Ausarbeitung im Fach Datensicherheit:

# Viren und Würmer unter UNIX

Hannes Seidel (992075),
Jan Tobias Mühlberg (992024)

10. Februar 2003

*„. . . jaaaa, aber sobald sich Linux etwas mehr verbreitet, wird es da auch ILOVEYOU, usw. geben. . . “*

# Inhaltsverzeichnis

# 1 Vorwort

Dank der Monopolstellung der Microsoft Betriebssysteme im Home- und Officebereich konzentriert sich der gemeine Hacker und Virenbastler auf diese Systeme besonders gern. Da ist es kein Wunder, daß es von Viren, Würmern und Trojanern für solche Systeme nur so wimmelt. Während sich vermutlich jeder Microsoft Nutzer schon einmal als Kammerjäger auf seinem Rechnersystem betätigen mußte, schmunzelt der UNIX Nutzer nur und kommentiert dies kurz mit dem Satz: „Mit einem ordentlichem Betriebssystem wäre das nicht passiert.“ Dieser Satz mag zwar in gewisser Hinsicht richtig sein, verschweigt aber auch die komplette Wahrheit, denn auch die Welt der UNIX Welt ist von Viren nicht verschont geblieben. Zwar kann man diese bisher noch an einer Hand abzählen, doch soll dies für uns kein Grund sein, nicht trotzdem darüber zu berichten.

# 2 Würmer und Trojanische Pferde

Der Apache Web-Server hat sich Dank seiner OpenSource Programmierung sehr stark im Internet verbreitet. Mehr als 60% aller öffentlicher Web-Sites im Internet laufen auf Apache-Basis. Durch die Beliebtheit und durch die vielen neuen Features, die in den Apache integriert wurden, ernannte sich dieser selbst zum Angriffsziel Nummer Eins. Im folgenden ein paar Beispiele, wie der Apache benutzt wurde, um Würmer zu verbreiten und Trojaner zu installieren.

## 2.1 Würmer

Die folgenden beschriebenen Würmer basieren auf der grundlegenden Idee des ersten Web-Wurms: Code Red. Dieser infizierte mehr als 350.000 Webseiten mit Microsoft IIS im Juli 2001 und verbreitete sich dadurch, daß die IIS- Server über Schwachstellen, die es erlaubten, Code auszuführen, kompromittiert wurden.

**ELF/Scalper** Im Juni 2002 wurde der ELF/Scalper Wurm entdeckt, welcher die Apache „HTTP Server chunk encoding stack overflow“ Vulnerabilität ausnutzt, um sich upzuloaden. Es ist ein Internet-Wurm, der sich auf FreeBSD-Systemen mit Apache-Web-Servern verbreitet. Gefahr besteht für Kombinationen aus FreeBSD 4.5 und Apache Versionen 1.3.20 - 1.3.24.
Der Wurm verbreitet sich, indem er systematisch alle IP-Adressen aus zufällig gewählten Klasse-B-Netzwerken überprüft. Wenn er eine gültige IP-Adresse findet, versucht er, sich mit einem Apache-Web-Server auf Port 80 zu verbinden und ihn über die kürzlich entdeckte Schwachstelle in der Kodierung beim Paket-Transfer zu beschädigen.
Ist der Wurm erfolgreich, überträgt er sich in uuencodierter Form zu der Datei „/tmp/.uua“ auf dem Remote-Host und uudecodiert sich in die Datei „/tmp/.a“, die er schließlich ausführt.
Elf/Scalper-A verfügt außerdem über Backdoor-Funktionalitäten auf dem UDP-Port 2001. Durch diese Funktionen kann ein Angreifer von fern verschiedene Formen von Denial-of-Service-Attacken auf andere Computer starten sowie Dateien auf dem Remote-Host nach E-Mail-Adressen durchsuchen, auf Webseiten zugreifen, E-Mails (Spam) versenden, Verbindungen zu anderen Ports öffnen und willkürliche Shell-Befehle ausführen.
Quellen und Weiterführende Informationen: [SARC], [SOPHOS-SC]

