# 3. Qualitätsrichtlinien und Methoden zur Bestimmung der Erkennungsgüte von Anti-Malware-Software

Nachdem im vorangegangenen Kapitel die Anforderungen an Anti-Malware-Software genauer betrachtet wurden, sollen in diesem Kapitel die Qualitätsmerkmale bei der Durchführung eines Anti-Malware-Testes sowie die dabei angewandten Methoden untersucht werden. Es werden insbesondere die Anforderungen, die die Güte eines Anti-Malware-Testes bestimmen, untersucht.

Zunächst werden allgemeine Vorgehensweisen und Methoden für das Testen von Software beschrieben (Abschnitt 3.1). Abschnitt 3.2 befaßt sich mit Anti-Malware-Tests im speziellen, und untersucht insbesondere Kriterien, die die Qualität eines Anti-Malware-Testes ausmachen. Am Ende des Kapitels werden kurz die Methoden zur Messung und zum Testen der in Kapitel 2 vorgestellten Qualitätskriterien beschrieben (Abschnitt 3.3 für quantitative Kriterien und Abschnitt 3.4 für qualitative Kriterien).

## 3.1 Testen von Software

Allgemein versteht man unter dem Testen von Software die "Überprüfung des Ein-/Ausgabeverhaltens eines Programms anhand der Spezifikation" (vgl. [Informatik-Duden 1993], S. 720ff). In einer guten Spezifikation eines Programms ist genau dargelegt, was das Programm leisten soll, unter welchen Bedingungen es dies leisten soll - und was es nicht leisten kann. Testen heißt, diese in der Spezifikation gemachten Angaben zu überprüfen.

Die praktische Durchführung des Testens lässt sich in der Regel in folgende Phasen unterteilen (vgl. [Informatik-Duden 1993], S. 720ff):

- Testplanung
- Testvorbereitung
- Testdurchführung
- Testauswertung

Die Planung ist die Basis eines jeden Tests; sie ermöglicht ein strukturiertes Vorgehen beim Testen. Pomberger und Blaschek erläutern die Planung so: "Zweck der Testplanung ist es, die Aufgaben, Ziele und Strategien für die Testausführung festzulegen" ([PombergerBlaschek 1996], S. 155). Die Testvorbereitung umfaßt die Ermittlung einer Testmenge, eine Beschreibung der Sollergebnisse für jedes Element der Testmenge und die Erstellung einer Testumgebung (vgl. [Informatik-Duden 1993], S. 720ff). Bei der eigentlichen Durchführung des Tests werden die Elemente der Testmenge an der Software als Eingaben getestet. Die Testdurchläufe werden während der Testdurchführung protokolliert. Die Testauswertung beinhaltet die Überprüfung der Testergebnisse mit den Sollergebnissen, die in der Testvorbereitungsphase für jedes Element der Testmenge festgelegt worden waren.

Unterschieden werden kann das Testen von Software nach Black-Box-Test (auch funktionaler Test) und White-Box-Test (auch Strukturtest) (vgl. [PombergerBlaschek 1996], S. 151f). Beim Black-Box-Test wird nur die nach außen sichtbare Funktionalität eines Programms getestet und mit der angegebenen Spezifikation verglichen. Beim White-Box-Test wird auch die innere Struktur des Programms getestet; notwendig ist dafür die Kenntnis der Ablaufstruktur des zu testenden Programms. Deshalb kann ein White-Box-Test nur dann vorgenommen werden, wenn der Quellcode (oder zumindest ein grober Ablaufplan) des zu testenden Programms dem Tester vorliegt.

## 3.2 Qualitätsrichtlinien für das Testen von Anti-Malware Software

Die in 3.1 aufgeführte Definition von Testen bezieht sich hauptsächlich auf das Aufspüren von Fehlern und auf die Überprüfung der Korrektheit von Programmen. Die Spezifikation eines Anti-Malware-Programms beinhaltet als Angabe, was die vorliegende Software leisten soll, in aller Regel die Erkennung von bösartiger Software. Da beim Testen dieser Software der Quellcode nicht vorliegt, ist nur ein reiner Black-Box-Test möglich, der die Software von außen betrachtet und die angegebene Funktionalität überprüft (s.o.). Beim Testen von Anti-Malware-Software steht daher das Testen der Güte der in Kapitel 2 beschriebenen Qualitätskriterien von Anti-Malware-Software im Vordergrund, da diese die Qualität der Erkennung bestimmen. Getestet werden soll die Erfüllung des eigentlichen Zieles der Software, nämlich die Erkennung von Malware.

Die Qualität eines Anti-Malware-Testes kann von verschiedenen Blickpunkten aus betrachtet werden:

- Anwender/Benutzer
- Hersteller
- Wissenschaft

Für den Anwender, der sich einen Marktüberblick über Anti-Malware Software machen möchte, stehen im Vordergrund Objektivität und die Herstellerunabhängigkeit des Tests, da er die Ergebnisse meist als Entscheidungsgrundlage für den Erwerb eines entsprechenden Produktes nutzen möchte. Da die Endanwender im Normalfall zu wenig Fachwissen über die Erkennung von bösartiger Software besitzen, um die Ergebnisse genau analysieren und sich ein eigenes Bild machen zu können, müssen sie sich auf die Ergebnisse eines Tests verlassen. Deshalb ist für die Benutzer von hoher Wichtigkeit, daß sie sich auf die Unabhängigkeit und die Objektivität eines Tests verlassen können.

Die Hersteller von Anti-Malware Produkten können oft großen Nutzen aus den Ergebnissen eines Tests ziehen, sofern diese detailliert nachvollziehbar sind und sinnvolle Aussagen über die Produkte beinhalten. Schließlich werden in einem Test (dem Hersteller bekannte oder unbekannte) Fehler aufgedeckt, die es zu verbessern gilt. Auch der Vergleich mit anderen

Produkten ist für die Hersteller wichtig, häufig wird ein positives Abschneiden beim Vergleich mit anderen Produkten zu Marketingzwecken genutzt.

