





# Red Hat System Administration I



**Red Hat Enterprise Linux 8.2 RH124**  
**Red Hat System Administration I**  
**Ausgabe 120200928**  
**Veröffentlicht 20200928**

Autoren: Fiona Allen, Victor Costea, Snehangshu Karmakar, Marc Kesler,  
Ed Parenti, Saumik Paul, Hervé Quatremain, Dallas Spohn  
Editor: Steven Bonneville, Ralph Rodriguez, David Sacco, Nicole Muller, Heather  
Charles, David O'Brien, Seth Kenlon

Copyright © 2020 Red Hat, Inc.

The contents of this course and all its modules and related materials, including handouts to audience members, are  
Copyright © 2020 Red Hat, Inc.

No part of this publication may be stored in a retrieval system, transmitted or reproduced in any way, including, but  
not limited to, photocopy, photograph, magnetic, electronic or other record, without the prior written permission of  
Red Hat, Inc.

This instructional program, including all material provided herein, is supplied without any guarantees from Red Hat,  
Inc. Red Hat, Inc. assumes no liability for damages or legal action arising from the use or misuse of contents or details  
contained herein.

If you believe Red Hat training materials are being used, copied, or otherwise improperly distributed, please send  
email to [training@redhat.com](mailto:training@redhat.com) or phone toll-free (USA) +1 (866) 626-2994 or +1 (919) 754-3700.

Red Hat, Red Hat Enterprise Linux, the Red Hat logo, JBoss, Hibernate, Fedora, the Infinity logo, and RHCE are  
trademarks of Red Hat, Inc., registered in the United States and other countries.

Linux® is the registered trademark of Linus Torvalds in the United States and other countries.

Java® is a registered trademark of Oracle and/or its affiliates.

XFS® is a registered trademark of Silicon Graphics International Corp. or its subsidiaries in the United States and/or  
other countries.

The OpenStack® word mark and the Square O Design, together or apart, are trademarks or registered trademarks  
of OpenStack Foundation in the United States and other countries, and are used with the OpenStack Foundation's  
permission. Red Hat, Inc. is not affiliated with, endorsed by, or sponsored by the OpenStack Foundation or the  
OpenStack community.

All other trademarks are the property of their respective owners.

Mitwirkende: Artur Glogowski, Latha Murthy, Samik Sanyal, Chetan Tiwary, Achyut Madhusudan,  
Rudolf Kastl, Rob Locke, Michael Phillips

<b>Dokumentkonventionen</b>	<b>ix</b>
<b>Einführung</b>	<b>xi</b>
Red Hat System Administration I .....	xi
Informationen zur Kursumgebung .....	xii
Landessprachliche Anpassung .....	xvi
<b>1. Erste Schritte mit Red Hat Enterprise Linux</b>	<b>1</b>
Was ist Linux? .....	2
Quiz: Erste Schritte mit Red Hat Enterprise Linux .....	9
Zusammenfassung .....	11
<b>2. Zugreifen auf die Befehlszeile</b>	<b>13</b>
Zugreifen auf die Befehlszeile .....	14
Quiz: Zugreifen auf die Befehlszeile .....	20
Zugreifen auf die Befehlszeile über den Desktop .....	24
Angeleitete Übung: Zugreifen auf die Befehlszeile über den Desktop .....	30
Ausführen von Befehlen an der Bash-Shell .....	32
Quiz: Ausführen von Befehlen an der Bash-Shell .....	39
Praktische Übung: Zugreifen auf die Befehlszeile .....	43
Zusammenfassung .....	49
<b>3. Verwalten von Dateien über die Befehlszeile</b>	<b>51</b>
Beschreiben der Hierarchiekonzepte des Linux-Dateisystems .....	52
Quiz: Beschreiben der Hierarchiekonzepte des Linux-Dateisystems .....	55
Angeben von Dateien nach Name .....	59
Quiz: Angeben von Dateien nach Name .....	65
Verwalten von Dateien mit Befehlszeilentools .....	69
Angeleitete Übung: Verwalten von Dateien mit Befehlszeilentools .....	75
Herstellen von Verknüpfungen zwischen Dateien .....	80
Angeleitete Übung: Herstellen von Verknüpfungen zwischen Dateien .....	84
Abgleichen von Dateinamen mit Shell-Erweiterungen .....	86
Quiz: Abgleichen von Dateinamen mit Shell-Erweiterungen .....	91
Praktische Übung: Verwalten von Dateien über die Befehlszeile .....	95
Zusammenfassung .....	105
<b>4. Abrufen von Hilfe in Red Hat Enterprise Linux</b>	<b>107</b>
Lesen von Handbuchseiten .....	108
Angeleitete Übung: Lesen von Handbuchseiten .....	112
Lesen der Info-Dokumentation .....	116
Angeleitete Übung: Lesen der Info-Dokumentation .....	120
Praktische Übung: Abrufen von Hilfe in Red Hat Enterprise Linux .....	123
Zusammenfassung .....	130
<b>5. Erstellen, Anzeigen und Bearbeiten von Textdateien</b>	<b>131</b>
Umleiten von Ausgaben an eine Datei oder ein Programm .....	132
Quiz: Umleiten von Ausgaben an eine Datei oder ein Programm .....	138
Bearbeiten von Textdateien an der Shell-Eingabeaufforderung .....	142
Angeleitete Übung: Bearbeiten von Textdateien an der Shell-Eingabeaufforderung .....	146
Ändern der Shell-Umgebung .....	148
Angeleitete Übung: Ändern der Shell-Umgebung .....	154
Praktische Übung: Erstellen, Anzeigen und Bearbeiten von Textdateien .....	157
Zusammenfassung .....	165
<b>6. Verwalten lokaler Benutzer und Gruppen</b>	<b>167</b>
Beschreiben von Benutzer- und Gruppenkonzepten .....	168
Quiz: Beschreiben von Benutzer- und Gruppenkonzepten .....	172
Zugriff als Superuser .....	176

Angeleitete Übung: Zugriff als Superuser .....	182
Verwalten lokaler Benutzerkonten .....	187
Angeleitete Übung: Verwalten lokaler Benutzerkonten .....	191
Verwalten lokaler Gruppenkonten .....	194
Angeleitete Übung: Verwalten lokaler Gruppenkonten .....	197
Verwalten von Benutzerpasswörtern .....	200
Angeleitete Übung: Verwalten von Benutzerpasswörtern .....	204
Praktische Übung: Verwalten lokaler Benutzer und Gruppen .....	208
Zusammenfassung .....	213
<b>7. Steuern des Zugriffs auf Dateien</b>	<b>215</b>
Interpretieren der Linux-Dateisystemberechtigungen .....	216
Quiz: Interpretieren der Linux-Dateisystemberechtigungen .....	221
Verwalten von Dateisystemberechtigungen über die Befehlszeile .....	225
Angeleitete Übung: Verwalten von Dateisystemberechtigungen über die Befehlszeile .....	229
Verwalten von Standardberechtigungen und Dateizugriff .....	233
Angeleitete Übung: Verwalten von Standardberechtigungen und Dateizugriff .....	238
Praktische Übung: Steuern des Zugriffs auf Dateien .....	243
Zusammenfassung .....	250
<b>8. Überwachen und Verwalten von Linux-Prozessen</b>	<b>251</b>
Auflisten von Prozessen .....	252
Quiz: Auflisten von Prozessen .....	258
Steuern von Jobs .....	260
Angeleitete Übung: Steuern von Jobs .....	263
Beenden von Prozessen .....	269
Angeleitete Übung: Beenden von Prozessen .....	276
Überwachen der Prozessaktivität .....	281
Angeleitete Übung: Überwachen der Prozessaktivität .....	285
Praktische Übung: Überwachen und Verwalten von Linux-Prozessen .....	290
Zusammenfassung .....	301
<b>9. Steuern von Services und Daemons</b>	<b>303</b>
Identifizieren automatisch gestarteter Systemprozesse .....	304
Angeleitete Übung: Identifizieren automatisch gestarteter Systemprozesse .....	310
Kontrollieren der Systemdienste .....	314
Angeleitete Übung: Kontrollieren der Systemdienste .....	319
Praktische Übung: Steuern von Diensten und Daemons .....	323
Zusammenfassung .....	327
<b>10. Konfigurieren und Sichern von SSH</b>	<b>329</b>
Zugreifen auf die Remote-Befehlszeile mit SSH .....	330
Angeleitete Übung: Zugreifen auf die Remote-Befehlszeile .....	334
Konfigurieren der schlüsselbasierten SSH-Authentifizierung .....	338
Angeleitete Übung: Konfigurieren der schlüsselbasierten SSH-Authentifizierung .....	343
Anpassen der OpenSSH-Servicekonfiguration .....	349
Angeleitete Übung: Anpassen der OpenSSH-Servicekonfiguration .....	351
Praktische Übung: Konfigurieren und Sichern von SSH .....	357
Zusammenfassung .....	364
<b>11. Analysieren und Speichern von Protokollen</b>	<b>365</b>
Beschreiben der Systemprotokollarchitektur .....	366
Quiz: Beschreiben der Systemprotokollarchitektur .....	368
Überprüfen von syslog-Dateien .....	372
Angeleitete Übung: Überprüfen von syslog-Dateien .....	376
Überprüfen von Systemjournal-Einträgen .....	379
Angeleitete Übung: Überprüfen von Systemjournal-Einträgen .....	385

Das Systemjournal bewahren .....	389
Angeleitete Übung: Das Systemjournal bewahren .....	392
Verwalten der genauen Uhrzeit .....	395
Angeleitete Übung: Verwalten der genauen Uhrzeit .....	399
Praktische Übung: Analysieren und Speichern von Protokollen .....	404
Zusammenfassung .....	410
<b>12. Netzwerkmanagement</b>	<b>411</b>
Beschreiben von Netzwerkkonzepten .....	412
Quiz: Beschreiben von Netzwerkkonzepten .....	425
Validieren der Netzwerkkonfiguration .....	429
Angeleitete Übung: Validieren der Netzwerkkonfiguration .....	435
Konfigurieren von Netzwerken über die Befehlszeile .....	438
Angeleitete Übung: Konfigurieren von Netzwerken über die Befehlszeile .....	444
Bearbeiten der Netzwerkconfigurationsdateien .....	450
Angeleitete Übung: Bearbeiten der Netzwerkconfigurationsdateien .....	454
Konfigurieren von Hostnamen und Namensauflösung .....	459
Angeleitete Übung: Konfigurieren von Hostnamen und Namensauflösung .....	462
Praktische Übung: Netzwerkmanagement .....	466
Zusammenfassung .....	472
<b>13. Archivieren und Übertragen von Dateien</b>	<b>473</b>
Verwalten von komprimierten tar-Archiven .....	474
Angeleitete Übung: Verwalten von komprimierten tar-Archiven .....	481
Sicheres Übertragen von Dateien zwischen Systemen .....	484
Angeleitete Übung: Sicheres Übertragen von Dateien zwischen Systemen .....	486
Sicheres Synchronisieren von Dateien zwischen Systemen .....	489
Angeleitete Übung: Sicheres Synchronisieren von Dateien zwischen Systemen .....	493
Praktische Übung: Archivieren und Übertragen von Dateien .....	495
Zusammenfassung .....	501
<b>14. Installieren und Aktualisieren von Softwarepaketen</b>	<b>503</b>
Registrieren von Systemen für den Red Hat Support .....	505
Quiz: Registrieren von Systemen für den Red Hat Support .....	509
Erläutern und Untersuchen von RPM-Softwarepaketen .....	511
Angeleitete Übung: Erläutern und Untersuchen von RPM-Softwarepaketen .....	517
Installieren und Aktualisieren von Softwarepaketen mit YUM .....	520
Angeleitete Übung: Installieren und Aktualisieren von Softwarepaketen mit YUM .....	527
Aktivieren von YUM-Software-Repositorys .....	533
Angeleitete Übung: Aktivieren von YUM-Software-Repositorys .....	537
Verwalten von Paketmodul-Streams .....	541
Angeleitete Übung: Verwalten von Paketmodul-Streams .....	548
Praktische Übung: Installieren und Aktualisieren von Softwarepaketen .....	553
Zusammenfassung .....	560
<b>15. Zugriff auf Linux-Dateisysteme</b>	<b>561</b>
Identifizieren von Dateisystemen und Geräten .....	562
Quiz: Identifizieren von Dateisystemen und Geräten .....	566
Mounten und Unmounten von Dateisystemen .....	568
Angeleitete Übung: Mounten und Unmounten von Dateisystemen .....	572
Suchen von Dateien im System .....	575
Angeleitete Übung: Suchen von Dateien im System .....	582
Praktische Übung: Zugriff auf Linux-Dateisysteme .....	585
Zusammenfassung .....	590
<b>16. Analysieren von Servern und Erhalten von Unterstützung</b>	<b>591</b>
Analysieren und Verwalten von Remote-Servern .....	592

Angeleitete Übung: Analysieren und Verwalten von Remote-Servern .....	608
Erhalten von Hilfe im Red Hat Customer Portal .....	613
Angeleitete Übung: Erhalten von Hilfe im Red Hat Customer Portal .....	623
Erkennen und Lösen von Problemen mit Red Hat Insights .....	625
Quiz: Erkennen und Lösen von Problemen mit Red Hat Insights .....	635
Zusammenfassung .....	637
<b>17. Ausführliche Wiederholung</b>	<b>639</b>
Ausführliche Wiederholung .....	640
Praktische Übung: Verwalten von Dateien über die Befehlszeile .....	645
Praktische Übung: Verwalten von Benutzern und Gruppen, Berechtigungen und Prozessen .....	654
Praktische Übung: Konfigurieren und Verwalten eines Servers .....	661
Praktische Übung: Verwalten von Netzwerken .....	669
Praktische Übung: Mounten von Dateisystemen und Suchen von Dateien .....	676

# Dokumentkonventionen



## Literaturhinweise

„Verweise“ geben an, wo Sie weitere Informationen zu einem Thema in externen Dokumentationen finden können.



## Anmerkung

„Hinweise“ sind Tipps, Tastenkombinationen oder alternative Ansätze für die vorliegende Aufgabe. Wenn Sie einen Hinweis ignorieren, hat dies normalerweise keine negativen Konsequenzen. Allerdings können Hinweise helfen, einen Vorgang zu optimieren.



## Wichtig

In den Feldern „Wichtig“ werden Details hervorgehoben, die andernfalls leicht übersehen werden könnten: Konfigurationsänderungen, die nur die aktuelle Sitzung betreffen, oder Dienste, die neu gestartet werden müssen, bevor ein Update angewendet werden kann. Wenn Sie ein Feld mit der Bezeichnung „Wichtig“ ignorieren, führt dies nicht zu Datenverlust, kann jedoch Irritationen und Frustration hervorrufen.



## Warnung

„Warnungen“ dürfen nicht ignoriert werden. Ignorierte Warnungen führen mit großer Wahrscheinlichkeit zu Datenverlust.



# Einführung

## Red Hat System Administration I

*Red Hat System Administration I* (RH124) wurde für IT-Experten entwickelt, die keine Erfahrung mit der Linux-Systemadministration haben. In diesem Kurs werden den Teilnehmern zentrale Kenntnisse für die Linux-Administration vermittelt, indem die wichtigsten Administrationsaufgaben behandelt werden. *Red Hat System Administration I* vermittelt zudem Grundlagenkenntnisse für Kursteilnehmer, die Linux-Systemadministratoren in Vollzeit werden möchten, indem Befehlszeilenkonzepte und Tools auf Enterprise-Ebene vorgestellt werden. Diese Konzepte werden im Folgekurs, *Red Hat System Administration II* (RH134), weiter ausgeführt.

### Lerninhalte

- Erlangen ausreichender Kenntnisse, um wichtige Systemadministrationsaufgaben unter Red Hat Enterprise Linux auszuführen
- Erlangen des grundlegenden Know-hows, das ein RHCSA-zertifizierter Red Hat Enterprise Linux-Systemadministrator benötigt

### Zielgruppe

- IT-Experten aus unterschiedlichen Fachbereichen, die grundlegende Linux-Administrationsaufgaben durchführen müssen, darunter Installation, Einrichten der Netzwerkkonnektivität, Verwalten des physischen Speichers und grundlegende Sicherheitsverwaltung.

### Voraussetzungen

- Es gibt keine formalen Voraussetzungen für diesen Kurs, Erfahrungen in der Systemadministration anderer Betriebssysteme sind jedoch von Vorteil.

# Informationen zur Kursumgebung

---

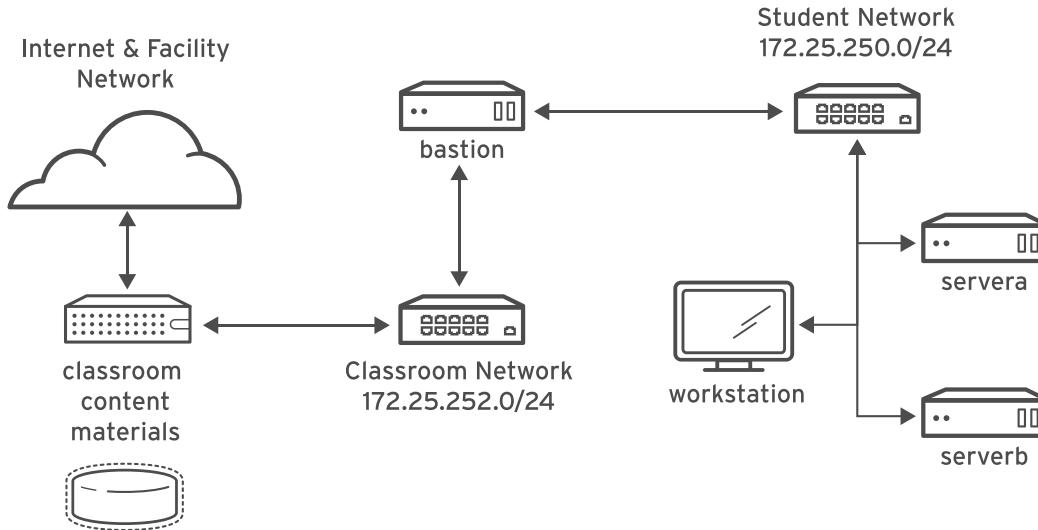


Abbildung 0.1: Kursumgebung

In diesem Kurs wird **workstation** als primäres Computersystem für praktische Übungen verwendet. Für diese Aktivitäten werden von den Teilnehmern zudem zwei weitere Rechner verwendet: **servera** und **serverb**. Alle drei Systeme befinden sich in der DNS Domain **lab.example.com**.

Alle Computersysteme für Teilnehmer verfügen über das standardmäßige Benutzerkonto **student** mit dem Passwort **student**. Das **root**-Passwort für alle Kursteilnehmer-Systeme lautet **redhat**.

## Kursraum-Rechner

Rechnername	IP-Adressen	Rolle
bastion.lab.example.com	172.25.250.254	Gateway-System zum Verbinden des privaten Netzwerks des Teilnehmers mit dem Kursraumservern (muss immer ausgeführt werden)
workstation.lab.example.com	172.25.250.9	Grafische Workstation für die Systemadministration
servera.lab.example.com	172.25.250.10	Erster Server
serverb.lab.example.com	172.25.250.11	Zweiter Server

Die Hauptfunktion von **bastion** besteht darin, als Router zwischen dem Kursraumnetzwerk und dem Netzwerk zu fungieren, das die Rechner der Kursteilnehmer verbindet. Wenn **bastion**

außer Betrieb ist, können andere Kursteilnehmer-Rechner nur auf Systeme im individuellen Kursteilnehmer-Netzwerk zugreifen.

Im Kursraum bieten verschiedene Systeme Unterstützung. Die beiden Server **content.example.com** und **materials.example.com** dienen als Quelle für Software- und Übungsmaterialien für praktische Übungen. Informationen zur Verwendung dieser Server finden Sie in der Anleitung der jeweiligen Übungen. Diese werden vom virtuellen Rechner **classroom.example.com** bereitgestellt. **classroom** und **bastion** sollten immer ausgeführt werden, damit die Übungsumgebung ordnungsgemäß verwendet wird.



### Anmerkung

Bei der Anmeldung bei **servera** oder **serverb** wird möglicherweise eine Meldung hinsichtlich der Aktivierung von **cockpit** angezeigt. Die Meldung kann ignoriert werden.

```
[student@workstation ~]$ ssh student@serverb
Warning: Permanently added 'serverb,172.25.250.11' (ECDSA) to the list of
known hosts.
Activate the web console with: systemctl enable --now cockpit.socket

[student@serverb ~]$
```

## Steuerung Ihrer Systeme

Ihnen werden Remote-Computer in einem Red Hat Online Learning-Kursraum zugewiesen. Der Zugriff darauf erfolgt über eine unter [rol.redhat.com](http://rol.redhat.com) [<http://rol.redhat.com>] gehostete Webanwendung. Sie sollten sich mithilfe Ihrer Anmelddaten für das Red Hat Customer Portal auf dieser Website anmelden.

## Steuern der virtuellen Rechner

Die virtuellen Rechner in Ihrer Kursumgebung werden über eine Webseite gesteuert. Der Status jedes virtuellen Rechners im Kursraum wird auf der unter der Registerkarte **Online Lab** befindlichen Seite angezeigt.

### Rechnerstatus

VM-Status	Beschreibung
STARTING	Der virtuelle Rechner wird hochgefahren.
STARTED	Der virtuelle Rechner wird ausgeführt und ist verfügbar (oder, falls noch hochgefahren wird, wird es bald sein.)
STOPPING	Der virtuelle Rechner wird heruntergefahren.
STOPPED	Der virtuelle Rechner ist vollständig heruntergefahren. Beim Starten fährt der virtuelle Rechner in denselben Status hoch, in dem er sich vor dem Herunterfahren befand. (Die Disk wurde nicht gelöscht.)
PUBLISHING	Der virtuelle Rechner wird anfänglich erstellt.
WAITING_TO_START	Der virtuelle Rechner wartet auf den Start anderer virtueller Rechner.

## Einführung

In Abhängigkeit des Zustands eines Rechners steht eine Auswahl der folgenden Aktionen zur Verfügung.

**Aktionen für Kursumgebung/Rechner**

<b>Schaltfläche oder Aktion</b>	<b>Beschreibung</b>
<b>PROVISION LAB</b>	Erstellen Sie den ROL-Kursraum. Hiermit werden sämtliche für die Kursumgebung erforderlichen virtuellen Rechner erstellt und gestartet. Dies dauert ggf. mehrere Minuten.
<b>DELETE LAB</b>	Entfernen Sie den ROL-Kursraum. Hiermit werden alle virtuellen Rechner im Kursraum entfernt. <b>Achtung: Alle auf den Disks gespeicherten Arbeiten gehen verloren.</b>
<b>START LAB</b>	Starten Sie alle virtuellen Rechner im Kursraum.
<b>SHUTDOWN LAB</b>	Halten Sie alle virtuellen Rechner im Kursraum an.
<b>OPEN CONSOLE</b>	Öffnet eine neue Registerkarte im Browser und stellt eine Verbindung zwischen Konsole und virtuellem Rechner her. Sie können sich direkt beim virtuellen Rechner anmelden und Befehle ausführen. In den meisten Fällen sollten Sie sich beim virtuellen Rechner <b>workstation</b> anmelden und <b>ssh</b> verwenden, um mit anderen virtuellen Rechnern eine Verbindung herzustellen.
<b>ACTION → Start</b>	Startet den virtuellen Rechner (d. h. schaltet ihn ein).
<b>ACTION → Shutdown</b>	Fährt den virtuellen Rechner ordnungsgemäß herunter, damit die Disk-Inhalte nicht verloren gehen.
<b>ACTION → Power Off</b>	Erzwingt ein Herunterfahren des virtuellen Rechners und behält die Inhalte seiner Disk bei. Dies entspricht der Stromabschaltung bei einem physischen Rechner.
<b>ACTION → Reset</b>	Erzwingt das Herunterfahren des virtuellen Rechners und setzt die Disk in den Ursprungszustand zurück. <b>Achtung: Sämtliche auf der Disk gespeicherte Arbeit geht verloren.</b>

Klicken Sie zu Beginn einer Übung, sofern Sie angewiesen wurden, einen einzelnen Knoten eines virtuellen Rechners zurückzusetzen, nur für den bestimmten virtuellen Rechner auf **ACTION → Reset**.

Klicken Sie zu Beginn einer Übung, sofern Sie angewiesen wurden, alle virtuellen Rechner zurückzusetzen, auf **ACTION → Reset**.

Wenn Sie die Kursumgebung auf ihren ursprünglichen Zustand beim Start des Kurses zurücksetzen möchten, können Sie auf **DELETE LAB** klicken, um die gesamte Kursumgebung zu entfernen. Nach dem Löschen des Labs können Sie auf **PROVISION LAB** klicken, um einen neuen Satz von Kurssystemen bereitzustellen.



### Warnung

Der Vorgang **DELETE LAB** kann nicht rückgängig gemacht werden. Die von Ihnen bis zu diesem Zeitpunkt in der Kursumgebung vorgenommene Arbeit geht verloren.

## Der Autostop-Timer

Die Registrierung bei Red Hat Online Learning ermöglicht Ihnen eine bestimmte Menge Zeit am Computer. Für den sparsamen Umgang mit der vorgegebenen Computerzeit verfügt die ROL-Kursumgebung über einen verknüpften Zählvorgang, der die Kursumgebung herunterfährt, wenn der Timer abgelaufen ist.

Klicken Sie zum Anpassen des Timers auf **MODIFY**, damit das Dialogfeld **New Autostop Time** angezeigt wird. Legen Sie die Anzahl der Stunden fest, bis der Kursraum automatisch angehalten wird. Beachten Sie, dass die maximale Dauer zehn Stunden beträgt. Klicken Sie auf **ADJUST TIME**, um diese Änderung auf die Timer-Einstellungen anzuwenden.

# Landessprachliche Anpassung

## Sprachauswahl auf Benutzerbasis

Ihre Benutzer bevorzugen für ihre Desktop-Umgebung eventuell eine andere Sprache als die Standardsprache des Systems. Für ihr Benutzerkonto möchten sie möglicherweise auch ein anderes Tastaturlayout oder eine andere Eingabemethode verwenden.

### Spracheinstellungen

In der GNOME-Desktopumgebung wird der Benutzer möglicherweise bei der ersten Anmeldung aufgefordert, seine bevorzugte Sprache und Eingabemethode einzustellen. Ist dies nicht der Fall, ist die Anwendung Region & Language für den einzelnen Benutzer der einfachste Weg, die bevorzugte Sprache und Eingabemethode anzupassen.

Sie können diese Anwendung auf zwei Arten starten. Sie können den Befehl **gnome-control-center region** über ein Terminal ausführen. Wählen Sie alternativ auf der oberen Leiste im Systemmenü in der rechten Ecke die Schaltfläche für die Einstellungen (das Symbol mit dem gekreuzten Schraubendreher und Schraubenschlüssel) im linken unteren Bereich des Menüs aus.

Wählen Sie in dem sich öffnenden Fenster Region & Language aus. Klicken Sie auf das Feld **Language**, und wählen Sie aus der daraufhin angezeigten Liste die bevorzugte Sprache aus. Dadurch wird auch die Einstellung für **Formats** auf die Standardwerte dieser Sprache aktualisiert. Bei Ihrer nächsten Anmeldung werden die Änderungen vollständig wirksam.

Diese Einstellungen wirken sich auf die GNOME-Desktop-Umgebung sowie auf alle Anwendungen, beispielsweise **gnome-terminal** aus, die innerhalb der Umgebung gestartet werden. Standardmäßig werden sie jedoch nicht auf dieses Benutzerkonto angewendet, wenn über eine **ssh**-Anmeldung von einem Remote-System oder über eine textbasierte Anmeldung von einer virtuellen Konsole (beispielsweise **tty5**) darauf zugegriffen wird.



### Anmerkung

Sie können festlegen, dass Ihre Shell-Umgebung dieselbe **LANG**-Einstellung wie Ihre grafische Umgebung verwendet, selbst wenn Sie sich über eine textbasierte virtuelle Konsole oder über **ssh** anmelden. Hierzu kann beispielsweise Code ähnlich dem folgenden in Ihrer `~/.bashrc`-Datei platziert werden. Der Code im folgenden Beispiel stellt die Sprache für eine Textanmeldung so ein, dass sie mit der zurzeit für die GNOME Desktop-Umgebung des Benutzers konfigurierten Sprache übereinstimmt:

```
i=$(grep 'Language=' /var/lib/AccountsService/users/${USER} \
| sed 's/Language=//')
if [ "$i" != "" ]; then
    export LANG=$i
fi
```

Japanisch, Koreanisch, Chinesisch und andere Sprachen mit nicht lateinischem Zeichensatz werden in textbasierten virtuellen Konsolen eventuell nicht einwandfrei angezeigt.

Durch Festlegen der Variablen **LANG** in der Befehlszeile kann für einzelne Befehle die Verwendung einer anderen Sprache festgelegt werden:

```
[user@host ~]$ LANG=fr_FR.utf8 date  
jeu. avril 25 17:55:01 CET 2019
```

Bei den nachfolgenden Befehlen erfolgt die Ausgabe wieder in der Standardsprache des Systems. Der Befehl **locale** kann verwendet werden, um den aktuellen Wert von **LANG** und andere relevante Umgebungsvariablen zu bestimmen.

## Einstellungen der Eingabemethode

GNOME 3 in Red Hat Enterprise Linux 7 oder höher verwendet automatisch das Auswahlsystem der Eingabemethode IBus, wodurch sich die Tastaturlayouts und Eingabemethoden schnell wechseln lassen.

Die Anwendung Region & Language kann auch zum Aktivieren alternativer Eingabemethoden verwendet werden. Im Anwendungsfenster von Region & Language zeigt das Feld **Input Sources** an, welche Eingabemethoden derzeit verfügbar sind. Standardmäßig ist eventuell **English (US)** die einzige verfügbare Methode. Markieren Sie **English (US)**, und klicken Sie auf das **Tastatur**-Symbol, um das aktuelle Tastaturlayout anzuzeigen.

Um eine weitere Eingabemethode hinzuzufügen, klicken Sie auf die Schaltfläche **+** im unteren linken Bereich des Fensters **Input Sources**. Das Fenster **Add an Input Source** wird geöffnet. Wählen Sie Ihre Sprache und anschließend die bevorzugte Eingabemethode oder das Tastaturlayout aus.

Wenn mehr als eine Eingabemethode konfiguriert ist, kann der Benutzer rasch zwischen diesen wechseln, indem er **Super+Space** (manchmal auch **Windows+Space**) eingibt. Auf der oberen GNOME-Leiste wird ein **Statusindikator** angezeigt, der über zwei Funktionen verfügt: Er gibt an, welche Ausgabemethode aktiv ist, und fungiert als Menü, das zum Wechseln zwischen Eingabemethoden oder zur Auswahl von erweiterten komplexeren Eingabemethoden verwendet werden kann.

Einige der Methoden sind mit Zahnräder gekennzeichnet. Diese verfügen über erweiterte Konfigurationsoptionen und Funktionen. Beispielsweise kann der Benutzer mit der japanischen Eingabemethode **Japanese (Kana Kanji)** Text im lateinischen Zeichensatz vorab bearbeiten und mit den Tasten **Pfeil nach unten** und **Pfeil nach oben** die zu verwendenden Zeichen auswählen.

Englischsprachige Benutzer in den USA finden dies ggf. ebenfalls hilfreich. Bei der Eingabemethode **English (United States)** lautet beispielsweise das Tastaturlayout **English (international AltGr dead keys)**, das die Taste **AltGr** (oder die rechte **Alt**-Taste) auf einer PC-Tastatur mit 104/105 Tasten als Zusatztaste in Form einer „zweiten Umschalttaste“ sowie als Aktivierungstaste für nicht belegte Tasten zur Eingabe zusätzlicher Zeichen behandelt. Zur Verfügung stehen auch Dvorak und andere Tastaturlayouts.



### Anmerkung

In der GNOME Desktop-Umgebung können alle Unicode-Zeichen eingegeben werden, sofern Sie den Unicode-Codepoint oder Unicode-Zahlenwert des Zeichens kennen. Drücken Sie auf **Strg+Umschalt+U** und dann auf den Codepoint. Nach dem Drücken von **Strg+Umschalt+U** erscheint ein unterstrichenes **u**, das angibt, dass das System auf die Eingabe des Unicode-Codepunkts wartet.

Beispielsweise hat der kleine griechische Buchstabe Lambda den Codepoint U +03BB und wird durch Eingabe von **Strg+Umschalt+U**, anschließend **03BB** und Drücken der **Eingabetaste** eingegeben.

## Einstellungen der Standardsprache des Systems

Die Standardsprache des Systems ist US-Englisch mit den Unicode-Zeichen der UTF-8-Codierung (**en\_US.utf8**). Dies lässt sich aber während oder nach der Installation ändern.

Der **root**-Benutzer kann in der Befehlszeile die systemweiten Ländereinstellungen mit dem Befehl **localectl** ändern. Wenn **localectl** ohne Argumente ausgeführt wird, werden die aktuellen Ländereinstellungen des Systems angezeigt.

Führen Sie zum Einstellen der systemweiten Standardsprache den Befehl **localectl set-locale LANG=locale** aus, wobei *locale* der entsprechende Wert für die **LANG**-Umgebungsvariable aus der Tabelle „Sprachcodereferenz“ in diesem Kapitel ist. Die Änderung wird für die Benutzer bei der nächsten Anmeldung wirksam und wird in der Datei **/etc/locale.conf** gespeichert.

```
[root@host ~]# localectl set-locale LANG=fr_FR.utf8
```

In GNOME kann ein Administrator diese Einstellung ändern, indem er in Region & Language in der rechten oberen Ecke des Fensters auf die Schaltfläche **Login Screen** klickt. Das Ändern der **Sprache** des grafischen Anmeldebildschirms wirkt sich auch auf die Einstellungen der Standardsprache des Systems aus, die in der Konfigurationsdatei **/etc/locale.conf** gespeichert sind.



### Wichtig

Textbasierte virtuelle Konsolen wie **tty4** sind hinsichtlich der Schriftarten, die sie anzeigen können, eingeschränkter als Terminals in einer virtuellen Konsole, die in einer grafischen Umgebung ausgeführt wird, oder als Pseudoterminals für **ssh**-Sitzungen. Japanische, koreanische und chinesische Zeichen beispielsweise werden in einer textbasierten virtuellen Konsole eventuell nicht richtig angezeigt. Daher sollten Sie in Betracht ziehen, Englisch oder eine andere Sprache mit einem lateinischen Zeichensatz als systemweite Standardeinstellung zu verwenden.

Gleichermaßen sind textbasierte virtuelle Konsolen bei den von ihnen unterstützten Eingabemethoden eingeschränkt. Dies wird separat über die grafische Desktopumgebung verwaltet. Die verfügbaren globalen Eingabeeinstellungen werden sowohl für textbasierte virtuelle Konsolen als auch für die grafische Umgebung anhand von **localectl** konfiguriert. Weitere Informationen finden Sie auf den Manpages **localectl(1)** und **vconsole.conf(5)**.

## Sprachpakete

Mit speziellen RPM-Paketen, die als *langpacks* bezeichnet werden, werden Sprachpakete installiert, die Unterstützung für bestimmte Sprachen hinzufügen. Diese Sprachpakete nutzen Abhängigkeiten, um zusätzliche RPM-Pakete, die Lokalisierungen, Verzeichnisse und Übersetzungen für andere Softwarepakete enthalten, automatisch auf Ihrem System zu installieren.

Verwenden Sie **yum list langpacks-\*** zum Auflisten der Sprachpakete, die installiert sind und möglicherweise installiert werden:

```
[root@host ~]# yum list langpacks-*
Updating Subscription Management repositories.
Updating Subscription Management repositories.
Installed Packages
langpacks-en.noarch      1.0-12.el8        @AppStream
Available Packages
langpacks-af.noarch       1.0-12.el8        rhel-8-for-x86_64-appstream-rpms
langpacks-am.noarch       1.0-12.el8        rhel-8-for-x86_64-appstream-rpms
langpacks-ar.noarch       1.0-12.el8        rhel-8-for-x86_64-appstream-rpms
langpacks-as.noarch       1.0-12.el8        rhel-8-for-x86_64-appstream-rpms
langpacks-ast.noarch      1.0-12.el8        rhel-8-for-x86_64-appstream-rpms
...output omitted...
```

Um Sprachunterstützung hinzuzufügen, installieren Sie das entsprechende Sprachpaket. Mit dem folgenden Befehl wird beispielsweise Unterstützung für Französisch hinzugefügt:

```
[root@host ~]# yum install langpacks-fr
```

Mit **yum repoquery --whatsupplements** können Sie bestimmen, welche RPM-Pakete durch ein Sprachpaket installiert werden können:

```
[root@host ~]# yum repoquery --whatsupplements langpacks-fr
Updating Subscription Management repositories.
Updating Subscription Management repositories.
Last metadata expiration check: 0:01:33 ago on Wed 06 Feb 2019 10:47:24 AM CST.
glibc-langpack-fr-0:2.28-18.el8.x86_64
gnome-getting-started-docs-fr-0:3.28.2-1.el8.noarch
hunspell-fr-0:6.2-1.el8.noarch
hyphen-fr-0:3.0-1.el8.noarch
libreoffice-langpack-fr-1:6.0.6.1-9.el8.x86_64
man-pages-fr-0:3.70-16.el8.noarch
mythes-fr-0:2.3-10.el8.noarch
```



### Wichtig

Sprachpakte nutzen schwache Abhängigkeiten für RPM, damit Zusatzpakte nur installiert werden, wenn das Kernpaket, das sie benötigt, ebenfalls installiert ist.

Wenn beispielsweise *langpacks-fr* wie in den vorherigen Beispielen gezeigt installiert wird, wird das Paket *mythes-fr* nur installiert, wenn auch der Thesaurus *mythes* auf dem System installiert ist.

Wenn *mythes* anschließend auf dem System installiert wird, wird das Paket *mythes-fr* aufgrund der schwachen Abhängigkeit des bereits installierten Pakets *langpacks-fr* ebenfalls automatisch installiert.



### Literaturhinweise

Manpages **locale(7)**, **localectl(1)**, **locale.conf(5)**, **vconsole.conf(5)**, **unicode(7)** und **utf-8(7)**

Konvertierungen zwischen den Namen der X11-Layouts der grafischen Desktopumgebung und deren Namen in **localectl** befinden sich in der Datei **/usr/share/X11/xkb/rules/base.lst**.

## Sprachcodereferenz



### Anmerkung

Diese Tabelle enthält möglicherweise nicht alle auf Ihrem System verfügbaren Sprachpakte. Verwenden Sie **yum info langpacks-SUFFIX**, um weitere Informationen zum jeweiligen Sprachpaket abzurufen.

### Sprachcodes

Sprache	Suffix für Sprachpakte	\$LANG-Wert
Englisch (US)	en	en_US.utf8
Assamesisch	as	as_IN.utf8
Bengalisch	bn	bn_IN.utf8
Chinesisch (vereinfacht)	zh_CN	zh_CN.utf8
Chinesisch (traditionell)	zh_TW	zh_TW.utf8
Französisch	fr	fr_FR.utf8
Deutsch	de	de_DE.utf8
Gujarati	gu	gu_IN.utf8
Hindi	hi	hi_IN.utf8

<b>Sprache</b>	<b>Suffix für Sprachpakete</b>	<b>\$LANG-Wert</b>
Italienisch	it	it_IT.utf8
Japanisch	ja	ja_JP.utf8
Kanaresisch	kn	kn_IN.utf8
Koreanisch	ko	ko_KR.utf8
Malayalam	ml	ml_IN.utf8
Marathisch	mr	mr_IN.utf8
Odia	oder	or_IN.utf8
Portugiesisch (Brasilien)	pt_BR	pt_BR.utf8
Punjabi	pa	pa_IN.utf8
Russisch	ru	ru_RU.utf8
Spanisch	es	es_ES.utf8
Tamilisch	ta	ta_IN.utf8
Telugu	te	te_IN.utf8



## Kapitel 1

# Erste Schritte mit Red Hat Enterprise Linux

### Ziel

Beschreiben und Definieren von Open Source, Linux, Linux-Distributionen und Red Hat Enterprise Linux

### Ziele

- Definieren und Erläutern des Zwecks von Linux, Open Source, Linux-Distributionen und Red Hat Enterprise Linux

### Abschnitte

- Was ist Linux?

### Test

Erste Schritte mit Red Hat Enterprise Linux

# Was ist Linux?

---

## Ziele

Nach Abschluss dieses Abschnitts sollten Sie in der Lage sein, den Zweck von Linux, Open Source, Linux-Distributionen und Red Hat Enterprise Linux zu definieren und zu erläutern.

## Gründe für das Erlernen von Linux

Linux ist eine wichtige Technologie, die IT-Experten kennen müssen.

Linux ist weit verbreitet, und wenn Sie das Internet nutzen, interagieren Sie in Ihrem Alltag wahrscheinlich bereits mit Linux-Systemen. Die offensichtlichste Art und Weise, wie Sie mit Linux-Systemen interagieren, ist das Browsen im World Wide Web und die Verwendung von E-Commerce-Sites zum Kaufen und Verkaufen von Produkten.

Linux wird jedoch noch für viel mehr verwendet. Linux verwaltet Point-of-Sale-Systeme und die Aktienmärkte der Welt und bildet auch die Basis von Smart-TVs und Unterhaltungssystemen in Flugzeugen. Die meisten der Top-500-Supercomputer weltweit werden mit Linux betrieben. Linux bietet die grundlegenden Technologien für die Cloud-Revolution und die Tools zur Erstellung der nächsten Generation von containerbasierten Microservices-Anwendungen, softwarebasierten Storage-Technologien und Big Data-Lösungen.

Im modernen Rechenzentrum sind Linux und Microsoft Windows die Hauptakteure, und die Bedeutung von Linux in diesem Bereich nimmt zu. Zu den vielen Gründen, warum Linux erlernt werden sollte, zählen:

- Windows-Benutzer müssen mit Linux zusammenarbeiten.
- Bei der Anwendungsentwicklung hostet Linux die Anwendung oder deren Laufzeit.
- Beim Cloud Computing verwenden die Cloud-Instanzen in Private oder Public Cloud-Umgebungen Linux als Betriebssystem.
- Bei mobilen Anwendungen oder beim Internet der Dinge (IoT) verwendet Ihr Gerät höchstwahrscheinlich das Betriebssystem Linux.
- Wenn Sie nach neuen beruflichen Chancen in der IT suchen, sind Linux-Kenntnisse sehr gefragt.

## Was sind die Vorteile von Linux?

Es gibt viele verschiedene Antworten auf die Frage: „Was sind die Vorteile von Linux?“ Drei davon sind:

- Linux ist Open Source Software.

Open Source bedeutet nicht nur, dass Sie sehen können, wie das System funktioniert. Sie können auch mit Änderungen experimentieren und diese frei an andere Benutzer weitergeben. Das Open Source-Modell bedeutet, dass Verbesserungen einfacher vorgenommen und dadurch Innovationen beschleunigt werden können.

- Linux bietet einfachen Zugriff auf eine leistungsstarke und skriptfähige Befehlszeilenschnittstelle (*Command-Line Interface, CLI*).

Linux basiert auf der grundlegenden Designphilosophie, dass Benutzer alle Verwaltungsaufgaben über die CLI ausführen können. Es vereinfacht Automatisierung, Bereitstellung und Provisioning und erleichtert sowohl die lokale als auch die Remote-Systemadministration. Im Gegensatz zu anderen Betriebssystemen wurden diese Fähigkeiten von Anfang an integriert, und es wurde immer davon ausgegangen, dass diese wichtigen Fähigkeiten auch nutzbar gemacht werden.

- Linux ist ein modulares Betriebssystem, mit dem Sie Komponenten einfach austauschen oder entfernen können.

Komponenten des Systems können bei Bedarf verbessert und aktualisiert werden. Ein Linux-System kann eine universelle Entwicklungs-Workstation oder eine extrem abgespeckte Software-Appliance sein.

## Was ist Open Source Software?

Open Source Software ist Software mit Quellcode, den jeder verwenden, untersuchen, ändern und teilen kann.

Quellcode ist ein Satz von für Menschen lesbaren Anweisungen, die zum Erstellen eines Programms verwendet werden. Er kann als Skript interpretiert oder in eine binäre ausführbare Datei kompiliert werden, die der Computer direkt ausführt. Beim Erstellen des Quellcodes wird dieser urheberrechtlich geschützt und der Inhaber des Urheberrechts kontrolliert die Bedingungen, unter denen die Software kopiert, angepasst und verteilt werden kann. Benutzer können diese Software unter einer Softwarelizenz verwenden.

Manche Software verfügt über Quellcode, den nur die Person, das Team oder das Unternehmen, die bzw. das ihn erstellt hat, einsehen, ändern oder verteilen kann. Diese Software wird manchmal als „proprietäre“ oder „Closed Source“-Software bezeichnet. Normalerweise erlaubt die Lizenz dem Endbenutzer lediglich, das Programm auszuführen, und gewährt keinen Zugriff oder nur streng limitierten Zugriff auf den Quellcode.

Open Source Software ist anders. Wenn der Copyright-Inhaber Software unter einer Open Source-Lizenz bereitstellt, erteilt er dem Benutzer das Recht, das Programm auszuführen und außerdem den Quellcode anzuzeigen, zu ändern, zu kompilieren und gebührenfrei an andere weiterzugeben.

Open Source fördert Zusammenarbeit, gemeinsame Nutzung, Transparenz und schnelle Innovationen, da es auch andere Personen außer den ursprünglichen Entwicklern ermutigt, Änderungen und Verbesserungen an der Software vorzunehmen und diese mit anderen zu teilen.

Nur weil die Software Open Source ist, bedeutet dies nicht, dass sie nicht kommerziell genutzt oder bereitgestellt werden kann. Open Source ist ein entscheidender Bestandteil des Geschäftsbetriebs vieler Unternehmen. Bestimmte Open Source-Lizenzen ermöglichen die Wiederverwendung von Code in proprietären Produkten. Open Source-Code kann verkauft werden, aber die Nutzungsbedingungen von echten Open Source-Lizenzen gestatten dem Kunden im Allgemeinen, den Quellcode weiterzugeben. Meist stellen Anbieter wie Red Hat kommerzielle Hilfe für Bereitstellung, Support und Erweiterung von auf Open Source-Produkten basierenden Lösungen bereit.

Open Source bietet dem Benutzer viele Vorteile:

- **Kontrolle:** Nachvollziehen, was der Code ausführt, und Ändern des Codes, um ihn zu verbessern
- **Schulung:** Lernen aus realem Code und Entwickeln nützlicherer Anwendungen

## Kapitel 1 | Erste Schritte mit Red Hat Enterprise Linux

- *Sicherheit:* Überprüfen von vertraulichem Code und Korrigieren des Codes mit oder ohne Hilfe der ursprünglichen Entwickler
- *Stabilität:* Weiterverwendung des Codes auch bei Wegfall des ursprünglichen Entwicklers oder Vertriebspartners

Fazit ist, dass mit Open Source durch Zusammenarbeit bessere Software mit höherer Rendite erstellt werden kann.

## Arten von Open Source-Lizenzen

Es gibt mehrere Möglichkeiten, Open Source Software bereitzustellen. Die Nutzungsbedingungen der Softwarelizenz steuern, wie der Quellcode mit anderem Code kombiniert oder wiederverwendet werden kann, und es gibt Hunderte verschiedener Open Source-Lizenzen. Bedingung für Open Source ist jedoch, dass Lizenzen Benutzern erlauben müssen, den Code frei zu verwenden, anzuseigen, zu ändern, zu kompilieren und zu verteilen.

Es gibt zwei allgemeine Klassen von Open Source-Lizenzen, die besonders wichtig sind:

- *Copyleft*-Lizenzen, die darauf ausgerichtet sind, dass Code als Open Source bestehen bleibt.
- *Freizügige* Lizenzen, die die Wiederverwendbarkeit von Code maximieren sollen.

Copyleft- oder „Share Alike“-Lizenzen setzen voraus, dass jeder, der den Quellcode mit oder ohne Änderungen bereitstellt, auch die Freiheit für andere Benutzer zum Kopieren, Ändern und Verteilen des Codes weitergeben muss. Der grundlegende Vorteil dieser Lizenzen besteht darin, dass sie dazu beitragen, vorhandenen Code und Verbesserungen an diesem Code offen zu halten und die Menge an verfügbarem Open Source-Code zu vergrößern. Zu den am häufigsten verwendeten Copyleft-Lizenzen zählen die *GNU General Public License (GPL)* und die *Lesser GNU Public License (LGPL)*.

Freizügige Lizenzen sind darauf ausgerichtet, die Wiederverwendbarkeit von Quellcode zu maximieren. Benutzer können den Quellcode für einen beliebigen Zweck verwenden, solange die Copyright- und Lizenzbestimmungen gewahrt bleiben; dies schließt auch die Wiederverwendung dieses Codes unter restriktiveren oder sogar proprietären Lizenzen ein. Dadurch kann dieser Code sehr einfach wiederverwendet werden, jedoch mit dem Risiko, dass nur proprietäre Verbesserungen erfolgen. Häufig verwendete freizügige Open Source-Lizenzen sind u. a. die *MIT/X11-Lizenz*, die *Simplified BSD License* und die *Apache Software License 2.0*.

## Wer entwickelt Open Source Software?

Es ist ein Missverständnis zu glauben, dass Open Source ausschließlich von einer „Armee von Freiwilligen“ oder sogar von einer Armee von Einzelpersonen und Red Hat entwickelt wird. Die Entwicklung von Open Source erfolgt heute überwiegend professionell. Viele Entwickler werden von ihren Unternehmen dafür bezahlt, an Open Source-Projekten zu arbeiten, um die von ihnen und ihren Kunden benötigten Verbesserungen zu erstellen und einzubringen.

Freiwillige und die akademische Community spielen eine wichtige Rolle und können insbesondere in neuen Technologiebereichen entscheidende Beiträge leisten. Die Kombination formeller und informeller Entwicklung bildet eine äußerst dynamische und produktive Umgebung.

## Wer ist Red Hat?

Red Hat ist der weltweit führende Anbieter von Open Source Software-Lösungen. Unsere in der Community entwickelten Lösungen umfassen zuverlässige und leistungsstarke Cloud-, Linux-, Middleware-, Storage- und Virtualisierungstechnologien. Die Mission von Red Hat besteht darin,

als Katalysator in Communitys von Kunden, Mitwirkenden und Partnern zu agieren, um bessere Technologien auf Grundlage des Open Source-Gedankens zu entwickeln.

Die Aufgabe von Red Hat ist es, Kunden dabei zu unterstützen, mit der Open Source-Community und ihren Partnern in Kontakt zu treten, um Open Source Software-Lösungen effektiv zu nutzen. Red Hat beteiligt sich aktiv an der Open Source-Community und unterstützt diese. Die langjährige Erfahrung hat das Unternehmen von der Bedeutung von Open Source für die Zukunft der IT-Branche überzeugt.

Am bekanntesten ist Red Hat für seine Teilnahme an der Linux-Community und für die Red Hat Enterprise Linux-Distribution. Red Hat ist aber auch in anderen Open Source Communitys sehr aktiv, einschließlich Middleware-Projekten, die unter anderem auf die JBoss-Entwickler-Community, Virtualisierungslösungen, Cloud-Technologien wie OpenStack und OpenShift sowie auf die softwarebasierten Ceph- und Gluster-Storage-Projekte ausgerichtet sind.

## Was ist eine Linux-Distribution?

Eine *Linux-Distribution* ist ein installierbares Betriebssystem, das auf einem Linux-Kernel aufgebaut ist und Benutzerprogramme und -bibliotheken unterstützt. Ein vollständiges Linux-Betriebssystem wird nicht von einem einzelnen Unternehmen entwickelt, sondern von einer Reihe unabhängiger Open Source-Entwicklungs-Communitys, die mit einzelnen Softwarekomponenten arbeiten. Eine Distribution bietet Benutzern eine einfache Möglichkeit, ein funktionierendes Linux-System zu installieren und zu verwalten.

1991 entwickelte ein junger Informatik-Student namens Linus Torvalds einen Unix-ähnlichen Kernel, den er *Linux* nannte und als Open Source Software unter der GPL lizenzierte. Der Kernel ist die Hauptkomponente des Betriebssystems, das Hardware, Arbeitsspeicher und die Planung von laufenden Programmen verwaltet. Dieser Linux-Kernel konnte dann durch andere Open Source Software ergänzt werden, z. B. Dienstprogramme und Programme aus dem GNU-Projekt, die grafische X Window System-Schnittstelle von MIT und viele andere Open Source-Komponenten, wie der Sendmail-Mail-Server oder der Apache-HTTP-Webserver, um ein vollständiges Unix-ähnliches Open Source-Betriebssystem zu erstellen.

Eine der Herausforderungen für Linux-Benutzer bestand jedoch darin, dass alle diese Teile aus vielen verschiedenen Quellen zusammengestellt werden mussten. Linux-Entwickler begannen bereits früh mit der Bereitstellung einer Distribution vorgefertigter und getester Tools, die Benutzer herunterladen und verwenden konnten, um ihre Linux-Systeme schnell einzurichten.

Es gibt viele verschiedene Linux-Distributionen mit unterschiedlichen Zielen und Kriterien für die Auswahl und Unterstützung der von ihrer Distribution bereitgestellten Software. Distributionen weisen jedoch im Allgemeinen viele gemeinsame Merkmale auf:

- Distributionen bestehen aus einem Linux-Kernel und unterstützenden Programmen aus dem Benutzerraum.
- Distributionen können klein sein und nur einem Zweck dienen oder Tausende von Open Source-Programmen enthalten.
- Distributionen müssen eine Methode zur Installation und Aktualisierung der Distribution und ihrer Komponenten bieten.
- Der Anbieter der Distribution muss diese Software unterstützen und im Idealfall direkt an der Community teilnehmen, die diese Software entwickelt.

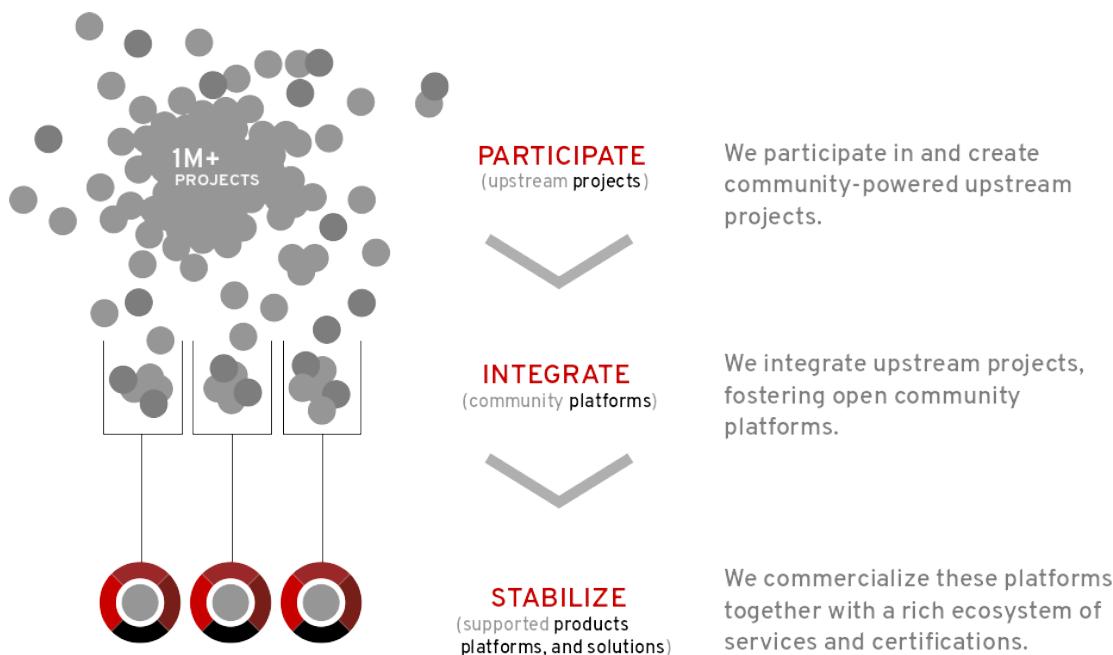
*Red Hat Enterprise Linux* ist die kommerzialisierte Linux-Distribution von Red Hat.

## Red Hat Enterprise Linux

### Entwicklung von Red Hat Enterprise Linux

Red Hat entwickelt und integriert Open Source Software über einen mehrstufigen Prozess in RHEL.

- Red Hat *beteiligt sich* an der Unterstützung einzelner Open Source-Projekte. Es trägt Code, Entwicklerzeit, Ressourcen und andere Unterstützung bei und arbeitet häufig mit Entwicklern anderer Linux-Distributionen zusammen. Dadurch wird die allgemeine Softwarequalität für alle verbessert.
- Red Hat sponsert und *integriert* Open Source-Projekte in die Community-basierte Linux-Distribution *Fedora*. Fedora stellt eine kostenlose Arbeitsumgebung zur Verfügung, die als Entwicklungslabor dienen kann und die Grundlage für die Funktionen bietet, die in die kommerzialisierten Produkte integriert werden.
- Red Hat *stabilisiert* die Software, um sicherstellen, dass sie für langfristige Unterstützung und Standardisierung bereit ist, und integriert sie in seine Unternehmensdistribution RHEL.



### Fedora

Fedora ist ein Community-Projekt, das ein vollständiges, kostenloses, Linux-basiertes Betriebssystem erstellt und veröffentlicht. Red Hat sponsert die Community und arbeitet mit Community-Vertretern zusammen, um die neueste Upstream-Software in eine sich schnell ändernde und sichere Distribution zu integrieren. Das Fedora-Projekt bringt alles zurück in die kostenlose Open Source-Welt, und jeder kann daran teilhaben.

Fedora ist jedoch auf Innovation und Exzellenz ausgelegt, nicht auf langfristige Stabilität. Alle sechs Monate gibt es neue Hauptaktualisierungen, die erhebliche Änderungen umfassen können. Fedora unterstützt Releases nur für etwa ein Jahr (zwei Hauptaktualisierungen). Dies macht es weniger geeignet für den Einsatz in Unternehmen.

## Red Hat Enterprise Linux

Red Hat Enterprise Linux (RHEL) ist die unternehmenstaugliche, kommerziell unterstützte Linux-Distribution von Red Hat. Es ist die führende Plattform für Open Source Computing und nicht nur eine Sammlung ausgereifter Open Source-Projekte. RHEL wurde umfassend getestet und verfügt über ein umfassendes System von Partnern, Hardware- und Softwarezertifizierungen, Beratungsdiensten, Schulungen sowie mehrjährigen Garantie- und Wartungsleistungen.

Red Hat basiert die Haupt-Releases von RHEL auf Fedora. Danach kann Red Hat allerdings auswählen, welche Pakete eingeschlossen werden sollen, weitere Verbesserungen vornehmen (die zurück an die Upstream-Projekte und Fedora geleitet werden) und Konfigurationsentscheidungen treffen, die den Bedürfnissen der Kunden entsprechen. Red Hat unterstützt Anbieter und Kunden dabei, mit der Open Source Community und bei der Upstream-Entwicklung zusammenzuarbeiten, um Lösungen zu entwickeln und Probleme zu beheben.

Red Hat Enterprise Linux verwendet ein subskriptionsbasiertes Distributionsmodell. Da es sich um Open Source Software handelt, ist dies keine Lizenzgebühr. Es wird stattdessen für Support, Wartung, Updates, Sicherheitspatches und den Zugriff auf die Knowledgebase im Red Hat Customer Portal (<http://access.redhat.com/>), Zertifizierungen und so weiter bezahlt. Der Kunde bezahlt für langfristigen Support und Fachwissen, Engagement und Unterstützung bei Bedarf.

Wenn Hauptaktualisierungen verfügbar sind, können Kunden nach Belieben zu diesen wechseln, ohne mehr zu bezahlen. Dadurch wird die Verwaltung der wirtschaftlichen und praktischen Aspekte von Systemaktualisierungen vereinfacht.

## CentOS

CentOS ist eine Community-basierte Linux-Distribution, die von einem Großteil der Open Source-Red Hat Enterprise Linux-Codebasis und anderen Quellen abgeleitet ist. Es ist kostenlos, einfach zu installieren und wird von Mitgliedern einer aktiven Benutzer-Community betreut und unterstützt, die unabhängig von Red Hat arbeiten.

In der folgenden Tabelle sind einige wichtige Unterschiede zwischen CentOS und Red Hat Enterprise Linux aufgeführt.

CentOS	Red Hat Enterprise Linux
Nur Self-Support	Es stehen verschiedene Support-Level zur Verfügung, darunter Standard-Support während der Geschäftszeiten, Premium-Support rund um die Uhr bei kritischen Problemen und Einsteiger-Support-Subskriptionen. Verschiedene SLA-Levels können gemischt und in einer Umgebung angepasst werden.
Die Errata-Produktion beginnt, wenn eine offizielle RHEL-Errata-Veröffentlichung verfügbar ist.	Schnelle Antwort von internen Entwicklern, Hotfixes sind möglicherweise vor der offiziellen RHEL-Errata-Veröffentlichung verfügbar.

CentOS	Red Hat Enterprise Linux
Paketupdates werden für die neueste Nebenversion bis zum Ende der Phase „RHEL Maintenance Support 2“ bereitgestellt.	Updates für ältere Nebenversionen sind im Programm „Extended Update Support“ (EUS) und noch Jahre nach dem Ende des „Maintenance Support 2“ durch das Programm „Extended Lifecycle Support“ (ELS) verfügbar.
Im Allgemeinen nicht von Softwareanbietern wie SAS, SAP und Oracle als unterstützte Plattform zertifiziert.	Tausende zertifizierte Anwendungen von Hunderten von ISVs.
Hilfe- und Dokumentationsressourcen sind in Foren, Mailinglisten, im Chat, auf der CentOS Project-Website und im Wiki sowie in anderen Community-Quellen verfügbar.	Dokumentation, Referenzarchitekturen, Fallstudien und Knowledgebase-Artikel sind im Red Hat Customer Portal erhältlich. Zugriff auf Red Hat Customer Portal Labs, eine Reihe von Tools, mit denen Sie die Leistung verbessern, Sicherheitsprobleme identifizieren oder bei Problemen helfen können. Optionale proaktive Systemanalyse mit Red Hat Insights, einem SaaS-Tool zur Bereitstellung von Echtzeit-Bewertung der Risiken in Bezug auf Leistung, Verfügbarkeit, Stabilität und Sicherheit.

## Testen von Red Hat Enterprise Linux

Es gibt viele verschiedene Möglichkeiten, Red Hat Enterprise Linux zu testen. Eine Möglichkeit ist das Herunterladen einer Evaluierungsversion von der Website unter <https://access.redhat.com/products/red-hat-enterprise-linux/evaluation>. Diese Seite enthält Links zu zusätzlichen Informationen.

Über das Red Hat Developer Program unter <https://developer.redhat.com> bietet Red Hat außerdem kostenlose Subskriptionen für eine Reihe von Produkten für Entwicklungszwecke. Mit diesen Subskriptionen können Entwickler ihre Software schnell entwickeln, einen Prototyp erstellen, testen und demonstrieren, um sie auf denselben Unternehmensprodukten bereitzustellen.

Ein anderer Ansatz besteht darin, eine Instanz von Red Hat Enterprise Linux über einen Cloud-Anbieter bereitzustellen. Zum Beispiel verfügt Red Hat über offizielle AMIs für Red Hat Enterprise Linux im Amazon AWS Marketplace.

Weitere Informationen finden Sie auf der Seite „Erste Schritte“ von Red Hat Enterprise Linux, auf die am Ende dieses Abschnitts verwiesen wird.



### Literaturhinweise

#### Erste Schritte mit Red Hat Enterprise Linux

<https://access.redhat.com/products/red-hat-enterprise-linux#getstarted>

#### Die Open Source-Lösung

<https://opensource.com/open-source-way>

## ► Quiz

# Erste Schritte mit Red Hat Enterprise Linux

Wählen Sie die richtigen Antworten auf die folgenden Fragen aus:

► 1. Welche zwei der folgenden Antworten nennen Vorteile von Open Source Software für den Benutzer? (Wählen Sie zwei Antworten aus.)

- a. Code kann auch bei Wegfall des ursprünglichen Entwicklers oder Vertriebspartners weiter verwendet werden.
- b. Vertrauliche Teile des Codes sind geschützt und nur für den ursprünglichen Entwickler verfügbar.
- c. Sie können aus realem Code lernen und effektivere Anwendungen entwickeln.
- d. Code bleibt offen, solange er sich in einem öffentlichen Repository befindet. Die Lizenz kann sich jedoch ändern, wenn der Code in proprietärer Software enthalten ist.

► 2. Welche zwei der folgenden Antworten beschreiben Methoden, wie Red Hat Produkte für die Zukunft entwickelt und mit der Community interagiert? (Wählen Sie zwei Antworten aus.)

- a. Sponsoren und Integrieren von Open Source-Projekten in das Community-basierte Fedora-Projekt
- b. Entwickeln spezifischer Integrationstools, die nur in Red Hat-Distributionen verfügbar sind
- c. Beteiligen an Upstream-Projekten
- d. Neuverpacken und Neulizenzieren von Community-Produkten

► 3. Welche zwei Aussagen beschreiben die Vorteile von Linux? (Wählen Sie zwei Antworten aus.)

- a. Linux wird ausschließlich von Freiwilligen entwickelt und ist somit ein kostengünstiges Betriebssystem.
- b. Linux ist modular aufgebaut und kann als vollständiger grafischer Desktop oder als kleine Appliance konfiguriert werden.
- c. Jedes Release von Linux bleibt mindestens ein Jahr lang auf einem bekannten Stand gesperrt, wodurch die Entwicklung kundenspezifischer Software vereinfacht wird.
- d. Linux beinhaltet eine leistungsstarke und skriptfähige Befehlszeilschnittstelle, die eine einfachere Automatisierung und Bereitstellung ermöglicht.

## ► Lösung

# Erste Schritte mit Red Hat Enterprise Linux

Wählen Sie die richtigen Antworten auf die folgenden Fragen aus:

- 1. Welche zwei der folgenden Antworten nennen Vorteile von Open Source Software für den Benutzer? (Wählen Sie zwei Antworten aus.)
- a. Code kann auch bei Wegfall des ursprünglichen Entwicklers oder Vertriebspartners weiter verwendet werden.
  - b. Vertrauliche Teile des Codes sind geschützt und nur für den ursprünglichen Entwickler verfügbar.
  - c. Sie können aus realem Code lernen und effektivere Anwendungen entwickeln.
  - d. Code bleibt offen, solange er sich in einem öffentlichen Repository befindet. Die Lizenz kann sich jedoch ändern, wenn der Code in proprietärer Software enthalten ist.
- 2. Welche zwei der folgenden Antworten beschreiben Methoden, wie Red Hat Produkte für die Zukunft entwickelt und mit der Community interagiert? (Wählen Sie zwei Antworten aus.)
- a. Sponsoren und Integrieren von Open Source-Projekten in das Community-basierte Fedora-Projekt
  - b. Entwickeln spezifischer Integrationstools, die nur in Red Hat-Distributionen verfügbar sind
  - c. Beteiligen an Upstream-Projekten
  - d. Neuverpacken und Neulizenzierten von Community-Produkten
- 3. Welche zwei Aussagen beschreiben die Vorteile von Linux? (Wählen Sie zwei Antworten aus.)
- a. Linux wird ausschließlich von Freiwilligen entwickelt und ist somit ein kostengünstiges Betriebssystem.
  - b. Linux ist modular aufgebaut und kann als vollständiger grafischer Desktop oder als kleine Appliance konfiguriert werden.
  - c. Jedes Release von Linux bleibt mindestens ein Jahr lang auf einem bekannten Stand gesperrt, wodurch die Entwicklung kundenspezifischer Software vereinfacht wird.
  - d. Linux beinhaltet eine leistungsstarke und skriptfähige Befehlszeilenschnittstelle, die eine einfachere Automatisierung und Bereitstellung ermöglicht.

# Zusammenfassung

---

In diesem Kapitel erhielten Sie folgende Informationen:

- Open Source Software ist Software mit Quellcode, den jeder kostenlos verwenden, untersuchen, ändern und teilen kann.
- Eine Linux-Distribution ist ein installierbares Betriebssystem, das auf einem Linux-Kernel aufgebaut ist und Benutzerprogramme und -bibliotheken unterstützt.
- Red Hat beteiligt sich an der Unterstützung und Bereitstellung von Code für Open Source-Projekte, sponsert und integriert Projektsoftware in Community-Distributionen und stabilisiert die Software, um sie als unterstützte unternehmenstaugliche Produkte anzubieten.
- Red Hat Enterprise Linux ist die unternehmenstaugliche, kommerziell unterstützte Linux-Distribution.



## Kapitel 2

# Zugreifen auf die Befehlszeile

### Ziel

Anmelden bei einem Linux-System und Ausführen einfacher Befehle über die Shell

### Ziele

- Anmelden bei einem Linux-System auf einer lokalen Textkonsole und Ausführen einfacher Befehle über die Shell
- Anmelden mit der GNOME 3-Desktopumgebung bei einem Linux-System und Ausführen von Befehlen an einer Shell-Eingabeaufforderung in einem Terminalprogramm
- Zeitsparende Verwendung von Tab-Vervollständigung, Befehlsverlauf und Tastenkombinationen für die Befehlszeilenbearbeitung zum Ausführen von Befehlen in der Bash-Shell

### Abschnitte

- Zugreifen auf die Befehlszeile (und Test)
- Zugreifen auf die Befehlszeile über den Desktop (und angeleitete Übung)
- Ausführen von Befehlen an der Bash-Shell (und Test)

### Praktische Übung

Zugreifen auf die Befehlszeile

# Zugreifen auf die Befehlszeile

## Ziele

Nach Abschluss dieses Abschnittes sollten Sie in der Lage sein, sich bei einem Linux-System anzumelden und einfache Befehle über die Shell auszuführen.

## Einführung in die Bash-Shell

Eine **Befehlszeile** ist eine textbasierte Schnittstelle, an der Anweisungen in ein Computersystem eingegeben werden können. Die Linux-Befehlszeile wird von einem Programm, der **Shell**, bereitgestellt. Im Laufe der Jahre wurden verschiedene Optionen für das Shell-Programm entwickelt und unterschiedliche Benutzer können für die Verwendung verschiedener Shells konfiguriert werden. Die meisten Benutzer verwenden jedoch den aktuellen Standard.

Unter Red Hat Enterprise Linux ist die GNU Bourne-Again Shell (**bash**) die Standard-Shell für Benutzer. Bash ist eine verbesserte Version der Bourne Shell (**sh**), einer der erfolgreichsten auf UNIX-ähnlichen Systemen eingesetzten Shells.

Wenn eine Shell interaktiv verwendet wird, zeigt sie eine Zeichenfolge an, wenn sie auf einen Befehl des Benutzers wartet. Die Zeichenfolge wird als **Shell-Eingabeaufforderung** bezeichnet. Wenn ein regulärer Benutzer eine Shell startet, endet die Standardeingabeaufforderung mit einem **\$**-Zeichen (siehe unten).

```
[user@host ~]$
```

Das Zeichen **\$** wird durch das Zeichen **#** ersetzt, wenn die Shell als Superuser **root** ausgeführt wird. Dadurch wird offensichtlicher, dass es sich um eine Superuser-Shell handelt, und Verschen und Fehler, die das gesamte System beeinträchtigen können, werden vermieden. Die Superuser-Shell-Eingabeaufforderung ist unten dargestellt.

```
[root@host ~]#
```

Die Nutzung von **bash** zum Ausführen von Befehlen kann sehr leistungsstark sein. Die **bash**-Shell stellt eine Skripting-Sprache bereit, die die Automatisierung von Aufgaben unterstützt. Die Shell bietet weitere Funktionen, mit denen sich Vorgänge vereinfachen oder ermöglichen lassen, die mit grafischen Werkzeugen nur schwer durchführbar sind.



### Anmerkung

Das Konzept der **bash**-Shell gleicht dem Befehlszeilen-Interpreter aus neueren Versionen von Microsoft Windows, **cmd .exe**, auch wenn **bash** eine kompliziertere Skripting-Sprache aufweist. Sie gleicht auch Windows PowerShell in Windows 7 und Windows Server 2008 R2. Administratoren, die mit dem Apple Mac und dem Dienstprogramm Terminal vertraut sind, werden feststellen, dass **bash** auch bei macOS die Standard-Shell ist.

## Shell-Grundlagen

Befehle, die an der Shell-Eingabeaufforderung eingegeben werden, bestehen aus drei grundlegenden Teilen:

- Auszuführender Befehl
- Optionen zum Anpassen des Befehlsverhaltens
- Argumente, die in der Regel Ziele des Befehls sind

Der **Befehl** ist der Name des Programms, das ausgeführt werden soll. Ihm können eine oder mehrere **Optionen** folgen, die das Verhalten oder die Wirkungsweise des Befehls anpassen. Optionen beginnen normalerweise mit einem oder zwei Bindestrichen (z. B. **-a** oder **--all**), damit sie leichter von Argumenten zu unterscheiden sind. Befehle können auch von einem oder mehreren **Argumenten** gefolgt werden, die häufig angeben, worauf der Befehl angewendet werden soll.

Der Befehl **usermod -L user01** besteht beispielsweise aus einem Befehl (**usermod**), einer Option (**-L**) und einem Argument (**user01**). Mit diesem Befehl wird das Passwort für das Konto des Benutzers **user01** gesperrt.

## Anmelden bei einem lokalen Computer

Um die Shell auszuführen, müssen Sie sich beim Computer über ein *Terminal* anmelden. Ein Terminal ist eine textbasierte Schnittstelle zur Eingabe von Befehlen und zur Darstellung der Ausgabe von einem Computersystem. Es stehen dazu mehrere Möglichkeiten zur Verfügung.

Der Computer verfügt möglicherweise über eine Hardwaretastatur und ein Display für die Eingabe und Ausgabe, die direkt mit dem Computer verbunden sind. Dies ist die *physische Konsole* des Linux-Rechners. Die physische Konsole unterstützt mehrere *virtuelle Konsole*n, die separate Terminals ausführen können. Jede virtuelle Konsole unterstützt eine eigene Anmeldesitzung. Sie können zwischen ihnen umschalten, indem Sie **Strg+Alt** und eine Funktionstaste (**F1** bis **F6**) gleichzeitig drücken. Die meisten dieser virtuellen Konsole führen ein Terminal aus, das eine Text-Anmeldeeingabeaufforderung bereitstellt. Wenn Sie Ihren Benutzernamen und Ihr Passwort richtig eingeben, melden Sie sich an und erhalten eine Shell-Eingabeaufforderung.

Der Computer kann eine grafische Anmeldeeingabeaufforderung auf einer der virtuellen Konsole n bereitstellen. Damit können Sie sich bei einer *grafischen Umgebung* anmelden. Die grafische Umgebung wird auch auf einer virtuellen Konsole ausgeführt. Um eine Shell-Eingabeaufforderung zu erhalten, müssen Sie ein Terminalprogramm in der grafischen Umgebung starten. Die Shell-Eingabeaufforderung wird in einem Anwendungsfenster Ihres grafischen Terminalprogramms bereitgestellt.



### Anmerkung

Viele Systemadministratoren entscheiden sich dafür, keine grafische Umgebung auf ihren Servern auszuführen. Dadurch können Ressourcen, die von der grafischen Umgebung benötigt werden, stattdessen von den Serverservices verwendet werden.

Wenn die grafische Umgebung verfügbar ist, wird in Red Hat Enterprise Linux 8 der Anmeldebildschirm auf der ersten virtuellen Konsole, **tty1**, ausgeführt. Auf den virtuellen Konsole n zwei bis sechs stehen fünf weitere Text-Anmeldeeingabeaufforderungen zur Verfügung.

## Kapitel 2 | Zugreifen auf die Befehlszeile

Wenn Sie sich über den grafischen Anmeldebildschirm anmelden, wird Ihre grafische Umgebung auf der ersten virtuellen Konsole gestartet, die derzeit nicht von einer Anmeldesitzung verwendet wird. Normalerweise ersetzt Ihre grafische Sitzung die Anmeldeeingabeaufforderung auf der zweiten virtuellen Konsole (**tty2**). Wenn diese Konsole jedoch von einer aktiven Text-Anmeldesitzung verwendet wird (nicht nur einer Anmeldeeingabeaufforderung), wird stattdessen die nächste freie virtuelle Konsole verwendet.

Der grafische Anmeldebildschirm wird auf der ersten virtuellen Konsole (**tty1**) weiter ausgeführt. Wenn Sie bereits bei einer grafischen Sitzung angemeldet sind, melden Sie sich als anderer Benutzer auf dem grafischen Anmeldebildschirm an oder wechseln Sie über den Menüeintrag **Switch User** den Benutzer in der grafischen Umgebung, ohne sich abzumelden. Für diesen Benutzer wird eine andere grafische Umgebung auf der nächsten freien virtuellen Konsole gestartet.

Wenn Sie sich von einer grafischen Umgebung abmelden, wird sie beendet und die physische Konsole wechselt automatisch zum grafischen Anmeldebildschirm auf der ersten virtuellen Konsole.



### Anmerkung

In Red Hat Enterprise Linux 6 und 7 wird der grafische Anmeldebildschirm auf der ersten virtuellen Konsole ausgeführt. Wenn Sie sich jedoch anmelden, ersetzt Ihre ursprüngliche grafische Umgebung den Anmeldebildschirm auf der ersten virtuellen Konsole, anstatt auf einer neuen virtuellen Konsole zu starten.

In Red Hat Enterprise Linux 5 und früher haben die ersten sechs virtuellen Konsolen immer Text-Anmeldeeingabeaufforderungen bereitgestellt. Die grafische Umgebung wird auf der virtuellen Konsole *sieben* ausgeführt (darauf kann mit **Strg+Alt+F7** zugegriffen werden).

Ein *monitorloser Server* verfügt über keine fest verbundene Tastatur und kein fest verbundenes Display. In einem Rechenzentrum können viele Racks mit monitorlosen Servern enthalten sein. Wenn nicht jeder Server mit einer Tastatur und einem Display ausgestattet ist, werden Platz und Kosten gespart. Damit sich Administratoren anmelden können, verfügt ein monitorloser Server möglicherweise über eine von seiner *seriellen Konsole* bereitgestellte Anmeldeeingabeaufforderung. Diese Anmeldeeingabeaufforderung wird auf einem seriellen Port ausgeführt, der mit einem Netzwerkkonsolenserver verbunden ist, um den Remote-Zugriff auf die serielle Konsole zu ermöglichen.

Die serielle Konsole würde normalerweise verwendet werden, um den Server zu reparieren, wenn seine Netzwerkkarte falsch konfiguriert und die Anmeldung über seine eigene Netzwerkverbindung nicht möglich ist. Meistens wird jedoch auf monitorlosen Server über das Netzwerk auf andere Weise zugegriffen.

## Anmelden über das Netzwerk

Linux-Benutzer und -Administratoren benötigen häufig Zugriff auf ein Remote-System über die Shell, indem sie eine Verbindung über das Netzwerk herstellen. In einer modernen Computing-Umgebung sind viele monitorlose Server eigentlich virtuelle Rechner oder werden als Public oder Private Cloud-Instanzen ausgeführt. Diese Systeme sind nicht physisch vorhanden und haben keine echten Hardwarekonsolen. Sie bieten möglicherweise nicht einmal Zugriff auf ihre (simulierte) physische oder serielle Konsole.

Unter Linux ist der häufigste Weg, eine Shell-Eingabeaufforderung auf einem Remote-System abzurufen, die Verwendung von Secure Shell (SSH). Die meisten Linux-Systeme

## Kapitel 2 | Zugreifen auf die Befehlszeile

(einschließlich Red Hat Enterprise Linux) und macOS stellen für diesen Zweck das OpenSSH-Befehlszeilenprogramm **ssh** bereit.

In diesem Beispiel meldet sich ein Benutzer mit einer Shell-Eingabeaufforderung auf dem Rechner **host** mit **ssh** beim Remote-Linux-System **remotehost** als Benutzer **remoteuser** an:

```
[user@host ~]$ ssh remoteuser@remotehost  
remoteuser@remotehost's password: password  
[remoteuser@remotehost ~]$
```

Der Befehl **ssh** verschlüsselt die Verbindung, um die Kommunikation gegen Abhören oder Abgreifen der Passwörter und des Inhalts zu schützen.

Einige Systeme (z. B. neue Cloud-Instanzen) lassen aus Sicherheitsgründen nicht zu, dass Benutzer sich mit einem Passwort mit **ssh** anmelden. Eine alternative Methode zur Authentifizierung bei einem Remote-Rechner ohne Eingabe eines Passworts ist die *Authentifizierung per Public Key*.

Bei dieser Authentifizierungsmethode verfügen Benutzer über eine spezielle Identitätsdatei, die einen *Private Key* enthält, der einem Passwort entspricht und den sie geheim halten. Ihr Konto auf dem Server ist mit einem übereinstimmenden *Public Key* konfiguriert, der nicht geheim sein muss. Bei der Anmeldung können Benutzer **ssh** konfigurieren, um den Private Key bereitzustellen. Wenn ihr entsprechender Public Key in diesem Konto auf diesem Remote-Server installiert ist, werden sie angemeldet, ohne dass sie nach einem Passwort gefragt werden.

Im nächsten Beispiel meldet sich ein Benutzer mit einer Shell-Eingabeaufforderung auf dem Rechner **host** bei **remotehost** als **remoteuser** mit **ssh** und mit der Authentifizierung per Public Key an. Mit der Option **-i** wird die Private-Key-Datei, **mylab.pem**, des Benutzers angegeben. Der übereinstimmende Public Key ist bereits als autorisierter Schlüssel im **remoteuser**-Konto eingerichtet.

```
[user@host ~]$ ssh -i mylab.pem remoteuser@remotehost  
[remoteuser@remotehost ~]$
```

Damit das funktioniert, darf die Private-Key-Datei nur für den Benutzer lesbar sein, der die Datei besitzt. Im vorherigen Beispiel, bei dem sich der Private Key in der Datei **mylab.pem** befindet, könnte dies mit dem Befehl **chmod 600 mylab.pem** sichergestellt werden. Wie Sie Dateiberechtigungen festlegen, wird in einem späteren Kapitel ausführlicher beschrieben.

Benutzer könnten auch Private Keys konfiguriert haben, die automatisch ausprobiert werden. Dieses Thema geht jedoch über den Rahmen dieses Abschnitts hinaus. Unter „Referenzen“ am Ende dieses Abschnitts finden Sie Links zu weiteren Informationen zu diesem Thema.



### Anmerkung

Wenn Sie sich zum ersten Mal bei einem neuen Rechner anmelden, wird eine Warnmeldung von **ssh** darüber angezeigt, dass die Authentizität des Hosts nicht festgestellt kann:

```
[user@host ~]$ ssh -i mylab.pem remoteuser@remotehost
The authenticity of host 'remotehost (192.0.2.42)' can't be established.
ECDSA key fingerprint is 47:bf:82:cd:fa:68:06:ee:d8:83:03:1a:bb:29:14:a3.
Are you sure you want to continue connecting (yes/no)? yes
[remoteuser@remotehost ~]$
```

Jedes Mal, wenn Sie eine Verbindung mit einem Remote-Host über **ssh** herstellen, sendet der Remote-Host seinen *Hostschlüssel* an **ssh**, um sich zu authentifizieren und eine verschlüsselte Kommunikation einzurichten. Der Befehl **ssh** vergleicht diesen Schlüssel mit einer Liste gespeicherter Hostschlüssel, um sicherzustellen, dass er nicht geändert wurde. Wenn sich der Hostschlüssel geändert hat, weist dies möglicherweise darauf hin, dass jemand versucht vorzugeben, er wäre der Host, um die Verbindung abzufangen. Dies wird auch als Man-in-the-Middle-Angriff bezeichnet. In SSH schützen Hostschlüssel vor Man-in-the-Middle-Angriffen. Diese Hostschlüssel sind für jeden Server eindeutig und müssen regelmäßig sowie bei einer vermuteten Kompromittierung geändert werden.

Sie erhalten diese Warnung, wenn auf Ihrem lokalen Rechner kein Hostschlüssel für den Remote-Host gespeichert ist. Wenn Sie **yes** eingeben, wird der vom Remote-Host gesendete Hostschlüssel akzeptiert und zur späteren Verwendung gespeichert. Die Anmeldung wird fortgesetzt und diese Meldung sollte nicht wieder angezeigt werden, wenn Sie eine Verbindung zu diesem Host herstellen. Wenn Sie **no** eingeben, wird der Hostschlüssel abgelehnt und die Verbindung geschlossen.

Ist auf dem lokalen Rechner ein Hostschlüssel gespeichert, der nicht mit dem tatsächlich vom Remote-Host gesendeten Schlüssel übereinstimmt, wird die Verbindung automatisch mit einer Warnung geschlossen.

## Abmelden

Wenn Sie mit der Verwendung der Shell fertig sind und das Programm beenden möchten, haben Sie verschiedene Möglichkeiten, die Sitzung zu beenden. Sie können den Befehl **exit** eingeben, um die aktuelle Shell-Sitzung zu beenden. Alternativ können Sie eine Sitzung beenden, indem Sie **Strg+D** drücken.

Das folgende Beispiel zeigt einen Benutzer, der sich von einer SSH-Sitzung abmeldet:

```
[remoteuser@remotehost ~]$ exit
logout
Connection to remotehost closed.
[user@host ~]$
```



### Literaturhinweise

Manpages **intro(1)**, **bash(1)**, **console(4)**, **pts(4)**, **ssh(1)** und **ssh-keygen(1)**

Hinweis: Einige Details auf der Manpage **console(4)**, die **init(8)** und **inittab(5)** betreffen, sind veraltet.

Weitere Informationen zu OpenSSH und zur Authentifizierung mit Public Keys finden Sie im Kapitel *Using secure communications between two systems with OpenSSH* im Handbuch *Red Hat Enterprise Linux 8 Securing networks* unter [https://access.redhat.com/documentation/en-us/red\\_hat\\_enterprise\\_linux/8/html-single/securing\\_networks/index#using-secure-communications-between-two-systems-with-openssh\\_securing-networks](https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux/8/html-single/securing_networks/index#using-secure-communications-between-two-systems-with-openssh_securing-networks)



### Anmerkung

Anweisungen zum Lesen von **man**-Seiten und andere Online-Hilfedokumentationen finden Sie am Ende des nächsten Abschnitts.

## ► Quiz

# Zugreifen auf die Befehlszeile

Wählen Sie die richtige Antwort auf die folgenden Fragen aus:

- ▶ 1. Welcher Begriff beschreibt den Interpreter, der als Zeichenfolgen eingegebene Befehle ausführt?
  - a. Befehl
  - b. Konsole
  - c. Shell
  - d. Terminal
  
- ▶ 2. Welcher Begriff beschreibt das visuelle Zeichen, das angibt, dass eine interaktive Shell auf die Befehlseingabe durch einen Benutzer wartet?
  - a. Argument
  - b. Befehl
  - c. Option
  - d. Eingabeaufforderung
  
- ▶ 3. Welcher Begriff beschreibt den Namen eines Programms, das ausgeführt werden soll?
  - a. Argument
  - b. Befehl
  - c. Option
  - d. Eingabeaufforderung
  
- ▶ 4. Welcher Begriff beschreibt den Teil der Befehlszeile, der das Verhalten eines Befehls anpasst?
  - a. Argument
  - b. Befehl
  - c. Option
  - d. Eingabeaufforderung
  
- ▶ 5. Welcher Begriff beschreibt den Teil der Befehlszeile, der das Ziel angibt, für das der Befehl ausgeführt werden soll?
  - a. Argument
  - b. Befehl
  - c. Option
  - d. Eingabeaufforderung

- **6. Welcher Begriff beschreibt den Hardwarebildschirm und die -tastatur, die zur Interaktion mit dem System verwendet werden?**
- a. Physische Konsole
  - b. Virtuelle Konsole
  - c. Shell
  - d. Terminal
- **7. Welcher Begriff beschreibt eine der verschiedenen logischen Konsolen, die jeweils eine unabhängige Anmeldesitzung unterstützen?**
- a. Physische Konsole
  - b. Virtuelle Konsole
  - c. Shell
  - d. Terminal
- **8. Welcher Begriff beschreibt eine Schnittstelle, die einen Bildschirm für die Ausgabe und eine Tastatur für die Eingabe in einer Shell-Sitzung bereitstellt?**
- a. Konsole
  - b. Virtuelle Konsole
  - c. Shell
  - d. Terminal

## ► Lösung

# Zugreifen auf die Befehlszeile

Wählen Sie die richtige Antwort auf die folgenden Fragen aus:

- ▶ 1. Welcher Begriff beschreibt den Interpreter, der als Zeichenfolgen eingegebene Befehle ausführt?
  - a. Befehl
  - b. Konsole
  - c. Shell
  - d. Terminal
  
- ▶ 2. Welcher Begriff beschreibt das visuelle Zeichen, das angibt, dass eine interaktive Shell auf die Befehlseingabe durch einen Benutzer wartet?
  - a. Argument
  - b. Befehl
  - c. Option
  - d. Eingabeaufforderung
  
- ▶ 3. Welcher Begriff beschreibt den Namen eines Programms, das ausgeführt werden soll?
  - a. Argument
  - b. Befehl
  - c. Option
  - d. Eingabeaufforderung
  
- ▶ 4. Welcher Begriff beschreibt den Teil der Befehlszeile, der das Verhalten eines Befehls anpasst?
  - a. Argument
  - b. Befehl
  - c. Option
  - d. Eingabeaufforderung
  
- ▶ 5. Welcher Begriff beschreibt den Teil der Befehlszeile, der das Ziel angibt, für das der Befehl ausgeführt werden soll?
  - a. Argument
  - b. Befehl
  - c. Option
  - d. Eingabeaufforderung

- **6. Welcher Begriff beschreibt den Hardwarebildschirm und die -tastatur, die zur Interaktion mit dem System verwendet werden?**
- a. Physische Konsole
  - b. Virtuelle Konsole
  - c. Shell
  - d. Terminal
- **7. Welcher Begriff beschreibt eine der verschiedenen logischen Konsolen, die jeweils eine unabhängige Anmeldesitzung unterstützen?**
- a. Physische Konsole
  - b. Virtuelle Konsole
  - c. Shell
  - d. Terminal
- **8. Welcher Begriff beschreibt eine Schnittstelle, die einen Bildschirm für die Ausgabe und eine Tastatur für die Eingabe in einer Shell-Sitzung bereitstellt?**
- a. Konsole
  - b. Virtuelle Konsole
  - c. Shell
  - d. Terminal

# Zugreifen auf die Befehlszeile über den Desktop

## Ziele

Nach Abschluss dieses Abschnitts sollten Sie in der Lage sein, sich mit der GNOME 3-Desktopumgebung beim Linux-System anzumelden, um Befehle an einer Shell-Eingabeaufforderung in ein Terminalprogramm einzugeben.

## Einführung in die GNOME-Desktopumgebung

Die *Desktopumgebung* ist eine grafische Benutzeroberfläche auf einem Linux-System. Die Standard-Desktopumgebung in Red Hat Enterprise Linux 8 wird von GNOME 3 bereitgestellt. Sie bietet Benutzern einen integrierten Desktop und über die grafische Umgebung hinaus eine einheitliche Entwicklungsplattform entweder über Wayland (Standard) oder über das alte X Window System.

GNOME Shell bietet die wichtigsten Benutzeroberflächenfunktionen für die GNOME-Desktopumgebung. Die Anwendung GNOME Shell kann umfassend angepasst werden. In Red Hat Enterprise Linux 8 ist das Erscheinungsbild von GNOME Shell standardmäßig auf das Design „Standard“ eingestellt, das in diesem Abschnitt verwendet wird. In Red Hat Enterprise Linux 7 wurde standardmäßig ein alternatives Design mit dem Namen „Classic“ verwendet, das dem Erscheinungsbild älterer GNOME-Versionen näher kam. Beide Designs können dauerhaft bei der Anmeldung ausgewählt werden, indem Sie auf das Zahnradsymbol neben der Schaltfläche **Sign In** klicken, die nach Auswahl Ihres Benutzerkontos, aber vor der Eingabe Ihres Passworts verfügbar ist.

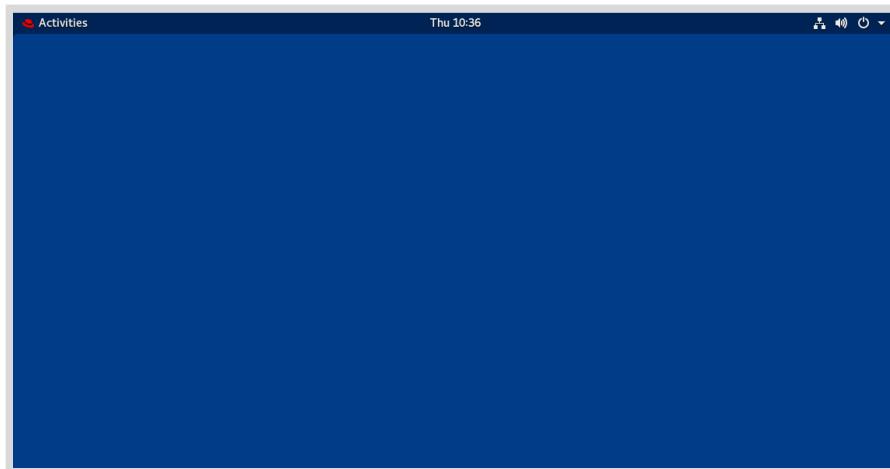
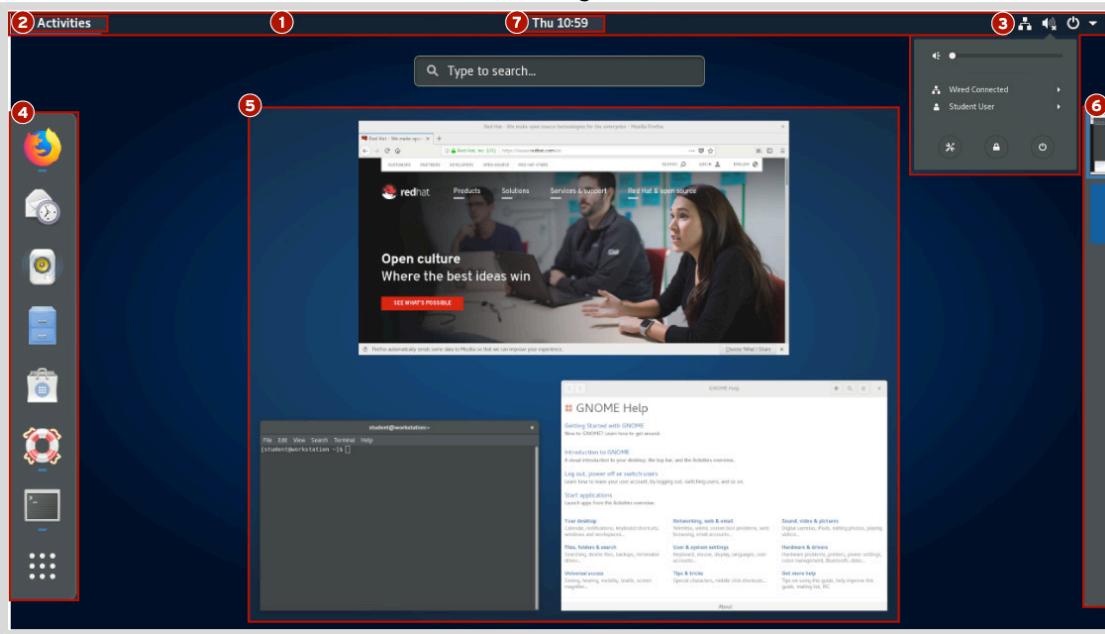


Abbildung 2.1: Ein leerer GNOME 3-Desktop

Wenn Sie sich das erste Mal als neuer Benutzer anmelden, wird ein anfängliches Setup-Programm ausgeführt, das Sie bei der Konfiguration der grundlegenden Benutzerkontoeinstellungen unterstützt. Anschließend wird die Anwendung GNOME Help auf dem Bildschirm **Getting Started with GNOME** gestartet. Dieser Bildschirm enthält Videos und Dokumentationen, die neuen Benutzern bei der Orientierung in der GNOME 3-Umgebung helfen. Sie können GNOME Help schnell starten, indem Sie auf der linken Seite der oberen Leiste auf **Activities** und in der angezeigten Dashboard-Leiste auf der linken Seite des Bildschirms auf das Rettungsring-Symbol klicken.

## Bestandteile der GNOME-Shell

Die Elemente der GNOME-Shell umfassen die folgenden Teile, wie in diesem Screenshot der GNOME-Shell im Übersichtsmodus „Activities“ dargestellt:



- ➊ **Obere Leiste:** Die Leiste, die am oberen Bildschirmrand angezeigt wird. Sie wird in der Übersicht „Activities“ und in Workspaces angezeigt. Die obere Leiste enthält die Schaltfläche **Activities** und steuert Volumen, Netzwerkfunktionen, Kalenderzugriff und Umschalten zwischen den Tastatureingabemethoden (falls mehrere konfiguriert sind).
- ➋ **Übersicht „Activities“:** In diesem besonderen Modus kann der Benutzer Fenster organisieren und Anwendungen starten. Die Übersicht „Activities“ kann durch Klicken auf die Schaltfläche **Activities** in der linken oberen Ecke der oberen Leiste oder durch Drücken der **Super-Taste** aufgerufen werden. Die **Super-Taste** (manchmal als **Windows-Taste** oder als **Befehlstaste** bezeichnet) befindet sich auf einer IBM PC 104/105- oder Apple-Tastatur in der Nähe der linken unteren Tastaturecke. Die drei Hauptbereiche der Übersicht „Activities“ sind die **Dashboard-Leiste** links auf dem Bildschirm, die **Fensterübersicht** in der Bildschirmitte und der **Workspace-Selektor** rechts auf dem Bildschirm.
- ➌ **Systemmenü:** Über das Menü in der rechten oberen Ecke können Sie die Helligkeit des Bildschirms anpassen und die Netzwerkverbindungen ein- oder ausschalten. Unter dem Untermenü für den Benutzernamen befinden sich Optionen zum Anpassen der Benutzerkontoeinstellungen und Abmelden vom System. Das Systemmenü enthält auch Schaltflächen zum Öffnen des Fensters **Settings**, zum Sperren des Bildschirms oder zum Herunterfahren des Systems.
- ➍ **Dashboard-Leiste:** Dies ist eine konfigurierbare Symboliste mit den bevorzugten Anwendungen des Benutzers, derzeit laufenden Anwendungen und einer **Raster-Schaltfläche** am Ende der Liste, mit der beliebige Anwendungen ausgewählt werden können. Starten Sie Anwendungen, indem Sie auf eines der Symbole klicken, oder verwenden Sie die Raster-Schaltfläche, um nach seltener verwendeten Anwendungen zu suchen. Die Dashboard-Leiste wird manchmal auch als **Dock** bezeichnet.
- ➎ **Fensterübersicht:** Ein Bereich in der Mitte der Aktivitätenübersicht, in dem Miniaturansichten aller im aktuellen Workspace aktiven Fenster angezeigt werden. Dadurch können Fenster in einem überladenen Workspace leichter in den Vordergrund geholt oder in einen anderen Workspace verschoben werden.

## Kapitel 2 | Zugreifen auf die Befehlszeile

- ⑥ **Workspace-Selektor:** Ein Bereich rechts neben der Aktivitätenübersicht, in dem Miniaturansichten aller aktiven Workspaces angezeigt werden. Hier können Workspaces ausgewählt und Fenster von einem Workspace in einen anderen verschoben werden.
- ⑦ **Meldungsleiste:** Über die Meldungsleiste können Benachrichtigungen geprüft werden, die von Anwendungen oder Systemkomponenten an GNOME gesendet werden. Wenn eine Benachrichtigung eingeht, wird sie in der Regel zuerst kurz als einzelne Zeile oben im Bildschirm angezeigt und ein dauerhaft blinkender Indikator wird in der Mitte der oberen Leiste neben der Uhr angezeigt, um den Benutzer darüber zu informieren, dass Benachrichtigungen eingegangen sind. Sie können die Meldungsleiste öffnen, um diese Benachrichtigungen zu prüfen, indem Sie auf die Uhr in der oberen Leiste klicken oder **Super+M** drücken. Die Meldungsleiste kann durch Klicken in die obere Leiste oder durch Drücken von **Esc** oder **Super+M** geschlossen werden.

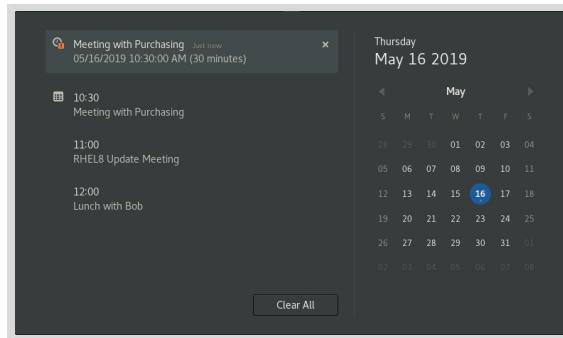


Abbildung 2.2: Nahaufnahme einer geöffnete Meldungsleiste

Sie können die von Ihrem Benutzerkonto verwendeten GNOME-Tastatkürzel anzeigen und bearbeiten. Öffnen Sie das Systemmenü auf der rechten Seite der oberen Leiste. Klicken Sie auf die Schaltfläche **Settings** am unteren Rand des Menüs auf der linken Seite. Wählen Sie im folgenden Anwendungsfenster aus dem linken Bereich **Devices → Keyboard** aus. Im rechten Bereich werden Ihre aktuellen Einstellungen für Tastenkombinationen angezeigt.



### Anmerkung

Einige Tastenkombinationen wie Funktionstasten oder die **Super**-Taste können möglicherweise nicht an einen virtuellen Rechner gesendet werden. Dies liegt daran, dass spezielle Tastatureingaben, die von diesen Tastenkombinationen verwendet werden, eventuell von Ihrem lokalen Betriebssystem oder von der Anwendung, mit der Sie auf den grafischen Desktop Ihres virtuellen Rechners zugreifen, abgefangen werden.



### Wichtig

In den aktuellen virtuellen und selbstgesteuerten Schulungsumgebungen von Red Hat kann die Verwendung der **Super**-Taste ein wenig schwierig sein. Sie können wahrscheinlich die **Super**-Taste Ihrer Tastatur nicht einfach verwenden, da sie häufig nicht von Ihrem Webbrowser an den virtuellen Rechner in der Kursumgebung übergeben wird.

Oben in Ihrem Browserfenster, das die Schnittstelle für den virtuellen Rechner anzeigt, sollte auf der rechten Seite ein Tastatursymbol vorhanden sein. Wenn Sie darauf klicken, wird eine Bildschirmtastatur angezeigt. Durch erneutes Klicken wird die Bildschirmtastatur wieder geschlossen.

Die Bildschirmtastatur behandelt **Super** als Umschalttaste, die häufig gedrückt gehalten wird, während eine andere Taste gedrückt wird. Wenn Sie einmal darauf klicken, wird sie gelb dargestellt, um anzusehen, dass die Taste gedrückt gehalten wird. Um also **Super+M** mit der Bildschirmtastatur einzugeben, klicken Sie auf **Super** und dann auf **M**.

Wenn Sie **Super** auf der Bildschirmtastatur nur kurz drücken und loslassen möchten, müssen Sie zweimal klicken. Der erste Klick „hält“ die **Super**-Taste gedrückt und der zweite Klick gibt sie wieder frei.

Die anderen Tasten, die von der Bildschirmtastatur als Umschalttasten behandelt werden (wie **Super**), sind **Umschalt**, **Strg**, **Alt** und die **Feststelltaste**. Die Tasten **Esc** und **Menü** werden wie normale Tasten behandelt und *nicht* wie Umschalttasten.

## Workspaces

Workspaces sind getrennte Desktop-Bildschirme, die unterschiedliche Anwendungsfenster enthalten. Damit kann die Arbeitsumgebung organisiert werden, indem geöffnete Anwendungsfenster nach Aufgabe gruppiert werden. So können zum Beispiel Fenster für eine bestimmte Systemwartungsaktivität (z. B. Einrichten eines neuen Remote-Servers) in einem Workspace gruppiert werden, während E-Mail- und andere Kommunikationsanwendungen einen anderen Workspace belegen.

Es gibt zwei einfache Methoden zum Wechseln zwischen Workspaces. Die vielleicht schnellste Methode ist das Drücken von **Strg+Alt+Pfeil nach oben** oder **Strg+Alt+Pfeil nach unten**, um sequenziell zwischen den Workspaces umzuschalten. Die zweite Methode besteht im Wechseln zur Übersicht **Activities**, in der Sie auf den gewünschten Workspace klicken können.

Ein Vorteil bei der Verwendung der Übersicht **Activities** ist, dass Sie auf die Fenster klicken und sie zwischen Workspaces verschieben können, indem Sie den **Workspace-Selektor** rechts auf dem Bildschirm und die **Fensterübersicht** in der Bildschirmmitte verwenden.



### Wichtig

Wie die **Super**-Taste werden in den aktuellen virtuellen und selbstgesteuerten Schulungsumgebungen von Red Hat die Tastenkombinationen **Strg+Alt** in der Regel nicht von Ihrem Webbrower an den virtuellen Rechner in der Kursumgebung übergeben.

Sie können diese Tastenkombinationen mit der Bildschirmstastatur eingeben, um zwischen Workspaces zu wechseln. Es müssen mindestens zwei Workspaces verwendet werden. Öffnen Sie die Bildschirmstastatur und klicken Sie auf **Strg, Alt** und dann entweder auf **Pfeil nach oben** oder **Pfeil nach unten**.

In diesen Schulungsumgebungen ist es jedoch im Allgemeinen einfacher, die Tastenkombinationen und die Bildschirmstastatur zu vermeiden. Wechseln Sie den Workspace, indem Sie auf die Schaltfläche **Activities** und dann im Workspace-Selektor rechts neben der Übersicht „Activities“ auf den Workspace klicken, zu dem Sie wechseln möchten.

## Starten eines Terminals

Um eine Shell-Eingabeaufforderung in GNOME zu erhalten, starten Sie eine grafische Terminal-Anwendung wie GNOME Terminal. Es stehen dazu mehrere Möglichkeiten zur Verfügung. Die zwei am häufigsten verwendeten Methoden sind unten aufgeführt:

- Wählen Sie in der **Aktivitätenübersicht** aus der **Dashboard**-Leiste die Option Terminal aus (entweder aus dem Favoritenbereich oder durch Suchen mit der **Raster**-Schaltfläche in der **Utilities**-Gruppe oder dem Suchfeld oben in der **Fensterübersicht**).
- Drücken Sie die Tastenkombination **Alt+F2**, um **Enter a Command** zu öffnen, und geben Sie **gnome-terminal** ein.

Wenn ein Terminalfenster geöffnet ist, wird eine Shell-Eingabeaufforderung für den Benutzer angezeigt, der das grafische Terminalprogramm gestartet hat. Die Shell-Eingabeaufforderung und die Titelleiste des Terminalfensters zeigen den aktuellen Benutzernamen, Hostnamen und das Arbeitsverzeichnis an.

## Sperren des Bildschirms oder Abmelden

Über das Systemmenü ganz rechts in der oberen Leiste können Sie den Bildschirm sperren oder sich vollständig abmelden.

Um den Bildschirm zu sperren, klicken Sie im Systemmenü in der rechten oberen Ecke auf die Sperr-Schaltfläche unten im Menü oder drücken Sie **Super+1** (was eventuell leichter zu merken ist als **Windows+1**). Der Bildschirm wird auch gesperrt, wenn die grafische Sitzung für einige Minuten inaktiv war.

Ein **Bildschirmsperrbild** wird angezeigt, das die Systemzeit und den Namen des angemeldeten Benutzers enthält. Um den Bildschirm zu entsperren, drücken Sie die **Eingabetaste** oder die **Leertaste**, um das Bildschirmsperrbild zu entfernen, und geben Sie dann das Benutzerpasswort in den **Sperrbildschirm** ein.

Um sich abzumelden und die aktuelle grafische Anmeldesitzung zu beenden, wählen Sie das Systemmenü in der rechten oberen Leiste aus und klicken auf **(Benutzer) → Log Out**. Ein Fenster wird angezeigt, das die Optionen zum Abbrechen (**Cancel**) oder Bestätigen der Abmeldung (**Log Out**) enthält.

## Abschalten oder Neubooten des Systems

Um das System herunterzufahren, klicken Sie im Systemmenü in der rechten oberen Ecke auf die Ein-/Aus-Schaltfläche unten im Menü oder drücken Sie **Strg+Alt+Entf**. Im angezeigten Dialogfeld können Sie auswählen, den Rechner abzuschalten (**Power Off**), neu zu starten (**Restart**) oder den Vorgang abzubrechen (**Cancel**). Wenn Sie hier keine Auswahl treffen, fährt das System automatisch nach 60 Sekunden herunter.