**Slapper** Slapper ist ein Netzwerk-Wurm, der sich auf Linux-Maschinen ausbreitet, indem er eine Sicherheitslücke ausnutzt, die im August 2002 in

OpenSSL-Libraries entdeckt wurde.
Der Wurm wurde am Freitag, dem 13. September 2002 spät abends erstmals in Osteuropa entdeckt und ähnelt sehr dem Scalper Apache Wurm. Der Wurm befällt Linux-Maschinen, auf denen der Apache Web-Server mit OpenSSL läuft. Mehr als 60% öffentlicher Web-Sites im Internet laufen auf Apache-Basis. Weniger als 10% davon dürften OpenSSL eingeschaltet haben. SSL (Secure Sockets Layer) wird in der Regel für Online-Geschäfte, Banktransaktionen und private Anwendungen genutzt.
Sobald eine Maschine infiziert wurde, versucht der Wurm sich erneut weiter auszubreiten. Zusätzlich enthält der Wurm Code um ein attackierendes Peer-to-Peer-Netzwerk aufzubauen, in dem infizierte Maschinen ferngesteuert veranlaßt werden, eine große Anzahl von Distributed Denial of Service (DDoS) Angriffe zu starten. Der Wurm breitet sich auf Intel Maschinen mit Linux Distributionen von RedHat, SuSE, Mandrake, Slackware oder Debian aus. Apache ( 1.3.26 - 1.3.6) und OpenSSL ( ¡ 0.96e) werden als Brutstätten genutzt. Am 16.9.2002 9:50 Uhr GMT gibt es aus 100 Ländern Berichte über Rechner, die mit Slapper infiziert sind.
Bis zum heutigen Tage sind fünf verschiedene Varianten des Slapper Wurms bekannt geworden. Die ersten drei unterscheiden sich jedoch nur in den Datei- und Prozeßnamen und den benutzten horchenden UDP Portnummern auf den befallenen Rechnern. Slapper.D ist ein Linux-Trojaner, der sich mit dem IRC-Server irc.zyclonicz.net verbindet (Kanal #devnull). Slapper.E ist eine leicht modifizierte Version, die derzeit beobachtet wird.

**VARIANTE: Slapper.A** Der Wurm infiziert das System, indem er eine uuencodierte Kopie von sich als /tmp/.uubugtraq ablegt. Er decodiert die Datei nach /tmp/.bugtraq.c und benutzt den gcc Compiler, um eine ausführbare Kopie als /tmp/.bugtraq zu erstellen, die dann ausgeführt wird.
Sodann startet der Wurm einen Scan über einen vordefinierten Bereich im Klasse A Netzwerk und sucht nach anfälligen Rechnern, indem er eine Verbindung auf Port 80 (httpd-Server) herstellt. Falls er erfolgreich ist und eine Verbindung zustande kommt, prüft er den Header des Servers auf das Vorkommen des Strings „Apache“. Falls dieses der Fall ist, versucht er eine SSL-Verbindung (Port 443) aufzubauen und falls auch dieses funktioniert, sucht er das System über die OpenSSL-Sicherheitslücke zu infizieren.
Der Wurm enthält ebenfalls eine Backdoor (Hintertür), die den Port 2002 UDP abhorcht und ferngesteuert kontrolliert werden kann. Die Backdoor-Funktion hat die Fähigkeit willkürlich Programme auf dem infizierten Rechner upzuloaden und zu starten. Sie enthält außerdem die Fähigkeit verschiedene Distributed Denial of Service (DDoS) Angriffe zu starten. Diese Back-

door ähnelt sehr der im Scalper Wurm.
Beseitigung: Der Wurm zeigt sich auf dem System als Prozeß mit dem Namen „.bugtraq“. Der Prozeß muß gestoppt werden und die Dateien „/tmp/.uubugtraq“, „/tmp/.buqtraq.c“ und „/tmp/.bugtraq“ müssen gelöscht werden. Der Apache Web-Server muß herunter gefahren w erden und OpenSSL muß mind. auf Version 0.9.6e upgedatet werden, um erneute Infektionen zu vermeiden.

