Systematisches Testen von Anti-Viren-Software

Die Autoren

Andreas Marx
Claus Rautenstrauch

Andreas Marx, BSc.,
Gega IT-Solutions GbR,
Klewitzstr. 7,
39112 Magdeburg,
E-Mail: amarx@gega-it.de;
Prof. Dr. Claus Rautenstrauch,
Otto-von-Guericke-Universität Magdeburg,
Institut für Technische und Betriebliche Informationssysteme,
Postfach 4120,
39016 Magdeburg,
E-Mail: rauten@iti.cs.uni-magdeburg.de

1 Bedrohungs- und Schadenspotenziale

Es vergeht kaum ein Jahr, in dem Computerviren wie Melissa, I-love-you, Code Red oder Nimda nicht für einige Tage Schlagzeilen in den Medien machen. Das *Bedrohungspotenzial*, das sich durch die Anzahl der Virenattacken pro Zeiteinheit äußert, steigt nach wie vor stetig [Mess02], was weniger durch die Produktivität der Virenentwickler, sondern vor allem durch den steigenden E-Mail-Verkehr verursacht wird. So hat sich das Verhältnis zwischen verseuchten zu nicht-verseuchten E-Mails bei etwa 1 : 200 eingependelt. Zur Anzahl der Virenattacken gibt es keine verlässlichen Angaben, aber die von zwei Anbietern von Anti-Viren-Software (AV-Software) permanent aktualisierten Zahlen zeigen, dass beide zusammen etwa 25,5 Mio. verseuchter E-Mails im Jahr 2002 aufgespürt haben [Mess02; GeCA02].

Aus der Bedrohung kann Schaden werden, wenn es einem Virus gelingt, einen Computer zu befallen („auszubrechen"). Im *ICSA Labs Virus Prevalence Survey 2001* [BrTi01] ist dokumentiert, dass in einer Untersuchungsgruppe von 300 Firmen mit je mehr als 500 Arbeitsplatzrechnern insgesamt 1.182.634 Virenvorfälle der 666.327 untersuchten PCs im Laufe von 20 Monaten (Januar 2000 bis August 2001) festgestellt wurden, d. h. 89 Vorfälle pro 1.000 PCs im Monat. Die durchschnittlichen direkten Kosten für die Schadensbeseitigung eines Virenvorfalls liegen demnach im Durchschnitt bei US-$ 69.000 mit einem Median von US-$ 5.500. Nimmt man noch die indirekten Kosten (Ausfälle in der Produktion, entgangene Aufträge etc.) mit hinzu, so werden die Gesamtkosten zwischen US-$ 50.000 und 500.000 pro Jahr für Virenunfälle pro Firma in der Untersuchungsgruppe hochgerechnet. Die Gesamtschadensbilanz pro Virus zeigt ähnliche Dimensionen. Nach einer Untersuchung von Computer Economics [CEI02] hat allein der I-love-you-Virus im Jahr 2000 einen weltweiten Schaden von 8,75 Mrd. US-$ verursacht. Bemerkenswert ist, dass der im Jahr 2001 aufgetretene und nicht weniger zerstörerisch und vermehrungsfreudig wirkende Nimda-Virus „nur" einen Schaden von 0,59 Mrd. US-$ verursacht haben soll. Offensichtlich hat I-love-you bewirkt, dass zunehmend *AV-Software* eingesetzt wird und so zumindest zu einer deutlichen Schadensbegrenzung beigetragen. Dies zeigt sich auch bei einer globalen Betrachtung von Bedrohungs- und Schadenspotenzialen. Trotz steigendem Bedrohungspotenzial war der durch Viren verursachte Gesamtschaden im Jahr 2001 erstmals rückläufig. Er betrug in 2001 13,2 Mrd. US-$ im Vergleich zu 17,2 Mrd. US-$ in 2000. Die *Schadenspotenziale* eines einzelnen Virus liegen damit im Multi-Millionen-Dollar-Bereich. AV-Software trägt damit offensichtlich flächendeckend dazu bei, dass das Bedrohungspotenzial in wesentlich geringerem Maße in ein Schadenspotenzial umgesetzt wird.

In der Wirtschaftsinformatik wird AV-Software eher selten diskutiert, obwohl die in Wirtschaftsunternehmen eingesetzten Informations- und Kommunikationssysteme nicht nur Hauptverbreitungsmedien, sondern auch Hauptangriffsziele solcher

Attacken sind. Integrierte Anwendungen, etwa Officepakete – allen voran vom besonders als Angriffsziel beliebten Weltmarktführer Microsoft –, bieten mit ihren vielen, schwer zu überschauenden und gleichzeitig einfach handhabbaren Funktionen ideale Bedingungen für die Schädlinge, da sie mehr auf Benutzerkomfort als auf Sicherheit ausgelegt sind. Die oben zitierten Zahlen zeigen zudem, dass AV-Software eine signifikante wirtschaftliche Bedeutung hat. Die Frage, welche AV-Software in einem Unternehmen einzusetzen ist, besitzt strategische Bedeutung, da potenziell alle Arbeitsplätze im Unternehmen betroffen sein können und eine langfristige und zuverlässige Absicherung der Informationssysteme notwendig ist.

Der Einsatz von AV-Software ist für Unternehmen damit ein notwendiges Übel, das Unternehmen jeder Branche und jeder Größe betrifft. Die oben skizzierten Zahlen zeigen, dass die Nullalternative, d. h. der ungeschützte Betrieb einer internetkonnektierten Informationsinfrastruktur, für verantwortungsbewusste Unternehmen nicht infrage kommt. Auf der anderen Seite ist AV-Software nicht direkt an Prozessen der betrieblichen Leistungserstellung beteiligt, sodass sich keine Kosten-Nutzen-Relation aus der Verbesserung der Geschäftsprozesse herleiten lässt. Die Kosten für AV-Software sind nicht vernachlässigbar: Neben den Lizenzkosten, die in der Regel mit der Anzahl Arbeitsplatzrechner korrelieren, schlagen insbesondere die Wartungskosten zu Buche, da die Schlagkräftigkeit der Software wesentlich von den Releaseständen abhängt. Veraltete AV-Software kann nicht vor aktuellen Viren schützen und die Releasezyklen hängen direkt von der ungebrochenen Kreativität der Virenprogrammierer ab. Bei der Auswahl von AV-Software steht das betriebliche Informationsmanagement in der Pflicht, sowohl leistungsfähige und zuverlässige wie auch kostengünstige und mit wenig Aufwand wartbare Systeme auszuwählen. Systematische AV-Softwaretests leisten einen signifikanten Beitrag zur Unterstützung der Auswahlentscheidung.

Der Test von AV-Software erfordert Spezialkenntnisse, die oftmals selbst Fachabteilungen von Unternehmen überfordern, da Methoden des konventionellen Software-Qualitätsmanagements zu kurz greifen. Unabhängige AV-Softwaretester haben daher die Rolle, regelmäßig AV-Software zu prüfen und die Ergebnisse zu publizieren. Im Folgenden wird nach einer kurzen Einführung in die Grundlagen von AV-Software gezeigt, wie derartige Tests durchgeführt werden. Die Autoren befassen sich mit dem regelmäßigen Testen von AV-Software und den methodischen Grundlagen seit mehr als sechs Jahren. Sie stehen in permanentem Kontakt mit den Herstellern von AV-Software und haben die Testergebnisse in namhaften Fachzeitschriften auch international publiziert.