### Literaturhinweise

GNOME-Hilfe

- **yelp**  
GNOME Help: *Getting Started with GNOME*
- **yelp help:gnome-help/getting-started**

## ► Angeleitete Übung

# Zugreifen auf die Befehlszeile über den Desktop

In dieser Übung melden Sie sich über den grafischen Display-Manager als regulärer Benutzer an, um sich mit der von GNOME 3 bereitgestellten GNOME-Standard-Desktopumgebung vertraut zu machen.

### Ergebnisse

Sie sollten in der Lage sein, sich mit der GNOME 3-Desktopumgebung bei einem Linux-System anzumelden und Befehle an einer Shell-Eingabeaufforderung in einem Terminalprogramm auszuführen.

### Bevor Sie Beginnen

Prüfen Sie, ob der virtuelle Rechner **workstation** ausgeführt wird. Führen Sie die folgenden Aufgaben auf **workstation** aus.

- ▶ 1. Melden Sie sich als Benutzer **student** mit dem Passwort **student** bei **workstation** an.
  - 1.1. Klicken Sie auf **workstation** im GNOME-Anmeldebildschirm auf das Benutzerkonto **student**. Geben Sie **student** ein, wenn Sie zur Passworteingabe aufgefordert werden.
  - 1.2. Klicken Sie auf **Sign In**.
- ▶ 2. Ändern Sie das Passwort für **student** von **student** in **55TurnK3y**.



#### Wichtig

Das Skript **finish** setzt das Passwort für den Benutzer **student** auf **student** zurück. Das Skript muss am Ende der Übung ausgeführt werden.

- 2.1. Am einfachsten öffnen Sie dafür das Terminal-Fenster und führen den Befehl **passwd** an der Shell-Eingabeaufforderung aus.  
Drücken Sie in der virtuellen Lernumgebung mit visueller Tastatur die **Super**-Taste zweimal, um die Übersicht **Activities** zu öffnen. Geben Sie **terminal** ein und drücken Sie dann die **Eingabetaste**, um das Terminal zu starten.
- 2.2. Ein Terminalfenster wird geöffnet. Geben Sie **passwd** an der Shell-Eingabeaufforderung ein. Ändern Sie das Passwort für „student“ von **student** in **55TurnK3y**.

## Kapitel 2 | Zugreifen auf die Befehlszeile

```
[student@workstation ~]$ passwd  
Changing password for user student.  
Current password: student  
New password: 55TurnK3y  
Retype new password: 55TurnK3y  
passwd: all authentication tokens updated successfully.
```

- ▶ **3.** Melden Sie sich ab und wieder als **student** mit dem Passwort **55TurnK3y** an, um das geänderte Passwort zu bestätigen.
  - 3.1. Klicken Sie auf das Systemmenü in der rechten oberen Ecke.
  - 3.2. Wählen Sie **Student User → Log Out** aus.
  - 3.3. Klicken Sie im angezeigten Bestätigungsdialogfeld auf **Log Out**.
  - 3.4. Klicken Sie im GNOME-Anmeldebildschirm auf das Benutzerkonto **student**. Wenn Sie aufgefordert werden, das Passwort einzugeben, geben Sie **55TurnK3y** ein.
  - 3.5. Klicken Sie auf **Sign In**.
- ▶ **4.** Sperren Sie den Bildschirm.
  - 4.1. Drücken Sie im Systemmenü in der rechten oberen Ecke auf die Sperr-Schaltfläche unten im Menü.
- ▶ **5.** Entsperren Sie den Bildschirm.
  - 5.1. Drücken Sie die **Eingabetaste**, um das Bildschirmsperrbild zu entfernen.
  - 5.2. Geben Sie im Feld **Password 55TurnK3y** als Passwort ein.
  - 5.3. Klicken auf **Unlock**.
- ▶ **6.** Legen Sie fest, wie **workstation** über die grafische Schnittstelle heruntergefahren werden soll, brechen Sie den Vorgang aber mit **Cancel** ab, d. h., fahren Sie das System nicht herunter.
  - 6.1. Drücken Sie im Systemmenü in der rechten oberen Ecke auf die Ein-/Aus-Schaltfläche unten im Menü. Ein Dialogfeld mit den Optionen zum Neustarten (**Restart**) oder Ausschalten (**Power Off**) des Rechners wird angezeigt.
  - 6.2. Klicken Sie im angezeigten Dialogfeld auf **Cancel**.

## Beenden

Führen Sie auf **workstation** das Skript **lab cli-desktop finish** aus, um diese Übung zu beenden.

```
[student@workstation ~]$ lab cli-desktop finish
```

Hiermit ist die angeleitete Übung beendet.

# Ausführen von Befehlen an der Bash-Shell

## Ziele

Nach Abschluss dieses Abschnitts sollten Sie in der Lage sein, Befehle an einer Shell-Eingabeaufforderung mit Bash-Tastenkombinationen auszuführen.

## Grundlegende Befehlssyntax

Die GNU Bourne-Again Shell (**bash**) ist ein Programm, das vom Benutzer eingegebene Befehle interpretiert. Jede an der Shell eingegebene Zeichenfolge kann aus bis zu drei Teilen bestehen: dem Befehl, den Optionen (die normalerweise mit - oder -- beginnen) und den Argumenten. Jedes an der Shell eingegebene Wort wird mit Leerzeichen von den anderen getrennt. Befehle sind Namen der Programme, die auf dem System installiert sind. Jeder Befehl hat eigenen Optionen und Argumente.

Wenn Sie zur Ausführung eines Befehls bereit sind, drücken Sie die **Eingabetaste**. Geben Sie jeden Befehl in einer separaten Zeile ein. Die Befehlausgabe wird angezeigt, bevor die nächste Shell-Eingabeaufforderung erscheint.

```
[user@host]$ whoami  
user  
[user@host]$
```

Wenn Sie mehr als einen Befehl in eine Zeile eingeben möchten, verwenden Sie ein Semikolon (;) als Trennzeichen. Ein Semikolon gehört zu einer Klasse von Zeichen, die als *Metazeichen* bezeichnet werden und die in **bash** eine besondere Bedeutung haben. In diesem Fall wird die Ausgabe von beiden Befehlen angezeigt, bevor die nächste Shell-Eingabeaufforderung erscheint.

Das folgende Beispiel zeigt, wie zwei Befehle (**command1** und **command2**) in der Befehlszeile kombiniert werden.

```
[user@host]$ command1;command2
```

## Beispiele für einfache Befehle

Der Befehl **date** zeigt das aktuelle Datum und die aktuelle Uhrzeit an. Er kann auch vom Superuser verwendet werden, um die Systemuhr einzustellen. Ein Argument, das mit einem Pluszeichen (+) beginnt, gibt eine Formatzeichenfolge für den date-Befehl an.

```
[user@host ~]$ date  
Sat Jan 26 08:13:50 IST 2019  
[user@host ~]$ date +%R  
08:13  
[user@host ~]$ date +%x  
01/26/2019
```

## Kapitel 2 | Zugreifen auf die Befehlszeile

Der Befehl **passwd** ändert das Passwort des Benutzers. Das ursprüngliche Passwort für das Konto muss angegeben werden, damit die Änderung zulässig ist. Standardmäßig ist **passwd** so konfiguriert, dass ein sicheres Passwort, bestehend aus Klein- und Großbuchstaben, Zahlen und Symbolen gefordert wird, das nicht auf einem Wörterbuchwort basiert. Der Superuser kann den Befehl **passwd** zum Ändern der Passwörter anderer Benutzer verwenden.

```
[user@host ~]$ passwd
Changing password for user user.
Current password: old_password
New password: new_password
Retype new password: new_password
passwd: all authentication tokens updated successfully.
```

Linux benötigt keine Dateinamenerweiterungen, um Dateien nach Typ zu klassifizieren. Mit dem Befehl **file** wird der Anfang des Dateiinhalts gescannt und angezeigt, um welchen Typ es sich handelt. Die zu klassifizierenden Dateien werden als Argumente an den Befehl übergeben.

```
[user@host ~]$ file /etc/passwd
/etc/passwd: ASCII text
[user@host ~]$ file /bin/passwd
/bin/passwd: setuid ELF 64-bit LSB shared object, x86-64, version 1
(SYSV), dynamically linked, interpreter /lib64/ld-linux-x86-64.so.2,
for GNU/Linux 3.2.0, BuildID[sha1]=a3637110e27e9a48dcfd9f38b4ae43388d32d0e4,
stripped
[user@host ~]$ file /home
/home: directory
```

## Anzeigen des Inhalts von Dateien

Einer der einfachsten und am häufigsten verwendeten Befehle in Linux ist **cat**. Mit dem Befehl **cat** können Sie einzelne oder mehrere Dateien erstellen, den Inhalt von Dateien anzeigen, den Inhalt mehrerer Dateien verketten und den Inhalt der Datei an ein Terminal oder an Dateien umleiten.

Das Beispiel demonstriert, wie der Inhalt der Datei **/etc/passwd** angezeigt wird.

```
[user@host ~]$ cat /etc/passwd
root:x:0:0:root:/root:/bin/bash
bin:x:1:1:bin:/bin:/sbin/nologin
daemon:x:2:2:daemon:/sbin:/sbin/nologin
adm:x:3:4:adm:/var/adm:/sbin/nologin
...output omitted...
```

Verwenden Sie den folgenden Befehl, um den Inhalt mehrerer Dateien anzuzeigen.

```
[user@host ~]$ cat file1 file2
Hello World!!
Introduction to Linux commands.
```

Einige Dateien sind sehr lang und können mehr Platz zum Anzeigen in Anspruch nehmen als das Terminal bereitstellt. Der Befehl **cat** zeigt den Inhalt einer Datei nicht als Seiten an. Mit dem Befehl **less** können Sie jeweils eine Seite einer Datei anzeigen und nach Belieben scrollen.

## Kapitel 2 | Zugreifen auf die Befehlszeile

Mit dem Befehl **less** scrollen Sie seitenweise vorwärts und rückwärts durch Dateien die länger sind, als in ein Terminalfenster passt. Verwenden Sie die Tasten **Pfeil nach oben** und **Pfeil nach unten**, um nach oben und unten zu scrollen. Drücken Sie **q**, um den Befehl zu beenden.

Die Befehle **head** und **tail** zeigen den Anfang bzw. das Ende einer Datei an. Standardmäßig werden mit diesen Befehlen 10 Zeilen einer Datei angezeigt, aber mit der Option **-n** kann eine andere Zeilenzahl angegeben werden. Die anzuseigende Datei wird als Argument an diese Befehle übergeben.

```
[user@host ~]$ head /etc/passwd
root:x:0:0:root:/root:/bin/bash
bin:x:1:1:bin:/bin:/sbin/nologin
daemon:x:2:2:daemon:/sbin:/sbin/nologin
adm:x:3:4:adm:/var/adm:/sbin/nologin
lp:x:4:7:lp:/var/spool/lpd:/sbin/nologin
sync:x:5:0:sync:/sbin:/bin/sync
shutdown:x:6:0:shutdown:/sbin:/sbin/shutdown
halt:x:7:0:halt:/sbin:/sbin/halt
mail:x:8:12:mail:/var/spool/mail:/sbin/nologin
operator:x:11:0:operator:/root:/sbin/nologin
[user@host ~]$ tail -n 3 /etc/passwd
gdm:x:42:42::/var/lib/gdm:/sbin/nologin
gnome-initial-setup:x:977:977::/run/gnome-initial-setup/:/sbin/nologin
avahi:x:70:70:Avahi mDNS/DNS-SD Stack:/var/run/avahi-daemon:/sbin/nologin
```

Mit dem Befehl **wc** werden Zeilen, Wörter und Zeichen in einer Datei gezählt. Es kann die Option **-l**, **-w** oder **-c** hinzugefügt werden, wenn nur Zeilen, Wörter bzw. Zeichen angezeigt werden sollen.

```
[user@host ~]$ wc /etc/passwd
 45 102 2480 /etc/passwd
[user@host ~]$ wc -l /etc/passwd ; wc -l /etc/group
45 /etc/passwd
70 /etc/group
[user@host ~]$ wc -c /etc/group /etc/hosts
 966 /etc/group
 516 /etc/hosts
1482 total
```

## Tab-Vervollständigung

Mit der *Tab-Vervollständigung* kann der Benutzer Befehle oder Dateinamen schnell ergänzen, sobald genügend Text an der Eingabeaufforderung eingegeben wurde, um sie eindeutig zu identifizieren. Wenn die eingegebenen Zeichen nicht eindeutig sind, drücken Sie zweimal die **Tabulatortaste**, um alle Befehle anzuzeigen, die mit den bereits eingegebenen Zeichen beginnen.

```
[user@host ~]$ pas① Tab+Tab
passwd      paste      pasuspender
[user@host ~]$ pass② Tab
[user@host ~]$ passwd
Changing password for user user.
Current password:
```

## Kapitel 2 | Zugreifen auf die Befehlszeile

- ① Drücken Sie **Tab** zweimal.
- ② Drücken Sie **Tab** einmal.

Die Tab-Vervollständigung kann zum Ergänzen von Dateinamen verwendet werden, die als Argumente für Befehle eingegeben werden. Wenn **Tabulatortaste** gedrückt wird, wird der Dateiname soweit wie möglich ergänzt. Beim zweiten Drücken der **Tabulatortaste** listet die Shell alle Dateien auf, die dem bisherigen Zeichenmuster entsprechen. Geben Sie weitere Zeichen ein, bis der Name eindeutig ist, und verwenden Sie die Tab-Vervollständigung, um den Befehl zu vervollständigen.

```
[user@host ~]$ ls /etc/pas①Tab
[user@host ~]$ ls /etc/passwd②Tab
passwd  passwd-
```

- ① ② Drücken Sie **Tab** einmal.

Argumente und Optionen können mit der Tab-Vervollständigung für zahlreiche Befehle abgeglichen werden. Der Befehl **useradd** wird vom Superuser, **root**, verwendet, um weitere Benutzer im System zu erstellen. Er hat zahlreiche Optionen, mit denen sich steuern lässt, wie sich der Befehl verhält. Nach der teilweisen Eingabe einer Option kann die Option mit der Tab-Vervollständigung vervollständigt werden. So müssen Sie weniger Zeichen eingeben.

```
[root@host ~]# useradd --①Tab+Tab
--base-dir      --groups          --no-log-init    --shell
--comment       --help            --non-unique     --skel
--create-home   --home-dir        --no-user-group --system
--defaults      --inactive        --password      --uid
--expiredate   --key             --root          --user-group
--gid           --no-create-home --selinux-user
[root@host ~]# useradd --
```

- ① Drücken Sie **Tab** zweimal.

## Fortsetzen eines langen Befehls in einer anderen Zeile

Befehle mit vielen Optionen und Argumenten können schnell lang werden und werden automatisch vom Befehlsfenster unterbrochen, wenn der Cursor den rechten Rand erreicht. Um die Lesbarkeit von Befehlen zu erleichtern, können Sie stattdessen einen langen Befehl mit mehr als einer Zeile eingeben.

Dazu verwenden Sie einen Backslash (\), der als *Escape-Zeichen* bezeichnet wird, um die Bedeutung des Zeichens unmittelbar nach dem Backslash zu ignorieren. Sie haben gelernt, dass die Eingabe eines Zeilenumbruchzeichens durch Drücken der **Eingabetaste** der Shell mitteilt, dass die Befehlseingabe abgeschlossen ist und der Befehl ausgeführt werden soll. Wenn Sie das Zeilenumbruchzeichen maskieren, wird die Shell angewiesen, in eine neue Befehlszeile zu wechseln, ohne den Befehl auszuführen. Die Shell bestätigt die Anforderung, indem sie ein Fortsetzungszeichen anzeigt, das als *sekundäre Eingabeaufforderung* bezeichnet wird und standardmäßig als das Größer-als-Zeichen (>) in einer leeren neuen Zeile angezeigt wird. Befehle können über viele Zeilen fortgesetzt werden.

```
[user@host]$ head -n 3 \
> /usr/share/dict/words \
> /usr/share/dict/linux.words
==> /usr/share/dict/words <=
```

## Kapitel 2 | Zugreifen auf die Befehlszeile

```
1080
10-point
10th

==> /usr/share/dict/linux.words <=
1080
10-point
10th
[user@host ~]$
```



### Wichtig

Das vorherige Bildschirmbeispiel zeigt, wie ein fortgesetzter Befehl einem typischen Benutzer angezeigt wird. Die Darstellung dieses Zeichens in Lernmaterialien, wie diesem Kursbuch, führt jedoch häufig zu Verwirrung. Neue Lernende können das zusätzliche Größer-als-Zeichen versehentlich als Teil des eingegebenen Befehls einfügen. Die Shell interpretiert ein eingegebenes Größer-als-Zeichen als *Prozessumleitung*, was der Benutzer nicht beabsichtigt hat. Die Prozessumleitung wird in einem später Kapitel behandelt.

Um diese Verwirrung zu vermeiden, werden in diesem Kursbuch keine sekundären Eingabeaufforderungen in den Bildschirmausgaben gezeigt. Ein Benutzer sieht die sekundäre Eingabeaufforderung weiterhin in seinem Shell-Fenster, das Kursmaterial zeigt jedoch absichtlich nur die einzugebenden Zeichen, wie im folgenden Beispiel gezeigt. Vergleichen Sie es mit dem vorherigen Bildschirmbeispiel.

```
[user@host]$ head -n 3 \
/usr/share/dict/words \
/usr/share/dict/linux.words
==> /usr/share/dict/words <=
1080
10-point
10th

==> /usr/share/dict/linux.words <=
1080
10-point
10th
[user@host ~]$
```

## Befehlsverlauf

Mit dem Befehl **history** wird eine Liste der zuvor ausgeführten Befehle angezeigt, denen eine Befehlsnummer vorangestellt ist.

Das Ausrufezeichen (!) ist ein Metazeichen, das zum Erweitern früherer Befehle dient, ohne sie erneut eingeben zu müssen. Mit dem Befehl **!number** wird der Befehl erweitert, der der angegebenen Nummer entspricht. Mit dem Befehl **!string** wird der letzte Befehl erweitert, der der angegebenen Zeichenfolge entspricht.

```
[user@host ~]$ history
...output omitted...
23 clear
```

```

24 who
25 pwd
26 ls /etc
27 uptime
28 ls -l
29 date
30 history
[user@host ~]$ !ls
ls -l
total 0
drwxr-xr-x. 2 user user 6 Mar 29 21:16 Desktop
...output omitted...
[user@host ~]$ !26
ls /etc
abrt hosts pulse
adjtime hosts.allow purple
aliases hosts.deny qemu-ga
...output omitted...

```

Mit den Pfeiltasten können Sie durch vorherige Befehle im Shell-Verlauf navigieren. Mit **Pfeil nach oben** bearbeiten Sie den vorherigen Befehl in der Verlaufsliste. Mit **Pfeil nach unten** bearbeiten Sie den nächsten Befehl in der Verlaufsliste. Mit **Pfeil nach links** und **Pfeil nach rechts** bewegen Sie den Cursor im aktuellen Befehl aus der Verlaufsliste nach links bzw. rechts, damit Sie ihn bearbeiten können, bevor Sie ihn ausführen.

Sie können entweder mit der Tastenkombination **Esc+.** oder **Alt+.** das letzte Wort des vorherigen Befehls an der aktuellen Cursorposition einfügen. Durch wiederholte Verwendung der Tastenkombination wird dieser Text durch das letzte Wort noch früherer Befehle im Verlauf ersetzt. Die Tastenkombination **Alt+.** ist besonders praktisch, da Sie **Alt** gedrückt halten und **.** wiederholt drücken können, um weitere frühere Befehle einfach zu durchlaufen.

## Bearbeiten der Befehlszeile

Bei interaktiver Verwendung hat die **bash** eine Funktion zur Befehlszeilenbearbeitung. Damit kann der Benutzer Texteditorbefehle verwenden, um durch die aktuell eingegebene Befehlszeile zu navigieren und sie zu ändern. Mit den Pfeiltasten können Sie durch den aktuellen Befehl und den Befehlsverlauf navigieren, der an früherer Stelle in dieser Sitzung eingegeben wurde. Die folgende Tabelle enthält leistungsstärkere Bearbeitungsbefehle.

### Nützliche Tastenkombinationen für die Befehlszeilenbearbeitung

Tastenkombination	Beschreibung
<b>Strg+A</b>	An den Anfang der Befehlszeile springen
<b>Strg+E</b>	An das Ende der Befehlszeile springen
<b>Strg+U</b>	Ab der Cursorposition bis zum Beginn der Befehlszeile löschen
<b>Strg+K</b>	Ab der Cursorposition bis zum Ende der Befehlszeile löschen
<b>Strg+Pfeil nach links</b>	Zum Anfang des vorherigen Worts in der Befehlszeile springen

Tastenkombination	Beschreibung
<b>Strg+Pfeil nach rechts</b>	Zum Ende des nächsten Worts in der Befehlszeile springen
<b>Strg+R</b>	Verlaufsliste der Befehle nach einem Muster durchsuchen

Es gibt weitere Befehle für die Befehlszeilenbearbeitung, aber diese sind für neue Benutzer die nützlichsten. Die weiteren Befehle finden Sie auf der Manpage **bash(1)**.



#### Literaturhinweise

Manpages **bash(1)**, **date(1)**, **file(1)**, **cat(1)**, **more(1)**, **less(1)**, **head(1)**, **passwd(1)**, **tail(1)** und **wc(1)**

## ► Quiz

# Ausführen von Befehlen an der Bash-Shell

Wählen Sie die richtigen Antworten auf die folgenden Fragen aus:

- ▶ 1. Welche Bash-Tastenkombination oder welcher Bash-Befehl springt zum Anfang des vorherigen Worts in der Befehlszeile?
  - a. Drücken von **Strg+Pfeil nach links**
  - b. Drücken von **Strg+K**
  - c. Drücken von **Strg+A**
  - d. **!string**
  - e. **!number**
  
- ▶ 2. Welche Bash-Tastenkombination oder welcher Bash-Befehl trennt Befehle in derselben Zeile?
  - a. Drücken von **Tab**
  - b. **history**
  - c. ;
  - d. **!string**
  - e. Drücken von **Esc+**.
  
- ▶ 3. Mit welcher Bash-Tastenkombination oder welchem Bash-Befehl werden Zeichen vom Cursor bis zum Ende der Befehlszeile gelöscht?
  - a. Drücken von **Strg+Pfeil nach links**
  - b. Drücken von **Strg+K**
  - c. Drücken von **Strg+A**
  - d. ;
  - e. Drücken von **Esc+**.
  
- ▶ 4. Mit welcher Tastenkombination oder welchem Bash-Befehl wird ein kürzlich ausgeführter Befehl anhand des übereinstimmenden Befehlsnamen erneut ausgeführt?
  - a. Drücken von **Tab**
  - b. **!number**
  - c. **!string**
  - d. **history**
  - e. Drücken von **Esc+**.

► 5. Mit welcher Tastenkombination oder welchem Bash-Befehl werden Befehle, Dateinamen und Optionen vervollständigt?

- a. ;
- b. **!number**
- c. **history**
- d. Drücken von **Tab**
- e. Drücken von **Esc+ .**

► 6. Welche Bash-Tastenkombination oder welcher Bash-Befehl führt einen bestimmten Befehl in der Verlaufsliste erneut aus?

- a. Drücken von **Tab**
- b. **!number**
- c. **!string**
- d. **history**
- e. Drücken von **Esc+ .**

► 7. Welche Bash-Tastenkombination oder welcher Bash-Befehl springt zum Anfang der Befehlszeile?

- a. **!number**
- b. **!string**
- c. Drücken von **Strg+Pfeil nach links**
- d. Drücken von **Strg+K**
- e. Drücken von **Strg+A**

► 8. Welche Bash-Tastenkombination oder welcher Bash-Befehl zeigt die Liste der vorherigen Befehle an?

- a. Drücken von **Tab**
- b. **!string**
- c. **!number**
- d. **history**
- e. Drücken von **Esc+ .**

► 9. Welche Bash-Tastenkombination oder welcher Bash-Befehl kopiert das letzte Argument von vorherigen Befehlen?

- a. Drücken von **Strg+K**
- b. Drücken von **Strg+A**
- c. **!number**
- d. Drücken von **Esc+ .**

## ► Lösung

# Ausführen von Befehlen an der Bash-Shell

Wählen Sie die richtigen Antworten auf die folgenden Fragen aus:

- 1. Welche Bash-Tastenkombination oder welcher Bash-Befehl springt zum Anfang des vorherigen Worts in der Befehlszeile?
- a. Drücken von **Strg+Pfeil nach links**
  - b. Drücken von **Strg+K**
  - c. Drücken von **Strg+A**
  - d. **!string**
  - e. **!number**
- 2. Welche Bash-Tastenkombination oder welcher Bash-Befehl trennt Befehle in derselben Zeile?
- a. Drücken von **Tab**
  - b. **history**
  - c. ;
  - d. **!string**
  - e. Drücken von **Esc+**.
- 3. Mit welcher Bash-Tastenkombination oder welchem Bash-Befehl werden Zeichen vom Cursor bis zum Ende der Befehlszeile gelöscht?
- a. Drücken von **Strg+Pfeil nach links**
  - b. Drücken von **Strg+K**
  - c. Drücken von **Strg+A**
  - d. ;
  - e. Drücken von **Esc+**.
- 4. Mit welcher Tastenkombination oder welchem Bash-Befehl wird ein kürzlich ausgeführter Befehl anhand des übereinstimmenden Befehlsnamen erneut ausgeführt?
- a. Drücken von **Tab**
  - b. **!number**
  - c. **!string**
  - d. **history**
  - e. Drücken von **Esc+**.

► 5. Mit welcher Tastenkombination oder welchem Bash-Befehl werden Befehle, Dateinamen und Optionen vervollständigt?

- a. ;
- b. *!number*
- c. history
- d. Drücken von Tab
- e. Drücken von Esc+ .

► 6. Welche Bash-Tastenkombination oder welcher Bash-Befehl führt einen bestimmten Befehl in der Verlaufsliste erneut aus?

- a. Drücken von Tab
- b. *!number*
- c. *!string*
- d. history
- e. Drücken von Esc+ .

► 7. Welche Bash-Tastenkombination oder welcher Bash-Befehl springt zum Anfang der Befehlszeile?

- a. *!number*
- b. *!string*
- c. Drücken von Strg+Pfeil nach links
- d. Drücken von Strg+K
- e. Drücken von Strg+A

► 8. Welche Bash-Tastenkombination oder welcher Bash-Befehl zeigt die Liste der vorherigen Befehle an?

- a. Drücken von Tab
- b. *!string*
- c. *!number*
- d. history
- e. Drücken von Esc+ .

► 9. Welche Bash-Tastenkombination oder welcher Bash-Befehl kopiert das letzte Argument von vorherigen Befehlen?

- a. Drücken von Strg+K
- b. Drücken von Strg+A
- c. *!number*
- d. Drücken von Esc+ .

## ► Praktische Übung

# Zugreifen auf die Befehlszeile

### Leistungscheckliste

In dieser praktischen Übung verwenden Sie die Bash-Shell, um Befehle auszuführen.

### Ergebnisse

- Erfolgreiches Ausführen einfacher Programme über die Bash-Shell-Befehlszeile
- Ausführen von Befehlen zum Identifizieren von Dateitypen und Anzeigen von Teilen von Textdateien
- Verwenden von „Tastenkombinationen“ für den Bash-Befehlsverlauf, um Befehle oder Teile von Befehlen effizienter zu wiederholen

### Bevor Sie Beginnen

Melden Sie sich als Benutzer **student** mit dem Passwort **student** bei **workstation** an.

Führen Sie auf **workstation** das Skript **lab cli-review start** aus, um eine bereinigte Übungsumgebung einzurichten. Das Skript kopiert auch die Datei **zcat** in das Benutzerverzeichnis von **student**.

```
[student@workstation ~]$ lab cli-review start
```

1. Zeigen Sie mit dem Befehl **date** die aktuelle Uhrzeit und das aktuelle Datum an.
2. Zeigen Sie die aktuelle Uhrzeit im 12-Stunden-Format an (zum Beispiel: 11:42:11 AM). Tipp: Die Formatzeichenfolge zur Anzeige dieser Ausgabe ist **%r**.
3. Was für eine Art Datei ist **/home/student/zcat**? Ist sie für Menschen lesbar?
4. Verwenden Sie den Befehl **wc** und Bash-Tastenkombinationen, um die Größe von **zcat** anzuzeigen.
5. Zeigen Sie die ersten 10 Zeilen von **zcat** an.
6. Zeigen Sie die letzten 10 Zeilen der Datei **zcat** an.
7. Wiederholen Sie den vorherigen Befehl exakt mit drei oder weniger Tastenanschlägen.
8. Wiederholen Sie den vorherigen Befehl, verwenden Sie jedoch die Option **-n 20** zum Anzeigen der letzten 20 Zeilen in der Datei. Verwenden Sie die Befehlszeilenbearbeitung, um dies mit möglichst wenigen Tastenanschlägen zu erreichen.
9. Verwenden Sie den Shell-Verlauf, um den Befehl **date +%r** erneut auszuführen.

### Bewertung

Führen Sie auf **workstation** das Skript **lab cli-review grade** aus, um den Erfolg dieser Übung zu bestätigen.

```
[student@workstation ~]$ lab cli-review grade
```

## Beenden

Führen Sie auf workstation das Skript **lab cli-review finish** aus, um die praktische Übung zu beenden.

```
[student@workstation ~]$ lab cli-review finish
```

Damit ist die praktische Übung abgeschlossen.

## ► Lösung

# Zugreifen auf die Befehlszeile

### Leistungscheckliste

In dieser praktischen Übung verwenden Sie die Bash-Shell, um Befehle auszuführen.

### Ergebnisse

- Erfolgreiches Ausführen einfacher Programme über die Bash-Shell-Befehlszeile
- Ausführen von Befehlen zum Identifizieren von Dateitypen und Anzeigen von Teilen von Textdateien
- Verwenden von „Tastenkombinationen“ für den Bash-Befehlsverlauf, um Befehle oder Teile von Befehlen effizienter zu wiederholen

### Bevor Sie Beginnen

Melden Sie sich als Benutzer **student** mit dem Passwort **student** bei **workstation** an.

Führen Sie auf **workstation** das Skript **lab cli-review start** aus, um eine bereinigte Übungsumgebung einzurichten. Das Skript kopiert auch die Datei **zcat** in das Benutzerverzeichnis von **student**.

```
[student@workstation ~]$ lab cli-review start
```

1. Zeigen Sie mit dem Befehl **date** die aktuelle Uhrzeit und das aktuelle Datum an.

```
[student@workstation ~]$ date  
Thu Jan 22 10:13:04 PDT 2019
```

2. Zeigen Sie die aktuelle Uhrzeit im 12-Stunden-Format an (zum Beispiel: 11:42:11 AM). Tipp: Die Formatzeichenfolge zur Anzeige dieser Ausgabe ist **%r**.  
Verwenden Sie das Argument **+%r** mit dem Befehl **date**, um die aktuelle Uhrzeit im 12-Stunden-Format anzuzeigen.

```
[student@workstation ~]$ date +%r  
10:14:07 AM
```

3. Was für eine Art Datei ist **/home/student/zcat**? Ist sie für Menschen lesbar?  
Verwenden Sie den Befehl **file**, um den Dateityp zu bestimmen.

```
[student@workstation ~]$ file zcat  
zcat: POSIX shell script, ASCII text executable
```

4. Verwenden Sie den Befehl **wc** und Bash-Tastenkombinationen, um die Größe von **zcat** anzuzeigen.

**Kapitel 2 |** Zugreifen auf die Befehlszeile

Mit dem Befehl **wc** können Sie die Anzahl der Zeilen, Wörter und Bytes im Skript **zcat** anzeigen. Verwenden Sie die Tastenkombination für den Bash-Verlauf **Esc+**. (die Tasten **Esc** und **.** gleichzeitig drücken), um das Argument des vorherigen Befehls wiederzuverwenden, anstatt den Dateinamen erneut einzugeben.

```
[student@workstation ~]$ wc Esc+.  
[student@workstation ~]$ wc zcat  
51 299 1983 zcat
```

5. Zeigen Sie die ersten 10 Zeilen von **zcat** an.

Mit dem Befehl **head** wird der Anfang der Datei angezeigt. Versuchen Sie, die Tastenkombination **Esc+**. erneut zu drücken.

```
[student@workstation ~]$ head Esc+.  
[student@workstation ~]$ head zcat  
#!/bin/sh  
# Uncompress files to standard output.  
  
# Copyright (C) 2007, 2010-2018 Free Software Foundation, Inc.  
  
# This program is free software; you can redistribute it and/or modify  
# it under the terms of the GNU General Public License as published by  
# the Free Software Foundation; either version 3 of the License, or  
# (at your option) any later version.
```

6. Zeigen Sie die letzten 10 Zeilen der Datei **zcat** an.

Verwenden Sie den Befehl **tail**, um die letzten 10 Zeilen der Datei **zcat** anzuzeigen.

```
[student@workstation ~]$ tail Esc+.  
[student@workstation ~]$ tail zcat  
With no FILE, or when FILE is -, read standard input.  
  
Report bugs to <bug-gzip@gnu.org>."  
  
case $1 in  
--help) printf '%s\n' "$usage" || exit 1;;  
--version) printf '%s\n' "$version" || exit 1;;  
esac  
  
exec gzip -cd "$@"
```

7. Wiederholen Sie den vorherigen Befehl exakt mit drei oder weniger Tastenanschlägen.

Wiederholen Sie den vorherigen Befehl exakt. Drücken Sie entweder die Taste **Pfeil nach oben** einmal, um einen Befehl im Befehlsverlauf zurückzunavigieren, und drücken Sie dann die **Eingabetaste** (= zwei Tastenanschläge) oder geben Sie die Tastenkombination **!!** ein

**Kapitel 2 |** Zugreifen auf die Befehlszeile

und drücken Sie dann die **Eingabetaste** (= drei Tastenanschläge), um den letzten Befehl im Befehlsverlauf auszuführen. (Versuchen Sie beides.)

```
[student@workstation]$ !!
tail zcat
With no FILE, or when FILE is -, read standard input.

Report bugs to <bug-gzip@gnu.org>.

case $1 in
--help)    printf '%s\n' "$usage"    || exit 1;;
--version) printf '%s\n' "$version" || exit 1;;
esac

exec gzip -cd "$@"
```

8. Wiederholen Sie den vorherigen Befehl, verwenden Sie jedoch die Option **-n 20** zum Anzeigen der letzten 20 Zeilen in der Datei. Verwenden Sie die Befehlszeilenbearbeitung, um dies mit möglichst wenigen Tastenanschlägen zu erreichen.

**Pfeil nach oben** zeigt den vorherigen Befehl an. **Strg+A** setzt den Cursor an den Zeilenanfang. **Strg+Pfeil nach rechts** springt zum nächsten Wort und fügt die Option **-n 20** hinzu und drücken der **Eingabetaste** führt den Befehl aus.

```
[student@workstation ~]$ tail -n 20 zcat
-l, --list          list compressed file contents
-q, --quiet         suppress all warnings
-r, --recursive    operate recursively on directories
-S, --suffix=SUF   use suffix SUF on compressed files
                   --synchronous synchronous output (safer if system crashes, but slower)
-t, --test          test compressed file integrity
-v, --verbose       verbose mode
                   --help           display this help and exit
                   --version        display version information and exit

With no FILE, or when FILE is -, read standard input.

Report bugs to <bug-gzip@gnu.org>.

case $1 in
--help)    printf '%s\n' "$usage"    || exit 1; exit;;
--version) printf '%s\n' "$version" || exit 1; exit;;
esac

exec gzip -cd "$@"
```

9. Verwenden Sie den Shell-Verlauf, um den Befehl **date +%r** erneut auszuführen. Zeigen Sie mit dem Befehl **history** die Liste der vorherigen Befehle an und suchen Sie den spezifischen Befehl **date**, der ausgeführt werden soll. Verwenden Sie **!number**, um den

## Kapitel 2 | Zugreifen auf die Befehlszeile

Befehl auszuführen. *number* ist dabei die Nummer des Befehls, der aus der Ausgabe des Befehls **history** verwendet werden soll.

Beachten Sie, dass sich Ihr Shell-Verlauf von dem folgenden Beispiel unterscheiden kann. Bestimmen Sie die zu verwendende Befehlsnummer basierend auf Ihrer Ausgabe des Befehls **history**.

```
[student@workstation ~]$ history
1 date
2 date +%r
3 file zcat
4 wc zcat
5 head zcat
6 tail zcat
7 tail -n 20 zcat
8 history
[student@workstation ~]$ !2
date +%r
10:49:56 AM
```

## Bewertung

Führen Sie auf **workstation** das Skript `lab cli-review grade` aus, um den Erfolg dieser Übung zu bestätigen.

```
[student@workstation ~]$ lab cli-review grade
```

## Beenden

Führen Sie auf `workstation` das Skript `lab cli-review finish` aus, um die praktische Übung zu beenden.

```
[student@workstation ~]$ lab cli-review finish
```

Damit ist die praktische Übung abgeschlossen.

# Zusammenfassung

---

In diesem Kapitel erhielten Sie folgende Informationen:

- Die Bash-Shell ist ein Befehlsinterpret, der interaktive Benutzer zur Eingabe von Linux-Befehlen auffordert.
- Viele Befehle verfügen über die Option **--help**, die eine Meldung oder einen Bildschirm zur Verwendung des jeweiligen Befehls anzeigt.
- Über Workspaces können mehrere Anwendungsfenster einfacher organisiert werden.
- Mit der Schaltfläche **Activities** in der linken oberen Ecke der oberen Leiste wird ein Übersichtsmodus geöffnet, in dem Benutzer Fenster organisieren und Anwendungen starten können.
- Mit dem Befehl **file** wird der Anfang des Dateiinhalts gescannt und angezeigt, um welchen Typ es sich handelt.
- Die Befehle **head** und **tail** zeigen den Anfang bzw. das Ende einer Datei an.
- Sie können mit der **Tab**-Vervollständigung Dateinamen bei der Eingabe als Befehlsargumente vervollständigen.



## Kapitel 3

# Verwalten von Dateien über die Befehlszeile

### Ziel

Kopieren, Verschieben, Erstellen, Löschen und Organisieren von Dateien über die Bash-Shell-Eingabeaufforderung

### Ziele

- Beschreiben, wie Linux Dateien organisiert, und Erläutern des Zwecks verschiedener Verzeichnisse in der Dateisystemhierarchie
- Angeben des Speicherorts von Dateien relativ zum aktuellen Arbeitsverzeichnis und nach absolutem Speicherort, Bestimmen und Ändern Ihres Arbeitsverzeichnisses sowie Auflisten des Inhalts von Verzeichnissen
- Erstellen, Kopieren, Verschieben und Entfernen von Dateien und Verzeichnissen
- Festlegen, dass mehrere Dateinamen die gleiche Datei referenzieren, unter Verwendung von Hardlinks und symbolischen Verknüpfungen (oder „Softlinks“)
- Effizientes Ausführen von Befehlen, die sich auf viele Dateien auswirken, unter Verwendung der Mustervergleichsfunktionen der Bash-Shell

### Abschnitte

- Beschreiben der Hierarchiekonzepte des Linux-Dateisystems (und Test)
- Angeben von Dateien nach Name (und Test)
- Verwalten von Dateien mit Befehlszeilentools (und angeleitete Übung)
- Erstellen von Links zwischen Dateien (und angeleitete Übung)
- Abgleichen von Dateinamen mit Shell-Erweiterungen (und Test)

### Praktische Übung

Verwalten von Dateien über die Befehlszeile

# Beschreiben der Hierarchiekonzepte des Linux-Dateisystems

---

## Ziele

Nach Abschluss dieses Abschnitts sollten Sie in der Lage sein, zu beschreiben, wie Linux Dateien organisiert, sowie den Zweck verschiedener Verzeichnisse in der Dateisystemhierarchie zu erläutern.

## Die Dateisystemhierarchie

Alle Dateien in einem Linux-System werden in Dateisystemen gespeichert, die in einer einzelnen *invertierten* Verzeichnisbaumstruktur organisiert sind, die als *Dateisystemhierarchie* bezeichnet wird. Diese Baumstruktur ist invertiert, da die Wurzel (root) des Baums an der Spitze der Hierarchie stehen muss und sich die Verzeichnisse und Unterverzeichnisse *unter* der Wurzel ausbreiten.

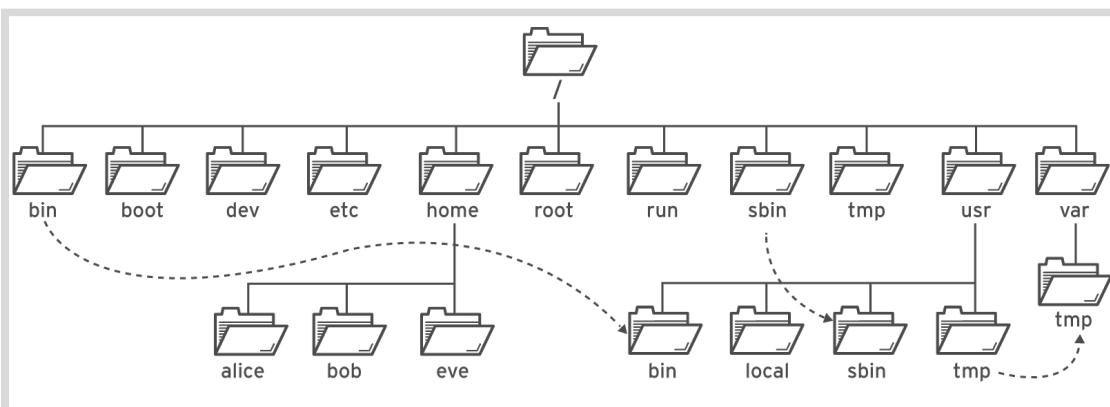


Abbildung 3.1: Wichtige Dateisystemverzeichnisse in Red Hat Enterprise Linux 8

Das Verzeichnis `/` ist das Root-Verzeichnis an der Spitze der Dateisystemhierarchie. Das Zeichen `/` wird auch als *Verzeichnistrennzeichen* in Dateinamen verwendet. Wenn `etc` beispielsweise ein Unterverzeichnis des Verzeichnisses `/` ist, könnten Sie dieses Verzeichnis mit `/etc` referenzieren. Entsprechend gilt, dass Sie für den Fall, dass das Verzeichnis `/etc` eine Datei mit dem Namen `issue` enthält, diese Datei mit `/etc/issue` referenzieren könnten.

Unterverzeichnisse von `/` werden für standardisierte Zwecke zur Organisation von Dateien nach Typ und Zweck eingesetzt. Dies erleichtert die Dateisuche. So wird beispielsweise im Root-Verzeichnis das Unterverzeichnis `/boot` zum Speichern der zum Starten des Systems benötigten Dateien verwendet.



### Anmerkung

Inhalte von Dateisystemverzeichnissen werden mit den folgenden Begriffen beschrieben:

- *statischer* Inhalt verbleibt unverändert, es sei denn, er wird explizit bearbeitet oder neu konfiguriert.
- *dynamischer* oder *variabler* Inhalt kann von aktiven Prozessen geändert oder ergänzt werden.
- *persistenter* Inhalt verbleibt nach dem Neubooten erhalten.
- *Laufzeit*-Inhalt ist prozess- oder systemspezifischer Inhalt, der durch Neubooten gelöscht wird.

In der nachstehenden Tabelle werden die wichtigsten Verzeichnisse im System nach Name und Zweck aufgeführt.

### Wichtige Verzeichnisse von Red Hat Enterprise Linux

Speicherort	Zweck
<b>/usr</b>	Installierte Software, freigegebene Bibliotheken, Include-Dateien und schreibgeschützte Programmdaten. Wichtige Unterverzeichnisse enthalten Folgendes: <ul style="list-style-type: none"> <li>• <b>/usr/bin</b>: Benutzerbefehle</li> <li>• <b>/usr/sbin</b>: Befehle zur Systemadministration</li> <li>• <b>/usr/local</b>: Lokal angepasste Software</li> </ul>
<b>/etc</b>	Für dieses System spezifische Konfigurationsdateien.
<b>/var</b>	Für dieses System spezifische variable Daten, die zwischen Startvorgängen beibehalten werden sollten. Dateien, die sich dynamisch ändern, z. B. Datenbanken, Cache-Verzeichnisse, Protokolldateien, vom Drucker gespoolte Dokumente und Websiteinhalte, befinden sich unter <b>/var</b> .
<b>/run</b>	Laufzeitdaten für Prozesse, die seit dem letzten Startvorgang gestartet wurden. Dies umfasst unter anderem Prozess-ID-Dateien und Sperrdateien. Die Inhalte dieses Verzeichnisses werden beim Neustart wiederhergestellt. Dieses Verzeichnis vereinigt <b>/var/run</b> und <b>/var/lock</b> aus früheren Versionen von Red Hat Enterprise Linux.
<b>/home</b>	Benutzerverzeichnisse, in denen reguläre Benutzer ihre persönlichen Daten und Konfigurationsdateien speichern.

Speicherort	Zweck
/root	Benutzerverzeichnisse für den administrativen Superuser, <b>root</b> .
/tmp	Ein beschreibbarer Speicherplatz für temporäre Dateien. Dateien, die 10 Tage lang nicht aufgerufen, verändert oder bearbeitet wurden, werden automatisch aus diesem Verzeichnis gelöscht. Es gibt außerdem ein weiteres temporäres Verzeichnis ( <b>/var/tmp</b> ), in dem Dateien, die 30 Tage lang nicht aufgerufen, verändert oder bearbeitet wurden, automatisch gelöscht werden.
/boot	Dateien, die für den Startvorgang erforderlich sind.
/dev	Enthält spezielle <i>Gerätedateien</i> , die vom System für den Zugriff auf Hardware verwendet werden.



### Wichtig

In Red Hat Enterprise Linux 7 und früher verfügen vier ältere Verzeichnisse in / über identische Inhalte wie ihre Entsprechungen in **/usr**:

- **/bin** und **/usr/bin**
- **/sbin** und **/usr/sbin**
- **/lib** und **/usr/lib**
- **/lib64** und **/usr/lib64**

In früheren Versionen von Red Hat Enterprise Linux handelte es sich hierbei um verschiedene Verzeichnisse mit unterschiedlichen Dateisätzen.

In Red Hat Enterprise Linux 7 und früher sind die Verzeichnisse in / symbolische Verknüpfungen mit den entsprechenden Verzeichnissen in **/usr**.



### Literaturhinweise

Manpage (7)[hier](#)

#### Die Seite „UsrMove-Feature“ von Fedora 17

<https://fedoraproject.org/wiki/Features/UsrMove>

## ► Quiz

# Beschreiben der Hierarchiekonzepte des Linux-Dateisystems

Wählen Sie die richtigen Antworten auf die folgenden Fragen aus:

- ▶ 1. Welches Verzeichnis enthält persistente, systemspezifische Konfigurationsdaten?
  - a. /etc
  - b. /root
  - c. /run
  - d. /usr
  
- ▶ 2. Welches Verzeichnis ist das oberste Verzeichnis in der Dateisystemhierarchie des Systems?
  - a. /etc
  - b. /
  - c. /home/root
  - d. /root
  
- ▶ 3. Welches Verzeichnis enthält Benutzerverzeichnisse?
  - a. /
  - b. /home
  - c. /root
  - d. /user
  
- ▶ 4. Welches Verzeichnis enthält temporäre Dateien?
  - a. /tmp
  - b. /trash
  - c. /run
  - d. /var
  
- ▶ 5. Welches Verzeichnis enthält dynamische Daten, zum Beispiel für Datenbanken und Websites?
  - a. /etc
  - b. /run
  - c. /usr
  - d. /var

► **6. Welches Verzeichnis ist das Benutzerverzeichnis des Administrator-Superusers?**

- a. /etc
- b. /
- c. /home/root
- d. /root

► **7. Welches Verzeichnis enthält reguläre Befehle und Dienstprogramme?**

- a. /commands
- b. /run
- c. /usr/bin
- d. /usr/sbin

► **8. Welches Verzeichnis enthält nicht persistente Prozesslaufzeitdaten?**

- a. /tmp
- b. /etc
- c. /run
- d. /var

► **9. Welches Verzeichnis enthält installierte Softwareprogramme und Bibliotheken?**

- a. /etc
- b. /lib
- c. /usr
- d. /var

## ► Lösung

# Beschreiben der Hierarchiekonzepte des Linux-Dateisystems

Wählen Sie die richtigen Antworten auf die folgenden Fragen aus:

► 1. Welches Verzeichnis enthält persistente, systemspezifische Konfigurationsdaten?

- a. /etc
- b. /root
- c. /run
- d. /usr

► 2. Welches Verzeichnis ist das oberste Verzeichnis in der Dateisystemhierarchie des Systems?

- a. /etc
- b. /
- c. /home/root
- d. /root

► 3. Welches Verzeichnis enthält Benutzerverzeichnisse?

- a. /
- b. /home
- c. /root
- d. /user

► 4. Welches Verzeichnis enthält temporäre Dateien?

- a. /tmp
- b. /trash
- c. /run
- d. /var

► 5. Welches Verzeichnis enthält dynamische Daten, zum Beispiel für Datenbanken und Websites?

- a. /etc
- b. /run
- c. /usr
- d. /var

► **6. Welches Verzeichnis ist das Benutzerverzeichnis des Administrator-Superusers?**

- a. /etc
- b. /
- c. /home/root
- d. /root

► **7. Welches Verzeichnis enthält reguläre Befehle und Dienstprogramme?**

- a. /commands
- b. /run
- c. /usr/bin
- d. /usr/sbin

► **8. Welches Verzeichnis enthält nicht persistente Prozesslaufzeitdaten?**

- a. /tmp
- b. /etc
- c. /run
- d. /var

► **9. Welches Verzeichnis enthält installierte Softwareprogramme und Bibliotheken?**

- a. /etc
- b. /lib
- c. /usr
- d. /var

# Angeben von Dateien nach Name

## Ziele

Nach Abschluss dieses Abschnitts sollten Sie in der Lage sein, den Speicherort von Dateien relativ zum aktuellen Arbeitsverzeichnis und nach absolutem Speicherort anzugeben, Ihre Arbeitsverzeichnisse festzulegen und zu ändern sowie den Inhalt von Verzeichnissen aufzulisten.

## Absolute und relative Pfade

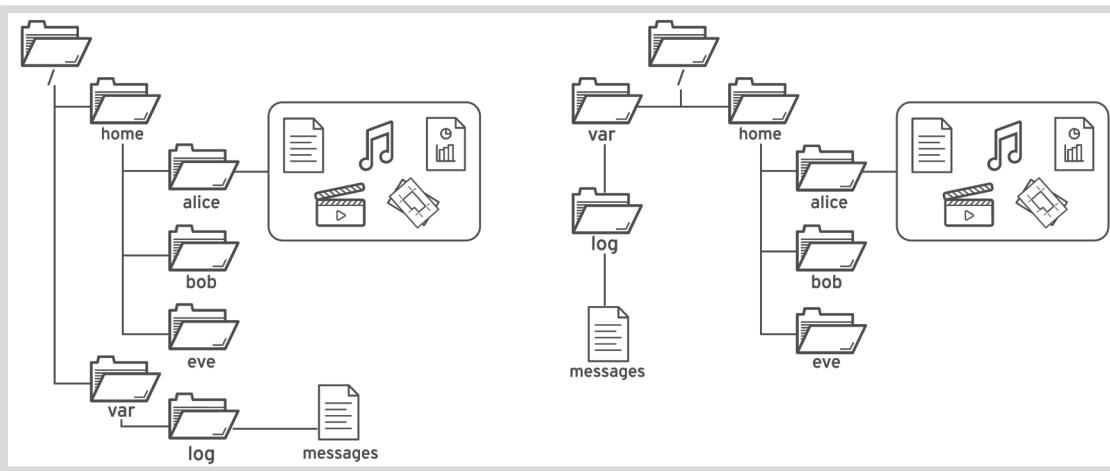


Abbildung 3.2: Die gängige Datei-Browseransicht (links) entspricht der Ansicht von oben nach unten (rechts).

Mit dem *Pfad* einer Datei oder eines Verzeichnisses wird der eindeutige Dateisystemspeicherort angegeben. Ein Dateipfad durchläuft eines oder mehrere benannte Unterverzeichnisse, angegeben durch einen Schrägstrich (/), bis das Ziel erreicht ist. Verzeichnisse, auch *Ordner* genannt, enthalten weitere Dateien und weitere Unterverzeichnisse. Sie können auf dieselbe Weise wie Dateien referenziert werden.



### Wichtig

Ein Leerzeichen ist als Teil eines Linux-Dateinamens zulässig. Leerzeichen werden jedoch auch von der Shell verwendet, um Optionen und Argumente in der Befehlszeile zu trennen. Wenn Sie einen Befehl eingeben, der einen Dateinamen mit einem Leerzeichen enthält, kann die Shell den Befehl falsch interpretieren und davon ausgehen, dass Sie einen neuen Dateinamen oder ein anderes Argument an dem Leerzeichen beginnen möchten.

Sie können dies vermeiden, indem Sie Dateinamen in Anführungszeichen setzen. Wenn Sie jedoch keine Leerzeichen in Dateinamen verwenden müssen, vermeiden Sie deren Verwendung einfach.

## Absolute Pfade

Ein *absoluter Pfad* ist ein *vollständig qualifizierter Name*, der den genauen Speicherort der Dateien in der Dateisystemhierarchie angibt. Er beginnt mit dem Root-Verzeichnis (/) und gibt jedes Unterverzeichnis an, das durchlaufen werden muss, um auf die jeweilige Datei zuzugreifen. Jede Datei in einem Dateisystem verfügt über einen eindeutigen absoluten Pfadnamen, der nach einer einfachen Regel zu erkennen ist: Ein Pfadname mit einem Schrägstrich (/) als erstes Zeichen ist ein absoluter Pfadname. Der absolute Pfadname für die Protokolldatei von Systemmeldungen lautet **/var/log/messages**. Die Eingabe absoluter Pfadnamen kann langwierig sein, daher können Dateien auch *relativ* zum aktuellen Arbeitsverzeichnis für Ihre Shell-Eingabeaufforderung gesucht werden.

## Das aktuelle Arbeitsverzeichnis und relative Pfade

Meldet sich ein Benutzer an und öffnet ein Befehlseingabefenster, wird in der Regel zuerst das Benutzerverzeichnis des Benutzers angezeigt. Für Systemprozesse ist auch ein Anfangsverzeichnis vorhanden. Benutzer und Prozesse navigieren gegebenenfalls zu anderen Verzeichnissen. Die Begriffe *Arbeitsverzeichnis* und *aktueller Arbeitsverzeichnis* beziehen sich auf den *aktuellen* Speicherort.

Genau wie mit einem absoluten Pfad wird mit einem *relativen Pfad* eine eindeutige Datei identifiziert, indem nur der Pfad angegeben wird, der zum Erreichen der Datei vom Arbeitsverzeichnis aus notwendig ist. Die Erkennung relativer Pfadnamen folgt einer einfachen Regel: Ein Pfadname mit *einem anderen Zeichen* als einem Schrägstrich als erstes Zeichen ist ein *relativer Pfadname*. Ein Benutzer könnte aus dem Verzeichnis **/var** auf die Meldungsprotokolldatei mit **log/messages** relativ verweisen.

Bei Linux-Dateisystemen, einschließlich jedoch nicht ausschließlich ext4, XFS, GFS2 und GlusterFS, wird nach Groß- und Kleinschreibung unterschieden. Mit der Erstellung von **FileCase.txt** und **filecase.txt** in einem Verzeichnis werden zwei eindeutige Dateien angelegt.

Nicht-Linux-Dateisysteme funktionieren möglicherweise anders. Beispielsweise behalten VFAT, NTFS von Microsoft und HFS+ von Apple die *Groß- und Kleinschreibung* bei. Auch wenn bei diesen Dateisystemen *nicht* nach Groß- und Kleinschreibung unterschieden wird, werden Dateinamen mit der bei der Dateierstellung ursprünglich verwendeten Groß- und Kleinschreibung angezeigt. Wenn Sie daher versuchen, die Dateien im vorherigen Beispiel in einem VFAT-Dateisystem zu erstellen, werden beide Namen so behandelt, als würden sie auf dieselbe Datei anstatt auf zwei verschiedene Dateien verweisen.

## Navigieren in Pfaden

Der Befehl **pwd** zeigt den vollständigen Pfadnamen des aktuellen Arbeitsverzeichnisses für diese Shell an. Damit können Sie die Syntax ermitteln, mit der Dateien mithilfe relativer Pfadnamen erreicht werden. Mit dem Befehl **ls** werden Verzeichnisinhalte für das angegebene Verzeichnis oder, falls kein Verzeichnis angegeben ist, für das aktuelle Arbeitsverzeichnis aufgeführt.

```
[user@host ~]$ pwd  
/home/user  
[user@host ~]$ ls  
Desktop Documents Downloads Music Pictures Public Templates Videos  
[user@host ~]$
```

Mit dem Befehl **cd** ändern Sie das aktuelle Arbeitsverzeichnis Ihrer Shell. Wenn Sie keine Argumente für den Befehl angeben, wechselt er in Ihr Benutzerverzeichnis.

### Kapitel 3 | Verwalten von Dateien über die Befehlszeile

Im folgenden Beispiel wird eine Mischung aus absoluten und relativen Pfaden mit dem Befehl **cd** zum Ändern des aktuellen Arbeitsverzeichnisses für die Shell verwendet.

```
[user@host ~]$ pwd  
/home/user  
[user@host ~]$ cd Videos  
[user@host Videos]$ pwd  
/home/user/Videos  
[user@host Videos]$ cd /home/user/Documents  
[user@host Documents]$ pwd  
/home/user/Documents  
[user@host Documents]$ cd  
[user@host ~]$ pwd  
/home/user  
[user@host ~]$
```

Wie Sie im vorherigen Beispiel sehen können, zeigt die Standard-Shell-Eingabeaufforderung auch die letzte Komponente des absoluten Pfads zum aktuellen Arbeitsverzeichnis an. Für **/home/user/Videos** wird beispielsweise nur **Videos** angezeigt. In der Eingabeaufforderung wird das Tilde-Zeichen (~) angezeigt, wenn es sich bei Ihrem aktuellen Arbeitsverzeichnis um Ihr Benutzerverzeichnis handelt.

Mit dem Befehl **touch** wird der Zeitstempel einer Datei normalerweise mit dem aktuellen Datum und der aktuellen Uhrzeit aktualisiert, ohne dass diese anderweitig verändert wird. Dies ist für die Erstellung leerer Dateien hilfreich, die dann zu Übungszwecken verwendet werden, da die Anwendung des Befehls „**touch**“ auf einen nicht vorhandenen Dateinamen zur Erstellung der Datei führt. Im folgenden Beispiel erstellt der Befehl **touch** Übungsdateien in den Unterverzeichnissen **Documents** und **Videos**.

```
[user@host ~]$ touch Videos/blockbuster1.ogg  
[user@host ~]$ touch Videos/blockbuster2.ogg  
[user@host ~]$ touch Documents/thesis_chapter1.odf  
[user@host ~]$ touch Documents/thesis_chapter2.odf  
[user@host ~]$
```

Für den Befehl **ls** stehen mehrere Optionen für die Anzeige von Dateiattributen zur Verfügung. Die am häufigsten verwendeten und hilfreichsten Attribute sind **-l** (Langformat), **-a** (alle Dateien, einschließlich *verborgener* Dateien) und **-R** (rekursiv zur Berücksichtigung des Inhalts aller Unterverzeichnisse).

```
[user@host ~]$ ls -l  
total 15  
drwxr-xr-x. 2 user user 4096 Feb  7 14:02 Desktop  
drwxr-xr-x. 2 user user 4096 Jan  9 15:00 Documents  
drwxr-xr-x. 3 user user 4096 Jan  9 15:00 Downloads  
drwxr-xr-x. 2 user user 4096 Jan  9 15:00 Music  
drwxr-xr-x. 2 user user 4096 Jan  9 15:00 Pictures  
drwxr-xr-x. 2 user user 4096 Jan  9 15:00 Public  
drwxr-xr-x. 2 user user 4096 Jan  9 15:00 Templates  
drwxr-xr-x. 2 user user 4096 Jan  9 15:00 Videos  
[user@host ~]$ ls -la  
total 15  
drwx----- 16 user user 4096 Feb  8 16:15 .
```

```
drwxr-xr-x. 6 root root 4096 Feb 8 16:13 ..
-rw----- 1 user user 22664 Feb 8 00:37 .bash_history
-rw-r--r-- 1 user user 18 Jul 9 2013 .bash_logout
-rw-r--r-- 1 user user 176 Jul 9 2013 .bash_profile
-rw-r--r-- 1 user user 124 Jul 9 2013 .bashrc
drwxr-xr-x. 4 user user 4096 Jan 20 14:02 .cache
drwxr-xr-x. 8 user user 4096 Feb 5 11:45 .config
drwxr-xr-x. 2 user user 4096 Feb 7 14:02 Desktop
drwxr-xr-x. 2 user user 4096 Jan 9 15:00 Documents
drwxr-xr-x. 3 user user 4096 Jan 25 20:48 Downloads
drwxr-xr-x. 11 user user 4096 Feb 6 13:07 .gnome2
drwx----- 2 user user 4096 Jan 20 14:02 .gnome2_private
-rw----- 1 user user 15190 Feb 8 09:49 .ICEauthority
drwxr-xr-x. 3 user user 4096 Jan 9 15:00 .local
drwxr-xr-x. 2 user user 4096 Jan 9 15:00 Music
drwxr-xr-x. 2 user user 4096 Jan 9 15:00 Pictures
drwxr-xr-x. 2 user user 4096 Jan 9 15:00 Public
drwxr-xr-x. 2 user user 4096 Jan 9 15:00 Templates
drwxr-xr-x. 2 user user 4096 Jan 9 15:00 Videos
[user@host ~]$
```

Die beiden Spezialverzeichnisse unten in der Liste verweisen auf das aktuelle Verzeichnis (.) und das *übergeordnete* Verzeichnis (..). Diese Spezialverzeichnisse sind in jedem Verzeichnis im System vorhanden. Sie werden ihre Nützlichkeit erkennen, wenn Sie Dateiverwaltungsbefehle verwenden.



### Wichtig

Mit einem Punkt (.) beginnende Dateinamen weisen auf *verborgene* Dateien hin; diese Dateien sind in der Normalansicht mit **ls** und anderen Befehlen nicht sichtbar. Hierbei handelt es sich *nicht* um eine Sicherheitsfunktion. Verbogene Dateien tragen dazu bei, dass notwendige Benutzerkonfigurationsdateien die Benutzerverzeichnisse nicht „überfüllen“. Viele Befehle verarbeiten verbogene Dateien nur mit bestimmten Befehlszeilenoptionen. Somit wird verhindert, dass die Konfiguration eines Benutzers versehentlich in andere Verzeichnisse oder zu anderen Benutzern kopiert wird.

Zum Schutz von *Dateiinhalten* vor unberechtigter Anzeige müssen *Dateiberechtigungen* eingesetzt werden.

```
[user@host ~]$ ls -R
.:
Desktop Documents Downloads Music Pictures Public Templates Videos

./Desktop:

./Documents:
thesis_chapter1.odf thesis_chapter2.odf

./Downloads:

./Music:
```

```
./Pictures:  
  
./Public:  
  
./Templates:  
  
./Videos:  
blockbuster1.ogg blockbuster2.ogg  
[user@host ~]$
```

Für den Befehl **cd** stehen viele Optionen zur Verfügung. Einige sind so nützlich, dass es sich lohnt, ihre Anwendung rechtzeitig zu üben und sie häufig einzusetzen. Mit dem Befehl **cd -** wird in das vorherige Verzeichnis gewechselt, in dem sich der Benutzer vor dem aktuellen Verzeichnis befand. Das folgende Beispiel, in dem zwischen zwei Verzeichnissen gewechselt wird, veranschaulicht dieses Verhalten. Dies ist hilfreich, wenn Sie eine Reihe ähnlicher Aufgaben bearbeiten.

```
[user@host ~]$ cd Videos  
[user@host Videos]$ pwd  
/home/user/Videos  
[user@host Videos]$ cd /home/user/Documents  
[user@host Documents]$ pwd  
/home/user/Documents  
[user@host Documents]$ cd -  
[user@host Videos]$ pwd  
/home/user/Videos  
[user@host Videos]$ cd -  
[user@host Documents]$ pwd  
/home/user/Documents  
[user@host Documents]$ cd -  
[user@host Videos]$ pwd  
/home/user/Videos  
[user@host Videos]$ cd  
[user@host ~]$
```

Der Befehl **cd ..** verwendet das verborgene Verzeichnis **..**, um eine Ebene höher in das *übergeordnete* Verzeichnis zu wechseln, ohne den genauen Namen des übergeordneten Verzeichnisses zu kennen. Im anderen verborgenen Verzeichnis (**.**) ist das *aktuelle Verzeichnis* für Befehle angegeben, bei denen der aktuelle Speicherort entweder das Ausgangs- oder das Zielargument ist, damit nicht der absolute Pfadname des Verzeichnisses eingegeben werden muss.

```
[user@host Videos]$ pwd  
/home/user/Videos  
[user@host Videos]$ cd .  
[user@host Videos]$ pwd  
/home/user/Videos  
[user@host Videos]$ cd ..  
[user@host ~]$ pwd  
/home/user  
[user@host ~]$ cd ..  
[user@host home]$ pwd  
/home  
[user@host home]$ cd ..
```

```
[user@host /]$ pwd  
/  
[user@host /]$ cd  
[user@host ~]$ pwd  
/home/user  
[user@host ~]$
```



### Literaturhinweise

**info libc 'file name resolution'** (*Referenzhandbuch für die GNU C Library*)

- Abschnitt 11.2.2: Dateinamenauflösung

Manpages **bash(1)**, **cd(1)**, **ls(1)**, **pwd(1)**, **unicode(7)** und **utf-8(7)**

### UTF-8 und Unicode

<http://www.utf-8.com/>

## ► Quiz

# Angeben von Dateien nach Name

Wählen Sie die richtigen Antworten auf die folgenden Fragen aus:

- ▶ 1. Mit welchem Befehl wird zum Benutzerverzeichnis des aktuellen Benutzers zurückgekehrt, vorausgesetzt, das aktuelle Arbeitsverzeichnis ist /tmp und das Benutzerverzeichnis ist /home/user?
  - a. cd
  - b. cd ..
  - c. cd .
  - d. cd \*
  - e. cd /home
  
- ▶ 2. Welcher Befehl zeigt den absoluten Pfadnamen des aktuellen Speicherorts an?
  - a. cd
  - b. pwd
  - c. ls ~
  - d. ls -d
  
- ▶ 3. Mit welchem Befehl wird immer zu dem Arbeitsverzeichnis zurückgekehrt, das vor dem aktuellen Arbeitsverzeichnis verwendet wurde?
  - a. cd -
  - b. cd -p
  - c. cd ~
  - d. cd ..
  
- ▶ 4. Mit welchem Befehl wird das Arbeitsverzeichnis immer um zwei Ebenen vom aktuellen Speicherort aus nach oben geändert?
  - a. cd ~
  - b. cd ../
  - c. cd ../../
  - d. cd -u2
  
- ▶ 5. Welcher Befehl listet Dateien am aktuellen Speicherort in einem langen Format und einschließlich versteckter Dateien auf?
  - a. llong ~
  - b. ls -a
  - c. ls -l
  - d. ls -al

► 6. Welcher Befehl ändert das Arbeitsverzeichnis immer in /bin?

- a. **cd bin**
- b. **cd /bin**
- c. **cd ~bin**
- d. **cd -bin**

► 7. Mit welchem Befehl wird das Arbeitsverzeichnis immer in das übergeordnete Verzeichnis des aktuellen Speicherorts geändert?

- a. **cd ~**
- b. **cd ..**
- c. **cd .../..**
- d. **cd -u1**

► 8. Mit welchem Befehl wird das Arbeitsverzeichnis in /tmp geändert, wenn das aktuelle Arbeitsverzeichnis /home/student ist?

- a. **cd tmp**
- b. **cd ..**
- c. **cd .../.../tmp**
- d. **cd ~tmp**

## ► Lösung

# Angeben von Dateien nach Name

Wählen Sie die richtigen Antworten auf die folgenden Fragen aus:

- ▶ 1. Mit welchem Befehl wird zum Benutzerverzeichnis des aktuellen Benutzers zurückgekehrt, vorausgesetzt, das aktuelle Arbeitsverzeichnis ist /tmp und das Benutzerverzeichnis ist /home/user?
  - a. cd
  - b. cd ..
  - c. cd .
  - d. cd \*
  - e. cd /home
  
- ▶ 2. Welcher Befehl zeigt den absoluten Pfadnamen des aktuellen Speicherorts an?
  - a. cd
  - b. pwd
  - c. ls ~
  - d. ls -d
  
- ▶ 3. Mit welchem Befehl wird immer zu dem Arbeitsverzeichnis zurückgekehrt, das vor dem aktuellen Arbeitsverzeichnis verwendet wurde?
  - a. cd -
  - b. cd -p
  - c. cd ~
  - d. cd ..
  
- ▶ 4. Mit welchem Befehl wird das Arbeitsverzeichnis immer um zwei Ebenen vom aktuellen Speicherort aus nach oben geändert?
  - a. cd ~
  - b. cd ../
  - c. cd .../..
  - d. cd -u2
  
- ▶ 5. Welcher Befehl listet Dateien am aktuellen Speicherort in einem langen Format und einschließlich versteckter Dateien auf?
  - a. llong ~
  - b. ls -a
  - c. ls -l
  - d. ls -al

► **6. Welcher Befehl ändert das Arbeitsverzeichnis immer in /bin?**

- a. cd bin
- b. cd /bin**
- c. cd ~bin
- d. cd -bin

► **7. Mit welchem Befehl wird das Arbeitsverzeichnis immer in das übergeordnete Verzeichnis des aktuellen Speicherorts geändert?**

- a. cd ~
- b. cd ..**
- c. cd ../..
- d. cd -u1

► **8. Mit welchem Befehl wird das Arbeitsverzeichnis in /tmp geändert, wenn das aktuelle Arbeitsverzeichnis /home/student ist?**

- a. cd tmp
- b. cd ..
- c. cd ../../tmp**
- d. cd ~tmp

# Verwalten von Dateien mit Befehlszeilentools

---

## Ziele

Nach Abschluss dieses Kapitels sollten Sie in der Lage sein, Dateien und Verzeichnisse zu erstellen, zu kopieren, zu verschieben und zu entfernen.

## Dateiverwaltung über die Befehlszeile

Zum Verwalten von Dateien müssen Sie in der Lage sein, Dateien zu erstellen, zu entfernen, zu kopieren und zu verschieben. Sie müssen sie zudem logisch in Verzeichnissen organisieren, die Sie auch erstellen, entfernen, kopieren und verschieben können müssen.

In der folgenden Tabelle sind einige der am häufigsten verwendeten Dateiverwaltungsbefehle zusammengefasst. Im weiteren Verlauf dieses Abschnitts werden Möglichkeiten zur detaillierten Verwendung dieser Befehle erläutert.

### Allgemeine Befehl für die Dateiverwaltung

Aktivität	Befehlssyntax
Verzeichnis erstellen	<b>mkdir directory</b>
Datei kopieren	<b>cp file new-file</b>
Verzeichnis und seinen Inhalt kopieren	<b>cp -r directory new-directory</b>
Datei oder Verzeichnis verschieben oder umbenennen	<b>mv file new-file</b>
Datei entfernen	<b>rm file</b>
Verzeichnis mit enthaltenen Dateien entfernen	<b>rm -r directory</b>
Leeres Verzeichnis entfernen	<b>rmdir directory</b>

## Erstellen von Verzeichnissen

Der Befehl **mkdir** erstellt ein oder mehrere Verzeichnisse oder Unterverzeichnisse. Als Argumente wird eine Liste von Pfaden zu den Verzeichnissen verwendet, die Sie erstellen möchten.

Der Befehl **mkdir** schlägt mit einem Fehler fehl, wenn das Verzeichnis bereits vorhanden ist oder wenn Sie versuchen, ein Unterverzeichnis in einem nicht vorhandenen Verzeichnis zu erstellen. Mit der Option **-p (parent)** werden fehlende übergeordnete Verzeichnisse für das angeforderte Ziel erstellt. Seien Sie bei der Verwendung des Befehls **mkdir -p** vorsichtig, da bei versehentlichen Schreibfehlern unbeabsichtigte Verzeichnisse ohne Fehlermeldung generiert werden können.

Nehmen Sie für das folgende Beispiel an, Sie würden versuchen, ein Verzeichnis im Verzeichnis **Videos** mit dem Namen **Watched** zu erstellen, aber Sie haben versehentlich den Buchstaben „s“ von **Videos** in Ihrem **mkdir**-Befehl weggelassen.

```
[user@host ~]$ mkdir Video/Watched  
mkdir: cannot create directory `Video/Watched': No such file or directory
```

Der Befehl **mkdir** schlägt fehl, da **Videos** falsch geschrieben wurde, und das Verzeichnis **Video** nicht vorhanden ist. Wenn Sie den Befehl **mkdir** mit der Option **-p** verwendet hätten, wäre das Verzeichnis **Video** erstellt worden, was Sie aber nicht beabsichtigt hatten, und das Unterverzeichnis **Watched** würde in diesem falschen Verzeichnis erstellt werden.

Wenn das übergeordnete Verzeichnis **Videos** korrekt geschrieben wird, ist die Erstellung des Unterverzeichnisses **Watched** erfolgreich.

```
[user@host ~]$ mkdir Videos/Watched  
[user@host ~]$ ls -R Videos  
Videos:/  
blockbuster1.ogg blockbuster2.ogg Watched  
  
Videos/Watched:
```

Im folgenden Beispiel werden Dateien und Verzeichnisse unter dem Verzeichnis **/home/user/Documents** organisiert. Verwenden Sie den Befehl **mkdir** und eine durch Leerzeichen getrennte Liste der Verzeichnisnamen zum Erstellen mehrerer Verzeichnisse.

```
[user@host ~]$ cd Documents  
[user@host Documents]$ mkdir ProjectX ProjectY  
[user@host Documents]$ ls  
ProjectX ProjectY
```

Verwenden Sie den Befehl **mkdir -p** und durch Leerzeichen getrennte relative Pfade für die Namen der Unterverzeichnisse zum Erstellen mehrerer Verzeichnisse mit Unterverzeichnissen.

```
[user@host Documents]$ mkdir -p Thesis/Chapter1 Thesis/Chapter2 Thesis/Chapter3  
[user@host Documents]$ cd  
[user@host ~]$ ls -R Videos Documents  
Documents:  
ProjectX ProjectY Thesis  
  
Documents/ProjectX:  
  
Documents/ProjectY:  
  
Documents/Thesis:  
Chapter1 Chapter2 Chapter3  
  
Documents/Thesis/Chapter1:  
  
Documents/Thesis/Chapter2:  
  
Documents/Thesis/Chapter3:  
  
Videos:  
blockbuster1.ogg blockbuster2.ogg Watched
```

Videos/Watched:

Mit dem letzten **mkdir**-Befehl wurden drei ChapterN-Unterverzeichnisse mit einem Befehl erstellt. Das fehlende übergeordnete Verzeichnis **Thesis** wurde mit der Option **-p** erzeugt.

## Kopieren von Dateien

Der Befehl **cp** kopiert eine Datei und erstellt eine neue Datei entweder im aktuellen Verzeichnis oder in einem angegebenen Verzeichnis. Mit dem Befehl können auch mehrere Dateien in ein Verzeichnis kopiert werden.



### Warnung

Wenn die Zielfile bereits vorhanden ist, überschreibt der Befehl **cp** diese Datei.

```
[user@host ~]$ cd Videos
[user@host Videos]$ cp blockbuster1.ogg blockbuster3.ogg
[user@host Videos]$ ls -l
total 0
-rw-rw-r-- 1 user user    0 Feb  8 16:23 blockbuster1.ogg
-rw-rw-r-- 1 user user    0 Feb  8 16:24 blockbuster2.ogg
-rw-rw-r-- 1 user user    0 Feb  8 16:34 blockbuster3.ogg
drwxrwxr-x. 2 user user 4096 Feb  8 16:05 Watched
[user@host Videos]$
```

Werden mehrere Dateien mit einem Befehl kopiert, muss das letzte Argument ein Verzeichnis sein. Kopierte Dateien behalten im neuen Verzeichnis ihren ursprünglichen Namen bei. Wenn im Zielverzeichnis eine Datei mit demselben Namen vorhanden ist, wird die vorhandene Datei überschrieben. Standardmäßig kopiert **cp** keine Verzeichnisse; sie werden ignoriert.

Im folgenden Beispiel werden zwei Verzeichnisse aufgelistet: **Thesis** und **ProjectX**. Nur das letzte Argument, **ProjectX**, ist als Ziel gültig. Das Verzeichnis **Thesis** wird ignoriert.

```
[user@host Videos]$ cd ../Documents
[user@host Documents]$ cp thesis_chapter1.odf thesis_chapter2.odf Thesis ProjectX
cp: omitting directory `Thesis'
[user@host Documents]$ ls Thesis ProjectX
ProjectX:
thesis_chapter1.odf  thesis_chapter2.odf

Thesis:
Chapter1  Chapter2  Chapter3
```

Mit dem ersten **cp**-Befehl wurde das Verzeichnis **Thesis** nicht kopiert, die Dateien **thesis\_chapter1.odf** und **thesis\_chapter2.odf** wurden jedoch kopiert.

Wenn Sie eine Datei in das aktuelle Arbeitsverzeichnis kopieren möchten, können Sie das Sonderverzeichnis **.** verwenden:

### Kapitel 3 | Verwalten von Dateien über die Befehlszeile

```
[user@host ~]$ cp /etc/hostname .
[user@host ~]$ cat hostname
host.example.com
[user@host ~]$
```

Verwenden Sie den Kopierbefehl mit der Option **-r** (*recursive*), um das Verzeichnis **Thesis** und dessen Inhalt in das Verzeichnis **ProjectX** zu kopieren.

```
[user@host Documents]$ cp -r Thesis ProjectX
[user@host Documents]$ ls -R ProjectX
ProjectX:
Thesis thesis_chapter1.odf thesis_chapter2.odf

ProjectX/Thesis:
Chapter1 Chapter2 Chapter3

ProjectX/Thesis/Chapter1:

ProjectX/Thesis/Chapter2:
thesis_chapter2.odf

ProjectX/Thesis/Chapter3:
```

## Verschieben von Dateien

Der Befehl **mv** verschiebt Dateien von einem Speicherort an einen anderen. Wenn Sie sich den absoluten Pfad zu einer Datei als ihren vollständigen Namen vorstellen, entspricht das Verschieben einer Datei dem Umbenennen einer Datei. Der Dateiinhalt wird dabei nicht verändert.

Verwenden Sie den Befehl **mv**, um eine Datei *umzubenennen*.

```
[user@host Videos]$ cd ../Documents
[user@host Documents]$ ls -l thesis*
-rw-rw-r--. 1 user user 0 Feb  6 21:16 thesis_chapter1.odf
-rw-rw-r--. 1 user user 0 Feb  6 21:16 thesis_chapter2.odf
[user@host Documents]$ mv thesis_chapter2.odf thesis_chapter2_reviewed.odf
[user@host Documents]$ ls -l thesis*
-rw-rw-r--. 1 user user 0 Feb  6 21:16 thesis_chapter1.odf
-rw-rw-r--. 1 user user 0 Feb  6 21:16 thesis_chapter2_reviewed.odf
```

Verwenden Sie den Befehl **mv**, um eine Datei in ein anderes Verzeichnis zu verschieben.

```
[user@host Documents]$ ls Thesis/Chapter1
[user@host Documents]$
[user@host Documents]$ mv thesis_chapter1.odf Thesis/Chapter1
[user@host Documents]$ ls Thesis/Chapter1
thesis_chapter1.odf
[user@host Documents]$ ls -l thesis*
-rw-rw-r--. 1 user user 0 Feb  6 21:16 thesis_chapter2_reviewed.odf
```

## Entfernen von Dateien und Verzeichnissen

Der Befehl **rm** entfernt Dateien. Standardmäßig entfernt **rm** keine Verzeichnisse, die Dateien enthalten, es sei denn, Sie fügen die Option **-r** oder **--recursive** hinzu.



### Wichtig

Es stehen keine Befehlszeilenfunktionen zum Rückgängigmachen einer Löschaktion und kein Papierkorb zur Verfügung, aus dem Dateien wiederhergestellt werden könnten.

Sie sollten daher immer überprüfen, ob Sie sich in Ihrem aktuellen Arbeitsverzeichnis befinden, bevor Sie eine Datei oder ein Verzeichnis entfernen.

```
[user@host Documents]$ pwd  
/home/student/Documents
```

Verwenden Sie den Befehl **rm** zum Entfernen einer einzelnen Datei aus Ihrem Arbeitsverzeichnis.

```
[user@host Documents]$ ls -l thesis*  
-rw-rw-r-- 1 user user 0 Feb  6 21:16 thesis_chapter2_reviewed.odf  
[user@host Documents]$ rm thesis_chapter2_reviewed.odf  
[user@host Documents]$ ls -l thesis*  
ls: cannot access 'thesis*': No such file or directory
```

Wenn Sie versuchen, mit dem Befehl **rm** ohne die Option **-r** ein Verzeichnis zu entfernen, schlägt der Befehl fehl.

```
[user@host Documents]$ rm Thesis/Chapter1  
rm: cannot remove `Thesis/Chapter1': Is a directory
```

Verwenden Sie den Befehl **rm -r**, um ein Unterverzeichnis und seinen Inhalt zu entfernen.

```
[user@host Documents]$ ls -R Thesis  
Thesis/:  
Chapter1 Chapter2 Chapter3  
  
Thesis/Chapter1:  
thesis_chapter1.odf  
  
Thesis/Chapter2:  
thesis_chapter2.odf  
  
Thesis/Chapter3:  
[user@host Documents]$ rm -r Thesis/Chapter1  
[user@host Documents]$ ls -l Thesis  
total 8  
drwxrwxr-x. 2 user user 4096 Feb 11 12:47 Chapter2  
drwxrwxr-x. 2 user user 4096 Feb 11 12:48 Chapter3
```

Der Befehl **rm -r** durchläuft zuerst jedes Unterverzeichnis und entfernt einzeln dessen Dateien, bevor er die Verzeichnisse entfernt. Sie können vor dem Löschen mit dem Befehl **rm -ri**

interaktiv eine Bestätigungsaufforderung anzeigen. Dies ist im Wesentlichen das Gegenteil der Option **-f**, die das Entfernen erzwingt, ohne dass der Benutzer zur Bestätigung aufgefordert wird.

```
[user@host Documents]$ rm -ri Thesis
rm: descend into directory `Thesis'? y
rm: descend into directory `Thesis/Chapter2'? y
rm: remove regular empty file `Thesis/Chapter2/thesis_chapter2.odf'? y
rm: remove directory `Thesis/Chapter2'? y
rm: remove directory `Thesis/Chapter3'? y
rm: remove directory `Thesis'? y
[user@host Documents]$
```

**Warnung**

Wenn Sie beide Optionen, **-i** und **-f**, angeben, hat die Option **-f** Priorität und Sie werden nicht zur Bestätigung aufgefordert, bevor **rm** Dateien löscht.

Im folgenden Beispiel entfernt der Befehl **rmdir** nur das leere Verzeichnis. Wie im vorherigen Beispiel müssen Sie den Befehl **rm -r** auch zum Entfernen eines Verzeichnisses verwenden, das Inhalt enthält.

```
[user@host Documents]$ pwd
/home/student/Documents
[user@host Documents]$ rmmdir ProjectY
[user@host Documents]$ rmmdir ProjectX
rmmdir: failed to remove `ProjectX': Directory not empty
[user@host Documents]$ rm -r ProjectX
[user@host Documents]$ ls -lR
.:
total 0
[user@host Documents]$
```

**Anmerkung**

Der Befehl **rm** ohne Optionen kann kein leeres Verzeichnis entfernen. Sie müssen den Befehl **rmdir** verwenden, **rm -d** (entspricht **rmdir**) oder **rm -r**.

**Literaturhinweise**

Manpages **cp(1)**, **mkdir(1)**, **mv(1)**, **rm(1)** und **rmdir(1)**

## ► Angeleitete Übung

# Verwalten von Dateien mit Befehlszeilentools

In dieser Übung erstellen, organisieren, kopieren und entfernen Sie Dateien und Verzeichnisse.

## Ergebnisse

Sie sollten in der Lage sein, Dateien und Verzeichnisse zu erstellen, zu organisieren, zu kopieren und zu entfernen.

## Bevor Sie Beginnen

Melden Sie sich als Benutzer **student** mit dem Passwort **student** bei **workstation** an.

Führen Sie auf **workstation** den Befehl **lab files-manage start** aus. Dieser Befehl führt ein Startskript aus, das ermittelt, ob der Rechner **servera** im Netzwerk erreichbar ist.

```
[student@workstation ~]$ lab files-manage start
```

- 1. Melden Sie sich mit dem Befehl **ssh** bei **servera** als Benutzer **student** an. Die Systeme sind für die Verwendung von SSH-Schlüsseln zur Authentifizierung konfiguriert. Daher ist kein Passwort erforderlich.

```
[student@workstation ~]$ ssh student@servera  
...output omitted...  
[student@servera ~]$
```

- 2. Erstellen Sie im Benutzerverzeichnis des Benutzers **student** mit dem Befehl **mkdir** drei Unterverzeichnisse: **Music**, **Pictures** und **Videos**.

```
[student@servera ~]$ mkdir Music Pictures Videos
```

- 3. Erstellen Sie anschließend ebenfalls im Benutzerverzeichnis des Benutzers **student** mit dem Befehl **touch** leere Übungsdateien, die in dieser Übung verwendet werden sollen.

- Erstellen Sie sechs Dateien mit Namen im Format **songX.mp3**.
- Erstellen Sie sechs Dateien mit Namen im Format **snapX.jpg**.
- Erstellen Sie sechs Dateien mit Namen im Format **filmX.avi**.

Ersetzen Sie für jeden Satz das „X“ durch die Zahlen 1 bis 6.

```
[student@servera ~]$ touch song1.mp3 song2.mp3 song3.mp3 song4.mp3 \  
song5.mp3 song6.mp3  
[student@servera ~]$ touch snap1.jpg snap2.jpg snap3.jpg snap4.jpg \  
snap5.jpg snap6.jpg
```

**Kapitel 3 |** Verwalten von Dateien über die Befehlszeile

```
[student@servera ~]$ touch film1.avi film2.avi film3.avi film4.avi \
film5.avi film6.avi
[student@servera ~]$ ls -l
total 0
-rw-rw-r--. 1 student student 0 Feb  4 18:23 film1.avi
-rw-rw-r--. 1 student student 0 Feb  4 18:23 film2.avi
-rw-rw-r--. 1 student student 0 Feb  4 18:23 film3.avi
-rw-rw-r--. 1 student student 0 Feb  4 18:23 film4.avi
-rw-rw-r--. 1 student student 0 Feb  4 18:23 film5.avi
-rw-rw-r--. 1 student student 0 Feb  4 18:23 film6.avi
drwxrwxr-x. 2 student student 6 Feb  4 18:23 Music
drwxrwxr-x. 2 student student 6 Feb  4 18:23 Pictures
-rw-rw-r--. 1 student student 0 Feb  4 18:23 snap1.jpg
-rw-rw-r--. 1 student student 0 Feb  4 18:23 snap2.jpg
-rw-rw-r--. 1 student student 0 Feb  4 18:23 snap3.jpg
-rw-rw-r--. 1 student student 0 Feb  4 18:23 snap4.jpg
-rw-rw-r--. 1 student student 0 Feb  4 18:23 snap5.jpg
-rw-rw-r--. 1 student student 0 Feb  4 18:23 snap6.jpg
-rw-rw-r--. 1 student student 0 Feb  4 18:23 song1.mp3
-rw-rw-r--. 1 student student 0 Feb  4 18:23 song2.mp3
-rw-rw-r--. 1 student student 0 Feb  4 18:23 song3.mp3
-rw-rw-r--. 1 student student 0 Feb  4 18:23 song4.mp3
-rw-rw-r--. 1 student student 0 Feb  4 18:23 song5.mp3
-rw-rw-r--. 1 student student 0 Feb  4 18:23 song6.mp3
drwxrwxr-x. 2 student student 6 Feb  4 18:23 Videos
```

- 4. Verschieben Sie ebenfalls im Benutzerverzeichnis des Benutzers **student** die song-Dateien in das Unterverzeichnis **Music**, die snap-Dateien in das Unterverzeichnis **Pictures** und die film-Dateien in das Unterverzeichnis **Videos**.

Wechseln Sie bei der Verteilung von Dateien von einem Speicherort an viele Speicherorte zunächst in das Verzeichnis mit den Quelldateien. Verwenden Sie die einfachste Pfadsyntax (absolut oder relativ), um das Ziel für jede Dateiverwaltungsaufgabe anzugeben.

```
[student@servera ~]$ mv song1.mp3 song2.mp3 song3.mp3 song4.mp3 \
song5.mp3 song6.mp3 Music
[student@servera ~]$ mv snap1.jpg snap2.jpg snap3.jpg snap4.jpg \
snap5.jpg snap6.jpg Pictures
[student@servera ~]$ mv film1.avi film2.avi film3.avi film4.avi \
film5.avi film6.avi Videos
[student@servera ~]$ ls -l Music Pictures Videos
Music:
total 0
-rw-rw-r--. 1 student student 0 Feb  4 18:23 song1.mp3
-rw-rw-r--. 1 student student 0 Feb  4 18:23 song2.mp3
-rw-rw-r--. 1 student student 0 Feb  4 18:23 song3.mp3
-rw-rw-r--. 1 student student 0 Feb  4 18:23 song4.mp3
-rw-rw-r--. 1 student student 0 Feb  4 18:23 song5.mp3
-rw-rw-r--. 1 student student 0 Feb  4 18:23 song6.mp3

Pictures:
total 0
-rw-rw-r--. 1 student student 0 Feb  4 18:23 snap1.jpg
-rw-rw-r--. 1 student student 0 Feb  4 18:23 snap2.jpg
-rw-rw-r--. 1 student student 0 Feb  4 18:23 snap3.jpg
```

**Kapitel 3 |** Verwalten von Dateien über die Befehlszeile

```
-rw-rw-r-- 1 student student 0 Feb  4 18:23 snap4.jpg
-rw-rw-r-- 1 student student 0 Feb  4 18:23 snap5.jpg
-rw-rw-r-- 1 student student 0 Feb  4 18:23 snap6.jpg
```

Videos:

```
total 0
-rw-rw-r-- 1 student student 0 Feb  4 18:23 film1.avi
-rw-rw-r-- 1 student student 0 Feb  4 18:23 film2.avi
-rw-rw-r-- 1 student student 0 Feb  4 18:23 film3.avi
-rw-rw-r-- 1 student student 0 Feb  4 18:23 film4.avi
-rw-rw-r-- 1 student student 0 Feb  4 18:23 film5.avi
-rw-rw-r-- 1 student student 0 Feb  4 18:23 film6.avi
```

- 5. Erstellen Sie ebenfalls im Benutzerverzeichnis des Benutzers **student** drei Unterverzeichnisse zur Organisation der Dateien in Projekten. Nennen Sie die Unterverzeichnisse **friends**, **family** und **work**. Erstellen Sie mit einem einzigen Befehl alle drei Unterverzeichnisse gleichzeitig.

Mithilfe dieser Verzeichnisse ordnen Sie die Dateien in Projekten neu an.

```
[student@servera ~]$ mkdir friends family work
[student@servera ~]$ ls -l
total 0
drwxrwxr-x. 2 student student 6 Feb  4 18:38 family
drwxrwxr-x. 2 student student 6 Feb  4 18:38 friends
drwxrwxr-x. 2 student student 108 Feb  4 18:36 Music
drwxrwxr-x. 2 student student 108 Feb  4 18:36 Pictures
drwxrwxr-x. 2 student student 108 Feb  4 18:36 Videos
drwxrwxr-x. 2 student student 6 Feb  4 18:38 work
```

- 6. Kopieren Sie eine Auswahl neuer Dateien in die Projektverzeichnisse **family** und **friends**. Verwenden Sie beliebig viele Befehle. Sie müssen nicht denselben Befehl verwenden wie im Beispiel. Wechseln Sie für jedes Projekt zuerst in das Projektverzeichnis und kopieren Sie dann die Ausgangsdateien in dieses Verzeichnis. Denken Sie daran, dass Sie Kopien erstellen, die Originaldateien bleiben daher an ihrem ursprünglichen Speicherort, nachdem die Dateien in die Projektverzeichnisse kopiert wurden.

- Kopieren Sie die Dateien (alle Typen), die die Nummern 1 und 2 enthalten, in das Unterverzeichnis **friends**.
- Kopieren Sie die Dateien (alle Typen), die die Nummern 3 und 4 enthalten, in das Unterverzeichnis **family**.

Beim Kopieren von Dateien von mehreren Speicherorten an einen Speicherort empfiehlt Red Hat, dass Sie vor dem Kopieren der Dateien in das Zielverzeichnis wechseln. Verwenden Sie die einfachste Pfadsyntax (absolut oder relativ), um die Quelle für jede Dateiverwaltungsaufgabe anzugeben.

```
[student@servera ~]$ cd friends
[student@servera friends]$ cp ~/Music/song1.mp3 ~/Music/song2.mp3 \
~/Pictures/snap1.jpg ~/Pictures/snap2.jpg ~/Videos/film1.avi \
~/Videos/film2.avi .
[student@servera friends]$ ls -l
total 0
-rw-rw-r-- 1 student student 0 Feb  4 18:42 film1.avi
```

**Kapitel 3 |** Verwalten von Dateien über die Befehlszeile

```
-rw-rw-r-- 1 student student 0 Feb  4 18:42 film2.avi
-rw-rw-r-- 1 student student 0 Feb  4 18:42 snap1.jpg
-rw-rw-r-- 1 student student 0 Feb  4 18:42 snap2.jpg
-rw-rw-r-- 1 student student 0 Feb  4 18:42 song1.mp3
-rw-rw-r-- 1 student student 0 Feb  4 18:42 song2.mp3
[student@servera friends]$ cd ../family
[student@servera family]$ cp ~/Music/song3.mp3 ~/Music/song4.mp3 \
~/Pictures/snap3.jpg ~/Pictures/snap4.jpg ~/Videos/film3.avi \
~/Videos/film4.avi .
[student@servera family]$ ls -l
total 0
-rw-rw-r-- 1 student student 0 Feb  4 18:44 film3.avi
-rw-rw-r-- 1 student student 0 Feb  4 18:44 film4.avi
-rw-rw-r-- 1 student student 0 Feb  4 18:44 snap3.jpg
-rw-rw-r-- 1 student student 0 Feb  4 18:44 snap4.jpg
-rw-rw-r-- 1 student student 0 Feb  4 18:44 song3.mp3
-rw-rw-r-- 1 student student 0 Feb  4 18:44 song4.mp3
```

- 7. Erstellen Sie für Ihr Arbeitsprojekt zusätzliche Kopien.

```
[student@servera family]$ cd ../work
[student@servera work]$ cp ~/Music/song5.mp3 ~/Music/song6.mp3 \
~/Pictures/snap5.jpg ~/Pictures/snap6.jpg \
~/Videos/film5.avi ~/Videos/film6.avi .
[student@servera work]$ ls -l
total 0
-rw-rw-r-- 1 student student 0 Feb  4 18:48 film5.avi
-rw-rw-r-- 1 student student 0 Feb  4 18:48 film6.avi
-rw-rw-r-- 1 student student 0 Feb  4 18:48 snap5.jpg
-rw-rw-r-- 1 student student 0 Feb  4 18:48 snap6.jpg
-rw-rw-r-- 1 student student 0 Feb  4 18:48 song5.mp3
-rw-rw-r-- 1 student student 0 Feb  4 18:48 song6.mp3
```

- 8. Ihre Projektaufgaben sind nun abgeschlossen und es ist an der Zeit, die Projekte zu bereinigen.

Wechseln Sie in das Benutzerverzeichnis des Benutzers **student**. Versuchen Sie beide Projektverzeichnisse, **family** und **friends**, mit einem einzigen **rmdir**-Befehl zu löschen.

```
[student@servera work]$ cd
[student@servera ~]$ rmkdir family friends
rmdir: failed to remove 'family': Directory not empty
rmdir: failed to remove 'friends': Directory not empty
```

Der Befehl **rmdir** sollte fehlschlagen, da beide Unterverzeichnisse Dateien enthalten.

**Kapitel 3 |** Verwalten von Dateien über die Befehlszeile

- 9. Verwenden Sie den Befehl **rm -r**, um beide Unterverzeichnisse, **family** und **friends**, sowie deren Inhalt rekursiv zu löschen.

```
[student@servera ~]$ rm -r family friends
[student@servera ~]$ ls -l
total 0
drwxrwxr-x. 2 student student 108 Feb  4 18:36 Music
drwxrwxr-x. 2 student student 108 Feb  4 18:36 Pictures
drwxrwxr-x. 2 student student 108 Feb  4 18:36 Videos
drwxrwxr-x. 2 student student 108 Feb  4 18:48 work
```

- 10. Löschen Sie alle Dateien im Arbeitsprojekt, löschen Sie jedoch nicht das Arbeitsverzeichnis.

```
[student@servera ~]$ cd work
[student@servera work]$ rm song5.mp3 song6.mp3 snap5.jpg snap6.jpg \
film5.avi film6.avi
[student@servera work]$ ls -l
total 0
```

- 11. Löschen Sie schließlich im Benutzerverzeichnis des Benutzers **student** mit dem Befehl **rmdir** das Verzeichnis **work**. Der Befehl müsste nun erfolgreich sein, da dieses nun leer ist.

```
[student@servera work]$ cd
[student@servera ~]$ rm -r work
[student@servera ~]$ ls -l
total 0
drwxrwxr-x. 2 student student 108 Feb  4 18:36 Music
drwxrwxr-x. 2 student student 108 Feb  4 18:36 Pictures
drwxrwxr-x. 2 student student 108 Feb  4 18:36 Videos
```

- 12. Beenden Sie **servera**.

```
[student@servera ~]$ exit
logout
Connection to servera closed.
[student@workstation ~]$
```

## Beenden

Führen Sie auf **workstation** das Skript **lab files-manage finish** aus, um diese Übung abzuschließen. Das Skript entfernt alle Verzeichnisse und Dateien, die während dieser Übung erstellt wurden.

```
[student@workstation ~]$ lab files-manage finish
```

Hiermit ist die angeleitete Übung beendet.

# Herstellen von Verknüpfungen zwischen Dateien

---

## Ziele

Nach Abschluss dieses Kapitels sollten Sie in der Lage sein, unter Verwendung von Hardlinks und symbolischen Verknüpfungen (oder „Softlinks“) festzulegen, dass mehrere Dateinamen die gleiche Datei referenzieren.

## Verwalten von Verknüpfungen zwischen Dateien

### Hardlinks und Softlinks

Es ist möglich, mehrere Namen zu erstellen, die auf dieselbe Datei verweisen. Es gibt zwei Möglichkeiten, dies zu tun: durch Erstellen eines *Hardlinks* in die Datei oder durch Erstellen eines *Softlinks* (manchmal auch als *symbolische Verknüpfung* bezeichnet) auf die Datei. Jede hat Vor- und Nachteile.

### Erstellen von Hardlinks

Jede Datei beginnt mit einer einzelnen festen Verknüpfung, vom Anfangsnamen bis zu den Daten im Dateisystem. Wenn Sie einen neuen Hardlink zu einer Datei erstellen, erstellen Sie einen anderen Namen, der auf dieselben Daten verweist. Der neue Hardlink verhält sich genauso wie der ursprüngliche Dateiname. Einmal erstellt, können Sie den Unterschied zwischen dem neuen Hardlink und dem ursprünglichen Namen der Datei nicht feststellen.

Sie können mit dem Befehl **ls -l** herausfinden, ob eine Datei mehrere Hardlinks hat. Der Befehl meldet unter anderem die *Verknüpfungsanzahl* jeder Datei, also die Anzahl der Hardlinks, die die Datei hat.

```
[user@host ~]$ pwd
/home/user
[user@host ~]$ ls -l newfile.txt
-rw-r--r--. 1 user user 0 Mar 11 19:19 newfile.txt
```

Im vorherigen Beispiel war die Verknüpfungsanzahl von **newfile.txt** 1. Sie hat genau einen absoluten Pfad, nämlich **/home/user/newfile.txt**.

Sie können mit dem Befehl **ln** einen neuen Hardlink (anderen Namen) erstellen, der auf eine vorhandene Datei verweist. Der Befehl benötigt mindestens zwei Argumente, einen Pfad zu der vorhandenen Datei und den Pfad zu dem Hardlink, den Sie erstellen möchten.

Das folgende Beispiel erstellt den Hardlink **newfile-link2.txt** für die vorhandene Datei **newfile.txt** im Verzeichnis **/tmp**.

```
[user@host ~]$ ln newfile.txt /tmp/newfile-hlink2.txt
[user@host ~]$ ls -l newfile.txt /tmp/newfile-hlink2.txt
-rw-rw-r--. 2 user user 12 Mar 11 19:19 newfile.txt
-rw-rw-r--. 2 user user 12 Mar 11 19:19 /tmp/newfile-hlink2.txt
```

Wenn Sie herausfinden möchten, ob zwei Dateien Hardlinks voneinander sind, können Sie die Option **-i** mit dem Befehl **ls** zum Auflisten der *Inode-Nummer* der Dateien verwenden. Wenn sich die Dateien im selben Dateisystem befinden (wird gleich behandelt) und ihre Inode-Nummern gleich sind, dann sind die Dateien Hardlinks, die auf die gleichen Daten verweisen.

```
[user@host ~]$ ls -il newfile.txt /tmp/newfile-hlink2.txt
8924107 -rw-rw-r--. 2 user user 12 Mar 11 19:19 newfile.txt
8924107 -rw-rw-r--. 2 user user 12 Mar 11 19:19 /tmp/newfile-hlink2.txt
```



### Wichtig

Alle Hardlinks, die auf dieselbe Datei verweisen, verfügen über dieselbe Verknüpfungsanzahl, dieselben Zugriffsberechtigungen, Benutzer- und Gruppenberechtigungen, Zeitstempel und Dateiinhalte. Wird eine dieser Informationen mit einem Hardlink geändert, wird diese neue Information für alle anderen Hardlinks, die auf diese Datei verweisen, ebenfalls übernommen. Dies liegt daran, dass jeder Hardlink auf dieselben Daten auf dem Speichergerät verweist.

Selbst wenn die Originaldatei gelöscht wird, ist der Inhalt der Datei weiter verfügbar, solange mindestens ein Hardlink besteht. Daten werden erst aus dem Speicher gelöscht, wenn der letzte Hardlink gelöscht wird.

```
[user@host ~]$ rm -f newfile.txt
[user@host ~]$ ls -l /tmp/newfile-hlink2.txt
-rw-rw-r--. 1 user user 12 Mar 11 19:19 /tmp/newfile-hlink2.txt
[user@host ~]$ cat /tmp/newfile-hlink2.txt
Hello World
```

## Einschränkungen von Hardlinks

Hardlinks haben einige Einschränkungen. Erstens können Hardlinks nur mit regulären Dateien verwendet werden. Sie können **ln** nicht verwenden, um einen Hardlink zu einem Verzeichnis oder einer speziellen Datei zu erstellen.

Zweitens können Hardlinks nur verwendet werden, wenn sich beide Dateien im selben *Dateisystem* befinden. Die Dateisystemhierarchie kann aus mehreren Speichergeräten bestehen. Abhängig von der Konfiguration Ihres Systems werden dieses Verzeichnis und sein Inhalt möglicherweise auf einem anderen Dateisystem gespeichert, wenn Sie in ein neues Verzeichnis wechseln.

Sie können mit dem Befehl **df** die Verzeichnisse auflisten, die sich auf anderen Dateisystemen befinden. Beispielsweise könnte die Ausgabe folgendermaßen aussehen:

```
[user@host ~]$ df
Filesystem      1K-blocks    Used Available Use% Mounted on
devtmpfs          886788     0   886788  0% /dev
tmpfs            902108     0   902108  0% /dev/shm
tmpfs            902108   8696   893412  1% /run
tmpfs            902108     0   902108  0% /sys/fs/cgroup
/dev/mapper/rhel_rhel8--root 10258432 1630460  8627972 16% /
/dev/sda1        1038336 167128   871208 17% /boot
tmpfs           180420     0   180420  0% /run/user/1000
[user@host ~]$
```

Dateien in zwei verschiedenen „Mounted on“-Verzeichnissen und deren Unterverzeichnissen befinden sich in verschiedenen Dateisystemen. (Die spezifischste Übereinstimmung erhält den Vorrang.) Sie können also für das System in diesem Beispiel einen Hardlink zwischen **/var/tmp/link1** und **/home/user/file** erstellen, weil sie beide Unterverzeichnisse von / sind, aber für kein anderes Verzeichnis auf der Liste. Aber Sie können keinen Hardlink zwischen **/boot/test/badlink** und **/home/user/file** erstellen, weil sich die erste Datei in einem Unterverzeichnis von **/boot** befindet (in der Liste „Mounted on“) und die zweite Datei nicht.

## Erstellen von Softlinks

Der Befehl **ln -s** erstellt einen Softlink, der auch „symbolische Verknüpfung“ genannt wird. Ein Softlink ist keine gewöhnliche Datei, sondern ein besonderer Dateityp, der auf eine vorhandene Datei oder ein vorhandenes Verzeichnis verweist.

Softlinks haben einige Vorteile gegenüber Hardlinks:

- Sie können zwei Dateien in verschiedenen Dateisystemen verknüpfen.
- Sie können auf ein Verzeichnis oder eine spezielle Datei verweisen, nicht nur auf eine reguläre Datei.

Im folgenden Beispiel wird mit dem Befehl **ln -s** ein neuer Softlink mit dem Namen **/tmp/newfile-symlink.txt** für die vorhandene Datei **/home/user/newfile-link2.txt** erstellt.

```
[user@host ~]$ ln -s /home/user/newfile-link2.txt /tmp/newfile-symlink.txt
[user@host ~]$ ls -l newfile-link2.txt /tmp/newfile-symlink.txt
-rw-rw-r--. 1 user user 12 Mar 11 19:19 newfile-link2.txt
lrwxrwxrwx. 1 user user 11 Mar 11 20:59 /tmp/newfile-symlink.txt -> /home/user/
newfile-link2.txt
[user@host ~]$ cat /tmp/newfile-symlink.txt
Soft Hello World
```

Im vorherigen Beispiel ist das erste Zeichen der langen Auflistung für **/tmp/newfile-symlink.txt** **l** anstatt **-**. Dies zeigt an, dass die Datei ein Softlink und keine reguläre Datei ist. (Ein **d** würde angeben, dass die Datei ein Verzeichnis ist.)

Wird die ursprüngliche reguläre Datei gelöscht, verweist der Softlink weiterhin auf die Datei, aber das Ziel ist nicht mehr vorhanden. Ein Softlink, der auf eine fehlende Datei verweist, wird als „dangling soft link“ (verlorener Softlink) bezeichnet.

```
[user@host ~]$ rm -f newfile-link2.txt
[user@host ~]$ ls -l /tmp/newfile-symlink.txt
lrwxrwxrwx. 1 user user 11 Mar 11 20:59 /tmp/newfile-symlink.txt -> /home/user/
newfile-link2.txt
[user@host ~]$ cat /tmp/newfile-symlink.txt
cat: /tmp/newfile-symlink.txt: No such file or directory
```



### Wichtig

Ein Nebeneffekt des verlorenen Softlinks im vorherigen Beispiel ist, dass wenn Sie später eine neue Datei mit dem gleichen Namen wie die gelöschte Datei (`/home/user/newfile-link2.txt`) erstellen, der Softlink nicht mehr „verloren“ ist, sondern auf die neue Datei verweist.

Hardlinks funktionieren nicht auf diese Weise. Wenn Sie einen Hardlink löschen und dann normale Tools verwenden (anstatt `ln`), um eine neue Datei mit demselben Namen zu erstellen, wird die neue Datei nicht mit der alten Datei verknüpft.