**VARIANTE: Slapper.B** Im Unterschied zu Slapper.A lauscht Slapper.B auf dem UDP Port 4156. Das System ist wahrscheinlich mit dem Slapper.B infiziert, wenn ein Prozeß mit dem Namen '.unlock' läuft. Nach beenden des Prozeßes müssen folgende Dateien gelöscht werden: „/tmp/.unlock“, „/tmp/.unlock.c“, „/tmp/.unlock.uu“, „/tmp/.update.c“ und „/tmp/update“.

**VARIANTE: Slapper.C** Slapper.C lauscht auf UDP Port 1978. Folgender Prozeß muß beendet werden: '.cinik' Danach müssen die Dateien „/tmp/.cinik“, „/tmp/.cinik.c“ und „/tmp/.cinik.uu“ gelöscht werden.

**Sicherheits-Hinweise von Linux-Distributionen**

| | |
|---|---|
| Debian | http://www.debian.org/security/2002/dsa-136 |
| Mandrake | http://www.mandrakelinux.com/en/security/2002/MDKSA-2002-046.php |
| RedHat | http://rhn.redhat.com/errata/RHSA-2002-155.html |
| SuSE | http://www.suse.de/de/security/2002_027_openssl.html |

**Vorbeugende Maßnahmen** Um in Zukunft einem Befall ähnlich aufgebauter Würmer entgegenzuwirken, hier ein paar Tips:
Legen Sie keine Kopien ihres Compilers (z.B. gcc) auf ihren Web-Server. übersetzen sie ihre Applikationen auf geschützten Rechnern und überspielen sie diese dann auf den Web-Server. Ohne den „gcc“ Compiler kann Slapper ihr System nicht infizieren.
Wenn sie unbedingt einen Compiler auf ihrem Web-Server haben müssen, verbieten sie dessen Ausführungsrechte für unpriviligierte Nutzer. Apache läuft normalerweise als Benutzer „nobody“; dies ist auch die UserID, welche Slapper nutzt, um ins System einzubrechen.
Lassen sie den WebServer in einer „chroot“ Umgebung laufen. Dies minimiert den Schaden, den der Wurm anrichten kann, da er nur einen kleinen Teil des Festplatteninhalts zu sehen bekommt. Ein „chrooted“ Slapper kann nicht die komplette Festplatte nach E-Mail Adressen scannen.
öffnen Sie keine Ports in ihrer Firewall, die sie nicht benutzen. Wenn Sie kein SSL nutzen, wird ein blockieren von Port 443 Slapper gar nicht erst ins

System lassen. Durch blockieren von Port 2002 (Slapper.A) kann der Wurm nicht mehr über seine Backdoor Funktion kontaktiert werden.
Gehen Sie sicher, daß alle Prozesse, die auf ihrem Rechner laufen, auch wirklich die Prozesse sind, die sie für den Betrieb benötigen.
Lesen sie die Security Announcements. Sie geben Hinweise auf mögliche Gefahren und bieten auch oftmals gleich den richtigen Link zum patchen einer anfälligen Programmversion.
Quellen und Weiterführende Informationen: [SOPHOS]

## 2.2 Trojaner

Neben der Verbreitung über WebServer, wie dem Apache, können sich Viren unter UNIX- artigen Betriebssystemen auch auf herkömmliche Art und Weise verbreiten. So das folgende Beispiel.

