

# Aufbau und Installation eines Webmail Servers unter Suse Linux 9.0 mit Openwebmail als Web-Oberfläche



# Inhaltsverzeichnis

<b>1</b>	<b>Vorwort</b>	<b>5</b>
<b>2</b>	<b>Projektinformationen</b>	<b>8</b>
<b>3</b>	<b>Projektvorbereitung</b>	<b>9</b>
3.1	W-Fragen .....	10
3.2	Gruppenziele .....	12
3.3	Zeitplan .....	13
<b>4</b>	<b>Verwendete Software</b>	<b>14</b>
4.1	Die Distribution SuSe 9.0 .....	15
4.1.1	Einleitung .....	15
4.1.2	Preise .....	15
4.1.3	Pro .....	16
4.1.4	Contra .....	16
4.2	Der Apache2 Webserver .....	17
4.2.1	Einleitung .....	17
4.2.2	Aufbau .....	18
4.2.3	Architektur bis Version 1.3 .....	18
4.2.4	Laufzeitverhalten .....	18
4.2.5	Die neue Apache2 Architektur .....	20
4.2.6	APR (Apache Portable Runtime) .....	20
4.2.7	MPMs (Multi Processing Modules) .....	20
4.3	Postfix .....	21
4.3.1	Einleitung .....	21
4.3.2	Queues .....	21
4.3.3	Aufbau .....	22
4.4	Openwebmail .....	24
4.4.1	Einleitung .....	24
4.4.2	Unabhängig durch Web-Login .....	24
4.4.3	Umfangreiche benutzerdefinierte Voreinstellungen .	24
4.4.4	Komfortables Mailsystem .....	24
4.4.5	Datenbank-kompatibles Adressbuch .....	25
4.4.6	Kalender jederzeit im Web verfügbar .....	25
4.4.7	Aufbau .....	25

## Inhaltsverzeichnis

4.5	Das Adduserscript .....	26
4.5.1	Einleitung .....	26
4.5.2	Funktion .....	26
4.6	Quota .....	27
4.6.1	Einleitung .....	27
4.6.2	Funktion .....	27
4.7	Webmin .....	28
4.7.1	Einleitung .....	28
4.7.2	Funktion .....	28
<b>5</b>	<b>Installation - Konfiguration</b>	<b>29</b>
5.1	SuSe 9.0 Standardinstallation .....	30
5.1.1	Anmerkung .....	30
5.2	Apache2 Installation .....	31
5.3	Postfix Konfiguration .....	32
5.3.1	Postfix im Einsatz .....	34
5.3.2	Befehle .....	34
5.3.3	Oportional .....	34
5.3.4	virtual domains .....	34
5.3.5	Mail Relaying durch POP before SMTP .....	35
5.3.6	Virens scanner .....	36
5.4	Installation von Diskquotas mit SuSe .....	37
5.4.1	Benötigte Pakete .....	37
5.4.2	Installation .....	37
5.5.	Openwebmail Installation und Konfiguration .....	40
5.5.1	Benötigte Pakete .....	40
5.5.2	Datei und Verzeichnis Rechte .....	40
5.5.3	Konfiguration .....	41
5.6	Installation des Adduserscriptes .....	44
5.6.1	Benötigte Pakete .....	44
5.6.2	Installation .....	45
5.7	Boot-Runlevel ändern .....	47
5.8	Häufig auftretende Fehlermeldungen .....	48

## Inhaltsverzeichnis

<b>6</b>	<b>Anhänge - Projektmanagement</b>	<b>49</b>
6.1	Gruppenregeln . . . . .	50
6.2	Persönliche Zielsetzung . . . . .	50
6.3	Erfahrungsberichte . . . . .	51
6.3.1	Markus Bode . . . . .	51
6.3.2	Mario Valceschini . . . . .	52
6.4	Protokolle . . . . .	53
6.5	Quellen . . . . .	59
6.6	Abschlussklärung . . . . .	60
6.7	Notizen . . . . .	61

# 1 Vorwort

Sehr verehrter Leser,

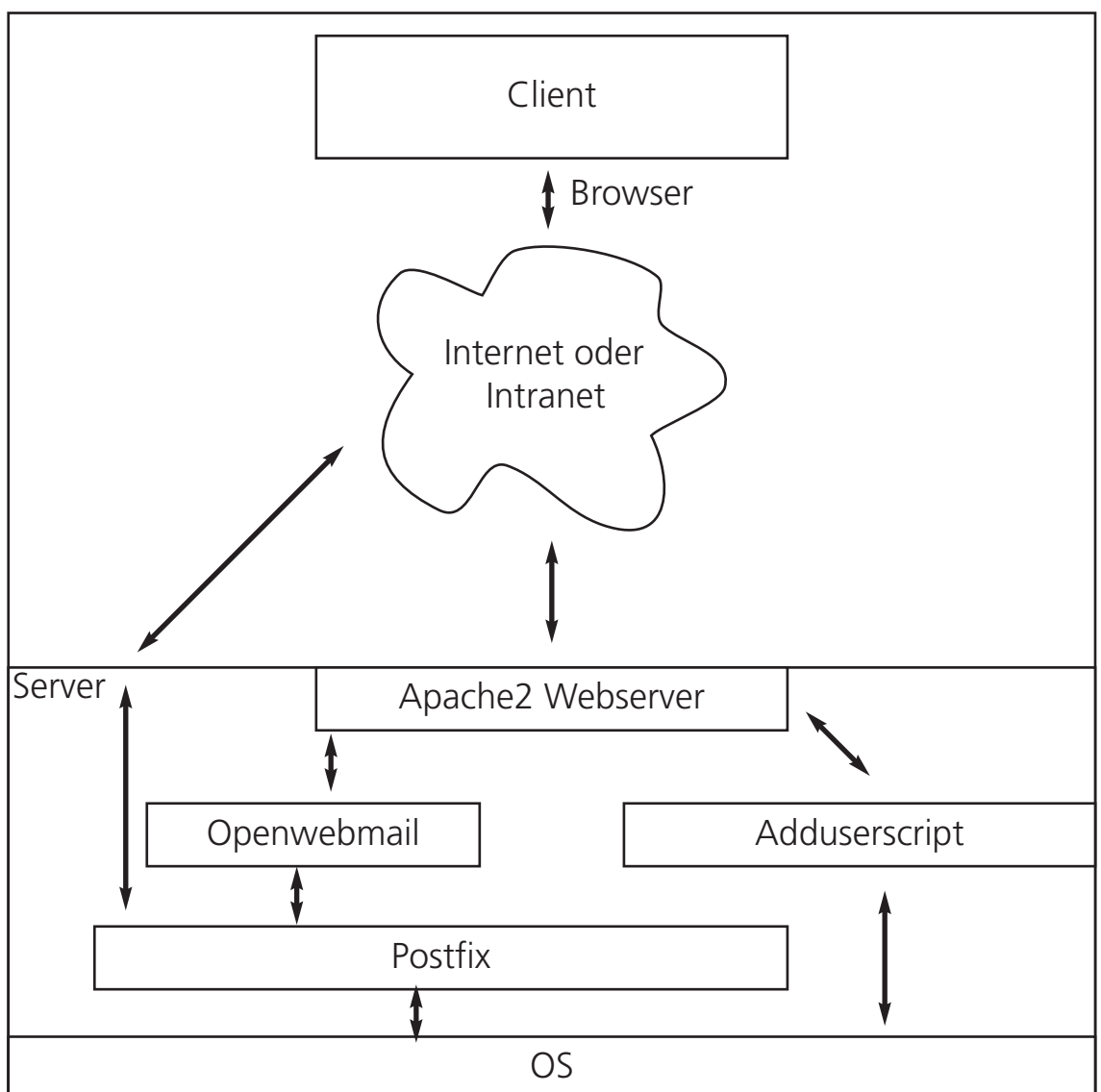
diese Dokumentation entstand im Rahmen eines Projekts der Klasse HBFM02 der Berufsbildenden Schule Neustadt a.d. Weinstrasse im Fach Kommunikationstechnik. In unserer Ausbildung zum staatlich geprüften Medienassistenten müssen wir in den 2 Jahren der schulischen Ausbildung pro Jahr ein größeres Projekt durchführen. Im ersten Jahr war das Projekt im Fach Projektmanagement angesiedelt. Dort erlernten wir die grundlegenden Techniken der Projektdurchführung, an welche wir uns bei der Durchführung unseres Projekts halten sollten. Im zweiten Jahr der Ausbildung durften wir uns das Projekt frei aussuchen. Da wir uns mehr für die Technik als für die Gestaltung interessieren, haben wir uns für ein Projekt im Bereich der Netzwerktechnik entschieden.

Nach Absprache mit Herrn Scheib entschieden wir uns für den Aufbau und Installation eines Postfix Webmail Servers unter dem Betriebssystem Linux, Distribution SuSe 9.0 mit Verwendung von Openwebmail als Webmail-Oberfläche. Durch dieses Projekt soll die E-Mail Kommunikation zwischen den Lehrern und Schülern an der BBS NW vereinfacht werden. Zunächst jedoch soll der Server seine Dienste im LAN (Local Area Network) der BBS NW zur Verfügung stellen, später wird er als öffentlicher E-mail Dienst der BBS-NW eingerichtet werden. Die Hard- und Software wurde uns von der BBS-NW zu Verfügung gestellt.

Da wir beide kaum Linux-Erfahrung hatten, wussten wir am Anfang des Projekts noch nicht genau wie wir vorzugehen hatten. Durch umfangreiche Webrecherche zu diesem Thema fanden wir jedoch nach längerem Suchen einige brauchbaren Seiten, welche uns bei der Installation und Konfiguration unterstützten.

Um die Abhängigkeit der Programme zueinander etwas zu verdeutlichen, finden Sie hier eine kleine Skizze der verwendeten Programme und deren Schnittstellenkommunikation.

