

# Windows IT Pro

Das Magazin für den Windows-Administrator

## 64-Bit-Lösungen

Mehr Kerne, mehr Kraft  
64-Bit-Prozessoren  
Zusammenarbeit mit dem  
Betriebssystem

### TOOLKIT:

- Firewall mit Selbstschutz
- Hardware per Skript inventarisieren
- Anwendungen sicher machen

### WISSEN:

- Longhorn Server Beta 2
- Administratoren kontrollieren

### SPECIAL:

- Kontrollsummen helfen beim Kampf gegen Spam
- Analyse-Tool: Schutz für die E-Mail
- Gefährlicher Trend: Image-Spam



tausendblauwerk 07

**Sonderdruck für GeNUA**

# Wehrhafte Einrichtung

von Alexander von Gernerl

*Trotz vieler anderer Sicherheitseinrichtungen bleibt die Firewall die zentrale Instanz in einem Netzwerk. Sie muss deshalb in der Lage sein, auch direkte Angriffe auf die Firewall selbst erfolgreich zu bekämpfen. Unser Autor stellt die Selbstschutzeinrichtungen einer solchen Sicherheitslösung vor.*

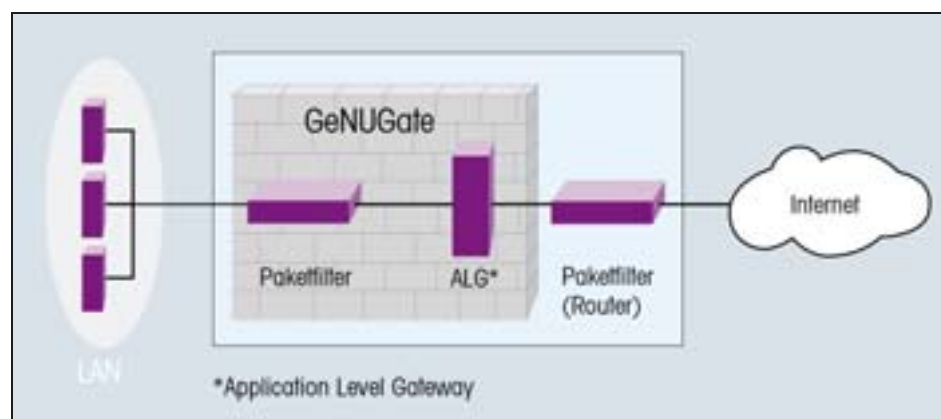
Eine Firewall stellt in der Regel die zentrale Sicherheitsinstanz eines Netzwerks dar. Aus diesem Grund ist es von entscheidender Bedeutung, dass diese Lösung allen direkten Angriffen standhalten kann. Die Entwickler dieser wichtigen Systeme besitzen zwei Möglichkeiten, dieser Art von Gefahren zu begegnen: Sie können beispielsweise ständig hinter den Angreifern und deren Attacken „herlaufen“ und die L cher stopfen, die diese in den Schutzwall gerissen haben. Der weitaus bessere Ansatz besteht aber darin, dass der Entwickler einfach den Spie umdreht: Er muss daf r Sorge tragen, dass sowohl das Sicherheitskonzept als auch die Selbstschutzeinrichtungen einer Firewall einer systematischen Analyse entspringen.

Alle Verbindungen, die von einem Netzwerk nach auen f hren, laufen auf der Firewall zusammen. Das bedeutet aber auch, dass jeder, der die Kontrolle  ber die Firewall besitzt, damit auch das gesamte Netz dahinter kontrolliert. Schlielich ist er damit in der Lage, alle ein- und ausgehenden Verbindungen zu belauschen oder zu manipulieren. Es ist also naheliegend, dass die Firewall nicht nur einen besonderen Schutz f r das interne Netz bieten sollte, sondern auch sich selbst angemessen gegen Angreifer abschirmen muss. Die Unversehrtheit der Firewall ist f r einen sicheren Netzbetrieb sehr wichtig.

Einfache Paketfilter sind hierbei relativ trivial abzuschirmen: Attacken auf Systeme k nnen nur  ber deren Schnittstellen zur Auenwelt erfolgen. Deshalb bietet hier ein reiner Filter relativ wenig Angriffsfl che, weil die Pakete dabei bereits in den Netzwerkroutrinen des Betriebssystemkerns bearbeitet werden. Zudem sind meist keine zus tzlichen Dienste gestartet, die noch an zus tzlichen Schnittstellen lauschen und deshalb extra abgesichert werden m ssten.

