Ihr Linux-Rechner wurde assimiliert – ist Widerstand zwecklos?

ROOTKITS UNTER LINUX

Sie sind eines der beliebtesten Hilfsmittel der Computer-Hacker: Rootkits – die mächtigsten und tückischsten aller Trojaner.[1] *Rootkit* bedeutet soviel wie „Administratorenausrüstung", also eine Art Ausstattung an Softwarewerkzeugen, die von Dritten unrechtmäßig und meist unbemerkt in ein Computersystem eingeschleust werden, um zukünftige Logins des Eindringlings zu verbergen, Prozesse zu verstecken, Daten zu kopieren und Eingaben mitzuverfolgen.

Der Einsatz eines Rootkits erfolgt erst nach einem geglückten Einbruch, indem es seine Existenz verschleiert sowie entsprechend einhergehende Tätigkeiten des Systems vor dem eigentlichen Systemadministrator verbirgt. So sorgt es beispielsweise dafür, dass der Eindringling auch zu einem späteren Zeitpunkt wieder Root-Status (gleichbedeutend mit Administratoren-Status) auf dem System erhält. Dazu eröffnet es dem Hacker so genannte *Backdoors*. Diese „Hintertüren" lassen sich in zwei Varianten unterteilen: Zum einen in lokale Backdoors auf dem System – diese setzen die interaktive Präsenz des Angreifers auf dem geknackten System voraus – und zum anderen in Netzwerk-Hintertüren – z.B. ein offener Port, über den der Hacker von einem anderen Rechner aus privilegiert in das System einsteigen kann. Rootkits sind der ultimative Schrecken eines jeden Systemverantwortlichen.

Während unter Windows Trojaner und Rootkits derzeit zu boomen scheinen (siehe dazu auch Artikel *Ungebetene Gäste: Trojaner am Windows-PC* in *Comment 04/1* bzw. unter `www.univie.ac.at/comment/04-1/041_10.html`), ist es um die Rootkits unter Unix etwas ruhiger geworden – aber der Schein trügt. Erst kürzlich wurden zwei neue Rootkits (Phalanx und eNYeLKM; Näheres dazu im Abschnitt *Schau trau – wem?*) für aktuelle Linux-Versionen veröffentlicht, die selbst von den besten Suchwerkzeugen nicht erkannt werden.

Der Angriffszyklus

Bevor ein Hacker sein Rootkit installieren kann, muss er sich zunächst Root-Rechte auf einem System verschaffen. Der dazu notwendige Einbruch und die folgenden Schritte laufen meist nach einem festen Schema ab: Zunächst wird der Angreifer so viele Informationen wie möglich über das potenzielle Opfer beschaffen. So sucht er zuerst nach Diensten, die vom System angeboten werden. Ein so genannter Port-Scan des Systems zeigt in der Regel rasch, welche dieser Dienste offen sind. Die Scan-Methoden sind dabei je nach Vorlieben der Hacker sehr unterschiedlich: Manche gehen direkt vor, andere setzen lieber Täuschungsmanöver ein, um auch geübte Systemverantwortliche auszutricksen.

Hat der Hacker Schwachstellen gefunden – meist sind dies Systeme mit unzureichender Wartung – führt er seinen Angriff durch. Dabei verwendet er für das jeweilige System und seine Schwachstellen geeignete, hoch spezialisierte Programme, die ihm den Einstieg in das System ermöglichen. Mit der Aneignung der nötigen Rechte ist es nunmehr für den Hacker wichtig, die Spuren des Einbruchs sofort zu verwischen, damit der eigentliche Systemverantwortliche nichts bemerkt. Oft sind in Rootkits bereits entsprechende Werkzeuge zum Bereinigen von Log-Dateien enthalten. Danach richtet sich der Hacker auf dem Rechner häuslich ein und installiert sein Rootkit. Mit diesem Schritt ist er wieder bereit, sein Unwesen an einem neuen Ort zu treiben – denn häufig sind die befallenen Systeme (z.B. die mit schnellem Internetzugang ausgestatteten Rechner an der Universität Wien) nicht selbst das Ziel, sondern nur ein Zwischenwirt für den Hacker zur Verschleierung seiner Spuren.

Fallbeispiel

Das Institut X leistet sich zur lokalen EDV-Unterstützung einen größeren SuSE Linux-Server mit einem lokalen SSH (*Secure Shell*)-Zugang, einem Web-Server sowie einem Windows Fileservice mittels Samba. Der Server steht nicht hinter einer Institutsfirewall, ist aber selbst durch eine IP-Tables Firewall geschützt. Es ist Ferienzeit. Der verantwortliche Administrator, eigentlich Wissenschaftler, ist auf wohlverdientem Urlaub. Während er die freie Zeit genießt, wird ein Sicherheitsproblem bei SuSE publiziert, was dem unbeaufsichtigten Server zum Verhängnis wird. Prompt wird die Lücke im System von einem Hacker entdeckt und genutzt.

Zunächst verschafft sich der Hacker unprivilegierten Zugang zum Server. Das schlecht gewählte Root-Passwort[2] ist durch Probieren rasch geknackt. Beobachtungen des Systems zeigen dem Hacker, dass kein Administrator am System aktiv ist und er in Ruhe schalten und walten kann.

Dazu vernichtet er zuerst jegliche Informationen, die seinen Einbruch enttarnen würden. Dann schließt er die Sicherheitslücke des Webservers – kein zweiter Hacker soll ihm den eroberten Platz streitig machen. Er besorgt sich das Rootkit Adore-ng und installiert es auf dem System. Nur im kurzen Zeitraum des Bootens wären Spuren des Einbruchs

1) Als Trojanische Pferde oder kurz Trojaner bezeichnet man im Computer-Jargon schädigende Programme, die als nützliche Programme getarnt sind oder zusammenhängend mit einem nützlichen Programm verbreitet werden, aber tatsächlich auf dem Computer im Verborgenen unerwünschte Aktionen ausführen können.

2) Tipps zur Wahl eines sicheren Passworts sind unter `www.univie.ac.at/ZID/passwort/` zu finden.

sichtbar – aber da schaut keiner hin. Die Tarnung ist perfekt. Die IP-Tables Firewall wird für einen weiteren Port geöffnet und eine zweite SSH dahinter versteckt. Der Prozess ist unsichtbar, die Log-Dateien, die der zweite `ssh`-Daemon herstellen würde, werden dank Adore-ng nicht geschrieben. Da das System nicht hinter einer Institutsfirewall steht, funktioniert der Zugang zur SSH auch über den neuen, getarnten Port.

Als der Administrator aus dem Urlaub kommt, ist das System in perfektem Wartungszustand. Allerdings ist genau das verdächtig. Ihm bleibt nichts anderes übrig als zu suchen. Tage vergehen – keine Spur. Der Hacker hat perfekt gearbeitet. Panik beim Administrator, der eigentlich eine Tagung vorbereiten müsste. Zerknirscht muss er seinen TeamkollegInnen eingestehen, dass er die Situation nicht im Griff hat.

Während eine weitere Woche vergeht, kopiert der Hacker größere Datenmengen auf den Server und versteckt sie mittels des Rootkits. Daraufhin muss der Administrator feststellen, dass 40% des Plattenplatzes verbraucht sind und das System sehr langsam ist. Allerdings gibt es keinen sichtlichen CPU-Verbrauch.

Nun ist eine genaue Systemanalyse nicht mehr aufzuschieben. Erst bei einer sehr detaillierten Untersuchung treten die Probleme sichtbar hervor. Illegale Videokopien, Musikstücke etc. hat der Hacker auf dem Server abgelegt – und vermutlich weiterverkauft. Eine forensische Analyse der Festplatte zeigt, welche Schritte der Hacker anfangs machte. Das Rootkit wird deaktiviert. Der zweite `ssh`-Daemon wird belassen – es soll herausgefunden werden, von wo der Hacker kommt. Nach dem Start des Rechners läßt dieser auch nicht lange auf sich warten, kommt danach aber nie wieder. Nur die Kunden wollen noch längere Zeit Daten vom System holen. Die Rechnerquelle, von welcher der Hacker kam, liegt irgendwo in Taiwan. Eine Rechtsverfolgung dorthin ist leider unmöglich.

Vorbeugung

Um nicht selbst einen derartigen Eingriff zu erleben und Hackern nicht ahnungslos ausgeliefert zu sein, ist Vorsorge der beste Schutz! Es sollte erst gar nicht soweit kommen, dass ein System von Eindringlingen geentert und übernommen wird. Frei nach Raumschiff Enterprise: Widerstand ist nicht zwecklos! Die besten Waffen hierfür sind:

- **Minimieren der möglichen Angriffsfläche:** Nicht benötigte Services gehören abgeschaltet. Verwenden Sie eine strikte Politik: Nur das ist erlaubt, was wirklich benötigt wird.