2 Anti-Viren-Software – Anforderungen und Funktionsweise

Wenn man die am Markt verfügbare AV-Software zum Ausgangspunkt einer weitergehenden Analyse heranzieht, ist zunächst festzustellen, dass der Begriff Virus zu kurz greift, da AV-Software nicht nur Viren, sondern auch anderen Formen schädigender Software erkennen und beseitigen kann. Als Oberbegriff für schädigende Software hat sich *Malware* (*malicious software*) etabliert [Ford02]. Zu Malware gehören neben Viren Würmer und Trojanische Pferde. Weiterhin lassen sich einige Grenzfälle sowie Pseudo-Malware unterscheiden (siehe Bild 1).

Am Markt ist eine Vielzahl von AV-Software verfügbar. Für Nutzer sind die Anforderungen an diese kaum durchschaubar. Aus seiner Sicht muss AV-Software einfach installierbar und handhabbar sein sowie zuverlässig mittels eines *Scanners* Viren erkennen (*detection*) und beseitigen (*disinfection*). Prozesse, die im Hintergrund neu übertragene Dateien mit dem Scanner überprüfen, werden *Wächter* genannt.

Oft werden AV-Programme als relativ einfache Datenbank mit vielen Signaturen, d. h. typischen Codesequenzen von Viren („Fingerabdrücke“), dargestellt, nach denen das Programm sucht. Auch wenn diese Grundidee heute noch zutrifft, handelt es sich mittlerweile um komplexe Software, die vielen Anforderungen gerecht werden muss.

Malware-Arten lassen sich nach folgendem Schema untergliedern: Als *Viren* bezeichnet man Programme, welche die Eigenschaft zur Vermehrung ohne Hilfe von außen besitzen, indem sie Kopien von sich selbst erstellen. Hierfür infizieren sie andere Programme und nisten sich dort so ein, dass sie beim Start der Anwendung ebenfalls aktiv werden können [Nai02]. Viren lassen sich wiederum unterscheiden in:

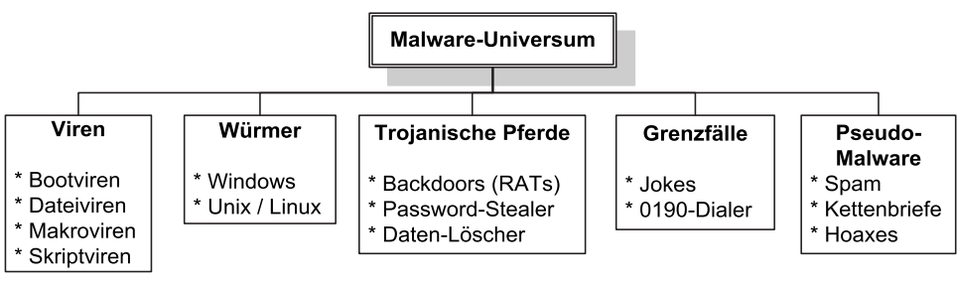

Bild 1 Klassifizierung von Malware-Arten

- Bootviren, die (Boot-) Systembereiche von Datenträgern befallen,
- Dateiviren (auch Binärviren genannt), die ausführbare Programmdateien befallen,
- Makroviren, die sich als Makros innerhalb von Dokumenten und Tabellen von Standard-Office-Anwendungen wie MS-Word oder MS-Excel einnisten,
- Skriptviren, die in oftmals betriebssystemnahen Skripten ausgeführt werden.

Würmer hingegen sind eigenständige Programme, die sich über lokale Netzwerke oder das Internet (etwa per E-Mail oder offene Laufwerksfreigaben) verbreiten, jedoch keine anderen Programme infizieren. Viren benötigen einen Wirt, Würmer hingegen nicht. Sie verbreiten sich typischerweise nur auf einem Betriebssystem, und zwar entweder – wie in den meisten Fällen – Windows oder verschiedenen Unix-Derivaten wie Linux. Daher werden sie anhand ihres Lebensraums (also: Betriebssystemumgebung) unterschieden.

Trojanische Pferde – oft auch kurz Trojaner genannt – verbreiten sich nicht selbstständig, beinhalten aber eine Schadfunktion. Meist sind sie als nützliche Programme getarnt, die als Haupt- oder Nebenfunktion einer destruktiven Tätigkeit nachgehen [Whal98]. *Hintertürenprogramme*, auch *Backdoors* oder *RATs* (*remote administration tools*) genannt, ermöglichen die Fernsteuerung eines Rechners ohne Wissen des Anwenders, um beispielsweise Daten zu stehlen oder zu manipulieren. Aus betrieblicher Sicht sind dies Werkzeuge zur Industriespionage. Bekannte Vertreter dieser Spezies sind z. B. *Back Orifice*, *Sub Seven* und *Netbus*, die vielen Anwendern unbemerkt untergeschoben werden und das System für Dritte öffnen. Aber auch Standardprogramme zur Fernadministration wie *PC Anywhere* oder *VNC* erlauben solche Aktivitäten. Einige Backdoors bieten sogar erheblich mehr Funktionen als kommerzielle Software wie z. B. eingebaute Skriptsprachen, „Nervfunktionen“ zum Ein- und Ausfahren des CD-Laufwerkes, Optionen zum Rechnerneustart, Aufzeichnungsoptionen (für Tastaturanschläge) usw. Die Programmierer dieser Hintertüren sehen in der Klassifizierung ihrer Software als „durch AV-Software erkennbare Malware“ eine Geschäftsschädigung [Kula01]. Daher kann man nicht immer eindeutig von Malware sprechen, da man die an sich schädlichen Programme auch produktiv einsetzen kann bzw. die kommerzielle Software auch als Malware einsetzbar ist. Ein Spezialfall der Trojaner sind Password-Stealer, die Passwörter durch direktes Auslesen von Dateien oder „Mithören“ von Benutzereingaben stehlen. Ebenso erwähnenswert sind Trojaner, die gezielt oder willkürlich Dateien löschen oder die Festplatte formatieren.

Die eindeutige Zuordnung eines Schadprogramms in eine der Kategorien Viren, Würmer und Trojanische Pferde ist heutzutage nicht immer möglich. Viele Malware-Produkte verfügen über multiple Ausbreitungs- und Schadmechanismen, die nicht eindeutig einer Klasse zuzuordnen sind. So breitet sich etwa *MTX* als Virus und Wurm aus und verfügt über eine Backdoor [Nai00]. Solche Mischformen treten immer häufiger auf.