Die Funktionsweise von Hardlinks und Softlinks kann wie folgt verglichen werden:

- Ein Hardlink ist ein Name, der auf Daten auf einem Speichergerät verweist
- Ein Softlink ist ein Name, der auf einen anderen Namen verweist, der auf Daten auf einem Speichergerät verweist

Ein Softlink kann auf ein Verzeichnis verweisen. Der Softlink verhält sich dann wie ein Verzeichnis. Wird mit `cd` zum Softlink gewechselt, wird das aktuelle Arbeitsverzeichnis zum verknüpften Verzeichnis. Einige Tools verfolgen möglicherweise, dass Sie über einen Softlink dorthin gelangt sind. Zum Beispiel aktualisiert `cd` standardmäßig Ihr aktuelles Arbeitsverzeichnis mit dem Namen des Softlinks und nicht mit dem Namen des tatsächlichen Verzeichnisses. (Die Option `-P` aktualisiert das Arbeitsverzeichnis stattdessen mit dem Namen des aktuellen Verzeichnisses.)

Im folgenden Beispiel wird ein Softlink mit dem Namen `/home/user/configfiles` erstellt, der auf das Verzeichnis `/etc` verweist.

```
[user@host ~]$ ln -s /etc /home/user/configfiles
[user@host ~]$ cd /home/user/configfiles
[user@host configfiles]$ pwd
/home/user/configfiles
```



### Literaturhinweise

Manpage `ln(1)`

**info** `ln` („`ln`“: Erstellt Verknüpfungen zwischen Dateien)

## ► Angeleitete Übung

# Herstellen von Verknüpfungen zwischen Dateien

In dieser Übung erstellen Sie Hardlinks und symbolische Links und vergleichen die Ergebnisse.

## Ergebnisse

Sie sollten in der Lage sein, Hardlinks und Softlinks zwischen Dateien zu erstellen.

## Bevor Sie Beginnen

Melden Sie sich als Benutzer **student** mit dem Passwort **student** bei **workstation** an.

Führen Sie auf **workstation** den Befehl **lab files-make start** aus. Dieser Befehl führt ein Startskript aus, das ermittelt, ob der Host **servera** im Netzwerk erreichbar ist, und die Dateien und Arbeitsverzeichnisse auf **servera** erstellt.

```
[student@workstation ~]$ lab files-make start
```

- 1. Melden Sie sich mit dem Befehl **ssh** bei **servera** als Benutzer **student** an. Die Systeme sind für die Verwendung von SSH-Schlüsseln zur Authentifizierung konfiguriert. Daher ist kein Passwort erforderlich.

```
[student@workstation ~]$ ssh student@servera  
...output omitted...  
[student@servera ~]$
```

- 2. Erstellen Sie den Hardlink **/home/student/backups/source.backup** für die vorhandene Datei **/home/student/files/source.file**.

- 2.1. Zeigen Sie die Linkanzahl für die Datei **/home/student/files/source.file** an.

```
[student@servera ~]$ ls -l files/source.file  
total 4  
-rw-r--r--. 1 student student 11 Mar 5 21:19 source.file
```

- 2.2. Erstellen Sie den Hardlink **/home/student/backups/source.backup**. Verknüpfen Sie diesen Hardlink mit der Datei **/home/student/files/source.file**.

```
[student@servera ~]$ ln /home/student/files/source.file \  
/home/student/backups/source.backup
```

- 2.3. Überprüfen Sie die Linkanzahl für die Originaldatei **/home/student/files/source.file** und die neue verknüpfte Datei **/home/student/backups/source.backup**. Die Linkanzahl sollte für beide Dateien **2** sein.

```
[student@servera ~]$ ls -l /home/student/files/  
-rw-r--r--. 2 student student 11 Mar 5 21:19 source.file  
[student@servera ~]$ ls -l /home/student/backups/  
-rw-r--r--. 2 student student 11 Mar 5 21:19 source.backup
```

- 3. Erstellen Sie den Softlink **/home/student/tempdir**, der auf das Verzeichnis **/tmp** auf **servera** verweist.

- 3.1. Erstellen Sie den Softlink **/home/student/tempdir** und verknüpfen Sie ihn mit **/tmp**.

```
[student@servera ~]$ ln -s /tmp /home/student/tempdir
```

- 3.2. Überprüfen Sie den neu erstellten Softlink mit dem Befehl **ls -l**.

```
[student@servera ~]$ ls -l /home/student/tempdir  
lrwxrwxrwx. 1 student student 4 Mar 5 22:04 /home/student/tempdir -> /tmp
```

- 4. Beenden Sie **servera**.

```
[student@servera ~]$ exit  
logout  
Connection to servera closed.  
[student@workstation ~]$
```

## Beenden

Führen Sie auf **workstation** das Skript **lab files-make finish** aus, um diese Übung abzuschließen. Dieses Skript entfernt alle Dateien und Verzeichnisse, die während der Übung auf **servera** erstellt wurden.

```
[student@workstation ~]$ lab files-make finish
```

Hiermit ist die angeleitete Übung beendet.

# Abgleichen von Dateinamen mit Shell-Erweiterungen

---

## Ziele

Nach Abschluss dieses Abschnitts sollten Sie in der Lage sein, unter Verwendung der Mustervergleichsfunktionen der Bash-Shell Befehle, die sich auf viele Dateien auswirken, effizient auszuführen.

## Befehlszeilenerweiterungen

Die Bash-Shell bietet mehrere Möglichkeiten zum Erweitern einer Befehlszeile, einschließlich *Mustervergleich*, Benutzerverzeichnis erweiterung, Zeichenfolgen generierung und Variablen substitution. Die wahrscheinlich leistungsstärkste davon ist Funktion zum Abgleich von Pfadnamen, die früher als *Globbing* bezeichnet wurde. Mit der Bash Globbing-Funktion, auch „Platzhalter“ genannt, kann eine große Anzahl von Dateien einfacher verwaltet werden. Mithilfe von „erweiterten“ Metazeichen, die gesuchte Datei- und Pfadnamen vergleichen, werden Befehle für einen bestimmten Satz Dateien auf einmal ausgeführt.

## Mustervergleich

Globbing ist eine Befehlsanalysefunktion für eine Shell, mit der ein Platzhaltermuster in eine Liste passender Pfadnamen aufgelöst wird. Die Metazeichen der Befehlszeile werden vor der Befehlausführung durch die Übereinstimmungsliste ersetzt. Bei Mustern, die keine Übereinstimmungen liefern, wird das Originalmuster als buchstäblicher Text angezeigt. Nachstehend sind gängige Metazeichen und Musterklassen aufgeführt.

**Tabelle der Metazeichen und Übereinstimmungen**

Muster	Übereinstimmungen
*	Eine Zeichenfolge mit null oder mehr Zeichen.
?	Jedes einzelne Zeichen.
[abc...]	Jedes Zeichen in der eingeschlossenen Klasse (eckigen Klammern).
[!abc...]	Alle <i>nicht</i> in der eingeschlossenen Klasse enthaltenen Zeichen.
[^abc...]	Alle <i>nicht</i> in der eingeschlossenen Klasse enthaltenen Zeichen.
[:alpha:]	Alle Zeichen des Alphabets.
[:lower:]	Alle Zeichen in Kleinbuchstaben.
[:upper:]	Alle Zeichen in Großbuchstaben.
[:alnum:]	Alle Zeichen des Alphabets oder Ziffern.
[:punct:]	Alle druckfähigen Zeichen, keine Leerzeichen oder alphanumerischen Zeichen.

Muster	Übereinstimmungen
<code>[:digit:]</code>	Eine beliebige Ziffer von 0 bis 9.
<code>[:space:]</code>	Jeder einzelne Leerraum. Dazu zählen Tabulatoren, Zeilenumbrüche, Absatzwechsel, Seitenvorschub oder Leerzeichen.

Nehmen Sie sich für die nächsten Beispiele an, Sie hätten die folgenden Befehle ausgeführt, um einige Beispieldateien zu erstellen.

```
[user@host ~]$ mkdir glob; cd glob
[user@host glob]$ touch alfa bravo charlie delta echo able baker cast dog easy
[user@host glob]$ ls
able alfa baker bravo cast charlie delta dog easy echo
[user@host glob]$
```

Im ersten Beispiel werden einfache Musterübereinstimmungen mit dem Sternchen (\*) und Fragezeichen (?) sowie eine Klasse von Zeichen verwendet, um Übereinstimmungen einiger dieser Dateinamen zu finden.

```
[user@host glob]$ ls a*
able alfa
[user@host glob]$ ls *a*
able alfa baker bravo cast charlie delta easy
[user@host glob]$ ls [ac]*
able alfa cast charlie
[user@host glob]$ ls ****
able alfa cast easy echo
[user@host glob]$ ls *****?
baker bravo delta
[user@host glob]$
```

## Tilde-Erweiterung

Das Tildezeichen (~) entspricht dem Benutzerverzeichnis des aktuellen Benutzers. Wenn die Tilde eine Zeichenfolge außer einem Schrägstrich (/) vorangeht, interpretiert die Shell die Zeichenfolge bis zu diesem Schrägstrich als Benutzernamen, sofern einer übereinstimmt, und ersetzt die Zeichenfolge durch den absoluten Pfad zum Benutzerverzeichnis dieses Benutzers. Gibt es keine Übereinstimmung mit einem Benutzernamen, wird eine Tilde gefolgt von der Zeichenfolge stattdessen verwendet.

Im folgenden Beispiel wird mit dem Befehl **echo** der Wert des Tilde-Zeichens angezeigt. Mit dem Befehl **echo** können auch die Werte von geschweiften Klammern und variablen Erweiterungszeichen und andere angezeigt werden.

```
[user@host glob]$ echo ~root
/root
[user@host glob]$ echo ~user
/home/user
[user@host glob]$ echo ~/glob
/home/user/glob
[user@host glob]$
```

## Klammernerweiterung

Mit der Klammernerweiterung wird eine beliebige Zeichenfolge generiert. Klammern enthalten eine durch Komma getrennte Liste von Zeichenfolgen oder einen Folgeausdruck. Das Ergebnis beinhaltet den Text vor oder nach der Klammerdefinition. Klammernerweiterungen können ineinander verschachtelt sein. Die Syntax mit zwei Punkten (..) wird ebenfalls zu einer Sequenz erweitert: **{m..p}** wird zu **m n o p** erweitert.

```
[user@host glob]$ echo {Sunday,Monday,Tuesday,Wednesday}.log
Sunday.log Monday.log Tuesday.log Wednesday.log
[user@host glob]$ echo file{1..3}.txt
file1.txt file2.txt file3.txt
[user@host glob]$ echo file{a..c}.txt
filea.txt fileb.txt filec.txt
[user@host glob]$ echo file{a,b}{1,2}.txt
filea1.txt filea2.txt fileb1.txt fileb2.txt
[user@host glob]$ echo file{a[1,2],b,c}.txt
filea1.txt filea2.txt fileb.txt filec.txt
[user@host glob]$
```

Eine praktische Anwendung der geschweiften Klammer besteht darin, schnell eine Reihe von Dateien oder Verzeichnissen zu erstellen.

```
[user@host glob]$ mkdir ..../RHEL{6,7,8}
[user@host glob]$ ls ..../RHEL*
RHEL6 RHEL7 RHEL8
[user@host glob]$
```

## Variablerweiterung

Eine Variable verhält sich wie ein benannter Container, in dem ein Wert gespeichert werden kann. Variablen ermöglichen den einfachen Zugriff auf und die Änderung der gespeicherten Daten entweder über die Befehlszeile oder über ein Shell-Skript.

Sie können mit der folgenden Syntax einer Variable Daten als Wert zuweisen:

```
[user@host ~]$ VARIABLENAME=value
```

Sie können die Variablerweiterung verwenden, um den Variablennamen in der Befehlszeile in seinen Wert zu konvertieren. Wenn eine Zeichenfolge mit einem Dollarzeichen (\$) beginnt, dann versucht die Shell, den Rest der Zeichenfolge als Variablennamen zu verwenden und ihn durch den Wert zu ersetzen, den die Variable hat.

```
[user@host ~]$ USERNAME=operator
[user@host ~]$ echo $USERNAME
operator
```

Um Fehler aufgrund anderer Shell-Erweiterungen zu vermeiden, können Sie den Namen der Variablen in geschweifte Klammern setzen, z. B.  **\${VARIABLENAME}** .

```
[user@host ~]$ USERNAME=operator  
[user@host ~]$ echo ${USERNAME}  
operator
```

Shell-Variablen und ihre Verwendung werden später in diesem Kurs ausführlicher behandelt.

## Befehlssubstitution

Mit der Befehlssubstitution kann die Ausgabe eines Befehls den eigentlichen Befehl in der Befehlszeile ersetzen. Die Befehlssubstitution erfolgt, wenn ein Befehl in Klammern eingeschlossen und ein Dollar-Zeichen (\$) vorangestellt ist. Mit dem Format **\$(*Befehl*)** können mehrere Befehlserweiterungen ineinander verschachtelt werden.

```
[user@host glob]$ echo Today is $(date +%A).  
Today is Wednesday.  
[user@host glob]$ echo The time is $(date +%M) minutes past $(date +%l%p).  
The time is 26 minutes past 11AM.  
[user@host glob]$
```



### Anmerkung

Eine ältere Form der Befehlssubstitution verwendet Backticks: `**Befehl**`.

Das Format mit Backticks hat Nachteile: 1) Backticks sind leicht mit einfachen Anführungszeichen zu verwechseln und 2) Backticks können nicht ineinander verschachtelt werden.

## Schützen von Argumenten vor Erweiterung

Viele Zeichen haben in der Bash-Shell eine bestimmte Bedeutung. Damit die Shell keine Shell-Erweiterungen für Teile Ihrer Befehlszeile durchführt, können Sie Zeichen und Zeichenfolgen *in Anführungszeichen* setzen und mit *Escape-Zeichen* versehen.

Der Backslash (\) ist ein Escape-Zeichen in der Bash-Shell. Es schützt das Zeichen, der ihm unmittelbar folgt, vor Erweiterung.

```
[user@host glob]$ echo The value of $HOME is your home directory.  
The value of /home/user is your home directory.  
[user@host glob]$ echo The value of \$HOME is your home directory.  
The value of $HOME is your home directory.  
[user@host glob]$
```

Im vorherigen Beispiel hat der Schutz des Dollarzeichens vor Erweiterung dazu geführt, dass Bash es als reguläres Zeichen behandelt und keine variable Erweiterung für **\$HOME** durchgeführt hat.

Zum Schutz langerer Zeichenfolgen werden diese in einfache ('') oder doppelte ("") Anführungszeichen eingeschlossen. Sie haben eine geringfügig unterschiedliche Wirkung. Einfache Anführungszeichen stoppen alle Shell-Erweiterungen. Doppelte Anführungszeichen stoppen die *meisten* Shell-Erweiterungen.

Verwenden Sie doppelte Anführungszeichen zur Unterdrückung von Globbing und von Shell-Erweiterung, ohne gleichzeitig auch Befehls- und Variablensubstitution zu unterdrücken.

```
[user@host glob]$ myhost=$(hostname -s); echo $myhost
host
[user@host glob]$ echo "***** hostname is ${myhost} *****"
***** hostname is host *****
[user@host glob]$
```

Wenn der gesamte Text exakt wie eingegeben interpretiert werden soll, verwenden Sie einzelne Anführungszeichen.

```
[user@host glob]$ echo "Will variable $myhost evaluate to $(hostname -s)?"
Will variable host evaluate to host?
[user@host glob]$ echo 'Will variable $myhost evaluate to $(hostname -s)?'
Will variable $myhost evaluate to $(hostname -s)?
[user@host glob]$
```



### Wichtig

Das einzelne Anführungszeichen (') und der Backtick (`) der Befehlssubstitution können leicht auf dem Bildschirm und auf der Tastatur verwechselt werden. Wenn Sie die beiden Zeichen verwechseln, führt dies zu unerwartetem Shell-Verhalten.



### Literaturhinweise

Manpages **bash(1)**, **cd(1)**, **glob(7)**, **isalpha(3)**, **ls(1)**, **path\_resolution(7)** und **pwd(1)**

## ► Quiz

# Abgleichen von Dateinamen mit Shell-Erweiterungen

Wählen Sie die richtigen Antworten auf die folgenden Fragen aus:

► 1. Welches Muster passt nur zu Dateinamen, die mit „b“ enden?

- a. **b\***
- b. \***b**
- c. \***b\***
- d. [!b]\*

► 2. Welches Muster passt nur zu Dateinamen, die mit „b“ beginnen?

- a. **b\***
- b. \***b**
- c. \***b\***
- d. [!b]\*

► 3. Welches Muster passt nur zu Dateinamen, deren erstes Zeichen kein „b“ ist?

- a. **b\***
- b. \***b**
- c. \***b\***
- d. [!b]\*

► 4. Welches Muster passt zu allen Dateinamen, die ein „b“ enthalten?

- a. **b\***
- b. \***b**
- c. \***b\***
- d. [!b]\*

► 5. Welches Muster passt nur zu Dateinamen, die eine Zahl enthalten?

- a. \*#\*
- b. \*[[:digit:]]\*
- c. \*[digit]\*
- d. [0-9]

► **6. Welches Muster passt nur zu Dateinamen, die mit einem Großbuchstaben beginnen?**

- a. ^?\*
- b. ^\*
- c. [upper]\*
- d. [:upper:]\*
- e. [[CAP]]\*

► **7. Welches Muster passt nur zu Dateinamen mit mindestens drei Zeichen?**

- a. ???\*
- b. ???
- c. \3\*
- d. +++\*
- e. ...\*

## ► Lösung

# Abgleichen von Dateinamen mit Shell-Erweiterungen

Wählen Sie die richtigen Antworten auf die folgenden Fragen aus:

► 1. Welches Muster passt nur zu Dateinamen, die mit „b“ enden?

- a. `b*`
- b. `*b`
- c. `*b*`
- d. `[!b]*`

► 2. Welches Muster passt nur zu Dateinamen, die mit „b“ beginnen?

- a. `b*`
- b. `*b`
- c. `*b*`
- d. `[!b]*`

► 3. Welches Muster passt nur zu Dateinamen, deren erstes Zeichen kein „b“ ist?

- a. `b*`
- b. `*b`
- c. `*b*`
- d. `[!b]*`

► 4. Welches Muster passt zu allen Dateinamen, die ein „b“ enthalten?

- a. `b*`
- b. `*b`
- c. `*b*`
- d. `[!b]*`

► 5. Welches Muster passt nur zu Dateinamen, die eine Zahl enthalten?

- a. `*#*`
- b. `*[[digit]]*`
- c. `*[digit]*`
- d. `[0-9]`

► **6. Welches Muster passt nur zu Dateinamen, die mit einem Großbuchstaben beginnen?**

- a. ^?\*
- b. ^\*
- c. [upper]\*
- d. [[:upper:]]\*
- e. [[CAP]]\*

► **7. Welches Muster passt nur zu Dateinamen mit mindestens drei Zeichen?**

- a. ???\*
- b. ???
- c. \3\*
- d. +++\*
- e. . . \*

## ► Praktische Übung

# Verwalten von Dateien über die Befehlszeile

### Leistungscheckliste

In dieser praktischen Übung erstellen, verschieben und entfernen Sie Dateien und Verzeichnisse effizient mit der Shell und einer Reihe von Techniken zum Dateinamensabgleich.

### Ergebnisse

Es werden folgende Fähigkeiten vermittelt:

- Verwenden von Platzhaltern zum Suchen und Bearbeiten von Dateien

### Bevor Sie Beginnen

Melden Sie sich als Benutzer **student** mit dem Passwort **student** bei **workstation** an.

Führen Sie auf **workstation** den Befehl **lab files-review start** aus. Der Befehl führt ein Startskript aus, das ermittelt, ob der Rechner **serverb** im Netzwerk erreichbar ist.

```
[student@workstation ~]$ lab files-review start
```

1. Melden Sie sich mit dem Befehl **ssh** bei **serverb** als **student** an. Die Systeme sind für die Verwendung von SSH-Schlüsseln zur Authentifizierung konfiguriert. Daher ist kein Passwort erforderlich.
2. Bevor Sie Projektdateien erstellen, erstellen Sie mit dem Befehl **mkdir** mit Klammererweiterung leere Projektplanungsdokumente im Verzeichnis **/home/student/Documents/project\_plans**. (Hinweis: Wenn **~/Documents** nicht vorhanden ist, wird sie mit der Option **-p** des Befehls **mkdir** erstellt.)  
Erstellen Sie zwei leere Dateien im Verzeichnis **~/Documents/project\_plans**: **season1\_project\_plan.odf** und **season2\_project\_plan.odf**.
3. Erstellen Sie Sätze mit leeren Übungsdateien für diese Übung. Verwenden Sie die Lösung zum Lernen und Üben, wenn Sie eine vorgesehene Tastenkombination zur Shell-Erweiterung nicht sofort verstehen. Mit der Tab-Vervollständigung der Shell können Sie Dateipfadnamen einfach suchen.  
Erstellen Sie insgesamt 12 Dateien mit den Namen **tv\_seasonX\_episodeY.ogg**. Ersetzen Sie dabei X durch die Staffelnummer und Y durch die Folge dieser Staffel – für zwei Staffeln mit jeweils sechs Folgen.
4. Sie sind Autor einer erfolgreichen Reihe von Mystery-Romanen. Die Kapitel Ihres nächsten Bestsellers werden gerade für die Veröffentlichung redigiert. Erstellen Sie insgesamt acht Dateien mit den Namen **mystery\_chapterX.odf**. Ersetzen Sie X durch die Nummern 1 bis 8.
5. Erstellen Sie mit einem einzigen Befehl zwei Unterverzeichnisse mit den Namen **season1** und **season2** unter dem Verzeichnis **Videos**, um die Fernsehfolgen zu strukturieren.

**Kapitel 3 |** Verwalten von Dateien über die Befehlszeile

6. Verschieben Sie die entsprechenden Fernsehfolgen in die Staffelunterverzeichnisse. Verwenden Sie nur zwei Befehle und geben Sie damit die Ziele mit relativer Syntax an.
7. Erstellen Sie mit nur einem Befehl eine zweistufige Verzeichniss hierarchie zur Strukturierung der Kapitel des Mystery-Buches. Erstellen Sie **my\_bestseller** unter dem Verzeichnis **Documents** und **chapters** unter dem neuen Verzeichnis **my\_bestseller**.
8. Erstellen Sie mit einem einzigen Befehl direkt unter dem Verzeichnis **my\_bestseller** drei weitere Unterverzeichnisse. Legen Sie **editor**, **changes** und **vacation** als Namen für die Unterverzeichnisse fest. Die Option **-p** (create parents) wird nicht benötigt, da das übergeordnete Verzeichnis **my\_bestseller** bereits vorhanden ist.
9. Wechseln Sie in das Verzeichnis **chapters**. Geben Sie mit der Tilde-Taste (~) für das Benutzerverzeichnis die Quelldateien an und verschieben Sie alle Buchkapitel in das Verzeichnis **chapters**, das nun Ihr aktuelles Verzeichnis ist. Wie sieht die einfachste Syntax zur Angabe des Zielverzeichnisses aus?
10. Sie haben die ersten beiden Kapitel zur Korrektur an den Editor geschickt. Verschieben Sie nur diese beiden Kapitel in das Verzeichnis **editor**, um zu vermeiden, dass sie während der Überprüfung geändert werden. Verwenden Sie ausgehend vom Kapitel-Unterverzeichnis die Klammererweiterung mit einem Bereich, um die zu verschiebenden Kapiteldateinamen und einen relativen Pfad für das Zielverzeichnis anzugeben.
11. Sie möchten im Urlaub die Kapitel 7 und 8 schreiben. Verschieben Sie die Dateien mit einem einzigen Befehl aus dem Verzeichnis **chapters** in das Verzeichnis **vacation**. Geben Sie die Kapiteldateinamen mit der Klammererweiterung mit einer Liste von Zeichenfolgen und ohne Platzhalterzeichen an.
12. Ändern Sie Ihr Arbeitsverzeichnis in **~/Videos/season2** und kopieren Sie dann die erste Folge der Staffel in das Verzeichnis **vacation**.
13. Wechseln Sie mit einem einzigen **cd**-Befehl aus Ihrem Arbeitsverzeichnis in das Verzeichnis **~/Documents/my\_bestseller/vacation**. Listen Sie die Dateien auf. Verwenden Sie das Argument für das vorherige Arbeitsverzeichnis, um zum Verzeichnis **season2** zurückzukehren. (Dies ist erfolgreich, wenn der letzte Verzeichniswechsel mit dem Befehl **cd** mit einem Befehl statt mit mehreren **cd**-Befehlen erfolgt ist.) Kopieren Sie im Verzeichnis **season2** die Datei der Episode 2 in das Verzeichnis **vacation**. Verwenden Sie die Tastenkombination erneut, um zum Verzeichnis **vacation** zurückzukehren.
14. Die Autoren der Kapitel 5 und 6 möchten mit möglichen Änderungen experimentieren. Kopieren Sie die beiden Dateien aus dem Verzeichnis **~/Documents/my\_bestseller/chapters** in das Verzeichnis **~/Documents/my\_bestseller/changes**, um zu verhindern, dass durch diese Änderungen die Originaldateien geändert werden. Navigieren Sie zum Verzeichnis **~/Documents/my\_bestseller**. Geben Sie mit dem Mustervergleich mit eckigen Klammern an, welche Kapitelnummern mit dem Dateinamenargument des Befehls **cp** übereinstimmen sollen.
15. Wechseln Sie aus dem aktuellen Verzeichnis in das Verzeichnis **changes**. Kopieren Sie mit dem Befehl **date +%F** mit Befehlssubstitution **mystery\_chapter5.odf** in eine neue Datei, die das vollständige Datum enthält. Der Name sollte die Form **mystery\_chapter5\_YYYY-MM-DD.odf** haben.  
Erstellen Sie eine weitere Kopie von **mystery\_chapter5.odf** und hängen Sie den aktuellen Zeitstempel an (als Anzahl der Sekunden seit der Epoche 1970-01-01 00:00 UTC), damit ein eindeutiger Dateiname erstellt wird. Verwenden Sie dazu den Befehl **date +%s** mit Befehlssubstitution.

16. Nach weiterer Prüfung entscheiden Sie, dass die Plotänderungen nicht erforderlich sind. Löschen Sie das Verzeichnis **changes**.  
Navigieren Sie ggf. zum Verzeichnis **changes** und löschen Sie alle Dateien in dem Verzeichnis. Sie können ein Verzeichnis nicht löschen, solange es das aktuelle Arbeitsverzeichnis ist. Wechseln Sie in das übergeordnete Verzeichnis des Verzeichnisses **changes**. Versuchen Sie, das leere Verzeichnis mit dem Befehl **rm** ohne die rekursive Option **-r** zu löschen. Dieser Versuch müsste fehlschlagen. Löschen Sie schließlich mit dem Befehl **rmdir** das leere Verzeichnis. Dieser Befehl wird erfolgreich ausgeführt.
17. Nach dem Urlaubsende wird das Verzeichnis **vacation** nicht mehr benötigt. Löschen Sie es mit dem Befehl **rm** mit der Option *recursive*.  
Kehren Sie abschließend in das Benutzerverzeichnis des Benutzers **student** zurück.
18. Erstellen Sie den Hardlink **~/Documents/backups/season2\_project\_plan.odf.back** auf die Datei **~/Documents/project\_plans/season2\_project\_plan.odf**. Ein Hardlink schützt vor versehentlichem Löschen der Originaldatei und hält die Backup-Datei bei Änderungen an der Originaldatei auf dem neuesten Stand.
19. Beenden Sie **serverb**.

```
[student@serverb ~]$ exit  
logout  
Connection to serverb closed.  
[student@workstation ~]$
```

## Bewertung

Führen Sie auf **workstation** das Skript **lab files-review grade** aus, um den Erfolg dieser praktischen Übung zu bestätigen.

```
[student@workstation ~]$ lab files-review grade
```

## Beenden

Führen Sie auf **workstation** das Skript **lab files-review finish** aus, um diese praktische Übung abzuschließen. Dieses Skript entfernt alle Dateien und Verzeichnisse, die während der Übung auf **serverb** erstellt wurden.

```
[student@workstation ~]$ lab files-review finish
```

Hiermit wird die praktische Übung abgeschlossen.

## ► Lösung

# Verwalten von Dateien über die Befehlszeile

### Leistungscheckliste

In dieser praktischen Übung erstellen, verschieben und entfernen Sie Dateien und Verzeichnisse effizient mit der Shell und einer Reihe von Techniken zum Dateinamensabgleich.

### Ergebnisse

Es werden folgende Fähigkeiten vermittelt:

- Verwenden von Platzhaltern zum Suchen und Bearbeiten von Dateien

### Bevor Sie Beginnen

Melden Sie sich als Benutzer **student** mit dem Passwort **student** bei **workstation** an.

Führen Sie auf **workstation** den Befehl **lab files-review start** aus. Der Befehl führt ein Startskript aus, das ermittelt, ob der Rechner **serverb** im Netzwerk erreichbar ist.

```
[student@workstation ~]$ lab files-review start
```

1. Melden Sie sich mit dem Befehl **ssh** bei **serverb** als **student** an. Die Systeme sind für die Verwendung von SSH-Schlüsseln zur Authentifizierung konfiguriert. Daher ist kein Passwort erforderlich.

```
[student@workstation ~]$ ssh student@serverb
...output omitted...
[student@serverb ~]$
```

2. Bevor Sie Projektdateien erstellen, erstellen Sie mit dem Befehl **mkdir** mit Klammererweiterung leere Projektplanungsdokumente im Verzeichnis **/home/student/Documents/project\_plans**. (Hinweis: Wenn **~/Documents** nicht vorhanden ist, wird sie mit der Option **-p** des Befehls **mkdir** erstellt.)

Erstellen Sie zwei leere Dateien im Verzeichnis **~/Documents/project\_plans**: **season1\_project\_plan.odf** und **season2\_project\_plan.odf**.

```
[student@serverb ~]$ mkdir -p ~/Documents/project_plans
[student@serverb ~]$ touch \
~/Documents/project_plans/{season1,season2}_project_plan.odf
[student@serverb ~]$ ls -1R Documents/
Documents/:
total 0
drwxrwxr-x. 2 student student 70 Jan 31 18:20 project_plans

Documents/project_plans:
```

```
total 0
-rw-rw-r-- 1 student student 0 Jan 31 18:20 season1_project_plan.odf
-rw-rw-r-- 1 student student 0 Jan 31 18:20 season2_project_plan.odf
```

3. Erstellen Sie Sätze mit leeren Übungsdateien für diese Übung. Verwenden Sie die Lösung zum Lernen und Üben, wenn Sie eine vorgesehene Tastenkombination zur Shell-Erweiterung nicht sofort verstehen. Mit der Tab-Vervollständigung der Shell können Sie Dateipfadnamen einfach suchen.

Erstellen Sie insgesamt 12 Dateien mit den Namen **tv\_seasonX\_episodeY.ogg**. Ersetzen Sie dabei X durch die Staffelnummer und Y durch die Folge dieser Staffel – für zwei Staffeln mit jeweils sechs Folgen.

```
[student@serverb ~]$ touch tv_season{1..2}_episode{1..6}.ogg
[student@serverb ~]$ ls tv*
tv_season1_episode1.ogg  tv_season1_episode5.ogg  tv_season2_episode3.ogg
tv_season1_episode2.ogg  tv_season1_episode6.ogg  tv_season2_episode4.ogg
tv_season1_episode3.ogg  tv_season2_episode1.ogg  tv_season2_episode5.ogg
tv_season1_episode4.ogg  tv_season2_episode2.ogg  tv_season2_episode6.ogg
```

4. Sie sind Autor einer erfolgreichen Reihe von Mystery-Romanen. Die Kapitel Ihres nächsten Bestsellers werden gerade für die Veröffentlichung redigiert. Erstellen Sie insgesamt acht Dateien mit den Namen **mystery\_chapterX.odf**. Ersetzen Sie X durch die Nummern 1 bis 8.

```
[student@serverb ~]$ touch mystery_chapter{1..8}.odf
[student@serverb ~]$ ls mys*
mystery_chapter1.odf  mystery_chapter4.odf  mystery_chapter7.odf
mystery_chapter2.odf  mystery_chapter5.odf  mystery_chapter8.odf
mystery_chapter3.odf  mystery_chapter6.odf
```

5. Erstellen Sie mit einem einzigen Befehl zwei Unterverzeichnisse mit den Namen **season1** und **season2** unter dem Verzeichnis **Videos**, um die Fernsehfolgen zu strukturieren.

```
[student@serverb ~]$ mkdir -p Videos/season{1..2}
[student@serverb ~]$ ls Videos
season1  season2
```

6. Verschieben Sie die entsprechenden Fernsehfolgen in die Staffelunterverzeichnisse. Verwenden Sie nur zwei Befehle und geben Sie damit die Ziele mit relativer Syntax an.

```
[student@serverb ~]$ mv tv_season1* Videos/season1
[student@serverb ~]$ mv tv_season2* Videos/season2
[student@serverb ~]$ ls -R Videos
Videos:
season1  season2

Videos/season1:
tv_season1_episode1.ogg  tv_season1_episode3.ogg  tv_season1_episode5.ogg
tv_season1_episode2.ogg  tv_season1_episode4.ogg  tv_season1_episode6.ogg

Videos/season2:
tv_season2_episode1.ogg  tv_season2_episode3.ogg  tv_season2_episode5.ogg
tv_season2_episode2.ogg  tv_season2_episode4.ogg  tv_season2_episode6.ogg
```

**Kapitel 3 |** Verwalten von Dateien über die Befehlszeile

7. Erstellen Sie mit nur einem Befehl eine zweistufige Verzeichnishierarchie zur Strukturierung der Kapitel des Mystery-Buches. Erstellen Sie **my\_bestseller** unter dem Verzeichnis **Documents** und **chapters** unter dem neuen Verzeichnis **my\_bestseller**.

```
[student@serverb ~]$ mkdir -p Documents/my_bestseller/chapters
[student@serverb ~]$ ls -R Documents
Documents:
my_bestseller project_plans

Documents/my_bestseller:
chapters

Documents/my_bestseller/chapters:
Documents/project_plans:
season1_project_plan.odf season2_project_plan.odf
```

8. Erstellen Sie mit einem einzigen Befehl direkt unter dem Verzeichnis **my\_bestseller** drei weitere Unterverzeichnisse. Legen Sie **editor**, **changes** und **vacation** als Namen für die Unterverzeichnisse fest. Die Option **-p** (create parents) wird nicht benötigt, da das übergeordnete Verzeichnis **my\_bestseller** bereits vorhanden ist.

```
[student@serverb ~]$ mkdir Documents/my_bestseller/{editor,changes,vacation}
[student@serverb ~]$ ls -R Documents
Documents:
my_bestseller project_plans

Documents/my_bestseller:
changes chapters editor vacation

Documents/my_bestseller/changes:

Documents/my_bestseller/chapters:

Documents/my_bestseller/editor:

Documents/my_bestseller/vacation:

Documents/project_plans:
season1_project_plan.odf season2_project_plan.odf
```

9. Wechseln Sie in das Verzeichnis **chapters**. Geben Sie mit der Tilde-Taste (~) für das Benutzerverzeichnis die Quelldateien an und verschieben Sie alle Buchkapitel in das Verzeichnis **chapters**, das nun Ihr aktuelles Verzeichnis ist. Wie sieht die einfachste Syntax zur Angabe des Zielverzeichnisses aus?

```
[student@serverb ~]$ cd Documents/my_bestseller/chapters
[student@serverb chapters]$ mv ~/mystery_chapter* .
[student@serverb chapters]$ ls
mystery_chapter1.odf mystery_chapter4.odf mystery_chapter7.odf
mystery_chapter2.odf mystery_chapter5.odf mystery_chapter8.odf
mystery_chapter3.odf mystery_chapter6.odf
```

**Kapitel 3 |** Verwalten von Dateien über die Befehlszeile

10. Sie haben die ersten beiden Kapitel zur Korrektur an den Editor geschickt. Verschieben Sie nur diese beiden Kapitel in das Verzeichnis **editor**, um zu vermeiden, dass sie während der Überprüfung geändert werden. Verwenden Sie ausgehend vom Kapitel-Unterverzeichnis die Klammererweiterung mit einem Bereich, um die zu verschiebenden Kapiteldateinamen und einen relativen Pfad für das Zielverzeichnis anzugeben.

```
[student@serverb chapters]$ mv mystery_chapter{1..2}.odf .../editor  
[student@serverb chapters]$ ls  
mystery_chapter3.odf mystery_chapter5.odf mystery_chapter7.odf  
mystery_chapter4.odf mystery_chapter6.odf mystery_chapter8.odf  
[student@serverb chapters]$ ls .../editor  
mystery_chapter1.odf mystery_chapter2.odf
```

11. Sie möchten im Urlaub die Kapitel 7 und 8 schreiben. Verschieben Sie die Dateien mit einem einzigen Befehl aus dem Verzeichnis **chapters** in das Verzeichnis **vacation**. Geben Sie die Kapiteldateinamen mit der Klammererweiterung mit einer Liste von Zeichenfolgen und ohne Platzhalterzeichen an.

```
[student@serverb chapters]$ mv mystery_chapter{7,8}.odf ..../vacation  
[student@serverb chapters]$ ls  
mystery_chapter3.odf mystery_chapter5.odf  
mystery_chapter4.odf mystery_chapter6.odf  
[student@serverb chapters]$ ls ..../vacation  
mystery_chapter7.odf mystery_chapter8.odf
```

12. Ändern Sie Ihr Arbeitsverzeichnis in **~/Videos/season2** und kopieren Sie dann die erste Folge der Staffel in das Verzeichnis **vacation**.

```
[student@serverb chapters]$ cd ~/Videos/season2  
[student@serverb season2]$ cp *episode1.ogg ~/Documents/my_bestseller/vacation
```

13. Wechseln Sie mit einem einzigen **cd**-Befehl aus Ihrem Arbeitsverzeichnis in das Verzeichnis **~/Documents/my\_bestseller/vacation**. Listen Sie die Dateien auf. Verwenden Sie das Argument für das vorherige Arbeitsverzeichnis, um zum Verzeichnis **season2** zurückzukehren. (Dies ist erfolgreich, wenn der letzte Verzeichniswechsel mit dem Befehl **cd** mit einem Befehl statt mit mehreren **cd**-Befehlen erfolgt ist.) Kopieren Sie im Verzeichnis **season2** die Datei der Episode 2 in das Verzeichnis **vacation**. Verwenden Sie die Tastenkombination erneut, um zum Verzeichnis **vacation** zurückzukehren.

```
[student@serverb season2]$ cd ~/Documents/my_bestseller/vacation  
[student@serverb vacation]$ ls  
mystery_chapter7.odf mystery_chapter8.odf tv_season2_episode1.ogg  
[student@serverb vacation]$ cd -  
/home/ec2-user/Videos/season2  
[student@serverb season2]$ cp *episode2.ogg ~/Documents/my_bestseller/vacation  
[student@serverb vacation]$ cd -  
/home/ec2-user/Documents/my_bestseller/vacation  
[student@serverb vacation]$ ls  
mystery_chapter7.odf tv_season2_episode1.ogg  
mystery_chapter8.odf tv_season2_episode2.ogg
```

14. Die Autoren der Kapitel 5 und 6 möchten mit möglichen Änderungen experimentieren. Kopieren Sie die beiden Dateien aus dem Verzeichnis **~/Documents/my\_bestseller/**

**Kapitel 3 |** Verwalten von Dateien über die Befehlszeile

**chapters** in das Verzeichnis **~/Documents/my\_bestseller/changes**, um zu verhindern, dass durch diese Änderungen die Originaldateien geändert werden. Navigieren Sie zum Verzeichnis **~/Documents/my\_bestseller**. Geben Sie mit dem Mustervergleich mit eckigen Klammern an, welche Kapitelnummern mit dem Dateinamenargument des Befehls **cp** übereinstimmen sollen.

```
[student@serverb vacation]$ cd ~/Documents/my_bestseller
[student@serverb my_bestseller]$ cp chapters/mystery_chapter[56].odf changes
[student@serverb my_bestseller]$ ls chapters
mystery_chapter3.odf mystery_chapter5.odf
mystery_chapter4.odf mystery_chapter6.odf
[student@serverb my_bestseller]$ ls changes
mystery_chapter5.odf mystery_chapter6.odf
```

15. Wechseln Sie aus dem aktuellen Verzeichnis in das Verzeichnis **changes**.

Kopieren Sie mit dem Befehl **date +%F** mit Befehlssubstitution **mystery\_chapter5.odf** in eine neue Datei, die das vollständige Datum enthält. Der Name sollte die Form **mystery\_chapter5\_YYYY-MM-DD.odf** haben.

Erstellen Sie eine weitere Kopie von **mystery\_chapter5.odf** und hängen Sie den aktuellen Zeitstempel an (als Anzahl der Sekunden seit der Epoche 1970-01-01 00:00 UTC), damit ein eindeutiger Dateiname erstellt wird. Verwenden Sie dazu den Befehl **date +%s** mit Befehlssubstitution.

```
[student@serverb my_bestseller]$ cd changes
[student@serverb changes]$ cp mystery_chapter5.odf \
mystery_chapter5_$(date +%F).odf
[student@serverb changes]$ cp mystery_chapter5.odf \
mystery_chapter5_$(date +%s).odf
[student@serverb changes]$ ls
mystery_chapter5_1492545076.odf mystery_chapter5.odf
mystery_chapter5_2017-04-18.odf mystery_chapter6.odf
```

16. Nach weiterer Prüfung entscheiden Sie, dass die Plotänderungen nicht erforderlich sind.

Löschen Sie das Verzeichnis **changes**.

Navigieren Sie ggf. zum Verzeichnis **changes** und löschen Sie alle Dateien in dem Verzeichnis. Sie können ein Verzeichnis nicht löschen, solange es das aktuelle Arbeitsverzeichnis ist. Wechseln Sie in das übergeordnete Verzeichnis des Verzeichnisses **changes**. Versuchen Sie, das leere Verzeichnis mit dem Befehl **rm** ohne die rekursive Option **-r** zu löschen. Dieser Versuch müsste fehlschlagen. Löschen Sie schließlich mit dem Befehl **rmdir** das leere Verzeichnis. Dieser Befehl wird erfolgreich ausgeführt.

```
[student@serverb changes]$ rm mystery*
[student@serverb changes]$ cd ..
[student@serverb my_bestseller]$ rm changes
rm: cannot remove 'changes': Is a directory
[student@serverb my_bestseller]$ rm -r changes
[student@serverb my_bestseller]$ ls
chapters editor vacation
```

17. Nach dem Urlaubsende wird das Verzeichnis **vacation** nicht mehr benötigt. Löschen Sie es mit dem Befehl **rm** mit der Option *recursive*.

Kehren Sie abschließend in das Benutzerverzeichnis des Benutzers **student** zurück.

```
[student@serverb my_bestseller]$ rm -r vacation
[student@serverb my_bestseller]$ ls
chapters editor
[student@serverb my_bestseller]$ cd
[student@serverb ~]$
```

18. Erstellen Sie den Hardlink **~/Documents/backups/season2\_project\_plan.odf.back** auf die Datei **~/Documents/project\_plans/season2\_project\_plan.odf**. Ein Hardlink schützt vor versehentlichem Löschen der Originaldatei und hält die Backup-Datei bei Änderungen an der Originaldatei auf dem neuesten Stand.

Beachten Sie, dass die Anzahl der Links für beide Dateien, **season2\_project\_plan.odf.back** und **season2\_project\_plan.odf, 2** ist.

```
[student@serverb ~]$ mkdir ~/Documents/backups
[student@serverb ~]$ ln ~/Documents/project_plans/season2_project_plan.odf \
~/Documents/backups/season2_project_plan.odf.back
[student@serverb ~]$ ls -lR ~/Documents/
/home/student/Documents/:
total 0
drwxrwxr-x. 2 student student 43 Jan 31 18:59 backups
drwxrwxr-x. 4 student student 36 Jan 31 19:42 my_bestseller
drwxrwxr-x. 2 student student 70 Jan 31 18:20 project_plans

/home/student/Documents/backups:
total 4
-rw-rw-r--. 2 student student 0 Jan 31 19:05 season2_project_plan.odf.back

/home/student/Documents/my_bestseller:
total 0
drwxrwxr-x. 2 student student 118 Jan 31 19:39 chapters
drwxrwxr-x. 2 student student 62 Jan 31 19:38 editor

/home/student/Documents/my_bestseller/chapters:
total 0
-rw-rw-r--. 1 student student 0 Jan 31 19:18 mystery_chapter3.odf
-rw-rw-r--. 1 student student 0 Jan 31 19:18 mystery_chapter4.odf
-rw-rw-r--. 1 student student 0 Jan 31 19:18 mystery_chapter5.odf
-rw-rw-r--. 1 student student 0 Jan 31 19:18 mystery_chapter6.odf

/home/student/Documents/my_bestseller/editor:
total 0
-rw-rw-r--. 1 student student 0 Jan 31 19:18 mystery_chapter1.odf
-rw-rw-r--. 1 student student 0 Jan 31 19:18 mystery_chapter2.odf

/home/student/Documents/project_plans:
total 4
-rw-rw-r--. 1 student student 0 Jan 31 18:20 season1_project_plan.odf
-rw-rw-r--. 2 student student 0 Jan 31 19:05 season2_project_plan.odf
```

**19.** Beenden Sie **serverb**.

```
[student@serverb ~]$ exit  
logout  
Connection to serverb closed.  
[student@workstation ~]$
```

## Bewertung

Führen Sie auf **workstation** das Skript **lab files-review grade** aus, um den Erfolg dieser praktischen Übung zu bestätigen.

```
[student@workstation ~]$ lab files-review grade
```

## Beenden

Führen Sie auf **workstation** das Skript **lab files-review finish** aus, um diese praktische Übung abzuschließen. Dieses Skript entfernt alle Dateien und Verzeichnisse, die während der Übung auf **serverb** erstellt wurden.

```
[student@workstation ~]$ lab files-review finish
```

Hiermit wird die praktische Übung abgeschlossen.

# Zusammenfassung

---

In diesem Kapitel erhielten Sie folgende Informationen:

- Dateien in einem Linux-System werden in einer einzelnen invertierten Verzeichnisbaumstruktur organisiert, die als Dateisystemhierarchie bezeichnet wird.
- Absolute Pfade beginnen mit einem / und geben den Speicherort einer Datei in der einer Dateisystemhierarchie an.
- Relative Pfade beginnen nicht mit einem / und geben den Speicherort einer Datei relativ zum aktuellen Arbeitsverzeichnis an.
- Fünf Tasturbefehle werden zum Verwalten von Dateien verwendet: **mkdir**, **rmdir**, **cp**, **mv** und **rm**.
- Hardlinks und Softlinks sind verschiedene Methoden, mit denen mehrere Dateinamen auf dieselben Daten verweisen.
- Die Bash-Shell bietet Mustervergleichs-, Erweiterungs- und Ersetzungsfunktionen, mit denen Sie Befehle effizient ausführen können.



## Kapitel 4

# Abrufen von Hilfe in Red Hat Enterprise Linux

### Ziel

Beheben von Problemen durch die Verwendung von lokalen Hilfesystemen

### Ziele

- Finden von Informationen auf den Handbuchseiten des lokalen Linux-Systems
- Finden von Informationen in der lokalen Dokumentation in GNU Info

### Abschnitte

- Lesen von Handbuchseiten (und angeleitete Übung)
- Lesen der Info-Dokumentation (und angeleitete Übung)

### Praktische Übung

Abrufen von Hilfe in Red Hat Enterprise Linux

# Lesen von Handbuchseiten

## Ziele

Nach Abschluss dieses Abschnitts sollten Sie in der Lage sein, Informationen in den Manpages des lokalen Linux-Systems zu finden.

## Einführung in den Befehl „man“

Eine Dokumentationsquelle, die im Allgemeinen auf dem lokalen System verfügbar ist, sind Systemhandbuchseiten oder *Manpages*. Diese Seiten werden als Teil der Softwarepakete ausgeliefert, für die sie Dokumentationen enthalten, und können über die Befehlszeile mit dem Befehl **man** aufgerufen werden.

Das historische Handbuch für Linux-Programmierer, aus dem die Manpages stammen, war so umfassend, dass es mehrere gedruckte Bände füllte. Jeder Abschnitt enthält Informationen zu einem bestimmten Thema.

### Allgemeine Abschnitte des Linux-Handbuchs

Abschnitt	Inhalt
1	Benutzerbefehle ( <i>ausführbare und Shell-Programme</i> )
2	Systemaufrufe ( <i>vom Benutzerbereich aufgerufene Kernel-Routinen</i> )
3	Bibliotheksfunktionen ( <i>von Programmbibliotheken bereitgestellt</i> )
4	Spezielle Dateien ( <i>wie z. B. Gerätedateien</i> )
5	Dateiformate ( <i>für viele Konfigurationsdateien und -strukturen</i> )
6	Spiele ( <i>historischer Abschnitt für Spaßprogramme</i> )
7	Konventionen, Standards und Verschiedenes ( <i>Protokolle, Dateisysteme</i> )
8	Systemadministration und privilegierte Befehle ( <i>Wartungsaufgaben</i> )
9	Linux-Kernel-API ( <i>interne Kernel-Aufrufe</i> )

Zur Unterscheidung identischer Themenamen in verschiedenen Abschnitten enthalten Manpage-Verweise die Abschnittsnummer in Klammern nach dem Thema. So wird beispielsweise mit **passwd(1)** der Befehl zum Ändern von Passwörtern beschrieben, während mit **passwd(5)** das Dateiformat **/etc/passwd** zum Speichern lokaler Benutzerkonten erläutert wird.

Verwenden Sie zum Lesen bestimmter Manpages **man topic**. Inhalte werden bildschirmweise angezeigt. Der Befehl **man** durchsucht Handbuchabschnitte in alphanumerischer Reihenfolge. So wird beispielsweise mit **man passwd** standardmäßig **passwd(1)** angezeigt. Geben Sie zur Anzeige des Manpage-Themas aus einem bestimmten Abschnitt das Abschnittsnummerargument an: mit **man 5 passwd** wird **passwd(5)** angezeigt.

## Navigieren und Suchen nach Manpages

Die Fähigkeit, effizient nach Themen zu suchen und durch Manpages zu navigieren, ist eine wichtige Kompetenz in der Administration. GUI-Tools erleichtern die Konfiguration allgemeiner Systemressourcen, aber die Verwendung der Befehlszeilenschnittstelle ist noch effizienter. Um in der Befehlszeile effektiv zu navigieren, müssen Sie in der Lage sein, die benötigten Informationen auf den Manpages zu finden.

In der nachstehenden Tabelle sind die grundlegenden Navigationsbefehle bei der Anzeige von Manpages aufgeführt:

### Navigation auf Manpages

Befehl	Ergebnis
<b>Leertaste</b>	Zum nächsten Bildschirm (nach unten) scrollen
<b>Bild ab</b>	Zum nächsten Bildschirm (nach unten) scrollen
<b>Bild auf</b>	Zum vorherigen Bildschirm (nach oben) scrollen
<b>Pfeil nach unten</b>	Zur nächsten Zeile (nach unten) scrollen
<b>Pfeil nach oben</b>	Eine Zeile zurückscrollen (nach oben)
<b>D</b>	Zum nächsten halben Bildschirm (nach unten) scrollen
<b>U</b>	Zum vorherigen halben Bildschirm (nach oben) scrollen
<b>/Zeichenfolge</b>	Vorwärtssuche (nach unten) nach <i>Zeichenfolge</i> auf der Manpage
<b>N</b>	Vorherige Vorwärtssuche (nach unten) auf der Manpage wiederholen
<b>Umschalt+N</b>	Vorherige Rückwärtssuche (nach oben) auf der Manpage wiederholen
<b>G</b>	Zum Anfang der Manpage
<b>Umschalt+G</b>	Zum Ende der Manpage
<b>Q</b>	<b>man</b> beenden und zur Befehlsshell-Eingabeaufforderung zurückkehren



#### Wichtig

Bei der Suche ermöglicht *Zeichenfolge* eine Syntax mit regulären Ausdrücken. Einfache Texteingaben (z. B. **passwd**) funktionieren zwar wie erwartet, reguläre Ausdrücke verwenden für genauere Suchangaben Metazeichen (z. B. **\$**, **\***, **.** und **^**). Daher kann die Suche mit Zeichenfolgen, die Metazeichen für Programmausdrücke enthalten, wie zum Beispiel **make \$\$\$**, zu unerwarteten Ergebnissen führen.

Reguläre Ausdrücke und Syntax werden in *Red Hat System Administration II* und unter dem man-Thema **regex(7)** erläutert.

## Lesen von Manpages

Jedes Thema ist in mehrere Teile gegliedert. Die meisten Themen haben die gleichen Überschriften und werden in derselben Reihenfolge dargestellt. Normalerweise enthält ein Thema nicht alle Überschriften, da nicht alle Überschriften für alle Themen gelten.

Übliche Überschriften sind:

### Überschriften

Überschrift	Beschreibung
NAME	Name des Themas. Normalerweise ein Befehl oder ein Dateiname. Sehr kurze Beschreibung.
SYNOPSIS	Übersicht über die Syntax des Befehls
DESCRIPTION	Ausführliche Beschreibung, um ein grundlegendes Verständnis des Themas zu vermitteln
OPTIONS	Erläuterung der Optionen für die Befehlsausführung
EXAMPLES	Beispiele für die Verwendung des Befehls, der Funktion oder der Datei.
FILES	Eine Liste von Dateien und Verzeichnissen, die sich auf die Manpage beziehen.
SEE ALSO	Verwandte Informationen, normalerweise Themen auf anderen Manpages
BUGS	Bekannte Fehler in der Software
AUTHOR	Informationen darüber, wer zur Entwicklung des Themas beigetragen hat

## Suchen von Manpages nach Schlüsselwort

Eine Schlüsselwortsuche nach Manpages wird mit **man -k keyword** durchgeführt, womit eine Liste mit Manpages-Themen mit Abschnittsnummern ausgegeben wird.

```
[student@desktopX ~]$ man -k passwd
checkPasswdAccess (3) - query the SELinux policy database in the kernel.
chpasswd (8)          - update passwords in batch mode
ckpasswd (8)          - nnrpd password authenticator
fgetpwent_r (3)       - get passwd file entry reentrantly
getpwent_r (3)        - get passwd file entry reentrantly
...
passwd (1)            - update user's authentication tokens
sslpasswd (1ssl)      - compute password hashes
passwd (5)            - password file
passwd.nntp (5)       - Passwords for connecting to remote NNTP servers
passwd2des (3)        - RFS password encryption
...
```

Häufig aufgerufene Themen zur Systemadministration finden Sie in den Abschnitten 1 (Benutzerbefehle), 5 (Dateiformate) und 8 (administrative Befehle). Administratoren, die mit bestimmten Fehlersuchtools arbeiten, nutzen auch Abschnitt 2 (Systemaufrufe). Die übrigen

Abschnitte dienen vorwiegend als Referenz für Programmierer oder für die fortgeschrittenen Administration.



### Anmerkung

Schlüsselwortsuchen erfolgen auf Grundlage eines mit dem Befehl **mandb(8)** generierten Index, der als **root** ausgeführt werden muss. Der Befehl wird täglich über **cron.daily** oder durch **anacrontab** innerhalb einer Stunde nach dem Startvorgang ausgeführt, wenn er überfällig ist.



### Wichtig

Mit der Option **-K** (Großbuchstabe) für den Befehl **man** wird eine Volltextsuche durchgeführt und nicht wie bei der Option **-k** nur nach Titeln und Beschreibungen gesucht. Eine Volltextsuche beansprucht mehr Systemressourcen und Zeit.



### Literaturhinweise

Manpages **man(1)**, **mandb(8)**, **man-pages(7)**, **less(1)**, **intro(1)**, **intro(2)**, **intro(5)**, **intro(7)**, **intro(8)**

## ► Angeleitete Übung

# Lesen von Handbuchseiten

In dieser Übung suchen Sie mit den Optionen und Argumenten von **man** nach relevanten Informationen.

## Ergebnisse

Sie sollten in der Lage sein, das **man**-Linux-Handbuchsystem zu verwenden und nützliche Informationen durch Suchen und Browsen zu finden.

## Bevor Sie Beginnen

Melden Sie sich als Benutzer **student** mit dem Passwort **student** bei **workstation** an.

Führen Sie auf **workstation** den Befehl **lab help-manual start** aus. Dieser Befehl erstellt auch eine Datei namens **manual**.

```
[student@workstation ~]$ lab help-manual start
```

- 1. Zeigen Sie auf **workstation** die Manpage **gedit** an. Zeigen Sie mit **gedit** die Optionen zum Bearbeiten einer bestimmten Datei in der Befehlszeile an.

Verwenden Sie eine der Optionen von der Manpage **gedit**, um die Datei **/home/student/manual** mit **gedit** und dem Cursor am Ende der Datei zu öffnen.

- 1.1. Zeigen Sie die Manpage **gedit** an.

```
[student@workstation ~]$ man gedit
```

```
GEDIT(1) General Commands Manual GEDIT(1)
NAME
    gedit - text editor for the GNOME Desktop

SYNOPSIS
    gedit [OPTION...] [FILE...] [+LINE[:COLUMN]]
    gedit [OPTION...] -
...output omitted...
```

- 1.2. Sehen Sie sich mit **gedit** die Optionen zum Bearbeiten einer bestimmten Datei in der Befehlszeile an.

```
...output omitted...
FILE Specifies the file to open when gedit starts.
...output omitted...
+LINE For the first file, go to the line specified by LINE (do not insert
a space between the "+" sign and the number). If LINE is missing, go to the last
line.
...output omitted...
```

Drücken Sie **q**, um die Manpage zu verlassen.

- 1.3. Öffnen Sie mit dem Befehl **gedit + manual**. Die fehlende Zeilennummer neben der Option + öffnet eine Datei, die als Argument mit dem Cursor am Ende der letzten Zeile übergeben wird.

```
[student@workstation ~]$ gedit + manual
```

```
the quick brown fox just came over to greet the lazy poodle!
```

Vergewissern Sie sich, dass die Datei mit dem Cursor am Ende der letzten Zeile geöffnet ist. Drücken Sie **Strg+q**, um die Anwendung zu schließen.

► 2. Lesen Sie die Manpage **su(1)**.

Beachten Sie, dass der Befehl **su** vom Benutzer **root** ausgeht, wenn der Benutzer nicht angegeben ist. Wenn auf den Befehl **su** ein einzelner Bindestrich (-) folgt, wird eine untergeordnete Login-Shell gestartet. Ohne den Bindestrich wird eine untergeordnete Shell ohne Anmeldefunktion erstellt, die der aktuellen Umgebung des Benutzers entspricht.

```
[student@workstation ~]$ man 1 su
```

```
SU(1) User Commands SU(1)
NAME
    su - run a command with substitute user and group ID

SYNOPSIS
    su [options] [-] [user [argument...]]

DESCRIPTION
    su allows to run commands with a substitute user and group ID.
    When called without arguments, su defaults to running an interactive
    shell as root.
...output omitted...
OPTIONS
...output omitted...
-, -l, --login
    Start the shell as a login shell with an environment similar to a real login
...output omitted...
```



### Anmerkung

Beachten Sie, dass durch Komma getrennte Optionen in einer einzelnen Zeile, wie z. B. **-**, **-l** und **--login**, alle zu demselben Verhalten führen.

Drücken Sie **q**, um die Manpage zu verlassen.

- 3. Der Befehl **man** hat auch eigene Handbuchseiten.

```
[student@workstation ~]$ man man
MAN(1)           Manual pager utils                         MAN(1)

NAME
    man - an interface to the on-line reference manuals
...output omitted...

DESCRIPTION
    man is the system's manual pager. Each page argument given to man is
    normally the name of a program, utility or function. The manual page
    associated with each of these arguments is then found and displayed.
    A section, if provided, will direct man to look only in that section
    of the manual.
...output omitted...
```

Drücken Sie **q**, um die Manpage zu verlassen.

- 4. Alle Manpages befinden sich in **/usr/share/man**. Suchen Sie mit dem Befehl **whereis** die Binär-, Quell- und Handbuchseiten im Verzeichnis **/usr/share/man**.

```
[student@workstation ~]$ whereis passwd
passwd: /usr/bin/passwd /etc/passwd /usr/share/man/man1/passwd.1.gz /usr/share/
man/man5/passwd.5.gz
```

- 5. Listen Sie mit dem Befehl **man -k zip** detaillierte Informationen über ein ZIP-Archiv auf.

```
[student@workstation ~]$ man -k zip
...output omitted...
zipinfo (1)      - list detailed information about a ZIP archive
zipnote (1)      - write the comments in zipfile to stdout, edit comments and
    rename files in zipfile
zipsplit (1)     - split a zipfile into smaller zipfiles
```

- 6. Rufen Sie mit dem Befehl **man -k boot** die Manpage auf, die eine Liste der Parameter enthält, die beim Booten an den Kernel übergeben werden können.

```
[student@workstation ~]$ man -k boot
...output omitted...
bootctl (1)      - Control the firmware and boot manager settings
bootparam (7)     - introduction to boot time parameters of the Linux kernel
bootup (7)       - System bootup process
...output omitted...
```

- 7. Suchen Sie mit **man -k ext4** den Befehl zum Anpassen der Parameter des ext4-Dateisystems.

```
[student@workstation ~]$ man -k ext4
...output omitted...
resize2fs (8)           - ext2/ext3/ext4 file system resizer
tune2fs (8)            - adjust tunable filesystem parameters on ext2/ext3/ext4
filesystems
```

## Beenden

Führen Sie auf **workstation** das Skript **lab help-manual finish** aus, um diese Übung zu beenden.

```
[student@workstation ~]$ lab help-manual finish
```

Hiermit ist die angeleitete Übung beendet.

# Lesen der Info-Dokumentation

## Ziele

Nach Abschluss dieses Abschnitts sollten die Kursteilnehmer in der Lage sein, Informationen in der lokalen GNU Info-Dokumentation zu finden.

## Einführung in GNU Info

Manpages weisen ein Format auf, das sich als Referenz für Befehle eignet, als allgemeine Dokumentation jedoch weniger nützlich ist. Für solche Dokumente wurde mit dem GNU-Projekt ein anderes Online-Dokumentationssystem mit der Bezeichnung *GNU Info* entwickelt. Info-Dokumente sind wichtige Ressourcen für ein Red Hat Enterprise Linux-System, da viele grundlegende Komponenten und Dienstprogramme, wie das *coreutils*-Paket und *glibc*-Standardbibliotheken, entweder mit dem GNU-Projekt entwickelt werden oder das Info-Dokumentensystem verwenden.



### Wichtig

Sie fragen sich vielleicht, warum es zwei lokale Dokumentationssysteme gibt, Manpages und Info-Dokumente. Einige der Gründe dafür sind praktischer Natur und andere haben damit zu tun, wie Linux und seine Anwendungen im Laufe der Jahre von verschiedenen Open Source-Communities entwickelt wurden.

Manpages haben ein wesentlich formelleres Format und dokumentieren normalerweise einen bestimmten Befehl oder eine bestimmte Funktion aus einem Softwarepaket. Sie sind als einzelne Textdateien strukturiert. Info-Dokumente decken in der Regel bestimmte Softwarepakete als Ganzes ab, enthalten eher praktischere Beispiele für die Verwendung der Software und sind als Hypertextdokumente strukturiert.

Sie sollten mit beiden Systemen vertraut sein, um die Ihnen vom System zur Verfügung gestellten Informationen optimal nutzen zu können.

## Lesen der Info-Dokumentation

Mit dem Befehl **pinfo** starten Sie den Info-Dokument-Viewer. **pinfo** wird mit dem obersten Verzeichnis geöffnet.

The screenshot shows a terminal window titled 'File: dir Node: Top This is the top of the INFO tree'. The text displayed is:

```

This (the Directory node) gives a menu of major topics.
Typing "q" exits, "?" lists all Info commands, "d" returns here,
"h" gives a primer for first-timers,
"mEmacs<Return>" visits the Emacs topic, etc.

In Emacs, you can click mouse button 2 on a menu item or cross reference
to select it.

* Menu:

Archiving
* Cpio: (cpio). Copy-in-copy-out archiver to tape or disk.
* Tar: (tar). Making tape (or disk) archives.

Basics
* Common options: (coreutils)Common options.
    Common options.
* Coreutils: (coreutils). Core GNU (file, text, shell) utilities.
* Date input formats: (coreutils)Date input formats.
* File permissions: (coreutils)File permissions.
    Access modes.
* Finding files: (find). Operating on files matching certain criteria.
* ed: (ed). The GNU Line Editor.

Viewing line 25/2002, 1%

```

Abbildung 4.1: Info-Dokument-Viewer „pinfo“, oberstes Verzeichnis

Die Info-Dokumentation ist ausführlich und mit Hyperlinks versehen. Sie können Info-Seiten in mehreren Formaten ausgeben. Im Gegensatz dazu sind Manpages für die Druckausgabe optimiert. Das Info-Format ist flexibler als Manpages und bietet eine genauere Erläuterung komplexer Befehle und Konzepte. Genau wie Manpages werden Info-Knoten von der Befehlszeile aus mit dem Befehl **pinfo** gelesen.

Eine typische Manpage enthält eine kurze Erläuterung zu einem bestimmten Thema, einem Befehl, einem Tool oder einer Datei. Die Info-Dokumentation ist ein ausführliches Dokument. Info bietet die folgenden Verbesserungen:

- Ein einziges Dokument für ein großes System, das alle für dieses System erforderlichen Informationen enthält
- Hyperlinks
- Vollständiger durchsuchbarer Dokumentindex
- Volltextsuche im gesamten Dokument

Für einige Befehle und Dienstprogramme gibt es sowohl **manpages** als auch Info-Dokumentation. In der Regel enthält die Info-Dokumentation detailliertere Informationen. Vergleichen Sie die Unterschiede in der **man**- und **pinfo**-Dokumentation für **tar**:

```
[user@host ~]$ man tar
[user@host ~]$ pinfo tar
```

Der **pinfo**-Reader liefert detailliertere Informationen als der ursprüngliche **info**-Befehl. Um ein bestimmtes Thema zu durchsuchen, verwenden Sie den Befehl **pinfo topic**. Der Befehl **pinfo** ohne Argument öffnet das oberste Verzeichnis. Neue Dokumentation wird bei der Installation der zugehörigen Softwarepakete in **pinfo** verfügbar.



### Anmerkung

Wenn für einen bestimmten von Ihnen gesuchten Eintrag kein Info-Thema im System vorhanden ist, sucht Info nach einer passenden Manpage und zeigt diese stattdessen an.

## Vergleich der Navigation in GNU Info und Manpage

Der Befehl **pinfo** und der Befehl **man** verwenden geringfügig unterschiedliche Tasten für die Navigation. In der folgenden Tabelle werden die Navigationstasten für beide Befehle verglichen:

### pinfo und man, Tastenzuordnungsvergleich

Navigation	pinfo	man
Zum nächsten Bildschirm (nach unten) scrollen	<b>Bild ab</b> oder <b>Leer</b>	<b>Bild ab</b> oder <b>Leer</b>
Zum vorherigen Bildschirm (nach oben) scrollen	<b>Bild auf</b> oder <b>b</b>	<b>Bild auf</b> oder <b>b</b>
Das Themenverzeichnis anzeigen	<b>d</b>	-
Zum nächsten halben Bildschirm (nach unten) scrollen	-	<b>d</b>
Den übergeordneten Knoten eines Themas anzeigen	<b>u</b>	-
Den Anfang eines Themas (nach oben) anzeigen	<b>POS1</b>	<b>g</b>
Zum vorherigen halben Bildschirm (nach oben) scrollen	-	<b>u</b>
Zum nächsten Hyperlink (nach unten) weiterblättern	<b>Pfeil nach unten</b>	-
Thema an der aktuellen Cursorposition öffnen	<b>Eingabe</b>	-
Zur nächsten Zeile (nach unten) scrollen	-	<b>Pfeil nach unten</b> oder <b>Eingabe</b>
Zum vorherigen Hyperlink (nach oben) zurückscrollen	<b>Pfeil nach oben</b>	-
Eine Zeile zurückscrollen (nach oben)	-	<b>Pfeil nach oben</b>
Muster suchen	<b>/Zeichenfolge</b>	<b>/Zeichenfolge</b>
Nächsten Knoten (Kapitel) im Thema anzeigen	<b>n</b>	-
Vorherige Vorwärtssuche (nach unten) wiederholen	<b>/ dann Eingabe</b>	<b>n</b>
Vorherigen Knoten (Kapitel) im Thema anzeigen	<b>p</b>	-
Vorherige Rückwärtssuche (nach oben) wiederholen	-	<b>N</b>

Navigation	<b>pinfo</b>	<b>man</b>
Programm beenden	<b>q</b>	<b>q</b>



### Literaturhinweise

**pinfo info** (*Info: Eine Einführung*)

**pinfo pinfo** (*Dokumentation für pinfo*)

Über das Projekt GNU

<http://www.gnu.org/gnu/thegnuproject.html>

Manpages **pinfo(1)** und **info(1)**

## ► Angeleitete Übung

# Lesen der Info-Dokumentation

In dieser Übung suchen Sie Informationen, die in GNU Info-Dokumenten gespeichert sind, indem Sie in diesen Dokumenten mit Befehlszeilentools navigieren.

## Ergebnisse

Sie sollten in der Lage sein, in der GNU Info-Dokumentation mit Befehlszeilentools zu navigieren.

## Bevor Sie Beginnen

Melden Sie sich als Benutzer **student** mit dem Passwort **student** bei **workstation** an.

Führen Sie auf **workstation** den Befehl **lab help-info start** aus.

```
[student@workstation ~]$ lab help-info start
```

- 1. Starten Sie auf **workstation pinfo** ohne Argumente.

```
[student@workstation ~]$ pinfo
```

- 2. Navigieren Sie zum Thema **Common options**.

Drücken Sie **Pfeil nach oben** oder **Pfeil nach unten**, bis **(coreutils) Common options** hervorgehoben wird.

### Basics

- \* Bash: ([bash](#)). The GNU Bourne-Again SHeLL.
- \* Common options: ([coreutils](#))[Common options](#).

Abbildung 4.2: Bash-Dokumentation

- 3. Drücken Sie die **Eingabetaste**, um dieses Thema anzuzeigen.

```
File: coreutils.info, Node: Common options, Next: Output of entire files, Prev: Introduction, Up: Top

2 Common options
*****



Certain options are available in all of these programs. Rather than
writing identical descriptions for each of the programs, they are
described here. (In fact, every GNU program accepts (or should accept)
these options.)

Normally options and operands can appear in any order, and programs
act as if all the options appear before any operands. For example,
'sort -r passwd -t :' acts like 'sort -r -t : passwd', since ':' is an
option-argument of '-t'. However, if the 'POSIXLY_CORRECT' environment
variable is set, options must appear before operands, unless otherwise
specified for a particular command.
```

Abbildung 4.3: Info-Thema „Common options“

- 4. Gehen Sie dieses Info-Thema durch. Sie lernen hier, ob lange Optionen abgekürzt werden können.  
Navigieren Sie mithilfe von **Bild ab** und **Bild auf** durch das Thema. In vielen Programmen können lange Optionen abgekürzt werden.
- 5. Stellen Sie fest, welche Bedeutung die Symbole `--` haben, wenn sie als Befehlsargument verwendet werden.  
Die Symbole `--` kennzeichnen in komplexen Befehlen, bei denen diese Unterscheidung vom Befehlszeilenparser der Shell möglicherweise nicht korrekt getroffen wird, das Ende der *Befehlsoptionen* und den Beginn der *Befehlsargumente*.
- 6. Scrollen Sie nach oben zum Knoten **GNU Coreutils**, ohne **pinfo** zu beenden.  
Drücken Sie **u**, um zum obersten Knoten des Themas zu gelangen.
- 7. Kehren Sie zum Thema der obersten Ebene zurück.  
Drücken Sie noch einmal **u**. Wenn Sie sich an der obersten Stelle eines Themenknotens befinden und nach oben scrollen, gelangen Sie wieder zum Themenverzeichnis. Wenn Sie an einer beliebigen Stelle oder in einem beliebigen Thema **d** drücken, gelangen Sie direkt zum Themenverzeichnis.
- 8. Suchen Sie nach dem Muster **coreutils** und wählen Sie dieses Thema aus.  
Drücken Sie **/** und geben Sie dann die Suchzeichenfolge „coreutils“ ein. Wenn das Thema hervorgehoben ist, drücken Sie die **Eingabetaste**.

```
* Coreutils: (coreutils).      Core GNU (file, text, shell) utilities.
```

Abbildung 4.4: Suchergebnis

- 9. Suchen Sie im Menü oben **Output of entire files** und wählen Sie es aus, indem Sie **n** drücken. Gehen Sie das Thema durch.  
Wählen Sie mit der **Eingabetaste cat invocation** aus. Gehen Sie das Thema mithilfe der Pfeiltasten durch.
- 10. Blättern Sie zwei Ebenen nach oben, um wieder zu **GNU Coreutils** zu gelangen. Blättern Sie zu **Summarizing files**.  
Drücken Sie die **Eingabetaste**, um das Thema auszuwählen und gehen Sie dann das Thema durch.
- 11. Drücken Sie **q**, um **pinfo** zu beenden.
- 12. Verwenden Sie den Befehl **pinfo** erneut und geben Sie in der Befehlszeile **coreutils** als Zielthema ein.

```
[student@workstation ~]$ pinfo coreutils
```

- 13. Wählen Sie das Thema **Disk usage** aus.  
Drücken Sie **Pfeil nach unten**, um **Disk usage** hervorzuheben, und drücken Sie dann die **Eingabetaste**, um dieses Thema auszuwählen.

► **14.** Lesen Sie die Unterthemen **df invocation** und **du invocation**.

Verwenden Sie die Pfeiltasten zum Hervorheben eines Themas, **Bild auf** und **Bild ab**, um den Text durchzugehen, und drücken Sie dann **u**, um eine Ebene nach oben zu gelangen. Drücken Sie **q**, um zu beenden, wenn Sie fertig sind.

## Beenden

Führen Sie auf **workstation** das Skript **lab help-info finish** aus, um diese Übung zu beenden.

```
[student@workstation ~]$ lab help-info finish
```

Hiermit ist die angeleitete Übung beendet.

## ► Praktische Übung

# Abrufen von Hilfe in Red Hat Enterprise Linux

### Leistungscheckliste

In dieser praktischen Übung schlagen Sie in Manpages und GNU Info-Dokumenten Informationen nach, die Ihnen bei der Durchführung von Aufgaben helfen.

### Ergebnisse

Es werden folgende Fähigkeiten vermittelt:

- Auffinden relevanter Befehle, indem Sie Manpages und Info-Knoten durchsuchen
- Erlernen neuer Optionen für häufig verwendete Dokumentationsbefehle
- Verwenden geeigneter Tools zum Anzeigen und Drucken von Dokumentation sowie anderer nicht als Text formatierter Dateien

### Bevor Sie Beginnen

Melden Sie sich als Benutzer **student** mit dem Passwort **student** bei **workstation** an.

Führen Sie auf **workstation** den Befehl **lab help-review start** aus.

```
[student@workstation ~]$ lab help-review start
```

1. Ermitteln Sie auf **workstation**, wie eine Manpage für den Druck vorbereitet wird. Suchen Sie insbesondere nach dem Format oder der Rendering-Sprache, die zum Drucken verwendet wird.
2. Erstellen Sie eine formatierte Ausgabedatei der Manpage **passwd**. Rufen Sie die Datei **passwd.ps** auf. Ermitteln Sie das Format des Dateiinhalts. Überprüfen Sie den Inhalt der Datei **passwd.ps**.



#### Anmerkung

Erstellen Sie mit folgendem Befehl eine formatierte Ausgabe der Manpage **passwd**:

```
[student@workstation ~]$ man -t passwd > passwd.ps
```

Das Symbol **>** leitet den Inhalt der Manpage zur Datei **passwd.ps** um. Dieser Befehl wird in einem folgenden Kapitel genauer beschrieben.

3. Sehen Sie sich anhand von **man** die Befehle zum Anzeigen und Drucken von PostScript-Dateien an.
4. Informieren Sie sich anhand von **evince(1)** über die Verwendung des Viewers im Vorschaumodus. Ermitteln Sie außerdem, wie Sie ein Dokument auf einer bestimmten Seite öffnen.

## Kapitel 4 | Abrufen von Hilfe in Red Hat Enterprise Linux

5. Zeigen Sie Ihre PostScript-Datei mit den verschiedenen gefundenen **evince**-Optionen an. Schließen Sie die Dokumentdatei, wenn Sie fertig sind.
6. Suchen Sie mit dem Befehl **man** auf **lp(1)** nach Informationen zum Drucken eines beliebigen Dokuments ab einer bestimmten Seite. Informieren Sie sich darüber, wie ohne tatsächliche Eingabe eines Befehls (da keine Drucker vorhanden sind) die Syntax für einen Befehl zum Drucken nur der Seiten 2 und 3 Ihrer PostScript-Datei aussehen würde.
7. Suchen Sie mit **pinfo** nach der GNU Info-Dokumentation zum **evince**-Viewer.
8. Öffnen Sie mit Firefox das Verzeichnis mit der Paketdokumentation des Systems und wechseln Sie in das Paketunterverzeichnis **man-db**. Zeigen Sie die bereitgestellten Handbücher an.
9. Suchen Sie mit dem Firefox-Browser das Paketunterverzeichnis **initscripts** und wechseln Sie dorthin. Zeigen Sie die Datei **sysconfig.txt** an, in der wichtige Systemkonfigurationsoptionen beschrieben werden, die im Verzeichnis **/etc/sysconfig** gespeichert sind.

## Bewertung

Führen Sie auf **workstation** **lab help-review grade** aus, um den Erfolg dieser Übung zu bestätigen.

```
[student@workstation ~]$ lab help-review grade
```

## Beenden

Führen Sie auf **workstation** das Skript **lab help-review finish** aus, um diese Übung zu beenden.

```
[student@workstation ~]$ lab help-review finish
```

Damit ist die praktische Übung abgeschlossen.

## ► Lösung

# Abrufen von Hilfe in Red Hat Enterprise Linux

### Leistungscheckliste

In dieser praktischen Übung schlagen Sie in Manpages und GNU Info-Dokumenten Informationen nach, die Ihnen bei der Durchführung von Aufgaben helfen.

### Ergebnisse

Es werden folgende Fähigkeiten vermittelt:

- Auffinden relevanter Befehle, indem Sie Manpages und Info-Knoten durchsuchen
- Erlernen neuer Optionen für häufig verwendete Dokumentationsbefehle
- Verwenden geeigneter Tools zum Anzeigen und Drucken von Dokumentation sowie anderer nicht als Text formatierter Dateien

### Bevor Sie Beginnen

Melden Sie sich als Benutzer **student** mit dem Passwort **student** bei **workstation** an.

Führen Sie auf **workstation** den Befehl **lab help-review start** aus.

```
[student@workstation ~]$ lab help-review start
```

1. Ermitteln Sie auf **workstation**, wie eine Manpage für den Druck vorbereitet wird. Suchen Sie insbesondere nach dem Format oder der Rendering-Sprache, die zum Drucken verwendet wird.
  - 1.1. Verwenden Sie den Befehl **man man**, um zu ermitteln, wie eine Manpage für den Druck vorbereitet wird.

```
[student@worksation ~]$ man man  
...output omitted...
```

Drücken Sie **q**, um die Manpage zu verlassen.



#### Anmerkung

Für **man** wird **-t** zur Vorbereitung einer Manpage für den Druck im PostScript-Format verwendet.

2. Erstellen Sie eine formatierte Ausgabedatei der Manpage **passwd**. Rufen Sie die Datei **passwd.ps** auf. Ermitteln Sie das Format des Dateiinhalts. Überprüfen Sie den Inhalt der Datei **passwd.ps**.



### Anmerkung

Erstellen Sie mit folgendem Befehl eine formatierte Ausgabe der Manpage **passwd**:

```
[student@workstation $]$ man -t passwd > passwd.ps
```

Das Symbol **>** leitet den Inhalt der Manpage zur Datei **passwd.ps** um. Dieser Befehl wird in einem folgenden Kapitel genauer beschrieben.

- 2.1. Erstellen Sie mit dem Befehl **man -t** eine formatierte Datei der Manpage **passwd**.

```
[student@workstation ~]$ man -t passwd > passwd.ps  
[student@workstation ~]$ ls -al  
...output omitted...  
-rw-rw-r--. 1 student student 19947 Feb 26 11:14 passwd.ps  
...output omitted...
```

- 2.2. Ermitteln Sie mit dem Befehl **file** das Format des Dateiinhalts.

```
[student@workstation ~]$ file /home/student/passwd.ps  
passwd.ps: PostScript document text conforming DSC level 3.0
```

- 2.3. Zeigen Sie mit dem Befehl **less** die Datei **/home/student/passwd.ps** an.

```
[student@workstation ~]$ less /home/student/passwd.ps  
%!PS-Adobe-3.0  
%%Creator: groff version 1.22.3  
%%CreationDate: Tue Feb 26 11:14:40 2019  
%%DocumentNeededResources: font Times-Roman  
%%+ font Times-Bold  
%%+ font Times-Italic  
%%+ font Symbol  
%%DocumentSuppliedResources: procset grops 1.22 3  
...output omitted...
```



### Anmerkung

Die Ausgabe von **file** gibt an, dass die Datei im PostScript-Format vorliegt, und Sie haben dies durch Anzeigen des Inhalts bestätigt. Beachten Sie die Header-Zeilen mit den PostScript-Informationen. Beenden Sie den Befehl **less** mit **q**.

3. Sehen Sie sich anhand von **man** die Befehle zum Anzeigen und Drucken von PostScript-Dateien an.

- 3.1. Sehen Sie sich anhand von **man** die Befehle zum Anzeigen und Drucken von PostScript-Dateien an.

```
[student@workstation ~]# man -k postscript viewer  
evince (1) - GNOME document viewer  
evince-previewer (1) - show a printing preview of PostScript and PDF documents  
evince-thumbnailer (1) - create png thumbnails from PostScript and PDF documents
```

```

gcm-viewer (1)      - GNOME Color Manager Profile Viewer Tool
gnome-logs (1)     - log viewer for the systemd journal
grops (1)          - PostScript driver for groff
pango-view (1)      - Pango text viewer
pluginviewer (8)    - list loadable SASL plugins and their properties

```



### Anmerkung

Mithilfe mehrerer Wörter mit der Option **-k** suchen Sie Manpages, die eine Übereinstimmung mit *einem* dieser Wörter aufweisen, das heißt, diejenigen mit „postscript“ oder „viewer“ in der Beschreibung. Beachten Sie die **evince(1)**-Befehle in der Ausgabe.

4. Informieren Sie sich anhand von **evince(1)** über die Verwendung des Viewers im Vorschaumodus. Ermitteln Sie außerdem, wie Sie ein Dokument auf einer bestimmten Seite öffnen.
  - 4.1. Informieren Sie sich anhand des Befehls **man evince** über die Verwendung des Viewers im Vorschaumodus.

```
[student@workstation ~]$ man evince
...output omitted...
```

Drücken Sie **q**, um die Manpage zu verlassen.



### Anmerkung

Mit der Option **-w** (oder **--preview**) wird **evince** im Vorschaumodus geöffnet. Mit der Option **-i** wird eine bestimmte Startseite angegeben.

5. Zeigen Sie Ihre PostScript-Datei mit den verschiedenen gefundenen **evince**-Optionen an. Schließen Sie die Dokumentdatei, wenn Sie fertig sind.
  - 5.1. Öffnen Sie mit dem Befehl **evince** die Datei **/home/student/passwd.ps**.

```
[student@workstation ~]$ evince /home/student/passwd.ps
```

- 5.2. Öffnen Sie mit dem Befehl **evince -w /home/student/passwd.ps** die Datei im Vorschaumodus.

```
[student@workstation ~]$ evince -w /home/student/passwd.ps
```

- 5.3. Öffnen Sie mit dem Befehl **evince -i 3 /home/student/passwd.ps** die Datei auf Seite 3.

```
[student@workstation ~]$ evince -i 3 /home/student/passwd.ps
```



### Anmerkung

Während im normalen **evince**-Modus die Anzeige im Vollbildmodus und im Präsentationsstil möglich ist, eignet sich der **evince**-Vorschaumodus zum schnellen Durchsuchen und Drucken. Beachten Sie das **Drucksymbol** oben.

**Kapitel 4 |** Abrufen von Hilfe in Red Hat Enterprise Linux

6. Suchen Sie mit dem Befehl **man** auf **lp(1)** nach Informationen zum Drucken eines beliebigen Dokuments ab einer bestimmten Seite. Informieren Sie sich darüber, wie ohne tatsächliche Eingabe eines Befehls (da keine Drucker vorhanden sind) die Syntax für einen Befehl zum Drucken nur der Seiten 2 und 3 Ihrer PostScript-Datei aussehen würde.

- 6.1. Ermitteln Sie mit dem Befehl **man lp**, wie bestimmte Seiten eines Dokuments gedruckt werden.

```
[student@workstation ~]$ man lp  
...output omitted...
```

Drücken Sie **q**, um die Manpage zu verlassen.

**Anmerkung**

Aus **lp(1)** können Sie entnehmen, dass die Option **-P** bestimmte Seiten angibt. Mit dem Befehl **lp** wird an den *Standarddrucker* gespoolet. Dabei wird nur der Seitenbereich von 2 bis 3 gesendet. Eine gültige Antwort lautet daher **lp passwd.ps -P 2-3**.

7. Suchen Sie mit **pinfo** nach der GNU Info-Dokumentation zum **evince**-Viewer.

- 7.1. Suchen Sie mit **pinfo command** nach der GNU Info-Dokumentation zum **evince**-Viewer.

```
[student@workstation ~]$ pinfo evince
```

**Anmerkung**

Beachten Sie, dass stattdessen die Manpage **evince(1)** angezeigt wird. Mit dem **pinfo**-Dokumentviewer wird eine entsprechende Manpage gesucht, wenn kein passender GNU-Dokumentationsknoten für das gewünschte Thema vorhanden ist. Drücken Sie **q**, um das Dienstprogramm zu beenden.

8. Öffnen Sie mit Firefox das Verzeichnis mit der Paketdokumentation des Systems und wechseln Sie in das Paketunterverzeichnis **man-db**. Zeigen Sie die bereitgestellten Handbücher an.

- 8.1. Zeigen Sie mit **firefox /usr/share/doc** die Systemdokumentation an. Wechseln Sie zum Unterverzeichnis **man-db**. Klicken Sie auf die Handbücher, um sie anzuzeigen.

```
[student@workstation ~]$ firefox /usr/share/doc
```

**Anmerkung**

Für jedes häufig verwendete Verzeichnis können Lesezeichen erstellt werden. Klicken Sie nach dem Durchsuchen des Verzeichnisses **man-db**, um die Textversion des Handbuchs zu öffnen und anzuzeigen, und schließen Sie sie anschließend wieder. Klicken Sie zum Öffnen der PostScript-Version. Wie bereits zuvor festgestellt, ist **evince** der Standardviewer für PostScript- und PDF-Dokumente. Möglicherweise möchten Sie diese Dokumente später noch einmal aufrufen, um mehr über **man** zu erfahren. Schließen Sie den **evince**-Viewer, wenn Sie fertig sind.

Index of file:///usr/share/doc/man-db/			
<a href="#">Up to higher level directory</a>			
Name	Size	Last Modified	
File: ChangeLog	51 KB	12/12/16 1:44:30 PM GMT+1	
File: NEWS	60 KB	11/7/18 4:46:16 PM GMT+1	
File: README	12 KB	12/11/16 12:44:45 AM GMT+1	
man-db-manual.ps	129 KB	11/7/18 4:47:06 PM GMT+1	
man-db-manual.txt	70 KB	11/7/18 4:47:01 PM GMT+1	

9. Suchen Sie mit dem Firefox-Browser das Paketunterverzeichnis **initscripts** und wechseln Sie dorthin. Zeigen Sie die Datei **sysconfig.txt** an, in der wichtige Systemkonfigurationsoptionen beschrieben werden, die im Verzeichnis **/etc/sysconfig** gespeichert sind.
- 9.1. Suchen Sie im Firefox-Browser das Paketunterverzeichnis **initscripts**. Beachten Sie, wie hilfreich ein Browser für die Suche und Anzeige lokaler Systemdokumentation ist. Schließen Sie das Dokument und Firefox, wenn Sie fertig sind.

## Bewertung

Führen Sie auf **workstation** **lab help-review grade** aus, um den Erfolg dieser Übung zu bestätigen.

```
[student@workstation ~]$ lab help-review grade
```

## Beenden

Führen Sie auf **workstation** das Skript **lab help-review finish** aus, um diese Übung zu beenden.

```
[student@workstation ~]$ lab help-review finish
```

Damit ist die praktische Übung abgeschlossen.

# Zusammenfassung

---

In diesem Kapitel erhielten Sie folgende Informationen:

- Manpages werden mit dem Befehl **man** angezeigt und bieten Informationen zu Komponenten eines Linux-Systems, z. B. zu Dateien, Befehlen und Funktionen.
- Per Konvention folgt bei einem Verweis auf eine Manpage dem Namen einer Seite die Abschnittsnummer in Klammern.
- Info-Dokumente werden mit dem Befehl **pinfo** angezeigt und bestehen aus einer Sammlung von Hypertext-Knoten, die Informationen zu Softwarepaketen als Ganzes bereitstellen.
- Die Navigationstasten von **man** und **pinfo** unterscheiden sich geringfügig.

## Kapitel 5

# Erstellen, Anzeigen und Bearbeiten von Textdateien

### Ziel

Erstellen, Anzeigen und Bearbeiten von Textdateien über die Befehlsausgabe oder in einem Editor

### Ziele

- Speichern der Befehlsausgabe oder Fehler in einer Datei mit Shell-Umleitung und Verarbeiten der Befehlsausgabe über mehrere Befehlszeilenprogramme mit Pipes
- Erstellen und Bearbeiten von Textdateien mit dem **vim**-Editor
- Verwenden von Shell-Variablen zur Ausführung von Befehlen und Bearbeiten von Bash-Startskripts zur Festlegung von Shell- und Umgebungsvariablen, um das Verhalten der Shell und von in der Shell ausgeführten Programmen zu ändern

### Abschnitte

- Umleiten von Ausgaben an eine Datei oder ein Programm (und angeleitete Übung)
- Bearbeiten von Textdateien an der Shell-Eingabeaufforderung (und angeleitete Übung)
- Ändern der Shell-Umgebung (und angeleitete Übung)

### Praktische Übung

Erstellen, Anzeigen und Bearbeiten von Textdateien

# Umleiten von Ausgaben an eine Datei oder ein Programm

---

## Ziele

Nach Abschluss dieses Abschnitts sollten Sie in der Lage sein, die Ausgabe oder Fehler in einer Datei mit Shell-Umleitung zu speichern und die Befehlausgabe über mehrere Befehlszeilenprogramme mit Pipes zu verarbeiten.

## Standardeingabe, Standardausgabe und Standard-Fehlerausgabe

Ein ausgeführtes Programm oder ein Prozess muss irgendwo eine Eingabe lesen und irgendwo eine Ausgabe schreiben. Eine Befehlausführung an der Shell-Eingabeaufforderung liest die Eingabe in der Regel von der Tastatur und sendet die Ausgabe an sein Terminalfenster.

Ein Prozess verwendet nummerierte Kanäle, sogenannte *Dateideskriptoren*, um Eingaben abzurufen und Ausgaben zu senden. Alle Prozesse beginnen mit mindestens drei Dateideskriptoren. **Standardeingabe** (Kanal 0) liest Eingaben von der Tastatur. **Standardausgabe** (Kanal 1) sendet die normale Ausgabe an das Terminal. **Standardfehler** (Kanal 2) sendet Fehlermeldungen an das Terminal. Wenn ein Programm separate Verbindungen zu anderen Dateien öffnet, kann es Dateideskriptoren mit höherer Nummerierung verwenden.

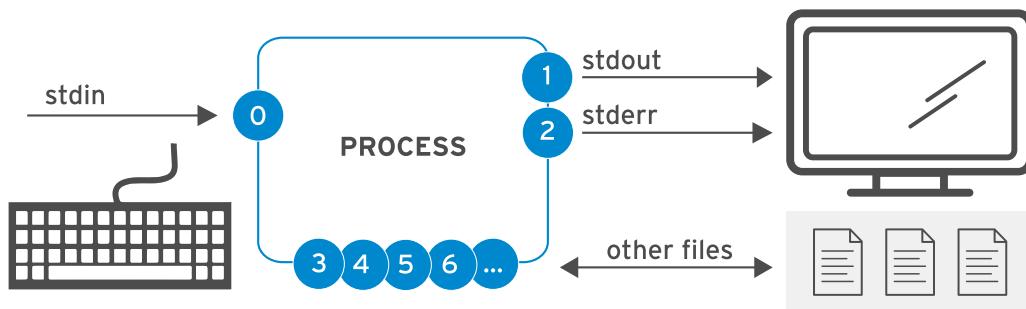


Abbildung 5.1: Prozess-I/O-Kanäle (Dateideskriptoren)

### Kanäle (Dateideskriptoren)

Nummer	Kanalname	Beschreibung	Standardverbindung	Verwendung
0	<b>stdin</b>	Standardeingabe	Tastatur	schreibgeschützt
1	<b>stdout</b>	Standardausgabe	Terminal	nur schreibberechtigt
2	<b>stderr</b>	Standard-Fehlerausgabe	Terminal	nur schreibberechtigt
3 +	<b>Dateiname</b>	Andere Dateien	keine	lese- und/oder schreibberechtigt

## Umleiten von Ausgaben in eine Datei

Die I/O-Umleitung ändert, wie der Prozess seine Eingabe oder Ausgabe erhält. Anstatt Eingaben über die Tastatur zu erhalten oder Ausgaben und Fehler an das Terminal zu senden, liest der Prozess aus Dateien oder schreibt in Dateien. Mit der Umleitung können Sie Meldungen in einer Datei speichern, die normalerweise an das Terminalfenster gesendet werden. Alternativ können Sie die Umleitung verwenden, um Ausgaben oder Fehler zu verwerfen, damit sie nicht auf dem Terminal angezeigt oder gespeichert werden.

Durch Umleiten von **stdout** wird die Anzeige der Prozessausgabe auf dem Terminal unterdrückt. Wie in der folgenden Tabelle gezeigt, wird durch das Umleiten von **stdout** alleine nicht die Anzeige von **stderr**-Fehlermeldungen am Terminal unterdrückt. Falls die Datei nicht existiert, wird sie erstellt. Falls die Datei nicht existiert und die Umleitung nicht an die Datei angehängt ist, wird der Inhalt der Datei überschrieben.

Wenn Sie Meldungen verwerfen möchten, dann verwirft die spezielle Datei **/dev/null** Kanalausgaben, die an sie umgeleitet wurden, im Hintergrund und ist daher immer eine leere Datei.

### Operatoren für die Ausgabeumleitung

Verwendung	Erläuterung	Visuelle Hilfe
<code>&gt; file</code>	Umleiten von <b>stdout</b> zum Überschreiben einer Datei	<pre> graph LR     K[Keyboard] -- "stdin" --&gt; P((PROCESS))     P -- "0" --&gt; F1["File"]     P -- "1" --&gt; M1["Monitor"]     P -- "2" --&gt; F2["File"]     F1 -- "stdout" --&gt; F2     M1 -- "stderr" --&gt; F2   </pre>
<code>&gt;&gt; file</code>	Umleiten von <b>stdout</b> zum Anhängen an eine Datei	<pre> graph LR     K[Keyboard] -- "stdin" --&gt; P((PROCESS))     P -- "0" --&gt; F1["File"]     P -- "1" --&gt; M1["Monitor"]     P -- "2" --&gt; F2["File"]     F1 -- "stdout" --&gt; F2     M1 -- "stderr" --&gt; F2   </pre>
<code>2&gt; file</code>	Umleiten von <b>stderr</b> zum Überschreiben einer Datei	<pre> graph LR     K[Keyboard] -- "stdin" --&gt; P((PROCESS))     P -- "0" --&gt; M1["Monitor"]     P -- "1" --&gt; F1["File"]     P -- "2" --&gt; F2["File"]     M1 -- "stdout" --&gt; F2     F1 -- "stderr" --&gt; F2   </pre>
<code>2&gt; /dev/null</code>	Verwerfen von <b>stderr</b> -Fehlermeldungen durch Umleiten an <b>/dev/null</b>	<pre> graph LR     K[Keyboard] -- "stdin" --&gt; P((PROCESS))     P -- "0" --&gt; M1["Monitor"]     P -- "1" --&gt; F1["File"]     P -- "2" --&gt; T["Trash Bin"]     M1 -- "stdout" --&gt; F1     T -- "stderr" --&gt; F1   </pre>
<code>&gt; file 2&gt;&amp;1</code>	Umleiten von <b>stdout</b> und <b>stderr</b> zum Überschreiben derselben Datei	<pre> graph LR     K[Keyboard] -- "stdin" --&gt; P((PROCESS))     P -- "0" --&gt; F1["File"]     P -- "1" --&gt; F2["File"]     P -- "2" --&gt; F2     F1 -- "stdout" --&gt; F2     F1 -- "stderr" --&gt; F2   </pre>
<code>&amp;&gt; file</code>	Umleiten von <b>stdout</b> und <b>stderr</b> zum Überschreiben derselben Datei	<pre> graph LR     K[Keyboard] -- "stdin" --&gt; P((PROCESS))     P -- "0" --&gt; F1["File"]     P -- "1" --&gt; F2["File"]     P -- "2" --&gt; F2     F1 -- "stdout" --&gt; F2     F1 -- "stderr" --&gt; F2   </pre>

Verwendung	Erläuterung	Visuelle Hilfe
<code>&gt; file 2&gt;&amp;1</code>	Umleiten von <b>stdout</b> und <b>stderr</b> zum Anhängen an dieselbe Datei	
<code>&amp;&gt;&gt; file</code>		



### Wichtig

Die Reihenfolge der Umleitungsvorgänge ist wichtig. Durch die folgende Sequenz werden Standardausgaben an **file** umgeleitet und anschließend Standard-Fehlernachrichten an denselben Ort wie Standardausgaben (**file**) umgeleitet.

```
> file 2>&1
```

Allerdings bewirkt die folgende Sequenz eine Umleitung in umgekehrter Reihenfolge. Dadurch wird die Standard-Fehlernachricht an den Standardort für Standardausgaben umgeleitet (das Terminalfenster, also keine Änderung) und *anschließend* werden nur Standardausgaben an **file** umgeleitet.

```
2>&1 > file
```

Deshalb bevorzugen einige Benutzer die Verwendung von Umleitungsoperatoren für die Zusammenführung:

<b>&amp;&gt;file</b>	anstelle	<b>&gt;file 2&gt;&amp;1</b>
	von	
<b>&amp;&gt;&gt;file</b>	anstelle	<b>&gt;&gt;file 2&gt;&amp;1</b> (in Bash 4/RHEL 6 und höher)
	von	

Andere Systemadministratoren und Programmierer, die auch andere Shells mit Bezug zu **bash** (Bourne-kompatible Shells genannt) zur Skripterstellung für Befehle verwenden, sind jedoch der Meinung, dass neuere Umleitungsoperatoren für die Zusammenführung vermieden werden sollten, da sie nicht standardisiert oder nicht in all diesen Shells implementiert sind und andere Einschränkungen für sie gelten.

Die Autoren dieses Kurses haben eine neutrale Haltung zu diesem Thema und in der Praxis lassen sich beide Syntaxen finden.

## Beispiele für die Ausgabeumleitung

Viele Routine-Administrationsaufgaben werden durch die Umleitung vereinfacht. Ziehen Sie die obige Tabelle für die folgenden Beispiele heran:

- Speichern Sie einen Zeitstempel zur späteren Referenz.

```
[user@host ~]$ date > /tmp/saved-timestamp
```

- Kopieren Sie die letzten 100 Zeilen aus einer Protokolldatei in eine andere Datei.

```
[user@host ~]$ tail -n 100 /var/log/dmesg > /tmp/last-100-boot-messages
```

- Verketten Sie vier Dateien zu einer.

```
[user@host ~]$ cat file1 file2 file3 file4 > /tmp/all-four-in-one
```

- Listen Sie die verborgenen und regulären Dateinamen des Benutzerverzeichnisses in einer Datei auf.

```
[user@host ~]$ ls -a > /tmp/my-file-names
```

- Fügen Sie die Ausgabe an eine vorhandene Datei an.

```
[user@host ~]$ echo "new line of information" >> /tmp/many-lines-of-information  
[user@host ~]$ diff previous-file current-file >> /tmp/tracking-changes-made
```

- Die nächsten Befehle generieren Fehlermeldungen, da einige Systemverzeichnisse für normale Benutzer nicht zugänglich sind. Beachten Sie, wie die Fehlermeldungen umgeleitet werden. Leiten Sie die Fehler in eine Datei um, während Sie die normale Befehlsausgabe am Terminal anzeigen.

```
[user@host ~]$ find /etc -name passwd 2> /tmp/errors
```

- Speichern Sie die Prozessausgabe und die Fehlermeldungen in getrennten Dateien.

```
[user@host ~]$ find /etc -name passwd > /tmp/output 2> /tmp/errors
```

- Ignorieren und verwerfen Sie Fehlermeldungen.

```
[user@host ~]$ find /etc -name passwd > /tmp/output 2> /dev/null
```

- Speichern Sie die Ausgabe und die generierten Fehler zusammen.

```
[user@host ~]$ find /etc -name passwd &> /tmp/save-both
```

- Hängen Sie die Ausgabe und die generierten Fehler an eine vorhandene Datei an.

```
[user@host ~]$ find /etc -name passwd >> /tmp/save-both 2>&1
```

## Errichten von Pipelines

Eine *Pipeline* ist eine Sequenz aus einem oder mehreren Befehlen, die durch das Pipe-Zeichen getrennt werden. Ein Pipe-Zeichen verbindet die Standardausgabe des ersten Befehls mit der Standardeingabe des nächsten Befehls.

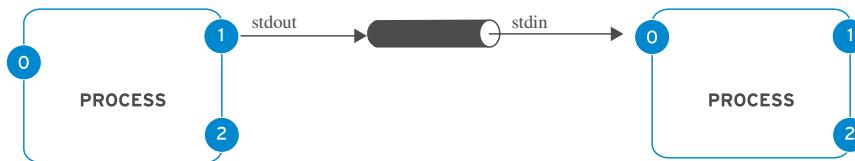


Abbildung 5.8: Prozess-I/O-Piping

Pipelines ermöglichen die Bearbeitung und Formatierung der Ausgabe eines Prozesses durch andere Prozesse, bevor sie an das Terminal ausgegeben wird. Man könnte sich das beispielsweise so vorstellen, dass Daten in der Pipeline von einem Prozess zu einem anderen „fließen“ und durch jeden Befehl in der Pipeline, den sie durchlaufen, leicht verändert werden.



### Anmerkung

Sowohl Pipelines als auch I/O-Umleitung verändern Standardausgaben und Standardeingaben. Durch Umleitung werden Standardausgaben an Dateien gesendet oder Standardeingaben von Dateien empfangen. Pipes senden Standardausgaben von einem Prozess an die Standardeingabe eines anderen Prozesses.

## Pipeline-Beispiele

Bei diesem Beispiel wird die Ausgabe des Befehls **ls** mit **less** auf jeweils einem Bildschirm auf dem Terminal angezeigt.

```
[user@host ~]$ ls -l /usr/bin | less
```

Die Ausgabe des Befehls **ls** wird an den Befehl **wc -l** weitergeleitet, der die Anzahl der Zeilen zählt, die von **ls** empfangen werden, und sie am Terminal ausgibt.

```
[user@host ~]$ ls | wc -l
```

In dieser Pipeline werden durch **head** die ersten 10 Zeilen der Ausgabe von **ls -t** ausgegeben, wobei das Endergebnis in eine Datei umgeleitet wird.

```
[user@host ~]$ ls -t | head -n 10 > /tmp/ten-last-changed-files
```

## Pipelines, Umleitung und der Befehl tee

Wird die Umleitung mit einer Pipeline kombiniert, richtet die Shell zuerst die gesamte Pipeline ein und leitet dann Eingaben/Ausgaben weiter. Wird die Ausgabeumleitung in der Mitte einer Pipeline verwendet, dann wird die Ausgabe an die Datei und nicht an den nächsten Befehl in der Pipeline geleitet.

In diesem Beispiel wird die Ausgabe des Befehls **ls** an die Datei geleitet und **less** zeigt nichts auf dem Terminal an.

```
[user@host ~]$ ls > /tmp/saved-output | less
```

Der Befehl **tee** umgeht diese Einschränkung. In einer Pipeline wird mit **tee** die Standardeingabe zur Standardausgabe kopiert und die Standardausgabe an die Dateien umgeleitet, die als

Argumente für den Befehl angegeben sind. Wenn Sie sich Daten als Wasser vorstellen, das durch ein Rohr fließt, können Sie sich **tee** als T-Stück im Rohr vorstellen, das Ausgaben in zwei Richtungen leitet.

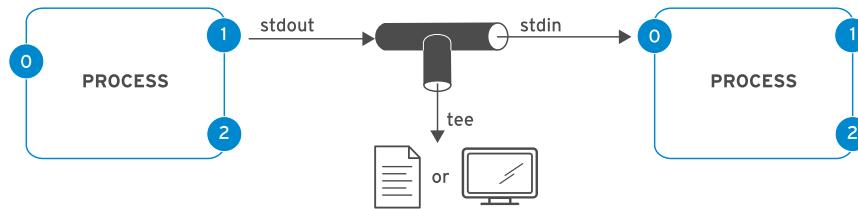


Abbildung 5.9: Prozess-I/O-Piping mit **tee**

## Pipeline-Beispiele unter Verwendung des **tee**-Befehls

In diesem Beispiel wird die Ausgabe des Befehls **ls** an die Datei umgeleitet und an **less** übergeben, damit sie auf jeweils einem Bildschirm auf dem Terminal angezeigt wird.

```
[user@host ~]$ ls -1 | tee /tmp/saved-output | less
```

Wird **tee** am Ende einer Pipeline verwendet, kann die finale Ausgabe eines Befehls gespeichert und gleichzeitig an das Terminal ausgegeben werden.

```
[user@host ~]$ ls -t | head -n 10 | tee /tmp/ten-last-changed-files
```



### Wichtig

Die Standard-Fehlerausgabe kann über ein Pipe umgeleitet werden, die Umleitungsoperatoren für die Zusammenführung (**&>** und **&>>**) können dafür jedoch nicht verwendet werden.

Nachfolgend finden Sie die korrekte Methode zum Umleiten von Standardausgaben und Standard-Fehlerausgaben durch ein Pipe:

```
[user@host ~]$ find -name /passwd 2>&1 | less
```



### Literaturhinweise

**info bash** (*Das GNU Bash-Referenzhandbuch*)

- Abschnitt 3.2.2: Pipelines
- Abschnitt 3.6: Umleitungen

**info coreutils 'tee invocation'** (*The GNU coreutils Manual*)

- Abschnitt 17.1: Umleiten der Ausgabe an mehrere Dateien oder Prozesse

Manpages **bash(1)**, **cat(1)**, **head(1)**, **less(1)**, **mail(1)**, **tee(1)**, **tty(1)**, **wc(1)**

## ► Quiz

# Umleiten von Ausgaben an eine Datei oder ein Programm

Wählen Sie die richtige Antwort auf die folgenden Fragen aus:

- ▶ 1. Welche Antwort zeigt die Ausgabe auf einem Terminal an und ignoriert alle Fehler?
  - a. &>file
  - b. 2>&gt;file
  - c. 2 >/dev/null
  - d. 1>/dev/null
  
- ▶ 2. Welche Antwort sendet die Ausgabe an eine Datei und sendet Fehler an eine andere Datei?
  - a. >file 2>file2
  - b. >file 1>file2
  - c. >file &2>file2
  - d. | tee file
  
- ▶ 3. Welche Antwort sendet sowohl Ausgaben als auch Fehler an eine Datei, wobei die Datei erstellt oder deren Inhalt überschrieben wird?
  - a. | tee file
  - b. 2 &>file
  - c. 1&>file
  - d. &>file
  
- ▶ 4. Welche Antwort sendet die Ausgabe und Fehler an dieselbe Datei und stellt sicher, dass der vorhandene Dateiinhalt erhalten bleibt?
  - a. >file 2>file2
  - b. &>file
  - c. >>file 2>&1
  - d. >>file 1>&1
  
- ▶ 5. Welche Antwort verwirft alle Meldungen, die normalerweise an das Terminal gesendet werden?
  - a. >file 2>file2
  - b. &>/dev/null
  - c. &>/dev/null 2>file
  - d. &>file

► **6. Welche Antwort sendet die Ausgabe gleichzeitig an den Bildschirm und an eine Datei?**

- a. &>/dev/null
- b. >file 2>file2
- c. | tee file
- d. | < file

► **7. Welche Antwort speichert die Ausgabe in einer Datei und verwirft Fehlermeldungen?**

- a. &>file
- b. | tee file 2>/dev/null
- c. > file 1>/dev/null
- d. > file 2>/dev/null

## ► Lösung

# Umleiten von Ausgaben an eine Datei oder ein Programm

Wählen Sie die richtige Antwort auf die folgenden Fragen aus:

► 1. Welche Antwort zeigt die Ausgabe auf einem Terminal an und ignoriert alle Fehler?