- **Eingeschränkter Zugriff auf das System:** Meist muss nicht jeder Dienst aus der ganzen Welt erreichbar sein. Oft ist die Einschränkung auf den Netzwerkbereich der Uni Wien (131.130.0.0/16) und einige externe Internetadressen völlig ausreichend. Dazu hilft Firewalling (Konfigurationstipps hierzu sind unter `www.univie.ac.at/ZID/anleitungen/ip-tables/` zu finden).

- **Das System aktuell halten:** Hier sind vor allem die Dienste angesprochen, die dem Netzwerk zur Verfügung gestellt werden.

- **Das System regelmäßig auf Rootkits überprüfen:** Ist ein System erst einmal geentert worden, kann man sich nur noch durch eine komplette Neuinstallation und ein kritisches Überdenken der angewandten Sicherheitsstrategien verlässlich Ruhe verschaffen.

- **Den Feind kennen lernen:** Das Verhalten von Rootkits, ihre Stärken und Schwächen, lernt man am besten zu verstehen, wenn man sich selbst Rootkits besorgt und sie untersucht.

Jeder, der an einem Linux-Rechner mit Netzwerkanschluss arbeitet bzw. diesen betreut, sollte sich mit Rootkit-Versionen und deren Eigenschaften näher vertraut machen. In welcher Form Rootkits vorgehen und welche Maßnahmen zum Schutz eines Systems getroffen werden können, ist im Folgenden detailliert beschrieben.

Schau trau – wem? Rootkit-Versionen, ihre Vorgehensweise und Gegenmaßnahmen

User Mode-Rootkits

Ist ein Rootkit einmal installiert, hat der Hacker oft leichtes Spiel. Je nach Art des Rootkits werden entweder die Ausgaben der Systemprogramme so modifiziert, dass zu versteckende Dateien oder zu verschleiernde Aktivitäten nicht angezeigt werden, oder es wird im Systemkern deren Ausgabe verhindert. Erstere sind die klassischen Rootkits – die so genannten *User Mode-Rootkits*. Das bedeutet, dass z.B. die Befehle `ls`, `du`, `df` oder `find`, aber auch `ifconfig`, `netstat`, `ps`, `top` etc. einfach falsche Ausgaben anzeigen. Solche Programme laufen – auch unter root – immer im *User Mode*[3] ab, daher ihr Name. So kann zum Beispiel auch das Programm `kill` so modifiziert werden, dass bestimmte Programme nicht ohne weiteres gestoppt werden können, obwohl der Systemadministrator als root den Befehl absetzt. Hintertüren zum Eindringen schafft das Rootkit mit trojanisierten Versionen von `login`, `sshd` oder `xinetd` etc. Etwaige unerwünschte Log-Einträge werden häufig durch einen modifizierten `syslog`-Daemon verhin-

3) In Unix und Unix-ähnlichen Betriebssystemen wie Linux ist der Kernel für alle privilegierten Operationen (Ein- und Ausgabe, Verwalten von Prozessen usw.) zuständig. Solange Prozesse keine solchen privilegierten Operationen brauchen, laufen sie im *User Mode*; wenn ein Prozess z.B. Ein- oder Ausgabeoperationen durchführt, fordert er über genormte Schnittstellen, so genannte *System Calls*, die erforderlichen Dienste vom Kernel an und läuft für die Dauer der Operation im *Kernel Mode*.

dert. Damit ein Hacker seinen Einfluss auf das System und seine Umgebung ausweiten kann, sind zusätzlich häufig auch Netzwerk-Sniffer im User Mode-Rootkit enthalten. Damit lassen sich Username-/Passwort-Kombinationen auf der Ebene des Netzwerkverkehrs ausspähen. Jede Autorisierung, die unverschlüsselt erfolgt, ist davon betroffen.

Da Unix im Allgemeinen ein offenes Betriebssystem ist und viele Teile der Programmquellen ohnehin öffentlich sind, ist die nachträgliche Trojanisierung eines Programms kein Problem. Wichtig für den „Hersteller" eines Rootkits ist nur, dass er dem Systemadministrator eine konsistent falsche Sicht der Vorgänge liefert. Daher muss für User Mode-Rootkits oft eine große Anzahl von Programmen des Betriebssystems modifiziert werden. Wird ein Programm oder ein Shell-Befehl übersehen (gerne z.B. die *Shell-expansion*), so hat der Systemadministrator noch die Möglichkeit, etwas per Zufall zu entdecken (oft hilft `echo *` bei User Mode-Rootkits statt eines `ls`- oder `find`-Befehls). Ein findiger Systemadministrator sollte sich vor dieser Art von Betrug so sichern, dass er ausschließlich eigene Kopien der Originalprogramme verwendet. So sollte er beispielsweise bei der Installation des Systems die Systemprogramme kopieren und auf eine CD brennen. Will er später sichere Programme verwenden, dann kann er mittels eines `mount`-Befehls diese CD an einem geeigneten „Mountpoint" (z.B. in `/mnt/cdrom`) einhängen und mit `chroot /mnt/cdrom` seiner Shell bekannt geben, dass er die Programme von der CD verwenden möchte. Allerdings kann auch `chroot` trojanisiert sein.

User Mode-Rootkits gehen zurück auf das Jahr 1989. Damals beschränkte man sich auf das Modifizieren der System-Logeinträge (`utmp`, `wtmp` und `lastlog`). Damit konnte ein Angreifer mittels der Befehle `who`, `w` oder `last` nicht gesehen werden; allerdings konnte man mittels der Prozessliste `ps` sehr wohl Befehle bei deren Ausführung erkennen. Deswegen sagen viele, dass Rootkits 1994 entstanden sind: Damals wurden außerdem die ersten Werkzeuge so umgeschrieben, dass inkludierte Netzwerk-Sniffer unverschlüsselte Netzwerkdaten unbemerkt analysieren und Username-/Passwort-Kombinationen protokollieren konnten. Das älteste Linux-Rootkit dürfte am 11. Oktober 1994 entstanden sein. Es beinhaltete trojanisierte Versionen der Programme `ps`, `netstat` und `login` – letzteres mit Hintertür zum Anmelden als Administrator. Der Begriff *Rootkit* wurde etwa 1995 geprägt. Zwei sehr bekannte Vertreter der User Mode-Rootkits sind LRK (*Linux RootKit*) und T0rnkit, welches ab März 2001 vom „Lion"-Wurm verwendet wurde.

Kernel Mode-Rootkits

Die andere – modernere, effizientere – Art der Rootkits sind die ***Kernel Mode-Rootkits***. Diese Rootkits modifizieren die Daten vor der Ausgabe auf der Ebene des Systemkerns. Sie filtern ungewünschte Informationen heraus, bevor sie allen User Mode-Programmen zur Verfügung stehen. Mit einem `ls`- oder `find`-Befehl erhält man die versteckte Information nicht, ebenso bekommen Programme wie `ps` oder `top` den versteckten Prozess nicht zu Gesicht. Relevante neue Log-Einträge werden gar nicht an den zuständigen Daemon geliefert. Aber auch Spuren des Einbruchs, sichtbar z.B. in den Logfiles `utmp`, `wtmp` oder in `messages`, können vor dem Systemverantwortlichen geheim gehalten werden. Die Daten stehen zwar auf der Festplatte, können aber nicht angezeigt werden. Die Täuschung eines guten Kernel Mode-Rootkits ist komplett. Kernel Mode-Rootkits können zudem noch mehr: Sie sind in der Lage, die Ausführung von Programmen umzuleiten, sodass anstelle eines Programms ein anderes unbemerkt zur Ausführung gelangt.

Der einfachste Weg, einen Systemkern zu ändern, wird über dynamisch ladbare Module eingeschlagen. Jedes moderne Betriebssystem hat derartige Methoden, um seine Funktionalität während der Laufzeit zu erweitern. Ältere Unix-Systemkerne konnten hingegen nur durch Neuübersetzen und einen Neustart geändert werden.

Das Ziel eines Kernel Mode-Rootkits ist, den trojanisierenden Programmcode im Bereich des Systemkerns unterzubringen, was gleichbedeutend mit einer Funktionsänderung des Betriebssystems ist. Folgende grundsätzliche Methoden stehen dafür zur Verfügung:

- **Ladbare Kernel-Module (LKMs):** LKM-Rootkits ersetzen im Allgemeinen Systemaufrufe des Betriebssystems. Damit werden zum Beispiel die Funktionen zum Anzeigen der Prozessliste, der Liste der Dateien zum Öffnen, Lesen oder Schreiben von Dateien etc. so geändert, dass zu versteckende Informationen herausgefiltert werden. Die neuen Module werden in der so genannten *System Call Table* eingetragen. Dieses Problem haben die Linux-Kernel-Entwickler erkannt und mit den Änderungen zum Kernel 2.6 erheblich erschwert. Ein anderer, modernerer Weg führt über das *Virtual File System* (VFS). Neben dem Laden eigener neuer Kernel-Module ist es auch möglich, existierende, vertrauenswürdige, immer vom System verwendete Kernelmodule zu infizieren. Damit bleibt das Rootkit genauso unsichtbar und wird beim Neustart des Systems immer mit dem anderen Modul mitgeladen. Beispiele für LKM-Rootkits sind Knark, Adore, Adore-ng, KIS und – neu – eNYeLKM.

