok. Akinek van lehetősége (telefonmodemje és anyagi fedezete a telefonszámlára), annak érdemes rendszeresen felhívnia ezeket az információs rendszereket. (Telefoszám: 00-1-408-973-9598, Symantec BBS, 2400 baud 8N1. Az angol nyelvű automata vírushírszolgálat: 00-1-408-252-3993.)

Már az eredeti programlemezen is találunk egy WHALE.DEF bővítőállományt, amely a Whale (Bálna) újgenerációs vírus mutációinak felismeréséhez szükséges információkkal bővíti a program tudását. Megjelenése óta ennek a könyvnek a megírásáig a következő javítások és korszerűsítések készültek a programhoz, amelyek a magyar szoftvercsatornákon hozzáférhetőek.

UPDATE01.DEF — 1991. január 25-én bocsátotta ki a Symantec a NAV-hoz. Néhány definíciót — szokás szerint — alaposan elrontottak. *Tartalma:* 12 Tricks trojan, Invader és Plastique két új verziója, Keypress, az AIDS-2 elleni felismerési algoritmusok. Ezenkívül a Whale állomány javításaként megjelent benne három új Bálna-mutáns detektálásához szükséges definíció: Whale G28, Whale G29, Whale G30.

UPDATE02.DEF — 1991. március 27-én bocsátották ki a felhasználói reklamációk hatására. Ebben módosítják az UDTADE01.DEF-ben megadott szekvenciák és irtási eljárások egy részét, de újabb vírusok ismeretével is bővítik a készletet:

Aids, Aids-2, Alabama, Ambulance (RedX), Amoeba (1392), Amstrad, Anarkia, Anarkia-2, Anti-Pascal 1, Anti-Pascal 2, Anticad (1253) Beeper, Best Wish, Black Monday, Brain-A/B, Burger, Cascade (170X)

related, Christmas (XA1), D31 Dark Avenger, Data Crime II, Data Crime II-B, Datacrime (1168), Datacrime (1280), DBase, Dear Nina, Den Zuk, Destructor Devil's Dance, Disk Killer, Do Nothing 640k, E.D.V. Eight Tunes (1971) Evil/V1226 related, F-Word, Father Christmas, Fellowship, Fish, Flash, Flip, Form, Frere Jacques, Frodo (4096), Fu Manchu, Ghostballs, Greek (Armageddon), Halloechen, Hammelburg (405), Happy Day, Icelandic-1, Icelandic-2, Icelandic-3, Invader, Italian-A, Italian-B, ItaVir, Jerspain, Jerusalem-A, Jerusalem-B related Jerusalem-D, Jojo, Joshi, July 13th, June 16, Kamikazi, Kennedy, Keypress, Korea LBC, Leapfrog, Lehigh, Leprosy-A, Leprosy-B, Liberty, Lisbon-A/B, Lovechild, Lozinsky, Mardi Bros, Mendoza, MG-1, MG-2, MG-3, MGTU, Microbes, MIX1-A, MIX1-B, Murphy 1/2, Nina, Nomenklatura, Number of the Beast (512) Off Stealth, Ontario, Oropax, Paris, Parity, Pentagon, Perfume, Pixel related, Plastique, Plastique 5.21, Print Screen, Prudents (1210), PSQR (1720), Saratoga, Saturday the 14th, Shake, Slow, Solano, Sorry, Stoned, Subliminal, Sunday, Suomi (1008), Suriv 1, Suriv 2, SVIR, Sylvia, Syslock (3551) Taiwan, Taiwan-2, Tel Aviv (1605), Tiny related Tiny-133, Tiny-158 related, Tiny-167, Tiny-198, Traceback (2930/3066), Trojan 1381, Typo Boot, Typo, Unknown Jerusalem-B, Unknown Plastique, Unknown Yankee Doodle, USSR-256, 37, USSR-257, 30, USSR-394, 65, USSR-600, 47, USSR-696, 41, USSR-711, 62, USSR-948, 94, V1024, V2000, V2100, V2P2/1260/Casper, V2P6(Z), V651 (Eddie 3), V800, Vacsina related, Valert (1554) Vcomm, Victor, Vien6, Vienna VHP-348 Vienna VHP-353 Vienna VHP-367 Vienna VHP-435 Vienna VHP-623 Vienna VHP-627 Vienna, Virdem, Virus 101, Virus 90, Virus-B, Voronezh 2.01, VP, W-13 A/B, Westwood, Yale/Alameda, Yankee Doodle TP33, Yankee Doodle TP34, Yankee Doodle TP38, Yankee Doodle TP41, Yankee Doodle TP42, Yankee Doodle TP44, Yankee Doodle TP45, Yankee Doodle TP46, Yankee-2 (1961), Zero-Bug, Zero-Hunt, 12 Tricks trojan, 492, 529, 691.

Még ugyanezen a napon jelent meg az UPDATE03.DEF, amelyben kijavították a Kamikazi vírus felismerését (túl sok volt a vakriadó a Turbo Pascal 6.0-val fordított állományokban egy rosszul megválasztott keresési szekvencia miatt), és az 1591 jelű vírussal bővült a választék. Ha mindegyik bővítést installáltuk, és a szükséges definíciócseréket elvégeztük, akkor kaptunk egy közepes minőségű, bár nagyon könnyen kezelhető detektáló-irtó programot.

A vevőszolgálat az első javítás kibocsátása után komoly munkát végzett: a program vírusismeretét kibővítette az ilyen programoknál szokásos mértékre. (Mint kiderült, korábban nem volt elegendő éles vírusuk, azokat csak később tudták pótolni.)

Most lássuk, milyen változtatásokat kell elvégezni a gyári lemez installálása után a megfelelő használhatóság érdekében!

Először is a vírusdefiníciók betöltése menüponttal be kell tölteni a WHALE.DEF állományt. Akkor végeztük el jól a műveletet, ha tartalma megjelenik a víruslistában. Utána a vírusdefiníciók törlése menüponttal ki kell irtani az AIDS-2 definíciót. Itt ugyanis rosszul adták meg a tárrezidens vírusrész adatait, így a vírust nem jelzi a program. Ezután be kell tölteni az UPDATE01.DEF-et. Majd ki kell irtani a következő vírusok definícióját:

Anarkia-2, Icelandic-1, Icelandic-2, Icelandic-3, Italian-A, Italian-B, Leprosy-A, Leprosy-B, MIX1-A, MIX1-B, Taiwan 2, Virus B.

Utána betöltjük az UPADATE02.DEF állományt, kitöröljük a listából a Kamikazi vírust (ha benne van), majd betöltjük az UPDATE03.DEF állományt. Windows alatti futtatáshoz a forgalmazó a következő tartalmú .BAT állományt ajánlja:

```
NAV /C+ /X+ /S+ /O+
WIN
NAV /X- /S-
REM A windows képernyőműveletei miatt kikapcsoltuk az
REM üzenetablakot, és a helyébe speciális
REM jelzőhangot aktiváltunk.
REM Kilépéskor visszaáll az eredeti állapot.
```

A Norton AntiVirus szép reményekre jogosító program. Csakis akkor tehet szert azonban nagyobb népszerűségre és akkor terjedhet el, ha a forgalmazó kiadja a bővítéshez alkalmazott programnyelvet, hogy minden országban elkészülhessenek a megfelelő kiegészítések a helyi vírusokról és vírusváltozatokról.

Szintén problémás, hogy a víruslista megtekintése után nem lehet eldönteni, ki tud-e irtani a program egy bizonyos vírust, vagy csak detektálja azt. Erről az ad felvilágosítást, hogy felajánlja-e a REPAIR parancsot.

A NAV.SYS állományt az ajánlással ellentétben nem szabad bekötni rendszermeghajtóként a CONFIG.SYS állományba! Ekkor ugyanis teleszemetel minden könyvtárat a maga hidden system ellenőrzőállományaival, amit minden futtatható programról hasonló néven elkészít. Ezek hossza 77 bájt, a kiterjesztés egyik karaktere pedig az aláhúzás jel. Ráadásul elég idegölő, amíg mindezt létrehozza.

Hálózatokon vírus keresésére vagy floppykon biztonsági ellenőrzésre például a PC-Scan vagy a McAfee-féle Scan programmal együtt alkalmazva nyújt elegendő biztonságot.

A többi Norton-programhoz hasonlóan a NAV is lemezről installálható. Használata egyszerű, értelemszerű. Ugyanakkor nem szabad elfogadni, hogy az installálási procedúra során, módosítsa a CONFIG.SYS és az AUTOEXEC.BAT állományokat. A program a megadott meghajtón a NAV alkönyvtárba helyezi el magát. Indítható menüvezérléssel, a NAV parancs begépelésével vagy parancssorból, paraméterezésel. Ebben ez esetben mindig memóriavizsgálatot végez, és figyelemmel kísérhetjük, hogy a mi példányunk milyen vírusokat tud megkeresni. A program másolásvédelem nélküli, csupán az első installáláskor beírja magába nevünket, cégcímünket. Paraméteres üzemmódban is mindig kiírja ezeket az adatokat. Nem szabadszoftver, Magyarországon a kereskedelemben (például a Cédrus Rt. kínálatában) a többi Norton-program között szerepel.

A program parancssori opcióit egy help képernyő alapján mutatjuk be:

The Norton AntiVirus
Copyright (c) Symantec Corp 1990. All rights reserved.
Version 1.0.0
This copy is licensed to:
Kis János
Cédrus Informatikai Rt., Budapest
Command line options:

NAV /?	— Help (megjeleníti a parancsori parancsokat).
NAV /A	— Minden meghajtót vizsgál.
NAV /M{M/C}	— Monokróm/színes üzemmód-átkapcsoló.
NAV /S{+/-}	— Engedélyezi/tiltja a speciális riasztó hangot.
NAV /O{+/-}	— Engedélyezi/tiltja a speciális riasztó rendszerüzenetet monokróm monitor esetén.
NAV /X{+/-}	— Engedélyezi/tiltja a speciális riasztó rendszerüzenetet színes monitor esetén.
NAV /C{+/-}	— Engedélyezi/tiltja a kompatibilitási üzemmódot.
NAV {Drive:}	— Megadott meghajtót vizsgál.
NAV {Directory}	— A megadott könyvtárat és az alatta lévő könyvtárfát vizsgálja.
NAV {Filename}	— A megadott állományokat vizsgálja. Dzsókerek (* és ?) használhatók.

Példa:

NAV /S+ /O- /X-	— Hang engedélyezve, rendszerüzenet monokróm és színes monitoron is letiltva.

A program egyszerűen kezelhető. Beállításai jelszóhoz kötötten elmenthetők, ezt módosítani csak a jelszó birtokában lehet. Vírusriasztás

esetében megadható az a szöveg is, amit ki kell írnia, például arra lehet kérni a felhasználót, hogy értesítse a rendszergazdát. Fontos tudnivaló: parancssorból csak keres.

Annak nincs értelme, hogy egyenként felsoroljuk, mely vírusokat tud irtani, és melyeket csak detektálni, mert a megjelenő UPDATE állományok remélhetőleg alaposan átformálják vírusismeretét. Nem ártott volna piacra bocsátása előtt egy kicsit érlelni ezt a meglehetősen ellentmondásos programcsomagot! Felületesen megírt felismerő és irtó algoritmusok az alapverzióban, a vevőszolgálat és szoftveres követés szervezetlensége és hiánya, a könyvtárak teleszemetelése rejtett állományokkal, a 286-os és az XT gépeken a system állomány kompatibilitási és futási problémái — mindez meggátolta, hogy ez a program Európában feliratkozzon a sikerprogramok listájára. Az új definíciós állományokkal azt legalább már elérték, hogy a program a korábbinál kevesebbet téved.

A magyar vírusok és a kelet-európai eredetű nagyobb vírustörzsek hiányosan vagy egyáltalán nem találhatóak meg e program vírusismeretében. Az az ellenőrzőösszegen alapuló általános eljárása például az újgenerációs vírusok esetében teljesen hatástalan.

AZ F-PROT CSOMAG

Válogatás egy izlandi svédasztalnál

A vírusirtás nemzetközi gyakorlatában alapszoftvernek számít az izlandi eredetű, szabadszoftverként elérhető F-Prot. Sok apró programból áll. Koncepciója azt az irányzatot tükrözi, amelyet annak idején az Antivir programcsomag elkészítésekor mi is alkalmaztunk. Hátránya: alapos programozási és rendszerismeret szükséges a használatához. Számos programja kitűnően alkalmazható, de vannak olyanok is, amelyek nem ajánlhatók, vagy kifejezetten kockázatosak. Bár nem akad össze sem a Venturával, sem a Windows programokkal, SYS állományát csak akkor használhatjuk, ha elfér a memóriában.

Talán ez az a külföldi program, amely a legtöbb hazai vírusverziót felismeri és részben irtja is, ami annak köszönhető, hogy a program írójának jók a kapcsolatai Kelet-Európa országaival. Víruslistája:

Agiplan, Alabama, Ambulance, Amoeba, Amstrad, Anthrax, AntiPascal, AntiPascal2, Armagedon, Attention, Bebe, Bestwish, Black Monday, Blood, Brain, Burger, Cancer, Carioca, Cascade, Christmas-J, Crazy Eddie, Dance, DataCrime A, DataCrime B, DataCrime-2, Datalock, dBase, Dec. 24, Den Zuk, Destructor, Diamond, DIR, Disk Killer, Doteater, Durban, Dyslexia, E.D.V., Eddie, Eddie-2, Fellowship, Filler, Fish 6, Flash, Flip, Flip Boot, Form, Frodo, Fumble, Guppy, Halloechen, Icelandic, Internal, Itavir, Jerusalem, Jerusalem, Jo-Jo, Joker-01, Joshi, Kemerovo, Kennedy, Korea, Lehigh, Leprosy, Liberty, Lisbon, Lozinsky, MG, MGTU, Mix1, MLTI, Murphy, Music Bug, New Vienna, Nichols, Nina, Nomenklatura, Ohio, Old Yankee, Oropax, Parity, Perfume, Phoenix, Ping-Pong, Ping-Pong 2, Piter, Pixel, Plastique, Polimer, Pretoria, PrintScreen, Prudents, Saddam, Shake, Slow, South Afr., South Afr., Stoned, Stupid, Suomi, Suriv 1.0, Suriv 2.0, SVC, Sverdlov, Svir, Swap, Sylvia, SysLock, Taiwan, Tiny, Tiny-family, Traceback, TUQ, Turbo, Turku, Typo, V-1, Boot, Vacsina, Valert, Vcomm, VFSI, Victor, Vienna, Virdem, Virus-101, Virus-90, Voronezh, VP, W13, Wisconsin, XA1, Yale, Yankee, Zero Bug, Zero Hunt, 13J, 217, 405, 417, 440, 492, 500, 512, 512, 516, 600, 696, 707, 711, 800, 948, 1049, 1067, 1075?, 1077, 1260, 1600, 2144, 2480, 5120, 8-Tunes.