Der Client kommuniziert mit dem Server über ein vorhandenes Intranet oder Internet. Dabei werden die Daten an den von den Programmen verwendeten Ports abgeliefert. Um mit Openwebmail seine E-Mails verwalten zu können, wird der Apache2 Webserver benötigt. Dabei ist Openwebmail nur eine Oberfläche für Postfix. Das Adduserscript unterstützt den Administrator, da es User selbstständig anlegen kann.



### Die Distribution SuSe 9.0:

Wir haben uns für eine Linux Distribution von SuSe entschieden, da diese auch für Anfänger geeignet ist. Diese Distribution der deutschen Firma SuSe bietet die beste deutsche Sprachunterstützung. Die Handbücher sind sehr umfangreich und verständlich geschrieben.

### **Das Paket Quota:**

Dieses Paket befindet sich auf dem SuSe 9.0 Installations Medium. Mittels der Software Quota lassen sich für beliebige User Plattenplatzbeschränkungen anlegen. Dies ist für einen Webmail-Dienst wichtig, da jeder User eine bestimmte Postfachgröße haben sollte.

### **Der Apache2 Webserver:**

Der Apache2 Webserver ist ein leicht konfigurierbarer Webserver. Zu diesem gibt es viele gute Dokumentationen. Durch seinen modularen Aufbau wurden die Portierungen auf verschiedene Betriebssysteme vereinfacht. Dies macht ihn zum weitverbreitetsten Webserver und er befindet sich auch auf dem SuSe 9.0 Installations Medium.

### **Der Mailedienst Postfix:**

Der MTA (Mail Transfer Agent) Postfix ist eine verbesserte Version des weitverbreiteten Sendmail. Postfix besitzt bei Ordnerstruktur und Konfigurationsdateien viele Ähnlichkeiten zu Sendmail. Durch seinen modularen Aufbau gibt es viele Erweiterungen, u.a. zum Virenschutz. Der Mailedienst wird standardmäßig bei der Default-Installation mitinstalliert.

### **Die Webmail-Oberfläche Openwebmail:**

Die Webmail-Oberfläche Openwebmail ist eine verbesserte Version von NeoMail. Openwebmail wurde mit einer Anderen Oberfläche versehen und mit vielen Features. Es beinhaltet umfangreiche Möglichkeiten der Mail-, Adressen- und Webspaceverwaltung.

### **Das Adduserscript:**

Um dem Administrator des Mailserver die Arbeit der Benutzererstellung abzunehmen, ist auf dem Server ein Adduserscript installiert. Mit diesem Script können sich Benutzer neue Accounts für den Zugang zu Openwebmail anlegen, ohne dass sie dabei Zugriff auf das Linux-System erhalten.

### **Das Linux Konfigurationstool Webmin:**

Um die administrative Arbeit für Linux etwas zu erleichtern haben wir das Linux Konfigurationstool Webmin installiert. Es vereint umfangreiche Einstellungsmöglichkeiten zu vielen Programmen unter Linux. Dabei ist es auch beliebig durch Module erweiterbar.

## 2 Projektinformationen

**Klasse:**

HBFM 02 BBS-NW

**Projektgruppe:**

Kommunikationstechnik 1

**Mitglieder:**

Markus Bode, Mario Valceschini

**Projektbetreuung:**

Herr Scheib

**Projektthema:**

Aufbau eines Webmaildienstes unter Verwendung von Apache und Postfix auf einem SuSe 9.0 Linux Server.



# 3 Projektvorbereitung

## 3.1 W-Fragen

### Was?

---

Aufbau eines Webmaildienstes unter Verwendung von Apache und Postfix auf einem SuSe 9.0 Linux Server.

### Wer?

---

Markus Bode und Mario Valceschini

### Warum?

---

Da jeder sich bei einem Freemail Anbieter ein Konto einrichten kann, gibt es keine richtige Übersicht über die E-Mail Adressen. Nicht jeder hat eine geeignete E-Mail Adresse die er für die Schule nutzen möchte. Um die E-Mail Kommunikation zwischen Lehrern und Schülern zu vereinfachen.

### Wozu?

---

Leichtere E-Mail Adressenverwaltung für BBS. Um eine gemeinsame Kommunikationsplattform aufzubauen, die dann die Kommunikation zwischen den verschiedenen Klassen fördern soll. Schüler können schneller über bevorstehende Aktionen benachrichtigt werden, z.B. über Stundenausfall oder Stundenplanänderung.

### Mit Wem?

---

Direkter Ansprechpartner ist Herr Scheib. Bei fachlichen Fragen wird auf die entsprechenden Fachlehrer zurückgegriffen.

### Für Wen?

---

Die Lehrer und Schüler der Schule BBS-NW.

### Wie?

---

Der Ablauf des Projektes ist durch die Anleitung des Scripts festgelegt und muss genau eingehalten werden. Die im Unterricht erlernten Methoden werden entsprechend der jeweiligen Projektphase angewendet.

## W-Fragen

Womit?

---

Die benötigten Mittel werden von der Schule gestellt.

Wann?

---

Während der Schulzeit arbeitet die Gruppe in der Schule, ansonsten kann außerhalb der Schule gearbeitet werden.

Wo?

---

Der Arbeitsraum, in dem die Projektgruppe zusammenarbeitet, ist B210.

## 3.2 Gruppenziele

- Die Installation und Konfiguration des Webmailservers soll zu dem festgelegten Termin abgeschlossen sein.
- Die Funktionen des Webmailservers sollen für alle Schüler der BBS-NW nutzbar sein.
- Die Dokumentation soll komplett und fristgemäß abgegeben werden.
- Die Weboberfläche soll überschaubar und verständlich gestaltet sein.

## 3.3 Zeitplan

- 1.) **Vorbereitung / Webrecherche**  
01.10.2003 - 15.11.2003
- 2.) **Festlegung der Ziele**  
16.11.2003 - 23.11.2003
- 3.) **Installation der Programme**  
24.11.2003 - 04.01.2004
- 4.) **Einarbeiten in Linux, Apache und Anwendersoftware**  
05.01.2004 - 12.01.2004
- 5.) **Konfiguration der Software**  
14.01.2004 - 18.01.2004
- 6.) **Testphase**  
19.01.2004 - 20.01.2004
- 7.) **Dokumentation (Gestaltung des Layouts)**  
21.01.2004 - 26.01.2004
- 8.) **Erstellen der Präsentation**  
27.01.2004 - 13.02.2004
- 9.) **Präsentationsvorbereitung (Materialien, Wahl der Medien)**  
14.02.2004 - 24.02.2004

# 4 Verwendete Software

## 4.1 Die Distribution SuSe 9.0



### 4.1.1 Einleitung:

---

Linux ist ein kostenloses Betriebssystem, welches unter der GPL (GNU Public Licence) verteilt wird. Linux wurde 1991 als Studienübung von Linus Torvalds entwickelt. Es ist keine Freeware, sondern unterliegt bestimmten Lizenzbedingungen. Die GPL soll sicherstellen, dass die Software für alle Benutzer frei ist. D.h. die unveränderte Software ist beliebig oft kopierbar. Die eigenen Programm Kopien dürfen verändert, sofern die Änderungen kenntlich gemacht und veröffentlicht werden. Unabhängig davon darf jeder eigene Distribution (Softwaresammlung) erstellen und anderen Leuten zur Verfügung stellen. Für die Distribution und eventuelle Zusatzleistungen (bei SuSE sind es das Installationsprogramm und Handbuch) darf Geld verlangt werden, nicht aber für die Linux-Software, welche kostenlos sein muss. SuSe verwendet das Redhat rpm Paketsystem. Diese Distribution ist Einsteigerfreundlich. Das Programm Yast hilft komfortabel das Betriebssystem zu administrieren. Da wir beide nicht soviel Linux Erfahrung hatten, war diese Distribution genau das richtige für uns.

### 4.1.2 Preise:

---

Je nach Version ist dieser unterschiedlich:

Die Personal, Professional Campusversion	kostet ca 49 Euro.
Die Professionalversion	kostet ca 79 Euro.

Stand: Anfang 2004

Zu beziehen u.a. bei Amazon.de.

## 4.1 Die Distribution SuSe 9.0

### 4.1.3 Pro:

---

- + Sehr einfache Installation
- + Gute Unterstützung von ISDN und DSL
- + Beste Deutsche Sprachunterstützung, da deutsche Distribution
- + Zentrales, einheitliches, grafisches Administrationstool (Yast2 oder Yast)
- + Guter Support, gute Support- und Hardwaredatenbank
- + Gute Handbücher werden mitgeliefert
- + Alle wichtigen Programme vorkompiliert
- + Live-CD erhältlich (zum Testen ohne Installation auf der Festplatte)
- + DVD wird mitgeliefert, kein Wechseln der CD während der Installation erforderlich

### 4.1.4 Contra:

---

- Bei der Personal-Version fehlen viele wichtige Entwicklerpakete
- Viele rpm-Pakete für Redhat funktionieren nicht