- **Patchen des laufenden Kernels (Modifikation des Speichers)**: Diese Rootkit-Art basiert auf der Änderung des Kernel-Abbilds (*Image*) im Speicher des Systems, welcher durch `/dev/kmem` repräsentiert wird. Rootkits dieser Art kommen ohne ladbare Kernel-Module aus und funktionieren direkt. Allerdings hat die Entwicklung rund um den Kernel 2.6 Rootkits dieser Art schwer behindert, da `/dev/kmem` nicht mehr zur Verfügung steht. Ein Einfügen eines Rootkits über `/dev/mem` ist erheblich komplizierter. SuckIT, das *Super User Control Kit*, basiert auf einer Änderung in `/dev/kmem` und ist nur noch auf älteren Linux-Kernels funktionstüchtig. Neu seit Herbst 2005 ist das Rootkit mit dem Namen Phalanx, welches auf dem Einfügen des schadhaften Codes direkt in `/dev/mem` basiert. Wegen der Schwierigkeiten

mit `/dev/mem` läuft Phalanx nicht auf allen Linux 2.6-Kernels, wurde aber auf zahlreichen Versionen von Fedora Core 4, Vanilla, Debian Gentoo und Ubuntu eingesetzt.

- **Disk Kernel-Rootkits:** Diese sind weniger elegant, aber nicht minder gefährlich. Sie werden durch Änderung der `/boot/vmlinuz`-Datei in einem Linux-System implementiert. Allerdings werden Disk Kernel-Rootkits erst nach einem Neustart des Systems aktiv, weshalb die Ursachen eines Systemneustarts immer zu hinterfragen sind. Ein Vertreter dieser Rootkit-Spezies ist kpatch.

Die ersten Kernel Mode-Rootkits entstanden 1997. Die seit damals häufigste Methode, ein Rootkit in den Kernel zu bekommen, war mittels ladbarem Kernel-Modul und einer Substitution der Systemaufrufe. Zum Vergleich: Das erste Kernel Mode-Rootkit unter Windows entstand 1999 für Windows NT.

Das Kernel Mode-Rootkit Adore-ng

Während die Rootkits Phalanx und eNYeLKM nicht ganz leicht zu installieren sind, lässt sich Adore-ng extrem einfach verwenden und funktioniert außerdem zuverlässig. Die Gefahr, diesem Rootkit auf einem System mit aktuellem Linux 2.6-Kernel zu begegnen, ist daher besonders hoch. Adore-ng funktioniert aber auch auf Systemen mit Linux 2.4-Kernel. Daher lohnt die Betrachtung dieses Rootkits besonders.

Hat der Eindringling einmal das Rootkit mittels `insmod` in den Kernel gebracht, kann es mittels des mitgebrachten Steuerprogramms `ava` (andere Namen können zur Täuschung durchaus vergeben werden) bedient werden. Der richtige Administrator wird das Rootkit jedenfalls durch ein `lsmod` nicht finden können. Das Rootkit verwendet einen so genannten *Adore-Key*, um das Kernel-Modul und das Steuerprogramm zu tarnen. Wenn nicht der Standardwert `fgjgggfd` verwendet wird, entzieht es sich einfachen Suchen. Weiters verwendet das Rootkit eine spezielle UID/GID-Kombination für Files, die zu verstecken sind. Im Jargon der Hacker sind das die ELITE_UID und ELITE_GID. Typischerweise werden hier große Zahlen genommen, was für das Auffinden von Adore-ng hilfreich sein kann.

Der Hacker kann über zwei Wege in das System kommen: Er installiert sich einen eigenen Netzwerkzugang, oder er kommt als lokaler unprivilegierter Nutzer. Als solcher wird er sehr rasch root. Das Steuerprogramm `ava` ermöglicht es, Befehle als root abzusetzen (welche alle versteckt laufen und sich einer Ansicht durch `ps` oder `top` entziehen). Mittels `ava r /bin/bash` bekommt der Angreifer eine Shell, von der er verdeckt operieren kann (siehe Abb. 1). Es ist einfach, Dateien oder Verzeichnisse vollständig vor dem Zugriff durch den Systemverantwortlichen zu entziehen. Mittels der Option `h` kann jede Datei, jedes Directory oder jeder symbolische Link im Filesystem verborgen werden. Mittels `cd` kann der Angreifer sich ein solchermaßen verstecktes Verzeichnis setzen – man muss nur wissen, wie es heißt. Dort ist alles wieder sichtbar, sofern es nicht noch einmal extra versteckt wird. In Abb. 2 ist dargestellt, wie ein Verzeichnis zum Verschwinden gebracht wird.

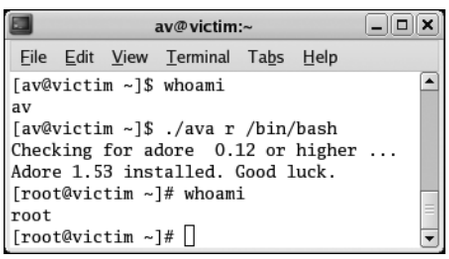


Abb. 1: Versteckte Root-Shell durch Adore-ng Rootkit

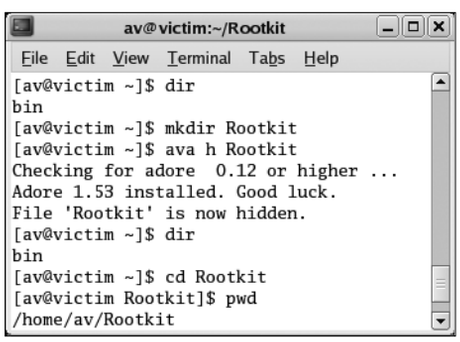


Abb. 2: Verstecken von Verzeichnissen durch Adore-ng Rootkit

Da das Kernel Mode-Rootkit (anders als ein User Mode-Rootkit) die Dateien vollständig verbirgt, ist ein verstecktes Verzeichnis oder eine versteckte Datei durch keines der regulären Systemwerkzeuge aufspürbar. Weder `echo` noch `find` können diese Dateien entdecken. In Abb. 3 ist zu sehen, wie Adore-ng durch das Verstecken eines Verzeichnisses auch die darin enthaltenen Dateien unsichtbar macht.

Die Versteckfunktionen von Adore-ng erstrecken sich nicht nur auf Dateien. Auch Prozesse können versteckt oder wieder sichtbar gemacht werden. Ein von einem versteckten Prozess erzeugter weiterer Prozess – zum Beispiel als Befehl auf einer Shell initiiert – ist ebenfalls unsichtbar. Das Verstecken ist sehr einfach, wenn das Rootkit bereits im Kernel ist. In Abb. 4 ist der Prozess (`bash`) zunächst sichtbar. Er wird anschließend unsichtbar gemacht. Jeder weitere Befehl, der über diesen Prozess abgegeben wird, ist in keinem Prozesslisting auffindbar. Schlimmer noch: Sogar jedes Logging eines versteckten Prozesses in die systemweiten Logfiles wird unterbunden. Damit ist für den Angreifer sichergestellt, dass er keine unbeabsichtigten Logfile-Einträge verursacht. Der ahnungslose Systemadmini-

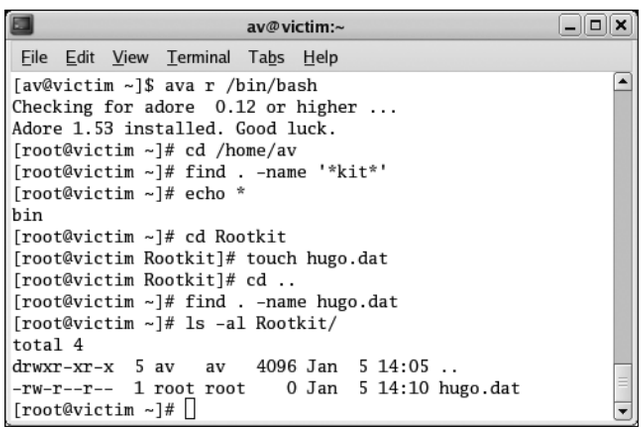


Abb. 3: Unauffindbare Dateien durch Adore-ng Rootkit

strator hat keine Chance, den Eindringling auf diesem Weg zu finden.