Az F-PROT 1.14 csomag programjai

F-DRIVER.SYS — Az ismert vírusokat keresi, és nem engedi elindulni azt a programot, amelyikben vírust talál. Elégséges biztonságot nyújt, jól használható. A CONFIG.SYS-be a következő sort kell beírni:

```
DEVICE=C:\F-PROT\F-DRIVER.SYS
```

Természetesen azt az útvonalat kell megadni, ahol a többi részprogram is elérhető, s azok legyenek ugyanabban az alkönyvtárban.

F-OSCHK.EXE — Ellenőrző összeget készít a bootról, a partíciós tábláról és a rendszerállományokról. Az újabb generációs vírusok egy része átveri. Először paraméter nélkül kell futtatni. Ekkor megadja azokat a számokat, amelyeket paraméterként be kell írni az AUTOEXEC.BAT állományba, ekképpen:

```
c:\F-PROT\F-OSCHK vvvvv wwwww xxxxx yyyyy zzzzz
```

Az egyes DOS verziókban a következő állományokat ellenőrzi:

PC-DOS	MS-DOS	DR-DOS
IBMBIO.COM	IO.SYS	DRBIOS.COM
IBMDOS.COM	MSDOS.SYS	DRBDOS.COM
COMMAND.COM	COMMAND.COM	COMMAND.COM

F-LOCK.EXE — Vírusfunkciót detektáló program, általános vírusvédelemre hivatott. Van nála hatásosabb, használatát nem ajánljuk.

F-SYSCHK.EXE — Ismert vírusokat keres a memóriában. Használata célszerű.

F-FCHK.EXE — Fertőzött állományokat keres, és azokból eltávolítja az ismert vírusokat. Használata ajánlható.

Opciói:

/LIST esetén az F-FCHK program jegyzőkönyvet készít a vizsgált állományokról, illetve megjeleníti azok listáját. A /LIST opció a DOS „kacsacsőr" karakterével átirányítható a megadott állományba:

```
F-FCHK C: /AUTO /LIST > REPORT.LIS
```

Ha a /AUTO kapcsolóval automatikus helyreállítást kértünk, amíg ez tart, a monitoron — néha bizony elég hosszan — a CURED... felirat látható. Ha kudarcot vall, akkor a következő rendszerüzenetet kapjuk:

```
Virus could not be removed.
```

A /MULTI kapcsoló az F-FCHK és az F-DISINF programok esetében több floppy egymás utáni vizsgálatát teszi lehetővé.

Az /ALL kapcsolóval az ellenőrzést minden állományra elvégzi.

F-DISINF.EXE — Eltávolítja a bootszektorból a fertőzést, ha azt ismert vírus okozta. Nem szabványos DOS-szal, például a DR DOS-szal is jól működik. Ilyenkor helyettesíti a McAfee-féle Clean és M-Disk programokat. Használata ajánlható. Kapcsolói ugyanazok, mint az F-FCHK programnak, ezenkívül a /AUTO automatikusan helyreállítja a károsodott programot, ha tudja. Természetesen egyszerre több kapcsoló is alkalmazható.

F-XLOCK.EXE — Ellenőrzőkódot ad a programokhoz, és többségüket tönkreteszi! Használata kerülendő.

F-UNLOCK.EXE — Az F-XLOCK.EXE programmal felrakott „vírusellenes farok" eltávolítója. Az önvédelemmel ellátott programok utána nem működnek, a másolásvédett szoftverek pedig vagy pusztítanak vagy csak nem működnek. Használata kerülendő.

F-XCHK.EXE — Csak a vírusellenes farokkal ellátott programokat engedi futni. Használata kerülendő. Mellette másik program nem indítható, csak az F-RUN.EXE segédprogrammal.

F-RUN.EXE — Ha az F-XCHK.EXE aktív, akkor ez a program-engedélyezi a „vírusellenes farok" nélküli programok futtatását. Használata kerülendő.

F-INOC.EXE — A bootszektort támadó egyes vírusok ellen állítólag immunizál. Nincs sok értelme a használatának.

F-POPUP.EXE — Kiválasztó képernyős program, az F-LOCK és az F-DLOCK használatához.

F-DLOCK.EXE — A merevlemezt védi a formázás és az írás ellen. Néhány vírus becsapja, ezért használata hamis biztonságérzetet nyújt. A formázó másolásvédelmek egy részénél a feltörés tesztelésére viszont alkalmazható. Egyébként kerülendő!

F-EX.EXE — Eltávolítja a memórából a programcsomag rezidens részeit, ha opcióként melléírjuk a kitejesztés nélküli neveket.

F-DIR.EXE — A hidden és system állományokat is kijelzi. Jól alkalmazható program, ha nem használunk Norton Commandert.

F-MMAP.EXE — A memóriatérképet mutatja a hooked vektorokkal. Jól sikerült segédprogram.

F-HIDE.EXE — Rejtetté tesz állományokat.

F-UNHIDE.EXE — Rejtett állományokat normális állományokká alakít.

F-BOOT.EXE — Megnézhető vele a bootszektor.

F-PBR.EXE — Megnézhető vele a pertíciós tábla.

F-NET.EXE — Novell-kiegészítés hálózatos használatra.

SIGN.TXT — A vírusdefiníciókat tárolja. Abban a könyvtárban kell lennie, ahol a programcsomag többi programja is van.

Ez a programcsomag nem hiányozhat a programozástechnikához értő vagy éppen vírustalanító szakember szoftverei közül. Programjai egyszerűek, s bár némelyik kifejezetten pocsék, többségük kiválóan használható. Külön előnye, hogy felkészítették az egzotikus DOS-verziókra, ami unikum az ilyen szoftverek között.

KÁRTYAJÁTÉKOK

„A vírusok nem kártyakompatibilisak"

Mielőtt részletesen ismertetnénk a vírusok ellen kártyákkal történő védekezés lehetőségeit és eredményeit, érdemes összefoglalni azokat a követelményeket, amelyeket egy jó vírusvédelmi kártyával szemben támasztanunk kell.

1. Lehetőleg minél kevesebb döntést bízzon a felhasználóra.
2. Csak akkor „szóljon", ha valóban baj van, és nem tud mit tenni. Különben maradjon teljesen észrevétlen.
3. Használata ne függjön a felhasználó akaratától, tehát ha bent van a gépben, akkor valóban védjen.
4. Legyen vírusismerete. A „katalógusvírusokkal" való fertőzést minden körülmények között akadályozza meg.
5. Legyen a vírusfunkciók felismerésére alkalmas általános védelme. Ne engedje meg az érzékeny részek (CMOS-RAM, boot, partíciós tábla, rendszerállományok) felülírását, a merevlemez szektorainak formázását.
6. Védő hatása a processzorok minden üzemmódjában érvényesüljön. (Például az MS-Windows védett módját is ismerje.)
7. Ne foglaljon el helyet a memóriából, és már a rendszer betöltődése után, a winchester bootfolyamata előtt aktivizálódjon.
8. Minden winchestertípuson működjön.
9. A vírusok által fizikailag megtámadhatatlan legyen, azaz a kártya lássa a memóriát, de a memóriából ne lehessen látni a kártyát.
10. A kulcsparaméterek tárolására legyen saját memóriája, hogy szükség esetén onnan is lehessen folytatni a bootfolyamatot. Ez akár elemmel védett RAM, akár ROM is lehet.
11. Könnyen felfrissíthető legyen a vírusismerete.
12. Támogassa a hozzáférésvédelmet és az adatbiztonsági szolgáltatásokat.
13. Semmilyen bevett szoftver (például Norton Utility, Norton Commander, védelem nélküli Ventura stb.) futását ne akadályozza és ne nehezítse meg.

14. Felhasználói programokból rendszerhívásokkal elérhető legyen, támogassa a programozókat. Legyen olyan programinterfésze, amely más alkalmazói szoftverekből meghívható. (Ha egy könyvelő program floppyt kér, tudja előtte ellenőrizni annak vírusmentességét.)

Megpróbáltunk a belföldi és külföldi kereskedelmi termékek között olyat találni, amelyik megfelel ezeknek a kritériumoknak. Az említésre sem méltó termékek sora mellett két tanulságos külföldi megoldást láttunk, s azok teszjének eredményét azért is érdemesnek tartjuk közreadni, mert jelen vannak a magyar piacon.

Thunderbyte PC Immunizer

A részben használható kártyák közé tartozik az ESaSS (Electronic Systems and Special Services) cég Thunderbyte PC Immunizer kártyája. Ebből a kártyából két verziót teszteltünk, az 1.00-t a győri Summatech Kft-től, a v1.10-est közvetlenül a fejlesztő cégtől vettük. A legújabb 2.0 verziót a budapesti Ifabón, a cég európai disztribútorának standján volt módunk „megizzasztani".

Az első, a Magyarországon vásárolt példány alaposan megkeserítette életünket: ékes holland nyelvű dokumentációját kínnal-keservvel sikerült megfejtenünk. Nemcsak nyelvi okokból, hanem mert az egyébként is trehány leírás egy soha forgalomba nem került tesztverzió állapotát tükrözte. Az újabb verzióról szerencsére már angol nyelvű dokumentációt kaptunk a fejlesztő cégtől.

A Thunderbyte kártya vírusvédelmi rendszere a kártyára szerelt BIOS-ban lévő szoftverrel és a kártyához adott lemezről installálható programmal működik.

A kártyához sok-sok kis programocskát adtak, amelyek közül a TBINSTAL szolgál a kártya konfigurálására. Ezt a programot a kártyának a gépbe helyezése előtt kell elindítani, hogy a program megállapíthassa a kártya BIOS-ának installálási címét és a kártya I/O címét. A kártyán lévő 1. kapcsolósor, azaz switch, egyes elemeit ennek megfelelően kell beállítani. A kártyán található 2. kapcsolósor a kártya hardverlehetőségeinek aktivizálására szolgál. A kapcsolók beállítása után tehetjük be a Thunderbyte kártyát a gépbe. Már itt megállapítható, hogy a kártya üzemeltetése alapos számítástechnikai ismereteket és a „víruslélektanban" való jártasságot igényel.

Fejlesztői szerint a kártya általános célú: nem egyes vírusok ellen készült, hanem vírusfunkciók ellen nyújt védelmet, tehát vírust nem öl, csak vírusfunkciókat detektál. A kártya BIOS-a a PC BIOS-a után fut le, de még a DOS előtt. Védelmi TBCONFIG.COM szoftverét a

CONFIG.SYS vagy az AUTOEXEC.BAT állományból kell installálni a DOS betöltése során. A merevlemezre írás engedélyezése vagy tiltása a vezérlőkábel megszakításával lehetséges, a kártyát elektromosan a merevlemez és a meghajtó közé iktatva.

A Thunderbyte kártya és a hozzá adott szoftver angol, német és holland nyelven üzenhet. Mi a német és az angol ágat jártuk végig. Ezek következetesek. Az üzenetek a kártyán lévő BIOS-ban vannak, így annak „locsogó" volta miatt vajmi kevés hely maradt az érdemi programokra. A kártya üzenetei a fenti nyelveken nem mindig egyértelmű formában, de minden esetben felhasználói döntéshez kötötték a rendszer további működését. Néhány esetben viszont a vírus e döntéstől függetlenül jön, lát és győz. (Polimer vírus.)

A kártya koncepciója: nem enged más programot (meghajtót) rezidenssé válni, csak azt, amit a meghatározott konfigurációs állományban (SET TB=xxxxxxxx.yyy) megadtunk neki. Figyeli a kártya BIOS-ába fixen beégetett EXE, COM, SYS kiterjesztéseket, és nem enged végrehajtható programokba írni. Mit tegyünk tehát, ha másfajta végrehajtható programunk is van, és védeni akarjuk? Mondjuk a GEM rendszer .APP állományai...? Lehetőségünk van a CMOS setup információváltozásainak figyelésére, továbbá többletszolgáltatásként a számítógép jelszóval is védhető. A jelszó a merevlemez 0. track 6. szektorának elejére íródik, a 2. kapcsolósor állásaival kódolva. Néhány bootvírus is ír erre a szektorra, s ha átmegy, akkor aztán kesereghetünk „kivasalt" merevlemezünk felett. A szovjet eredetű Mirror vírus ezt 5–10 másodperc alatt készséggel megteszi nekünk.

Az élő katalógusvírusokkal végzett „éleslövészethez" használt gépkonfiguráció: 80386-os AT, Phoenix BIOS, TVGA VGA monitorkártya, két merevlemez (egy 80 és egy 40 MB-os), ahol a gépben további 2 Mbájt RAM volt. Az operációs rendszer MS–DOS 4.01, BIGDOS partícióval. A kártya szoftverét (TBCONFIG) a CONFIG.SYS állományba helyeztük. A rezidens program egyformán indítható meghajtóprogramként a CONFIG.SYS-ből, és programként az AUTOEXEC.BAT-ból. Meghajtóprogramként jóval nagyobb a paraméterezési lehetőség, ezért ezt az utat jártuk végig. Mindezeken felül az exetended memóriával dolgozó FASTX programot használtuk a lemezműveletek gyorsítására. A fertőzött programokat DOS-ból és a Norton Commander 3.00 alól indítottuk el.

A probléma már a Norton Commandernél kezdődött. Amikor egy új alkönyvtárat akartunk létrehozni a floppyn, a kártya rögtön szólt: a bootszektor megváltozik (lehet, hogy a cache-program miatt?). Ugyanezt tapasztaltuk, amikor Norton Commanderrel állományokat másoltunk

ki merevlemezről floppyra. A floppylemezek inicializálásához pedig külön trükköket kellett alkalmazni, amolyan „jobb kezemmel megfogom a bal fülemet" típusú akciókat. Közben a kártya folyton sipákolt.

Ugyanezen a gépen megvizsgáltuk a kártyát az MS-DOS 5.00 béta-verziójával is. Egyértelműen megállapíthatjuk, hogy a Thunderbyte két eltérő verziójú tesztpéldánya nem működik ezzel a DOS-szal. Floppyról a DOS 5.00 nem tölthető be. Merevlemezről megkísérelve, a kártya már a betöltés folyamán vírust jelez, ami természetesen csak vaklárma. Amikor nagy nehezen betöltöttük az operációs rendszert, a kártya szoftvere nem érzékelte a kártyát.