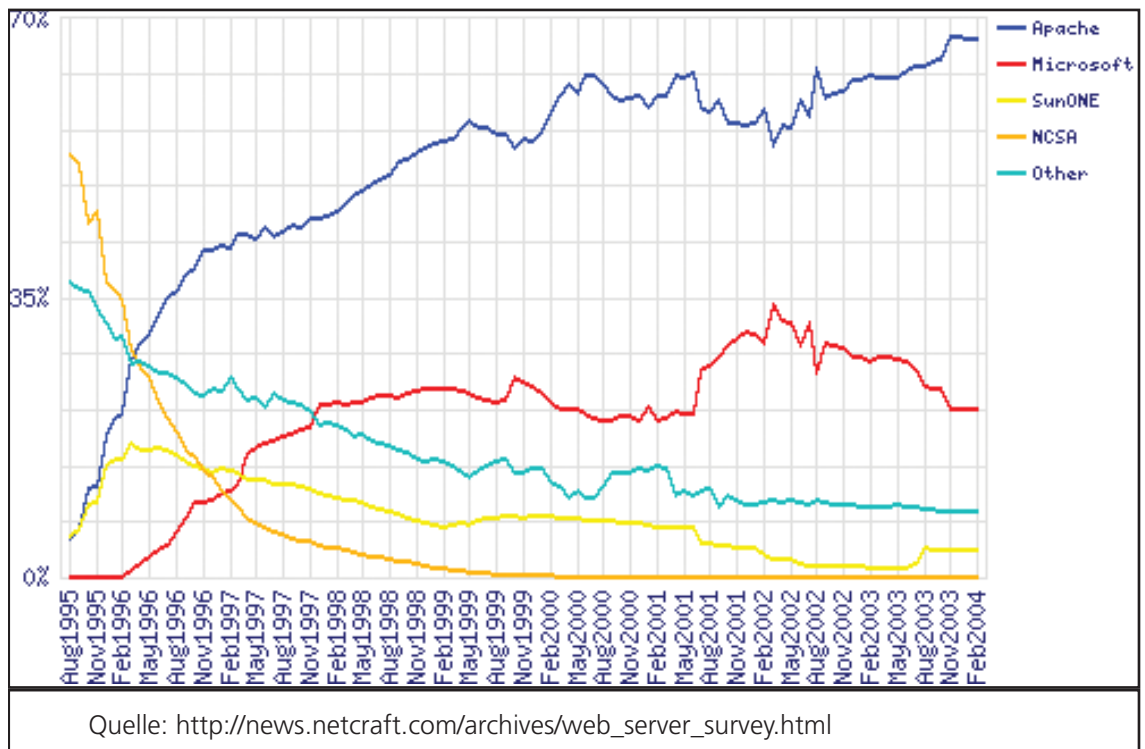


## 4.2 Der Apache2 Webserver



### 4.2.1 Einleitung:

Der Apache Webserver ist ungefähr seit 6 Jahren der am meisten verwendete Webserver weltweit. Seit der Apache Version 1.3 gibt es Portierungen für Windows, OS2 und BeOS. Das Apache Projekt ist heute, neben Linux selbst, das erfolgreichste und bekannteste Open-Source-Projekt. Es entstand Mitte der 90er aus dem NCSA Webserver, der zu dieser Zeit weit verbreitet war. Da dessen Entwicklung jedoch eingestellt wurde, gründeten ein paar Programmierer die Apache Group.



## 4.2 Der Apache2 Webserver

### 4.2.2 Aufbau:

---

Die meisten Apache2-Webserver-Konfigurationsdateien befinden sich in */etc/apache2/*. Die Hauptkonfigurationsdatei ist die Datei *httpd.conf*. In ihr werden die wichtigsten Einstellungen und die zu ladenden Module festgelegt.

### 4.2.3 Architektur bis Version 1.3:

---

Bis zur Apache Version 1.3 blieb die Grundarchitektur des Webserver weitgehend gleich.

### 4.2.4 Laufzeitverhalten:

---

Bis zur Apache Version 1.2.x unterstützte der Webserver nur die verschiedenen Unix-Betriebssysteme.

Der Apache Webserver arbeitete als sogenannter Preforking-Server: Der Vater-Prozess legt nach seinem Start eine in der Konfigurationsdatei mit der Option „MinSpareServers“ festgelegte minimale Anzahl von Kopien von sich selbst (Kind-Prozessen) an. Diese Kind-Prozesse warten auf eingehende http-Anfragen. Falls mehr http-Anfragen eingehen, als die vorhandenen Kind-Prozesse beantworten können, werden rechtzeitig weitere Kind-Prozesse des Ursprungsprozesses erstellt. Bei Überschreitung von der eingestellten Anzahl „MaxSpareServers“ werden diese wieder gelöscht.

Der Apache Webserver wurde in der Version 1.3 auf die Windows-Plattform portiert.

Dabei gibt es unter Windows eine Besonderheit. Im Gegensatz zu UNIX-Betriebssystemen ist das Kopieren laufender Prozesse nicht möglich. Als einzige Möglichkeit kann man komplett neue Prozesse erstellen. Dies würde aber bedeuten, dass die gesamte Startphase von Apache Webserver inklusive der Verarbeitung der Konfigurations-direktiven durchlaufen werden muss.

Um den Apache Webserver auch unter Windows effektiv nutzen zu können, wurde ein anderes Laufzeitverhalten implementiert.

Daher gibt es unter Windows zwei Apache-Prozesse:

Der erste ist für die Beantwortung der ankommenden HTTP-Anfragen zuständig, der zweite überwacht den ersten Prozess, um bei einem Absturz diesen neu zu starten.



## 4.2 Der Apache2 Webserver

Innerhalb des ersten Prozesses, der die http-Anfragen bearbeitet, laufen mehrere einzelne Threads. Wie bei UNIX werden je nach Beanspruchung der Threads neue erstellt oder alte aus dem Speicher entfernt.

Bei der Portierung wurden die Teile des Apache-Quelltext, die für die Ursprungs-Prozess-Erzeugung zuständig waren so umgeschrieben, dass die beiden Laufzeitverhalten (Threads/Prozesse) realisiert werden konnten.

Um nach der Portierung im Code noch zwischen den beiden Versionen unterscheiden zu können, wurden *#ifdef*-Anweisungen verwendet.

Eine *#ifdef*-Anweisung kommt aus der C/C++ Programmiersprache und überprüft, ob ein Name mit der Anweisung *#define* festgelegt worden ist, bei welcher ein Symbol definiert wird. Dieses entspricht einem konstanten Wert. Der Compiler ersetzt im Quelltext alle Vorkommen dieses Symbols durch den Wert, welches das Symbol repräsentiert. Wenn die Anweisung *#define* festgelegt wurde, wird der nachfolgende Block ausgeführt, ansonsten wird er übersprungen.

Durch die Portierung wurde der Code unübersichtlicher und dadurch schwierig zu pflegen, insbesondere wenn weitere Portierungen implementiert werden. Um Apache wieder leichter portierbar zu machen, wurde eine neue modulbasierende Architektur entwickelt.

## 4.2 Der Apache2 Webserver

### 4.2.5 die neue Apache 2 Architektur:

---

Bei der neuen Architektur wurden alle plattformspezifischen Codeteile aus den Quelltexten vom restlichen Apache-Code getrennt. Durch diese Kapselung lässt sich der Apache Webserver wieder leichter portieren. Nun gibt es eine Unterscheidung in APR und MPM Programmteile:

### 4.2.6 APR (Apache Portable Runtime):

---

In der neuen Architektur benutzt der Apache Webserver nicht mehr die POSIX-Schnittstellen. Auf nicht-Unix-Betriebssystemen konnten diese bei falscher Implementation zur Folge haben, dass der Server dort nicht zufriedenstellend lief.

Die Apache Portable Runtime ist eine Bibliothek. Diese bildet zwischen den jeweiligen Betriebssystemen und dem Apache 2 Webserver eine Schnittstelle. Das APR besitzt eine eigene API und bietet grundlegende Funktionen eines Betriebssystems. Eine API ist eine Schnittstelle zur Programmierung von Anwendungen.

Dazu gehören unter anderem File-I/O, Netzwerk-I/O, Speicher-verwaltung, Thread- und Prozessverwaltung u.s.w.

Die APR verwendet bei jeder Gelegenheit native System-Aufrufe.

Um den Programmierern Portierungen zu erleichtern, gleichen sich die APR-Methoden den POSIX-Methoden äußerlich.

### 4.2.7 MPMs (Multi Processing Modules):

---

In diesen Modulen befindet sich der betriebssystemspezifische Teil des Codes. Dieser ist für die Verwaltung von Prozessen und/oder Threads zuständig. Je nach verwendetem MPM werden Prozesse oder Threads verwendet.

Dadurch ist der Quelltext strukturierter und die einzelnen MPMs können auf die jeweiligen Betriebssysteme optimiert werden.

Dies führt bei Nicht-Unix-Betriebssystemen im Vergleich zur alten 1.3er Architektur zu Geschwindigkeitssteigerungen. Für UNIX gibt es verschiedene MPM Module. Diese besitzen unterschiedliche Laufzeitverhalten und werden ständig weiteroptimiert.

## 4.3 Postfix



### POSTFIX

#### 4.3.1 Einleitung:

Wietse Venema ist der Autor des Projekts *VMailer*. Da dieser Name aber einem von IBM eingetragenen Markenzeichen zu ähnlich war, musste er das Projekt umbenennen. Er wollte bei seinem Projekt ein schnelles, einfach zu administrierendes und sicheres Programmpaket entwickeln. Nebenbei sollte das Programm auch zu Sendmail kompatibel sein. Der modulare Aufbau von Postfix ist in der Grafik von linux-magazin.de verdeutlicht. Die Kommunikation der Module findet über die UNIX-Domain-Sockets statt. Dadurch kann man Fehler leichter finden und Sicherheitslöcher schneller stopfen. Durch die höhere Sicherheit gegenüber Sendmail, das zwingend als root laufen muss, kann der Angreifer unmöglich root-Rechte bekommen, da Postfix einen eigenen User für die Administration anlegt.

Um die Sicherheit weiter zu erhöhen arbeitet Postfix mit vier verschiedenen Queues: *maildrop*, *incoming*, *active* und *deferred*. Postfix kann aber immer nur eine einzige Mail gleichzeitig bearbeiten, weshalb es seine *active* Queue klein hält, was Ressourcen spart.

#### 4.3.2 Queues:

---

##### **maildrop**

Hier landen alle lokal gesendeten Mails und werden nach Überprüfung in die "incoming" Queue kopiert.

##### **incoming**

Aus dieser Queue werden die Mails in die "active" Queue verschoben und nach und nach abgearbeitet.

