Jan-Hendrik Terstegge / Julian Frede
Ernst-Barlach-Gymnasium Unna
Jahrgangsstufe 12.2
2000/2001

# Konfiguration und Sicherheitsaspekte des c't ODS (Offenes Deutsches Schulnetz) Kommunikationsservers 3.0 am Beispiel des Pestalozzi-Gymnasiums Unna

| | | |
|---|---|---|
| Fach: | Informatik | Note: |
| Fachlehrer: | Müller | |
| Abgabedatum: | 02.03.2001 | _____ |
| | | (Datum) (Unterschrift) |

# INHALTSVERZEICHNIS

1. Inhaltsverzeichnis ..... S. 2
2. Vorwort ..... S. 3
3. Sicherheitsaspekte ..... S. 4
4. Absicherung des root-Zugangs ..... S. 10
5. Automatisierung der Einwahl ..... S. 13
6. Samba ..... S. 14
7. E-Mail Austausch ..... S. 18
8. Anpassungen des Web-Interfaces ..... S. 20
9. Quellenverzeichnis ..... S. 21

# VORWORT

Wir, d.h. Jan-Hendrik Terstegge und Julian Frede, haben das Thema „Konfiguration und Sicherheitsaspekte des c't ODS (Offenes Deutsches Schulnetz) Kommunikationsservers 3.0 am Beispiel des Pestalozzi-Gymnasiums Unna" gewählt, da der c't ODS Kommunikationsserver 3.0 auf einem Linux-System aufbaut, und Linux zum einen zu unseren persönlichen Interessen gehört, andererseits aber auch das Betriebssystem der Zukunft sein wird. Da wir uns selber in unserer Freizeit mit dem Betriebssystem Linux beschäftigen, erschien uns dieses Thema angemessen um in Zusammenarbeit den großen Bereich „Konfiguration und Sicherheitsaspekte" zu bearbeiten.

Wir haben natürlich viele Aspekte des c't ODS Kommunikationsservers nicht berücksichtigt (wie z.B. die Einrichtung und Anwendung von Diskquotas), aber alle Aspekte zu berücksichtigen, würde den Rahmen dieser Facharbeit bei weitem sprengen.

Ebenso sind nicht alle online-Quellen beigelegt, da beispielsweise alleine Quelle [05][1] einen Umfang von 100 Seiten hat. Allerdings sind alle Quellen auf der beigelegten CD-ROM enthalten. Die Quellen können mit dem Programm Acrobat Reader eingesehen werden, das unter http://www.acrobat.com erhältlich ist.

Unsere Absicht beim Verfassen der Facharbeit, war, die wichtigsten Aspekte des c't ODS Kommunikationsservers 3.0 dem Leser näherzubringen und ihm einige Antworten auf wichtige Fragen zu geben.

Jan-Hendrik Terstegge Julian Frede

[1] [05]: Auer, Karl & Allison, James. smb.conf Manpage

## SICHERHEITSASPEKTE

### (JAN-HENDRIK TERSTEGGE)

Der c't ODS Kommunikationsserver 3.0 ist von Natur aus relativ schlecht gegen Eindringlinge von außen gesichert. Um die Sicherheit aller Daten auf dem Server zu gewährleisten, müssen verschiedene Dinge durchgeführt werden:

Als erstes sollte man einen Firewall zwischen sich und das Internet bringen, der unbefugte Eindringlinge von außen abwehrt. Dies ist bei einem Anschluß an die meisten Internet-Provider aber nicht notwendig, da fast alle ISP's (Internet Service Provider) bereits einen Firewall zwischen Internet und Benutzer geschaltet haben.

Trotzdem sind unter Linux bereits standardmäßig einige Sicherheitsmaßnahmen getroffen, um Sicherheit zu gewährleisten falls es doch gelingen sollte, den Firewall des ISP's zu überlisten.

Die beiden Dateien `/etc/hosts.allow` und `/etc/hosts.deny` sorgen dafür, das nur lokale Benutzer die eine IP-Adresse im Bereich 192.168.x.x bzw. 255.255.128.x haben, wobei 'x' eine beliebige Zahl zwischen 0 und 255 sein kann.

