Analysetools für Firewall-Logfiles

Nur fürs Protokoll

Netfilter-Firewalls erzeugen sehr detaillierte Protokolle, die kaum jemand manuell auswerten will. Logfile-Analysetools wie IPtables Log Analyzer, Wallfire WFlogs und FWlogwatch helfen dem Admin dabei, den Überblick zu behalten und die relevanten Meldungen auszufiltern. Ralf Spenneberg

Für den sicheren Betrieb einer Firewall muss der Administrator im Bilde sein: Er sollte darum möglichst viele Vorgänge protokollieren lassen. Das darf aber nicht dazu führen, dass er in der Datenflut versinkt und in den täglich mehrere MByte großen Logfiles die relevanten Einträge übersieht.

Protokollhelfer

Einen Ausweg bieten Protokoll-Analysetools. Aus den vielen Programmen für diese Aufgabe hat das Linux-Magazin drei herausgepickt: IPtables Log Analyzer [1], WFlogs aus dem Wallfire-Projekt [2] und FWlogwatch [3]. Alle werten Logfiles in zahlreichen Formaten aus und präsentieren die Ergebnisse übersichtlich auf HTML-Seiten, WFlogs und FWlogwatch bieten zusätzlich einen Echtzeitmodus an. IPtables Log Analyzer nutzt als einziges der vorgestellten Tools eine Datenbank zum Speichern der Meldungen.

Was dort ein spezieller Feeder erledigt, beherrscht Harald Weltes Ulogd [4] bereits von Haus aus. Er ersetzt das klassische Syslog-System. Doch leider existieren bisher kaum freie Analysewerkzeuge, die auf der Ulog-Datenbank aufsetzen. Die Anwendung Ulogd-php [5] macht hier den Anfang. Im Gegensatz zu allen anderen Protokollsystemen kann Ulogd auch die Pakete, die zur Firewallmeldung geführt haben, in der Datenbank mitprotokollieren.

IPtables Log Analyzer

Der IPtables Log Analyzer präsentiert ein IPtables-Protokoll für Linux 2.4 oder 2.6 in Form einer übersichtlichen HTML-Seite (siehe Abbildung 1). Das Tool besteht aus drei Komponenten. Ein Database Feeder schreibt die Protokolleinträge in eine MySQL-Datenbank. Über ein Webinterface greift der Admin anschließend auf den Inhalt der Datenbank zu. Der Database Feeder, die Datenbank und das Webinterface lassen sich entweder gemeinsam auf demselben Rechner oder auf unterschiedlichen Hosts installieren. Bei letzterer Variante sammelt die Datenbank auf Wunsch die Protokolle mehrere Firewalls.

Hat er sich für eine Architektur entschieden, erzeugt und initialisiert der Administrator in MySQL eine Datenbank namens »iptables« mit Zugriffsrechten für die Benutzer »iptables_admin« und »iptables_user« und legt darin Tabellen an (Listing 1). Damit Netfilter auch ein Protokoll erzeugt, sind passende IPtables-Regeln nötig. Sinnvoll sind zwei benutzerdefinierte Ketten (Listing 2).

Ketten anlegen

Anstelle von »-j ACCEPT« verwendet IPtables nun immer »-j LOG_ACCEPT«. Für Shorewall [6] oder die Suse Firewall on CD [7] sind diese Änderungen nicht erforderlich. Suse wird das eigene kommerzielle Firewall-Produkt allerdings nicht weiterentwickeln. Besonders dessen Anwender sind daher in Zukunft auf Werkzeuge und Updates aus der Open-Source-Welt angewiesen.

Dann installiert der Admin noch das Webinterface. Hierzu verschiebt er das »web«-Verzeichnis in das Document-Root des Webservers und passt die »config.php«-Datei an die Einstellungen der Datenbank und des Webservers (Benutzer, Kennwort, URL …) an. Als letzte Aktionen sind Installation und Aktivie-

rung des Database-Feeders erforderlich. Auch hier ist noch der Datenbankbenutzer anzupassen.

IPtables Log Analyzer kennt drei Feeder-Varianten: »feed_db.pl«, »feed_db-shorewall.pl« und »feed_db-suse.php«. Für den automatischen Start des Feeders verschiebt der Administrator noch das mitgelieferte Startskript »scripts/iptablelog« nach »/etc/init.d« und erzeugt die Verknüpfungen in den »rc«-Verzeichnissen.