##### **active**

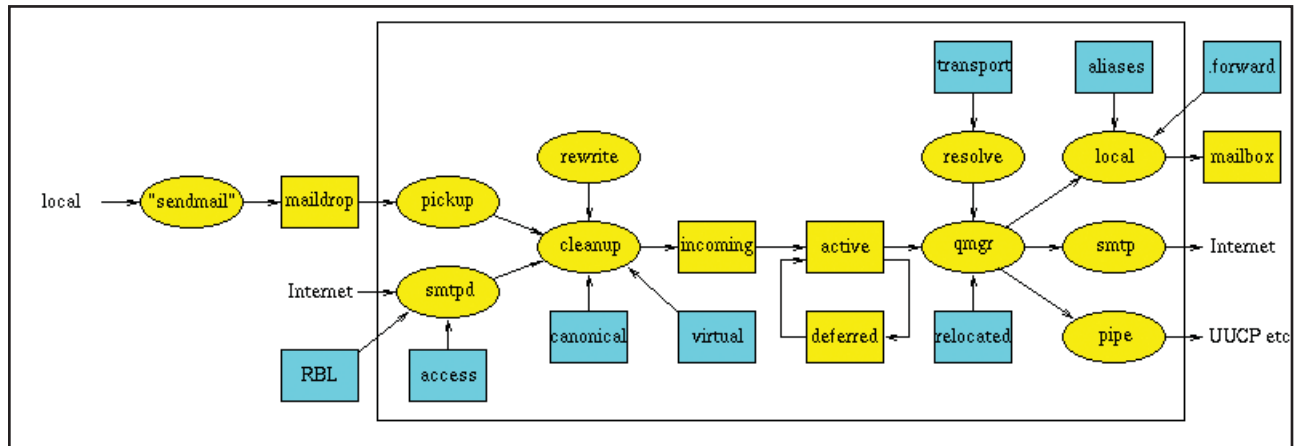
Hier landen alle Mails, die durch den Postfix Queue-Manager gerade bearbeitet und (lokal oder remote) ausgeliefert werden.

##### **deferred**

Hier landen alle Mails, die Postfix nicht ausliefern kann.

## 4.3 Postfix

### 4.3.3 Aufbau:



Die Grafik zeigt den modularen Aufbau von Postfix. Hierbei bedeuten:

- \* gelbe Ellipsen Programme
- \* gelbe Kästen Mail-Queues oder -Dateien
- \* blaue Kästen (Nachschlage-) Tabellen
- \* Programme in der umrandeten Box, welche unter der Kontrolle des Postfix master Daemons laufen

Dateien in diesem Kasten gehören dem Postfix-Mail-System an.

Quelle: <http://www.linux-magazin.de/Artikel/ausgabe/2000/06/Postfix/postfix.html>

Bei der SuSe 9.0 Distribution befinden sich die Konfigurationsdateien von Postfix in */etc/postfix*. Nun folgt ein kleiner Überblick über die einzelnen Dateien.

*/etc/postfix/main.cf*: Dies ist die wichtigste Postfix Konfigurationsdatei. In dieser gut auf englisch dokumentierten Datei befinden sich alle grundsätzlichen Funktionen, die von den default-Werten aus */etc/postfix/main.cf.default* abweichen.

*/etc/postfix/master.cf*: Diese Datei dient den Optionen um die Ressourcen-Belegung von Postfix zu konfigurieren.

*/etc/postfix/access*: Konfiguration des E-Mail Einganges.

*/etc/postfix/aliases*: In dieser Datei werden lokale Email-Adressen anderen Adressen zugeordnet.

## 4.3 Postfix

*/etc/postfix/canonical*: Die Funktion dieser Datei entspricht */etc/aliases*. Dabei dient sie dazu externen E-Mail-Adressen neue Adressen zu geben. Eine Einstellung in dieser Datei wird nötig, da manche Mailclients Probleme mit der lokalen Adresse haben, bzw. diese nicht selbständig umsetzen können. Es können neben Empfängeradressen auch Absenderadressen ausgetauscht werden.

*/etc/postfix/relocated*: Falls sich eine E-Mail Adresse geändert hat, kann dies ähnlich wie in der Datei */etc/postfix/access* eingetragen werden.

*/etc/postfix/transport*: Hier können die verwendeten Protokolle zum Mail-Transport (z.B. SMTP oder UUCP) eingestellt und konfiguriert werden.

*/etc/postfix/virtual*: Diese Datei ist mit der Datei */etc/aliases* vergleichbar, allerdings hat sie erweiterte Optionen. Eine zusätzliche Option wäre z.B. externe Mail-Weiterleitung.

## 4.4 Openwebmail



### 4.4.1 Einleitung:

---

Open WebMail ist ein Webmail-System, das auf Neomail v1.14 von Ernie Miller aufbaut. Das Webmail-System bietet umfangreiche Möglichkeiten der Mail-, Adressen- und Webspaceverwaltung.

### 4.4.2 Unabhängig durch Web-Login:

---

Der Zugang zu dem angelegten Webmail-Account läuft in unserem Falle über die Systembenutzer mit ihren Namen und Passwörtern. Neue Benutzer können ganz einfach in UNIX mit adduser oder über verschiedene adduser-Scripte angelegt werden. Es können verschiedene Accounts unterschiedliche E-mail-Accounts über einen Zugang steuern. Es können aber auch über ein E-mail-Programm und einem dafür eingerichteten OpenWebmail-Web-Account E-mails parallel empfangen werden. Umfangreiche Programm-Hilfen sind auch vorhanden.

### 4.4.3 Umfangreiche benutzerdefinierte Voreinstellungen:

---

Es gibt die Möglichkeit der Sprachwahl aus über 30 Sprachen, der Designwahl der Programmumgebung, der Sicherungs-rythmen u.v.m.

### 4.4.4 Komfortables Mailsystem:

---

OpenWebmail besitzt umfangreiche Funktionen für den Empfang und die Versendung von E-mails. Die E-mail-Empfänger können direkt aus dem Adressbuch ausgewählt werden. Für sich selbst können mehrere E-mail-Aliase angelegt werden. Eine E-mail-Weiterleitung kann ebenso wie ein Autoresponder einfach eingerichtet werden. Die E-mails können übersichtlich in benutzerdefinierten Ordnern abgelegt werden. Über umfangreiche Suchfunktionen können Mailinhalte gefunden werden. Durch Mailfiltereinstellungen kann man sich vor Spamming und bekannten virenbehafteten Mails schützen. E-mail-Logfiles sind auf dem Server einsehbar.



## 4.4 Openwebmail

### 4.4.5 Datenbank-kompatibles Adressbuch:

---

Bestehende Adressen aus dem Outlook und Netscape Adressbuch können importiert oder jederzeit in andere Systeme exportiert werden. Das Adressbuch selbst ist übersichtlich aufgebaut und mit dem Mailsystem verbunden.

### 4.4.6 Kalender jederzeit im Web verfügbar:

---

Der integrierte Kalender erlaubt eine schnelle Übersicht über alle wichtigen eingetragenen Termine. Diese können in verschiedenen Ansichten (Tag, Woche, Monat, Jahr) angesehen werden. Es können sogar Dokumente und Webseiten über Hyperlinks eingebunden werden. Durch den für mehrere User zugänglich gemachten Kalender können einfache Möglichkeiten der Terminplanung genutzt werden.

### 4.4.7 Aufbau:

---

Der Standardinstallationsordner ist */srv/www/cgi-bin/openwebmail*. In diesem Ordner liegen die ganzen Perl-Scripte. Die Konfigurationsdateien befinden sich im Unterordner *etc/*. Die Hauptkonfigurationsdatei heißt *openwebmail.conf*, in dieser werden die von den Standardeinstellungen abweichenden Konfigurationsparameter gespeichert. Die Default-Einstellungen sind in der Datei *openwebmail.conf.default* gespeichert.

## 4.5 Das Adduserscript

### 4.5.1 Einleitung:

---

Das verwendete Adduserscript stammt von Arthur Corliss ([www.postman.net](http://www.postman.net)). Sein Script legt für Openwebmail Benutzer-Accounts an. Die Benutzer können mit ihrem Benutzernamen und Passwort Zugang zu Openwebmail erhalten.

### 4.5.2 Funktion:

---

Als erstes wird mit dem Befehl `checkacct` überprüft, ob der gewünschte Benutzer schon vorhanden ist. Das Script verwendet den Befehl `useradd`, mit dem es neue Benutzer anlegt. Dabei kann es die in der `adduser.conf` festgelegten quotas auf den neu erstellten Account anwenden. Dem neu erstellten Benutzer wird durch das Script auch ein vorher definiertes Homeverzeichnis angelegt. Darin liegen die Mails und der Webespace des Benutzers. Die angelegten Benutzer können sich aber nicht im System einloggen, da ihre verwendete Login-Shell `/bin/false` ist und diese keinen Login erlaubt.

## 4.6 Quota

### 4.6.1 Einleitung:

---

Mit Hilfe von Diskquota ist es möglich, jedem Benutzer eines Systems nur einen vom Administrator bestimmten maximal belegbaren Plattenplatz in einer Partition zuzuweisen. Wird der vordefinierte Wert überschritten, so erhält der Benutzer eine Warnung per E-mail. Verringert dieser nicht innerhalb eines definierten Zeitraums seinen Platzbedarf, so wird ihm das Weiterschreiben auf der Partition verwehrt, bis er wieder im Rahmen des ihm erlaubten Plattenplatzes ist.

### 4.6.2 Funktion:

---

In der Datei */etc/fstab* befinden sich alle gemounteten Partitionen und der verwendete Mountpunkt. In der Tabelle muss der Eintrag der Partition auf die Quota angewendet werden um den Eintrag *usrquota* ergänzt. Nun muss noch eine Option in der Datei */etc/rc.config* geändert werden. Der Eintrag *START\_QUOTA=no* muss auf *START\_QUOTA=yes* umgestellt werden. Dieser Eintrag setzt auch das Script *boot.quota*, das mit dem Befehl *insserv boot.quota* im Verzeichnis */etc/init.d* aufgerufen wird. Erst nach einem Neustart des Systems werden die Quotas angewendet. Am einfachsten können die Einstellungen über das Quota-Webmin-Modul vorgenommen werden. Um die Quotas aktuell zu halten müssen die Befehle */sbin/quotacheck -a* und */usr/sbin/warnquota* als cron-job ausgeführt werden. Die Quota Installation wird auf Seite 37 erklärt.