Dafür steht in `/etc/hosts.allow` *`ALL:LOCAL, 192.168.0.0 / 255.255.128.0`* und in `/etc/hosts.deny` *`ALL:ALL`*.

Das heißt, das allen (*`ALL:ALL`* aus `/etc/hosts.deny`) Benutzern außer den lokalen (*`ALL:LOCAL`* aus `/etc/hosts.allow`) Benutzern der Zugriff verwehrt wird.

Da es trotzdem noch immer wieder Sicherheitslücken in einige Programmen gibt, und die meistens über offene Kommunikationsports ausgenutzt werden, um in einen Rechner einzudringen, müssen alle Ports außer den besonders wichtigen geschlossen werden.

Um festzustellen, welche Ports vor der Absicherung offen sind, kann z.B. das Programm nmap das unter `http://www.insecure.org/nmap/` erhältlich ist verwendet werden.

Bei einem Scan der offenen Ports mit nmap vor der Absicherung sind folgende Ports offen:

| PORT | DIENST |
|---|---|
| 21 | ftp |
| 22 | ssh |
| 23 | telnet |
| 25 | smtp |
| 37 | time |
| 53 | domain |
| 79 | finger |
| 80 | http |
| 110 | pop-3 |
| 113 | auth |
| 119 | nntp |
| 137 | netbios-sn |
| 138 | netbios-dgm |
| 139 | netbios-ssh |
| 143 | imap2 |
| 443 | https |
| 515 | printer |
| 517 | talk |
| 518 | ntalk |
| 540 | uucp |
| 901 | samba-swat |
| 3130 | squid-ipc |
| 8080 | http-proxy |

Nachdem man festgestellt hat, welche Ports offen sind, öffnet man die Datei /etc/inetd.conf und kommentiert alle unwichtigen Ports aus.
Im vorliegenden c't ODS-Kommunikationsserver 3.0 müssen folgende Zeilen auskommentiert werden:

| | | | | | | |
|---|---|---|---|---|---|---|
| ftp | stream | tcp | nowait | root | /usr /sbin /tcpd | /usr /sbin /in.ftpd -a |
| telnet | stream | tcp | nowait | root | /usr /sbin /tcpd | /usr /sbin /in.telnetd |
| talk | dgram | udp | wait | root | /usr /sbin /tcpd | /usr /sbin /in.ntalkd |
| ntalk | dgram | udp | wait | root | /usr /sbin /tcpd | /usr /sbin /in.ntalkd |
| finger | stream | tcp | nowait | daemon | /usr /sbin /tcpd | /usr /sbin /in.fingerd |
| imap2 | stream | tcp | nowait | root | /usr /sbin /tcpd | /usr /sbin /imapd |

Nachdem man diese Zeilen auskommentiert hat, bleiben noch folgende Ports offen, wie man bei einem erneuten nmap-Durchlauf sieht:

| PORT | DIENST | ERKLÄRUNG |
|---|---|---|
| 22 | ssh | Wird benötigt, um das System von außen zu administrieren. |
| 25 | smtp | Wird benötigt, um E-Mails auszuliefern. |
| 37 | time | Wird benötigt, um die Zeit richtig einzustellen. Das ist wichtig, damit Cron-Jobs zeitlich genau |

| | | |
|---|---|---|
| | | ablaufen. |
| 53 | domain | Wird benötigt, um Domainservices bereitzustellen. |
| 80 | http | Wird benötigt, um das Webinterface bereitzustellen. |
| 110 | pop-3 | Wird benötigt, um E-Mails abzufragen. |
| 113 | auth | Wird zur Benutzerzertifizierung bei der Anmeldung im System benötigt |
| 119 | nntp | Wird benötigt, um News bereitzustellen. |
| 137, 138, 139 | netbios-ns, netbios-dgm, netbios-ssh | Diese Dienste werden von Samba benötigt. |
| 443 | https | Wird benötigt, um eine sichere Webverbindung zum Server herzustellen. |
| 515 | printer | Wird benötigt, um über das Netzwerk einen Drucker von mehreren Rechnern aus anzusprechen. |
| 540 | uucp | Wird ebenfalls (smtp auch) benötigt, um E-Mails abzufragen. |
| 901 | samba-swat | Wird benötigt, um das Samba Web Administration Tool zu benutzen. |
| 3130 | squid-ipc | Wird vom Cache-Dienst Squid benötigt. |
| 8080 | http-proxy | Wird vom Dienst http benötigt. |

Danach ist das System immer noch relativ schlecht abgesichert. Zwar hat der von außen kommende Eindringling jetzt keine ungenutzten Angriffsstellen mehr, trotzdem bleiben noch genügend Ports übrig, anhand derer er sich des Systems bemächtigen könnte.