Für die Wissenschaft, die weniger an den Ergebnissen einzelner Produkte, sondern an der Gesamtentwicklung der Erkennung von bösartiger Software interessiert ist, sind Vollständigkeit, Nachvollziehbarkeit und Aussagekraft der Ergebnisse wichtige Qualitätsmerkmale eines Tests. Denn nur Aussagen, die die genannten Eigenschaften besitzen, können als wissenschaftlich relevant betrachtet werden.

Alle genannten Aspekte beeinflussen aus unterschiedlichen Blickwinkeln die Qualität eines Anti-Malware-Testes und werden zusammenfassend aufgezählt:

- Vollständigkeit
- Größe und Qualität der Testmenge
- Nachvollziehbarkeit der Ergebnisse
- Objektivität und Herstellerunabhängigkeit
- Aussagekraft der Ergebnisse
- Nützlichkeit der Ergebnisse

Es folgt eine detaillierte Betrachtung der einzelnen Gesichtspunkte hinsichtlich der Qualität eines Anti-Malware Tests.

### 3.2.1 Vollständigkeit und Größe der Testmenge

Will man Fehler in einer Software durch Testen aufdecken, so ist es in der Regel nicht möglich, sämtliche Eingabewerte auf die (laut Spezifikation) gewünschten Ausgabewerte zu überprüfen (vgl. [PombergerBlaschek 1996], S. 156: "...vollständiges Testen auch bei einfachen Testobjekten in der Regel unmöglich ist. Die Anzahl an Kombinationen von Eingabedaten {Testfällen} ist schon bei einfachen Testobjekten so groß, daß derartige Tests nicht realisiert werden können."). Deshalb wird meistens aus einer Vielzahl von Möglichkeiten eine Testmenge von Eingaben ausgewählt und die Ausgabe vom Programm auf diese Eingabe überprüft. Die richtige Auswahl von Testfällen ist entscheidend für die erfolgreiche Aufdeckung von Fehlverhalten der Software.

Bei Sicherheitsprogrammen ist eine umfangreiche Testmenge sinnvoll, da eine hohe Sicherheit für den Benutzer getestet werden soll. Ein Antivirenprogramm erkennt laut Spezifikation bösartige Software und ist in der Lage, diese zu entfernen. Diese Eigenschaft muß möglichst vollständig getestet werden, damit eine glaubhafte Aussage über die bereitgestellte Sicherheit eines Programms gemacht werden kann. Es ist zwar auch im Falle von Sicherheitssoftware nicht möglich, jede mögliche Eingabe zu überprüfen (das wäre auf den Scanprozeß betrachtet die Erzeugung jeder möglichen Zeichenfolge als Datei), aber man muß die Malware, die man zum Testen zur Verfügung hat, vollständig testen und nicht nur Stichproben daraus zum Testen nehmen. Mit anderen Worten: will man die Erkennung von bösartiger Software testen, so kann man gutartige Software stichprobenartig in die Testmenge

einstreuen (für das Testen nach sogenannten "false positives", vgl. Abschnitt 1.2), die bösartige Software muß aber (soweit vorhanden) vollständig als Eingabe getestet werden.

Der Informatik-Duden definiert eine ideale Testmenge zum Testen eines Programms auf seine Korrektheit (entsprechend der Definition in Abschnitt 2.3.4) folgendermaßen ([Informatik-Duden 1993], S. 723):

> "Eine Testmenge T zu einem Programm P heißt ideal, falls P genau dann korrekt ist, wenn P für alle Testwerte aus T korrekte Ergebnisse liefert."

Weiterhin schreibt der Informatik-Duden zum Sinn einer idealen Testmenge: "Findet man zu einem Problem und einem Programm die ideale Testmenge, so ist viel gewonnen: Man braucht das Programm nicht zu verifizieren, sondern führt es für alle (also endlich viele) Eingabewerte der Testmenge aus und kann dann bereits mit absoluter Sicherheit entscheiden, ob das Programm korrekt ist oder nicht" ([Informatik-Duden 1993], S. 723). Eine Testmenge ist also ideal, wenn sie zum Auffinden von Fehlern in einem Programm vollständig ist. Eine ideale Testmenge zum Testen von Programmen ist deshalb so sinnvoll, weil mit ihrer Hilfe gesicherte Aussagen über die Qualität von Programmen gemacht werden können.

Für das Problem der Erkennung von bösartiger Software ließe sich eine ideale Testmenge nur durch die Integration aller existierenden Malware in die Testmenge bilden, da aus der Erkennung von bestimmten Viren oder maliziös verseuchten Dateien keine Rückschlüsse auf die Erkennung anderer Malware gezogen werden können. Als Beispiel betrachte man das Testen von Zuverlässigkeit der Virenerkennung bei den Tests des VTC. Einige getestete Scanner erkennen alle Musterdateien aller Viren in einer Datenbank bis auf einige, wenige Dateien. Eine kleinere, diese Dateien nicht enthaltende Testmenge würde in so einem Fall zu falschen Rückschlüssen über die Erkennungszuverlässigkeit solcher Scanner führen, da in diesem Falle alle im Test vorhandenen Musterdateien erkannt würden und man so den Scanner als zuverlässig erachten würde. Es gilt deshalb[28]: Je größer die Testmenge, desto mehr nähert sich die Testmenge der idealen Testmenge.

Leider steht dem Tester nicht die vollständige Menge an Malware zur Verfügung[29]. Andernfalls wäre ein vollständiges Testen aller bösartigen Software auf die gewünschte Ausgabe laut Spezifikation, nämlich die Erkennung der bösartigen Software, zumindest theoretisch möglich. Ein solcher Test gäbe als Ergebnis die reale Erkennungsrate pro getestetem Produkt als Prozentsatz aller vorhandenen Malwareobjekte an. Es ist aber unmöglich, alle auf der Welt vorhandenen, mit Malware infizierten Dateien als Samples in einer Datenbank zu sammeln und dann Anti-Malware-Software auf Erkennung der Objekte in dieser Datenbank zu testen.

---

[28] für Testmengen bösartiger Software zum Testen der Malware-Erkennung

[29] Selbst wenn es möglich wäre, durch Integration aller vorhandener Malware eine ideale Testmenge für Anti-Malware-Programme zu bilden, so wäre diese Testmenge immer nur für einen kurzen Zeitraum ideal, nämlich so lange, bis neue Malware entdeckt wird.

