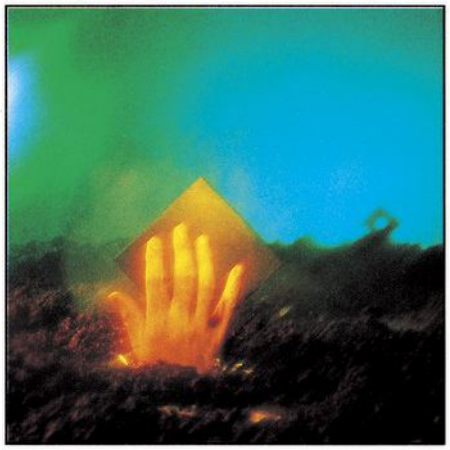

Schützer kontra Knackis

Wie kann man ein Programm sinnvoll schützen? Und wie kann man solch einen Schutz knacken? Um diese Fragen zu klären, unterhielten wir uns mit einigen Programmierern und Knackern. Sie erklärten uns ausführlich die Grundlagen des Softwareschutzes, des Knackens und des Kopierens.

vielfältigungsstücke eines Werkes herstellen. Das heißt man darf die Kopie eines Originalprogramms (nicht einer Kopie oder Knackversion) an bis zu sieben Freunde verteilen, die diese aber nicht noch einmal kopieren dürfen. Hier taucht allerdings wieder die Frage auf, was man alles unter den Begriff »Freunde« setzt.

Nehmen wir jetzt aber einmal an, 20 Leute tun sich zusammen, um ein Originalprogramm zu kaufen. Sowohl das Urheberrechtsgesetz als auch bisherige Gerichtsurteile lassen die Frage offen, ob alle zusammen oder jeder von ihnen sieben Kopien für den eigenen Gebrauch herstellen darf. Immerhin wären das 140 legale Kopien..... also keine Raubkopien, sondern »dezentralisierte Sicherheitsbackups«.

Das gesunde Rechtsempfinden eines Richters wird sicher all die oben genannten Überlegungen über den Haufen werfen, aber trotzdem: Das deutsche Urheberrechtsgesetz weist mehr Löcher auf als der Schweizer Käse. Das UrhG schützt nur deutsche Staatsangehörige (§120). Aufgrund des Assimilationsprinzips (Welturheberrechtsabkommen, Artikel II und Berner Übereinkunft, Artikel 3) werden Ausländer im Inland wie Inländer geschützt. Aus diesem Grunde ist auch der amerikanische Copyright Act in der Bundesrepublik nicht anwendbar. Schade, denn dieses Gesetz weist weit weniger Unklarheiten auf: Computerwerke sind darin schon gesondert geregelt.

Und wenn Sie nun sagen, das wäre alles an den Haaren herbeigezogen, dann haben Sie vollkommen recht. Aber sind die Anwälte der Softwarefirmen nicht auch nur dazu da, ihr Recht an den Haaren herbeizuziehen, weil das Gesetz so viele Lücken aufweist?

(M. Kohlen/aa)

Als wir uns nach den Grundlagen der Schutztechniken erkundigten, wurden wir gleich vor eine Frage gestellt: Meinten wir nun den Programm- oder den Kopierschutz? Man klärte uns auf, daß man bei Schutzmethoden folgende Unterscheidung machen muß: Als Programmschutz werden programmtechnische Maßnahmen bezeichnet, die im Computer ablaufen und das Knacken verhindern sollen. Ein Kopierschutz hingegen befindet sich auf der Diskette oder Kassette und soll das Programm vor Kopierversuchen sichern. Nur die Kombination dieser beiden Techniken ist heutzutage zum Schützen von Programmen sinnvoll.

Programmschutz

Ein Programmschutz bedeutet, daß es einem Außenstehenden fast unmöglich gemacht wird, die Arbeitsweise des Programms zu verfolgen. Hier gibt es viele Techniken, die sich allerdings auf die Maschinensprachebene beschränken. Die erste und am weitesten verbreitetste ist die Codierung von Programmteilen. Nur ein kleiner Teil des Programms liegt lauffertig vor, der Rest ist in irgendeiner Form codiert und wird erst bei Bedarf decodiert. Das allein wäre aber nicht effektiv genug: Meistens liegen die Decodierprogramme ebenfalls codiert vor, müssen also von anderen Teilen decodiert werden, und so weiter. Eine Verschachtelung von bis zu zwanzig Codier- und Decodier-Routinen, die sich vielleicht noch gegenseitig aufrufen, ist keine Seltenheit.