Nun sollte man noch das System, bzw. die Sicherheits-wichtigen Softwarepakete einzeln aus dem Netz kopieren, kompilieren und installieren. Dies ist besonders wichtig, da laufend neue Sicherheitslücken in Linux-Paketen entdeckt werden, und jeder erfahrene Angreifer (und

auch sogenannte „Script-Kiddies“) das nötige Wissen haben, diese Sicherheitslücken auszunutzen und in das System einzudringen.
Beispielsweise sollte man die aktuellste Version der OpenSSH installieren. Das geht folgendermaßen:

1. Die Quelltexte aus dem Internet saugen (über www.openssh.com), beispielsweise von `ftp://ftp.de.openbsd.org/pub/Unix/OpenBSD/OpenSSH/openssh-2.5.1p1.tgz`
   Hierbei muß darauf geachtet werden, daß man die korrekte Version der OpenSSH (nämlich die auf andere Betriebssysteme portable downloadet)
2. Danach die mit tar und gzip gepackte Datei entpacken:
   ‚`tar xvfz openssh-2.5.1p1.tgz`‘
   Die Parameter xvfz bedeuten folgendes:
   - x = eXtract (entpacken)
   - v = Verbose (genaue Informationen dabei liefern)
   - f = die hinten erwähnte Datei (File) (in diesem Fall ‚openssh-2.5.1.tgz‘) soll verwendet werden
   - z = das ganze soll über das gZip Programm laufen, da es auch hiermit noch gepackt worden ist.
3. Nun mit ‚`cd ./openssh-2.5.1p1`‘ in das Verzeichnis ‚openssh-2.5.1p1‘ wechseln, in das die Quelltexte entpackt worden sind.
4. Nun muß mit ‚`./configure`‘ der Konfigurationsprozess gestartet werden, in dem sich ein selbstständiges Programm was im Paket enthalten ist, alle Einstellungen selber zuweist, die für die Kompilierung notwendig sind.
5. Nun startet man mit ‚`make`‘ den eigentlichen Kompilationsprozess, der, wenn erfolgreich verlaufen
6. Als letztes muß man ‚`make install`‘ auf der Kommandozeile eingeben, um das fertig kompilierte Programm in die verschiedenen, erforderlichen Verzeichnisse zu installieren.

Diese hier am Beispiel ‚OpenSSH' gezeigte Vorgehensweise zur Neuinstallation eines Programmes funktioniert für gewöhnlich bei fast allen Programmen auf die gleiche Art und Weise (über ‚`./configure`', ‚`./make`', ‚`./make install`'), sollte dies nicht der Fall sein, wird empfohlen, nach einer Datei namens ‚`README`' oder ‚`INSTALL`' zu schauen, da diese meistens die Vorgehensweise erläutern'.

Quellen: [09], [10], [11]

## ABSICHERUNG DES ROOT-ZUGANGES

**(JULIAN FREDE)**

- Sicherheitsspezifisch:
  Wie sichert man den root-Zugang des arktur ab?

Zuerst einmal sei gesagt das es absolute Sicherheit auf einem System nicht gibt. Denn wenn man einen Computer absolut sicher machen will gibt es nur die Möglichkeit ihn seiner sämtlichen Fähigkeiten zu berauben.
Außerdem trifft auch der unter Sicherheitsfachmännern gebräuchlichen Satz „Was ein Mensch verschließt kann ein Mensch auch öffnen!" zu.
Natürlich kann man als Administrator für ein höchstmögliches Maß an Sicherheit auf seinem Server sorgen. Hierzu gibt es folgende Möglichkeiten:

a) Man vergibt nur hoch sichere Passwörter:

Man kann die User anweisen sichere Passwörter zu Benutzen. Sie sollten zum Beispiel nicht ihren Vornamen oder ihr Geburtsdatum als Passwort verwenden.
Ebenso sollten sie von jeglichen von Außenstehenden leicht zu erschließenden Passwörten abstand nehmen.
Ein gutes Beispiel für ein unsicheres Passwort ist das zur Zeit auf dem uns zur verfügunstehenden ODS-Server vom sysadm (12345) ein sicheres Passwort wäre dagegen „tf7gdfzu".

b) Man schränkt den Netzwerkzugriff des Servers ein:

Hierzu sollte man sämtliche ungenutzte Ports des Servers in der Datei `/etc/inetd.conf` aus:

ftp: der Port für den ftp-Daemon, den Zugang gemäß dem File Transfer Protocol

telnet: der Port des Telnet Daemons, der das Login eines Users über das Netzwerk erlaubt

talk: Der Port des Talk Daemons, der die Kommunikation von Usern über das Netzwerk erlaubt.

ntalk: Der Port des Ntalk Daemons, der die Kommunikation von Usern über das Netzwerk erlaubt.

finger: Der Port des Finger Daemons, der das Abfragen von Userprofilen über das Netzwerk erlaubt.

imap2 : Der Port des Imap2 Daemons, der das Austauschens von Mails gemäß des Imap2 Protokolls erlaubt.

Nach dieser Prozedur sind noch folgende Ports offen:

ssh: Wird benötigt, um das System von außen zu administrieren.

smtp: Wird benötigt, um E-Mails auszuliefern.

time: Wird benötigt, um die Zeit richtig einzustellen.

domain: Wird benötigt, um Domainservices bereitzustellen.

http: Wird benötigt, um das Webinterface bereitzustellen.

pop-3: Wird benötigt, um E-Mails abzufragen.

auth: Wird zur Benutzerzertifizierung bei der Anmeldung im System benötigt.

nntp: Wird benötigt, um News bereitzustellen.

netbios-ns, netbios-dgm, netbios-ssh: Diese Dienste werden von Samba benötigt.

https: Wird benötigt, um eine sichere Webverbindung zum Server herzustellen.

printer: Wird benötigt, um über das Netzwerk einen Drucker von mehreren Rechnern aus anzusprechen.

uucp: Wird ebenfalls (smtp auch) benötigt, um E-Mails abzufragen.

samba-swat: Wird benötigt, um das Samba Web Administration Tool zu benutzen.
squid-ipc: Wird vom Cache-Dienst Squid benötigt.
http-proxy: Wird vom Dienst http benötigt.

c) Die Administration als User root.

Für die Administration über das Netwerk als User root empfiehlt sich meiner Meinung nach nur der Lokale oder der Netzwerkzugang über die SecureShell sicher.
Da dieses Programm sämtliche im Netzwerk übermittelte Daten mit 40Bit verschlüsselt. Diese doch relativ geringe doch für Schulzwecke ausreichende Verschlüsselung beruht darauf, dass die USA Exportbeschränkungen bei Sicherheitsrelevanten Programmen keine Ausfuhr von Verschlüsselungen über 40Bit zulassen.
Meiner Meinung nach ist mit Hilfe dieser Einstellungen für ein für den Arktur höchstmögliches Maß an Sicherheit gesorgt.
Natürlich kann ich nicht für absolute Sicherheit garantieren. Um auch weiterhin für höhere Sicherheit zu Sorgen sollte man regelmäßige Updates durchführen. Und die einschlägigen Internetseiten lesen (wie z.B. `http://www.rootshell.org`).

Anmerkung des Autors: Die Überschneidungen mit dem Punkt Sicherheitsaspekte dieser Facharbeit zweckmäßig und gewollt.
Quellen: [10], [11]

## AUTOMATISIERUNG DER EINWAHL

### (JULIAN FREDE)

- Besteht die Möglichkeit, dass in Abhängigkeit von den User-Rechten die Internet-Verbindung automatisch (!) aufgebaut wird, ohne dass das Admin-Interface benutzt werden muss? Wie sieht es mit dem Aufbau der Internetverbindung aus?

