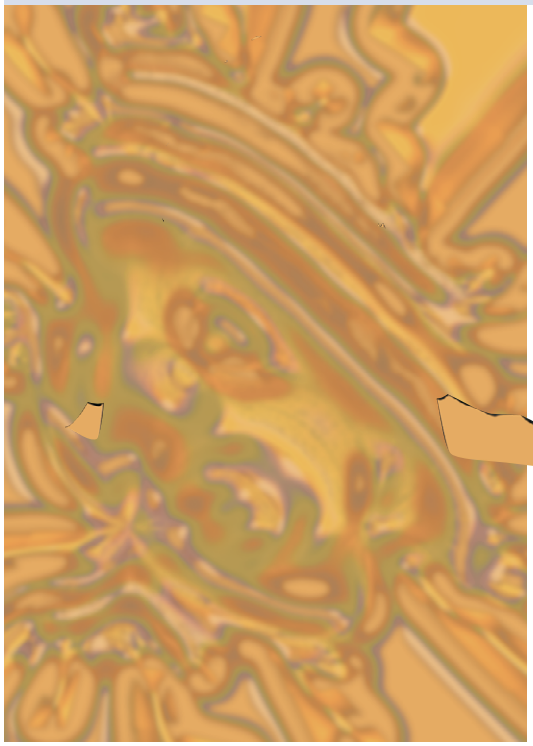
Hybrid-IDS überwacht sowohl Rechner als auch Netzwerk

Großer Auftakt

Gängige Intrusion-Detection-Systeme beobachten nur den einzelnen Rechner (Host-IDS) oder nur das Netzwerk (NIDS). Zusammengenommen versprechen die Sensordaten aber bessere Informationen über Eindringlinge. Das Open-Source-IDS Prelude ist HIDS und NIDS zugleich. Ralf Spenneberg



Das Intrusion-Detection-System Prelude [1] wurde 1998 von Yoann Vandoorselaere als GPL-Projekt gestartet. Ursprünglich wollte Yoann ein modulares NIDS entwickeln (Netzwerk-Intrusion-Detection-System). Während gleichzeitig Martin Roesch die Arbeit an Snort [2] begann, entwickelte sich Prelude von einem NIDS zu einem hybriden IDS. Yoann integrierte sowohl NIDS als auch Host-IDS-Funktionen in Prelude.

Schlägt Snort und Tripwire

Prelude ist damit ein wesentlich mächtigeres und umfassenderes Intrusion-Detection-System als zum Beispiel Snort oder Tripwire alleine: Während Snort zwar den Netzwerkverkehr untersucht, aber lokale Einbrüche nicht erkennt, analysiert Tripwire [3] nur das Dateisystem eines Rechners. Beide Systeme sind alleine nicht dazu in der Lage, verschiedene Ereignisse zu sammeln und zu korrelieren.

Für diese Aufgabe gibt es zwar im Fall von Snort einige zusätzliche Werkzeuge wie ACID [4] oder SGUIL [5], die die Daten mehrerer Sensoren zusammenfassen, doch muss der IDS-Betreiber diese Tools zusätzlich installieren sowie einzeln warten. Bei Prelude erhält er alle Werkzeuge, Komponenten und Funktionen aus einer Hand.

Komponenten

Das Prelude-System besteht aus vier Komponenten: Libprelude, Sensoren, Manager und Frontend. Die wichtigste Komponente ist die Libprelude-Bibliothek. Sie stellt die zentralen Kommunikationsdienste zwischen den Komponenten zur Verfügung. Hierfür setzt Libprelude das Intrusion Detection Message Exchange Format IDMEF [5] ein. Mit diesem standardisierten XML-Format können IDS-Anwendungen herstellerunabhängig Nachrichten austauschen. Um die Performance zu verbessern, nutzt Prelude eine eigene binäre Variante des IDMEF-Protokolls.

Die Prelude-Sensoren zeichnen Ereignisse auf und melden sie an die Manager. Libprelude ist verantwortlich für hochverfügbare Verbindungen der Sensoren zu den Managern. Erreicht ein Sensor seinen Manager nicht, wählt die Bibliothek selbstständig einen anderen Manager. Steht kein weiterer zur Verfügung, speichert sie die Informationen lokal in einer Datei (Caching), um sie später an einen Manager zu übertragen. So stellt Libprelude sicher, dass keine Daten verloren gehen.