- a. &>file
- b. 2>&gt;file
- c. 2 >/dev/null
- d. 1 >/dev/null

► 2. Welche Antwort sendet die Ausgabe an eine Datei und sendet Fehler an eine andere Datei?

- a. >file 2>file2
- b. >file 1>file2
- c. >file &2>file2
- d. | tee file

► 3. Welche Antwort sendet sowohl Ausgaben als auch Fehler an eine Datei, wobei die Datei erstellt oder deren Inhalt überschrieben wird?

- a. | tee file
- b. 2 &>file
- c. 1 &>file
- d. &>file

► 4. Welche Antwort sendet die Ausgabe und Fehler an dieselbe Datei und stellt sicher, dass der vorhandene Dateiinhalt erhalten bleibt?

- a. >file 2>file2
- b. &>file
- c. >>file 2>&1
- d. >>file 1>&1

► 5. Welche Antwort verwirft alle Meldungen, die normalerweise an das Terminal gesendet werden?

- a. >file 2>file2
- b. &>/dev/null
- c. &>/dev/null 2>file
- d. &>file

► **6. Welche Antwort sendet die Ausgabe gleichzeitig an den Bildschirm und an eine Datei?**

- a. &>/dev/null
- b. >file 2>file2
- c. | tee file
- d. | < file

► **7. Welche Antwort speichert die Ausgabe in einer Datei und verwirft Fehlermeldungen?**

- a. &>file
- b. | tee file 2>/dev/null
- c. > file 1>/dev/null
- d. > file 2>/dev/null

# Bearbeiten von Textdateien an der Shell-Eingabeaufforderung

---

## Ziele

Nach Abschluss dieses Abschnitts sollten Sie in der Lage sein, Textdateien über die Befehlszeile mit dem **vim**-Editor zu erstellen und zu bearbeiten.

## Bearbeiten von Dateien mit Vim

Ein wesentliches Designprinzip von Linux besteht darin, dass Informationen und Konfigurationseinstellungen im Allgemeinen in textbasierten Dateien gespeichert werden. Diese Dateien können auf verschiedene Arten als Listen von Einstellungen, in INI-ähnliche Formate wie strukturiertes XML oder YAML usw. strukturiert werden. Der Vorteil von Textdateien ist jedoch, dass sie mit jedem einfachen Texteditor angezeigt und bearbeitet werden können.

Vim ist eine verbesserte Version des **vi**-Editors, der mit Linux- und UNIX-Systemen verteilt wird. Vim ist hochgradig konfigurierbar und effizient für erfahrene Benutzer. Er umfasst Funktionen wie Split-Screen-Editing, Farbformatierung und Markierung für die Textbearbeitung.

## Warum sollte Vim erlernt werden?

Sie sollten wissen, wie Sie mindestens einen Texteditor verwenden, der von der textbasierten Shell-Eingabeaufforderung aus verwendet werden kann. Sie können dann textbasierte Konfigurationsdateien aus einem Terminalfenster oder von Remote-Anmeldungen durch **ssh** oder die Web Console bearbeiten. Sie benötigen dann keinen Zugriff auf einen grafischen Desktop, um Dateien auf einem Server zu bearbeiten, und tatsächlich muss auf diesem Server möglicherweise überhaupt keine grafische Desktopumgebung ausgeführt werden.

Aber warum sollten Sie Vim erlernen, anstatt andere Möglichkeiten zu nutzen? Der Hauptgrund dafür ist, dass Vim fast immer auf einem Server installiert ist, wenn überhaupt ein Texteditor vorhanden ist. Das liegt daran, dass **vi** durch den POSIX-Standard angegeben wurde, dem Linux und viele andere UNIX-ähnliche Betriebssysteme weitgehend entsprechen.

Darüber hinaus wird Vim oft als **vi**-Implementierung auf anderen gängigen Betriebssystemen oder Distributionen verwendet. Zum Beispiel enthält macOS derzeit standardmäßig eine Lightweight-Installation von Vim. Die für Linux erlernten Vim-Kenntnisse können Ihnen also auch dabei helfen, andere Aufgaben zu erledigen.

## Starten von Vim

Vim kann in Red Hat Enterprise Linux auf zwei verschiedene Arten installiert sein. Dies kann sich auf die Funktionen und Vim-Befehle auswirken, die Ihnen zur Verfügung stehen.

Auf Ihrem Server ist möglicherweise nur das Paket *vim-minimal* installiert. Dies ist eine sehr abgespeckte Installation, die nur den Kernfunktionssatz und den grundlegenden **vi**-Befehl enthält. In diesem Fall können Sie mit **vi filename** eine Datei zum Bearbeiten öffnen und alle in diesem Abschnitt beschriebenen Kernfunktionen stehen Ihnen zur Verfügung.

Alternativ könnte auf Ihrem Server das Paket *vim-enhanced* installiert sein. Dieses Paket bietet einen viel umfassenderen Funktionssatz, ein Online-Hilfesystem und ein Tutorial-Programm. Verwenden Sie den Befehl **vim**, um Vim in diesem erweiterten Modus zu starten.

```
[user@host ~]$ vim filename
```

Auf jeden Fall funktionieren die in diesem Abschnitt beschriebenen Kernfunktionen mit beiden Befehlen.



### Anmerkung

Wenn *vim-enhanced* installiert ist, wird für reguläre Benutzer ein Shell-Alias festgelegt, damit sie bei Ausführung des Befehls **vi** automatisch den Befehl **vim** erhalten. Dies gilt nicht für **root** und andere Benutzer mit UIDs unter 200 (die von Systemservices verwendet werden).

Wenn Sie Dateien als Benutzer **root** bearbeiten und Sie erwarten, dass **vi** im erweiterten Modus ausgeführt wird, kann dies eine Überraschung sein. Ebenso wenn *vim-enhanced* installiert ist und ein regulärer Benutzer aus irgendeinem Grund das einfache **vi** möchte, muss er möglicherweise **\vi** verwenden, um den Alias vorübergehend zu überschreiben.

Fortgeschrittene Benutzer können **\vi --version** und **vim --version** verwenden, um die Funktionssätze der beiden Befehle zu vergleichen.

## Vim-Betriebsmodi

Ein besonderes Merkmal von Vim ist, dass er mehrere *Betriebsmodi* bereitstellt, wie *Befehlsmodus*, *erweiterter Befehlsmodus*, *Bearbeitungsmodus* und *visueller Modus*. Je nach Modus geben Sie möglicherweise Befehle aus, bearbeiten Text oder arbeiten mit Textblöcken. Als neuer Vim-Benutzer sollten Sie immer Ihren aktuellen Modus kennen, da Tastatureingaben in verschiedenen Modi unterschiedliche Auswirkungen haben.

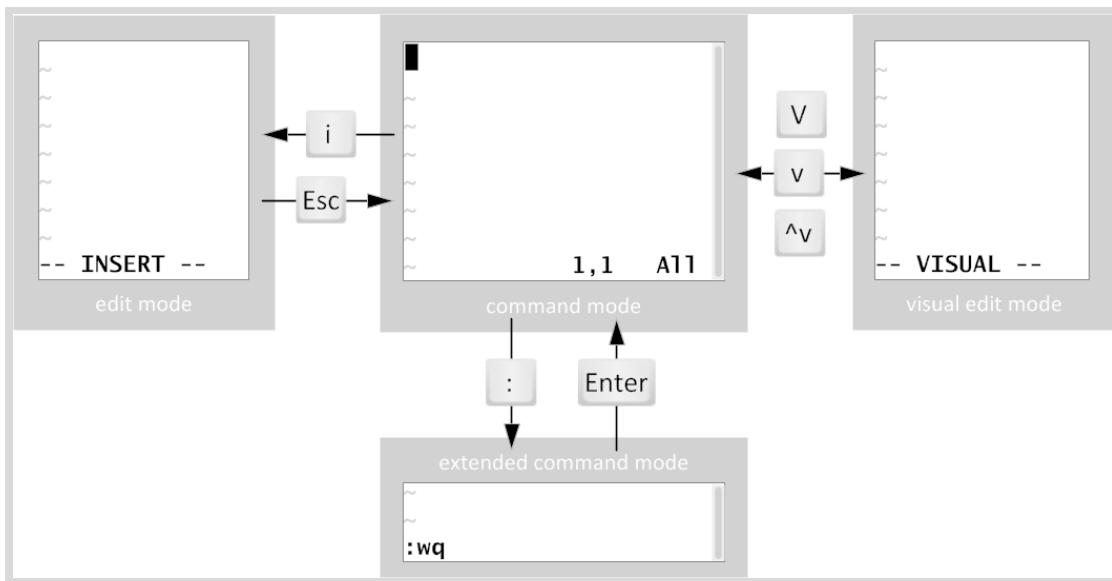


Abbildung 5.10: Wechseln zwischen Vim-Modi

Wenn Sie Vim zum ersten Mal öffnen, ist der *Befehlsmodus* aktiv, der zum Navigieren, Ausschneiden und Einfügen und für andere Textbearbeitungsfunktionen verwendet wird. Wechseln Sie mit einem Tastendruck in die einzelnen anderen Modi, um auf bestimmte Bearbeitungsfunktionen zuzugreifen:

## Kapitel 5 | Erstellen, Anzeigen und Bearbeiten von Textdateien

- Mit der Eingabe von **i** wechseln Sie in den *Einfügemodus*, in dem alle eingegebenen Texte zum Dateinhalt hinzugefügt werden. Drücken Sie die **Esc**-Taste, um zum Befehlsmodus zurückzuwechseln.
- Mit **v** wechseln Sie in den *visuellen Modus*, in dem mehrere Zeichen für die Textbearbeitung ausgewählt werden können. Verwenden Sie **Umschalt+v**, um mehrere Zeilen auszuwählen, und **Strg+v**, um einen Block auszuwählen. Durch Drücken der gleichen Tasten wie beim Wechseln in den visuellen Modus (**v**, **Umschalt+v** oder **Strg+v**) verlassen Sie diesen Modus wieder.
- Mit **:** wechseln Sie in den *erweiterten Befehlsmodus* für Aufgaben wie Schreiben der Dateien (um sie zu speichern) und zum Beenden des Vim-Editors.



### Anmerkung

Wenn Sie sich nicht sicher sind, in welchem Modus sich Vim befindet, können Sie versuchen, einige Male **Esc** zu drücken, um in den Befehlsmodus zurückzukehren. Das Drücken von **Esc** im Befehlsmodus richtet keinen Schaden an, daher sind ein paar zusätzliche Tasteneingaben kein Problem.

## Minimaler, grundlegender Vim-Workflow

Vim verfügt über effiziente, koordinierte Tastatureingaben für erweiterte Bearbeitungsaufgaben. Wenn Vim auch nach einiger Übung sehr nützlich sein kann, können sich neue Benutzer durch die zahlreichen Funktionen überfordert fühlen.

Die Taste **i** versetzt Vim in den Einfügemodus. Der gesamte danach eingegebene Text wird als Dateinhalt behandelt, bis Sie den Einfügemodus verlassen. Mit der Taste **Esc** verlassen Sie den Einfügemodus und wechseln in Vim zum Befehlsmodus zurück. Mit der Taste **u** wird die letzte Bearbeitung rückgängig gemacht. Drücken Sie die Taste **x**, um ein einzelnes Zeichen zu löschen. Der Befehl **:w** schreibt (speichert) die Datei und verbleibt für weitere Bearbeitungen im Befehlsmodus. Der Befehl **:wq** schreibt (speichert) die Datei und beendet Vim. Mit dem Befehl **:q!** wird Vim beendet und alle Änderungen an der Datei seit der letzten Speicherung werden verworfen. Der Vim-Benutzer muss diese Befehle erlernen, um eine Bearbeitungsaufgabe auszuführen.

## Neuanordnen von vorhandenem Text

In Vim wird Kopieren und Einfügen als *yank and put* bezeichnet. Dafür werden die Befehlszeichen **y** und **p** verwendet. Setzen Sie den Cursor auf das erste auszuählende Zeichen und wechseln Sie in den visuellen Modus. Verwenden Sie die Pfeiltasten, um die visuelle Auswahl zu erweitern. Drücken Sie dann **y**, um die Auswahl in den Arbeitsspeicher zu kopieren (*yank*). Positionieren Sie den Cursor an der neuen Position und drücken Sie **p**, um die Auswahl an der Cursorposition einzufügen (*put*).

## Visueller Modus in Vim

Der visuelle Modus bietet eine großartige Möglichkeit, Text zu markieren und zu bearbeiten. Es gibt drei Tasteneingaben:

- Zeichenmodus: **v**
- Zeilenmodus: **Umschalt+v**
- Blockmodus: **Strg+v**

Der Zeichenmodus markiert Sätze in einem Textblock. Das Wort **VISUAL** wird am unteren Bildschirmrand angezeigt. Drücken Sie **v**, um in den visuellen Zeichenmodus zu wechseln.

**Umschalt+v** wechselt in den Zeilenmodus. **VISUAL LINE** wird am unteren Bildschirmrand angezeigt.

Der visuelle Blockmodus eignet sich perfekt zum Bearbeiten von Datendateien. Drücken Sie am Cursor **Strg+v**, um in den visuellen Blockmodus zu wechseln. **VISUAL BLOCK** wird am unteren Bildschirmrand angezeigt. Verwenden Sie die Pfeiltasten, um den zu ändernden Abschnitt zu markieren.



### Anmerkung

Vim verfügt über viele Funktionen, aber Sie sollten zuerst den grundlegenden Workflow beherrschen. Sie müssen nicht den gesamten Editor und seine Funktionen auf einmal verstehen. Machen Sie sich durch Üben mit diesen Grundlagen vertraut und erweitern Sie Ihr Vim-Vokabular, indem Sie zusätzliche Vim-Befehle (Tasteneingaben) erlernen.

Die Übung für diesen Abschnitt führt Sie in den Befehl **vimtutor** ein. Dieses Tutorial, das mit *vim-enhanced* ausgeliefert wird, ist eine hervorragende Möglichkeit, die Kernfunktionalität von Vim zu erlernen.



### Literaturhinweise

Manpage **vim(1)**

Der Befehl **:help** in **vim** (wenn das Paket *vim-enhanced* installiert ist).

### Der Vim-Editor

<http://www.vim.org/>

### Erste Schritte mit dem visuellen Modus von Vim

<https://opensource.com/article/19/2/getting-started-vim-visual-mode>

## ► Angeleitete Übung

# Bearbeiten von Textdateien an der Shell-Eingabeaufforderung

In dieser Übung verwenden Sie **vimtutor**, um grundlegende Bearbeitungstechniken im vim-Editor zu üben.

## Ergebnisse

Es werden folgende Fähigkeiten vermittelt:

- Bearbeiten von Dateien mit Vim
- Erwerben von Kompetenz in Vim mit **vimtutor**

## Bevor Sie Beginnen

Melden Sie sich als Benutzer **student** mit dem Passwort **student** bei **workstation** an.

Führen Sie auf **workstation** den Befehl **lab edit-vim start** aus. Dieses Skript überprüft, ob der Zielserver ausgeführt wird.

```
[student@workstation ~]$ lab edit-vim start
```

- 1. Melden Sie sich mit dem Befehl **ssh** bei **servera** an.

```
[student@workstation ~]$ ssh student@servera  
...output omitted...  
[student@servera ~]$
```

- 2. Öffnen Sie **vimtutor**. Lesen Sie den Begrüßungsbildschirm und bearbeiten Sie *Lesson 1.1*.

```
[student@servera ~]$ vimtutor
```

In der Präsentation erfolgt die Navigation mit den Pfeiltasten. Zur Zeit der Entwicklung von **vi** konnten sich Benutzer nicht darauf verlassen, Pfeiltasten oder funktionierende Tastaturbelegungen für Pfeiltasten zu haben, um den Cursor zu bewegen. Daher wurde **vi** ursprünglich so entwickelt, dass der Cursor mit Befehlen für Standardzeichtasten bewegt wurde, wie die nebeneinander angeordneten Tasten **H**, **J**, **K** und **L**.

So können Sie sich die Tasten merken:

**hang back**, **jump down**, **kick up**, **leap forward**.

- 3. Arbeiten Sie im **vimtutor**-Fenster *Lesson 1.2* durch.

In dieser Lektion lernen Benutzer, wie sie das Programm beenden, ohne unerwünschte Änderungen beizubehalten. Alle Änderungen gehen verloren. Manchmal ist es vorzuziehen, eine kritische Datei in einem inkorrekt Zustand zu belassen.

- 4. Arbeiten Sie im **vimtutor**-Fenster *Lesson 1.3* durch.

Vim verfügt über schnelle, effiziente Tastatureingaben, um eine genaue Anzahl Wörter, Zeilen, Sätze und Absätze zu löschen. Jede Bearbeitungsaufgabe kann aber durchgeführt werden, indem Sie **x** für das Löschen einzelner Zeichen verwenden.

- **5.** Arbeiten Sie im **vimtutor**-Fenster *Lesson 1.4* durch.

Bei den meisten Bearbeitungsaufgaben wird als erste Taste **i** gedrückt.

- **6.** Arbeiten Sie im **vimtutor**-Fenster *Lesson 1.5* durch.

Im Kurs wurde nur der Befehl **i** (*insert*) als Taste zum Wechseln in den Bearbeitungsmodus behandelt. Diese **vimtutor**-Lektion zeigt im Einfügemodus weitere verfügbare Tastatureingaben zum Ändern der Cursorposition. Im Einfügemodus ist eingegebener Text immer Dateiinhalt.

- **7.** Arbeiten Sie im **vimtutor**-Fenster *Lesson 1.6* durch.

Geben Sie **:wq** ein, um die Datei zu speichern und den Editor zu verlassen.

- **8.** Lesen Sie im **vimtutor**-Fenster die *Lesson 1 Summary*.

Der Befehl **vimtutor** umfasst sechs weitere mehrstufige Lektionen. Diese Lektionen werden nicht als Teil dieses Kurses behandelt. Sie können sie aber auf eigene Faust erkunden, um weitere Informationen zu erhalten.

- **9.** Beenden Sie **servera**.

```
[student@servera ~]$ exit
logout
Connection to servera closed.
[student@workstation ~]$
```

## Beenden

Führen Sie auf **workstation** das Skript **lab edit-vim finish** aus, um diese Übung zu beenden.

```
[student@workstation ~]$ lab edit-vim finish
```

Hiermit ist die angeleitete Übung beendet.

# Ändern der Shell-Umgebung

## Ziele

Nach Abschluss dieses Abschnitts sollten Sie in der Lage sein, Shell-Variablen zur Ausführung von Befehlen und zum Bearbeiten von Bash-Startskripts zur Festlegung von Shell- und Umgebungsvariablen festzulegen, um das Verhalten der Shell und von Programmen zu ändern, die in der Shell ausgeführt werden.

## Verwenden von Shell-Variablen

Die Bash-Shell ermöglicht die Festlegung von *Shell-Variablen*, um Befehle auszuführen oder das Verhalten der Shell zu ändern. Sie können Shell-Variablen auch als *Umgebungsvariablen* exportieren, die automatisch in Programme kopiert werden, die beim Start von dieser Shell ausgeführt werden. Anhand von Variablen können Sie die Ausführung eines Befehls mit einem langen Argument vereinfachen oder eine allgemeine Einstellung auf Befehle anwenden, die von dieser Shell ausgeführt werden.

Shell-Variablen sind für eine bestimmte Shell-Sitzung eindeutig. Wenn Sie zwei Terminalfenster oder zwei unabhängige Anmeldesitzungen auf demselben Remote-Server ausführen, führen Sie zwei Shells aus. Jede Shell verfügt über eigene Werte für ihre Shell-Variablen.

## Zuweisen von Werten zu Variablen

Weisen Sie einer Shell-Variable mit der folgenden Syntax einen Wert zu:

```
VARIABLENAME=value
```

Variablennamen können Groß- oder Kleinbuchstaben, Ziffern und den Unterstrich (\_) enthalten. Die folgenden Befehle legen beispielsweise Shell-Variablen fest:

```
[user@host ~]$ COUNT=40
[user@host ~]$ first_name=John
[user@host ~]$ file1=/tmp/abc
[user@host ~]$ _ID=RH123
```

Denken Sie daran, dass diese Änderung nur die Shell betrifft, in der Sie den Befehl ausführen, nicht jedoch alle anderen Shells, die Sie auf diesem Server ausführen.

Mit dem Befehl **set** können Sie alle aktuell festgelegten Shell-Variablen auflisten. (Der Befehl listet auch alle Shell-Funktionen auf, die Sie ignorieren können.) Diese Liste ist ziemlich lang, daher können Sie die Ausgabe an den Befehl **less** leiten, um die Ausgabe seitenweise anzuzeigen.

```
[user@host ~]$ set | less
BASH=/usr/bin/bash
BASHOPTS=checkwinsize:cmdhist:complete_fullquote:expand_aliases:extglob:extquote:
force_fignore:histappend:interactive_comments:progcomp:promptvars:sourcepath
BASHRC_SOURCE=Y
...output omitted...
```

## Abrufen von Werten mit Variablenweiterung

Sie können anhand der Variablenweiterung auf den Wert einer Variable verweisen, die Sie festgelegt haben. Stellen Sie dazu dem Namen der Variable ein Dollarzeichen (\$) voran. Im folgenden Beispiel gibt der Befehl **echo** den Rest der eingegebenen Befehlszeile aus, aber nachdem die Variablenweiterung durchgeführt wurde.

Der folgende Befehl legt beispielsweise die Shell-Variable **COUNT** auf **40** fest.

```
[user@host ~]$ COUNT=40
```

Wenn Sie den Befehl **echo COUNT** eingeben, wird die Zeichenfolge **COUNT** ausgegeben.

```
[user@host ~]$ echo COUNT  
COUNT
```

Wenn Sie aber den Befehl **echo \$COUNT** eingeben, wird der Wert der Variable **COUNT** ausgegeben.

```
[user@host ~]$ echo $COUNT  
40
```

Ein praktischeres Beispiel wäre die Verwendung einer Variable, um auf einen langen Dateinamen für mehrere Befehle zu verweisen.

```
[user@host ~]$ file1=/tmp/tmp.z9pXW0HqcC  
[user@host ~]$ ls -l $file1  
-rw----- 1 student student 1452 Jan 22 14:39 /tmp/tmp.z9pXW0HqcC  
[user@host ~]$ rm $file1  
[user@host ~]$ ls -l $file1  
total 0
```



### Wichtig

Wenn an den Variablennamen nachstehende Zeichen angehängt sind, müssen Sie den Variablennamen möglicherweise mit geschweiften Klammern schützen. Sie können geschweifte Klammern immer in Variablenweiterungen verwenden, aber Sie werden auch viele Beispiele sehen, bei denen diese nicht benötigt und daher weggelassen werden.

Im folgenden Beispiel versucht der erste **echo**-Befehl, die nicht vorhandene Variable **COUNTx** zu erweitern, was keinen Fehler verursacht, sondern einfach nichts zurückgibt.

```
[user@host ~]$ echo Repeat $COUNTx  
Repeat  
[user@host ~]$ echo Repeat ${COUNT}x  
Repeat 40x
```

## Bash mit Shell-Variablen konfigurieren

Einige Shell-Variablen werden beim Start der Bash(-Shell) festgelegt, können jedoch geändert werden, um das Verhalten der Shell anzupassen.

Zwei Shell-Variablen, die sich auf den Shell-Verlauf und den Befehl **history** auswirken, sind beispielsweise **HISTFILE** und **HISTFILESIZE**. Wenn **HISTFILE** festgelegt ist, gibt sie den Speicherort einer Datei an, in der der Shell-Verlauf gespeichert wird, wenn der Befehl beendet ist. Standardmäßig ist dies die Datei `~/.bash_history` des Benutzers. Die Variable **HISTFILESIZE** gibt an, wie viele Befehle aus dem Verlauf in dieser Datei gespeichert werden sollen.

Ein anderes Beispiel ist **PS1**, eine Shell-Variable, die das Erscheinungsbild der Shell-Eingabeaufforderung steuert. Wenn Sie diesen Wert ändern, wird das Erscheinungsbild der Shell-Eingabeaufforderung geändert. Der Abschnitt „PROMPTING“ auf der Manpage **bash(1)** enthält eine Auflistung von Sonderzeichenerweiterungen, die von der Eingabeaufforderung unterstützt werden.

```
[user@host ~]$ PS1="bash\$ "
bash$ PS1="[\u@\h \w]\$ "
[user@host ~]$
```

Diese beiden Punkte sollten Sie beim obigen Beispiel beachten: Erstens: Es ist praktisch immer wünschenswert, an die Eingabeaufforderung als letztes Zeichen ein Leerzeichen anzuhängen, da der von PS1 festgelegte Wert eine Eingabeaufforderung ist. Zweitens: Wenn der Wert einer Variable eine Form von Leerraum enthält, z. B. Leerzeichen, Tabulator oder Absatzwechsel, muss der Wert immer in einfache oder doppelte Anführungszeichen gesetzt werden. Das ist nicht optional. Wenn die Anführungszeichen weggelassen werden, treten unerwartete Ergebnisse auf. Überprüfen Sie das PS1-Beispiel oben und beachten Sie, dass sowohl die Empfehlung (Leerzeichen) als auch die Regel (Anführungszeichen) eingehalten wurden.

## Konfigurieren von Programmen mit Umgebungsvariablen

Die Shell stellt für die Programme, die Sie von dieser Shell aus ausführen, eine *Umgebung* bereit. Diese Umgebung enthält unter anderem Informationen zum aktuellen Arbeitsverzeichnis im Dateisystem, zu den Befehlszeilenoptionen, die an das Programm übergeben werden, und zu den Werten von *Umgebungsvariablen*. Die Programme können diese Umgebungsvariablen verwenden, um ihr Verhalten oder ihre Standardeinstellungen zu ändern.

Shell-Variablen, die keine Umgebungsvariablen sind, können nur von der Shell verwendet werden. Umgebungsvariablen können von der Shell *und* von Programmen verwendet werden, die auf dieser Shell ausgeführt werden.



### Anmerkung

**HISTFILE**, **HISTFILESIZE** und **PS1**, die im vorherigen Abschnitt behandelt wurden, müssen nicht als Umgebungsvariablen exportiert werden, da sie nur von der Shell selbst verwendet werden, nicht von den Programmen, die Sie auf der Shell ausführen.

Sie können jede in der Shell definierte Variable in eine Umgebungsvariable umwandeln, indem Sie sie für den Export mit dem Befehl **export** kennzeichnen.

```
[user@host ~]$ EDITOR=vim  
[user@host ~]$ export EDITOR
```

Sie können eine Variable in einem Schritt festlegen und exportieren:

```
[user@host ~]$ export EDITOR=vim
```

Anwendungen und Sitzungen verwenden diese Variablen, um ihr Verhalten festzulegen. Beispielsweise setzt die Shell beim Start automatisch die Variable **HOME** auf den Dateinamen des Benutzerverzeichnisses. Dies kann verwendet werden, damit Programme feststellen können, wo Dateien gespeichert werden sollen.

Ein anderes Beispiel ist **LANG**, die das Gebietsschema festlegt. Damit werden die bevorzugte Sprache für die Programmausgabe, der Zeichensatz, die Formatierung von Datumsangaben, Zahlen und Währungen sowie die Sortierreihenfolge für Programme festgelegt. Wenn die Variable **auf en\_US.UTF-8** festgelegt ist, verwendet das Gebietsschema US-Englisch mit der Unicode-Zeichenkodierung UTF-8. Wenn die Variable auf etwas anderes, beispielsweise **fr\_FR.UTF-8**, festgelegt ist, wird Französisch mit der Unicode-Codierung UTF-8 verwendet.

```
[user@host ~]$ date  
Tue Jan 22 16:37:45 CST 2019  
[user@host ~]$ export LANG=fr_FR.UTF-8  
[user@host ~]$ date  
mar. janv. 22 16:38:14 CST 2019
```

Eine weitere wichtige Umgebungsvariable ist **PATH**. Die Variable **PATH** enthält eine Liste von durch Doppelpunkt getrennter Verzeichnisse, die Programme enthalten:

```
[user@host ~]$ echo $PATH  
/home/user/.local/bin:/home/user/bin:/usr/share/Modules/bin:/usr/local/bin:/usr/  
bin:/usr/local/sbin:/usr/sbin
```

Wenn Sie einen Befehl wie **ls** ausführen, durchsucht die Shell der Reihenfolge nach jedes dieser Verzeichnisse nach der ausführbaren Datei **ls** und führt die erste übereinstimmende Datei aus, die gefunden wird. (Bei einem typischen System ist das **/usr/bin/ls**.)

Sie können problemlos weitere Verzeichnisse am Ende Ihres **PATH** hinzufügen. Möglicherweise verfügen Sie über ausführbare Programme oder Skripts, die Sie wie reguläre Befehle in **/home/user/sbin** ausführen möchten. Sie können **/home/user/sbin** am Ende Ihres **PATH** für die aktuelle Sitzung wie folgt hinzufügen:

```
[user@host ~]$ export PATH=${PATH}:/home/user/sbin
```

Um alle Umgebungsvariablen für eine bestimmte Shell aufzulisten, führen Sie den Befehl **env** aus:

```
[user@host ~]$ env  
...output omitted...  
LANG=en_US.UTF-8  
HISTCONTROL=ignoredups  
HOSTNAME=host.example.com  
XDG_SESSION_ID=4  
...output omitted...
```

## Festlegen des Standard-Texteditors

Die Umgebungsvariable **EDITOR** gibt das Programm an, das Sie als Standard-Texteditor für Befehlszeilenprogramme verwenden möchten. Wenn nichts angegeben ist, verwenden viele Programme **vi** oder **vim**, Sie können diese Einstellung jedoch bei Bedarf überschreiben:

```
[user@host ~]$ export EDITOR=nano
```



### Wichtig

Normalerweise haben Umgebungsvariablen und Shell-Variablen, die automatisch von der Shell festgelegt werden, Namen in Großbuchstaben. Wenn Sie eigene Variablen festlegen, können Sie auch Namen verwenden, die aus Kleinbuchstaben bestehen, um Namenskollisionen zu vermeiden.

## Automatisches Festlegen von Variablen

Wenn Sie Shell- oder Umgebungsvariablen beim Start Ihrer Shell automatisch festlegen möchten, können Sie die Bash-Startskripts bearbeiten. Wenn Sie die Bash(-Shell) starten, werden mehrere Textdateien mit Shell-Befehlen ausgeführt, die die Shell-Umgebung initialisieren.

Welche Skripts genau ausgeführt werden, hängt davon ab, wie die Shell gestartet wurde, ob es sich um eine interaktive Login-Shell, eine interaktive Shell ohne Anmeldefunktion oder ein Shell-Skript handelt.

Angenommen, es sind die Standarddateien **/etc/profile**, **/etc/bashrc** und **~/.bash\_profile** vorhanden. Wenn Sie eine Änderung an Ihrem Benutzerkonto vornehmen möchten, die sich auf alle Ihre interaktiven Shell-Eingabeaufforderungen beim Start auswirkt, bearbeiten Sie die Datei **~/.bashrc**. Sie können beispielsweise den Standardeditor dieses Kontos auf **nano** festlegen, indem Sie die zu lesenden Datei bearbeiten:

```
# .bashrc  
  
# Source global definitions  
if [ -f /etc/bashrc ]; then  
    . /etc/bashrc  
fi  
  
# User specific environment  
PATH="$HOME/.local/bin:$HOME/bin:$PATH"  
export PATH  
  
# User specific aliases and functions  
export EDITOR=nano
```



### Anmerkung

Am besten passen Sie Einstellungen für alle Benutzerkonten an, indem Sie eine Datei hinzufügen, deren Name auf `.sh` endet und die Änderungen für das Verzeichnis `/etc/profile.d` enthält. Dazu müssen Sie als `root`-Benutzer angemeldet sein.

## Zurücksetzen und Exportieren von Variablen

Verwenden Sie den Befehl `unset`, um eine Variable vollständig zu löschen und vom Export auszuschließen:

```
[user@host ~]$ echo $file1  
/tmp/tmp.z9pXW0Hqcc  
[user@host ~]$ unset file1  
[user@host ~]$ echo $file1  
  
[user@host ~]$
```

Mit dem Befehl `export -n` schließen Sie eine Variable von Export aus, ohne sie zurückzusetzen:

```
[user@host ~]$ export -n PS1
```



### Literaturhinweise

Manpages `bash(1)`, `env(1)` und `builtins(1)`

## ► Angeleitete Übung

# Ändern der Shell-Umgebung

In dieser Übung verwenden Sie Shell-Variablen und die Variablenerweiterung, um Befehle auszuführen, und legen eine Umgebungsvariable fest, um den Standardeditor für neue Shells anzupassen.

### Ergebnisse:

Es werden folgende Fähigkeiten vermittelt:

- Bearbeiten des Benutzerprofils
- Erstellen einer lokalen Variable
- Erstellen einer Umgebungsvariable

### Bevor Sie Beginnen

Melden Sie sich als Benutzer **student** mit dem Passwort **student** bei **workstation** an.

Führen Sie auf **workstation** den Befehl **lab edit-shell start** aus. Dieses Skript überprüft, ob der Zielserver ausgeführt wird.

```
[student@workstation ~]$ lab edit-shell start
```

- 1. Ändern Sie die Shell-Variable **PS1** des Benutzers „student“ in **[\u@\h \t \w]\$** (denken Sie daran, den Wert von PS1 in Anführungszeichen zu setzen und nach dem Dollarzeichen ein Leerzeichen einzugeben). Damit wird der Eingabeaufforderung die Zeit hinzugefügt.

- 1.1. Melden Sie sich auf **workstation** mit dem Befehl **ssh** bei **servera** an.

```
[student@workstation ~]$ ssh student@servera  
...output omitted...  
[student@servera ~]$
```

- 1.2. Verwenden Sie Vim zum Bearbeiten der Konfigurationsdatei **~/.bashrc**.

```
[student@servera ~]$ vim ~/.bashrc
```

- 1.3. Fügen Sie der Datei **~/.bashrc** die Shell-Variable **PS1** und deren Wert hinzu. Denken Sie daran, ein Leerzeichen am Ende des von Ihnen festgelegten Wertes einzufügen und den gesamten Wert in Anführungszeichen zu setzen, einschließlich des nachgestellten Leerzeichens.

```
...output omitted...
# User specific environment and startup programs
PATH="$HOME/.local/bin:$HOME/bin:$PATH"
PS1='[\u@\h \t \w]$ '
export PATH
```

- 1.4. Melden Sie sich von **servera** ab und melden Sie sich erneut mit dem Befehl **ssh** an, um die Eingabeaufforderung zu aktualisieren.

```
[student@servera ~]$ exit
logout
Connection to servera closed.
[student@workstation ~]$ ssh student@servera
...output omitted...
[student@servera 14:45:05 ~]$
```

- 2. Weisen Sie einer lokalen Shell-Variablen einen Wert zu. Variablennamen können Groß- oder Kleinbuchstaben, Ziffern und den Unterstrich enthalten. Rufen Sie den Variablenwert ab.

- 2.1. Erstellen Sie eine neue Variable mit dem Namen **file** und dem Wert **tmp.zdkei083**. Die Datei **tmp.zdkei083** ist im Benutzerverzeichnis **student** vorhanden.

```
[student@servera 14:47:05 ~]$ file=tmp.zdkei083
```

- 2.2. Rufen Sie den Wert der Variable **file** ab.

```
[student@servera 14:48:35 ~]$ echo $file
tmp.zdkei083
```

- 2.3. Verwenden Sie den Variablenamen **file** und den Befehl **ls -l** zum Auflisten der Datei **tmp.zdkei083**. Löschen Sie mit dem Befehl **rm** und dem Variablenamen **file** die Datei **tmp.zdkei083**. Überprüfen Sie, ob die Datei gelöscht wurde.

```
[student@servera 14:59:07 ~]$ ls -l $file
-rw-rw-r-- 1 student student 0 Jan 23 14:59 tmp.zdkei083
[student@servera 14:59:10 ~]$ rm $file
[student@servera 14:59:15 ~]$ ls -l $file
ls: cannot access 'tmp.zdkei083': No such file or directory
```

- 3. Weisen Sie der Variable **editor** einen Wert zu. Verwenden Sie einen Befehl, um die Variable in eine Umgebungsvariable zu ändern.

```
[student@servera 14:46:40 ~]$ export EDITOR=vim
[student@servera 14:46:55 ~]$ echo $EDITOR
vim
```

- 4. Beenden Sie **servera**.

```
[student@servera 14:47:11 ~]$ exit  
logout  
Connection to servera closed.  
[student@workstation ~]$
```

## Beenden

Führen Sie auf **workstation** das Skript **lab edit-shell finish** aus, um diese Übung zu beenden.

```
[student@workstation ~]$ lab edit-shell finish
```

Hiermit ist die angeleitete Übung beendet.

## ► Praktische Übung

# Erstellen, Anzeigen und Bearbeiten von Textdateien

### Leistungscheckliste

In dieser Übung bearbeiten Sie eine Textdatei mit dem **vim**-Editor.

### Ergebnisse

Es werden folgende Fähigkeiten vermittelt:

- Verwenden von Vim zur Dateibearbeitung
- Verwenden des visuellen Modus zur Vereinfachung der Dateibearbeitung

### Bevor Sie Beginnen

Melden Sie sich als Benutzer **student** mit dem Passwort **student** bei **workstation** an.

Führen Sie auf **workstation** den Befehl **lab edit-review start** aus.

```
[student@workstation ~]$ lab edit-review start
```

1. Leiten Sie eine ausführliche Auflistung aller Inhalte im Benutzerverzeichnis von „student“, einschließlich der versteckten Verzeichnisse und Dateien, in eine Datei namens **editing\_final\_lab.txt** um.
  2. Bearbeiten Sie die Datei mit Vim.
  3. Entfernen Sie die ersten drei Zeilen. Wechseln Sie mit dem großgeschriebenen **V** in den zeilenbasierten visuellen Modus.
  4. Entfernen Sie Spalten in der ersten Zeile. Wechseln Sie mit einem kleingeschriebenen **v** in den visuellen Modus. Das kleingeschriebene **v** wählt nur Zeichen in einer einzelnen Zeile aus. Die Spalten nach **-rw-** sollten gelöscht werden.
  5. Entfernen Sie die Spalten und den nachfolgenden Punkt (.) in den restlichen Zeilen. Verwenden Sie den visuellen Blockmodus. Wechseln Sie mit der Steuersequenz **Strg+v** in den visuellen Blockmodus. Verwenden Sie diese Tastenfolge, um einen Zeichenblock in mehreren Zeilen auszuwählen. Die Spalten nach **-rw-** sollten gelöscht werden.
  6. Verwenden Sie den visuellen Blockmodus, um die vierte Spalte zu entfernen.
  7. Entfernen Sie die Spalte „time“, aber belassen Sie „month“ und „day“ in allen Zeilen.
  8. Entfernen Sie die Zeilen **Desktop** und **Public**. Wechseln Sie mit dem großgeschriebenen **V** in den visuellen Modus.
  9. Verwenden Sie den Befehl **:wq** zum Speichern und Beenden der Datei. Legen Sie eine Sicherungskopie an, indem Sie das Datum (in Sekunden) verwenden, um einen eindeutigen Dateinamen zu erstellen.
- Der folgende **copy**-Befehl ist sehr lang und sollte als eine einzelne Zeile eingegeben werden.

## Kapitel 5 | Erstellen, Anzeigen und Bearbeiten von Textdateien

10. Hängen Sie eine gestrichelte Linie an die Datei an. Die gestrichelte Linie sollte mindestens aus 12 Strichen bestehen.  
Der folgende **echo**-Befehl ist sehr lang und sollte als eine einzelne Zeile eingegeben werden.
11. Hängen Sie eine Verzeichnisaufstellung an das Verzeichnis **Documents** an. Listen Sie die Verzeichnisaufstellung auf dem Terminal auf und senden Sie die Liste mit einer Befehlszeile an die Datei **editing\_final\_lab.txt**.
12. Überprüfen Sie, ob sich die Verzeichnisaufstellung am Ende der Übungsdatei befindet.

## Bewertung

Führen Sie auf **workstation** den Befehl **lab edit-review grade** aus, um den Erfolg dieser Übung zu bestätigen.

```
[student@workstation ~]$ lab edit-review grade
```

## Beenden

Führen Sie auf **workstation** das Skript **lab edit-review finish** aus, um diese Übung zu beenden.

```
[student@workstation ~]$ lab edit-review finish
```

Damit ist die praktische Übung abgeschlossen.

## ► Lösung

# Erstellen, Anzeigen und Bearbeiten von Textdateien

### Leistungscheckliste

In dieser Übung bearbeiten Sie eine Textdatei mit dem **vim**-Editor.

### Ergebnisse

Es werden folgende Fähigkeiten vermittelt:

- Verwenden von Vim zur Dateibearbeitung
- Verwenden des visuellen Modus zur Vereinfachung der Dateibearbeitung

### Bevor Sie Beginnen

Melden Sie sich als Benutzer **student** mit dem Passwort **student** bei **workstation** an.

Führen Sie auf **workstation** den Befehl **lab edit-review start** aus.

```
[student@workstation ~]$ lab edit-review start
```

1. Leiten Sie eine ausführliche Auflistung aller Inhalte im Benutzerverzeichnis von „student“, einschließlich der versteckten Verzeichnisse und Dateien, in eine Datei namens **editing\_final\_lab.txt** um.



#### Anmerkung

Die Ausgabe stimmt möglicherweise nicht genau mit den gezeigten Beispielen überein.

Leiten Sie auf „workstation“ aus dem Benutzerverzeichnis **student** mit dem Befehl **ls -al** eine lange Auflistung des gesamten Inhalts in eine Datei mit dem Namen **editing\_final\_lab.txt** um.

```
[student@workstation ~]$ ls -al > editing_final_lab.txt
```

2. Bearbeiten Sie die Datei mit Vim.

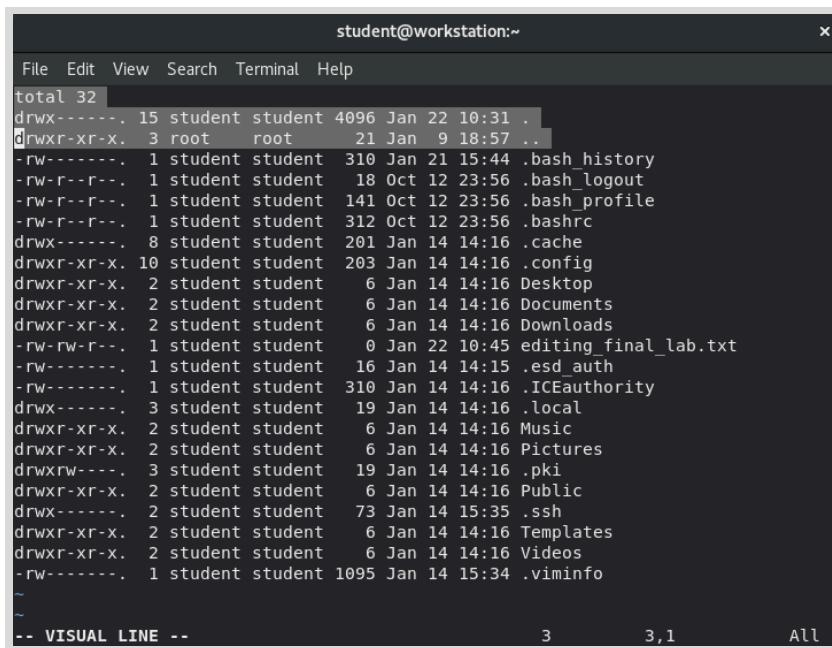
```
[student@workstation ~]$ vim editing_final_lab.txt
```

3. Entfernen Sie die ersten drei Zeilen. Wechseln Sie mit dem großgeschriebenen **V** in den zeilenbasierten visuellen Modus.

Verwenden Sie die Pfeiltasten, um den Cursor am ersten Zeichen in der ersten Zeile zu positionieren. Wechseln Sie mit **Umschalt+V** in den zeilenbasierten visuellen Modus.

**Kapitel 5 |** Erstellen, Anzeigen und Bearbeiten von Textdateien

Drücken Sie zweimal die Taste „Pfeil nach unten“, um nach unten zu scrollen und die ersten drei Zeilen auszuwählen. Löschen Sie die Zeilen mit **x**.



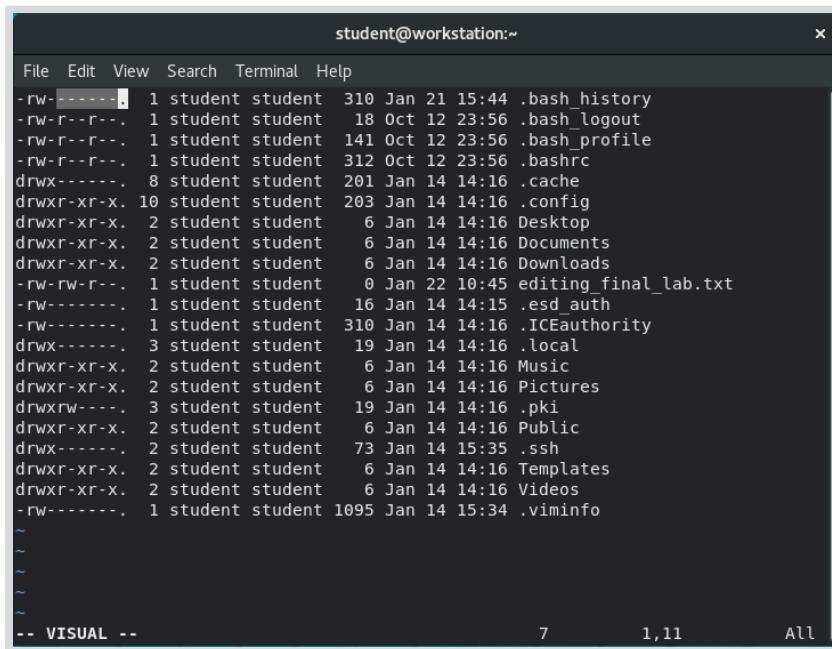
```
student@workstation:~$ ls
total 32
drwx----- 15 student student 4096 Jan 22 10:31 .
drwxr-xr-x  3 root   root    21 Jan  9 18:57 ..
-rw-r--r--  1 student student  310 Jan 21 15:44 .bash_history
-rw-r--r--  1 student student  18 Oct 12 23:56 .bash_logout
-rw-r--r--  1 student student 141 Oct 12 23:56 .bash_profile
-rw-r--r--  1 student student 312 Oct 12 23:56 .bashrc
drwx-----  8 student student 201 Jan 14 14:16 .cache
drwxr-xr-x 10 student student 203 Jan 14 14:16 .config
drwxr-xr-x  2 student student   6 Jan 14 14:16 Desktop
drwxr-xr-x  2 student student   6 Jan 14 14:16 Documents
drwxr-xr-x  2 student student   6 Jan 14 14:16 Downloads
-rw-rw-r--  1 student student   0 Jan 22 10:45 editing_final_lab.txt
-rw-----  1 student student  16 Jan 14 14:15 .esd_auth
-rw-----  1 student student 310 Jan 14 14:16 .ICEauthority
drwx-----  3 student student  19 Jan 14 14:16 .local
drwxr-xr-x  2 student student   6 Jan 14 14:16 Music
drwxr-xr-x  2 student student   6 Jan 14 14:16 Pictures
drwxrw----  3 student student  19 Jan 14 14:16 .pki
drwxr-xr-x  2 student student   6 Jan 14 14:16 Public
drwx-----  2 student student  73 Jan 14 15:35 .ssh
drwxr-xr-x  2 student student   6 Jan 14 14:16 Templates
drwxr-xr-x  2 student student   6 Jan 14 14:16 Videos
-rw-----  1 student student 1095 Jan 14 15:34 .viminfo
~
~
```

-- VISUAL LINE --

3            3,1            All

- 4.** Entfernen Sie Spalten in der ersten Zeile. Wechseln Sie mit einem kleingeschriebenen **v** in den visuellen Modus. Das kleingeschriebene **v** wählt nur Zeichen in einer einzelnen Zeile aus. Die Spalten nach **-rw-** sollten gelöscht werden.

Positionieren Sie den Cursor mithilfe der Pfeiltasten am ersten Zeichen. Wechseln Sie mit einem kleingeschriebenen **v** in den visuellen Modus. Positionieren Sie den Cursor mithilfe der Pfeiltasten am letzten Zeichen. Löschen Sie die Auswahl mit **x**.



```
student@workstation:~$ ls
-rw-----  1 student student 310 Jan 21 15:44 .bash_history
-rw-r--r--  1 student student 18 Oct 12 23:56 .bash_logout
-rw-r--r--  1 student student 141 Oct 12 23:56 .bash_profile
drwx-----  8 student student 201 Jan 14 14:16 .cache
drwxr-xr-x 10 student student 203 Jan 14 14:16 .config
drwxr-xr-x  2 student student   6 Jan 14 14:16 Desktop
drwxr-xr-x  2 student student   6 Jan 14 14:16 Documents
drwxr-xr-x  2 student student   6 Jan 14 14:16 Downloads
-rw-rw-r--  1 student student  0 Jan 22 10:45 editing_final_lab.txt
-rw-----  1 student student 16 Jan 14 14:15 .esd_auth
-rw-----  1 student student 310 Jan 14 14:16 .ICEauthority
drwx-----  3 student student  19 Jan 14 14:16 .local
drwxr-xr-x  2 student student   6 Jan 14 14:16 Music
drwxr-xr-x  2 student student   6 Jan 14 14:16 Pictures
drwxrw----  3 student student  19 Jan 14 14:16 .pki
drwxr-xr-x  2 student student   6 Jan 14 14:16 Public
drwx-----  2 student student  73 Jan 14 15:35 .ssh
drwxr-xr-x  2 student student   6 Jan 14 14:16 Templates
drwxr-xr-x  2 student student   6 Jan 14 14:16 Videos
-rw-----  1 student student 1095 Jan 14 15:34 .viminfo
~
~
```

-- VISUAL --

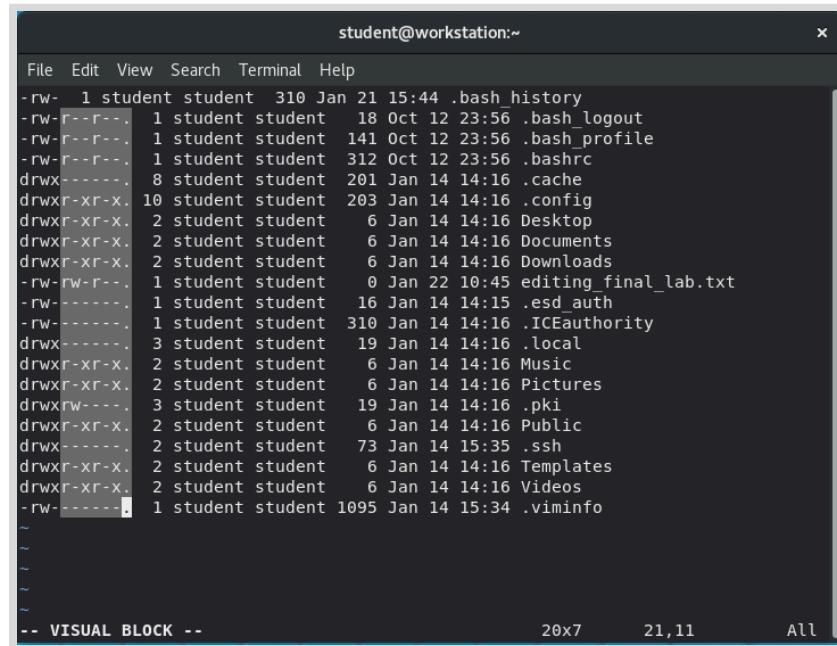
7            1,11            All

- 5.** Entfernen Sie die Spalten und den nachfolgenden Punkt (.) in den restlichen Zeilen. Verwenden Sie den visuellen Blockmodus. Wechseln Sie mit der Steuersequenz **Strg+V**

## Kapitel 5 | Erstellen, Anzeigen und Bearbeiten von Textdateien

in den visuellen Blockmodus. Verwenden Sie diese Tastenfolge, um einen Zeichenblock in mehreren Zeilen auszuwählen. Die Spalten nach **-rw-** sollten gelöscht werden.

Positionieren Sie den Cursor mithilfe der Pfeiltasten am ersten Zeichen. Wechseln Sie mit der Steuersequenz **Strg+V** in den visuellen Modus. Verwenden Sie die Pfeiltasten, um den Cursor am letzten Zeichen der Spalte in der letzten Zeile zu positionieren. Löschen Sie die Auswahl mit **X**.



```
student@workstation:~
```

```
File Edit View Search Terminal Help
```

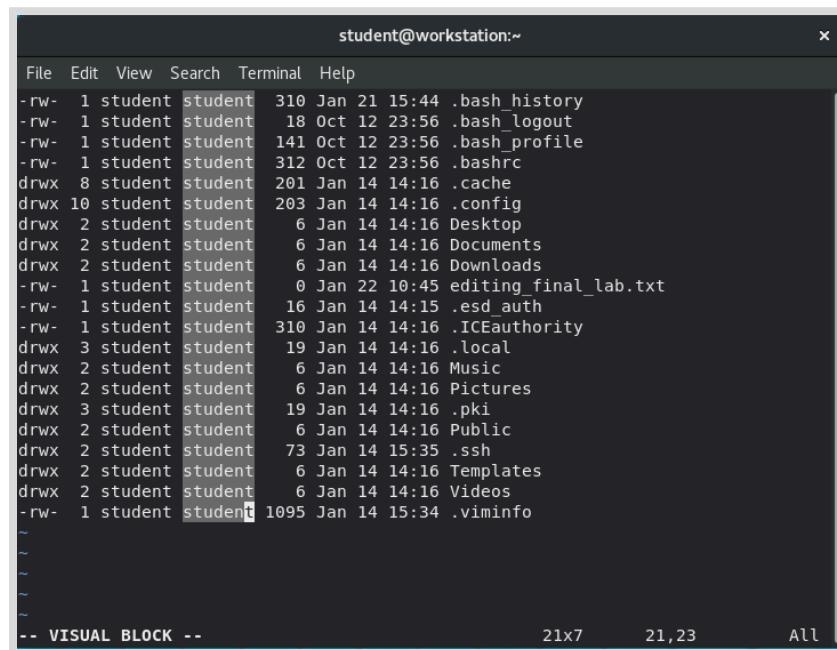
```
-rw- 1 student student 310 Jan 21 15:44 .bash_history
-rw- 1 student student 18 Oct 12 23:56 .bash_logout
-rw- 1 student student 141 Oct 12 23:56 .bash_profile
-rw- 1 student student 312 Oct 12 23:56 .bashrc
drwx 8 student student 201 Jan 14 14:16 .cache
drwxr-xr-x 10 student student 203 Jan 14 14:16 .config
drwxr-xr-x 2 student student 6 Jan 14 14:16 Desktop
drwxr-xr-x 2 student student 6 Jan 14 14:16 Documents
drwxr-xr-x 2 student student 6 Jan 14 14:16 Downloads
-rw-rw-r-- 1 student student 0 Jan 22 10:45 editing_final_lab.txt
-rw----- 1 student student 16 Jan 14 14:15 .esd_auth
-rw----- 1 student student 310 Jan 14 14:16 .ICEauthority
drwx----- 3 student student 19 Jan 14 14:16 .local
drwxr-xr-x 2 student student 6 Jan 14 14:16 Music
drwxr-xr-x 2 student student 6 Jan 14 14:16 Pictures
drwxrwm--- 3 student student 19 Jan 14 14:16 .pki
drwxr-xr-x 2 student student 6 Jan 14 14:16 Public
drwx----- 2 student student 73 Jan 14 15:35 .ssh
drwxr-xr-x 2 student student 6 Jan 14 14:16 Templates
drwxr-xr-x 2 student student 6 Jan 14 14:16 Videos
-rw----- 1 student student 1095 Jan 14 15:34 .viminfo
```

```
-- VISUAL BLOCK --
```

```
20x7 21,11 All
```

6. Verwenden Sie den visuellen Blockmodus, um die vierte Spalte zu entfernen.

Positionieren Sie den Cursor mit den Pfeiltasten am ersten Zeichen der vierten Spalte. Rufen Sie den visuellen Blockmodus mit **Strg+V** auf. Positionieren Sie den Cursor mit den Pfeiltasten am letzten Zeichen in der letzten Zeile der vierten Spalte. Löschen Sie die Auswahl mit **X**.



```
student@workstation:~
```

```
File Edit View Search Terminal Help
```

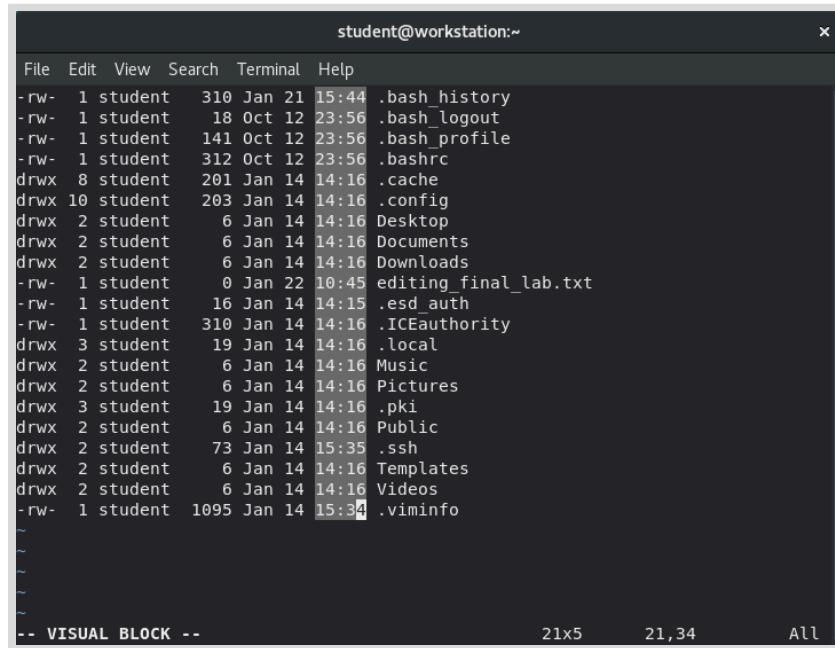
```
-rw- 1 student student 310 Jan 21 15:44 .bash_history
-rw- 1 student student 18 Oct 12 23:56 .bash_logout
-rw- 1 student student 141 Oct 12 23:56 .bash_profile
-rw- 1 student student 312 Oct 12 23:56 .bashrc
drwx 8 student student 201 Jan 14 14:16 .cache
drwxr-xr-x 10 student student 203 Jan 14 14:16 .config
drwxr-xr-x 2 student student 6 Jan 14 14:16 Desktop
drwxr-xr-x 2 student student 6 Jan 14 14:16 Documents
drwxr-xr-x 2 student student 6 Jan 14 14:16 Downloads
-rw-rw-r-- 1 student student 0 Jan 22 10:45 editing_final_lab.txt
-rw----- 1 student student 16 Jan 14 14:15 .esd_auth
-rw----- 1 student student 310 Jan 14 14:16 .ICEauthority
drwx----- 3 student student 19 Jan 14 14:16 .local
drwxr-xr-x 2 student student 6 Jan 14 14:16 Music
drwxr-xr-x 2 student student 6 Jan 14 14:16 Pictures
drwxrwm--- 3 student student 19 Jan 14 14:16 .pki
drwxr-xr-x 2 student student 6 Jan 14 14:16 Public
drwx----- 2 student student 73 Jan 14 15:35 .ssh
drwxr-xr-x 2 student student 6 Jan 14 14:16 Templates
drwxr-xr-x 2 student student 6 Jan 14 14:16 Videos
-rw----- 1 student student 1095 Jan 14 15:34 .viminfo
```

```
-- VISUAL BLOCK --
```

```
21x7 21,23 All
```

7. Entfernen Sie die Spalte „time“, aber belassen Sie „month“ und „day“ in allen Zeilen.

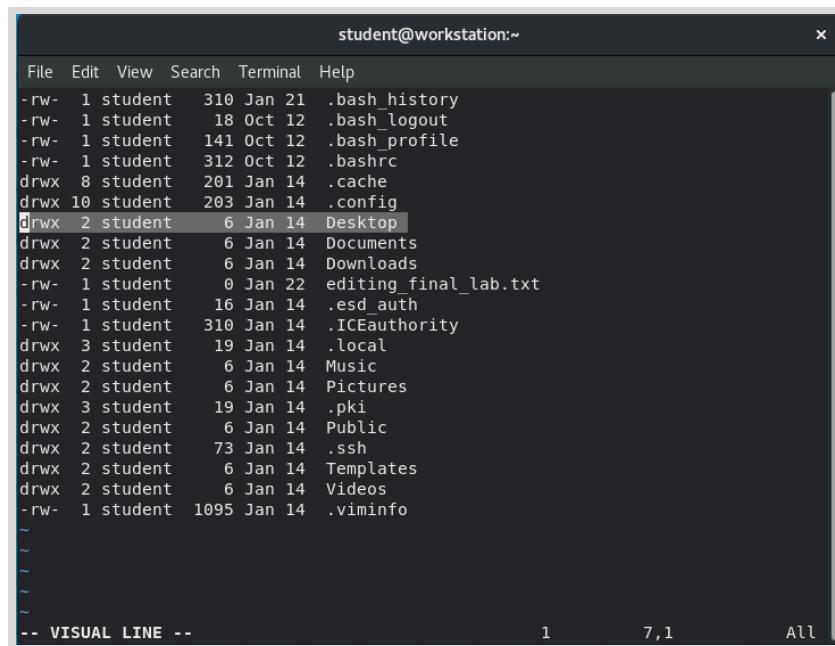
Positionieren Sie den Cursor mithilfe der Pfeiltasten am ersten Zeichen. Rufen Sie den visuellen Blockmodus mit **Strg+V** auf. Verwenden Sie die Pfeiltasten, um den Cursor am letzten Zeichen und der letzten Zeile der Spalte „time“ zu positionieren. Löschen Sie die Auswahl mit **x**.



```
student@workstation:~  
File Edit View Search Terminal Help  
-rw- 1 student 310 Jan 21 15:44 .bash_history  
-rw- 1 student 18 Oct 12 23:56 .bash_logout  
-rw- 1 student 141 Oct 12 23:56 .bash_profile  
-rw- 1 student 312 Oct 12 23:56 .bashrc  
drwx 8 student 201 Jan 14 14:16 .cache  
drwx 10 student 203 Jan 14 14:16 .config  
drwx 2 student 6 Jan 14 14:16 Desktop  
drwx 2 student 6 Jan 14 14:16 Documents  
drwx 2 student 6 Jan 14 14:16 Downloads  
-rw- 1 student 0 Jan 22 10:45 editing_final_lab.txt  
-rw- 1 student 16 Jan 14 14:15 .esd_auth  
-rw- 1 student 310 Jan 14 14:16 .ICEauthority  
drwx 3 student 19 Jan 14 14:16 .local  
drwx 2 student 6 Jan 14 14:16 Music  
drwx 2 student 6 Jan 14 14:16 Pictures  
drwx 3 student 19 Jan 14 14:16 .pki  
drwx 2 student 6 Jan 14 14:16 Public  
drwx 2 student 73 Jan 14 15:35 .ssh  
drwx 2 student 6 Jan 14 14:16 Templates  
drwx 2 student 6 Jan 14 14:16 Videos  
-rw- 1 student 1095 Jan 14 15:34 .viminfo  
~  
~  
~  
~  
~  
-- VISUAL BLOCK -- 21x5 21,34 All
```

8. Entfernen Sie die Zeilen **Desktop** und **Public**. Wechseln Sie mit dem großgeschriebenen **V** in den visuellen Modus.

Verwenden Sie die Pfeiltasten, um den Cursor an einem beliebigen Zeichen der Zeile **Desktop** zu positionieren. Wechseln Sie mit dem großgeschriebenen **V** in den visuellen Modus. Die vollständige Zeile wird ausgewählt. Löschen Sie die Auswahl mit **x**. Wiederholen Sie den Vorgang für die Zeile **Public**.



```
student@workstation:~  
File Edit View Search Terminal Help  
-rw- 1 student 310 Jan 21 .bash_history  
-rw- 1 student 18 Oct 12 .bash_logout  
-rw- 1 student 141 Oct 12 .bash_profile  
-rw- 1 student 312 Oct 12 .bashrc  
drwx 8 student 201 Jan 14 .cache  
drwx 10 student 203 Jan 14 .config  
drwx 2 student 6 Jan 14 Desktop  
drwx 2 student 6 Jan 14 Documents  
drwx 2 student 6 Jan 14 Downloads  
-rw- 1 student 0 Jan 22 editing_final_lab.txt  
-rw- 1 student 16 Jan 14 .esd_auth  
-rw- 1 student 310 Jan 14 .ICEauthority  
drwx 3 student 19 Jan 14 .local  
drwx 2 student 6 Jan 14 Music  
drwx 2 student 6 Jan 14 Pictures  
drwx 3 student 19 Jan 14 .pki  
drwx 2 student 6 Jan 14 Public  
drwx 2 student 73 Jan 14 .ssh  
drwx 2 student 6 Jan 14 Templates  
drwx 2 student 6 Jan 14 Videos  
-rw- 1 student 1095 Jan 14 .viminfo  
~  
~  
~  
~  
~  
-- VISUAL LINE -- 1 7,1 All
```

9. Verwenden Sie den Befehl :wq zum Speichern und Beenden der Datei. Legen Sie eine Sicherungskopie an, indem Sie das Datum (in Sekunden) verwenden, um einen eindeutigen Dateinamen zu erstellen.

A screenshot of a terminal window titled "student@workstation:~". The window shows a list of files in the current directory (~). The files include .bash\_history, .bash\_logout, .bash\_profile, .bashrc, .cache, .config, Documents, Downloads, editing\_final\_lab.txt, .esd\_auth, .ICEauthority, .local, Music, Pictures, .pki, .ssh, Templates, Videos, and .viminfo. At the bottom of the terminal window, the command ":wq" is visible in the input field.

Der folgende **copy**-Befehl ist sehr lang und sollte als eine einzelne Zeile eingegeben werden.

```
[student@workstation ~]$ cp editing_final_lab.txt  
editing_final_lab_$(date +%s).txt
```

10. Hängen Sie eine gestrichelte Linie an die Datei an. Die gestrichelte Linie sollte mindestens aus 12 Strichen bestehen.

Der folgende **echo**-Befehl ist sehr lang und sollte als eine einzelne Zeile eingegeben werden.

```
[student@workstation ~]$ echo "-----"  
-> editing_final_lab.txt
```

11. Hängen Sie eine Verzeichnisaufstellung an das Verzeichnis **Documents** an. Listen Sie die Verzeichnisaufstellung auf dem Terminal auf und senden Sie die Liste mit einer Befehlszeile an die Datei **editing\_final\_lab.txt**.

```
[student@workstation ~]$ ls Documents/ | tee -a editing_final_lab.txt  
lab_review.txt
```

12. Überprüfen Sie, ob sich die Verzeichnisaufstellung am Ende der Übungsdatei befindet.

```
[student@workstation ~]$ cat editing_final_lab.txt  
-rw- 1 student 310 Jan 21 .bash_history  
-rw- 1 student 18 Oct 12 .bash_logout  
-rw- 1 student 141 Oct 12 .bash_profile  
-rw- 1 student 312 Oct 12 .bashrc  
drwx 8 student 201 Jan 14 .cache  
drwx 10 student 203 Jan 14 .config  
drwx 2 student 6 Jan 14 Documents
```

## Kapitel 5 | Erstellen, Anzeigen und Bearbeiten von Textdateien

```
drwx  2 student   6 Jan 14  Downloads
-rw-  1 student   0 Jan 22  editing_final_lab.txt
-rw-  1 student  16 Jan 14  .esd_auth
-rw-  1 student 310 Jan 14  .ICEauthority
drwx  3 student  19 Jan 14  .local
drwx  2 student  6 Jan 14  Music
drwx  2 student  6 Jan 14  Pictures
drwx  3 student  19 Jan 14  .pki
drwx  2 student  73 Jan 14  .ssh
drwx  2 student  6 Jan 14  Templates
drwx  2 student  6 Jan 14  Videos
-rw-  1 student 1095 Jan 14  .viminfo
-----
lab_review.txt
```

## Bewertung

Führen Sie auf **workstation** den Befehl **lab edit-review grade** aus, um den Erfolg dieser Übung zu bestätigen.

```
[student@workstation ~]$ lab edit-review grade
```

## Beenden

Führen Sie auf **workstation** das Skript **lab edit-review finish** aus, um diese Übung zu beenden.

```
[student@workstation ~]$ lab edit-review finish
```

Damit ist die praktische Übung abgeschlossen.

# Zusammenfassung

---

In diesem Kapitel erhielten Sie folgende Informationen:

- Ausgeführte Programme oder Prozesse haben drei Standardkommunikationskanäle: Standardeingabe, Standardausgabe und Standardfehler.
- Sie können mit der I/O-Umleitung die Standardeingabe von einer Datei lesen oder die Ausgabe oder Fehler aus einem Prozess in eine Datei schreiben.
- Mit Pipelines kann die Standardausgabe eines Prozesses mit der Standardeingabe eines anderen Prozesses verbunden und die Ausgabe formatiert oder komplexe Befehle erstellt werden.
- Sie sollten wissen, wie mindestens ein Befehlszeilen-Texteditor verwendet wird, und Vim ist in der Regel installiert.
- Shell-Variablen unterstützen die Ausführung von Befehlen und sind für eine bestimmte Shell-Sitzung eindeutig.
- Umgebungsvariablen können Ihnen dabei helfen, das Verhalten der Shell oder der von ihr gestarteten Prozesse zu konfigurieren.



## Kapitel 6

# Verwalten lokaler Benutzer und Gruppen

### Ziel

Erstellen, Verwalten und Löschen lokaler Benutzer und Gruppen und Verwalten lokaler Passwortrichtlinien

### Ziele

- Beschreiben des Zwecks von Benutzern und Gruppen auf einem Linux-System
- Wechseln zum Superuser-Konto, um ein Linux-System zu verwalten, und anderen Benutzern mit dem Befehl **sudo** Zugriff als Superuser zu gewähren
- Erstellen, Ändern und Löschen lokal definierter Benutzerkonten
- Erstellen, Ändern und Löschen lokal definierter Gruppenkonten
- Festlegen einer Passwortverwaltungsrichtlinie für Benutzer und manuelles Sperren und Entsperren von Benutzerkonten

### Abschnitte

- Beschreiben von Benutzer- und Gruppenkonzepten (und Test)
- Erhalten von Superuser-Zugriff (und angeleitete Übung)
- Verwalten lokaler Benutzerkonten (und angeleitete Übung)
- Verwalten lokaler Gruppenkonten (und angeleitete Übung)
- Verwalten von Benutzerpasswörtern (und angeleitete Übung)

### Praktische Übung

Verwalten lokaler Linux-Benutzer und -Gruppen

# Beschreiben von Benutzer- und Gruppenkonzepten

## Ziele

Nach Abschluss dieses Abschnitts sollten Sie in der Lage sein, den Zweck von Benutzern und Gruppen auf einem Linux-System zu beschreiben.

## Was ist ein Benutzer?

Ein *Benutzerkonto* wird verwendet, um Sicherheitsgrenzen zwischen verschiedenen Personen und Programmen festzulegen, die Befehle ausführen können.

Benutzer haben *Benutzernamen*, damit sie für menschliche Benutzer identifiziert werden können und einfacher mit ihnen gearbeitet werden kann. Intern unterscheidet das System Benutzerkonten anhand der ihnen zugewiesenen eindeutigen Identifikationsnummer, der *Benutzer-ID* oder *UID*. Einem von Personen verwendeten Benutzerkonto wird in der Regel ein geheimes *Passwort* zugewiesen, mit dem der Benutzer bei der Anmeldung beweist, dass er der autorisierte Benutzer ist.

Benutzerkonten haben für die Systemsicherheit eine grundlegende Bedeutung. Jeder Prozess (ausgeführtes Programm) auf dem System wird als ein bestimmter Benutzer ausgeführt. Jede Datei hat einen bestimmten Benutzer als Eigentümer. Aufgrund der Zuordnung von Dateien zu Eigentümern kann das System den Zugriff für Benutzer der Dateien steuern. Der zu einem ausgeführten Prozess gehörige Benutzer bestimmt, welche Dateien und Verzeichnisse dem Prozess zur Verfügung stehen.

Es gibt drei Haupttypen von Benutzerkonten: *Superuser*, *Systembenutzer* und *regulärer Benutzer*.

- Das Benutzerkonto *Superuser* ist für die Verwaltung des Systems vorgesehen. Der Name des Superuser lautet **root** und das Benutzerkonto hat die UID 0. Der Superuser hat vollständigen Zugriff auf das System.
- Das System verfügt über *Systembenutzer*-Konten, die von Prozessen verwendet werden, die unterstützende Services bereitstellen. Diese Prozesse oder *Daemons* müssen normalerweise nicht als Superuser ausgeführt werden. Sie sind unprivilegierten Benutzerkonten zugeordnet, mit denen sie ihre Dateien und anderen Ressourcen voreinander und vor regulären Benutzern im System schützen können. Benutzer melden sich nicht interaktiv mit einem Systembenutzerkonto an.
- Die meisten Benutzer haben *reguläre Benutzerkonten*, die sie für ihre tägliche Arbeit verwenden. Wie Systembenutzer haben reguläre Benutzer nur eingeschränkten Zugriff auf das System.

Mit dem Befehl **id** können Sie Informationen zum aktuell angemeldeten Benutzer anzeigen.

```
[user01@host ~]$ id  
uid=1000(user01) gid=1000(user01) groups=1000(user01)  
context=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023
```

Um grundlegende Informationen über einen anderen Benutzer anzuzeigen, übergeben Sie den Benutzernamen als Argument an den Befehl **id**.

```
[user01@host]$ id user02
uid=1002(user02) gid=1001(user02) groups=1001(user02)
context=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023
```

Der Eigentümer einer Datei wird mit dem Befehl **ls -l** angezeigt. Der Eigentümer eines Verzeichnisses wird mit dem Befehl **ls -ld** angezeigt. In der folgenden Ausgabe enthält die dritte Spalte den Benutzernamen.

```
[user01@host ~]$ ls -l file1
-rw-rw-r--. 1 user01 user01 0 Feb 5 11:10 file1
[user01@host]$ ls -ld dir1
drwxrwxr-x. 2 user01 user01 6 Feb 5 11:10 dir1
```

Verwenden Sie den Befehl **ps**, um Prozessinformationen anzuzeigen. Standardmäßig werden nur Prozesse in der aktuellen Shell angezeigt. Fügen Sie die Option **a** hinzu, um alle Prozesse eines Terminals anzuzeigen. Fügen Sie dem Befehl die Option **u** hinzu, um den mit einem Prozess verknüpften Benutzer anzuzeigen. In der folgenden Ausgabe enthält die erste Spalte den Benutzernamen.

```
[user01@host]$ ps -au
USER      PID %CPU %MEM    VSZ   RSS TTY      STAT START   TIME COMMAND
root      777  0.0  0.0 225752 1496 tty1      Ss+  11:03   0:00 /sbin/agetty -o -
p -- \u --noclear tty1 linux
root      780  0.0  0.1 225392 2064 ttyS0     Ss+  11:03   0:00 /sbin/agetty -o -
p -- \u --keep-baud 115200,38400,9600
user01    1207  0.0  0.2 234044 5104 pts/0     Ss   11:09   0:00 -bash
user01    1319  0.0  0.2 266904 3876 pts/0     R+   11:33   0:00 ps au
```

Durch die vorhergehenden Befehle werden die Namen der Benutzer angezeigt, das Betriebssystem verfolgt Benutzer jedoch anhand von UIDs. Die Zuordnung von Benutzernamen zu UIDs ist in einer Datenbank mit den Kontoinformationen enthalten. Standardmäßig wird die Datei **/etc/passwd** zum Speichern von Informationen zu lokalen Benutzern verwendet.

Jede Zeile in der Datei **/etc/passwd** enthält Informationen zu einem Benutzer. Die Datei ist in bis zu sieben, durch Doppelpunkt getrennte Felder unterteilt. Hier ist ein Beispiel für eine Zeile aus **/etc/passwd**:

```
① user01:②x:③1000:④1000:⑤User One:⑥/home/user01:⑦/bin/bash
```

- ① Benutzername für diesen Benutzer (**user01**).
- ② Das Passwort des Benutzers wurde hier in verschlüsselter Form gespeichert. Das Passwort wurde in die Datei **/etc/shadow** verschoben, die später behandelt wird. Dieses Feld sollte immer **x** enthalten.
- ③ Die UID-Nummer für dieses Benutzerkonto (**1000**).
- ④ Die GID-Nummer für die primäre Gruppe dieses Benutzerkontos (**1000**). Gruppen werden später in diesem Abschnitt behandelt.
- ⑤ Der reale Name für diesen Benutzer (**User One**).
- ⑥ Das Benutzerverzeichnis für diesen Benutzer (**/home/user01**). Das ist das erste Arbeitsverzeichnis beim Start der Shell. Es enthält die Daten und Konfigurationseinstellungen des Benutzers.
- ⑦ Das Standard-Shell-Programm für diesen Benutzer, das bei der Anmeldung ausgeführt wird (**/bin/bash**). Bei einem regulären Benutzer ist dies im Regelfall das Programm, das die

Befehlseingabeaufforderung des Benutzers bereitstellt. Ein Systembenutzer könnte **/sbin/nologin** verwenden, wenn interaktive Anmeldungen für diesen Benutzer nicht zulässig sind.

## Was ist eine Gruppe?

Eine Gruppe ist eine Sammlung von Benutzern, die den Zugriff auf Dateien und andere Systemressourcen gemeinsam nutzen müssen. Mit Gruppen kann mehreren Benutzern anstelle eines einzelnen Benutzers Zugriff auf Dateien gewährt werden.

Wie Benutzer haben Gruppen *Gruppennamen*, um die Arbeit mit ihnen zu erleichtern. Intern unterscheidet das System Gruppen anhand der ihnen zugewiesenen eindeutigen Identifikationsnummer, der *Gruppen-ID* oder *GID*.

Die Zuordnung von Gruppennamen zu GIDs ist in einer Datenbank mit den Gruppenkontoinformationen enthalten. Standardmäßig wird die Datei **/etc/group** zum Speichern von Informationen zu lokalen Gruppen verwendet.

Jede Zeile in der Datei **/etc/group** enthält Informationen zu einer Gruppe. Jeder Gruppeneintrag ist in vier, durch Doppelpunkt getrennte Felder unterteilt. Hier ist ein Beispiel für eine Zeile aus **/etc/group**:

❶ group01:❷x:❸10000:❹user01,user02,user03

- ❶ Gruppenname für diese Gruppe (**group01**).
- ❷ Veraltetes Gruppenpasswortfeld. Dieses Feld sollte immer **x** enthalten.
- ❸ Die GID-Nummer für diese Gruppe (**10000**).
- ❹ Eine Liste der Benutzer, die Mitglieder dieser Gruppe als Zusatzgruppe sind (**user01, user02, user03**). Primäre (oder Standard-) und Zusatzgruppen werden später in diesem Abschnitt behandelt.

## Primäre Gruppen und Zusatzgruppen

Jeder Benutzer gehört genau einer primären Gruppe an. Für lokale Benutzer ist dies die Gruppe, die nach GID-Nummer in der Datei **/etc/passwd** aufgeführt ist. Standardmäßig ist dies die Gruppe, in der neue Dateien gespeichert werden, die vom Benutzer erstellt wurden.

Normalerweise wird beim Erstellen eines neuen regulären Benutzers eine neue Gruppe mit demselben Namen wie dieser Benutzer erstellt. Diese Gruppe wird als primäre Gruppe für den neuen Benutzer verwendet und dieser Benutzer ist das einzige Mitglied dieser *User Private Group*. Die Verwaltung von Dateiberechtigungen wird dadurch vereinfacht, worauf später in diesem Kurs noch eingegangen wird.

Benutzer können auch *Zusatzgruppen* haben. Die Mitgliedschaft in Zusatzgruppen wird von der Datei **/etc/group** bestimmt. Benutzer erhalten Zugriff auf Dateien, je nachdem, ob eine ihrer Gruppen Zugriff darauf hat. Es spielt keine Rolle, ob die Gruppe oder Gruppen, die Zugriff haben, für den Benutzer primäre oder Zusatzgruppen sind.

Wenn beispielsweise der Benutzer **user01** eine primäre Gruppe **user01** und Zusatzgruppen **wheel** und **webadmin** hat, kann dieser Benutzer Dateien lesen, die von einer dieser drei Gruppen gelesen werden können.

Mit dem Befehl **id** können auch Informationen zur Gruppenmitgliedschaft eines Benutzers gesucht werden.

```
[user03@host ~]$ id  
uid=1003(user03) gid=1003(user03) groups=1003(user03),10(wheel),10000(group01)  
context=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023
```

Im vorherigen Beispiel hat **user03** die Gruppe **user03** als primäre Gruppe (**gid**). Das Element **groups** listet alle Gruppen für diesen Benutzer auf und außer der primären Gruppe **user03** hat der Benutzer die Gruppen **wheel** und **group01** als Zusatzgruppen.



### Literaturhinweise

Manpages **id(1)**, **passwd(5)** und **group(5)**

**info libc** (*Referenzhandbuch für die GNU C Library*)

- Abschnitt 30: Benutzer und Gruppen

(Hinweis: Das Paket *glibc-devel* muss installiert sein, damit dieser Infoknoten verfügbar ist.)

## ► Quiz

# Beschreiben von Benutzer- und Gruppenkonzepten

Wählen Sie die richtige Antwort auf die folgenden Fragen aus:

- ▶ **1. Welches Element repräsentiert eine Nummer, die den Benutzer auf grundlegendster Ebene identifiziert?**
  - a. Primärer Benutzer
  - b. UID
  - c. GID
  - d. Benutzername
  
- ▶ **2. Welches Element repräsentiert das Programm, das die Befehlszeileneingabeaufforderung des Benutzers bereitstellt?**
  - a. Primäre Shell
  - b. Benutzerverzeichnis
  - c. Login-Shell
  - d. Befehlsname
  
- ▶ **3. Welches Element oder welche Datei stellt den Speicherort der lokalen Gruppeninformationen dar?**
  - a. Benutzerverzeichnis
  - b. **/etc/passwd**
  - c. **/etc/GID**
  - d. **/etc/group**
  
- ▶ **4. Welches Element oder welche Datei stellt den Speicherort der persönlichen Dateien des Benutzers dar?**
  - a. Benutzerverzeichnis
  - b. Login-Shell
  - c. **/etc/passwd**
  - d. **/etc/group**
  
- ▶ **5. Welches Element repräsentiert eine Nummer, die die Gruppe auf grundlegendster Ebene identifiziert?**
  - a. Primäre Gruppe
  - b. UID
  - c. GID
  - d. groupid

► **6. Welches Element oder welche Datei stellt den Speicherort der lokalen Benutzerkontoinformationen dar?**

- a. Benutzerverzeichnis
- b. **/etc/passwd**
- c. **/etc/UID**
- d. **/etc/group**

► **7. Was ist das vierte Feld der Datei /etc/passwd?**

- a. Benutzerverzeichnis
- b. UID
- c. Login-Shell
- d. Primäre Gruppe

## ► Lösung

# Beschreiben von Benutzer- und Gruppenkonzepten

Wählen Sie die richtige Antwort auf die folgenden Fragen aus:

- ▶ **1. Welches Element repräsentiert eine Nummer, die den Benutzer auf grundlegendster Ebene identifiziert?**
  - a. Primärer Benutzer
  - b. UID
  - c. GID
  - d. Benutzername
- ▶ **2. Welches Element repräsentiert das Programm, das die Befehlszeileneingabeaufforderung des Benutzers bereitstellt?**
  - a. Primäre Shell
  - b. Benutzerverzeichnis
  - c. Login-Shell
  - d. Befehlsname
- ▶ **3. Welches Element oder welche Datei stellt den Speicherort der lokalen Gruppeninformationen dar?**
  - a. Benutzerverzeichnis
  - b. /etc/passwd
  - c. /etc/GID
  - d. /etc/group
- ▶ **4. Welches Element oder welche Datei stellt den Speicherort der persönlichen Dateien des Benutzers dar?**
  - a. Benutzerverzeichnis
  - b. Login-Shell
  - c. /etc/passwd
  - d. /etc/group
- ▶ **5. Welches Element repräsentiert eine Nummer, die die Gruppe auf grundlegendster Ebene identifiziert?**
  - a. Primäre Gruppe
  - b. UID
  - c. GID
  - d. groupid

► **6. Welches Element oder welche Datei stellt den Speicherort der lokalen Benutzerkontoinformationen dar?**

- a. Benutzerverzeichnis
- b. **/etc/passwd**
- c. **/etc/UID**
- d. **/etc/group**

► **7. Was ist das vierte Feld der Datei /etc/passwd?**

- a. Benutzerverzeichnis
- b. UID
- c. Login-Shell
- d. Primäre Gruppe

# Zugriff als Superuser

## Ziele

Nach Abschluss dieses Abschnitts sind Sie in der Lage, zum Superuser-Konto zu wechseln, um ein Linux-System zu verwalten und anderen Benutzern mit dem Befehl **sudo** Zugriff als Superuser zu gewähren.

## Der Superuser

Bei den meisten Betriebssystemen gibt es eine Art *Superuser* – einen Benutzer, der das gesamte System steuern kann. In Red Hat Enterprise Linux das ist der **root**-Benutzer. Dieser Benutzer kann normale Berechtigungen im Dateisystem außer Kraft setzen und wird für die Systemverwaltung und -administration verwendet. Um Aufgaben wie die Installation oder Deinstallation von Software durchzuführen sowie Systemdateien und -verzeichnisse zu verwalten, müssen Benutzer die entsprechenden Berechtigungen an den Benutzer **root** übertragen.

Nur der **root**-Benutzer unter den regulären Benutzern kann die meisten Geräte steuern, aber es gibt einige Ausnahmen. Zum Beispiel können reguläre Benutzer Wechselmedien wie USB-Geräte steuern. Reguläre Benutzer dürfen auf einem Wechseldatenträger Dateien hinzufügen und entfernen sowie andere Verwaltungsaufgaben ausführen. Zur Verwaltung von „festen“ Festplattenlaufwerken ist jedoch standardmäßig nur **root** berechtigt.

Die uneingeschränkten Berechtigungen gehen allerdings mit einer größeren Verantwortung einher. Der **root**-Benutzer hat unbegrenzte Möglichkeiten, das System zu beschädigen, indem er Dateien und Verzeichnisse entfernt, Benutzerkonten löscht, Schwachstellen hinzufügt usw. Wenn das Konto des **root**-Benutzers kompromittiert ist, können Angreifer die administrative Kontrolle über das System übernehmen. In diesem Kurs wird Administratoren empfohlen, sich grundsätzlich als regulärer Benutzer anzumelden und nur bei Bedarf die Berechtigungen von **root** zu nutzen.

Das **root**-Konto auf Linux entspricht in etwa dem lokalen Administratorkonto unter Microsoft Windows. Unter Linux melden sich die meisten Systemadministratoren beim System als unprivilegierter Benutzer an und nutzen bestimmte Tools, um vorübergehend **root**-Berechtigungen zu erlangen.



### Warnung

In der Vergangenheit war es übliche Praxis auf Microsoft Windows, dass lokale **Administrator**-Benutzer sich direkt anmeldeten, um Aufgaben der Systemadministration durchzuführen. Dies ist zwar unter Linux möglich, Red Hat empfiehlt aber, dass sich Systemadministratoren nicht direkt als **root** anmelden. Stattdessen sollten sie sich als reguläre Benutzer anmelden und andere Mechanismen (z. B. **su**, **sudo** oder PolicyKit) nutzen, um vorübergehend Superuser-Berechtigungen zu erlangen.

Bei einer Anmeldung als Superuser wird die gesamte Desktop-Umgebung völlig unnötig mit Administratorrechten ausgeführt. In einer derartigen Situation hätte eine Sicherheitslücke, die normalerweise nur das Benutzerkonto kompromittieren würde, das Potenzial für einen systemweiten Missbrauch.

## Wechseln von Benutzern

Mit dem Befehl **su** können Benutzer in ein anderes Benutzerkonto wechseln. Wenn Sie **su** aus einem regulären Benutzerkonto ausführen, werden Sie aufgefordert, das Passwort des Benutzerkontos einzugeben, zu dem Sie wechseln möchten. Wenn **root su** ausführt, müssen Sie das Passwort des Benutzers nicht eingeben.

```
[user01@host ~]$ su - user02  
Password:  
[user02@host ~]$
```

Wenn Sie den Benutzernamen weglassen, versucht der Befehl **su** oder **su -** standardmäßig, zu **root** zu wechseln.

```
[user01@host ~]$ su -  
Password:  
[root@host ~]#
```

Mit dem Befehl **su** wird eine *Shell ohne Anmeldung* gestartet, mit dem Befehl **su -** (mit der Bindestrich-Option) hingegen wird eine *Login-Shell* gestartet. Der Hauptunterschied zwischen diesen beiden Befehlen ist, dass Sie mit **su -** die Shell-Umgebung so einrichten, als ob Sie sich mit diesem Benutzer normal angemeldet hätten, während **su** lediglich als dieser Benutzer eine Shell startet, jedoch die originalen Umgebungseinstellungen des Benutzers beibehält.

In den meisten Fällen sollten Administratoren **su -** ausführen, um eine Shell mit den normalen Umgebungseinstellungen des Zielbenutzers zu erhalten. Weitere Informationen finden Sie auf der Manpage **bash(1)**.



### Anmerkung

Der Befehl **su** wird meist verwendet, um eine Befehlszeilenschnittstelle (Shell-Eingabeaufforderung) aufzurufen, die als anderer Benutzer (üblicherweise **root**) ausgeführt wird. Mit der Option **-c** kann der Befehl jedoch auch wie das Windows-Dienstprogramm **runas** verwendet werden, um ein beliebiges Programm als anderer Benutzer auszuführen. Führen Sie **info su** aus, um weitere Details anzuzeigen.

## Ausführen von Befehlen mit Sudo

In einigen Fällen verfügt das Konto des **root**-Benutzers aus Sicherheitsgründen möglicherweise über kein gültiges Passwort. In diesem Fall können sich Benutzer beim System nicht direkt als **root** mit einem Passwort anmelden und **su** kann nicht verwendet werden, um eine interaktive Shell zu erhalten. Mit dem Tool **sudo** erhalten Sie in diesem Fall **root**-Zugriff.

Im Unterschied zu **su** müssen Benutzer bei **sudo** ihr eigenes Passwort für die Authentifizierung eingeben, nicht das Passwort des Benutzerkontos, auf das sie zugreifen möchten. Das heißt, Benutzer, die mit **sudo** Befehle als **root** ausführen, müssen das **root**-Passwort nicht kennen. Stattdessen verwenden sie ihre eigenen Passwörter, um den Zugriff zu authentifizieren.

Zusätzlich kann **sudo** so konfiguriert werden, dass bestimmte Benutzer jeden Befehl als anderer Benutzer oder nur bestimmte Befehle als dieser Benutzer ausführen können.

**Kapitel 6 |** Verwalten lokaler Benutzer und Gruppen

Wenn beispielsweise **sudo** so konfiguriert wurde, dass der Benutzer **user01** den Befehl **usermod** als **root** ausführen darf, könnte **user01** den folgenden Befehl ausführen, um ein Benutzerkonto zu sperren oder zu entsperren:

```
[user01@host ~]$ sudo usermod -L user02
[sudo] password for user01:
[user01@host ~]$ su - user02
Password:
su: Authentication failure
[user01@host ~]$
```

Wenn ein Benutzer versucht, einen Befehl als anderer Benutzer auszuführen, und die **sudo**-Konfiguration dies nicht zulässt, wird der Befehl blockiert, der Versuch wird protokolliert und es wird standardmäßig eine E-Mail an den **root**-Benutzer gesendet.

```
[user02@host ~]$ sudo tail /var/log/secure
[sudo] password for user02:
user02 is not in the sudoers file. This incident will be reported.
[user02@host ~]$
```

Ein zusätzlicher Vorteil bei der Verwendung von **sudo** liegt darin, dass alle ausgeführten Befehle standardmäßig in **/var/log/secure** protokolliert werden.

```
[user01@host ~]$ sudo tail /var/log/secure
...output omitted...
Feb 6 20:45:46 host sudo[2577]: user01 : TTY=pts/0 ; PWD=/home/user01 ;
USER=root ; COMMAND=/sbin/usermod -L user02
...output omitted...
```

In Red Hat Enterprise Linux 7 und Red Hat Enterprise Linux 8 können alle Mitglieder der Gruppe **wheel** mit **sudo** Befehle als beliebige Benutzer, einschließlich **root**, ausführen. Die Benutzer werden nach ihrem jeweils eigenen Passwort gefragt. Dies ist eine Änderung gegenüber Red Hat Enterprise Linux 6 und früheren Versionen. In diesen Versionen haben Benutzer, die Mitglieder der Gruppe **wheel** waren, diesen administrativen Zugriff standardmäßig nicht erhalten.

**Warnung**

Standardmäßig hat RHEL 6 der Gruppe **wheel** keine speziellen Berechtigungen erteilt. Sites, die diese Gruppe bislang genutzt haben, sind möglicherweise überrascht, wenn RHEL 7 und RHEL 8 automatisch allen Mitgliedern von **wheel** volle **sudo**-Berechtigungen einräumen. Auf diese Weise könnten unautorisierte Benutzer administrativen Zugriff auf RHEL 7- und RHEL 8-Systeme erlangen.

In der Vergangenheit haben UNIX-ähnliche Systeme die Mitgliedschaft in der Gruppe **wheel** dazu genutzt, den Zugriff als Superuser zu gewähren oder zu steuern.

## Interaktive Root-Shell mit Sudo

Wenn im System ein nicht administratives Benutzerkonto vorhanden ist, das mit **sudo** den Befehl **su** ausführen kann, können Sie **sudo su -** von diesem Konto aus ausführen, um eine interaktive **root**-Benutzer-Shell zu erhalten. Das ist möglich, weil **sudo su -** als **root** ausführt und **root** kein Passwort eingeben muss, um **su** zu verwenden.

## Kapitel 6 | Verwalten lokaler Benutzer und Gruppen

Ein anderer Weg, um auf das **root**-Benutzerkonto mit **sudo** zuzugreifen, ist die Verwendung des Befehls **sudo -i**. Dieser Befehl wechselt zum **root**-Benutzerkonto und führt die Standard-Shell dieses Benutzers (meistens **bash**) und zugehörige Shell-Anmeldeeskripts aus. Wenn Sie nur die Shell ausführen möchten, können Sie den Befehl **sudo -s** verwenden.

Ein Administrator kann beispielsweise eine interaktive Shell als **root** auf einer AWS EC2-Instanz mithilfe der SSH-Public-Key-Authentifizierung erhalten, um sich als regulärer Benutzer **ec2-user** anzumelden und dann durch Ausführen von **sudo -i** die Shell des Benutzers **root** zu erhalten.

```
[ec2-user@host ~]$ sudo -i  
[sudo] password for ec2-user:  
[root@host ~]#
```

Die Befehle **sudo su -** und **sudo -i** verhalten sich nicht genau gleich. Dies wird am Ende des Abschnitts kurz erläutert.

## Konfigurieren von Sudo

Die Hauptkonfigurationsdatei für **sudo** ist **/etc/sudoers**. Um Probleme zu vermeiden, wenn mehrere Administratoren versuchen, die Datei gleichzeitig zu bearbeiten, sollte sie nur mit dem Sonderbefehl **visudo** bearbeitet werden.

Die folgende Zeile aus der Datei **/etc/sudoers** ermöglicht beispielsweise den **sudo**-Zugriff für Mitglieder der Gruppe **wheel**.

```
%wheel      ALL=(ALL)      ALL
```

In dieser Zeile ist **%wheel** der Benutzer oder die Gruppe, für die die Regel gilt. Ein **%** gibt an, dass es sich um eine Gruppe, **wheel**, handelt. **ALL=(ALL)** gibt an, dass auf jedem Host, der diese Datei enthalten könnte, **wheel** jeden Befehl ausführen kann. **ALL** am Ende gibt an, dass **wheel** diese Befehle als jeder beliebige Benutzer auf dem System ausführen kann.

Standardmäßig bezieht **/etc/sudoers** auch den Inhalt aller Dateien im Verzeichnis **/etc/sudoers.d** als Teil der Konfigurationsdatei ein. Dadurch kann ein Administrator **sudo**-Zugriff für einen Benutzer durch einfaches Ablegen einer entsprechenden Datei in diesem Verzeichnis hinzufügen.



### Anmerkung

Die Verwendung von Zusatzdateien im Verzeichnis **/etc/sudoers.d** ist bequem und einfach. Sie können den **sudo**-Zugriff einfach aktivieren oder deaktivieren, indem Sie eine Datei in das Verzeichnis kopieren oder aus dem Verzeichnis entfernen.

In diesem Kurs erstellen und entfernen Sie Dateien im Verzeichnis **/etc/sudoers.d**, um den **sudo**-Zugriff für Benutzer und Gruppen zu konfigurieren.

Um den vollständigen **sudo**-Zugriff für den Benutzer **user01** zu aktivieren, könnten Sie **/etc/sudoers.d/user01** mit folgendem Inhalt erstellen:

```
user01  ALL=(ALL)  ALL
```

## Kapitel 6 | Verwalten lokaler Benutzer und Gruppen

Um den vollständigen **sudo**-Zugriff für die Gruppe **group01** zu aktivieren, könnten Sie **/etc/sudoers.d/group01** mit folgendem Inhalt erstellen:

```
%group01  ALL=(ALL)  ALL
```

Es ist auch möglich, **sudo** einzurichten, um einem Benutzer zu erlauben, Befehle als anderer Benutzer auszuführen, ohne sein Passwort einzugeben.

```
ansible  ALL=(ALL)  NOPASSWD:ALL
```

Zwar gibt es offensichtliche Sicherheitsrisiken, wenn einem Benutzer oder einer Gruppe dieses Zugriffsrecht eingeräumt wird, es wird jedoch häufig bei Cloud-Instanzen, virtuellen Rechnern und Bereitstellungssystemen zur Konfiguration der Server verwendet. Das Benutzerkonto mit diesem Zugriff muss sorgfältig geschützt werden und erfordert möglicherweise die SSH-Public-Key-Authentifizierung, damit ein Benutzer auf einem Remote-System überhaupt darauf zugreifen kann.

Das offizielle AMI für Red Hat Enterprise Linux im Amazon Web Services Marketplace wird beispielsweise mit gesperrten Passwörtern für die Benutzer **root** und **ec2-user** ausgeliefert. Das Benutzerkonto **ec2-user** ist so eingerichtet, dass ein interaktiver Remote-Zugriff über die SSH-Public-Key-Authentifizierung möglich ist. Der Benutzer **ec2-user** kann auch jeden Befehl als **root** ohne Passwort ausführen, da die letzte Zeile der AMI-Datei **/etc/sudoers** wie folgt lautet:

```
ec2-user  ALL=(ALL)  NOPASSWD: ALL
```

Die Anforderung zur Eingabe eines Passworts für **sudo** kann wieder aktiviert werden oder andere Änderungen können vorgenommen werden, um die Sicherheit im Rahmen des Konfigurationsprozesses des Systems zu erhöhen.



### Anmerkung

In diesem Kurs wird möglicherweise **sudo su** - anstelle von **sudo -i** verwendet. Beide Befehle funktionieren, aber es gibt einige geringfügige Unterschiede zwischen ihnen.

Der Befehl **sudo su** - richtet die **root**-Umgebung genau wie bei einer normalen Anmeldung ein, weil der Befehl **su** - die von **sudo** vorgenommenen Einstellungen ignoriert und die Umgebung von Grund auf neu einrichtet.

Die Standardkonfiguration des Befehls **sudo -i** richtet einige Details der **root**-Benutzerumgebung anders als bei einer normalen Anmeldung ein. Zum Beispiel legt der Befehl die Umgebungsvariable **PATH** etwas anders fest. Dies wirkt sich darauf aus, wo die Shell nach Befehlen sucht.

Sie können das Verhalten von **sudo -i** an **su** - anpassen, indem Sie **/etc/sudoers** mit **visudo** bearbeiten. Suchen Sie die Zeile

```
Defaults    secure_path = /sbin:/bin:/usr/sbin:/usr/bin
```

und ersetzen Sie sie durch die folgenden zwei Zeilen:

```
Defaults    secure_path = /usr/local/bin:/usr/bin  
Defaults>root  secure_path = /usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin
```

Für die meisten Zwecke ist dies kein wesentlicher Unterschied. Aus Gründen der Konsistenz der **PATH**-Einstellungen auf Systemen mit der Standarddatei **/etc/sudoers** verwenden die Autoren dieses Kurses in Beispielen **sudo -i**.



### Literaturhinweise

Manpages **su(1)**, **sudo(8)**, **visudo(8)** und **sudoers(5)**

**info libc persona** (*GNU C Library Reference Manual*)

- Abschnitt 30.2: Die Rolle eines Prozesses

(Hinweis: Das Paket *glibc-devel* muss installiert sein, damit dieser Infoknoten verfügbar ist.)

## ► Angeleitete Übung

# Zugriff als Superuser

In dieser Übung üben Sie den Wechsel zum **root**-Konto und die Ausführung von Befehlen als **root**.

## Ergebnisse

Es werden folgende Fähigkeiten vermittelt:

- Wechseln zu **root** mit **sudo** und Zugreifen auf die interaktive Shell als **root** ohne Kenntnis des Passworts des Superusers
- Erklären, wie **su** und **sudo** - sich auf die Shell-Umgebung auswirken können, wenn die Anmeldeskripts ausgeführt oder nicht ausgeführt werden
- Verwenden von **sudo** für die Ausführung anderer Befehle als **root**

## Bevor Sie Beginnen

Melden Sie sich als Benutzer **student** mit dem Passwort **student** bei **workstation** an.

Führen Sie auf **workstation** das Skript **lab users-sudo start** aus, um diese Übung zu beginnen. Dieses Skript erstellt die erforderlichen Benutzerkonten und Dateien, um die Umgebung korrekt einzurichten.

```
[student@workstation ~]$ lab users-sudo start
```

### ► 1. Öffnen Sie auf **workstation** als **student** eine SSH-Sitzung zu **servera**.

```
[student@workstation ~]$ ssh student@servera  
...output omitted...  
[student@servera ~]$
```

### ► 2. Untersuchen Sie die Shell-Umgebung von **student**. Zeigen Sie die aktuellen Benutzer- und Gruppeninformationen sowie das aktuelle Arbeitsverzeichnis an. Zeigen Sie auch die Umgebungsvariablen an, die das Benutzerverzeichnis und die Speicherorte der ausführbaren Dateien des Benutzers angeben.

#### 2.1. Führen Sie **id** aus, um die aktuellen Benutzer- und Gruppeninformationen anzuzeigen.

```
[student@servera ~]$ id  
uid=1000(student) gid=1000(student) groups=1000(student),10(wheel)  
context=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023
```

#### 2.2. Führen Sie **pwd** aus, um das aktuelle Arbeitsverzeichnis anzuzeigen.

```
[student@servera ~]$ pwd  
/home/student
```

- 2.3. Geben Sie die Werte der Variablen **HOME** und **PATH** aus, um das Benutzerverzeichnis bzw. den Pfad der ausführbaren Dateien des Benutzers zu ermitteln.

```
[student@servera ~]$ echo $HOME  
/home/student  
[student@servera ~]$ echo $PATH  
/home/student/.local/bin:/home/student/bin:/usr/local/bin:/usr/bin:/usr/local/sbin:/usr/sbin
```

- 3. Wechseln zu **root** in einer Shell ohne Anmeldung und untersuchen Sie die neue Shell-Umgebung.
- 3.1. Führen Sie **sudo su** an der Shell-Eingabeaufforderung aus, um als **root**-Benutzer zu fungieren.

```
[student@servera ~]$ sudo su  
[sudo] password for student: student  
[root@servera student]#
```

- 3.2. Führen Sie **id** aus, um die aktuellen Benutzer- und Gruppeninformationen anzuzeigen.

```
[root@servera student]# id  
uid=0(root) gid=0(root) groups=0(root)  
context=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023
```

- 3.3. Führen Sie **pwd** aus, um das aktuelle Arbeitsverzeichnis anzuzeigen.

```
[root@servera student]# pwd  
/home/student
```

- 3.4. Geben Sie die Werte der Variablen **HOME** und **PATH** aus, um das Benutzerverzeichnis bzw. den Pfad der ausführbaren Dateien des Benutzers zu ermitteln.

```
[root@servera student]# echo $HOME  
/root  
[root@servera student]# echo $PATH  
/sbin:/bin:/usr/sbin:/usr/bin:/usr/local/sbin:/usr/local/bin
```

Wenn Sie bereits Erfahrung mit Linux und dem Befehl **su** haben, haben Sie möglicherweise erwartet, dass **su** ohne die Bindestrich-Option (-), um als **root** zu fungieren, dazu führen würde, dass der aktuelle **PATH** von **student** beibehalten wird. Das ist nicht passiert. Wie Sie im nächsten Schritt sehen, ist dies auch nicht der übliche **PATH** zu **root**.

Was ist passiert? Der Unterschied ist, dass Sie **su** nicht direkt ausgeführt haben. Stattdessen haben Sie **su** als **root** mit **sudo** ausgeführt, weil Sie das Passwort des Superusers nicht besitzen. Der Befehl **sudo** überschreibt aus Sicherheitsgründen

**Kapitel 6 |** Verwalten lokaler Benutzer und Gruppen

die Variable **PATH** der ursprünglichen Umgebung. Jeder Befehl, der nach der anfänglichen Überschreibung ausgeführt wird, kann die Variable **PATH** aktualisieren, wie Sie in den folgenden Schritten gezeigt wird.

- 3.5. Beenden Sie die Shell des **root**-Benutzers, um zur Shell des **student**-Benutzers zurückzukehren.

```
[root@servera student]# exit  
exit  
[student@servera ~]$
```

- 4. Wechseln zu **root** in einer Login-Shell und untersuchen Sie die neue Shell-Umgebung.

- 4.1. Führen Sie **sudo su -** an der Shell-Eingabeaufforderung aus, um als **root**-Benutzer zu fungieren.

```
[student@servera ~]$ sudo su -  
[root@servera ~]#
```

Beachten Sie den Unterschied der Shell-Eingabeaufforderung im Vergleich zu der von **sudo su** im vorherigen Schritt.

**sudo** kann Sie abhängig von der Timeout-Periode von **sudo** dazu auffordern oder nicht, das Passwort für **student** einzugeben. Die Standard-Timeout-Periode beträgt fünf Minuten. Wenn Sie sich bei **sudo** innerhalb der letzten fünf Minuten authentifiziert haben, fragt **sudo** Sie nicht nach dem Passwort. Wenn seit der Authentifizierung bei **sudo** mehr als fünf Minuten vergangen sind, müssen Sie **student** als Passwort eingeben, um bei **sudo** authentifiziert zu werden.

- 4.2. Führen Sie **id** aus, um die aktuellen Benutzer- und Gruppeninformationen anzuzeigen.

```
[root@servera ~]# id  
uid=0(root) gid=0(root) groups=0(root)  
context=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023
```

- 4.3. Führen Sie **pwd** aus, um das aktuelle Arbeitsverzeichnis anzuzeigen.

```
[root@servera ~]# pwd  
/root
```

- 4.4. Geben Sie die Werte der Variablen **HOME** und **PATH** aus, um das Benutzerarbeitsverzeichnis bzw. den Pfad der ausführbaren Dateien des Benutzers zu ermitteln.

```
[root@servera ~]# echo $HOME  
/root  
[root@servera ~]# echo $PATH  
/usr/local/sbin:/usr/local/bin:/usr/sbin:/usr/bin:/root/bin
```

Wie im vorherigen Schritt hat der Befehl **su -** die Shell-Anmeldebeschreibung für **root** ausgeführt und die Variable **PATH** auf einen anderen Wert festgelegt, nachdem **sudo** die Variable **PATH** in den Einstellungen in der Shell-Umgebung des Benutzers **student** zurückgesetzt hat. Der Befehl **su** ohne die Bindestrich-Option (-) hat das nicht ausgeführt.