Man kann sich die Bedeutung der Größe der Testmenge auch durch statistische Überlegungen klar machen. Die Erkennungsrate beim Testen auf einer durch Sammeln von Musterdateien zusammengestellten Datenbank hat als Aussageziel die Annäherung an die oben beschriebene "reale" Erkennungsrate. Je größer die Datenbank, auf der getestet wird, ist, desto mehr nähert sich statistisch die Erkennungsrate der realen Erkennungsrate an. Dies liegt daran, daß einzelne, nicht direkt zufällige, aber untypische Fehler der Software bei der Erkennung mit wachsender Größe der Datenbank immer mehr vernachlässigt werden können. In der Statistik bezeichnet man dies als "empirisches Gesetz der großen Zahlen" (siehe zum Beispiel [Hübner 1996], S. 16f).

Man stelle sich beispielsweise vor, man habe eine Datenbank, in der von jedem Virus *nur eine* befallene Datei enthalten ist. Erkennt ein Produkt die Datei nicht als viral, ist unklar, ob das Produkt diesen Virus generell nicht erkennt oder ob es nicht alle befallenen Objekte dieses Virus erkennt. Ferner kann selbst bei einer Erkennung der Musterdatei keine genaue Aussage darüber gemacht werden, wieviel Prozent aller befallenen Objekte dieses Virus das Produkt erkennt, ob allgemein die Erkennung also eher die Ausnahme oder die Regel darstellt. Um exakt aussagen zu können, wieviele aller mit diesem Virus verseuchten Dateien das Produkt erkennt, müßte man im Besitz aller dieser Dateien sein (was in der Realität nicht umsetzbar ist), und hätte dann die reale Erkennungsrate[30]. Es sollte klar sein, daß die gemessene Erkennungszuverlässigkeit bei einer Datenbank mit vielen Musterdateien pro Virus statistisch betrachtet näher an dem realen Wert liegt als bei einer Datenbank mit wenigen Musterdateien pro Virus. Einzelne Ausnahmen an Erkennungen und Nichterkennungen gleichen sich bei einer großen Anzahl an Musterdateien aus, so daß die gemessene Erkennungsrate (auf die Datenbank gesamt betrachtet) und Erkennungszuverlässigkeit sowie -genauigkeit (auf einzelne Viren oder Virenfamilien betrachtet) dann aussagekräftiger sind.

Deshalb ist eine gewisse (im Sinne von statistisch ausreichend) Größe der Datenbanken eines Anti-Malware-Testes zur Gewinnung von auf die Allgemeinheit übertragbarer Aussagen in zweierlei Hinsicht nötig:

- die Gesamtzahl der Objekte in einer Datenbank muß möglichst groß sein, damit die gewonnene Aussage über die Erkennungsrate auf der in der Datenbank enthaltenen Art von Malware als Annäherungswert der "realen" Erkennungsrate erachtet werden kann
- die Anzahl an Samples pro Virus muß möglichst groß sein, damit die gewonnene Aussage über die Erkennungszuverlässigkeit sowie -genauigkeit auf der in der Datenbank enthaltenen Malware als Annäherungswert an die Realität erachtet werden kann

[30] in diesem Beispiel genauer: die Erkennungszuverlässigkeit, da die zuverlässige Erkennung mehrerer mit demselben Virus verseuchten Dateien betrachtet wird (vgl. 2.2.2).

### 3.2.2 Qualität der Testmenge

Da bereits ab einer geringen Größe der Datenbanken (und erst recht bei einer statistisch aussagekräftigen Größe) die einzelnen Musterdateien nicht manuell auf das tatsächliche Enthalten von Malware überprüft werden können, ist die Qualität der Objekte in einer Datenbank hinsichtlich der Korrektheit der Musterdateien sicherzustellen. Mit Korrektheit der Musterdateien ist gemeint, daß diese auch tatsächlich mit dem angegebenen Virus infiziert sind. Da die infizierten Musterdateien unter Umständen (wie beim VTC) aus einer Vielzahl von Quellen kommen, ist diese Art der Qualität der verseuchten Dateien nicht immer einheitlich. Die Qualität der erhaltenen Musterdateien wird durch die Zuverlässigkeit und Professionalität der Quellen für die Virendatenbanken bestimmt. So kommt es beispielsweise vor, daß in einer Kollektion von Viren einige Dateien enthalten sind, die nicht eindeutig mit einem Virus verseucht sind[31], oder die sogar völlig "sauber" sind und versehentlich mit in die Kollektion von Viren gelangt sind.

Nach Erhalt der Viren kann die Qualität der Datenbank nur noch durch automatische Prüfungen verbessert werden. Hierzu lässt man beispielsweise eine Vielzahl von bekanntermaßen guten Malwareerkennungsprogrammen über die Datenbanken laufen und schaut sich die Dateien, die von keinem Programm als bösartig erkannt wurden, manuell daraufhin an, ob es sich tatsächlich um Malware handelt. Die Qualität der zum Testen von Anti-Malware-Software verwendeten Testmenge ist deshalb wichtig, weil eine Testmenge mit vielen nicht richtig infizierten oder sauberen, aber von den Testern als maliziös eingestuften Dateien die Messung der Erkennung verfälscht.

### 3.2.3 Nachvollziehbarkeit der Ergebnisse

Bei jeder wissenschaftlichen Untersuchung ist die Nachvollziehbarkeit der Ergebnisse von äußerster Wichtigkeit, weil die durch die ermittelten Ergebnisse getroffenen Aussagen sonst nicht bewiesen sind. Ohne in ihrer Entstehung nachvollziehbare Ergebnisse handelt es sich aus wissenschaftlicher Sicht um bedeutungslose Aussagen.