Für die Übertragung wählt die Bibliothek automatisch das am besten geeignete Protokoll: Bei Verbindungen lokal auf einem Rechner überträgt Libprelude die Informationen als Klartext über Unix-Domain-Sockets. Muss sie die Daten übers Netzwerk übertragen, baut die Library eine SSL-Verbindung auf.

Prelude liefert zwei Sensoren mit: Prelude-NIDS (Netzwerk-Überwachung mit modifizierter Libpcap) und Prelude-LML (Log Monitoring Lackey, überwacht Hosts). Auch fremde Anwendungen können als Prelude-fähige Sensoren dienen, wenn sie die Libprelude-Bibliothek einbinden. Die Programme müssen dazu nur minimal angepasst werden. Zum Beispiel enthält Libsafe seit der Version 2.0-11 bereits die entsprechenden Änderungen, auch für Honeyd, Nessus, Samhain, Snort und Systrace gibt es passende Patches (siehe Kasten „Fremde Sensoren“).

Snort-Regeln in Prelude

Das Prelude-NIDS verwendet dieselbe Regelsyntax wie das weit verbreitete Snort. Damit können Administratoren die bereits vorhandenen, mächtigen Snort-Regelsätze mit Prelude weiter verwenden. Dieser Sensor unterstützt alle Snort-Features, etwa IP-Defragmentierung und TCP-Reassemblierung.

Eigene Plugins erweitern das Prelude-NIDS um einige Funktionen. Das HTTP-

Decoding-Plugin normalisiert HTTP-Anfragen und erkennt beispielsweise auch Angriffe, die sich mit Unicode-Zeichenkodierung tarnen. Weitere Decoder-Plugins gibt es für FTP, Telnet und das RPC-Protokoll. Außerdem unterstützt Prelude-NIDS ein Port-Scan-Detection-Plugin, ein Polymorphic-Shell-Code-Detection- sowie ein ARP-Spoof-Plugin. Letzteres erkennt typische Angriffe auf ARP-Protokollebene [7].

Prelude-LML ist der HIDS-Sensor von Prelude. Er analysiert die Protokollmeldungen des lokalen Rechners oder mehrerer entfernter Syslog-Clients. Dabei unterstützt der Prelude Log Monitoring Lackey (LML) unter anderem die Protokollformate von Netfilter, Checkpoint, NT-Syslog, Portsentry, Sshd und GR-Security. Die Funktionen lassen sich recht einfach anpassen und erweitern, da LML Perl-kompatible reguläre Ausdrücke (PCRE) beim Analysieren der Protokollmeldungen einsetzt.

Protokoll-Formate

Nachdem ein Sensor ein Ereignis erkannt hat, sendet er es mit Libprelude an einen Manager. Der Prelude-Manager nimmt die Meldung entgegen und protokolliert sie in einem geeigneten Format. Er unterstützt unter anderem Datenbanken (MySQL, PostgreSQL und Oracle) oder schreibt die Protokolle im XML-Format oder als Klartext.

Um die Prelude-Daten in einer Datenbank zu analysieren, stehen mehrere Frontends zur Verfügung: Das GTK-2-Frontend läuft direkt in der grafischen Oberfläche, während das Prelude-IDS-Webinterface Piwi über einen Webbrowser bedient wird. Zwei weitere Oberflächen befinden sich in der Entwicklung.

Prelude installieren und konfigurieren

Die Installation von Prelude [1] ist recht einfach, da die Entwickler sowohl RPM- als auch Debian-Pakete zur Verfügung stellen. Wer die neuesten Prelude-Funktionen (zum Beispiel das neue GTK-Interface) nutzen möchte, übersetzt und installiert Prelude aber besser aus den Quellen. Dabei ist die Reihenfolge wichtig: Erst Libprelude, dann den Rest, da die weiteren Programme schon beim Übersetzen die Bibliothek brauchen. Alle Programmpakete verwenden Autoconf, der übliche Aufruf »./configure && make && make install« funktioniert. Beim Configure-Skript kann man wie gewohnt Pfade angeben und einzelne Funktionen an- oder abschalten, zum Beispiel »--enable-mysql«.