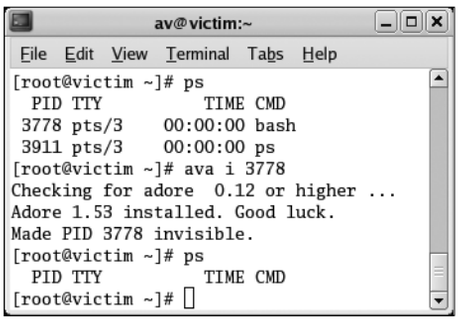


Abb. 4: Verstecken von Prozessen durch Adore-ng Rootkit

Fährtenlesen bei Rootkits

Da Rootkits die moderne Weiterentwicklung des Trojanischen Pferdes sind, ist es grundsätzlich schwierig, Spuren zu finden. Die zentrale Eigenschaft eines Kernel Mode-Rootkits, die Daten *vor* ihrer Verfügbarkeit zu fälschen, macht das erfolgreiche Aufspüren besonders schwer. Es gibt zwar einige Programme zum Suchen von Rootkits, jedoch hat sich im Test gezeigt, dass deren Auskünfte nicht unbedingt zutreffen müssen. Ein Klassiker unter den frei verfügbaren Rootkit-Suchwerkzeugen ist chkrootkit. Ein weiteres, sehr gutes freies Werkzeug zum Aufspüren von Rootkits ist auch der Rootkit Hunter oder kurz rkhunter. Beide finden eine überwiegende Vielzahl von User Mode- als auch Kernel Mode-Rootkits, versagen aber durchaus bei deren modernsten Vertretern. Für beide Suchwerkzeuge gilt: Im Falle von User Mode-Rootkits ist die Verfügbarkeit unverfälschter Originalprogramme für eine verlässliche Suche unumgänglich. Es ist daher sinnvoll, sich nach der Installation eines Linux-Systems die bereits genannten Systemprogramme auf eine CD zu brennen, sodass diese Werkzeuge darauf zugreifen können.

chkrootkit

chkrootkit läuft unter root auf dem System, welches zu untersuchen ist. Die Software ist rasch installiert. Unter `www.chkrootkit.org` kann die neueste Version heruntergeladen werden. Die beigefügte Prüfsumme ist mittels `md5sum` zu verifizieren. Der *Tarball* (die gezippte Tar-Datei) wird mittels `tar xvzf chkrootkit_`*`versionsnummer`*`.tar.gz` entpackt, wobei *`versionsnummer`* die aktuelle Softwareversion von chkrootkit ist. Anschließend wechselt man in das Verzeichnis der neu ausgepackten Software und tippt den Befehl `make sense` ein. Das Programm wird nun erstellt und kann mittels `./chkrootkit` aufgerufen werden. Wer einen sicheren Satz von Systemprogrammen auf CD verfügbar hat, verwendet den Befehl `./chkrootkit -p /mnt/cdrom`, wenn `/mnt/cdrom` der Mountpoint der CD ist. chkrootkit benötigt die Programme `awk`, `cut`, `echo`, `egrep`, `find`, `head`, `id`, `ls`, `netstat`, `ps`, `sed`, `strings` und `uname` und findet derzeit 60 verschiedene Rootkits.

rkhunter

rkhunter, verfügbar unter `www.rootkit.nl`, ist ein etwas komplexeres Werkzeug als chkrootkit. Die Software beschränkt sich nicht nur auf das direkte Suchen, sie ist auch in der Lage, einen Vergleich wichtiger Dateien zur letzten Suche zu ziehen, und prüft auch einige riskante Systemeinstellungen. Anders als chkrootkit wird rkhunter im System installiert. Nach dem Herunterladen des Tarballs wird rkhunter analog zu chkrootkit entpackt. Die Installation geschieht dann im Softwareverzeichnis des entpackten rkhunter mittels `./installer.sh`. Standardmäßig installiert sich die Software unter `/usr/local/rkhunter`. Leider ist dies für Hacker sofort ersichtlich, sodass diese, nachdem sie root-Rechte erhalten haben, die Konfigurationen des rkhunter so abändern können, dass die Ergebnisse eines Suchlaufes nicht mehr stimmen müssen. Deswegen – und weil Prüfsummen wichtiger Dateien dort abgelegt werden – ist zu empfehlen, die Installation von `installer.sh` so abzuändern, dass mittels des Parameters `-installdir` *`Installationsverzeichnis`* ein sicheres Verzeichnis angegeben wird, das nur zum Zeitpunkt des Tests auf dem System gemountet ist. Der USB-Stick des Systemadministrators wäre zum Beispiel ein geeigneter Ort. Nach dem Test sollte der Befehl `umount` und das Abziehen des Sticks nicht vergessen werden. Im Unterschied zu chkrootkit prüft rkhunter auch verdächtige Internetports (siehe Artikel *Firewalls: Schutz vor Gefahren aus dem Internet* in *Comment 02/2*, Seite 14 bzw. unter `www.univie.ac.at/comment/02-2/022_14.html`). So installieren manche Rootkits Netzwerk-Hintertüren, die permanent auf bestimmten Ports auf Verbindungsaufnahme des Hackers warten. Dies ist für die Rootkit-Suche an sich ein Vorteil, da manche Rootkits nur auf diesem Wege gefunden werden können. Allerdings kommt es auch vor, dass rkhunter zu viele Ports verdächtigt (mehr Informationen dazu sind unter dem URL `www.rootkit.nl` zu finden). Der Befehl `rkhunter` hat zahlreiche mögliche Parameter. Ein regulärer Test wäre mittels `rkhunter -c` durchzuführen.

Eigenes Fährtenlesen

Während beide Werkzeuge beim Suchen von User Mode-Rootkits recht erfolgreich sein können, ist ihre Effizienz bei den moderneren Kernel Mode-Rootkits nicht groß. Aktuelle Versionen von Adore-ng, Phalanx oder enYeLKM werden nicht gefunden. Welche Möglichkeiten hat nun der Administrator, um zu erkennen, dass sein System ein Problem hat? Bei nicht allzu großen Servern kann er sich den Netzwerkverkehr näher ansehen. Das ist beispielsweise mit einem Monitoring-Werkzeug wie ethereal möglich, welches in den meisten Linux-Distributionen enthalten ist (siehe z.B. auch `www.ethereal.com`).

Reisen die Benutzer des Systems viel (d.h. kommen sie von vielen unterschiedlichen, nicht vorhersagbaren IP-Adressen), dann kann das schwierig werden. Ein zweiter Nutzen des ethereal-Sniffers ist, dass er – unter root betrieben – die Netzwerkkarte in den so genannten „promiskuitiven Modus“ setzt. Sie empfängt dann auch Datenpakete, die nicht unbedingt an das System adressiert sind, und leitet sie an das Betriebssystem weiter. Damit kann auch der Sniffer Netzwerkverkehr aufnehmen, der nicht direkt für das System bestimmt ist. Ein Rootkit, das beim Befehl `ifconfig` die Anzeige des promiskuitiven Modus unterbindet, um dem Administrator die Tätigkeit des Sniffens nicht anzuzeigen, kann so gefunden werden (bei manchen Linux-

Versionen wird dies allerdings von Haus aus nicht angezeigt). Schaltet der Administrator ethereal ein und ist das Rootkit aktiv und verhindert die Anzeige des promiskuitiven Modus, so weiß der Verantwortliche, dass mit seinem System etwas nicht in Ordnung ist. Das Programm ifpromisc des Suchwerkzeugs chkrootkit bewerkstelligt die verlässliche Anzeige. Dazu ruft man entweder chkrootkit selbst auf und liest die Ergebnisse, oder man startet `ifpromisc` direkt. Wenn ein Programm wie ethereal die Netzwerkkarte in den promiskuitiven Modus versetzt hat, zeigt ifpromisc folgendes an: `eth0: PF_PACKET(/usr/sbin/ethereal)`, wobei der Pfad zu ethereal in der Anzeige wichtig ist: Er sollte mit der Originalplatzierung des Programms ethereal übereinstimmen. Das kann mit dem Befehl `which ethereal` überprüft werden.