**Linux Remote Shell Trojaner (kurz RST)** Seine Aktivierung über einen beliebigen UDP-Port macht ihn besonders gefährlich. Der Remote Shell Trojaner RST.b, es gibt zur Zeit zwei Arten (RST.a und RST.b), gelangt als E-Mail-Attachment oder über das Internet in herunter geladenen Dateien auf den Rechner und verbreitet sich selbständig weiter. Auf befallenen Systemen öffnet der RST.b eine Hintertür, woraufhin das System auf Datenverkehr auf beliebigen UDP-Ports wartet. über diese Hintertür kann er nun beim Empfang spezieller UDP-Pakete eine TCP Verbindung zum Angreifer aufbauen und eine Shell für diesen starten. Jeder beliebige Befehl kann nun auf dem Wirt ausgeführt werden. Dies hat verheerende Folgen. Sichere Daten können ausspioniert, geändert oder gelöscht werden. Inwieweit über den Trojaner Root-Rechte auf dem System erlangt werden können ist noch unbekannt.
Außerdem kann RST.b sich selbst ständig vervielfältigen, wodurch die Wahrscheinlichkeit besteht, daß er sich über Binär-Dateien auf dem infizierten Host verbreitet - eine Funktion, die es trojanischen Pferden und Viren schon früher ermöglicht hat, auch andere Betriebssysteme wie etwa Microsoft Windows zu befallen. Zusätzlich erhöht wird das Gefahrenpotenzial dadurch, daß RST.b Hackern die Möglichkeit bietet, das Internet nach infizierten Systemen zu durchsuchen, wodurch die Verbreitungsgeschwindigkeit und die Möglichkeit eines Angriffs weiter steigt.
Unter https://www.qualys.com/forms/remoteshellb.html werden kostenlos Tools zur Erkennung von RST.b und zur Säuberung infizierter Systeme angeboten.
Quellen und Weiterführende Informationen: [QUALYS]

# 3 File- und Bootviren, E-Mail Würmer

Nachdem der vorherige Abschnitt dieser Ausarbeitung sich mit Würmern, die beispielsweise Web-Server oder vergleichbare Dienste befallen, auseinandergesetzt hat, sollen in diesem Teil unserer Ausarbeitung die von E-Mail-Würmern bzw. konventionellen File- und Bootviren ausgehenden Bedrohungen für die Nutzer unixoider Betriebssysteme etwas unter die Lupe genommen werden. Vor allem die E-Mail-Würmer erfreuen sich derzeit unter einem bekannten und sehr populären, von einem in Redmond beheimateten Softwaregiganten herausgebrachten Betriebssystem einer ganz erschreckenden Beliebtheit, wohingegen die Relevanz der File- und Bootviren hier stark zurück gegangen zu sein scheint.
Wie sieht es also mit solchen Bedrohungen bei der Nutzung von UNIX-ähnlichen Systemen aus? Ist sie überhaupt Vorhanden? Wann immer diese Frage irgendwo im Internet gestellt wird, melden sich sofort Stimmen mit der Behauptung, der offensichtliche Mangel an Viren für Systeme wie Solaris, HPUX oder auch Linux läge vor allem in deren geringer Verbreitung und mit dem Vermehrten Einsatz von Linux würde sich das schon ändern, eine Behauptung, die sich bereits damit entkräften läßt, daß unixoide Systeme zwar derzeit auf Desktop-Computern nur eine nominale Verbreitung aufweisen, jedoch in der Schicht der Server, also der Schicht von Computern, die im Internet Dienste wie beispielsweise E-Mail, FTP oder HTTP anbieten, dominierend sind. Ein Angreifer, der am Verursachen von Schaden oder an der Informationsbeschaffung interessiert ist, hätte in einem solchen System, so es denn Angriffspunkte gäbe natürlich den idealen Punkt zur Durchsetzung seiner Ziele gefunden. Anstatt also jeden einzelnen Arbeitzplatzrechner zu penetrieren um eine möglichst große Schadwirkung zu entfalten, könnte ein geschickter Programmierer hier mit einem Schlag gleich hunderte, wenn nicht tausende Nutzer vom Zugriff auf ihre eigenen Daten abhalten. Warum kommt es nun also nicht häufiger zu solchen Disastern? Sind die Systeme zu sicher? Wie funktionieren beispielsweise E-Mail-Würmer überhaupt und unter welchen Umständen können sie Schaden anrichten?

## 3.1 Grundsätzliches

Natürlich sind auch die unixoiden Betriebssysteme mit all den von ihnen bereitgestellten Diensten nicht so sicher, als daß überhaupt keine Möglichkeiten für eine Penetration bestünden. Dementsprechend hat auch der durchschnittliche Viren-Entwickler durchaus Chancen, den Einen oder Anderen Volltreffer zu landen.