Nach der Installation muss der Administrator die Datenbank für den Prelude-Manager erzeugen. Hierfür haben die Entwickler ein praktisches Skript zur Verfügung gestellt:

```
prelude-manager-db-create.sh
```

Es fragt interaktiv verschiedene Informationen ab, um die Datenbank zu erzeugen. Das Skript unterstützt sowohl MySQL als auch PostgreSQL-Installationen, es legt eine neue Datenbank für Prelude an und erzeugt einen Datenbankbenutzer. Hierfür muss die Datenbank bereits gestartet sein.

Logging in die Datenbank

Zum Abschluss gibt das Skript noch die passenden Konfigurationseinstellungen für den Prelude-Manager bekannt:

```
[MySQL]
dbhost = localhost;
dbport = 3306;
dbname = prelude;
dbuser = prelude;
dbpass = xxxxxx;
```

Diese Änderungen trägt der Admin in die Datei »*/Etc-Pfad*/prelude-manager/prelude-manager.conf« ein. Der Etc-Pfad lautet bei einer Quelltextinstallation per Default »/usr/local/etc«, die RPM- und Debian-Pakete verwenden »/etc«. Achtung: Bei den neuesten Betaversionen hat sich die Syntax bereits geändert. Künftig wird die Datenbank mit folgenden Parametern konfiguriert:

```
[db]
format = classic
type = mysql
host = localhost;
port = 3306;
name = prelude;
user = prelude;
pass = xxxxx;
```

Auch die Sensoren wollen konfiguriert sein. Listing 1 zeigt eine typische Konfigurationsdatei für Prelude-LML; mit ihr schickt der Sensor seine Nachrichten an

Fremde Sensoren

Prelude arbeitet auf Wunsch mit fremder Software zusammen und nutzt diese als zusätzliche Sensoren.

Libsafe: Diese Bibliothek vereitelt viele Buffer-Overflow-Angriffe. Der Admin kann jedem Programm die Libsafe unterschieben, wenn er die Library in »/etc/ld.so.preload« einträgt oder die »LD_PRELOAD«-Umgebungsvariable setzt. Der dynamische Linker lädt dann als Erstes Libsafe. Sie tauscht gefährliche und unsichere C-Funktionen durch sichere Varianten aus. Die neuen Funktionen erkennen den Buffer-Overflow und beenden das Programm, bevor Schaden entsteht. Ist Libsafe auf einem Prelude-System installiert, meldet es ab Version 2.11 automatisch Buffer-Overflows an den Prelude Manager.

Honeyd: Ist dieses virtuelle Honeynet mit einem Patch versehen, meldet es jede neue Verbindung an Prelude. So lassen sich auch diese Ereignisse über das Prelude-Frontend mit anderen Events korrelieren. [http://www.citi.umich.edu/u/provos/honeyd/]

Nessus: Der Vulnerability-Scanner sucht übers Netzwerk nach verwundbaren Diensten auf den erreichbaren Rechnern. Mit einem Patch versehen protokolliert Nessus seine Ergebnisse zu Prelude. [http://www.nessus.org]

Samhain: Der File-Integrity-Checker (ähnlich Tripwire) kann seit Version 1.8.2 auch Prelude als Logging-Zielsystem verwenden. Ein Patch ist nicht mehr nötig. [http://la-samhna.de/samhain/]

Snort: Prelude arbeitet von sich aus bereits mit Snort-Regelsätze. Wer lieber das Snort-Original statt des Prelude-NIDS einsetzt, muss nur ein Patch einspielen, dann funktioniert Snort auch als Prelude-Sensor. [http://www.snort.org]

Systrace: Dieses Sicherheitssystem beschränkt und überwacht die Privilegien eines Prozesses und kann ihn beim Überschreiten bestimmter Grenzen beenden [8]. Spielt der Administrator ein Patch ein, meldet Systrace diese Ereignisse an Prelude. [http://www.citi.umich.edu/u/provos/systrace/]

Listing 1: Prelude-LML

```
01 manager-addr = 127.0.0.1;
02 #[Udp-Srvr]
03 #port = 514
04 #addr = 0.0.0.0
05 file = /var/log/messages
06 rotation-interval = 3600
07 [SimpleMod]
08 ruleset=/usr/local/etc/prelude-lml/ruleset/simple.rules;
```

