Benutzer und Gruppen

Bei der Benutzerverwaltung wird geregelt, wer auf welche Dateien/Verzeichnisse zugreifen, wer welche Programme ausführen und wer auf Hardwarekomponenten zugreifen darf. Unter Linux werden mehrere Benutzer neben den eigentlichen Benutzern eingerichtet um Aufgaben zu verteilen. Viele Systemdienste haben ihre eigenen Benutzer. Erstellt ein Benutzer eine Datei ist die Datei immer Eigentum des Benutzers und der Gruppe zugeordnet die der Hauptgruppe des Benutzers entspricht.

Bei der Zuordnung von Benutzer und Gruppen gibt es jedoch Distributonsabweichungen. So wird z.B. bei Debian ein neu erstellter Nutzer „michael“ auch der Gruppe „michael“ zugewiesen. Bei anderen Distributionen könnte dies bei Nutzer „michael“ die Gruppe „user“ sein.

Alle Benutzer und Gruppenidentitäten werden als Zahlen „UID“ (userID) und „GID“ (gruppenID) im Prozesskontrollblock hinterlegt. Der user mit der UID 0: ist immer der Administator (root).

Diese IDs unterscheiden sich noch in:

- RUID, RGID ⇒ reale Benutzer/Gruppenident. des gestarteten Prozesses.
- EUID, EGID ⇒ effective Identität. Hier kann durch setzen des SetUID-Flags einem User mehr Rechte für bestimmte Aufgaben gewähren.
- SUID, SGID ⇒ bei der saved Identität. Sind mehr Rechte für User in EUID gesetzt und EUID unterscheidet sich von RUID wird der Wert in der SUID abgelegt, um die Identitäten leichter zu verändern.
- FSUID, FSGID ⇒ Identität für Zugriff auf Dateisysteme.

Standartisierte User und Gruppen

Unter Unix, Linux Systemen gibt es folgende drei Typen von Benutzer:

- `root` ⇒ Bezeichnet den SuperUser oder Systemadministrator und hat uneingeschränkte Rechte auf das System.
- `normaler User` ⇒ verwendet das System um damit zu Arbeiten, dabei hat der Nutzer uneingeschränkten Zugriff auf seine eigenen Dateien, jedoch nur eingeschränkten Zugriff auf das System.
- `System User` ⇒ Als System Nutzer bezeichnet man z.B. Dämonen und Serverdienste, die nicht interaktiv an der Arbeit am Computer vorgesehen sind. Der Apache Webserver nutzt einen eigenen User wie „www-data“, „httpd“, „www-run“ um eine möglichst hohe Systemsicherheit zu gewährleisten.

Verschiedene standardisierte Linux User:

- `individueller-username` ⇒ wird der Gruppe individueller-username hinzugefügt.
- `root` ⇒ Systemadministrator.
- `nobody` ⇒ vergibt nur ein minimum an Rechten, wird von Prozessen als Benutzerkennung verwendet.
- `cupsys` ⇒ Benutzer des Druckerdienstes
- `www-data` ⇒ Benutzer des Apache Webservers unter Debian.
- `apache` ⇒ Benutzer des Apache Webservers unter CentOS.

Verschiedene standardisierte Gruppen:

- `adm` ⇒ User der Gruppe adm können Logdateien einsehen.
- `cdrom` ⇒ CD Laufwerke benutzen.
- `floppy` ⇒ Diskettenlaufwerk benutzen.
- `audio` ⇒ Audiogeräte verwenden.
- `plugdev` ⇒ Booten von Externen Medien.
- `admin` ⇒ sudo ausführen. (Debian)
- `wheel` ⇒ sudo ausführen. (CentOS)

Alle Benutzer werden in der Datei `/etc/passwd` mit `Loginname, Name, UID, GID, Homeverzeichnis und Shell` gespeichert.

Gruppen haben die Aufgabe, verschiedenen Benutzern einen gemeinsamen Zugriff auf Dateien zu ermöglichen. Dabei wird jeder Benutzer einer bestimmten Gruppe zugeordnet. Ein Benutzer kann beliebig vielen Gruppen zugeordnet sein. Alle Gruppen werden in der Datei `/etc/group` gespeichert. Bei der Zuordnung von Benutzern und Gruppen gibt es unterschiedliche Ansätze. Bei SuSE wird ein neuer Benutzer Standardmäßig der Gruppe „users“ zugeordnet. Bei Debian und Red Hat bekommt der Benutzer seine eigene primäre Gruppe die dem Nutzernamen entspricht.

Konfigurationsdateien

- `/etc/default/useradd` ⇒ Datei mit Vorgaben für neue Accounts.
- `/etc/skel` ⇒ Verzeichnis mit Standarddateien für neue Accounts (option -m).
- `/etc/passwd` ⇒ listet alle Accounts auf.
- `/etc/group` ⇒ listet Gruppennamen und GID.
- `/etc/shadow` ⇒ verschlüsselte Passwörter + Zeitlimits für Passwörter.
- `/etc/login.defs` ⇒ Loginstandards festlegen/bearbeiten.
- `/etc/skel/*` ⇒ Der Inhalt dieses Verzeichnisses wird beim Anlegen eines neuen Nutzers in sein Heimatverzeichnis kopiert.

