Sicheren Linux-Debian-Server aufsetzen

== Vorwort ==
Diese Anleitung ist für fortgeschrittene Linuxbenutzer gedacht. Ich werde keinerlei Grundlagen erklären und auch nicht auf jede Einzelheit eingehen. Manche Befehle können in neueren Versionen abweichen, z. Bsp, Downloadlinks. Diese sollten dann bei Bedarf durch die neueren ersetzt werden. Dabei würde ich mich freuen, wenn man die Links dann auch direkt im Artikel abändern würde.
== SSH ==
Für die Kommunikation zwischen dem Server und dem Administrator kommt der SSH-Dienst zum Einsatz. Normalerweise sollte dieser schon auf dem Server installiert sein. Falls nicht, kann dies mit

Quellcode

```
1. aptitude install ssh-server
```

nachträglich gemacht werden.
Die SSH-Konfiguration sollte so angepasst werden, dass sich der Benutzer root nicht mehr direkt per SSH anmelden kann. Dazu wird die Zeile

Quellcode

```
1. PermitRootLogin yes
```

in

Quellcode

```
1. PermitRootLogin no
```

geändert. Dies hat den Vorteil, dass die meisten Login-Versuche fremder schon fehlschlagen und man für das erfolgreiche ***Kompremettieren*** mindestens zwei Passwörter braucht, das des normalen Benutzers und dass von root. Andererseits underbindet es die Faulheit des Administrators, wenn dieser nur eine kleine Aufgabe zu erledigen hat, welche auch un unpreveligierter Benutzer ausführen kann.
Weiterhin sollte der SSH-Daemon gegen Bruteforce-Attacken gesichert werden. Dies lässt sich einfach mit dem Programm denyhosts bewerkstelligen. Es sperrt einfach die IP-Adresse des PCs der sich mehrmals erfolglos versucht anzumelden.
Installiert wird denyhosts mit

Quellcode

```
1. aptitude install denyhosts
```

Danach sollte die Konfigurationsdatei von denyhosts, welche unter /etc liegt, bearbeitet werden.
Zuerst wird die Frist definiert, nach der die Einträge wieder gelöscht werden sollen. Dazu verändern wir die Zeile

Quellcode

```
1. PURGE_DENY =
```

in

Quellcode

```
1. PURGE_DENY = 5d
```

Damit werden Einträge, die älter als fünf Tage sind gelöscht. Dann sollte noch die Zeile mit dem Inhalt

Quellcode

```
1. DENY_THRESHOLD_INVALID = 5
```

überprüft werden. Hier wird die Anzahl der Versuche definiert, die man hat, bevor denyhosts die IP auf die Sperrliste setzt.
== Apache ==
Jetzt wird der Webserver. hier Apache2, mit PHP und MySql installiert. Dazu gibt man in der Shell folgendes ein:

Quellcode

```
1. aptitude install apache2 php5 mysql libapache2-mod-php5
```

Damit wird ein solides Grundsystem installiert, welches den meisten Anforderungen erst einmal ausreichen sollten.
Für die komfortable Administration bzw. Konfiguration des MySQL-Datenbanksystems wird PHPMyAdmin mit dem Befehl

Quellcode

```
1. aptitude install phpmyadmin
```

installiert.
Jetzt wird PHPMyAdmin entsprechend abgesichert, so dass niemand mehr das übertragene Passwort entschlüsseln kann. Dazu muss man ein SSL-Zertifikat erstellen. Für den privaten Gebrauch sollte es ausreichen, wenn man das Zertifikat selbst signiert. Dadurch wird zwar z. Bsp. der Firefox eine entsprechende Warnung anzeigen, wenn wir auf die Seite gehen möchten, diese kann jedoch getrost ignoriert werden.
Wer sein Zertifikat für andere glaubwürdiger machen möchte, kann auf CaCert als kostenlose Certificate Authority nehmen, jedoch werden die Warnungen erst nach dem Import des Rootzertifikates von CaCert verschwinden. Alternativ kann man natürlich auch ein Zertifikat von Thawte oder VeriSign für mehrere Hundert Euro im Jahr nehmen.
Das Zertifikat wird mit

Quellcode

```
1. make-ssl-cert generate-default-snakeoil --force-overwrite
```

erzeugt und gleichzeitig unterschrieben.
Genaueres kann man in /usr/share/doc/apache2.2-common/README\Debian.gz nachlesen.