

Serverdokumentation

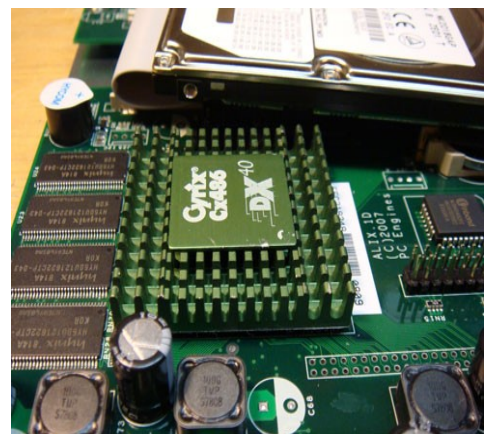
© Hermann Holthaus

Freien, den 02.05.09

Hier beschreibe ich wie man einen Eisfair-Server aufsetzt und als Wetterrechner laufen lässt. Einige grundlegende Kenntnisse in Linux sollten vorhanden sein.

Als Hardware dient ein Alix1d Board der Firma PC-Engines, es ist etwa so groß wie ein DIN A5 Blatt und arbeitet mit einem AMD Prozessor der mit 500MHz läuft. Der Speicher ist 256MB groß.

Weiterhin hat das Board einen CF-Slot, einen IDE und einen Mini PCI Anschluss (für Wlan oder so). Ein normaler PCI Bus ist auch vorhanden. Nach außen sind VGA, RS232, 2xUSB, Maus/Tastatur, Sound und Spannungsversorgung vorhanden. Zur Versorgung reicht ein 12V Steckernetzteil mit 1,2 A, das Board benötigt etwa 5-6 Watt Leistung.



Der Aufbau wurde noch etwas geändert, die CPU hat einen alten Kühlkörper vom einem 486er bekommen. Weiterhin wurden im Gehäuse noch 2 LED für Power und HDD angebracht.

Das mitgelieferte Gehäuse wurde außerdem noch mit einigen Lüftungslöchern versehen.



Zur Installation des Betriebssystems ist es am einfachsten, die Laptopplatte in einen normalen PC zu stecken und von der CD-ROM das Eisfair (Version 1.5.5.c) zu booten und auf der Platte zu installieren. Die Konfiguration wird hinterher auf dem Zielrechner gemacht. Nachdem das Grundsystem auf der Platte ist, kann man den PC herunterfahren und die Platte an das Alix-Board anschließen. Nach Anschluss von Tastatur und Bildschirm sollte das Eis hochfahren. Als erstes wird

die Konfiguration so bearbeitet, dass der Rechner eine Netzwerkverbindung aufbauen kann und sich die zusätzlichen Pakete vom Server ziehen kann. Der erforderliche Treiber für die Netzwerkkarte ist: *via-rhine*. Zur Konfiguration hält man sich erst einmal an die Hilfe zum Eissystem. Entweder man loggt sich mit der Tastatur direkt ein oder über das Netzwerk mit ssh, was ich persönlich einfacher finde.

Folgende Pakete sind zum sinnvollen Betrieb als Wettercomputer notwendig:

- Netdate
- Inetservices
- Apache2
- PHP5 für Apache
- Webalizer
- GCC
- Mysql 5.0.67
- Header Library
- PHPmyadmin
- Development Environment for Eisfair

Nachdem die Pakete einigermaßen eingestellt sind und laufen, macht man sich an die open2300 Software.

Herunterladen kann man sich die Software hier: http://sourceforge.net/project/showfiles.php?group_id=103374

Nach dem Entpacken des Paketes mit *tar -xvzf open2300-1.10.tar.gz* wechselt man in das neue Verzeichnis „open2300-1.10“. Mit *make all* werden die einzelnen Programme erzeugt und mit *make install* nach */usr/local/bin* installiert. Einzelheiten dazu stehen in der Hilfedatei zum Paket. Die zentralen Einstellungen werden in der *open2300.conf* Datei gemacht. Diese Datei dann in das richtige Verzeichnis kopieren, siehe Anleitung open2300.