A kártyát boot- és fájlvírusokkal egyaránt teszteltük. A bootvírusok közül Stoned/Marijuana vírussal fertőzött lemezt tettünk az A: meghajtóba, és megpróbáltuk a vírust betelepíteni. A kártya nagyon jól kivédte a támadást. Ránézett a floppyra, és inkább a merevlemezről töltötte be az operációs rendszert. Hasonlóképpen védte ki a (c)Brain floppyfertőző bootvírust is. Utána az Alt billentyű lenyomásával kikapcsoltuk a kártyát, és megfertőztük a gépet a Stoned/Marijuana vírussal. Amikor már a kártyával újraindítottuk a rendszert, a DOS betöltési folyamata a CONFIG.SYS fájlnál leállt (valószínűleg a TBCONFIG.COM program memóriaelhelyezési gondja miatt), a gép pedig lefagyott és csak a kártya kiiktatása illetve a vírus kiirtása után vált ismét használhatóvá. (Ki dolgozik mindig „tüzelésre kész" csavarhúzóval?)

Tesztünk során Print Screen bootvírussal is megfertőztünk egy nem rendszerlemezt. Ha erről a lemezről akartunk operációs rendszert betölteni — mint amikor valaki figyelmetlenül benne felejti a floppyt a meghajtóban —, akkor a BIOS (Phoenix) a rendszer újraindítását kérte az F1 billentyűvel, amitől le is merevedett a gép. Ogre/Disk Killer vírusra és a Ping-Pong bootvírus XT-verziójára a kártya hasonlóan viselkedett. (Ez utóbbi az AT-t kártya nélkül is lefagyasztja.)

Érdekesen viselkedett a Form bootvírussal szemben is, amivel egy szintén nem rendszerlemezt fertőztünk meg. Rendszerindításkor a kártya figyelmeztetett, hogy valaki a lemezkezelő megszakításon keresztül írni akar a merevlemez bootszektorába. Ha nem engedélyezzük a közvetlen lemezírást *(Abort (Y/N) Y)*, akkor a DOS betöltődik, de a vírus is aktív, rezidens lesz. Ezt követően a DOS betöltési folyamata a CONFIG.SYS állománynál leáll, és a rendszer lefagy (valószínűleg az ott elhelyezett TBCONFIG.COM programot nem tudja betölteni). A számítógép ilyenkor csak a főkapcsolóval vagy a reset gombbal éleszthető fel, a Ctrl-Alt-Del kombináció nem működik.

A bootvírusok beépüléséhez egyébként nem okvetlenül szükséges rendszerindításra alkalmas bootlemez. Elég megformázni egy akármi-

lyen lemezt, ugyanis a lemez betöltő rutinja, amely kiírja, hogy nem rendszerlemez, elegendő ahhoz, hogy a bootvírus ráakaszkodjon, és a második bootfolyamat során már benn is marad a tárban!

Szintén érdekes a helyzet az Invader/Plastique v5.21 fájl- és bootvírussal is, melyet fájlfertőzésből indítottunk. A kártya figyelmeztetett arra, hogy valamilyen program rezidenssé válik, mi pedig engedélyeztük. (Honnan tudja szegény felhasználó, hogy mi akarhat rezidenssé válni?! Automatikusan nyomja a Yest, különösen azután, hogy a lemezmásolás a Nortonnal állományonként két gombnyomást igényel.) Ezt követően a vírus aktivizálódott. Amikor a bootszektorba akart írni, a kártya ki is jelezte (elcsípte), de sajnos a vírus válaszunktól függetlenül (igen/nem) megfertőzte a bootszektort.

A kártyával az .EXE állományoknál az Invader vírus fertőzése megakadályozható, de a .COM programokat szó nélkül megfertőzi.

Fájlvírusok esetén a kártya igen furcsán viselkedik. Erre magyarázatot maga a program írója adott: a kártyát nem éles vírusokkal, hanem vírusszimulátorokkal tesztelték. Amint Frans Veldman sajnálkozva és tréfásan megfogalmazta: *a vírusok nem kártyakompatibilisak.*

A Thunderbyte csak néhány fájlvírus esetén hatásos. Többször produkált viszont olyan vírus-effektusokat, ami még a vírusok forráskódját alaposan ismerve is meglepetés volt. Például a magyar Polimer vírus vígan elsétált mellette. Az Eddie/Dark Avenger vírust pedig kifejezetten doppingolta a kártya: normális esetben Eddie csak egyszer épül be egy állományba, a kártya megléte esetén pedig akárhányszor.

A kártya a Victor/Iván v1.0 verzióját korrekten felfedezi, és a fertőzést megakadályozza. Hasonlóképpen viselkedik a Jerusalem/PLO vírussal és a Vienna/648-cal is. Érdekes a Vacsinával való bensőséges kapcsolata. Ha Norton Commander alól futtatjuk a vírusos programot, jelzi az INT 21 cseréjét. Ennek nem engedélyezése esetén a vírus sem fertőz. Ha rossz gombot nyomva engedélyezzük, akkor a kártya folyamatosan vonyít, a rendszer leáll és csak a főkapcsolóval kelthető életre. Ha nem a Norton Commander alól indítjuk a programot, a helyzet annyiban más, hogy az engedélyező, azaz rossz válaszra a DOS *Devide overflow* üzenettel kiakad, de legalább befogja a hangszóróját!

Az új generációs vírusok közé tartozik a 4096, amely egyszerre több verzióban is felbukkant Dél-Magyarországon. A kártya ennek a fertőzését kivédi. Nem szól semmit, és a vírus ugyan rezidenssé válik, de nem fertőz tovább. Lényeges pozitívum.

Az éles vírusokkal folyatatott nyúzópróba és kínvallatás után a kártyát kiadtuk fejlesztőinknek, hogy szokásos körülmények között teszteljék működését. Egyikük egy hétig kínlódott a kártyával, a másik hama-

rabb feladta. Véleményük összességében az volt, hogy a kártya részötleteiben jó, de nem jól kezelhető, mert minden vírusgyanús esetben a használónak kell eldöntenie, hogy engedélyezi-e az adott műveletet, vagy sem. (Aki pedig még akkor is nehezen tud dönteni, ha jól beszél angolul vagy németül, és az üzenetek pontosan vannak megfogalmazva.)

A kártya jelenlegi tudásával a gyakorlatban nehézkesen használható, nem elég megbízható és hamis biztonságérzetet kelt. Egyes vírusok működését csak részben vagy egyáltalán nem akadályozza meg, néhányét pedig még serkenti is.

Teljesen véd az alábbi vírusok ellen:

Stoned, Ping-Pong '86, Töltögető, (c)Brain, Yankee Doodle, Vienna/648, Victor, Jerusalem, 1260, 4096, Mixer, Barcelona, Perfume, Syslock, Oropax, Time/Monxla, Joker, Devil's Dance, Invader/Plastique (.EXE verzió).

Szerencsés válaszok esetén részlegesen véd az alábbiak ellen:

Form boot, Eddie, V2000, April 1.

A Thunderbyte kártyával ugyanúgy, vagy még inkább fertőznek:

Polimer, Invader/Plastique (.COM verzió).

A kártya mellett is rendszerösszeomlást, adatvesztést okoznak:

Ogre/Disk killer, Print Screen, Invader/Plastique (bootrutin), Vacsina, Alabama, W-513, Flip, Vcomm.

Eredményeinket elküldtük a fejlesztő cégnek is. Első reakciójuk dacos válasz volt a vírusok kártyakompatibilitásáról... De rövidesen megjelent az EXIT_FIX.COM rezidens program (a Thunderbyte-kártyákhoz), amellyel a rendszer lemerevedéseit akarták megakadályozni. Mi pedig megkaptuk a Thunderbyte v2.00 *message compilert*, s cserébe a cég kikérte véleményünket. Itt a rendszerüzeneteket már a szoftver tartalmazza és a disztribútorok fordítják be a szoftverbe az adott ország nyelvén. De a kártyán sajnos továbbra is átmennek az eddigi „kártyajáró" vírusok.

Virus Guardian

1990 novemberétől több forgalmazónál megjelent a *Virus Guardian* vírusvédelmi kártya. Izraeli fejlesztés, melyben ők nem láttak fantáziát, ezért addigi eredményeiket eladták Tajvanba.

Első ránézésre roppant érdekesnek tűnt. Szimpatikus benne, hogy a kártyán akkumulátor van, amivel bizonyos alapinformációkat a gép kikapcsolása után is megőrizhet. Azután feltűnt a kártyán egy hosszú, speciális IC is... amiről azonban kiderült, hogy a tajvani gyártó egy fém-

lappal leragasztotta a kártya BIOS-át és a kártyán lévő CMOS-t, mintegy véglegesítve a kártya szoftverét is. Ezzel nem tudták ugyan megakadályozni a ROM tartalmának kimásolását és visszafejtését, csak a vevő zsebéből lopják ki vele a pénzt, mert aki át akar térni egy fejlettebb változatra, annak újból meg kell vásárolnia az egész kártyát nem csupán az új BIOS-t és a szoftvert, amire valójában szüksége lenne.

A teszteléshez a kártyát behelyeztük egy IBM kompatibilis gépbe. (AT 286-os, 40 Mbájtos merevlemez, EGA kártya, 5,25" és 3,5" floppymeghajtók.) Rendszerindítás után a kártya mindjárt be is jelentkezett, teljes képernyővel.

Először a kártya jelszavát kellett megadni. (Mi a kártya sorozatszámát használtuk erre.) A jelszót minden újabb rendszerindításnál be kell gépelni, mert megköveteli. Ha nem adunk meg jelszót, a kártya akkor is bejelentkezik, de az Enter billentyűvel kell továbblépkedni a kérdéseken.

Nem mondhatjuk el, hogy a Virus Guardian szerényen meghúzódva, csendben és láthatatlanul dolgozik. Függetlenül attól, hogy szükség van-e a kijelzésre, állandóan visít és megpróbál látványosan működni.

A kártyát egy olyan „új" számítógépbe tettük be, melyet szoftverekkel még ne, töltöttünk fel. A konfiguráció beállítása után a kártya megjegyezte, hogy a DOS mekkora szabad memóriával rendelkezik, és azt elraktározta a CMOS-ába. Amikor az AUTOEXEC.BAT állományban elhelyeztük a DOS rezidens APPEND parancsát, a kártya minden rendszerindításkor visított. Ha kivettük az APPEND parancsot, a visítás megszűnt. Vajon miért tartja vírusnak a DOS parancsot? Haragudott a GRAFTABL és a PRINT rezidens DOS-utasításokra is.

A kártyához adott ködös dokumentációban nem találtunk semmilyen utalást arra, hogy ebben az esetben hogyan kell eljárni. Végül rájöttünk a megoldásra: ki kell venni az akkumulátort, pihentetni egy kicsit, s utána újból betehetjük a gépbe. (Az akkumulátor tárolta a DOS által átadott szabad memóriát.) A rendszer újból elindítva belépéskor először ismét a sorozatszámát kéri. (Még jó, hogy ráragasztottuk a gépre és a kártyára is!)

A gépnek szoftverekkel való feltöltése során létrehoztunk néhány alkönyvtárat. Természtesen nem minden állomány került oda, ahová szerettük volna. Ezeket az állományokat a Norton Commander v3.00-val átmásoltuk a kívánt alkönyvtárba, a régieket pedig töröltük, vagyis nem a move parancsot használtuk. A VG (Virus Guardian) kártya azonban nem hagyja olyan egyszerűen törölni az általa figyelt COM, EXE, SYS stb. állományokat, hanem minden törlésre a szokásos visítás kíséretében kétszer rákérdez. 60 állomány = 120 visítás + 120 Y (yes) billentyű lenyomása...

Amikor végre használni akartuk a gépet, a kártya egyes szoftvereknél érthetetlen módon lemerevítette azt. Például a Carmel szoftver Turbo Antivirus TNVIRUS.EXE programjánál is. Ugyanennek a programcsomagnak a TSAFE rezidens vírusvédelmi része némi visítások közepette ugyan, de működésbe lépett. Ez egy kicsit meglepett minket. Ismét megpróbáltuk elindítani a TNVIRUS programot. Csoda történt! A program minden további nélkül elindult. Mintha a rezidens programok figyelésével valami gond lenne ezen a kártyán. Ezt követően minden olyan program indítása előtt, amely előzőleg lefagyott, lefuttatuk a TSAFE programot, és azok kitűnően működtek!

A vírusok vizsgálatakor a VG kártya az általunk elindított összes bootvírust kivédte, néhány apró megjegyzéssel. A Töltögető bootvírus eljutott a pusztító rutin által kiírt szövegig, de a merevlemez tartalmában nem okozott kárt. A Den-Zuk bootvírus eljutott a *Non system disk* rendszerüzenet kiírásáig, és a rendszer lefagyott, tehát a vírus átjutott a védelmi rendszeren.

Ha a kártya bootvírust észlel, figyelmezteti a felhasználót, és tiszta rendszerlemezt kér a DOS betöltéséhez. Eddig rendben is van, de a kártya nem minden esetben ad hidegindítást, hanem folytatja a rendszer betöltését a behelyezett másik lemezről, de minden ellenőrzés nélkül! Márpedig, ahol *egy* vírus van, ott lehet több is. A kártya szóvá teszi ugyan, hogy az operációs rendszert tartalmazó lemez fertőzött, s hogy valaki a megadott számú fizikai merevlemezre akar írni... de akkor más késő, a vírus odaért.

A fájlvírusok közül a VG kártyát mindjárt az egyik legveszélyesebbel, az Invaderrel teszteltük, amely ellen a Thunderbyte kártya sem véd kielégítően. A kártya észreveszi, hogy valaki rezidenssé akar válni és figyelmeztet erre, ám eközben az Invader vírus már bele is írt a bootszektorba. Ezt a fertőzést csak akkor vettük észre, amikor egy lefagyáskor ismét rendszert akartunk indítani a merevlemezről. A kártya, érzékelvén a bootvírust, nem engedte a rendszerindítást. (Nem szól idejében, de véd a fertőzés *után*. És itt jöhet a csavarhúzó, a szerelés, majd a vírusirtás...)

A VG kártya fájlvírusok elleni védelmi elvét a következőképpen fogalmazhatjuk meg: „engedem, hogy a vírus rezidenssé, aktívvá váljon, de nem engedem, hogy tovább fertőzzön". Meglepő volt, hogy a kártya a szokásos fájlvírusok némelyikével szemben is „érdekesen" viselkedett. A fejlesztők arra sem gondoltak, hogy a grafikus szoftverek mellett is lehetnek vírusok, és a kártya üzenetei teljesen tönkreteszik, absztakt mintákkal dekorálják a grafikus képernyőt. (Talán takarékosságból Hercules monokróm monitoron fejlesztették?)

Az 1701/Cascade vírussal fertőzött program (Norton Commander alóli) elindítása után a kártya egyszerűen lefagyott. A 1704-es vírus lefuttatása után pedig a kurzor folyamatosan szaladgált a képernyőn, mintha végtelen ciklusba esett volna. Egy klasszikus baci kiakasztotta, amit igazán nem vártunk volna tőle. A 648/Vienna elindítása után mindig a *Write protect disk* üzenetet kapjuk meg.