Informationen über Benutzer

Informationen und Konfigurationsdateien über Benutzer und Gruppenzugehörigkeiten.

id

Reale und effektive Benutzer- und Gruppen-IDs ausgeben.

Syntax:

id [opt] [nutzername]

Optionen:

-a ⇒ bei Debian ignoriert (für Kompatibilität mit anderen Versionen). -Z ⇒ Nur Sicherheitskontext des aktuellen Benutzers ausgeben, (funktioniert nur auf einem Kernel mit SELinux). -g ⇒ Nur effective

GruppenID ausgeben. -G ⇒ Alle GruppenIDs ausgeben. -n ⇒ Einen Gruppennamen ausgeben. (Im Vorgabe-Format ist es nicht möglich, nur Namen oder echte IDs auszugeben). -r ⇒ reale statt effective IDs ausgeben. (Im Vorgabe-Format ist es nicht möglich, nur Namen oder echte IDs auszugeben). -u ⇒ Nur effective UserID ausgeben. –help ⇒ kurze Hilfe. –version ⇒ Versionsinformationen anzeigen.

Beispiel: rudi@home:~$ id uid=1000(rudi) gid=1000(rudi) Gruppen=1000(rudi),24(cdrom),25(floppy),29(audio),30(dip),44(video),46(plugdev),108(netdev),109(bluetooth),112(fuse),115(scanner) rudi@home:~$ id -G 1000 24 25 29 30 44 46 108 109 112 115

---

passwd

Mit „passwd“ lassen sich Passwörter von Benutzer und/oder Gruppen verändern. Benutzer und Gruppenangehörige dürfen nur ihr eigenes Passwort, „root“ hingegen alles bearbeiten. Wenn „passwd“ ohne Nutzernamen und Optionen eingesetzt wird, kann das Passwort des angemeldeten Nutzers geändert werden.

Syntax:

passwd [opt] [nutzername]

Optionen:

-a, –all ⇒ Zeigt den Status aller Benutzer. (kann nur in Verbindung mit -S verwendet werden.) z.B „passwd -S -a“. -d, –delete ⇒ Löscht das Passwort eines Benutzers. -e, –expire ⇒ Lässt das Passwort eines Benutzers sofort verfallen, der Nutzer müsste beim nächsten Login sein Passwort ändern. -i [tage], –inactive [tage] ⇒ Deaktiviert Konto, nachdem das Passwort nach xxTagen abgelaufen ist. Wenn ein Nutzer ein Passwort länger als „inaktive“ Tage hat, kann sich der Nutzer auf dem Konto nicht mehr anmelden. -l, –lock ⇒ Sperrt das Passwort des Accounts. Der Nutzer könnte sich z.B. noch über SSH keys am System anmelden. -n [tage], –mindays [tage] ⇒ Anzahl der Tage zwischen Änderungen des Passworts. -S, –status ⇒ Zeigt Account Status Informationen in 7 Felder: Feld1=Loginname; Feld2=PasswortInfo (L=gesperrtes)(NP=kein)(P=benutztes) Passwort; Feld3=Datum des letzten Passwortwechsels; Feld4=min.Passwortgülitigkeit[Tage]; Feld5=max.Passwortgülitigkeit[tage]; Feld6=warnung vor Passdel.[tage]; Feld7=Gültigkeitsdauer[Tage]. -u, –unlock ⇒ reaktiviert ein gesperrtes Passwort. -w [tage], –warndays[tage] ⇒ Anzahl der Tage, an dem der Benutzer eine Warnung erhält, bis das Passwort ungültig wird. -x [tage], –maxdays[tage] ⇒ Anzahl der Tage, die das Passwort gültig bleibt. -h, –help ⇒ kurze Hilfe.

Beispiel: michael@home:~$ passwd michael Geben Sie ein neues UNIX-Passwort ein: Geben Sie das neue UNIX-Passwort erneut ein: passwd: Passwort erfolgreich geändert

michael@home:~$ passwd -S michael michael P 06/25/2012 0 99999 7 -1

/etc/passwd

Die Datei „/etc/passwd“ listet alle Metadaten der Nutzer, deren Gruppenzugehörigkeit und Shell auf. Da diese Datei von jedem lesbar sein muß, wird das Passwort in der Datei „/etc/shadow“, die nur von

„root“ und/oder der Gruppe „shadow“ lesbar sein muß.

less /etc/passwd ….. root:x:0:0:tf,,,:/root:/bin/bash daemon:x:1:1:daemon:/usr/sbin:/bin/sh bin:x:2:2:bin:/bin:/bin/sh sys:x:3:3:sys:/dev:/bin/sh ….. michael:x:1000:1000:michael,,,:/home/michael:/bin/bash …..

Erstes Feld “michael” ⇒ Benutzername (Loginname). Zweites Feld “x” ⇒ Zeigt verschlüsseltes Passwort (von /etc/shadow). Zum sperren des Kontos kann man hier ein „*“ eintragen. Drittes Feld “1000” ⇒ UID (userID). Viertes Feld “1000” ⇒ GID (groupID). Fünftes Feld “michael” ⇒ GECOS Feld, kann Infos über Nutzer enthalten z.B. voller Name, Tel. usw. Kann über finger abgefragt werden. Sechstes Feld “/home/michael” ⇒ Heimatverzeichnis des Users. Siebtes Feld “/bin/bash” ⇒ Login Shell des Nutzers. Hier kann auch „/bin/false“ angegeben werden, sollte der Nutzer sich nicht anmelden dürfen, sondern nur Mails abholen.