## 4.7 Webmin

### 4.7.1 Einleitung:

---

Webmin ist ein modulbasiertes Linux-Konfigurationstool, das von Jamie Cameron in Perl geschrieben wurde. Je nach vorhandenen eingebundenen Modulen können verschiedene Einstellungen einfach per Browser geändert werden.

### 4.7.2 Funktion:

---

Wenn Webmin installiert ist, kann man eine verschlüsselte Verbindung zu Webmin mittels Eingabe von <https://localhost:10000> in einem Browser aufrufen. Nach Eingabe von Benutzer root und dem dazugehörigen Passwort stehen viele Eingabemöglichkeiten offen. Durch Einbinden von neuen Modulen ist das Programm beliebig erweiterbar. Die Konfiguration der einzelnen Module erfolgt per Button oder Texteingabefeld im Browser. Diese Eingaben werden auf die Text-Konfigurationsdateien angewendet.

# 5 Installation - Konfiguration

## 5.1 SuSe 9.0 Standardinstallation

Bei der Installation werden alle vom SuSe 9.0 Installer gemachten Paketvorschläge übernommen. Für das einfache Arbeiten unter SuSe sollte die Sektion *Grafisches Grundsystem* ausgewählt werden.

Lediglich müssen später in der Installation die Netzwerkkarteneinstellungen je nach vorhandenem Netzwerk eingerichtet werden. In den Netzwerkkarten-Optionen kann man den Rechnernamen und die verwendete Domain eingegeben.

Die Domain und der Rechnername können aber auch später über *yast* nachträglich leicht eingestellt werden.

Im Falle der BBS-NW wurde der Rechnername *mailintern* und die Domain *bbs-nw.de* verwendet.

Während der Installation wurde das frisch installierte SuSe 9.0 System mit dem *Onlineupdate* auf den aktuellen Stand gebracht.

### 5.1.1 Anmerkung:

---

Für die folgenden Installationen und Konfigurationen muss man sich als *root* im System einloggen. Befehle, Variablen, Optionen und Parameter sind mit einer Kursiven Schrift ausgezeichnet.

## 5.2 Apache2 Installation

Der Apache2 Server wird am einfachsten über *yast* installiert.

Im *yast* muss man die Paketanzeige auf *Sektionen* umstellen.

Nun muss man alle Pakete in *Einfacher Webserver mit Apache2* auswählen, damit diese installiert werden können.

Zusätzlich werden die Pakete *Apache2-mod\_perl* und *apache2\_metuxmpm* über *suchen* ausgewählt.

Eine Konfiguration von Apache2 muss nicht vorgenommen werden, da die *default* Einstellungen für unsere Zwecke ausreichend sind.

Um diese Einstellungen zu überprüfen, muss man in der Datei mit *joe*

*/etc/apache2/httpd.conf*

bei *'Main' server configuration* überprüfen, ob

```
Include /etc/apache2/default-server.conf
```

verwendet wird. Nun sollte man noch in den beiden Dateien nachsehen, ob die Einträge

```
ScriptAlias /cgi-bin/ "/srv/www/cgi-bin/"
```

(gibt den vollständigen Pfad zur *ScriptAlias /cgi-bin/* an) und folgende Zugriffsberechtigungen für das Verzeichnis */srv/www/cgi-bin*

```
<Directory "/srv/www/cgi-bin">
  AllowOverride None
  Options +ExecCGI -Includes
  Order allow,deny
  Allow from all
</Directory>
```

gesetzt sind. Diese Berechtigungen werden für die CGI-Script-Ausführung benötigt.

## 5.3 Postfix Konfiguration

Postfix wird bei einer Standard SuSe 9.0 Installation schon mitinstalliert, deshalb muss es lediglich konfiguriert werden.

Da Postfix aus Sicherheitsgründen nicht als root laufen sollte, wird bei der Installation ein Benutzer Postfix angelegt.

Die Konfigurationsdateien liegen alle in einem einzigen Verzeichnis, nämlich in */etc/postfix*. Es muss nicht viel eingestellt werden. Die Defaulteinstellungen sind geeignet gewählt, um ein einfaches Mailsystem abzudecken.

Nur in der Datei */etc/postfix/main.cf* sind einige Einträge anzupassen.

Wenn der Mailserver rein intern betrieben werden soll, kann der Eintrag `smtp` auskommentiert werden. Dieser verhindert Zugriff von außen auf den Mailserver.

```
#smtp
```

Angabe des Hostnamens für den Mailserver

```
myhostname = mailintern.bbs-nw.de
```

Angabe der Domain für den Mailserver

```
mydomain = bbs-nw.de
```

Die Variable *myorigin* setzt den rechten Teil der Absenderadresse (*X@MYORIGIN*) der Mails fest. Wird diese Variable nicht verwendet, nutzt Postfix den *Hostnamen* als Absenderadresse, etwa *X@mailintern.bbs-nw.de*. Meist ist die Form der Adresse *X@bbs-nw.de* erwünscht. Vorausgesetzt der MX-Eintrag (Mail EXchanger) im Nameserver ist richtig gesetzt, kann man dies durch den Eintrag

```
myorigin = $mydomain
```

erreichen. Der MX-Eintrag liefert die Information, von welchem Host in der Domäne die E-Mail verarbeitet wird und an welches System eingehende E-Mails geliefert werden sollen.

Der *inet\_interfaces* Parameter gibt die Netzwerk-Interface-Adresse an, an der der Mailserver E-mails empfängt. Als Standardeinstellung nutzt Postfix alle aktiven Interfaces des Systems.

```
inet_interfaces = $myhostname
```



## 5.3 Postfix Konfiguration

Der *mydestination* Parameter gibt die Domain an, von der dieses System sich selbst als Ziel betrachtet.

```
mydestination = $myhostname
```

Der *unknown\_local\_recipient\_reject\_code* Parameter gibt den *SMTP Server Response Code* an, wenn eine Empfänger-Domain *\$mydestination* oder *\$inet\_interfaces* beinhaltet, auch wenn keine Variable in *\$local\_recipient\_maps* eingetragen wurde und die Empfänger-Adresse oder die lokale Adresse nicht gefunden wird. Der Code *550* bewirkt das die Mail sofort gesendet wird. Bei Code *450* kann es ein paar Stunden dauern, bis die Mail ankommt.

```
unknown_local_recipient_reject_code = 550
```

Setzt man diese Werte nicht explizit, ermittelt Postfix sie durch Systemaufrufe. Es bleibt noch anzumerken, dass jede zugewiesene Variable an beliebiger anderer Stelle der Konfigurationsdatei wieder verwendet werden kann.

Lediglich die beiden zentralen Dateien *main.cf* und *master.cf* müssen bearbeitet werden. Die Konfiguration der *main.cf* haben wir bereits abgeschlossen. In der Datei *master.cf* kann man den Ressourcenverbrauch von Postfix steuern. Hier kann unter anderem angegeben werden, wie viele Prozesse maximal gleichzeitig laufen dürfen. Darin müssen aber keine Werte geändert werden.

Falls das Programm noch nicht laufen sollte, wird es mit dem Befehl

```
rcpostfix start
```

gestartet. Wenn es schon läuft muss der Befehl

```
rcpostfix reload
```

einggegeben werden.

## 5.3 Postfix Konfiguration

### 5.3.1 Postfix im Einsatz:

#### 5.3.2 Befehle:

---

Ähnlich dem Sendmail Kommando *newaliases* ist *postalias* nun für die Mail-Aliases zuständig.

*postcat* zeigt in der aktuellen Postfix-Version den Inhalt der Queues an.

*postconf* zeigt die in der Datei main.cf gesetzten Werte an.

*postkick* dient als Schnittstelle zu internen Kommunikationskanälen, etwa für Shellskripts.

*postlock* stellt ein Postfix-kompatibles Mailbox-Locking zur Verfügung, z.B. für Shellskripts.

*postlog* stellt für externe Programme oder Shellskripts eine Postfix-kompatible Logging-Funktion bereit.

*postmap* stellt in etwa die selbe Funktion wie makemap zur Verfügung und generiert die Lookup-Tabellen, je nach System als hash (db) oder dbm.

*postsuper* ist das Tool, das die Mails in den verschiedenen Queues verschiebt. Das Postfix-Startskript führt dieses Kommando aus.

### 5.3.3 Opportional:

#### 5.3.4 virtual domains:

---

Postfix unterstützt auch *virtual domains*. Diese werden in der Datei */etc/postfix/virtual* standardmäßig eingerichtet.

Beispieleinträge:

*bbs-nw.de*  
*mario@bbs-nw.de*  
*markus@bbs-nw.de*

*dieser Teil der ersten Zeile ist beliebig*  
*mario-v@anderedomain.de*  
*markus-b@anderedomain.de*



## 5.3 Postfix Konfiguration

In der ersten Zeile wird die Domäne ohne Mailadresse angegeben. Der rechte Teil der Zeile ist beliebig. Als nächstes folgen die E-mail-Adressen, die Postfix verwendet. Im rechten Teil stehen (existierende) Benutzeraccounts oder externe E-mail-Adressen. Dorthin werden die Mails weitergeleitet. Um *virtual domains* zu aktivieren, muss in der Datei */etc/postfix/main.cf*

```
virtual_maps = hash:/etc/postfix/virtual
```

eingetragen werden.

Jetzt muss mit dem Befehl

```
postmap /etc/postfix/virtual
```

die Hashtabelle für Postfix generiert werden. Falls über eine Dialin-Verbindung die Mails versendet werden, kann es notwendig sein, die Relay-Domain des Providers in der Datei */etc/postfix/main.cf*

```
relay_domains = $mydestination, einwahl.provider.de, noch.einer.de
```

einzutragen. Wenn dies nicht geschieht, akzeptiert Postfix unter Umständen die Versendung der Mail an eine Adresse außerhalb der von Postfix verwalteten Hosts nicht.

### 5.3.5 Mail Relaying durch POP before SMTP:

---

Um unkontrollierte E-mail-Spam-Versendung über den Mailserver zu verhindern, gibt es die Möglichkeit, dass sich der Benutzer bei dem Zugriff authentifiziert.