Teljesen véd a kártya a következő vírusokkal szemben: Stoned, Ogre/Disk killer, Print Screen, Ping-Pong '86, Form boot, (c)Brain.

Részben véd a Töltögető ellen.

Nem véd, illetve rendszerösszeomlást okoz az alábbiak esetén: Invader/Plastique boot rutin, Den Zuk, 1701/1704 Cascade.

A kártyához adott szoftverek nincseek összhangban magával a kártyával. A partíciós tábla elmentése még zavartalan, visszatöltéskor azonban a kártya visít, jelezve, hogy alacsonyszintű műveletet hajtunk végre. (Honnan tudja a felhasználó, hogy egy-egy technológiai folyamat milyen szinten zajlik le a DOS-hierarchiában?) Ha ezt automatikusan lelövi, ahogy a bacikat kell, akkor a merevlemez ledöglik, és örökké a *Disk error* hibaüzenetben „gyönyörködhetünk". Alapvető, adatbiztonságot veszélyeztető hiba.

Top Virus Guard

Végeredményben a külföldi kártyák tesztelésekor szerzett kedvezőtlen tapasztalatok ösztönöztek bennünket olyan saját vírusvédelmi kártya megtervezésére és kifejlesztésére, amely megfelel a fejezet bevezetőjében megfogalmazott követelményeknek. Felfogásunk szerint ugyanis a jövő vírusvédelmi eszköze olyan hardvereszköz lehet, amelyet szoftver támogat. Ki kellett azonban próbálni, hogy ebből a jelenlegi technikai háttérrel mennyit lehet megvalósítani. A Safe Kft-nél elkészítettük a Top Virus Guard (TVG) kártyát, melynek bétatesztes változatát 1991 májusában, a budapesti Ifabón mutattuk be.

E vírusvédelmi kártya lényegesen nagyobb területet fog át, mint az egyszerű vírusvédelem, és tudása az általunk ismert és tesztelt vírusvédelmi kártyákét felülmúlja. Úgy terveztük meg, hogy a védelem továbbfejlesztésekor ne kelljen a kártyákat mindig lecserélni, hanem elegendő legyen a programot korszerűsíteni, illetve a meglévő egységhez kiegészítő „kártya-oldalkocsikat" kapcsolni.

Jól látható tendencia továbbá a számítástechnikai rendszerekben tárolt adatok és programok sérülés és lopás elleni biztosítása. (Nevezzük leegyszerűsítve vírusbiztosításnak.) A fejlesztés során erre is gondolni

kellett, így a TVG kiegészítő szoftverrel és elektronikával egyben adat- és vagyonvédelmi funkciókat is ellát.

A TVG vírusvédelme két fő területre oszlik:

1. Ismert vírusok elleni, konkrét vírusismereten alapuló védekezés.
2. Ismeretlen vírusok elleni, általános eljárásokon és algoritmusokon alapuló védelem, amely a vírusfunkciót felismeri és leblokkolja.

Az ismert vírusok elleni védekezés az eddigi szoftveres eredmények felhasználására épülhet. A kártya szoftvere a BIOS után, de még az operációs rendszer betöltése előtt aktivizálódik, az operációs rendszertől teljesen független állománykezeléssel. Itt még egy problémába ütközik a programfejlesztő, ami miatt a szoftverírók általában elkerülik az ilyen megoldásokat: igen sokféle merevlemez és meghajtó van forgalomban, és azoknak a védelemhez felhasználható, nem publikált vezérlőparancsai jelentősen eltérnek. Hasonló gondokat okoznak a speciális alaplapok is.

A TVG kártya az operációs rendszer betöltése előtt elvégzi a bootvírusok ellenőrzését. Ismert bootvírusok jelenléte esetén kiírja a vírus nevét, és igény szerint ki is irtja a floppyról. Ha ismeretlen bootvírusokkal találkozik, akkor az általános védelmi rutin aktivizálódik, amely tájékoztat a vírus jelenlétéről, és nem engedi azt aktivizálódni.

Nagyon nehéz általános célú vírusvédelmi szoftvert írni. Ismert vírussal találkozva rendszerint az általános célú vírusvédelmi kártya is csak a felhasználó számára semmitmondó üzeneteket tud kiírni, például *Interrupt csere. Engedélyezi? Igen / Nem... A program a fizikai #1 lemezcilinder 0. fej 1. szektorába ír. Engedélyezi? (Igen / Nem).* A TVG kártya ilyen hibaüzeneteket csak akkor küld a felhasználónak, ha már a mintegy 200–400 vírusra és azok változataira ellenőrizte a programot, és mégis valami rendellenességet tapasztalt.

Ha valamelyik bootvírus mégis kijátszaná a kártya hardver- és szoftvervédelmét, és megfertőzné a merevlemez bootszektorát, a vírusvédelmi rendszer azt is észleli, és az eredeti bootszektornak és partíciós táblának az akkumulátorral táplált memóriában őrzött adatai alapján helyre tudja állítani a sértetlen állapotot.

A fájlvírusok elleni védelmi rendszer a következőképpen működik. Minden indítás előtt ellenőrzi az összes programot, hogy talál-e ismert vírust. Ez a művelet olyan gyorsan történik, hogy a használó észre sem veszi. Amennyiben ismert vírust talál, azt rögtön el is távolítja és a műveletet automatikusan dokumentálja. A rendszert fel lehet készíteni a vírusátírási algoritmusokra is. Például létezik olyan eljárás, amely a

hazánkból kiindult, NOP-beszúrásos vírusátírási technika ellen véd. Fontos az analízis gyorsasága és megbízhatósága.

Ezt követően aktivizálódik az általános célú vírusvédelmi rendszer, amely megakadályozza az ismeretlen vírusok terjedését, fertőzését és így az adatvesztést. Minden floppyról és logikai egységről végzett másolást ellenőriz, így gátolja meg, hogy a winchesterre vírus kerülhessen. Az ötlet McAfee VCOPY programjából született: ellenőrizni kell az operációs rendszert, ha már a Microsoft nem gondolt rá. Ha valamilyen vírusgyanús műveletnél a felhasználónak kell döntenie, arról is dokumentum készül. Mindezeken túl a TVG kártya letilthatja a floppyegységeket (szükség esetén csak a rendszergazda engedélyezheti új programok felvitelét a merevlemezre), és biztosítja, hogy csak érvényes jelszóval lehessen a gépet használni. Ez természetesen nem kötelező, csupán választható lehetőség, de az illetéktelen hozzáférés ellen kellő biztosítékot nyújt. Kiegészítő modullal a hozzáférés engedélyezésének eszköze akár intelligens memóriakártya, akár az USA hadseregében alkalmazott ujjlenyomat- vagy recehártyakép-vizsgáló berendezés is lehet.

Ahhoz, hogy a védelmet széles körben alkalmazni lehessen és szervesen kapcsolódhasson más programokhoz, minden programozó számára meg kell adni a jól definiált programozói felületet. Ebből a célból a jogos használóknak a TVG kártya lehetőséget ad a funkcióhívásokra, és arra, hogy saját szoftverükben is kihasználhassák a kártya és a hozzá tartozó programok víruskereső/vírusölő funkcióját. Ez különösen a streamer kimentő/visszamentő szoftvereknél célszerű.

Mélyen beavatkozva a PC lelkivilágába, meg kellett oldani az operációs rendszer betöltését kívánság szerint a B: floppyegységről is. Ennek akkor van jelentősége, ha különböző típusú floppynk van és az installálandó rendszer csak bootolás után helyezhető üzembe. (Például jól jött most, amikor az MS-DOS 5.0 bétatesztje csak 720-as 3,5"-os floppyn volt hozzáférhető, ami általában minden gépben, így nálunk is a B: jelű lemezegység.) Érdemes volt megoldani a merevlemezek *low level*, illetve *high level* formázás elleni védelmét is, mert ezáltal a formázással büntető másolásvédelmek és a trójai programok nem tudják tönkretenni az állományokat.

A vírusvédelmi rendszert felkészítettük olyan vírusok ellen is, amelyek a Ctrl–ALt–Del típusú rendszerindítást átvészelik. A kártya a CMOS-szal (AT/286/386/486) rendelkező számítógépek védelmét is megoldja. Ez a funkció megakadályozza a CMOS-ba író programok és másolásvédelmek (például a Lotus 1-2-3 R3.1 magyar változat) futását a védelem miatt, de cserébe védelmet nyújt minden olyan vírus, trójai

program vagy büntető másolásvédelem ellen, amely a merevlemezt típusának menet közbeni átírásával teszi tönkre.

Számba vettük a TVG hardver–szoftver kombinációjának előnyeit az eddig csak programozástechnikai eszközöket alkalmazó megoldásokkal szemben. Mint könyvünk korábbi fejezeteiben kifejtettük: az egyes víruskereső és vírusölő programokat bizonyos rendszerességgel futtatni kell. Ez lehetővé teszi a vírusok megtalálását és kiirtását, de nem oldja meg a vírusmegelőzést. A rendszerbe jutott bootvírusok esetében is csak vírusölésről lehet szó, mert a bootvírus az operációs rendszer betöltésének folyamatába épül be, s hagyományos technológiával azt megelőzően semmilyen szoftver nem indulhat el. Ha pedig a betöltést módosítjuk, könnyen adatvesztés történik, még a merevlemez is tönkremehet, s legjobb esetben inkompatibilitással kell számolni. Hasonló a helyzet, ha a védelemre tárrezidens szoftvert alkalmazunk, s még a memória mérete is jelentősen csökken.

A TVG koncepciójában installálás után a rendszer csak kis mértékben függ az embertől. Nem kell rendszeres vírusellenőrzést tartani, mert ez állandóan folyik. Nem kell semmilyen vírusölő programot lefuttatni, mert a kártya minden programot még a számítógépbe jutása előtt ellenőriz, valós idejű módban és észrevétlenül. A bootvírusok detektálása és a számítógép vírusvédelme tehát még azelőtt megtörténik, hogy a vírusok a rendszerbe bejutnának.

A teljeskörű naplózási rendszerrel pontosan megállapíthatjuk, hogy melyik programunk volt vírushordozó. A szokásos szoftveres megoldásokkal már csak a fertőzött állományok mutathatók ki, de az nem, hogy melyik program és mikor hozta be a vírust. Ez részletkérdésnek tűnik, pedig igen fontos kideríteni a fertőzés forrását, s hogy az eset véletlen volt-e vagy szabotázs. A naplózás biztonságtechnikai szempontból és az adatvédelmi biztonság szempontjából egyaránt kulcsfontosságú. Ez a rész ugyanúgy működik, mint a pénztárgépek „spiclidoboza": adott ideig mindent tárol és adatai a használó által nem módosíthatók, csak leolvashatók.

A TVG pontosan úgy működik, mint egy memóriarezidens vírusvédelmi rendszer, azzal a különbséggel, hogy egyetlen bájtot sem foglal el a memóriából. A kártya szoftverének futtatásáról magára a kártyára felszerelt, 256 kbájtos RAM gondoskodik. A TVG rendszer ismert vírusaira, azok azonosítási és ölési eljárásaira, valamint a kártya általános vírusfigyelő rendszerére — a vírusok fejlődésének figyelembevételével — folyamatos bővítést adható. A szoftververzió cseréje nem igényli a kártya cseréjét, és a BIOS esetleges változtatása is egyszerűen megoldható.

A kártya extra szolgáltatása a merevlemez valódi, fizikai írásvédel-

me. Általa a problémás programok tesztje, vagy a csak olvasható adatbázisok védelme éppoly egyszerű, mint amikor a 3,5"-es floppy fizikai írásvédelmét egy műanyag pecek eltolásával ki- vagy bekapcsoljuk. Minden szoftveres írásvédelmi rendszer kijátszható valamilyen úton. Sportoltunk vele mi is eleget. De ugyanezt teszik a vírusgyártó kisiparosok is. A merevlemez valódi fizikai írásvédelme viszont csak az írásvezeték megszakításával oldható meg. A TVG kártya — ha már úgyis ott van —, él ezzel a lehetőséggel.

A kártya vagyonvédelmi rendszere a Dataplan Rt. által kifejlesztett HISEC rendszerhez kapcsolódik, egy kiegészítő panelen keresztül. (Külön is megvásárolható a vagyonvédelmi modul, amely a számítógép kinyitását, ellopását és illetéktelen használatát akadályozza meg.) A vagyonvédelmi rendszer más jellegű riasztó rendszerekhez (tűzvédelmi, magnós stb.) is kapcsolódhat.

A kártya használata nem igényel vírusismeretet és szaktudást. A kártya érzékeli, hogy mikor kapcsolták ki a gépet, s ezután bekapcsoláskor automatikusan teljes vírusellenőrzést végez. A TVG kártya azt is érzékeli, ha kivették a gépből. Ismételt visszahelyezés esetén pedig ezt a tényt dokumentálja is. Mivel maga is víruskereső funkciókat lát el, a kártyához külön víruskereső nem tartozik.

Amennyiben a TVG kártyák elterjednek, azok naplóállományainak elektronikus összesítésével rendszeresen kaphatunk országos statisztikát a vírushelyzetről, természetesen a használókkal együttműködve. A különösen fontos adatok védelmére (pl. Nemzetvédelmi Minisztérium, Belügyminisztérium) kiegészítésként valós idejű online adattitkosító modul is csatlakoztatható.

E könyv írásakor még tart a TVG kártya felületszerelt végleges változatának áramköri tervezése. Az áramköri lap gyártására valószínűleg az USA-ban kerül sor, az ottani MIL (az átlagosnál szigorúbb, katonai) szabványok szerint.

A VÍRUSOK ÖSSZEFOGLALÓ TÁBLÁZATA

A McAfee Associates 1991. júliusi listája alapján

Név A McAfee-féle egységes referencianevek.

( ) Gömbölyű zárójelben az azonos eljárással felismerhető vírusváltozatok száma. (A nyugat-európai és amerikai szakirodalom alapján.)

[ ] Szögletes zárójelben a McAfee által javasolt rövidített kód az egységes számítógépes nyilvántartáshoz és a vírusirtó programokban való azonosításhoz.

A vírus fertőzési helyére utaló jelölés:

\+ Igen

\- Nem

A fertőzött programok méretnövekedése:

n/a Nincs megadott hossz vagy nem jellemző a vírusra.

— Nem értelmezhető (pl. felülíráskor).

A károkozás módjának betűjelzései:

B Károsítja vagy felülírja a bootrekordot.

D Adatállományokat károsít vagy felülír.

F A merevlemezt, a floppyt vagy azok egyes részeit formázza.

L Közvetlenül vagy közvetve rossz állománykapcsolatokat okoz.

O Az operációs rendszer működését befolyásolja.

P Program- és overlay-állományokat károsít vagy felülír.