- 4.5. Beenden Sie die Shell des **root**-Benutzers, um zur Shell des **student**-Benutzers zurückzukehren.

```
[root@servera ~]# exit  
logout  
[student@servera ~]$
```

- 5. Überprüfen Sie, ob der Benutzer **operator1** so konfiguriert ist, dass jeder Benutzer beliebige Befehle mit **sudo** ausführen kann.

```
[student@servera ~]$ sudo cat /etc/sudoers.d/operator1  
operator1 ALL=(ALL) ALL
```

- 6. Fungieren Sie als **operator1** und zeigen Sie den Inhalt von **/var/log/messages** an. Kopieren Sie **/etc/motd** nach **/etc/motdOLD** und entfernen Sie sie (**/etc/motdOLD**). Diese Vorgänge erfordern Administratorrechte. Führen Sie daher **sudo** diese Befehle als Superuser aus. Wechseln Sie nicht mit **sudo su** oder **sudo su -** zu „root“. Verwenden Sie **redhat** als Passwort für **operator1**.

- 6.1. Wechseln zu **operator1**.

```
[student@servera ~]$ su - operator1  
Password: redhat  
[operator1@servera ~]$
```

- 6.2. Versuchen Sie, die letzten fünf Zeilen von **/var/log/messages** anzuzeigen, ohne **sudo** zu verwenden. Dies sollte fehlgeschlagen.

```
[operator1@servera ~]$ tail -5 /var/log/messages  
tail: cannot open '/var/log/messages' for reading: Permission denied
```

- 6.3. Versuchen Sie, die letzten fünf Zeilen von **/var/log/messages** mit **sudo** anzuzeigen. Dies sollte erfolgreich sein.

```
[operator1@servera ~]$ sudo tail -5 /var/log/messages  
[sudo] password for operator1: redhat  
Jan 23 15:53:36 servera su[2304]: FAILED SU (to operator1) student on pts/1  
Jan 23 15:53:51 servera su[2307]: FAILED SU (to operator1) student on pts/1  
Jan 23 15:53:58 servera su[2310]: FAILED SU (to operator1) student on pts/1  
Jan 23 15:54:12 servera su[2322]: (to operator1) student on pts/1  
Jan 23 15:54:25 servera su[2353]: (to operator1) student on pts/1
```



### Anmerkung

Die vorherige Ausgabe kann auf Ihrem System abweichen.

- 6.4. Versuchen Sie, eine Kopie von **/etc/motd** als **/etc/motdOLD** zu erstellen, ohne **sudo** zu verwenden. Dies sollte fehlgeschlagen.

```
[operator1@servera ~]$ cp /etc/motd /etc/motdOLD  
cp: cannot create regular file '/etc/motdOLD': Permission denied
```

- 6.5. Versuchen Sie, eine Kopie von **/etc/motd** als **/etc/motdOLD** mit **sudo** zu erstellen. Dies sollte erfolgreich sein.

```
[operator1@servera ~]$ sudo cp /etc/motd /etc/motdOLD  
[operator1@servera ~]$
```

- 6.6. Versuchen Sie, **/etc/motdOLD** zu löschen, ohne **sudo** zu verwenden. Dies sollte fehlschlagen.

```
[operator1@servera ~]$ rm /etc/motdOLD  
rm: remove write-protected regular empty file '/etc/motdOLD'? y  
rm: cannot remove '/etc/motdOLD': Permission denied  
[operator1@servera ~]$
```

- 6.7. Versuchen Sie, **/etc/motdOLD** mit **sudo** zu löschen. Dies sollte erfolgreich sein.

```
[operator1@servera ~]$ sudo rm /etc/motdOLD  
[operator1@servera ~]$
```

- 6.8. Beenden Sie die Shell des **operator1**-Benutzers, um zur Shell des **student**-Benutzers zurückzukehren.

```
[operator1@servera ~]$ exit  
logout  
[student@servera ~]$
```

- 6.9. Melden Sie sich von **servera** ab.

```
[student@servera ~]$ exit  
logout  
Connection to servera closed.  
[student@workstation ~]$
```

## Beenden

Führen Sie auf **workstation** das Skript **lab users-sudo finish** aus, um diese Übung zu beenden. Dieses Skript löscht die zu Beginn der Übung erstellten Benutzerkonten und Dateien, um eine bereinigte Umgebung sicherzustellen.

```
[student@workstation ~]$ lab users-sudo finish
```

Hiermit ist die angeleitete Übung beendet.

# Verwalten lokaler Benutzerkonten

---

## Ziele

Nach Abschluss dieses Abschnitts sollten Sie in der Lage sein, lokale Benutzerkonten zu erstellen, zu bearbeiten und zu löschen.

## Verwalten lokaler Benutzer

Zum Verwalten lokaler Benutzerkonten steht eine Reihe von Befehlszeilentoools zur Verfügung.

### Erstellen von Benutzern über die Befehlszeile

- Der Befehl **useradd** *username* erstellt einen neuen Benutzer mit dem Namen **username**. Er richtet das Benutzerverzeichnis und die Kontoinformationen des Benutzers ein und erstellt eine private Gruppe für den Benutzer **username**. Zu diesem Zeitpunkt verfügt das Benutzerkonto über kein gültiges Passwort und der Benutzer kann sich erst anmelden, wenn ein Passwort festgelegt ist.
- Der Befehl **useradd --help** zeigt die allgemeinen Optionen an, mit denen die Standardwerte überschrieben werden können. In den meisten Fällen stehen mit dem Befehl **usermod** dieselben Optionen zur Verfügung, um einen bestehenden Benutzer zu bearbeiten.
- Manche Standardwerte wie etwa der Bereich gültiger UID-Nummern und Standardregeln für Passwortalterung werden aus der Datei **/etc/login.defs** gelesen. Die Werte in dieser Datei kommen nur beim Erstellen neuer Benutzer zum Einsatz. Änderungen an dieser Datei haben keine Auswirkungen auf bestehende Benutzer.

### Ändern vorhandener Benutzer über die Befehlszeile

- Der Befehl **usermod --help** zeigt die allgemeinen Optionen an, mit denen ein Benutzerkonto geändert werden kann. Zu den gängigen Optionen gehören:

<b>usermod-Optionen:</b>	<b>Verwendung</b>
<b>-c, --comment COMMENT</b>	Fügt dem Kommentarfeld den tatsächlichen Namen des Benutzers hinzu.
<b>-g, --gid GROUP</b>	Legt die primäre Gruppe für das Benutzerkonto fest.
<b>-G, --groups GROUPS</b>	Legt eine durch Komma getrennte Liste von Zusatzgruppen für das Benutzerkonto fest.
<b>-a, --append</b>	Wird mit der Option <b>-G</b> verwendet, um die Zusatzgruppe dem Satz der aktuellen Gruppenmitgliedschaften des Benutzers hinzuzufügen, anstatt den Satz der Zusatzgruppen durch einen neuen Satz zu ersetzen.
<b>-d, --home HOME_DIR</b>	Legt ein bestimmtes Benuterverzeichnis für das Benutzerkonto fest.
<b>-m, --move-home</b>	Verschiebt das Benuterverzeichnis des Benutzers an einen neuen Speicherort. Muss in Kombination mit der Option <b>-d</b> verwendet werden.
<b>-s, --shell SHELL</b>	Legt eine bestimmte Anmelde-Shell für das Benutzerkonto fest.
<b>-L, --lock</b>	Sperrt das Benutzerkonto.
<b>-U, --unlock</b>	Entsperrt das Benutzerkonto.

### Löschen von Benutzern über die Befehlszeile

- Der Befehl **userdel username** entfernt die Details von **username** aus **/etc/passwd**, behält aber das Benuterverzeichnis des Benutzers bei.
- Der Befehl **userdel -r username** entfernt die Details von **username** aus **/etc/passwd** und löscht auch das Benuterverzeichnis des Benutzers.



### Warnung

Wenn ein Benutzer mit **userdel** ohne die Option **-r** entfernt wird, verbleiben Dateien im System, die einen Eigentümer ohne zugewiesene UID haben. Dies kann auch vorkommen, wenn eine Datei, deren Eigentümer ein gelöschter Benutzer ist, außerhalb des Benutzerverzeichnisses dieses Benutzers vorhanden ist. Diese Situation kann zu Informationslecks und anderen Sicherheitsproblemen führen.

In Red Hat Enterprise Linux 7 und Red Hat Enterprise Linux 8 weist der Befehl **useradd** neuen Benutzern die erste freie UID zu, die größer oder gleich 1000 ist, sofern Sie nicht explizit eine mit der Option **-u** angeben.

Auf folgende Weise kann es zu Informationslecks kommen. Falls die erste freie UID zuvor zu einem mittlerweile vom System gelöschten Benutzerkonto gehört hat, wird die UID des früheren Benutzers dem neuen Benutzer zugeordnet; dieser wird somit Eigentümer der verbleibenden Dateien des früheren Benutzers.

Das folgende Szenario verdeutlicht diese Situation.

```
[root@host ~]# useradd user01
[root@host ~]# ls -l /home
drwx----- 3 user01  user01    74 Feb  4 15:22 user01
[root@host ~]# userdel user01
[root@host ~]# ls -l /home
drwx----- 3    1000   1000    74 Feb  4 15:22 user01
[root@host ~]# useradd user02
[root@host ~]# ls -l /home
drwx----- 3 user02      user02      74 Feb  4 15:23 user02
drwx----- 3 user02      user02      74 Feb  4 15:22 user01
```

Wie Sie sehen, ist **user02** nun Eigentümer aller Dateien, deren Eigentümer zuvor **user01** war.

Je nach Situation ist eine Lösung für dieses Problem, alle Dateien ohne Eigentümer vom System zu löschen, wenn der Benutzer gelöscht wird, der die Dateien erstellt hat. Eine andere Lösung besteht darin, die Dateien ohne Eigentümer einem anderen Benutzer manuell zuzuweisen. Der Benutzer **root** kann mit dem Befehl **find** / - **nouser** -o **-nogroup** alle Dateien und Verzeichnisse ohne Eigentümer finden.

### Einrichten von Passwörtern über die Befehlszeile

- Der Befehl **passwd username** legt das anfängliche Passwort fest oder ändert das vorhandene Passwort von **username**.
- Der Benutzer **root** kann ein Passwort auf jeden beliebigen Wert festlegen. Es wird eine Meldung angezeigt, wenn das Passwort nicht die empfohlenen Mindestkriterien erfüllt; auf sie folgt jedoch eine Eingabeaufforderung zur erneuten Eingabe des neuen Passworts und alle Token werden erfolgreich aktualisiert.

```
[root@host ~]# passwd user01
Changing password for user user01.
New password: redhat
BAD PASSWORD: The password fails the dictionary check - it is based on a
dictionary word
Retype new password: redhat
passwd: all authentication tokens updated successfully.
[root@host ~]#
```

- Ein regulärer Benutzer muss ein Passwort mit einer Länge von mindestens acht Zeichen wählen, das zudem nicht auf einem Wort im Wörterbuch, dem Benutzernamen oder dem vorherigen Passwort beruht.

### UID-Bereiche

Red Hat Enterprise Linux nutzt bestimmte UID-Nummern und Nummernbereiche für spezifische Zwecke.

- UID 0* wird immer dem Superuser-Konto **root** zugewiesen.
- Der Bereich *UID 1–200* ist „Systembenutzern“ vorbehalten und wird von Red Hat statisch an Systemprozesse vergeben.
- Der Bereich *UID 201–999* ist für „Systembenutzer“ vorgesehen, die von Systemprozessen verwendet werden und nicht Eigentümer von Dateien im Dateisystem sind. Sie werden in der Regel bei der Installation der Software, die sie benötigt, dynamisch aus dem Pool der verfügbaren Nummern zugewiesen. Programme werden als diese „nicht privilegierten“ Systembenutzer ausgeführt, um ihren Zugriff auf genau diejenigen Ressourcen zu beschränken, die sie für ihr Funktionieren benötigen.
- UID 1000+* ist der Nummernbereich, der für die Zuweisung an reguläre Benutzer verfügbar ist.



#### Anmerkung

Vor RHEL 7 war es üblich, die UID-Nummern 1–499 für Systembenutzer und die UID-Nummern ab 500 für reguläre Benutzer zu verwenden. Die von **useradd** und **groupadd** genutzten Standardwerte können in der Datei **/etc/login.defs** geändert werden.



#### Literaturhinweise

Manpages **useradd(8)**, **usermod(8)**, **userdel(8)**

## ► Angeleitete Übung

# Verwalten lokaler Benutzerkonten

In dieser Übung erstellen Sie mehrere Benutzer in Ihrem System und legen Passwörter für diese Benutzer fest.

## Ergebnisse

Sie sollten in der Lage sein, ein Linux-System mit zusätzlichen Benutzerkonten zu konfigurieren.

## Bevor Sie Beginnen

Melden Sie sich als Benutzer **student** mit dem Passwort **student** bei **workstation** an.

Führen Sie auf **workstation** das Skript **lab users-manage start** aus, um diese Übung zu beginnen. Durch das Skript wird gewährleistet, dass die Umgebung richtig eingerichtet ist.

```
[student@workstation ~]$ lab users-manage start
```

- 1. Öffnen Sie auf **workstation** als **student** eine SSH-Sitzung zu **servera**.

```
[student@workstation ~]$ ssh student@servera
...output omitted...
[student@servera ~]$
```

- 2. Wechseln Sie auf **servera** mit **sudo** zu **root**, wobei die Umgebung des Benutzers **root** umgewandelt wird.

```
[student@servera ~]$ sudo su -
[sudo] password for student: student
[root@servera ~]#
```

- 3. Erstellen Sie den Benutzer **operator1** und überprüfen Sie, ob er im System vorhanden ist.

```
[root@servera ~]# useradd operator1
[root@servera ~]# tail /etc/passwd
...output omitted...
operator1:x:1002:1002::/home/operator1:/bin/bash
```

- 4. Legen Sie das Passwort für **operator1** auf **redhat** fest.

```
[root@servera ~]# passwd operator1
Changing password for user operator1.
New password: redhat
BAD PASSWORD: The password is shorter than 8 characters
Retype new password: redhat
passwd: all authentication tokens updated successfully.
```

- 5. Erstellen Sie die zusätzlichen Benutzer **operator2** und **operator3**. Legen Sie deren Passwörter auf **redhat** fest.

- 5.1. Fügen Sie den Benutzer **operator2** hinzu. Legen Sie das Passwort für **operator2** auf **redhat** fest.

```
[root@servera ~]# useradd operator2
[root@servera ~]# passwd operator2
Changing password for user operator2.
New password: redhat
BAD PASSWORD: The password is shorter than 8 characters
Retype new password: redhat
passwd: all authentication tokens updated successfully.
```

- 5.2. Fügen Sie den Benutzer **operator3** hinzu. Legen Sie das Passwort für **operator3** auf **redhat** fest.

```
[root@servera ~]# useradd operator3
[root@servera ~]# passwd operator3
Changing password for user operator3.
New password: redhat
BAD PASSWORD: The password is shorter than 8 characters
Retype new password: redhat
passwd: all authentication tokens updated successfully.
```

- 6. Aktualisieren Sie die Benutzerkonten **operator1** und **operator2**, um die Kommentare **Operator One** bzw. **Operator Two** einzubeziehen. Überprüfen Sie, ob die Kommentare erfolgreich hinzugefügt wurden.

- 6.1. Führen Sie **usermod -c** aus, um die Kommentare des Benutzerkontos **operator1** zu aktualisieren.

```
[root@servera ~]# usermod -c "Operator One" operator1
```

- 6.2. Führen Sie **usermod -c** aus, um die Kommentare des Benutzerkontos **operator2** zu aktualisieren.

```
[root@servera ~]# usermod -c "Operator Two" operator2
```

- 6.3. Vergewissern Sie sich, dass die Kommentare für beide Benutzer, **operator1** und **operator2**, in den Benutzerdatensätzen vorhanden sind.

```
[root@servera ~]# tail /etc/passwd
...output omitted...
operator1:x:1002:1002:Operator One:/home/operator1:/bin/bash
operator2:x:1003:1003:Operator Two:/home/operator2:/bin/bash
operator3:x:1004:1004::/home/operator3:/bin/bash
```

- 7. Löschen Sie den Benutzer **operator3** zusammen mit allen persönlichen Daten des Benutzers. Überprüfen Sie, ob der Benutzer erfolgreich gelöscht wurde.

- 7.1. Entfernen Sie den Benutzer **operator3** aus dem System.

```
[root@servera ~]# userdel -r operator3
```

- 7.2. Überprüfen Sie, ob **operator3** erfolgreich gelöscht wurde.

```
[root@servera ~]# tail /etc/passwd
...output omitted...
operator1:x:1002:1002:Operator One:/home/operator1:/bin/bash
operator2:x:1003:1003:Operator Two:/home/operator2:/bin/bash
```

Beachten Sie, dass in der vorherigen Ausgabe die Benutzerkontoinformationen von **operator3** nicht angezeigt werden.

- 7.3. Beenden Sie die Shell des **root**-Benutzers, um zur Shell des **student**-Benutzers zurückzukehren.

```
[root@servera ~]# exit
logout
[student@servera ~]$
```

- 7.4. Melden Sie sich von **servera** ab.

```
[student@servera ~]$ exit
logout
Connection to servera closed.
[student@workstation ~]$
```

## Beenden

Führen Sie auf **workstation** das Skript **lab users-manage finish** aus, um diese Übung zu beenden. Durch das Skript wird gewährleistet, dass die Umgebung bereinigt ist.

```
[student@workstation ~]$ lab users-manage finish
```

Hiermit ist die angeleitete Übung beendet.

# Verwalten lokaler Gruppenkonten

## Ziele

Nach Abschluss dieses Abschnitts sollten die Teilnehmer in der Lage sein, lokale Gruppenkonten zu erstellen, zu bearbeiten und zu löschen.

## Verwalten lokaler Gruppen

Bevor ein Benutzer einer Gruppe hinzugefügt werden kann, muss diese erstellt werden. Zum Verwalten lokaler Gruppenkonten werden verschiedene Befehlszeilentools verwendet.

### Erstellen von Gruppen über die Befehlszeile

- Der Befehl **groupadd** erstellt Gruppen. Ohne Optionen verwendet der Befehl **groupadd** beim Erstellen von Gruppen die nächste verfügbare GID aus dem Bereich, der in der Datei **/etc/login.defs** angegeben ist.
- Die Option **-g** gibt eine bestimmte GID für die zu verwendende Gruppe an.

```
[user01@host ~]$ sudo groupadd -g 10000 group01
[user01@host ~]$ tail /etc/group
...output omitted...
group01:x:10000:
```



### Anmerkung

Angesichts der automatischen Erstellung von UPGs (GID 1000+) ist es im Allgemeinen empfehlenswert, einen Bereich von GIDs für Zusatzgruppen zu reservieren. Ein höherer Bereich verhindert eine Kollision mit einer Systemgruppe (GID 0-999).

- Mit der Option **-r** wird eine Systemgruppe erstellt, die eine GID im Bereich der gültigen System-GIDs aus der Datei **/etc/login.defs** verwendet. Die Konfigurationselemente **SYS\_GID\_MIN** und **SYS\_GID\_MAX** in **/etc/login.defs** definieren den Bereich der System-GIDs.

```
[user01@host ~]$ sudo groupadd -r group02
[user01@host ~]$ tail /etc/group
...output omitted...
group01:x:10000:
group02:x:988:
```

### Ändern vorhandener Gruppen über die Befehlszeile

- Der Befehl **groupmod** ändert die Eigenschaften einer vorhandenen Gruppe. Mit der Option **-n** wird ein neuer Name für die Gruppe angegeben.

```
[user01@host ~]$ sudo groupmod -n group0022 group02
[user01@host ~]$ tail /etc/group
...output omitted...
group0022:x:988:
```

Beachten Sie, dass der Gruppenname von **group02** auf **group0022** aktualisiert wird.

- Mit der Option **-g** wird eine neue GID festgelegt.

```
[user01@host ~]$ sudo groupmod -g 20000 group0022
[user01@host ~]$ tail /etc/group
...output omitted...
group0022:x:20000:
```

Beachten Sie, dass die GID von **988** auf **20000** aktualisiert wird.

### Löschen von Gruppen über die Befehlszeile

- Der Befehl **groupdel** entfernt Gruppen.

```
[user01@host ~]$ sudo groupdel group0022
```



#### Anmerkung

Sie können eine Gruppe nicht löschen, wenn sie die primäre Gruppe eines bestehenden Benutzers ist. Stellen Sie ebenso wie bei **userdel** auf allen Dateisystemen sicher, dass keine Dateien auf dem System verbleiben, deren Eigentümer die Gruppe ist.

### Ändern der Gruppenmitgliedschaft über die Befehlszeile

- Die Mitgliedschaft bei einer Gruppe wird durch die Benutzerverwaltung gesteuert. Mit dem Befehl **usermod -g** ändern Sie die primäre Gruppe eines Benutzers.

```
[user01@host ~]$ id user02
uid=1006(user02) gid=1008(user02) groups=1008(user02)
[user01@host ~]$ sudo usermod -g group01 user02
[user01@host ~]$ id user02
uid=1006(user02) gid=10000(group01) groups=10000(group01)
```

- Mit dem Befehl **usermod -aG** fügen Sie einer Zusatzgruppe einen Benutzer hinzu.

```
[user01@host ~]$ id user03
uid=1007(user03) gid=1009(user03) groups=1009(user03)
[user01@host ~]$ sudo usermod -aG group01 user03
[user01@host ~]$ id user03
uid=1007(user03) gid=1009(user03) groups=1009(user03),10000(group01)
```

 **Wichtig**

Durch Verwendung der Option **-a** arbeitet **usermod** im Modus *Anhängen* (append). Ohne **-a** wird der Benutzer aus allen seinen aktuellen Zusatzgruppen entfernt, die nicht in der Liste der Option **-G** enthalten sind.



**Literaturhinweise**

Manpages **group(5)**, **groupadd(8)**, **groupdel(8)** und **usermod(8)**

## ► Angeleitete Übung

# Verwalten lokaler Gruppenkonten

In dieser Übung erstellen Sie Gruppen, verwenden sie als Zusatzgruppen für bestimmte Benutzer, ohne die primären Gruppen dieser Benutzer zu ändern, und konfigurieren eine der Gruppen mit sudo-Zugriff, um Befehle als **root** auszuführen.

## Ergebnisse

Es werden folgende Fähigkeiten vermittelt:

- Erstellen von Gruppen und Verwenden dieser Gruppen als Zusatzgruppen
- Konfigurieren des vollständigen sudo-Zugriffs für eine Gruppe

## Bevor Sie Beginnen

Melden Sie sich als Benutzer **student** mit dem Passwort **student** bei **workstation** an.

Führen Sie auf **workstation** das Skript **lab users-group-manage start** aus, um diese Übung zu beginnen. Dieses Skript erstellt die erforderlichen Benutzerkonten, um die Umgebung korrekt einzurichten.

```
[student@workstation ~]$ lab users-group-manage start
```

- 1. Öffnen Sie auf **workstation** als **student** eine SSH-Sitzung zu **servera**.

```
[student@workstation ~]$ ssh student@servera
...output omitted...
[student@servera ~]$
```

- 2. Wechseln Sie auf **servera** mit **sudo** zu **root**, wobei die gesamte Umgebung des Benutzers **root** geerbt wird.

```
[student@servera ~]$ sudo su -
[sudo] password for student: student
[root@servera ~]#
```

- 3. Erstellen Sie die Zusatzgruppe **operators** mit der GID 30000.

```
[root@servera ~]# groupadd -g 30000 operators
```

- 4. Erstellen Sie eine weitere Zusatzgruppe namens **admin**.

```
[root@servera ~]# groupadd admin
```

- 5. Überprüfen Sie, ob die beiden Zusatzgruppen, **operators** und **admin**, vorhanden sind.

```
[root@servera ~]# tail /etc/group  
...output omitted...  
operators:x:30000:  
admin:x:30001:
```

- 6. Stellen Sie sicher, dass die Benutzer **operator1**, **operator2** und **operator3** zur Gruppe **operators** gehören.

- 6.1. Fügen Sie **operators** die Benutzer **operator1**, **operator2** und **operator3** hinzu.

```
[root@servera ~]# usermod -aG operators operator1  
[root@servera ~]# usermod -aG operators operator2  
[root@servera ~]# usermod -aG operators operator3
```

- 6.2. Überprüfen Sie, ob die Benutzer erfolgreich der Gruppe hinzugefügt wurden.

```
[root@servera ~]# id operator1  
uid=1002(operator1) gid=1002(operator1) groups=1002(operator1),30000(operators)  
[root@servera ~]# id operator2  
uid=1003(operator2) gid=1003(operator2) groups=1003(operator2),30000(operators)  
[root@servera ~]# id operator3  
uid=1004(operator3) gid=1004(operator3) groups=1004(operator3),30000(operators)
```

- 7. Stellen Sie sicher, dass die Benutzer **sysadmin1**, **sysadmin2** und **sysadmin3** zur Gruppe **admin** gehören. Aktivieren Sie Administratorrechte für alle Gruppenmitglieder von **admin**. Überprüfen Sie, ob die Mitglieder von **admin** Verwaltungsbefehle ausführen können.

- 7.1. Fügen Sie **admin** die Benutzer **sysadmin1**, **sysadmin2** und **sysadmin3** hinzu.

```
[root@servera ~]# usermod -aG admin sysadmin1  
[root@servera ~]# usermod -aG admin sysadmin2  
[root@servera ~]# usermod -aG admin sysadmin3
```

- 7.2. Überprüfen Sie, ob die Benutzer erfolgreich der Gruppe hinzugefügt wurden.

```
[root@servera ~]# id sysadmin1  
uid=1005(sysadmin1) gid=1005(sysadmin1) groups=1005(sysadmin1),30001(admin)  
[root@servera ~]# id sysadmin2  
uid=1006(sysadmin2) gid=1006(sysadmin2) groups=1006(sysadmin2),30001(admin)  
[root@servera ~]# id sysadmin3  
uid=1007(sysadmin3) gid=1007(sysadmin3) groups=1007(sysadmin3),30001(admin)
```

- 7.3. Untersuchen Sie **/etc/group**, um die Mitgliedschaften in den Zusatzgruppen zu überprüfen.

```
[root@servera ~]# tail /etc/group  
...output omitted...  
operators:x:30000:operator1,operator2,operator3  
admin:x:30001:sysadmin1,sysadmin2,sysadmin3
```

- 7.4. Erstellen Sie die Datei **/etc/sudoers.d/admin** so, dass die Mitglieder von **admin** vollständige Administratorrechte besitzen.

```
[root@servera ~]# echo "%admin ALL=(ALL) ALL" >> /etc/sudoers.d/admin
```

- 7.5. Wechseln zu **sysadmin1** (ein Mitglied von **admin**) und vergewissern Sie sich, dass Sie einen **sudo**-Befehl als **sysadmin1** ausführen können.

```
[root@servera ~]# su - sysadmin1
[sysadmin1@servera ~]$ sudo cat /etc/sudoers.d/admin
[sudo] password for sysadmin1: redhat
%admin ALL=(ALL) ALL
```

- 7.6. Beenden Sie die Shell des **sysadmin1**-Benutzers, um zur Shell des **root**-Benutzers zurückzukehren.

```
[sysadmin1@servera ~]$ exit
logout
[root@servera ~]#
```

- 7.7. Beenden Sie die Shell des **root**-Benutzers, um zur Shell des **student**-Benutzers zurückzukehren.

```
[root@servera ~]# exit
logout
[student@servera ~]$
```

- 7.8. Melden Sie sich von **servera** ab.

```
[student@servera ~]$ exit
logout
Connection to servera closed.
[student@workstation ~]$
```

## Beenden

Führen Sie auf **workstation** das Skript **lab users-group-manage finish** aus, um diese Übung zu beenden. Dieses Skript löscht die zu Beginn der Übung erstellten Benutzerkonten.

```
[student@workstation ~]$ lab users-group-manage finish
```

Hiermit ist die angeleitete Übung beendet.

# Verwalten von Benutzerpasswörtern

## Ziele

Nach Abschluss dieses Abschnitts sollten Sie in der Lage sein, eine Passwortverwaltungsrichtlinie für Benutzer festzulegen und Benutzerkonten manuell zu sperren und zu entsperren.

## Shadow-Passwörter und Passwortrichtlinie

Früher wurden verschlüsselte Passwörter in der Datei **/etc/passwd** gespeichert, die von jedem gelesen werden konnte. Dies galt als relativ sicher, bis Wörterbuchangriffe auf verschlüsselte Passwörter alltäglich wurden. Zu diesem Zeitpunkt wurden die verschlüsselten Passwörter in eine separate Datei **/etc/shadow** verschoben, die nur von **root** gelesen werden kann. Dank dieser Datei konnten auch Funktionen wie der Passwortablauf implementiert werden.

Wie in **/etc/passwd** enthält die Datei **/etc/shadow** für jeden Benutzer eine Zeile. Eine Musterzeile von **/etc/shadow** mit neun durch Doppelpunkt getrennten Feldern ist unten dargestellt.

```
❶ user03:❷$6$CSSx...output omitted...:❸17933:❹0:❺99999:❻7:❼2:❽18113:❾
```

- ❶ Benutzername des Benutzerkontos, zu dem dieses Passwort gehört.
- ❷ Das verschlüsselte Passwort des Benutzers. Das Format verschlüsselter Passwörter wird später in diesem Abschnitt beschrieben.
- ❸ Der Tag, an dem das Passwort zuletzt geändert wurde. Dies wird in Tagen seit dem 01.01.1970 festgelegt und in der UTC-Zeitzone berechnet.
- ❹ Die Mindestanzahl von Tagen, die seit der letzten Passwortänderung vergehen muss, bevor der Benutzer das Passwort erneut ändern kann.
- ❺ Die maximale Anzahl von Tagen, die ohne Passwortänderung vergehen kann, bevor das Passwort abläuft. Ein leeres Feld bedeutet, dass das Passwort nicht nach einer bestimmten Zeit ab der letzten Änderung abläuft.
- ❻ Warnzeitraum. Der Benutzer wird über ein ablaufendes Passwort informiert, wenn er sich innerhalb dieser Anzahl von Tagen vor Ablauf der Frist anmeldet.
- ❼ Inaktivitätszeitraum. Wenn das Passwort abgelaufen ist, wird es noch für diese Anzahl von Tagen für die Anmeldung akzeptiert. Nach Ablauf dieser Frist wird das Konto gesperrt.
- ❽ Der Tag, an dem das Konto abläuft. Dies wird in Tagen seit dem 01.01.1970 festgelegt und in der UTC-Zeitzone berechnet. Ein leeres Feld bedeutet, dass es nicht an einem bestimmten Datum abläuft.
- ❾ Das letzte Feld ist in der Regel leer und für zukünftige Zwecke reserviert.

## Format eines verschlüsselten Passworts

Das Feld für das verschlüsselte Passwort speichert drei Informationen: den verwendeten *Hashalgorithmus*, das *Salt* und den verschlüsselten *Hashwert*. Jede Information wird durch das Zeichen **\$** getrennt.

```
$❶6$❷CSSx...output omitted...
```

- ➊ Der für dieses Passwort verwendete Hashalgorithmus. Die Zahl **6** gibt an, dass es sich um einen SHA-512-Hash handelt. Das ist die Standardeinstellung in Red Hat Enterprise Linux 8. Eine **1** würde MD5 angeben, eine **5** SHA-256.
- ➋ Das zur Verschlüsselung des Passworts verwendete Salt. Dieses wird anfangs zufällig ausgewählt.
- ➌ Der verschlüsselte Hashwert des Benutzerpassworts. Das Salt und das unverschlüsselte Passwort werden kombiniert und verschlüsselt, um den verschlüsselten Hashwert des Passworts zu bilden.

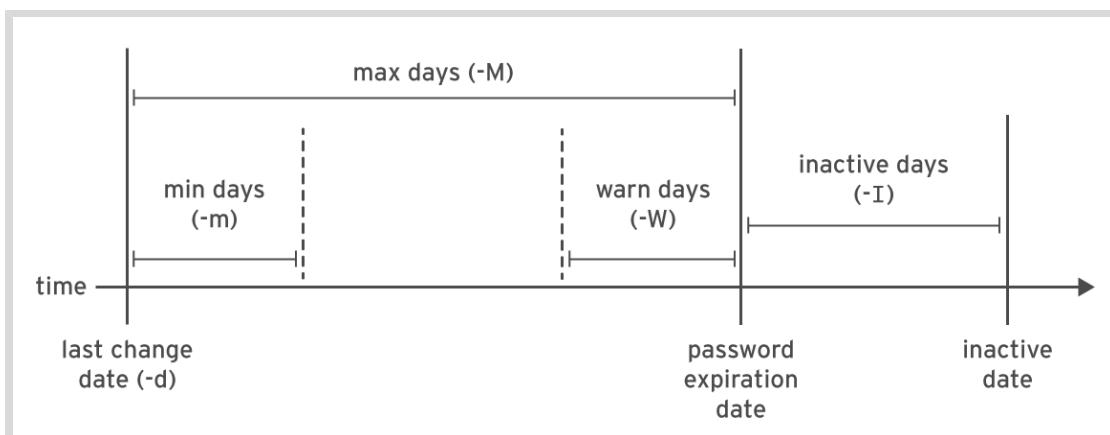
Der Hauptgrund für die Kombination eines Salt mit dem Passwort ist die Abwehr von Angriffen mit vorab berechneten Listen von Passwort-Hashes. Durch das Hinzufügen von Salts werden die resultierenden Hashes geändert und die vorberechnete Liste nutzlos. Wenn ein Angreifer eine Kopie einer **/etc/shadow**-Datei abrufen kann, die Salts verwendet, muss er ein Brute-Force-Passwort erraten, was mehr Zeit und Aufwand erfordert.

## Passwortverifizierung

Wenn sich ein Benutzer anmelden möchte, sucht das System nach dem Eintrag des Benutzers in der Datei **/etc/shadow**, kombiniert das Salt für den Benutzer mit dem eingegebenen, unverschlüsselten Passwort und verschlüsselt diese mittels des angegebenen Hashalgorithmus. Wenn das Ergebnis mit dem verschlüsselten Hashwert übereinstimmt, hat der Benutzer das richtige Passwort eingegeben. Wenn das Ergebnis nicht mit dem verschlüsselten Hashwert übereinstimmt, hat der Benutzer das falsche Passwort eingegeben und der Anmeldeversuch schlägt fehl. Mit dieser Methode kann das System feststellen, ob der Benutzer das richtige Passwort eingegeben hat, ohne das Passwort in einer Form speichern zu müssen, die zum Anmelden verwendet werden könnte.

## Konfigurieren des Passwortablaufs

Das folgende Diagramm zeigt die relevanten Parameter zum Passwortablauf, die mit dem Befehl **chage** angepasst werden können, um eine Richtlinie zum Passwortablauf zu implementieren.



```
[user01@host ~]$ sudo chage -m 0 -M 90 -W 7 -I 14 user03
```

Der vorherigen **chage**-Befehl verwendet die Optionen **-m**, **-M**, **-W** und **-I** zum Festlegen des Mindestalters, des Höchstalters, des Warnzeitraums bzw. des Zeitraums der Inaktivität des Benutzerpassworts.

Der Befehl **chage -d 0 user03** verlangt, dass der Benutzer **user03** bei der nächsten Anmeldung sein Passwort aktualisiert.

## Kapitel 6 | Verwalten lokaler Benutzer und Gruppen

Der Befehl **chage -1 user03** zeigt die Details des Passwortablaufs von **user03** an.

Der Befehl **chage -E 2019-08-05 user03** bewirkt, dass das Konto des Benutzers **user03** am 05.08.2019 (im Format JJJJ-MM-TT) abläuft.



### Anmerkung

Mit dem Befehl **date** kann ein Datum in der Zukunft berechnet werden. Die Option **-u** zeigt die Zeit in UTC an.

```
[user01@host ~]$ date -d "+45 days" -u  
Thu May 23 17:01:20 UTC 2019
```

Bearbeiten Sie die Konfigurationselemente für den Passwortablauf in der Datei **/etc/login.defs**, um die Standardrichtlinien für den Passwortablauf festzulegen. **PASS\_MAX\_DAYS** legt das standardmäßige Höchstalter des Passworts fest. **PASS\_MIN\_DAYS** legt das standardmäßige Mindestalter des Passworts fest. **PASS\_WARN\_AGE** legt den standardmäßigen Warnzeitraum des Passworts fest. Jede Änderung der Standardrichtlinien für den Passwortablauf gilt nur für neue Benutzer. Die vorhandenen Benutzer verwenden weiterhin die alten Einstellungen für den Passwortablauf und nicht die neuen.

## Einschränken des Zugriffs

Sie können mit dem Befehl **chage** das Ablaufdatum des Kontos festlegen. Wenn dieses Datum erreicht ist, kann sich der Benutzer nicht mehr interaktiv beim System anmelden. Mit dem Befehl **usermod** und der Option **-L** kann ein Konto gesperrt werden.

```
[user01@host ~]$ sudo usermod -L user03  
[user01@host ~]$ su - user03  
Password: redhat  
su: Authentication failure
```

Wenn ein Mitarbeiter das Unternehmen verlässt, kann der Administrator das Benutzerkonto mit dem Befehl **usermod** sperren und schließen. Das Datum muss als Tage seit dem 01.01.1970 im Format JJJJ-MM-TT angegeben werden.

```
[user01@host ~]$ sudo usermod -L -e 2019-10-05 user03
```

Der obige Befehl **usermod** verwendet die Option **-e**, um das Ablaufdatum für das angegebene Benutzerkonto festzulegen. Die Option **-L** sperrt das Passwort des Benutzers.

Das Sperren des Kontos verhindert, dass der Benutzer sich mit einem Passwort beim System authentifizieren kann. Das ist die empfohlene Methode, um einem Mitarbeiter, der das Unternehmen verlassen hat, den Zugang zum System zu verwehren. Sollte der Mitarbeiter zu einem späteren Zeitpunkt wieder zurückkehren, kann das Benutzerkonto mit dem Befehl **usermod -U** wieder entsperrt werden. Ändern Sie auch das Ablaufdatum, falls das Konto außerdem abgelaufen war.

## Die nologin-Shell

Die **nologin**-Shell fungiert als Ersatz-Shell für die Benutzerkonten, die nicht für die interaktive Anmeldung beim System vorgesehen sind. Es ist aus Sicherheitsgründen ratsam, die Anmeldung eines Kontos bei einem System zu deaktivieren, wenn das Konto den Service nicht benötigt.

Beispielsweise benötigt ein Mail-Server evtl. ein Konto, um Post zu speichern, und ein Benutzerpasswort zur Mail-Client-Authentifizierung, um Post zu empfangen. Der Benutzer muss sich aber nicht direkt bei dem System anmelden.

Dieses Problem wird gewöhnlich mit dem Festlegen der Login-Shell des Benutzers auf **/sbin/nologin** gelöst. Wenn der Benutzer versucht, sich direkt beim System anzumelden, schließt die **nologin**-Shell die Verbindung.

```
[user01@host ~]$ usermod -s /sbin/nologin user03  
[user01@host ~]$ su - user03  
Last login: Wed Feb  6 17:03:06 IST 2019 on pts/0  
This account is currently not available.
```



### Wichtig

Die **nologin**-Shell verhindert die interaktive Verwendung des Systems, aber nicht den gesamten Zugriff. Benutzer könnten sich weiterhin authentifizieren und Dateien beispielsweise über Webanwendungen, Dateitransferprotokolle oder E-Mail-Reader hoch- oder herunterladen, wenn Sie das Passwort des Benutzers zu Authentifizierung verwenden.



### Literaturhinweise

Manpages **chage(1)**, **usermod(8)**, **shadow(5)**, **crypt(3)**

## ► Angeleitete Übung

# Verwalten von Benutzerpasswörtern

In dieser Übung legen Sie Passwortrichtlinien für mehrere Benutzer fest.

## Ergebnisse

Es werden folgende Fähigkeiten vermittelt:

- Erzwingen einer Passwortänderung bei der ersten Anmeldung des Benutzers beim System
- Erzwingen einer Passwortänderung alle 90 Tage
- Festlegen, dass das Konto nach 180 Tagen ab dem aktuellen Datum abläuft

## Bevor Sie Beginnen

Melden Sie sich als Benutzer **student** mit dem Passwort **student** bei **workstation** an.

Führen Sie auf **workstation** das Skript **lab users-pw-manage start** aus, um diese Übung zu beginnen. Dieses Skript erstellt die erforderlichen Benutzerkonten und Dateien, um sicherzustellen, dass die Umgebung korrekt eingerichtet ist.

```
[student@workstation ~]$ lab users-pw-manage start
```

### ► 1. Öffnen Sie auf **workstation** als **student** eine SSH-Sitzung zu **servera**.

```
[student@workstation ~]$ ssh student@servera  
...output omitted...  
[student@servera ~]$
```

### ► 2. Üben Sie auf **servera** das Sperren und Entsperren von Benutzerkonten als **student**.

#### 2.1. Sperren Sie als **student** mit Administratorrechten das Konto **operator1**.

```
[student@servera ~]$ sudo usermod -L operator1  
[sudo] password for student:
```

#### 2.2. Versuchen Sie, sich als **operator1** anzumelden. Dies sollte fehlschlagen.

```
[student@servera ~]$ su - operator1  
Password: redhat  
su: Authentication failure
```

#### 2.3. Entsperren Sie das Konto **operator1**.

```
[student@servera ~]$ sudo usermod -U operator1
```

- 2.4. Versuchen Sie erneut, sich als **operator1** anzumelden. Dies sollte erfolgreich sein.

```
[student@servera ~]$ su - operator1  
Password: redhat  
...output omitted...  
[operator1@servera ~]$
```

- 2.5. Beenden Sie die Shell des **operator1**-Benutzers, um zur Shell des **student**-Benutzers zurückzukehren.

```
[operator1@servera ~]$ exit  
logout
```

- 3. Ändern Sie die Passwortrichtlinie für das Konto **operator1** so, dass alle 90 Tage das Passwort geändert werden muss. Vergewissern Sie sich, dass der Passwortablauf erfolgreich festgelegt wurde.

- 3.1. Legen Sie das Höchstalter des Benutzerpassworts des Benutzers **operator1** auf 90 Tage fest.

```
[student@servera ~]$ sudo chage -M 90 operator1
```

- 3.2. Überprüfen Sie, ob das Passwort des Benutzers **operator1** 90 Tage nach der Änderung abläuft.

```
[student@servera ~]$ sudo chage -l operator1  
Last password change      : Jan 25, 2019  
Password expires          : Apr 25, 2019  
Password inactive         : never  
Account expires           : never  
Minimum number of days between password change   : 0  
Maximum number of days between password change   : 90  
Number of days of warning before password expires : 7
```

- 4. Forcieren Sie eine Passwortänderung für das Konto **operator1** bei der ersten Anmeldung.

```
[student@servera ~]$ sudo chage -d 0 operator1
```

- 5. Melden Sie sich als **operator1** an und ändern Sie das Passwort in **forsooth123**. Kehren Sie nach dem Festlegen des Passworts zur Shell des Benutzers **student** zurück.

- 5.1. Melden Sie sich als **operator1** an und ändern Sie das Passwort in **forsooth123**, wenn Sie dazu aufgefordert werden.

```
[student@servera ~]$ su - operator1
Password: redhat
You are required to change your password immediately (administrator enforced)
Current password: redhat
New password: forsooth123
Retype new password: forsooth123
...output omitted...
[operator1@servera ~]$
```

- 5.2. Beenden Sie die Shell des **operator1**-Benutzers, um zur Shell des **student**-Benutzers zurückzukehren.

```
[operator1@servera ~]$ exit
logout
```

- ▶ 6. Legen Sie fest, dass das Konto **operator1** nach 180 Tagen ab dem aktuellen Datum abläuft. Hinweis: Mit **date -d "+180 days"** erhalten Sie das Datum und die Uhrzeit 180 Tage ab dem aktuellen Datum und der aktuellen Uhrzeit.
- 6.1. Legen Sie ein Datum 180 Tage in der Zukunft fest. Verwenden Sie das Format **%F** mit dem Befehl **date**, um den genauen Wert zu erhalten.

```
[student@servera ~]$ date -d "+180 days" +%F
2019-07-24
```

Abhängig vom aktuellen Datum und der aktuellen Uhrzeit auf Ihrem System erhalten Sie im folgenden Schritt möglicherweise einen anderen Wert.

- 6.2. Legen Sie fest, dass das Konto an dem im vorherigen Schritt angezeigten Datum abläuft.

```
[student@servera ~]$ sudo chage -E 2019-07-24 operator1
```

- 6.3. Überprüfen Sie, ob das Ablaufdatum des Kontos erfolgreich festgelegt wurde.

```
[student@servera ~]$ sudo chage -l operator1
Last password change      : Jan 25, 2019
Password expires          : Apr 25, 2019
Password inactive         : never
Account expires           : Jul 24, 2019
Minimum number of days between password change   : 0
Maximum number of days between password change   : 90
Number of days of warning before password expires : 7
```

- ▶ 7. Legen Sie fest, dass die Passwörter für alle Benutzer 180 Tage nach dem aktuellen Datum ablaufen. Verwenden Sie Administratorrechte, um die Konfigurationsdatei zu bearbeiten.
- 7.1. Legen Sie **PASS\_MAX\_DAYS** in **/etc/login.defs** auf **180** fest. Verwenden Sie Administratorrechte, wenn Sie die Datei mit dem Texteditor öffnen. Sie können den Befehl **sudo vim /etc/login.defs** verwenden, um diesen Schritt auszuführen.

```
...output omitted...
# Password aging controls:
#
#      PASS_MAX_DAYS   Maximum number of days a password may be
#      used.
#      PASS_MIN_DAYS   Minimum number of days allowed between
#      password changes.
#      PASS_MIN_LEN     Minimum acceptable password length.
#      PASS_WARN_AGE    Number of days warning given before a
#      password expires.
#
PASS_MAX_DAYS  180
PASS_MIN_DAYS  0
PASS_MIN_LEN   5
PASS_WARN_AGE  7
...output omitted...
```



### Wichtig

Die Standardeinstellungen für Passwort- und Kontoablauf gelten für neue Benutzer, jedoch nicht für vorhandene Benutzer.

7.2. Melden Sie sich von **servera** ab.

```
[student@servera ~]$ exit
logout
Connection to servera closed.
[student@workstation ~]$
```

## Beenden

Führen Sie auf **workstation** das Skript **lab users-pw-manage finish** aus, um diese Übung zu beenden. Dieses Skript löscht die zu Beginn der Übung erstellten Benutzerkonten und Dateien, um eine bereinigte Umgebung sicherzustellen.

```
[student@workstation ~]$ lab users-pw-manage finish
```

Hiermit ist die angeleitete Übung beendet.

## ► Praktische Übung

# Verwalten lokaler Benutzer und Gruppen

### Leistungscheckliste

In dieser praktischen Übung legen Sie eine lokale Standardrichtlinie für Passwörter fest, erstellen eine Zusatzgruppe für drei Benutzer, lassen zu, dass diese Gruppe als **root** mit **sudo** Befehle ausführen kann, und ändern die Passwortrichtlinie für einen Benutzer.

### Ergebnisse

Es werden folgende Fähigkeiten vermittelt:

- Festlegen einer Standardrichtlinie für den Passwortablauf für das Passwort des lokalen Benutzers
- Erstellen einer Gruppe und Verwenden dieser Gruppe als Zusatzgruppe für neue Benutzer
- Erstellen von drei neuen Benutzern mit der neuen Gruppe als Zusatzgruppe
- Konfigurieren der Gruppenmitglieder der Zusatzgruppe, damit sie Befehle als beliebiger Benutzer mit **sudo** ausführen können
- Festlegen einer benutzerspezifischen Richtlinie für den Passwortablauf

### Bevor Sie Beginnen

Melden Sie sich als Benutzer **student** mit dem Passwort **student** bei **workstation** an.

Führen Sie auf **workstation** das Skript **lab users-review start** aus, um diese Übung zu beginnen. Dieses Skript erstellt die erforderlichen Dateien, um sicherzustellen, dass die Umgebung korrekt eingerichtet ist.

```
[student@workstation ~]$ lab users-review start
```

1. Öffnen Sie auf **workstation** als **student** eine SSH-Sitzung zu **serverb**.
2. Stellen Sie auf **serverb** sicher, dass die Passwörter der neu erstellten Benutzer alle 30 Tage geändert werden müssen.
3. Erstellen Sie die neue Gruppe **consultants** mit der GID **35000**.
4. Konfigurieren Sie Administratorrechte für alle Mitglieder von **consultants**, damit sie Befehle als beliebige Benutzer ausführen können.
5. Erstellen Sie die Benutzer **consultant1**, **consultant2** und **consultant3** mit **consultants** als ihre Zusatzgruppe.
6. Legen Sie die Konten **consultant1**, **consultant2** und **consultant3** so fest, dass sie in 90 Tagen ab dem aktuellen Datum ablaufen.
7. Ändern Sie die Passwortrichtlinie für das Konto **consultant2** so, dass alle 15 Tage das Passwort geändert werden muss.

8. Setzen Sie außerdem durch, dass die Benutzer **consultant1**, **consultant2** und **consultant3** Ihre Passwörter bei der ersten Anmeldung ändern müssen.

## Bewertung

Führen Sie auf **workstation** den Befehl **lab users-review grade** aus, um den Erfolg dieser Übung zu bestätigen.

```
[student@workstation ~]$ lab users-review grade
```

## Beenden

Führen Sie auf **workstation** das Skript **lab users-review finish** aus, um die praktische Übung abzuschließen. Dieses Skript löscht die während der Übung erstellten Benutzerkonten und Dateien, um eine bereinigte Umgebung sicherzustellen.

```
[student@workstation ~]$ lab users-review finish
```

Damit ist die praktische Übung abgeschlossen.

## ► Lösung

# Verwalten lokaler Benutzer und Gruppen

### Leistungscheckliste

In dieser praktischen Übung legen Sie eine lokale Standardrichtlinie für Passwörter fest, erstellen eine Zusatzgruppe für drei Benutzer, lassen zu, dass diese Gruppe als **root** mit **sudo** Befehle ausführen kann, und ändern die Passwortrichtlinie für einen Benutzer.

### Ergebnisse

Es werden folgende Fähigkeiten vermittelt:

- Festlegen einer Standardrichtlinie für den Passwortablauf für das Passwort des lokalen Benutzers
- Erstellen einer Gruppe und Verwenden dieser Gruppe als Zusatzgruppe für neue Benutzer
- Erstellen von drei neuen Benutzern mit der neuen Gruppe als Zusatzgruppe
- Konfigurieren der Gruppenmitglieder der Zusatzgruppe, damit sie Befehle als beliebiger Benutzer mit **sudo** ausführen können
- Festlegen einer benutzerspezifischen Richtlinie für den Passwortablauf

### Bevor Sie Beginnen

Melden Sie sich als Benutzer **student** mit dem Passwort **student** bei **workstation** an.

Führen Sie auf **workstation** das Skript **lab users-review start** aus, um diese Übung zu beginnen. Dieses Skript erstellt die erforderlichen Dateien, um sicherzustellen, dass die Umgebung korrekt eingerichtet ist.

```
[student@workstation ~]$ lab users-review start
```

1. Öffnen Sie auf **workstation** als **student** eine SSH-Sitzung zu **serverb**.

```
[student@workstation ~]$ ssh student@serverb
...output omitted...
[student@serverb ~]$
```

2. Stellen Sie auf **serverb** sicher, dass die Passwörter der neu erstellten Benutzer alle 30 Tage geändert werden müssen.
  - 2.1. Legen Sie **PASS\_MAX\_DAYS** in **/etc/login.defs** auf **30** fest. Verwenden Sie Administratorrechte, wenn Sie die Datei mit dem Texteditor öffnen. Sie können den Befehl **sudo vim /etc/login.defs** verwenden, um diesen Schritt auszuführen. Geben Sie als Passwort **student** ein, wenn **sudo** Sie zur Eingabe des Passworts des Benutzers **student** auffordert.

```
...output omitted...
# Password aging controls:
#
#      PASS_MAX_DAYS   Maximum number of days a password may be
#      used.
#      PASS_MIN_DAYS   Minimum number of days allowed between
#      password changes.
#      PASS_MIN_LEN    Minimum acceptable password length.
#      PASS_WARN_AGE   Number of days warning given before a
#      password expires.
#
PASS_MAX_DAYS    30
PASS_MIN_DAYS    0
PASS_MIN_LEN     5
PASS_WARN_AGE    7
...output omitted...
```

3. Erstellen Sie die neue Gruppe **consultants** mit der GID **35000**.

```
[student@serverb ~]$ sudo groupadd -g 35000 consultants
```

4. Konfigurieren Sie Administratorrechte für alle Mitglieder von **consultants**, damit sie Befehle als beliebige Benutzer ausführen können.
    - 4.1. Erstellen Sie die neue Datei **/etc/sudoers.d/consultants** mit folgendem Inhalt: Sie können den Befehl **sudo vim /etc/sudoers.d/consultants** verwenden, um diesen Schritt auszuführen.

```
%consultants  ALL=(ALL) ALL
```

  - 5. Erstellen Sie die Benutzer **consultant1**, **consultant2** und **consultant3** mit **consultants** als ihre Zusatzgruppe.
- ```
[student@serverb ~]$ sudo useradd -G consultants consultant1
[student@serverb ~]$ sudo useradd -G consultants consultant2
[student@serverb ~]$ sudo useradd -G consultants consultant3
```
6. Legen Sie die Konten **consultant1**, **consultant2** und **consultant3** so fest, dass sie in 90 Tagen ab dem aktuellen Datum ablaufen.
    - 6.1. Legen Sie das Datum 90 Tage in der Zukunft fest. Abhängig vom aktuellen Datum und der aktuellen Uhrzeit auf Ihrem System erhalten Sie verglichen mit der folgenden Ausgabe möglicherweise einen anderen Wert.

```
[student@serverb ~]$ date -d "+90 days" +%F
2019-04-28
```

  - 6.2. Legen Sie das Ablaufdatum der Konten **consultant1**, **consultant2** und **consultant3** auf denselben Wert fest, der im vorherigen Schritt ermittelt wurde.

## Kapitel 6 | Verwalten lokaler Benutzer und Gruppen

```
[student@serverb ~]$ sudo chage -E 2019-04-28 consultant1  
[student@serverb ~]$ sudo chage -E 2019-04-28 consultant2  
[student@serverb ~]$ sudo chage -E 2019-04-28 consultant3
```

7. Ändern Sie die Passwortrichtlinie für das Konto **consultant2** so, dass alle 15 Tage das Passwort geändert werden muss.

```
[student@serverb ~]$ sudo chage -M 15 consultant2
```

8. Setzen Sie außerdem durch, dass die Benutzer **consultant1**, **consultant2** und **consultant3** Ihre Passwörter bei der ersten Anmeldung ändern müssen.

- 8.1. Legen Sie den letzten Tag der Passwortänderung auf **0** fest, sodass die Benutzer gezwungen sind, das Passwort bei jeder ersten Anmeldung beim System zu ändern.

```
[student@serverb ~]$ sudo chage -d 0 consultant1  
[student@serverb ~]$ sudo chage -d 0 consultant2  
[student@serverb ~]$ sudo chage -d 0 consultant3
```

- 8.2. Melden Sie sich von **serverb** ab.

```
[student@serverb ~]$ exit  
logout  
Connection to serverb closed.
```

## Bewertung

Führen Sie auf **workstation** den Befehl **lab users-review grade** aus, um den Erfolg dieser Übung zu bestätigen.

```
[student@workstation ~]$ lab users-review grade
```

## Beenden

Führen Sie auf **workstation** das Skript **lab users-review finish** aus, um die praktische Übung abzuschließen. Dieses Skript löscht die während der Übung erstellten Benutzerkonten und Dateien, um eine bereinigte Umgebung sicherzustellen.

```
[student@workstation ~]$ lab users-review finish
```

Damit ist die praktische Übung abgeschlossen.

# Zusammenfassung

---

In diesem Kapitel erhielten Sie folgende Informationen:

- Es gibt drei Haupttypen von Benutzerkonten: Superuser, Systembenutzer und regulärer Benutzer.
- Ein Benutzer muss eine primäre Gruppe haben und kann Mitglied einer oder mehrerer Zusatzgruppen sein.
- Die drei entscheidenden Dateien mit Benutzer- und Gruppeninformationen sind **/etc/passwd**, **/etc/group** und **/etc/shadow**.
- Mit den Befehlen **su** und **sudo** können Befehle als Superuser ausgeführt werden.
- Mit den Befehlen **useradd**, **usermod** und **userdel** können Benutzer verwaltet werden.
- Mit den Befehlen **groupadd**, **groupmod** und **groupdel** können Gruppen verwaltet werden.
- Mit dem Befehl **chage** können Einstellungen für den Passwortablauf für Benutzer konfiguriert und angezeigt werden.



## Kapitel 7

# Steuern des Zugriffs auf Dateien

### Ziel

Einrichten von Linux-Dateisystemberechtigungen für Dateien und Interpretieren der Sicherheitseffekte verschiedener Berechtigungseinstellungen

### Ziele

- Auflisten der Dateisystemberechtigungen für Dateien und Verzeichnisse und Interpretieren der Auswirkungen dieser Berechtigungen auf den Zugriff von Benutzern und Gruppen
- Ändern der Berechtigungen und Eigentümerschaft von Dateien mit Befehlszeilentools
- Steuern der Standardberechtigungen neuer, von Benutzern erstellter Dateien, Erläutern der Auswirkungen besonderer Berechtigungen und Verwenden spezieller Berechtigungen und Standardberechtigungen zum Festlegen des Gruppeneigentümers von in einem bestimmten Verzeichnis erstellten Dateien

### Abschnitte

- Interpretieren der Linux-Dateisystemberechtigungen (und Test)
- Verwalten von Dateisystemberechtigungen über die Befehlszeile (und angeleitete Übung)
- Verwalten von Standardberechtigungen und Dateizugriff (und angeleitete Übung)

### Praktische Übung

Steuern des Zugriffs auf Dateien

# Interpretieren der Linux-Dateisystemberechtigungen

---

## Ziele

Nach Abschluss dieses Abschnitts sollten Sie in der Lage sein, die Dateisystemberechtigungen für Dateien und Verzeichnisse aufzulisten und die Auswirkungen dieser Berechtigungen auf den Zugriff von Benutzern und Gruppen zu interpretieren.

## Dateisystemberechtigungen unter Linux

*Dateiberechtigungen* steuern den Zugriff auf Dateien. Linux-Dateiberechtigungen sind einfach und flexibel, leicht verständlich und umsetzbar, eignen sich aber dennoch für die meisten Berechtigungsfälle.

Dateien haben drei Benutzerkategorien, für die Dateiberechtigungen gelten. Eigentümer einer Datei ist ein Benutzer; in der Regel derjenige Benutzer, der die Datei erstellt hat. Darüber hinaus ist auch eine einzelne Gruppe Eigentümer einer Datei; in der Regel die primäre Gruppe des Benutzers, der die Datei erstellt hat (dies kann jedoch geändert werden). Für den Eigentümerbenutzer, die Eigentümergruppe und alle anderen Benutzer im System, die nicht der Eigentümerbenutzer oder Mitglied der Eigentümergruppe sind, können unterschiedliche Berechtigungen festgelegt werden.

Spezifischere Berechtigungen haben Vorrang. *Benutzerberechtigungen* haben Vorgang vor *Gruppenberechtigungen*, die wiederum Vorgang vor *anderen* Berechtigungen haben.

In Abbildung 7.1 ist **joshua** Mitglied der Gruppen **joshua** und **web** und **allison** ist Mitglied der Gruppen **allison**, **wheel** und **web**. Wenn **joshua** und **allison** zusammenarbeiten müssen, sollten die Dateien der Gruppe **web** zugewiesen werden und Gruppenberechtigungen sollten entsprechend festgelegt werden.

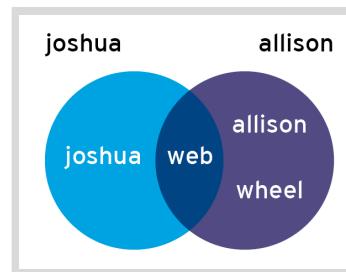


Abbildung 7.1: Beispiel einer Gruppenmitgliedschaft, um die Zusammenarbeit zu erleichtern

Es gelten drei Berechtigungskategorien: Lesen (read), Schreiben (write) und Ausführen (execute). Die folgende Tabelle enthält die Erklärungen dazu, wie diese Berechtigungen sich auf den Zugriff auf Dateien und Verzeichnisse auswirken.

### Auswirkungen von Berechtigungen auf Dateien und Verzeichnisse

| Berechtigung | Auswirkung auf Dateien          | Auswirkung auf Verzeichnisse                               |
|--------------|---------------------------------|------------------------------------------------------------|
| r (read)     | Dateiinhalt kann gelesen werden | Verzeichnisinhalt (die Dateinamen) kann aufgelistet werden |

| Berechtigung | Auswirkung auf Dateien                       | Auswirkung auf Verzeichnisse                                                                                                                                                 |
|--------------|----------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| w (write)    | Dateiinhalt kann geändert werden             | Alle Dateien im Verzeichnis können erstellt oder gelöscht werden                                                                                                             |
| x (execute)  | Dateien können als Befehle ausgeführt werden | Das Verzeichnis kann zum aktuellen Arbeitsverzeichnis werden. (Sie können <b>cd</b> ausführen, benötigen aber auch Leseberechtigung, um die gefundenen Dateien aufzulisten.) |

Benutzer haben in der Regel sowohl Lese- als auch Ausführungs berechtigungen für schreibgeschützte Verzeichnisse, damit sie das Verzeichnis auflisten und auf seine Inhalte zum Lesen zugreifen können. Wenn ein Benutzer nur Lesezugriff auf ein Verzeichnis hat, kann er die Namen der enthaltenen Dateien anzeigen. Jedoch kann er weder auf die Dateien noch auf weitere Informationen, wie Berechtigungen oder Zeitstempel, zugreifen. Wenn ein Benutzer nur Ausführungszugriff auf ein Verzeichnis hat, kann er keine Dateinamen im Verzeichnis auflisten. Wenn der Benutzer den Namen einer Datei kennt, für die er Leseberechtigung hat, kann er von außerhalb des Verzeichnisses auf den Inhalt dieser Datei zugreifen, indem er den relativen Dateinamen explizit angibt.

Eine Datei kann von jedem, der Eigentümer des Verzeichnisses ist, in dem sie sich befindet, oder Schreibberechtigungen dafür besitzt, entfernt werden, unabhängig davon, wer Eigentümer der Datei ist oder welche Berechtigungen für die Datei selbst eingetragen sind. Dies kann durch eine spezielle Berechtigung umgangen werden, die als *Sticky Bit* bezeichnet und später in diesem Kapitel behandelt wird.



### Anmerkung

Linux-Dateiberechtigungen funktionieren anders als das vom NTFS-Dateisystem für Microsoft Windows verwendete Berechtigungssystem.

Unter Linux gelten Berechtigungen nur für die Datei oder das Verzeichnis, für die sie festgelegt sind. Das heißt, Berechtigungen für ein Verzeichnis werden nicht automatisch für Unterverzeichnisse oder enthaltene Dateien übernommen. Berechtigungen für ein Verzeichnis können jedoch den Zugriff auf den Inhalt des Verzeichnisses blockieren, je nachdem, wie restriktiv sie sind.

Die Berechtigung **read** für ein Verzeichnis unter Linux entspricht etwa der Berechtigung **List folder contents** unter Windows.

Die Berechtigung **write** für ein Verzeichnis unter Linux entspricht etwa der Berechtigung **Modify** unter Windows; sie impliziert, dass Dateien und Unterverzeichnisse gelöscht werden können. Wenn unter Linux **write** und **sticky bit** für ein Verzeichnis festgelegt sind, kann es nur vom Eigentümer einer enthaltenen Datei oder eines Unterverzeichnisses gelöscht werden. Dies entspricht etwa dem Verhalten der Berechtigung **Write** unter Windows.

Der Linux-Root-Benutzer hat die Berechtigung für alle Dateien, was der Berechtigung **Full Control** unter Windows entspricht. Der Zugriff des Root-Benutzers kann jedoch durch die SELinux-Richtlinie des Systems mit Prozess- und Sicherheitskontexten eingeschränkt werden. SELinux wird in einem anderen Kurs behandelt.

## Anzeigen von Datei- und Verzeichnisberechtigungen und -eigentümerschaft

Die Option **-l** des Befehls **ls** zeigt detaillierte Informationen über Berechtigungen und Eigentümerschaft an:

```
[user@host~]$ ls -l test  
-rw-rw-r--. 1 student student 0 Feb 8 17:36 test
```

Verwenden Sie die Option **-ld** zum Anzeigen detaillierter Informationen zu einem Verzeichnis selbst und nicht zu dessen Inhalt.

```
[user@host ~]$ ls -ld /home  
drwxr-xr-x. 5 root root 4096 Jan 31 22:00 /home
```

Das erste Zeichen der ausführlichen (langen) Auflistung ist der Dateityp, der wie folgt interpretiert wird.

- **-** ist eine reguläre Datei.
- **d** ist ein Verzeichnis.
- **l** ist ein Softlink.
- Andere Zeichen stehen für Hardwaregeräte (**b** und **c**) oder andere Dateien für spezielle Zwecke (**p** und **s**).

Die nächsten neun Zeichen sind die Dateiberechtigungen. Dies sind drei Sätze von drei Zeichen: Berechtigungen, die für den Benutzer gelten, der Eigentümer der Datei ist, die Gruppe, die Eigentümer der Datei ist, und alle anderen Benutzer. Wenn **rwx** angezeigt wird, dann hat diese Kategorie alle drei Berechtigungen: Lesen, Schreiben und Ausführen. Wenn ein Buchstabe durch **-** ersetzt wurde, dann hat diese Kategorie diese Berechtigung nicht.

Der erste Name nach der Anzahl der Verknüpfungen gibt den Benutzer an, der Eigentümer der Datei ist, und der zweite Name gibt die Gruppe an, die Eigentümer der Datei ist.

Im obigen Beispiel sind also die Berechtigungen für den Benutzer **student** durch den ersten Satz von drei Zeichen angegeben. Der Benutzer **student** hat für **test** Lese- und Schreibberechtigungen, aber keine Ausführungsberechtigung.

Die Gruppe **student** wird durch den zweiten Satz von drei Zeichen angegeben: sie hat auch Lese- und Schreibberechtigungen für **test**, aber keine Ausführungsberechtigung.

Die Berechtigungen aller anderen Benutzer werden durch den dritten Satz von drei Zeichen angegeben: Sie haben nur Leseberechtigung für **test**.

Es gilt immer der spezifischste Berechtigungssatz. Wenn also der Benutzer **student** andere Berechtigungen als die Gruppe **student** hat und der Benutzer **student** auch Mitglied dieser Gruppe ist, dann gelten die Benutzerberechtigungen.

## Beispiele für Auswirkungen von Berechtigungen

Die folgenden Beispiele veranschaulichen die Wechselwirkung von Dateiberechtigungen. In diesen Beispielen werden vier Benutzer mit den folgenden Gruppenmitgliedschaften verwendet:

| Benutzer         | Gruppenmitgliedschaften       |
|------------------|-------------------------------|
| <b>operator1</b> | <b>operator1, consultant1</b> |

| Benutzer           | Gruppenmitgliedschaften       |
|--------------------|-------------------------------|
| <b>database1</b>   | <b>database1, consultant1</b> |
| <b>database2</b>   | <b>database2, operator2</b>   |
| <b>contractor1</b> | <b>contractor1, operator2</b> |

Diese Benutzer arbeiten mit Dateien im Verzeichnis **dir**. Dies ist eine lange Auflistung der Dateien in diesem Verzeichnis:

```
[database1@host dir]$ ls -la
total 24
drwxrwxr-x.. 2 database1 consultant1 4096 Apr  4 10:23 .
drwxr-xr-x.. 10 root      root       4096 Apr  1 17:34 ..
-rw-rw-r--.. 1 operator1 operator1   1024 Apr  4 11:02 lfile1
-rw-r--r--.. 1 operator1 consultant1 3144 Apr  4 11:02 lfile2
-rw-rw-r--.. 1 database1 consultant1 10234 Apr  4 10:14 rfile1
-rw-r-----.. 1 database1 consultant1 2048 Apr  4 10:18 rfile2
```

Die Option **-a** zeigt die Berechtigungen für versteckte Dateien an, einschließlich der speziellen Dateien, die zur Darstellung des Verzeichnisses und seines übergeordneten Verzeichnisses verwendet werden. In diesem Beispiel gibt **.** die Berechtigungen von **dir** selbst und **..** die Berechtigungen des übergeordneten Verzeichnisses wieder.

Welche Berechtigungen hat **rfile1**? Der Benutzer, der Eigentümer der Datei (**database1**) ist, hat Lese- und Schreibberechtigung, aber keine Ausführungs berechtigung. Die Gruppe, die Eigentümer der Datei (**consultant1**) ist, hat Lese- und Schreibberechtigung, aber keine Ausführungs berechtigung. Alle anderen Benutzer haben Lese-, aber keine Schreib- oder Ausführungs berechtigung.

In der folgenden Tabelle werden einige der Auswirkungen dieses Berechtigungssatzes für diese Benutzer erläutert:

| Auswirkung                                                                                                                              | Warum trifft das zu?                                                                                                                                        |
|-----------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Der Benutzer <b>operator1</b> kann den Inhalt von <b>rfile1</b> ändern.                                                                 | Der Benutzer <b>operator1</b> ist Mitglied der Gruppe <b>consultant1</b> und diese Gruppe hat sowohl Lese- als auch Schreibberechtigung für <b>rfile1</b> . |
| Der Benutzer <b>database1</b> kann den Inhalt der Datei <b>rfile2</b> anzeigen und ändern.                                              | Der Benutzer <b>database1</b> ist der Eigentümer der Datei <b>rfile2</b> und hat deshalb sowohl Lese- als auch Schreibzugriff darauf.                       |
| Der Benutzer <b>operator1</b> kann den Inhalt von <b>rfile2</b> anzeigen, aber nicht ändern (ohne sie zu löschen und neu zu erstellen). | Der Benutzer <b>operator1</b> ist Mitglied der Gruppe <b>consultant1</b> und diese Gruppe hat nur Lesezugriff auf <b>rfile2</b> .                           |
| Die Benutzer <b>database2</b> und <b>contractor1</b> haben keinen Zugriff auf den Inhalt von <b>rfile2</b> .                            | Andere Berechtigungen gelten für die Benutzer <b>database2</b> und <b>contractor1</b> und diese Berechtigungen enthalten weder Lese- noch Schreibzugriff.   |

| Auswirkung                                                                                                                                                           | Warum trifft das zu?                                                                                                                                                                                                                                                                |
|----------------------------------------------------------------------------------------------------------------------------------------------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>operator1</b> ist der einzige Benutzer, der den Inhalt von <b>lfile1</b> ändern kann (ohne sie zu löschen und neu zu erstellen).                                  | Benutzer und Gruppe <b>operator1</b> haben Schreibberechtigung für die Datei, andere Benutzer nicht. Aber einziges Mitglied der Gruppe <b>operator1</b> ist der Benutzer <b>operator1</b> .                                                                                         |
| Der Benutzer <b>database2</b> kann den Inhalt von <b>lfile2</b> ändern.                                                                                              | Der Benutzer <b>database2</b> ist weder der Eigentümer der Datei noch Mitglied der Gruppe <b>consultant1</b> , deshalb gelten <b>andere</b> Berechtigungen. Diese gewähren Schreibberechtigung.                                                                                     |
| Der Benutzer <b>database1</b> kann den Inhalt von <b>lfile2</b> anzeigen, aber den Inhalt von <b>lfile2</b> nicht ändern (ohne sie zu löschen und neu zu erstellen). | Der Benutzer <b>database1</b> ist Mitglied der Gruppe <b>consultant1</b> und diese Gruppe hat nur Lesezugriff für <b>lfile2</b> . Auch wenn <b>andere</b> über Schreibberechtigung verfügt, haben die Gruppenberechtigungen Vorrang.                                                |
| Der Benutzer <b>database1</b> kann <b>lfile1</b> und <b>lfile2</b> löschen.                                                                                          | Der Benutzer <b>database1</b> hat Schreibberechtigung für das Verzeichnis, das die beiden Dateien (mit . angezeigt) enthält, kann folglich jede Datei in diesem Verzeichnis löschen. Dies gilt auch dann, wenn <b>database1</b> keine Schreibberechtigung für die Datei selbst hat. |



### Literaturhinweise

Manpage **ls(1)**

(GNU Coreutils)**info coreutils**

- Abschnitt 13: Ändern von Dateiattributen

## ► Quiz

# Interpretieren der Linux-Dateisystemberechtigungen

Lesen Sie die folgenden Informationen durch und beantworten Sie die Testfragen.

Das System verfügt über vier Benutzer, die den folgenden Gruppen zugeordnet sind:

- Der Benutzer **consultant1** ist Mitglied der Gruppen **consultant1** und **database1**
- Der Benutzer **operator1** ist Mitglied der Gruppen **operator1** und **database1**
- Der Benutzer **contractor1** ist Mitglied der Gruppen **contractor1** und **contractor3**
- Der Benutzer **operator2** ist Mitglied der Gruppen **operator2** und **contractor3**

Das aktuelle Verzeichnis (.) enthält vier Dateien mit den folgenden Berechtigungsinformationen:

```
drwxrwxr-x. operator1 database1 .
-rw-rw-r--. consultant1 consultant1 lfile1
-rw-r--rw-. consultant1 database1 lfile2
-rw-rw-r--. operator1 database1 rfile1
-rw-r-----. operator1 database1 rfile2
```

- 1. Welche reguläre Datei ist Eigentum von **operator1** und kann von allen Benutzern gelesen werden?
- lfile1**
  - lfile2**
  - rfile1**
  - rfile2**
- 2. Welche Datei kann vom Benutzer **contractor1** geändert werden?
- lfile1**
  - lfile2**
  - rfile1**
  - rfile2**
- 3. Welche Datei kann vom Benutzer **operator2** nicht gelesen werden?
- lfile1**
  - lfile2**
  - rfile1**
  - rfile2**

► **4. Welche Datei ist Eigentum der Gruppe consultant1?**

- a. **lfile1**
- b. **lfile2**
- c. **rfile1**
- d. **rfile2**

► **5. Welche Dateien können vom Benutzer operator1 gelöscht werden?**

- a. **rfile1**
- b. **rfile2**
- c. Alle aufgeführten Dateien
- d. Keine der aufgeführten Dateien

► **6. Welche Dateien können vom Benutzer operator2 gelöscht werden?**

- a. **lfile1**
- b. **lfile2**
- c. Alle aufgeführten Dateien
- d. Keine der aufgeführten Dateien

## ► Lösung

# Interpretieren der Linux-Dateisystemberechtigungen

Lesen Sie die folgenden Informationen durch und beantworten Sie die Testfragen.

Das System verfügt über vier Benutzer, die den folgenden Gruppen zugeordnet sind:

- Der Benutzer **consultant1** ist Mitglied der Gruppen **consultant1** und **database1**
- Der Benutzer **operator1** ist Mitglied der Gruppen **operator1** und **database1**
- Der Benutzer **contractor1** ist Mitglied der Gruppen **contractor1** und **contractor3**
- Der Benutzer **operator2** ist Mitglied der Gruppen **operator2** und **contractor3**

Das aktuelle Verzeichnis (.) enthält vier Dateien mit den folgenden Berechtigungsinformationen:

```
drwxrwxr-x. operator1 database1 .
-rw-rw-r--. consultant1 consultant1 lfile1
-rw-r--rw-. consultant1 database1 lfile2
-rw-rw-r--. operator1 database1 rfile1
-rw-r-----. operator1 database1 rfile2
```

► 1. Welche reguläre Datei ist Eigentum von **operator1** und kann von allen Benutzern gelesen werden?

- a. **lfile1**
- b. **lfile2**
- c. **rfile1**
- d. **rfile2**

► 2. Welche Datei kann vom Benutzer **contractor1** geändert werden?

- a. **lfile1**
- b. **lfile2**
- c. **rfile1**
- d. **rfile2**

► 3. Welche Datei kann vom Benutzer **operator2** nicht gelesen werden?

- a. **lfile1**
- b. **lfile2**
- c. **rfile1**
- d. **rfile2**

► **4. Welche Datei ist Eigentum der Gruppe consultant1?**

- a. **lfile1**
- b. lfile2
- c. rfile1
- d. rfile2

► **5. Welche Dateien können vom Benutzer operator1 gelöscht werden?**

- a. rfile1
- b. **rfile2**
- c. Alle aufgeführten Dateien
- d. Keine der aufgeführten Dateien

► **6. Welche Dateien können vom Benutzer operator2 gelöscht werden?**

- a. **lfile1**
- b. **lfile2**
- c. Alle aufgeführten Dateien
- d. Keine der aufgeführten Dateien

# Verwalten von Dateisystemberechtigungen über die Befehlszeile

## Ziele

Nach Abschluss dieses Abschnitts sollten Sie in der Lage sein, die Dateiberechtigungen und -eigentümerschaft über die Befehlszeile zu ändern.

## Ändern von Datei- und Verzeichnisberechtigungen

Über die Befehlszeile können Berechtigungen mit dem Befehl **chmod**, kurz für „change mode“ (Berechtigungen werden auch als *Modus* einer Datei bezeichnet), geändert werden. Der Befehl **chmod** erwartet eine Berechtigungsanweisung, gefolgt von einer Liste mit zu ändernden Dateien oder Verzeichnissen. Die Berechtigungsanweisung kann entweder symbolisch (symbolische Methode) oder numerisch (numerische Methode) angegeben werden.

## Ändern von Berechtigungen mit der symbolischen Methode

```
chmod WhoWhatWhich file|directory
```

- Wer ist u, g, o, a (*für Benutzer, Gruppe, Andere, Alle*)
- Was ist +, -, =(*für hinzufügen, entfernen, genau festlegen*)
- Welches ist r, w, x (*für Lesen, Schreiben, Ausführen*)

Die *symbolische* Methode zur Änderung von Dateiberechtigungen verwendet zur Darstellung der verschiedenen Berechtigungsgruppen Buchstaben: **u** für Benutzer, **g** für Gruppe, **o** für Andere und **a** für Alle.

Bei der symbolischen Methode ist es nicht notwendig, einen vollständigen neuen Satz Berechtigungen festzulegen. Stattdessen können eine oder mehrere der bestehenden Berechtigungen einfach geändert werden. Verwenden Sie **+** oder **-**, um Berechtigungen hinzuzufügen oder zu entfernen, oder verwenden Sie **=**, um den gesamten Satz für eine Gruppe von Berechtigungen zu ersetzen.

Die Berechtigungen selbst werden durch einzelne Buchstaben dargestellt: **r** für Lesen, **w** für Schreiben und **x** für Ausführen. Bei Verwendung von **chmod** zum Ändern von Berechtigungen mit der symbolischen Methode wird durch ein großes **X** als Berechtigungs-Flag nur dann eine Ausführungsberechtigung hinzugefügt, wenn die Datei ein Verzeichnis ist oder die Ausführungsberechtigung bereits für Benutzer, Gruppe oder Andere festgelegt ist.



### Anmerkung

Der Befehl **chmod** unterstützt die Option **-R** für die rekursive Festlegung von Berechtigungen für die Dateien in einer vollständigen Verzeichnisstruktur. Bei Verwendung der Option **-R** kann es nützlich sein, Berechtigungen symbolisch mit der Option **X** festzulegen. Dadurch kann die Berechtigung zur Ausführung (Suche) für Verzeichnisse festgelegt werden, sodass ihre Inhalte aufgerufen werden können, ohne die Berechtigungen der meisten Dateien zu ändern. Verwenden Sie die Option **X** mit Bedacht: Wenn für eine Datei eine Ausführungsberechtigung festgelegt ist, legt **X** auch für diese Datei die angegebene Ausführungsberechtigung fest. Beispielsweise wird mit dem folgenden Befehl rekursiv Lese- und Schreibzugriff für **demodir** und alle untergeordneten Dateien für den Gruppeneigentümer festgelegt, jedoch werden keine Ausführungsberechtigungen für Verzeichnisse und Dateien festgelegt, bei denen bereits Ausführungsberechtigungen für Benutzer, Gruppe oder Andere existieren.

```
[root@host opt]# chmod -R g+rwx demodir
```

### Beispiele

- Entfernen Sie die Lese- und Schreibberechtigung für **file1** für Gruppe und Andere:

```
[user@host ~]$ chmod go-rw file1
```

- Fügen Sie die Berechtigung zum Ausführen von **file2** für alle hinzu:

```
[user@host ~]$ chmod a+x file2
```

## Ändern von Berechtigungen mit der numerischen Methode

Im Beispiel unten steht das Zeichen # für eine Ziffer.

```
chmod ### file|directory
```

- Jede Ziffer steht für Berechtigungen für eine Zugriffsebene: Benutzer, Gruppe, Andere.
- Die Ziffer wird berechnet, indem für jede Berechtigung, die Sie hinzufügen möchten, Zahlen addiert werden, 4 für Lesen, 2 für Schreiben und 1 für Ausführen.

Bei der *numerischen* Methode werden die Berechtigungen durch eine dreistellige *oktale* Zahl dargestellt. (Bei erweiterten Berechtigungen kann sie auch vierstellig sein.) Eine einzelne Oktalziffer kann einen beliebigen Wert von 0 bis 7 darstellen.

In der dreistelligen Oktaldarstellung (numerisch) von Berechtigungen steht jede Ziffer für eine Zugriffsebene; von links nach rechts: Benutzer, Gruppe und Andere. So bestimmen Sie jede Ziffer:

- Beginnen Sie mit 0.
- Wenn die Leseberechtigung für diese Zugriffsebene vorhanden sein sollte, fügen Sie 4 hinzu.
- Wenn die Schreibberechtigung vorhanden sein sollte, fügen Sie 2 hinzu.
- Wenn die Ausführungsberechtigung vorhanden sein sollte, fügen Sie 1 hinzu.

Überprüfen Sie die Berechtigungen **-rwxr-x---**. Für den Benutzer wird **rwx** wie folgt berechnet:  $4+2+1=7$ . Für die Gruppe wird **r-x** als  $4+0+1=5$  berechnet und für andere Benutzer wird **- - -** mit 0

dargestellt. Wenn Sie diese drei Werte zusammenfassen, lautet die numerische Darstellung dieser Berechtigungen 750.

Diese Berechnung kann auch in die entgegengesetzte Richtung vorgenommen werden.

Betrachten Sie die Berechtigungen 640. Bei den Benutzerberechtigungen steht 6 für Lesen (4) und Schreiben (2) und wird als **rw-** dargestellt. Bei den Gruppenberechtigungen steht 4 nur für Lesen (4) und wird als **r--** dargestellt. Die 0 für Andere enthält keine Berechtigungen (**---**), der symbolische Berechtigungssatz für die entsprechende Datei lautet also **-rw-r----**.

Erfahrene Administratoren verwenden häufig numerische Berechtigungen, weil sie schneller einzugeben und anzusagen sind, aber trotzdem die vollständige Kontrolle über alle Berechtigungen ermöglichen.

### Beispiele

- Legen Sie für **samplefile** Lese- und Schreibberechtigung für Benutzer, Leseberechtigung für Gruppe und Andere fest:

```
[user@host ~]$ chmod 644 samplefile
```

- Legen Sie für **sampledir** Lese-, Schreib- und Ausführungs berechtigungen für Benutzer, Lese- und Ausführungsrechte für Gruppe und keine Berechtigungen für Andere fest:

```
[user@host ~]$ chmod 750 sampledir
```

## Ändern der Benutzer- oder Gruppeneigentümerschaft für Dateien und Verzeichnisse

Eine neu erstellte Datei ist Eigentum des Benutzers, der die Datei erstellt hat. Standardmäßig sind neue Dateien auch das Eigentum der primären Gruppe des Benutzers, der die jeweilige Datei erstellt hat. In Red Hat Enterprise Linux ist die primäre Gruppe eines Benutzers normalerweise eine private Gruppe mit nur diesem Benutzer als Mitglied. Um den Zugriff auf eine Datei basierend auf der Gruppenmitgliedschaft zu gewähren, muss die Gruppe, die Eigentümer der Datei ist, unter Umständen geändert werden.

Nur **root** kann den Benutzer, der Eigentümer einer Datei ist, ändern. Die Gruppeneigentümerschaft kann hingegen von **root** oder dem Dateieigentümer festgelegt werden. **root** kann die Dateieigentümerschaft jeder Gruppe übertragen, aber reguläre Benutzer können die Eigentümerschaft einer Datei nur an eine Gruppe übertragen, in denen sie Mitglied sind.

Die Dateieigentümerschaft kann mit dem Befehl **chown** (Eigentümer ändern) geändert werden. Mit dem folgenden Befehl kann beispielsweise die Eigentümerschaft der Datei **test\_file** an **student** übertragen werden:

```
[root@host ~]# chown student test_file
```

**chown** kann mit der Option **-R** verwendet werden, um rekursiv die Eigentümerschaft eines gesamten Verzeichnisbaums zu ändern. Der folgende Befehl überträgt die Eigentümerschaft von **test\_dir** und sämtlichen darunterliegenden Dateien und Unterverzeichnissen an **student**:

```
[root@host ~]# chown -R student test_dir
```

## Kapitel 7 | Steuern des Zugriffs auf Dateien

Mit dem Befehl **chown** kann auch die Gruppeneigentümerschaft einer Datei geändert werden, indem ein Doppelpunkt (:) vor den Gruppennamen gesetzt wird. Beispielsweise kann mit dem folgenden Befehl die Gruppeneigentümerschaft des **test\_dir**-Verzeichnisses in **admins** geändert werden:

```
[root@host ~]# chown :admins test_dir
```

Mit dem Befehl **chown** können Eigentümer und Gruppe sogar gleichzeitig geändert werden, indem die Syntax **owner:group** verwendet wird. Beispiel: Um den Eigentümer von **test\_dir** in **visitor** und die Gruppe in **guests** zu ändern, verwenden Sie folgenden Befehl:

```
[root@host ~]# chown visitor:guests test_dir
```

Anstatt **chown** zu verwenden, ändern einige Benutzer die Gruppeneigentümerschaft mit dem Befehl **chgrp**. Dieser Befehl funktioniert genauso wie **chown**, außer dass er nur verwendet wird, um die Gruppeneigentümerschaft zu ändern, und dass der Doppelpunkt (:) vor dem Gruppennamen nicht erforderlich ist.



### Wichtig

Sie können auf Beispiele von **chown**-Befehlen stoßen, die eine alternative Syntax verwenden, bei der Eigentümer und Gruppe durch einen Punkt anstelle eines Doppelpunkts getrennt werden:

```
[root@host ~]# chown owner.group filename
```

Sie sollten diese Syntax nicht verwenden. Verwenden Sie immer einen Doppelpunkt.

Ein Punkt ist ein gültiges Zeichen in einem Benutzernamen, ein Doppelpunkt jedoch nicht. Wenn der Benutzer **enoch.root**, der Benutzer **enoch** und die Gruppe **root** auf dem System vorhanden sind, ist das Ergebnis von **chown enoch.root filename**, dass der Benutzer **enoch.root** Eigentümer von **filename** ist. Möglicherweise haben Sie versucht, die Eigentümerschaft der Datei für den Benutzer **enoch** und der Gruppe **root** festzulegen. Das kann verwirrend sein.

Wenn Sie beim gleichzeitigen Festlegen von Benutzer und Gruppe immer die Doppelpunkt-Syntax für **chown** verwenden, sind die Ergebnisse stets leicht vorhersagbar.



### Literaturhinweise

Manpages **ls(1)**, **chmod(1)**, **chown(1)** und **chgrp(1)**

## ► Angeleitete Übung

# Verwalten von Dateisystemberechtigungen über die Befehlszeile

In dieser Übung verwenden Sie Dateisystemberechtigungen, um ein Verzeichnis zu erstellen, in dem alle Mitglieder einer bestimmten Gruppe Dateien hinzufügen und löschen können.

## Ergebnisse

Sie sollten in der Lage sein, ein gemeinschaftliches Verzeichnis zu erstellen, auf das alle Mitglieder einer bestimmten Gruppe zugreifen können.

## Bevor Sie Beginnen

Melden Sie sich als Benutzer **student** mit dem Passwort **student** bei **workstation** an.

Führen Sie auf **workstation** den Befehl **lab perms-cli start** aus. Das Startskript erstellt eine Gruppe mit dem Namen **consultants** und den beiden Benutzern **consultant1** und **consultant2**.

```
[student@workstation ~]$ lab perms-cli start
```

- 1. Melden Sie sich auf **workstation** mit dem Befehl **ssh** bei **servera** als Benutzer **student** an.

```
[student@workstation ~]$ ssh student@servera  
...output omitted...  
[student@servera ~]$
```

- 2. Wechseln Sie mit dem Passwort **redhat** zum **root**-Benutzer.

```
[student@servera ~]$ su -  
Password: redhat  
[root@servera ~]#
```

- 3. Erstellen Sie mit dem Befehl **mkdir** das Verzeichnis **/home/consultants**.

```
[root@servera ~]# mkdir /home/consultants
```

- 4. Ändern Sie mit dem Befehl **chown** den Gruppeneigentümer des Verzeichnisses **consultants** in **consultants**.

```
[root@servera ~]# chown :consultants /home/consultants
```

- 5. Stellen Sie sicher, dass die Mitglieder der Gruppe **consultants** berechtigt sind, Dateien im Verzeichnis **/home/consultants** zu erstellen und zu löschen. Die Berechtigungen sollten verhindern, dass andere auf die Dateien zugreifen können.

- 5.1. Überprüfen Sie mit dem Befehl **ls**, ob die Mitglieder der Gruppe **consultants** berechtigt sind, Dateien im Verzeichnis **/home/consultants** zu erstellen und zu löschen.

```
[root@servera ~]# ls -ld /home/consultants
drwxr-xr-x. 2 root     consultants   6 Feb  1 12:08 /home/consultants
```

Beachten Sie, dass die Gruppe **consultants** derzeit über keine Schreibberechtigung verfügt.

- 5.2. Fügen Sie mit dem Befehl **chmod** der Gruppe **consultants** Schreibberechtigungen hinzu.

```
[root@servera ~]# chmod g+w /home/consultants
[root@servera ~]# ls -ld /home/consultants
drwxrwxr-x. 2 root consultants 6 Feb  1 13:21 /home/consultants
```

- 5.3. Verhindern Sie mit dem Befehl **chmod**, dass andere auf Dateien im Verzeichnis **/home/consultants** zugreifen.

```
[root@servera ~]# chmod 770 /home/consultants
[root@servera ~]# ls -ld /home/consultants
drwxrwx---. 2 root consultants 6 Feb  1 12:08 /home/consultants/
```

- 6. Beenden Sie die Root-Shell und wechseln Sie zum Benutzer **consultant1**. Das Passwort lautet **redhat**.

```
[root@servera ~]# exit
logout
[student@servera ~]$
[student@servera ~]$ su - consultant1
Password: redhat
```

- 7. Navigieren Sie zum Verzeichnis **/home/consultants** und erstellen Sie eine Datei namens **consultant1.txt**.

- 7.1. Wechseln Sie mit dem Befehl **cd** in das Verzeichnis **/home/consultants**.

```
[consultant1@servera ~]$ cd /home/consultants
```

- 7.2. Erstellen Sie mit dem Befehl **touch** eine leere Datei namens **consultant1.txt**.

```
[consultant1@servera consultants]$ touch consultant1.txt
```

- 8. Listen Sie mit dem Befehl **ls -l** den Standardbenutzer und die Gruppeneigentümerschaft der neuen Datei und ihre Berechtigungen auf.

```
[consultant1@servera consultants]$ ls -l consultant1.txt  
-rw-rw-r-- 1 consultant1 consultant1 0 Feb 1 12:53 consultant1.txt
```

- 9. Stellen Sie sicher, dass alle Mitglieder der Gruppe **consultants** die Datei **consultant1.txt** bearbeiten können. Ändern Sie die Gruppeneigentümer der Datei **consultant1.txt** in **consultants**.
- 9.1. Ändern Sie mit dem Befehl **chown** den Gruppeneigentümer der Datei **consultant1.txt** in **consultants**.

```
[consultant1@servera consultants]$ chown :consultants consultant1.txt
```

- 9.2. Listen Sie mit dem Befehl **ls** und der Option **-l** den neuen Eigentümer der Datei **consultant1.txt** auf.

```
[consultant1@servera consultants]$ ls -l consultant1.txt  
-rw-rw-r-- 1 consultant1 consultants 0 Feb 1 12:53 consultant1.txt
```

- 10. Beenden Sie die Shell und wechseln Sie zum Benutzer **consultant2**. Das Passwort lautet **redhat**.

```
[consultant1@servera consultants]$ exit  
logout  
[student@servera ~]$ su - consultant2  
Password: redhat  
[consultant2@servera ~]$
```

- 11. Navigieren Sie zum Verzeichnis **/home/consultants**. Stellen Sie sicher, dass der Benutzer **consultant2** der Datei **consultant1.txt** Inhalte hinzufügen kann. Beenden Sie die Shell.
- 11.1. Wechseln Sie mit dem Befehl **cd** in das Verzeichnis **/home/consultants**. Fügen Sie mit dem Befehl **echo** der Datei **consultant1.txt** **Text** hinzu.

```
[consultant2@servera ~]$ cd /home/consultants/  
[consultant2@servera consultants]$ echo "text" >> consultant1.txt  
[consultant2@servera consultants]$
```

- 11.2. Überprüfen Sie mit dem Befehl **cat**, ob der Text der Datei **consultant1.txt** hinzugefügt wurde.

```
[consultant2@servera consultants]$ cat consultant1.txt  
text  
[consultant2@servera consultants]$
```

- 11.3. Beenden Sie die Shell.

```
[consultant2@servera consultants]$ exit  
logout  
[student@servera ~]$
```

- 12. Melden Sie sich von **servera** ab.

```
[student@servera ~]$ exit  
logout  
Connection to servera closed.  
[student@workstation ~]$
```

## Beenden

Führen Sie auf **workstation** das Skript **lab perms-cli finish** aus, um diese Übung zu beenden.

```
[student@workstation ~]$ lab perms-cli finish
```

Hiermit ist die angeleitete Übung beendet.

# Verwalten von Standardberechtigungen und Dateizugriff

## Ziele

Am Ende dieses Abschnitts sollten Sie zu Folgendem in der Lage sein:

- Steuern der Standardberechtigungen für neue, von Benutzern erstellte Dateien
- Erläutern der Auswirkung spezieller Berechtigungen
- Verwenden spezieller Berechtigungen und Standardberechtigungen, um die Gruppeneigentümer von in einem bestimmten Verzeichnis erstellten Dateien festzulegen

## Spezielle Berechtigungen

Spezielle Berechtigungen bilden einen vierten Berechtigungstyp zusätzlich zu den grundlegenden Typen Benutzer, Gruppe und Andere. Wie der Name schon sagt, bieten diese Berechtigungen zusätzliche zugriffsbezogene Funktionen, die über die grundlegenden Berechtigungstypen hinausgehen. In diesem Abschnitt werden die Auswirkungen spezieller Berechtigungen beschrieben, die in der folgenden Tabelle zusammengefasst sind.

### Auswirkungen spezieller Berechtigungen auf Dateien und Verzeichnisse

| Spezielle Berechtigung | Auswirkung auf Dateien                                                                  | Auswirkung auf Verzeichnisse                                                                                                                                                                |
|------------------------|-----------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>u+s</b> (suid)      | Datei wird als Eigentümerbenutzer der Datei ausgeführt, nicht als ausführender Benutzer | Keine Auswirkung                                                                                                                                                                            |
| <b>g+s</b> (sgid)      | Datei wird als Eigentümergruppe der Datei ausgeführt                                    | Eigentümergruppe des Verzeichnisses wird als Eigentümergruppe neu erstellter Dateien übernommen                                                                                             |
| <b>o+t</b> (sticky)    | Keine Auswirkung                                                                        | Benutzer mit Zugriff auf das Verzeichnis können nur Dateien entfernen, deren Eigentümer sie sind; sie können keine Dateien entfernen oder speichern, deren Eigentümer andere Benutzer sind. |

Die Berechtigung `setuid` für eine ausführbare Datei bewirkt, dass Befehle unter dem Eigentümer Benutzer der Datei ausgeführt werden, nicht unter dem Benutzer, der den Befehl ausgeführt hat. Ein Beispiel hierfür ist der Befehl **passwd**:

```
[user@host ~]$ ls -l /usr/bin/passwd
-rwsr-xr-x. 1 root root 35504 Jul 16 2010 /usr/bin/passwd
```

## Kapitel 7 | Steuern des Zugriffs auf Dateien

In einer langen Liste erkennen Sie setuid-Berechtigungen an einem kleinen **s**, wo Sie normalerweise ein **x** (Ausführungsberechtigungen des Eigentümers) erwarten würden. Wenn der Eigentümer keine Ausführungsberechtigungen besitzt, wird stattdessen ein großes **S** angezeigt.

Die spezielle Berechtigung *setgid* für ein Verzeichnis bedeutet, dass in diesem Verzeichnis erstellte Dateien die Gruppeneigentümerschaft vom Verzeichnis anstatt vom erstellenden Benutzer erben. Dies wird häufig bei gemeinschaftlichen Verzeichnissen für Gruppen verwendet, um eine Datei automatisch von der privaten Standardgruppe in die gemeinsame Gruppe zu ändern, oder wenn Dateien in einem Verzeichnis immer Eigentum einer speziellen Gruppe sein sollen. Ein Beispiel dafür ist das Verzeichnis **/run/log/journal**:

```
[user@host ~]$ ls -ld /run/log/journal  
drwxr-sr-x. 3 root systemd-journal 60 May 18 09:15 /run/log/journal
```

Wenn *setgid* für eine ausführbare Datei festgelegt ist, werden Befehle ähnlich wie bei *setuid* unter der Gruppe ausgeführt, die Eigentümer dieser Datei ist, nicht unter dem Benutzer, der den Befehl ausgeführt hat. Ein Beispiel hierfür ist der Befehl **locate**:

```
[user@host ~]$ ls -ld /usr/bin/locate  
-rwx--s--x. 1 root slocate 47128 Aug 12 17:17 /usr/bin/locate
```

In einer langen Liste erkennen Sie *setgid*-Berechtigungen an einem kleinen **s**, wo Sie normalerweise ein **x** (Ausführungsberechtigungen der Gruppe) erwarten würden. Wenn die Gruppe keine Ausführungsberechtigungen besitzt, wird stattdessen ein großes **S** angezeigt.

Das *Sticky Bit* für ein Verzeichnis legt eine besondere Einschränkung beim Löschen von Dateien fest. Nur der Eigentümer der Datei (und **root**) kann Dateien in dem Verzeichnis löschen. Ein Beispiel ist **/tmp**:

```
[user@host ~]$ ls -ld /tmp  
drwxrwxrwt. 39 root root 4096 Feb 8 20:52 /tmp
```

In einer langen Liste erkennen Sie *sticky*-Berechtigungen an einem kleinen **t**, wo Sie normalerweise ein **x** (Ausführungsberechtigungen von anderen) erwarten würden. Wenn andere keine Ausführungsberechtigungen besitzen, wird stattdessen ein großes **T** angezeigt.

### Festlegen spezieller Berechtigungen

- Symbolisch: *setuid* = **u+s**; *setgid* = **g+s**; *sticky* = **o+t**
- Numerisch (vierte vorangehende Zahl): *setuid* = 4; *setgid* = 2; *sticky* = 1

### Beispiele

- Fügen Sie das *setgid*-Bit für **directory** hinzu:

```
[user@host ~]# chmod g+s directory
```

- Setzen Sie das *setgid*-Bit und fügen Sie Berechtigungen zum Lesen/Schreiben/Ausführen für Benutzer und Gruppe ohne Zugriff für andere Benutzer auf **directory** hinzu:

```
[user@host ~]# chmod 2770 directory
```

## Standarddateiberechtigungen

Wenn Sie eine neue Datei oder ein neues Verzeichnis erstellen, werden anfängliche Berechtigungen zugewiesen. Es gibt zwei Aspekte, die diese anfänglichen Berechtigungen beeinflussen. Der erste bezieht sich darauf, ob Sie eine reguläre Datei oder ein Verzeichnis erstellen. Der zweite betrifft die aktuelle *Umask*.

Wenn Sie ein neues Verzeichnis erstellen, weist das Betriebssystem diesem zunächst die oktalen Berechtigungen 0777 zu (**drwxrwxrwx**) zu. Wenn Sie eine neue reguläre Datei erstellen, weist das Betriebssystem ihr die oktalen Berechtigungen 0666 zu (**-rw-rw-rw-**) zu. Sie müssen einer regulären Datei immer explizit Ausführungsberechtigungen hinzufügen. Dies erschwert es einem Angreifer, einen Netzwerkservice zu kompromittieren, indem er eine neue Datei erstellt und sofort als Programm ausführt.

Die Shell-Sitzung legt allerdings auch eine Umask fest, um die anfänglich festgelegten Berechtigungen weiter einzuschränken. Diese Unmask ist eine oktale Bitmaske, mit der Berechtigungen neuer, von einem Prozess erstellter Dateien und Verzeichnisse gelöscht werden. Wird ein Bit in der Umask gesetzt, wird die entsprechende Berechtigung in den neuen Dateien gelöscht. Beispielsweise löscht die Umask 0002 das Schreiben-Bit für andere Benutzer. Die führenden Nullen bedeuten, dass die speziellen, Benutzer- und Gruppenberechtigungen nicht gelöscht wurden. Eine Umask mit dem Wert 0077 löscht alle Gruppen- und anderen Berechtigungen von neu erstellten Dateien.

Der Befehl **umask** ohne Argumente zeigt den aktuellen Wert der Shell-Umask an:

```
[user@host ~]$ umask  
0002
```

Verwenden Sie den Befehl **umask** mit einem einstelligen numerischen Argument, um die Umask der aktuellen Shell zu ändern. Das numerische Argument sollte ein oktaler Wert entsprechend des neuen Umask-Werts sein. Sie können führende Nullen in der Umask weglassen.

Die Systemstandards für Umask-Werte von Bash-Shell-Benutzern werden in den Dateien **/etc/profile** und **/etc/bashrc** definiert. Benutzer können die Standardwerte des Systems in den Dateien **.bash\_profile** und **.bashrc** in ihren Benutzerverzeichnissen überschreiben.

### Beispiel für Umask

Im folgenden Beispiel wird erläutert, wie sich die Umask auf die Berechtigungen von Dateien und Verzeichnissen auswirkt. Sehen Sie sich die standardmäßigen Umask-Berechtigungen für Dateien und Verzeichnisse in der aktuellen Shell an. Der Eigentümer und die Gruppe verfügen über Lese- und Schreibberechtigung für Dateien, während andere für das Lesen konfiguriert sind. Der Eigentümer und die Gruppe verfügen über Berechtigungen zum Lesen, Schreiben und Ausführen für Verzeichnisse. Die einzige Berechtigung für andere ist Lesen.

```
[user@host ~]$ umask  
0002  
[user@host ~]$ touch default  
[user@host ~]$ ls -l default.txt  
-rw-rw-r--. 1 user user 0 May  9 01:54 default.txt  
[user@host ~]$ mkdir default  
[user@host ~]$ ls -ld default  
drwxrwxr-x. 2 user user 0 May  9 01:54 default
```

## Kapitel 7 | Steuern des Zugriffs auf Dateien

Wenn Sie den Umask-Wert auf 0 setzen, ändern sich die Dateiberechtigungen für andere von Lesen in Lesen und Schreiben. Die Verzeichnisberechtigungen für andere ändern sich von Lesen und Ausführen zu Lesen, Schreiben und Ausführen.

```
[user@host ~]$ umask 0
[user@host ~]$ touch zero.txt
[user@host ~]$ ls -l zero.txt
-rw-rw-rw-. 1 user user 0 May  9 01:54 zero.txt
[user@host ~]$ mkdir zero
[user@host ~]$ ls -ld zero
drwxrwxrwx. 2 user user 0 May  9 01:54 zero
```

Um alle Datei- und Verzeichnisberechtigungen für andere zu maskieren, setzen Sie den umask-Wert auf 007.

```
[user@host ~]$ umask 007
[user@host ~]$ touch seven.txt
[user@host ~]$ ls -l seven.txt
-rw-rw----. 1 user user 0 May  9 01:55 seven.txt
[user@host ~]$ mkdir seven
[user@host ~]$ ls -ld seven
drwxrwx---. 2 user user 0 May  9 01:54 seven
```

Eine Umask von 027 stellt sicher, dass neue Dateien über Lese- und Schreibberechtigungen für den Benutzer und Leseberechtigungen für die Gruppe verfügen. Neue Verzeichnisse haben Lese- und Schreibzugriff für die Gruppe und keine Berechtigungen für andere.

```
[user@host ~]$ umask 027
[user@host ~]$ touch two-seven.txt
[user@host ~]$ ls -l two-seven.txt
-rw-r-----. 1 user user 0 May  9 01:55 two-seven.txt
[user@host ~]$ mkdir two-seven
[user@host ~]$ ls -ld two-seven
drwxr-x---. 2 user user 0 May  9 01:54 two-seven
```

Die Standard-Umask für Benutzer wird von den Shell-Startskripten festgelegt. Wenn die UID Ihres Benutzerkontos 200 oder höher ist und Ihr Benutzername und der Name Ihrer primärer Gruppe identisch sind, wird Ihnen standardmäßig die Umask 002 zugewiesen. Andernfalls lautet Ihre Umask 022.

Als **root** können Sie dies ändern, indem Sie ein Shell-Startskript mit dem Namen **/etc/profile.d/local-umask.sh** hinzufügen, das ungefähr so aussieht wie die Ausgabe in diesem Beispiel:

```
[root@host ~]# cat /etc/profile.d/local-umask.sh
# Overrides default umask configuration
if [ $UID -gt 199 ] && [ "`id -gn`" = "`id -un`" ]; then
    umask 007
else
    umask 022
fi
```

Im obigen Beispiel wird die Umask für Benutzer mit einer UID größer als 199 und einem übereinstimmenden Benutzer- und primären Gruppennamen auf 007 und für alle anderen auf 022 festgelegt. Wenn Sie die Umask für alle auf 022 festlegen möchten, können Sie diese Datei mit nur dem folgenden Inhalt erstellen:

```
# Overrides default umask configuration  
umask 022
```

Um sicherzustellen, dass globale Umask-Änderungen wirksam werden, müssen Sie sich von der Shell abmelden und erneut anmelden. Bis zu diesem Zeitpunkt ist die in der aktuellen Shell konfigurierte Umask noch aktiv.



### Literaturhinweise

Manpages **bash(1)**, **ls(1)**, **chmod(1)** und **umask(1)**

## ► Angeleitete Übung

# Verwalten von Standardberechtigungen und Dateizugriff

In dieser Übung steuern Sie mit Umask-Einstellungen und der Berechtigung setgid die Berechtigungen für neue Dateien, die in einem Verzeichnis erstellt werden.

## Ergebnisse

Es werden folgende Fähigkeiten vermittelt:

- Erstellen eines gemeinschaftlichen Verzeichnisses, in dem die Eigentümerschaft neuer Dateien automatisch auf die Gruppe **operators** übertragen wird
- Experimentieren mit verschiedenen Umask-Einstellungen
- Anpassen der Standardberechtigungen für bestimmte Benutzer
- Überprüfen der Richtigkeit Ihrer Änderung

## Bevor Sie Beginnen

Melden Sie sich als Benutzer **student** mit dem Passwort **student** bei **workstation** an.

