

Zugriffsrechte für Dateien und Ordner

Einleitung

In Linux herrscht ein striktes Berechtigungssystem. Wir können für jeden Ordner festlegen, wer welche Datei schreiben, lesen oder ausführen darf. Diese Konfiguration machen wir über das Terminal. Ansonsten können wir die Berechtigungen auch über die GUI setzen.

Zur Verwaltung der Berechtigungen verwenden wir den Befehl **chmod**. Mit diesen können wir Berechtigungen festlegen, verändern oder ganz entfernen.

Syntax von chmod

Wenn wir den Befehl **chmod** verwenden möchten, müssen wir erstmal die Syntax des Befehls nachvollziehen.

```
chmod [optionen] <maske> <datei>
```

Unter **maske** verstehen wir die Berechtigungsmaske. Die Zuteilung kann Symbolisch oder Numerisch umgesetzt werden.

Symbolische Zuteilung

Wenn wir die Berechtigungen über Symbole und Buchstaben ändern möchten, wird die Maske in 3 Teilbereiche aufgeteilt.

- Benutzerkategorie
- Operator
- Rechte

Die Rechte werden von der Benutzerkategorie immer durch einen Operator getrennt. Der Operator gibt an, ob die Rechte jeweils hinzugefügt, entfernt oder gesetzt werden.

Benutzerkategorie	
u	Besitzer
g	Gruppe
o	Andere
a	All (Besitzer, Gruppe, Andere)

Operator	
+	Rechte hinzufügen
-	Rechte entfernen
=	Rechte neu setzen

Rechte	
r	Lesen
w	Schreiben
x	Ausführen

Syntax Beispiele:

```
chmod a+rwx datei.txt
```

```
chmod +x script.sh
```

```
chmod u=rw,g=rw,o=r datei.txt
```

Numerische Zuteilung

Im Gegenteil zu der symbolischen Zuordnung können Berechtigungen auch über die numerische Zuteilung gesetzt werden. Dort wird eine dreistellige Zahl angegeben. Diese teilt mit, welche Berechtigungen, welche Ebene bekommt.

- **1. Zahl** = Besitzer
- **2. Zahl** = Gruppe
- **3. Zahl** = Andere

Die Rechte werden dann mit einer Nummer identifiziert. Die Zahl **751** gibt z.B. die Berechtigung für den Besitzer **Vollzugriff**, für die Gruppe **Lesen, Ausführen** und für andere auf **Nur Ausführen**.

Berechtigungen	
7	Vollzugriff
6	Lesen, Schreiben
5	Lesen, Ausführen
4	Nur Lesen
3	Schreiben, Ausführen
2	Nur Schreiben
1	Nur Ausführen
0	Keine Berechtigungen

Syntaxbeispiele:

```
chmod 777 datei.txt
```

```
chmod -R 700 /footer/topbar
```

Optionen

Um den Befehl jetzt noch intelligenter zu gestalten, gibt es die **Optionen**. Mit den Optionen können wir dem Befehl noch etwas mitgeben, wie er sich verhalten soll. Die Option geben wir mit einem **Bindestrich** vorne dran an, und wir schreiben dann die entsprechenden Buchstaben dahinter.

Optionen	
-c	hat die gleiche Funktion wie -v, gibt aber nur Rückmeldung wenn etwas geändert wird.
-f	Unterdrückt Fehlermeldungen
-R	Damit werden die Berechtigungen Rekursiv auf Unterverzeichnisse und Unterdateien geschrieben.
-v	Zeigt alles an was der Befehl gerade am System macht.

Revision #3

Created 28 April 2022 12:41:08 by Phillip U.

Updated 14 May 2024 07:57:47 by Phillip U.