Es ist spezifisch nicht möglich dieses zu realisieren. Der User (Benutzer) meldet sich bei der Netzwerkanmeldung nicht als User (Benutzer) auf dem 'Arktur' c't ODS-Kommunikationsserver an, sondern teilt nur, via SMB Protokoll, dem auf 'Arktur' laufendem Daemon (Systemhintergrundprozeß) 'Samba' mit, dass er jetzt im Netzwerk verfügbar ist und seine Verzeichnisse gerne eingebunden haben will. Das bedeutet das der User (Benutzer) zu keiner Zeit auf 'Arktur' angemeldet ist, und somit keine Rechte hat ein Einwahlscript auszuführen.
Generell besteht doch eine Möglichkeit dieses zu realisieren, indem der Daemon 'Samba' im Quelltext angepaßt wird. Man muß ihn um die Funktion erweitern, dass er bei jeder Anforderung von Netzwerkverzeichnissen sich den User (Benutzer) extrahiert, kontrolliert ob dieser Mitglied einer Gruppe mit Internetanwahl-Erlaubniss ist, und die Verbindung dann gegebenenfalls aufbaut.
Der Verbindungsaufbau wird intern über das Skript `/etc/ppp/inet-on [VerbindungsName]` gestartet. Der Verbindungsabbau wird über `/etc/ppp/inet-off [VerbindungsName]` eingeleitet. Dieses ist auch über SecureShell möglich.

## SAMBA

### (JAN-HENDRIK TERSTEGGE)

- Lassen sich neben den Homeverzeichnissen der einzelnen User- auch Gruppenverzeichnisse einrichten? Dabei soll ein User durchaus Zugriff auf mehrere Gruppen haben können. Die Benutzer-/Gruppen-verwaltung darf jedoch die Benutzerverwaltung des ODS-Servers nicht zerstören. Evtl. sind hier manuelle Änderungen an smb.conf nötig.

Es ist mit Samba nicht möglich, neben den Homeverzeichnissen der einzelnen User auch einzelne Gruppenverzeichnisse einzurichten ohne im Quelltext von Samba 'herumzupfuschen'.
Es ist nur möglich, über die Option `[homes]` in der `/etc/samba/smb.conf` jedem einzelnen Benutzer ein eigenes Verzeichnis anzubieten. Die Option [homes] hat mehrere Einstellungsmöglichkeiten:

| OPTION | ERKLÄRUNG |
|---|---|
| `[homes]` | Mit dieser Option wird dem Benutzer ein Homeverzeichnis zugewiesen. Beim Anmelden am Samba-Server wird dem Benutzer automatisch ein Verzeichnis namens `/home/username` zugewiesen wobei ‚username' identisch mit dem Anmeldenamen ist. |
| `; available = no` | Mit dieser Option kann eingestellt werden, ob der Service verfügbar ist. Wenn erreicht werden soll, das |

| | |
|---|---|
| | die Benutzer nicht an ihre Homeverzeichnisse kommen, braucht man nur das Semikolon entfernen und Samba neuzustarten. |
| `comment = Hauptverzeichnis` | Dieser Text erscheint im Dateimanager des Anwenders, wenn dieser auf seine Shares zugreift. |
| `browseable = no` | Hiermit wird angegeben, ob dieses Verzeichnis auch von anderen Benutzern eingesehen werden darf. |
| `read only = no` | Legt fest, ob der Benutzer auch in das Verzeichnis schreiben darf |
| `create mode = 0755` | Diese Option legt fest, welche Unix-Dateiattribute Dateien unter DOS bzw. Windows erhalten. |
| `wide links = no` | Hiermit wird festgelegt, ob Links (Softlinks) auf andere Dateien oder Verzeichnisse verfügbar sind. |

Quellen: [01], [02], [03], [04], [05]

- Können Verzeichnisse für einzelne User völlig ausgeblendet werden?

Über die oben erwähnte Option ‚`browseable`' kann festgesetzt werden, das das Verzeichnis des Benutzers für alle anderen Benutzer nicht zugänglich ist, d.h. sie können das Verzeichnis weder benutzen, noch hineinschauen oder durchsuchen.

Quellen: [01], [03]

- Sind für verschiedene User auch unterschiedliche Einwahlscripte / Batch-Dateien möglich. Können Verzeichnisse für einzelne User völlig ausgeblendet werden?