Nachvollziehbarkeit eines Tests wird dadurch erreicht, daß die einzelnen Testschritte und das Testverfahren detailliert dokumentiert sind. Zusätzlich müssen Berechnung und Bewertung der Testergebnisse nachvollziehbar sein, indem Zwischenergebnisse und Bewertungsmethoden veröffentlicht werden, so daß jeder Leser des Tests die angegebene Bewertung durch eigene Berechnung wiederholen kann. Bei einem guten Anti-Malware-Test

[31] Nicht eindeutige Infektion bedeutet zum Beispiel, daß ein Virus eine Opferdatei nur teilweise befallen hat (etwa weil er bei der Infektion unterbrochen wurde oder eine ungewöhnliche Systemkonstellation die vollständige Infektion verhindert hat), und so die befallene Datei nicht weiter zur Replikation fähig ist. Die besagte Datei ist zwar maliziös verseucht, aber nicht mit dem Virus, da sie den Viruscode nicht vollständig zur weiteren Verbreitung enthält. Einige Hersteller erkennen durch ihre Signatur solche Dateien, da ja Bruchstücke des Viruscodes in der Datei enthalten sind. In einem Test sollten allerdings nur virale Musterdateien - gemäß der Definition in Kapitel 1 - verwendet werden; sie müssen also zur Replikation fähig sein. Deshalb müssen nicht eindeutig infizierte Dateien aus der Testmenge entfernt werden.

sind deshalb alle einzelnen Testergebnisse detailliert verfügbar, auch wenn als Testergebnisse vornehmlich kumulierte und durchschnittliche Werte interessieren.

### 3.2.4 Objektivität und Herstellerunabhängigkeit

Da auch die Hersteller von Anti-Malware-Produkten einen Nutzen aus den Ergebnissen eines Tests von Anti-Malware-Software ziehen (s.o.), werden einige Tests von Herstellerseite aus unterstützt. So ist zum Beispiel allgemein bekannt, daß das Magazin *Virus Bulletin*, welches regelmäßig Anti-Malware-Tests durchführt, den gleichen Besitzer hat wie die Software *Sophos-Antivirus* (siehe [Bjergstrom 2001], S.1: „It is a potential weakness that it {Virus Bulletin} is published and/or owned by the people behind the anti-malware vendor Sophos Ltd ...”). Im Prinzip ist an einer solchen Unterstützung nichts auszusetzen, da ja mitunter auch Fachwissen und die Bedürfnisse der Hersteller mit in das Testen einfließen. Dennoch sind aus Sicht der Benutzer und vor allem aus Sicht der Wissenschaft, die an wahrheitsgemäßen, objektiven Aussagen interessiert ist, Zweifel angebracht bezüglich der Objektivität des Testens und der Auswertung eines von Herstellerseite unterstützen Tests. Eine faire Behandlung aller Teilnehmer eines solchen Tests erscheint fraglich, da die anderen Teilnehmer Konkurrenten des unterstützenden Herstellers sind und es bei Testergebnissen letztendlich auch um Marktanteile und Umsatz geht. Diese Tatsache legt die Vermutung nahe, daß ein den Test unterstützendes Produkt vorteilhafter dargestellt wird.

Will man sich der Fairneß und Gleichbehandlung aller Produkte in einem Test von Anti-Malware-Software sicher sein, so ist die Unabhängigkeit von Herstellern eine Richtlinie für die Güte des Tests.

### 3.2.5 Aussagekraft und Nützlichkeit der Ergebnisse

Unterschiedliche Leser eines Anti-Malware-Testes haben in aller Regel unterschiedliche Motive und Ausgangssituationen (zum Beispiel privater Gebrauch zum Schutz eines Einzelplatzsystems, unternehmensweiter Schutz aller Clienten eines Netzwerkes, Einsatz auf einem Gateway-Rechner), zu deren Schutz sie Anti-Malware-Programme einsetzen möchten. Deshalb müssen die Ergebnisaussagen eines Tests einen allgemeinen Charakter haben, um für verschiedene Zielgruppen nützlich und für die Allgemeinheit aussagekräftig zu sein. Die Qualität eines Tests von Software wird nicht nur durch die Qualität der gemessenen Ergebnisse hinsichtlich Nachvollziehbarkeit, Qualität der Testmenge und Ähnlichem bestimmt, sondern auch anhand der eigentlichen Nützlichkeit der Ergebnisse (wie Erkennungsrate, Erkennungszuverlässigkeit und -genauigkeit, usw., vgl. Abschnitt 2.2 und 2.3) für Benutzer, Hersteller und Wissenschaft. Mit anderen Worten ist neben dem *wie gut* nicht zuletzt auch wichtig, *was* überhaupt gemessen und getestet wurde.

Für die Hersteller von Antivirensoftware sind exakte und sehr detaillierte Angaben über die Erkennung und genaue Identifikation einzelner Viren von großem Nutzen. Durch solche Ergebnisse können sie ihr Produkt verbessern und bisher unbekannte Schwächen und Fehler

aufdecken. Bekommen die Hersteller, wie bei den VTC-Tests[32], nach dem Test die nicht erkannten Musterdateien (*missed samples*) zugeschickt, so können sie ihre Produkt durch Erstellung entsprechender Signaturen und Überprüfen der Gründe für die Nichterkennung erheblich verbessern. In diesem Fall tragen die Testergebnisse auch zu einem verbesserten Gesamtschutz der Benutzer - und somit zur Bekämpfung der durch Malware existenten Bedrohung von Computersystemen insgesamt - bei.

Für die Benutzer, die einen Test häufig als Entscheidungsgrundlage für einen Kauf von Antivirensoftware nehmen, ist der Vergleich der einzelnen Produkte untereinander wichtig. Außerdem ist es für Benutzer wie Unternehmen, die genau wissen, welchen speziellen Gefahren sie in Teilbereichen ausgesetzt sind, von Vorteil, wenn in Bezug auf unterschiedliche Malwarearten differenzierte Ergebnisse der einzelnen (quantitativen) Qualitätskriterien vorliegen. So können sie unabhängig von der Bewertung des Tests (Nachvollziehbarkeit der Ergebnisse vorausgesetzt) das für die eigenen Bedürfnisse optimale Produkt auswählen.