/etc/group

Konfigurationsdatei für Benutzergruppen.

less /etc/group ….. root:x:0: daemon:x:1: bin:x:2: sys:x:3: adm:x:4: tty:x:5: disk:x:6: lp:x:7: mail:x:8: news:x:9: bettina:x:502: verkauf:x:503:bettina …..

Erstes Feld “bettina” ⇒ Name der Gruppe. Zweites Feld “x” ⇒ Verschlüsseltes Passwort oder “x”, ein Verweis zur /etc/gshadow. Drittes Feld “502” ⇒ Gruppen ID (GID). Viertes Feld “bettina” ⇒ Liste der Gruppenmitglieder, bei mehreren Gruppenmitglieder durch Komma getrennt.

groups

Der Befehl „groups“ gibt die Gruppenzugehörigkeit der Nutzer aus. Dabei kann der Administrator „root“ mit „groups [Nutzername]“ alle Gruppenzugehörigkeiten der registrierten Nutzer einsehen, der angemeldete Nutzer selbst, nur seine eigenen Gruppenzugehörigkeiten.

michael@home:~$ groups michael cdrom floppy audio video plugdev scanner

---

/etc/shadow

In der Datei „/etc/shadow“, dass nur von „root“ lesbar ist, wird der Nutzername, das verschlüsselte Passwort und weitere Felder zur Gültigkeit gespeichert.

… michael:3jX6784T$XXXXXXXXXXXXXXXXXXXXXXXXXXXXXXKA1jWfli0:XXX23:0:XXX90:5::: bettina:1F7vuc.lg$bhr1g/GOb94zCQyUhHDhy.:15659:7:92:7:14:15916: daemon:*:XXX23:0:XXX90:5::: bin:*:XXX23:0:XXX90:5::: sys:*:XXX23:0:XXX90:5::: …

- Erstes Feld “bettina” ⇒ Nutzernamen muß mit dem in der „/etc/passwd“ übereinstimmen.
- Zweites Feld “1F7vuc…” ⇒ Verschlüsseltes Passwort. Sollte hier ein [*] oder [!] stehen, ist der Login gesperrt.
- Drittes Feld “15659” ⇒ Letzte Passwortänderung. Wert wird in Tagen ab 1. Januar 1970 angezeigt (Unix Timestamp).

- Viertes Feld “7” ⇒ Minimale Gültigkeitsdauer des Passworts (in Tagen). In wievielen Tagen das Passwort erneut geändert werden muß.
- Fünftes Feld “92” ⇒ Maximale Gültigkeitsdauer des Passworts (in Tagen).
- Sechstes Feld “7” ⇒ Zeigt in Tagen, wann Nutzer vor Ablauf des Passwortes gewarnt wird.
- Siebtes Feld “14”⇒ Tage bis zur Deaktivierung des Accounts nach Ablauf des Passworts. Wieviele Tage nach Ablauf des Passworts der Nutzer sich noch einloggen darf.
- Achtes Feld “15916” ⇒ Zeigt im Linux Timesamp (Zeit ab 1.1.1970) Ablaufdatum des Accounts.
- Neuntes Feld ⇒ Frei für spätere Erweiterungen.

/etc/login.defs

In der Datei „/etc/login.defs“ befinden sich Möglichkeiten zur Administration von Passwörter und Logins. Hier kann man z.B. einstellen wie lange die Wartezeit nach einem gescheiterten Loginversuch ist und wieviele Loginversuche gestattet sind. Je nach Einstellung dieser Datei wird bei jedem fehlerhaften Login eine Meldung in „/var/log/faillog“ im Binärformat gespeichert.

---

faillog

Die Datei „/var/log/faillog“ protokolliert, sofern in der Datei „/etc/login.defs“ eingestellt die fehlerhaften Login Versuche.

Syntax:

faillog [opt] [args]

Optionen:

-a, –all ⇒ Zeigt Fehlgeschlagene Loginversuche aller Nutzer. -h, –help ⇒ Kurze Hilfe. -l [sec], –lock-secs [sec] ⇒ Bei fehlerhaften Login wird der Account für [x] Sekunden gesperrt. -m [x], –maximum ⇒ setzt [x] Anzahl der Loginversuche. -r, –reset ⇒ Setzt den Zähler für maximale Loginversuche zurück. -t [tage], –time [tage] ⇒ Zeigt mehr Tage, anstatt nur kürzliche fehlerhafte Login Einträge. -u [login], –user [login] ⇒ Zeigt fehlerhafte Login Versuche von [Nutzer].

michael@home:~$ faillog -a Login Fehlver. Maximum Letzter Auf

root 0 0 01/01/70 01:00:00 +0100 daemon 0 0 01/01/70 01:00:00 +0100 bin 0 0 01/01/70 01:00:00 +0100 sys 0 0 01/01/70 01:00:00 +0100 …. Zeigt fehlerhafte Loginversuche aller Benutzer.

Beispiele: faillog -u michael -m 5 Für User michael werden 5 fehlerhafte Loginversuche gestattet, danach wird der Login blockiert und kann danach nur von „root“ freigeschalten werden.

faillog -u michael -r Der Loginzähler wurde zurückgesetzt, der Benutzer „michael“ kann sich wieder einloggen.