**Der Inhalt bewirkt den Unterschied: Application Level Gateway.** Ganz anders verh lt es sich hingegen bei den Application Level Gateways. Diese aufw ndigen Firewall-Systeme filtern die abgerufenen Inhalte wie etwa Mails, Webseiten oder Downloads auf Viren und aktive Inhalte (JavaScript, ActiveX und  hnliche M g-

Netz. Denkbare Aufgaben solcher Programme sind beispielsweise die Filterung der besuchten Webseiten, heruntergeladener Dateien und des E-Mail-Verkehrs. Jeder dieser Proxies ist f r sich wieder eine komplexe Software, die den normalen Gesetzen der Softwareentwicklung unterworfen und damit auch potenziell fehlerbehaftet ist.



*Sicherheit wie sie auch vom BSI empfohlen wird: P-A-P bedeutet Paketfilter, Application-Level-Gateway, Paketfilter sind drei Stufen, die eine erh hte Sicherheit bei einer Firewall erm glichen.*

*(Quelle: Genua)*

lichkeiten). Diese Filterung des Contents f r das innere Netz gewinnt heutzutage immer mehr an Bedeutung, weil die Client-Systeme hinter der Firewall oft Ziel von Trojanern oder Mail-W rmern sind. Durch die Interpretation des Inhalts  ffnet sich eine neue Schneise f r Angriffe gegen die Firewall selbst. Dieser Gefahr muss auf Systemebene wirksam begegnet werden.

So werden f r die  berpr fung der durchgeleiteten Protokolle meist Programme eingesetzt, die Netzwerkverkehr aufnehmen, seinen Inhalt filtern und wieder ins Netz zur ckschicken. Diese Programme werden „Proxies“ (Stellvertreter) genannt,  hneln einem Web-Proxy und behandeln die Anfragen f r die Client-Systeme im inneren

Gelingt es einem Angreifer nun, einen Fehler in einem Application Level Proxy auszunutzen, kann er die Kontrolle  ber die Firewall erlangen. Das ist der Super-GAU f r jeden Administrator. Mit fortschreitender Entwicklung kommt erschwerend hinzu, dass eine Firewall immer mehr Protokolle unterst tzen muss, so kam beispielsweise durch die Verbreitung von VoIP-Anwendungen unl ngst das SIP-Protokoll hinzu. Durch die steigende Komplexit t der Aufgaben einer Firewall, gelangt immer mehr Code auf die Firewall-Systeme, der potenziell auch angreifbar ist. Dadurch ist der Selbstschutz der Systeme und folglich die Sicherheit der dahinter liegenden Netze gef hrdet.

## Sicherheitsumgebung: Der Cage der GeNUGate

- Beim Cage handelte es sich um eine erweiterte „chroot()“-Umgebung, wie sie auf Unix/Linux-Systemen zum Einsatz kommt.
- Ein KILL-Kommando auf Prozesse außerhalb dieser Umgebung ist verboten.
- Ein nochmaliger Aufruf von „chroot()“ ist verboten, sodass ein Angreifer die Umgebung nicht einfach verlassen kann.
- Die Befehle „mount“ und „mknod“, mit deren Hilfe Manipulation und Zugriff auf Dateisysteme gelingen könnte, funktionieren in dieser Umgebung nicht mehr.
- Andere „Systemcalls“ unterliegen strengen Prüfungen.