Für die Wissenschaft sind nur Ergebnisse der Messung quantitativer Kriterien interessant, da qualitative Kriterien nicht objektiv bewertet werden können (vgl. Abschnitt 2.3). Hierbei werden Aussagen über die Erkennung von bestimmten Malwarearten im allgemeinen benötigt, und nicht die Werte einzelner Produkte. Desweiteren interessieren Durchschnittswerte aller Scanner und die Entwicklung der Erkennung und anderer Kriterien über einen längeren Zeitraum, das heißt der Fortschritt von Anti-Malware-Software als Gesamtheit betrachtet.

Die Aussagekraft der Ergebnisse eines Tests der Erkennung von bösartiger Software wird hauptsächlich durch die Qualität und Größe der Testmenge sowie durch die Nachvollziehbarkeit der Ergebnisse und Dokumentation des Vorgehens beim Testen bestimmt.

## 3.3 Methoden für die Messung von quantitativen Kriterien

Die in Abschnitt 2.2 erläuterten quantitativen Kriterien für die Qualität eines Anti-Malware-Produktes sind:

- Erkennungsrate
- Erkennungsgenauigkeit
- Erkennungszuverlässigkeit
- Häufigkeit von Falschmeldungen
- Unterstützung von Datenformaten
- Geschwindigkeit

[32] Das Virus Test Center ist die einzige Institution, die den Herstellern nichterkannte Musterdateien zuschickt (vergleiche auch Kapitel 5).

- Reparatur von infizierten Dateien

Grundsätzlich erfolgt die Messung der Erkennung mit Hilfe sogenannter Protokoll-Dateien (auch Reportdateien genannt), die von den Anti-Malware-Produkten beim Scannen der Datenbanken erzeugt werden. Abbildung 3.A zeigt einen beispielhaften Ausschnitt einer Protokolldatei[33]. In jeder Zeile der Protokolldatei hat der Scanner den Namen einer gescannten Datei gespeichert. Rechts neben dem Dateinamen steht (getrennt durch "...") die Diagnose des Scanners. Bei erfolgreicher Erkennung (im Beispiel werden alle aufgelisteten Dateien als viral erkannt) wird der Name des erkannten Virus zur genauen Identifikation angegeben. Am Ende der Reportdatei listet dieser Scanner auch noch eine Zusammenfassung des Scanvorgangs auf, inklusive der für den Vorgang benötigten Zeit.

```
...
R:\XM\WEIT\A\XM_000_.XLS ... Found the X97M/Weit.gen virus !!!
R:\XM\WEIT\A\XM_001_.XLS ... Found the X97M/Weit.gen virus !!!
R:\XM\WEIT\A\_X_X_X_X.XLS ... Found the X97M/Weit.gen virus !!!
R:\XM\WEIT\B\BOOK1.XLS ... Found the X97M/Weit.gen virus !!!
R:\XM\WEIT\B\XM_000_.XLS ... Found the X97M/Weit.gen virus !!!
R:\XM\WEIT\B\XM_001_.XLS ... Found the X97M/Weit.gen virus !!!
R:\XM\YOHIMBE\A\XMYOHIMB.XLS ... Found the XM/Yohimbe.a virus !!!
R:\XM\YOHIMBE\A\YOHIMBE.XLS ... Found the XM/Yohimbe.a virus !!!
R:\XM\YOHIMBE\A\YOHIMBE1.XLS ... Found the XM/Yohimbe.a virus !!!
R:\XM\YOHIMBE\A\YOHIMBE2.XLS ... Found the XM/Yohimbe.a virus !!!
R:\XM\YOHIMBE\A\YOHIMBE3.XLS ... Found the XM/Yohimbe.a virus !!!
R:\XM\YOHIMBE\A\YOHIMBE4.XLS ... Found the XM/Yohimbe.a virus !!!
R:\XM\YOHIMBE\B\LAURIE.XLS ... Found the XM/Yohimbe.b virus !!!
R:\XM\YOHIMBE\B\LAURIE1.XLS ... Found the XM/Yohimbe.b virus !!!

Summary report on R:\XM\*.*
File(s)
      Total files: .....     789
      Clean: .....            0
      Possibly Infected: .....     789


Time: 00:02.16
```

Abbildung 3.A: Ausschnitt aus dem Protokoll eines Virenscanners

[33] Protokoll von NAI VirusScan aus dem VTC Test 2001-10, siehe SCN.RAR\W2K\MAC in [VTC 2001-10b]

Aus derartig aufgebauten Protokolldateien können die Ergebnisse hinsichtlich der aufgelisteten Kriterien (s.o.) abgeleitet werden. Im folgenden Abschnitt wird für die einzelnen Kriterien beschrieben, wie die Gesamtergebnisse aus Protokolldateien gewonnen werden.

### 3.3.1 Messung von quantitativen Qualitätskriterien

Die Meßwerte von quantitativen Qualitätskriterien werden im Zuge der Auswertung von Testprotokollen gewonnen. Auswertung bedeutet den Vergleich der Testprotokolle mit den Datenbanken. Dies geschieht in der Regel durch entsprechende Auswertungsprogramme, die skriptgesteuert automatisch die in den Testprotokollen gemeldeten Erkennungen mit den Datenbanken vergleichen. Anders wäre ein solcher Vergleich bei großen Datenbanken kaum machbar. Dazu muß zu den Datenbanken vorliegen, welche Dateien mit welchem Virus infiziert sind, und welche Dateien nur als *false positives* in die Datenbank eingestreut wurden. Die Meßwerte der Qualitätskriterien werden dabei folgendermaßen gewonnen:

| | |
|---|---|
| *Erkennungsrate:* | Da in den Scan-Protokollen genau die Erkennung jedes einzelnen Objektes gemeldet wird, ist ersichtlich, welche Dateien ein Produkt als bösartig erkannt hat und welche nicht. So kann durch den Abgleich der Protokolle mit den Datenbanken die Erkennungsrate berechnet werden. |
| *Erkennungsgenauigkeit:* | Zu jedem als Malware erkannten Objekt in einer Datenbank wird im Scan-Protokoll eine Bezeichnung zu dieser gefundenen Malware geliefert. Dadurch können komplexere Auswertungsskripte die Benamung der gefundenen bösartigen Objekte auf ihre Genauigkeit überprüfen. |
| *Erkennungszuverlässigkeit:* | Erkennungszuverlässigkeit wird dadurch getestet, daß in die Datenbanken pro Virus eine Vielzahl unterschiedlicher, mit diesem Virus infizierter Musterdateien integriert wird. In den Testprotokollen ist dann nachprüfbar, ob ein Produkt alle Musterdateien eines Virus erkannt hat oder wieviele Musterdateien es erkannt hat. |
| *Häufigkeit von Falschmeldungen:* | Die potentiell falsche Identifikation von sauberen Dateien als Malware wird dadurch gemessen, daß man an unterschiedlichen Stellen saubere Dateien in die Datenbank einstreut. In den Protokolldateien ist nach dem Scannen klar ablesbar, ob ein Produkt diese Dateien als maliziös (was einer Falschmeldung entspräche) oder sauber meldet. |
| *Unterstützung von Dateiformaten:* | Durch die Integration von Virensamples unterschiedlicher Dateiformate wird die Erkennung |

dieser Dateiformate automatisch mitgetestet. Zur Messung der Erkennungsrate einzelner Dateiformate ist es sinnvoll, diese in separate Datenbanken oder Unterverzeichnisse zu sortieren, so daß bei der Auswertung die entsprechende Erkennungsrate leicht berechnet werden kann. Gleiches gilt auch für mit unterschiedlichen Verfahren komprimierte Viren.

*Geschwindigkeit:* Die für das Scannen einer Datenbank benötigte Zeit kann in den meisten Fällen direkt aus der Report-Datei entnommen werden. Wird dieser Wert nicht im Protokoll mit aufgelistet, oder traut man den Angaben der Produkte nicht, so muß die Dauer eines Tests von Hand oder durch einen automatisierten Scan-Aufruf, welcher die Zeit stoppt, gemessen werden. Letzteres kann beim Scannen per CLI-Aufruf durch eine Batch-Datei erfolgen, welche vor und nach Aufruf des eigentlichen Scans die Systemzeit speichert.

*Reparatur von infizierten Dateien*: Die einwandfreie Reparatur von befallenen Dateien ist nicht einfach durch einen Scanvorgang und das Auswerten des Protokolls zu testen. Hierbei müssen die von den Scannern gesäuberten Dateien genau untersucht werden, um die Korrektheit der Säuberung zu bestätigen. Diese Überprüfung muß bei großen Mengen an Dateien nach Möglichkeit automatisch ablaufen. Ein Verfahren zur Messung und speziellen Bewertung der Reparatur von mit Malware infizierten Dateien ist im Virus Test Center der Universität Hamburg von Martin Retsch und Stefan Tode entwickelt worden (vgl. [RetschTode 2000] und Abschnitt 4.3.2).

Da ein Anti-Viren-Produkt grundsätzlich in zwei verschiedenen Betriebsarten ausführbar ist, nämlich im *On-access* Modus und im *On-demand* Modus (zur Erläuterung der Modi siehe Kapitel 1), liefern sämtliche quantitativen Qualitätskriterien jeweils nur in bezug auf die benutzte Betriebsart eine Aussage. Denn die Ergebnisse eines Scanners bei *On-demand*-Scannen und *On-access*-Überwachung sind nicht unbedingt identisch (vgl. beispielsweise in Tests von Anti-Malware-Software unter [AV-Test 2002a], [ICSA-Labs 2002] oder [VirusBulletin 2002]). In den folgenden beiden Abschnitten wird auf die Unterschiede beim Testen der Erkennung dieser beiden Modi von Anti-Malware-Software eingegangen.

### 3.3.2 Testen der Erkennung im On-demand Modus

Beim Testen im Modus *On-demand* wird normalerweise die zu testende Software in einer vorher definierten Testumgebung installiert. Dann werden die entsprechenden Optionen im Programm eingestellt, vor allem wird die Protokollierung des Scannens aktiviert. Im nächsten Schritt wird der Scanprozeß auf einer der Datenbanken gestartet, auf die der Testrechner entweder über Netzwerk Zugriff hat, oder die lokal auf dem Testrechner plaziert sind. Dieser Schritt wird für alle zu testenden Datenbanken wiederholt. Werden mehrere Produkte getestet, so muß vor Installation eines neuen Produktes meist das Betriebssystem erneut installiert werden, da bei einer Deinstallation (zum Beispiel unter Microsoft-Windows Systemen) nicht auszuschließen ist, daß noch Restdateien im System verbleiben, die die Funktion eines nachfolgend installierten Programms negativ beeinflussen können. So ist es möglich, nacheinander (oder bei Netzwerkzugriff auf die Datenbanken auch gleichzeitig auf mehreren Rechnern) mehrere Produkte zu testen. Danach werden die Testprotokolle ausgewertet, wodurch das Testergebnis entsteht. Die einzelnen Schritte sind in chronologischer Reihenfolge:

1. Testumgebung definieren
2. Testumgebung installieren
3. Datenbanken in Testumgebung verfügbar machen
4. Produkt installieren
5. Optionen einstellen
6. Scan auf Datenbank anwerfen
7. 6 wiederholen, bis alle Datenbanken getestet sind
8. 4 bis 7 wiederholen, bis alle Produkte getestet sind
9. Testprotokolle der Scanvorgänge aller Produkte auf allen Datenbanken auswerten

### 3.3.3 Testen der Erkennung im On-access Modus

Das Testen der Erkennung von Anti-Malware-Software im *On-access* Modus ist schwieriger als das Testen im *On-demand* Modus. Dies liegt daran, daß die zu testende Datenbank nicht einfach auf eine Festplattenpartition plaziert und dann der Scanprozeß gestartet werden kann. Stattdessen muß vom Test der *Zugriff* (engl. "access") auf die als Testmenge fungierenden Dateien simuliert werden. Ein Zugriff auf eine Datei besteht zum Beispiel bei folgenden Aktionen, die alle aktiv vom Benutzer vorgenommen werden:

- Datei öffnen
- Datei kopieren
- Datei ausführen

Diese Aktionen müssen vom Test nachgestellt werden, damit die On-access Erkennung getestet werden kann. Die Schwierigkeit beim Testen der On-access Erkennung großer Datenbanken besteht daher in der Konzeption einer geeigneten Testumgebung, welche den Benutzerzugriff auf Dateien simulieren soll. Solch eine Testumgebung muß die aufgelisteten Dateizugriffe automatisch simulieren können, damit ein Testen der On-access Erkennung von großen Datenbanken praktisch möglich wird.