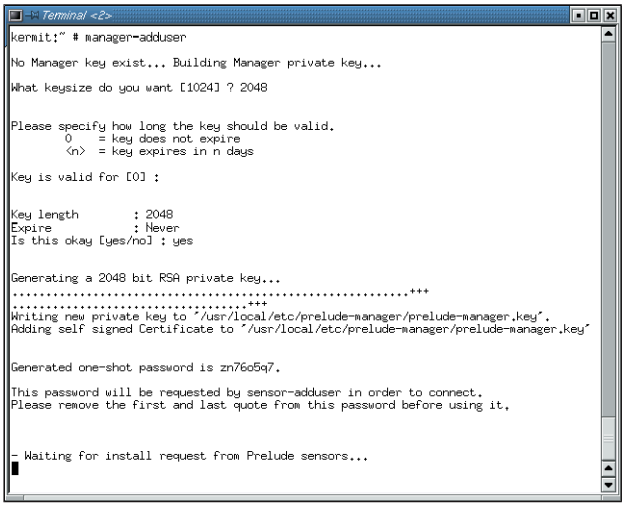


Abbildung 1: Der Admin bereitet den Prelude-Manager mit »manager-adduser« darauf vor, einen neuen Sensor zu akzeptieren. Der Manager erzeugt daraufhin ein Schlüsselpaar und teilt dem Administrator ein One-Shot-Passwort mit, mit dem sich später der Sensor ausweisen muss.

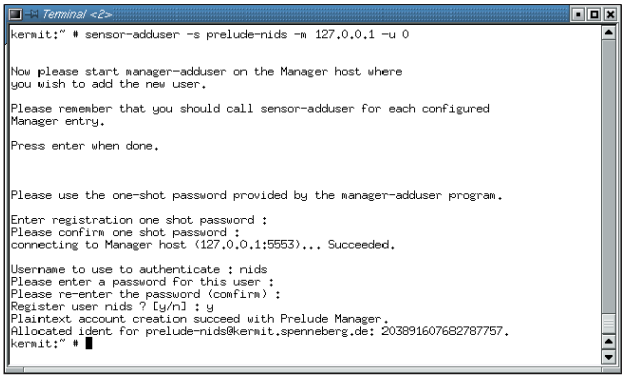


Abbildung 2: Prelude-Sensoren müssen sich bei ihrem Manager registrieren: »sensor-adduser« erledigt dies und authentifiziert den Sensor auch. Beim Erstkontakt tippt der Admin das One-Shot-Passwort ein, das er per »manager-adduser« erfahren hat. Danach läuft die Authentifizierung automatisch.

den lokalen Prelude-Manager. Der Prelude-LML ist in der Lage, entweder Protokolle über das Netzwerk entgegenzunehmen oder lokale Dateien zu analysieren. Gleichzeitig kann er beide Funktionen aber nicht wahrnehmen. Syslog-Meldungen würde er auf Port 514 empfangen, wenn die Zeilen 2 bis 4 nicht auskommentiert wären.

In Listing 2 ist die Konfigurationsdatei für das Prelude-NIDS zu sehen. Der Sensor liest Snort-Regeln, wie in Zeile 8 zu erkennen ist. Sensoren und Manager vertrauen sich nicht blind, die Sensoren müssen sich beim Manager erst registrieren. Die beiden Komponenten tauschen dabei Schlüssel aus, mit denen sie sich fortan authentifizieren.

Listing 2: Prelude-NIDS

```
01 [Prelude NIDS]
02 manager-addr = 127.0.0.1;
03 user = prelude;
04 [Tcp-Reasm]
05 statefull-only;
06 both;
07 [SnortRules]
08 ruleset=/usr/local/etc/prelude-nids/ruleset/
   prelude.rules;
09 [ScanDetect]
10 high-port-cnx-count = 50;
11 low-port-cnx-count = 5;
12 cnx-ttl = 60;
13 [Shellcode]
14 nops_raise_alert = 60;
15 [TelnetMod]
16 port-list = 23 21;
17 [HttpMod]
18 double-encode;
19 flip-backslash;
20 max-whitespace = 10;
21 codepage-file = /usr/local/etc/prelude-nids/
   unitable.txt;
22 codepage-number = 437;
23 port-list = 80 8080;
```

Gesprächsaufnahme

Der Prelude-Betreiber muss den Manager für neue Sensoren vorbereiten. Dazu ruft er auf dem Manager-Rechner den in Abbildung 1 gezeigten Befehl »manager-adduser« aus. Wurde Prelude mit OpenSSL-Unterstützung installiert, erzeugt dieser Befehl zunächst ein RSA-Schlüsselpaar (für die Authentifizierung und das Verschlüsseln der Kommunikation) und ein One-Shot-Kennwort.