Egyéb jelölések:

* A rejtett, „stealth" (lopakodó) technikát alkalmazó vírusok az utóbbi időben jelentek meg. Nem a hagyományos módon válnak rezidenssé, illetve a DOS számára az állomány eredeti állapotát színlelik.

** Névváltozás.

*** Eredeti magyar vírus. Nevüket a külföldi szakirodalom nem mindig írja pontosan. (Például a Turbo Kukac említésekor néha szinte azonosítani sem lehet, hogy melyik vírusról van szó.)

**** Miközben a Crash vírus kódja lefut, nagy lármát csapva teljes rendszerösszeomlást okoz, emiatt egyéb tulajdonságairól nem lehet képet alkotni. Új víruscsalád első tagjának látszik.

MEGNEVEZÉS			A FERTŐZÉS HELYE											
Név	Változatok száma	Rövidített névkód	Hagyományosan rejtett	A víruskódot „titkosítja"	Rezidens része marad a tárban	COMMAND.COM	.COM állományok	.EXE állományok	Overlay-állományok	A floppy bootrekordja	A merevlemez bootszektora	A merevlemez partíciós táblája	A fertőzött program méretnövekedése	A KÁROKOZÁS MÓDJA
AGI-Plan		[AGI]	–	–	–	+	+	–	–	–	–	–	1536	O P L
AIDS	(4)	[Aids]	–	–	–	–	+	–	–	–	–	–	—	Felülír
AirCop	(3)	[AirCop]	–	–	+	–	–	–	–	+	–	–	n/a	B O
Alabama	(3)	[Alabama]	–	–	+	–	–	+	–	–	–	–	1560	O P L
Alameda	(2)	[Alameda]	–	–	+	–	–	–	–	+	–	–	n/a	B
Amstrad	(5)	[Amst]	–	–	–	–	+	–	–	–	–	–	847	P
Anarkia	(2)	[Ana]	–	–	+	–	+	+	–	–	–	–	1813	O P D
Anthrax Boot	(2)	[Atx]	–	–	+	–	–	–	–	–	–	+	n/a	O P D
Anthrax File	(4)	[Atx]	–	–	+	+	+	+	–	–	–	–	1206	O P D
Arab Virus		[Ar]	–	–	+	–	+	–	–	–	–	–	834	O P
Armagedon	(3)	[Arma]	–	–	+	+	+	–	–	–	–	–	1079	O P
Ashar		[Brain]	–	–	+	–	–	–	–	+	–	–	n/a	B
Australian		[Aust]	–	–	+	+	+	+	+	–	–	–	1433	O P L D
Austria	(3)	[Austria]	–	–	–	+	+	+	–	–	–	–	—	Felülír
Azusa		[Azusa]	–	–	+	–	–	–	–	+	–	+	n/a	D O B L
Bad Boy	(2)	[BB]	–	–	+	+	+	–	–	–	–	–	1000	O P D
BadGuy		[BG]	–	–	–	+	+	–	–	–	–	–	265	O L
Bandit		[Ban]	–	–	+	+	+	+	+	–	–	–	988	O D
BeBe		[BeBe]	–	–	–	+	+	–	–	–	–	–	1004	O P D
Beeper	(2)	[Beep]	–	–	+	–	+	–	–	–	–	–	482	O P D
Best Wish		[BWish]	–	–	–	+	+	+	+	–	–	–	1024	O P D
Black Monday	(2)	[BMON]	–	–	+	+	+	+	+	–	–	–	1055	L O P D
Bljec	(8)	[Blj]	–	–	–	+	+	–	–	–	–	–	369	O,P
Blood-2		[B-2]	–	–	–	–	+	–	–	–	–	–	427	O P D
Bloody!		[Bloody]	–	+	+	–	–	–	–	+	–	+	n/a	B O

MEGNEVEZÉS			A FERTŐZÉS HELYE											
Név	Változatok száma	Rövidített névkód	Hagyományosan rejtett*	A víruskódot „titkosítja"*	Rezidens része marad a tárban	COMMAND.COM	.COM állományok	.EXE állományok	Overlay-állományok	A floppy bootrekordja	A merevlemez bootszektora	A merevlemez partíciós táblája	A fertőzött program méretnövekedése	A KÁROKOZÁS MÓDJA
Brain	(3)	[Brain]	–	–	+	–	–	–	–	+	–	–	n/a	B
Brain Slayer		[Slay]	–	–	+	–	+	+	+	–	–	–	5120	O P L D
Cancer		[Cn]	–	–	–	–	+	–	–	–	–	–	1480	O P D
Carioca	(2)	[Carioca]	–	–	+	–	+	–	–	–	–	–	951	O P
Cascade-B	(9)	[170x]	–	+	+	–	+	–	–	–	–	–	1704	O P
Casino		[Casino]	–	–	+	+	+	–	–	–	–	–	n/a	O P L
Casper		[Casper]	–	+	–	+	+	–	–	–	–	–	1200	L O P D
Chaos		[Chaos]	–	–	+	–	–	–	–	+	+	–	n/a	B O D F
Christmas-J		[C-J]	–	–	+	+	+	+	–	–	–	–	600	O P
ChristmasViolator		[CVio]	–	–	–	?	+	–	–	–	–	–	n/a	O P D
Crash****		[Crash]	–	–	–	–	–	–	–	–	–	–	n/a,	
Curse Boot		[Curse]	–	–	+	–	–	–	–	+	+	–	n/a	B O
Dark Avenger	(4)	[DAv]	–	–	+	+	+	+	+	–	–	–	1800	O P L
Darth Vader	(6)	[Darth]	–	–	+	+	+	–	–	–	–	–	Változó	O L
Datacrime	(2)	[Crime]	–	+	–	–	+	–	–	–	–	–	1280	P F
Datacrime II	(2)	[Crime-2]	–	+	–	–	+	+	–	–	–	–	1514	P F
Datacrime II-B		[Crime-2]	–	+	–	+	+	+	–	–	–	–	1917	P F
Datacrime-B		[Crime-B]	–	+	–	–	+	–	–	–	–	–	1168	P F
DataLock		[Data]	–	–	+	–	–	+	–	–	–	–	920	O P
Dbase		[Dbase]	–	–	+	–	+	–	–	–	–	–	1864	D O P
Den Zuk	(3)	[Zuk]	–	–	+	–	–	–	–	+	–	–	n/a	O B
Destructor		[Dest]	–	–	+	+	+	+	+	–	–	–	1150	O P
Devil's Dance		[Dance]	–	–	+	–	+	–	–	–	–	–	941	D O P L
Dir-Vir		[DVir]	+	–	+	+	+	–	–	–	–	–	691	O P D

MEGNEVEZÉS			A FERTŐZÉS HELYE											
Név	Változatok száma	Rövidített névkód	Hagyományosan rejtett	A víruskódot „titkosítja"	Rezidens része marad a tárban	COMMAND.COM	.COM állományok	.EXE állományok	Overlay-állományok	A floppy bootrekordja	A merevlemez bootszektora	A merevlemez particiós táblája	A fertőzött program méretnövekedése	A KÁROKOZÁS MÓDJA
Disk Killer	(4)	[Killer]	–	–	+	–	–	–	–	+	+	–	n/a	B O P D F
Do-Nothing		[Nothing]	–	–	+	–	+	–	–	–	–	–	608	P
Doom2		[Dm2]	–	–	+	–	+	+	–	–	–	–	2504	O P D L
Dot Killer		[Dot]	–	–	+	+	+	–	–	–	–	–	944	O P
EDV	(2)	[EDV]	+	–	+	–	–	–	–	+	+	+	n/a	B O
Empire	(3)	[Emp]	+	+	+	–	–	–	–	+	+	–	n/a	O P
Enigma		[Enigma]	–	+	+	–	–	+	+	–	–	–	1755	O P
Exterminator		[Ext]	–	–	+	+	+	+	–	–	–	–	451	O L D
Father Christmas		[FC]	–	–	–	+	+	–	–	–	–	–	1881	O P
Fellowship	(3)	[Fellow]	–	–	+	–	–	+	–	–	–	–	1022	O P D L
Fingers		[Fing]	–	–	+	+	+	+	+	–	–	–	1322	O P D
Fish-6	(2)	[Fish]	+	+	+	+	+	+	+	–	–	–	3584	O P L
Flash		[Flash]	–	–	+	+	+	+	–	–	–	–	688	O P D L
Flip	(4)	[Flip]	–	+	+	+	+	+	+	–	–	–	2343	O P D L
Form	(2)	[Form]	–	–	+	–	–	–	–	+	+	–	n/a	B O D
Frere Jacques		[Frere]	–	–	+	–	+	+	+	–	–	–	1811	O P
Friday 13th COM		[Fri13]	–	–	–	–	+	–	–	–	–	–	512	P
Frogs		[Frogs]	–	–	+	+	+	–	–	–	–	–	1500	O P
Fu Manchu	(4)	[Fu]	–	–	+	–	+	+	+	–	–	–	2086	O P
Ghost Boot		[Ghost]	–	–	+	–	–	–	–	+	+	–	n/a	B O
Ghost COM		[Ghost]	–	–	–	–	+	–	–	–	–	–	2351	B P
Guppy		[Guppy]	–	–	+	+	+	–	–	–	–	–	152	O P
Goblin		[Gobl]	–	–	+	+	+	–	–	–	–	–	1951	O P L
Greemlin		[Gree]	+	–	+	+	+	+	+	–	–	–	1146	O P L D

MEGNEVEZÉS			A FERTŐZÉS HELYE											
Név	Változatok száma	Rövidített névkód	Hagyományosan rejtett	A víruskódot „titkosítja"	Rezidens része marad a tárban	COMMAND.COM	.COM állományok	.EXE állományok	Overlay-állományok	A floppy bootrekordja	A merevlemez bootszektora	A merevlemez partíciós táblája	A fertőzött program méretnövekedése	A KÁROKOZÁS MÓDJA
Growing Block		[Grb]	–	–	+	+	+	+	+	–	–	–	1446	O P L D
Happy Day		[Happy]	–	–	–	+	+	–	–	–	–	–	453	O P
Happy New Year		[HNew]	–	–	+	+	+	+	+	–	–	–	1865	O P
Holocaust		[Holo]	+	–	+	+	+	–	–	–	–	–	3784	O P L D
Horse	(7)	[Hrs]	–	–	+	+	+	+	+	–	–	–	1154	O P L
Horse Boot		[H-B]	–	–	+	+	+	–	–	+	+	–	n/a	B P
Hybrid		[Hybrid]	–	–	–	+	+	–	–	–	–	–	1306	O P L
Hymn		[Hymn]	–	–	+	+	+	+	+	–	–	–	642	O P D
Hymn-2		[H-2]	–	–	+	+	+	+	+	–	–	–	1962	O P L
Icelandic	(4)	[Ice]	–	–	+	–	–	+	–	–	–	–	642	O P
Icelandic II		[Ice-2]	–	–	+	–	–	+	–	–	–	–	661	O P
Icelandic-3		[Ice-3]	–	–	+	–	–	+	–	–	–	–	853	O P
IKV528		[I528]	–	–	–	+	+	–	–	–	–	–	528	O P
Incom		[Inc]	–	–	–	–	+	–	–	–	–	–	648	O P
Invader	(4)	[Invader]	–	+	+	–	+	+	+	+	+	–	4096	B L O P D
Iraqi Warrior		[Iwar]	–	–	–	+	+	–	–	–	–	–	777	O P L D
ItaVir		[Ita]	–	–	–	–	–	+	–	–	–	–	3880	O P L B
Jeff		[Jeff]	–	–	–	+	+	–	–	–	–	–	828	O P D F
Jerusalem	(41)	[Jeru]	–	–	+	–	+	+	+	–	–	–	1808	O P
Jerusalem-B		[Jeru]	–	–	+	–	+	+	+	–	–	–	1808	O P
JoJo	(3)	[JoJo]	–	–	+	–	+	–	–	–	–	–	1701	O P
Joker		[Joke]	–	–	+	+	+	–	–	–	–	–	n/a	O P
Joshi	(4)	[Joshi]	+	–	+	–	–	–	–	+	+	+	n/a	B O D
July 13th		[J13]	–	+	–	–	–	+	–	–	–	–	1201	O P D L

MEGNEVEZÉS			A FERTŐZÉS HELYE											
Név	Változatok száma	Rövidített névkód	Hagyományosan rejtett*	A víruskódot „titkosítja"*	Rezidens része marad a tárban	COMMAND.COM	.COM állományok	.EXE állományok	Overlay-állományok	A floppy bootrekordja	A merevlemez bootszektora	A merevlemez partíciós táblája	A fertőzött program méretnövekedése	A KÁROKOZÁS MÓDJA
June 16th		[June16]	–	–	–	+	+	–	–	–	–	–	1726	F O P L
Justice		[Just]	–	–	+	+	+	–	–	–	–	–	1242	O P
Kennedy	(3)	[Kennedy]	–	–	+	–	+	–	–	–	–	–	308	O P
Keme	(5)	[Keme]	–	–	+	+	+	–	–	–	–	–	Változó	O L P D
Keypress	(3)	[Key]	–	–	+	+	+	+	–	–	–	–	1232	O P D
Korea	(4)	[Korea]	–	–	–	–	–	–	–	+	+	–	n/a	B O
Kukac***			–	–	–	+	+	–	–	–	–	–	448	O
Kukac/Turbo		[Kuka]	–	–	+	+	+	–	–	–	–	–	—	Felülír
Label		[Label]	–	–	+	+	+	–	–	–	–	–	—	Felülír
Lazy		[Lazy]	–	–	+	+	+	–	–	–	–	–	720	O P
Leapfrog Virus		[Leap]	–	–	+	?	+	–	–	–	–	–	516	O P D
Leech		[Leech]	+	+	+	+	+	–	–	–	–	–	934	O P L D
Lehigh		[Lehigh]	–	–	+	+	–	–	–	–	–	–	—	Felülír P F
Leprosy	(5)	[Lep]	–	–	+	+	+	+	+	–	–	–	—	Felülír
Leprosy-B		[Lepb]	–	–	–	+	+	+	–	–	–	–	—	Felülír
Liberty	(2)	[Liberty]	–	–	+	+	+	+	+	–	–	–	2862	O P
Lisbon	(2)	[Lisb]	–	–	–	–	+	–	–	–	–	–	648	P
Little Pieces		[LP]	–	–	+	–	+	+	–	–	–	–	1374	O P
Loa Duong		[Loa]	–	–	+	–	–	–	–	+	+	+	n/a	B O P L
Love Child	(3)	[LC]	–	–	+	+	+	–	–	–	–	–	488	O D
Lozinsky		[Loz]	–	–	–	+	+	–	–	–	–	–	1023	O P D
Lucifer		[Luc]	+	–	+	+	+	+	+	–	–	–	1086	O P D L
Mardi Bros.	(3)	[Mardi]	–	–	+	–	–	–	–	+	+	–	n/a	B O

