

Hard- und Soft-Links

ITS-Net-Lin

Sebastian Meisel

28. November 2024

1 1. Grundlagen: Hard- und Soft-Links

Links in Linux ermöglichen es, Dateien oder Verzeichnisse auf flexible Weise zu referenzieren. Es gibt zwei Haupttypen von Links: **Hard-Links** und **Soft-Links**. In dieser Einführung werden die Unterschiede und die Befehle zur Erstellung und Verwaltung von Links erläutert.

1.1 Hard-Links

- Ein Hard-Link ist ein direkter Verweis auf die gleiche Datei im Dateisystem.
- Alle Hard-Links einer Datei teilen sich denselben Inode¹.
- Änderungen an der Datei sind durch alle Hard-Links sichtbar.
- Einschränkungen: Keine Verknüpfung mit Verzeichnissen oder über Dateisystemgrenzen (Partitionen) hinweg.

1.2 Der Befehl 'ln' für Hard-Links

Der Befehl 'ln' wird verwendet, um einen Hard-Link zu erstellen.

```
ln <zieldatei> <hardlinkname>
```

Beispiel:

```
ln datei.txt hardlink.txt
```

Ergebnis:

- Beide Dateien (Original und Link) teilen denselben Inode.
- Änderungen an einer Datei wirken sich auf die andere aus.

2 Soft-Links (Symbolische Links)

- Ein Soft-Link ist eine Verknüpfung, die auf den Pfad einer Datei oder eines Verzeichnisses zeigt.
- Soft-Links haben einen eigenen Inode und speichern den Zielpfad als Inhalt.
- Kann über Dateisystemgrenzen hinweg verwendet werden.
- Funktioniert nicht, wenn die Zieldatei gelöscht wird (Broken Link).

¹Eine Inode ist ein Datensatz im Dateisystem, der alle wichtigen Informationen über eine Datei enthält, wie zum Beispiel, wo die Datei auf der Festplatte gespeichert ist. Sie beschreibt also nicht die Datei selbst, sondern ihre Metadaten und den Speicherort.

2.1 Der Befehl 'ln -s' für Soft-Links

Mit dem Befehl 'ln -s' können Sie einen Soft-Link erstellen.

```
ln -s <zielpfad> <softlinkname>
```

Beispiel:

```
ln -s /home/user/datei.txt softlink.txt
```

Ergebnis:

- 'softlink.txt' verweist auf '/home/user/datei.txt'.
- Wenn 'datei.txt' gelöscht wird, zeigt der Soft-Link ins Leere.

3 Vergleich: Hard-Links vs. Soft-Links

Eigenschaft	Hard-Link	Soft-Link
Inode geteilt?	Ja	Nein
Funktioniert bei Zielfeilelöschung?	Ja	Nein (Broken Link)
Über Dateisystemgrenzen?	Nein	Ja
Verzeichnisunterstützung?	Nein	Ja