Das eben generierte Passwort gibt der Admin beim ersten Verbindungsaufbau des Sensors an: »sensor-adduser -s *Sensorname* -u *UID* -m *Manager*«. Als Sensorname sind »prelude-nids« oder »prelude-lml« möglich, die Option »-u« legt den Benutzer fest, in dessen Kontext anschließend der Sensor arbeiten soll. Das Programm fragt das vom Manager ausgegebene One-Shot-Kennwort interaktiv ab (siehe Abbildung 2). Die Anmeldung muss für jeden Sensor einzeln erfolgen. Sind auf einem Rechner sowohl der NIDS- als auch der LML-Sensor installiert, dann ist die Anmeldung zweimal erforderlich.

Nun können die Prelude-Komponenten starten, beginnend mit dem Manager (Abbildung 3). Der lädt seine Plugins und baut die Verbindung zur Datenbank auf. Dann startet der Admin die Sensoren. Abbildung 4 zeigt den Prelude-LML-Sensor, er beobachtet die »/var/log/messages«-Datei. Um zu bemerken, wann jemand das File verändert hat, nutzt Prelude nach Möglichkeit den File Alteration Monitor (FAM) [12].

Um das IDS zu überwachen und die Ereignisse in der Datenbank zu analysieren, stehen vier Oberflächen zur Wahl. So hat Pablo Belin mit Gprelude eine grafische GTK-Anwendung geschrieben, die die Ereignisse anzeigt (Abbildung 5). Der Anwender kann mit diesem Tool auch in den Ereignissen suchen oder Filter anlegen, um nur Untergruppen der Meldungen zu sehen.

Mit Pylude steht ein weiteres grafisches Interface zur Wahl (Abbildung 6). Wie der Name bereits andeutet, ist es in Python geschrieben, es nutzt die QT-Bindings für sein GUI. Die Oberfläche erinnert etwas an Windows XP. Pylude bietet ähnliche Möglichkeiten wie Gprelude. Es benötigt aber die Libpreludedb, die derzeit nur in den Prelude-Betaversionen (Trunk) enthalten ist.

GUI und Webinterface

Neben den GUIs gibt es noch zwei Webschnittstellen für den Zugriff auf die Prelude-Datenbank: Piwi und Prewikka. Während sich Piwi bereits in einem recht fortgeschrittenen und ausgereiften Zustand befindet, ist Prewikka noch in der Entwicklung (und nur im Prelude-CVS verfügbar). Piwi lässt sich ziemlich

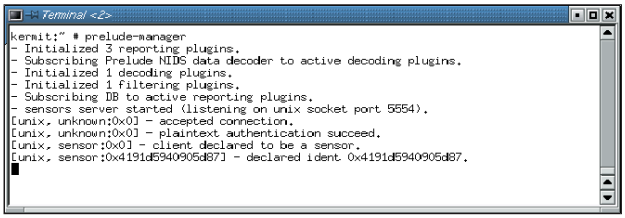


Abbildung 3: Beim Starten lädt der Prelude-Manager seine Plugins, verbindet sich mit der Datenbank (alle Zeilen, die mit »-« beginnen) und nimmt dann Verbindungen seiner Sensoren entgegen (diese Zeilen beginnen mit »[«).

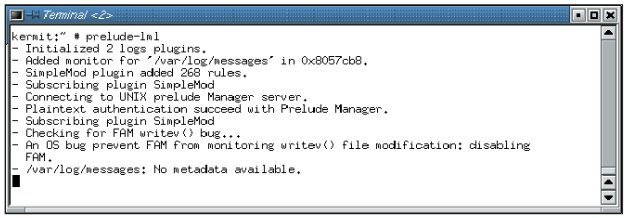


Abbildung 4: Der Prelude-LML-Sensor beobachtet die Logdatei »/var/log/messages«. Hierfür versucht er den File Alteration Monitor (FAM) einzusetzen – jedoch verhindert hier ein Bug, dass Prelude den FAM verwenden kann.

einfach installieren. Für seinen vollen Funktionsumfang benötigt es noch Ettercap [9] – Piwi kann die Ettercap-Datenbanken nutzen.

