Linux Security

Linux Dateiberechtigungen

Table des matières

1. Definitionen

Benutzerklassen

Das herkömmliche POSIX Rechtekonzept kennt drei Klassen von Benutzern für die Rechtevergabe im Dateisystem: Eigentümer (engl. owner), Gruppe (engl. group) und andere Benutzer oder den „Rest der Welt“ (engl. other).

Pro Benutzerklasse lassen sich jeweils die drei Berechtigungsbits (engl. permission bits) für Lesezugriff (r), für Schreibzugriff (w) und für Ausführbarkeit (x) vergeben.

2. Zugriffskontrollen für Ordnern (Verzeichnissen)

2.1. Zugriffsarten :

r read (lesen) : Den Inhalt auslesen

w write (schreiben) : Dateien und Unterverzeichnisse in dem Verzeichnis erstellen, umbenennen, löschen und deren Dateirechte verändern

x execute (ausführen) : in das Verzeichnis wechseln und dort Dateien oder Unterverzeichnisse erreichen. Ohne das Lesen-Recht darf der Verzeichnisinhalt jedoch nicht ausgelesen werden

2.2. Operationen

Verzeichnisinhalt auslesen (ls)
Verzeichnis löschen (rm -r)

Dateien eintragen (cp, mv, ln)

3. Zugriffskontrollen für Dateien

3.1. Zugriffsarten :

r read (lesen) : Lesen oder Kopieren des Dateininhalts

w write (schreiben) : Schreiben oder Andern von Dateiinhalten

x execute (ausführen) : Das Ausführen von Programmen, die in Dateien enthalten sind

3.2. Operationen

Lesen (cat, more, less, pg)
Schreiben (cat, >, ed)
Umbennenen (mv)
Prg starten

Tabelle 2 verdeutlicht, dass *w* und *x* für Verzeichnisse anders interpretiert werden als dies bei Dateien der Fall ist. Das Schreibrecht kann man wie folgt interpretieren: Die Liste der enthaltenen Dateien darf verändert oder ergänzt werden.
Ist das *x* gesetzt, dann gilt: Das Verzeichnis darf aktuelles Verzeichnis oder Bestandteil eines Pfadnamens werden. Zur Übersicht zeigt Abbildung 1 den Ablauf der Zugriffskontrolle eines Prozesses auf eine Datei.

Beispiel :

Eigner- und Zugriffsrechte sind zentrale Punkte des Systemsicherheit.

```
linsrv1:~ # umask
0022
```

umask Value Octal (xyz)	Default File Permissions	666 - xyz	Default Directory Permissions	777 - xyz
000	rw-rw-rw	666	rwxrwxrwx	777
002	rw-rw-r--	664	rwxrwxr-x	775
022	rw-r--r--	644	rwxr-xr-x	755
026	rw-r-----	640	rwxr-x--x	751
046	rw--w----	620	rwx-wx--x	731
062	rw----r--	604	rwx--xr-x	715
066	rw-------	600	rwx--x--x	711
222	r--r--r--	444	r-xr-xr-x	555
600	---rw-rw-	066	--xrwxrwx	177
666	---------	000	--x--x--x	111
777	---------	000	---------	000

4. Verzeichnisse

777 – 022 = 755 = r(4) + w(2) + x(1) | r(4) + ~~w(0)~~ + x(1) | r(4) + ~~w(0)~~ + x(1)

```
drwxr-xr-x  2 oracle oinstall     6 Sep 24 16:06 Test
```

- d Ein Verzeichnis
- Die Ausführberechtigungen sind da, so dass alle Benutzer sich den Inhalt des Verzeichnisses anzeigen lassen können.
- Der Eigner (Owner) kann lesen, schreiben, ausführen
- Die Gruppe kann nur lesen und ausführen
- Sonstige können nur lesen und ausführen

5. Dateiberechtigungen

666 – 022 = 644 = r(4) + w(2) + ~~x(0)~~ | r(4) + ~~w(2)~~ + ~~x(0)~~ | r(4) + ~~w(0)~~ + x(0)

```
-rw-r--r--  1 oracle oinstall     0 Sep 24 16:12 file
```