Fast immer wird das Codier-Prinzip mit dem Verschiebe-Prinzip gepaart: Die einzelnen Programmteile werden durch Verschieberoutinen über den gesamten Arbeitsspeicher von 64 KByte verstreut. Freaks sagen dazu auch »Spreaded Code«. Der arme Knacker hat dann praktisch keine Chance, irgendwo einen Monitor oder ein sonstiges zusätzliches Programm unterzubringen, da in jeder Ecke des Speichers ein paar Byte des Programms stehen. Weiterer Vorteil ist die damit verbundene Reset-Sperre: Man bringt Teile des Programms in Zeropage, Prozessorstack und Bildschirmspeicher unter, denn die werden bei einem Reset, so ausgetüftelt er auch sein mag, immer teilweise gelöscht.

Die nächste Schutzmethode sind die sogenannten »Illegalen Opcodes«, Maschinensprachebefehle, die offiziell gar nicht existieren. Über diese Opcodes haben wir ja schon öfter berichtet. Inzwischen sind die Softwarefirmen von der Verwendung der »Illegals« abgekommen. Aus zwei Gründen: Einerseits kennt die sowieso schon fast jeder, andererseits funktionieren die Opcodes, die man zum Schutz am besten brauchen könnte, nicht auf allen C 64. Der Grund dafür ist, das auch Fremdhersteller die 6510-CPU für Commodore in Lizenz fertigen und daß sich die einzelnen Typen nicht exakt entsprechen. Ungefähr die Hälfte der illegalen Opcodes verhalten sich auf verschiedenen Gruppen von C 64 völlig unterschiedlich. Das bringt natürlich so manchen ausgetüftelten Programmschutz zu Fall, denn was will man mit einem Programm, das nur auf einem Viertel aller C 64 läuft?

Letzte und schwierigste Methode: Selbstmodifizierender Code. Dieser Zungenbrecher bezeichnet nichts weiter, als daß sich ein Programm selbst verändert. In einem Programmteil wandelt sich dann zum Beispiel ein Sprungbefehl nach $7698 zu einem nach $4435 um oder ein LDA # wird zu einem STX. Das läßt sich sowohl sinnvoll in Programmen einsetzen, gerade in sehr schnellen Grafik-Routinen, aber auch nur zur Verwirrung des Betrachters. Es geht sogar noch schlimmer: Ein in den Interrupt eingehängtes Programm ändert perio-

disch Teile des Hauptprogramms. Dann verwandelt sich ein Sprungbefehl, der ins Leere geht, auf einmal in einen sinnvollen, und das erst direkt bevor er ausgeführt wird. Die Verwandlung selbst kann man nicht per Monitor nachverfolgen, da sie aus dem Interrupt heraus geschieht. Allerdings sei eines gesagt: Solchen Code zu schreiben ist schlimmer als ihn zu knacken. Große Softwarefirmen entwickeln deswegen diesen selbstmodifizierenden Code mit anderen Computern als dem C 64, beispielsweise mit dem IBM-PC. Ein speziell dafür ausgelegter Assembler übernimmt dann automatisch die Knochenarbeit, den Code zu erstellen.

Wenn Knacken zur Routine wird

Wenn Programme aber so gut geschützt sind, wie kann man sie dann knacken? Auch hier wurden wir sofort eines Besseren belehrt: Meistens ist ja nur der Kopierschutz so versteckt. Hat man den entfernt, braucht man sich nicht mehr um Gemeinheiten im Hauptprogramm zu kümmern. Normalerweise erstreckt sich so ein Schutzprogramm über einige wenige Teilprogramme, die von der Diskette nachgeladen werden, und ist insgesamt auch nur wenige KByte lang. Da muß man dann halt alles ausdrucken und auf dem Papier nachverfolgen, Taktzyklen zählen und sich ständig Notizen machen, welche Speicherstellen wie verändert werden. So nach und nach tastet man sich dann vor, bis man den Kopierschutz entschlüsselt vor sich liegen hat und diesen dann entfernt. Bei den neueren Schutzprogrammen muß man aber fast schon hochgradig schizophren sein, will man drei oder vier Programme, die praktisch gleichzeitig ablaufen und sich gegenseitig verändern, durchschauen.