Zu den Grenzfällen gehören *Scherzprogramme* (*Jokes*) und 0190-Dialer. Einige Jokes werden von AV-Software entdeckt, d. h., dass einige AV-Programme einige Jokes erkennen, aber kein Programm wirklich viele Jokes zuverlässig erkennt. Jokes sind nur im entfernten Sinne Malware, da sie keine vorsätzlichen Datenschäden verursachen. Vielmehr kommt es auf die Reaktion des Anwenders an: So gibt es Jokes, die in bestimmten Zeitabständen Fehlermeldungen anzeigen oder den Bildschirm auf den Kopf stellen und so erhebliche Verunsicherungen und darauf folgende panikartige Reaktionen (z. B. das Ausschalten des Rechners ohne vorherige Sicherung der gerade bearbeiteten Daten) hervorrufen können.

Bei *(0190-)Dialern* ist noch unklar, ob es sich ebenfalls um Malware handelt [Marx02d]. Eigentlich ist die Abrechnung von Kleinstbeträgen über die Telefonrechnung eine legitime Zahlungsmethode, die insbesondere von Erotikseitenbetreibern genutzt wird. Viele Dialer-Anbieter missachten aber den *Verhaltenskodex für Telefonmehrwertdienste* [Fst01]. Sie deaktivieren etwa Dialer-Warnprogramme [Tiet02] oder versuchen, diese durch verschiedene Tricks zu umgehen [Boro02], sodass einige AV-Hersteller diese Produkte in ihren Datenbanken aufgenommen haben [Siem02]. Dies währte allerdings nicht sehr lange, da die entsprechenden AV-Anbieter wegen Geschäftsschädigung abgemahnt wurden.

Zwar bezieht sich der Begriff Malware ausdrücklich auf ausführbare Software, aber oft werden unter dem Begriff Pseudo-Malware auch *Hoaxes* (Falschmeldungen über angebliche Viren), *Kettenbriefe* und *Spams* (unerwünschte Werbesendungen) der Malware zugeordnet, wenngleich es sich beispielsweise um reine (E-Mail-) Texte handelt [Ziem02]. Die Kosten von Hoaxes können sehr schnell explodieren, wenn nur ein paar Mitarbeiter Mails bekommen, lesen, diese firmenintern und extern weiterleiten und vielleicht noch Dateien vom System löschen, wie es oft in den Falschmeldungen als Anleitung zu lesen ist [Fuhs98]. Von AV-Software werden sie nicht erkannt, da es sich nicht um ausführbaren Code handelt, sondern vielmehr um reinen Text bzw. HTML-Seiten. Schutz bietet hier Content-Security-Software, die E-Mails auf Gateway- oder Groupware-Systemen nach bestimmten Texten filtern kann.

Kernpunkte für das Management

Anti-Viren-Software (AV-Software) gewinnt auf Grund der enormen Schäden, die verschiedene Arten von Malware verursachen, zunehmende Bedeutung für Unternehmen. Es wird ein systematisches Testverfahren für AV-Software vorgestellt, mit dem die Leistungsfähigkeit verschiedener Systeme hinsichtlich Virenerkennung und -beseitigung geprüft werden kann.

- Viren und andere Schädlinge haben nach wie vor enorme Schadenspotenziale. AV-Software ist wirksamer Schutz auch vor erheblichen finanziellen Schäden.
- Systematische Testprozeduren unterstützen die Auswahl aus verschiedenen Systemen nach unterschiedlichen Leistungskriterien.

Stichworte: Anti-Viren-Software, AV-Software, Malware, Sicherheit, Testverfahren

Bezüglich des Erscheinungsbilds von Malware sind *unverschlüsselte*, *verschlüsselte*, *polymorph (vielgestaltig) verschlüsselte* und *gepackte (laufzeitkomprimierte)* Formen unterscheidbar:

- Unverschlüsselte Malware kann durch einfachen Vergleich der Codesequenz mit den in der AV-Datenbank gespeicherten Signaturen erkannt werden.
- Bei verschlüsselten Viren ist es erforderlich, das Virus zunächst zu entschlüsseln, um einen solchen Vergleich durchzuführen. Meistens ist dies relativ einfach, da der Programmcode, der das Virus letztlich entschlüsselt, selbst nicht verschlüsselt und konstant ist, sodass man das Virus oft bereits daran eindeutig erkennen kann.
- Bei polymorpher Malware ist die ist Entschlüsselungsroutine nicht mehr statisch, sondern wird zur Laufzeit durch eine Codesequenz erzeugt. Man kann den Virus nun nicht mehr am Entschlüsselungscode erkennen, sondern es muss vielmehr versucht werden, Schritt für Schritt per *Code-Emulation* auf einer virtuellen CPU das Verschlüsselungsverfahren nachzuvollziehen. Ist der Virus erst einmal mit Hilfe der Code-Emulation in einer simulierten Umgebung entschlüsselt worden, kann er ebenfalls wieder einfach per Signaturvergleich erkannt werden.
- Bei laufzeitkomprimierter Malware werden Verfahren genutzt, die ein komprimiertes Programm weiterhin direkt ausführbar lassen, aber deutlich weniger Speicherplatz beanspruchen. Wenn ein Virus in einer solchen Datei enthalten ist, so wird er mit komprimiert und ist somit bis zum Entpacken nicht anhand seiner Signatur erkennbar.

Bei verschlüsselter, polymorpher und laufzeitkomprimierter Malware kann man per Code-Emulation auch das Verhalten eines potenziell verdächtigen Programms näher untersuchen und bei Verdachtsbestätigung Alarm auslösen [Nach98].

Weiterhin wird verlangt, dass die Software in verschiedensten Dateiformaten nach Malware suchen kann, d. h., dass auch Viren, die in OLE-Objekten, PDF-Dokumenten oder ARJ- bzw. ZIP-Archiven versteckt sind, entdeckt werden müssen. Eine weitere Anforderung ist Plattformunabhängigkeit bezogen auf verschiedene Betriebssysteme sowie Prozessor- und Hardware-Architekturen. Die meisten AV-Programme sind daher in ANSI-C geschrieben. Assemblercodes werden nur noch selten verwendet.

3 AV-Software-Tests im Detail

AV-Software wird mit dem Ziel getestet, Leistungen und Schwächen der Programme zu ermitteln sowie Verbesserungen und Weiterentwicklungen der Produkte zu dokumentieren, um Systemadministratoren entscheidungsunterstützende Informationen für die Auswahl von Produkten zu liefern. Weiterhin werden die Testergebnisse auch in Zeitschriften als Testberichte veröffentlicht.

Wesentliche Grundlage für die Tests ist eine große Malware-Datenbank mit infizierten Dateien (die so genannte *Virenkollektion*), die ständig mit neuen Viren aktualisiert wird. Weil es in der Praxis sowohl extrem schwierig ist, eine vollständige Virensammlung zu bekommen, als auch nachzuweisen, dass es sich bei dem gesammelten Material wirklich um Viren handelt, sind nur wenige locker miteinander verbundene Einrichtungen weltweit in der Lage, solche Tests außerhalb der Laboratorien der AV-Softwarehersteller durchzuführen. Dazu gehören die ICSA (International Computer Security Association, USA; www.icsalabs.com), die Westcoast Labs (England; www.check-mark.com), das Virus Bulletin (England; www.virusbtn.com), die Universität Hamburg (Deutschland; http://agn-www.informatik.uni-hamburg.de) und die Universität Magdeburg (Deutschland; www.av-test.org).