WFlogs

WFlogs ist das Protokoll-Analysetool des Wallfire-Projekts [2], lässt sich jedoch auch unabhängig davon einsetzen. Das modular aufgebaute Programm liest und verarbeitet Netfilter-, IPchains-, IPfilter-, Cisco-PIX-, Cisco-IOS- sowie Snort-Protokolle. Seine Ausgabe schreibt das Tool in den Formaten Text, HTML, XML oder dem interaktiven Echtzeitmodus namens Human. Eine Datenbankanbindung fehlt jedoch. Zusätzlich kann es Netfilter-, IPchains- und IPfilter-Protokolle ineinander umwandeln.

Die Installation von WFlogs ist unter Debian denkbar simpel. In Debian Sid ist WFlogs enthalten, für Woody gibt es unter [8] Pakete. Auf anderen Distributionen ist WFlogs aus den Quellen zu installieren. Es benötigt die ebenfalls aus dem Wallfire-Projekt [2] stammende WFnetobjs-Bibliothek. Zusätzlich empfiehlt sich die alternative DNS-Bibliothek »adns« [9], die DNS-Namen asynchron auflöst. Übersetzt wird WFlogs mit dem klassischen »./configure; make; make install«, die WFnetojbs-Verzeichnisse muss der Admin möglicherweise beim Configure-Aufruf angeben.

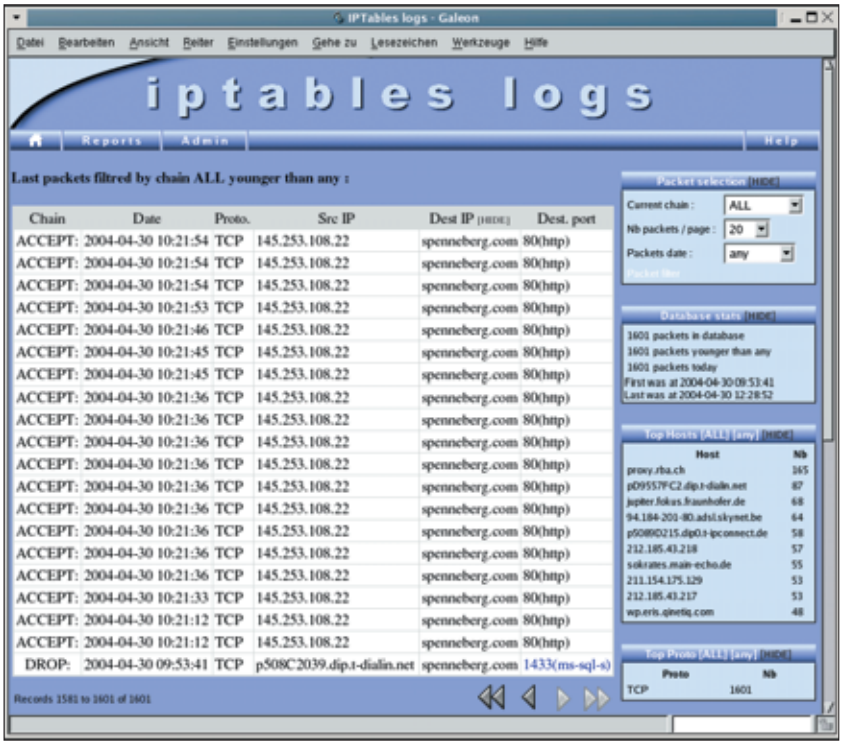

Abbildung 1: Der IPtables Log Analyzer bietet übersichtlichen Zugriff auf die Firewall-Protokolle.