Führen Sie auf **workstation** den Befehl **lab perms-default start** aus. Dieser Befehl führt ein Startskript aus, das ermittelt, ob **servera** im Netzwerk erreichbar ist. Das Skript erstellt auch die Gruppe **operators** und den Benutzer **operator1** auf **servera**.

```
[student@workstation ~]$ lab perms-default start
```

- 1. Melden Sie sich mit dem Befehl **ssh** bei **servera** als Benutzer **student** an.

```
[student@workstation ~]$ ssh student@servera
...output omitted...
[student@servera ~]$
```

- 2. Wechseln Sie mit dem Befehl **su** zum Benutzer **operator1**, verwenden Sie dazu **redhat** als Passwort.

```
[student@servera ~]$ su - operator1
Password: redhat
[operator1@servera ~]$
```

- 3. Zeigen Sie mit dem Befehl **umask** den Standard-Umask-Wert des Benutzers **operator1** an.

```
[operator1@servera ~]$ umask
0002
```

- 4. Erstellen Sie ein neues Verzeichnis mit dem Namen **/tmp/shared**. Erstellen Sie im Verzeichnis **/tmp/shared** die Datei **defaults**. Sehen Sie sich die Standardberechtigungen an.
- 4.1. Erstellen Sie mit dem Befehl **mkdir** das Verzeichnis **/tmp/shared**. Listen Sie mit dem Befehl **ls -ld** die Berechtigungen des neuen Verzeichnisses auf.

```
[operator1@servera ~]$ mkdir /tmp/shared  
[operator1@servera ~]$ ls -ld /tmp/shared  
drwxrwxr-x. 2 operator1 operator1 6 Feb 4 14:06 /tmp/shared
```

- 4.2. Erstellen Sie mit dem Befehl **touch** im Verzeichnis **/tmp/shared** eine Datei mit dem Namen **defaults**.

```
[operator1@servera ~]$ touch /tmp/shared/defaults
```

- 4.3. Listen Sie mit dem Befehl **ls -l** die Berechtigungen der neuen Datei auf.

```
[operator1@servera ~]$ ls -l /tmp/shared/defaults  
-rw-rw-r--. 1 operator1 operator1 0 Feb 4 14:09 /tmp/shared/defaults
```

- 5. Ändern Sie den Gruppeneigentümer von **/tmp/shared** in **operators**. Überprüfen Sie die neue Eigentümerschaft und die Berechtigungen.
- 5.1. Ändern Sie mit dem Befehl **chown** den Gruppeneigentümer des Verzeichnisses **/tmp/shared** in **operators**.

```
[operator1@servera ~]$ chown :operators /tmp/shared
```

- 5.2. Listen Sie mit dem Befehl **ls -ld** die Berechtigungen des neuen Verzeichnisses **/tmp/shared** auf.

```
[operator1@servera ~]$ ls -ld /tmp/shared  
drwxrwxr-x. 2 operator1 operators 22 Feb 4 14:09 /tmp/shared
```

- 5.3. Erstellen Sie mit dem Befehl **touch** im Verzeichnis **/tmp/shared** eine Datei mit dem Namen **group**. Listen Sie mit dem Befehl **ls -l** die Dateiberechtigungen auf.

```
[operator1@servera ~]$ touch /tmp/shared/group  
[operator1@servera ~]$ ls -l /tmp/shared/group  
-rw-rw-r--. 1 operator1 operator1 0 Feb 4 17:00 /tmp/shared/group
```



### Anmerkung

Der Gruppeneigentümer der Datei **/tmp/shared/group** ist nicht **operators**, sondern **operator1**.

- 6. Vergewissern Sie sich, dass der Eigentümer der im Verzeichnis **/tmp/shared** erstellten Dateien die Gruppe **operators** ist.

**Kapitel 7 |** Steuern des Zugriffs auf Dateien

- 6.1. Legen Sie mit dem Befehl **chmod** die Gruppen-ID für das Verzeichnis **/tmp/shared** auf die Gruppe **operators** fest.

```
[operator1@servera ~]$ chmod g+s /tmp/shared
```

- 6.2. Erstellen Sie mit dem Befehl **touch** im Verzeichnis **/tmp/shared** eine neue Datei mit dem Namen **operations\_database.txt**.

```
[operator1@servera ~]$ touch /tmp/shared/operations_database.txt
```

- 6.3. Überprüfen Sie mit dem Befehl **ls -l**, ob die Gruppe **operators** der Gruppeneigentümer der neuen Datei ist.

```
[operator1@servera ~]$ ls -l /tmp/shared/operations_database.txt
-rw-rw-r--. 1 operator1 operators 0 Feb  4 16:11 /tmp/shared/
operations_database.txt
```

- 7. Erstellen Sie im Verzeichnis **/tmp/shared** eine neue Datei mit dem Namen **operations\_network.txt**. Zeigen Sie die Eigentümerschaft und die Berechtigungen an. Ändern Sie die **Umask** für **operator1**. Erstellen Sie eine neue Datei namens **operations\_production.txt**. Zeigen Sie die Eigentümerschaft und die Berechtigungen der Datei **operations\_production.txt** an.

- 7.1. Erstellen Sie mit dem Befehl **touch** im Verzeichnis **/tmp/shared** eine Datei mit dem Namen **operations\_network.txt**.

```
[operator1@servera ~]$ touch /tmp/shared/operations_network.txt
```

- 7.2. Listen Sie mit dem Befehl **ls -l** die Berechtigungen der Datei **operations\_network.txt** auf.

```
[operator1@servera ~]$ ls -l /tmp/shared/operations_network.txt
-rw-rw-r--. 1 operator1 operators 5 Feb  0 15:43 /tmp/shared/
operations_network.txt
```

- 7.3. Ändern Sie mit dem Befehl **umask** die Umask für den Benutzer **operator1** in 027. Überprüfen Sie mit dem Befehl **umask** die Änderung.

```
[operator1@servera ~]$ umask 027
[operator1@servera ~]$ umask
0027
```

- 7.4. Erstellen Sie mit dem Befehl **touch** im Verzeichnis **/tmp/shared/** eine neue Datei mit dem Namen **operations\_production.txt**. Überprüfen Sie mit dem Befehl **ls -l**, ob neu erstellte Dateien mit Lesezugriff für die Gruppe **operators** und ohne Zugriff für andere Benutzer erstellt wurden.

```
[operator1@servera ~]$ touch /tmp/shared/operations_production.txt  
[operator1@servera ~]$ ls -l /tmp/shared/operations_production.txt  
-rw-r----- 1 operator1 operators 0 Feb 0 15:56 /tmp/shared/  
operations_production.txt
```

- 8. Öffnen Sie ein neues Terminalfenster und melden Sie sich bei **servera** als **operator1** an.

```
[student@workstation ~]$ ssh operator1@servera  
...output omitted...  
[operator1@servera ~]$
```

- 9. Zeigen Sie den Umask-Wert für **operator1** an.

```
[operator1@servera ~]$ umask  
0002
```

- 10. Ändern Sie die Standard-Umask für den Benutzer **operator1**. Die neue Umask gestattet keinem Benutzer außerhalb der Gruppe den Zugriff. Vergewissern Sie sich, dass die Umask geändert wurde.

- 10.1. Ändern Sie mit dem Befehl **echo** die Umask für den Benutzer **operator1** in 007.

```
[operator1@servera ~]$ echo "umask 007" >> ~/.bashrc  
[operator1@servera ~]$ cat ~/.bashrc  
# .bashrc  
  
# Source global definitions  
if [ -f /etc/bashrc ]; then  
    . /etc/bashrc  
fi  
  
# Uncomment the following line if you don't like systemctl's auto-paging feature:  
# export SYSTEMD_PAGER=  
  
# User specific aliases and functions  
umask 007
```

- 10.2. Melden Sie sich ab und melden Sie sich als Benutzer **operator1** wieder an.  
Überprüfen Sie mit dem Befehl **umask**, ob die Änderung dauerhaft ist.

```
[operator1@servera ~]$ exit  
logout  
Connection to servera closed.  
[student@workstation ~]$ ssh operator1@servera  
...output omitted...  
[operator1@servera ~]$ umask  
0007
```

- 11. Beenden Sie auf **servera** alle Shells für die Benutzer **operator1** und **student**.



### Warnung

Beenden Sie alle Shells, die von **operator1** geöffnet wurden. Wenn Sie nicht alle Shells beenden, schlägt das Skript zum Beenden fehl.

```
[operator1@servera ~]$ exit  
logout  
Connection to servera closed.  
[student@workstation ~]$
```

## Beenden

Führen Sie auf **workstation** das Skript **lab perms-default finish** aus, um diese Übung zu beenden.

```
[student@workstation ~]$ lab perms-default finish
```

Hiermit ist die angeleitete Übung beendet.

## ► Praktische Übung

# Steuern des Zugriffs auf Dateien

### Leistungscheckliste

In dieser Übung Sie konfigurieren Berechtigungen für Dateien und richten ein Verzeichnis ein, in dem Benutzer einer bestimmten Gruppe Dateien auf dem lokalen Dateisystem freigeben können.

### Ergebnisse

Es werden folgende Fähigkeiten vermittelt:

- Erstellen eines Verzeichnisses, in dem Benutzer gemeinsam an Dateien arbeiten können
- Erstellen von Dateien, denen automatisch die Gruppeneigentümerschaft zugewiesen wird
- Erstellen von Dateien, auf die außerhalb der Gruppe nicht zugegriffen werden kann

### Bevor Sie Beginnen

Melden Sie sich als Benutzer **student** mit dem Passwort **student** bei **workstation** an.

Führen Sie auf **workstation** den Befehl **lab perms-review start** aus. Dieser Befehl führt ein Startskript aus, das ermittelt, ob **serverb** im Netzwerk erreichbar ist. Das Skript erstellt auch die Gruppe **techdocs** und die drei Benutzer **tech1**, **tech2** und **database1**.

```
[student@workstation ~]$ lab perms-review start
```

1. Melden Sie sich mit dem Befehl **ssh** bei **serverb** als Benutzer **student** an. Wechseln Sie auf **serverb** zu **root**, verwenden Sie dazu das Passwort **redhat**.
2. Erstellen Sie das Verzeichnis **/home/techdocs**.
3. Ändern Sie den Gruppeneigentümer des Verzeichnisses **/home/techdocs** in die Gruppe **techdocs**.
4. Stellen Sie sicher, dass die Benutzer in der Gruppe **techdocs** aktuell keine Dateien im Verzeichnis **/home/techdocs** erstellen können.
5. Legen Sie Berechtigungen für das Verzeichnis **/home/techdocs** fest. Konfigurieren Sie für das Verzeichnis **/home/techdocs** für den Eigentümer/Benutzer und die Gruppe die Berechtigungen „setgid“ (2), „read/write/execute“ (7) und keine Berechtigungen (0) für andere Benutzer.
6. Überprüfen Sie, ob die Berechtigungen ordnungsgemäß festgelegt sind.
7. Vergewissern Sie sich, dass die Benutzer in der Gruppe **techdocs** jetzt Dateien im Verzeichnis **/home/techdocs** erstellen und bearbeiten können. Benutzer, die kein Mitglied der Gruppe **techdocs** sind, können keine Dateien im Verzeichnis **/home/techdocs** bearbeiten oder erstellen. Die Benutzer **tech1** und **tech2** sind Mitglieder der Gruppe **techdocs**. Der Benutzer **database1** ist kein Mitglied dieser Gruppe.

8. Ändern Sie die globalen Anmeldeskripts. Regulären Benutzern sollte eine Umask-Einstellung zugewiesen sein, die verhindert, dass andere neue Dateien und Verzeichnisse anzeigen oder modifizieren können.
9. Melden Sie sich von **serverb** ab.

```
[student@serverb ~]$ exit  
logout  
Connection to serverb closed.
```

## Bewertung

Führen Sie auf **workstation** das Skript **lab perms-review grade** aus, um den Erfolg dieser Übung zu bestätigen.

```
[student@workstation ~]$ lab perms-review grade
```

## Beenden

Führen Sie auf **workstation** das Skript **lab perms-review finish** aus, um die praktische Übung zu beenden.

```
[student@workstation ~]$ lab perms-review finish
```

Hiermit wird die praktische Übung abgeschlossen.

## ► Lösung

# Steuern des Zugriffs auf Dateien

### Leistungscheckliste

In dieser Übung Sie konfigurieren Berechtigungen für Dateien und richten ein Verzeichnis ein, in dem Benutzer einer bestimmten Gruppe Dateien auf dem lokalen Dateisystem freigeben können.

### Ergebnisse

Es werden folgende Fähigkeiten vermittelt:

- Erstellen eines Verzeichnisses, in dem Benutzer gemeinsam an Dateien arbeiten können
- Erstellen von Dateien, denen automatisch die Gruppeneigentümerschaft zugewiesen wird
- Erstellen von Dateien, auf die außerhalb der Gruppe nicht zugegriffen werden kann

### Bevor Sie Beginnen

Melden Sie sich als Benutzer **student** mit dem Passwort **student** bei **workstation** an.

Führen Sie auf **workstation** den Befehl **lab perms-review start** aus. Dieser Befehl führt ein Startskript aus, das ermittelt, ob **serverb** im Netzwerk erreichbar ist. Das Skript erstellt auch die Gruppe **techdocs** und die drei Benutzer **tech1**, **tech2** und **database1**.

```
[student@workstation ~]$ lab perms-review start
```

1. Melden Sie sich mit dem Befehl **ssh** bei **serverb** als Benutzer **student** an. Wechseln Sie auf **serverb** zu **root**, verwenden Sie dazu das Passwort **redhat**.

```
[student@workstation ~]$ ssh student@serverb
...output omitted...
[student@serverb ~]$ su -
Password: redhat
[root@serverb ~]#
```

2. Erstellen Sie das Verzeichnis **/home/techdocs**.

- 2.1. Erstellen Sie mit dem Befehl **mkdir** ein Verzeichnis mit dem Namen **/home/techdocs**.

```
[root@serverb ~]# mkdir /home/techdocs
```

3. Ändern Sie den Gruppeneigentümer des Verzeichnisses **/home/techdocs** in die Gruppe **techdocs**.

- 3.1. Ändern Sie mit dem Befehl **chown** den Gruppeneigentümer des Verzeichnisses **/home/techdocs** in die Gruppe **techdocs**.

```
[root@serverb ~]# chown :techdocs /home/techdocs
```

4. Stellen Sie sicher, dass die Benutzer in der Gruppe **techdocs** aktuell keine Dateien im Verzeichnis **/home/techdocs** erstellen können.

- 4.1. Wechseln Sie mit dem Befehl **su** zum Benutzer **tech1**.

```
[root@serverb ~]# su - tech1  
[tech1@serverb ~]$
```

- 4.2. Erstellen Sie mit **touch** eine Datei mit dem Namen **techdoc1.txt** im Verzeichnis **/home/techdocs**.

```
[tech1@serverb ~]$ touch /home/techdocs/techdoc1.txt  
touch: cannot touch '/home/techdocs/techdoc1.txt': Permission denied
```



### Anmerkung

Beachten Sie Folgendes: Obwohl **techdocs** Eigentümer des Verzeichnisses **techdocs** ist und **tech1** Teil der **techdocs**-Gruppe ist, kann in diesem Verzeichnis keine neue Datei erstellt werden. Das liegt daran, dass die Gruppe **techdocs** über keine Schreibberechtigung verfügt. Zeigen Sie mit dem Befehl **ls -ld** die Berechtigungen an.

```
[tech1@serverb ~]$ ls -ld /home/techdocs/  
drwxr-xr-x. 2 root techdocs 6 Feb 5 16:05 /home/techdocs/
```

5. Legen Sie Berechtigungen für das Verzeichnis **/home/techdocs** fest. Konfigurieren Sie für das Verzeichnis **/home/techdocs** für den Eigentümer/Benutzer und die Gruppe die Berechtigungen „setgid“ (2), „read/write/execute“ (7) und keine Berechtigungen (0) für andere Benutzer.

- 5.1. Beenden Sie die Benutzer-Shell **tech1**.

```
[tech1@serverb ~]$ exit  
logout  
[root@serverb ~]#
```

- 5.2. Legen Sie mit dem Befehl **chmod** die Gruppenberechtigung für das Verzeichnis **/home/techdocs** fest. Konfigurieren Sie für das Verzeichnis **/home/techdocs** für den Eigentümer/Benutzer und die Gruppe die Berechtigungen „setgid“ (2), „read/write/execute“ (7) und keine Berechtigungen (0) für andere Benutzer.

```
[root@serverb ~]# chmod 2770 /home/techdocs
```

6. Überprüfen Sie, ob die Berechtigungen ordnungsgemäß festgelegt sind.

```
[root@serverb ~]# ls -ld /home/techdocs  
drwxrws---. 2 root techdocs 6 Feb 4 18:12 /home/techdocs/
```

Beachten Sie, dass die Gruppe **techdocs** jetzt über Schreibberechtigungen verfügt.

7. Vergewissern Sie sich, dass die Benutzer in der Gruppe **techdocs** jetzt Dateien im Verzeichnis **/home/techdocs** erstellen und bearbeiten können. Benutzer, die kein Mitglied der Gruppe **techdocs** sind, können keine Dateien im Verzeichnis **/home/techdocs** bearbeiten oder erstellen. Die Benutzer **tech1** und **tech2** sind Mitglieder der Gruppe **techdocs**. Der Benutzer **database1** ist kein Mitglied dieser Gruppe.
  - 7.1. Wechseln Sie zum Benutzer **tech1**. Erstellen Sie mit **touch** eine Datei mit dem Namen **techdoc1.txt** im Verzeichnis **/home/techdocs**. Beenden Sie die Benutzer-Shell **tech1**.

```
[root@serverb ~]# su - tech1
[tech1@serverb ~]$ touch /home/techdocs/techdoc1.txt
[tech1@serverb ~]$ ls -l /home/techdocs/techdoc1.txt
-rw-rw-r-- 1 tech1 techdocs 0 Feb  5 16:42 /home/techdocs/techdoc1.txt
[tech1@serverb ~]$ exit
logout
[root@serverb ~]#
```

- 7.2. Wechseln Sie zum Benutzer **tech2**. Fügen Sie mit dem Befehl **echo** der Datei **/home/techdocs/techdoc1.txt** Inhalte hinzu. Beenden Sie die Benutzer-Shell **tech2**.

```
[root@serverb ~]# su - tech2
[tech2@serverb ~]$ cd /home/techdocs
[tech2@serverb techdocs]$ echo "This is the first tech doc." > techdoc1.txt
[tech2@serverb techdocs]$ exit
logout
[root@serverb ~]#
```

- 7.3. Wechseln Sie zum Benutzer **database1**. Fügen Sie mit dem Befehl **echo** der Datei **/home/techdocs/techdoc1.txt** Inhalte hinzu. Daraufhin wird die Meldung **Permission Denied** (Zugang verweigert) angezeigt. Vergewissern Sie sich mit dem Befehl **ls -l**, dass **database1** keinen Zugriff auf die Datei hat. Beenden Sie die Benutzer-Shell **database1**.

Der folgende **echo**-Befehl ist sehr lang und sollte als eine einzelne Zeile eingegeben werden.

```
[root@serverb ~]# su - database1
[database1@serverb ~]$ echo "This is the first tech doc." >> /home/techdocs/
techdoc1.txt
-bash: /home/techdocs/techdoc1.txt: Permission denied
[database1@serverb ~]$ ls -l /home/techdocs/techdoc1.txt
ls: cannot access '/home/techdocs/techdoc1.txt': Permission denied
[database1@serverb ~]$ exit
logout
[root@serverb ~]#
```

8. Ändern Sie die globalen Anmeldeskripts. Regulären Benutzern sollte eine Umask-Einstellung zugewiesen sein, die verhindert, dass andere neue Dateien und Verzeichnisse anzeigen oder modifizieren können.

- 8.1. Ermitteln Sie die Umask des Benutzers **student**. Wechseln Sie mit dem Befehl **su - student** zur Anmelde-Shell **student**. Beenden Sie die Shell, wenn Sie fertig sind.

```
[root@serverb ~]# su - student
[student@serverb ~]$ umask
0002
[student@serverb ~]$ exit
logout
[root@serverb ~]#
```

- 8.2. Erstellen Sie die Datei **/etc/profile.d/local-umask.sh** mit dem folgenden Inhalt, um die Umask für Benutzer mit einer UID größer als **199** und einem übereinstimmenden Benutzer- und primären Gruppennamen auf **007** und für alle anderen auf **022** festzulegen.

```
# Overrides default umask configuration
if [ $UID -gt 199 ] && [ "`id -gn`" = "`id -un`" ]; then
    umask 007
else
    umask 022
fi
```

- 8.3. Melden Sie sich von der Shell ab und wieder an als **student** an, um zu überprüfen, ob die globale Umask in **007** geändert wurde.

```
[root@serverb ~]# exit
logout
[student@serverb ~]$ exit
logout
Connection to serverb closed.
[student@workstation ~]$ ssh student@serverb
...output omitted...
[student@serverb ~]$ umask
0007
```

9. Melden Sie sich von **serverb** ab.

```
[student@serverb ~]$ exit
logout
Connection to serverb closed.
```

## Bewertung

Führen Sie auf **workstation** das Skript **lab perms-review grade** aus, um den Erfolg dieser Übung zu bestätigen.

```
[student@workstation ~]$ lab perms-review grade
```

## Beenden

Führen Sie auf **workstation** das Skript **lab perms-review finish** aus, um die praktische Übung zu beenden.

```
[student@workstation ~]$ lab perms-review finish
```

Hiermit wird die praktische Übung abgeschlossen.

# Zusammenfassung

---

In diesem Kapitel erhielten Sie folgende Informationen:

- Es gibt drei Kategorien, für die Dateiberechtigungen gelten. Ein Benutzer, eine einzelne Gruppe und andere Benutzer können Eigentümer einer Datei sein. Es gilt immer die spezifischste Berechtigung. Benutzer-Berechtigungen haben Vorrang vor Gruppen-Berechtigungen und Gruppen-Berechtigungen haben Vorrang vor anderen Berechtigungen.
- Mit der Option **-l** des Befehls **ls** wird die Dateiauflistung um die Dateiberechtigungen und den Eigentümer der Dateien erweitert.
- Der Befehl **chmod** ändert Dateiberechtigungen von der Befehlszeile aus. Es gibt zwei Methoden, um Berechtigungen darzustellen: symbolisch (Buchstaben) und numerisch (Ziffern).
- Der Befehl **chown** ändert die Eigentümerschaft einer Datei. Die Option **-R** ändert die Eigentümerschaft eines Verzeichnisbaums rekursiv.
- Der Befehl **umask** ohne Argumente zeigt den aktuellen Umask-Wert der Shell an. Jeder Prozess im System hat eine Umask. Die Umask-Standardwerte von Bash-Shell-Benutzern werden in den Dateien **/etc/profile** und **/etc/bashrc** definiert.

## Kapitel 8

# Überwachen und Verwalten von Linux-Prozessen

### Ziel

Evaluieren und Steuern von auf einem Red Hat Enterprise Linux-System ausgeführten Prozessen

### Ziele

- Abrufen von Informationen zu auf dem System ausgeführten Programmen zum Ermitteln und Steuern von Status, Ressourcennutzung und Eigentümerschaft
- Verwalten mehrerer, von derselben Terminalsitzung gestarteter Prozesse mit der Bash-Jobsteuerung
- Steuern und Beenden von nicht mit der Shell verbundenen Prozessen und erzwungenes Beenden von Benutzersitzungen und -prozessen
- Beschreiben der durchschnittlichen Systemauslastung und Ermitteln von Prozessen mit hohem Ressourcenverbrauch auf einem Server

### Abschnitte

- Auflisten von Prozessen (und Test)
- Steuern von Jobs (und angeleitete Übung)
- Beenden von Prozessen (und angeleitete Übung)
- Überwachen der Prozessaktivität (und angeleitete Übung)

### Praktische Übung

Überwachen und Verwalten von Linux-Prozessen

# Auflisten von Prozessen

## Ziele

Nach Abschluss dieses Abschnitts, sollten Sie in der Lage sein, Informationen zu auf einem System ausgeführten Programmen zum Ermitteln und Steuern des Status, der Ressourcennutzung und der Eigentümerschaft abzurufen.

## Definition eines Prozesses

Ein Prozess ist eine laufende Instanz einer gestarteten ausführbaren Datei. Ein Prozess besteht aus Folgendem:

- Einem Adressraum von zugewiesenem Arbeitsspeicher
- Sicherheitseigenschaften wie Eigentümeranmeldedaten und -berechtigungen
- Einem oder mehreren Ausführungs-Threads von Programmcode
- Prozessstatus

Die *Umgebung* eines Prozesses beinhaltet Folgendes:

- Lokale und globale Variablen
- Einen aktuellen Scheduler-Kontext
- Zugewiesene Systemressourcen wie Dateideskriptoren und Netzwerkports

Ein bestehender (*übergeordneter*) Prozess dupliziert seinen eigenen Adressraum (**fork**), um eine neue (*untergeordnete*) Prozessstruktur zu erstellen. Jeder neue Prozess erhält aus Nachverfolgungs- und Sicherheitsgründen eine eindeutige *Prozess-ID* (PID). Die PID und die *ID des übergeordneten Prozesses* (PPID) sind Elemente der neuen Prozessumgebung. Jeder Prozess kann untergeordnete Prozesse erstellen. Alle Prozesse sind auf einem Red Hat Enterprise Linux 8-System vom ersten Systemprozess **systemd** abgeleitet.

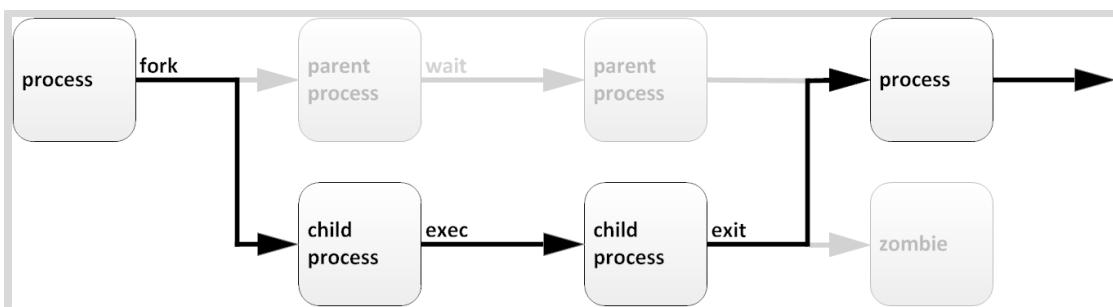


Abbildung 8.1: Prozess-Lebenszyklus

Über die Routine `fork` erbt ein untergeordneter Prozess die Sicherheitsmerkmale, alte und aktuelle Dateideskriptoren, Port- und Ressourcenberechtigungen, Umgebungsvariablen und den Programmcode. Ein untergeordneter Prozess kann dann eigenen Programmcode ausführen (`exec`). Im Normalfall befindet sich ein übergeordneter Prozess im *Ruhezustand*, während der untergeordnete Prozess ausgeführt wird, und stellt die Anforderung (`wait`), bei Abschluss des untergeordneten Prozesses benachrichtigt zu werden. Bei Beendigung hat der untergeordnete

Prozess seine Ressourcen und seine Umgebung bereits geschlossen oder verworfen. Die einzige verbleibende Ressource, *Zombie* genannt, ist ein Eintrag in der Prozesstabelle. Der übergeordnete Prozess erhält bei Beendigung des untergeordneten Prozesses das Signal zur Aktivität, bereinigt in der Prozesstabelle den Eintrag des untergeordneten Prozesses und gibt so die letzte Ressource des untergeordneten Prozesses frei. Der übergeordnete Prozess setzt dann die Ausführung des eigenen Programmcodes fort.

## Beschreiben von Prozessstatus

In einem Multitasking-Betriebssystem kann jede CPU (oder jeder CPU-Kern) zu jedem Zeitpunkt einen Prozess verarbeiten. Während ein Prozess ausgeführt wird, ändern sich dessen unmittelbare Anforderungen an die CPU-Zeit und die Ressourcenzuweisung. Prozesse erhalten einen *Status*, der sich den aktuellen Anforderungen entsprechend ändert.

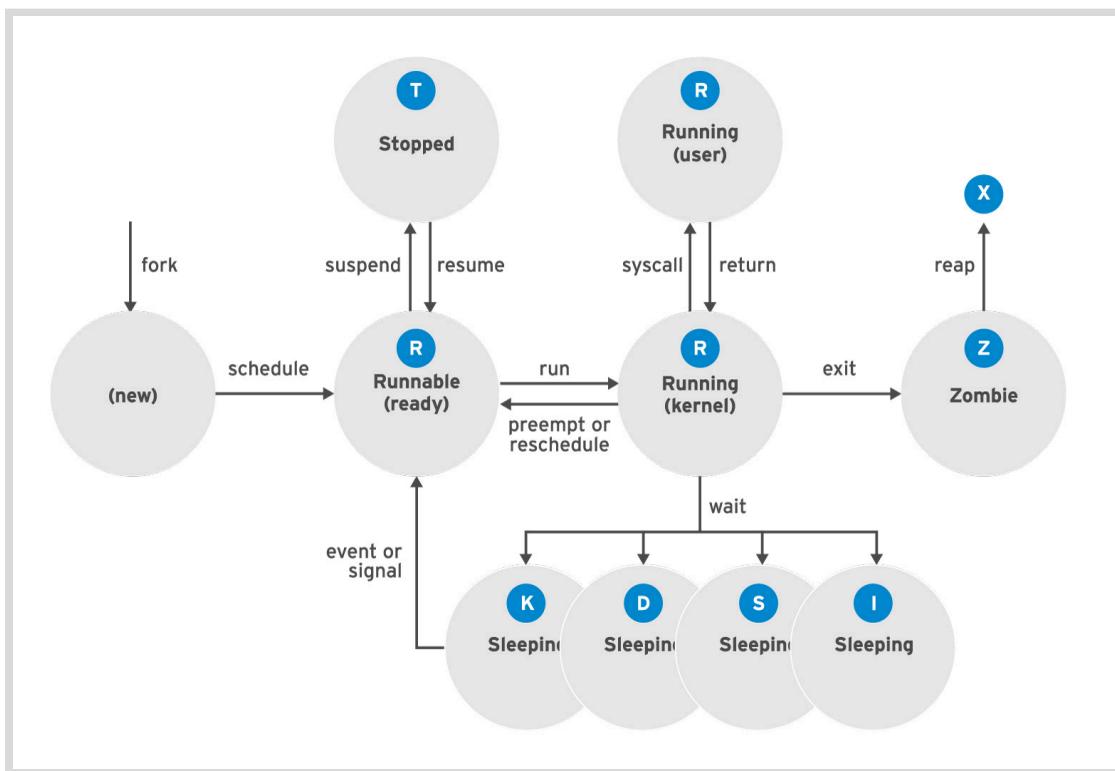


Abbildung 8.2: Linux-Prozessstatus

Linux-Prozessstatus sind im vorhergehenden Diagramm dargestellt und werden in der folgenden Tabelle näher beschrieben.

### Linux-Prozessstatus

| Name    | Flag | Statusname und Beschreibung definiert durch Kernel                                                                                                                                                                                                                                                                                                  |
|---------|------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Running | R    | TASK_RUNNING: Der Prozess wird entweder auf einer CPU ausgeführt oder wartet darauf, ausgeführt zu werden. Prozesse können Benutzerroutinen oder Kernel-Routinen (Systemaufrufe) ausführen oder in die Warteschlange gestellt werden, wo sie zur Ausführung bereit stehen, wenn sie sich im Status <i>Running</i> (oder <i>Runnable</i> ) befinden. |

| Name     | Flag | Statusname und Beschreibung definiert durch Kernel                                                                                                                                                                                                                                                                                                                                          |
|----------|------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Sleeping | S    | TASK_INTERRUPTIBLE: Der Prozess wartet auf eine Bedingung wie eine Hardware-Anforderung, den Zugang zu Systemressourcen oder ein Signal. Sobald ein Ereignis oder ein Signal die Bedingung erfüllt, wechselt der Prozess wieder in den Status <i>Running</i> .                                                                                                                              |
|          | D    | TASK_UNINTERRUPTIBLE: Dieser Prozess befindet sich ebenfalls im <i>Ruhezustand</i> , reagiert aber im Gegensatz zum Status <b>S</b> nicht auf Signale. Dieser Status wird verwendet, wenn eine Unterbrechung des Prozesses unvorhergesehene Folgen haben kann.                                                                                                                              |
|          | K    | TASK_KILLABLE: Dieser Status entspricht dem nicht unterbrechbaren Status <b>D</b> . Eine wartende Aufgabe kann hier jedoch auf das Signal zur vollständigen Beendigung reagieren. Dienstprogramme zeigen <i>beendbare</i> Prozesse häufig als Status <b>D</b> an.                                                                                                                           |
|          | I    | TASK_REPORT_IDLE: Eine Teilmenge des Status <b>D</b> . Der Kernel zählt diese Prozesse bei der Berechnung der durchschnittlichen Auslastung nicht mit. Wird für Kernel-Threads verwendet. Die Flags TASK_UNINTERRUPTIBLE und TASK_NOLOAD sind festgelegt. Ähnlich wie bei TASK_KILLABLE ist dies auch eine Teilmenge des Status <b>D</b> . Dieser Status akzeptiert Signale zur Beendigung. |
| Stopped  | T    | TASK_STOPPED: Der Prozess wurde <i>angehalten</i> (ausgesetzt). Dies erfolgt im Regelfall durch ein Signal eines Benutzers oder eines anderen Prozesses. Der Prozess kann durch ein weiteres Signal fortgesetzt und somit wieder <i>ausgeführt</i> werden.                                                                                                                                  |
|          | T    | TASK_TRACED: Ein Prozess, der gerade gedebuggt wird, ist für den Moment <i>angehalten</i> und weist dadurch ebenfalls den Status <b>T</b> auf.                                                                                                                                                                                                                                              |
| Zombie   | Z    | EXIT_ZOMBIE: Ein untergeordneter Prozess sendet beim Beenden ein Signal an den übergeordneten Prozess. Alle Ressourcen mit Ausnahme der Prozess-ID (PID) werden freigegeben.                                                                                                                                                                                                                |
|          | X    | EXIT_DEAD: Sobald der übergeordnete Prozess die verbleibenden untergeordneten Prozessstrukturen bereinigt ( <i>beendet</i> ) hat, wird der Prozess endgültig freigegeben. Dieser Status tritt in Dienstprogrammen, die Prozesse auflisten, niemals auf.                                                                                                                                     |

## Bedeutung von Prozessstatus

Bei der Problembehandlung eines Systems ist es wichtig zu verstehen, wie der Kernel mit Prozessen kommuniziert und wie Prozesse miteinander kommunizieren. Bei der Prozesserstellung weist das System dem Prozess einen Status zu. In der Spalte **S** des Befehls **top** oder der Spalte **STAT** von **ps** wird der Status jedes Prozesses angezeigt. Auf einem System mit einer einzelnen CPU kann gleichzeitig nur ein Prozess ausgeführt werden. Es ist möglich, dass mehrere Prozesse mit dem Status **R** vorhanden sind. Sie werden jedoch nicht alle nacheinander ausgeführt, einige befinden sich im *Wartestatus*.

```
[user@host ~]$ top
PID USER PR NI VIRT RES SHR S %CPU %MEM TIME+ COMMAND
 1 root 20 0 244344 13684 9024 S 0.0 0.7 0:02.46 systemd
 2 root 20 0 0 0 0 S 0.0 0.0 0:00.00 kthreadd
...output omitted...
```

```
[user@host ~]$ ps aux
USER PID %CPU %MEM VSZ RSS TTY STAT START TIME COMMAND
...output omitted...
root 2 0.0 0.0 0 0 ? S 11:57 0:00 [kthreadd]
student 3448 0.0 0.2 266904 3836 pts/0 R+ 18:07 0:00 ps aux
...output omitted...
```

Ein Prozess kann mit Signalen ausgesetzt, angehalten, fortgesetzt, beendet und unterbrochen werden. Signale werden im weiteren Verlauf des Kapitels ausführlicher behandelt. Signale können von anderen Prozessen, vom Kernel selbst oder von beim System angemeldeten Benutzern verwendet werden.

## Auflisten von Prozessen

Mit dem Befehl **ps** wird eine Auflistung aktueller Prozesse angezeigt. Damit können detaillierte Prozessinformationen abrufen werden, wie z. B.

- Benutzeridentifikation (UID), die die Prozessberechtigungen festlegt
- Eindeutige Prozess-ID (PID)
- Bereits verbrauchte CPU- und Echtzeit
- Speicher, den der Prozess an verschiedenen Speicherorten zugewiesen hat
- Speicherort der **stdout** des Prozesses, auch als *Kontrollterminal* bezeichnet
- Aktueller Prozessstatus



### Wichtig

Die Linux-Version des Befehls **ps** unterstützt die folgenden drei Optionsformate:

- UNIX (POSIX)-Optionen, die gruppiert werden können und denen ein Bindestrich vorausgehen muss
- BSD-Optionen, die gruppiert werden können und keinen Bindestrich beinhalten dürfen
- Lange GNU-Optionen, denen zwei Bindestriche vorausgehen müssen

Beispielsweise ist **ps -aux** nicht derselbe Befehl wie **ps aux**.

Die wahrscheinlich am häufigsten verwendete Optionsgruppe, **aux**, zeigt alle Prozesse einschließlich der Prozesse ohne Kontrollterminal an. Eine lange Auflistung (Optionen **lax**) bietet mehr technische Details, kann aber unter Umständen schneller aufgerufen werden, weil keine Benutzernamen nachgeschlagen werden. Die ähnliche UNIX-Syntax verwendet die Optionen **-ef**, um alle Prozesse aufzulisten.

```
[user@host ~]$ ps aux
USER      PID %CPU %MEM    VSZ   RSS TTY      STAT START  TIME COMMAND
root      1  0.1  0.1  51648  7504 ?        Ss   17:45  0:03 /usr/lib/systemd/
syst
root      2  0.0  0.0     0     0 ?        S    17:45  0:00 [kthreadd]
root      3  0.0  0.0     0     0 ?        S    17:45  0:00 [ksoftirqd/0]
root      5  0.0  0.0     0     0 ?        S<  17:45  0:00 [kworker/0:0H]
root      7  0.0  0.0     0     0 ?        S    17:45  0:00 [migration/0]
...output omitted...
[user@host ~]$ ps lax
F  UID  PID  PPID PRI  NI    VSZ   RSS WCHAN  STAT TTY      TIME COMMAND
4  0    1    0  20   0  51648  7504 ep_pol Ss   ?    0:03 /usr/lib/
systemd/
1  0    2    0  20   0     0  kthrea S    ?    0:00 [kthreadd]
1  0    3    2  20   0     0  smpboo S    ?    0:00 [ksoftirqd/0]
1  0    5    2  0 -20   0     0  worker S<  ?    0:00 [kworker/0:0H]
1  0    7    2 -100  -    0     0  smpboo S    ?    0:00 [migration/0]
...output omitted...
[user@host ~]$ ps -ef
UID      PID  PPID  C STIME TTY          TIME CMD
root      1    0  0 17:45 ?        00:00:03 /usr/lib/systemd/systemd --
switched-ro
root      2    0  0 17:45 ?        00:00:00 [kthreadd]
root      3    2  0 17:45 ?        00:00:00 [ksoftirqd/0]
root      5    2  0 17:45 ?        00:00:00 [kworker/0:0H]
root      7    2  0 17:45 ?        00:00:00 [migration/0]
...output omitted...
```

Standardmäßig listet der Befehl **ps** ohne ausgewählte Optionen alle Prozesse mit der effektiven Benutzer-ID (EUID) des aktuellen Benutzers auf, die zum selben Terminal gehören, auf dem der Befehl **ps** ausgeführt wurde.

- Prozesse in eckigen Klammern (für gewöhnlich ganz oben in der Auflistung) sind geplante Kernel-Threads.
- Zombies werden als **exiting** oder **defunct** angezeigt.
- Die Ausgabe von **ps** wird einmal angezeigt. Mit **top** zeigen Sie eine dynamisch aktualisierte Prozessanzeige an.
- **ps** kann im Baumformat dargestellt werden, sodass Sie die Beziehungen zwischen übergeordneten und untergeordneten Prozessen anzeigen können.
- Die Standardausgabe wird nach der Prozess-ID-Nummer sortiert. Auf den ersten Blick scheint dies eine chronologische Reihenfolge zu sein. Der Kernel verwendet jedoch Prozess-IDs wieder, sodass die Reihenfolge weniger strukturiert ist, als sie erscheint. Verwenden Sie zum Sortieren die Option **-o** oder **--sort**. Die Anzeige entspricht der Systemprozesstabelle, die Tabellenzeilen wiederverwendet, wenn Prozesse beendet oder neu erstellt werden. Die Liste kann chronologisch angeordnet sein. Dies muss aber nicht der Fall sein, außer die Optionen **-o** oder **--sort** werden explizit verwendet.



### Literaturhinweise

**info libc signal** (*GNU C Library Reference Manual*)

- Abschnitt 24: Signalverarbeitung

**info libc processes** (*Referenzhandbuch für die GNU C Library*)

- Abschnitt 26: Prozesse

Manpages **ps(1)** und **signal(7)**

## ► Quiz

# Auflisten von Prozessen

Wählen Sie die richtigen Antworten auf die folgenden Fragen aus:

► 1. Welcher Zustand stellt einen Prozess dar, der angehalten oder ausgesetzt wurde?

- a. D
- b. R
- c. S
- d. T
- e. Z

► 2. Welcher Zustand stellt einen Prozess dar, der alle Ressourcen außer der PID freigegeben hat?

- a. D
- b. R
- c. S
- d. T
- e. Z

► 3. Welchen Prozess verwendet ein übergeordneter Prozess, um einen neuen untergeordneten Prozess zu erstellen?

- a. exec
- b. fork
- c. zombie
- d. syscall
- e. reap

► 4. Welcher Zustand stellt einen Prozess dar, der sich im Ruhezustand befindet, bis eine Bedingung erfüllt ist?

- a. D
- b. R
- c. S
- d. T
- e. Z

## ► Lösung

# Auflisten von Prozessen

Wählen Sie die richtigen Antworten auf die folgenden Fragen aus:

- ▶ 1. Welcher Zustand stellt einen Prozess dar, der angehalten oder ausgesetzt wurde?
  - a. D
  - b. R
  - c. S
  - d. T
  - e. Z
  
- ▶ 2. Welcher Zustand stellt einen Prozess dar, der alle Ressourcen außer der PID freigegeben hat?
  - a. D
  - b. R
  - c. S
  - d. T
  - e. Z
  
- ▶ 3. Welchen Prozess verwendet ein übergeordneter Prozess, um einen neuen untergeordneten Prozess zu erstellen?
  - a. exec
  - b. fork
  - c. zombie
  - d. syscall
  - e. reap
  
- ▶ 4. Welcher Zustand stellt einen Prozess dar, der sich im Ruhezustand befindet, bis eine Bedingung erfüllt ist?
  - a. D
  - b. R
  - c. S
  - d. T
  - e. Z

# Steuern von Jobs

---

## Ziele

Nach Abschluss dieses Abschnitts sollten Sie in der Lage sein, mehrere, von derselben Terminalsitzung gestartete Prozesse mit der Bash-Jobsteuerung zu verwalten.

## Beschreiben von Jobs und Sitzungen

**Jobsteuerung** ist eine Funktion der Shell, über die eine einzelne Shell-Instanz mehrere Befehle ausführen und verwalten kann.

Ein *Job* ist mit jeder an einer Shell-Eingabeaufforderung eingegebenen Pipeline verbunden. Alle Prozesse in dieser Pipeline sind Teil des Jobs und Mitglieder derselben *Prozessgruppe*. Wenn nur ein Befehl an einer Shell-Eingabeaufforderung eingegeben wird, kann dies als minimale „Pipeline“ eines Befehls angesehen werden, und es wird ein Job mit nur einem Mitglied erstellt.

Es kann jeweils nur ein Job Eingaben und über die Tastatur generierte Signale aus einem bestimmten Terminalfenster lesen. Prozesse, die Teil dieses Jobs sind, sind *Vordergrundprozesse* dieses *Kontrollterminals*.

Ein *Hintergrundprozess* dieses Kontrollterminals ist Mitglied eines beliebigen anderen mit diesem Terminal verbundenen Jobs. Hintergrundprozesse eines Terminals können keine Eingaben lesen oder über die Tastatur generierte Interrupts vom Terminal empfangen, sie können jedoch möglicherweise in das Terminal schreiben. Ein Job im Hintergrund kann angehalten (unterbrochen) werden sein oder ausgeführt werden. Versucht ein ausgeführter Hintergrundjob aus dem Terminal zu lesen, wird er automatisch unterbrochen.

Jedes Terminal ist eine eigenständige *Sitzung* und kann einen Vordergrundprozess und eine beliebige Anzahl unabhängiger Hintergrundprozesse aufweisen. Ein Job ist Teil von genau einer Sitzung: nämlich der, die zu seinem Kontrollterminal gehört.

Mit dem Befehl **ps** wird der Gerätename des Kontrollterminals eines Prozesses in der Spalte **TTY** angezeigt. Einige Prozesse, wie *System-Daemons*, werden durch das System, nicht von einer Shell-Eingabeaufforderung gestartet. Diese Prozesse haben kein Kontrollterminal, sind kein Mitglied eines Jobs und können nicht in den Vordergrund geholt werden. Der Befehl **ps** zeigt ein Fragezeichen (?) in der Spalte **TTY** für diese Prozesse an.

## Ausführen von Jobs im Hintergrund

Jeder Befehl und jede Pipeline kann im Hintergrund gestartet werden, indem ein Ampersand-Zeichen (&) am Ende der Befehlszeile angehängt wird. Die Bash-Shell zeigt eine *Jobnummer* (einzigartig in der Sitzung) sowie die PID des neuen untergeordneten Prozesses an. Die Shell wartet nicht auf den untergeordneten Prozess, der beendet werden soll, sondern zeigt erneut die Shell-Eingabeaufforderung an.

```
[user@host ~]$ sleep 10000 &
[1] 5947
[user@host ~]$
```



### Anmerkung

Wird eine Befehlszeile, die eine Pipe enthält, mit einem Ampersand-Zeichen in den Hintergrund verschoben, wird die PID des letzten Befehls in der Pipeline als Ausgabe verwendet. Alle Prozesse in der Pipeline sind weiterhin Teil dieses Jobs.

```
[user@host ~]$ example_command | sort | mail -s "Sort output" &
[1] 5998
```

Mit dem Befehl **jobs** können Sie die Liste der Jobs anzeigen, die Bash für eine bestimmte Sitzung verfolgt.

```
[user@host ~]$ jobs
[1]+  Running                  sleep 10000 &
[user@host ~]$
```

Ein Hintergrundjob kann mit dem Befehl **fg** und Angabe der Job-ID (%Jobnummer) in den Vordergrund geholt werden.

```
[user@host ~]$ fg %1
sleep 10000
```

Im vorherigen Beispiel wird der Befehl **sleep** auf dem Kontrollterminal nun im Vordergrund ausgeführt. Die Shell ist jetzt wieder inaktiv, bis der untergeordnete Prozess abgeschlossen ist.

Um einen Vordergrundprozess in den Hintergrund zu verschieben, drücken Sie zuerst auf die über die Tastatur erzeugte Anforderung zum *Unterbrechen* (**Strg+z**) auf dem Terminal.

```
sleep 10000
^Z
[1]+  Stopped                  sleep 10000
[user@host ~]$
```

Der Job wird sofort in den Hintergrund verschoben und unterbrochen.

Mit dem Befehl **ps j** werden Informationen zu Jobs angezeigt. Die PID ist die eindeutige *Prozess-ID* des Prozesses. Die PPID ist die PID des diesem Prozess *übergeordneten Prozesses*, des Prozesses, der ihn gestartet (forked) hat. Die PGID ist die PID des *Prozessgruppenleiters*, in der Regel der erste Prozess in der Pipeline des Jobs. Die SID ist die PID des *Sitzungsleiters*, der bei einem Job in der Regel die interaktive Shell ist, die auf dem zugehörigen Kontrollterminal ausgeführt wird. Da der Beispielbefehl **sleep** derzeit ausgesetzt ist, lautet der Prozessstatus **T**.

```
[user@host ~]$ ps j
  PPID   PID   PGID   SID TTY      TPGID STAT   UID    TIME COMMAND
 2764  2768  2768  2768 pts/0      6377 Ss    1000   0:00 /bin/bash
 2768  5947  5947  2768 pts/0      6377 T     1000   0:00 sleep 10000
 2768  6377  6377  2768 pts/0      6377 R+    1000   0:00 ps j
```

Verwenden Sie den Befehl **bg** mit derselben Job-ID, um den ausgesetzten Prozess im Hintergrund zu starten.

```
[user@host ~]$ bg %1  
[1]+ sleep 10000 &
```

Die Shell zeigt eine Warnmeldung an, wenn Benutzer versuchen, ein Terminalfenster (eine Sitzung) mit ausgesetzten Jobs zu schließen. Wenn der Benutzer das Fenster dennoch schließt, werden die unterbrochenen Jobs beendet.



### Anmerkung

Beachten Sie das **+**-Zeichen nach **[1]** in den obigen Beispielen. Das **+**-Zeichen gibt an, dass dieser Job der aktuelle Standardjob ist. Das heißt, wenn ein Befehl verwendet wird, der ein **%Jobnummer**-Argument erwartet und keine Jobnummer angegeben ist, dann die Aktion für den Job mit dem **+**-Indikator ausgeführt wird.



### Literaturhinweise

Bash-Infoseite (*Das GNU Bash-Referenzhandbuch*)  
<https://www.gnu.org/software/bash/manual>

- Abschnitt 7: Jobsteuerung

Manpages **bash(1)**, **builtins(1)**, **ps(1)**, **sleep(1)**

## ► Angeleitete Übung

# Steuern von Jobs

In dieser Übung verwenden Sie die Jobsteuerung, um mehrere Prozesse zu starten, auszusetzen und in den Hintergrund sowie in Vordergrund zu verschieben.

## Ergebnisse

Sie sollten in der Lage sein, mit der Jobsteuerung Benutzerprozesse anzuhalten und erneut zu starten.

## Bevor Sie Beginnen

Melden Sie sich als Benutzer **student** mit dem Passwort **student** bei **workstation** an.

Führen Sie auf **workstation** den Befehl **lab processes-control start** aus. Dieses Skript stellt sicher, dass **servera** erreichbar ist.

```
[student@workstation ~]$ lab processes-control start
```

- 1. Öffnen Sie auf **workstation** zwei Terminalfenster nebeneinander. In diesem Abschnitt werden diese beiden Terminals als *links* und *rechts* bezeichnet. Melden Sie sich auf jedem Terminal mit dem Befehl **ssh** bei **servera** als Benutzer **student** an.

```
[student@workstation ~]$ ssh student@servera  
...output omitted...  
[student@servera ~]$
```

- 2. Erstellen Sie im *linken* Fenster ein neues Verzeichnis mit dem Namen **/home/student/bin**. Erstellen Sie im neuen Verzeichnis ein Shell-Skript namens **control**. Machen Sie das Skript ausführbar.
- 2.1. Erstellen Sie mit dem Befehl **mkdir** ein neues Verzeichnis mit dem Namen **/home/student/bin**.

```
[student@servera ~]$ mkdir /home/student/bin
```

- 2.2. Erstellen Sie mit dem Befehl **vim** im Verzeichnis **/home/student/bin** ein Skript mit dem Namen **control**. Drücken Sie die Taste **i**, um in den interaktiven Modus von Vim zu wechseln. Speichern Sie die Datei mit dem Befehl **:wq**.

```
[student@servera ~]$ vim /home/student/bin/control  
#!/bin/bash  
while true; do  
    echo -n "$@ " >> ~/control_outfile  
    sleep 1  
done
```



### Anmerkung

Das Skript `control` wird bis zum Ende ausgeführt. Es hängt einmal pro Sekunde Befehlszeilenargumente an die Datei `~/control_outfile` an.

- 2.3. Wandeln Sie die Datei `control` mit dem Befehl `chmod` in eine ausführbare Datei um.

```
[student@servera ~]$ chmod +x /home/student/bin/control
```

- 3. Führen Sie das `control`-Skript aus. Das Skript hängt fortgesetzt, in Abständen von einer Sekunde, das Wort „technical“ und ein Leerzeichen an die Datei `~/control_outfile` an.



### Anmerkung

Sie können das `control`-Skript ausführen, weil es sich in Ihrem `PATH` befindet und ausführbar gemacht wurde.

```
[student@servera ~]$ control technical
```

- 4. Verwenden Sie in der rechten Terminal-Shell den Befehl `tail` mit der Option `-f`, um zu überprüfen, ob der neue Prozess in die Datei `/home/student/control_outfile` schreibt.

```
[student@servera ~]$ tail -f ~/control_outfile
technical technical technical technical
...output omitted...
```

- 5. Drücken Sie in der linken Terminal-Shell **Strg+z**, um den ausgeführten Prozess auszusetzen. Die Shell zeigt die Job-ID in eckigen Klammern an. Stellen Sie im rechten Fenster sicher, dass die Prozessausgabe wirklich angehalten wurde.

```
^Z
[1]+  Stopped                  control technical
[student@servera ~]$
```

```
technical technical technical technical
...no further output...
```

- 6. Zeigen Sie in der linken Terminal-Shell die Liste `jobs` an. Denken Sie daran, dass das `+-` Zeichen den Standardjob angibt. Starten Sie den Job im Hintergrund neu. Überprüfen Sie im rechten Fenster, ob die Prozessausgabe wieder aktiv ist.

- 6.1. Zeigen Sie mit dem Befehl `jobs` die Liste der Jobs an.

```
[student@servera ~]$ jobs
[1]+  Stopped                  control technical
```

- 6.2. Starten Sie mit dem Befehl `bg` den Job `control` im Hintergrund erneut.

```
[student@servera ~]$ bg  
[1]+ control technical &
```

- 6.3. Überprüfen Sie mit dem Befehl **jobs**, ob der Job **control** wieder ausgeführt wird.

```
[student@servera ~]$ jobs  
[1]+ Running control technical &
```

- 6.4. Vergewissern Sie sich in der rechten Terminal-Shell, dass der Prozess **tail** eine Ausgabe erzeugt.

```
...output omitted...  
technical technical technical technical technical technical technical technical
```

- 7. Starten Sie in der linken Terminal-Shell zwei weitere **control**-Prozesse, um Text an die Datei **~/output** anzuhängen. Verwenden Sie das Ampersand-Zeichen (&), um die Prozesse im Hintergrund zu starten. Ersetzen Sie **technical** durch **documents** und dann durch **database**. Durch Ersetzen der Argumente können die drei Prozesse einfacher unterschieden werden.

```
[student@servera ~]$ control documents &  
[2] 6579  
[student@servera ~]$  
[student@servera ~]$ control database &  
[3] 6654
```



### Anmerkung

Die Jobnummer jedes neuen Prozesses wird in eckigen Klammern ausgegeben. Die zweite Zahl ist die eindeutige systemweite Prozess-ID-Nummer (PID) für den Prozess.

- 8. Zeigen Sie in der linken Terminal-Shell mit dem Befehl **jobs** die drei ausgeführten Prozesse an. Überprüfen Sie in der rechten Terminal-Shell, ob alle drei Prozesse Text an die Datei anhängen.

```
[student@servera ~]$ jobs  
[1] Running control technical &  
[2]- Running control documents &  
[3]+ Running control database &
```

```
...output omitted...  
technical documents database technical documents database technical documents  
database technical documents database  
...output omitted...
```

- 9. Unterbrechen Sie den Prozess **control technical**. Überprüfen Sie, ob der Prozess unterbrochen wurde. Beenden Sie den Prozess **control documents** und überprüfen Sie, ob er beendet wurde.

- 9.1. Holen Sie in der linken Terminal-Shell mit dem Befehl **fg** und der Job-ID den Prozess **control technical** in den Vordergrund. Drücken Sie **Strg+z**, um den Prozess auszusetzen. Überprüfen Sie mit dem Befehl **jobs**, ob der Prozess unterbrochen wurde.

```
[student@servera ~]$ fg %1
control technical
^Z
[1]+  Stopped                  control technical
[student@servera ~]$ jobs
[1]+  Stopped                  control technical
[2]   Running                 control documents &
[3]-  Running                 control database &
```

- 9.2. Vergewissern Sie sich in der rechten Terminal-Shell, dass der Prozess **control technical** keine weitere Ausgabe mehr sendet.

```
database documents database documents database
...no further output...
```

- 9.3. Holen Sie in der linken Terminal-Shell mit dem Befehl **fg** und der Job-ID den Prozess **control documents** in den Vordergrund. Drücken Sie **Strg+c**, um den Prozess zu beenden. Überprüfen Sie mit dem Befehl **jobs**, ob der Prozess beendet wurde.

```
[student@servera ~]$ fg %2
control documents
^C
[student@servera ~]$ jobs
[1]+  Stopped                  control technical
[3]-  Running                 control database &
```

- 9.4. Vergewissern Sie sich in der rechten Terminal-Shell, dass der Prozess **control documents** keine weitere Ausgabe mehr sendet.

```
...output omitted...
database database database database database database database
...no further output...
```

- 10. Zeigen Sie im linken Fenster mit dem Befehl **ps** und der Option **jT** die verbleibenden Jobs an. Die unterbrochenen Jobs weisen den Status **T** auf. Die anderen Hintergrundjobs befinden sich im Ruhezustand (**S**).

```
[student@servera ~]$ ps jT
  PPID  PID  PGID  SID TTY      TPGID STAT   UID    TIME COMMAND
 27277 27278 27278 27278 pts/1    28702 Ss   1000   0:00 -bash
 27278 28234 28234 27278 pts/1    28702 T    1000   0:00 /bin/bash /home/student/
bin/control technical
 27278 28251 28251 27278 pts/1    28702 S    1000   0:00 /bin/bash /home/student/
bin/control database
 28234 28316 28234 27278 pts/1    28702 T    1000   0:00 sleep 1
 28251 28701 28251 27278 pts/1    28702 S    1000   0:00 sleep 1
 27278 28702 28702 27278 pts/1    28702 R+   1000   0:00 ps jT
```

- 11. Zeigen Sie im linken Fenster mit dem Befehl **jobs** die aktuellen Jobs an. Beenden Sie den Prozess **control database** und überprüfen Sie, ob er beendet wurde.

```
[student@servera ~]$ jobs
[1]+  Stopped                  control technical
[3]-  Running                  control database &
```

Holen Sie mit dem Befehl **fg** und der Job-ID den Prozess **control database** in den Vordergrund. Drücken Sie **Strg+c**, um den Prozess zu beenden. Überprüfen Sie mit dem Befehl „**jobs**“, ob der Prozess beendet wurde.

```
[student@servera ~]$ fg %3
control database
^C
[student@servera ~]$ jobs
[1]+  Stopped                  control technical
```

- 12. Halten Sie in der rechten Terminal-Shell mit **Strg+c** den Befehl **tail** an. Löschen Sie mit dem Befehl **rm** die Datei **~/control\_outfile**.

```
...output omitted...
Ctrl+c
[student@servera ~]$ rm ~/control_outfile
```

- 13. Melden Sie sich von **servera** auf beiden Terminals ab.

```
[student@servera ~]$ exit
logout
Connection to servera closed.
```

```
[student@servera ~]$ exit
logout
Connection to servera closed.
```

## Beenden

Führen Sie auf **workstation** das Skript **lab\_processes-control\_finish** aus, um diese Übung zu beenden.

```
[student@workstation ~]$ lab processes-control finish
```

Hiermit ist die angeleitete Übung beendet.

# Beenden von Prozessen

## Ziele

Am Ende dieses Abschnitts sollten Sie zu Folgendem in der Lage sein:

- Verwenden von Befehlen zum Beenden von und Kommunizieren mit Prozessen
- Definieren der Charakteristika eines Daemon-Prozesses
- Beenden von Benutzersitzungen und -prozessen

## Steuern von Prozessen über Signale

Ein Signal ist ein Software-Interrupt eines Prozesses. Signale melden Ereignisse an ein ausgeführtes Programm. Ereignisse, die ein Signal generieren, können durch einen Fehler, ein externes Ereignis (eine I/O-Anforderung oder Zeitüberschreitung) oder die explizite Verwendung eines Befehls, der ein Signal sendet, oder durch eine Tastatureingabe ausgelöst werden.

In der folgenden Tabelle sind die grundlegenden Signale aufgeführt, die Systemadministratoren für die Routine-Prozessverwaltung verwenden. Sie können auf Signale entweder mit ihrem Kurznamen (HUP) oder vollständigen Namen (SIGHUP) verweisen.

### Grundlegende Prozessverwaltungssignale

| Signalnummer | Kurzname | Definition                      | Zweck                                                                                                                                                                                       |
|--------------|----------|---------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 1            | HUP      | Blockierung des Arbeitsablaufes | Wird zum Berichten einer Beendigung des Steuerprozesses eines Terminals verwendet. Wird auch zum Reinitialisieren eines Prozesses (Neuladen einer Konfiguration) ohne Beendigung verwendet. |
| 2            | INT      | Tastatur-Interrupt              | Führt zur Beendigung des Programms. Kann blockiert oder verarbeitet werden. Wird durch Drücken der Tastenkombination INTR ( <b>Strg+c</b> ) gesendet.                                       |
| 3            | QUIT     | Tastatur-Beendigung             | Ähnlich wie SIGINT; fügt ein Prozessabbild bei Beendigung hinzu. Wird durch Drücken der Tastenkombination QUIT ( <b>Strg+\</b> ) gesendet.                                                  |

| Signalnummer | Kurzname | Definition                       | Zweck                                                                                                                                                                                                     |
|--------------|----------|----------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 9            | KILL     | Beendigung, nicht blockierbar    | Führt zur sofortigen Programmbeendigung. Kann nicht blockiert, ignoriert oder verarbeitet werden; führt immer zur Beendigung.                                                                             |
| 15 Standard  | TERM     | Beendigung                       | Führt zur Beendigung des Programms. Kann im Gegensatz zu SIGKILL blockiert, ignoriert oder verarbeitet werden. Die „höfliche“ Art, ein Programm zur Beendigung aufzufordern; ermöglicht Selbsterneuerung. |
| 18           | CONT     | Fortführung                      | Wird zur Fortsetzung an einen Prozess gesendet, falls dieser unterbrochen wurde. Kann nicht blockiert werden. Führt selbst beim Verarbeiten zur Fortsetzung des Prozesses.                                |
| 19           | STOP     | Unterbrechung, nicht blockierbar | Unterbricht den Prozess. Kann nicht blockiert oder verarbeitet werden.                                                                                                                                    |
| 20           | TSTP     | Tastatur-Unterbrechung           | Kann im Gegensatz zu SIGSTOP blockiert, ignoriert oder verarbeitet werden. Wird durch Drücken der Tastenkombination SUSP ( <b>Strg+z</b> ) gesendet.                                                      |



### Anmerkung

Die Signalnummern variieren auf den verschiedenen Linux-Plattformen; die Signalnamen und deren Bedeutung sind allerdings standardisiert. Für Befehle sollten Sie die Signalnamen anstatt der Signalnummern verwenden. Die in diesem Abschnitt behandelten Nummern beziehen sich auf x86\_64-Systeme.

Jedes Signal hat eine *Standardaktion*. Für gewöhnlich ist das eine der folgenden:

- **Term** – Beendet ein Programm sofort
- **Core** – Erstellt ein Speicherabbild (Speicherauszug), bevor das Programm beendet wird
- **Stop** – Unterbricht ein Programm (suspend) und wartet auf dessen Fortsetzung (resume)

Programme können durch Implementieren von Handler-Routinen, die die Standardaktion eines Signals ignorieren, ersetzen oder erweitern, vorbereitet werden, auf erwartete Ereignissignale zu reagieren.

## Befehle zum Senden von Signalen durch explizite Anforderungen

Sie senden ein Signal an den aktuellen Vordergrundprozess durch folgende Tastatureingaben: Aussetzen (**Strg+z**), Beenden (**Strg+c**) oder Speicherauszug (**Strg+\**) des Prozesses. Sie müssen jedoch für das Senden eines Signals an einen Hintergrundprozess oder an einen Prozess in einer anderen Sitzung Befehle verwenden, die ein Signal senden.

Signale können entweder als Optionen per Name (z. B. **-HUP** oder **-SIGHUP**) oder per Nummer (die zugehörige **-1**) angegeben werden. Benutzer können eigene Prozesse beenden, benötigen aber root-Berechtigungen, um die Prozesse anderer Benutzer zu beenden.

Mit dem Befehl **kill** wird ein Signal an einen Prozess über die PID-Nummer gesendet. Trotz seines Namens können mit diesem Befehl alle Signale gesendet werden, nicht nur die zum Beenden von Programmen. Sie können mit dem Befehl **kill -1** die Namen und Nummern aller verfügbaren Signale anzeigen.

```
[user@host ~]$ kill -1
 1) SIGHUP      2) SIGINT      3) SIGQUIT      4) SIGILL      5) SIGTRAP
 6) SIGABRT     7) SIGBUS      8) SIGFPE       9) SIGKILL     10) SIGUSR1
11) SIGSEGV     12) SIGUSR2     13) SIGPIPE     14) SIGALRM     15) SIGTERM
16) SIGSTKFLT   17) SIGCHLD     18) SIGCONT     19) SIGSTOP     20) SIGTSTP
...output omitted...
[user@host ~]$ ps aux | grep job
5194  0.0  0.1 222448  2980 pts/1    S    16:39  0:00 /bin/bash /home/user/bin/
control job1
5199  0.0  0.1 222448  3132 pts/1    S    16:39  0:00 /bin/bash /home/user/bin/
control job2
5205  0.0  0.1 222448  3124 pts/1    S    16:39  0:00 /bin/bash /home/user/bin/
control job3
5430  0.0  0.0 221860  1096 pts/1   S+   16:41  0:00 grep --color=auto job
[user@host ~]$ kill 5194
[user@host ~]$ ps aux | grep job
user  5199  0.0  0.1 222448  3132 pts/1    S    16:39  0:00 /bin/bash /home/
user/bin/control job2
user  5205  0.0  0.1 222448  3124 pts/1    S    16:39  0:00 /bin/bash /home/
user/bin/control job3
user  5783  0.0  0.0 221860   964 pts/1   S+   16:43  0:00 grep --color=auto
job
[1]  Terminated                  control job1
[user@host ~]$ kill -9 5199
[user@host ~]$ ps aux | grep job
user  5205  0.0  0.1 222448  3124 pts/1    S    16:39  0:00 /bin/bash /home/
user/bin/control job3
user  5930  0.0  0.0 221860  1048 pts/1   S+   16:44  0:00 grep --color=auto
job
[2]- Killed                      control job2
[user@host ~]$ kill -SIGTERM 5205
user  5986  0.0  0.0 221860  1048 pts/1   S+   16:45  0:00 grep --color=auto
job
[3]+ Terminated                  control job3
```

Der Befehl **killall** kann mehrere Prozesse basierend auf ihrem Befehlsnamen signalisieren.

```
[user@host ~]$ ps aux | grep job
5194 0.0 0.1 222448 2980 pts/1 S 16:39 0:00 /bin/bash /home/user/bin/
control job1
5199 0.0 0.1 222448 3132 pts/1 S 16:39 0:00 /bin/bash /home/user/bin/
control job2
5205 0.0 0.1 222448 3124 pts/1 S 16:39 0:00 /bin/bash /home/user/bin/
control job3
5430 0.0 0.0 221860 1096 pts/1 S+ 16:41 0:00 grep --color=auto job
[user@host ~]$ killall control
[1] Terminated control job1
[2]- Terminated control job2
[3]+ Terminated control job3
[user@host ~]$
```

Verwenden Sie **pkill**, um ein Signal an einen oder mehrere Prozesse zu senden, die den Auswahlkriterien entsprechen. Auswahlkriterien können ein Befehlsname, ein Prozess eines bestimmten Benutzers oder alle systemweiten Prozesse sein. Der Befehl **pkill** umfasst erweiterte Auswahlkriterien:

- Command – Prozesse mit einem Befehlsnamen, die einem Muster entsprechen
- UID – Prozesse, die Eigentum eines Linux-Benutzerkontos (effektiv oder real) sind
- GID – Prozesse, die Eigentum eines Linux-Gruppenkontos (effektiv oder real) sind
- Parent – Untergeordnete Prozesse eines bestimmten übergeordneten Prozesses
- Terminal – Prozesse, die auf einem bestimmten Kontrollterminal ausgeführt werden

```
[user@host ~]$ ps aux | grep pkill
user 5992 0.0 0.1 222448 3040 pts/1 S 16:59 0:00 /bin/bash /home/
user/bin/control pkill1
user 5996 0.0 0.1 222448 3048 pts/1 S 16:59 0:00 /bin/bash /home/
user/bin/control pkill2
user 6004 0.0 0.1 222448 3048 pts/1 S 16:59 0:00 /bin/bash /home/
user/bin/control pkill3
[user@host ~]$ pkill control
[1] Terminated control pkill1
[2]- Terminated control pkill2
[user@host ~]$ ps aux | grep pkill
user 6219 0.0 0.0 221860 1052 pts/1 S+ 17:00 0:00 grep --color=auto
pkill
[3]+ Terminated control pkill3
[user@host ~]$ ps aux | grep test
user 6281 0.0 0.1 222448 3012 pts/1 S 17:04 0:00 /bin/bash /home/
user/bin/control test1
user 6285 0.0 0.1 222448 3128 pts/1 S 17:04 0:00 /bin/bash /home/
user/bin/control test2
user 6292 0.0 0.1 222448 3064 pts/1 S 17:04 0:00 /bin/bash /home/
user/bin/control test3
user 6318 0.0 0.0 221860 1080 pts/1 S+ 17:04 0:00 grep --color=auto
test
[user@host ~]$ pkill -U user
[user@host ~]$ ps aux | grep test
user 6870 0.0 0.0 221860 1048 pts/0 S+ 17:07 0:00 grep --color=auto
test
[user@host ~]$
```

## Abmelden von Benutzern als Administrator

Möglicherweise müssen Sie andere Benutzer aus verschiedenen Gründen abmelden. Einige der vielen Gründe beziehen sich auf Folgendes: Der Benutzer hat eine Sicherheitsverletzung begangen, der Benutzer hat möglicherweise überlastete Ressourcen, der Benutzer hat eventuell ein nicht reagierendes System oder der Benutzer hat unzulässigen Zugriff auf Materialien. In diesen Fällen müssen Sie möglicherweise die Sitzung mit Signalen administrativ beenden.

Um einen Benutzer abzumelden, ermitteln Sie zuerst die Anmeldesitzung, die beendet werden soll. Listen Sie mit dem Befehl **w** die Benutzeranmeldungen und die aktuell ausgeführten Prozesse auf. Achten Sie auf die Spalten **TTY** und **FROM**, um die zu schließenden Sitzungen zu bestimmen.

Alle Benutzeranmeldesitzungen sind einem Endgerät (TTY) zugeordnet. Wenn der Gerätename der Form **pts/N** entspricht, das Gerät ist ein *Pseudo-Terminal*, das mit einem grafischen Terminalfenster oder einer Remote-Anmeldesitzung verbunden ist. Wenn der Gerätename der Form **ttyN** entspricht, arbeitet der Benutzer auf einer Systemkonsole, einer alternativen Konsole oder einem anderen direkt verbundenen Terminalgerät.

```
[user@host ~]$ w
12:43:06 up 27 min,  5 users,  load average: 0.03, 0.17, 0.66
USER     TTY      FROM          LOGIN@    IDLE   JCPU   PCPU WHAT
root     tty2          12:26  14:58  0.04s  0.04s -bash
bob      tty3          12:28  14:42  0.02s  0.02s -bash
user     pts/1  desk.example.com 12:41   2.00s  0.03s  0.03s w
[user@host ~]$
```

Finden Sie durch Anzeigen der Sitzungsanmeldezeit heraus, wie lange ein Benutzer bereits im System ist. Für jede Sitzung werden durch den aktuellen Job verbrauchte CPU-Ressourcen, einschließlich Hintergrundaufgaben und untergeordnete Prozesse, in der Spalte **JCPU** angezeigt. Die aktuelle CPU-Auslastung durch Vordergrundprozesse befindet sich in der Spalte **PCPU**.

Sie können Signale entweder an einzelne oder alle Prozesse und Sitzungen senden. Um alle Prozesse eines Benutzers zu beenden, verwenden Sie den Befehl **pkill**. Da der Anfangsprozess einer Anmeldesitzung (*Sitzungsleiter*) dazu gedacht ist, Anfragen zur Beendigung der Sitzung zu verarbeiten und unbeabsichtigte Tastatursignale zu ignorieren, wird für das Beenden aller Prozesse und Login-Shells des Benutzers das Signal SIGKILL benötigt.



### Wichtig

SIGKILL wird von Administratoren häufig zu schnell verwendet.

Da das Signal SIGKILL weder verarbeitet noch ignoriert werden kann, führt es immer zur Beendigung. Es forciert jedoch die Beendigung, ohne dem Prozess die Möglichkeit zur Selbstbereinigung zu geben. Es wird daher empfohlen, zuerst SIGTERM zu senden, dann SIGINT zu versuchen und nur, wenn beide fehlgeschlagen, SIGKILL zu senden.

Ermitteln Sie zuerst mit **pgrep** die zu beendenen PID-Nummern. Das funktioniert ähnlich wie **pkill**, einschließlich der gleichen Optionen, außer dass **pgrep** Prozesse auflistet, anstatt sie zu beenden.

```
[root@host ~]# pgrep -l -u bob
6964 bash
6998 sleep
6999 sleep
7000 sleep
[root@host ~]# pkill -SIGKILL -u bob
[root@host ~]# pgrep -l -u bob
[root@host ~]#
```

Wenn sich Prozesse, die Aufmerksamkeit benötigen, in derselben Anmeldesitzung befinden, müssen unter Umständen nicht alle Prozesse des Benutzers beendet werden. Ermitteln Sie mit dem Befehl **w** das Kontrollterminal der Sitzung und beenden Sie dann nur die Prozesse mit derselben Terminal-ID. Sofern nicht **SIGKILL** verwendet wird, verarbeitet der Sitzungsleiter (in diesem Fall die Bash-Login-Shell) erfolgreich die Anforderung zur Beendigung und wird weiter ausgeführt. Alle anderen Prozesse der Sitzung werden jedoch beendet.

```
[root@host ~]# pgrep -l -u bob
7391 bash
7426 sleep
7427 sleep
7428 sleep
[root@host ~]# w -h -u bob
bob      tty3      18:37    5:04   0.03s  0.03s -bash
[root@host ~]# pkill -t tty3
[root@host ~]# pgrep -l -u bob
7391 bash
[root@host ~]# pkill -SIGKILL -t tty3
[root@host ~]# pgrep -l -u bob
[root@host ~]#
```

Dieselbe selektive Methode zum Beenden von Prozessen kann auch bei übergeordneten und untergeordneten Prozessen angewendet werden. Verwenden Sie den Befehl **pstree**, um einen Prozessbaum für das System oder einen individuellen Benutzer aufzurufen. Verwenden Sie die PID des übergeordneten Prozesses, um alle von ihm erstellten untergeordneten Prozesse zu beenden. In diesem Fall wird die übergeordnete Bash-Login-Shell weiterhin ausgeführt, da das Signal nur an ihre untergeordneten Prozesse gerichtet wurde.

```
[root@host ~]# pstree -p bob
bash(8391)—sleep(8425)
           └─sleep(8426)
             └─sleep(8427)
[root@host ~]# pkill -P 8391
[root@host ~]# pgrep -l -u bob
bash(8391)
[root@host ~]# pkill -SIGKILL -P 8391
[root@host ~]# pgrep -l -u bob
bash(8391)
[root@host ~]#
```



### Literaturhinweise

**info libc signal** (*Referenzhandbuch für die GNU C Library*)

- Abschnitt 24: Signalverarbeitung

**info libc processes** (*Referenzhandbuch für die GNU C Library*)

- Abschnitt 26: Prozesse

Manpages **kill(1)**, **killall(1)**, **pgrep(1)**, **pkill(1)**, **pstree(1)**, **signal(7)** und **w(1)**

## ► Angeleitete Übung

# Beenden von Prozessen

In dieser Übung verwenden Sie Signale, um Prozesse zu verwalten und zu unterbrechen.

## Ergebnisse

Sie sollten in der Lage sein, mehrere Shell-Prozesse zu starten und zu unterbrechen.

## Bevor Sie Beginnen

Melden Sie sich als Benutzer **student** mit dem Passwort **student** bei **workstation** an.

Führen Sie auf **workstation** den Befehl **lab processes-kill start** aus. Dieser Befehl führt ein Startskript aus, das ermittelt, ob der Host **servera** im Netzwerk erreichbar ist.

```
[student@workstation ~]$ lab processes-kill start
```

- ▶ 1. Öffnen Sie auf **workstation** zwei Terminalfenster nebeneinander. In diesem Abschnitt werden diese Terminals als *links* und *rechts* bezeichnet. Melden Sie sich auf jedem Terminal mit dem Befehl **ssh** bei **servera** als Benutzer **student** an.

```
[student@workstation ~]$ ssh student@servera
...output omitted...
[student@servera ~]$
```

- ▶ 2. Erstellen Sie im *linken* Fenster ein neues Verzeichnis mit dem Namen **/home/student/bin**. Erstellen Sie im neuen Verzeichnis ein Shell-Skript namens **killing**. Machen Sie das Skript ausführbar.
- 2.1. Erstellen Sie mit dem Befehl **mkdir** ein neues Verzeichnis mit dem Namen **/home/student/bin**.

```
[student@servera ~]$ mkdir /home/student/bin
```

- 2.2. Erstellen Sie mit dem Befehl **vim** im Verzeichnis **/home/student/bin** ein Skript mit dem Namen **killing**. Drücken Sie die Taste **i**, um in den interaktiven Modus von Vim zu wechseln. Speichern Sie die Datei mit dem Befehl **:wq**.

```
[student@servera ~]$ vim /home/student/bin/killing
#!/bin/bash
while true; do
    echo -n "$@" >> ~/killing_outfile
    sleep 5
done
```



### Anmerkung

Das Skript **killing** wird bis zum Ende ausgeführt. Es hängt alle 5 Sekunden Befehlszeilenargumente an die Datei **~/killing\_outfile** an.

- 2.3. Wandeln Sie die Datei **killing** mit dem Befehl **chmod** in eine ausführbare Datei um.

```
[student@servera ~]$ chmod +x /home/student/bin/killing
```

- 3. Wechseln Sie in der linken Terminal-Shell mit dem Befehl **cd** in das Verzeichnis **/home/student/bin/**. Starten Sie drei **killing**-Prozesse mit den Argumenten **network**, **interface** bzw. **connection**. Starten Sie die drei Prozesse **network**, **interface** und **connection**. Verwenden Sie das Ampersand-Zeichen (**&**), um die Prozesse im Hintergrund zu starten.

```
[student@servera ~]$ cd /home/student/bin  
[student@servera bin]$ killing network &  
[1] 3460  
[student@servera bin]$ killing interface &  
[2] 3482  
[student@servera bin]$ killing connection &  
[3] 3516
```

Die Prozesse haben unterschiedliche PID-Nummern.

- 4. Überprüfen Sie in der rechten Terminal-Shell mit dem Befehl **tail** und der Option **-f**, ob alle drei Prozesse Text an die Datei **/home/student/killing\_outfile** anhängen.

```
[student@servera ~]$ tail -f ~/killing_outfile  
network interface network connection interface network connection interface  
network  
...output omitted...
```

- 5. Listen Sie in der linken Terminal-Shell mit dem Befehl **jobs** die Jobs auf.

```
[student@servera bin]$ jobs  
[1] Running killing network &  
[2]- Running killing interface &  
[3]+ Running killing connection &
```

- 6. Verwenden Sie Signale, um den Prozess **network** zu unterbrechen. Überprüfen Sie, ob der Prozess **network** unterbrochen wurde. Vergewissern Sie sich in der rechten Terminal-Shell, dass der Prozess **network** keine weitere Ausgabe mehr an **~/killing\_output** anhängt.

- 6.1. Unterbrechen Sie mit **kill** und der Option **-SIGSTOP** den Prozess **network**. Führen Sie **jobs** aus, um zu überprüfen, ob der Prozess unterbrochen wurde.

```
[student@servera bin]$ kill -SIGSTOP %1
[1]+  Stopped                  killing network
[student@servera bin]$ jobs
[1]+  Stopped                  killing network
[2]   Running                 killing interface &
[3]-  Running                 killing connection &
```

- 6.2. Sehen Sie sich in der rechten Terminal-Shell die Ausgabe des Befehls **tail** an. Vergewissern Sie sich, dass das Wort **network** nicht mehr an die Datei **~/killing\_outfile** angehängt wird.

```
...output omitted...
interface connection interface connection interface connection interface
```

- 7. Beenden Sie in der linken Terminal-Shell den Prozess **interface** mit Signalen. Vergewissern Sie sich, dass der Prozess **interface** nicht mehr angezeigt wird. Vergewissern Sie sich in der rechten Terminal-Shell, dass die Ausgabe des Prozesses **interface** nicht mehr an die Datei **~/killing\_outfile** angehängt wird.
- 7.1. Beenden Sie mit dem Befehl **kill** und der Option **-SIGTERM** den Prozess **interface**. Führen Sie den Befehl **jobs** aus, um zu überprüfen, ob der Prozess beendet wurde.

```
[student@servera bin]$ kill -SIGTERM %2
[student@servera bin]$ jobs
[1]+  Stopped                  killing network
[2]  Terminated                killing interface
[3]-  Running                 killing connection &
```

- 7.2. Sehen Sie sich in der rechten Terminal-Shell die Ausgabe des Befehls **tail** an. Vergewissern Sie sich, dass das Wort **interface** nicht mehr an die Datei **~/killing\_outfile** angehängt wird.

```
...output omitted...
connection connection connection connection connection connection connection
connection
```

- 8. Setzen Sie in der linken Terminal-Shell den Prozess **network** anhand von Signalen fort. Überprüfen Sie, ob der Prozess **network ausgeführt** wird. Überprüfen Sie im rechten Fenster, ob die Ausgabe des Prozesses **network** an die Datei **~/killing\_outfile** angehängt wird.
- 8.1. Setzen Sie mit dem Befehl **kill** mit **-SIGCONT** den Prozess **network** wieder fort. Überprüfen Sie mit dem Befehl **jobs**, ob der Prozess **ausgeführt** wird.

```
[student@servera bin]$ kill -SIGCONT %1
[student@servera bin]$ jobs
[1]+  Running                  killing network &
[3]-  Running                 killing connection &
```

- 8.2. Sehen Sie sich in der rechten Terminal-Shell die Ausgabe des Befehls **tail** an. Überprüfen Sie, ob das Wort **network** an die Datei **~/killing\_outfile** angehängt wird.

```
...output omitted...
network connection network connection network connection network connection
network connection
```

- 9. Beenden Sie in der linken Terminal-Shell die verbleibenden zwei Jobs. Stellen Sie sicher, dass alle Jobs beendet wurden und keine Ausgabe mehr erfolgt.
- 9.1. Beenden Sie mit dem Befehl **kill** und der Option **-SIGTERM** den Prozess **network**. Verwenden Sie den gleichen Befehl, um den Prozess **connection** zu beenden.

```
[student@servera bin]$ kill -SIGTERM %1
[student@servera bin]$ kill -SIGTERM %3
[1]+  Terminated          killing network
[student@servera bin]$ jobs
[3]+  Terminated          killing connection
```

- 10. Listen Sie in der linken Terminal-Shell die **tail**-Prozesse auf, die in allen geöffneten Terminal-Shells ausgeführt werden. Beenden Sie die ausgeführten „tail“-Befehle. Vergewissern Sie sich, dass der Prozess nicht mehr ausgeführt wird.
- 10.1. Listen Sie mit dem Befehl **ps** und der Option **-ef** alle ausgeführten „tail“-Prozesse auf. Verfeinern Sie die Suche mit dem Befehl **grep**.

```
[student@servera bin]$ ps -ef | grep tail
student  4581 31358  0 10:02 pts/0    00:00:00 tail -f killing_outfile
student  4869 2252  0 10:33 pts/1    00:00:00 grep --color=auto tail
```

- 10.2. Beenden Sie mit dem Befehl **pkill** und der Option **-SIGTERM** den **tail**-Prozess. Vergewissern Sie sich mit **ps**, dass der Prozess nicht mehr vorhanden ist.

```
[student@servera bin]$ pkill -SIGTERM tail
[student@servera bin]$ ps -ef | grep tail
student  4874 2252  0 10:36 pts/1    00:00:00 grep --color=auto tail
```

- 10.3. Vergewissern Sie sich in der rechten Terminal-Shell, dass der Befehl **tail** nicht mehr ausgeführt wird.

```
...output omitted...
network connection network connection network connection Terminated
[student@servera ~]$
```

- 11. Beenden Sie beide Terminalfenster. Wenn Sie nicht alle Sitzungen beenden, schlägt das Skript zum Beenden fehl.

```
[student@servera bin]$ exit  
logout  
Connection to servera closed.  
[student@workstation ~]$
```

```
[student@servera ~]$ exit  
logout  
Connection to servera closed.  
[student@workstation ~]$
```

## Beenden

Führen Sie auf **workstation** das Skript **lab processes-kill finish** aus, um diese Übung zu beenden.

```
[student@workstation ~]$ lab processes-kill finish
```

Hiermit ist die angeleitete Übung beendet.

# Überwachen der Prozessaktivität

---

## Ziele

Nach Abschluss dieses Abschnitts sollten Sie in der Lage sein, die durchschnittliche Systemauslastung zu beschreiben und Prozesse mit hohem Ressourcenverbrauch auf einem Server zu ermitteln.

## Beschreiben der durchschnittlichen Systemauslastung

*Durchschnittlichen Systemauslastung* ist eine vom Linux-Kernel bereitgestellte Messung, mit der die wahrgenommene Systemauslastung im Zeitverlauf auf einfache Weise dargestellt werden kann. Sie kann als grobe Maßeinheit dafür dienen, wie viele Systemressourcenanforderungen ausstehen, und um zu bestimmen, ob die Systemauslastung im Zeitverlauf zu- oder abnimmt.

Alle fünf Sekunden erfasst der Kernel die aktuelle *Auslastung*, basierend auf der Anzahl der Prozesse im ausführbaren und unterbrechungsfreien Zustand. Diese Zahl wird akkumuliert und als exponentieller gleitender Durchschnitt der letzten 1, 5 und 15 Minuten angegeben.

## Verstehen der Berechnung der durchschnittlichen Systemauslastung unter Linux

Die durchschnittliche Systemauslastung stellt die wahrgenommene Systemauslastung über einen bestimmten Zeitraum dar. Linux ermittelt diesen Wert, indem erfasst wird, wie viele Prozesse zur Ausführung auf einer CPU bereit sind und wie viele Prozesse auf den Abschluss einer Laufwerks- oder Netzwerk-I/O warten.

- Die Auslastung basiert auf dem laufenden Durchschnitt der Anzahl der Prozesse, die zur Ausführung bereit sind (im Prozesszustand **R**) und auf den Abschluss der I/O warten (im Prozesszustand **D**).
- Einige UNIX-Systeme beziehen sich zur Berechnung der Systemauslastung nur auf die CPU-Auslastung oder die Länge der Warteschlangen. Linux enthält auch die Laufwerks- oder Netzwerkauslastung, da dies die Systemleistung ebenso stark beeinflussen kann wie die CPU-Auslastung. Wenn die durchschnittliche Auslastung bei minimaler CPU-Aktivität sehr hoch ist, sehen Sie sich die Laufwerks- und Netzwerkaktivitäten genauer an.

Die durchschnittliche Auslastung ist eine grobe Messung der Anzahl der Prozesse, die derzeit auf den Abschluss einer Anforderung warten, bevor sie etwas anderes ausführen können. Die Anforderung könnte sich auf CPU-Zeit beziehen, um den Prozess auszuführen. Alternativ kann die Anforderung ein kritischer Laufwerks-/I/O-Vorgang sein, der abgeschlossen werden muss, und der Prozess kann erst auf der CPU ausgeführt werden, wenn die Anforderung abgeschlossen ist, auch wenn die CPU sich im Leerlauf befindet. In beiden Fällen wirkt sich dies auf die Systemauslastung aus und das System scheint langsamer zu laufen, da Prozesse auf die Ausführung warten.

## Interpretieren der angezeigten Werte der durchschnittlichen Systemauslastung

Der Befehl **uptime** ist eine Möglichkeit, die aktuelle durchschnittliche Auslastung anzuzeigen. Der Befehl gibt die aktuelle Uhrzeit, die Betriebsdauer des Rechners, die Anzahl der ausgeführten Benutzersitzungen und die aktuelle durchschnittliche Auslastung aus.

```
[user@host ~]$ uptime  
15:29:03 up 14 min, 2 users, load average: 2.92, 4.48, 5.20
```

Die drei Werte für die durchschnittliche Auslastung stellen die Auslastung in den letzten 1, 5 und 15 Minuten dar. Ein kurzer Blick reicht aus, um zu erkennen, ob die Systemauslastung ansteigt oder abfällt.

Wenn der Hauptbeitrag zur durchschnittlichen Auslastung von Prozessen stammt, die auf die CPU warten, können Sie den ungefähren Auslastungswert *pro CPU* berechnen, um zu ermitteln, ob lange Wartezeiten im System vorherrschen.

Mit dem Befehl **lscpu** können Sie ermitteln, über wie viele CPUs ein System verfügt.

Im folgenden Beispiel ist das System ein Dual-Core-System mit einem Socket und zwei Hyperthreads pro Kern. Vereinfacht gesagt, behandelt Linux dieses System für Terminierungszwecke als ein System mit vier CPUs.

```
[user@host ~]$ lscpu  
Architecture:           x86_64  
CPU op-mode(s):        32-bit, 64-bit  
Byte Order:            Little Endian  
CPU(s):                4  
On-line CPU(s) list:  0-3  
Thread(s) per core:   2  
Core(s) per socket:   2  
Socket(s):             1  
NUMA node(s):          1  
...output omitted...
```

Stellen Sie sich für einen Moment vor, dass der einzige Beitrag zur Auslastung von Prozessen stammt, die CPU-Zeit benötigen. Dann können Sie die angezeigten durchschnittlichen Auslastungswerte durch die Anzahl der logischen CPUs im System teilen. Ein Wert unter 1 bedeutet angemessene Ressourcenauslastung und minimale Wartezeiten. Ein Wert über 1 bedeutet volle Ressourcenauslastung und eine gewisse Prozessverzögerung.

```
# From lscpu, the system has four logical CPUs, so divide by 4:  
#                                     load average: 2.92, 4.48, 5.20  
#         divide by number of logical CPUs:    4    4    4  
#   -----  
#                                     per-CPU load average: 0.73  1.12  1.30  
#  
# This system's load average appears to be decreasing.  
# With a load average of 2.92 on four CPUs, all CPUs were in use ~73% of the time.  
# During the last 5 minutes, the system was overloaded by ~12%.  
# During the last 15 minutes, the system was overloaded by ~30%.
```

Eine CPU-Warteschlange im Leerlauf hat eine Auslastung von 0. Jeder Prozess, der auf eine CPU wartet, addiert die Anzahl 1 zur Auslastung hinzu. Wenn ein Prozess auf einer CPU ausgeführt wird, ist die Auslastung eins; die Ressource (die CPU) ist in Gebrauch, aber es warten keine Anforderungen. Wenn dieser Prozess eine volle Minute lang läuft, ist sein Beitrag zur einminütigen durchschnittlichen Auslastung 1.

Allerdings werden Prozesse, die wegen einer ausgelasteten Laufwerks- oder Netzwerkressource nicht unterbrechbar auf I/O warten, mitgezählt und erhöhen die durchschnittliche Auslastung. Obwohl dies kein Hinweis auf die CPU-Auslastung ist, werden diese Prozesse zur Warteschlangenzahl hinzugefügt, da sie auf Ressourcen warten und erst dann auf einer CPU ausgeführt werden können, wenn sie diese erhalten. Aufgrund von Ressourcenbeschränkungen, die dazu führen, dass Prozesse nicht ausgeführt werden, ist das System weiterhin ausgelastet.

Bis zur vollen Ressourcenauslastung bleibt die durchschnittliche Systemauslastung unter 1, da Prozesse selten in der Warteschlange gestellt werden. Die durchschnittliche Systemauslastung erhöht sich erst dann, wenn Anforderungen durch die Ressourcenauslastung in der Warteschlange hängen bleiben und von der Auslastungsberechnungsroutine gezählt werden. Wenn die Ressourcenauslastung sich 100 % nähert, wird jede weitere Anforderung in die Warteschlange gestellt.

Eine Reihe zusätzlicher Tools gibt die durchschnittliche Auslastung an, darunter **w** und **top**.

## Prozessüberwachung in Echtzeit

Das Programm **top** bietet eine dynamische Anzeige der Systemprozesse, wobei eine Überschriftenzeile gefolgt von einer Liste der Prozesse und Threads ähnlich der Informationen durch den Befehl **ps** angezeigt wird. Im Gegensatz zu der statischen **ps**-Übersicht, die nur eine Momentaufnahme bietet, werden die Informationen bei **top** ständig nach einer individuell festgelegten Zeitspanne aktualisiert. Außerdem können die Spalten neu angeordnet, sortiert und hervorgehoben werden. Benutzerkonfigurationen können gespeichert und als Standardeinstellung festgelegt werden.

Die Standardspalten haben dieselben Bezeichnungen wie bei anderen Tools:

- Die Prozess-ID (**PID**)
- Der Benutzername (**USER**) ist der Eigentümer des Prozesses.
- Virtueller Speicher (**VIRT**) ist der gesamte vom Prozess belegte Speicher inklusive der residenten Gruppe, gemeinsam genutzten Bibliotheken und aller zugeordneten und getauschten Speicherseiten. (**VSZ** im Befehl **ps** genannt.)
- Der residente Speicher (**RES**) ist der vom Prozess belegte physische Speicher inklusive aller residenten, gemeinsam genutzten Objekte. (**RSS** im Befehl **ps** genannt.)
- Der Prozessstatus (**S**) wird wie folgt angezeigt:
  - **D** = Unterbrechungsfreier Ruhezustand
  - **R** = Ausgeführt oder ausführbar
  - **S** = Inaktiv
  - **T** = Angehalten oder nachverfolgt
  - **Z** = Zombie
- Die CPU-Zeit (**TIME**) zeigt die gesamte Verarbeitungszeit seit Prozessstart an. Diese kann so eingestellt werden, dass die gesamte Verarbeitungszeit der untergeordneten Prozesse mit angezeigt wird.
- Der Befehlsname des Prozesses (**COMMAND**)

## Grundlegende Tastaturbefehle in top

| Taste                   | Zweck                                                                                                                                |
|-------------------------|--------------------------------------------------------------------------------------------------------------------------------------|
| ? oder h                | Hilfe zu den interaktiven Tasturbefehlen                                                                                             |
| l, t, m                 | Umschaltbefehle für die Überschriftenzeilen Auslastung, Threads und Speicher                                                         |
| 1                       | Umschaltbefehle für das Anzeigen individueller CPUs oder eines Gesamtwerts aller CPUs in der Überschriftenzeile                      |
| s <sup>(1)</sup>        | Ändern der (Bild-)Wiederholrate in Sekunden mit Kommatrennung (z. B. 0,5, 1, 5)                                                      |
| b                       | Umschaltbefehl für das inverse Hervorheben <b>laufender</b> Prozesse; Standardeinstellung ist nur Fettformatierung                   |
| <b>Umschalt+b</b>       | Dieser Befehl aktiviert die Verwendung der Fettformatierung in der Überschriftenzeile und für <i>laufende</i> Prozesse.              |
| <b>Umschalt+h</b>       | Umschaltbefehle für Threads; individuelle Threads oder Prozesszusammenfassung anzeigen                                               |
| u,<br><b>Umschalt+u</b> | Filter für Benutzernamen (effektiv oder real)                                                                                        |
| <b>Umschalt+m</b>       | Dieser Befehl sortiert die Prozesse nach Speicherbelegung in absteigender Reihenfolge.                                               |
| <b>Umschalt+p</b>       | Dieser Befehl sortiert Prozesse nach Prozessorauslastung in absteigender Reihenfolge.                                                |
| k <sup>(1)</sup>        | Beenden eines Prozesses. Geben Sie bei Aufforderung <b>PID</b> gefolgt von <b>signal</b> ein.                                        |
| r <sup>(1)</sup>        | Priorisieren eines Prozesses. Geben Sie bei Aufforderung <b>PID</b> gefolgt von <b>nice_value</b> ein.                               |
| <b>Umschalt+w</b>       | Schreiben (Speichern) der aktuellen Anzeigekonfiguration zur Verwendung beim nächsten Neustart von <b>top</b> .                      |
| q                       | Beenden                                                                                                                              |
| f                       | Verwalten der Spalten, indem Sie Felder aktivieren oder deaktivieren. Hier können Sie auch das Sortierfeld für <b>top</b> festlegen. |
| Hinweis:                | <sup>(1)</sup> Nicht verfügbar, falls top im abgesicherten Modus gestartet wurde. Siehe <b>top(1)</b> .                              |



### Literaturhinweise

Manpages **ps(1)**, **top(1)**, **uptime(1)** und **w(1)**

## ► Angeleitete Übung

# Überwachen der Prozessaktivität

In dieser Übung verwenden Sie den Befehl **top**, um ausgeführte Prozesse dynamisch zu untersuchen und zu steuern.

## Ergebnisse

Sie sollten in der Lage sein, Prozesse in Echtzeit zu verwalten.

## Bevor Sie Beginnen

Melden Sie sich als Benutzer **student** mit dem Passwort **student** bei **workstation** an.

Führen Sie auf **workstation** den Befehl **lab processes-monitor start** aus. Dieser Befehl führt ein Startskript aus, das ermittelt, ob der Host **servera** im Netzwerk erreichbar ist.

```
[student@workstation ~]$ lab processes-monitor start
```

- 1. Öffnen Sie auf **workstation** zwei Terminalfenster nebeneinander. Diese Terminals werden als *links* und *rechts* bezeichnet. Melden Sie sich auf jedem Terminal mit dem Befehl **ssh** bei **servera** als Benutzer **student** an.

```
[student@workstation ~]$ ssh student@servera  
...output omitted...  
[student@servera ~]$
```

- 2. Erstellen Sie in der *linken* Terminal-Shell ein neues Verzeichnis mit dem Namen **/home/student/bin**. Erstellen Sie im neuen Verzeichnis ein Shell-Skript namens **monitor**, das künstliche CPU-Last erzeugt. Stellen Sie sicher, dass das Skript ausführbar ist.
- 2.1. Erstellen Sie mit dem Befehl **mkdir** ein neues Verzeichnis mit dem Namen **/home/student/bin**.

```
[student@servera ~]$ mkdir /home/student/bin
```

- 2.2. Erstellen Sie ein Skript mit dem Namen **monitor** im Verzeichnis **/home/student/bin**, das den folgenden Inhalt hat:

```
#!/bin/bash  
while true; do  
    var=1  
    while [[ var -lt 60000 ]]; do  
        var=$($var+1)  
    done  
    sleep 1  
done
```

## Kapitel 8 | Überwachen und Verwalten von Linux-Prozessen

Das Skript **monitor** wird bis zum Ende ausgeführt. Es erzeugt künstliche CPU-Last durch Ausführen von 60.000 Additionsproblemen. Das Skript befindet sich dann eine Sekunde lang im Ruhezustand, setzt die Variable zurück und beginnt erneut.

- 2.3. Wandeln Sie die Datei **monitor** mit dem Befehl **chmod** in eine ausführbare Datei um.

```
[student@servera ~]$ chmod a+x /home/student/bin/monitor
```

- 3. Führen Sie in der rechten Terminal-Shell das Dienstprogramm **top** aus. Verändern Sie die Größe des Fensters so, dass es die maximal mögliche Höhe hat.

```
[student@servera ~]$ top
top - 12:13:03 up 11 days, 58 min, 3 users, load average: 0.00, 0.00, 0.00
Tasks: 113 total, 2 running, 111 sleeping, 0 stopped, 0 zombie
%Cpu(s): 0.2 us, 0.0 sy, 0.0 ni, 99.8 id, 0.0 wa, 0.0 hi, 0.0 si, 0.0 st
MiB Mem : 1829.4 total, 1377.3 free, 193.9 used, 258.2 buff/cache
MiB Swap: 1024.0 total, 1024.0 free, 0.0 used. 1476.1 avail Mem

PID USER PR NI VIRT RES SHR S %CPU %MEM TIME+ COMMAND
5861 root 20 0 0 0 0 I 0.3 0.0 0:00.71 kworker/1:3-
events
6068 student 20 0 273564 4300 3688 R 0.3 0.2 0:00.01 top
1 root 20 0 178680 13424 8924 S 0.0 0.7 0:04.03 systemd
2 root 20 0 0 0 0 S 0.0 0.0 0:00.03 kthreadd
3 root 0 -20 0 0 0 I 0.0 0.0 0:00.00 rcu_gp
...output omitted...
```

- 4. Ermitteln Sie in der linken Terminal-Shell mit dem Befehl **lscpu** die Anzahl der logischen CPUs auf diesem virtuellen Rechner.

```
[student@servera ~]$ lscpu
Architecture: x86_64
CPU op-mode(s): 32-bit, 64-bit
Byte Order: Little Endian
CPU(s): 2
...output omitted...
```

- 5. Führen Sie in der linken Terminal-Shell eine einzelne Instanz der ausführbaren Datei **monitor** aus. Verwenden Sie das Ampersand-Zeichen (&), um die Prozesse im Hintergrund auszuführen.

```
[student@servera ~]$ monitor &
[1] 6071
```

- 6. Sehen Sie sich in der rechten Terminal-Shell die **top**-Anzeige an. Aktivieren Sie die Überschriftenzeilen für Auslastung, Threads und Speicher mittels der Tastaturbefehle **1**, **t** und **m**. Vergewissern Sie sich anschließend, dass alle Überschriftenzeilen angezeigt werden.
- 7. Sehen Sie sich die Prozess-ID (PID) für **monitor** an. Sehen Sie sich die CPU-Auslastung des Prozesses an. Diese sollte zwischen 15 und 25 % liegen.

```
[student@servera ~]$ top
PID USER      PR  NI    VIRT    RES    SHR S %CPU %MEM     TIME+ COMMAND
071 student    20   0 222448  2964  2716 S 18.7  0.2  0:27.35 monitor
...output omitted...
```

Sehen Sie sich die durchschnittliche Systemauslastung an. Die durchschnittliche Auslastung für eine Minute ist aktuell geringer als der Wert 1. Der gemessene Wert kann durch Ressourcenkonflikte mit einem anderen virtuellen Rechner oder dem virtuellen Host beeinflusst werden.

```
top - 12:23:45 up 11 days, 1:09, 3 users, load average: 0.21, 0.14, 0.05
```

- 8. Führen Sie in der linken Terminal-Shell eine zweite Instanz von **monitor** aus. Verwenden Sie das Ampersand-Zeichen (&), um die Prozesse im Hintergrund auszuführen.

```
[student@servera ~]$ monitor &
[2] 6498
```

- 9. Sehen Sie sich in der rechten Terminal-Shell die Prozess-ID (PID) für den zweiten **monitor**-Prozess an. Sehen Sie sich die CPU-Auslastung des Prozesses an. Diese sollte ebenfalls zwischen 15 und 25 % liegen.

```
[student@servera ~]$ top
PID USER      PR  NI    VIRT    RES    SHR S %CPU %MEM     TIME+ COMMAND
6071 student    20   0 222448  2964  2716 S 19.0  0.2  1:36.53 monitor
6498 student    20   0 222448  2996  2748 R 15.7  0.2  0:16.34 monitor
...output omitted...
```

Zeigen Sie erneut die durchschnittliche Auslastung bei 1 Minute an, die immer noch kleiner als 1 ist. Warten Sie mindestens eine Minute, damit sich die Berechnung an den neuen Workload anpassen kann.

```
top - 12:27:39 up 11 days, 1:13, 3 users, load average: 0.36, 0.25, 0.11
```

- 10. Führen Sie in der linken Terminal-Shell eine dritte Instanz von **monitor** aus. Verwenden Sie das Ampersand-Zeichen (&), um die Prozesse im Hintergrund auszuführen.

```
[student@servera ~]$ monitor &
[3] 6881
```

- 11. Sehen Sie sich in der rechten Terminal-Shell die Prozess-ID (PID) für den dritten **monitor**-Prozess an. Sehen Sie sich die CPU-Auslastung des Prozesses an. Diese sollte wieder zwischen 15 und 25 % liegen.

```
[student@servera ~]$ top
  PID USER      PR  NI    VIRT    RES   SHR S %CPU %MEM     TIME+ COMMAND
6881 student    20   0 222448  3032  2784 S 18.6  0.2  0:11.48 monitor
6498 student    20   0 222448  2996  2748 S 15.6  0.2  0:47.86 monitor
6071 student    20   0 222448  2964  2716 S 18.1  0.2  2:07.86 monitor
```

Um die durchschnittliche Auslastung auf über 1 zu steigern, müssen Sie weitere **monitor**-Prozesse starten. Die Einrichtung des Unterrichtsraums verfügt über 2 CPUs, sodass nur 3 Prozesse nicht ausreichen, um sie zu beladen. Starten Sie drei weitere **monitor**-Prozesse. Zeigen Sie erneut die durchschnittliche Auslastung an, die nun über 1 liegen sollte. Warten Sie mindestens eine Minute, damit sich die Berechnung an den neuen Workload anpassen kann.

```
[student@servera ~]$ monitor &
[4] 10708
[student@servera ~]$ monitor &
[5] 11122
[student@servera ~]$ monitor &
[6] 11338
```

```
top - 12:42:32 up 11 days,  1:28,  3 users,  load average: 1.23, 2.50, 1.54
```

- 12. Wenn Sie sich alle Werte für die durchschnittliche Auslastung angesehen haben, beenden Sie alle **monitor**-Prozesse in **top**.

- 12.1. Drücken Sie in der rechten Terminal-Shell **k**. Sehen Sie sich das Eingabefeld an, das zwischen Überschriftenzeile und Spalten steht.

```
...output omitted...
PID to signal/kill [default pid = 11338]
```

- 12.2. An der Eingabeaufforderung wurden die **monitor**-Prozesse oben in der Liste ausgewählt. Drücken Sie die **Eingabetaste**, um den Prozess zu beenden.

```
...output omitted...
Send pid 11338 signal [15/sigterm]
```

- 12.3. Drücken Sie erneut die **Eingabetaste**, um das SIGTERM-Signal 15 zu bestätigen.

Stellen Sie sicher, dass der ausgewählte Prozess nicht mehr in **top** angezeigt wird. Falls die PID weiterhin angezeigt wird, führen Sie die Schritte zum Beenden der Instanz erneut aus. Geben Sie jetzt allerdings das SIGKILL-Signal 9 ein, wenn Sie zur Eingabe aufgefordert werden.

```
6498 student 20 0 222448 2996 2748 R 22.9 0.2 5:31.47 monitor
6881 student 20 0 222448 3032 2784 R 21.3 0.2 4:54.47 monitor
11122 student 20 0 222448 2984 2736 R 15.3 0.2 2:32.48 monitor
6071 student 20 0 222448 2964 2716 S 15.0 0.2 6:50.90 monitor
10708 student 20 0 222448 3032 2784 S 14.6 0.2 2:53.46 monitor
```

- ▶ **13.** Wiederholen Sie den letzten Schritt für jede der verbleibenden **monitor**-Instanzen. Vergewissern Sie sich, dass keine **monitor**-Prozesse mehr in **top** vorhanden sind.
- ▶ **14.** Drücken Sie in der rechten Terminal-Shell **q**, um **top** zu beenden. Beenden Sie auf **servera** beide Terminalfenster.

```
[student@servera ~]$ exit
logout
Connection to servera closed.
[student@workstation ~]$
```

```
[student@servera ~]$ exit
logout
Connection to servera closed.
[student@workstation ~]$
```

## Beenden

Führen Sie auf **workstation** das Skript **lab processes-monitor finish** aus, um diese Übung zu beenden.

```
[student@workstation ~]$ lab processes-monitor finish
```

Hiermit ist die angeleitete Übung beendet.

## ► Praktische Übung

# Überwachen und Verwalten von Linux-Prozessen

### Leistungscheckliste

In dieser praktischen Übung lokalisieren und verwalten Sie Prozesse, die am meisten Ressourcen auf einem System verbrauchen.

### Ergebnisse

Sie sollten in der Lage sein, Prozesse mit **top** als Prozessmanagement-Tool zu verwalten.

### Bevor Sie Beginnen

Melden Sie sich als Benutzer **student** mit dem Passwort **student** bei **workstation** an.

Führen Sie auf **workstation** den Befehl **lab processes-review start** aus. Dieser Befehl führt ein Startskript aus, um zu ermitteln, ob der Host **serverb** im Netzwerk erreichbar ist.

```
[student@workstation ~]$ lab processes-review start
```

1. Öffnen Sie auf **workstation** zwei Terminalfenster nebeneinander. In diesem Abschnitt werden diese Terminals als *links* und *rechts* bezeichnet. Melden Sie sich auf jedem Terminal bei **serverb** als Benutzer **student** an.  
Erstellen Sie ein Skript namens **process101**, das künstliche CPU-Last erzeugt. Erstellen Sie das Skript im Verzeichnis **/home/student/bin**.

```
#!/bin/bash
while true; do
    var=1
    while [[ var -lt 50000 ]]; do
        var=$((var+1))
    done
    sleep 1
done
```

2. Führen Sie im rechten Fenster das Dienstprogramm **top** aus.
3. Ermitteln Sie in der linken Terminal-Shell die Anzahl der logischen CPUs auf dem virtuellen Rechner. Führen Sie das Skript **process101** im Hintergrund aus.
4. Sehen Sie sich in der rechten Terminal-Shell die **top**-Anzeige an. Wechseln Sie zwischen „Load“, „Threads“ und „Memory“. Sehen Sie sich die Prozess-ID (PID) für **process101** an. Sehen Sie sich den CPU-Prozentsatz an. Er sollte etwa zwischen 10 % und 15 % liegen. Vergewissern Sie sich, dass **top** die CPU-Auslastung anzeigt, sobald Sie „Load“, „Threads“ und „Memory“ angezeigt haben.