Es ist nicht möglich für verschiedene User unterschiedliche Einwahlscripte zu spezifizieren. Das Problem besteht darin, das in der smb.conf die mit SWAT[2] automatisch erstellt wird, unter der Option 'logon script', nur ein Script spezifiziert werden kann. Dieses Script, das im Verzeichnis `/etc/samba/scripts` abgelegt wird, wird von dem anmeldenden Rechner heruntergeladen und ausgeführt. Dadurch bedingt wird immer nur die gleiche Datei ausgeführt.
Quellen: [01], [02], [03]

- Inwieweit kann man Samba so konfigurieren, das die höchstmögliche Sicherheit gewährleistet wird?

Bei der Konfiguration des Systems in Bezug auf Systemsicherheit müssen auch Samba-Konfigurationsoptionen entsprechend abgeändert werden, um höchstmögliche Sicherheit zu garantieren. Dazu müssen mit Hilfe von SWAT mehrere Optionen verändert werden: Die Variable 'hosts allow' muß auf 'ALL:LOCAL' gesetzt werden, damit alle Rechner im lokalen Netzwerk auf den Samba Server zugreifen können. Gleichzeitig muß die Variable 'hosts deny' auf 'ALL:ALL' gesetzt werden, damit alle anderen Rechner davon ausgeschlossen werden, Samba zu benutzen. Die Option 'encrypt passwords' muß auf 'NO' gesetzt bleiben, da alle Windows-Versionen unter NT 4.0 SP 3 oder Windows 98 plain text Passwörter übermitteln, d.h. die Passwörter werden nicht verschlüsselt übermittelt. Eine höhere Sicherheitsstufe könnte nur dadurch erreicht werden, das man eine der obengenannten Windows-Versionen einsetzt. Die Variable 'password level' muß von der Voreinstellung '8' auf '0' gesetzt werden. Einige Betriebssystem haben Probleme mit unterschiedlicher Groß- und

Kleinschreibung von Passwörtern und in der Art und Weise wie sie übermittelt werden. Die Voreinstellung '8' probiert alle Möglichkeiten der Groß- und Kleinschreibung automatisch aus, um das korrekte Passwort zu ermitteln. Die Option '0' wird verwendet um nur zwei verschiedene Möglichkeiten der Passwortverwendung zuzulassen: Zum einen in der Art und Weise, wie das Passwort geschrieben ist, mit Groß- und Kleinschreibung, falls verwendet, zum anderen in kompletter Kleinschreibung. Die Option '0' bietet die höchstmögliche Sicherheit im Umgang mit Passwörtern unter Samba. Die anderen Variablen die man mit dem Samba Web Administration Tool einstellen kann ('workgroup', 'netbios name', 'server string', 'interfaces', 'guest account', 'log level', 'map archive', mangled names', 'logon script', 'domain logons', 'os level', 'preferred master', 'local master', 'domain master', 'dns proxy', 'wins support' und 'wins server') müssen nicht verändert werden, um das erstrebte Resultat zu erreichen.
Quellen: [03], [04], [05]

[2] SWAT: **S**amba **W**eb **A**dministration **T**ool

# E – MAIL AUSTAUSCH

## (JULIAN FREDE)

- Fetchmail: Das Empfangen/Versenden per UUCP über Ping e.V. funktioniert richtig. Interessant sind aber folgende Fragestellung:
  1. Abruf der E-Mails auch noch von zusätzlichen Providern (z.B. mehrere Postfächer bei T-Online per POP 3)
  2. Zuordnung von E-Mail-Adressen (auch in anderer Form, z.B. informatik@pgu.de) zu Usernamen (z.B. mueller auf arktur)

Zuerst muss man über die Adminoberfäche des Arktur den Emailaustausch per SMTP[3] konfigurieren. Danach sollte man die Datei `/etc/mail/fetchmailrc` mit einem Editor (mcedit oder vi) als root wie folgt bearbeiten:

```
poll "[Mailserver des Accounts]"
protocol pop3
localdomains "[Die Lokale Domäne]"
username "[Benutzername]"
password "[Passwort]"
is * fetchall [wenn die Usernamen Lokal und auf dem
Account übereinstimmen. ODER:]
[Adresse vor dem @] is [Lokaler User] dieses sollte für
jeden User eingetragen werden, der eine Spezifische von
seinem Usernamen abweichen Adresse hat.
```

Um mehrere Accounts abzufragen müssen jeweils neue, dieser Form entsprechende Einträge in der Datei angelegt werden.