Im Anti-Viren-Bereich sind die Release-Zyklen von Updates extrem schnell (stündlich bis spätestens wöchentlich), da ständig neue Viren freigesetzt werden. Die Wahrscheinlichkeit dafür, dass durch den Einsatz veralteter Releases in einem Unternehmen Schaden entsteht, ist bei AV-Software deutlich größer ist als bei anderen Systemen. Anwender können daher trotz Schutzprogrammen nicht sicher darauf vertrauen, ein virenfreies System zu haben, da laufend neue Schädlinge auftreten, die sich Dank Internet immer schneller verbreiten können. Während die Hersteller von AV-Software noch an Updates arbeiten, können Rechner bereits leicht anderweitig infiziert sein. Diese Eigenschaft ist auch bei der Bewertung der Testergebnisse zu berücksichtigen, die nur Schnappschüsse der Situation zu bestimmten Zeitpunkten sein können.

Bei AV-Softwaretests lassen sich die Phasen Testvorbereitung, -durchführung und -dokumentation unterscheiden [Marx00]. Im Folgenden werden diese im Detail vorgestellt. Der Schwerpunkt liegt dabei auf den speziellen Abläufen und Rahmenbedingungen von AV-Software.

3.1 Testvorbereitung

Projektplan

Ein AV-Softwaretest beginnt mit dem Erstellen eines Projektplans, der Testziele, Rahmenbedingungen und eine Abschätzung der benötigten Ressourcen wie Budget, Personal sowie Hard- und Software enthält. Die zeitliche Abschätzung ist insbesondere bei den ersten Tests problematisch. Bei späteren Tests ist dies einfacher, da z. B. die Virensammlung nicht neu aufgebaut werden muss und man im Laufe der Zeit Erfahrungen sammelt, die den Testprozess durch Vorkenntnisse zu Stärken und Schwächen von Produkten sowie über die Automatisierung von Testprozeduren beschleunigen.

Testkriterien

Der zweite Schritt ist die Festlegung der Testkriterien, anhand derer die Programme verglichen werden. Neben den speziellen Kriterien wie Erkennung (*detection*) und Beseitigung (*disinfection*) müssen auch allgemeine Dienstleistungen der Softwareanbieter (Unterstützung bei Viren-Problemen bzw. akutem Befall, (Schnelligkeit des) Update-Services des Anbieters, Erreichbarkeit usw.) berücksichtigt werden. In einem Scoringmodell wird die Gewichtung der einzelnen Testkriterien als Grundlage für eine spätere Gesamtbewertung der Testkandidaten festgelegt. Für die Festlegung der Gewichtung empfiehlt es sich, schon im Vorfeld des Tests einzelne Programme näher anzuschauen und deren positive bzw. negative Leistungsmerkmale festzuhalten sowie Nutzerbefragungen durchzuführen.

Da AV-Softwarehersteller ihre Sourcecodes nicht offen legen (einzige bislang bekannte Ausnahme: OpenAntiVirus [Open02]), sind die Testfälle auf *Blackbox-Tests* eingeschränkt [PoBl96, 151f.]. Für derartige Tests kommen folgende Kriterien infrage [Wall90, 194f.]:

- *Vollständigkeit* ist gegeben, wenn alle wichtigen Komponenten wie Scanner und Wächter sowie Notfalldatenträger und Internet-Update-Funktionen enthalten sind.
- Ein System ist *robust*, wenn es bei *Massen-* bzw. *Belastungstests* viele verschiedene infizierte und nicht-infizierte Dateien fehlerfrei in großen Verzeichnisstrukturen scannen kann, auch wenn diese mitunter beschädigt sind [Marx02a], und wenn die angebotenen Maßnahmen für gefundene Viren wie Löschen, Umbenennen oder in Quarantäne verschieben ordnungsgemäß arbeiten. Weiterhin ist ein System *robust,* wenn es unter *hoher Arbeitslast* auf einem Server mit vielen angeschlossenen Clients, die parallel darauf zugreifen, ordnungsgemäß und zuverlässig arbeitet.
- *Effizienz* bedeutet bezogen auf den Scanner, dass er nicht zu viele *Ressourcen* belegt, da der Wächter bei vielen Dateioperationen das zu überwachende System zur Laufzeit bremsen würde.
- Darüber hinaus verlangt der Benutzer *Kompatibilität* mit der weiteren verwendeten Hard- und Software, insbesondere den verschiedenen Betriebssystemversionen. Dies muss gewährleistet sein, da die AV-Software sonst die Systemstabilität beeinträchtigt und dann möglicherweise sogar vom Anwender deaktiviert wird.

Daneben sind auch konventionelle Softwarequalitätsmerkmale einzubeziehen. Dies betrifft z. B. die *Benutzerfreundlichkeit* des Programms, hier insbesondere auch die Genauigkeit des Feedbacks, wenn ein Schädling entdeckt wird. Ebenso ist die *Sicherheit* zu berücksichtigen. Eine AV-Software benötigt grundsätzlich Administratorrechte, wodurch sich bei unsachgemäßer Handhabung Lücken öffnen könnten, z. B. wenn temporäre Dateien in ungeschützten Verzeichnissen angelegt werden. Daher muss sichergestellt werden, dass nur Administratoren verschiedene Programmeinstellungen ändern können. Ferner sind auch die *Installations-* und *Deinstallationsprozeduren* dahingehend zu prüfen, ob z. B. Neustarts nötig sind (was auf Servern kritisch sein kann), welche Dateien und *registry keys* sie ändern und ob diese Modifikationen auch wieder rückgängig gemacht werden können.

Aufbau der Testumgebung

In der Praxis gestaltet sich der Aufbau der Malware-Kollektion, die Viren, Würmer, Trojanische Pferde und andere Bedrohungen enthält, oftmals als schwierig [Marx02b, 11ff]. Hierfür ist zunächst ein abgesichertes System empfehlenswert, das – wie auch das eigentliche Testnetzwerk – aus Sicherheitsgründen nicht mit dem Internet oder anderen Produktivnetzwerken verbunden ist. Der Zugang zur Testumgebung muss weiterhin durch bauliche (Zutritts- und Zugangsberechtigungen) und organisatorische Maßnahmen eingeschränkt sein. Online-Verschlüsselung aller Daten und regelmäßige verschlüsselte Backups sind obligatorisch, schließlich handelt es sich bei Malware um gefährliche Programme, die, wenn sie in falsche Hände geraten, großen Schaden anrichten können. Für den Aufbau einer initialen Malware-Kollektion gibt es prinzipiell folgende Wege:

- Man greift auf vorhandene Testergebnisse von anerkannten Organisationen zurück und ergänzt die Ergebnisse auf Basis eigener Tests mit wenigen aktuellen Viren. Ein Indikator für Häufigkeit und Aktualität von Viren und Würmern ist die monatlich aktualisierte WildList [Well02]. Etwa 70 weltweit agierende Reporter sammeln hierfür validierte Infektionsmeldungen von befallenden Rechnern. Viren, die in der WildList aufgeführt werden, bezeichnet man als *ITW*-Viren (In-the-Wild), also Viren, die in *freier Wildbahn* vorkommen und somit häufig bei Anwendern anzutreffen sind.
- Virensammlungen werden von Herstellern der Schutzprogramme übernommen. Allerdings wird kaum ein Anbieter seine Virensammlung an einen Unbekannten herausgeben und wenn doch, so wäre es einseitig, die Programme anderer Hersteller gegen eine solche Sammlung antreten zu lassen. Welcher Entwickler gibt schon Viren heraus, die sein Programm nicht erkennt? Erst eine Vielzahl an Herstellersammlungen kann dieses Risiko mindern.
- Virensimulatoren erzeugen Dateien, die Bruchstücke bekannter Viren an nicht relevanten Stellen enthalten. Solche „Testviren“ sind definitionsgemäß keine Viren, sondern Programme, die eine Meldung ausgeben und sich selbst beenden. Werden sie trotzdem erkannt, ist dies ein Fehlalarm. Daher ist von solchen Testviren Abstand zu nehmen. Auch die Idee, selbst Viren zu schreiben oder sich mittels *virus construction kits* (VCKs, „Viren-Baukästen“) artifizielle Viren zusammenzuklicken, ist nicht angebracht, da schon genug Viren existieren und die Konstruktionsmuster solcher Viren den AV-Herstellern bekannt sind.

Zumindest in der Anfangsphase muss daher auf Virensammlungen zurückgegriffen werden, die auf verschiedenen einschlägig bekannten Internetseiten zu finden sind. Diese sind aber meist in schlechtem Zustand und beinhalten viele harmlose Dateien, etwa Anleitungen oder Beschreibungen zu Viren. Aber auch bereits (schlecht) desinfizierte Viren, zerstörte Programme, Archivdateien und normale (d. h. nicht-virale) Anwendungen sind hier oft zu finden. Bei solchen Sammlungen beinhalten häufig 10–20 % „Virenschrott“, was eine aufwändige Reinigungsphase erfordert.

Nach Übernahme und Reinigung der Sammlungen sind zunächst alle Archive zu entpacken und die Dateien nach bestimmten Merkmalen zu gruppieren, etwa nach Textdateien, Win32- und DOS-Anwendungen sowie andere Dokumente. Dabei sind alle mehrfach vorhandenen Dateien auszusortieren. Die Vorsortierung wird anschließend immer weiter nach Vireneigenschaften bis letztlich zum einzelnen Virus untergliedert [Bont93].

Danach werden alle Viren in einer speziell gesicherten Umgebung, wo sich die Viren nach einer Aktivierung schadlos ausbreiten können, wie z. B. Vmware, einer virtuellen Maschine, die einen PC emuliert, repliziert, um festzustellen, ob es sich wirklich um Viren handelt und um infizierte Testdateien zu erstellen, die in der Ausgangsdatei nicht enthalten sind. Nur diese replizierten Samples dürfen in einem Test verwendet werden, da ansonsten Mogeln (*cheating*) von Herstellern insbesondere bei polymorphen Viren nicht ausgeschlossen werden kann. Solche Viren können zwar in einer vom Hersteller weitergegebenen Datei gefunden werden, indem dieser diese (eine) zufällig generierte Signatur des Virus kennt, in Samples aber nur dann, wenn der Erkennungsmechanismus wirklich die Polymorphieeigenschaft berücksichtigt.

Eine Virenkollektion umfasst leicht einige Tausend Dateien, sodass sie sinnvoll strukturiert werden muss. Sie ist die Grundlage für die Selektion von *Testsets*, d. h. eine Menge verseuchter Dateien, die für einzelne Tests herangezogen werden. Als Strukturierungsschema bietet sich der „Lebens-

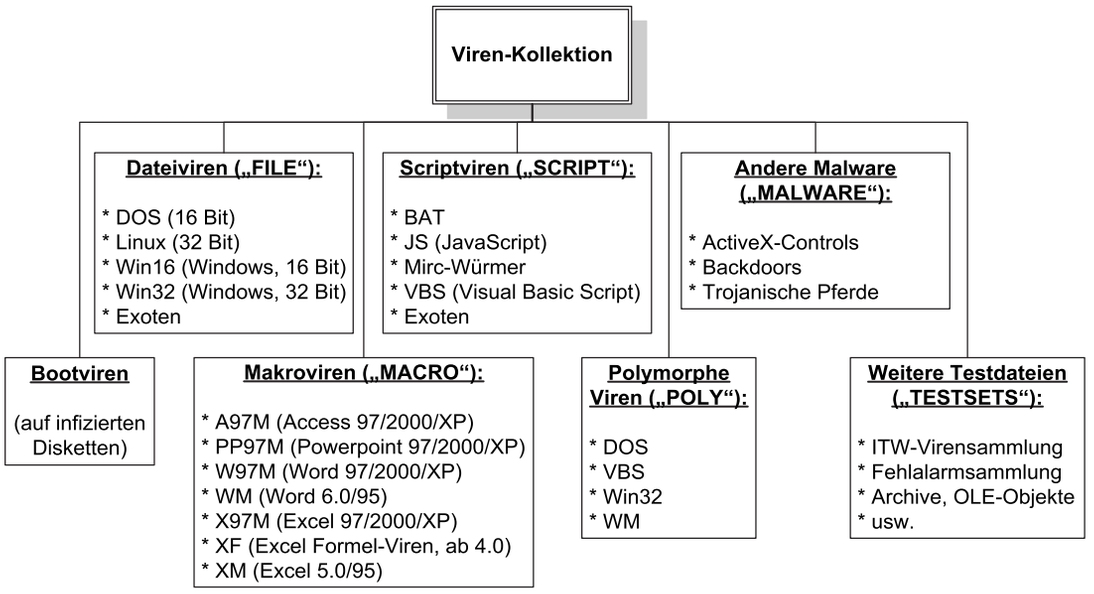

Bild 2 Aufbau der Virensammlung von AV-Test.org

raum“ der Viren an. Bild 2 zeigt die aktuelle Verzeichnisstruktur der Virensammlung von AV-Test.org. Würmer befinden sich jeweils in den entsprechenden Verzeichnissen der Virenarten, da sie sich nur in ihrem Ausbreitungs-, aber nicht in ihrem Schädigungsverhalten von Viren unterscheiden.

Bootviren nisten sich in Systembereichen wie dem *Bootsektor* oder dem *master boot record* (MBR) von Disketten und Festplatten ein. Daher werden sie auf separaten Datenträgern aufbewahrt. *Dateiviren* infizieren 16- oder 32-Bit-Programme verschiedener Betriebssysteme. Innerhalb der Kategorien werden die Viren alphabetisch sortiert in separaten Unterverzeichnissen abgelegt. Die „klassischen“ *Makroviren* befallen die Microsoft-Office-Anwendungen Word, Excel, Powerpoint und Access in den verschiedenen Versionen. Einzelne Exemplare verseuchen auch Dateien von Ami Pro oder MS-Project. *Skriptviren* schließlich sind oft als BAT-Dateien, JavaScript-Code (JS) oder in Visual Basic Script (VBS) implementiert. Viele Werkzeuge im Kontext *internet relay chat* (IRC) besitzen umfangreichen Skriptsprachen, mit deren Hilfe man oft mit nur einer Programmzeile schon einen Wurm erstellen kann. Insbesondere das weit verbreitete Chat-Programm „Mirc“ tut sich hier hervor.

