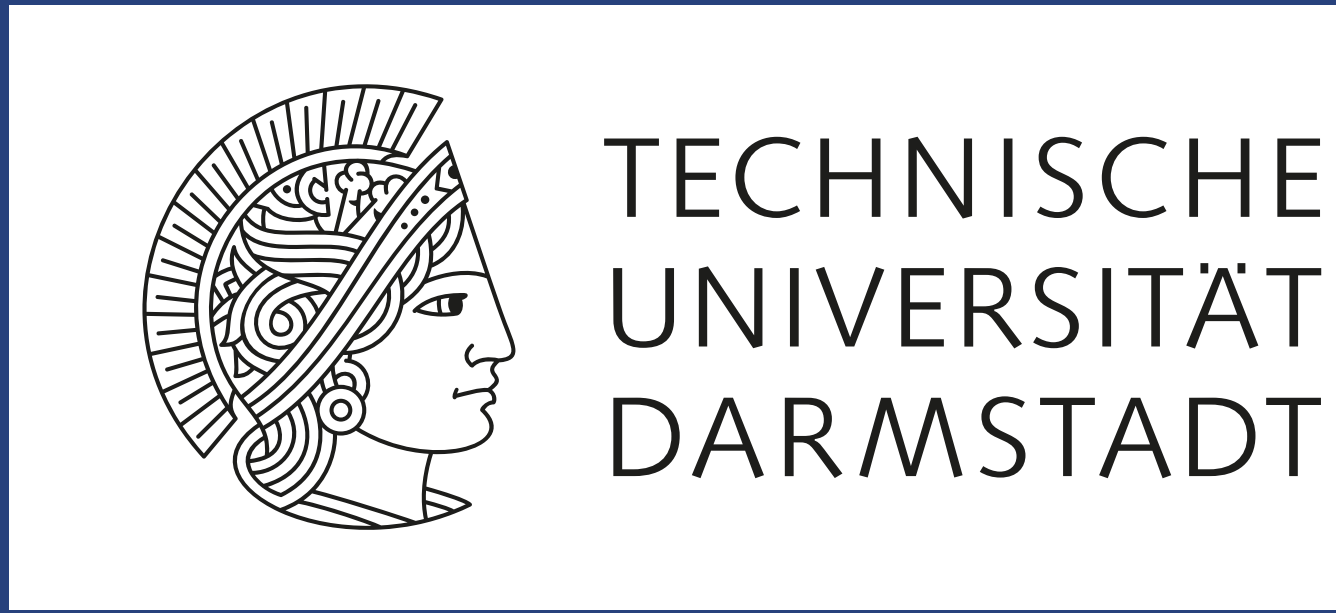


# Linux Cheat Sheet



Tim Carlo Pöpke, Florian Piana

## Zusammenfassung

Dieses Dokument soll Ihnen das Arbeiten auf dem bereitgestellten Linux Server vereinfachen, indem es Ihnen einen Einblick auf einige besonders wichtige Befehle gibt.

Die Befehle sind nicht klausurrelevant. Wir empfehlen Ihnen die Befehle trotzdem zu verinnerlichen, da Sie Ihre Testate auf dem Server testen und ausführen können müssen.

Auf einigen Tastaturen ist die Taste `Strg` als `Ctrl` aufgeführt. Falls dies bei Ihrer Tastatur der Fall sein sollte, können Sie die Taste `Ctrl` nutzen, wenn die Taste `Strg` gefordert wird.

### Basics

- Befehle werden mit `Enter`, `↵` bestätigt.
- Mit `↑`, `↓` kann man sich den Verlauf der bereits abgeschickten Befehle anzeigen lassen.
- Eine ausführbare Datei, lässt sich mit dem Befehl `./DATEINAME` in der Konsole aufrufen.

### Der Texteditor nano

Um Code zu schreiben, können Sie den Texteditor *nano* verwenden.

- `nano DATEINAME` Erstellt oder öffnet eine Datei mit dem angegebenen Dateinamen in *nano*.
- `←`, `→`, `↑`, `↓` Bewegen den Cursor.
- `Strg + S` Speichert die Datei.
- `Strg + X` Schließt *nano*. (Denken Sie daran vorher zu speichern!)
- `Strg + C` Bricht die momentane Operation ab (kann hilfreich sein, wenn Sie versehentlich in einem Untermenü von *nanolanden*).

### Dateisystem

- `cd DATEIPFAD` Geht im Dateisystem zu dem angegeben *Dateipfad*.
- `cd ..` Geht im Dateisystem eine Ebene höher.
- `ls` Zeigt den aktuellen Ordnerinhalt an.
- `pwd` Zeigt an in welchem Dateipfad Sie sich aktuell befinden.
- `mkdir ORDNERNAME` Erstellt einen Ordner mit dem angegebenen Namen.
- `rm DATEINAME` Löscht die angegebene Datei.
- `rm -r ORDNERNAME` Entfernt den angegebenen Ordner sowie dessen Inhalt.
- `mv ALTERPFAD NEUERPFAD` Verschiebt ALTERPFAD in NEUERPFAD.
- `cp QUELLPFAD ZIELPFAD` Kopiert QUELLPFAD in ZIELPFAD. Hei bei können Dateien oder Ordner angegeben werden

### Vom Source-Code zur ausführbaren Datei

Um den geschriebenen Assembler-Code auf dem ARM-Server des ISPs ausführen zu können, müssen die folgenden beiden Befehle nacheinander ausgeführt werden.

- `arm-linux-gnueabi-hf-gcc -o OUTPUTEXECUTABLE INPUTDATEI` Startet eine Vielzahl an Tools hintereinander, um den gesamten Compile-Prozess (Präprozessieren, Compilieren, Assemblieren und Linken) auszuführen. `INPUTDATEI` kann Assembly-Code, Objektdatei oder C-Code sein. `OUTPUTEXECUTABLE` ist der Name, den die daraus entstehende, ausführbare Datei haben soll.
- `arm-linux-gnueabi-hf-as -o OUTPUTOBJECTFILE INPUTASSEMBLYCODE` assembliert den in `INPUTASSEMBLYCODE` angegebenen Assemblercode in eine Objektdatei mit dem Namen `OUTPUTOBJECTFILE`.
- `objdump -d OBJEKTDAT EI` Disassembliert die abgegebene Objektdatei und gibt viele darin enthaltene Informationen aus.