Das Piwi-Verzeichnis muss in einem Bereich liegen, auf den der Apache-Webserver zugreifen darf. In der Piwi-Datei »Functions/config.pl« sind die Parameter für den Datenbankzugriff anzupassen. Auch in der Apache-Konfiguration sind Änderungen nötig, damit der Webserver die Piwi-Perl-Skripte ausführt. Dabei kann Piwi die Mod_perl-Erweiterung des Apache und Apache::DBI nutzen:

```
<Files *.pl
  SetHandler perl-script
  PerlHandler Apache::PerlRun
  PerlSendHeader On
</Files>
```

Ohne Mod_perl muss Apache die Piwi-Skripte über das klassische CGI-Verfahren aufrufen:

```
<Directory "/var/www/html/piwi/">
  Options ExecCGI
  AddHandler cgi-script .pl
</Directory>
```

Alle Piwi-Verzeichnisse und ebenfalls alle Perl-Skripte benötigen die Rechte »rx« für den Webserver, in »generated/« muss der Webserver zusätzlich auch schreiben dürfen.

Voraussetzungen für Piwi

Der erste Piwi-Aufruf prüft die Installationsvoraussetzungen. Falls einzelne Rechte nicht stimmen oder Perl-Module fehlen, zeigt die Weboberfläche eine Seite mit Fehlermeldungen. Piwi benötigt die Perl-Module Socket, CGI und Date::Calc. Zusätzlich zu empfehlen sind Geo::IP, PDF::API2 und Ghostscript. Das Perl-Modul Geo::IP verwendet die Geo-IP-Datenbank [10], um IP-Adressen der Angreifer nach Ländern auflösen. Ist die Installation erfolgreich, wird der Administrator mit einem mächtigen Webinterface belohnt (Abbildung 7).

Neue Weboberfläche

Die Piwi-Alternative namens Prewikka befindet noch in der Entwicklung, soll nach Auskunft der Programmierer Miika Keskinen und Markus Alkio aber Piwi in naher Zukunft überholen und ablösen. Für die Installation der Python-basierten Oberfläche muss der Administrator lediglich das Prewikka-Verzeichnis in den Bereich des Webservers kopieren, die Rechte anpassen und die beigelegte »apache.conf«-Konfiguration für den Webserver übernehmen.

Zusätzlich sind das Python-MySQL-Modul [11] und eine Erweiterung der Daten-

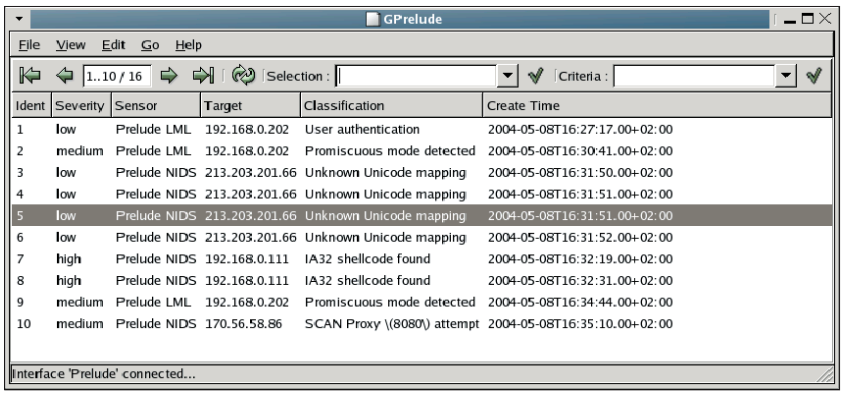


Abbildung 5: Die grafische Prelude-Oberfläche Gprelude verbindet sich mit der Prelude-Log-Datenbank und zeigt die protokollierten Ereignisse übersichtlich an. Der Benutzer kann die Einträge durchsuchen und sich mit Filtern auf einzelne Gruppen konzentrieren.