### Ein holistischer Ansatz: Firewalls als Gesamtlösung betrachten.

Wie kann der Administratoren diesem Problem wirksam begegnen? Zunächst empfiehlt es sich, die Firewall als Gesamtprodukt aus Betriebssystem, Hardware und Firewall-Software zu betrachten, als so genannte Appliance. Das ist deshalb wichtig, weil Fehler in der Firewall-Software ausgenutzt werden können, um in die darunter liegende Betriebssystemstruktur einzudringen und umgekehrt. Auch die Hardware darf nicht vernachlässigt werden, denn manche Architekturen stellen wirksamere Mechanismen zur Absicherung des Betriebssystems bereit als andere. Die Sicherheit ist ein verketteter Prozess, bei dem das schwächste Glied und nicht die stärkste Komponente über das Schicksal des Gesamtsystems entscheidet. Im Folgenden wird der Selbstschutz einer Appliance anhand des Beispiels der Firewall GeNUGate betrachtet. Diese Firewall des Herstellers Genua verwendet als Betriebssystem OpenBSD auf der i386-Architektur. Die Entscheidung für das freie Unix-Derivat OpenBSD fiel deshalb, weil nur bei diesem System der Fokus der Entwicklung gänzlich auf Sicherheit liegt. Diese Konzentration auf den Sicherheitsaspekt zieht sich dabei durch alle Designentscheidungen. Mit den i386-kompatiblen PCs als Plattform steht darüber hinaus eine große Auswahl an Hardware zur Verfügung, auch wenn diese Architektur gerade in der Praxis häufig unter ihren über die Jahrzehnte gewachsenen Strukturen leidet. Mit einem sicheren Betriebssystem und einer Standardarchitektur als Grundlage kann nun versucht werden, die möglichen Sicherheitslücken der Application Level Proxies zu schließen.

### Der Selbstschutz der Firewall: Mehrstufigkeit ist wichtig.

Die Philosophie für den Selbstschutz der hier vorgestellten Firewall basiert auf Mehrstufigkeit: Selbst wenn ein Proxy bereits von einem Angreifer gekapert sein sollte und somit zum Ausgangspunkt für weitere Aktionen wird,

stemmen sich dem Angriff viele verschiedene Mechanismen entgegen. Da ist beispielsweise das Konzept der Relays zu nennen: Ein Proxy auf dem System basiert zum Großteil aus einem oft getesteten und bei praktisch allen Protokollen eingesetzten gemeinsamen Unterbau. Nur der protokollspezifische Teil wird pro unterstützte Anwendung extra hinzugefügt. Durch diese Wiederverwendung von bewährtem Code wird dem Prinzip der steigenden Anzahl von Fehlern mit steigender Anzahl von Lines of Code (LOC) Rechnung getragen und Fehlerquellen können gezielt ausgeschlossen werden.

Die Firewall besitzt aber noch eine Besonderheit: Die Relays wurden nicht in einer kompilierten Sprache wie etwa C realisiert, bei der zunächst Code in Maschinensprache übersetzt wird, um dann zu einem späteren Zeitpunkt ausgeführt zu werden. Der Großteil der Software auf dem System ist in einer

nächst im Interpreter ankommt. Jetzt muss der Angreifer also noch aus dem Interpreter ausbrechen, um auf die tiefere Ebene zu gelangen. Als nächste Sicherheitsmaßnahme arbeiten alle Prozesse, die sich auf einem solchen System direkt mit der Außenwelt unterhalten, in einer speziell eingeschränkten Ablaufumgebung, dem so genannten Cage. Hierbei handelt es sich um noch stärker eingeschränkte chroot()-Umgebungen, deren Möglichkeiten und Fähigkeiten im Kasten auf dieser Seite aufgeführt sind.

### Der Prozessmaster: Er hat alle wichtigen Prozesse im Blick.

Eine zusätzliche Schicht im Sicherheitsmodell namens Prozessmaster, stellt eine zentrale Instanz für die Integrität auf dem System dar. Hierbei handelt es sich um einen dem Kernel speziell bekannten Prozess, der bestimmte Voraussetzungen für die Funktionsfähigkeit und Sicherheit der Appliance permanent überprüft. So kümmert sich diese Komponente beispielsweise darum, dass das Logging stets funktioniert, der Cron-Prozess (der Hintergrundprozess für automatische Batch-Prozesse), der das regelmäßige Starten von Aktionen betreut kontinuierlich läuft und dass auch andere wichtige Systemfunktionen verfügbar sind. Der Prozessmaster wiederum wird vom Kernel selbst überwacht. Antwortet er eine bestimmte Zeit nicht mehr auf die Anfragen des Kernels, so geht das System von einer Manipulation oder einer Fehlfunktion aus – die Firewall wird neu gebootet.