Durch „faillog -m 5“ kann mann die Anzahl der Loginversuche global festlegen. Dann sollte „root“ „faillog -u root -m 0“ ausführen, ansonsten kann es passieren, dass sich „root“ nicht mehr Anmelden kann, wenn ein anderer Benutzer sich 5x vergeblich als „root“ anmelden wollte.

gpasswd

Mit dem Kommando „gpasswd“ kann ein Gruppenpasswort vergeben werden, dieses wird in /etc/gshadow gespeichert. Ein Gruppenpasswort ist nötig, sollte ein Benutzer mit dem Kommando newgrp die aktive Gruppe wechseln, da die aktive Gruppe festlegt, welcher Gruppe neue Dateien angehören.

Verwaltung des Shadowsystems

pwconv

Dieses Programm erstellt oder aktualisiert die Datei „/etc/shadow“, dabei sucht „pwconv“ in der Datei „/etc/passwd“ nach Passwörter und verschiebt diese nach „/etc/shadow“, sollte dies nicht schon geschehen sein (x im Passwortfeld der passwd). Optionen sind bei diesem Kommando nur notwendig, sollten sich die Dateien nicht im Verzeichnis „/etc“ befinden.

pwunconv

Hier werden verschlüsselte Passwörter aus der Datei „/etc/shadow“ in die „/etc/passwd“ verschoben und anschließend „/etc/shadow“ gelöscht.

pwck

Prüft die Integrität der Dateien „/etc/passwd“ und „/etc/shadow“.

Beispiel: root@home:~# pwck pwck: Keine Änderungen

grpconv

Das Kommando ist aquivalent zu „pwconv“, es werden lediglich die Dateien „/etc/group“ und „/etc/gshadow“ eingelesen.

grpunconv

Das Kommando ist aquivalent zu „pwunconv“, es werden lediglich die Dateien „/etc/group“ und „/etc/gshadow“ eingelesen.

grpck

Prüft die Integrität der Dateien „/etc/group“ und „/etc/gshadow“ (wie „pwck“).

User wechseln

Linux und BSD sind mehrbenutzer Systeme. Um sich z.B. als Systemadministator einzuloggen ist der Nutzer „root“ vorgesehen. Nur „root“ kann wichtige Konfigurationen vornehmen.

Hierfür gibt es folgende Befehle:

su

Mit „su“ (Switch User) läßt sich eine neue Shell des gewählten Nutzers starten. Hierfür ist das Passwort des Nutzers notwendig, nur „root“ darf in alle Nutzer Shells abtauchen.

Syntax:

su [opt] [nutzername]

Optionen: -l ⇒ Login-Shell starten. -m ⇒ In neuer Shell, aktuelle Umgebungsvariablen beibehalten. -s /bin/[shell] ⇒ Die zu startende Shell z.B. „/bin/bash“.

michael@home:~$ su root Passwort: root@home:/home/michael# Oben wird der „su“ Befehl ohne Optionen ausgeführt. „root“ bekommt eine Shell und befindet sich im letzten Verzeichnis des Nutzers „michael“.

michael@home:~$ su - root Passwort: root@home:~# Hier wurde vor dem Nutzernamen ein [-] hinzugefügt, dadurch landet man beim Benutzerwechsel gleich im jeweiligen Heimatverzeichnis des angegebenen Nutzers.

Mit der Option -c [user] kann ein gewünschter Befehl als [user x] ausgeführt werden.

sudo

Hier wird keine Shell als „root“ oder „nutzerrname“ gestartet, sondern führt das Kommando nur einmalig mit „root“, „nutzername“ rechten aus. Wird auch hier kein Nutzer angegeben, startet das Programm mit „root“ rechten. In der Konfigurationsdatei „/etc/sudoers“ kann das erlauben oder verbieten der Nutzung von „sudo“ für die jeweiligen Nutzer gesteuert werden. Der „sudo“ Befehl wird Standardmäßig bei Ubuntu Systeme verwendet.

Sinnvoll ist das z.B., wenn ein Nutzer gelegentlich, für ein Kommando „root“ Rechte benötigt.

Syntax:

sudo [opt] [user] [kommando]

Optionen:

-b ⇒ übergibt das sudo Kommando als Hintergrundprozess. Kann jedoch mit job controll nicht manipuliert werden. -h ⇒ Kurze Hilfe. -v ⇒ User kann Timestamp des Prozesses erneuern. -u ⇒ Benutzer(identität annehmen).

sudo ohne -u [useridentität] ausgeführt, wird root als Identität genommen.

SetUID / SetGID

Durch das setzen von „SetUID / SetGID“ Bits können die Berechtigungen eines anderen Nutzers verwendet werden. Diese Rechte werden auf Dateien gesetzt und mit den Rechten des Besitzers oder der Gruppe der Datei ausgeführt. Weitere Informationen hierzu in Dateirechte.

Nutzer bearbeiten

Für die verschiedenen Aufgaben zur Benutzerverwaltung gibt verschiedene Kommandos zur Bearbeitung. Die Befehle „adduser“, „deluser“, „addgroup“ und „delgroup“ sind Debian spezifische

Kommandos. Die Standardkommandos zur Nutzerverwaltung, „useradd“, „groupadd“ usw. berücksichtigen bei Debian Systeme die in den „/etc/adduser.conf“ und „/etc/deluser.conf“ definierten Regeln. Bei Red Hat und Fedora sind die Kommandos „adduser“, „addgroup“ usw. lediglich Links auf „useradd“ bzw. „groupadd“ usw., weshalb sich hier die Syntax nicht unterscheidet, wie z.B. bei Debian .