MEGNEVEZÉS			A FERTŐZÉS HELYE											
Név	Változatok száma	Rövidített névkód	Hagyományosan rejtett*	A víruskódot „titkosítja"*	Rezidens része marad a tárban	COMMAND.COM	.COM állományok	.EXE állományok	Overlay-állományok	A floppy bootrekordja	A merevlemez bootszektora	A merevlemez partíciós táblája	A fertőzött program méretnövekedése	A KÁROKOZÁS MÓDJA
MGTU Virus	(2)	[MGTU]	–	–	–	+	+	–	–	–	–	–	273	O P D
Michelangelo		[Michel]	–	–	+	–	–	–	–	+	+	+	n/a	O P L
Microbes		[Micro]	–	–	+	–	–	–	–	+	+	–	n/a	B O D
Mir		[Mir]	–	–	+	+	+	+	+	–	–	–	1745	O P L
Mirror	(2)	[Mirror]	–	–	+	–	–	+	–	–	–	–	928	O P
Mix1		[Ice]	–	–	+	–	–	+	–	–	–	–	1618	O P
Mix2		[MX2]	–	–	+	+	+	+	+	–	–	–	2280	O P
Monxla		[Monxla]	–	–	–	+	+	–	–	–	–	–	939	O P
Monxla-B		[MonB]	–	–	–	+	+	–	–	–	–	–	535	O P L
Murphy		[Murphy]	–	–	+	+	+	+	+	–	–	–	1277	O P
Music Bug	(3)	[MBug]	–	–	+	–	–	–	–	+	+	–	n/a	B O
Necrophilia		[Nec]	–	–	+	+	+	–	–	–	–	–	Változó	O P L D
New Jerusalem		[Jeru]	–	–	+	–	+	+	+	–	–	–	1808	O P
Nina		[Nina]	–	–	+	+	+	–	–	–	–	–	256	O P D
Nomenclature	(4)	[Nom]	–	–	+	+	+	+	+	–	–	–	1024	O P D
Off Stealth		[Off]	+	–	+	+	+	+	+	–	–	–	1689	O P D
Ohio		[Ohio]	–	–	+	–	–	–	–	+	–	–	n/a	B
Ontario		[Ont]	–	+	+	+	+	+	–	–	–	–	Változó	O P D
Oropax	(5)	[Oro]	–	–	+	–	+	–	–	–	–	–	2773	P O
P1	(6)	[P1r]	–	+	+	–	+	–	–	–	–	–	Változó	O P D L
Pakistani Brain**			–	–	+	–	–	–	–	+	–	–	n/a	B
Paris		[Paris]	–	–	–	+	+	+	+	–	–	–	4909	O P D L
Parity		[Parity]	–	–	–	+	+	–	–	–	–	–	441	O P D

Név	Változatok száma	Rövidített névkód	Hagyományosan rejtett*	A víruskódot „titkosítja"*	Rezidens része marad a tárban	COMMAND.COM	.COM állományok	.EXE állományok	Overlay-állományok	A floppy bootrekordja	A merevlemez bootszektora	A merevlemez partíciós táblájá	A fertőzött program méretnövekedése	A KÁROKOZÁS MÓDJA
Payday	(2)	[Payday]	–	–	+	–	+	+	+	–	–	–	1808	P
Pentagon		[Pentagon]	–	–	–	–	–	–	–	+	–	–	n/a	B
Perfume	(2)	[Fume]	–	–	–	–	+	–	–	–	–	–	765	P
Pest	(3)	[Pest]	–	–	+	+	+	+	+	–	–	–	1910	O P L
Ping Pong	(3)	[Ping]	–	–	+	–	–	–	–	+	–	–	n/a	O B
Ping Pong-B	(7)	[Ping]	–	–	+	–	–	–	–	+	+	–	n/a	O B
Pixel	(5)	[Pix]	–	–	–	+	+	–	–	-	–	–	779	O P
Phantom		[Phant]	–	–	+	+	+	–	–	–	–	–	2253	O P
Plague		[Plague]	–	–	–	–	+	+	–	–	–	–	—	Felülír
Plastique	(9)	[P1q]	–	–	+	+	+	+	+	–	–	–	3012	O P D
Polimer***		[Polimer]	–	–	–	+	+	–	–	–	–	–	512	O P D
Polish-2		[P-2]	–	–	+	+	+	–	–	–	–	–	512	O P D
Polish 217		[P-217]	–	–	–	+	+	–	–	–	–	–	217	O P D
Print Screen	(2)	[Prtscr]	–	–	+	–	–	–	–	+	+	–	n/a	B O D
RedX	(2)	[Redx]	–	–	–	+	+	–	–	–	–	–	796	O P
Saddam		[Saddam]	–	–	+	+	+	–	–	–	–	–	919	O P D L
Saratoga		[Ice]	–	–	+	–	–	+	–	–	–	–	632	O P
Saturday 14th	(3)	[Sat14]	–	–	+	–	+	+	+	–	–	–	685	F O P L
Scott's Valley		[Sval]	–	+	+	–	+	+	+	–	–	–	2133	L O P D
Sentinel		[Sent]	–	–	+	+	+	+	+	–	–	–	4625	L O P D
Shadow Byte		[Sha]	–	–	–	+	+	–	–	–	–	–	723	O P
Shake		[Shake]	–	–	+	–	+	–	–	–	–	–	476	O P
Skism		[Skism]	–	–	+	–	+	+	+	–	–	–	1815	O P
Slow	(5)	[Slow]	–	+	+	–	+	+	+	–	–	–	1721	O P L
Solano	(3)	[Solano]	–	–	+	–	+	–	–	–	–	–	2000	O P L

Név	Változatok száma	Rövidített névkód	Hagyományosan rejtett*	A víruskódot „titkosítja"*	Rezidens része marad a tárb	COMMAND.COM	.COM állományok	.EXE állományok	Overlay-állományok	A floppy bootrekordja	A merevlemez bootszektora	A merevlemez partíciós tábl	A fertőzött program méretnövekedése
Sorry		[Sorry]	–	–	+	+	+	–	–	–	–	–	731
Spar		[Spar]	–	–	+	+	+	+	–	–	–	–	1255
Spyer		[Spyer]	–	–	+	–	+	+	+	–	–	–	1181
Staf		[Staf]	–	–	+	+	+	–	–	–	–	–	2083
Star Dot		[Sdot]	–	–	+	–	+	–	–	–	–	–	801
Stone-90		[S90]	–	–	–	+	+	–	–	–	–	–	961
Stoned-II		[Stoned]	–	–	+	–	–	–	–	+	–	+	n/a
Stoned/Marijuana	(5)	[Stoned]	–	–	+	–	–	–	–	+	–	+	n/a
Striker		[STR]	–	–	–	–	+	–	–	–	–	–	46
Subliminal	(3)	[Sub]	–	–	+	+	+	–	–	–	–	–	1496
Sunday	(4)	[Sunday]	–	–	+	–	+	+	+	–	–	–	1636
Suriv01	(2)	[SurivA]	–	–	+	–	+	–	–	–	–	–	897
Suriv02		[SurivA]	–	+	–	–	+	–	–	–	–	–	1488
Suriv03		[SurivB]	–	–	+	–	+	+	+	–	–	–	n/a
Sverdlov		[Sverd]	–	–	+	+	+	+	+	–	–	–	1962
Swap Boot		[Swap]	–	–	+	–	–	–	–	+	–	–	n/
Swiss 143		[S143]	–	–	–	+	+	–	–	–	–	–	14
Sylvia/Holland		[Holland]	–	–	–	–	+	–	–	–	–	–	133
S-847		[S847]	–	–	+	–	+	–	–	–	–	–	85
Taiwan	(3)	[Taiwan]	–	–	–	–	+	–	–	–	–	–	70
Taiwan3		[T3]	–	–	+	+	+	+	+	–	–	–	290
Taiwan4		[T4]	–	–	+	+	+	+	+	–	–	–	257
TCC		[TCC]	–	–	–	+	+	+	+	–	–	–	490
Telecom File		[Tele]	–	+	+	–	+	–	–	–	–	–	370
Telecom Boot		[Tele]	–	+	+	–	–	–	–	–	+	+	n/

MEGNEVEZÉS			A FERTŐZÉS HELYE											
Név	Változatok száma	Rövidített névkód	Hagyományosan rejtett*	A víruskódot „titkosítja"*	Rezidens része marad a tárban	COMMAND.COM	.COM állományok	.EXE állományok	Overlay-állományok	A floppy bootrekordja	A merevlemez bootszektora	A merevlemez partíciós táblája	A fertőzött program méretnövekedése	A KÁROKOZÁS MÓDJA
Terror		[Ter]	–	–	+	+	+	+	+	–	–	–	1085	O P F
Tester		[TV]	–	–	+	+	+	–	–	–	–	–	1000	O P
Tiny	(13)	[Tiny]	–	–	–	+	+	–	–	–	–	–	163	O P
Tiny-133		[T133]	–	–	–	+	+	–	–	–	–	–	133	O P
Traceback	(3)	[3066]	–	–	+	–	+	+	–	–	–	–	3066	P
Tumen V0.5		[Tum5]	–	–	+	+	+	–	–	–	–	–	1663	O P L D
Tumen V2.0		[Tum2]	–	–	+	+	+	–	–	–	–	–	1092	O P L D
Turbo Kukac***			–	–	–	+	+	–	–	–	–	–	512	O
Typo Boot		[Typo]	–	–	+	–	–	–	–	+	+	–	n/a	O B
Typo/Fumble		[Typo]	–	–	+	–	+	–	–	–	–	–	867	O P
Töltögető/Fill***			–	–	–	–	–	–	–	+	+	+	n/a	B P
Unrecognized Boot Sector			–	–	+	–	–	–	–	+	+	–	n/a	B O P
USSR		[USSR]	–	+	–	–	–	+	–	–	–	–	575	O P
USSR	(3)	[USSR]	–	+	–	–	–	+	–	–	–	–	575	O P
USSR-256		[U256]	–	+	–	+	+	–	–	–	–	–	256	P D
USSR-257		[U257]	–	+	–	+	+	–	–	–	–	–	257	P D
USSR-311		[U311]	–	–	–	–	+	–	–	–	–	–	321	O P
USSR-394		[U394]	–	+	–	+	+	–	–	–	–	–	394	P D
USSR-492		[U492]	–	–	–	–	+	–	–	–	–	–	492	O P
USSR-516		[U516]	–	–	+	+	+	–	–	–	–	–	516	O P
USSR-529		[U529]	–	–	+	+	+	–	–	–	–	–	529	O P
USSR-600		[U600]	–	+	–	+	+	–	–	–	–	–	600	P D
USSR-696		[U696]	–	+	–	–	+	–	–	–	–	–	696	P D
USSR-707		[U707]	–	+	–	+	+	–	–	–	–	–	707	P D

MEGNEVEZÉS			A FERTŐZÉS HELYE											
Név	Változatok száma	Rövidített névkód	Hagyományosan rejtett	A víruskódot „titkosítja"	Rezidens része marad a tárban	COMMAND.COM	.COM állományok	.EXE állományok	Overlay-állományok	A floppy bootrekordja	A merevlemez bootszektora	A merevlemez particiós táblája	A fertőzött program méretnövekedése	A KÁROKOZÁS MÓDJA
USSR-711		[U711]	–	+	–	–	+	–	–	–	–	–	711	P D
USSR-830		[U830]	–	–	+	+	+	–	–	–	–	–	830	O P
USSR-948		[U948]	–	+	–	–	+	+	+	–	–	–	948	O P D
USSR-1049		[U1049]	–	–	+	+	+	–	–	–	–	–	1049	O P
USSR-2144		[U2144]	–	+	+	+	+	+	+	–	–	–	2144	L O P D
V-299		[V299]	–	–	–	–	+	–	–	–	–	–	299	O P D
V-555		[555]	–	–	+	+	+	+	+	–	–	–	555	O P L
V-961		[V961]	–	–	+	+	+	–	–	–	–	–	961	O P
V800	(3)	[V800]	+	+	+	–	+	–	–	–	–	–	n/a	O P L
V-801		[V801]	–	–	+	+	+	+	+	–	–	–	801	O P L
V2000	(3)	[2000]	–	–	+	+	+	+	+	–	–	–	2000	O P L
V2100	(2)	[2100]	–	–	+	–	+	+	–	–	–	–	2100	O P D L
Vacsina	(5)	[Vacs]	–	–	+	–	+	+	+	–	–	–	1206	O P
Vacsina V05			–	–	+	–	+	+	+	–	–	–	1217	O P
Vacsina V16			–	–	+	–	+	+	+	–	–	–	1530	O P
Vacsina V24		[Vacs]	–	–	+	–	+	+	+	–	–	–	1760	O P
Vcomm	(5)	[Vcomm]	–	–	–	–	–	+	–	–	–	–	1074	O P L
Victor	(2)	[Victor]	–	–	+	+	+	+	+	–	–	–	2458	P D L
Vienna-B		[Vienna]	–	–	–	–	+	–	–	–	–	–	648	P
Vienna/648	(23)	[Vienna]	–	–	–	–	+	–	–	–	–	–	648	P
Violator	(6)	[Vio]	–	–	–	+	+	–	–	–	–	–	1055	O P D
Virus-90		[90]	–	–	+	–	+	–	–	–	–	–	857	P
Virus-101		[101]	–	+	+	+	+	+	+	+	–	–	2560	P
Voronezh		[Voro]	–	+	+	+	+	+	+	–	–	–	1600	O P D
W-13	(4)	[W13]	–	–	–	–	+	–	–	–	–	–	532	O P

