

# Linux

- [Linux Befehle](#)
- [Linux](#)
- [Linux Befehlaufbau](#)

# Linux Befehle

## Rechtesystem unter Linux

Typ	User (u)	Group (g)	Other (o)
Die Vergaben gelten nur als Beispiel. Man kann z. B. für alle Typen die gleichen Rechte vergeben!	read = 4 write = 2	read = 4 write = 2	ex = 1

## Standardrechte (Nummerisch)

644	Wird verwendet bei Dateien
755	Wird verwendet bei Verzeichnissen
007	Nicht zu empfehlen!!!
777	Wird für Debugging verwendet

## Benutzerrechte erstellen

Befehl	Bedeutung
<code>chmod</code>	Dateirechte verändern
<code>chown</code>	Besitzer wechseln
<code>chgroup</code>	Gruppe wechseln
<code>w</code>	Schreibberechtigung
<code>r</code>	Leseberechtigung
<code>x</code>	Ausführungsberechtigung
<code>+</code>	Berechtigung erteilen
<code>-</code>	Berechtigung entziehen

## Weitere Linux Befehle

Befehl	Bedeutung
<code>ls</code>	Verzeichnis anzeigen
<code>cd</code>	Verzeichnis öffnen
<code>mv</code>	Datei oder Verzeichnis verschieben oder umbenennen
<code>cp</code>	Datei oder Verzeichnis in ein anderes Verzeichnis kopieren
<code>rm</code>	Datei löschen
<code>rmdir</code>	Verzeichnis löschen
<code>mkdir</code>	Verzeichnis erstellen
<code>grep</code>	Suche innerhalb einer Datei
<code>find</code>	Suche innerhalb eines Verzeichnis
<code>echo</code>	Gibt einen nachfolgenden Text aus
<code>wc</code>	Wortzähler

## Spezielle Zeichen

Befehl	Bedeutung
<code>&gt;</code>	Dieses Zeichen nennt man Pipe und überschreibt eine Ausgabe von z. B. <code>ls</code> in eine Datei
<code>&gt;&gt;</code>	Hier wird die Ausgabe in einer Datei angehängt
<code>&lt;&lt;ENDE</code>	In Kombination mit <code>grep</code> kann man hier Verschiedene Begriffe auflisten und spezifisch gesucht werden
<code>*</code>	Platzhalter für 0.1 bis unendlich Zeichen
<code>?</code>	Platzhalter für genau 1 Zeichen

!	Ein Ausrufezeichen in Klammern bedeutet, dass alle Zeichen ausser das definierte zulässig sind. (z. B. [! Aa])
[]	Innerhalb der Klammern werden Buchstaben, Zeichen und Zahlen definiert, um Bereiche zu definieren. (z. B. [a-z])
;	Das Semikolon wird verwendet, um mehrere Befehle hintereinander zu verketteten. (Nachteil: Auch wenn der erste Befehl fehlschlägt, wird der nächste trotzdem ausgeführt)
&&	Logisches Und: Wenn z. B. der erste Befehl der Verkettung fehlschlägt, wird der nächste nicht Automatisch ausgeführt

# Linux

*Im Remote Desktop Manager hat jeder Mitarbeitende eine Root-Berechtigung auf den Linux-Servern.*

# Linux Befehlaufbau

Befehl: *grep* [Optionen] Pfad [Datei]

<code>grep -v</code>	Alle Zeilen heraussuchen, die das angegebene Textmuster nicht enthalten.
<code>grep -w</code>	Nur Zeilen heraussuchen, in denen das Suchmuster als ganzes Wort enthalten ist.
<code>grep -n</code>	Die Zeilennummern, in denen der Text enthalten ist, ausgeben.
<code>grep -H</code>	Den Dateinamen aller Dateien ausgeben, die den angegebenen Text enthalten.