Im „Poly“-Verzeichnis sind mehrere zehntausend infizierte Dateien von polymorph verschlüsselten Viren zu finden. Ebenso enthält die Malware-Sammlung Tausende von Trojanischen Pferden, gefährlichen ActiveX-Controls und Backdoor-Programmen im Bereich „andere Malware“. Weiterhin sind verschiedene Testdateien vorhanden (dazu später mehr).

Auch hier gibt es wieder Malware, die mehrere der oben genannten Lebensräume befallen kann. So sind einige Dateiviren wie *One_Half* bekannt, die auch die Systembereiche von Datenträgern befallen. Sie werden als *Hybridviren* bezeichnet und infizierte Dateien sind im „File“-Bereich eingeordnet, während infizierte Bootbereiche auf Disketten gespeichert werden.

Viren können in den unterschiedlichsten Arten von Dateien versteckt sein und auf ihre Aktivierung warten – etwa in Archiven (ARJ, LHA, RAR, ZIP), in einem laufzeitkomprimierten Programm (Online-Packer), in eingebetteten OLE-Objekten (etwa in einer DOC-Datei, die sich in einer XLS-Datei befindet), in einem kennwortgeschützten Office-Dokument oder in Datenbanken gängiger E-Mail-Anwendungen. Solche Fälle sind in einem Test systematisch zu berücksichtigen.

Zum Abschluss der Phase „Aufbau der Testumgebung“ ist noch eine Dokumentation über die Sammlung zu erstellen. Eine gute Sortierung begünstigt die Erstellung eines Dateibaums mit der jeweils enthaltenen Anzahl Viren sowie eine kurze Gesamtstatistik mit der Zahl der unterschiedlichen Viren bzw. Dateien.

Produktbeschaffung

In den letzten Jahren ist der AV-Software-Markt durch Konsolidierungen übersichtlicher geworden. Klammert man OEM-Versionen aus, die sich meist nur leicht von den originalen Versionen unterscheiden, sind heute etwa 30 marktgängige Programme zu finden, wobei drei Hersteller (Symantec, Network Associates und Trend Micro) marktführend sind. Linksammlungen,

etwa vom Virus Bulletin [Viru02] oder AV-Test.org [AvTe02], sowie weitere Webrecherchen von Lesern, die solche Tests selbst durchführen bzw. nachvollziehen wollen, erleichtern die Suche nach geeigneten Testkandidaten.

Vor dem Test werden per E-Mail oder Fax die Hersteller über den Test informiert und um ein Rezensionsexemplar gebeten. Hierbei sind kurz Motivation und Anlass des Tests, Testkriterien und die Rahmenbedingungen bekannt zu geben. Hierzu gehört, unter welchen Betriebssystemen getestet wird, ob Netzwerk- oder Einzelplatz-Produkte untersucht und welche weiteren Informationen benötigt werden (etwa gedruckte Dokumentationen und Preislisten).

3.2 Testdurchführung

Installation

Ein Test beginnt mit der Installation der Testkandidaten, also der AV-Programme. Danach wird die Software per (automatischem oder manuellem) Internet- oder CD-Update auf den neuesten Stand gebracht. Man trennt die Testrechner anschließend vom Internet, sodass sich versehentlich aktivierte Viren nicht außerhalb der kontrollierten Umgebung weiterverbreiten können. Konfigurationsänderungen an den Programmen sind nachvollziehbar zu dokumentieren, um in der Lage zu sein, bei auftretenden Problemen den Originalzustand wiederherzustellen. Abschließend wird die Installationspartition mit einem Cloning-Programm wie Symantec Ghost oder Powerquest Drive Image auf eine andere Partition gesichert, um die exakte Testumgebung des Programms mit allen eingespielten (Online-)Updates jederzeit wiederherstellen zu können. Dies kann nötig sein, wenn das System aus Versehen verseucht wird, aber auch, wenn man mehrere Programme nacheinander auf dem demselben Rechner testet oder Rückfragen des Herstellers während oder nach dem Test zu beantworten sind.

Online-Virenscanner (z. B. über den Internet Explorer zu installierende ActiveX-Controls) und -Virenscan-Services (etwa E-Mail-Schutzsysteme) bleiben zu diesem Zeitpunkt bei den Tests unberücksichtigt. Bei beiden Systemen ist eine aufgebaute Internet-Verbindung nötig, was bei dem Umgang mit Viren zu gefährlichen Situationen führen kann. Dies kann die versehendliche Infektion eines Systems sein und zu einer Virenausbreitung über das Internet führen. Es besteht aber auch die Gefahr, dass Kopien der verwendeten Dateien beim Anbieter zurückbleiben. Da dieser dann die Testdateien exakt kennt, sind später objektive Tests nicht mehr möglich.

Virenerkennung und Testdateien

Nach den genannten Vorbereitungen beginnt der eigentliche Test der AV-Software mit einer Prüfung der Virenscanner. Zuerst wird die Scanleistung der Programme geprüft, sowohl für die Viren, als auch für andere Testdateien, etwa Archive, die infizierte Dateien enthalten. Für einen Test des Desktop-Virenscanners wird ein Suchlauf über die Virensammlung gestartet (*on-demand scan*). Für einen Test des On-Access-Wächters kopiert man im einfachsten Fall ein Verzeichnis mit infizierten Dateien in einen anderen Ordner und schaut, was die Software – je nach Einstellung – blockiert, umbenennt, desinfiziert oder löscht. Im Falle von Server-Produkten kopiert man infizierte Dateien von dem Server oder auf den Server, wobei auf dem Client kein AV-Schutz installiert sein darf. Bei E-Mail-Schutzlösungen muss man alle infizierten Dateien – am besten skriptgesteuert – per E-Mail zum Scanner schicken.

Um herauszufinden, welche Dateien erkannt bzw. nicht erkannt werden, verwendet man vorzugsweise die Reportfunktionen der Scanner. Diese führen alle infizierten bzw. überprüften Dateien auf. Bislang gibt es hierfür noch keine Standardformate (verwendet werden TXT, CSV, HTML oder DBF), was die Auswertung der verschiedenen Berichte mit den unterschiedlich dargestellten Informationen erschwert. Auf keinen Fall darf die vom Scanner angezeigte Zahl der Infektionen unkritisch weiterverwendet werden, denn jeder Hersteller zählt nach einem anderen Schema (auch doppelt) und so kann es vorkommen, dass mehr Dateien als infiziert erkannt werden, als tatsächlich vorhanden sind.

