Name:	Lehrgang: Betriebssysteme	Datum:
Arbeitsblatt Nr.	Dateizugriffsrechte unter LINUX	Seite 1 von 2

In den Inodes von Dateien werden u.a. die Zugriffsrechte auf Dateien verwaltet. Die Rechtetabelle legt fest, was welcher Benutzer mit welchem Verzeichnis bzw. mit welcher Datei machen darf. Durch die Anmeldung (Login) mit Benutzername und zugehörigem Passwort am System authentifiziert sich der Benutzer und erwirbt dadurch die ihm vom Dateisystem zugestandenen Rechte an Verzeichnissen und Dateien.

Verzeichnisse

Recht	*Bedeutung*
r	Der Inhalt des Verzeichnisses kann eingesehen werden
w	Der Inhalt des Verzeichnisses kann verändert werden
x	Man kann in das Verzeichnis wechseln und auf die Verwaltungsinformationen der darin enthaltenen Dateien zugreifen

Verzeichnisrechte bei LINUX

In der Praxis treten folgende Rechtekombinationen bei den Verzeichnisrechten auf:

Recht	*Bedeutung*
keines	Gestattet keine Zugriffsmöglichkeit auf dieses Verzeichnis und seine Unterverzeichnisse
r und x	Erlaubt Informationen über das Verzeichnis und seinen Inhalt. Darin enthaltene Dateien können • gelesen • kopiert • gedruckt • ausgeführt werden. Es kann ein Link auf Dateien erstellt werden, sofern die Zugriffsrechte an der Datei bestehen.
Alle (rwx)	Alle verwaltungstechnischen Arbeiten an Dateien sind erlaubt

Gebräuchliche Verzeichnisrechte bei LINUX

Dateien

Recht	*Bedeutung*
r	Der Inhalt der Datei kann eingesehen werden
w	Der Inhalt der Datei kann verändert und die Datei dann unter ihrem bisherigen Namen abgespeichert werden
x	Die Datei kann ausgeführt werden (nur bei Programmdateien!)

Dateirechte bei LINUX

Befehl	*Rechte am Verzeichnis*	*Rechte an der Datei*
cat	rx	r
cat >	rwx	-
cp	rx	r
mv	rwx Zusätzlich rwx am Zielverzeichnis	-
ln	rx Zusätzlich rwx am Zielverzeichnis	-
rm	rwx	-
file	rx	r
„starten“	rx	x
„edit“	rx	rw

Dateioperationen und benötigte Rechte

```
root@andrea>ls -al /home
total 5
drwxr-xr-x   5 root     root         1024 Sep 30 07:51 .
drwxr-xr-x  19 root     root         1024 Jun  4 18:49 ..
drwxr-xr-x   9 andrea   users        1024 Jun  5 00:13 andrea
drwxr-xr-x   7 root     root         1024 Jun 28 22:18 inet
drwxr-xr-x   7 uhomm    users        1024 Jul  2 07:30 uhomm

root@andrea>ls -al /home/andrea
total 13
drwxr-xr-x   9 andrea   users        1024 Jun  5 00:13 .
drwxr-xr-x   5 root     root         1024 Sep 30 07:51 ..
-rw-r--r--   1 andrea   users        5742 Dec  8  1998 .Xdefaults
lrwxrwxrwx   1 andrea   users          10 Jun  4 23:45 .Xresources -> .Xdefaults
-rw-------   1 andrea   users         232 Jul  2 19:48 .bash_history
drwx------   2 andrea   users        1024 Jun  5 00:13 .cedit
-rw-------   1 andrea   users           0 May  8  1996 .dayplan.priv
drwx------   2 andrea   users        1024 Jun  4 18:48 .grok
drwxr-xr-x   2 andrea   users        1024 Jun  4 18:48 .hotjava
-rw-r--r--   1 andrea   users        5376 Aug 28  1996 private
```

Beispiele von Inhaltsverzeichnissen in langer Form

Besondere Rechte

s-Recht

Unter LINUX sind noch einige besondere Rechte vorhanden. So ist es z.B. bei bestimmten Programmen, die dem Superuser `root` gehören, möglich, daß diese trotzdem von jedem Benutzer ausführbar sind. Ein entsprechendes Programm ist z.B. dasjenige zur Änderung des Systempasswortes. Dieses Programm muß in der Passwortdatei Änderungen vornehmen können. Eine Arbeit, die normalerweise aufgrund der Rechte an der Passwortdatei (nicht am Passwort-Programm!) lediglich durch den Superuser `root` ausgeführt werden darf.