- Eine einfache Datei
- Oracle = Eigner, oinstall = Gruppe

  Der Eigner (Owner) kann lesen, schreiben aber nicht ausführen
  Die Gruppe kann nur lesen
  Sonstige können nur lesen

6. Eigner, Gruppe und Berechtigungen ändern

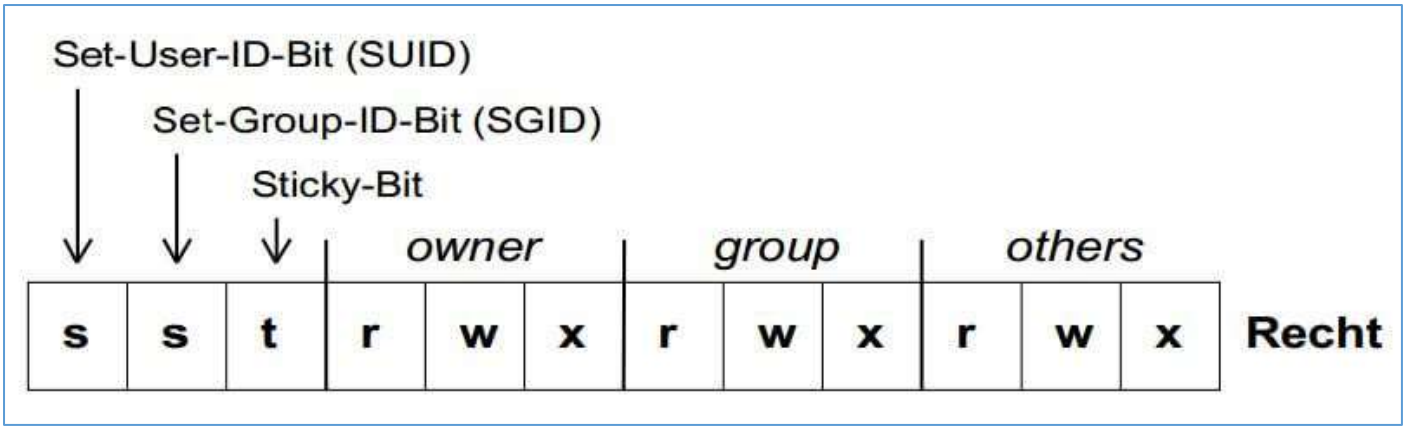
Mit dem Befehl chown ändern Sie die Eignerschaft einer Datei und mit chgrp die Gruppe.