## Offenes Betriebssystem als Firewall-Grundlage: OpenBSD

- Direkter „Nachkomme“ des Ur-Unix von 1969
- Freies, 4.4BSD-kompatibles Betriebssystem
- Aktive Entwicklung, halbjährlicher Release-Zyklus
- Absoluter Fokus auf Sicherheit
- Konsequenter Einsatz von Kryptographie
- Frei exportierbar, da kanadisches Projekt
- Projektleiter: Theo de Raadt
- Webadresse: <http://www.openbsd.org/>

interpretierten Sprache geschrieben, etwa die Skriptsprache Perl. Die Programme befinden sich deshalb als Skripts auf dem System und werden zur Laufzeit vom Perl-Interpreter ausgeführt. Was zunächst seltsam anmutet, entpuppt sich beim zweiten Blick als weitere Hürde: Wenn ein Hacker mit einem Angriff, beispielsweise einem Buffer Overflow oder Format String Exploit, erfolgreich ist, landet er in C bereits auf der System-Call-Ebene, während er bei Perl zu-

Dies führt aber im Gegensatz zu anderen Systemen nicht zu einem erneuten Hochfahren der Firewall: Hier wurde es vom Hersteller so geregelt, dass die Appliance bewusst im Boot-Prozess anhält und dann auf eine Bestätigung durch den Administrator wartet. Dieser Mechanismus schützt davor, dass Angreifer von entfernten Rechnern aus die Firewall manipulieren und anschließend gleich neu starten können, um ihre Änderungen wirksam werden zu lassen.

Bleibt die Frage wie die Entwickler einer solchen Lösung verhindern wollen, dass der sichere Boot-Vorgang selbst nicht ausgehebelt wird. Hier helfen die so genannten BSD-File-Flags des Unix-Betriebssystems. Durch ihren Einsatz können bestimmte Dateien trotz Schreibzugriff auf das Verzeichnis als unveränderbar markiert und somit versiegelt werden. Bei der hier vorgestellten Lösung wurde dies für praktisch alle ausführbaren Dateien, den Kernel und wichtigen Skripts gemacht. Durchgesetzt werden diese File-Flags mithilfe des so genannten Securelevels: Sobald die Firewall Datenpakete annimmt, also von außen erreichbar ist, hat der Securelevel mindestens den Wert 1 und die File-Flags sind aktiv. Das bedeutet aber auch, dass es für einen Angreifer von außen via Netzwerk unmöglich ist, Systemdateien der Firewall in irgendeiner Weise zu manipulieren. Nur wenn der Securelevel auf 0 steht, sind die File-Flags außer Kraft und lassen Veränderungen zu. Der Wert 0 wird aber nur dann erreicht, wenn die Firewall vom Administrator vor Ort gebootet wird. Während des Bootens nimmt das System aber grundsätzlich über das Netzwerk keine Datenpakete von außen an.

### Doppelter Einsatz: Zwei Firewall-Systeme in einer Reihe.

Noch mehr Sicherheit wird durch die Abtrennung des Paketfilters von der Funktion des Application-Proxys erreicht: Weil immer noch eine zweite Filterung auf einem physikalisch vollkommen getrennten Firewall-System erfolgt, ist ein Eindringling selbst nach vollständiger Übernahme der Application-Level-Maschine nicht in der Lage, das Netz nach außen total zu öffnen. Schließlich kann er auch zu diesem Zeitpunkt die Regeln des Paketfilters nicht verändern. Dies ist nämlich nur durch direkten Eingriff des Administrators möglich, der dazu den physikalischen Zugang zur Firewall-Appliance besitzen muss. Um diese Aufgabe durchzuführen, muss der Administrator ein Boot-Medium in Form eines USB-Sticks oder einer Floppy geschrieben haben und mit diesem den Paketfilter neu starten. Diese Aufgaben sind dabei sicher keine Aktion, die ein Systembetreuer im Vorbeigehen oder gar aus Versehen und ohne Nachdenken durchführen könnte.

Ein wichtiger Faktor bei der Entscheidung für eine Sicherheitslösung betrifft immer wieder die Glaubwürdigkeit des Herstellers: Auch wenn dieser die Mechanismen des eigenen Produkts in den schönsten Farben schildert – die Kunden können und wollen häufig auch die Qualität der Sicherheitsfunktionen bis hinunter zum Quellcode in der Regel nicht überprüfen. Abhilfe schafft

## Übersicht: Zertifizierung einer Firewall

- Internationaler Standard Common Criteria
- Die Stufen EAL 1 bis 7 beschreiben Intensität der Prüfung
- Ab EAL 4 muss Quellcode offengelegt werden
- Bis EAL 4 weltweit anerkannt
- In Deutschland ist dafür das Bundesamt für Sicherheit in der Informationstechnik (BSI) zuständig ([www.bsi.de](http://www.bsi.de))



hier eine sorgfältige Analyse durch eine vertrauenswürdige dritte Partei, die den behaupteten Selbstschutz der Firewall noch einmal einer kritischen Prüfung unterzieht. Noch besser ist es, wenn die unabhängige Überprüfung selbst wieder einem Standard unterworfen ist, um die Ergebnisse verschiedener Hersteller vergleichbar zu machen.