Als nächstes kommt die *mysql2300.c* dran. Diese Datei wurde durch die o.g. Vorgänge nicht automatisch kompiliert.

Also, noch mal Aufruf von *make mysql2300*. Die Ausgabe am Prompt sieht dann so aus:

```
eis # gcc -Wall -O3 -DVERSION=\"1.10\" -o mysql2300 mysql2300.c rw2300.c linux2300.c -lm  
-I/usr/include/mysql -L/usr/lib/mysql -lmysqlclient -lz .....Das „-lz“ ist jetzt noch nicht vorhanden,  
es muss von Hand eingefügt werden.
```

Also: Achtung: beim *make mysql2300* unbedingt *-lz* anfügen !!

Jetzt sollte der Compiler ohne Fehler durchlaufen. Die so entstandene Datei wird in das gleiche Verzeichnis wie die anderen open2300 Daten kopiert. Dann noch die Rechte vergeben und fertig.

Dann kann man sich an die Einrichtung der Datenbank machen. Dazu benutzt man am besten das Tool "PHPmyadmin". Als erstes wird eine neue Datenbank angelegt:

- Datenbankname: open2300

Dann noch einen Benutzer anlegen:

- Benutzername: open2300
- Benutzerpasswort: mysql2300

Der Benutzer bekommt dann noch alle Rechte und dann wird in der Datenbank eine Tabelle erzeugt



mit dem Namen weather: einfach auf importieren klicken und die Datei mysql2300.sql hoch laden. Jetzt kann man zum testen den Befehl mysql2300 eingeben, es sollte keine Fehlermeldungen geben. Damit die Programme hinterher auch automatisch funktionieren ist es notwendig in der crontab entsprechende Einstellungen vorzunehmen. Also im Hauptmenü von Eis unter „1-System Administration“, dann „6-cron Administration“ den Punkt Edit Konfiguration aus wählen.

Die Konfiguration sollte in etwa so aussehen:

`/etc/config.d/cron`

`START_CRON = yes`

`CRON_N = 1`

`CRON_1_NAME = mysql2300`

`CRON_1_ACTIVE = yes`

`CRON_1_TIMES = 0-59/5 * * * *`

`CRON_1_COMMAND = root /usr/local/bin/mysql2300`

Dann schließt man den Editor und die Programme laufen. Jetzt werden alle 5min neue Werte in die Datenbank geschrieben.

Für den ersten Test schließt man die WS2300 an und lässt das System etwa 15min laufen. Dann schaut man in der Datenbank nach ob die entsprechenden Daten geschrieben wurden, wenn ja, dann läuft die Software richtig.

Zum Schluss kommt die Webseite dran, die Daten sollen ja schließlich dem Nutzer per Internet zugänglich gemacht werden. Das Programmpaket WeatherOffice-0.5 aus dem Internet herunterladen <http://sourceforge.net/projects/weatheroffice> und in den Ordner /var/www/htdocs/ entpacken. Der Packer erzeugt das Verzeichnis weatherOffice-0.5. Hier findet man jetzt die PHP Seiten usw. In den entsprechenden Dateien müssen jetzt nur noch diverse Einstellungen und Eintragungen vorgenommen werden, ganz nach persönlichem Bedarf.

Mit dem Webbrowser ruft man jetzt die Seite auf, die Daten sollten alle vorhanden sein.

Zusätze....

Für die Datensicherung habe ich dem System einen kleinen USB-Stick gegönnt, so ca. 4GB.



Unter dem Eis-Menü den Stick mounten und in der Mysql-Config den Backuppfad entsprechend einstellen, die automatische Sicherung einschalten und schon hat man eine tägliche Sicherung der Datenbank auf einem externen Medium.

Eine weitere Änderung betrifft den Dienst Netdate, hier ist es notwendig die Adresse des aufgerufenen Dienstes zu ändern. Bei der vorkonfigurierten Einstellung kommt es vor dass keine Synchronisation mit dem Zeitserver stattfindet. Ich habe hier den Zeitserver : „tcp ntp2.fau.de“ eingetragen. Diese Einstellung läuft ohne Fehler.