Beim Anlegen eines neuen Nutzers kann das Verzeichnis „/etc/skel/“ so modifiziert werden, dass die Verzeichnisstruktur innerhalb von „/etc/skel/“ in das Heimatverzeichnis des neuen Nutzers kopiert und mit den jeweiligen Rechten ausgestattet werden.

useradd

Hinzufügen eines neuen Nutzers mit dem Befehl „useradd“.

Syntax:

useradd [Optionen] [nutzername]

Optionen:

-d [/home/name] ⇒ Verwendet “name” anstatt Benutzername als Namen für das Heimatverzeichnis des neuen Benutzers. -D ⇒ Zeigt Standardeinstellungen von „useradd“. -b ⇒ Default Heimatverzeichnis. -s [shell] ⇒ Standardshell für neuen Benutzer. -c ⇒ Comment, Hier kann beliebiger Text stehen. -u [uid] ⇒ Setzt gewünschte Benutzerid [uid] für neuen Benutzer. -g [gid] ⇒ Standardgruppe für neuen Benutzer. -G [group1, group2] ⇒ Benutzer in weitere Gruppen hinzufügen. –ingroup [group] ⇒ Standardgruppe für Benutzer. -m ⇒ Heimatverzeichnis anlegen und Standardverzeichnisse von „/etc/skel“ Verzeichnis hineinkopieren (siehe /home). -e [YYYY-MM-DD] ⇒ Gültigkeitsdauer des Accounts (Ablaufdatum). -f [tage] ⇒ Anzahl der Tage, nachdem ein Account verfällt nachdem das Passwort ungültig geworden ist. -1 deaktiviert dieses Feature(default). -h ⇒ Hilfe Anzeigen und Exit. -k ⇒ Zeigt auf Skeleton Datei, die Verzeichnisse in „/home/[nutzername]/[Verzeichnisse]“ anlegt. -l ⇒ Nutzer nicht in die „faillog“ und „lastlog“ Datenbanken eintragen.

Beispiele: useradd -m -s /bin/bash -c testuser michael Hier wird der neue Nutzer „michael“ in die Gruppe „michael“ und ein Heimatverzeichnis „/home/michael/*“ mit Standardverzeichnissen aus den Dateien in „/etc/skel“ angelegt. Als Shell bekommt der neue Nutzer „michael“ die Standardshell „bash“ zugewiesen, als Kommentar wurde „testuser“ definiert.

root@home:~# useradd -D GROUP=100 HOME=/home INACTIVE=-1 EXPIRE= SHELL=/bin/sh SKEL=/etc/skel CREATE_MAIL_SPOOL=no Zeigt die Standardeinstellungen zu „useradd“. Die Konfigurationsdatei zu „useradd“ befindet sich in der Datei „/etc/default/useradd“.

Soll mit „useradd -g“ der neue Nutzer einer Gruppe zugeführt werden, muß diese bereits angelegt sein.

Legt man einen neuen Nutzer ohne die Option [-m] an, kann es bei der Anmeldung zu Fehlermeldungen kommen.

adduser (Debian)

Ist ein Debian spezifisches Kommando, wobei ein angelegter Benutzer zu seiner Gruppe, jedoch keiner Systemgruppe hinzugefügt wird. Nach dem Start von „adduser“ wird ein Dialog in der Shell angezeigt, der auf Eingaben wartet.

Gibt man hier keine Optionen an, werden die Voreinstellungen der Datei „/etc/adduser.conf“ (Standard Konfiguratinsdatei von „adduser“ und „addgroup“) verwendet, sowie die Konfigurationsdateien von „/etc/skel“ in das Heimatverzeichnis des neuen Nutzers hineinkopiert. „adduser“ vergibt dann die erste freie „UID“ des in der Konfigurationsdatei festgelegten Bereichs.

Syntax:

adduser [optionen] [benutzernamen]

Optionen:

–home [pfad/to/home] ⇒ Heimatverzeichnis und Arbeitsverzeichnisse aus „/etc/skel“ werden hier angelegt. –no-create-home ⇒ Legt kein Heimatverzeichnis für neuen Nutzer an. –conf [pfad zur/datei] ⇒ Verwendet [datei] als Konfigurationsdatei. –ingroup [gruppe] ⇒ Neuer Nutzer wird der Gruppe [gruppe] hinzugefügt, dabei muß die Gruppe schon vorhanden sein. –shell [/bin/shell] ⇒ Festlegen der Loginshell des neuen Nutzers. Gibt man hier „/bin/false“ an, bekommt der neue Nutzer keine Login-Shell. –disabled-login ⇒ Anmeldung ohne Passwort. –disabled-password ⇒ Anmeldung mit SSH-RSA Schlüsseln ist möglich jedoch ohne Passwort. –uid [UID] ⇒ Setzt die UID als Zahl auf UID. –firstuid ⇒ Die UID Vergabe festlegen. –lastuid ⇒ Die UID Vergabe festlegen. –gid [GID] ⇒ Bei neuer Gruppe wird die Zahl GID der Gruppe übergeben. –group ⇒ Gruppe für Nutzer anlegen. –help ⇒ Kurze Hilfe und Exit. –quiet ⇒ Nur Warnungen und Fehler anzeigen, Meldungen werden unterdrückt. –debug ⇒ Ausführliche Fehlermeldungen ausgeben. –system ⇒ Systemnutzer oder Systemgruppe einrichten.