Ein Nachteil, den manche Softwarefirmen allerdings selbst zu verschulden haben: Wird ein und derselbe Programmschutz über eine längere Zeit beibehalten, so schreiben sich manche Knacker ein Programm, das automatisch Originale dieses Herstellers entschützt. Wir bekamen als Beispiel ein Programm vorgeführt, das innerhalb von 40 Sekunden die neuen Electronic-Arts-Originale, über die nachher noch zu sprechen sein wird, voll kopierfähig macht.

Damit waren wir beim zweiten Punkt angelangt: den Kopierschutzmethoden. Der Programmschutz dient meist nur dazu, das Kopierschutzprogramm zu verstecken.

Kopierschutz

Der Kopierschutz selbst schützt nicht vor Knack- sondern vor Kopierversuchen (oder soll dies zumindest). Wir erfuhren die gängigsten Verfahren des Kassetten- und Diskettenschutzes.

Bei Kassetten ist es eigentlich ganz einfach: Man entwickelt ein eigenes Aufzeichnungsformat und die dazu passenden Leseroutinen. Neben dem Kopierschutz ist dann auch gleichzeitig ein Turbo-Lader realisierbar. Ein brandneues Schutzsystem ist so ausgetüftelt, daß das Laden des Programmes unmöglich ist, wenn der C 64 minimal verändert worden oder ein zusätzliches Peripheriegerät (Floppy, Drucker oder auch nur Modul im Expansionport) angeschlossen ist. Dann schwankt die Spannungsversorgung des C 64 und somit auch die Motorsteuerung des Recorders minimal, und das haarscharfe Timing der Leserroutine bricht zusammen.

Solche Kassetten mit zwei Hi-Fi-Tapedecks zu überspielen klappt meist nicht, denn die sind zu gut und zerstören das Signal beim Überspielen durch Frequenz- und Dynamikkorrekturen. Mit billigen Recordern geht's auch nicht, weil sich da die Motorschwankungen beim Überspielen addieren. Und ein Kassettenkopierprogramm für den C 64, das alle möglichen Formate kopiert, kann aus verschiedenen technischen Gründen nicht geschrieben werden. So ist beispielsweise die Motorsteuerung durch den Computer nicht präzise genug. Kassettenprogramme sind gegen Kopierversuche im großen und ganzen besser schützbar als Diskettenprogramme. Deswegen wird englische Software hauptsächlich auf Kassette angeboten. In Amerika und Deutschland ist hingegen die Floppy so weit verbreitet, daß sich Programme nur auf Diskette in genügend großen Mengen verkaufen lassen.

Um einiges vielfältiger sind daher die Methoden des Diskettenschutzes. Die gängigsten und aktuellsten Methoden wurden uns im Schnelldurchgang vorgestellt:

Da wären erst einmal die »Errors«, künstlich auf die Diskette aufgebrachte Lesefehler. Meist werden dann auch noch Daten in den fehlerhaften Blöcken versteckt. Diese Methode ist aber viel zu bekannt, und die meisten Fehler werden von jedem zweitklassigen Kopierprogramm einfach mit übertragen. Ein Lesefehler besonderer Art sind die Killertracks, Tracks, die komplett mit Synchronmarkierungen vollgeschrieben wurden. Will man normal auf einen solchen Track zugreifen, hängt sich die Floppy unweigerlich auf. Eine Zeitlang waren auch physikalische Fehler im Gespräch: Die Diskette wird an einigen Stellen beschädigt und dann wird versucht, diese Stelle zu beschreiben. Doch dieses Verfahren ist sehr kompliziert und teuer in der Herstellung, sollen alle Disketten gleich defekt sein. Es wird deswegen nur von kleinen Firmen für Programme verwendet, die in nicht allzuhohen Stückzahlen auf den Markt kommen. Sowas ist natürlich nicht softwaremäßig kopierbar, aber so mancher hat es schon durch optische Kontrolle des Originals und gezieltes Zerstören der Kopie geschafft.

Interessanter als Errors sind da schon die Blockheader-Manipulationen. In einem Blockheader, der jedem Datenblock vorangeht, sind für den Disk-Controller wichtige Daten abgelegt. Deren Manipulation kann zu vielfältigen Ergebnissen führen: Vertauschung der Reihenfolge der Sektoren auf einem Track, Blöcke die doppelt mit unterschiedlichen Daten vorhanden sind, illegale Track- und Sektor-Nummern auf legalen Positionen, Vertauschung zweier Tracks miteinander und so weiter...