## 3.2 Bootviren

Eine inzwischen scheinbar dem Aussterben nahe Virenfamilie sind die Bootviren. Sie werden bereits beim Booten eines Rechners aktiv weil ihr Code üblicher Weise aus dem Bootsector des Datenträgers gelesen wird, von dem der Rechner startet. Dies funktioniert unter PC-Betriebssystemen sehr gut weil hier besagter Bootsector bereits Teil des Betriebssystem-Codes ist. Ist das Virus dann gestartet, kann es nach belieben alle weiteren in die Laufwerke des Computers eingelegten und schreibbaren mobilen Datenträger wie beispielsweise Disketten infizieren, d.h. seinen eigenen Code in den Bootsector des jeweiligen Datenträgers schreiben. Um das Virus nun auf einen weiteren Rechner zu übertragen, muß der bislang nicht infizierte Rechner zumindest einmalig von einer infizierten Diskette starten, damit das Virus lokale Datenträger infizieren kann. So weit zur Theorie.
Leider funktioniert dieses Prinzip bei unixoiden Betriebssystemen nicht: Hier (Beispiel: LINUX) wird der Bootprozeß von einer kleinen Software im Bootsector, dem Boot-Manager eingeleitet. Dieser startet wiederum den Kernel, also das eigentliche Betriebssystem, daß jedoch nicht darauf aufbaut sondern eine eigene Speicherverwaltung, ein eigenes Prozeßmanagement und eine eigene Systemkomponentenverwaltung nutzt. Das Ergebnis dessen ist, daß der Virus-Code nicht weiter ausgeführt wird.
Andere UNIXe gehen hier ähnlich vor oder nutzen gar keinen Bootsector sondern starten den Kernel des Betriebssystems direkt über eine BIOS-Routine, so daß ein Bootvirus gar keine Angriffsfläche hat.
Bei x86- oder m68k-basierten unixoiden Betriebssystemen hat eine Infektion mit einem Bootvirus üblicher Weise das nicht mehr Funktionieren des Bootvorgangs zur Folge weil der Bootsector auf ungeeignete Art und Weise verändert wird - das Bootvirus ging von einem „normalen“ Betriebssystem aus. Schutzmechanismen hiergegen gibt es keine. Es liegt beim Nutzer bzw. Administrator, nicht von unsicheren Datenträgern zu booten.

## 3.3 Fileviren

Fileviren, ebenfalls eine recht alte Virenfamilie, haben in der Vergangenheit der PC-Entwicklung zweifellos eine große Rolle gespielt. Nahezu alle PC-Betriebssysteme hatten mit dem Problem zu kämpfen: Der Virus-Code befindet sich hierbei, meistens eingebettet in nutzbringenden Code, in einem Ausführbaren Programm. Dieses benimmt sich wie vor der Infektion mit einem Virus, führt jedoch unbemerkt zusätzliche Funktionen aus. So wird üblicher Weise der Virus verbreitet oder führt eine Schadensfunktion aus. Dies war unter PC-Betriebssystemen auch nie ein Problem - die Systeme