Eine Methode dies zu erreichen bietet *POP before SMTP*. SMTP besitzt keine Authentifizierungsmethode. Durch Erweiterungen des SMTP-Protokolls bietet SMTP die Möglichkeit zur Authentifizierung. Dabei gibt es nur einen Nachteil, nämlich der verwendete Client muss diese auch unterstützen.

Mittels eines einfachen Tricks ist dies möglich: Die Authentifikation erfolgt über den Benutzernamen und das Passwort, wenn sich der Client (Benutzer) in den POP-Server einloggt und seine Mails abholt. Gleichzeitig wird die soeben authentifizierte IP in einer Tabelle eingetragen. Beim Versenden von E-mails sieht Postfix in dieser Tabelle nach um herauszufinden ob diese IP eingetragen ist. Erst wenn dies der

## 5.3 Postfix Konfiguration

Fall ist, sendet Postfix die Mails weiter. Nach einer gewissen Zeit löscht Postfix die Einträge wieder.

Das von Postfix verwendete Authorisationspaket *DRACD* (Dynamic Relay Authorization Control Daemon) wurde ursprünglich für Sendmail entwickelt. Es besteht aus einem Patch für den POP-Daemon *qpopper* von Qualcomm und einem Dämon. Dieser Patch bewirkt, dass sich *qpopper* nach erfolgreicher Authentifizierung des Benutzers, mit dem *dracd* verbindet. Dabei teilt es diesem die aktuelle Adresse des Rechners mit. Der *DRACD* schreibt diese dann in eine Datenbank, welche Postfix auswertet.

In der Datei */etc/postfix/main.cf* muss:

```
smtpd_recipient_restrictions = permit_mynetworks,  
check_client_access hash:/etc/postfix/client_access,  
btree:/etc/mail/dracd, check_relay_domains
```

eingetragen werden. Bei *btree:/etc/mail/dracd* wird der Pfad zur Datenbank angegeben. Dabei muss die *.db* Endung weggelassen werden.

### 5.3.6 Virens Scanner:

---

Es gibt auch ein Modul für Postfix um die eingehenden E-mails nach Viren zu durchsuchen.

Der Eintrag in der Datei */etc/postfix/main.cf* muss

```
mailbox_command = /local/bin/viruscheck
```

lauten. Dazu wird jedoch ein diese Funktion unterstützender Virens Scanner benötigt.

## 5.4 Installation von Diskquotas mit SuSe

(falls quotas für die Openwebmail Benutzer gewünscht werden)

### 5.4.1 Benötigte Pakete:

---

*quota* (SuSe 9.0 DVD/CD)

### 5.4.2 Instalation:

---

Zuerst muss ein neuer Benutzer für unsere Zwecke mit *yast* angelegt werden, zum Beispiel mit dem Namen *postfach*. Dort wird unter Details die Standartgruppe *mail* eingestellt, und alle anderen Gruppen abgewählt.

In unserem Fall wurde die Partition */dev/hda8* mit dem Dateisystem reiserfs am Mountpunkt */* in das Dateisystem eingebunden. Um Diskquotas für die Benutzer auf dieser Partition anzulegen, muss in der Datei */etc/fstab* der Eintrag für */dev/hda8* um die Option *usrquota* ergänzt werden.

*joe /etc/fstab*

```
/dev/hda8      /           defaults,usrquota    1 1
```

Mit dem Befehl *cd /* wird nun auf das Wurzelverzeichnis gewechselt.

Dort wird mit dem Befehl *touch /aquota.user* eine leere Datei erstellt. Nun werden die Rechte der Datei auf read und write für root gesetzt. Macht man das nicht, kann jeder die Datei verändern und sich selbst das Quota erhöhen.

*chmod 600 /aquota.user*

Als nächstes muss die Verwaltungsdatei eingerichtet werden, was das Programm *quotacheck*, das im *quota* Paket enthalten ist übernimmt.

*quotacheck -avMF vfsr0*

Jetzt ist es an der Zeit für einen Benutzer quotas anzulegen. Dies erfolgt über den Befehl *edquota*, der für die Eingabe der Werte den Editor *vi* verwendet.

## 5.4 Installation von Diskquotas mit SuSe

*edquota -u postfach*

Nun muss die Taste *i* gedrückt werden, um in den Editmodus zu wechseln.

Und so sollte der Eintrag lauten, wenn der Benutzer postfach eine Beschränkung von 2 MB als weiches und hartes Limit besitzen soll.

```
mailintern:/ # quota postfach
Disk quotas for user postfach (uid 501):
  Filesystem blocks  quota  limit  grace  files  quota  limit  grace
  /dev/hda8   71    2048  2048           28     0     0
```

Mit dem Drücken von *Esc* verlässt man den Editmodus. Das File wird durch Eingeben von *:wq* gespeichert und geschlossen.

Nach der Einrichtung sollten die Diskquotas auch aktiviert werden. Danach liefert *repquota* den derzeitigen Stand der Dinge. Der Befehl

*quotaon -F vfsr0 /*

schaltet das Quota ein, dabei steht *-F vfsr0* für das Dateisystem. Bei dem Mountpunkt */* bekommt man die Meldung *Device or resource busy*. Diese Meldung kann ignoriert werden. Sie besagt nur, dass er den Mountpunkt nicht unmounten und danach wieder neu mounten kann. Die Lösung ist einen Neustart nach dem Fertigkonfigurieren von Quota durchzuführen. Der Befehl

*repquota -av*

wird die Quotas anzeigen.

Das Quota ist mit den derzeitigen Einstellungen nach dem Neustart nicht mehr aktiv. Dafür gibt es im Verzeichnis */etc/init.d* einige init-Scripte die es für uns automatisch aktivieren.

Das Script *Quota* startet die Quotas und *boot.quota* sorgt dafür, dass nach einem unsauberen Beenden des Dateisystems die Quotas überprüft werden. Der Befehl *insserv* bewirkt, dass das angegebene Script nun auch in dem im Script eingestellten Runlevel startet.

*cd /etc/init.d*

## 5.4 Installation von Diskquotas mit SuSe

*insserv quota*  
*insserv boot.quota*

Mit dem Befehl `quota` kann man sich die vorhandenen Quotas für jeden Benutzer ansehen.

z.B. *quota postfach*

```
mailintern:/ # quota postfach
Disk quotas for user postfach (uid 501):
  Filesystem blocks  quota  limit  grace  files  quota  limit  grace
  /dev/hda8   71    2048  2048           28     0     0
```

oder *repquota -av* für alle Benutzer.

```
mailintern:/ # repquota -av
*** Report for user quotas on device /dev/hda8
Block grace time: 7days; Inode grace time: 7days

User                used      Block limits          File limits
                   soft    hard  grace    used  soft  hard  grace
-----
root                -- 2703109      0      0           121184  0  0
lp                  -- 67           0      0           23     0  0
mail                -- 1            0      0           2     0  0
news                -- 2            0      0           11     0  0
uucp                -- 1            0      0           4     0  0
games              -- 1            0      0           2     0  0
man                 -- 946          0      0          1990     0  0
at                  -- 1            0      0           5     0  0
wwwrun              -- 1            0      0           6     0  0
postfix             -- 8            0      0           82     0  0
mysql               -- 9            0      0           3     0  0
ntp                 -- 1            0      0           2     0  0
nobody              -- 1            0      0           2     0  0
markus              -- 253          0      0           76     0  0
postfach            -- 71    2048  2048           28     0  0
test123             -- 156    2048  2048           43     0  0
testxzy             -- 70    2048  2048           22     0  0
test543             -- 181    2048  2048           46     0  0
testhm              -- 144    2048  2048           42     0  0
#1000               -- 50         0      0           12     0  0
#100892             -- 110        0      0           21     0  0

Statistics:
Total blocks: 11
Data blocks: 1
Entries: 21
Used average: 21.000000
```

## 5.5 Openwebmail Installation und Konfiguration

Als erstes Openwebmail von [Openwebmail.de](http://Openwebmail.de) downloaden.  
Bei den Downloads gibt es für SuSe extra rpm Files. Diese können verwendet werden. Es muss aber darauf geachtet werden das die Versionsnummern nicht kleiner als die angegebenen sind.

### 5.5.1 Benötigte Pakete:

---

*perl\_CGI\_Application (SuSe DVD/CD) mindestens v. 2.74*  
*libnet (SuSe DVD/CD) mindestens v. 1.0901*  
*perl\_Text\_Iconf (SuSe DVD/CD) mindestens v. 1.2*  
*make (SuSe DVD/CD)*  
*uudeview (SuSe DVD/CD)*  
*gcc (SuSe DVD/CD)*  
*db40 (SuSe DVD/CD)*  
*db40-devel (SuSe DVD/CD)*  
*quota (SuSe DVD/CD) falls für die Benutzer ein Quota gewünscht wird.*  
*MIME-Base64 mindestens v 2.12 z.b. über links2linux.de suchen*

Als erstes wird das Paket *MIME-Base64* mit dem Befehl

```
tar -zxvBpf MIME-Base64 [TAB]
```

entpackt. Dann mit dem Befehl

```
cd /MIME-Base64
```

in den entpackten Ordner wechseln.

Das *MIME-Base64* Modul wird durch Eingabe von dem Befehl

```
perl Makefile.PL  
make  
make install
```

compiliert und installiert.

### 5.5.2 Datei und Verzeichnis Rechte:

---

Alle Dateien in dem Verzeichnis */srv/www/cgi-bin/openwebmail* müssen dem User *root* und der Gruppe *mail* angehören. Dagegen müssen die Verzeichnisse alle dem User *root* und der Gruppe *root* angehören.