Beispiel: adduser michael –ingroup gruppe Hier wird der neue Nutzer „michael“ der Gruppe [gruppe] hinzugefügt.

Standardmäßig wird bei neu angelegten Benutzern das Heimatverzeichnis Systemweit (auch von anderen Benutzern) lesbar sein. Will man das nicht, kann man den unten genannten Befehl verwenden.

root@home:~# dpkg-reconfigure adduser

adduser.jpg

usermod

Nutzer Einstellungen bearbeiten.

Syntax:

usermod [opt] [nutzername]

Optionen: -d [pfad/zu] ⇒ Neues Heimatverzeichnis für [nutzername]. -m ⇒ Den Inhalt des Heimat-Verzeichnisses an den neuen Ort verschieben.(Nur mit [-d] benutzen). -c [kommentar] ⇒ Kommentar im GECOS-Feld. -e [yyyy-mm-dd] ⇒ Ablaufdatum für Nutzerkonto setzen. -f [tage] ⇒ Anzahl der Tage, nachdem Passwort abgelaufen ist, bis das Nutzerkonto dauerhaft deaktiviert wird. -l [nutzername] ⇒ Neuen Loginnamen festlegen. -s [shell] ⇒ Neue Standardshell zuweisen. -u [uid] ⇒ Ändert die userid. (!) -g [gid] ⇒ Weist dem Nutzer eine neue Standard Gruppe zu. -G [gruppe1][gruppe2].. ⇒ Weist dem Nutzer mehreren Gruppen zu. Wird diese Option ohne -a ausgeführt, wird der Nutzer den angegebenen Gruppen hinzugefügt und aus allen anderen entfernt! -a ⇒ Benutzer zu zusätzlichen Gruppen hinzufügen, ohne ihn dabei aus anderen Gruppen zu entfernen.(Nur mit der Option -G anwenden). -L [nutzername] ⇒ Login sperren. -U [nutzername] ⇒ Login entsperren. -o ⇒ Benutzung von nicht einmaliger UID erlauben. -p ⇒ Neues Passwort für Nutzer setzen (verschlüsseltes Passwort

von crypt). Besser ist es, passwd zu verwenden. -Z ⇒ Neue SELinux-Benutzer-Zuordnung für den Benutzerzugang. -h ⇒ Kurze Hilfe.

Zum ändern der User Einstellungen darf der Nutzer nicht angemeldet sein.

Beispiele: root@home:/# usermod -G video,audio,www-data michael Hier wird der Nutzer „michael“ den Gruppen „video“, „audio“ und „www-data“ hinzugefügt.

root@home:/# usermod -d /home/verwaltung/michael -m michael Den Nutzer „michael“ mit seinen Heimatverzeichnissen nach „/home/verwaltung/michael“ umziehen.

userdel

Löscht einen Nutzer und seine Verzeichnisse.

Syntax:

userdel [opt] [nutzername]

Optionen: -r ⇒ Löscht Heimat und Mail Verzeichnisse des Nutzers. -f ⇒ Löscht das Nutzerkonto auch wenn der Nutzer eingeloggt ist. Es werden auch Dateien im Heimatverzeichnis gelöscht, die nicht dem Nutzer gehören. -h ⇒ Zeigt Hilfe an und Ende.

Rückgabewerte: 0 ⇒ Kommando erfolgreich. 1 ⇒ Kann nicht in „passwd“ schreiben. 2 ⇒ Syntaxfehler im Kommando. 6 ⇒ Ausgewählter Nutzer existiert nicht. 8 ⇒ Nutzer ist gerade eingeloggt. 10 ⇒ Kann nicht in „group“ schreiben. 12 ⇒ Kann Heimatverzeichnis nicht löschen.

Beispiel: [root@home ~]# userdel -rf karl && echo $? 0 Löscht den Nutzer „karl“ inclusive Heimatverzeichnis, Mailverzeichnis und Dateien im Nutzerverzeichnis die nicht „Karl“ gehören und gibt dabei den Wert [0] (Aktion erfolgreich) zurück.

deluser (Debian)

Nutzer und Gruppen aus dem System entfernen. Bedinungsfreundlichere Frontends für „userdel“ und „groupdel“. Ohne Optionen entfernt „deluser“ den Benutzer, ohne das Heimatverzeichnis, die Emails und andere dem Benutzer gehörende Dateien zu löschen.

Wird „deluser“ mit der Option „–group“ angewendet, werden auch die jeweiligen Gruppen vom System entfernt.