Die meisten dieser Verfahren führen dazu, daß Teile der Diskette nur noch mit speziellen Programmen gelesen werden können. Die sind dann natürlich, wie oben beschrieben, versteckt.

Nächster Ansatzpunkt sind Halbspuren (Halftracks) und illegale Tracks. Der Schreib-/Lesekopf der 1541 kann in Halbspurschritten bewegt werden. Allerdings ist es nicht möglich, eine Halbspur zu beschreiben, ohne die beiden angrenzenden Tracks teilweise zu löschen. Aber immerhin kann man so, verzichtet man auf die Tracks 12 und 13, den Track 12,5 beschreiben und ihn entweder als Track 12 oder Track 13 verwenden. Einer von beiden geht dabei völlig verloren. Wird beim Lesen von Sektoren dieses Tracks abgefragt, um wieviele Halbspurschritte sich der Kopf bewegen mußte um diesen Track zu erreichen, lassen sich Original und Kopie voneinander unterscheiden.

Illegale Tracks sind die Tracks 36 bis 42. Diese Tracks sind, obwohl offiziell nicht vorhanden, beschreib- und lesbar, allerdings auch nur mit speziellen Programmen und bei Disketten guter Qualität.

Die letzte einfache Schutzmöglichkeit ist das Manipulieren von Lücken. Solche Lücken befinden sich immer zwischen Blockheader und Datenblock und umgekehrt. Man kann die Länge dieser Lücken verändern oder aber Daten in ihnen verstecken.

Der ewige Wettlauf

Da sich alle diese Verfahren aber, wie später noch beschrieben wird, recht einfach kopieren lassen, fahren die Softwarefirmen in letzter Zeit ein sehr schweres Geschütz auf: Fremdformate. Die Datenaufzeichnung auf der 1541 unterliegt vielen Spielregeln. Wenn man diese nicht nur teilweise übertritt, sondern völlig außer Kraft setzt, so erhält man Ansammlungen von Bytes auf der Diskette, die das DOS der 1541 nicht mehr verarbeiten kann. Diese Daten können dann nur noch mit Spezialprogrammen gelesen werden. Eine Möglichkeit eines Fremdformates wäre zum Beispiel die Auflösung der Sektor-Struktur auf einem Track. Hinter einer einzigen Synchronmarkierung befinden sich dann direkt aufeinanderfolgend die Datenbytes. Wenn man hier geschickt arbeitet, kann man alle Kopierversuche mit einfachen Kopierprogrammen unterbinden. Doch haben Programmierer inzwischen auch hier Mittel und Wege zum Kopieren gefunden.

Einen einzigen Kopierschutz gibt es bisher, der garantiert nicht rein softwaremäßig und ohne Umbau der 1541 kopiert werden kann. Dieser Schutz wird bisher nur von Electronic-Arts verwendet. Diese Firma kopiert ihre Disketten mit einer speziellen Kopiermaschine, die einen zu breiten Tonkopf hat, so daß sie immer zwei Tracks gleichzeitig beschreiben kann. Daraus folgt, daß die Tracks 34, 34,5 und 35 völlig identisch und parallel zueinander sind. Beim Lesen von Blöcken des Tracks 34 kann also der Kopf in Halbspurschritten nach innen und wieder zurück nach außen bewegt werden, ohne daß Leseprobleme auftreten. Dieser an sich einfache Schutz ist nicht mit der normalen 1541 kopierbar, da diese ja im eigentlichen Sinne nicht halbspurfähig ist.

Um bei den Firmen zu bleiben: Wir wollten wissen, wie die anderen großen Firmen schützen. Schlechtester Schützer ist augenblicklich Activision, die immer noch mit Errors arbeitet. Broderbund benutzt neuerdings ein einfaches eigenes Format auf den Tracks 36 und 37 gekoppelt mit Killertracks. Epyx arbeitete bisher mit Lücken, stellt aber gerade auf ein neues Verfahren um. Data Becker verwendet gefüllte Lücken und minipulierte Header auf den Tracks 1 bis 40, eine gefährliche Sache, weil schon damit leicht verstellte 1541-Laufwerke nicht mehr zurecht kommen. Datasoft schließlich treibt den größten Aufwand mit einem eigenen ausgeklügelten Format auf fast allen Tracks.