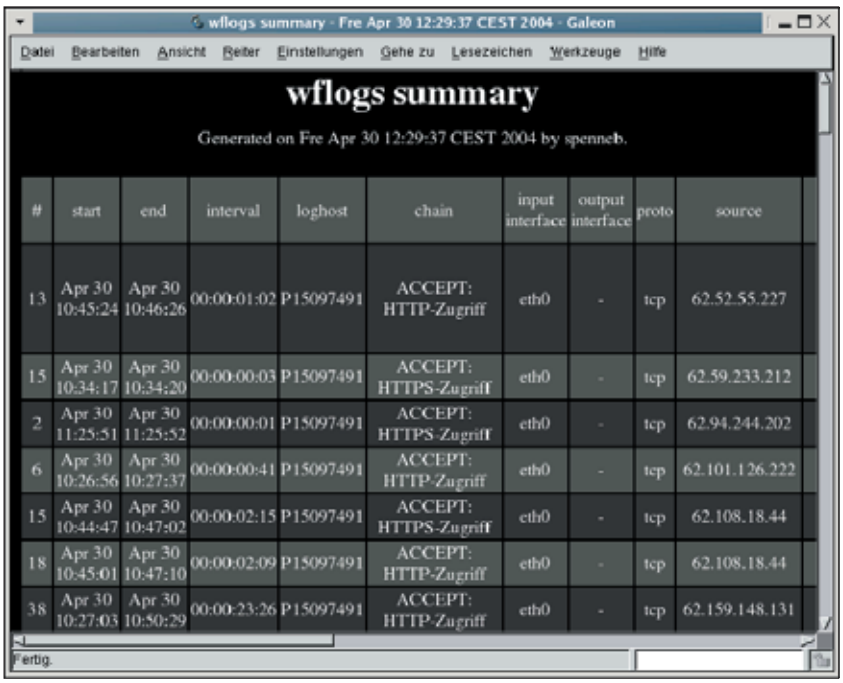

Abbildung 2: Die WFlogs-Summary-Seite zeigt, wie viele Pakete von welcher Quelle protokolliert wurden.

Netfilter zu HTML

WFlogs bearbeitet Firewall-Logs offline oder online. Der folgende Befehl generiert beispielsweise aus einem Netfilter-Logfile eine Übersicht im HTML-Format (Abbildung 2):

```
wflogs -i netfilter -o html ➥
netfilter.log > logs.html
```

Im Realtime-Modus analysiert WFlogs neue Einträge in der Protokolldatei und

Listing 1: MySQL-Datenbank

```
01 # mysql -u root -p
02 mysql> create database iptables;
03 mysql> grant create,select,insert on iptables.* to
   iptables_admin@localhost identified by 'g3h31m';
04 mysql> grant create,select on iptables.* to
   iptables_user@localhost identified by 'auchgeheim';
05 mysql> quit
06 # cat sql/db.sql | mysql -u iptables_admin -p iptables
```

Listing 2: IPtables Log Analyzer

```
01 iptables -N LOG_DROP
02 iptables -A LOG_DROP -j LOG --log-tcp-options
   --log-ip-options --log-prefix '[IPTABLES DROP] : '
03 iptables -A LOG_DROP -j DROP
04 iptables -N LOG_ACCEPT
05 iptables -A LOG_ACCEPT -j LOG --log-tcp-options
   --log-ip-options --log-prefix '[IPTABLES ACCEPT] : '
06 iptables -A LOG_ACCEPT -j ACCEPT
```

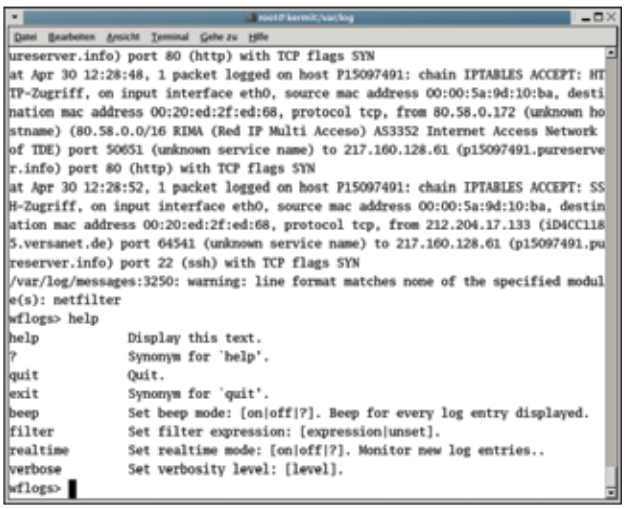

Abbildung 3: Per eingebauter Shell steuert der Benutzer WFlogs im interaktiven Modus. Das Tool zeigt die Meldungen in Echtzeit und filtert sie auf Wunsch.

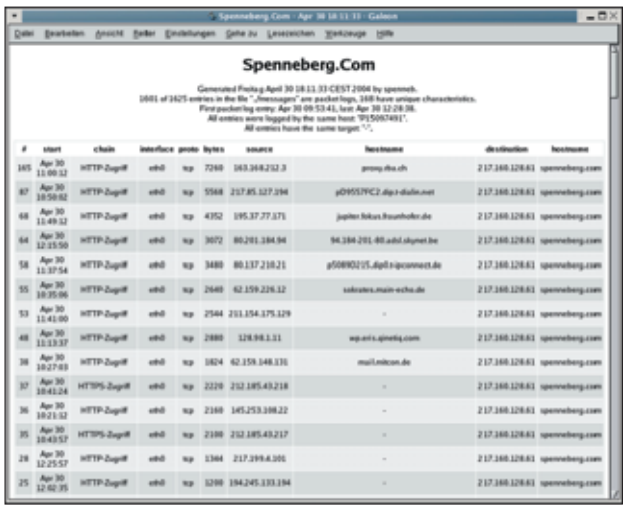

Abbildung 4: Der Zusammenfassungsmodus (Summary Mode) von FWlogwatch gibt einen Überblick über das Firewallprotokoll.