5. Deaktivieren Sie die Fettformatierung in der Anzeige. Speichern Sie die Konfiguration, damit sie beim nächsten Start von top wieder geladen wird. Überprüfen Sie, ob die Änderungen gespeichert wurden.
6. Kopieren Sie das Skript **process101** in eine neue Datei mit dem Namen **process102**. Bearbeiten Sie das Skript, um mehr künstliche CPU-Last zu erzeugen. Erhöhen Sie die Last von fünfzigtausend auf hunderttausend. Starten Sie den Prozess **process102** im Hintergrund.
7. Überprüfen Sie in der rechten Terminal-Shell, ob der Prozess ausgeführt wird und die meisten CPU-Ressourcen verwendet. Die Auslastung sollte zwischen 25 % und 35 % liegen.
8. Die durchschnittliche Auslastung liegt immer noch unter 1. Kopieren Sie **process101** in ein neues Skript mit dem Namen **process103**. Erhöhen Sie die Additionsanzahl auf achthunderttausend. Starten Sie **process103** im Hintergrund. Überprüfen Sie, ob die durchschnittliche Auslastung über 1 liegt. Es kann einige Minuten dauern, bis sich die durchschnittliche Auslastung ändert.
9. Ändern Sie in der linken Terminal-Shell den Benutzer in **root**. Unterbrechen Sie den Prozess **process101**. Listen Sie die verbleibenden Jobs auf. Sie werden feststellen, dass der Prozessstatus für **process101** jetzt **T** lautet.
10. Setzen Sie den Prozess **process101** fort.
11. Beenden Sie in der Befehlszeile **process101**, **process102** und **process103**. Vergewissern Sie sich, dass die Prozesse nicht mehr in **top** angezeigt werden.
12. Beenden Sie in der linken Terminal-Shell den Benutzer **root**. Halten Sie in der rechten Terminal-Shell den Befehl **top** an. Beenden Sie auf **serverb** beide Terminalfenster.

## Bewertung

Führen Sie auf **workstation** das Skript **lab processes-review grade** aus, um den Erfolg dieser Übung zu bestätigen.

```
[student@workstation ~]$ lab processes-review grade
```

## Beenden

Führen Sie auf **workstation** das Skript **lab processes-review finish** aus, um die praktische Übung zu beenden.

```
[student@workstation ~]$ lab processes-review finish
```

Damit ist die praktische Übung abgeschlossen.

## ► Lösung

# Überwachen und Verwalten von Linux-Prozessen

### Leistungscheckliste

In dieser praktischen Übung lokalisieren und verwalten Sie Prozesse, die am meisten Ressourcen auf einem System verbrauchen.

### Ergebnisse

Sie sollten in der Lage sein, Prozesse mit **top** als Prozessmanagement-Tool zu verwalten.

### Bevor Sie Beginnen

Melden Sie sich als Benutzer **student** mit dem Passwort **student** bei **workstation** an.

Führen Sie auf **workstation** den Befehl **lab processes-review start** aus. Dieser Befehl führt ein Startskript aus, um zu ermitteln, ob der Host **serverb** im Netzwerk erreichbar ist.

```
[student@workstation ~]$ lab processes-review start
```

1. Öffnen Sie auf **workstation** zwei Terminalfenster nebeneinander. In diesem Abschnitt werden diese Terminals als *links* und *rechts* bezeichnet. Melden Sie sich auf jedem Terminal bei **serverb** als Benutzer **student** an.  
Erstellen Sie ein Skript namens **process101**, das künstliche CPU-Last erzeugt. Erstellen Sie das Skript im Verzeichnis **/home/student/bin**.

```
#!/bin/bash
while true; do
    var=1
    while [[ var -lt 50000 ]]; do
        var=$((var+1))
    done
    sleep 1
done
```

- 1.1. Öffnen Sie auf **workstation** zwei Terminalfenster nebeneinander. Melden Sie sich auf jedem Terminal mit dem Befehl **ssh** bei **serverb** als Benutzer **student** an.

```
[student@workstation ~]$ ssh student@serverb
...output omitted...
[student@serverb ~]$
```

- 1.2. Erstellen Sie in der linken Terminal-Shell mit dem Befehl **mkdir** das Verzeichnis **/home/student/bin**.

```
[student@serverb ~]$ mkdir /home/student/bin
```

- 1.3. Erstellen Sie in der linken Terminal-Shell mit dem Befehl **vim** das Skript **process101**. Drücken Sie die Taste **i**, um in den interaktiven Modus zu wechseln. Geben Sie **:wq** ein, um die Datei zu speichern.

```
[student@serverb ~]$ vim /home/student/bin/process101
#!/bin/bash
while true; do
    var=1
    while [[ var -lt 50000 ]]; do
        var=$((var+1))
    done
    sleep 1
done
```

- 1.4. Wandeln Sie die Datei **process101** mit dem Befehl **chmod** in eine ausführbare Datei um.

```
[student@serverb ~]$ chmod +x /home/student/bin/process101
```

2. Führen Sie im rechten Fenster das Dienstprogramm **top** aus.

- 2.1. Führen Sie im rechten Fenster das Dienstprogramm **top** aus. Verändern Sie die Größe des Fensters so, dass es die maximal mögliche Höhe hat.

```
[student@serverb ~]$ top
top - 13:47:06 up 19 min,  2 users,  load average: 0.00, 0.00, 0.00
Tasks: 110 total,   1 running, 109 sleeping,   0 stopped,   0 zombie
%Cpu(s):  0.0 us,  3.1 sy,  0.0 ni, 96.9 id,  0.0 wa,  0.0 hi,  0.0 si,  0.0 st
MiB Mem : 1829.4 total, 1439.1 free,   171.9 used,   218.4 buff/cache
MiB Swap: 1024.0 total, 1024.0 free,     0.0 used. 1499.6 avail Mem

PID USER      PR  NI      VIRT      RES      SHR S %CPU %MEM     TIME COMMAND
 1 root      20   0 178536  13488  8996 S  0.0  0.7  0:01.15 systemd
 2 root      20   0          0          0      0 S  0.0  0.0  0:00.00 kthreadd
 3 root      0 -20          0          0      0 I  0.0  0.0  0:00.00 rcu_gp
 4 root      0 -20          0          0      0 I  0.0  0.0  0:00.00 rcu_par_gp
 6 root      0 -20          0          0      0 I  0.0  0.0  0:00.00 kworker/0:0H-
kblockd
...output omitted...
```

3. Ermitteln Sie in der linken Terminal-Shell die Anzahl der logischen CPUs auf dem virtuellen Rechner. Führen Sie das Skript **process101** im Hintergrund aus.

- 3.1. Ermitteln Sie mit dem Befehl **grep** die Anzahl der logischen CPUs.

```
[student@serverb ~]$ grep "model name" /proc/cpuinfo | wc -l
2
```

- 3.2. Wechseln Sie mit dem Befehl **cd** in das Verzeichnis **/home/student/bin**. Führen Sie das Skript **process101** im Hintergrund aus.

```
[student@serverb ~]$ cd /home/student/bin
[student@serverb bin]$ process101 &
[1] 1180
```

4. Sehen Sie sich in der rechten Terminal-Shell die **top**-Anzeige an. Wechseln Sie zwischen „Load“, „Threads“ und „Memory“. Sehen Sie sich die Prozess-ID (PID) für **process101** an. Sehen Sie sich den CPU-Prozentsatz an. Er sollte etwa zwischen 10 % und 15 % liegen. Vergewissern Sie sich, dass **top** die CPU-Auslastung anzeigt, sobald Sie „Load“, „Threads“ und „Memory“ angezeigt haben.

#### 4.1. Drücken Sie **Umschalt+m**.

```
top - 13:56:24 up 28 min,  2 users,  load average: 0.21, 0.08, 0.02
Tasks: 112 total,   2 running, 110 sleeping,   0 stopped,   0 zombie
%Cpu(s):  5.8 us,  1.3 sy,  0.0 ni, 92.8 id,  0.0 wa,  0.0 hi,  0.0 si,  0.0 st
MiB Mem : 1829.4 total,  1438.1 free,   172.7 used,   218.6 buff/cache
MiB Swap: 1024.0 total,  1024.0 free,      0.0 used. 1498.7 avail Mem

  PID USER      PR  NI    VIRT    RES    SHR S %CPU %MEM     TIME+ COMMAND
  705 root      20   0  409956  34880  33620 S  0.0  1.9  0:00.04 sssd_nss
  706 root      20   0  454304  34472  14304 S  0.0  1.8  0:00.62 firewalld
  725 root      20   0  611348  28244  14076 S  0.0  1.5  0:00.27 tuned
  663 polkitd   20   0 1907312  23876  16040 S  0.0  1.3  0:00.04 polkitd
  718 root      20   0  600316  17176  14832 S  0.0  0.9  0:00.06 NetworkManager
...output omitted...
```



#### Anmerkung

Beachten Sie, dass beim Umschalten von top in den *memory*-Modus **process101** nicht mehr der erste Prozess ist. Sie können **Umschalt+p** drücken, um zur CPU-Auslastung zurückzukehren.

#### 4.2. Drücken Sie **m**.

```
top - 09:32:52 up 20:05,  2 users,  load average: 0.18, 0.10, 0.03
Tasks: 112 total,   2 running, 110 sleeping,   0 stopped,   0 zombie
%Cpu(s):  7.8/1.5      9[|||||||||]                                ]
MiB Mem : 18.3/1829.4   [|||||||||||||||||||]                      ]
MiB Swap:  0.0/1024.0   [   ]          ]
  PID USER      PR  NI    VIRT    RES    SHR S %CPU %MEM     TIME+ COMMAND
  705 root      20   0  409956  34880  33620 S  0.0  1.9  0:00.04 sssd_nss
  706 root      20   0  454304  34472  14304 S  0.0  1.8  0:00.62 firewalld
  725 root      20   0  611348  28244  14076 S  0.0  1.5  0:00.30 tuned
  663 polkitd   20   0 1907312  23876  16040 S  0.0  1.3  0:00.04 polkitd
  718 root      20   0  600316  17176  14832 S  0.0  0.9  0:00.07 NetworkManager
...output omitted...
```

#### 4.3. Drücken Sie **t**.

```
Tasks: 113 total,   2 running, 111 sleeping,   0 stopped,   0 zombie
%Cpu(s):  7.8/1.5      9[|||||||||]                                ]
MiB Mem : 1829.4 total,  1436.7 free,   173.7 used,   219.0 buff/cache
```

```
MiB Swap: 1024.0 total, 1024.0 free, 0.0 used. 1497.7 avail Mem

  PID USER      PR  NI    VIRT    RES    SHR S %CPU %MEM     TIME+ COMMAND
1180 student    20   0 222448  3056  2808 S 12.0  0.2  1:59.94 process101
  705 root      20   0 409956 34880 33620 S  0.0  1.9  0:00.04 sssd_nss
  706 root      20   0 454304 34472 14304 S  0.0  1.8  0:00.62 firewalld
  725 root      20   0 611348 28244 14076 S  0.0  1.5  0:00.30 tuned
  663 polkitd   20   0 1907312 23876 16040 S  0.0  1.3  0:00.04 polkitd
  718 root      20   0 600316 17176 14832 S  0.0  0.9  0:00.07 NetworkManager
...output omitted...
```

#### 4.4. Drücken Sie **Umschalt+p**.

```
top - 09:35:48 up 20:08, 2 users, load average: 0.10, 0.10, 0.04
Tasks: 110 total, 4 running, 106 sleeping, 0 stopped, 0 zombie
%Cpu(s): 6.8 us, 1.0 sy, 0.0 ni, 92.2 id, 0.0 wa, 0.0 hi, 0.0 si, 0.0 st
MiB Mem : 18.3/1829.4 [||||||||||||||||||]
MiB Swap: 0.0/1024.0 []

  PID USER      PR  NI    VIRT    RES    SHR S %CPU %MEM     TIME+ COMMAND
27179 student    20   0 222448  3060  2812 R 15.6  0.2  1:13.47 process101
...output omitted...
```

5. Deaktivieren Sie die Fettformatierung in der Anzeige. Speichern Sie die Konfiguration, damit sie beim nächsten Start von top wieder geladen wird. Überprüfen Sie, ob die Änderungen gespeichert wurden.

##### 5.1. Drücken Sie **Umschalt+b**, um die Fettformatierung zu deaktivieren.

```
top - 19:40:30 up 6:12, 2 users, load average: 0.11, 0.12, 0.09
Tasks: 112 total, 1 running, 111 sleeping, 0 stopped, 0 zombie
%Cpu(s): 7.6/1.5 9[|||||||||]
MiB Mem : 18.2/1829.4 [||||||||||||||||||]
MiB Swap: 0.0/1024.0 [
```

##### 5.2. Drücken Sie **Umschalt+w**, um diese Konfiguration zu speichern. Die Standardkonfiguration wird in der Datei **toprc** im Verzeichnis **/home/student/.config/procps** gespeichert. Vergewissern Sie sich in der linken Terminal-Shell, dass die Datei **toprc** vorhanden ist.

```
[student@serverb bin]$ ls -l /home/student/.config/procps/toprc
-rw-rw-r-- 1 student student 966 Feb 18 19:45 /home/student/.config/procps/toprc
```

- 5.3. Beenden Sie in der rechten Terminal-Shell **top** und starten Sie es anschließend erneut. Überprüfen Sie, ob die neue Anzeige die gespeicherte Konfiguration verwendet.

```
top - 00:58:21 up 43 min, 2 users, load average: 0.29, 0.28, 0.20
Tasks: 105 total, 1 running, 104 sleeping, 0 stopped, 0 zombie
%Cpu(s): 11.0/1.8 13[|||||||||||||]
MiB Mem : 18.7/1829.0 [||||||||||||||||||]
MiB Swap: 0.0/0.0 [
```

6. Kopieren Sie das Skript **process101** in eine neue Datei mit dem Namen **process102**. Bearbeiten Sie das Skript, um mehr künstliche CPU-Last zu erzeugen. Erhöhen Sie die Last von fünfzigtausend auf hunderttausend. Starten Sie den Prozess **process102** im Hintergrund.

- 6.1. Kopieren Sie in der linken Terminal-Shell mit dem Befehl **cp** das Skript **process101** in **process102**.

```
[student@serverb bin]$ cp process101 process102
```

- 6.2. Verwenden Sie den Befehl **vim**, um das Skript **process102** zu bearbeiten. Erhöhen Sie die Additionsprobleme von fünfzigtausend auf hunderttausend. Wechseln Sie mit **i** in den interaktiven Modus. Geben Sie **:wq** ein, um die Datei zu speichern.

```
[student@serverb bin]$ vim process102
#!/bin/bash
while true; do
    var=1
    while [[ var -lt 100000 ]]; do
        var=$((var+1))
    done
    sleep 1
done
```

- 6.3. Starten Sie den Prozess **process102** im Hintergrund.

```
[student@serverb bin]$ process102 &
[2] 20723
```

- 6.4. Überprüfen Sie mit dem Befehl **jobs**, ob beide Prozesse im Hintergrund ausgeführt werden.

```
[student@serverb bin]$ jobs
[1]-  Running                  process101 &
[2]+  Running                  process102 &
```

7. Überprüfen Sie in der rechten Terminal-Shell, ob der Prozess ausgeführt wird und die meisten CPU-Ressourcen verwendet. Die Auslastung sollte zwischen 25 % und 35 % liegen.

- 7.1. Überprüfen Sie in der rechten Terminal-Shell, ob der Prozess ausgeführt wird und die meisten CPU-Ressourcen verwendet. Die Auslastung sollte zwischen 25 % und 35 % liegen.

```
top - 20:14:16 up  6:46,  2 users,  load average: 0.58, 0.34, 0.18
Tasks: 112 total,   2 running, 110 sleeping,   0 stopped,   0 zombie
499 %Cpu(s):  0.0 us,  0.0 sy,  0.0 ni,100.0 id,  0.0 wa,  0.0 hi,  0.0 si,  0.0
      st
500 MiB Mem : 1829.4 total, 1428.7 free,   179.0 used,   221.8 buff/cache
501 MiB Swap: 1024.0 total, 1024.0 free,     0.0 used. 1492.1 avail Mem

 PID USER      PR  NI      VIRT      RES      SHR S %CPU %MEM      TIME+ COMMAND
```

```
20723 student 20 0 222448 3016 2764 S 24.7 0.2 0:53.28 process102
1180 student 20 0 222448 3056 2808 S 12.0 0.2 58:01.56 process101
...output omitted...
```



### Anmerkung

Wenn **process101** und **process102** nicht angezeigt werden, drücken Sie oben in der Prozessliste **Umschalt+p**, um sicherzustellen, dass top nach CPU-Auslastung sortiert wird.

8. Die durchschnittliche Auslastung liegt immer noch unter 1. Kopieren Sie **process101** in ein neues Skript mit dem Namen **process103**. Erhöhen Sie die Additionsanzahl auf achthunderttausend. Starten Sie **process103** im Hintergrund. Überprüfen Sie, ob die durchschnittliche Auslastung über 1 liegt. Es kann einige Minuten dauern, bis sich die durchschnittliche Auslastung ändert.
  - 8.1. Überprüfen Sie in der rechten Terminal-Shell, ob die durchschnittliche Auslastung unter 1 liegt.

```
top - 20:24:13 up 6:56, 2 users, load average: 0.43, 0.41, 0.29
...output omitted...
```

- 8.2. Kopieren Sie in der linken Terminal-Shell mit dem Befehl **cp** das Skript **process101** in ein neues Skript mit dem Namen **process103**.

```
[student@serverb bin]$ cp process101 process103
```

- 8.3. Verwenden Sie in der linken Terminal-Shell den Befehl **vim**, um das Skript **process103** zu bearbeiten. Erhöhen Sie die Additionsanzahl auf achthunderttausend. Wechseln Sie mit **i** in den interaktiven Modus. Geben Sie **:wq** ein, um die Datei zu speichern.

```
[student@serverb bin]$ vim process103
#!/bin/bash
while true; do
    var=1
    while [[ var -lt 800000 ]]; do
        var=$((var+1))
    done
    sleep 1
done
```

- 8.4. Starten Sie **process103** im Hintergrund. Die CPU-Auslastung liegt zwischen 60 % und 85 %.

```
[student@serverb bin]$ process103 &
[3] 22751
```

- 8.5. Überprüfen Sie, ob alle drei Jobs im Hintergrund ausgeführt werden.

```
[student@serverb bin]$ jobs
[1]  Running                  process101 &
[2]- Running                  process102 &
[3]+ Running                  process103 &
```

- 8.6. Überprüfen Sie sich im rechten Terminalfenster, ob die durchschnittliche Auslastung über 1 liegt.

```
top - 20:45:34 up 7:17, 2 users, load average: 1.10, 0.90, 0.64
```

9. Ändern Sie in der linken Terminal-Shell den Benutzer in **root**. Unterbrechen Sie den Prozess **process101**. Listen Sie die verbleibenden Jobs auf. Sie werden feststellen, dass der Prozessstatus für **process101** jetzt **T** lautet.

- 9.1. Führen Sie den Befehl **su -** aus, um **root**-Berechtigungen zu erhalten. Das Passwort lautet **redhat**.

```
[student@serverb bin]$ su -
Password: redhat
```

- 9.2. Unterbrechen Sie mit dem Befehl **pkill** und der Option **-SIGSTOP** den Prozess **process101**.

```
[root@serverb ~]# pkill -SIGSTOP process101
```

- 9.3. Vergewissern Sie sich in der rechten Terminal-Shell, dass der Prozess **process101** nicht mehr ausgeführt wird.

```
top - 20:52:01 up 7:24, 2 users, load average: 1.19, 1.19, 0.89
Tasks: 112 total, 2 running, 110 sleeping, 0 stopped, 0 zombie
499 %Cpu(s): 0.0 us, 0.0 sy, 0.0 ni,100.0 id, 0.0 wa, 0.0 hi, 0.0 si, 0.0 st
500 MiB Mem : 1829.4 total, 1428.7 free, 179.0 used, 221.8 buff/cache
501 MiB Swap: 1024.0 total, 1024.0 free, 0.0 used. 1492.1 avail Mem

 PID USER      PR  NI    VIRT    RES    SHR S %CPU %MEM     TIME+ COMMAND
24043 student   20    0 222448  2992  2744 R 66.1    0.2   6:59.50 process103
20723 student   20    0 222448  3016  2764 R 29.9    0.2  11:04.84 process102
...output omitted...
```

- 9.4. Führen Sie in der linken Terminal-Shell den Befehl **ps jT** aus, um die verbleibenden Jobs anzuzeigen.

```
[root@serverb ~]# ps jT
  PPID  PID  PGID  SID TTY      TPGID STAT   UID    TIME COMMAND
...output omitted...
 27138 1180 1180 27138 pts/0      28558 T     1000   3:06 /bin/bash /home/student/
bin/process101
 27138 20723 20723 27138 pts/0      28558 R     1000   1:23 /bin/bash /home/student/
bin/process102
 27138 24043 24043 27138 pts/0      28558 R     1000   2:35 /bin/bash /home/student/
bin/process103
...output omitted...
```

Beachten Sie, dass **process101** den Status **T** hat. Dies bedeutet, dass der Prozess derzeit unterbrochen ist.

#### 10. Setzen Sie den Prozess **process101** fort.

- 10.1. Setzen Sie in der linken Terminal-Shell mit dem Befehl **pkill** und der Option **-SIGCONT** den Prozess **process101** fort.

```
[root@serverb ~]# pkill -SIGCONT process101
```

- 10.2. Überprüfen Sie in der rechten Terminal-Shell, ob der Prozess wieder ausgeführt wird.

```
top - 20:57:02 up 7:29, 2 users, load average: 1.14, 1.20, 0.99
Tasks: 112 total, 2 running, 110 sleeping, 0 stopped, 0 zombie
%Cpu(s): 0.0 us, 0.0 sy, 0.0 ni,100.0 id, 0.0 wa, 0.0 hi, 0.0 si, 0.0 st
MiB Mem : 1829.4 total, 1428.7 free, 179.0 used, 221.8 buff/cache
MiB Swap: 1024.0 total, 1024.0 free, 0.0 used. 1492.1 avail Mem

 PID USER      PR  NI    VIRT    RES    SHR S %CPU %MEM     TIME+ COMMAND
24043 student    20    0 222448  2992  2744 R 66.8  0.2  10:40.61 process103
20723 student    20    0 222448  3016  2764 S 24.9  0.2  12:25.10 process102
1180 student    20    0 222448  3056  2808 S 17.9  0.2  64:07.99 process101
```

#### 11. Beenden Sie in der Befehlszeile **process101**, **process102** und **process103**. Vergewissern Sie sich, dass die Prozesse nicht mehr in **top** angezeigt werden.

- 11.1. Beenden Sie in der linken Terminal-Shell mit dem Befehl **pkill process101**, **process102** und **process103**.

```
[root@serverb ~]# pkill process101
[root@serverb ~]# pkill process102
[root@serverb ~]# pkill process103
```

- 11.2. Vergewissern Sie sich in der rechten Terminal-Shell, dass die Prozesse nicht mehr in **top** angezeigt werden.

```
top - 21:05:06 up 7:37, 2 users, load average: 1.26, 1.29, 1.12
Tasks: 112 total, 2 running, 110 sleeping, 0 stopped, 0 zombie
499 %Cpu(s): 0.0 us, 0.0 sy, 0.0 ni,100.0 id, 0.0 wa, 0.0 hi, 0.0 si, 0.0 st
500 MiB Mem : 1829.4 total, 1428.7 free, 179.0 used, 221.8 buff/cache
```

```
501 MiB Swap: 1024.0 total, 1024.0 free, 0.0 used. 1492.1 avail Mem

PID USER PR NI VIRT RES SHR S %CPU %MEM TIME+ COMMAND
 1 root 20 0 178536 13488 8996 S 0.0 0.7 0:01.21 systemd
 2 root 20 0 0 0 0 S 0.0 0.0 0:00.00 kthreadd
 3 root 0 -20 0 0 0 I 0.0 0.0 0:00.00 rcu_gp
...output omitted...
```

12. Beenden Sie in der linken Terminal-Shell den Benutzer **root**. Halten Sie in der rechten Terminal-Shell den Befehl **top** an. Beenden Sie auf **serverb** beide Terminalfenster.

- 12.1. Melden Sie mit dem Befehl **exit** den Benutzer **root** ab.

```
[root@serverb ~]# exit
logout
[1]  Terminated                  process101
[2]  Terminated                  process102
[3]-  Terminated                  process103
```

- 12.2. Beenden Sie alle Terminalfenster.

```
[student@serverb bin]$ exit
[student@workstation ~]$
```

- 12.3. Drücken Sie in der rechten Terminal-Shell **q**, um **top** zu beenden. Melden Sie sich mit dem Befehl **exit** ab.

```
[student@serverb ~]$ exit
logout
Connection to serverb closed.
[student@workstation ~]$
```

## Bewertung

Führen Sie auf **workstation** das Skript **lab processes-review grade** aus, um den Erfolg dieser Übung zu bestätigen.

```
[student@workstation ~]$ lab processes-review grade
```

## Beenden

Führen Sie auf **workstation** das Skript **lab processes-review finish** aus, um die praktische Übung zu beenden.

```
[student@workstation ~]$ lab processes-review finish
```

Damit ist die praktische Übung abgeschlossen.

# Zusammenfassung

---

In diesem Kapitel erhielten Sie folgende Informationen:

- Ein Prozess ist eine laufende Instanz eines ausführbaren Programms. Prozessen kann der Status „Ausgeführt“, „Im Ruhezustand“, „Angehalten“ oder „Zombie“ zugewiesen werden. Mit dem Befehl **ps** werden Prozesse aufgelistet.
- Jedes Terminal ist eine eigenständige Sitzung und kann einen Vordergrundprozess und unabhängige Hintergrundprozesse aufweisen. Der Befehl **jobs** zeigt Prozesse innerhalb einer Terminalsitzung an.
- Ein Signal ist ein Software-Interrupt, der Ereignisse an ein ausführendes Programm meldet. Die Befehle **kill**, **pkill** und **killall** verwenden Signale zum Steuern von Prozessen.
- Die durchschnittliche Auslastung ist eine Schätzung der Auslastung des Systems. Mit dem Befehl **top**, **uptime** oder **w** können Sie die Werte zur durchschnittlichen Auslastung anzeigen.



## Kapitel 9

# Steuern von Services und Daemons

### Ziel

Steuern und Überwachen von Netzwerkservices und System-Daemons mit Systemd

### Ziele

- Auflisten der System-Daemons und Netzwerkservices, die von **systemd**-Service- und Socket-Units gestartet wurden
- Steuern von System-Daemons und Netzwerkservices mit **systemctl**

### Abschnitte

- Identifizieren von automatisch gestarteten Systemprozessen (und angeleitete Übung)
- Kontrollieren der Systemservices (und angeleitete Übung)

### Praktische Übung

Steuern von Diensten und Daemons

# Identifizieren automatisch gestarteter Systemprozesse

## Ziele

Nach Abschluss dieses Abschnitts sollten Sie in der Lage sein, die System-Daemons und Netzwerkservices aufzulisten, die durch **systemd**-Service- und Socket-Units gestartet werden.

## Einführung in **systemd**

Der **systemd**-Daemon verwaltet den Start für Linux, einschließlich des Servicestarts und der allgemeinen Serviceverwaltung. Systemressourcen, Server-Daemons und andere Prozesse können sowohl beim Systemstart als auch auf einem laufenden System aktiviert werden.

Daemons sind Prozesse, die entweder im Hintergrund warten oder ausgeführt werden und verschiedene Aufgaben ausführen. Normalerweise werden Daemons beim Systemstart automatisch ausgeführt, bis das System heruntergefahren oder der Daemon manuell beendet wird. Die Namen vieler Daemon-Programme enden mit dem Buchstaben **d**.

Ein Service im Sinne von **systemd** bezieht sich häufig auf einen oder mehrere Daemons, aber durch Starten oder Beenden kann ein Service auch einmalige Änderungen am Systemstatus vornehmen, ohne dass danach ein Daemon-Prozess ausgeführt wird (genannt **oneshot**).

In Red Hat Enterprise Linux ist der erste Prozess (PID 1), der gestartet wird **systemd**. Dies sind einige der Funktionen von **systemd**:

- Parallelisierung (gleichzeitiger Start mehrerer Services), durch die der Bootvorgang des Systems beschleunigt wird.
- Start von Daemons auf Abfrage ohne separaten Service.
- Automatisches Service-Abhängigkeitsmanagement, das lange Timeouts verhindern kann. Zum Beispiel versucht ein netzwerkabhängiger Service nicht zu starten, solange das Netzwerk nicht verfügbar ist.
- Methode zur Nachverfolgung zusammenhängender Prozesse mittels Linux-Kontrollgruppen

## Beschreiben von Service-Units

**systemd** verwendet *Units*, um verschiedene Arten von Objekten zu verwalten. Hier finden Sie eine Liste der gängigen Unit-Typen:

- Service-Units tragen die Erweiterung **.service** und stellen Systemservices dar. Mit diesem Unit-Typ werden häufig verwendete Daemons wie ein Webserver gestartet.
- Socket-Unitshaben die Erweiterung **.socket** und stellen die Sockets dar, die für die Kommunikation zwischen Prozessen (IPC), die **systemd** überwachen soll, verantwortlich sind. Wenn ein Client eine Verbindung zum Socket herstellt, startet **systemd** einen Daemon und leitet die Verbindung weiter. Mit Socket-Units wird das Ausführen eines Service beim Systemstart verzögert oder auf Abfrage ein seltener verwendeter Service gestartet.
- Path-Units haben die Erweiterung **.path** und verzögern die Aktivierung eines Service, bis eine bestimmte Dateisystemänderung vorgenommen wurde. Sie kommen häufig bei Services mit spool-Verzeichnissen wie Druckersystemen zum Einsatz.

Der Befehl **systemctl** wird zur Verwaltung von Units verwendet. Zeigen Sie beispielsweise verfügbare Unit-Typen mit dem Befehl **systemctl -t help**.



### Wichtig

Durch die Verwendung von **systemctl** können Sie Unit-Namen, Prozessbaumeinträge und Einheitenbeschreibungen abkürzen.

## Auflisten von Service-Units

Den Befehl **systemctl** verwendet man, um den aktuellen Status des Systems zu prüfen. Der folgende Befehl listet beispielsweise alle aktuell geladenen Service-Units auf und paginiert die Ausgabe mit **less**.

```
[root@host ~]# systemctl list-units --type=service
UNIT                                     LOAD ACTIVE SUB   DESCRIPTION
atd.service                               loaded active running Job spooling tools
auditd.service                            loaded active running Security Auditing Service
chronyd.service                           loaded active running NTP client/server
crond.service                            loaded active running Command Scheduler
dbus.service                              loaded active running D-Bus System Message Bus
...output omitted...
```

Die obige Ausgabe begrenzt den aufgeführten Einheitentyp auf Service-Units mit der Option **--type=service**. Die Ausgabe besteht aus den folgenden Spalten:

### Spalten in der Befehlausgabe **systemctl list-units**.

#### UNIT

Der Name der Service-Unit.

#### LOAD

Ob **systemd** die Konfiguration der Unit ordnungsgemäß analysiert und die Unit gespeichert hat.

#### ACTIVE

Der hochrangige Aktivierungszustand der Unit. Diese Informationen zeigen an, ob das Gerät erfolgreich gestartet wurde oder nicht.

#### SUB

Der Aktivierungszustand der Unit auf niedriger Stufe. Diese Informationen zeigen detailliertere Informationen zur Unit an. Die Informationen variieren je nach Gerätetyp, Zustand und Ausführungsart der Unit.

#### BESCHREIBUNG

Die Kurzbeschreibung der Unit.

Standardmäßig listet der Befehl **systemctl list-units --type=service** nur die Service-Units mit dem Aktivierungsstatus **active** auf. Die Option **--all** listet alle Service-Units unabhängig vom Aktivierungsstatus auf. Verwenden Sie die Option **--state=** zum Filtern nach den Werten **LOAD**, **ACTIVE** oder **SUB** in den Feldern.

```
[root@host ~]# systemctl list-units --type=service --all
UNIT                                     LOAD ACTIVE SUB   DESCRIPTION
atd.service                               loaded active running Job spooling tools
```

**Kapitel 9 |** Steuern von Services und Daemons

```
auditd.service           loaded   active  running Security Auditing ...
auth-rpcgss-module.service loaded   inactive dead    Kernel Module ...
chronyd.service          loaded   active  running NTP client/server
cpupower.service         loaded   inactive dead    Configure CPU power ...
crond.service            loaded   active  running Command Scheduler
dbus.service              loaded   active  running D-Bus System Message Bus
● display-manager.service not-found inactive dead    display-manager.service
...output omitted...
```

Der Befehl **systemctl** ohne Argumente listet Units auf, die geladen und aktiv sind.

```
[root@host ~]# systemctl
UNIT                      LOAD ACTIVE SUB      DESCRIPTION
proc-sys-fs-binfmt_misc.automount  loaded active waiting  Arbitrary...
sys-devices-....device        loaded active plugged  Virtio network...
sys-subsystem-net-devices-ens3.device loaded active plugged  Virtio network...
...
-.mount                   loaded active mounted  Root Mount
boot.mount                loaded active mounted  /boot
...
systemd-ask-password-plymouth.path  loaded active waiting  Forward Password...
systemd-ask-password-wall.path     loaded active waiting  Forward Password...
init.scope                 loaded active running   System and Servi...
session-1.scope            loaded active running   Session 1 of...
atd.service                loaded active running   Job spooling tools
auditd.service             loaded active running   Security Auditing...
chronyd.service            loaded active running   NTP client/server
crond.service              loaded active running   Command Scheduler
...output omitted...
```

Der Befehl **systemctl list-units** zeigt Units an, die der Service **systemd** versucht zu parsen und in den Speicher zu laden. Es werden keine installierten Services angezeigt, die nicht aktiviert sind. Um den Status aller installierten Unit-Dateien anzuzeigen, verwenden Sie den Befehl **systemctl list-unit-files**. Beispiel:

```
[root@host ~]# systemctl list-unit-files --type=service
UNIT FILE                      STATE
arp-ethers.service              disabled
atd.service                     enabled
auditd.service                  enabled
auth-rpcgss-module.service     static
autovt@.service                 enabled
blk-availability.service       disabled
...output omitted...
```

In der Ausgabe des Befehls **systemctl list-units-files**, sind **aktiviert**, **deaktiviert**, **statisch** und **maskiert** gültige Einträge für die **STATUS**-Felder.

## Anzeigen von Servicestatus

Zeigen Sie den Status einer bestimmten Unit mit **systemctl status name.type** an. Wenn kein Unit-Typ angegeben wird, zeigt **systemctl** den Status einer Service-Unit an, sofern eine existiert.

```
[root@host ~]# systemctl status sshd.service
● sshd.service - OpenSSH server daemon
  Loaded: loaded (/usr/lib/systemd/system/sshd.service; enabled; vendor preset: enabled)
  Active: active (running) since Thu 2019-02-14 12:07:45 IST; 7h ago
    Main PID: 1073 (sshd)
       CGroup: /system.slice/sshd.service
                 └─1073 /usr/sbin/sshd -D ...

Feb 14 11:51:39 host.example.com systemd[1]: Started OpenSSH server daemon.
Feb 14 11:51:39 host.example.com sshd[1073]: Could not load host key: /etc/...
Feb 14 11:51:39 host.example.com sshd[1073]: Server listening on 0.0.0.0 ...
Feb 14 11:51:39 host.example.com sshd[1073]: Server listening on :: port 22.
Feb 14 11:53:21 host.example.com sshd[1270]: error: Could not load host k...
Feb 14 11:53:22 host.example.com sshd[1270]: Accepted password for root f...
...output omitted...
```

Dieser Befehl zeigt den aktuellen Status des Service an. Die Bedeutung der Felder:

#### Service-Unit-Informationen

| Feld      | Beschreibung                                                       |
|-----------|--------------------------------------------------------------------|
| Geladen   | Ob die Service-Unit in den Speicher geladen wurde.                 |
| Active    | Ob die Service-Unit läuft und wenn ja, seit wann sie läuft.        |
| Haupt-PID | Die Hauptprozess-ID des Service, einschließlich des Befehlsnamens. |
| Status    | Zusätzliche Informationen zu dem Service.                          |

Die Statusanzeige enthält mehrere Schlüsselwörter, die Aufschluss über den Status des Service geben:

#### Servicezustände in der Ausgabe von systemctl

| Schlüsselwort    | Beschreibung                                     |
|------------------|--------------------------------------------------|
| geladen          | Konfigurationsdatei der Unit wurde verarbeitet   |
| active (läuft)   | Führt einen oder mehrere Prozesse aus            |
| active (beendet) | Hat einen einmaligen Prozess erfolgreich beendet |
| active (wartend) | Wird ausgeführt, aber wartet auf ein Ereignis    |
| inaktiv          | Wird nicht ausgeführt                            |
| aktiviert        | Wird beim Systemstart ausgeführt.                |
| deaktiviert      | Wird beim Systemstart nicht ausgeführt           |

| Schlüsselwort | Beschreibung                                                                                    |
|---------------|-------------------------------------------------------------------------------------------------|
| statisch      | Kann nicht aktiviert werden, aber kann automatisch durch eine aktivierte Unit gestartet werden. |

**Anmerkung**

Der Befehl **systemctl status NAME** ersetzt den Befehl **service NAME status** in Red Hat Enterprise Linux 6 und älteren Versionen.

## Überprüfen des Status eines Service

Der Befehl **systemctl** bietet Methoden zur Überprüfung der spezifischen Status eines Service. Verwenden Sie beispielsweise den folgenden Befehl, um zu überprüfen, ob eine Service-Unit gerade aktiv ist (läuft):

```
[root@host ~]# systemctl is-active sshd.service  
active
```

Der Befehl gibt den Status der Service-Unit zurück, der normalerweise **active** oder **inactive** lautet.

Führen Sie den folgenden Befehl aus, um zu überprüfen, ob eine Service-Unit beim Systemboot automatisch gestartet werden kann:

```
[root@host ~]# systemctl is-enabled sshd.service  
enabled
```

Der Befehl gibt zurück, ob das Starten der Service-Unit beim Systemboot aktiviert ist. Normalerweise ist dies entweder **enabled** oder **disabled**.

Führen Sie den folgenden Befehl aus, um zu überprüfen, ob die Unit während des Startvorgangs ausgefallen ist:

```
[root@host ~]# systemctl is-failed sshd.service  
active
```

Der Befehl gibt entweder **active** zurück, wenn sie richtig läuft oder **failed**, wenn beim Starten ein Fehler aufgetreten ist. Falls die Unit gestoppt wurde, wird **unknown** oder **inactive** zurückgemeldet.

Um alle fehlerhaften Units aufzulisten, führen Sie den Befehl **systemctl --failed --type=service** aus.



### Literaturhinweise

Manpages **systemd(1)**, **systemd.unit(5)**, **systemd.service(5)**,  
**systemd.socket(5)** und **systemctl(1)**

Weitere Informationen finden Sie im Kapitel *Managing services with systemd* im Handbuch *Red Hat Enterprise Linux 8 Configuring basic system settings* unter [https://access.redhat.com/documentation/en-us/red\\_hat\\_enterprise\\_linux/8/html-single/configuring\\_basic\\_system\\_settings/managing-services-with-systemd\\_configuring-basic-system-settings#managing-services-with-systemd\\_configuring-basic-system-settings](https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux/8/html-single/configuring_basic_system_settings/managing-services-with-systemd_configuring-basic-system-settings#managing-services-with-systemd_configuring-basic-system-settings)

## ► Angeleitete Übung

# Identifizieren automatisch gestarteter Systemprozesse

In dieser Übung listen Sie die installierten Service-Units auf und ermitteln, welche Services auf einem Server derzeit aktiviert und aktiv sind.

## Ergebnisse

Sie sollten in der Lage sein, installierte Service-Units aufzulisten und aktive und aktivierte Services im System zu identifizieren.

## Bevor Sie Beginnen

Melden Sie sich als Benutzer **student** mit dem Passwort **student** bei **workstation** an.

Führen Sie auf **workstation** den Befehl **lab services-identify start** aus. Dieser Befehl führt ein Startskript aus, das ermittelt, ob der Host **servera** im Netzwerk erreichbar ist.

```
[student@workstation ~]$ lab services-identify start
```

- 1. Melden Sie sich mit dem Befehl **ssh** bei **servera** als Benutzer **student** an. Die Systeme sind für die Verwendung von SSH-Schlüsseln zur Authentifizierung konfiguriert. Daher ist kein Passwort erforderlich, um sich bei **servera** anzumelden.

```
[student@workstation ~]$ ssh student@servera  
...output omitted...  
[student@servera ~]$
```

- 2. Listen Sie alle auf **servera** installierten Service-Units auf.

```
[student@servera ~]$ systemctl list-units --type=service  
UNIT           LOAD   ACTIVE SUB     DESCRIPTION  
atd.service    loaded  active  running Job spooling tools  
auditd.service loaded  active  running Security Auditing Service  
chronyd.service loaded  active  running NTP client/server  
crond.service  loaded  active  running Command Scheduler  
dbus.service   loaded  active  running D-Bus System Message Bus  
...output omitted...
```

Drücken Sie **q**, um den Befehl zu beenden.

- 3. Rufen Sie die Liste aller aktiven und inaktiven Socket-Units auf **servera** auf.

```
[student@servera ~]$ systemctl list-units --type=socket --all
UNIT              LOAD   ACTIVE   SUB      DESCRIPTION
dbus.socket       loaded  active   running  D-Bus System Message Bus Socket
dm-event.socket   loaded  active   listening Device-mapper event daemon FIFOs
lvm2-lvmpoold.socket loaded  active   listening LVM2 poll daemon socket
...output omitted...
systemd-udevd-control.socket    loaded  active   running  udev Control Socket
systemd-udevd-kernel.socket     loaded  active   running  udev Kernel Socket

LOAD   = Reflects whether the unit definition was properly loaded.
ACTIVE = The high-level unit activation state, i.e. generalization of SUB.
SUB    = The low-level unit activation state, values depend on unit type.

12 loaded units listed.
To show all installed unit files use 'systemctl list-unit-files'.
```

- 4. Sehen Sie sich den Status des Service **chrony**d an. Dieser Service wird zum Synchronisieren der Netzwerkzeit (NTP) verwendet.
- 4.1. Rufen Sie den Status des Service **chrony**d auf. Achten Sie auf die Prozess-ID des aktiven Daemons.

```
[student@servera ~]$ systemctl status chronyd
● chronyd.service - NTP client/server
  Loaded: loaded (/usr/lib/systemd/system/chronyd.service; enabled; vendor
  preset: enabled)
  Active: active (running) since Wed 2019-02-06 12:46:57 IST; 4h 7min ago
    Docs: man:chronyd(8)
          man:chrony.conf(5)
  Process: 684 ExecStartPost=/usr/libexec/chrony-helper update-daemon
  (code=exited, status=0/SUCCESS)
  Process: 673 ExecStart=/usr/sbin/chronyd $OPTIONS (code=exited, status=0/
  SUCCESS)
  Main PID: 680 (chronyd)
    Tasks: 1 (limit: 11406)
   Memory: 1.5M
      CGroup: /system.slice/chronyd.service
              └─680 /usr/sbin/chronyd

... servera.lab.example.com systemd[1]: Starting NTP client/server...
...output omitted...
... servera.lab.example.com systemd[1]: Started NTP client/server.
... servera.lab.example.com chronyd[680]: Source 172.25.254.254 offline
... servera.lab.example.com chronyd[680]: Source 172.25.254.254 online
... servera.lab.example.com chronyd[680]: Selected source 172.25.254.254
```

Drücken Sie **q**, um den Befehl zu beenden.

- 4.2. Stellen Sie sicher, dass der aufgeführte Daemon ausgeführt wird. Im vorherigen Befehl lautet die Ausgabe der Prozess-ID, die dem Service **chrony**d zugeordnet ist, 680. Die Prozess-ID kann auf Ihrem System davon abweichen.

```
[student@servera ~]$ ps -p 680
 PID TTY      TIME CMD
 680 ?        00:00:00 chronyd
```

- 5. Sehen Sie sich den Status des Service **sshd** an. Dieser Service wird für die gesicherte, verschlüsselte Kommunikation zwischen Systemen verwendet.

- 5.1. Finden Sie heraus, ob der Service **sshd** beim Systemboot ausgeführt werden soll.

```
[student@servera ~]$ systemctl is-enabled sshd
enabled
```

- 5.2. Finden Sie heraus, ob der Service **sshd** aktiv ist, ohne alle Statusinformationen anzuzeigen.

```
[student@servera ~]$ systemctl is-active sshd
active
```

- 5.3. Zeigen Sie den Status des Service **sshd** auf.

```
[student@servera ~]$ systemctl status sshd
● sshd.service - OpenSSH server daemon
  Loaded: loaded (/usr/lib/systemd/system/sshd.service; enabled; vendor preset: enabled)
  Active: active (running) since Wed 2019-02-06 12:46:58 IST; 4h 21min ago
    Docs: man:sshd(8)
          man:sshd_config(5)
  Main PID: 720 (sshd)
     Tasks: 1 (limit: 11406)
    Memory: 5.8M
      CGroup: /system.slice/sshd.service
           └─720 /usr/sbin/sshd -D -oCiphers=aes256-gcm@openssh.com,
              chacha20-poly1305@openssh.com,aes256-ctr,
              aes256-cbc,aes128-gcm@openssh.com,aes128-ctr,
              aes128-cbc -oMACs= hmac-sha2-256-etm@openssh.com,hmac-sha>

... servera.lab.example.com systemd[1]: Starting OpenSSH server daemon...
... servera.lab.example.com sshd[720]: Server listening on 0.0.0.0 port 22.
... servera.lab.example.com systemd[1]: Started OpenSSH server daemon.
... servera.lab.example.com sshd[720]: Server listening on :: port 22.
... output omitted...
... servera.lab.example.com sshd[1380]: pam_unix(sshd:session): session opened for
user student by (uid=0)
```

Drücken Sie **q**, um den Befehl zu beenden.

- 6. Rufen Sie die Liste der aktivierten oder deaktivierten Status aller Service-Units auf.

```
[student@servera ~]$ systemctl list-unit-files --type=service
UNIT FILE                      STATE
arp-ethers.service               disabled
atd.service                      enabled
```

```
auditd.service           enabled
auth-rpcgss-module.service static
autovt@.service          enabled
blk-availability.service disabled
chrony-dnssrv@.service   static
chrony-wait.service      disabled
chronyd.service          enabled
...output omitted...
```

Drücken Sie **q**, um den Befehl zu beenden.

► 7. Beenden Sie **servera**.

```
[student@servera ~]$ exit
logout
Connection to servera closed.
[student@workstation]$
```

## Beenden

Führen Sie auf **workstation** das Skript **lab services-identify finish** aus, um diese Übung zu beenden.

```
[student@workstation ~]$ lab services-identify finish
```

Hiermit ist die angeleitete Übung beendet.

# Kontrollieren der Systemdienste

## Ziele

Nach Abschluss dieses Abschnitts sollten Sie System-Daemons und Netzwerkservices mittels des Befehls **systemctl** kontrollieren können.

## Starten und Stoppen von Services

Services müssen aus verschiedenen Gründen gestoppt oder manuell gestartet werden: Möglicherweise muss der Service aktualisiert werden, die Konfigurationsdatei muss möglicherweise geändert werden oder ein Service muss möglicherweise deinstalliert werden oder ein Administrator kann einen selten verwendeten Service manuell starten.

Um einen Service zu starten, vergewissern Sie sich zunächst, dass er nicht mit **systemctl status** ausgeführt wird. Verwenden Sie dann den Befehl **systemctl start** als **root**-Benutzer (mit **sudo** falls notwendig). Das folgende Beispiel zeigt, wie Sie den Service **sshd.service** starten:

```
[root@host ~]# systemctl start sshd.service
```

Wenn kein Servicetyp mit dem Servicenamen angegeben ist, sucht der Service **systemd** nach **.service**-Dateien für die Serviceverwaltung in Befehlen. Daher kann der obenstehende Befehl ausgeführt werden:

```
[root@host ~]# systemctl start sshd
```

Um einen gerade laufenden Service zu stoppen, verwenden Sie das Argument **stop** mit dem Befehl **systemctl**. Das folgende Beispiel zeigt, wie Sie den Service **sshd.service** stoppen:

```
[root@host ~]# systemctl stop sshd.service
```

## Neustarten und erneutes Laden von Services

Während eines Neustarts eines laufenden Service wird der Service angehalten und dann gestartet. Beim Neustart des Service ändert sich die Prozess-ID und beim Start wird eine neue Prozess-ID zugeordnet. Um einen laufenden Service neu zu starten, verwenden Sie das **restart** Argument mit dem Befehl **systemctl**. Das folgende Beispiel zeigt, wie Sie den Service **sshd.service** erneut starten:

```
[root@host ~]# systemctl restart sshd.service
```

Einige Services können ihre Konfigurationsdateien ohne Neustart erneut laden. Dieser Vorgang wird als *service reload* (Service neu laden) bezeichnet. Beim erneuten Laden eines Service wird die mit verschiedenen Serviceprozessen verknüpfte Prozess-ID nicht geändert. Um einen laufenden Service neu zu laden, verwenden Sie das Argument **reload** mit dem Befehl **systemctl**. Das folgende Beispiel zeigt, wie Sie den Service **sshd.service** neu laden:

```
[root@host ~]# systemctl reload sshd.service
```

Wenn Sie nicht sicher sind, ob der Service über die Funktionalität zum erneuten Laden der Konfigurationsdatei verfügt, verwenden Sie das Argument **reload-or-restart** mit dem Befehl **systemctl**. Der Befehl lädt die Konfigurationsänderungen neu, wenn die Funktion zum erneuten Laden verfügbar ist. Andernfalls startet der Befehl den Service neu, um die neuen Konfigurationsänderungen zu implementieren:

```
[root@host ~]# systemctl reload-or-restart sshd.service
```

## Auflisten von Unit-Abhängigkeiten

Bei einigen Services müssen zuerst andere Services ausgeführt werden, wodurch Abhängigkeiten zu den anderen Services geschaffen werden. Andere Services werden beim Systemboot nicht gestartet, lediglich bei Bedarf. In beiden Fällen starten systemd und **systemctl** Services nach Bedarf, entweder um die Abhängigkeit aufzulösen oder um einen selten verwendeten Service zu starten. Wenn beispielsweise der CUPS-Druckservice nicht ausgeführt wird und eine Datei im Druckspool-Verzeichnis abgelegt wird, startet das System mit CUPS zusammenhängende Daemons oder Befehle, um die Druckanforderung zu erfüllen.

```
[root@host ~]# systemctl stop cups.service
Warning: Stopping cups, but it can still be activated by:
  cups.path
  cups.socket
```

Sie müssen alle drei Units deaktivieren, um den Druckerservice auf einem System komplett zu deaktivieren. Durch das Deaktivieren des Service werden die Abhängigkeiten deaktiviert.

Der Befehl **systemctl list-dependencies UNIT** zeigt eine Hierarchiezuordnung von Abhängigkeiten an, um die Service-Unit zu starten. Um umgekehrte Abhängigkeiten (Units, die vom angegebenen Wert abhängen) aufzulisten, benutzen Sie die Option **--reverse** mit dem Befehl.

```
[root@host ~]# systemctl list-dependencies sshd.service
sshd.service
• └─system.slice
• └─sshd-keygen.target
•   └─sshd-keygen@ecdsa.service
•   └─sshd-keygen@ed25519.service
•   └─sshd-keygen@rsa.service
•   └─sysinit.target
...output omitted...
```

## Maskieren und Demaskieren von Services

Hin und wieder kann es vorkommen, dass auf einem System verschiedene Services installiert sind, die im Konflikt miteinander stehen. Es gibt beispielsweise mehrere Methoden zum Verwalten von Mail-Servern (z. B. **Postfix** und **Sendmail**). Das Maskieren eines Service verhindert, dass ein Administrator versehentlich einen Service startet, wodurch ein Konflikt entsteht. Durch die Maskierung wird in den Konfigurationsverzeichnissen eine Verknüpfung mit der Datei **/dev/null** erstellt, die den Start des Service verhindert.

```
[root@host ~]# systemctl mask sendmail.service  
Created symlink /etc/systemd/system/sendmail.service → /dev/null.
```

```
[root@host ~]# systemctl list-unit-files --type=service  
UNIT FILE STATE  
...output omitted...  
sendmail.service masked  
...output omitted...
```

Der Versuch, eine maskierte Service-Unit zu starten, schlägt mit der folgenden Ausgabe fehl:

```
[root@host ~]# systemctl start sendmail.service  
Failed to start sendmail.service: Unit sendmail.service is masked.
```

Verwenden Sie den Befehl **systemctl unmask** zum Demaskieren der Service-Unit.

```
[root@host ~]# systemctl unmask sendmail  
Removed /etc/systemd/system/sendmail.service.
```



### Wichtig

Ein deaktivierter Service kann beim Systemboot automatisch oder durch andere Unit-Dateien gestartet werden. Er wird jedoch nicht automatisch beim Systemboot gestartet. Ein maskierter Service kann nicht manuell oder automatisch gestartet werden.

## Konfigurieren von Services für das Starten oder Beenden beim Systemboot

Das Starten eines Service auf einem laufenden System garantiert nicht, dass der Service beim nächsten Booten des Systems automatisch gestartet wird. Ebenso führt das Beenden eines Service auf einem laufenden System nicht dazu, dass er beim nächsten Booten des Systems nicht wieder gestartet wird. Entsprechende Links in den **systemd** Konfigurationsverzeichnissen ermöglichen es, den Service beim Systemboot zu starten. Die Befehle **systemctl** können diese Links erstellen oder entfernen.

Um einen Service beim Systemboot zu starten, verwenden Sie den Befehl **systemctl enable**.

```
[root@root ~]# systemctl enable sshd.service  
Created symlink /etc/systemd/system/multi-user.target.wants/sshd.service → /usr/  
lib/systemd/system/sshd.service.
```

Der obige Befehl erstellt einen symbolischen Link aus der Datei der Service-Unit, normalerweise im Verzeichnis **/usr/lib/systemd/system**, an den Ort auf der Festplatte, wo **systemd** nach Dateien sucht, die sich im Verzeichnis **/etc/systemd/system/TARGETNAME.target.wants** befinden. Durch die Aktivierung eines Service wird der Service in der aktuellen Sitzung nicht gestartet. Um den Service zu starten und so zu konfigurieren, dass er beim Systemboot automatisch gestartet wird, führen Sie die Befehle **systemctl start** und **systemctl enable** aus.

Um den automatischen Start des Service zu deaktivieren, verwenden Sie den folgenden Befehl, der den symbolischen Link entfernt, der beim Aktivieren eines Service erstellt wurde. Beachten Sie, dass das Deaktivieren eines Service den Service nicht beendet.

```
[root@host ~]# systemctl disable sshd.service
Removed /etc/systemd/system/multi-user.target.wants/sshd.service.
```

Um zu überprüfen, ob der Service aktiviert oder deaktiviert ist, verwenden Sie den Befehl **systemctl is-enabled**.

## Übersicht der systemctl-Befehle

Services können auf einem laufenden System gestartet oder beendet werden. Ebenso kann festgelegt werden, ob sie beim Systemstart ausgeführt werden oder nicht.

### Nützliche Befehle für die Serviceverwaltung

| Aufgabe                                                                                                      | Befehl                                  |
|--------------------------------------------------------------------------------------------------------------|-----------------------------------------|
| Detaillierte Informationen zum Status einer Unit anzeigen.                                                   | <b>systemctl status UNIT</b>            |
| Einen Service auf einem laufenden System beenden.                                                            | <b>systemctl stop UNIT</b>              |
| Einen Service auf einem laufenden System starten.                                                            | <b>systemctl start UNIT</b>             |
| Einen Service auf einem laufenden System neu starten.                                                        | <b>systemctl restart UNIT</b>           |
| Neu laden der Konfigurationsdatei eines laufenden Service.                                                   | <b>systemctl reload UNIT</b>            |
| Das Ausführen eines Service vollständig unterbinden, sowohl durch manuelle Eingabe als auch beim Systemboot. | <b>systemctl mask UNIT</b>              |
| Einen maskierten Service verfügbar machen.                                                                   | <b>systemctl unmask UNIT</b>            |
| Einen Service konfigurieren, damit er beim Systemstart ausgeführt wird.                                      | <b>systemctl enable UNIT</b>            |
| Einen Service konfigurieren, damit er beim Systemstart nicht ausgeführt wird.                                | <b>systemctl disable UNIT</b>           |
| Einheiten auflisten, die zum Starten der angegebenen Unit benötigt werden.                                   | <b>systemctl list-dependencies UNIT</b> |



### Literaturhinweise

Manpages **systemd(1)**, **systemd.unit(5)**, **systemd.service(5)**,  
**systemd.socket(5)** und **systemctl(1)**

Weitere Informationen finden Sie im Kapitel *Managing services with systemd* im Handbuch *Red Hat Enterprise Linux 8 Configuring basic system settings* unter [https://access.redhat.com/documentation/en-us/red\\_hat\\_enterprise\\_linux/8/html-single/configuring\\_basic\\_system\\_settings/managing-services-with-systemd\\_configuring-basic-system-settings#managing-system-services\\_managing-services-with-systemd](https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux/8/html-single/configuring_basic_system_settings/managing-services-with-systemd_configuring-basic-system-settings#managing-system-services_managing-services-with-systemd)

## ► Angeleitete Übung

# Kontrollieren der Systemdienste

In dieser Übung werden Sie **systemctl** verwenden, um einen durch Systemd verwalteten Service zu stoppen, zu starten, neu zu starten, neu zu laden, zu aktivieren und zu deaktivieren.

## Ergebnisse

Sie sollten in der Lage sein, den Befehl **systemctl** zur Steuerung von **systemd**-verwalteten Services zu verwenden.

## Bevor Sie Beginnen

Melden Sie sich als Benutzer **student** mit dem Passwort **student** bei **workstation** an.

Führen Sie auf **workstation** den Befehl **lab services-control start** aus. Dieser Befehl führt ein Startskript aus, das ermittelt, ob der Host **servera** im Netzwerk erreichbar ist. Das Skript sorgt auch dafür, dass die Services **sshd** und **chronyd** weiter auf **servera** laufen.

```
[student@workstation ~]$ lab services-control start
```

- 1. Melden Sie sich mit dem Befehl **ssh** bei **servera** als Benutzer **student** an. Die Systeme sind für die Verwendung von SSH-Schlüsseln zur Authentifizierung konfiguriert. Daher ist kein Passwort erforderlich.

```
[student@workstation ~]$ ssh student@servera
...output omitted...
[student@servera ~]$
```

- 2. Führen Sie die Befehle **systemctl restart** und **systemctl reload** für den **sshd**-Service aus. Beachten Sie die unterschiedlichen Ergebnisse bei der Ausführung dieser Befehle.
- 2.1. Rufen Sie den Status des Service **sshd** auf. Beachten Sie die Prozess-ID des Daemons **sshd**.

```
[student@servera ~]$ systemctl status sshd
● sshd.service - OpenSSH server daemon
  Loaded: loaded (/usr/lib/systemd/system/sshd.service; enabled; vendor preset: enabled)
  Active: active (running) since Wed 2019-02-06 23:50:42 EST; 9min ago
    Docs: man:sshd(8)
          man:sshd_config(5)
   Main PID: 759 (sshd)
```

**Kapitel 9** | Steuern von Services und Daemons

```
Tasks: 1 (limit: 11407)
Memory: 5.9M
...output omitted...
```

Drücken Sie **q**, um den Befehl zu beenden.

- 2.2. Starten Sie den Service **sshd** neu und sehen Sie sich dessen Status an. Die Prozess-ID des Daemons muss sich ändern.

```
[student@servera ~]$ sudo systemctl restart sshd
[sudo] password for student: student
[student@servera ~]$ systemctl status sshd
● sshd.service - OpenSSH server daemon
  Loaded: loaded (/usr/lib/systemd/system/sshd.service; enabled; vendor preset: enabled)
  Active: active (running) since Wed 2019-02-06 23:50:42 EST; 9min ago
    Docs: man:sshd(8)
          man:sshd_config(5)
Main PID: 1132 (sshd)
  Tasks: 1 (limit: 11407)
  Memory: 5.9M
...output omitted...
```

In der vorherigen Ausgabe wurde die Prozess-ID 759 in 1132 geändert (auf Ihrem System werden es wahrscheinlich andere Zahlen sein). Drücken Sie **q**, um den Befehl zu beenden.

- 2.3. Laden Sie den Service **sshd** neu und sehen Sie sich dessen Status an. Die Prozess-ID des Daemons darf sich nicht ändern und die Verbindungen werden nicht unterbrochen.

```
[student@servera ~]$ sudo systemctl reload sshd
[student@servera ~]$ systemctl status sshd
● sshd.service - OpenSSH server daemon
  Loaded: loaded (/usr/lib/systemd/system/sshd.service; enabled; vendor preset: enabled)
  Active: active (running) since Wed 2019-02-06 23:50:42 EST; 9min ago
    Docs: man:sshd(8)
          man:sshd_config(5)
Main PID: 1132 (sshd)
  Tasks: 1 (limit: 11407)
  Memory: 5.9M
...output omitted...
```

Drücken Sie **q**, um den Befehl zu beenden.

- 3. Stellen Sie sicher, dass der Service **chronyd** ausgeführt wird.

```
[student@servera ~]$ systemctl status chronyd
● chronyd.service - NTP client/server
  Loaded: loaded (/usr/lib/systemd/system/chronyd.service; enabled; vendor
  preset: enabled)
    Active: active (running) since Wed 2019-02-06 23:50:38 EST; 1h 25min ago
      Docs: man:chronyd(8)
...output omitted...
```

Drücken Sie **q**, um den Befehl zu beenden.

- 4. Beenden Sie den Service **chronyd** und sehen Sie sich dessen Status an.

```
[student@servera ~]$ sudo systemctl stop chronyd
[student@servera ~]$ systemctl status chronyd
● chronyd.service - NTP client/server
  Loaded: loaded (/usr/lib/systemd/system/chronyd.service; enabled; vendor
  preset: enabled)
  Active: inactive (dead) since Thu 2019-02-07 01:20:34 EST; 44s ago
    ...output omitted...
... servera.lab.example.com chronyd[710]: System clock wrong by 1.349113 seconds,
adjustment started
... servera.lab.example.com systemd[1]: Stopping NTP client/server...
... servera.lab.example.com systemd[1]: Stopped NTP client/server.
```

Drücken Sie **q**, um den Befehl zu beenden.

- 5. Finden Sie heraus, ob der Service **sshd** für die Ausführung beim Booten des Systems aktiviert ist.

```
[student@server ~]$ systemctl is-enabled chronyd
enabled
```

- 6. Starten Sie **servera** neu und sehen Sie sich danach den Status des Service **chronyd** an.

```
[student@servera ~]$ sudo systemctl reboot
Connection to servera closed by remote host.
Connection to servera closed.
[student@workstation ~]$
```

Melden Sie sich als **student** auf **servera** an und sehen Sie sich den Status des Service**chronyd** an.

```
[student@workstation ~]$ ssh student@servera
...output omitted...
[student@servera ~]$ systemctl status chronyd
● chronyd.service - NTP client/server
  Loaded: loaded (/usr/lib/systemd/system/chronyd.service; enabled; vendor
  preset: enabled)
  Active: active (running) since Thu 2019-02-07 01:48:26 EST; 5min ago
    ...output omitted...
```

Drücken Sie **q**, um den Befehl zu beenden.

## Kapitel 9 | Steuern von Services und Daemons

- 7. Deaktivieren Sie den Service **chronyd**, damit dieser nicht beim Booten des Systems ausgeführt wird, und sehen Sie sich danach dessen Status an.

```
[student@servera ~]$ sudo systemctl disable chronyd
[sudo] password for student:
Removed /etc/systemd/system/multi-user.target.wants/chronyd.service.
[student@servera ~]$ systemctl status chronyd
● chronyd.service - NTP client/server
  Loaded: loaded (/usr/lib/systemd/system/chronyd.service; disabled; vendor
  preset: enabled)
    Active: active (running) since Thu 2019-02-07 01:48:26 EST; 5min ago
      ...output omitted...
```

Drücken Sie **q**, um den Befehl zu beenden.

- 8. Starten Sie **servera** neu und sehen Sie sich danach den Status des Service **chronyd** an.

```
[student@servera ~]$ sudo systemctl reboot
Connection to servera closed by remote host.
Connection to servera closed.
[student@workstation ~]$
```

Melden Sie sich als **student** auf **servera** an und sehen Sie sich den Status des Service**chronyd** an.

```
[student@workstation ~]$ ssh student@servera
...output omitted...
[student@servera ~]$ systemctl status chronyd
● chronyd.service - NTP client/server
  Loaded: loaded (/usr/lib/systemd/system/chronyd.service; disabled; vendor
  preset: enabled)
    Active: inactive (dead)
      Docs: man:chronyd(8)
             man:chrony.conf(5)
```

- 9. Beenden Sie **servera**.

```
[student@servera ~]$ exit
logout
Connection to servera closed.
[student@workstation]$
```

## Beenden

Führen Sie auf **workstation** das Skript **lab services-control finish** aus, um diese Übung zu beenden.

```
[student@workstation ~]$ lab services-control finish
```

Hiermit ist die angeleitete Übung beendet.

## ► Praktische Übung

# Steuern von Diensten und Daemons

### Leistungscheckliste

In dieser Übung werden Sie verschiedene Services so konfigurieren, dass sie auf der Grundlage einer von Ihnen bereitgestellten Spezifikation aktiviert oder deaktiviert, ausgeführt oder angehalten werden.

### Ergebnisse

Sie sollten in der Lage sein, Services zu aktivieren, zu deaktivieren, zu starten und zu stoppen.

### Bevor Sie Beginnen

Melden Sie sich als Benutzer **student** mit dem Passwort **student** bei **workstation** an.

Führen Sie auf **workstation** den Befehl **lab services-review start** aus. Dieser Befehl führt ein Startskript aus, das ermittelt, ob der Host **serverb** im Netzwerk erreichbar ist. Das Skript sorgt auch dafür, dass die Services **psacct** und **rsyslog** entsprechend auf **serverb** konfiguriert sind.

```
[student@workstation ~]$ lab services-review start
```

1. Starten Sie auf **serverb** den Service **psacct**.
2. Konfigurieren Sie den Service **psacct** so, dass er bei jedem Systemstart gestartet wird.
3. Beenden Sie den Service **rsyslog**.
4. Konfigurieren Sie den Service **rsyslog** so, dass dieser nicht beim Booten des Systems ausgeführt wird.
5. Starten Sie **serverb** neu, bevor Sie die praktische Übung bewerten.

### Bewertung

Führen Sie auf **workstation** das Skript **lab services-review grade** aus, um den Erfolg dieser praktischen Übung zu bestätigen.

```
[student@workstation ~]$ lab services-review grade
```

### Beenden

Führen Sie auf **workstation** das Skript **lab services-review finish** aus, um die praktische Übung zu beenden.

```
[student@workstation ~]$ lab services-review finish
```

Hiermit wird die praktische Übung abgeschlossen.

## ► Lösung

# Steuern von Diensten und Daemons

### Leistungscheckliste

In dieser Übung werden Sie verschiedene Services so konfigurieren, dass sie auf der Grundlage einer von Ihnen bereitgestellten Spezifikation aktiviert oder deaktiviert, ausgeführt oder angehalten werden.

### Ergebnisse

Sie sollten in der Lage sein, Services zu aktivieren, zu deaktivieren, zu starten und zu stoppen.

### Bevor Sie Beginnen

Melden Sie sich als Benutzer **student** mit dem Passwort **student** bei **workstation** an.

Führen Sie auf **workstation** den Befehl **lab services-review start** aus. Dieser Befehl führt ein Startskript aus, das ermittelt, ob der Host **serverb** im Netzwerk erreichbar ist. Das Skript sorgt auch dafür, dass die Services **psacct** und **rsyslog** entsprechend auf **serverb** konfiguriert sind.

```
[student@workstation ~]$ lab services-review start
```

- Starten Sie auf **serverb** den Service **psacct**.

- Melden Sie sich mit dem Befehl **ssh** bei **serverb** als **student** an.

```
[student@workstation ~]$ ssh student@serverb
...output omitted...
[student@serverb ~]$
```

- Führen Sie den Befehl **systemctl** aus, um den Status des Service **psacct** abzufragen. Beachten Sie, dass **psacct** beim Systemstart gestoppt und deaktiviert wird.

```
[student@serverb ~]$ systemctl status psacct
● psacct.service - Kernel process accounting
  Loaded: loaded (/usr/lib/systemd/system/psacct.service; disabled; vendor
  preset: disabled)
    Active: inactive (dead)
```

- Starten Sie den **psacct**-Service.

```
[student@serverb ~]$ sudo systemctl start psacct
[sudo] password for student: student
[student@serverb ~]$
```

- 1.4. Stellen Sie sicher, dass der Service **psacct** ausgeführt wird.

```
[student@serverb ~]$ systemctl is-active psacct  
active
```

2. Konfigurieren Sie den Service **psacct** so, dass er bei jedem Systemstart gestartet wird.

- 2.1. Aktivieren Sie den Service **psacct** so, dass er bei jedem Systemstart gestartet wird.

```
[student@serverb ~]$ sudo systemctl enable psacct  
Created symlink /etc/systemd/system/multi-user.target.wants/psacct.service → /usr/  
lib/systemd/system/psacct.service.
```

- 2.2. Überprüfen Sie, ob der Service **psacct** beim Systemstart aktiviert wird.

```
[student@serverb ~]$ systemctl is-enabled psacct  
enabled
```

3. Beenden Sie den Service **rsyslog**.

- 3.1. Führen Sie den Befehl **systemctl** aus, um den Status des Service **rsyslog** abzufragen. Beachten Sie, dass der Service **rsyslog** läuft und aktiviert ist, um bei einem Booten des Systems gestartet zu werden.

```
[student@serverb ~]$ systemctl status rsyslog  
● rsyslog.service - System Logging Service  
  Loaded: loaded (/usr/lib/systemd/system/rsyslog.service; enabled; vendor  
  preset: enabled)  
  Active: active (running) since Fri 2019-02-08 10:16:00 IST; 2h 34min ago  
    ...output omitted...
```

Drücken Sie **q**, um den Befehl zu beenden.

- 3.2. Beenden Sie den Service **rsyslog**.

```
[student@serverb ~]$ sudo systemctl stop rsyslog  
[sudo] password for student: student  
[student@serverb ~]$
```

- 3.3. Überprüfen Sie, ob der **rsyslog**-Service ordnungsgemäß beendet wurde.

```
[student@serverb ~]$ systemctl is-active rsyslog  
inactive
```

4. Konfigurieren Sie den Service **rsyslog** so, dass dieser nicht beim Booten des Systems ausgeführt wird.

- 4.1. Konfigurieren Sie den Service **rsyslog** so, dass dieser nicht beim Booten des Systems ausgeführt wird.

```
[student@serverb ~]$ sudo systemctl disable rsyslog
Removed /etc/systemd/system/syslog.service.
Removed /etc/systemd/system/multi-user.target.wants/rsyslog.service.
```

4.2. Überprüfen Sie, ob der Service **rsyslog** beim Booten des Systems deaktiviert ist.

```
[student@serverb ~]$ systemctl is-enabled rsyslog
disabled
```

5. Starten Sie **serverb** neu, bevor Sie die praktische Übung bewerten.

```
[student@serverb ~]$ sudo systemctl reboot
Connection to serverb closed by remote host.
Connection to serverb closed.
[student@workstation ~]$
```

## Bewertung

Führen Sie auf **workstation** das Skript **lab services-review grade** aus, um den Erfolg dieser praktischen Übung zu bestätigen.

```
[student@workstation ~]$ lab services-review grade
```

## Beenden

Führen Sie auf **workstation** das Skript **lab services-review finish** aus, um die praktische Übung zu beenden.

```
[student@workstation ~]$ lab services-review finish
```

Hiermit wird die praktische Übung abgeschlossen.

# Zusammenfassung

---

In diesem Kapitel erhielten Sie folgende Informationen:

- Mit **systemd** können Systemressourcen, Server-Daemons und andere Prozesse sowohl beim Systemstart als auch auf einem laufenden System aktiviert werden.
- Verwenden Sie **systemctl**, um Services zu starten, zu stoppen, zu aktivieren und zu deaktivieren.
- Verwenden Sie den Befehl **systemctl status**, um den Status der durch **systemd** gestarteten System-Daemons und Netzwerkservices zu bestimmen.
- Der Befehl **systemctl list-dependencies** listet alle Service-Units auf, von denen eine bestimmte Service-Unit abhängig ist.
- **systemd** kann eine Service-Unit maskieren, sodass sie nicht ausgeführt wird, um Abhängigkeiten zu erfüllen.



## Kapitel 10

# Konfigurieren und Sichern von SSH

### Ziel

Konfigurieren von sicherem Befehlszeilenservice auf Remote-Systemen mit OpenSSH

### Ziele

- Anmelden bei einem Remote-System und Ausführen von Befehlen mit **ssh**
- Konfigurieren der schlüsselbasierten Authentifizierung für ein Benutzerkonto zur sicheren und passwortlosen Anmeldung bei Remote-Systemen
- Einschränken der direkten Anmeldung als root und Deaktivieren der passwortbasierten Authentifizierung für den OpenSSH-Service

### Abschnitte

- Zugreifen auf die Remote-Befehlszeile mit SSH (und angeleitete Übung)
- Konfigurieren der auf SSH-Schlüssel basierten Authentifizierung (und angeleitete Übung)
- Anpassen der OpenSSH-Servicekonfiguration (und angeleitete Übung)

### Praktische Übung

Konfigurieren und Sichern von SSH

# Zugreifen auf die Remote-Befehlszeile mit SSH

---

## Ziele

Nach Abschluss dieses Abschnittes sollten Sie in der Lage sein, sich bei einem Remote-System anzumelden und Befehle mit **ssh** auszuführen.

## Was ist OpenSSH?

OpenSSH implementiert das Secure Shell- oder SSH-Protokoll in Red Hat Enterprise Linux-Systeme. Das SSH-Protokoll ermöglicht es Systemen, verschlüsselt und sicher über ein unsicheres Netzwerk zu kommunizieren.

Sie können mit dem Befehl **ssh** eine sichere Verbindung zu einem Remote-System herstellen, sich als bestimmter Benutzer authentifizieren und als dieser Benutzer eine interaktive Shell-Sitzung auf dem Remote-System abrufen. Sie können mit dem Befehl **ssh** auch einen einzelnen Befehl auf dem Remote-System ausführen, ohne eine interaktive Shell auszuführen.

## Secure Shell-Beispiele

Mit dem folgenden **ssh**-Befehl würden Sie sich auf dem Remote-Server **remotehost** mit demselben Benutzernamen wie der aktuelle lokale Benutzer anmelden. In diesem Beispiel werden Sie vom Remote-System aufgefordert, sich mit dem Passwort dieses Benutzers zu authentifizieren.

```
[user01@host ~]$ ssh remotehost
user01@remotehost's password: redhat
...output omitted...
[user01@remotehost ~]$
```

Mit dem Befehl **exit** können Sie sich vom Remote-System abmelden.

```
[user01@remotehost ~]$ exit
logout
Connection to remotehost closed.
[user01@host ~]$
```

Mit dem nächsten **ssh**-Befehl würden Sie sich auf dem Remote-Server **remotehost** mit dem Benutzernamen **user02** anmelden. Sie werden wieder vom Remote-System aufgefordert, sich mit dem Passwort des Benutzers zu authentifizieren.

```
[user01@host ~]$ ssh user02@remotehost
user02@remotehost's password: shadowman
...output omitted...
[user02@remotehost ~]$
```

Dieser **ssh**-Befehl würde den Befehl **hostname** auf dem Remote-System **remotehost** als Benutzer **user02** ohne Zugriff auf die interaktive Remote-Shell ausführen.

```
[user01@host ~]$ ssh user02@remotehost hostname
user02@remotehost's password: shadowman
remotehost.lab.example.com
[user01@host ~]$
```

Beachten Sie, dass der vorherige Befehl die Ausgabe im Terminal des lokalen Systems angezeigt hat.

## Identifizieren von Remote-Benutzern

Mit dem Befehl **w** zeigen Sie eine Liste der aktuell bei dem Computer angemeldeten Benutzer an. Das ist besonders hilfreich, um anzusehen, welche Benutzer mit **ssh** von welchen Remote-Standorten angemeldet sind sowie welche Aktivitäten sie durchführen.

```
[user01@host ~]$ ssh user01@remotehost
user01@remotehost's password: redhat
[user01@remotehost ~]$ w
16:13:38 up 36 min, 1 user, load average: 0.00, 0.00, 0.00
USER    TTY      FROM          LOGIN@    IDLE   JCPU   PCPU WHAT
user02  pts/0    172.25.250.10  16:13     7:30   0.01s  0.01s -bash
user01  pts/1    172.25.250.10  16:24     3.00s  0.01s  0.00s w
[user02@remotehost ~]$
```

Die vorherige Ausgabe zeigt, dass sich der Benutzer **user02** heute um **16:13** beim System auf dem Pseudoterminal **0** vom Host mit der IP-Adresse **172.25.250.10** aus angemeldet hat und sieben Minuten und dreißig Sekunden an einer Shell-Eingabeaufforderung inaktiv war. Die vorherige Ausgabe zeigt auch, dass sich der Benutzer **user01** beim System auf dem Pseudoterminal **1** angemeldet hat und nach der Ausführung des Befehls **w** seit drei Sekunden inaktiv war.

## SSH-Hostschlüssel

SSH sichert die Kommunikation durch Verschlüsselung mit Public Keys (öffentliche Schlüssel). Wenn ein SSH-Client eine Verbindung zu einem SSH-Server herstellt, sendet der Server vor der Anmeldung eine Kopie seines Public Key an den Client. Damit werden die sichere Verschlüsselung des Kommunikationskanals und die Authentifizierung des Servers beim Client ermöglicht.

Wenn ein Benutzer mit dem Befehl **ssh** eine Verbindung zu einem SSH-Server herstellt, überprüft dieser Befehl, ob eine Kopie des Public Key für diesen Server in den lokalen Dateien mit den bekannten Hosts vorhanden ist. Der Systemadministrator kann ihn in **/etc/ssh/ssh\_known\_hosts** vorkonfiguriert haben oder der Benutzer kann in seinem Benutzerverzeichnis eine **~/.ssh/known\_hosts**-Datei haben, die den Schlüssel enthält.

Wenn der Client über eine Kopie des Schlüssels verfügt, vergleicht **ssh** den Schlüssel aus den Dateien mit den bekannten Hosts für diesen Server mit dem erhaltenen Schlüssel. Wenn die Schlüssel nicht übereinstimmen, geht der Client davon aus, dass der Netzwerkdatenverkehr zu dem Server abgehört wird oder der Server kompromittiert wurde, und versucht, vom Benutzer die Bestätigung zum Fortsetzen oder Abbrechen der Verbindung zu erhalten.



### Anmerkung

Legen Sie den Parameter **StrictHostKeyChecking** in der benutzerspezifischen Datei `~/.ssh/config` oder in der systemweiten Datei `/etc/ssh/ssh_config` auf **yes** fest, damit der Befehl **ssh** die SSH-Verbindung immer abbricht, wenn die Public Keys nicht übereinstimmen.

Wenn der Client über keine Kopie des Public Key in seinen Dateien mit den bekannten Hosts verfügt, fragt der Befehl **ssh** Sie, ob Sie sich trotzdem anmelden möchten. In diesem Fall wird eine Kopie des Public Key in Ihrer `~/.ssh/known_hosts`-Datei gespeichert, damit die Identität des Servers in Zukunft automatisch bestätigt werden kann.

```
[user01@host ~]$ ssh newhost
The authenticity of host 'remotehost (172.25.250.12)' can't be established.
ECDSA key fingerprint is SHA256:qaS0PToLrq1C02XGk1A0iY7CaP7aPKimerDoaUkv720.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'newhost,172.25.250.12' (ECDSA) to the list of known
hosts.
user01@newhost's password: redhat
...output omitted...
[user01@newhost ~]$
```

## SSH-Schlüsselverwaltung für bekannte Hosts

Wenn der Public Key eines Servers geändert wird, weil der Schlüssel aufgrund eines Festplattenfehlers verloren gegangen ist oder aus legitimen Gründen ersetzt wurde, müssen Sie die Dateien mit den bekannten Hosts bearbeiten, um sicherzustellen, dass der Eintrag für den alten Public Key durch einen Eintrag mit dem neuen Public Key ersetzt wird, damit Sie sich fehlerfrei anmelden können.

Public Keys werden in der Datei `/etc/ssh/ssh_known_hosts` und in der Datei `~/.ssh/known_hosts` jedes Benutzers auf dem SSH-Client gespeichert. Jeder Schlüssel befindet sich in einer Zeile. Das erste Feld ist eine Liste von Hostnamen und IP-Adressen, die diesen Public Key gemeinsam nutzen. Das zweite Feld ist der Verschlüsselungsalgorithmus für den Schlüssel. Das letzte Feld ist der Schlüssel selbst.

```
[user01@host ~]$ cat ~/.ssh/known_hosts
remotehost,172.25.250.11 ecdsa-sha2-nistp256
AAAAE2VjZHNhLXNoYTItbm1zdHAyNTYAAAIBm1zdHAyNTYAAABBB0sEi0e+F1aNT6jul8Ag5Nj
+RViZ10yE2w6iYUr+1fPt0IF0Ea0gFZ1LXM37VFTxdgFxHS3D5WhnIfb+68zf8+w=
```

Jeder Remote-SSH-Server, mit dem Sie eine Verbindung herstellen, speichert seinen Public Key im Verzeichnis `/etc/ssh` in Dateien mit der Erweiterung `.pub`.

```
[user01@remotehost ~]$ ls /etc/ssh/*key.pub
/etc/ssh/ssh_host_ecdsa_key.pub  /etc/ssh/ssh_host_ed25519_key.pub  /etc/ssh/
ssh_host_rsa_key.pub
```



### Anmerkung

Einträge, die den `ssh_host_*key.pub`-Dateien eines Servers entsprechen, sollten Sie Ihrer `~/.ssh/known_hosts`-Datei oder der systemweiten `/etc/ssh/ssh_known_hosts`-Datei hinzuzufügen.



### Literaturhinweise

Manpages `ssh(1)`, `w(1)` und `hostname(1)`

Weitere Informationen finden Sie im Kapitel *Using secure communications between two systems with OpenSSH* im Handbuch *Red Hat Enterprise Linux 8 Securing networks* unter

[https://access.redhat.com/documentation/en-us/red\\_hat\\_enterprise\\_linux/8/html-single/securing\\_networks/index#using-secure-communications-between-two-systems-with-openssh\\_securing-networks](https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux/8/html-single/securing_networks/index#using-secure-communications-between-two-systems-with-openssh_securing-networks)

## ► Angeleitete Übung

# Zugreifen auf die Remote-Befehlszeile

In dieser Übung melden Sie sich als verschiedene Benutzer auf einem Remote-System an und führen dort Befehle aus.

## Ergebnisse

Es werden folgende Fähigkeiten vermittelt:

- Anmelden bei einem Remote-System
- Ausführen von Befehlen mit der sicheren OpenSSH-Shell

## Bevor Sie Beginnen

Melden Sie sich als Benutzer **student** mit dem Passwort **student** bei **workstation** an.

Führen Sie auf **workstation** das Skript **lab ssh-access start** aus, um diese Übung zu beginnen. Das Skript stellt sicher, dass die Umgebung richtig eingerichtet ist.

```
[student@workstation ~]$ lab ssh-access start
```

### ► 1. Öffnen Sie auf **workstation** als **student** eine SSH-Sitzung zu **servera**.

```
[student@workstation ~]$ ssh student@servera
...output omitted...
[student@servera ~]$
```

### ► 2. Öffnen Sie als **student** eine SSH-Sitzung zu **serverb**. Akzeptieren Sie den Hostschlüssel. Geben Sie **student** als Passwort ein, wenn Sie zur Eingabe eines Passworts für den Benutzer **student** auf **serverb** aufgefordert werden.

```
[student@servera ~]$ ssh student@serverb
The authenticity of host 'serverb (172.25.250.11)' can't be established.
ECDSA key fingerprint is SHA256:ERTdjoo0IrIwVSZQnqD5or+JbXfidg0udb3DXBuHWzA.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'serverb,172.25.250.11' (ECDSA) to the list of known
hosts.
student@serverb's password: student
...output omitted...
[student@serverb ~]$
```

Der Hostschlüssel wird in der Datei **/home/student/.ssh/known\_hosts** auf **servera** erfasst, um **serverb** zu identifizieren, weil der Benutzer **student** die SSH-Verbindung von **servera** initiiert hat. Wenn die Datei **/home/student/.ssh/known\_hosts** nicht bereits vorhanden ist, wird sie als neue Datei zusammen mit einem neuen Eintrag darin erstellt. Der Befehl **ssh** kann nicht ordnungsgemäß ausgeführt werden, wenn der Schlüssel des Remote-Hosts nicht mit dem gespeicherten Schlüssel übereinstimmt.

- 3. Führen Sie den Befehl **w** aus, um die Benutzer anzuzeigen, die aktuell bei **serverb** angemeldet sind.

```
[student@serverb ~]$ w
18:49:29 up 2:55, 1 user, load average: 0.00, 0.00, 0.00
USER      TTY      FROM          LOGIN@    IDLE     JCPU    PCPU WHAT
student   pts/0    172.25.250.10    18:33    0.00s  0.01s  0.00s w
```

Die vorherige Ausgabe zeigt an, dass sich der Benutzer **student** beim System von einem Host mit der IP-Adresse **172.25.250.10** aus angemeldet hat. Dabei handelt es sich um **servera** im Kursraumnetzwerk.



### Anmerkung

Die IP-Adresse eines Systems identifiziert das System in einem Netzwerk. Im weiteren Verlauf dieses Kapitels werden IP-Adressen ausführlicher behandelt.

- 4. Beenden Sie die Shell des Benutzers **student** auf **serverb**.

```
[student@serverb ~]$ exit
logout
Connection to serverb closed.
[student@servera ~]$
```

- 5. Öffnen Sie als **root** eine SSH-Sitzung zu **serverb**. Verwenden Sie **redhat** als Passwort des **root**-Benutzers.

```
[student@servera ~]$ ssh root@serverb
root@serverb's password: redhat
...output omitted...
[root@serverb ~]#
```

Beachten Sie, dass der vorherige Befehl **ssh** Sie nicht aufgefordert hat, den Hostschlüssel zu akzeptieren, da er unter den bekannten Hosts gefunden wurde. Sollte sich die Identität von **serverb** ändern, fordert OpenSSH Sie auf, den neuen Hostschlüssel zu bestätigen und zu akzeptieren.

- 6. Führen Sie den Befehl **w** aus, um die Benutzer anzuzeigen, die aktuell bei **serverb** angemeldet sind.

```
[root@serverb ~]# w
19:10:28 up 3:16, 1 user, load average: 0.00, 0.00, 0.00
USER      TTY      FROM          LOGIN@    IDLE     JCPU    PCPU WHAT
root      pts/0    172.25.250.10    19:09    1.00s  0.01s  0.00s w
```

Die vorhergehende Ausgabe zeigt an, dass sich der Benutzer **root** beim System von einem Host mit der IP-Adresse **172.25.250.10** aus angemeldet hat. Dabei handelt es sich um **servera** im Kursraumnetzwerk.

- 7. Beenden Sie die Shell des Benutzers **root** auf **serverb**.

```
[root@serverb ~]# exit
logout
Connection to serverb closed.
[student@servera ~]$
```

- 8. Entfernen Sie die Datei **/home/student/.ssh/known\_hosts** von **servera**. Dies bewirkt, dass **ssh** die erfassten Identitäten der Remote-Systeme verliert.

```
[student@servera ~]$ rm /home/student/.ssh/known_hosts
```

Hostschlüssel können sich aus berechtigten Gründen ändern: Möglicherweise wurde der Remote-Rechner aufgrund eines Hardwarefehlers ausgetauscht oder der Remote-Rechner wurde neu installiert. Im Regelfall sollte nur der Schlüsseleintrag für diesen bestimmten Host in der Datei **known\_hosts** gelöscht werden. Da diese **known\_hosts**-Datei nur einen Eintrag hat, können Sie die gesamte Datei entfernen.

- 9. Öffnen Sie als **student** eine SSH-Sitzung zu **serverb**. Akzeptieren Sie auf Anfrage den Hostschlüssel. Geben Sie **student** als Passwort ein, wenn Sie zur Eingabe eines Passworts für den Benutzer **student** auf **serverb** aufgefordert werden.

```
[student@servera ~]$ ssh student@serverb
The authenticity of host 'serverb (172.25.250.11)' can't be established.
ECDSA key fingerprint is SHA256:ERTdjoo0IrIwVSZQnqD5or+JbXfidg0udb3DXBuHWZA.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added 'serverb,172.25.250.11' (ECDSA) to the list of known
hosts.
student@serverb's password: student
...output omitted...
[student@serverb ~]$
```

Beachten Sie, dass der Befehl **ssh** Sie dazu aufgefordert hat, den Hostschlüssel zu akzeptieren oder abzulehnen, da für den Remote-Host kein Schlüssel gefunden wurde.

- 10. Beenden Sie die Shell des Benutzers **student** auf **serverb** und vergewissern Sie sich, dass eine neue Instanz von **known\_hosts** auf **servera** vorhanden ist.

```
[student@serverb ~]$ exit
logout
Connection to serverb closed.
[student@servera ~]$ ls -l /home/student/.ssh/known_hosts
-rw-r--r--. 1 student student 183 Feb 1 20:26 /home/student/.ssh/known_hosts
```

- 11. Überprüfen Sie, ob die neue Instanz der Datei **known\_hosts** den Hostschlüssel von **serverb** hat.

```
[student@servera ~]$ cat /home/student/.ssh/known_hosts
serverb,172.25.250.11 ecdsa-sha2-nistp256
AAAAE2VjZHNhLXNoYTItbmlzdHAyNTYAAAIbm1zdHAyNTYAAABBI9LEYEhwmu1rNqnbBPukH2Ba0/
QBAu9wbS4m03B3MIhhXWKFFNa/U1NjY8NDpEM+hkJe/GmnkcEYMLbCfd9nMA=
```

Die tatsächliche Ausgabe kann davon abweichen.

- 12. Führen Sie **hostname** auf **serverb** remote aus, ohne auf die interaktive Shell zuzugreifen.

```
[student@servera ~]$ ssh student@serverb hostname  
student@serverb's password: student  
serverb.lab.example.com
```

Der vorherige Befehl hat den vollständigen Hostnamen des Remote-Systems auf **serverb** angezeigt.

- 13. Beenden Sie die Shell des Benutzers **student** auf **servera**.

```
[student@servera ~]$ exit  
logout  
Connection to servera closed.
```

## Beenden

Führen Sie auf **workstation** das Skript **lab ssh-access finish** aus, um diese Übung zu beenden.

```
[student@workstation ~]$ lab ssh-access finish
```

Hiermit ist die angeleitete Übung beendet.

# Konfigurieren der schlüsselbasierten SSH-Authentifizierung

---

## Ziele

Nach Abschluss dieses Abschnitts, sollten Sie in der Lage sein, ein Benutzerkonto für die schlüsselbasierte Authentifizierung zur sicheren und passwortlosen Anmeldung bei Remote-Systemen zu konfigurieren.

## Schlüsselbasierte SSH-Authentifizierung

Sie können einen SSH-Server so konfigurieren, dass Sie sich ohne Passwort über die schlüsselbasierte SSH-Authentifizierung authentifizieren können. Dies basiert auf einem Private-Public-Schlüsselschema.

Dazu generieren Sie ein übereinstimmendes Paar kryptografischer Schlüsseldateien. Eine ist ein Private Key (privater Schlüssel), die andere ein übereinstimmender Public Key (öffentlicher Schlüssel). Die Datei mit dem Private Key dient dabei als Authentifizierungsanmeldedaten und muss, wie ein Passwort, geschützt und geheim gehalten werden. Der Public Key wird auf die Systeme kopiert, mit denen der Benutzer eine Verbindung herstellen möchte, und dient der Verifizierung des Private Key. Der Public Key muss nicht geheim gehalten werden.

Sie legen eine Kopie des Public Key in Ihrem Benutzerkonto auf dem Server ab. Wenn Sie versuchen, sich anzumelden, kann der SSH-Server mit dem Public Key eine Abfrage ausgeben, die nur mit dem Private Key richtig beantwortet werden kann. Infolgedessen kann Ihr **ssh**-Client Ihre Anmeldung beim Server automatisch mit Ihrer eindeutigen Kopie des Private Key authentifizieren. Dank dieses Verfahrens können Sie sicher auf andere Systeme zuzugreifen, ohne jedes Mal ein Passwort interaktiv eingeben zu müssen.

## Generieren von SSH-Schlüsseln

Um einen Private Key und einen übereinstimmenden Public Key für die Authentifizierung zu erstellen, verwenden Sie den Befehl **ssh-keygen**. Standardmäßig werden Ihr Private und Ihr Public Key in Ihren `~/.ssh/id_rsa`-Dateien bzw. Ihren `~/.ssh/id_rsa.pub`-Dateien gespeichert.

```
[user@host ~]$ ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/home/user/.ssh/id_rsa): Enter
Created directory '/home/user/.ssh'.
Enter passphrase (empty for no passphrase): Enter
Enter same passphrase again: Enter
Your identification has been saved in /home/user/.ssh/id_rsa.
Your public key has been saved in /home/user/.ssh/id_rsa.pub.
The key fingerprint is:
SHA256:vxutUNPIO3QDCyvkYm1oIx35hmMrHpPKWFdIYu3HV+w user@host.lab.example.com
The key's randomart image is:
+---[RSA 2048]---+
|           |
|   .   .   |
|   o   o   |
```

```
| . = o   o . |
| o + = S E . |
| ..o o + * + |
| .% 0 . + B . |
|= *o0 . . + * |
|++ . . +. |
+---[SHA256]---
```

Falls Sie keine Passphrase angeben, wenn Sie von **ssh-keygen** dazu aufgefordert werden, ist der generierte Private Key nicht sicher. In diesem Fall kann jeder Benutzer mit Ihrer Private-Key-Datei diese zur Authentifizierung verwenden. Wenn Sie eine Passphrase festlegen, müssen Sie diese Passphrase eingeben, wenn Sie den Private Key zur Authentifizierung verwenden. (Daher würden Sie zur Authentifizierung die Passphrase des Private Key anstelle Ihres Passworts auf dem Remote-Host verwenden.)

Sie können das Hilfsprogramm **ssh-agent** ausführen, das Ihre Passphrase für den Private Key vorübergehend zwischenspeichern kann, um eine echte passwortlose Authentifizierung zu ermöglichen. Dies wird weiter unten in diesem Abschnitt behandelt.

Das folgende Beispiel für den Befehl **ssh-keygen** zeigt die Erstellung eines durch eine Passphrase geschützten Private Key und des Public Key.

```
[user@host ~]$ ssh-keygen -f .ssh/key-with-pass
Generating public/private rsa key pair.
Enter passphrase (empty for no passphrase):
Enter same passphrase again:
Your identification has been saved in .ssh/key-with-pass.
Your public key has been saved in .ssh/key-with-pass.pub.
The key fingerprint is:
SHA256:w3GGB7EyHURY4a0cNPKmhNKS7d1YsMVLvFZJ77VxAo user@host.lab.example.com
The key's randomart image is:
+---[RSA 2048]---+
|     . + = .o ...
|     = B XEo o. .
|     . o O X =.... .
|     = = = B = o. .
|= + * * S . .
|. + = o + . .
| + .
|
|
+---[SHA256]---
```

Die Option **-f** mit dem Befehl **ssh-keygen** legt die Dateien fest, in denen die Schlüssel gespeichert werden. Im vorherigen Beispiel werden die Private und Public Keys des Benutzers in einer der Dateien **/home/user/.ssh/key-with-pass** **/home/user/.ssh/key-with-pass.pub** gespeichert.

**Warnung**

Wenn Sie während der weiteren SSH-Schlüsselpaar-Generierung einen eindeutigen Dateinamen angeben, werden Sie aufgefordert, das Überschreiben der vorhandenen Dateien **id\_rsa** und **id\_rsa.pub** zuzulassen. Wenn Sie die vorhandenen Dateien **id\_rsa** und **id\_rsa.pub** überschreiben, dann müssen Sie den alten Public Key auf allen SSH-Servern, auf denen sich Ihr alter Public Key befindet, durch den neuen ersetzen.

Sobald die SSH-Schlüssel generiert wurden, werden sie standardmäßig im Verzeichnis **.ssh/** des Benutzerverzeichnisses gespeichert. Der Berechtigungsmodus muss für den Private Key 600 und für den Public Key 644 lauten.

## Freigeben des Public Key

Bevor die schlüsselbasierte Authentifizierung verwendet werden kann, muss der Public Key auf das Zielsystem kopiert werden. Der Befehl **ssh-copy-id** kopiert den Public Key des SSH-Schlüsselpaares auf das Zielsystem. Wenn Sie den Pfad zur Datei mit dem Public Key während der Ausführung von **ssh-copy-id** weglassen, wird die Standarddatei **/home/user/.ssh/id\_rsa.pub** verwendet.

```
[user@host ~]$ ssh-copy-id -i .ssh/key-with-pass.pub user@remotehost
/usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed: "/home/user/.ssh/
id_rsa.pub"
/usr/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to filter
out any that are already installed
/usr/bin/ssh-copy-id: INFO: 1 key(s) remain to be installed -- if you are prompted
now it is to install the new keys
user@remotehost's password: redhat
Number of key(s) added: 1

Now try logging into the machine, with: "ssh 'user@remotehost'"
and check to make sure that only the key(s) you wanted were added.
```

Nachdem der Public Key erfolgreich auf ein Remote-System übertragen wurde, können Sie sich bei der Anmeldung beim Remote-System über SSH mit dem entsprechenden Private Key beim Remote-System authentifizieren. Wenn Sie den Pfad zur Datei mit dem Private Key während der Ausführung des Befehls **ssh** weglassen, wird die Standarddatei **/home/user/.ssh/id\_rsa** verwendet.

```
[user@host ~]$ ssh -i .ssh/key-with-pass user@remotehost
Enter passphrase for key '.ssh/key-with-pass': redhatpass
...output omitted...
[user@remotehost ~]$ exit
logout
Connection to remotehost closed.
[user@host ~]$
```

## Verwenden von ssh-agent für die nicht interaktive Authentifizierung

Wenn Ihr Private Key für SSH mit einer Passphrase geschützt ist, müssen Sie normalerweise die Passphrase eingeben, um den Private Key für die Authentifizierung zu verwenden. Sie können jedoch mit dem Programm **ssh-agent** die Passphrase vorübergehend im Arbeitsspeicher zwischenspeichern. Wenn Sie sich dann mit SSH mit dem Private Key bei einem anderen System anmelden, stellt **ssh-agent** automatisch die Passphrase für Sie bereit. Dies ist praktisch und kann die Sicherheit verbessern, indem weniger Gelegenheit zum „Schulter-Surfen“ bei der Eingabe der Passphrase geboten wird.

Je nach Konfiguration Ihres lokalen Systems wird das Programm **ssh-agent** möglicherweise automatisch gestartet und für Sie konfiguriert, wenn Sie sich zum ersten Mal bei der grafischen GNOME-Desktopumgebung anmelden.

Wenn Sie sich an einer Textkonsole anmelden, sich mit **ssh** anmelden oder **sudo** oder **su** verwenden, müssen Sie wahrscheinlich **ssh-agent** manuell für diese Sitzung starten. Sie können dies mit dem folgenden Befehl ausführen:

```
[user@host ~]$ eval $(ssh-agent)  
Agent pid 10155  
[user@host ~]$
```



### Anmerkung

Wenn Sie **ssh-agent** ausführen, werden einige Shell-Befehle ausgegeben. Sie müssen diese Befehle ausführen, um Umgebungsvariablen festzulegen, die von Programmen wie **ssh-add** für die Kommunikation verwendet werden. Der Befehl **eval \$(ssh-agent)** startet **ssh-agent** und führt diese Befehle aus, um diese Umgebungsvariablen für diese Shell-Sitzung automatisch festzulegen. Der Befehl zeigt auch die PID des **ssh-agent**-Prozesses an.

Wenn **ssh-agent** ausgeführt wird, müssen Sie ihm die Passphrase für Ihren Private Key oder Ihre Private Keys mitteilen. Sie können dies mit dem Befehl **ssh-add** ausführen:

Die folgenden **ssh-add**-Befehle fügen die Private Keys aus **/home/user/.ssh/id\_rsa** (Standard) und aus **/home/user/.ssh/key-with-pass**-Dateien hinzu.

```
[user@host ~]$ ssh-add  
Identity added: /home/user/.ssh/id_rsa (user@host.lab.example.com)  
[user@host ~]$ ssh-add .ssh/key-with-pass  
Enter passphrase for .ssh/key-with-pass: redhatpass  
Identity added: .ssh/key-with-pass (user@host.lab.example.com)
```

Nach erfolgreichem Hinzufügen der Private Keys zum Prozess **ssh-agent** können Sie eine SSH-Verbindung mit dem Befehl **ssh** aufrufen. Wenn Sie eine andere Private-Key-Datei als die Standarddatei **/home/user/.ssh/id\_rsa** verwenden, dann müssen Sie die Option **-i** mit dem Befehl **ssh** verwenden, um den Pfad zur Private-Key-Datei anzugeben.

Im folgenden Beispiel für den Befehl **ssh** wird die Standarddatei des Private Key zur Authentifizierung bei einem SSH-Server verwendet.

```
[user@host ~]$ ssh user@remotehost
Last login: Fri Apr  5 10:53:50 2019 from host.example.com
[user@remotehost ~]$
```

Im folgenden Beispiel für den Befehl **ssh** wird die Datei **/home/user/.ssh/key-with-pass** (nicht Standard) des Private Key zur Authentifizierung bei einem SSH-Server verwendet. Der Private Key im folgenden Beispiel wurde bereits entschlüsselt und seinem übergeordneten **ssh-agent**-Prozess hinzugefügt, daher fordert Sie der Befehl **ssh** nicht dazu auf, den Private Key durch interaktive Eingabe der Passphrase zu entschlüsseln.

```
[user@host ~]$ ssh -i .ssh/key-with-pass user@remotehost
Last login: Mon Apr  8 09:44:20 2019 from host.example.com
[user@remotehost ~]$
```

Wenn Sie sich von der Sitzung abmelden, die **ssh-agent** gestartet hat, wird der Prozess beendet und die Passphrasen für Ihre Private Keys werden aus dem Speicher gelöscht.



#### Literaturhinweise

Manpages **ssh-keygen(1)**, **ssh-copy-id(1)**, **ssh-agent(1)**, **ssh-add(1)**

## ► Angeleitete Übung

# Konfigurieren der schlüsselbasierten SSH-Authentifizierung

In dieser Übung konfigurieren Sie einen Benutzer für die Verwendung der schlüsselbasierten Authentifizierung für SSH.

## Ergebnisse

Es werden folgende Fähigkeiten vermittelt:

- Generieren eines SSH-Schlüsselpaares ohne Passphrasenschutz
- Generieren eines SSH-Schlüsselpaares mit Passphrasenschutz
- Authentifizieren mit beiden SSH-Schlüsseln, mit und ohne Passphrasenschutz

## Bevor Sie Beginnen

Melden Sie sich als Benutzer **student** mit dem Passwort **student** bei **workstation** an.

Führen Sie auf **workstation** das Skript **lab ssh-configure start** aus, um diese Übung zu beginnen. Dieses Skript erstellt die erforderlichen Benutzerkonten.

```
[student@workstation ~]$ lab ssh-configure start
```

- 1. Öffnen Sie auf **workstation** als **student** eine SSH-Sitzung zu **serverb**.

```
[student@workstation ~]$ ssh student@serverb  
...output omitted...  
[student@serverb ~]$
```

- 2. Wechseln Sie auf **serverb** mit dem Befehl **su** zum Benutzer **operator1**. Verwenden Sie **redhat** als Passwort für **operator1**.

```
[student@serverb ~]$ su - operator1  
Password: redhat  
[operator1@serverb ~]$
```

- 3. Generieren Sie mit dem Befehl **ssh-keygen** SSH-Schlüssel. Geben Sie keine Passphrase ein.

```
[operator1@serverb ~]$ ssh-keygen  
Generating public/private rsa key pair.  
Enter file in which to save the key (/home/operator1/.ssh/id_rsa): Enter  
Created directory '/home/operator1/.ssh'.  
Enter passphrase (empty for no passphrase): Enter  
Enter same passphrase again: Enter
```

**Kapitel 10 | Konfigurieren und Sichern von SSH**

```
Your identification has been saved in /home/operator1/.ssh/id_rsa.  
Your public key has been saved in /home/operator1/.ssh/id_rsa.pub.  
The key fingerprint is:  
SHA256:JainiQdnRosC+xXh0qsJQQLzBNULdb+jJbyrcZQBERI  
operator1@serverb.lab.example.com  
The key's randomart image is:  
+---[RSA 2048]---+  
|E+*+000 . . . . |  
|.= o o o . . . |  
|o.. = . . o . . |  
|+. + * . o . . |  
|+ = X . S + . . |  
| + @ + = . . . . |  
|. + = o . . . . |  
|o . . . . . . . |  
+---[SHA256]---+
```

- 4. Senden Sie mit dem Befehl **ssh-copy-id** den Public Key des SSH-Schlüsselpaares an **operator1** auf **servera**. Verwenden Sie auf **servera redhat** als Passwort für **operator1**.

```
[operator1@serverb ~]$ ssh-copy-id operator1@servera  
/usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed: "/home/  
operator1/.ssh/id_rsa.pub"  
The authenticity of host 'servera (172.25.250.10)' can't be established.  
ECDSA key fingerprint is SHA256:ERTdjoo0IrIwVSZQnqD5or+JbXfidg0udb3DXBuHWZA.  
Are you sure you want to continue connecting (yes/no)? yes  
/usr/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to filter  
out any that are already installed  
/usr/bin/ssh-copy-id: INFO: 1 key(s) remain to be installed -- if you are prompted  
now it is to install the new keys  
operator1@servera's password: redhat  
Number of key(s) added: 1  
  
Now try logging into the machine, with: "ssh 'operator1@servera'"  
and check to make sure that only the key(s) you wanted were added.
```

- 5. Führen Sie den Befehl **hostname** auf **servera** remote über SSH ohne Zugriff auf die interaktive Remote-Shell aus.

```
[operator1@serverb ~]$ ssh operator1@servera hostname  
servera.lab.example.com
```

Beachten Sie, dass der vorherige **ssh**-Befehl Sie nicht zur Eingabe eines Passworts aufgefordert hat, da der Private Key ohne Passphrase für den exportierten Public Key verwendet wurde, um sich als **operator1** auf **servera** zu authentifizieren. Dieser Ansatz ist nicht sicher, da sich jeder, der Zugriff auf die Private-Key-Datei hat, bei **servera** als **operator1** anmelden kann. Die sichere Alternative besteht darin, den Private Key mit einer Passphrase zu schützen. Dies wird im nächsten Schritt ausgeführt.

- 6. Generieren Sie mit dem Befehl **ssh-keygen** einen weiteren Satz von SSH-Schlüsseln mit Passphrasenschutz. Speichern Sie den Schlüssel als **/home/operator1/.ssh/key2**. Verwenden Sie **redhatpass** als Passphrase des Private Key.



### Warnung

Wenn Sie die Datei nicht angeben, in der der Schlüssel gespeichert wird, wird die Standarddatei (**/home/user/.ssh/id\_rsa**) verwendet. Sie haben den Standarddateinamen bereits beim Generieren von SSH-Schlüsseln im vorherigen Schritt verwendet. Daher müssen Sie unbedingt eine Nicht-Standarddatei angeben, andernfalls werden die vorhandenen SSH-Schlüssel überschrieben.

```
[operator1@serverb ~]$ ssh-keygen -f .ssh/key2
Generating public/private rsa key pair.
Enter passphrase (empty for no passphrase): redhatpass
Enter same passphrase again: redhatpass
Your identification has been saved in .ssh/key2.
Your public key has been saved in .ssh/key2.pub.
The key fingerprint is:
SHA256:0CtCjfPm5QrbPBgqbEIWCCw5AI4oSlMEbgLrBQ1HWKI
operator1@serverb.lab.example.com
The key's randomart image is:
+---[RSA 2048]---+
|O=X*          |
|OB=.          |
|E*o.          |
|Booo .         |
|.= . o S      |
|+.o o          |
|+.oo+ o        |
|+o.0.+         |
|+. . =o.       |
+---[SHA256]---
```

- 7. Senden Sie mit dem Befehl **ssh-copy-id** den Public Key des durch Passphrase geschützten Schlüsselpaares an **operator1** auf **servera**.

```
[operator1@serverb ~]$ ssh-copy-id -i .ssh/key2.pub operator1@servera
/usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed: ".ssh/key2.pub"
/usr/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to filter
out any that are already installed
/usr/bin/ssh-copy-id: INFO: 1 key(s) remain to be installed -- if you are prompted
now it is to install the new keys

Number of key(s) added: 1

Now try logging into the machine, with: "ssh 'operator1@servera'"
and check to make sure that only the key(s) you wanted were added.
```

Beachten Sie, dass der vorherige **ssh-copy-id**-Befehl Sie nicht zur Eingabe eines Passworts aufgefordert hat, da der Public Key des Private Key ohne Passphrase verwendet wurde, den Sie im vorherigen Schritt auf **servera** exportiert haben.