In zunehmendem Maße sind auch Tests interessant, bei denen nicht-infizierte Dateien – z. B. von Windows selbst, von Office-Paketen und von fast jeder PC-Fachzeitschrift beiliegenden CDs – mit in den Test einbezogen werden. Schlägt hier der Scanner oder Wächter Alarm, obwohl es sich bekanntermaßen nicht um eine Malware-Datei handelt, sollte dieser Fehlalarm negativ im Testbericht berücksichtigt werden.

Desinfektion

Naive Benutzer gehen oftmals davon aus, dass eine verseuchte Datei nach der Reinigung durch einen Virenscanner wieder mit dem Original vor der Infektion identisch wäre. Allerdings zerstören viele Viren Informationen in den befallenden Dateien unwiderruflich. Zudem kann auch der Virenscanner fehlerhaft arbeiten.

Nach einem Suchlauf über die Malware-Kollektion mit der Option „Reinigen“ erhält man, wenn alles gut geht, eine unverseuchte Dateisammlung. Um den genauen Zustand zu prüfen, werden wieder die Reportdateien dahingehend analysiert, welche Dateien desinfiziert werden konnten und ob überhaupt alle Dateien erkannt wurden (quantitativer Zustand).

Für den Qualitätsvergleich werden zunächst andere Virenscanner herangezogen. Dabei wird geprüft, ob die Dateien auch von anderen Virenscannern als verseucht entdeckt werden und ob sich die gereinigten Programme noch starten und benutzen lassen bzw. ob die desinfizierten Makrodateien noch problemlos mit den verschiedenen Office-Versionen zusammenarbeiten. Dabei können Alleinstellungsmerkmale und Wettbewerbsvorteile einzelner Scanner herausgestellt werden.

Zusätzliche Funktionstests

Neben diesen Standardtests sind weitere Untersuchungen interessant. Hierzu gehört beispielsweise, wie der Virenscanner reagiert, wenn das Testsystem bereits infiziert ist: Kann er das System reinigen und wenn ja, wie? Gibt es irgendwo Stolperstellen wie etwa falsche Bezeichnungen in den Menüs sowie umständliche Installation und Wartung? Ist ein Notfall-Rettungssystem vorhanden – kann etwa ein sauberes System vom Installationsdatenträger gestartet werden [Marx02c, 10ff.]? Sind vorweg erstellte Notfalldisketten lauffähig?

Auch die generelle Arbeitsweise auf nicht-infizierten Systemen ist interessant: Gibt es Probleme mit dem Scanner während eines normalen Arbeitstages? Hierzu kann es auch sinnvoll sein, einen Virenscanner für einige Zeit auf nicht-kritischen Produktivsystemen einzusetzen und wie gewohnt zu arbeiten. Oft zeigen sich erst hier kleinere Schwächen oder gar Stabilitätsprobleme,

die im Test auf Grund der kurzen Zeitspanne gar nicht aufgetreten oder aufgefallen sind.

Netzwerk-Funktionen

Während bei einem Einzelplatzprodukt nur selten relevant ist, ob es sich von anderen Rechnern aus fernsteuern und konfigurieren lässt, ist dies bei Client/Server-Lösungen wesentlich für einen Einsatz in einem heterogenen Netzwerk. Hierzu zählen etwa die Remote-Installation von Clients und Servern (wie ist sie möglich, unter welchen Betriebssystemen funktioniert sie), das automatische Einspielen von Updates sowie eine zentrale Überwachung der Systeme (Reportdateien, Viren-Quarantäne).

Diese Funktionen müssen mit unterschiedlichen Einstellungen in verschiedenen Szenarien mit und ohne Viren getestet werden. Insbesondere Grenzfälle können sehr interessant sein, wie z. B. das Verhalten, wenn ein Virus nicht entfernt werden konnte.

Weiterhin gibt es einige interessante Kriterien, die in der Praxis kaum testbar sind. Hierzu zählt z. B. die Reaktionszeit eines Herstellers auf einen akuten Viren-Ausbruch (*outbreak*). Oftmals wartet man auf einen Ausbruch mitunter Monate, dann jedoch müssen sehr schnell Personal und Testsysteme zur Verfügung stehen, um die Webseiten der Hersteller zu überwachen und die Updates innerhalb kurzer Zeit zu testen.

3.3 Ergebnisdarstellung und Nachbereitung

Nachdem alle Daten bezogen auf die Testkriterien gesammelt worden sind, beginnt die Aufbereitung der Ergebnisse zu verwertbaren Informationen mittels Tabellenkalkulation und Textverarbeitung. Aus den Werten in den Tabellen lassen sich über einfache Formeln Kennzahlen wie die sog. Detection- oder Disinfection-Quota (Verhältnis zwischen verseuchten und gefundenen bzw. gereinigten Dateien) berechnen und auch aus statistisch aggregierten Werten Grafiken erstellen. Zur weiteren Dokumentation des Tests gehört eine Testbeschreibung (wie wurde welcher Test mit welchen Mitteln unter welchen Systemen genau durchgeführt), eine Teilnehmerliste mit Kontaktdaten zu den Herstellern, eine Liste mit den zum Test verwendeten Dateien (Virenliste) sowie die üblichen rechtlichen Haftungsausschlüsse und Copyrights.

Danach findet eine abschließende Qualitätskontrolle statt, die nochmals die verschiedenen Details (z. B. begriffliche Konsistenz und Tippfehler) beleuchtet. Vor der Veröffentlichung werden die Ergebnisse kurz nur den Herstellern zur Verfügung stellt. Dies gibt ihnen die Möglichkeit, Unstimmigkeiten zu bereinigen.

Zur Testnachbereitung können auch Updates der Ergebnisse selbst gehören, etwa Fehlerkorrekturen, die dokumentiert werden müssen, aber auch die Diskussion mit den Herstellern und anderen Interessenten. Hier ergeben sich oft Anhaltspunkte für zukünftige Testkriterien oder Verbesserungsmöglichkeiten für bestehende Abläufe. In einigen Fällen ist es sinnvoll, nicht mehr aktuelle oder überholte Testkriterien zukünftig entfallen zu lassen.

4 Fazit

Viren und andere Malware zählen zu den wichtigen Themen in der Computerbranche. In der Wirtschaftsinformatik werden sie aber zu unrecht am Rande behandelt, obwohl sie für alle IuK-Systeme kritisch sind und hohe Schäden verursachen können. Daher ist es unumgänglich, an allen Angriffspunkten Anti-Viren-Lösungen einzusetzen. AV-Tests können eine wesentliche Entscheidungshilfe bei der AV-Softwareauswahl leisten.

In diesem Beitrag wurde gezeigt, dass solche Tests sehr aufwändig sind und langjährige Erfahrung sowie einen sicheren und vertrauensvollen Zugang zur „Szene“ voraussetzen. Unternehmen sind daher in den seltensten Fällen in der Lage, solche Tests selbst in der gezeigten Form durchzuführen. Auf Grund der genannten Rahmenbedingungen kann kaum empfohlen werden, eigene Testlabors einzurichten. Der Rückgriff auf Testergebnisse unabhängiger AV-Tester mit unzweifelhafter Reputation, die regelmäßig und neutral AV-Software testen und die Ergebnisse publizieren, ermöglicht dagegen die zuverlässige Informationsgewinnung für Unternehmen zur erfolgreichen Implementierung einer abgesicherten Virenabwehr.

