

Linux

Advanced

Labor-Umgebung

Für diese Schulung sind entweder **lokale VMs** oder ein **Cloud-Labor** vorgesehen.

Hostname IP (Vagrant) IP (Cloud) Rolle

node1	192.168.56.20	TBA	AlmaLinux-Host
node2	192.168.56.30	TBA	Ubuntu-Host

Für die Benutzung wurde der User-Account `user` angelegt, das dazugehörige **Passwort** lautet `test123`. Mittels `sudo` können auch privilegierte Kommandos ausgeführt werden.

In der Vagrant-Umgebung erfolgt der Login wie folgt:

```
$ vagrant ssh [NAME]
$ sudo su - user
```

Für den Login in der Cloud-Umgebung sind die folgende Kommandos notwendig:

```
$ ssh user@<ip-adresse>
```

Vagrant

Das Labor kann später auch zu Lernzwecken auf dem **eigenen Rechner** aufgesetzt werden. Hierfür werden benötigt:

- [HashiCorp Vagrant \(https://www.vagrantup.com\)](https://www.vagrantup.com)
- [Oracle VirtualBox \(https://www.virtualbox.org\)](https://www.virtualbox.org)
- 16 GB Arbeitsspeicher
- mindestens 40 GB Festplattenspeicher

Der Schulung liegt ein Ordner bei, der die entsprechenden Konfigurationen enthält. Mithilfe von Vagrant können die drei VMs vollautomatisiert erstellt und konfiguriert werden.

Zur Bereitstellung eine Kommandozeile öffnen und in den **Schulungsordner** wechseln:

```
$ cd Vagrant
$ vagrant up
...
PLAY RECAP
*****
ubuntu          : ok=16   changed=0    unreachable=0    failed=0
skipped=0       rescued=0   ignored=0
almalinux       : ok=16   changed=0    unreachable=0    failed=0
skipped=0       rescued=0   ignored=0
```

Die Bereitstellung kann bis zu **30 Minuten dauern**, es wird ein Rocky Linux 8-Template heruntergeladen.

Kapitel-Zusammenfassungen

Dienste und weitere Ressourcen

- **Systemd** ist als Init-System der erste Prozess während des Boots
 - verwaltet u.a. Dienste und Sitzungen
 - aggressive **Parallelisierung**
- bietet zahlreiche Zusatzfunktionen, wie z.B. **Logging**
- Verwaltung über das `systemctl`-Kommando
- Neben Diensten werden auch verwaltet
 - Mountpoints
 - Veränderungen in Pfaden
 - Regelmäßige auszuführende Units
- Cronjobs waren früher der einzige Weg regelmäßig Aufgaben auszuführen
 - moderne Systeme nutzen sowohl **Vixie-Cron** als auch `anacron`
- Durch reine Textkonfigurationen werden Aufgaben geplant
- neben systemweiten Konfiguration können auch User eigene Jobs einplanen

Logging

- Anwendungen schreiben i.d.R. **Protokolldateien**
 - diese können im **Dateisystem** oder im zentralen **Journal** stehen
- `rsyslog` implementiert das `syslog`-Protokoll
 - es kann Nachrichten filtern und auf Dateien verteilen
 - auch eine Weiterleitung an externe Systeme ist möglich

Storage

- Linux unterstützt zahlreiche verschiedene Storage-Protokolle und -Medien
- **skaliert** von Kleinstrechnern über Notebooks und Server bis hin zu Mainframes
- Der IO-Scheduler `blk-mq` wurde schon mit **über 15. Mio.** IOPS getestet
- zahlreiche **Dateisysteme** werden zur Installation und Benutzung unterstützt
- Mit **LVM** können Partitionen dynamisch verändert werden
- Durch den Einsatz von **LUKS** können Speicher verschlüsselt werden
- Mehrere Festplatten können im **RAID**-Verbund genutzt werden, um Durchsatz, Kapazität oder Ausfallsicherheit zu erhöhen

Systemüberwachung

- Linux bietet zahlreiche Tools zur Systemüberwachung, u.a.:
 - Allgemein: `top`, `htop`, `nmon`, `sar`

- Prozesse: ps, top, htop
- Storage: iotop, iostat
- Speicher: free, top, htop, vmstat, smem, mpstat, swapon
- Netzwerk: netstat, ss, iptraf
- Linux unterscheidet zwei verschiedene Speicherarten:
 - **Physischer Speicher** = verbauter RAM mit verbindlichen Speicheradressen
 - **Virtueller Speicher** = vom Betriebssystem abstrahierte Adressierung
- Linux nutzt ungenutzten Speicher als Dateisystem-Cache

Gängige Services

- Linux bietet eine breite Fülle an **Diensten**
 - u.a Web-/Mail-/Proxy-Server, Datenbanken, Monitoring und VPN
- **Apache** ist einer bekanntesten Webserver
 - durch modulares Design erweiterbar, z.B um **Skriptsprachen**
 - VirtualHosts erlauben das parallele Hosten mehrere Webseiten
- **MariaDB** ist ein relationales Datenbankmanagementsystem
 - weit verbreitet, wird häufig für Web-Anwendungen genutzt
- **Samba** ist eine Suite für Windows-Interoperabilität
 - bietet u.a Datenaustausch- und Domain Controller-Dienste
- **NFS** erlaubt Betriebssystem-unabhängige Dateiübertragungen
- Zahlreiche Server-Komponenten unterstützen beim Versenden und Zustellen von **Mails**