

1.1

5 gängige Linux-Distributionen und ihre Besonderheiten:

1. Ubuntu:

Ubuntu ist eine benutzerfreundliche Linux-Distribution, die auf Debian basiert. Sie wird oft von Anfängern verwendet und bietet eine breite Palette von Anwendungen und Software im Software Center. Die Desktop-Version wird alle sechs Monate aktualisiert, während die LTS-Versionen (Long-Term Support) für fünf Jahre unterstützt werden.

2. Fedora:

Fedora ist eine Distribution, die von der Fedora-Community entwickelt wird und als Vorläufer für Red Hat Enterprise Linux dient. Sie ist bekannt für ihren Fokus auf aktuelle Software und Technologien. Fedora verwendet den DNF-Paketmanager und hat relativ kurze Support-Zeiträume.

3. CentOS:

CentOS war eine Linux-Distribution, die auf Red Hat Enterprise Linux (RHEL) basierte und eine kostenfreie Alternative zu RHEL darstellte. Beachte jedoch, dass CentOS im Dezember 2020 durch CentOS Stream ersetzt wurde, was zu Veränderungen in der Community geführt hat.

4. Debian:

Debian ist eine stabile und weit verbreitete Linux-Distribution, die für ihre robuste und zuverlässige Natur bekannt ist. Sie verwendet den APT-Paketmanager und ist die Basis vieler anderer Distributionen, darunter Ubuntu.

5. Arch Linux:

Arch Linux ist eine minimalistische und rollende Distribution. Sie erfordert ein gewisses Maß an Linux-Kenntnissen, da sie ein DIY-Ansatz ist und es dem Benutzer ermöglicht, das System nach eigenen Vorlieben anzupassen. Arch Linux verwendet das Pacman-Paketmanagement und ist bekannt für seine Aktualität.

1.2

Multitasking in Bezug auf Betriebssysteme bedeutet, dass das Betriebssystem die Fähigkeit besitzt, mehrere Aufgaben oder Prozesse gleichzeitig auszuführen.

1.3

Der Unterschied zwischen einem Betriebssystem (BS) und einem Kernel besteht darin, dass das Betriebssystem ein umfassendes Softwarepaket ist, das den gesamten Computerbetrieb steuert und verwaltet.

Der Kernel hingegen ist der zentrale Bestandteil des Betriebssystems, der direkt mit der Hardware des Computers kommuniziert und Aufgaben wie die Verwaltung von Prozessen, Speicher und Hardware-Ressourcen übernimmt.

1.4

CLI (Command Line Interface) ist eine textbasierte Benutzerschnittstelle, bei der der Benutzer Befehle in einer Textzeile eingibt, um mit dem Betriebssystem zu interagieren.

CLI bietet präzise Kontrolle und Automatisierungsmöglichkeiten, ist jedoch oft komplexer und erfordert mehr technisches Wissen.

GUI (Graphical User Interface) hingegen bietet eine grafische Darstellung von Anwendungen und Aufgaben, die durch Mausinteraktionen und visuelle Elemente gesteuert werden.

GUI ist benutzerfreundlicher und intuitiver, eignet sich jedoch möglicherweise nicht für fortgeschrittene Aufgaben oder Skripting.

1.5

Eine Shell ist eine textbasierte Benutzerschnittstelle, die es dem Benutzer ermöglicht, Befehle an das Betriebssystem zu senden. Sie fungiert als Vermittler zwischen dem Benutzer und dem Kernel. Zwei gängige Shells in Linux sind:

- Bash (Bourne Again Shell): Bash ist eine weit verbreitete Standard-Shell in den meisten Linux-Distributionen.

- Zsh (Z Shell): Zsh ist eine erweiterte Shell, die auf Bash aufbaut und zusätzliche Funktionen wie intelligente Vervollständigung und erweiterte Konfigurationsmöglichkeiten bietet.

2.1

"Open Source" bezieht sich auf Software, deren Quellcode öffentlich verfügbar ist und von der Gemeinschaft überprüft, geändert und weiterentwickelt werden kann.

- Freie Verfügbarkeit: Der Quellcode der Software ist für jeden zugänglich und kann ohne Beschränkungen eingesehen werden.
- Freie Veränderbarkeit: Benutzer können den Quellcode ändern und an ihre Bedürfnisse anpassen.
- Gemeinschaftliche Zusammenarbeit: Die Softwareentwicklung basiert oft auf Zusammenarbeit und Mitwirkung einer breiten Gemeinschaft von Entwicklern und Nutzern.
- Transparenz: Die Entwicklung ist transparent, und Fehler oder Schwachstellen können von der Gemeinschaft identifiziert und behoben werden.

2.2

Vorteile von Open Source-Software:

1. Kostenfreiheit: Open Source-Software ist in der Regel kostenlos verfügbar.
2. Anpassbarkeit und Flexibilität: Da der Quellcode geändert werden kann, können Benutzer die Software an ihre speziellen Anforderungen anpassen.
3. Sicherheit und Transparenz: Der offene Quellcode ermöglicht es der Gemeinschaft, Sicherheitslücken zu identifizieren und zu beheben.

Nachteile von Open Source-Software:

1. Mangel an Support: Open Source-Software wird oft von Communities entwickelt, was bedeutet, dass der kommerzielle Support begrenzter sein kann als bei proprietärer Software.
2. Kompatibilität und Interoperabilität: Einige Open Source-Software kann mit proprietären Systemen und Dateiformaten weniger kompatibel sein.
3. Lernkurve: Die Anpassung von Open Source-Software erfordert oft technisches Wissen und eine gewisse Einarbeitungszeit.

```
losem@LTV: ~/LinuxPraxis Ke X + v
losem@LTV:~/octave$ cd
losem@LTV:~$ mkdir "LinuxPraxis Kevin"
losem@LTV:~$ ls
'LinuxPraxis Kevin' Techstarter octave
losem@LTV:~$ cd LinuxPraxis\ Kevin/
losem@LTV:~/LinuxPraxis Kevin$ touch kevin1.txt kevin2.txt kevin3.txt
losem@LTV:~/LinuxPraxis Kevin$ ls
kevin1.txt kevin2.txt kevin3.txt
losem@LTV:~/LinuxPraxis Kevin$ mkdir Unterordner
losem@LTV:~/LinuxPraxis Kevin$ mv kevin2.txt Unterordner/
losem@LTV:~/LinuxPraxis Kevin$ rm kevin1.txt
losem@LTV:~/LinuxPraxis Kevin$ echo "Hallo Techstarter!" >> kevin3.txt
losem@LTV:~/LinuxPraxis Kevin$ cat kevin3.txt
Hallo Techstarter!
losem@LTV:~/LinuxPraxis Kevin$ cp kevin3.txt kopie.txt
losem@LTV:~/LinuxPraxis Kevin$ |
```