Damit nun aber jeder User sein Passwort ändern kann, existiert ein Programm `passwd`, das Änderungen in der Passwortdatei vornehmen darf. Hierzu schlüpft der User für die Zeit des Programmablaufs in die Rolle des jeweiligen Programmbesitzers; in diesem Falle `root`.

Erkennbar ist dieses Verhalten am **s-Recht** für den User anstelle des x-Rechtes

```
-rwsr-xr-x   1 root     root        32916 Dec 11  1998 passwd
```

s-Recht bei der Programmdatei passwd

t-Recht

Hin und wieder kann man auf Dateien stoßen, die anstelle des x-Rechtes für die anderen (das letzte Recht in der Tabelle) einen Eintrag t haben. Dieses „Sticky bit“ genannte Recht bezieht sich ebenfalls auf die Ausführung einer Programmdatei.

Die Besonderheit ist in diesem Fall, daß der Programmcode nach dem erstmaligen Start der Datei nicht mehr aus dem Arbeitsspeicher entfernt wird. Andere Benutzer sparen beim Start des entsprechenden Programms die Zeit, die normalerweise für das Laden in den Arbeitsspeicher erforderlich ist. Allerdings hat dies auch einen Haken: Werden zuviele Programme, die mit dem t-Recht versehen sind, gestartet, ist der Arbeitsspeicher u.U. schnell voll.

Beide Rechte, s-Recht und t-Recht, können nur durch den Superuser vergeben werden!

Befehle zur Rechtevergabe

Verzeichnis- und Dateirechte werden durch den Befehl `chmod` (**ch**ange **mod**us) verändert. Rechte lassen sich auf zwei Arten verändern:

`chmod` **werwiewas** *Datei(en)*

`chmod` **Oktalzahl** *Datei(en)*

```
chmod u=rx testdatei
chmod go+rx testdir
chmod 755 *
chmod 740 all*
```

Beispiele für chmod

wer		***wie***		***was***	
u	user	+	Recht wird zusätzlich vergeben	r	Read-Recht
g	group	-	Recht wird entzogen	w	Write-Recht
o	others	=	Rechte werden entsprechend ersetzt	x	Execute-Recht
a	all				

Rechtevergabe bei chmod

Die Rechte können ebenfalls in Form einer Oktalzahl angegeben werden. Jede Ziffer der Oktalzahl **ugo** beschreibt das Recht für die entsprechende Gruppe. Eine Oktalzahl wird durch drei Bit dargestellt. Jedes Bit steht für ein entsprechendes Recht. Durch Addition der entsprechenden Zahlenwerte entsteht die Oktalziffer und damit das Recht für die betreffende Gruppe. Soll kein Recht vergeben werden, wird für die Rechtekombination des Benutzerbereiches die Ziffer 0 verwendet.

u			***g***			***o***		
r	w	x	r	w	x	r	w	x
4	2	1	4	2	1	4	2	1
7			7			7		

Oktalzahl bei Rechtevergabe

Setzen des s-Rechtes und des t-Rechtes

Mit `chmod` **u+s** *Dateiname* bzw. `chmod` **g+s** *Dateiname* bzw. `chmod` **o+t** *Dateiname* oder durch Voranstellen der Oktalziffern 4 für u+s, 2 für g+s oder 1 für o+t werden durch den Superuser die betreffenden Rechte erteilt.

Beispiel: `chmod 4755 /bin/passwd`