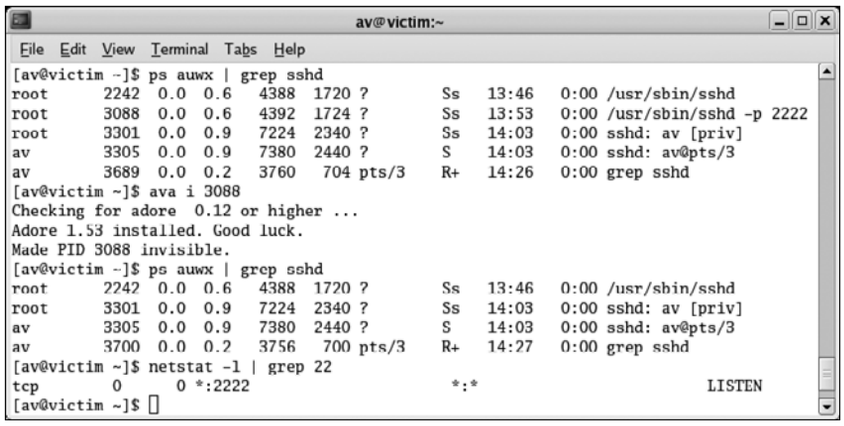


Abb. 5: Versteckt operierender SSH-Daemon auf Port 2222, welcher aufgrund ungeschickter Installation mittels `netstat -l` enttarnt werden kann

Da viele Kernel Mode-Rootkits ELITE_UID/GID-Kombinationen verwenden, die noch dazu aufgrund der Gefahr von Kollision und Entdeckung meist nicht innerhalb des regulären Nutzerbereiches (der Bereich, den der Administrator für die Nutzerkennungen angelegt hat) vergeben werden, lässt sich dies nutzen, um das Vorhandensein dieser Trojaner aufzuspüren. Dazu hilft uns Knoppix (eine komplett von CD oder DVD lauffähige Zusammenstellung von GNU/Linux-Software, zu finden unter `www.knoppix.de`). Durch die Vorgangsweise, Linux von CD zu booten und auf CD zu betreiben, ist Knoppix gegenüber Angriffen sehr sicher: Ein Neustart, und die Standardkonfiguration wird wiederhergestellt. Bootet man Knoppix von der CD, kann man mittels `ls -alR <Dateisystem>` das entsprechende Dateisystem rekursiv, also den gesamten Verzeichnisbaum, auflisten. Auch wenn das Listing sehr lang wird – große UID/GID-Nummern fallen mit Sicherheit auf. Sie sind unter den genannten Voraussetzungen ein nahezu untrügliches Zeichen für das Vorhandensein eines Rootkits. Knoppix kann zudem hilfreich sein, ein „sauberes“ chkrootkit laufen zu lassen.

Im Falle von Adore-ng gibt es für *ScriptKiddies* (Hacker, die vorgefertigte Schadprogramme verwenden, aber eigentlich keine Ahnung haben, was sie machen) allerdings auch Fallen. Installiert der Angreifer beispielsweise eine zweite Secure Shell zur Verbindungsaufnahme auf einem alternativen `ssh`-Netzwerkport (die Ports 2222 und 7350 werden von Adore-ng standardmäßig vorgeschlagen und sollten vor einem Listing durch `netstat -l` versteckt sein), so hat er glücklicherweise viele Möglichkeiten, etwas falsch zu machen, und der Zustand, dass ein Daemon hinter einem dieser Ports lauscht, wird nicht verborgen (siehe Abb. 5). Wie es allerdings richtig geht, wird hier nicht verraten.

Ausblick

Im Anschluss finden Sie den weiterführenden Artikel *Sonys digitaler Hausfriedensbruch*, der sich mit dem Einsatz von Hacker-Methoden durch Firmen beschäftigt. Ferner ist geplant, in einer der nächsten *Comment*-Ausgaben über Rootkits unter Windows zu berichten sowie über ausgefeilte Methoden, um diese Eindringlinge abzuwehren.

Aron Vrtala ■

SONYS DIGITALER HAUSFRIEDENSBRUCH

Wenn Firmen Hacker-Methoden anwenden

Der Musikverlag Sony BMG Music Entertainment hat eine ganze Reihe von CD-Titeln mit einem *Digital Rights Management* (DRM) namens XCP-Aurora versehen. Dieser Kopierschutz der britischen Software-Firma First4Internet (`www.first4internet.com`) greift jedoch unter Windows allzu tief ins System ein: Ende Oktober 2005 entdeckte der Sicherheitsexperte Mark Russinovich von Sysinternals (`www.sysinternals.com`), dass Sony auf den mittels DRM kopiergeschützten CDs ein Rootkit einsetzt, welches sich den Blicken der PC-EigentümerInnen entzieht. Der Kopierschutz installiert u.a. Filtertreiber für Festplatten und CD-ROM-Laufwerke. Damit kontrolliert die Trojaner-Software[1] die Zugriffe auf die Medien und speichert im Verborgenen Nutzungsinformationen. Selbstredend wird diese Software weder in der Software-Liste der Systemsteuerung angezeigt noch lässt sie sich mittels Uninstaller deinstallieren.

1) siehe Artikel *Ungebetene Gäste: Trojaner am Windows-PC* in *Comment 04/1*, Seite 10 bzw. unter `www.univie.ac.at/comment/04-1/041_10.html`

Das XCP-Rootkit (es dürfte in seiner Funktionsweise dem ersten Windows-Rootkit *NT Rootkit* angelehnt sein, das 1999 von Greg Hoglund entwickelt wurde) versteckt die ihm zugehörigen Dateien, Verzeichnisse, Prozesse und Registry-Einträge. Es versteckt alles, dessen Name mit `$sys$` beginnt. Daher kann sich mit Hilfe dieses Kopierschutzes unerwünschte, von HackerInnen stammende Software ebenfalls verbergen. Wenige Tage nach dem Bekanntwerden des Rootkits war bereits der Trojaner *Breplibot* im Umlauf, der `$sys$` als Tarnung verwendet. Ein weiterer solcher Trojaner ist z.B. *Backdoor.IRC.Snyd.A* – er installiert eine Hintertür zum Einstieg in das Windows-System. Wegen dieser Features von XCP wird in einschlägigen Internet-Foren bereits zum Kauf von Sony-CDs geraten.

Für die EntwicklerInnen von Trojanern und Viren ging ein Traum in Erfüllung: XCP-Aurora ist das erste käufliche Rootkit, ironischerweise mit LGPL-Lizenz (siehe `www.gnu.org/copyleft/lesser.html`). Für Sony BMG ging der Schuss nach hinten los: Auch Sonys eigene Spiele können mittels des Rootkits in geknackter Version verwendet werden (eine Musik-CD ist weit billiger als die Spiele).

Für die AnwenderInnen ist das XCP-Rootkit ein doppelter Schaden. Nicht nur, dass es Tür und Tor für Angriffe öffnet, es ist auch schlecht programmiert: Der installierte Treiber fragt jede zweite Sekunde alle laufenden Prozesse nach geöffneten Dateien ab, um sicherzustellen, dass kein Programm unbemerkt zu viele Kopien der geschützten Dateien produziert. Nachdem Programme unter Windows üblicherweise sehr viele Dateien geöffnet haben, bedeutet diese Vorgangsweise einen massiven Performanceverlust für den PC (geprüft wird auch dann, wenn sich gar keine CD im Laufwerk befindet). Darüber hinaus kann das Rootkit in bestimmten Situationen das System zum Absturz bringen und verursacht zusätzlichen Netzwerkverkehr: Es meldet die Verwendung der CDs an die Herstellerfirma und bietet Sony BMG damit die Möglichkeit, Nutzungsprofile zu erstellen.

Sony BMG demonstriert, wie Firmen um jeden Preis industrielle Interessen durchzusetzen versuchen. Mit dem Rootkit konfrontiert, reagierte Sony zunächst sehr unsensibel: *„Ich glaube, die meisten Menschen wissen gar nicht, was ein Rootkit ist, warum sollen sie sich also darum kümmern?“*, meinte Thomas Hesse, Vorsitzender der Abteilung *Global Digital Business* bei Sony BMG. Da die Kritik an Sony trotzdem nicht verstummen wollte, bietet die Firma seit einiger Zeit auf der Webseite `http://cp.sonybmg.com/xcp/` einen Deinstaller (der anfangs selbst eine große Sicherheitslücke in betroffene Windows-Systeme riss) sowie einen CD-Austausch an.

Das Vorgehen von Sony BMG löste eine Menge Fragen und Diskussionen über Gegenwart und Zukunft von Kopierschutzmechanismen aus. Es ist das erste Mal, dass eine Firma zum Schutz geistigen Eigentums und der damit verbundenen Rechte Hackerwerkzeuge einsetzt. Der weltweite Sturm der Entrüstung wird solche Methoden hoffentlich zukünftig verhindern.

Aron Vrtala ■

LAMPORTTAUEPSILONXI

Textverarbeitung und mehr

Hinter diesem kryptischen Namen steckt das in manchen Fachgebieten wohl weltweit am öftesten verwendete, aber auch in der breiten Öffentlichkeit am wenigsten bekannte computergestützte Textverarbeitungssystem: LaTeX. Nahezu jede naturwissenschaftliche Publikation, von Diplomarbeiten über Fachartikel bis hin zu Fachbüchern, wurde mit LaTeX geschrieben. LaTeX ist für alle Betriebssysteme kostenlos erhältlich, die zahlreichen BenutzerInnen und verfügbaren Module machen es zu einem leistungsstarken Werkzeug in der modernen Computerwelt. Der folgende Artikel soll einen ersten Einblick in LaTeX und einige Starthilfen für den Beginn bieten; es wird jedoch bewusst darauf verzichtet, die Leserschaft mit dem „eigentlichen LaTeX“, sprich der Befehlsstruktur, zu quälen. Eine auch nur halbwegs vollständige Anleitung zu LaTeX würde den Rahmen des *Comment* ohnehin bei weitem sprengen.