Kopierprogramme, die jeder haben will

Jetzt fragten wir natürlich noch nach den gängigen Kopierverfahren und -programmen.

Beschränkt man sich auf Kopierprogramme, die einen Kopierschutz mitkopieren sollen, gibt es derzeit drei Prinzipien. Dabei kann jedes Verfahren ganz einfach auf Halbspuren und illegale Tracks ausgeweitet werden, so daß wir diese nicht weiter beachten.

Beim ersten Verfahren, dem »Header-Copy«, werden Blockheader und Datenblock, vielleicht noch die Lücken kopiert. Das Verfahren kopiert viele aber nicht alle Fehler und die meisten Header-Manipulationen, aber nicht Killertracks oder gar Fremdformate. Ein Vertreter dieser Gattung ist »Turbo-Nibbler«, ein weiterer »Superclone«. Allerdings ist diese Methode inzwischen veraltet, da die kleinen Details, die diese Kopierprogramme zwangsläufig übersehen müssen, von den Softwarefirmen heutzutage gezielt eingesetzt werden.

Die zweite Gruppe von Kopierprogrammen sind die Synchronmarken-orientierten Programme (Sync-Copy). Hier wird nicht mehr zwischen Blockheader, Datenblock und Lücke unterschieden. Der Blockheader und der Datenblock werden immer von einer Synchronmarkierung, kurz Sync genannt, eingeleitet. Sync-orientierte Programme kopieren nun immer von Sync zu Sync, ohne sich darum zu kümmern, was sie eigentlich kopieren. Mit diesem Verfahren werden alle erzeugbaren Fehler, alle Blockheader- und Lückenmanipulationen und die einfachen Fremdformate kopiert. Dieses Verfahren versagt nur, wenn gar keine Syncs vorhanden sind oder wenn sich mehr als 1000 Byte zwischen zwei Syncs befinden. Dann reicht die Kapazität des Floppy-Pufferspeichers nicht mehr aus, die Daten zwischen den Syncs zu speichern. Bekanntester Vertreter dieser Gruppe ist »Doubble Image«. Einziger Nachteil dieses Verfahrens ist, daß die Disketten mindestens siebenmal gewechselt werden müssen, weil hier viel mehr Daten gelesen und geschrieben werden als normalerweise üblich.

Der Sieger steht noch nicht fest

Das letzte Verfahren ist noch nicht ausgereift, es handelt sich um »Burst-Copys«. Beim Burst-Verfahren wird auf keine Markierung auf der Diskette mehr Rücksicht genommen, ein Track wird während einer einzigen Umdrehung komplett gelesen und geschrieben, egal wie sein Inhalt aussieht. Damit würde praktisch alles kopiert werden, bis auf den Electronic-Arts-Schutz. Im Augenblick gibt es noch kein voll funktionsfähiges Burst-Copy, da die 1541 eigentlich nicht dafür ausgelegt ist. Die eine Alternative wäre, mit einem System zu arbeiten, das den seriellen Bus extrem beschleunigt. Die bisherigen Vertreter, Turbo Access und SpeedDos, sind hier hart an der Grenze der benötigten Geschwindigkeit. Die andere Möglichkeit ist eine RAM-Erweiterung der 1541 um mindestens 10 KByte, in der ein Track komplett zwischengespeichert werden kann, bevor er über den seriellen Bus geht.

Um auch die letzte Bastion, den Electronic-Arts-Schutz, zu kopieren, müßte man dann noch die Mechanik der 1541 gegen die eines 80-Spur-Laufwerkes tauschen oder aber das Indexloch, das auf der 1541 nicht verwendet wird, mit einer Lichtschranke nachrüsten, um parallele Halbspurformatierungen zu ermöglichen. Mit einem solchen hochgepäppeltem Laufwerk wäre wohl kein Kopierschutz mehr sicher.

Der Wettlauf zwischen Schutz- und Knackprogrammen geht also weiter. Die Frage, ob es wirklich keinen perfekten Programm- oder Kopierschutz gibt, kann man nur mit einem Augenzwinkern beantworten: »Ein schlechtes Programm, das keiner knacken oder kopieren will!«

(B. Schneider/aa)