Um das Überschreiben dieser mühsam per Hand eingetragen Daten durch sysadm zu verhindern müssen folgende Zeile in der Datei `/usr/lib/ods-server/bin/smtp-config` auskommentiert werden: (Zeilen 81 - 90)

```
# awk -F':' '{
#print "poll\t\"" $1 "\""
#print "protocol\t" $2
#print "username\t\"" $3 "\""
#print "password\t\"" $4 "\""
#print "is * fetchall "
#}' SYS > fetchmailrc
#
```

[3] SimpleMailTransferProtocol

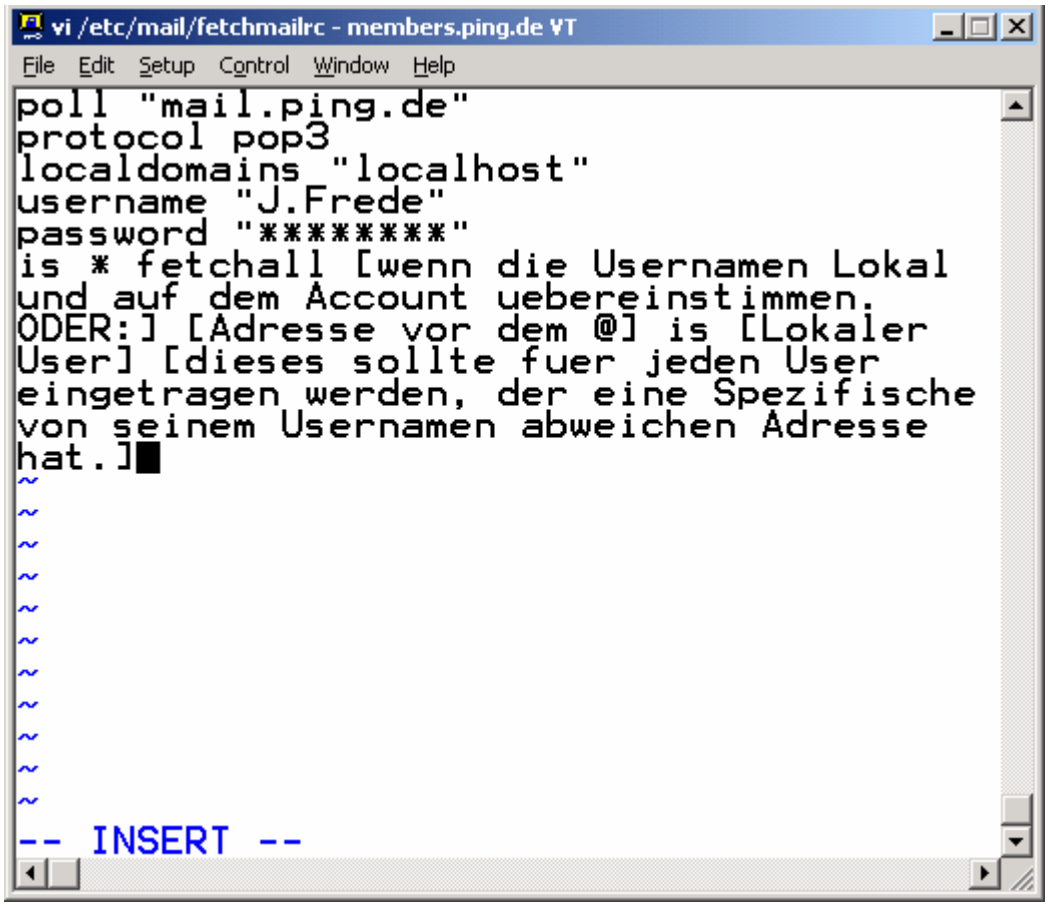
```
# chown root.root fetchmailrc
# chmod 600 fetchmailrc
```

Und ausserdem müssen in der Datei `/usr/lib/ods-server/bin/mailaustausch` die Zeilen 46 bis 49 auskommentiert werden. Es sollte dann wie folgt aussehen.

```
# if tcpping $SMTPSERVER 25 ; then
#echo `date` Sende Mails an Provider >> $LOGFILE
#sendmail -q
# fi
```

Die Änderungen an der Datei `/etc/mail/fetchmailrc` können unter Umständen bis zu einem Tag brauchen ehe sie in Wirkung treten.