MEGNEVEZÉS			A FERTŐZÉS HELYE											
Név	Változatok száma	Rövidített névkód	Hagyományosan rejtett*	A víruskódot „titkosítja"	Rezidens része marad a tárban	COMMAND.COM	.COM állományok	.EXE állományok	Overlay-állományok	A floppy bootrekordja	A merevlemez bootszektora	A merevlemez partíciós táblája	A fertőzött program méretnövekedése	A KÁROKOZÁS MÓDJA
Warrior		[War]	–	–	+	–	–	+	–	–	–	–	1024	O P D
Whale	(3)	[Whale]	+	+	+	+	+	+	+	–	–	–	9216	L O P D
Wisconsin		[Wisc]	–	+	–	+	+	–	–	–	–	–	825	O P D
Wolfman	(2)	[Wolf]	–	–	+	+	+	+	–	–	–	–	2064	O P
XA1		[XA1]	–	+	–	–	+	–	–	–	–	–	1539	F O P L
Yankee-2		[Doodle2]	–	–	–	–	+	+	–	–	–	–	1961	O P
Yankee-H3**			–	–	–	–	+	+	–	–	-	–	2932	O P
Yankee-H4**			–	–	–	–	+	+	–	–	–	–	2941	O P
Yankee Doodle	(6)	[Doodle]	–	–	+	–	+	+	–	–	–	–	2885	O P
Yankee Doodle v1			–	–	+	–	+	+	–	–	–	–	2890	O P
Yankee Doodle v2			–	–	+	–	+	+	–	–	–	–	2940	O P
Yankee Doodle v3			–	–	+	–	+	+	–	–	–	–	2772	O P
ZeroHunt		[Hunt]	+	+	+	–	+	–	–	–	–	–	n/a	O P D
144		[144]	–	–	–	+	+	–	–	–	–	–	144	O P
268-Plus		[268P]	–	–	–	+	+	–	–	–	–	–	270	O P L D
337		[337]	–	–	+	+	+	–	–	–	–	–	337	O L
400	(5)	[400]	–	–	+	–	+	–	–	–	–	–	Változó	O P D
405		[405]	–	–	–	–	+	–	–	–	–	–	—	Felülír
453		[453]	–	–	+	+	+	–	–	–	–	–	453	O P
483		[483]	–	–	+	+	+	–	–	–	–	–	483	O P
510		[510]	–	–	–	+	+	–	–	–	–	–	510	O L
512	(5)	[512]	+	–	+	+	+	–	–	–	–	–	—	O P L
529		[529]	–	–	+	+	+	–	–	–	–	–	529	O P D
651		[651]	–	–	+	–	+	–	–	–	–	–	651	O P D

MEGNEVEZÉS			A FERTŐZÉS HELYE											
Név	Változatok száma	Rövidített névkód	Hagyományosan rejtett*	A víruskódot „titkosítja"*	Rezidens része marad a tárban	COMMAND.COM	.COM állományok	.EXE állományok	Overlay-állományok	A floppy bootrekordja	A merevlemez bootszektora	A merevlemez partíciós táblája	A fertőzött program méretnövekedése	A KÁROKOZÁS MÓDJA
733		[733]	–	–	–	+	+	–	–	–	–	–	733	O P D L
777		[777]	–	–	+	+	+	–	–	–	–	–	777	O P
903		[903]	–	–	+	+	+	–	–	–	–	–	903	O P
923		[923]	–	–	+	+	+	+	+	–	–	–	923	O P L D
1008		[1008]	–	+	+	+	+	–	–	–	–	–	1008	O P D L
1024	(2)	[1024]	–	–	+	+	+	–	–	–	–	–	1024	O P
1024PSRC		[PS10]	–	–	+	+	+	–	–	–	–	–	1024	O P
1067		[1067]	–	–	+	+	+	–	–	–	–	–	1067	O P L
1210		[1210]	–	–	+	–	+	–	–	–	–	–	1210	O P L
1226	(3)	[1226]	–	+	+	+	+	+	+	–	–	–	1226	O P D
1253 Boot		[1253]	–	–	+	–	–	–	–	+	+	+	n/a	O P D L
1253 COM		[1253]	–	–	+	+	+	–	–	–	–	–	1253	O P D L
1260	(3)	[1260]	–	+	–	–	+	–	–	–	–	–	1260	P
1260	(3)	[1260]	–	+	–	–	+	–	–	–	–	–	1260	P
1381		[1381]	–	–	–	–	–	+	+	–	–	–	1381	O P
1392		[1392]	–	–	+	+	+	+	–	–	–	–	1392	O P L
1536/Zero Bug		[Zero]	–	–	+	–	+	–	–	–	–	–	1536	O P
1559		[1559]	–	–	+	+	+	+	–	–	–	–	1554	O P L
1575/1591	(2)	[15xx]	–	–	+	+	+	+	–	–	–	–	Változó	O P L
1605		[1605]	–	–	+	+	+	+	–	–	–	–	1605	L O P D
1701/Cascade	(12)	[170x]	–	+	+	–	+	–	–	–	–	–	1701	O P
1704 Format		[170x]	–	+	+	–	+	–	–	–	–	–	1704	O P F
1704/Cascade		[170x]	–	+	+	–	+	–	–	–	–	–	1704	O P
1720	(2)	[1720]	–	–	+	–	+	+	+	–	–	–	1720	F O P L

MEGNEVEZÉS			A FERTŐZÉS HELYE												
Név	Változatok száma	Rövidített névkód	Hagyományosan rejtett	A víruskódot „titkosítja"	Rezidens része marad a tárban	COMMAND.COM	.COM állományok	.EXE állományok	Overlay-állományok	A floppy bootrekordja	A merevlemez bootszektora	A merevlemez partíciós táblája	A fertőzött program méretnövekedése	A KÁROKOZÁS MÓDJA
1720	(3)	[1720]	–	–	+	–	+	+	+	–	–	–	1720	F O P L
1963		[1963]	+	–	+	+	+	+	+	–	–	–	1963	O P L D
1971/8 Tunes	(2)	[1971]	–	–	+	–	+	+	+	–	–	–	1971	O P
2930		[2930]	–	–	+	–	+	+	–	–	–	–	2930	P
3445		[3445]	+	+	+	–	+	+	–	–	–	–	3445	O P D L
3551/Syslock		[Syslock]	–	+	–	–	+	+	–	–	–	–	3551	P D
4096	(4)	[4096]	+	–	+	+	+	+	+	–	–	–	4096	D O P L
5120	(3)	[5120]	–	–	–	+	+	+	+	–	–	–	5120	O P D L
7808		[7808]	–	–	+	+	+	+	+	–	–	–	7808	O P L D

UTÓSZÓ

Végére érkezvén az Új viruslélektannak, láthatják, mennyire megváltozott a vírusvilág azóta, hogy megjelent az első hazai vírus szakkönyv. Mindenkinek fáj, amikor saját koncepciójáról meg kell állapítania, hogy az túlhaladott. A továbblépéshez azonban ezt be kellett vallanunk. Sok tapasztalat birtokában és hosszas vívódás után döntöttünk úgy, hogy szemléletet kell váltanunk, és a vírusok elleni védelem további kutatását a szoftveres megoldások helyett a kártyára kell alapoznunk. Mégpedig saját fejlesztésűre, mert az általunk ismert hazai és külföldi próbálkozások alapvető szakmai követelményeinknek sem feleltek meg. Ugyanakkor a korábban sikeres szoftveres vírusvédelmet sem hanyagolhatjuk el, mert jó ideig még arra is szükség lesz.

Így ezen kettős koncepció jegyében jelentettük meg — az Ifabo 1991-es budapesti rendezvénye alkalmából — PC-SCAN szoftverrendszerünket, amely McAfee kitűnő programjánál jóval több kelet-európai vírusváltozatot ismer fel. Mire e könyv megjelenik, kapható lesz PC-CLEAN programunk is, amellyel a PC-SCAN által felismert vírusokat lehet kitakarítani.

E könyvünk szerves kiegészítőjeként hamarosan megjelenik a bevezetőben már említett Vírushatározó is, amelyben az IBM-kompatibilis gépek vírusairól részletezve és rendszerezve ismertetjük mindazt, amit sikerült megtudnunk a számítástechnika eme szörnyszülötteiről. Tartsanak velünk a második kötet néhány helyen a krimi izgalmát idéző panoptikumában!

A szerzők

A könyv szerzői a következő címeken érhetők el:

Szegedi Imre
Safe Kft.
1134 Budapest
XIII., Gidófalvy u. 31.
Telefon: 140-7681
Telefon/Fax: 183-3267

Kis János
1027 Budapest
II., Mártírok útja 24.
Telefon: 116-8896 (lakás)
Telefon/Fax: 113-3591 (munkahely)

IRODALOMJEGYZÉK

A számítógépes vírusok kutatásának szakterületén kevés a valóban forrásértékű publikáció. Az információk megszerzésében sokszor kell nem közölhető forrásokra és személyes kapcsolatokra támaszkodni. A legjobban használható, bár nehezen elérhető kiadványok ebben a témakörben a számítógépes programfeltörők (a „hackerek") kiadványai. Hasonlóan jól használhatók, de még nehezebben beszerezhetők egyes országoknak az adatbűnözéssel, illetve az adatbiztonsági ajánlásokkal kapcsolatos belső anyagai. Az USA-ban az FBI, Németországban az Alkotmányvédő Hivatal végez az adatbűnözés megelőzését szolgáló tájékoztató tevékenységet. E szervezetek nyugati szakkönyvtárakban és vállalatoknál fellelhető tanulmányai értékes információforrások.

Szegedi Imre 1990 szeptemberében védte meg doktori disszertációját a számítógépes vírusok témakörében. Annak forrásjegyzékét egészítettük ki az első víruskönyv szakirodalmi válogatásának összeállításakor, listánk pedig azóta tovább bővült. Az anyagok egy része nincs meg a hazai szakkönyvtárakban, azokhoz a Hamburgi Egyetemi Központi Könyvtárból, a bécsi Technische Hochschule könyvtárából és más nyugat-európai könyvtárakból jutottunk hozzá, könyvtárközi kölcsönzéssel vagy személyes látogatások alkalmával.

Az *Új víruslélektan* és a *Vírushatározó* irodalomjegyzékét összevontuk, és mindkét kötetben közreadjuk.

1. Adney, William A. – Kavanagh, Douglas E.: The data bandits. In: Byte, 1989. 1. 167-270. p.
2. Alaplap mikroszámítógép magazin mágneslemez melléklettel. A „Vírusőrjárat" rovatban rendszeresen közöl cikkeket a vírusokról. Publ.: Cédrus Informatikai Rt., Budapest. (A továbbiakban: Alaplap.)
3. Auf der Knie. In: Der Spiegel, 1988, 11. 7. 294. p.
4. Bayerische Hackerpost. München. (A bajor számítógépbetörők szakmai fóruma.)
5. Bombenstimmung. In: Computer Live, 11/1990. Összeállítás. Kivonatosan közli a Computer Panoráma 1991. januári száma.
6. Börstler, Torsten – Fischer, Christoph: Sabotage vorprogrammiert!

Computer-Viren bedrohen Datenbestände. In: CAK Nr. 8. 1989. okt. 44-53. p.
7. Brown, Richard: Interrupt list rel. 91. 1. 5. Szövegállomány. Via HomeBase BBS, USA 1991.
8. Brunnstein, Klaus: Blindes Vertrauen in den Computer. Unterschätztes Risiko. In: Bild der Wissenschaft, 2/1988. 96. p.
9. Brunnstein, Klaus: Mythen und Fakten über Computer-Viren. In: Chip, 1989. 3. 50-56. p.
10. Brunnstein, Klaus: PC-Viren. Dichtung und Wahrheit. In: Computer Magazin, 1989. 9. 47-49. p.
11. Brunnstein, Klaus: Risiken der Informationsverarbeitung. In: Computer Magazin, 1989, 1-2. 31-35. p.
12. Brunnstein, Klaus: Viren-Telex mit Virus-Katalog. Ein monatlicher Informationsbrief für Datensicherheit, 1989-1990. Ed.: Vogel Verlag, Würzburg.
13. Brunnstein, Klaus: Über Viren, Würmer und andere seltsame Geister in Computersystemen — Ein kleines Informatik-Bestiarium. In: Angewandte Informatik, 1987. 10. 397. p.
14. Brunnstein, Klaus: Zur Klassifikation von Computer-Viren. Der Computer Virus Katalog. In:Tagungsband der 19. GI-Jahrestagung.
15. Brunnstein, Klaus – Fischer-Hübner, S.; Swimmer M.: Classification of Computer Anomalies Security Conference, New York 1991.
16. Brunnstein, Klaus – Fischer-Hübner, S.; Swimmer M.: Concepts of an Expert System for Virus Detection. In: IFIP TC11 Security Conference, 1991
17. Brunnstein, Klaus – Fischer-Hübner, S.: Risk Analysis of Trusted Computer Systems In: IFIP TC11 Security Conference, 1990 Helsiki
18. Bunge: Die jüngsten Prüfungsergebnisse des Bundesrechnungshofes zur Datensicherheit. In: 14. DAFTA Tagung in Köln. Gesellschaft für Datenschutz und Datensicherheit e. V. 1990 november.
19. Burger, Ralph: Das groe Computer-Viren Buch. Ed.: Data Becker GmbH, Düsseldorf–Wien, 1987. (Későbbi kiadásait részben átírták, aktualizálták.)
20. Burger, Ralph: Das groe PC Viren Schutzpaket. Ed.: Data Becker GmbH, Düsseldorf–Wien, 1989.
21. Buruzs Tamás: A számítógépes vírusok természetrajza. Szakdolgozat 1991/S-3. Ed.: Kandó Kálmán Villamosipari Műszaki Főiskola, Budapest. A szakdolgozat a benne lévő teljes víruskód (Potyogós) miatt csak szakmai kutatás céljára hozzáférhető, zárt anyag!

22. Buruzs Tamás: Rezidens Virus Killer (RVK). Dokumentációs állományai.
23. Buruzs Tamás: Self Protection Systetms (SPS). Dokumentációs állományai.
24. Cohen, Fred: „Computer Viruses". Dissertation. University of Southern California. Ed.: USC, 1985.
25. Cohen, Fred: Computer Viruses: Theory and Experiments. Ed.: University of Southern California, 8/1984. Reprint in Computer & Security, 6/1984.
26. Cohen, Fred: Models of Practical Defenses against Computer-Viruses. In: Computer & Security, 2/1989.
27. Computerworld-Számítástechnika. Rendszeresen közölt a vírusokra vonatkozó információkat és előrejelzéseket. Publ.: IDG Lapkiadó Kft., Budapest. (A továbbiakban: CWI illetve IDG.)
28. Cremer, Dorothea – Pohl, Harmuth: Zur Computerkriminalität im 5. StÄG der DDR und 2. WiKG der Bundesrepublik aus der Sicht der Informationstechnik 1.– 2. In: Datenschutz und Datensicherung 1990/10. 493-497. p. és 1990/11. 551-558. p.
29. Datenschleuder. Hamburg. (A Chaos hackercsoport folyóirata.) Ed.: Chaos, Hamburg.
30. Dierstein, Rüdiger: Anmerkungen zur Rechtslage. Programm-manipulationen — Trojanische Pferde, Viren und ihre Bekämpfung. Ed.: Carls-Cranz-Gesellschaft e.V. Oberpfaffenhofen 1991 April.
31. Dierstein, Rüdiger: Computer-Viren In: DFVLR Institutsbericht, IB 582/6, 1986 Juli.
32. Dierstein, Rüdiger: Computer-Viren 1. In: KES, 1985. 03. 77-86. p.
33. Dierstein, Rüdiger: Computer-Viren 2. In: KES, 1985. 04. 125-135. p.
34. Dierstein, Rüdiger: Computer-Viren 1. In: Output, Nr. 1986/8. 33-40. p.
35. Dierstein, Rüdiger: Computer-Viren 2. In: Output, Nr. 1986/10. 43-47. p.
36. Dierstein, Rüdiger: Computer-Viren — Was man jetzt darüber wissen mu. In: PM Computerheft 1989. März-Apr. 16-21. p.
37. Dierstein, Rüdiger: Computer Virus. In: Conference Proceedings of the Scuricom 86, 4th Worldwide Congress on Computer and Communications Security and Protection, Paris, 1986. Febr.
38. Dierstein, R.: Das Israel Virus. In: KES, 2/1988.
39. Dierstein, Rüdiger: Die neue Gefahr — Computer Viren In: KES, Zeitschrift für Kommunikations- und EDV-Sicherheit, (továbbiakban: KES), 3/85-4/85, Peter Hohl Verlag, Ingelheim.