waren bis vor kurzem ausschließlich für den Einzelnutzerbetrieb gedacht und der einzelne Nutzer hatte freien Zugriff auf alle physikalischen und logischen Komponenten des Systems. Damit ist es lediglich eine Frage der Zeit, bis jedes ausführbare Programm auf einem System infiziert ist. Da dies auch Programme auf eingelegten mobilen Datenträgern einschließt, ist auch die Übertragung zum nächsten System gesichert.
Echte Probleme haben Fileviren mit Systemen, die eine strikte Trennung der einzelnen Nutzer durchführen. Hier hat ein Virus zwar immer noch die Möglichkeit einzelne Programme zu infizieren, jedoch nur solche auf die der aktive Nutzer Schreibzugriff hat - und davon sollte es nur sehr wenige geben. All die anderen Systemkomponenten kann er sich zwar anschauen, jedoch nicht modifizieren. Ebenso arg eingeschränkt sind auch seine Möglichkeiten, Schaden anzurichten. Die werden nämlich durch die gleichen, oben genannten Zugriffsschutzmethoden auf den Datenbestand des Nutzers beschränkt.
Während die Mehrbenutzerfähigkeit, damit verbunden die Aufteilung der einzelnen Privilegien, in die Welt der PC-Betriebssysteme erst vor einigen Jahren Einzug hielt gehört dies bereits seit Jahrzehnten zum Standard-Repertoire der UNIX-artigen Systeme. Dies hatte zur Folge, daß sich die Fileviren hier nie in der Form entwickelt haben, wie sie in anderen Welten üblich waren. Nach umfangreichen Recherchen im Internet und in alten Zeitschriften, war es uns möglich, ganze drei Fileviren ausfindig zu machen. Zwei davon sind etwa 20 Jahre alt, liefen unter MVS und breiteten sich auch hier aufgrund o.g. Gründe nicht aus. Der Dritte hat erst drei Jahre auf seinem angeblich etwa etwa 500 Bytes großen Buckel, wurde jedoch nur zu Testzwecken unter Linux entwickelt, besitzt keinen Schadensfunktion und kam nie in Umlauf. Im übrigen ist er als echtes Open-Source Projekt natürlich auch in Quellen öffentlich verfügbar.
Letztlich zielen alle diese Viren lediglich auf die Dummheit eines Administrators ab. Solange niemand seinen Root-Account für nichtadministrative Aufgaben mißbraucht oder sonst irgendwie grob fahrlässig mit diesen doch fast Gott-gleichen Privilegien umgeht, ist ein unixoides Betriebssystem im ganzen Sicher vor Angriffen durch Fileviren. Die Infektion der Dateien einzelner Nutzer mit Shell-Account bleibt weiterhin möglich, setzt jedoch ebenfalls einen erhöhten Aufwand an Dummheit voraus. Schließlich gibt es heute vor allem im Open-Source Bereich genügend Möglichkeiten, die Authentizität und Korrektheit der eingesetzten Software zu überprüfen, insbesondere weil ein Großteil der Anwender und Administratoren Wörter wie „Raubkopie“ erst in einem Fremdwörterbuch nachschlagen müßten um deren Bedeutung zu begreifen.
Wieauchimmer: Weil ja unter unixoiden Systemen alles ganz hervorragend dokumentiert ist, gibt es hier auch hervorragende Informationen über die Im-

plementation von Fileviren, auf die ich an dieser Stelle verweisen möchte. Das Standardwerk hierfür ist die „Linux Virus Writing HOWTO“. Das Dokument wird im Quellenverzeichnis unter dem [LINVIR] aufgelistet.

## 3.4 E-Mail Würmer

E-Mail Würmer sind eine ganz beängstigende Erfindung der PC-Neuzeit. Die Möglichkeit, ausführbaren Code in eine E-Mail zu stecken ist nicht wirklich schlimm, vielmehr oftmals ganz nützlich, die Tatsache jedoch, daß es MUAs[1] gibt, die diesen Code nach einem einzelnen Mausklick bzw. ganz ohne jede Bestätigung ausführen ist schlichtweg erschreckend und völlig unverständlich.
Während sich viele UNIX-Nutzer bereits über E-Mails aufregen, die HTML-Code und das noch verhältnismäßig harmlose JavaScript enthalten, einfach nur, weil es ohne von einem Web-Browser interpretiert zu werden, oftmals nicht lesbar ist und gegen die Netiquette verstößt, müssen sich Nutzer gängiger PC-Betriebssysteme damit abfinden, daß der mit dem System mitgelieferte MUA die gesamte Darstellungslogik des ebenfalls mitgelieferten Web-Browsers nutzt und man dies auch nicht abschalten kann. Erwähnenswert ist auch, daß es unter UNIXen nichts mit ActiveX[2] vergleichbares gibt. Auch wenn hier viele mächtige Skriptsprachen verfügbar sind, kam noch kein Programmierer auf die Idee, eine davon dahingehend in einen MUA zu verbauen, als daß in der Mail vorhandener Code ausgeführt wird und beliebig unsinnige und schadhafte Dinge auf einem System machen kann. Folglich funktionieren alle derzeit verfügbaren E-Mail Würmer unter UNIX-ähnlichen Betriebssystemen nicht.
Was es jedoch für diese Systeme gibt, sind Virenscanner, die Mails auf dem Server, also dem UNIX-Host nach Viren durchsuchen, bevor diese von Endnutzern mit unsicheren MUAs abgeholt und gelesen werden.
Abschließend hier noch ein E-Mail Wurm, den ich vor einigen Wochen erhalten habe:

```
Date: Mon, 3 Feb 2003 12:29:33 +0100
From: xxxxxxxx xxxxxxxx <xxxxxxxx@xxxxxxxxxxxxxxxxxxx>
Subject: so gehts...
To: Jan Tobias Muehlberg <muehlber@fh-brandenburg.de>
User-Agent: Mutt/1.2.5i

Hallo
```