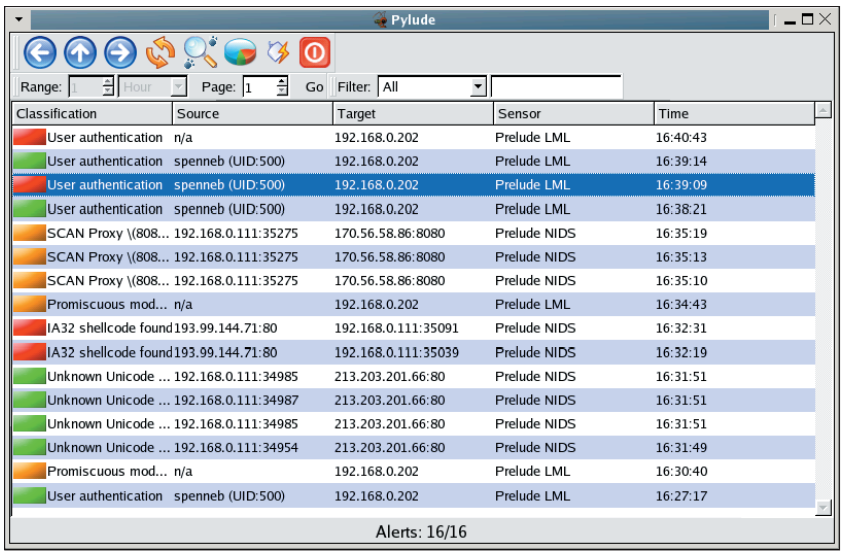


Abbildung 6: Das Pyprelude-GUI hat einen ähnlichen Funktionsumfang wie Gprelude (Abbildung 5), ist aber in Python entwickelt und verwendet die brandneue Libpreludedb.

bank erforderlich: »mysql -u prelude -p prelude < frontend.sql«. Anschließend steht der Zugriff auf das Prewikka-Webinterface offen (Abbildung 8). Der Administrator muss sich dazu authentifizieren, voreingestellt sind als User und Kennwort jeweils »admin«. Das Passwort sollte er als Erstes ändern, dann kann der Admin einen der in der Datenbank verfügbaren Sensoren in Prewikka registrieren (Abbildung 9). Achtung: Alle Felder bei der Registratur sind auszufüllen, andernfalls können Fehler auftreten.

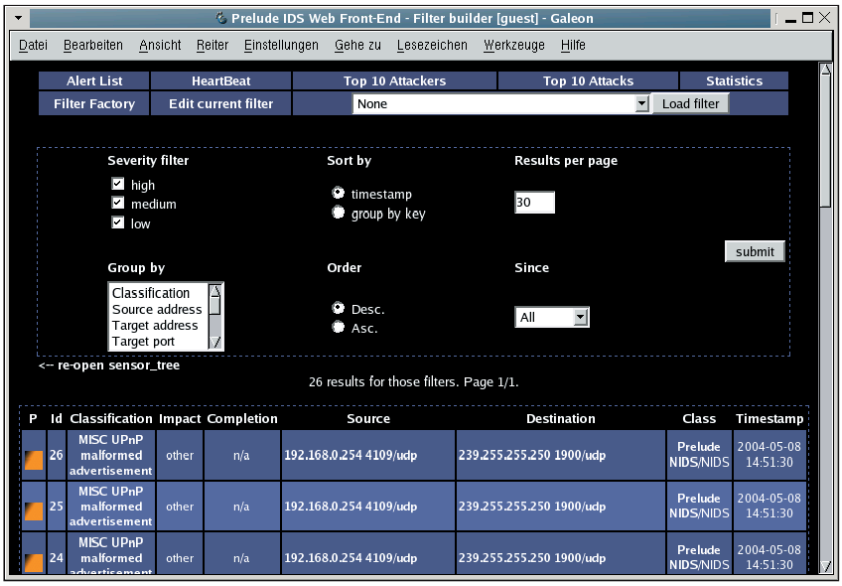


Abbildung 7: Das Piwi-Webinterface bietet von allen Management-Oberflächen derzeit den mächtigsten Zugang zu den Prelude-Protokolldaten.