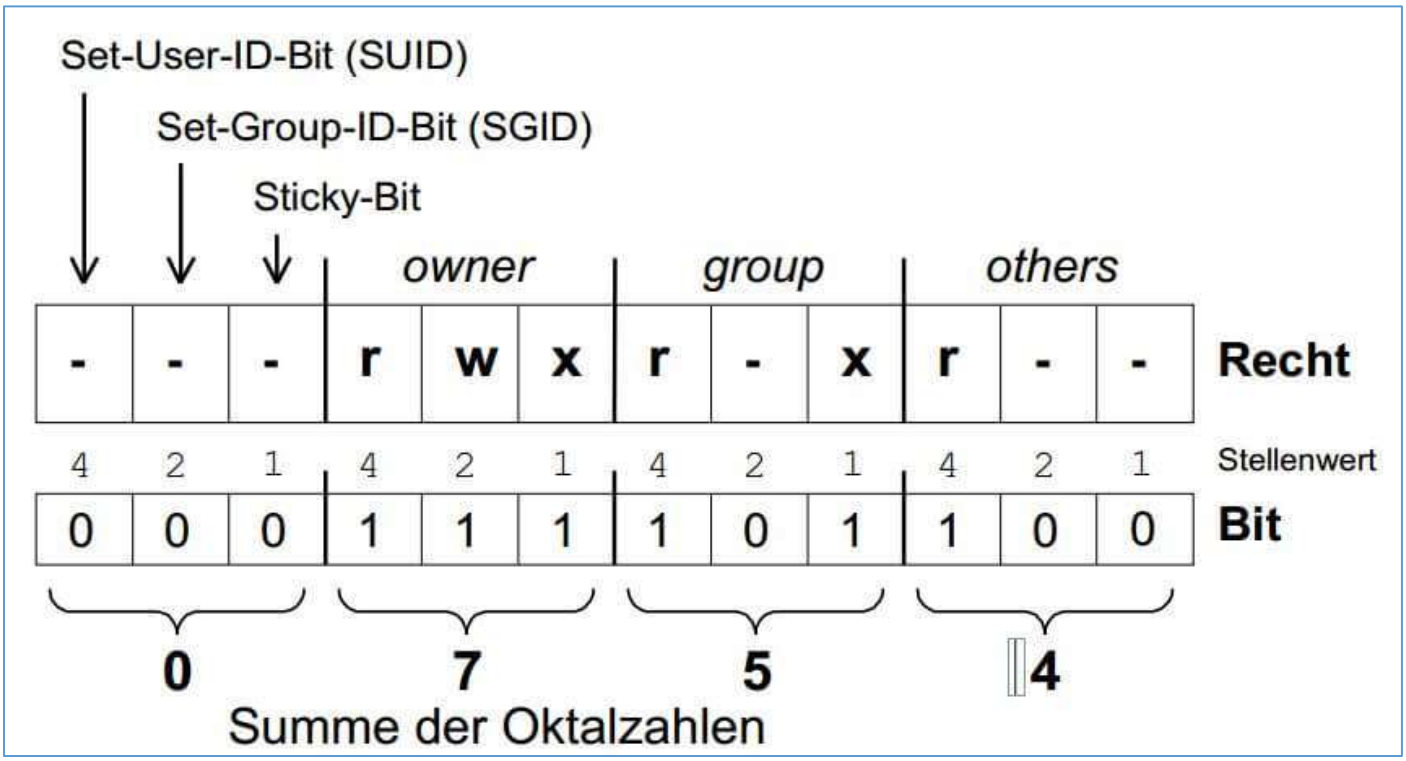
Unter Linux kann chown nur von root verwendet werden.

```
oracle@linsrv1:~/Test> chown yves file
chown: changing ownership of 'file': Operation not permitted
```

Jeder Benutzer kann seine Gruppe (oinstall) in eine andere Gruppe (dba) ändern, zu der er gehört.

```
oracle@linsrv1:~/Test> chgrp dba file
oracle@linsrv1:~/Test> ls -la
total 4
drwxr-xr-x  2 oracle oinstall   18 Sep 24 16:12 .
drwxr-xr-x 19 oracle oinstall 4096 Sep 24 16:06 ..
-rw-r--r--  1 oracle dba         0 Sep 24 16:12 file
```


Set-User-ID-Bit (SUID)
Set-Group-ID-Bit (SGID)
Sticky-Bit
owner
group
others
s s t r w x r w x r w x
Recht


Set-User-ID-Bit (SUID)
Set-Group-ID-Bit (SGID)
Sticky-Bit
owner
group
others
- - - r w x r - x r - -
Recht
4 2 1 4 2 1 4 2 1 4 2 1
Stellenwert
0 0 0 1 1 1 1 0 1 1 0 0
Bit
0 7 5 4
Summe der Oktalzahlen

7. Linux File Permissions Code

Linux File Permission Codes

Permissions	Binary	Octal	Description
`---`	000	0	No permissions
`--x`	001	1	Execute-only permission
`-w-`	010	2	Write-only permission
`-wx`	011	3	Write and execute permissions
`r--`	100	4	Read-only permission
`r-x`	101	5	Read and execute permissions
`rw-`	110	6	Read and write permissions
`rwx`	111	7	Read, write, and execute permissions

Unix Permission Mask	R	W	X	Decimal Representation
`---`	0	0	0	0
`--x`	0	0	1	1
`-w-`	0	1	0	2
`-wx`	0	1	1	3
`r--`	1	0	0	4
`r-x`	1	0	1	5
`rw-`	1	1	0	6
`rwx`	1	1	1	7