gibt diese auf dem Bildschirm aus. In einer kleinen Shell kann der Admin das Verhalten von WFlogs dabei auch interaktiv modifizieren. So wertet es im interaktiven Realtime-Modus die Datei »/var/log/warn« aus:

```
wflogs -RI -o human /var/log/warn
```

Mit der Option »-P« verarbeitet WFlogs auch alle früheren Meldungen in der Datei. Nicht-Firewall-Meldungen stören WFlogs dabei nicht (Abbildung 3).

Ausgesiebt

Mächtige Filter beschränken die Ausgabe auf bestimmte Meldungen. Der folgende Filter aus der WFlogs-Dokumentation zeigt alle abgelehnten Telnet- und SSH-Verbindungen, und zwar ausgehend vom Netzwerk 10.0.0.0/8 innerhalb der letzten drei Tage:

Listing 3: FWlogwatch-Echtzeitmodus mit Webserver

```
01 realtime_response = yes
02 parser = n
03 run_as = fwloguser
04 recent = 600
05 alert_threshold = 5
06 notify = yes
07 notification_script = /usr/sbin/fwlw_notify
08 server_status = yes
09 bind_to = 127.0.0.1
10 listen_port = 8888
11 status_user = ralf
12 status_password = i0QlAmOg4PrAA
13 refresh = 10
```

```
wflogs -f '$start_time >= [this 3 days ago] && $start_time < [this 2 days ago] && $chainlabel =~ /(DROP|REJECT)/ && $sipaddr == 10.0.0.0/8 && $protocol == tcp && ($dport == ssh || $dport == telnet) && ($tcpflags & SYN)' -i netfilter -o text --summary=no
```

FWlogwatch

FWlogwatch hat Boris Wesslowski für das RUS-CERT der Universität Stuttgart entwickelt. Mittlerweile gibt es das Analysetool in der Version 1.0 [3] unter der GPL-Lizenz.

FWlogwatch lässt sich in drei Modi betreiben: Zusammenfassungsmodus (Log Summary Mode), Berichtsmodus (Interactive Report Mode) oder Echtzeitmodus (Realtime Response Mode). Im ersten erzeugt es Text- oder HTML-Seiten mit einer Zusammenfassung der Firewall-Protokollmeldungen (Abbildung 4). Im Report Mode generiert FWlogwatch automatisch Incident-Berichte, die der Admin anschließend an die betroffenen Stellen senden kann.

Im dritten, dem Echtzeitmodus, reagiert FWlogwatch automatisch auf einen Angriff, zum Beispiel indem es Skripte ausführt, E-Mails verschickt oder die Firewallregeln selbstständig entsprechend abwandelt. Über einen eingebauten Webserver kann der Admin gleichzeitig den Status von FWlogwatch im Browser beobachten.

Das Tool unterstützt die Protokollformate von IPchains (Option »i«), Netfilter (»n«), IPfilter (»f«), IPFW (»b«), Cisco IOS (»c«), Cisco PIX (»p«), Netscreen (»e«), Windows XP (»w«), Elsa Lancom (»l«) und Snort (»s«). Zur Installation genügt ein simples »make && make install && make install-config«. Für Red Hat Linux und Debian stellt Boris Wesslowski

Abbildung 5: Der eingebaute Webserver von FWlogwatch informiert über den aktuellen Status der Firewall, zum Beispiel wie viele Meldungen FWlogwatch im angegebenen Zeitraum ausgewertet hat.

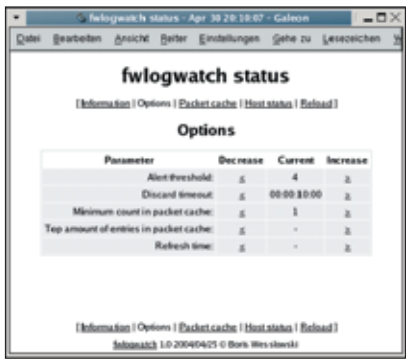

Abbildung 6: Im Browser kann der Anwender FWlogwatch konfigurieren. Der Alert Threshold bestimmt die Anzahl der Meldungen, ab der FWlogwatch seine Antwortskripte startet.