Ein Beispiel für eine komplexe Testumgebung zum Testen von Antivirus-Software im On-access Modus ist das sogenannte "Automatic and Controlled Macro Virus Execution and Automating the Windows Environment" System von Marco Helenius (Universität Tampere, Finnland, [Helenius 1998]). Das System kann automatisch Viren replizieren und das Windows-System so automatisieren, daß zum Beispiel Viren ausgeführt oder kopiert werden können. Die Testumgebung besteht aus drei Rechnern: Einem Opfer-PC, einem Monitor-PC, und einem Netzwerk-PC. Auf dem Opfer-PC wird maliziöser Code ausgeführt, den sich der Opfer-PC vom Netzwerk-PC holt. Der Monitor-PC überwacht den Opfer-PC und kann auf ihm über eine Tastatursteuerung[34] Befehle ausführen sowie ihn neu booten. Außerdem kann der Monitor-PC das Aufspielen eines "sauberen" (das heißt vor einer Infektion angefertigten) Betriebssystemimages auf den Opfer-PC veranlassen, bevor er ihn neu startet. Dieses Image wird auf dem Netzwerk-PC gespeichert und kann so jederzeit auf den Opfer-PC aufgespielt werden.

Ein anderes Konzept zur Durchführung von Anti-Malware-Tests im On-access Modus wird zur Zeit[35] im Virus Test Center der Universität Hamburg von Ulrike Siekierski ([Siekierski 2002]) entwickelt, mit dem Ziel, in Zukunft auch im VTC On-access Tests durchzuführen.

Die quantitativen Qualitätskriterien werden ebenso wie beim On-demand Testen durch Auswertung der beim Testen erzeugten Protokolle erlangt (s.o.). Die Protokolle können jedoch beim On-access-Scannen anders aufgebaut sein, da kein "Durchscannen" eines Laufwerks möglich ist, sondern mitunter vor jeder zu scannenden Datei neu gebootet werden muß. Dadurch entsteht eine Vielzahl von einzelnen Protokolldateien, die aber im Prinzip ähnlich automatisiert ausgewertet werden kann, wie eine einzelne Protokolldatei beim On-demand-Scannen.

Durch die beschriebene Simulation des Zugriffs auf Dateien ist das Testen der Erkennung von bösartiger Software im On-access Modus nicht nur aufwendiger in der Planung und Installation der Testumgebung, sondern auch im Hinblick auf die benötigte Zeit. Das genannte System von Marco Helenius braucht zum Beispiel auf einem Pentium-I-Rechner mit 90 Mhz eine Woche, um nur 650 Makroviren durch ferngesteuerten Zugriff zu replizieren ([Helenius 1998], S.11). Für den Test von Virenscannern, die im Hintergrund den Zugriff auf Dateien überwachen, ist zwar nicht die vollständige Replikation von Viren, sondern nur der einfache Zugriff auf entsprechende Dateien ferngesteuert auszuführen. Die größte Menge an Zeit wird aber beim ferngesteuerten Neustarten des Opfer-PC verbraucht. Somit wird deutlich, daß das Testen von On-access-Erkennung wesentlich zeitaufwendiger ist als das Testen von On-demand-Erkennung.

---

[34] Solch eine Tastatursteuerung kann zum Beispiel realisiert werden, indem vom Monitor-PC eine direkte Verbindung zum Tastatureingang des Opfer-PC besteht, über die der Monitor-PC entsprechende Signale senden kann, die dann als Tastaturbefehle vom Opfer-PC interpretiert werden (vgl. [Helenius 1998], S.6)

[35] Juni 2002

## 3.4 Methoden für die Qualitätsbestimmung von qualitativen Kriterien

Die in Abschnitt 2.3 identifizierten qualitativen Kriterien für die Qualität eines Anti-Malware-Produktes sind:

- Bedienbarkeit
- Benutzerfreundlichkeit
- Stabilität
- Funktionalität
- Korrektheit
- Anpaßbarkeit
- Wartungsfreundlichkeit
- Administrierbarkeit und -aufwand

Die Qualität dieser Gesichtspunkte ist nicht einfach zu testen, da keine absoluten Werte gemessen werden können, sondern subjektive Empfindungen des Testers in die Bewertung eingehen. Einige Tests bewerten daher diese Kriterien auch nicht mit absoluten Zahlen, sondern mit den Bezeichnungen von Schulnoten (sehr gut, gut, befriedigend, ausreichend, mangelhaft, ungenügend) oder ähnlichen Bewertungszeichen (so zum Beispiel die Zeitschrift *c't* mit "--" bis "++", siehe [MarxBrauch 2001]), welche tendenziell die Ausprägung der Qualität des Produktes unter dem entsprechendem Gesichtspunkt aufzeigen sollen. Ein so geartetes Bewertungsschema zeigt den subjektiven Charakter der Aussagen und verdeutlicht so dem Leser, daß diese Aussagen nur die Meinung des Testers wiedergeben.

Für fast alle Kriterien gilt, daß bei einem systematischen Test als Testmenge eine Reihe von Aufgaben aufgestellt wird. Diese Aufgaben werden so gewählt, daß sie möglichst als repräsentativ für das generelle Verhalten der Software bezüglich des Testkriteriums anzusehen sind. Der Tester versucht, die Aufgaben mit der jeweils zu testenden Software durchzuführen. Das Abschneiden wird dann vom Tester bewertet. Häufig findet keine eigentliche Qualitätsbestimmung statt, sondern es wird nur nach der Funktionalität hinsichtlich eines Kriteriums bewertet, oft sogar ohne diese Funktionalität detailliert zu überprüfen (vgl. [PC Professionell 2002], S. 106).