Die Geschichte von LaTeX reicht zurück bis ins Jahr 1977: Damals begann Donald E. Knuth an der Stanford University ein Textverarbeitungssystem zu entwickeln, das später als **T**au**e**psilon**X**i (TeX) einen weltweiten Siegeszug antrat. Die Zielsetzung war relativ klar umrissen: Die AutorInnen wissenschaftlicher Bücher sollten mit Hilfe eines universalen computergestützten Textverarbeitungssystems mathematische Formeln so editieren können, dass diese exakt so dargestellt wurden wie gewünscht. Die entsprechenden Schriftarten sollten mit METAFONT, einer eigens entwickelten Beschreibungssprache für Vektorschriften (siehe `http://de.wikipedia.org/wiki/Metafont`), definiert werden.

Auch wenn es im Jahr 2006 etwas befremdend klingt: Das Setzen von mathematischen Formeln für den Druck der Fachliteratur war bis vor 20 Jahren fast so langwierig wie das Errechnen der Formeln selbst und nur von Spezialisten des Buchdrucks zu bewerkstelligen, was sich natürlich auf den Preis auswirkte. Auch TeX war anfangs noch relativ unflexibel und gekennzeichnet von vielen Fehlern (die später sukzessive korrigiert wurden – heute ist TeX praktisch fehlerfrei). Deshalb entwickelte Leslie Lamport im Jahr 1982 das La(mport)TeX-System, das durch eine relativ einfache

LaTeX-Linksammlung

Archive & Linkseiten

`www.ctan.org`

Comprehensive TeX Archive Network, der Einstiegspunkt schlechthin – hier findet man so ziemlich alles, was es über LaTeX gibt

`www.dante.de`

Deutschsprachige Anwendervereinigung TeX e.V.

`www.esm.psu.edu/mac-tex/`

alles über LaTeX und Mac OS X

`http://staff.ttu.ee/~alahe/alatex.html`

umfangreichste Linksammlung zum Thema

Einführungen

`www.uni-giessen.de/hrz/tex/cookbook/cookbook.html`

ein Kochbuch für EinsteigerInnen

`www.infosun.fmi.uni-passau.de/infosun/software/latex/latex_tips.html`

jede Menge Tipps und Tricks

`www.weinelt.de/latex/`

alle Befehle, sehr übersichtlich dargestellt

`http://tex.loria.fr/graph-pack/grf/grf.htm`

alles über LaTeX und Grafiken

`http://sites.inka.de/picasso/latex.html`

alles über LaTeX und Grafiken, in Deutsch

`http://latex.tugraz.at/`

LaTeX@TUG, umfangreiches Projekt der Technischen Universität Graz

Editoren

`www.winedt.com`

WinEdt, ein ASCII-Editor für Windows (*„with a strong predisposition towards the creation of [La]TeX documents“*)

`www.lyx.org`

LyX, der erste WYSIWYM-Editor für LaTeX (*What You See Is What You Mean* – dabei wird zwar die logische Textauszeichnung, wie Überschriften oder Listen, am Bildschirm angezeigt, nicht aber die endgültige Formatierung des Textes)

`www.xm1math.net/texmaker/`

Texmaker, ein Editor für alle Betriebssysteme

`www.winshell.de`

WinShell für LaTeX mit vielen Zusatzprogrammen

`www.tex-tools.de/cms/`

WinTeX XP, vor allem für Windows-BenutzerInnen sehr zu empfehlen

Makros & Packages

`www.miktex.org`

MiKTeX, ein komplettes LaTeX-System für fast alle Betriebssysteme

`www.tug.org/texlive/`

TeX Live, eine TeX-Distribution für AIX, Mac OS, IRIX, Linux, Unix, Sun, Windows

`www.tug.org/mactex/`

MacTeX

`www.tug.org/teTeX/`

teTeX, eine vollständige TeX-Distribution für Unix-kompatible Systeme

Werkzeuge

`www.tug.org/yandy/`

Y&Y's Werkzeuge

`www.cs.wisc.edu/~ghost/`

Ghostscript, Ghostview und GSview, die Klassiker

`www.dessci.com/de/`

MathType, ein Formeleditor für MS-Word, der auch LaTeX-Formate ausgeben kann

`http://word2tex.com/`

Word2Tex, verwandelt MS-Word-Dokumente in LaTeX, ganz brauchbar

`http://latex.sehnot.de/`

beliebter LaTeX-Generator, nur für einfache Anwendungen geeignet

Zeichensätze

`http://texcatalogue.sarovar.org/bytopic.html#languages`

vollständige Liste aller unterstützten Zeichensätze

Befehlskette zur Kontrolle der Dokumentstruktur und des Layouts besticht. Mittlerweile ist LaTeX2e(psilon) der verwendete Standard; er basiert auf der aktuellen TeX-Version 3.141592.[1)]

Was ist LaTeX?

Die knochentrockene Definition lautet: TeX ist ein äußerst flexibles, rechnerunabhängiges Satzsystem zum Erstellen von Dokumenten in Buchdruckqualität; LaTeX ist ein integrierter Satz von Ergänzungen (*Makros*) zu TeX, die vorgefertigte Formatierungsanweisungen (*Styles*) enthalten, was das Arbeiten mit TeX wesentlich vereinfacht.

Mit Hilfe von Steuerbefehlen entscheidet LaTeX, wie etwas im Dokument angeordnet bzw. dargestellt wird, und ist daher in etwa mit HTML vergleichbar. Der wesentliche Unterschied: Bei einer HTML-Datei interpretiert der Browser die eingebundenen Steuerbefehle und zeigt die Inhalte des Dokuments dann entsprechend an; das Endergebnis kann jedoch – abhängig von Faktoren wie beispielsweise den am jeweiligen PC installierten Schriftarten oder den verwendeten Spracheinstellungen – recht unterschiedlich ausfallen. Ein LaTeX-Dokument hingegen muss zunächst mit einer speziellen Software übersetzt („kompiliert“) werden; erst die daraus resultierende DVI-Datei[2)] kann mit Hilfe eines weiteren Zusatzprogramms am Bildschirm grafisch angezeigt werden. Im Gegensatz zu HTML ist LaTeX darüber hinaus sehr kompromisslos, was Syntaxfehler anbelangt: Ist der Quellcode der Datei nicht korrekt, wird eine HTML-Datei in den meisten Fällen nur unansehnlich, ein LaTeX-Dokument aber gar nicht kompiliert.

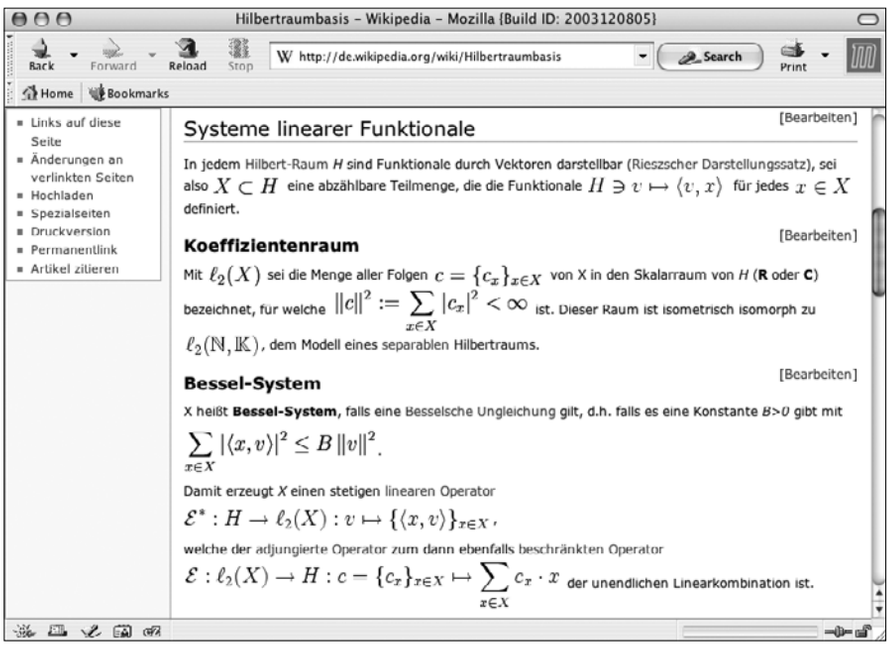
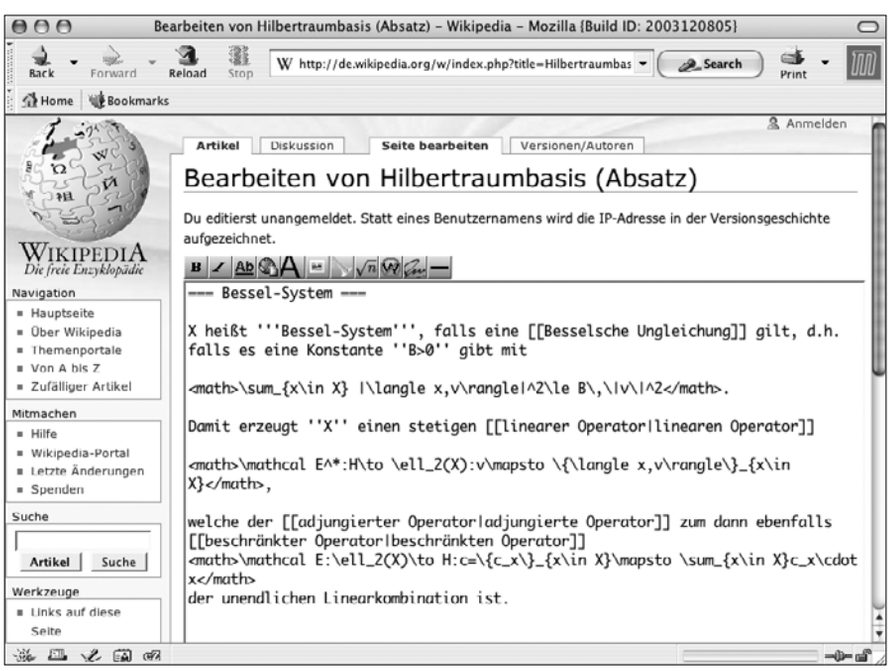