Die folgende Grafik zeigt meine persönliche `/etc/mail/fetchmailrc` mit der ich meine E-Mails abholen kann.



## ANPASSUNGEN DES WEB-INTERFACES

### (JULIAN FREDE)

- Welche Möglichkeiten bestehen bei dem ODS-Kommunikationsserver Arktur das Webinterface zu verändern.

Der c't ODS Kommunikationsserver stellt dem Benutzer/Administrator eine einfach zu handhabende Möglichkeit der Administration oder Nutzung in Form eines Webinterfaces zur Verfügung.
Dieses Interface hat sich bei unserer Nutzung als äußerst nützlich erwiesen.
Die Anpassungsmöglichkeiten sind allerdings extrem Beschränkt.
Dieses begründet sich in der Tatsache, dass, wenn sich ausdrücken darf, es ein ziemliches Sammelsurium an Scripten ist. Teilweise sind es Shellscripte (Bash[4]). Jedoch bestehen sie größtenteils aus Skripten die in der Interpretersprache Perl gehalten sind. Allerdings gibt es auch kompilierte cgi[5] Skripte. Da man diese nicht anpassen kann ohne ziemlich in die Struktur einzugreifen, habe ich dieses nicht gemacht. Natürlich besteht diese Möglichkeit dennoch.
Aus Sicht eines Hobby Softwareentwicklers und Administrator verstehe ich die Entwickler von Arktur nicht da es doch möglich wäre auf bestehende Interfacesoftware wie Webmin aufzusetzen. Die den Zweck der Administration doch sehr gut erfüllt.
Dies ist meiner Meinung nach auch ein großes Manko des Arktur.

Der Administrator könnte jedoch Webmin nachträglich installieren, und die Oberfläche nach seinen Vorstellungen anpassen.

---

[4] Bourne Again Shell
[5] Common Gateway Interface

Hierzu braucht der Administrator jedoch erhebliche Kenntnis von der Linux Struktur und eine gewisse Einarbeitungszeit die mancher Administrator wohl nicht investieren will.

Alles in allem halte ich das Webinterface über für ausreichend und zweckerfüllend.

## QUELLENVERZEICHNIS

**[01]** Verschiedene Autoren. Samba FAQ. [online]
Kein Datum angegeben. Version vom 12.02.2001
http://de.samba.org/samba/docs/FAQ/index.html

**[02]** Hertel, Chris. Samba: An Introduction [online]
Version vom 10.06.1999.
http://de.samba.org/samba/docs/SambaIntro.html

**[03]** Sharpe, Richard. Just what is SMB?. [online]
Version 1.2 vom 27.09.1999.
http://anu.samba.org/cifs/docs/what-is-smb.html

**[04]** Auer, Karl & Allison, James. Samba Manpage (7) [online]
Version vom 23.10.1998
http://de.samba.org/samba/docs/man/samba.7.html

**[05]** Auer, Karl & Allison, James. smb.conf Manpage (5) [online]
Version vom 23.10.1998
http://de.samba.org/samba/docs/man/smb.conf.5.html

**[06]** Kirmse, Heinz-Dietrich. Problemlösungen für Arktur [online]
Version vom 08.02.2001
http://www.erg.slf.th.schule.de/arktur/faq/

**[07]** Klaproth, Reiner. Einrichten des ODS-Kommunikationsservers [online]
Version vom 05.08.2000
http://www.arktur-schule.de/inhalt.htm

**[08]** Kukula, Rüdiger. Linux (Hilfestellungen und Problemlösungen zum Arktur-Linux-Server) [online]
Version vom 12.02.2001
http://www.rkukula.de/Linux/linux.htm

**[09]** Fyodor (NMAP – Stealth Port Scanner for Network Security Auditing, General Internet Exploration & Hack) [online]
Version vom 04.01.2001
http://www.insecure.org/nmap

**[10]** SSH (OpenSSH) [online]
Version vom 26.02.2001
http://www.openssh.com

**[11]** Rootshell [online]
Version vom 26.02.2001
http://www.rootshell.org

**[12]** Webmin [online]
Version vom 27.02.2001
http://www.webmin.com/webmin/