40. Dierstein, Rüdiger: Programm-Manipulation-Computer. Viren und deren Bekämpfung. In: Recht der Datenverarbeitung RDV, Heft 3. 1989. 101. p.
41. Fisher, Christoph: Grundlagen der Virenbekämpfung In: PC Woche, 1991/10(1). 4-16. p.
42. Fisher, Christoph: Tarnkappaviren sind nicht unentdeckbar. In: Datenschutz Berater, 1991/3, 1-4. pp 14.(3)
43. Die Hackerbibel. Vol. 1.-2. (Német hacker-kiadvány.) Ed.: Chaos, Hamburg.
44. Ducan, R. (compiled): The MS-DOS Encyclopedia. Ed: Microsoft Press, Redmond, Washington, 1988.
45. Elmer-DeWin, P.: Invasion of the Data Snatchers! In: Time, 26/9/1988. 62. p.
46. Experimente mit Computer-Viren. A KES 2/87. száma idézi a Die Datenschleuder underground lapot. (No.18. 2/1987.)
47. Fites, P.; Johnston, P.; Kratz, M.: The Computer Virus Crisis. Ed.: Van Nostrand Reinhold, N.Y. 1989.
48. Flu_Shot+ ver.1.5 User manual. Ed.: Software Concepts Design, N.Y. 1989.
49. Frost, David: The Complete Computer Virus Handbook. Ed.: Price Waterhouse, 1988.
50. Greenberg, R.M.: Know the Viral Enemy. In: Byte, 6/1989. 275. p.
51. Günter, Frhr. von Gravenreuth: Computer Viren, Datenspione, Crasher und Cracker. In: Neue Zeitschrift für Strafrecht, Heft. 5. 1989. 201-248. p.
52. Günter, Frhr. von Gravenreuth: Rechtliche Beurteilung von Computer Viren: GI Fachgespräch, Okt. 1989. Springer Verlag, Tagungsband der 19. GI-Jahrestagung. 1989. Band 1. 619-628. p.
53. Hirst, Joe: List of known PC viruses. Publ.: British Computer Virus Research Center, Brighton/Essex, 1989.
54. Hoppenrath, D.: Computerviren: Problem oder Psychose. In: Computer Persönlich, 3/1989. 45. p.
55. Hoppenrath, D.: Impfung via Software. In: Computer Persönlich, 3/1989. 48. p.
56. Hoppenrath, D.: Kranke Programme. In: PC-Magazin, 35/1988. 20. p.
57. Hozzászólás vírusügyben. In: CWI, 1988. 25. szám.
58. Goodwin, Jim: Virus Information Summary List. In: VSUM9003.ZIP 1990-02-18 from Homebase/CVIA Bulletin Board BBS, USA.
59. Kane, Pamela: V.I.R.U.S. Protection. Vital information resources

under siege. Foreword by Dvorak, John C. Ed.: Batham Books New York, 1989. & Dr. Panda Utilities by Andy Hopkins from Paralex Ltd. New York.
60. Kastenmüller, S.: Erkennen von Computer-Viren. In: KES 4/1988.
61. Kis János: A tiltott gyümölcs mindig kívánatos. In: Alaplap, 1990. 9. szám, 38. p.
62. Kis János: Egy veszélylehetőség realitássá vált. Virtank.doc, a Prgdoki 2.11...2.13 verzióihoz adott összefoglaló dokumentációs állomány. Szamizdatként Budapesten, Kecskeméten. 1988-1989.
63. Kis János: Hogyan kell vírust írni? In: Delta-Impulzus, 1989. 9. szám, (V. 6.), 40. p.
64. Kis János: Modern trójai háború. In: Delta-Impulzus, 1989. 8. szám, (IV. 22.), 24. p.
65. Kötél Gyula: A programfejlesztés módszertani kérdései a katonai információfeldolgozási rendszerek fejlesztésében. Egyetemi doktori értekezés. Ed.: Zrínyi Miklós Katonai Akadémia 287/6/88 nyt. sz, Budapest. 1990. (Kutatási célra hozzáférhető.)
66. Küzdelem a számítógépes vírusok ellen. Steve, R. – White David – M. Chess Cheng – Jimmy Kuo tanulmánya az IBM részére In: Floppy.Lap mágneslemez folyóirat. Cédrus Kt. 1991/1.-2.-3. Kis János utószavával. A fordítás alapjául szolgáló szöveg az IBM kutatási jelentések sorozatában jelent meg: Report Number RC 14405 1989 IBM Los Angeles Scientific Center Los Angeles, CA
67. Labor, Zeitschrift für Word Processing. (Víruscikkek, adatátvitel.) Technikai szamizdat. Ed.: Labor c/o Glaser, D-2000 Hamburg 50, Hospital Str. 61.
68. Másolás? Védelem? (A hónap témája. Összeállítás.) In: Alaplap, 1991. 1. szám.
69. McAfee, John: Scanxx.DOC, Cleanxx.DOC, Netscan.DOC, Vshieldxx.DOC, Mdisk.doc, Virlist.txt szoftver-dokumentációs állományok. 1988-1991.
70. McAfee, John: The virus cure. In: Datamation, 1989. 02. 15. 29-40. p.
71. Mosich, Donna: Norton Antivirus User Manual 1.0.1 Ed.: Szmatec Corp. USA Cupertino CA. 199O és a program dokumentációs állományai.
72. MS-DOS-Viren erkennen und bekämpfen. Chip Special, No. 82005/90003 1. Aufl. Ed.: Vogel Verlag, Würzburg, 1990.
73. Mutopf, Günther: Drei Schritte zur Heilung. In: Chip, 11/1989. Ed.: Vogel Verlag, Würzburg, 1989.
74. Mutopf, Günther (comp.): Trojanische Pferde, Viren und Wür-

mer. Eine ernstzunehmende Gefahr für PC-Anwender. Ed.: PerComp Verlag GmbH, Hamburg, 1989.
75. Mutopf, Günther: Wenn die Programme auf der Platte Amok laufen. Serie In: Die Computerwoche, 5/1990, 34. p., 6/1990, 26. p., 7/1990. 30.p.
76. Péntek 13-a! Vírusölő program. In: CWI, 1989. 36. szám.
77. Rablók és pandúrok. In: CWI, 1989. 6. szám.
78. Roberts, R.: Computer Viruses. Ed.: Compute! Books, Greensboro, NC, 1988.
79. Rubenking, N.J.: Infection Protection. In: PC Magazine, 4/1989. 193. p.
80. Schöneburg, E.: Computer Centre Risk Analysis by Expert Systems. In: Dornier Post 1/1987. Dornier GmbH, Friedrichshafen.
81. Schöneburg, E.: Computer-Viren — Eine aktuelle Bedrohung für Computer-Systeme. In: Dornier Post, 1/1987. Dornier GmbH, Friedrichshafen.
82. Schöneburg, E.: Computer-Viren und Trojanische Pferde. Gefährliche Softwareangriffe an Computersysteme. In: Neue Zürcher Zeitung, 1987. 9. 29.
83. Schöneburg, Eberhard – Heinzmann, Frank – Namyslik, Frank: Computer-Viren. Gefahren und Schutzmöglichkeiten. Ed.: Markt und Technik Verlag, Haar bei München, 1989.
84. Schöneburg, Eberhard – Heinzmann, Frank – Namyslik, Frank: Virus Power Pack (Programm und Buch). Ed.: Markt und Technik Verlag, Haar bei München, 1989.
85. Shapira, Eli – Sherman, Yuval: Turbo Anti Virus Toolkit *Tntvirus* ver. 6.80A dokumentációs állománya és felhasználói kézikönyve. Ed.: Carmel Software, Haifa, 1990.
86. Shapira, Eli – Sherman, Yuval: Turbo Anti Virus Toolkit *Tntvirus* ver. 6.71B demó verzió dokumentációs állomány. Ed.: Carmel Software, Haifa, 1990.
87. Skulason, Fridrik: F-Prot antivirus programcsomag dokumentációs állományai. Ed.: Reykjavik, 1991 febr.
88. Small business Innovation Research Program. Ed.: US Defense Dept, Washington DC. 1990. (A 45. oldaltól víruspályázat: Computer Virus and Electronic Counter Measures)
89. Sperber, J.: Virusfieber. In: Microcomputer Zeitschrift, 7/1988. 74. p.
90. Számítógépvírusok, avagy ki fél a cyberpunkoktól? In: CWI, 1989. 31. szám.
91. Szegedi Imre: Harc az adatgyilkosok ellen. In: Alaplap. 1990. 8. szám, 32. p.

92. Szegedi Imre: Megindult a hazai vírustenyésztés? In: Alaplap, 1990. 10. szám, 36. p.
93. Szegedi Imre: Személyi számítógépes vírusok elterjedésének veszélyei és az ellenük való védekezés a Magyar Honvédségben. Első magyar víruskönyv. (Doktori értekezés.) Magyar Honvédség, Zrínyi Miklós Katonai Akadémia 587/4/90, Budapest, 1990. (A benne közölt teljes víruskódok miatt nem publikálható anyag.)
94. Szegedi Imre: Szisztematikus doktorálás. In: Alaplap, 1990. 9. szám 36. p.
95. Technical Notes on AIDS DISK Trojan Mail Information. In: AIDSTECH.ZIP, 1989-12-23. From: Homebase/CVIA Bulletin Board BBS, USA.
96. Terjed a vírusjárvány az Egyesült Államokban. In: CWI, 1989. 22. szám.
97. Tűzre, vízre, adatokra vigyázzatok. In: CWI, 1989. 34. szám.
98. Újabb gyógyszer a Péntek 13-a ellen. In: CWI, 1989. 40. szám.
99. Veldman, Frans: A TBSCAN és a TBSCANX szekvenciális víruskereső programok leírásai: TBSCAN.DOC (1990 12. 15) és TBSCANX.DOC (1991. 04. 02) Thunderbyte BBS.
100. Verborgener Befehl — Bericht Cohens Arbeit. In: Der Spiegel, 4/1987.
101. Védőoltás vírus ellen. In: CWI, 1989. 22. szám.
102. Vírusok. In: CWI, 1988. 13. szám.
103. Vírusvadászat. (A hónap témája. Összeállítás.) In: Alaplap, 1991. 8. szám.
104. Woehlebier, H.: Der Weihnachtsbaum, der um die Welt ging. In: KES, 1/1988.

TARTALOMJEGYZÉK

Elöljáróban
avagy NIL NOCERE 5
Másolásvédelem és biztonság
Amikor a pandúrból rabló lesz 9
Hadibacik
Célpont a vezérlőrendszer 24
AIDS-tájékoztató lemez
Informatikai merénylet trójai módra 33
Az adatfeldolgozás szabadsága
Ki kit és mi mit véd? 50
Vírustipológia
A családfa változásai 64
Mit tegyünk, ha jön a vírus...?
...És hogy ne jöjjön! 73
Milyen jó a vírusvédelem?
Minden kaput becsukni 85
Számháború vaklármával
A szükséges és elégséges védelem 96
A Scan-család
Semmi sem tökéletes 100
Szekvenciális keresőprogramok
A vírusok különös ismertetőjele 116
Norton a vírusszínre lép
Nem mind arany... 131
Az F-Prot csomag
Válogatás egy izlandi svédasztalnál 136
Kártyajátékok
„A vírusok nem kártyakompatibilisak" 140
A vírusok összefoglaló táblázata
A McAfee Associates 1991. júniusi listája alapján 153
Utószó 167
Irodalomjegyzék 168

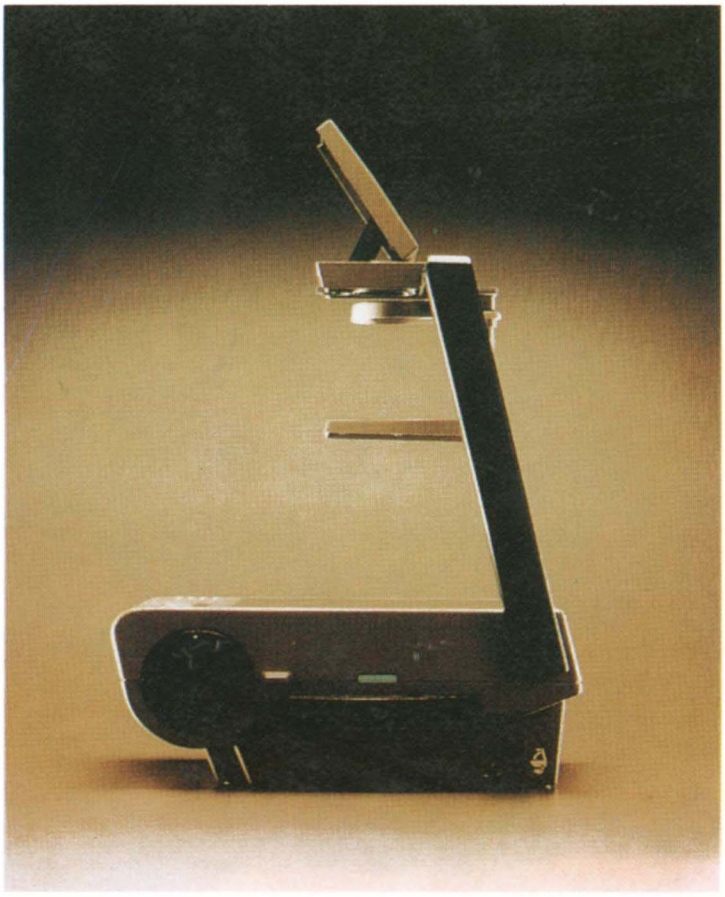
256,- FT

Ez a helyes írásvetítő!

Nappali fény mellett is kitűnő képet varázsolnak a vászonra a Polaroid írásvetítők.

A könnyen hordozható legkisebb modell mindössze 4,5 kg.

Előadások látványossá tételéhez ideális útitárs!



Többféle változatban kaphatók a

FLOPPYLAND

számítástechnikai szaküzletben
(Budapest V., Váci utca 84. Telefon/Fax: 118-2651)
és országszerte a Cédrus Rt. viszonteladóinál.

Polaroid