### Zweifel ausräumen: Zertifizierung belegt Selbstschutz.

Hierzu kann der Hersteller einer Appliance eine Zertifizierung nach Common Criteria (CC) durchführen. Dabei handelt es sich um ein weltweit anerkanntes Verfahren zur Qualitätsprüfung bei IT-Sicherheitssystemen, das sieben Stufen von EAL 1 bis 7 (EAL: Evaluation Assurance Level) zur Verfügung stellt. Diese Level geben dabei die Prüftiefe an, also wie genau das System untersucht wurde. Eine Firewall als zentrales Element der IT-Sicherheit sollte hier den Level EAL 4 erfüllen. Höhere Stufen sind auf komplexe Systeme wie Firewalls aufgrund des großen Aufwands nicht anwendbar. Je nach Land ist dafür eine bestimmte Stelle zuständig, in Deutschland ist es das Bundesamt für Sicherheit in der Informationstechnik (BSI). Die in diesem Artikel vorgestellte Firewall

GeNUGate ist vom BSI nach CC in der Stufe EAL 4+ zertifiziert. Das soll das Attribut „+“ anzeigen, dass bei bestimmten Kriterien über die Anforderungen des Levels EAL 4 hinausgegangen wurde: So entspricht die von der Firewall erfüllte Stufe der Schwachstellenanalyse im Kontext der CC-Version 2.3 dem Baustein „AVA\_VLA.4“, obwohl an dieser Stelle nur „AVA\_VLA.2“ verlangt gewesen wäre. Für eine Zertifizierung haben die Prüfer zunächst aufgrund der Entwurfsdokumente eine aktive Suche nach Schwachstellen durchgeführt, die durch das Design bedingt sein könnten. Hierzu gehörte auch eine nochmalige genaue Überprüfung der zuvor vom Hersteller festgelegten Security Targets: Dabei muss beispielsweise geklärt werden, ob die Firewall-Lösung wirklich die Sicherheitsfunktionen zur Verfügung stellt, die in den Targets angegeben werden. Durch die Tests versuchen die Experten zum Beispiel festzustellen, ob es möglich ist, dass einige dieser Funktionen eventuell doch umgangen werden können.

Danach gingen die Fachleute auf die Suche nach allen verfügbaren Sicherheitshinweisen für die eingesetzten Softwarebestandteile: Dabei wurde sowohl nach entsprechenden Informationen zum Betriebssystem OpenBSD als auch nach solchen für die zusätzlich installierten Softwarekomponenten wie etwa Sendmail, den Nameserver BIND oder die Verschlüsselungssoftware GnuPG gesucht. Hierbei handelte es sich um eine weitgehend mechanische, dafür aber wichtige Arbeit: Tatsächlich fanden die Experten dabei unter den hunderten eingesetzten Softwarekomponenten drei Pakete, die noch nicht auf dem allerneuesten Stand waren. Diese wurden vom Hersteller auf den aktuellen Stand gebracht. Schließlich musste sich die Firewall in der Praxis gegen Attacks wehren, die mit einem breiten Arsenal von Security-Scannern, Pen-Test-Tools und Hacker-Skripten ausgeführt wurden: Nessus, nmap, die BSI BOSS Security Suite und andere. Eine Firewall kann immer nur so gut ist wie das Angreiferszenario, das bei der Entwicklung zugrunde gelegt wird. Wer nur ein Sammelsurium von Sicherheitsfeatures aneinanderreihet, der wird auf lange Sicht ein vernünftiges Konzept vermissen lassen. (fms)

## Der Autor:

Alexander von Gernler ist unter der Mail-Adresse [mgrunk@openbsd.org](mailto:mgrunk@openbsd.org) erreichbar. Er ist OpenBSD-Committer und arbeitet als Softwareentwickler bei der GeNUA in Kirchheim.