Syntax: deluser [opt] [nutzer] deluser [opt] [nutzergruppe]

Optionen: –remove-home ⇒ Löscht auch das Heimatverzeichnis und den E-mail Puffer. –remove-all-files ⇒ Löscht alle Dateien des Benutzers vom System. –backup ⇒ Sichert die Benutzerdateien vor dem löschen als „username.tar.gz“ ins aktuelle Verzeichnis. –backup-to [verz./pfad] ⇒ Sichert die Benutzerdateien vor dem löschen als „username.tar.gz“ ins angegebene Verzeichnis. –system ⇒ Nutzer oder Gruppe nur entfernen, wenn es ein System-Nutzer oder System-Gruppe ist. –group ⇒ Entfernt eine Gruppe. Die Primäre Gruppe eines bestehenden Nutzers kann nicht gelöscht werden. –only-if-empty ⇒ Entfernt die Gruppe nur, wenn sie keine Mitglieder mehr hat.

Rückgabewerte: 0 ⇒ Kommando erfolgreich ausgeführt. 1 ⇒ Zu löschender Nutzer hatte kein Systemkonto. 2 ⇒ Nutzer existiert nicht. 3 ⇒ Gruppe existiert nicht. 4 ⇒ Interner Fehler. 5 ⇒ Gruppe wurde nicht gelöscht, da sie noch Mitglieder hat. 6 ⇒ Der Nutzer gehört nicht zur angegebenen

Gruppe. 7 ⇒ Nutzer kann nicht aus seiner primären Gruppe entfernt werden. 8 ⇒ Erforderliches Paket (perl-modules) ist nicht installiert. 9 ⇒ Für die Entfernung von „root“ ist die Option –force erforderlich.

chfn

Nutzerkonto Informationen auflisten / zuweisen. Diese Informationen werden z.B. von dem Programm finger verwendet. Ein Nutzer darf nur seine eigenen Felder bearbeiten, diese können in der Datei /etc/login.defs eingeschränkt werden. „chfn“ [nutzername] ohne Optionen, geht alle Infos durch und wartet auf Bestätigung.

root@home:/# chfn michael Benutzerinformationen für michael werden geändert. Geben Sie einen neuen Wert an oder drücken Sie ENTER für den Standardwert

```
Vollständiger Name [michael Dampf]:
Raumnummer []:
Telefon geschäftlich [0123456]:
Telefon privat [0123456]:
Sonstiges []:
```

Syntax: chfn [opt] [nutzername]

Optionen: -f [voller name] -r [room nr.] -w [tel. geschaftl.] -h [tel. privat] -o [sonstiges]

chsh

Das Kommando „chsh“ (change shell), dient zum ändern der „Login-Shell“. Dabei darf ein eingeloggter Nutzer nur seine eigene, „root“ die Shells aller Nutzer ändern. In der Datei /etc/shells sind alle möglichen Login-Shells aufgelistet.

Syntax: chsh [opt] [nutzername]

Optionen: -h ⇒ Zeigt Kurzhilfe an. -l ⇒ Liste aus „/etc/shells“. -s [/bin/shell] ⇒ Ändert Loginshell des Nutzers, ohne diese Option wird die Standard Login-Shell zugewiesen.

Beispiel: chsh -s /bin/csh michael Hier wird dem Nutzer „michael“ die csh Shell als Login-Shell zugewiesen.

„chsh“ ohne die Option -s startet das Programm im interaktiven Modus, gibt man eine Shell an, wird die Login-Shell geändert, gibt man ein leeres Zeichen ein, bleibt der alte Wert erhalten.

chage

Ist ein Kommandozeilen Werkzeug zum festlegen der Tage zwischen Passwort änderungen, b.z.w., wann ein Benutzer sein Passwort ändern muß. Als „root“ ausgeführt erwartet das Programm eine Eingabe, als Benutzer kann man mit der Option [-l] seine Parameter einsehen.

Syntax: chage [opt] [nutzer]

Optionen: -d [YYYY-MM-DD] ⇒ Tag der letzten Kennwortänderung ändern. -E [YYYY-MM-DD] ⇒ Ablauf des Benutzerzugangs. -I [tage] ⇒ Anzahl der Tage, nachdem Passwort abgelaufen ist, bis Account dauerhaft deaktiviert wird. -l ⇒ Accountinfos anzeigen. -m [tage] ⇒ Minimale Gültigkeitsdauer des Passworts. Wird hier [0] angegeben, muß der Nutzer beim nächsten Login sein Passwort ändern. -M [tage] ⇒ Maximale Gültigkeitsdauer des Passworts. -W [tage] ⇒ Wieviele Tage vorher wird vom Ablauf

des Passwortes gewarnt. -h ⇒ Hilfe anzeigen und Ende.

Rückgabewerte: 0 ⇒ Kommando wurde erfolgreich abgeschlossen. 1 ⇒ Zugriff Verweigert. 2 ⇒ Syntax Fehler. 15 ⇒ Kann „shadow“ oder „passwd“ Datei nicht finden.

Beispiele: michael@home:~$ chage -l michael Letzte Passwortänderung : Jan 04, 2010 Passwort läuft ab : nie Passwort inaktiv : nie Benutzerzugang läuft ab : nie Minimale Anzahl der Tage zwischen Passwortänderungen : 0 Maximale Anzahl der Tage zwischen Passwortänderungen : 99999 Anzahl Tage, an denen vor Passwortablauf gewarnt wird : 7

root@home:# chage michael Passwortalterung für michael wird geändert. Geben Sie einen neuen Wert an oder drücken Sie ENTER für den Standardwert

```
Minimales Passwortalter [0]:
Maximales Passwortalter [99999]:
Letzte Passwortänderung (JJJJ-MM-TT) [2010-01-04]:
Passwortablaufwarnung [7]:
Passwort inaktiv [-1]:
Ablaufdatum des Benutzerzugangs (JJJJ-MM-TT) [2012-12-31]:
```

Gruppen bearbeiten

Für die verschiedenen Aufgaben zur Benutzerverwaltung gibt es, wie oben schon Beschrieben mehr Kommandos zur Bearbeitung. Das gilt natürlich auch für die Gruppenbearbeitung.

groupadd

Eine neue Gruppe hinzufügen.