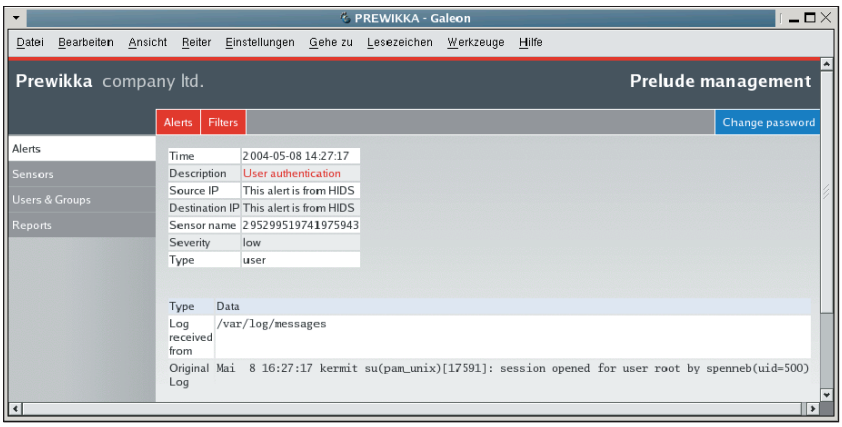


Abbildung 8: Nach der Authentifizierung gibt das Prewikka-Webinterface den Zugriff frei. Hier ist eine Warnung in der Detail-Ansicht zu sehen.

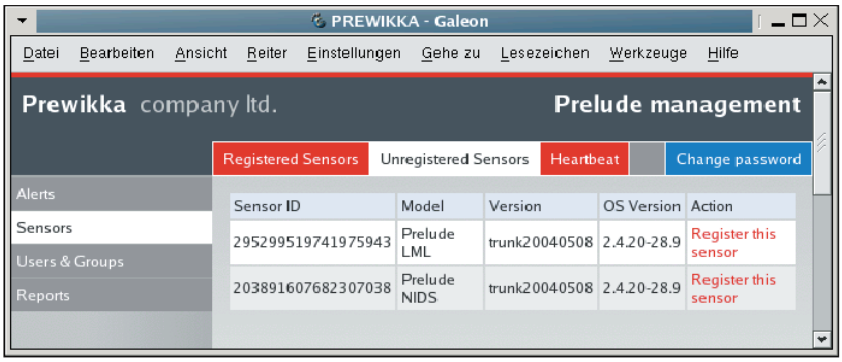


Abbildung 9: Bevor Prewikka einen Sensor unterstützt, muss ihn der Administrator registrieren. In der »Unregistered Sensors«-Ansicht zeigt Prewikka, welche Sensoren laut Prelude-Datenbank noch verfügbar sind.

Fazit

Prelude ist als Hybrid-IDS eine sehr interessante Entwicklung. Es verbindet erstmals unter Linux NIDS- und HIDS-Funktion: Prelude speichert die Ereignisse zentral und korreliert sie. Für Prelude steht eine große Anzahl zusätzlicher Sensoren zur Verfügung, beispielsweise Libsafe oder Honeyd und seit kurzem auch Samhain. Prelude bringt damit alle Voraussetzungen mit, um sich zu einem der mächtigsten Intrusion-Detection-Systeme im Open-Source-Bereich zu entwickeln. *(fjl)*

Infos

[1] Prelude-IDS: [http://www.prelude-ids.org]
[2] Snort: [http://www.snort.org]
[3] Tripwire: [http://www.tripwire.org]
[4] ACID, Analysis Console for Intrusion Databases: [http://acidlab.sourceforge.net]
[5] SGUIL, a Tcl/Tk Interface for Network Security Monitoring: [http://sguil.sf.net]
[6] IDMEF: [http://www.ietf.org/internet-drafts/draft-ietf-idwg-idmef-xml-11.txt]
[7] Achim Leitner und Thomas Demuth, „Angriffstechnik im lokalen Netz – ARP-Spoofing und -Poisoning“: Linux-Magazin 06/04, S. 34
[8] Marius Aamodt Eriksen und Niels Provos, „Enges Korsett – Systrace setzt Regeln für erlaubte Systemaufrufe durch“: Linux-Magazin 01/03, S. 32
[9] Ettercap: [http://ettercap.sf.net]
[10] Geo-IP: [http://www.maxmind.com/geoip/api/c.shtml]
[11] Python-MySQL-Modul: [http://sourceforge.net/projects/mysql-python]
[12] FAM, File Alteration Monitor: [http://oss.sgi.com/projects/fam/]

Der Autor

Ralf Spenneberg arbeitet als freier Unix/Linux-Trainer und Autor. Er veröffentlichte 2002 sein erstes Buch „Intrusion Detection für Linux-Server“ und Ende 2003 „VPN mit Linux“. Demnächst wird sein drittes Buch „Intrusion Detection und Prevention mit Snort und Co.“ erscheinen.