Aufgrund der beschriebenen Subjektivität werden die in diesem Abschnitt beschriebenen Methoden nicht als *Testen* (womit eher die nachvollziehbare Messung objektiver Testergebnisse gemeint ist) sondern - allgemeiner - als *Bestimmung* der Qualität hinsichtlich der genannten Gesichtspunkte bezeichnet. Da die Untersuchung dieser Gesichtspunkte oftmals keine eindeutigen Ergebniswerte, sondern Bewertungen durch Aussagen liefert, können bei vielen der Kriterien keine üblichen Bewertungsmethoden angegeben werden. Stattdessen wird im folgenden das in Tests (hauptsächlich von Computerzeitschriften)

übliche - in der Regel nicht ausreichend dokumentierte - Vorgehen zur Bewertung der einzelnen Kriterien beschrieben[36]:

*Bedienbarkeit und Benutzerfreundlichkeit:* Bedienbarkeit und Benutzerfreundlichkeit lassen sich kaum systematisch testen. Deshalb wird in der Regel der Gesamteindruck wiedergegeben, der beim Gebrauch der Software entstanden ist. Ergänzt wird dieser Gesamteindruck häufig durch während des Testens besonders aufgefallene Eigenschaften (positive wie negative). Obwohl ein derartiges Vorgehen recht üblich ist (vgl. [MarxBrauch 2001]) und objektive Bewertungskriterien für Bedienbarkeit und Benutzerfreundlichkeit schwer aufzustellen sind, ist es eindeutig subjektiv. Der individuelle Eindruck des jeweiligen Testers prägt das Ergebnis.

*Stabilität:* Da Stabilitätsprobleme sehr häufig Kompatibilitätsprobleme sind oder im Zusammenhang mit einer konkreten Installation und Konfiguration auftreten, ist es fast unmöglich, Stabilität sinnvoll in einer Testumgebung zu testen. Deshalb werden normalerweise keine gezielten Tests nach diesem Kriterium vorgenommen, sondern beim Testen aufgetretene Stabilitätsprobleme im Testbericht erwähnt.

*Funktionalität:* Die Funktionalität der unterschiedlichen Produkte lässt sich meist recht einfach anhand der beiliegenden Anleitungen und Produktbeschreibungen feststellen. Ein Vergleich kann tabellarisch erfolgen, indem links alle Funktionen aufgelistet sind und die einzelnen Spalten jeweils für ein Produkt das Vorhandensein dieser Funktion markieren oder nicht. Die Überprüfung der Funktionalität kann recht aufwendig sein, es reicht aber das Testen jeweils einer Eingabe zur reinen Überprüfung der Funktionalität, ein intensiver Test beträfe die Korrektheit der Funktionen.

*Korrektheit:* Da ein formaler Korrektheitsbeweis (Verifikation) bei großen Programmen nicht durchführbar ist (vgl. [PombergerBlaschek 1996], S. 156) und der Quellcode dem Benutzer

[36] Auf die in Abschnitt 3.3 vorgenommene Unterscheidung nach On-access Modus und On-demand Modus wird an dieser Stelle verzichtet, weil die Bestimmung der Kriterien nicht - wie bei quantitativen Kriterien - grundsätzlich unterschiedlich vorgenommen wird. Dies liegt daran, daß die Qualität der in diesem Abschnitt beschriebenen Kriterien in der Regel nicht automatisiert bestimmt werden kann (vgl. Abschnitt 2.3). Dennoch besitzen auch qualitative Kriterien unterschiedliche Ausprägungen je nach Betriebsmodus der Anti-Malware-Software. Deshalb muß auch bei diesen Kriterien die Betriebsart, auf die sich die Testergebnisse beziehen, stets mit angegeben werden.

beziehungsweise Tester von Anti-Malware-Software nicht vorliegt, kann die Korrektheit eines Programms nur durch eine Auswahl von Testfällen ermittelt werden. Ziel ist es, wie bereits mehrfach angedeutet (vgl. Abschnitt 3.2.1, *ideale Testmenge*) die Testfälle so zu wählen, das sie als repräsentativ für die Korrektheit gelten können. Die Korrektheit wird also - zwar mehr oder weniger intensiv - stichprobenartig getestet. Auch für die Korrektheit gilt (wie für die Stabilität), daß sie kaum gezielt getestet wird, sondern Fehler in der Software eher zufällig beim Testen auftauchen und dann berichtet werden.

*Anpaßbarkeit:* Die Installation auf verschiedenen Plattformen, die Einbindung in bestehende Netzwerkstrukturen und die Möglichkeit zur Speicherung von bestimmten Einstellungskombinationen sind Beispiele für gut zu testende Fähigkeiten von Programmen, die die Anpaßbarkeit ausmachen. Das Testen erfolgt in der Regel durch eine Festlegung von durchzuführenden Anpassungen als Testfälle und die Bewertung der einzelnen Programme hinsichtlich der gewählten Testfälle.

*Wartungsfreundlichkeit:* Die Wartung von Anti-Malware-Programmen betrifft hauptsächlich die Durchführung von Updates (Signatur, Engine und Programme). Die Güte der Wartung kann zum einen über die Funktionalität (bezüglich der Durchführung und der Möglichkeiten) von Updates bestimmt werden. Zum anderen spielt die Bedienbarkeit der durchzuführenden Aktionen eine Rolle bei der Bewertung der Wartungsfreundlichkeit.

*Administrierbarkeit und -aufwand:* Gute Administrierbarkeit ist hauptsächlich durch entsprechende Funktionen und eine gute Bedienbarkeit dieser Funktionen gegeben, die Qualitätsbestimmung beider Eigenschaften wurde gesondert betrachtet (s.o.).
Der Administrationsaufwand lässt sich nur schwer im Rahmen eines Softwaretests ausmachen, da viele zeitaufwändige Probleme erst im Laufe der Zeit eines Softwareeinsatzes auftreten. Außerdem ist der Aufwand bei der Administration stark von der Umgebung abhängig, in die die Anti-Malware-Software eingebunden werden muß. Deshalb lassen sich kaum mehr allgemeingültige Aussagen im Rahmen eines Tests von Software machen als die Angabe des Gesamteindrucks der Administrierbarkeit, ergänzt durch besonders auffallende Punkte.