In diesem Beitrag wurden Testmethodik und Hintergründe von AV-Tests offengelegt, um die praktische und wissenschaftliche Relevanz des Themas nahe zu bringen und die Aufmerksamkeit auf diese Thematik zu lenken. AV-Tests sind der Schlüssel zur sinnvollen AV-Softwareauswahl – und damit auch zur Verhinderung von Millionenschäden in Unternehmen.

Literatur

[AvTe02] *AV-Test.org:* Anti-Virus Links. http://www.av-test.org/sites/links.php3?lang=en&extra=viren&sort=1, Abruf am 2002-11-01.

[Bont93] *Bontchev, V.:* Analysis and Maintenance of a Clean Virus Library. http://www.virusbtn.com/OtherPapers/VirLib, Abruf am 2002-11-01.

[Boro02] *Borowski, S.:* Dialer – Die Tricks unseriöser Anbieter. http://www.dialerschutz.de/home/Tricks/tricks.html, Abruf am 2002-11-01.

[BrTi01] *Bridwell, L. M.; Tippett, P.:* ICSA Labs Virus Prevalence Survey 2001. http://www.trusecure.com/download/dispatch/vps-survey-2001.pdf, TrueSecure Corp, Herndon (VA) 2001, Abruf am 2003-01-06.

[CEI02] *Computer Economics:* Malicious Code Attacks Had $ 13.2 Billion Economic Impact in 2001. http://www.computereconomics.com/article.cfm?id=133, 2002-01-04, Abruf am 2003-01-05.

[Ford02] *Ford, R.:* Malware. http://www.malware.org/malware.htm, Abruf am 2002-11-01.

[Fst01] *Freiwillige Selbstkontrolle Telefonmehrwertdienste e.V.:* Verhaltenskodex für Telefonmehrwertdienste. http://www.fst-ev.org/ger/druck/verhaltenskodex.html, Abruf am 2002-11-01.

[Fuhs98] *Fuhs, H.:* Internet Hoaxes: Konzeptionelle Gesichtspunkte und praktische Auswirkungen. http://www.vhm.haitec.de/konferenz/1998/vortraege/hoax.htm, Abruf am 2002-11-01.

[GeCA03] *GeCAD:* Realtime Virus Statistics. http://www.rav.ro/ravmsstats/, Abruf am 2003-01-06.

[Kula01] *Kulakow, S.:* NetBus 2.1, Is It Still a Trojan Horse or an Actual Valid Remote Control Administration Tool? http://rr.sans.org/malicious/netbus21.php, Abruf am 2002-11-01.

[Marx00] *Marx, A.:* A Guideline to Anti-Malware-Software testing. In: EICAR 2000 Best Paper Proceedings, pp. 218–253. Online verfügbar unter: http://www.av-test.org/down/papers/2000-02_eicar_2000.zip.

[Marx02a] *Marx, A.:* Trouble Makers. In: Virus Bulletin 01/2002, S. 14–15. Online verfügbar unter: http://www.av-test.org/down/papers/2002-01_vb_trouble.pdf.

[Marx02b] *Marx, A.:* Test Lab Installation. In: Virus Bulletin 02/2002, S. 11–13. Online verfügbar unter: http://www.av-test.org/down/papers/2002-02_vb_testlab.pdf.

[Marx02c] *Marx, A.:* Rescue Me: Updating Anti-Virus Rescue Systems. In: Virus Bulletin 02/2002, S. 10–12. Online verfügbar unter: http://www.av-test.org/down/papers/2002-05_vb_rescue.pdf.

[Marx02d] *Marx, A.:* (Porn) Dialers – a New Class of Malware? In: Virus Bulletin 12/2002, S. 12–13. Online verfügbar unter: http://www.av-test.org/down/papers/2002-12_vb_dialer.pdf.

[Mess02] *MessageLabs:* VirusEye Virus Count. http://www.messagelabs.com/viruseye/default.asp?by=all, Abruf am 2003-01-06.

[Nach98] *Nachenberg, C.:* Staying Ahead of the Virus Writers: An in-depth look at heuristics. Proceedings of the Virus Bulletin Conference '98, pp. 85-98.

[Nai00] *Network Associates:* Virus Information Library: W95/MTX.gen@M. http://vil.nai.com/vil/content/v_98797.htm, Abruf am 2002-11-01.

[Nai02] *Network Associates:* Virus Glossary. http://www.mcafeeb2b.com/naicommon/avert/avert-research-center/virus-glossary.asp, Abruf am 2002-11-01.

[Open02] *Open Anti-Virus:* Mission Statement. http://www.openantivirus.org/mission.php, Abruf am 2002-11-01.

[PoBl96] *Pomberger, G.; Blaschek, G.:* Software-Engineering: Prototyping und objektorientierte Software-Entwicklung. 2. Auflage. Hanser, München, Wien, 1996.

[Siem02] *Siemens, M.:* Nummer Sicher statt Nummer 0190, AntiVir schützt auch vor überhöhter Telefonrechnung durch kostenintensive Dialer. http://www.antivir.de/news/2002/19_04_02.htm, Abruf am 2002-11-01.

[Tiet02] *Tietz, T.:* Test – Dialer setzt 0190-Schutzsoftware außer Gefecht. http://www.trojaner-info.de/news/dialer_warnkiller.shtml, Abruf am 2002-11-01.

[Viru02] *Virus Bulletin:* Useful Links, Anti-Virus Product Developer Index. http://www.virusbtn.com/AVLinks, Abruf am 2002-11-01.

[Wall90] *Wallmüller, E.:* Software-Qualitätssicherung in der Praxis. Hanser, München, Wien, 1990.

[Well02] *Wells, J.:* The WildList Organisation International. http://www.wildlist.org, Abruf am 2002-04-25.

[Whal98] *Whalley, I.:* Testing Times for Trojans. In: Proceedings of the Virus Bulletin Conference '99, pp. 55-68.

[Ziem02] *Ziemann, F.*: Hoax-Info Service, Über Computer-Viren, die keine sind (sog. „Hoaxes") und andere Falschmeldungen und Gerüchte. http://www.tu-berlin.de/www/software/hoax.shtml, Abruf am 2002-11-01.

Abstract

Systematic testing of anti-virus software

The application of anti-virus software (AV software) in companies is of increasing importance, caused by the enormous damages of different kinds of malware (malicious software). Features of different AV software systems vary in particular through the fast sequence of releases offered by different vendors. The reason for this release bombing is the still unbounded creativity of malware programmers. Therefore, it can only be analyzed through extensive and systematic tests, which software fits the current requirements regarding detection and disinfection of malware. In this paper first the potentials of damages caused by different kinds of malware will be described, followed by a presentation of a systematic test method for AV software.

Keywords: anti-virus software (av software), malware, security, test procedures