Syntax: groupadd [opt] [gruppe]

Optionen: -g [GID] ⇒ Gruppen ID als Zahl setzen. -f ⇒ –force ⇒ Beendet Kommando, wenn Gruppe bereits existiert. -h ⇒ Zeigt kurze Hilfe. -p ⇒ Ein Gruppenpasswort setzen. -r ⇒ Erstellt eine Systemgruppe.

groupmod

Gruppeneigenschaften ändern.

Syntax: groupmod [opt] [gruppe]

Optionen: -A [user] ⇒ Fügt Nutzer zu einer Gruppe hinzu. -R [user] ⇒ Entfernt Nutzer von Gruppe. -n [gruppe] ⇒ Gruppenname ändern. Die Gruppen-ID und Mitglieder der Gruppe bleiben erhalten. -g [GID] ⇒ GruppenID ändern. -p [password] ⇒ Gruppenpasswort setzen. -h ⇒ Kurze Hilfe.

groupdel

Gruppen löschen. Eine Übergabe von Optionen ist nicht erforderlich.

Syntax: groupdel [opt] [gruppe]

Optionen: wie userdel.

delgroup (Debian)

Gruppe löschen. Debian spezifisch.

Syntax: delgroup [opt] [gruppe]

Optionen: wie deluser.

gpasswd

Das Kommando „gpasswd“ dient zum bearbeiten von Gruppenzugehörigkeiten (/etc/group) und Gruppenpasswörter (/etc/gshadow).

Syntax: gpasswd [opt] [benutzer] [gruppe]

Optionen: -a ⇒ Fügt Benutzer einer Gruppe hinzu. -d ⇒ Löscht Benutzer aus einer Gruppe. -r ⇒ Löscht das Gruppenpasswort der Gruppe. -A [user,..] ⇒ Erstellt eine Liste von Adminstrativen Benutzern. -M [user,..] ⇒ Erstellt eine Liste von Gruppenmitgliedern. -h ⇒ Kurze Hilfe.

/etc/gshadow

Hier werden Gruppenname und Gruppenpasswort gespeichert.

…. cdrom:*::michael floppy:*::michael tape:*:: sudo:*:: audio:*::michael ….

Feld ⇒ Gruppenname. Feld ⇒ Gruppenpasswort(verschlüsselt). Feld ⇒ Gruppenadministratoren. Feld ⇒ Gruppenmitglieder.

Gruppenadministratoren dürfen, außer „root“, das Gruppenpasswort ändern, Benutzer der Gruppe hinzufügen und Benutzer aus der Gruppe entfernen. Gruppenadministrator(en) darf nur „root“ über „gpasswd -A [user,..]“ festlegen.

newgrp

Mit diesem Befehl kann man ohne sich ab und wieder anzumelden, während einer Sitzung als neue Gruppe anmelden. Wird „newgrp - [gruppe]“ angewendet, wird die Benutzerumgebung so neu gestartet als ob sich der Benutzer neu angemeldet hätte, ohne den Schalter [-] bleibt die aktuelle Benutzerumgebung und Arbeitsverzeichnisse unverändert, hierbei ist jedoch zu beachten, dass beim Wechsel in eine andere Gruppe eine neue Shell gestartet wird und ggf. gesetzte Variablen exportiert werden müssen.

Syntax:

- `newgrp [-] [gruppe]`

Wenn kein Gruppenname angegeben wird, wechselt der Benutzer in die Standardgruppe, die in der „/etc/passwd“ festgelegt ist. „newgrp“ wird versuchen die Gruppe der Gruppensammlung des Benutzers hinzuzufügen. Sollte der Benutzer nicht Mitglied der Gruppe und nicht „root“ sein, die Gruppe ein Passwort besitzt, wird nach einem Passwort gefragt und der Benutzer kann einer neuen Gruppe beitreten. Verweigert wird der Zugriff jedoch, sollte das Gruppenpasswort leer oder der Benutzer nicht Mitglied der Gruppe sein.

Befindet sich ein Eintrag zur Gruppe in der „/etc/gshadow“ wird die Mitgliederliste und Passwort

dieser Datei verwendet, ansonsten werden die Einträge der „/etc/group“ zum Einsatz.

Das Verhalten von „newgrp“ kann in der Datei „/etc/login.defs“ konfiguriert werden.

Tritt man einer neuen Gruppe bei hat dies Auswirkungen auf neu erstellte Dateien, da eine Datei nur einer Gruppe angehören kann. Ändert man die GID nicht, wird eine neue Datei mit der primären Gruppe aus der „/etc/passwd“ hinzugefügt. (Siehe auch) Dateirechte.

http://kreativgarten.bplaced.net/doku.php?id=benutzer_u._gruppen

Last update: **2019/05/27 16:45**