## 5.5 Openwebmail Installation und Konfiguration

```
chown root.root /srv/www/cgi-bin/openwebmail
chown root.mail /srv/www/cgi-bin/openwebmail/*
```

Zuweisung der Dateirechte für *Suidperl*:

```
chmod 4555 /usr/bin/suidperl
```

Jede Datei im Verzeichnis */srv/www/cgi-bin/openwebmail* mit dem Namen *openwebmail\*.pl* muss *chmod 4555* gesetzt werden und den Owner *root.mail* haben.

```
chmod 4555 /srv/www/cgi-bin/openwebmail/openwebmail*.pl
chown root.mail /srv/www/cgi-bin/openwebmail/openwebmail*.pl
```

### 5.5.3 Konfiguration:

---

Überprüfen und Ändern der Einstellungen von:

```
/srv/www/cgi-bin/openwebmail/etc/openwebmail.conf
```

Auslesen des Domainnamens aus Postfix:

```
domainnames      auto
```

Verwendetes Authentifikationsmodul:

```
auth_module      auth_unix.pl
```

Haupt-Mail-Verwaltungsordner:

```
mailspooldir     /var/spool/mail
```

Datenbankformat:

```
dbm_ext          .db
```

```
dbmopen_ext     none
```

```
dbmopen_haslock no
```

Pfad zum Openwebmail-CGI-Ordner:

```
ow_cgidir       /srv/www/cgi-bin/openwebmail
```

Url zum Openwebmail-CGI-Ordner:

```
ow_cgiurl       /cgi-bin/openwebmail
```

## 5.5 Openwebmail Installation und Konfiguration

Url zu den html-Files:

*ow\_htmldir*                      */data/openwebmail*

Pfad zum Openwebmail logfile:

*logfile*                              */var/log/openwebmail.log*

Pfad zur verwendeten Rechtschreibprüfung:

*spellcheck*                          */usr/bin/aspell*

Standard Openwebmail-Sprache:

*default\_dictionary*              *german*

E-Mail Attachment Limit in kb:

*attlimit*                              *1024*

Webspace Upload Limit in kb:

*webdisk\_uploadlimit*      *2048*

Standardsignatur unter E-Mails ändern:

*<default\_signature>*

*--*

*Open WebMail Project (<http://openwebmail.org>)*

*</default\_signature>*

Nun müssen noch die Einstellungen in der Datei */srv/www/cgi-bin/openwebmail/auth/auth\_unix.pl* angepasst werden.

```
my $unix_passwdfile_plaintext="/etc/passwd";
my $unix_passwdfile_encrypted="/etc/shadow";
my $unix_passwdmkdb="";
my $check_shell=0;
```

Als nächstes muss die Datei */etc/openwebmail\_path.conf* angelegt werden:

```
touch /etc/openwebmail_path.conf
```

In der Datei muss der Pfad zu Openwebmail angegeben werden:

```
/srv/www/cgi-bin/openwebmail/
```

## 5.5 Openwebmail Installation und Konfiguration

Openwebmail wird durch Eingabe von

```
/srv/www/cgi-bin/openwebmail/openwebmail-tool.pl --init
```

initialisiert.

Nun muss die Datei */srv/www/cgi-bin/openwebmail.pl* editiert werden:  
In *Zeile 1* muss das Argument *-T* entfernt werden.

Im Browser kann man durch Eingabe von

```
http://mailserver/cgi-bin/openwebmail/openwebmail.pl
```

testen, ob Openwebmail läuft.

## 5.6 Installation des Adduserscriptes

### 5.6.1 Benötigte Pakete:

---

*ParsePlainConfig* (über [links2linux.de](http://links2linux.de) suchen)  
*perl\_Template-Toolkit* (SuSe 9.0 DVD/CD)

Das Paket *ParsePlainConfig* muss durch Eingabe von

```
tar -zxvBpf ParsePlainConfig [TAB]
```

entpackt werden. Nun muss mit dem Befehl

```
cd /ParsePlainConfig [TAB]
```

in das Verzeichnis gewechselt werden, dort muss

```
perl Makefile.PL  
make  
make install
```

ausgeführt werden.

Das Script kann von [www.openwebmail.de](http://www.openwebmail.de) geladen werden, indem man *Strg + F* drückt und nach *adduser* sucht.

Wenn das File geladen wurde, wird es nun mit

```
tar -zxvBpf adduser.pl-0.2.tar.gz
```

entpackt und mit

```
cd /adduser.pl-0.2
```

in das Verzeichnis gewechselt.

## 5.6 Installation des Adduserscriptes

### 5.6.2 Installation:

---

Die Installation wird mit dem Befehl

```
perl Makefile.PL
```

gestartet. Alle Standardangaben sollen übernommen werden.

*Die CGI Dir ist /srv/www/cgi-bin/*

*Die UID ist 30 (für Apache2 Webserver)*

Das Verzeichnis */srv/www/etc* wird als nächstes mit dem Befehl

```
md /srv/www/etc
```

erstellt. In dieses Verzeichnis muss die *adduser.conf* aus dem *adduser.pl-0.2/sample/* Ordner kopiert werden. Desweiteren muss die *signup.html* aus dem selben Ordner nach */srv/www/htdocs/* kopiert werden.

Nun wird mit joe die Datei *adduser.conf* editiert, um einige Optionen anzupassen:

Scriptlog Directory:

```
LOG: /etc/log/adduser.log
```

Pfad zur *signup.html*:

```
REFERER: http://mailintern.bbs-nw.de/signup.html
```

Accounterstellung anschalten:

```
ALLOW: 1
```

Standard Gruppe festlegen:

```
GROUP: mail
```

Home Verzeichnis der neu erstellten Benutzer:

```
HOME_ROOT: /home/webmail/users
```

Wenn Quota erwünscht is, muss unser extra dafür erstellter Benutzer postfach angegeben werden. Dadurch werden für alle neu erstellten Accounts dessen Quotas verwendet

## 5.6 Installation des Adduserscriptes

*QUOTAP: default-user oder postfach*

E-mail domain angeben:

*DOMAIN: mailintern.bbs-nw.de*

Nun müssen die Dateien *error.ttk* und *success.ttk* aus dem *adduser.pl-0.2/sample* Ordner in das mit

*md /srv/www/cgi-bin/lib/*

erstellte Verzeichnis kopiert werden.

Als nächstes muss mit *joe* die Datei *adduser.pl* editiert werden.

*joe /srv/www/cgi-bin/adduser.pl*

In *Zeile 1* müssen die Argumente *-wT* entfernt werden.

```
use constant ERRHTML      => './lib/error.ttk';
use constant SUCHTML     => './lib/success.ttk';
```

*Zeile 69* muss mit einem darvorgestellten *#* auskommentiert werden, da das Script die *UID* vom *Apache2* für falsch hält. Nach Rückfragen mit dem Autor ist die verwendete *UID* aber richtig. Durch das auskommentieren wird die Abfrage umgangen.

Die Pfade von Sendmail müssen für Postfix umgeschrieben werden:

```
# Modify the virtusertable
open(VIRT, "+< /etc/postfix/virtual") ||

# Remake the databases
system("cd /etc/postfix && @[postmap] all 1>/dev/null 2>&1");
```

Nun ist auch diese Installation abgeschlossen.

## 5.7 Boot-Runlevel ändern

Wenn auf dem Rechner nichts anderes als Serverdienste, also z.B. der Openwebmaildienst laufen soll, kann man den Standard Runlevel, in das das System booten soll, anpassen. Dies erfolgt mit einem Texteditor in der Datei */etc/inittab*

```
# The default runlevel is defined here  
id:5:initdefault:
```

Dieser Eintrag kann auf

```
# The default runlevel is defined here  
id:3:initdefault:
```

geändert werden. Hier noch eine Übersicht über die vorhandenen Runlevel

```
runlevel 0 - System beenden (Nicht für diesen Befehl verwenden!)  
runlevel 1 - Einzelbenutzermodus  
runlevel 2 - Lokaler Multibenutzermodus ohne Remote-Netzwerk  
runlevel 3 - Voller Multibenutzermodus mit Netzwerk  
runlevel 4 - Nicht verwendet  
runlevel 5 - Voller Multibenutzermodus mit Netzwerk und Grafik  
runlevel 6 - System rebooten (Nicht für diesen Befehl verwenden!)
```

In dieser Übersicht wird deutlich, dass der Server am besten im runlevel 3 läuft, wenn keine grafische Oberfläche (runlevel 5) verwendet werden muss.

## 5.8 Häufig auftretende Fehlermeldungen

1. Bei der Fehlermeldung *Premature end script headers: xxxx.pl* muss in der Shell die Dateirechte von *Suidperl* neu gesetzt werden.

```
chmod 4555 /usr/bin/suidperl
```

Diese verändern sich wenn neue Perl-Pakete installiert wurden.

2. Bei der Fehlermeldung *Invalid Args in line 1* müssen die Argumente in *Zeile 1* mittels eines Editors gelöscht werden.



# 6 Anhänge

-

# Projektmanagement

## 6 Anhänge - Projektmanagement

### 6.1 Gruppenregeln:

---

Alle halten sich an die Zeitangaben und Termine.

Bei Meinungsverschiedenheiten wird ein gemeinsamer Nenner gefunden.

Alle Gruppenmitglieder müssen sich an die verschiedenen Regeln halten

Alle Gruppenmitglieder sind gleichberechtigt

### 6.2 Persönliche Ziesetzung:

---

Unser Ziel ist es ...

... Erfahrungen durch die Projektarbeit zu erhalten.

... präsentieren zu können.

... in einer Gruppe arbeiten zu lernen.

... die Projektmappe termingerecht abzugeben.

... mindestens die Note 2 auf die Projektarbeit zu bekommen.

... unser Wissen über die verwendete Software zu vertiefen.

... sich in Zukunft besser an Zeitangaben zu halten.

... sich in Zukunft besser an Abgabetermine zu halten.

## 6.3 Erfahrungsberichte

### 6.3.1 Markus Bode:

---

Als wir uns das Projekt herausgesucht hatten, wusste ich noch nicht genau was auf mich zukommt. Nachdem ich mir durch Webrecherche einen groben Überblick verschafft hatte, konnte ich schon etwas mehr mit dem ganzen Thema anfangen. Überhaupt kann ich sagen, dass über die Dauer des Projekts Google meine Hauptquelle für Informationen und Hilfe war. Um auch für das Projekt daheim arbeiten zu können, habe ich die gleiche Softwareumgebung wie auf dem verwendeten Rechner in der Schule installiert.