Abb. 1 (oben): Wikipedia-Seite mit LaTeX-Elementen
Abb. 2 (unten): Bearbeiten dieser Seite in Wikipedia



Der große Vorteil von LaTeX liegt darin, dass DVI-Dateien unabhängig von Betriebssystem, Schriftsätzen und Spracheinstellungen auf jedem Computer identisch aussehen – eine Eigenschaft, die der Output der meisten kommerziellen Desktop-Programme nicht besitzt. Im Vergleich mit Textverarbeitungsprogrammen wie MS-Word punktet LaTeX vor allem, wenn mathematische Zeichen gefragt sind: Das Setzen von Formeln aller Art gelingt mit LaTeX um Klassen besser. Abgesehen von der etwas gewöhnungsbedürftigen Bedienung ist LaTeX aber ohnehin mit professionellen Layout-Programmen (z.B. Adobe InDesign, Quark XPress) viel näher verwandt als mit Textverarbeitungen.

Wie bereits erwähnt, ist LaTeX für alle Betriebssysteme kostenlos erhältlich. Die Anzahl der BenutzerInnen weltweit und die Fülle der verfügbaren Werkzeuge sind beachtlich. Egal welche Frage ansteht – für jedes LaTeX-Problem gibt

1) Donald E. Knuth hatte 1977 vorgeschlagen, die transzendente Zahl π als endgültige Versionsnummer für TeX zu verwenden. Das ist die für seinen verschrobenen Humor typische Art anzudeuten, dass es nie eine endgültige Version geben wird: Eine transzendente Zahl hat unendlich viele Stellen nach dem Komma.

2) DVI steht für ***dev**ice **i**ndependent*, also geräteunabhängig.

es bereits eine Lösung, die in den unzähligen Foren und Beschreibungen einfach gefunden werden kann.

Anwendungsbereiche

Natürlich würde niemand seinen Einkaufszettel schnell einmal mit LaTeX schreiben; beim Setzen von mathematischen Formeln ist es aber immer noch der Standard schlechthin. Viele naturwissenschaftliche Fachpublikationen werden von den Verlagen nur als LaTeX-Dokumente akzeptiert, und auch an der Universität Wien sind nahezu alle Diplomarbeiten und Dissertationen in diesem Bereich mit Hilfe von LaTeX entstanden.

Darüber hinaus gibt es auch unzählige AnwenderInnen in anderen Fachgebieten, vor allem in der Linguistik. Dank METAFONT ist LaTeX in Bezug auf Zeichensätze sehr flexibel: Neben allen gängigen afrikanischen, asiatischen, europäischen und indianischen Sprachen sind beispielsweise auch Zeichensätze für die Hieroglyphen und das Olmekische verfügbar. Diese können ganz einfach in jedes Dokument eingebunden werden – ohne zusätzlichen Installationsaufwand und unabhängig von der Sprache des Betriebssystems. Eine vollständige Liste der unterstützten Zeichensätze ist unter `http://texcatalogue.sarovar.org/bytopic.html#languages` zu finden.

Eine sehr nette LaTeX-Anwendung aus der Musikwissenschaft sei hier ebenfalls erwähnt: Mit MusiXTeX (siehe `http://icking-music-archive.org/software/indexmt6.html`) ist es möglich, Noten in einem professionellen Layout darzustellen.

Auch Wikipedia setzt auf LaTeX: Am Beispiel des Wiki-Quellcodes der Webseite `http://de.wikipedia.org/wiki/Hilbertraumbasis` ist erkennbar, wie einfach die MediaWiki-Software das Einbinden von LaTeX in HTML macht (siehe Abb. 1 & 2 auf Seite 27; Näheres zu Wiki-Software finden Sie im Artikel *WIKI – Back to the Future* auf Seite 49).

Mit LaTeX arbeiten

Im *Comprehensive TeX Archive Network* (`www.ctan.org`) – jenem Ort im Internet, wo jede Suche nach „LaTeX-Allerlei" gestartet werden sollte – steht neben zahlreichen Zusatzprogrammen, Dokumentationen, Tutorials etc. auch das LaTeX-Basispaket zum Download bereit. Die Installation gestaltet sich in der Regel völlig problemlos.

Bevor man beginnt, mit LaTeX zu arbeiten, sollte man sich vor Augen führen, dass jedes LaTeX-Dokument grundsätzlich immer gleich und absolut logisch aufgebaut ist:

1. **Vorspann**: Hier wird die globale Struktur des Dokuments festgelegt. Das sind im Allgemeinen das Papierformat, die Textbreite und -höhe, Seitenränder, Seitenkopf und -fuss sowie die einzubindenden Module für Grafiken und Sprachen.

2. **Textteil**: Der Textteil umfasst den gesamten Inhalt des Dokuments – Titelseite, Text, Bilder und verschiedene Verzeichnisse (Inhalt, Tabellen, Bilder).

3. **Bibliografie**: Diese beinhaltet eine Auflistung der im Textteil verwendeten Literatur.

Ist man mit dem LaTeX-Befehlssatz – der, wie eingangs erwähnt, nicht Gegenstand dieses Artikels sein soll – vertraut, so kann man nun mit jedem beliebigen Text-Editor ein LaTeX-Dokument verfassen. In diesem Fall empfiehlt es sich, die Datei hin und wieder zu speichern, zu kompilieren und die DVI-Datei dahingehend zu überprüfen, ob sie dem gewünschten Ergebnis entspricht.

In der Regel wird man jedoch eine bequemere Variante der Bearbeitung wählen. Mittlerweile sind viele LaTeX-Editoren verfügbar (siehe *LaTeX-Linksammlung* auf Seite 26), die in punkto Funktionalität und Bedienung mit HTML-Editoren vergleichbar sind und in den meisten Fällen auch eine integrierte Vorschaufunktion enthalten. Zudem gibt es vorgefertigte „Grundgerüste" für verschiedene Dokumenttypen (z.B. für Fachartikel), sodass man auch ohne Kenntnisse des Befehlssatzes relativ einfach LaTeX-Dateien erstellen kann.

Wie bereits angesprochen, ist eine Datei, die in LaTeX geschrieben wurde, nicht unmittelbar verwendbar. Erst nachdem sie mit einem so genannten LaTeX-Compiler in eine DVI-Datei umgewandelt wurde, kann sie am Bildschirm grafisch dargestellt werden – dann allerdings *device independent*, also unter jedem Betriebssystem in absolut identischer Form. Die dafür benötigten Werkzeuge wie z.B. der Previewer *Yap* sind fester Bestandteil jedes LaTeX-Pakets.

DVI- und LaTeX-Dateien können praktischerweise in viele andere Dateiformate konvertiert werden. Die folgenden Makros sind ebenfalls in den meisten Paketen enthalten (wenn nicht, ist ein Link zur Software angegeben):

- **HTML:** *latex2html* und *dvi2html*
- **PDF:** *latex2pdf* und *dvi2pdf*
- **Postscript:** *dvi2ps*
- **RTF:** *latex2rtf* (`http://latex2rtf.sourceforge.net/`)
- **GIF und PNG:** *Textogif* (`www.fourmilab.ch/webtools/textogif/`)

Im Allgemeinen sollte man sowohl die LaTeX- (bevorzugt) als auch die DVI-Datei in das gewünschte Format umwandeln und die Resultate vergleichen: Vor allem wenn Grafiken eingebunden sind, kann es in seltenen Fällen zu einem unterschiedlichen Ergebnis kommen, weil die Papierformate (zum Beispiel A4 und/oder Letter) nicht immer einheitlich interpretiert werden. Die Umwandlung in eine MS-Word-Datei wird vom klassischen LaTeX-Paket nicht unterstützt, sondern ist nur mit Hilfe „externer" Programme möglich.