auf der FWlogwatch-Homepage auch fertige Pakete bereit.
Das Verhalten von FWlogwatch bestimmt der Admin entweder über die sehr gut kommentierte Konfigurationsdatei oder beim Aufruf über die Kommandozeile. Die mitgelieferte Manpage erklärt die Optionen. Folgender Befehl etwa startet FWlogwatch im Summary-Modus:

```
fwlogwatch -b -Pn -U 'Spenneberg.Com' ➥
-p -n -N -o output.html -t -w ➥
/var/log/messages
```

Die Option »-Pn« aktiviert den Netfilter-Parser. Mit »-U« gibt der Anwender eine Überschrift für die Zusammenfassung an. Die Option »-o« definiert die Ausgabedatei, »-w« fordert die Ausgabe in HTML an. »-n« und »-N« aktivieren die Namensauflösung für Rechner und Dienste. Das Ergebnis ist eine HTML-Zusammenfassung des Firewallprotokolls wie in Abbildung 4.

Reaktionsschnell

Am interessantesten ist der Einsatz von FWlogwatch im Echtzeitmodus. Hier bietet es die Möglichkeit, sowohl direkt auf Protokollmeldungen zu reagieren als auch den aktuellen Zustand über einen Webbrowser anzuzeigen. FWlogwatch arbeitet dazu als Daemon im Hintergrund und beobachtet die Protokolldatei. Sendet der Administrator »SIGHUP«, liest es seine Konfigurationsdatei neu ein. Bei einem »SIGUSR1« öffnet es seine Protokolldatei neu. Das ist besonders bei rotierten Logfiles sinnvoll.

Bei der Reaktion auf Protokollmeldungen legen Schwellenwerte fest, ab wann FWlogwatch Benachrichtigungs- oder Reaktionsskripte startet. Zwei Konfigurationsoptionen sind hier relevant: »recent« (»-l«) definiert den zu beobachtenden Zeitraum, »alert_threshold« (»-a«) die Anzahl der Ereignisse, die in diesem Zeitraum die Reaktion auslösen. Listing 3 zeigt eine Beispielkonfiguration. Hier ist FWlogwatch für den Echtzeitmodus mit dem Netfilter-Parser konfiguriert. Der Prozess läuft unter der Benutzerkennung »fwloguser«.
Wird der Schwellenwert von fünf Verbindungen in 600 Sekunden überschritten, ruft FWlogwatch »fwlw_notify« auf, das eine benutzerdefinierte Aktion ausführt. Zusätzlich bietet es einen Webserver auf 127.0.0.1:8888 an, bei dem sich der Anwender im Browser als Benutzer »ralf« mit dem Kennwort »kennwort« anmelden darf. FWlogwatch verwendet DES-verschlüsselte Passwörter, die zum Beispiel der Befehl »htpasswd -nb *User Password*« erzeugt. Sobald sich der Benutzer auf dieser Seite anmeldet, erscheint die in Abbildung 5 dargestellte Ansicht. Dort kann er viele weitere FWlogwatch-Einstellungen komfortabel im Browser ändern (Abbildung 6).

Auswahl

FWlogwatch glänzt mit seinem enormen Funktionsumfang, vom einfachen Summary bis zum Echtzeitmodus mit frei programmierbaren Reaktionen. Auch die anderen hier vorgestellten Werkzeuge lohnen einen Blick. Wer mächtige Filter für seine Abfragen braucht, ist mit WFlogs gut bedient. Der IPtables Log Analyzer ist vor allem wegen seiner Datenbankunterstützung interessant. So kann der Admin die Meldungen per SQL-Statement nach beliebigen Kriterien durchsuchen und ist nicht an das Webfrontend gebunden. *(eba)* ■

Infos

[1] IPtables Log Analyzer: [http://www.gege.org/iptables/]
[2] Wallfire-Projekt (WFlogs und WFnetobjs): [http://www.wallfire.org]
[3] FWlogwatch: [http://fwlogwatch.inside-security.de]
[4] Ulogd: [http://gnumonks.org/projects/ulogd]
[5] Ulogd-PHP: [http://www.inl.fr/download/ulog-php.html]
[6] Shorewall-Firewall: [http://shorewall.sourceforge.net]
[7] Suse-Firewall: [http://www.suse.de/en/business/products/suse_business/firewall/]
[8] WFlogs, Debian-Woody-Pakete: [http://people.debian.org/~kelbert/]
[9] GNU ADNS: [http://www.chiark.greenend.org.uk/~ian/adns/]

Der Autor

Ralf Spenneberg arbeitet als freier Unix/Linux-Trainer. Er veröffentlichte 2002 sein erstes Buch

„Intrusion Detection für Linux Server", gefolgt von „VPN mit Linux". Demnächst erscheint „Intrusion Detection und Prevention mit Snort und Co.".