Viele Probleme bereitete es mir, eine passende Installationsanleitung für Openwebmail im Internet zu finden. Diese gab es nicht, was uns einige Probleme brachte. Ich stellte erst nach einigen Versuchen fest, dass die für das Programm Sendmail als Unterbau geschrieben wurde. Die Konfiguration von Postfix verlief um einiges problemloser.

Das Adduser-Script stellte auch noch einige Herausforderungen bei der Installation an uns, da es Probleme mit der richtigen UID für den Apache2 hatte. Dazu hatten wir zum Testen noch viel zu viele Pakete von Programmen, die nicht benötigt wurden, installiert. Erst bei der dritten Installation des Systems lief von Anfang an alles so, wie es sollte.

Es gab auch Probleme mit Perl. Nach jeder Änderung eines Perl-Paketes mussten die Ordnerrechte für `suidperl` neu gesetzt werden. Bis wir den Fehler fanden verloren wir viel Zeit. Es ist frustrierend wenn man mehrere Stunden an einem Problem arbeitet, ohne sichtbar weiterzukommen.

Was mich am Schluss doch überraschte, war, dass für wir für das Schreiben der Dokumentation am meisten Zeit brauchten. Dabei hatten wir uns mit der veranschlagten Zeit etwas verschätzt. Ansonsten, nachdem die Schwierigkeiten im Projekt einmal überwunden waren, hat es mir viel Spaß gemacht. Mit meinem Gruppenpartner habe ich im Team gut gearbeitet.

### 6.3.2 Mario Valceschini:

---

Als wir uns für den Webmail-Server als Projektarbeit entschieden haben, wusste ich nicht so recht ob wir uns mit dieser Aufgabe etwas übernommen haben. Ich habe mich noch nicht so oft mit dem Betriebssystem Linux beschäftigt, geschweige denn, mit dem Aufbau eines Webmail-Servers. Bei der Installation des Servers stießen wir immer wieder auf kleinere Probleme, da es schwer war eine gute Installationsanleitung für Openwebmail im Internet zu finden. Als wir nach einigen Neuinstallationen ein stabiles System hatten, bekamen wir grössere Probleme mit dem Perl Programm. Nachdem dieses Problem beseitigt war, gab es noch kleinere Schwierigkeiten mit dem Aufbau der index.html, welche zur u.a. Anmeldung an dem Openwebmail-Server dient.

Wir haben bemerkt, dass das Erstellen der Dokumentation doch zeitintensiver war, als wir es eigentlich vorgesehen hatten. Die Suche nach Fehlern und nach Lösungen über das Netz haben oft mehrere Stunden in Anspruch genommen. Alles in allem hat mir das Arbeiten an dem Projekt viel Spass gemacht, auch das Arbeiten mit Markus verlief ohne Probleme.

## 6.4 Protokolle

Datum: 17.11.2003  
Uhrzeit: 15.00 - 17.00  
Protokollführer: Bode  
Anwesende: Bode, Valceschini  
Abwesende: keine

Festlegung der Ziele

Datum: 30.11.2003  
Uhrzeit: 14.00 - 17.00  
Protokollführer: Bode  
Anwesende: Bode, Valceschini  
Abwesende: keine

SuSe 9.0 Installation

Datum: 09.12.2003  
Uhrzeit: 14.30 - 16.00  
Protokollführer: Bode  
Anwesende: Bode, Valceschini  
Abwesende: keine

SuSe 9.0 Installation

Datum: 09.01.2004  
Uhrzeit: 13.00 - 16.00  
Protokollführer: Bode  
Anwesende: Bode, Valceschini  
Abwesende: keine

Einarbeitung in SuSe 9.0

## 6.4 Protokolle

Datum: 15.01.2004  
Uhrzeit: 15.00 - 17.00  
Protokollführer: Bode  
Anwesende: Bode, Valceschini  
Abwesende: keine

Auswahl der Benutzerweboberfläche für Postfix.

Datum: 19.01.2004  
Uhrzeit: 8.00 - 17.00  
Protokollführer: Bode  
Anwesende: Bode, Valceschini  
Abwesende: keine

Konfiguration von Postfix.  
Installation von Openwebmail.

Datum: 20.01.2004  
Uhrzeit: 8.00 - 17.00  
Protokollführer: Bode  
Anwesende: Bode, Valceschini  
Abwesende: keine

Installation und Konfiguration von Openwebmail.

Datum: 21.01.2004  
Uhrzeit: 8.00 - 17.00  
Protokollführer: Bode  
Anwesende: Bode, Valceschini  
Abwesende: keine

Installation des Adduserscript für Openwebmail.

## 6.4 Protokolle

Datum: 22.01.2004  
Uhrzeit: 8.00 - 17.00  
Protokollführer: Bode  
Anwesende: Bode, Valceschini  
Abwesende: keine

Konfiguration von Postfix, Openwebmail und dem Adduserscript.

Datum: 23.01.2004  
Uhrzeit: 8.00 - 16.00  
Protokollführer: Bode  
Anwesende: Bode, Valceschini  
Abwesende: keine

Installation und Einrichtung des Quotas für die Webmailbenutzer.

Datum: 22.01.2004  
Uhrzeit: 8.00 - 13.00  
Protokollführer: Bode  
Anwesende: Bode, Valceschini  
Abwesende: keine

Erstellung der Website für das Projekt.

Datum: 22.01.2004  
Uhrzeit: 8.00 - 13.00  
Protokollführer: Bode  
Anwesende: Bode, Valceschini  
Abwesende: keine

Erstellung der Website für das Projekt.

## 6.4 Protokolle

Datum: 27.01.2004  
Uhrzeit: 8.00 - 13.00  
Protokollführer: Bode  
Anwesende: Bode, Valceschini  
Abwesende: keine

Erstellung der Dokumentation.

Datum: 28.01.2004  
Uhrzeit: 8.00 - 13.00  
Protokollführer: Bode  
Anwesende: Bode, Valceschini  
Abwesende: keine

Erstellung der Dokumentation.

Datum: 29.01.2004  
Uhrzeit: 8.00 - 13.00  
Protokollführer: Bode  
Anwesende: Bode, Valceschini  
Abwesende: keine

Erstellung der Dokumentation.

Datum: 01.02.2004  
Uhrzeit: 15.00 - 17.00  
Protokollführer: Bode  
Anwesende: Bode  
Abwesende: Valceschini

Erstellung der Dokumentation.



## 6.4 Protokolle

Datum: 02.02.2004  
Uhrzeit: 11.30 - 13.00  
Protokollführer: Bode  
Anwesende: Bode, Valceschini  
Abwesende: keine

Erstellung des Designs der Dokumentation.

Datum: 03.02.2004  
Uhrzeit: 11.30 - 13.00  
Protokollführer: Bode  
Anwesende: Bode, Valceschini  
Abwesende: keine

Erstellung des Designs der Dokumentation.

Datum: 04.02.2004  
Uhrzeit: 15.30 - 18.00  
Protokollführer: Bode  
Anwesende: Bode, Valceschini  
Abwesende: keine

Erstellung der Dokumentation.

Datum: 05.02.2004  
Uhrzeit: 08.00 - 10.30  
Protokollführer: Bode  
Anwesende: Bode, Valceschini  
Abwesende: keine

Erstellung der Dokumentation.

## 6.4 Protokolle

Datum: 06.02.2004  
Uhrzeit: 15.00 - 17.00  
Protokollführer: Bode  
Anwesende: Bode, Valceschini  
Abwesende: keine

Erstellung der Dokumentation.

Datum: 08.02.2004  
Uhrzeit: 15.00 - 18.00  
Protokollführer: Bode  
Anwesende: Bode  
Abwesende: Valceschini

Erstellung der Dokumentation.

Datum: 09.02.2004  
Uhrzeit: 16.00 - 18.00  
Protokollführer: Bode  
Anwesende: Bode, Valceschini  
Abwesende: keine

Erstellung der Dokumentation.

Datum: 10.02.2004  
Uhrzeit: 15.00 - 22.00  
Protokollführer: Bode  
Anwesende: Bode, Valceschini  
Abwesende: keine

Erstellung der Dokumentation.

## 6.5 Quellen

GNU General Public License:

<http://www.suse.de/de/private/support/licenses/gpl.html>

Postfix Installationsanleitung:

<http://www.linux-magazin.de/Artikel/ausgabe/2000/06/Postfix/postfix.html>

<http://www.linux-tin.org/tin.german/setupguide/server/suse/html/mailserver.htm>

Informationen über Apache2:

[http://news.netcraft.com/archives/web\\_server\\_survey.html](http://news.netcraft.com/archives/web_server_survey.html)

<http://www.linux-magazin.de/Artikel/ausgabe/2002/10/apache2/apache2.html>

<http://www.fh-wedel.de/~si/seminare/ws01/Ausarbeitung/b.apache/apache2.htm>

Homepage von Openwebmail:

<http://www.openwebmail.de>

Openwebmail Installationsanweisung:

<http://www.torsten-brumm.de/data/html/owm-install-de.html>

POP-before-SMTP Mini-HowTo

<http://www.mdlug.de/bibo/knowhow/netz/popbsmtp.shtml>

[http://www.daciuk.de/sdb/de/html/hacht\\_pop\\_before\\_smtp\\_using\\_drac.html](http://www.daciuk.de/sdb/de/html/hacht_pop_before_smtp_using_drac.html)

## 6.6 Abschlusserklärung

Wir erklären, dass die Projektarbeit selbstständig und ohne fremde Hilfe verfasst wurde und keine anderen als die angegebenen Mittel verwendet wurden.

Wir versichern, dass alle wörtlichen und sinngemäßen Übernahmen aus anderen Werken als solche gekennzeichnet werden.

---

Markus Bode

---

Mario Valceschini

## 6.7 Notizen

## 6.7 Notizen

## 6.7 Notizen