[1] Mail User Agent, Mail Reader
[2] eine Skriptsprache von Microsoft

```
Ich bin ein Linux-Virus. Leider habe ich keine Moeglichkeit
direkt auf Ihr System zuzugreifen. Ich bitte Sie hiermit,
bitte bitte eine weitere Shell aufzumachen, sich mit 'su -'
Root-Rechte zu verschaffen, dort 'rm -rf /' einzugeben, und
aeh vorher noch diese Mail an alle Leute aus ihrem Adressbuch
weiterzuleiten. Vielen Dank fuer Ihre Kooperation!
```

## 3.5 Fazit

Wer unixoide Systeme einsetzt hat mit klassischen Viren und E-Mail Würmern keine Probleme, nicht heute und voraussichtlich auch nicht in der Zukunft. Allenfalls muss er sich mit Sicherheitslöchern in diversen Bibliotheken und Servern herumschlagen, die dann wiederum von Würmern genutzt werden könnten. Die gibt es aber bei anderen Systemen ganz genauso.
Ein kaum zu überschätzender Pluspunkt ist hierbei die Tatsache, daß fast alle UNIXe im Quellcode zugänglich sind, im Falle von LINUX sogar auf einem Open-Source Entwicklungskonzept basieren. Allein der Fakt, daß sehr viele Leute die Möglichkeit haben, den Quellcode einer Applikation einzusehen, sollte die Entwickler zu sauberer und gewissenhafter Arbeit zwingen. Abgesehen davon sehen viele Augen natürlich immer mehr Schwachstellen als nur einige wenige.
Auch die Offenlegung möglicher Sicherheitslücken und Programmierfehler ist hierbei ein außerordentlich wichtiger Aspekt: Es soll Softwareproduzenten geben, die Sicherheitslücken erst dann publizieren, wenn der entsprechende Patch erschienen ist. Ein Anwender allerdings, der über bekannte Probleme mit seinem System in Unkenntnis gelassen wird kann sich kaum vor Angriffen schützen.
Die Frage, ob nun der Einsatz eines unixoiden Systems sinnvoller als der eines typischen PC-Betriebssystems ist, ist und bleibt in den meisten Fällen eine Marketing-Entscheidung, auch wenn aus technischer Sicht die Frage gar nicht erst gestellt werden dürfte.

# 4 Quellen

*Folgende literarische Quellen und Web-Dokumente wurden zur Erstellung dieser Ausarbeitung verwendet:*


| | |
|---|---|
| [LINVIR] | Web-Dokument: Linux Virus Writing HOWTO<br>http://www.lwfug.org/ abartoli/<br>virus-writing-HOWTO/_html/ |
| [QUALYS] | Web-Dokument: http://www.qualys.com |
| [SARC] | Web-Dokument: http://www.sarc.com/avcenter/venc/data/<br>freebsd.scalper.worm.html |
| [SOPHOS] | Web-Dokument: http://www.sophos.com/virusinfo/analyses |
| [SOPHOS-SC] | Web-Dokument: http://www.sophos.de/virusinfo/analyses/<br>elfscalpera.html |