Ernst Paunzen ■

NEUE STANDARDSOFTWARE

Neue Produkte (Stand: 1. März 2006)

- Adobe InCopy CS2 1.0 für Win. und Mac
- Apple iLife 06 für Mac
- Apple iWork 06 für Mac
- Corel Draw X3 für Win.
- Corel PaintShop Pro X für Win.
- Endnote 9 für Win. und Mac
- ESRI ArcGIS 9.1 (siehe auch Seite 30)
- FileMaker Pro 8.0 für Win. und Mac
- InfoZoom Prof. 4.0 für Win.
- Macromedia Captivate (Robodemo) 1.01 für Win.
- Macromedia Dreamweaver 8 für Win. und Mac
- Macromedia Fireworks 8 für Win. und Mac
- Macromedia Flash 8 Prof. für Win. und Mac
- MS-AutoRoute Euro 2006 für Win.
- MS-Digital Image Suite 2006 für Win.
- MS-Encarta Premium 2006 für Win.
- MS-Money Deluxe 2006 für Win. (nur englisch)
- MS-Visual Studio Prof. 2005 für Win.
- Nero 7 für Win.
- Omnipage 15 für Win.
- Symantec Antivirus 10.0 für Mac
- Symantec Client Security 3.0 für Win.
- Symantec Norton Ghost 10.0 für Win.
- Symantec Norton Internet Security 2006 für Win.
- Symantec Norton SystemWorks 2006 für Win.
- Symantec PC Anywhere 11.5 für Win.

Updates (Stand: 1. März 2006)

- Exceed 11 2006 für Win. (bisher 10)
- LabVIEW 8.0 für Win., Mac, Linux (bisher – bzw. für Solaris unverändert – 7.1)
- Mathematica 5.2 für Win., Linux, Mac, Unix (bisher 5.1)
- MATLAB 7.0 R14 SP3 für Win. und Unix (bisher 6.5 R13)
- MS-Office 2003 für Win. inkl. ServicePack 2 (bisher ohne ServicePack)
- MS-Virtual Server 2005 R2 (bisher ohne R2)

Alle Informationen zur Standardsoftware finden Sie unter `www.univie.ac.at/ZID/standardsoftware/`

Peter Wienerroither ■

GEOINFORMATIK-SOFTWARE ARCGIS 9

Inklusive kostenloser Lizenzen für Studierende

Der Zentrale Informatikdienst stellt mit Beginn des Sommersemesters 2006 allen Universitäts-MitarbeiterInnen die Software ArcGIS 9 der Firma ESRI, einem der führenden Hersteller im Bereich der Geografischen Informationssysteme (GIS), als Campuslizenz zur Verfügung. Gegen eine geringe jährliche Gebühr (siehe weiter unten) kann jeder Interessent eine Lizenz erwerben, die zudem beliebig viele kostenlose Lizenzen für Studierende enthält. Derzeit wurden bereits Bestellungen von 11 Instituten für insgesamt 200 Lizenzen getätigt sowie 380 Lizenzen für Studierende angemeldet.

Die Software umfasst eine Reihe integrierter Anwendungen und Schnittstellen, mit denen beispielsweise das Erstellen von Karten, raumbezogenen Analysen, Datenbearbeitung und -umwandlung, Visualisierung sowie Geoverarbeitung möglich ist, wobei ArcView für eine umfassende Datennutzung, Kartenerstellung und Analyse geeignet ist. Daneben bietet ArcEditor zusätzliche Funktionen zur raumbezogenen Bearbeitung und Datenerstellung wie Datenmodellierung, Topologie oder Geodatenbanken und ist insbesondere für die Editierung komplexer Datenbestände in Mehrbenutzer-Umgebungen ausgelegt.

In der ArcGIS-Produktfamilie werden die beiden Produkte ArcInfo und ArcView auf eine gemeinsame technologische Basis zusammengeführt. War ArcInfo primär ein vollwertiges GIS auf Unix und ArcView eine reine Visualisierungssoftware auf PC, so hat die technische und anwendungsorientierte Weiterentwicklung zu einem intensiveren Zusammenspiel beider Produkte geführt. Die Software orientiert sich dabei am Bedarf der AnwenderInnen. Neben komplexen Programmen mit umfangreicher Funktionalität werden zunehmend einfach zu bedienende, funktionale Desktop-Programme gewünscht, die durch so genannte *Extensions* individuell erweiterbar sind.

ArcGIS trägt diesem Konzept Rechnung: Die Basisprodukte ArcView, ArcEditor und ArcInfo können sowohl auf einzelnen PCs (Einzelplatz-Lizenz) als auch in einem Netzwerk (Floater-Lizenz) installiert werden. Bei der Floater-Variante besteht zudem der Vorteil, dass die ArcGIS Extensions beliebig kombiniert verwendet werden können, während bei Einzelplatz-Lizenzen nur auf Programme auf diesem spezifizierten Rechner zugegriffen wird.

Der ZID bietet die gesamte ArcGIS-Software in Form von zwei Paketen an (der genaue Inhalt beider Pakete ist unter `www.univie.ac.at/ZID/software-news/` zu finden):

- **Paket 1:** ArcGIS 9 (inkl. ArcGIS Desktop, ArcInfo Workstation, ESRI Data & Maps u.a.)
  Gebühr: € 21,– je Lizenz für ein Jahr
- **Paket 2:** ArcGIS Server
  Gebühr: € 72,– je Lizenz für ein Jahr

Die Software läuft prinzipiell über Lizenzserver. Die für Standalone-Lizenzen – z.B. für Notebooks – benötigten Kopierschutzstecker (*Dongles*) sind in USB- oder Parallelport-Ausführung erhältlich und kosten € 48,– pro Stück.

Aufgrund des großen Umfanges von ArcGIS 9 erhalten LizenzbestellerInnen leihweise eine externe Festplatte mit der gesamten Software, wobei nur die bestellten Pakete freigeschaltet werden. Diese Festplatte darf nicht weitergegeben und die Software nur vom Lizenzinhaber installiert werden. Die Anzahl der Studierendenlizenzen muss dem Zentralen Informatikdienst jeweils zum 1. Februar und 1. September eines Jahres bekannt gegeben werden. Die Ausgabe dieser Lizenzen erfolgt derzeit noch auf einer kostenpflichtigen DVD.

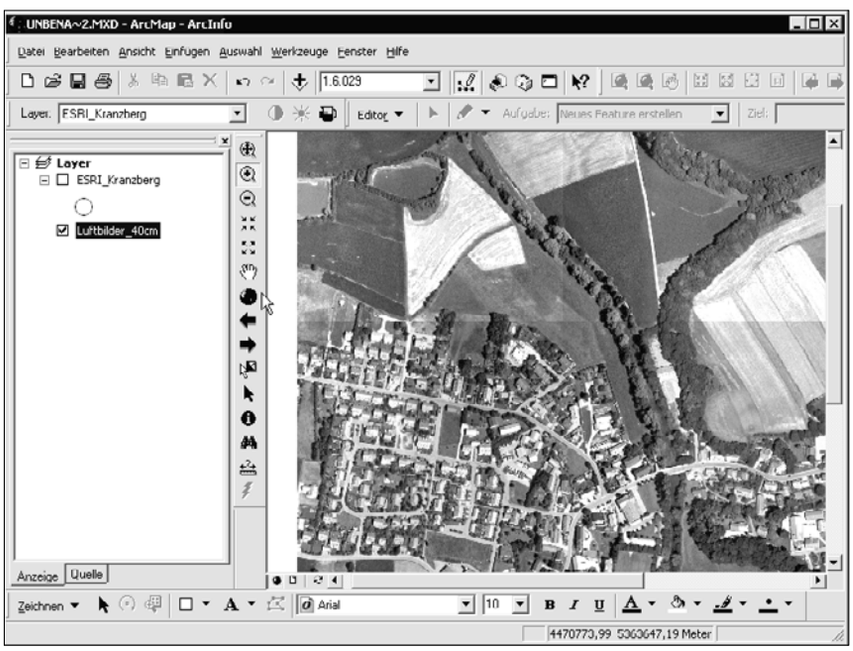


Abb. 1: ArcGIS 9 – Anwendungsbeispiel in ArcMap

Bei Interesse an der ArcGIS 9-Campuslizenz wenden Sie sich bitte per eMail an `software.zid@univie.ac.at`.

Umfassende Informationen zur Software ArcGIS, Demo-Filme zu vielen Features sowie Downloads von Software-Supplements und Dokumentationen finden Sie auf der Webseite der Firma ESRI (`www.esri-germany.de`).

Katharina Lüthke ■