- 8. Führen Sie den Befehl **hostname** auf **servera** remote über SSH ohne Zugriff auf die interaktive Remote-Shell aus. Verwenden Sie **/home/operator1/.ssh/key2** als Identitätsdatei. Geben Sie **redhatpass** als Passphrase an, die Sie im vorherigen Schritt für den Private Key festgelegt haben.

```
[operator1@serverb ~]$ ssh -i .ssh/key2 operator1@servera hostname  
Enter passphrase for key '.ssh/key2': redhatpass  
servera.lab.example.com
```

Beachten Sie, dass der vorherige **ssh**-Befehl Sie zur Eingabe der Passphrase aufgefordert hat, mit der Sie den Private Key des SSH-Schlüsselpaares geschützt haben. Diese Passphrase schützt den Private Key. Wenn ein Angreifer Zugriff auf den Private Key erhält, kann er ihn nicht für den Zugriff auf andere Systeme verwenden, da der Private Key selbst mit einer Passphrase geschützt ist. Der Befehl **ssh** verwendet eine andere Passphrase als die für **operator1** auf **servera**, daher müssen Benutzer beide kennen.

Sie können **ssh-agent** verwenden, wie im folgenden Schritt gezeigt, um das interaktive Eingeben der Passphrase während der Anmeldung mit SSH zu vermeiden. Die Verwendung von **ssh-agent** ist in Situationen, in denen sich die Administratoren regelmäßig bei Remote-Systemen anmelden, komfortabler und sicherer.

- 9. Führen Sie in der Bash-Shell **ssh-agent** aus und fügen Sie der Shell-Sitzung den durch die Passphrase geschützten Private Key (**/home/operator1/.ssh/key2**) des SSH-Schlüsselpaares hinzu.

```
[operator1@serverb ~]$ eval $(ssh-agent)  
Agent pid 21032  
[operator1@serverb ~]$ ssh-add .ssh/key2  
Enter passphrase for .ssh/key2: redhatpass  
Identity added: .ssh/key2 (operator1@serverb.lab.example.com)
```

Der vorherige **eval**-Befehl hat **ssh-agent** gestartet und diese Shell-Sitzung so konfiguriert, dass er verwendet wird. Sie haben dann mit **ssh-add** den entsperrten Private Key **ssh-agent** zur Verfügung gestellt.

- 10. Führen Sie den Befehl **hostname** auf **servera** remote ohne Zugriff auf die interaktive Remote-Shell aus. Verwenden Sie **/home/operator1/.ssh/key2** als Identitätsdatei.

```
[operator1@serverb ~]$ ssh -i .ssh/key2 operator1@servera hostname  
servera.lab.example.com
```

Beachten Sie, dass der vorherige **ssh**-Befehl Sie nicht aufgefordert hat, die Passphrase interaktiv einzugeben.

- 11. Öffnen Sie auf **workstation** ein weiteres Terminal und öffnen Sie als **student** eine SSH-Sitzung zu **serverb**.

```
[student@workstation ~]$ ssh student@serverb  
...output omitted...  
[student@serverb ~]$
```

- 12. Wechseln Sie auf **serverb** mit dem Befehl **su** zu **operator1** und rufen Sie eine SSH-Verbindung zu **servera** auf. Verwenden Sie **/home/operator1/.ssh/key2** als Identitätsdatei zur Authentifizierung mit den SSH-Schlüsseln.

- 12.1. Wechseln Sie mit dem Befehl **su** zum Benutzer **operator1**. Verwenden Sie **redhat** als Passwort für **operator1**.

```
[student@serverb ~]$ su - operator1  
Password: redhat  
[operator1@serverb ~]$
```

- 12.2. Öffnen Sie als **operator1** eine SSH-Sitzung zu **servera**.

```
[operator1@serverb ~]$ ssh -i .ssh/key2 operator1@servera  
Enter passphrase for key '.ssh/key2': redhatpass  
...output omitted...  
[operator1@servera ~]$
```

Beachten Sie, dass der vorherige Befehl **ssh** Sie aufgefordert hat, die Passphrase interaktiv einzugeben, da Sie die SSH-Verbindung nicht von der Shell aus aufgerufen haben, mit der Sie **ssh-agent** gestartet haben.

- 13. Beenden Sie alle Shells, die Sie im zweiten Terminal verwenden.

- 13.1. Melden Sie sich von **servera** ab.

```
[operator1@servera ~]$ exit  
logout  
Connection to servera closed.  
[operator1@serverb ~]$
```

- 13.2. Beenden Sie die Shells **operator1** und **student** auf **serverb**, um zur Shell des Benutzers **student** auf **workstation** zurückzukehren.

```
[operator1@serverb ~]$ exit  
logout  
[student@serverb ~]$ exit  
logout  
Connection to serverb closed.  
[student@workstation ~]$
```

- 13.3. Schließen Sie das zweite Terminal auf **workstation**.

```
[student@workstation ~]$ exit
```

- 14. Melden Sie sich von **serverb** auf dem ersten Terminal ab und beenden Sie diese Übung.

- 14.1. Beenden Sie auf dem ersten Terminal die Shell des Benutzers **operator1** auf **serverb**.

```
[operator1@serverb ~]$ exit  
logout  
[student@serverb ~]$
```

Der Befehl **exit** hat bewirkt, dass die Shell des Benutzers **operator1** sowie die Shell-Sitzung, in der **ssh-agent** aktiv war, beendet und zur Shell des Benutzers **student** auf **serverb** zurückgekehrt wurde.

- 14.2. Beenden Sie die Shell des Benutzers **student** auf **serverb**, um zur Shell des Benutzers **student** auf **workstation** zurückzukehren.

```
[student@serverb ~]$ exit  
logout  
Connection to serverb closed.  
[student@workstation ~]$
```

## Beenden

Führen Sie auf **workstation** das Skript **lab ssh-configure finish** aus, um diese Übung zu beenden.

```
[student@workstation ~]$ lab ssh-configure finish
```

Hiermit ist die angeleitete Übung beendet.

# Anpassen der OpenSSH-Servicekonfiguration

---

## Ziele

Nach Abschluss dieses Abschnitts sollten Sie in der Lage sein, direkte Anmeldungen als **root** einzuschränken und die passwortbasierte Authentifizierung für den OpenSSH-Service zu deaktivieren.

## Konfigurieren des OpenSSH-Servers

Der OpenSSH-Service wird von einem Daemon namens **sshd** bereitgestellt. Seine Hauptkonfigurationsdatei ist **/etc/ssh/sshd\_config**.

Die Standardkonfiguration des OpenSSH-Servers funktioniert gut. Möglicherweise möchten Sie jedoch einige Änderungen vornehmen, um die Sicherheit Ihres Systems zu erhöhen. Es gibt zwei häufige Änderungen, die Sie eventuell vornehmen möchten. Möglicherweise möchten Sie die direkte Remote-Anmeldung beim Benutzerkonto **root** sowie die passwortbasierte Authentifizierung verhindern (zugunsten der SSH-Authentifizierung mit Private Keys).

## Verhindern der Anmeldung des Superusers mit SSH

Es gilt als gute Praxis, die direkte Anmeldung beim Benutzerkonto **root** von Remote-Systemen aus nicht zuzulassen. Zu den Risiken der direkten Anmeldung als **root** zählen:

- Der Benutzername **root** ist standardmäßig auf jedem Linux-System vorhanden, weshalb ein potenzieller Angreifer lediglich das Passwort anstatt der Kombination aus gültigem Benutzernamen und Passwort erraten muss. Dies reduziert die Komplexität für einen Angreifer.
- Der **root**-Benutzer verfügt über uneingeschränkte Berechtigungen, sodass dessen Kompromittierung zu maximalem Schaden für das System führen kann.
- Aus der Überwachungsperspektive kann es schwierig sein festzustellen, als welcher berechtigte Benutzer sich als **root** angemeldet und Änderungen vorgenommen hat. Wenn Benutzer sich als reguläre Benutzer anmelden und zum Benutzerkonto **root** wechseln müssen, wird ein Protokollereignis generiert, mit dem Nachprüfbarkeit gewährleistet werden kann.

Der OpenSSH-Server verwendet die Konfigurationseinstellung **PermitRootLogin** in der Konfigurationsdatei **/etc/ssh/sshd\_config**, um Benutzern die Anmeldung als **root** zu erlauben oder zu verweigern.

```
PermitRootLogin yes
```

Wenn der Parameter **PermitRootLogin** auf **yes** festgelegt ist (Standardeinstellung), können sich Benutzer als **root** anmelden. Legen Sie den Wert auf **no** fest, um dies zu verhindern. Setzen Sie alternativ den Parameter **PermitRootLogin** auf **without-password**, um die passwortbasierte Authentifizierung für **root** zu verhindern, die auf dem Private Key basierte Authentifizierung jedoch zuzulassen.

Der SSH-Server (**sshd**) muss neu geladen werden, damit die Änderungen wirksam werden.

```
[root@host ~]# systemctl reload sshd
```

## Verhindern der passwortbasierten Authentifizierung für SSH

Das Zulassen nur auf Private Keys basierter Anmeldungen über die Remote-Befehlszeile hat verschiedene Vorteile:

- Angreifer können keine Angriffe zum Erraten von Passwörtern verwenden, um remote in bekannte Benutzerkonten des Systems einzudringen.
- Mit durch Passphrase geschützten Private Keys benötigt ein Angreifer sowohl die Passphrase als auch eine Kopie des Private Key. Bei Passwörtern benötigt ein Angreifer nur das Passwort.
- Durch die Verwendung von durch Passphrase geschützter Private Keys in Verbindung mit **ssh-agent** wird die Passphrase weniger häufig aufgedeckt, da sie weniger häufig eingegeben wird, und die Anmeldung ist für den Benutzer bequemer.

Der OpenSSH-Server verwendet den Parameter **PasswordAuthentication** in der Konfigurationsdatei **/etc/ssh/sshd\_config**, um zu steuern, ob Benutzer sich beim System per passwortbasierter Authentifizierung anmelden können.

```
PasswordAuthentication yes
```

Der Standwert **yes** für den Parameter **PasswordAuthentication** in der Konfigurationsdatei **/etc/ssh/sshd\_config** bewirkt, dass der SSH-Server die passwortbasierte Authentifizierung bei der Anmeldung zulässt. Der Wert **no** für **PasswordAuthentication** verhindert, dass Benutzer die passwortbasierte Authentifizierung verwenden.

Denken Sie daran, dass Sie den **sshd**-Service neu laden müssen, wenn Sie die Datei **/etc/ssh/sshd\_config** ändern, damit die Änderungen wirksam werden.



### Wichtig

Vergessen Sie auch Folgendes nicht: Wenn Sie die passwortbasierte Authentifizierung für **ssh** ausschalten, müssen Sie eine Möglichkeit haben, um sicherzustellen, dass die Datei **~/.ssh/authorized\_keys** des Benutzers auf dem Remote-Server mit seinem Public Key gefüllt wird, damit er sich anmelden kann.



### Literaturhinweise

Manpages **ssh(1)**, **sshd\_config(5)**

## ► Angeleitete Übung

# Anpassen der OpenSSH-Servicekonfiguration

In dieser Übung deaktivieren Sie direkte Anmeldungen als **root** und die passwortbasierte Authentifizierung für den OpenSSH-Service auf einem Server.

## Ergebnisse

Es werden folgende Fähigkeiten vermittelt:

- Deaktivieren direkter Anmeldungen als **root** über **ssh**
- Deaktivieren der passwortbasierten Authentifizierung für Remote-Benutzer zum Herstellen einer Verbindung über SSH

## Bevor Sie Beginnen

Melden Sie sich als Benutzer **student** mit dem Passwort **student** bei **workstation** an.

Führen Sie auf **workstation** das Skript **lab ssh-customize start** aus, um diese Übung zu beginnen. Dieses Skript erstellt die erforderlichen Benutzerkonten und Dateien.

```
[student@workstation ~]$ lab ssh-customize start
```

- 1. Öffnen Sie auf **workstation** als **student** eine SSH-Sitzung zu **serverb**.

```
[student@workstation ~]$ ssh student@serverb
...output omitted...
[student@serverb ~]$
```

- 2. Wechseln Sie auf **serverb** mit dem Befehl **su** zu **operator2**. Verwenden Sie **redhat** als Passwort für **operator2**.

```
[student@serverb ~]$ su - operator2
Password: redhat
[operator2@serverb ~]$
```

- 3. Generieren Sie mit dem Befehl **ssh-keygen** SSH-Schlüssel. Geben Sie keine Passphrase für die Schlüssel ein.

```
[operator2@serverb ~]$ ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/home/operator2/.ssh/id_rsa): Enter
Created directory '/home/operator2/.ssh'.
Enter passphrase (empty for no passphrase): Enter
Enter same passphrase again: Enter
Your identification has been saved in /home/operator2/.ssh/id_rsa.
```

**Kapitel 10 | Konfigurieren und Sichern von SSH**

```
Your public key has been saved in /home/operator2/.ssh/id_rsa.pub.
The key fingerprint is:
SHA256:JainiQdnRosC+xXh0qsJQQLzBNULdb+jJbyrCZQBERI
operator1@serverb.lab.example.com
The key's randomart image is:
+---[RSA 2048]----+
|E+*+ooo .      |
|.= o.o o .    |
|o.. = . . o   |
|+. + * . o .  |
|+= X . S +    |
| + @ + = .    |
| . + = o      |
| .o . . . .   |
|o     o..      |
+---[SHA256]-----+
```

- 4. Senden Sie mit dem Befehl **ssh-copy-id** den Public Key des SSH-Schlüsselpaares an **operator2** auf **servera**. Verwenden Sie auf **servera redhat** als Passwort für **operator2**.

```
[operator2@serverb ~]$ ssh-copy-id operator2@servera
/usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed: "/home/
operator1/.ssh/id_rsa.pub"
The authenticity of host 'servera (172.25.250.10)' can't be established.
ECDSA key fingerprint is SHA256:ERTdjoo0IrIwVSZQnqD5or+JbXfidg0udb3DXBuHWzA.
Are you sure you want to continue connecting (yes/no)? yes
/usr/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to filter
out any that are already installed
/usr/bin/ssh-copy-id: INFO: 1 key(s) remain to be installed -- if you are prompted
now it is to install the new keys
operator2@servera's password: redhat
Number of key(s) added: 1

Now try logging into the machine, with: "ssh 'operator2@servera'"
and check to make sure that only the key(s) you wanted were added.
```

- 5. Überprüfen Sie, ob Sie sich erfolgreich bei **servera** als **operator2** mit den SSH-Schlüsseln anmelden können.

- 5.1. Öffnen Sie als **operator2** eine SSH-Sitzung zu **servera**.

```
[operator2@serverb ~]$ ssh operator2@servera
...output omitted...
[operator2@servera ~]$
```

Beachten Sie, dass der vorherige Befehl **ssh** SSH-Schlüssel zur Authentifizierung verwendet.

- 5.2. Melden Sie sich von **servera** ab.

```
[operator2@servera ~]$ exit  
logout  
Connection to servera closed.
```

- 6. Überprüfen Sie, ob Sie sich erfolgreich bei **servera** als **root** mit dem Passwort **redhat** anmelden können.

- 6.1. Öffnen Sie als **root** eine SSH-Sitzung zu **servera** mit dem Passwort **redhat**.

```
[operator2@serverb ~]$ ssh root@servera  
root@servera's password: redhat  
...output omitted...  
[root@servera ~]#
```

Beachten Sie, dass der vorherige Befehl **ssh** das Passwort des Superusers zur Authentifizierung verwendet hat, da für den Superuser keine SSH-Schlüssel vorhanden sind.

- 6.2. Melden Sie sich von **servera** ab.

```
[root@servera ~]# exit  
logout  
Connection to servera closed.  
[operator2@serverb ~]$
```

- 7. Überprüfen Sie, ob Sie sich erfolgreich bei **servera** als **operator3** mit dem Passwort **redhat** anmelden können.

- 7.1. Öffnen Sie als **operator3** eine SSH-Sitzung zu **servera** mit dem Passwort **redhat**.

```
[operator2@serverb ~]$ ssh operator3@servera  
operator3@servera's password: redhat  
...output omitted...  
[operator3@servera ~]$
```

Beachten Sie, dass der vorherige Befehl **ssh** das Passwort von **operator3** zur Authentifizierung verwendet hat, da für **operator3** keine SSH-Schlüssel vorhanden sind.

- 7.2. Melden Sie sich von **servera** ab.

```
[operator3@servera ~]# exit  
logout  
Connection to servera closed.  
[operator2@serverb ~]$
```

- 8. Konfigurieren Sie **sshd** auf **servera**, um zu verhindern, dass sich Benutzer als **root** anmelden. Verwenden Sie **redhat** als Passwort des Superusers, falls erforderlich.

- 8.1. Öffnen Sie als **operator2** eine SSH-Sitzung zu **servera** mit den SSH-Schlüsseln.

```
[operator2@serverb ~]$ ssh operator2@servera
...output omitted...
[operator2@servera ~]$
```

- 8.2. Wechseln Sie auf **servera** zu **root**. Verwenden Sie **redhat** als Passwort des **root**-Benutzers.

```
[operator2@servera ~]$ su -
Password: redhat
[root@servera ~]#
```

- 8.3. Legen Sie in der Datei **/etc/ssh/sshd\_config** **PermitRootLogin** auf **no** fest und laden Sie **sshd** neu. Sie können **vim /etc/ssh/sshd\_config** verwenden, um die Konfigurationsdatei von **sshd** zu bearbeiten.

```
...output omitted...
PermitRootLogin no
...output omitted...
[root@servera ~]# systemctl reload sshd
```

- 8.4. Öffnen Sie auf **workstation** ein weiteres Terminal und öffnen Sie eine SSH-Sitzung zu **serverb** als **operator2**. Versuchen Sie, sich auf **serverb** bei **servera** als **root** anzumelden. Dies sollte fehlgeschlagen, da Sie im vorherigen Schritt die **root**-Benutzeranmeldung über SSH deaktiviert haben.



### Anmerkung

Die passwortbasierte Anmeldung ist in der Kursumgebung bereits zwischen **workstation** und **serverb** konfiguriert.

```
[student@workstation ~]$ ssh operator2@serverb
...output omitted...
[operator2@serverb ~]$ ssh root@servera
root@servera's password: redhat
Permission denied, please try again.
root@servera's password: redhat
Permission denied, please try again.
root@servera's password: redhat
root@servera: Permission denied (publickey,gssapi-keyex,gssapi-with-mic,password).
```

Standardmäßig versucht der Befehl **ssh** zuerst eine Authentifizierung mit der schlüsselbasierten Authentifizierung und, wenn dies fehlgeschlägt, die passwortbasierte Authentifizierung.

- 9. Konfigurieren Sie **sshd** auf **servera** so, dass sich Benutzer nur mit SSH-Schlüsseln und nicht mit ihren Passwörtern authentifizieren können.

- 9.1. Kehren Sie zum ersten Terminal zurück, auf dem die Shell des Benutzers **root** auf **servera** aktiv ist. Legen Sie in der Datei **/etc/ssh/sshd\_config** **PasswordAuthentication** auf **no** fest und laden Sie **sshd** neu. Sie können **vim /**

**/etc/ssh/sshd\_config** verwenden, um die Konfigurationsdatei von **sshd** zu bearbeiten.

```
...output omitted...
PasswordAuthentication no
...output omitted...
[root@servera ~]# systemctl reload sshd
```

- 9.2. Wechseln Sie zum zweiten Terminal, auf dem die Shell des Benutzers **operator2** auf **serverb** aktiv ist, und melden Sie sich bei **servera** als **operator3** an. Dies sollte fehlschlagen, da keine SSH-Schlüssel für **operator3** konfiguriert sind, und der **sshd**-Service auf **servera** die Verwendung von Passwörtern für die Authentifizierung nicht zulässt.

```
[operator2@serverb ~]$ ssh operator3@servera
operator3@servera: Permission denied (publickey,gssapi-keyex,gssapi-with-mic).
```



#### Anmerkung

Um eine größere Genauigkeit zu erreichen, können Sie die expliziten Optionen **-o PubkeyAuthentication=no** und **-o PasswordAuthentication=yes** mit dem Befehl **ssh** verwenden. Dadurch können Sie die Standardeinstellungen des Befehls **ssh** überschreiben und sicher ermitteln, dass der vorausgehende Befehl aufgrund der von Ihnen im vorherigen Schritt in **/etc/ssh/sshd\_config** angepassten Einstellungen fehlschlägt.

- 9.3. Kehren Sie zum ersten Terminal zurück, auf dem die Shell des Benutzers **root** auf **servera** aktiv ist. Überprüfen Sie, ob **PubkeyAuthentication** in **/etc/ssh/sshd\_config** aktiviert ist. Sie können **vim /etc/ssh/sshd\_config** verwenden, um die Konfigurationsdatei von **sshd** anzuzeigen.

```
...output omitted...
#PubkeyAuthentication yes
...output omitted...
```

Beachten Sie, dass die Zeile **PubkeyAuthentication** mit einem Kommentarzeichen versehen ist. Jede kommentierte Zeile in dieser Datei verwendet den Standardwert. Kommentierte Zeilen geben die Standardwerte eines Parameters an. Die Public-Key-Authentifizierung von SSH ist standardmäßig aktiv, wie aus der kommentierten Zeile hervorgeht.

- 9.4. Kehren Sie zum zweiten Terminal zurück, auf dem die Shell des Benutzers **operator2** auf **serverb** aktiv ist, und melden Sie sich bei **servera** als **operator2** an. Dies sollte erfolgreich sein, da die SSH-Schlüssel für **operator2** für die Anmeldung bei **servera** von **serverb** aus konfiguriert sind.

```
[operator2@serverb ~]$ ssh operator2@servera
...output omitted...
[operator2@servera ~]$
```

- 9.5. Beenden Sie im zweiten Terminal die Shell des Benutzers **operator2** auf **servera** und **serverb**.

```
[operator2@servera ~]$ exit  
logout  
Connection to servera closed.  
[operator2@serverb ~]$ exit  
logout  
Connection to serverb closed.  
[student@workstation ~]$
```

9.6. Schließen Sie das zweite Terminal auf **workstation**.

```
[student@workstation ~]$ exit
```

9.7. Beenden Sie im ersten Terminal die Shell des Benutzers **root** auf **servera**.

```
[root@servera ~]# exit  
logout
```

9.8. Beenden Sie im ersten Terminal die Shell des Benutzers **operator2** auf **servera** und **serverb**.

```
[operator2@servera ~]$ exit  
logout  
Connection to servera closed.  
[operator2@serverb ~]$ exit  
logout  
[student@serverb ~]$
```

9.9. Melden Sie sich von **serverb** ab und kehren Sie zur Shell des Benutzers **student** auf **workstation** zurück.

```
[student@serverb ~]$ exit  
logout  
Connection to serverb closed.  
[student@workstation ~]$
```

## Beenden

Führen Sie auf **workstation** das Skript **lab ssh-customize finish** aus, um diese Übung zu beenden.

```
[student@workstation ~]$ lab ssh-customize finish
```

Hiermit ist die angeleitete Übung beendet.

## ► Praktische Übung

# Konfigurieren und Sichern von SSH

### Leistungscheckliste

In dieser praktischen Übung richten Sie die schlüsselbasierte Authentifizierung für Benutzer ein und deaktivieren die direkte Anmeldung als **root** sowie die Passwortauthentifizierung für alle Benutzer für den OpenSSH-Service auf einem Server.

### Ergebnisse

Es werden folgende Fähigkeiten vermittelt:

- Authentifizieren mit SSH-Schlüsseln
- Verhindern der direkten Anmeldung von Benutzern als **root** über **ssh**
- Verhindern der Anmeldung von Benutzern beim System per passwortbasierter SSH-Authentifizierung

### Bevor Sie Beginnen

Melden Sie sich als Benutzer **student** mit dem Passwort **student** bei **workstation** an.

Führen Sie auf **workstation** das Skript **lab ssh-review start** aus, um diese Übung zu beginnen. Dieses Skript erstellt die erforderlichen Benutzerkonten und Dateien.

```
[student@workstation ~]$ lab ssh-review start
```

1. Öffnen Sie auf **workstation** als **student** eine SSH-Sitzung zu **servera**.
2. Wechseln Sie auf **servera** mit dem Befehl **su** zu **production1**.
3. Generieren Sie mit dem Befehl **ssh-keygen** SSH-Schlüssel ohne Passphrase für **production1** auf **servera**.
4. Senden Sie mit dem Befehl **ssh-copy-id** den Public Key des SSH-Schlüsselpaares an **production1** auf **serverb**.
5. Überprüfen Sie, ob **production1** sich erfolgreich bei **serverb** mit den SSH-Schlüsseln anmelden kann.
6. Konfigurieren Sie **sshd** auf **serverb**, um zu verhindern, dass sich Benutzer als **root** anmelden. Verwenden Sie als Passwort des Superusers **redhat**.
7. Konfigurieren Sie **sshd** auf **serverb** so, dass sich Benutzer nur mit SSH-Schlüsseln und nicht mit ihren Passwörtern authentifizieren können.

### Bewertung

Führen Sie auf **workstation** den Befehl **lab ssh-review grade** aus, um den Erfolg dieser Übung zu bestätigen.

```
[student@workstation ~]$ lab ssh-review grade
```

## Beenden

Führen Sie auf **workstation** das Skript **lab ssh-review finish** aus, um die praktische Übung abzuschließen.

```
[student@workstation ~]$ lab ssh-review finish
```

Hiermit wird die praktische Übung abgeschlossen.

## ► Lösung

# Konfigurieren und Sichern von SSH

### Leistungscheckliste

In dieser praktischen Übung richten Sie die schlüsselbasierte Authentifizierung für Benutzer ein und deaktivieren die direkte Anmeldung als **root** sowie die Passwortauthentifizierung für alle Benutzer für den OpenSSH-Service auf einem Server.

### Ergebnisse

Es werden folgende Fähigkeiten vermittelt:

- Authentifizieren mit SSH-Schlüsseln
- Verhindern der direkten Anmeldung von Benutzern als **root** über **ssh**
- Verhindern der Anmeldung von Benutzern beim System per passwortbasierter SSH-Authentifizierung

### Bevor Sie Beginnen

Melden Sie sich als Benutzer **student** mit dem Passwort **student** bei **workstation** an.

Führen Sie auf **workstation** das Skript **lab ssh-review start** aus, um diese Übung zu beginnen. Dieses Skript erstellt die erforderlichen Benutzerkonten und Dateien.

```
[student@workstation ~]$ lab ssh-review start
```

1. Öffnen Sie auf **workstation** als **student** eine SSH-Sitzung zu **servera**.

```
[student@workstation ~]$ ssh student@servera
...output omitted...
[student@servera ~]$
```

2. Wechseln Sie auf **servera** mit dem Befehl **su** zu **production1**.

```
[student@servera ~]$ su - production1
Password: redhat
[production1@servera ~]$
```

3. Generieren Sie mit dem Befehl **ssh-keygen** SSH-Schlüssel ohne Passphrase für **production1** auf **servera**.

```
[production1@servera ~]$ ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/home/production1/.ssh/id_rsa): Enter
Created directory '/home/production1/.ssh'.
Enter passphrase (empty for no passphrase): Enter
Enter same passphrase again: Enter
```

**Kapitel 10 | Konfigurieren und Sichern von SSH**

```
Your identification has been saved in /home/production1/.ssh/id_rsa.
Your public key has been saved in /home/production1/.ssh/id_rsa.pub.
The key fingerprint is:
SHA256:CsWCAmW0g5qaJujLzIAcengNj3u21kbrPP4Ys13PXCA
    production1@servera.lab.example.com
The key's randomart image is:
+---[RSA 2048]----+
|..o
|o+ . .
|= o . o
|.+   o
|o...   E .
|*o.= ... .
|Xo+ +oo.. .
|Oo .+==+ + .
| *o+o=*o. +
+---[SHA256]-----+
```

- 4.** Senden Sie mit dem Befehl **ssh-copy-id** den Public Key des SSH-Schlüsselpaars an **production1** auf **serverb**.

```
[production1@servera ~]$ ssh-copy-id production1@serverb
/usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed: "/home/
production1/.ssh/id_rsa.pub"
The authenticity of host 'serverb (172.25.250.11)' can't be established.
ECDSA key fingerprint is SHA256:ERTdjoo0IrIwVSZQnqD5or+JbXfidg0udb3DXBuHWzA.
Are you sure you want to continue connecting (yes/no)? yes
/usr/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to filter
out any that are already installed
/usr/bin/ssh-copy-id: INFO: 1 key(s) remain to be installed -- if you are prompted
now it is to install the new keys
production1@serverb's password: redhat
Number of key(s) added: 1

Now try logging into the machine, with: "ssh 'production1@serverb'"
and check to make sure that only the key(s) you wanted were added.
```

- 5.** Überprüfen Sie, ob **production1** sich erfolgreich bei **serverb** mit den SSH-Schlüsseln anmelden kann.

```
[production1@servera ~]$ ssh production1@serverb
...output omitted...
[production1@serverb ~]$
```

- 6.** Konfigurieren Sie **sshd** auf **serverb**, um zu verhindern, dass sich Benutzer als **root** anmelden. Verwenden Sie als Passwort des Superusers **redhat**.

- 6.1. Wechseln Sie auf **serverb** mit dem Befehl **su -** zu **root**.

```
[production1@serverb ~]$ su -
Password: redhat
[root@serverb ~]#
```

- 6.2. Legen Sie in der Datei `/etc/ssh/sshd_config` **PermitRootLogin** auf **no** fest und laden Sie **sshd** neu. Sie können **vim** `/etc/ssh/sshd_config` verwenden, um die Konfigurationsdatei von **sshd** zu bearbeiten.

```
...output omitted...
PermitRootLogin no
...output omitted...
[root@serverb ~]# systemctl reload sshd.service
```

- 6.3. Öffnen Sie auf **workstation** ein weiteres Terminal und öffnen Sie als **production1** eine SSH-Sitzung zu **servera**. Versuchen Sie, sich auf **servera** bei **serverb** als **root** anzumelden. Dies sollte fehlgeschlagen, da Sie im vorherigen Schritt die **root**-Benutzeranmeldung über SSH deaktiviert haben.



### Anmerkung

Die passwortbasierte Anmeldung ist in der Kursumgebung bereits zwischen **workstation** und **servera** konfiguriert.

```
[student@workstation ~]$ ssh production1@servera
...output omitted...
[production1@servera ~]$ ssh root@serverb
root@serverb's password: redhat
Permission denied, please try again.
root@serverb's password: redhat
Permission denied, please try again.
root@serverb's password: redhat
root@serverb: Permission denied (publickey,gssapi-keyex,gssapi-with-mic,password).
[production1@servera ~]$
```

Der vorherige Befehl **ssh** ist nach drei fehlgeschlagenen Anmeldeversuchen bei **servera** als **root** zurückgekehrt. Standardmäßig verwendet der Befehl **ssh** SSH-Schlüssel zur Authentifizierung. Wenn er jedoch die erforderlichen Schlüssel des Benutzers nicht findet, fordert er das Passwort des Benutzers für die Authentifizierung an.

7. Konfigurieren Sie **sshd** auf **serverb** so, dass sich Benutzer nur mit SSH-Schlüsseln und nicht mit ihren Passwörtern authentifizieren können.

- 7.1. Kehren Sie zum ersten Terminal zurück, auf dem die Shell des Benutzers **root** auf **serverb** aktiv ist. Legen Sie in der Datei `/etc/ssh/sshd_config` **PasswordAuthentication** auf **no** fest und laden Sie **sshd** neu. Sie können **vim** `/etc/ssh/sshd_config` verwenden, um die Konfigurationsdatei von **sshd** zu bearbeiten.

```
...output omitted...
PasswordAuthentication no
...output omitted...
[root@serverb ~]# systemctl reload sshd
```

- 7.2. Wechseln Sie zum zweiten Terminal, auf dem die Shell des Benutzers **production1** auf **servera** aktiv ist, und melden Sie sich bei **serverb** als **production2** an. Dies sollte fehlgeschlagen, da keine SSH-Schlüssel für **production2** konfiguriert

sind, und der **sshd**-Service auf **serverb** die Verwendung von Passwörtern für die Authentifizierung nicht zulässt.

```
[production1@servera ~]$ ssh production2@serverb  
production2@serverb: Permission denied (publickey,gssapi-keyex,gssapi-with-mic).
```



### Anmerkung

Um eine größere Genauigkeit zu erreichen, können Sie die expliziten Optionen **-o PubkeyAuthentication=no** und **-o PasswordAuthentication=yes** mit dem Befehl **ssh** verwenden. Dadurch können Sie die Standardeinstellungen des Befehls **ssh** überschreiben und sicher ermitteln, ob der vorausgehende Befehl aufgrund der von Ihnen im vorherigen Schritt in **/etc/ssh/sshd\_config** angepassten Einstellungen tatsächlich fehlschlägt.

- 7.3. Kehren Sie zum ersten Terminal zurück, auf dem die Shell des Benutzers **root** auf **serverb** aktiv ist. Überprüfen Sie, ob **PubkeyAuthentication** in **/etc/ssh/sshd\_config** aktiviert ist. Sie können **vim /etc/ssh/sshd\_config** verwenden, um die Konfigurationsdatei von **sshd** anzuzeigen.

```
...output omitted...  
#PubkeyAuthentication yes  
...output omitted...
```

Beachten Sie, dass die Zeile **PubkeyAuthentication** mit einem Kommentarzeichen versehen ist. Jede kommentierte Zeile in dieser Datei verwendet den Standardwert. Kommentierte Zeilen geben die Standardwerte eines Parameters an. Die Public-Key-Authentifizierung von SSH ist standardmäßig aktiv, wie aus der kommentierten Zeile hervorgeht.

- 7.4. Kehren Sie zum zweiten Terminal zurück, auf dem die Shell des Benutzers **production1** auf **servera** aktiv ist, und melden Sie sich bei **serverb** als **production1** an. Dies sollte erfolgreich sein, da die SSH-Schlüssel für **production1** für die Anmeldung bei **serverb** von **servera** aus konfiguriert sind.

```
[production1@servera ~]$ ssh production1@serverb  
...output omitted...  
[production1@serverb ~]$
```

- 7.5. Beenden Sie im zweiten Terminal die Shell des Benutzers **production1** auf **serverb** und **servera**.

```
[production1@serverb ~]$ exit  
logout  
Connection to serverb closed.  
[production1@servera ~]$ exit  
logout  
[student@workstation ~]$
```

- 7.6. Schließen Sie das zweite Terminal auf **workstation**.

```
[student@workstation ~]$ exit
```

- 7.7. Beenden Sie im ersten Terminal die Shell des Benutzers **root** auf **serverb**.

```
[root@serverb ~]# exit  
logout
```

- 7.8. Beenden Sie im ersten Terminal die Shell des Benutzers **production1** auf **serverb** und **servera**.

```
[production1@serverb ~]$ exit  
logout  
Connection to serverb closed.  
[production1@servera ~]$ exit  
logout  
[student@servera ~]$
```

- 7.9. Melden Sie sich von **servera** ab und kehren Sie zur Shell des Benutzers **student** auf **workstation** zurück.

```
[student@servera ~]$ exit  
logout  
Connection to servera closed.  
[student@workstation ~]$
```

## Bewertung

Führen Sie auf **workstation** den Befehl **lab ssh-review grade** aus, um den Erfolg dieser Übung zu bestätigen.

```
[student@workstation ~]$ lab ssh-review grade
```

## Beenden

Führen Sie auf **workstation** das Skript **lab ssh-review finish** aus, um die praktische Übung abzuschließen.

```
[student@workstation ~]$ lab ssh-review finish
```

Hiermit wird die praktische Übung abgeschlossen.

# Zusammenfassung

---

In diesem Kapitel erhielten Sie folgende Informationen:

- Der Befehl **ssh** ermöglicht Benutzern, auf Remote-Systeme sicher mit dem SSH-Protokoll zuzugreifen.
- Ein Client-System speichert die Identitäten der Remote-Server in `~/.ssh/known_hosts` und `/etc/ssh/ssh_known_hosts`.
- SSH unterstützt sowohl die passwortbasierte als auch die schlüsselbasierte Authentifizierung.
- Der Befehl **ssh-keygen** generiert ein SSH-Schlüsselpaar für die Authentifizierung. Der Befehl **ssh-copy-id** exportiert den Public Key auf Remote-Systeme.
- Der **sshd**-Service implementiert das SSH-Protokoll auf Red Hat Enterprise Linux-Systemen.
- Es ist ein empfohlenes Verfahren, **sshd** so zu konfigurieren, dass Remote-Anmeldungen als **root** deaktiviert sind und eine Authentifizierung mit Public Keys anstatt einer passwortbasierten Authentifizierung gefordert wird.

## Kapitel 11

# Analysieren und Speichern von Protokollen

### Ziel

Suchen und Auswerten von Systemprotokolldateien zur Fehlerbehebung.

### Ziele

- Beschreiben der grundlegenden Protokollierungsarchitektur, die von Red Hat Enterprise Linux zum Aufzeichnen von Ereignissen verwendet wird.
- Interpretieren von Ereignissen in relevanten Systemprotokolldateien zur Fehlerbehebung oder Prüfung des Systemstatus.
- Suchen und Interpretieren von Einträgen im Systemjournal zur Fehlerbehebung oder Prüfung des Systemstatus.
- Konfigurieren des Systemjournals, sodass die Aufzeichnung von Ereignissen beim Neubooten eines Servers erhalten bleibt.
- Beibehalten der genauen Zeitsynchronisierung mithilfe von NTP und konfigurieren der Zeitzone, um korrekte Zeitstempel für Ereignisse zu gewährleisten, die vom Systemjournal und den Protokollen aufgezeichnet werden.

### Abschnitte

- Beschreiben der Systemprotokollarchitektur (und Test)
- Überprüfen von Systemprotokolldateien (und angeleitete Übung)
- Überprüfen von Systemjournal-Einträgen (und angeleitete Übung)
- Bewahren des Systemjournals (und angeleitete Übung)
- Verwalten der genauen Uhrzeit (und angeleitete Übung)

### Praktische Übung

Analysieren und Speichern von Protokollen

# Beschreiben der Systemprotokollarchitektur

---

## Ziele

Nach Abschluss dieses Abschnitts sollten Sie in der Lage sein, die grundlegende Protokollierungsarchitektur zu beschreiben, die von Red Hat Enterprise Linux zum Aufzeichnen von Ereignissen verwendet wird.

## Systemprotokollierung

Prozesse und der Betriebssystem-Kernel zeichnen Ereignisse mittels Protokollen auf. Diese Protokolle werden für das System-Auditing und die Fehlersuche genutzt.

Viele Systeme zeichnen Ereignisprotokolle in Textdateien auf, die im Verzeichnis **/var/log** gespeichert werden. Diese Protokolle können mit normalen Textdienstprogrammen wie **less** und **tail** untersucht werden.

Red Hat Enterprise Linux umfasst ein standardisiertes Protokollierungssystem basierend auf dem **Syslog**-Protokoll. Dieses System wird von vielen Programmen verwendet, um Ereignisse aufzuzeichnen und sie in Protokolldateien zu organisieren. Die Services **systemd-journald** und **rsyslog** verarbeiten in Red Hat Enterprise Linux 8 die Syslog-Meldungen.

Der Service **systemd-journald** ist das zentrale Element der Ereignisprotokollierungsarchitektur des Betriebssystems. Er erfasst Ereignismeldungen von vielen Quellen, wie dem Kernel, Ausgaben aus den frühen Stadien des Systembootvorgangs, Standardausgaben und Standardfehler von Daemons bei ihrem Start und ihrer Ausführung sowie Syslog-Ereignisse. Der Service strukturiert diese dann in ein Standardformat um und schreibt sie in ein strukturiertes, indiziertes Systemjournal. Standardmäßig wird dieses Journal in einem Dateisystem gespeichert, das nach Bootvorgängen nicht bestehen bleibt.

Der Service **rsyslog** liest jedoch Syslog-Meldungen, die von **systemd-journald** empfangen werden, aus dem Journal, sobald sie eingehen. Anschließend verarbeitet er die Syslog-Ereignisse und zeichnet sie in seinen Protokolldateien auf oder leitet sie an andere Services weiter, je nach seiner Konfiguration.

Der Service **rsyslog** sortiert die Syslog-Meldungen und schreibt sie in Protokolldateien, die nach Bootvorgängen in **/var/log** erhalten bleiben. Der Service **rsyslog** sortiert die Protokollmeldungen nach Typ des Programms, das die jeweilige Meldung gesendet hat, oder nach *Facility* und der Priorität jeder Syslog-Meldung in bestimmte Protokolldateien.

Zusätzlich zu den Syslog-Meldungsdateien enthält das Verzeichnis **/var/log** Protokolldateien von anderen Services auf dem System. In der folgenden Tabelle sind einige nützliche Dateien aus dem Verzeichnis **/var/log** aufgeführt.

**Ausgewählte Systemprotokolldateien**

| Protokolldatei           | Typ der gespeicherten Meldung                                                                                                                                                                                                            |
|--------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>/var/log/messages</b> | Die meisten Syslog-Meldungen werden hier gespeichert. Ausgenommen davon sind Meldungen zur Authentifizierung, für die E-Mail-Verarbeitung, zur Ausführung terminierter Jobs und solche, die nur für die Fehlerbehebung verwendet werden. |
| <b>/var/log/secure</b>   | Syslog-Meldungen im Zusammenhang mit Sicherheits- und Authentifizierungsereignissen.                                                                                                                                                     |
| <b>/var/log/maillog</b>  | Syslog-Meldungen im Zusammenhang mit dem Mailserver.                                                                                                                                                                                     |
| <b>/var/log/cron</b>     | Syslog-Meldungen im Zusammenhang mit der Ausführung terminierter Jobs.                                                                                                                                                                   |
| <b>/var/log/boot.log</b> | Konsolenmeldungen im Zusammenhang mit dem Systemstart, nicht von Syslog.                                                                                                                                                                 |

**Anmerkung**

Einige Anwendungen verwenden syslog nicht, um Protokollnachrichten zu verwalten, legen aber normalerweise dennoch ihre Protokolldateien in einem Unterverzeichnis von /var/log ab. Der Apache-Webserver beispielsweise speichert Protokollmeldungen in Dateien in einem Unterverzeichnis des Verzeichnisses **/var/log**.

**Literaturhinweise**

Manpages **systemd-journald.service(8)**, **rsyslogd(8)** und **rsyslog.conf(5)**

Weitere Informationen finden Sie im Abschnitt *Troubleshooting problems using log files* im Handbuch *Red Hat Enterprise Linux 8 Configuring basic system settings* unter

[https://access.redhat.com/documentation/en-us/red\\_hat\\_enterprise\\_linux/8/html-single/configuring\\_basic\\_system\\_settings/index#troubleshooting-problems-using-log-files\\_getting-started-with-system-administration](https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux/8/html-single/configuring_basic_system_settings/index#troubleshooting-problems-using-log-files_getting-started-with-system-administration)

## ► Quiz

# Beschreiben der Systemprotokollarchitektur

Wählen Sie die richtige Antwort auf die folgenden Fragen aus:

- ▶ 1. In welche dieser Protokolldateien werden die meisten Syslog-Meldungen mit Ausnahme derjenigen, die sich auf Authentifizierung, E-Mail, terminierte Jobs und Debuggen beziehen, gespeichert?
  - a. /var/log/maillog
  - b. /var/log/boot.log
  - c. /var/log/messages
  - d. /var/log/secure
- ▶ 2. In welcher Protokolldatei werden Syslog-Meldungen gespeichert, die sich auf Sicherheits- und Authentifizierungsvorgänge im System beziehen?
  - a. /var/log/maillog
  - b. /var/log/boot.log
  - c. /var/log/messages
  - d. /var/log/secure
- ▶ 3. Welcher Service sortiert und organisiert Syslog-Meldungen in Dateien in /var/log?
  - a. rsyslog
  - b. systemd-journald
  - c. auditd
  - d. abgestimmt
- ▶ 4. Welches Verzeichnis enthält die von Menschen lesbaren Syslog-Dateien?
  - a. /sys/kernel/debug
  - b. /var/log/journal
  - c. /run/log/journal
  - d. /var/log
- ▶ 5. In welcher Datei werden Syslog-Meldungen im Zusammenhang mit dem Mailserver gespeichert?
  - a. /var/log/lastlog
  - b. /var/log/maillog
  - c. /var/log/tallylog
  - d. /var/log/boot.log

- ▶ 6. In welcher Datei werden Syslog-Meldungen im Zusammenhang mit terminierten Jobs gespeichert?
  - a. /var/log/cron
  - b. /var/log/tallylog
  - c. /var/log/spooler
  - d. /var/log/secure
  
- ▶ 7. In welcher Datei werden Konsolenmeldungen im Zusammenhang mit dem Systemstart gespeichert?
  - a. /var/log/messages
  - b. /var/log/cron
  - c. /var/log/boot.log
  - d. /var/log/secure

## ► Lösung

# Beschreiben der Systemprotokollarchitektur

Wählen Sie die richtige Antwort auf die folgenden Fragen aus:

- ▶ 1. In welche dieser Protokolldateien werden die meisten Syslog-Meldungen mit Ausnahme derjenigen, die sich auf Authentifizierung, E-Mail, terminierte Jobs und Debuggen beziehen, gespeichert?
  - a. /var/log/maillog
  - b. /var/log/boot.log
  - c. /var/log/messages
  - d. /var/log/secure
  
- ▶ 2. In welcher Protokolldatei werden Syslog-Meldungen gespeichert, die sich auf Sicherheits- und Authentifizierungsvorgänge im System beziehen?
  - a. /var/log/maillog
  - b. /var/log/boot.log
  - c. /var/log/messages
  - d. /var/log/secure
  
- ▶ 3. Welcher Service sortiert und organisiert Syslog-Meldungen in Dateien in /var/log?
  - a. rsyslog
  - b. systemd-journald
  - c. auditd
  - d. abgestimmt
  
- ▶ 4. Welches Verzeichnis enthält die von Menschen lesbaren Syslog-Dateien?
  - a. /sys/kernel/debug
  - b. /var/log/journal
  - c. /run/log/journal
  - d. /var/log
  
- ▶ 5. In welcher Datei werden Syslog-Meldungen im Zusammenhang mit dem Mailserver gespeichert?
  - a. /var/log/lastlog
  - b. /var/log/maillog
  - c. /var/log/tallylog
  - d. /var/log/boot.log

► 6. In welcher Datei werden Syslog-Meldungen im Zusammenhang mit terminierten Jobs gespeichert?

- a. **/var/log/cron**
- b. **/var/log/tallylog**
- c. **/var/log/spooler**
- d. **/var/log/secure**

► 7. In welcher Datei werden Konsolenmeldungen im Zusammenhang mit dem Systemstart gespeichert?

- a. **/var/log/messages**
- b. **/var/log/cron**
- c. **/var/log/boot.log**
- d. **/var/log/secure**

# Überprüfen von syslog-Dateien

## Ziele

Nachdem Sie diesen Abschnitt durchgearbeitet haben, sollten Sie Ereignisse in den relevanten syslog-Dateien interpretieren können, um Probleme zu beheben oder den Systemstatus zu überprüfen.

## Protokollieren von Ereignissen im System

Viele Programme verwenden das **syslog**-Protokoll für die Protokollierung von Ereignissen auf dem System. Jede Protokollmeldung wird in eine Facility (Art der Meldung) und eine Priorität (Schweregrad der Meldung) eingeteilt. Die verfügbaren Facilitys sind auf der Manpage **rsyslog.conf(5)** dokumentiert.

In der folgenden Tabelle sind die acht Standard-Syslog-Prioritäten von der höchsten bis zur niedrigsten aufgeführt.

### Übersicht der Systemprotokoll-Prioritäten

| Code | Priorität | Schweregrad                                   |
|------|-----------|-----------------------------------------------|
| 0    | emerg     | Das System kann nicht mehr verwendet werden.  |
| 1    | Alarm     | Es müssen sofort Maßnahmen ergriffen werden   |
| 2    | crit      | Es liegt ein schwerwiegender Fehler vor       |
| 3    | err       | Es liegt ein nicht schwerwiegender Fehler vor |
| 4    | Warnung   | Es liegt eine Warnung vor                     |
| 5    | Hinweis   | Das Ereignis ist normal, aber bedeutend       |
| 6    | Info      | Es liegt ein Informationsereignis vor         |
| 7    | debug     | Es liegt eine Debugging-Meldung vor           |

Der Service **rsyslog** bestimmt anhand der Facility und der Priorität von Protokollmeldungen, wie diese gehandhabt werden. Dies wird durch Regeln in der Datei **/etc/rsyslog.conf** und in jeder Datei im Verzeichnis **/etc/rsyslog.d** mit der Dateinamenerweiterung **.conf** konfiguriert. Softwarepakete können ganz einfach Regeln hinzufügen, indem eine entsprechende Datei im Verzeichnis **/etc/rsyslog.d** installiert wird.

Jede Regel, die das Sortieren von Syslog-Meldungen steuert, ist eine Zeile in einer der Konfigurationsdateien. Die linke Hälfte jeder Zeile gibt die Facility und den Schweregrad der Syslog-Meldung an, die mit der Regel übereinstimmt. Die rechte Hälfte jeder Zeile gibt an, in welcher Datei die Protokollmeldung gespeichert werden soll (oder wo sonst die Meldung gespeichert werden soll). Ein Sternchen (\*) ist ein Platzhalter, der allen Werten entspricht.

In der folgenden Zeile werden beispielsweise Meldungen in der Datei **/var/log/secure** aufgezeichnet, die an die Facility **authpriv** mit einer beliebigen Priorität gesendet wurden:

```
authpriv.*          /var/log/secure
```

Protokollmeldungen stimmen manchmal mit mehr als einer Regel in **rsyslog.conf** überein. In solchen Fällen wird eine Meldung in mehreren Protokolldateien gespeichert. Zum Begrenzen der gespeicherten Meldungen bedeutet das Schlüsselwort **none** im Prioritätsfeld, dass keine Meldungen für die angegebene Facility in der angegebenen Datei gespeichert werden sollen.

Statt der Protokollierung von Syslog-Meldungen in einer Datei können diese auch auf den Terminals aller angemeldeten Benutzer ausgegeben werden. Die Datei **rsyslog.conf** enthält eine Einstellung zum Ausgeben aller Syslog-Meldungen mit der Priorität **emerg** auf den Terminals aller angemeldeten Benutzer.

## Beispielregeln für Rsyslog

```
##### RULES #####
# Log all kernel messages to the console.
# Logging much else clutters up the screen.
#kern.*                      /dev/console

# Log anything (except mail) of level info or higher.
# Don't log private authentication messages!
*.info;mail.none;authpriv.none;cron.none      /var/log/messages

# The authpriv file has restricted access.
authpriv.*                     /var/log/secure

# Log all the mail messages in one place.
mail.*                         -/var/log/maillog

# Log cron stuff
cron.*                          /var/log/cron

# Everybody gets emergency messages
*. emerg                         :omusrmsg:*

# Save news errors of level crit and higher in a special file.
uucp,news.crit                  /var/log/spooler

# Save boot messages also to boot.log
local7.*                        /var/log/boot.log
```



### Anmerkung

Das Syslog-Subsystem bietet viele weitere Funktionen, die über den Rahmen dieses Kurses hinausgehen. Wenn Sie dieses Thema weiter erkunden möchten, sollten Sie die Manpage **rsyslog.conf(5)** und die umfangreiche HTML-Dokumentation in **/usr/share/doc/rsyslog/html/index.html** aufrufen, die im Paket *rsyslog-doc* enthalten und im AppStream-Repository von Red Hat Enterprise Linux 8 verfügbar ist.

## Rotation der Protokolldateien

Das Tool **logrotate** rotiert die Protokolldateien so, dass sie nicht zu viel Speicherplatz in dem Dateisystem beanspruchen, welches das Verzeichnis **/var/log** enthält. Wenn eine Protokolldatei rotiert wird, wird sie mit einer Erweiterung umbenannt, die das Datum der Rotation angibt. Zum Beispiel kann die alte Datei **/var/log/messages** zu **/var/log/messages-20190130** werden, wenn sie am 30.01.2019 rotiert wird. Nachdem die alte Protokolldatei rotiert wurde, wird eine neue Protokolldatei erstellt und der Service, der in sie schreibt, benachrichtigt.

Nach einer bestimmten Anzahl von Rotationen – üblicherweise nach vier Wochen – wird die älteste Protokolldatei gelöscht, um Festplattenspeicherplatz zu sparen. Ein terminierter Job führt das Programm **logrotate** aus, um zu ermitteln, ob Protokolle rotiert werden müssen. Die meisten Protokolldateien werden wöchentlich rotiert. Manche Dateien werden jedoch von **logrotate** schneller oder langsamer rotiert, wenn sie eine bestimmte Größe erreichen.

Die Konfiguration von **logrotate** wird in diesem Kurs nicht erläutert. Weitere Informationen finden Sie auf der Manpage **logrotate(8)**.

## Analysieren eines Syslog-Eintrags

Protokollmeldungen beginnen mit der ältesten Meldung am Anfang und der neuesten Meldung am Ende der Protokolldatei. Der Service **rsyslog** verwendet für die Aufzeichnung von Einträgen in Protokolldateien ein Standardformat. Das folgende Beispiel veranschaulicht den Aufbau einer Protokollmeldung in der Protokolldatei **/var/log/secure**:

```
① Feb 11 20:11:48 ② localhost ③ sshd[1433]: ④ Failed password for student from
172.25.0.10 port 59344 ssh2
```

- ① Der Zeitstempel zum Zeitpunkt der Aufzeichnung des Protokolleintrags
- ② Der Host, von dem die Protokollmeldung gesendet wurde
- ③ Das Programm oder der Prozessname und die PID-Nummer, die die Protokollmeldung gesendet haben
- ④ Die eigentliche gesendete Meldung

## Überwachen von Protokollen

Für das Reproduzieren von Problemen ist es hilfreich, eine oder mehrere Protokolldateien auf Ereignisse hin zu überwachen. Mit dem Befehl **tail -f /path/to/file** werden die letzten zehn Zeilen der angegebenen Datei sowie fortlaufend neue Zeilen ausgegeben, während sie geschrieben werden.

Um z. B. fehlgeschlagene Anmeldeversuche zu überwachen, führen Sie den Befehl **tail** erst in einem und dann in einem anderen Terminal aus und führen Sie den Befehl **ssh** als **root**-Benutzer aus, während ein Benutzer versucht, sich im System anzumelden.

Führen Sie im ersten Terminal den folgenden **tail**-Befehl aus:

```
[root@host ~]# tail -f /var/log/secure
```

Führen Sie im zweiten Terminal den folgenden **ssh**-Befehl aus:

```
[root@host ~]# ssh root@localhost  
root@localhost's password: redhat  
...output omitted...  
[root@host ~]#
```

Kehren Sie zum ersten Terminal zurück und sehen Sie sich die Protokolle an.

```
...output omitted...  
Feb 10 09:01:13 host sshd[2712]: Accepted password for root from 172.25.254.254  
port 56801 ssh2  
Feb 10 09:01:13 host sshd[2712]: pam_unix(sshd:session): session opened for user  
root by (uid=0)
```

## Manuelles Senden von Syslog-Meldungen

Mit dem Befehl **logger** können Meldungen an den **rsyslog**-Service gesendet werden. Standardmäßig wird die Meldung an die Facility **user** mit der Priorität **notice** (**user.notice**) gesendet, sofern mit der Option **-p** nicht anders angegeben. Es ist sinnvoll, Änderungen in der **rsyslog**-Servicekonfiguration zu testen.

Um eine Meldung an den Service **rsyslog** zu senden, der in der Protokolldatei **/var/log/boot.log** aufgezeichnet wird, führen Sie den folgenden **logger**-Befehl aus:

```
[root@host ~]# logger -p local7.notice "Log entry created on host"
```



### Literaturhinweise

Manpages **logger(1)**, **tail(1)**, **rsyslog.conf(5)** und **logrotate(8)**

### rsyslog Manual

- **/usr/share/doc/rsyslog/html/index.html** – wird vom Paket *rsyslog-doc* bereitgestellt

Weitere Informationen finden Sie im Abschnitt *Troubleshooting problems using log files* im Handbuch *Red Hat Enterprise Linux 8 Configuring basic system settings* unter

[https://access.redhat.com/documentation/en-us/red\\_hat\\_enterprise\\_linux/8/html-single/configuring\\_basic\\_system\\_settings/index#troubleshooting-problems-using-log-files\\_getting-started-with-system-administration](https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux/8/html-single/configuring_basic_system_settings/index#troubleshooting-problems-using-log-files_getting-started-with-system-administration)

## ► Angeleitete Übung

# Überprüfen von syslog-Dateien

In dieser Übung konfigurieren Sie **rsyslog** neu, damit bestimmte Protokollmeldungen in eine neue Datei geschrieben werden.

## Ergebnisse

Sie sollten in der Lage sein, den **rsyslog**-Service so zu konfigurieren, dass alle Protokollmeldungen mit Priorität **debug** in die Protokolldatei **/var/log/messages-debug** geschrieben werden.

## Bevor Sie Beginnen

Melden Sie sich als Benutzer **student** mit dem Passwort **student** bei **workstation** an.

Führen Sie auf **workstation** das Skript **lab log-configure start** aus, um diese Übung zu beginnen. Das Skript stellt sicher, dass die Umgebung richtig eingerichtet ist.

```
[student@workstation ~]$ lab log-configure start
```

- 1. Öffnen Sie auf **workstation** als **student** eine SSH-Sitzung zu **servera**.

```
[student@workstation ~]$ ssh student@servera
...output omitted...
[student@servera ~]$
```

- 2. Konfigurieren Sie **rsyslog** auf **servera** so, dass alle Meldungen mit der Priorität **debug** oder höher für jeden Service in der neuen Protokolldatei **/var/log/messages-debug** protokolliert werden, indem Sie die **rsyslog**-Konfigurationsdatei **/etc/rsyslog.d/debug.conf** hinzufügen.

- 2.1. Verwenden Sie den Befehl **sudo -i**, um zum **root**-Benutzer zu wechseln. Geben Sie **student** als Passwort für den Benutzer **student** ein, wenn während der Ausführung des Befehls **sudo -i** danach gefragt wird.

```
[student@servera ~]$ sudo -i
[sudo] password for student: student
[root@servera ~]#
```

- 2.2. Erstellen Sie die Datei **/etc/rsyslog.d/debug.conf** mit den erforderlichen Einträgen, um alle Protokollmeldungen mit der Priorität **debug** zu **/var/log/messages-debug** umzuleiten. Sie können den Befehl **vim /etc/rsyslog.d/debug.conf** verwenden, um die Datei mit dem folgenden Inhalt zu erstellen.

```
*.debug /var/log/messages-debug
```

Diese Konfigurationszeile fängt Syslog-Meldungen mit einer beliebigen Facility und der Prioritätsstufe **debug** oder höher ab. Der Service **rsyslog** schreibt die entsprechenden Meldungen in die Datei **/var/log/messages-debug**. Der Platzhalter (\*) im Feld **facility** oder **priority** der Konfigurationszeile gibt eine beliebige Facility oder Priorität an.

- 2.3. Starten Sie den **rsyslog**-Service neu.

```
[root@servera ~]# systemctl restart rsyslog
```

- 3. Stellen Sie sicher, dass alle Protokollmeldungen mit der Priorität **debug** in der Datei **/var/log/messages-debug** erscheinen.

- 3.1. Verwenden Sie den Befehl **logger** mit der Option **-p** zum Erzeugen einer Protokollmeldung mit der Facility **user** und der Priorität **debug**.

```
[root@servera ~]# logger -p user.debug "Debug Message Test"
```

- 3.2. Verwenden Sie den Befehl **tail** zum Anzeigen der letzten zehn Protokollmeldungen aus der **/var/log/messages-debug**-Datei und überprüfen Sie, ob Sie neben anderen Protokollmeldungen die Meldung **Debug Message Test** sehen.

```
[root@servera ~]# tail /var/log/messages-debug
Feb 13 18:22:38 servera systemd[1]: Stopping System Logging Service...
Feb 13 18:22:38 servera rsyslogd[25176]: [origin software="rsyslogd"
  swVersion="8.37.0-9.el8" x-pid="25176" x-info="http://www.rsyslog.com"] exiting
  on signal 15.
Feb 13 18:22:38 servera systemd[1]: Stopped System Logging Service.
Feb 13 18:22:38 servera systemd[1]: Starting System Logging Service...
Feb 13 18:22:38 servera rsyslogd[25410]: environment variable TZ is not set, auto
  correcting this to TZ=/etc/localtime [v8.37.0-9.el8 try http://www.rsyslog.com/
  e/2442 ]
Feb 13 18:22:38 servera systemd[1]: Started System Logging Service.
Feb 13 18:22:38 servera rsyslogd[25410]: [origin software="rsyslogd"
  swVersion="8.37.0-9.el8" x-pid="25410" x-info="http://www.rsyslog.com"] start
Feb 13 18:27:58 servera student[25416]: Debug Message Test
```

- 3.3. Verlassen Sie die Benutzer-Shells **root** und **student** auf **servera** und kehren Sie zur Shell des Benutzers **student** auf **workstation** zurück.

```
[root@servera ~]# exit
logout
[student@servera ~]$ exit
logout
Connection to servera closed.
[student@workstation ~]$
```

## Beenden

Führen Sie auf **workstation** das Skript **lab log-configure finish** aus, um diese Übung zu beenden. Das Skript stellt sicher, dass die Umgebung als eine bereinigte Übungsumgebung eingerichtet wird.

```
[student@workstation ~]$ lab log-configure finish
```

Hiermit ist die angeleitete Übung beendet.

# Überprüfen von Systemjournal-Einträgen

## Ziele

Nach Abschluss dieses Abschnitts sollten Sie in der Lage sein, Einträge im Systemjournal zu finden und zu interpretieren, um Probleme zu beheben oder den Systemstatus zu überprüfen.

## Finden von Ereignissen

Der Service **systemd-journald** speichert Protokolldaten in einer strukturierten, indizierten Binärdatei, die als Journal bezeichnet wird. Diese Daten umfassen zusätzliche Informationen zum Protokollereignis. Im Fall von Syslog-Ereignissen können diese Daten z. B. die Facility und die Priorität der ursprünglichen Meldung enthalten.



### Wichtig

In Red Hat Enterprise Linux 8 enthält standardmäßig das Verzeichnis **/run/log** das Systemjournal. Der Inhalt des Verzeichnisses **/run/log** wird nach einem Bootvorgang gelöscht. Sie können diese Einstellung ändern. Wie Sie dies tun, wird weiter unten in diesem Kapitel erläutert.

Verwenden Sie zum Abrufen von Protokollmeldungen aus dem Journal den Befehl **journalctl**. Mit diesem Befehl können Sie alle Meldungen im Journal anzeigen oder anhand einer Vielzahl von Optionen und Kriterien nach bestimmten Ereignissen suchen. Wenn Sie den Befehl als **root** ausführen, haben Sie vollen Zugriff auf das Journal. Reguläre Benutzer können diesen Befehl ebenfalls verwenden, können jedoch möglicherweise bestimmte Meldungen nicht sehen.

```
[root@host ~]# journalctl
...output omitted...
Feb 21 17:46:25 host.lab.example.com systemd[24263]: Stopped target Sockets.
Feb 21 17:46:25 host.lab.example.com systemd[24263]: Closed D-Bus User Message Bus
Socket.
Feb 21 17:46:25 host.lab.example.com systemd[24263]: Closed Multimedia System.
Feb 21 17:46:25 host.lab.example.com systemd[24263]: Reached target Shutdown.
Feb 21 17:46:25 host.lab.example.com systemd[24263]: Starting Exit the Session...
Feb 21 17:46:25 host.lab.example.com systemd[24268]: pam_unix(systemd-
user:session): session c>
Feb 21 17:46:25 host.lab.example.com systemd[1]: Stopped User Manager for UID
1001.
Feb 21 17:46:25 host.lab.example.com systemd[1]: user-runtime-dir@1001.service:
Unit not needed
Feb 21 17:46:25 host.lab.example.com systemd[1]: Stopping /run/user/1001 mount
wrapper...
Feb 21 17:46:25 host.lab.example.com systemd[1]: Removed slice User Slice of UID
1001.
Feb 21 17:46:25 host.lab.example.com systemd[1]: Stopped /run/user/1001 mount
wrapper.
Feb 21 17:46:36 host.lab.example.com sshd[24434]: Accepted publickey for root from
172.25.250.>
```

**Kapitel 11 |** Analysieren und Speichern von Protokollen

```
Feb 21 17:46:37 host.lab.example.com systemd[1]: Started Session 20 of user root.
Feb 21 17:46:37 host.lab.example.com systemd-logind[708]: New session 20 of user
root.
Feb 21 17:46:37 host.lab.example.com sshd[24434]: pam_unix(sshd:session): session
opened for u>
Feb 21 18:01:01 host.lab.example.com CROND[24468]: (root) CMD (run-parts /etc/
cron.hourly)
Feb 21 18:01:01 host.lab.example.com run-parts[24471]: (/etc/cron.hourly) starting
@anacron
Feb 21 18:01:01 host.lab.example.com run-parts[24477]: (/etc/cron.hourly) finished
@anacron
lines 1464-1487/1487 (END) q
```

Der Befehl **journalctl** hebt wichtige Protokollmeldungen hervor: Meldungen mit den Prioritäten **notice** oder **warning** erscheinen fett, während Meldungen der Priorität **error** oder höher in roter Schrift angezeigt werden.

Der Schlüssel für eine erfolgreiche Nutzung des Journals für die Fehlerbehebung und Prüfung liegt in der Einschränkung von Journalsuchen, sodass nur relevante Ausgaben angezeigt werden.

Standardmäßig werden mit **journalctl -n** die letzten zehn Protokolleinträge angezeigt. Sie können dies mit einem optionalen Argument anpassen, das angibt, wie viele Protokolleinträge angezeigt werden sollen. Führen Sie den folgenden **journalctl**-Befehl aus, um die letzten fünf Protokolleinträge anzuzeigen:

```
[root@host ~]# journalctl -n 5
-- Logs begin at Wed 2019-02-20 16:01:17 +07, end at Thu 2019-02-21 18:01:01 +07.
--
...output omitted...
Feb 21 17:46:37 host.lab.example.com systemd-logind[708]: New session 20 of user
root.
Feb 21 17:46:37 host.lab.example.com sshd[24434]: pam_unix(sshd:session): session
opened for u>
Feb 21 18:01:01 host.lab.example.com CROND[24468]: (root) CMD (run-parts /etc/
cron.hourly)
Feb 21 18:01:01 host.lab.example.com run-parts[24471]: (/etc/cron.hourly) starting
@anacron
Feb 21 18:01:01 host.lab.example.com run-parts[24477]: (/etc/cron.hourly) finished
@anacron
lines 1-6/6 (END) q
```

Ähnlich wie mit dem Befehl **tail -f** werden mit dem Befehl **journalctl -f** die letzten zehn Zeilen des Systemjournals sowie fortlaufend neue Journaleinträge, während sie in das Journal geschrieben werden, ausgegeben. Um den Prozess **journalctl -f** zu verlassen, verwenden Sie die **Strg+C**-Tastenkombination.

```
[root@host ~]# journalctl -f
-- Logs begin at Wed 2019-02-20 16:01:17 +07. --
...output omitted...
Feb 21 18:01:01 host.lab.example.com run-parts[24477]: (/etc/cron.hourly) finished
@anacron
Feb 21 18:22:42 host.lab.example.com sshd[24437]: Received disconnect from
172.25.250.250 port 48710:11: disconnected by user
```

```

Feb 21 18:22:42 host.lab.example.com sshd[24437]: Disconnected from user root
172.25.250.250 port 48710
Feb 21 18:22:42 host.lab.example.com sshd[24434]: pam_unix(sshd:session): session
closed for user root
Feb 21 18:22:42 host.lab.example.com systemd-logind[708]: Session 20 logged out.
Waiting for processes to exit.
Feb 21 18:22:42 host.lab.example.com systemd-logind[708]: Removed session 20.
Feb 21 18:22:43 host.lab.example.com sshd[24499]: Accepted
publickey for root from 172.25.250.250 port 48714 ssh2: RSA
SHA256:1UGybTe52L2jzEJa1HLVKn9QUCKrTv3ZxxnMjol1Fro
Feb 21 18:22:44 host.lab.example.com systemd-logind[708]: New session 21 of user
root.
Feb 21 18:22:44 host.lab.example.com systemd[1]: Started Session 21 of user root.
Feb 21 18:22:44 host.lab.example.com sshd[24499]: pam_unix(sshd:session): session
opened for user root by (uid=0)
^C
[root@host ~]#

```

Für die Behebung von Problemen ist es hilfreich, die Ausgabe des Journals nach der Priorität der Journaleinträge zu filtern. Der Befehl **journalctl -p** wird mit dem Namen oder der Nummer einer Prioritätsstufe ausgeführt und zeigt die Journaleinträge mit dieser Priorität oder höher an. Der Befehl **journalctl** kennt die Prioritätsstufen **debug, info, notice, warning, err, crit, alert** und **emerg**.

Führen Sie den folgenden **journalctl**-Befehl zum Auflisten der Journaleinträge mit der Priorität **err** oder höher aus:

```

[root@host ~]# journalctl -p err
-- Logs begin at Wed 2019-02-20 16:01:17 +07, end at Thu 2019-02-21 18:01:01 +07.
--
..output omitted...
Feb 20 16:01:17 host.lab.example.com kernel: Detected CPU family 6 model 13
stepping 3
Feb 20 16:01:17 host.lab.example.com kernel: Warning: Intel Processor - this
hardware has not undergone testing by Red Hat and might not be certif>
Feb 20 16:01:20 host.lab.example.com smartd[669]: DEVICESCAN failed: glob(3)
aborted matching pattern /dev/discs/disc*
Feb 20 16:01:20 host.lab.example.com smartd[669]: In the system's table of devices
NO devices found to scan
lines 1-5/5 (END) q

```

Wenn Sie nach bestimmten Ereignissen suchen, können Sie die Ausgabe auf einen bestimmten Zeitrahmen zu beschränken. Der Befehl **journalctl** verfügt über zwei Optionen, mit denen die Ausgabe auf einen bestimmten Zeitrahmen beschränkt werden kann: die Optionen **--since** und **--until**. Beide Optionen benötigen ein Zeitargument im Format "JJJJ-MM-TT hh:mm:ss" (die Anführungszeichen sind erforderlich, um den Platz in der Option freizuhalten). Wenn das Datum weggelassen wird, wird für den Befehl das aktuelle Datum angenommen; wenn die Uhrzeit weggelassen wird, wird angenommen, dass der ganze Tag ab 00:00:00 gemeint ist. Für beide Optionen sind neben dem Datum und der Uhrzeit auch die Argumente **yesterday, today** und **tomorrow** gültig.

Führen Sie den folgenden **journalctl**-Befehl aus, um alle Journaleinträge aus den heutigen Aufzeichnungen aufzulisten.

```
[root@host ~]# journalctl --since today
-- Logs begin at Wed 2019-02-20 16:01:17 +07, end at Thu 2019-02-21 18:31:14 +07.
--
...output omitted...
Feb 21 18:22:44 host.lab.example.com systemd-logind[708]: New session 21 of user
root.
Feb 21 18:22:44 host.lab.example.com systemd[1]: Started Session 21 of user root.
Feb 21 18:22:44 host.lab.example.com sshd[24499]: pam_unix(sshd:session): session
opened for user root by (uid=0)
Feb 21 18:31:13 host.lab.example.com systemd[1]: Starting dnf makecache...
Feb 21 18:31:14 host.lab.example.com dnf[24533]: Red Hat Enterprise Linux 8.0
AppStream (dvd) 637 kB/s | 2.8 kB 00:00
Feb 21 18:31:14 host.lab.example.com dnf[24533]: Red Hat Enterprise Linux 8.0
BaseOS (dvd) 795 kB/s | 2.7 kB 00:00
Feb 21 18:31:14 host.lab.example.com dnf[24533]: Metadata cache created.
Feb 21 18:31:14 host.lab.example.com systemd[1]: Started dnf makecache.
lines 533-569/569 (END) q
```

Führen Sie den folgenden Befehl **journalctl** aus, um alle Journaleinträge aufzulisten, die zwischen **10.02.2019 20:30:00** und **13.02.2019 12:00:00** liegen.

```
[root@host ~]# journalctl --since "2019-02-10 20:30:00" \
--until "2019-02-13 12:00:00"
...output omitted...
```

Sie können auch alle Einträge ab einer Zeit relativ zum aktuellen Zeitpunkt festlegen. Um beispielsweise alle Einträge in der letzten Stunde anzugeben, können Sie den folgenden Befehl verwenden:

```
[root@host ~]# journalctl --since "-1 hour"
...output omitted...
```



### Anmerkung

Sie können andere, anspruchsvollere Zeitangaben mit den Optionen **--since** und **--until** angeben. Einige Beispiele finden Sie auf der Manpage **systemd.time(7)**.

Neben dem sichtbaren Inhalt des Journals sind an die Protokolleinträge Felder angehängt, die nur angezeigt werden, wenn eine ausführliche Ausgabe aktiviert ist. Jedes der angezeigten zusätzlichen Felder kann verwendet werden, um die Ausgabe eines Journaleintrags zu filtern. Dies ist nützlich, um die Ausgabe komplexer Suchen nach bestimmten Ereignissen im Journal einzuschränken.

```
[root@host ~]# journalctl -o verbose
-- Logs begin at Wed 2019-02-20 16:01:17 +07, end at Thu 2019-02-21 18:31:14 +07.
--
...output omitted...
Thu 2019-02-21 18:31:14.509128 +07...
    _BOOT_ID=4409bbf54680496d94e090de9e4a9e23
```

```
_MACHINE_ID=73ab164e278e48be9bf80e80714a8cd5
SYSLOG_FACILITY=3
SYSLOG_IDENTIFIER=systemd
_UID=0
_GID=0
CODE_FILE=../src/core/job.c
CODE_LINE=826
CODE_FUNC=job_log_status_message
JOB_TYPE=start
JOB_RESULT=done
MESSAGE_ID=39f53479d3a045ac8e11786248231fbf
_TRANSPORT=journal
_PID=1
_COMM=systemd
_EXE=/usr/lib/systemd/systemd
_CMDLINE=/usr/lib/systemd/systemd --switched-root --system --deserialize 18
_CAP_EFFECTIVE=3fffffff
_SELINUX_CONTEXT=system_u:system_r:init_t:s0
_SYSTEMD_CGROUP=/init.scope
_SYSTEMD_UNIT=init.scope
_SYSTEMD_SLICE=--.slice
UNIT=dnf-makecache.service
MESSAGE=Started dnf makecache.
_HOSTNAME=host.lab.example.com
INVOCATION_ID=d6f90184663f4309835a3e8ab647cb0e
_SOURCE_REALTIME_TIMESTAMP=1550748674509128
lines 32239-32275/32275 (END) q
```

In der folgenden Liste sind die gemeinsamen Felder des Systemjournals aufgeführt, in denen nach Zeilen gesucht werden kann, die für einen bestimmten Prozess oder ein bestimmtes Ereignis relevant sind.

- `_COMM` ist der Name des Befehls
- `_EXE` ist der Pfad zu der ausführbaren Datei für den Prozess
- `_PID` ist die PID des Prozesses
- `_UID` ist die UID des Benutzers, der den Prozess ausführt
- `_SYSTEMD_UNIT` ist die systemd-Unit, die den Prozess gestartet hat

Mehrere der Systemjournalfelder können zu einer granularen Suchabfrage mit dem Befehl `journalctl` kombiniert werden. Zum Beispiel zeigt folgender `journalctl`-Befehl alle Journaleinträge an, die sich auf die `sshd.service` `systemd`-Unit aus einem Prozess mit der PID 1182 beziehen.

```
[root@host ~]# journalctl _SYSTEMD_UNIT=sshd.service _PID=1182
Apr 03 19:34:27 host.lab.example.com sshd[1182]: Accepted password for root
from ::1 port 52778 ssh2
Apr 03 19:34:28 host.lab.example.com sshd[1182]: pam_unix(sshd:session): session
opened for user root by (uid=0)
...output omitted...
```



### Anmerkung

Eine Liste mit häufig verwendeten Journalfeldern finden Sie auf der Manpage `systemd.journal-fields(7)`.



### Literaturhinweise

Manpage **journalctl(1)**, **systemd.journal-fields(7)** und **systemd.time(7)**

Weitere Informationen finden Sie im Abschnitt *Troubleshooting problems using log files* im Handbuch *Red Hat Enterprise Linux 8 Configuring basic system settings* unter

[https://access.redhat.com/documentation/en-us/red\\_hat\\_enterprise\\_linux/8/html-single/configuring\\_basic\\_system\\_settings/index#troubleshooting-problems-using-log-files\\_getting-started-with-system-administration](https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux/8/html-single/configuring_basic_system_settings/index#troubleshooting-problems-using-log-files_getting-started-with-system-administration)

## ► Angeleitete Übung

# Überprüfen von Systemjournal-Einträgen

In dieser Übung durchsuchen Sie das Systemjournal nach Einträgen, die Ereignisse aufzeichnen, die bestimmten Kriterien entsprechen.

## Ergebnisse

Sie sollten in der Lage sein, das Systemjournal nach Einträgen zu durchsuchen, die Ereignisse nach verschiedenen Kriterien aufzeichnen.

## Bevor Sie Beginnen

Melden Sie sich als Benutzer **student** mit dem Passwort **student** bei **workstation** an.

Führen Sie auf **workstation** das Skript **lab log-query start** aus, um diese Übung zu beginnen. Das Skript stellt sicher, dass die Umgebung richtig eingerichtet ist.

```
[student@workstation ~]$ lab log-query start
```

- 1. Öffnen Sie auf **workstation** als **student** eine SSH-Sitzung zu **servera**.

```
[student@workstation ~]$ ssh student@servera
...output omitted...
[student@servera ~]$
```

- 2. Verwenden Sie die Übereinstimmung **\_PID=1** mit dem **journalctl**-Befehl, um nur Protokollereignisse anzuzeigen, die im Prozess **systemd** mit der Prozesskennung 1 auf **servera** laufen. Um **journalctl** zu beenden, drücken Sie **q**.

```
[student@servera ~]$ journalctl _PID=1
...output omitted...
Feb 13 13:21:08 localhost systemd[1]: Found device /dev/disk/by-uuid/
cdf61ded-534c-4bd6-b458-cab18b1a72ea.
Feb 13 13:21:08 localhost systemd[1]: Started dracut initqueue hook.
Feb 13 13:21:08 localhost systemd[1]: Found device /dev/disk/by-
uuid/44330f15-2f9d-4745-ae2e-20844f22762d.
Feb 13 13:21:08 localhost systemd[1]: Reached target Initrd Root Device.
lines 1-5/5 (END) q
[student@servera ~]$
```



### Anmerkung

Der Befehl **journalctl** kann auf Ihrem System eine andere Ausgabe erzeugen.

- 3. Verwenden Sie die Übereinstimmung **\_UID=81** mit dem Befehl **journalctl**, um alle Protokollereignisse anzuzeigen, die von einem Systemservice stammen, der mit der

**Kapitel 11 |** Analysieren und Speichern von Protokollen

Prozesskennung 81 auf **servera** gestartet wurde. Um **journalctl** zu beenden, drücken Sie **q**.

```
[student@servera ~]$ journalctl _UID=81
...output omitted...
Feb 22 01:29:09 servera.lab.example.com dbus-daemon[672]: [system] Activating via
systemd: service name='org.freedesktop.nm_dispatcher'
Feb 22 01:29:09 servera.lab.example.com dbus-daemon[672]: [system] Successfully
activated service 'org.freedesktop.nm_dispatcher'
lines 1-5/5 (END) q
[student@servera ~]$
```

- 4. Verwenden Sie die Option **-p warning** mit dem Befehl **journalctl** zum Anzeigen von Protokollereignissen mit Priorität **warning** und höher auf **servera**. Um **journalctl** zu beenden, drücken Sie **q**.

```
[student@servera ~]$ journalctl -p warning
...output omitted...
Feb 13 13:21:07 localhost kernel: Detected CPU family 6 model 13 stepping 3
Feb 13 13:21:07 localhost kernel: Warning: Intel Processor - this hardware has not
undergone testing by Red Hat and might not >
Feb 13 13:21:07 localhost kernel: acpi PNP0A03:00: fail to add MMCONFIG
information, can't access extended PCI configuration s>
Feb 13 13:21:07 localhost rpc.statd[288]: Running as root. chown /var/lib/nfs/
statd to choose different user
Feb 13 13:21:07 localhost rpc.idmapd[293]: Setting log level to 0
...output omitted...
Feb 13 13:21:13 servera.lab.example.com rsyslogd[1172]: environment variable TZ is
not set, auto correcting this to TZ=/etc/lo>
Feb 13 14:51:42 servera.lab.example.com systemd[1]: cgroup compatibility
translation between legacy and unified hierarchy sett>
Feb 13 17:15:37 servera.lab.example.com rsyslogd[25176]: environment variable TZ
is not set, auto correcting this to TZ=/etc/l>
Feb 13 18:22:38 servera.lab.example.com rsyslogd[25410]: environment variable TZ
is not set, auto correcting this to TZ=/etc/l>
Feb 13 18:47:55 servera.lab.example.com rsyslogd[25731]: environment variable TZ
is not set, auto correcting this to TZ=/etc/l>
lines 1-17/17 (END) q
[student@servera ~]$
```

- 5. Zeigen Sie alle Protokollereignisse an, die in den vergangenen 10 Minuten ab der aktuellen Uhrzeit auf **servera** aufgezeichnet wurden.

- 5.1. Verwenden Sie die Option **--since** mit dem Befehl **journalctl**, um alle Protokollereignisse anzuzeigen, die in den vergangenen 10 Minuten auf **servera** aufgezeichnet wurden. Um **journalctl** zu beenden, drücken Sie **q**.

```
[student@servera ~]$ journalctl --since "-10min"
...output omitted...
Feb 13 22:31:01 servera.lab.example.com CROND[25890]: (root) CMD (run-parts /etc/
cron.hourly)
Feb 13 22:31:01 servera.lab.example.com run-parts[25893]: (/etc/cron.hourly)
starting Qanacron
```

```

Feb 13 22:31:01 servera.lab.example.com run-parts[25899]: (/etc/cron.hourly)
  finished @anacron
Feb 13 22:31:41 servera.lab.example.com sshd[25901]: Bad protocol version
  identification 'brain' from 172.25.250.254 port 37450
Feb 13 22:31:42 servera.lab.example.com sshd[25902]: Accepted publickey for root
  from 172.25.250.254 port 37452 ssh2: RSA SHA256:...
Feb 13 22:31:42 servera.lab.example.com systemd[1]: Started /run/user/0 mount
  wrapper.
Feb 13 22:31:42 servera.lab.example.com systemd[1]: Created slice User Slice of
  UID 0.
Feb 13 22:31:42 servera.lab.example.com systemd[1]: Starting User Manager for UID
  0...
Feb 13 22:31:42 servera.lab.example.com systemd[1]: Started Session 118 of user
  root.
Feb 13 22:31:42 servera.lab.example.com systemd-logind[712]: New session 118 of
  user root.
Feb 13 22:31:42 servera.lab.example.com systemd[25906]: pam_unix(systemd-
  user:session): session opened for user root by (uid=0)
  ...output omitted...
lines 1-32/84 39% q
[student@servera ~]$
```

- 6. Verwenden Sie die Option **--since** und die Übereinstimmung **\_SYSTEMD\_UNIT="sshd.service"** mit dem Befehl **journalctl**, um alle Protokollereignisse anzuzeigen, die vom Service **sshd** stammen und seit **09:00:00** heute Morgen auf **servera** aufgezeichnet wurden. Um **journalctl** zu beenden, drücken Sie **q**.



### Anmerkung

Sie befinden sich möglicherweise in derselben Zeitzone wie Ihr Kursraum.  
Überprüfen Sie die Zeit auf **servera** und passen Sie den **--since**-Wert bei Bedarf entsprechend an.

```

[student@servera ~]$ journalctl --since 9:00:00 _SYSTEMD_UNIT="sshd.service"
  ...output omitted...
Feb 13 13:21:12 servera.lab.example.com sshd[727]: Server listening on 0.0.0.0
  port 22.
Feb 13 13:21:12 servera.lab.example.com sshd[727]: Server listening on :: port 22.
Feb 13 13:22:07 servera.lab.example.com sshd[1238]: Accepted publickey for student
  from 172.25.250.250 port 50590 ssh2: RSA SHA256:...
Feb 13 13:22:07 servera.lab.example.com sshd[1238]: pam_unix(sshd:session):
  session opened for user student by (uid=0)
Feb 13 13:22:08 servera.lab.example.com sshd[1238]: pam_unix(sshd:session):
  session closed for user student
Feb 13 13:25:47 servera.lab.example.com sshd[1289]: Accepted publickey for root
  from 172.25.250.254 port 37194 ssh2: RSA SHA256:...
Feb 13 13:25:47 servera.lab.example.com sshd[1289]: pam_unix(sshd:session):
  session opened for user root by (uid=0)
Feb 13 13:25:47 servera.lab.example.com sshd[1289]: pam_unix(sshd:session):
  session closed for user root
Feb 13 13:25:48 servera.lab.example.com sshd[1316]: Accepted publickey for root
  from 172.25.250.254 port 37196 ssh2: RSA SHA256:...
```

```
Feb 13 13:25:48 servera.lab.example.com sshd[1316]: pam_unix(sshd:session):  
    session opened for user root by (uid=0)  
Feb 13 13:25:48 servera.lab.example.com sshd[1316]: pam_unix(sshd:session):  
    session closed for user root  
Feb 13 13:26:07 servera.lab.example.com sshd[1355]: Accepted publickey for student  
    from 172.25.250.254 port 37198 ssh2: RSA SH>  
Feb 13 13:26:07 servera.lab.example.com sshd[1355]: pam_unix(sshd:session):  
    session opened for user student by (uid=0)  
Feb 13 13:52:28 servera.lab.example.com sshd[1473]: Accepted publickey for root  
    from 172.25.250.254 port 37218 ssh2: RSA SHA25>  
Feb 13 13:52:28 servera.lab.example.com sshd[1473]: pam_unix(sshd:session):  
    session opened for user root by (uid=0)  
...output omitted...  
lines 1-32 q  
[student@servera ~]$
```

- 7. Melden Sie sich von **servera** ab.

```
[student@servera ~]$ exit  
logout  
Connection to servera closed.  
[student@workstation ~]$
```

## Beenden

Führen Sie auf **workstation** das Skript **lab log-query finish** aus, um diese Übung zu beenden. Das Skript stellt sicher, dass die Umgebung als eine bereinigte Übungsumgebung eingerichtet wird.

```
[student@workstation ~]$ lab log-query finish
```

Hiermit ist die angeleitete Übung beendet.

# Das Systemjournal bewahren

## Ziele

Nach Abschluss dieses Abschnitts sind Sie in der Lage, das Systemjournal so zu konfigurieren, dass die Aufzeichnung von Ereignissen beim Bootvorgang eines Servers erhalten bleibt.

## Dauerhaftes Speichern des Systemjournals

Standardmäßig wird das Systemjournal in Verzeichnis `/run/log/journal` aufbewahrt. Das bedeutet, die Journale werden bei einem Bootvorgang des Systems gelöscht. Sie können die Konfigurationseinstellungen des Services `systemd-journald` in der `/etc/systemd/journald.conf`-Datei ändern, damit die Journale auch nach dem Booten bestehen bleiben.

Der Parameter **Storage** in der Datei `/etc/systemd/journald.conf` definiert, ob Systemjournale persistent oder volatil beim Booten gespeichert werden sollen. Legen Sie diesen Parameter wie folgt auf **persistent**, **volatile**, **auto** oder **none** fest:

- **persistent**: Speichert Journale im Verzeichnis `/var/log/journal`, das nach dem Booten bestehen bleibt.  
Wenn das Verzeichnis `/var/log/journal` nicht existiert, erstellt es der Service `systemd-journald`.
- **volatile**: Speichert Journale vorübergehend im Verzeichnis `/run/log/journal`.  
Da das Dateisystem `/run` temporär ist und nur im Laufzeitspeicher vorhanden ist, bleiben darin gespeicherte Daten, einschließlich Systemjournalen, nach dem Booten nicht erhalten.

- **auto**: Wenn das Verzeichnis `/var/log/journal` existiert, dann verwendet `systemd-journald` einen permanenten Speicher, andernfalls wird die Speicherplatznutzung begrenzt.

Dies ist die Standardaktion, wenn der Parameter **Storage** nicht festgelegt ist.

- **none**: Kein Storage verwenden. Alle Protokolle werden gelöscht, die Protokollweiterleitung funktioniert jedoch weiterhin erwartungsgemäß.

Der Vorteil eines persistent gespeicherten Systemjournals ist, dass die historischen Daten beim Booten sofort verfügbar sind. Jedoch werden auch bei einem persistenten Journal nicht alle Daten für immer aufbewahrt. Das Journal verfügt über einen integrierten Protokoll-Rotationsmechanismus, der monatlich ausgelöst wird. Darüber hinaus ist standardmäßig festgelegt, dass das Journal nicht größer als 10 % des Dateisystems, in dem es sich befindet, werden darf oder nur so viel Speicherplatz belegen darf, dass mehr als 15 % des Dateisystems frei bleiben. Diese Werte können sowohl für die Laufzeit als auch für persistente Journale in `/etc/systemd/journald.conf` angepasst werden. Die aktuellen Grenzen der Journalgröße werden protokolliert, wenn der Prozess `systemd-journald` beginnt. Die folgende Befehlsausgabe zeigt die Journaleinträge, welche die aktuellen Größenbeschränkungen widerspiegeln:

```
[user@host ~]$ journalctl | grep -E 'Runtime|System journal'
Feb 25 13:01:46 localhost systemd-journald[147]: Runtime journal (/run/log/
journal/ae06db7da89142138408d77efea9229c) is 8.0M, max 91.4M, 83.4M free.
Feb 25 13:01:48 remotehost.lab.example.com systemd-journald[548]: Runtime journal
(/run/log/journal/73ab164e278e48be9bf80e80714a8cd5) is 8.0M, max 91.4M, 83.4M
free.
Feb 25 13:01:48 remotehost.lab.example.com systemd-journald[548]: System journal
(/var/log/journal/73ab164e278e48be9bf80e80714a8cd5) is 8.0M, max 3.7G, 3.7G free.
Feb 25 13:01:48 remotehost.lab.example.com systemd[1]: Starting Tell Plymouth To
Write Out Runtime Data...
Feb 25 13:01:48 remotehost.lab.example.com systemd[1]: Started Tell Plymouth To
Write Out Runtime Data.
```



### Anmerkung

In **grep** oben dient das Pipe-Symbol (**|**) als ein oder-Operator. Daher findet **grep** jede Zeile, die entweder die Zeichenfolge **Runtime** oder **System journal** in der Ausgabe **journalctl** enthält. Dadurch werden die aktuellen Größengrenzen für die volatile (**Runtime**) Journal-Speicherung sowie die persistente (**System**) Journal-Speicherung abgerufen.

## Konfigurieren persistenter Systemjournale

Um den Service **systemd-journald** so zu konfigurieren, dass Systemjournale nach dem Booten bewahrt werden, legen Sie **Storage** in der Datei **/etc/systemd/journald.conf** auf **persistent** fest. Führen Sie den Texteditor Ihrer Wahl als Superuser aus, um die Datei **/etc/systemd/journald.conf** zu bearbeiten.

```
[Journal]
Storage=persistent
...output omitted...
```

Starten Sie nach der Bearbeitung der Konfigurationsdatei den Service **systemd-journald** neu, damit die Konfigurationsänderungen wirksam werden.

```
[root@host ~]# systemctl restart systemd-journald
```

Wenn der Service **systemd-journald** erfolgreich neu gestartet wurde, können Sie sehen, dass das Verzeichnis **/var/log/journal** erstellt wurde und eines oder mehrere Unterverzeichnisse enthält. Diese Unterverzeichnisse enthalten hexadezimale Zeichen in ihren Langnamen und enthalten **\*.journal**-Dateien. Die **\*.journal**-Dateien sind die Binärdateien, in denen die strukturierten und indizierten Journaleinträge gespeichert werden.

```
[root@host ~]# ls /var/log/journal
73ab164e278e48be9bf80e80714a8cd5
[root@host ~]# ls /var/log/journal/73ab164e278e48be9bf80e80714a8cd5
system.journal user-1000.journal
```

Wenn die Systemjournale auch nach dem Booten bestehen bleiben, erhalten Sie eine umfangreiche Anzahl von Einträgen in der Ausgabe des Befehls **journalctl**, der Einträge aus dem aktuellen Bootvorgang des Systems sowie aus vorherigen enthält. Um die Ausgabe auf einen

bestimmten Bootvorgang des Systems zu beschränken, verwenden Sie die Option **-b** mit dem Befehl **journalctl**. Folgender Befehl **journalctl** ruft die Einträge ab, die auf den ersten Bootvorgang des Systems beschränkt sind:

```
[root@host ~]# journalctl -b 1  
...output omitted...
```

Folgender Befehl **journalctl** ruft die Einträge ab, die auf den zweiten Bootvorgang des Systems beschränkt sind: Das folgende Argument ist nur dann von Bedeutung, wenn das System mindestens zweimal neu gebootet wurde:

```
[root@host ~]# journalctl -b 2
```

Der folgende Befehl **journalctl** ruft die Einträge ab, die auf den aktuellen Bootvorgang des Systems beschränkt sind:

```
[root@host ~]# journalctl -b
```



### Anmerkung

Wenn Sie ein persistentes Journal für die Fehlerbehebung bei einem Systemabsturz verwenden, muss die Journalabfrage gewöhnlich auf den vor dem Absturz durchgeführten Bootvorgang begrenzt werden. Der Option **-b** kann eine negative Zahl hinzugefügt werden, die angeibt, wie viele vorherige Bootvorgänge des Systems die Ausgabe enthalten soll. Beispielsweise wird die Ausgabe mit **journalctl -b -1** auf den vorherigen Bootvorgang begrenzt.



### Literaturhinweise

Manpages **systemd-journald.conf(5)**, **systemd-journald(8)**

Weitere Informationen finden Sie im Abschnitt *Troubleshooting problems using log files* im Handbuch *Red Hat Enterprise Linux 8 Configuring basic system settings* unter

[https://access.redhat.com/documentation/en-us/red\\_hat\\_enterprise\\_linux/8/html-single/configuring\\_basic\\_system\\_settings/index#troubleshooting-problems-using-log-files\\_getting-started-with-system-administration](https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux/8/html-single/configuring_basic_system_settings/index#troubleshooting-problems-using-log-files_getting-started-with-system-administration)

## ► Angeleitete Übung

# Das Systemjournal bewahren

In dieser Übung konfigurieren Sie das Systemjournal so, dass seine Daten nach einem Bootvorgang erhalten bleiben.

## Ergebnisse

Sie sollten nun das Systemjournal so konfigurieren können, dass seine Daten nach einem Bootvorgang erhalten bleiben.

## Bevor Sie Beginnen

Melden Sie sich als Benutzer **student** mit dem Passwort **student** bei **workstation** an.

Führen Sie auf **workstation** das Skript **lab log-preserve start** aus, um diese Übung zu beginnen. Durch das Skript wird gewährleistet, dass die Umgebung richtig eingerichtet ist.

```
[student@workstation ~]$ lab log-preserve start
```

- 1. Öffnen Sie auf **workstation** als **student** eine SSH-Sitzung zu **servera**.

```
[student@workstation ~]$ ssh student@servera  
...output omitted...  
[student@servera ~]$
```

- 2. Vergewissern Sie sich als Superuser, dass das Verzeichnis **/var/log/journal** nicht vorhanden ist. Listen Sie mit dem Befehl **ls** den Inhalt des Verzeichnisses **/var/log/journal** auf. Erhöhen Sie mit **sudo** die Berechtigungen des Benutzers **student**. Verwenden Sie **student** bei Anforderung als Passwort.

```
[student@servera ~]$ sudo ls /var/log/journal  
[sudo] password for student: student  
ls: cannot access '/var/log/journal': No such file or directory
```

Da das Verzeichnis **/var/log/journal** nicht vorhanden ist, bewahrt der Service **systemd-journald** seine Protokolldaten nicht.

- 3. Konfigurieren Sie den Service **systemd-journald** auf **servera** so, dass seine Journale nach einem Bootvorgang bestehen bleiben.

- 3.1. Heben Sie die Auskommentierung der Zeile **Storage=auto** in der Datei **/etc/systemd/journald.conf** auf und legen Sie **Storage** auf **persistent** fest. Sie können den Befehl **sudo vim /etc/systemd/journald.conf** verwenden, um die Konfigurationsdatei anzupassen. Geben Sie **/Storage=auto** im **vim**-Befehlsmodus ein, um nach der Zeile **Storage=auto** zu suchen.

```
...output omitted...
[Journal]
Storage=persistent
...output omitted...
```

- 3.2. Starten Sie mit dem Befehl **systemctl** den Service **systemd-journald** neu, damit die Konfigurationsänderungen wirksam werden.

```
[student@servera ~]$ sudo systemctl restart systemd-journald.service
```

- 4. Überprüfen Sie, ob der Service **systemd-journald** auf **servera** seine Journale auch nach Bootvorgängen bewahrt.

- 4.1. Starten Sie mit dem Befehl **systemctl reboot servera** neu.

```
[student@servera ~]$ sudo systemctl reboot
Connection to servera closed by remote host.
Connection to servera closed.
[student@workstation ~]$
```

Beachten Sie, dass die SSH-Verbindung beendet wurde, sobald Sie das System **servera** neu gestartet haben.

- 4.2. Öffnen Sie eine SSH-Sitzung zu **servera**.

```
[student@workstation ~]$ ssh student@servera
...output omitted...
[student@servera ~]$
```

- 4.3. Verwenden Sie den Befehl **ls** zur Bestätigung, dass das Verzeichnis **/var/log/journal** existiert. Das Verzeichnis **/var/log/journal** enthält ein Unterverzeichnis mit einem langen hexadezimalen Namen. Die Journaldateien befinden sich in diesem Verzeichnis. Der Name des Unterverzeichnisses kann auf Ihrem System abweichen.

```
[student@servera ~]$ sudo ls /var/log/journal
[sudo] password for student: student
73ab164e278e48be9bf80e80714a8cd5
[student@servera ~]$ sudo ls \
/var/log/journal/73ab164e278e48be9bf80e80714a8cd5
system.journal user-1000.journal
```

- 4.4. Melden Sie sich von **servera** ab.

```
[student@servera ~]$ exit
logout
Connection to servera closed.
```

## Beenden

Führen Sie auf **workstation** das Skript **lab log-preserve finish** aus, um diese Übung zu beenden. Das Skript stellt sicher, dass die Umgebung als eine bereinigte Übungsumgebung eingerichtet wird.

```
[student@workstation ~]$ lab log-preserve finish
```

Hiermit ist die angeleitete Übung beendet.

# Verwalten der genauen Uhrzeit

---

## Ziele

Nach Abschluss dieses Abschnitts sollten in der Lage sein, die genaue Zeitsynchronisierung mithilfe von NTP beizubehalten und Zeitzonen zu konfigurieren, um korrekte Zeitstempel für Ereignisse zu gewährleisten, die vom Systemjournal und den Protokollen aufgezeichnet werden.

## Einrichten von lokalen Uhren und Zeitzonen

Für die systemübergreifende Analyse von Protokolldateien ist eine genaue und synchronisierte Systemzeit unerlässlich. Das *Network Time Protocol (NTP)* ist ein Standard für Systeme zur Synchronisierung mit Zeitangaben aus dem Internet. Systeme können genaue Zeitangaben von öffentlichen NTP-Services aus dem Internet wie dem NTP-Pool-Projekt abrufen. Eine andere Möglichkeit besteht darin, die Zeitangaben von hochwertigen Hardwareuhren für lokale Clients zum Abruf bereitzustellen.

Mit dem **timedatectl**-Befehl wird eine Übersicht über die aktuellen zeitbezogenen Systemeinstellungen, u. a. die aktuellen Zeit-, Zeitzonen- und NTP-Synchronisierungseinstellungen des Systems, angezeigt.

```
[user@host ~]$ timedatectl
    Local time: Fri 2019-04-05 16:10:29 CDT
    Universal time: Fri 2019-04-05 21:10:29 UTC
          RTC time: Fri 2019-04-05 21:10:29
            Time zone: America/Chicago (CDT, -0500)
      System clock synchronized: yes
        NTP service: active
       RTC in local TZ: no
```

Eine verfügbare Datenbank mit Zeitzonen kann mit dem Befehl **timedatectl list-timezones** aufgelistet werden.

```
[user@host ~]$ timedatectl list-timezones
Africa/Abidjan
Africa/Accra
Africa/Addis_Ababa
Africa/Algiers
Africa/Asmara
Africa/Bamako
...
```

Zeitzonennamen basieren auf der öffentlichen Zeitzonendatenbank, die von IANA verwaltet wird. Zeitzonen werden nach einem Kontinent oder Ozean und dann gewöhnlich – jedoch nicht immer – nach der größten Stadt innerhalb des Zeitzonenbereichs benannt. Zum Beispiel stellt „America/Denver“ den größten Teil der US-Zeitzone „Mountain Time Zone“ dar.

In Fällen, in denen Orte innerhalb einer Zeitzone unterschiedliche Regeln für die Sommerzeit haben, ist die Auswahl des richtigen Namens möglicherweise nicht intuitiv. Zum Beispiel wird in

## Kapitel 11 | Analysieren und Speichern von Protokollen

den Vereinigten Staaten im Großteil des Bundesstaates Arizona (US-Zeitzone „Mountain Time Zone“) keine Anpassung an die Sommerzeit vorgenommen. Dabei handelt es sich um die Zeitzone „America/Phoenix“.

Der Befehl **tzselect** ist nützlich, um die richtigen Zeitzonennamen in der zoneinfo-Datenbank zu identifizieren. Er ist interaktiv und fordert den Benutzer zur Beantwortung von Fragen zum Standort des Systems auf. Daraufhin wird der Name der richtigen Zeitzone ausgegeben. Mit dem Befehl wird keine Änderung an den Zeitzoneinstellungen des Systems vorgenommen.

Der Superuser kann die Systemeinstellung ändern, um die aktuelle Zeitzone mithilfe von Befehl **timedatectl set-timezone** zu aktualisieren. Der folgende **timedatectl**-Befehl aktualisiert die aktuelle Zeitzone auf **America/Phoenix**.

```
[root@host ~]# timedatectl set-timezone America/Phoenix
[root@host ~]# timedatectl
    Local time: Fri 2019-04-05 14:12:39 MST
    Universal time: Fri 2019-04-05 21:12:39 UTC
        RTC time: Fri 2019-04-05 21:12:39
       Time zone: America/Phoenix (MST, -0700)
System clock synchronized: yes
          NTP service: active
     RTC in local TZ: no
```



### Anmerkung

Wenn Sie auf einem bestimmten Server die koordinierte Weltzeit (UTC) verwenden müssen, setzen Sie die Zeitzone auf UTC. Der Befehl **tzselect** beinhaltet nicht den Namen der UTC-Zeitzone. Verwenden Sie den **timedatectl set-timezone UTC** Befehl, um die aktuelle Zeitzone des Systems einzustellen **UTC**.

Verwenden Sie den Befehl **timedatectl set-time**, um die aktuelle Zeitzone des Systems zu ändern. Die Zeit wird im Format „*JJJJ-MM-TT hh:mm:ss*“ angegeben. Das Datum oder die Uhrzeit können weggelassen werden. Der folgende **timedatectl**-Befehl ändert die aktuelle Zeitzone auf **09:00:00**.

```
[root@host ~]# timedatectl set-time 9:00:00
[root@host ~]# timedatectl
    Local time: Fri 2019-04-05 09:00:27 MST
    Universal time: Fri 2019-04-05 16:00:27 UTC
        RTC time: Fri 2019-04-05 16:00:27
       Time zone: America/Phoenix (MST, -0700)
System clock synchronized: yes
          NTP service: active
     RTC in local TZ: no
```

Mit dem Befehl **timedatectl set-ntp** wird die NTP-Synchronisierung für eine automatische Zeitanpassung aktiviert oder deaktiviert. Zur Aktivierung oder Deaktivierung der Option ist das Argument **true** bzw. **false** erforderlich. Der folgende **timedatectl**-Befehl aktiviert die NTP-Synchronisierung.

```
[root@host ~]# timedatectl set-ntp true
```



### Anmerkung

In Red Hat Enterprise Linux 8 passt der Befehl `timedatectl set-ntp` an, ob der NTP-Service **chrony** ausgeführt wird oder nicht. Andere Linux-Distributionen verwenden diese Einstellung möglicherweise, um einen anderen NTP- oder SNTP-Service anzupassen.

Durch Aktivieren oder Deaktivieren von NTP mithilfe anderer Dienstprogramme in Red Hat Enterprise Linux, wie in der grafischen GNOME-Anwendung Settings, wird auch diese Einstellung aktualisiert.

## Konfigurieren und Überwachen von Chrony

Durch den **chrony**-Service bleibt die für gewöhnlich nicht korrekte, lokale Hardwareuhr (RTC) aktuell, indem sie mit den konfigurierten NTP-Servern synchronisiert wird. Wenn keine Netzwerkverbindung verfügbar ist, berechnet **chrony** die RTC-Uhrendrift, die in **driftfile** in der angegebenen Konfigurationsdatei `/etc/chrony.conf` aufgezeichnet wird.

Standardmäßig verwendet der **chrony**-Service Server aus dem NTP-Pool-Projekt für die Zeit-Synchronisierung und muss nicht zusätzlich konfiguriert werden. Es kann nützlich sein, die NTP-Server zu wechseln, wenn sich das fragliche System in einem isolierten Netzwerk befindet.

Das **Stratum** der NTP-Zeitquelle bestimmt die Qualität. Das Stratum bestimmt die Anzahl der Hops zwischen einem System und einer leistungsstarken Referenzuhr. Die Referenzuhr ist ein **stratum 0**-Zeitgeber. Ein direkt angeschlossener NTP-Server ist ein **stratum 1**, und ein System, das die Uhrzeit mit dem NTP-Server synchronisiert, ist ein **stratum 2**-Zeitgeber.

Es gibt zwei Gruppen von Zeitgebern in der Konfigurationsdatei `/etc/chrony.conf`: **server** und **peer**. Der Server befindet sich ein Stratum oberhalb des lokalen NTP-Servers, während sich der Peer auf derselben Stratum-Ebene befindet. Sie können mehr als einen server und mehr als einen peer angeben, einen pro Zeile.

Das erste Argument in der **server**-Zeile gibt die IP-Adresse oder den DNS-Namen des NTP-Servers an. Nach der IP-Adresse oder dem Namen des Servers können Sie eine Reihe von Optionen für den Server auflisten. Es wird empfohlen, die Option **iburst** zu verwenden, da nach dem Start des Service vier Messungen in einem kurzen Zeitraum erfolgen, um eine genauere anfängliche Uhrsynchronisierung zu ermöglichen.

Die folgende **server classroom.example.com iburst**-Zeile in der `/etc/chrony.conf`-Datei verursacht, dass der **chrony**-Service `classroom.example.com` als NTP-Zeitquelle verwendet.

```
# Use public servers from the pool.ntp.org project.  
...output omitted...  
server classroom.example.com iburst  
...output omitted...
```

Nachdem Sie **chrony** auf den lokalen Zeitgeber, `classroom.example.com`, verwiesen haben, sollten Sie den Service neu starten.

```
[root@host ~]# systemctl restart chrony
```

Der **chronyc**-Befehl funktioniert wie ein Client des **chrony**-Service. Nach dem Einrichten der NTP-Synchronisierung sollten Sie sicherstellen, dass das lokale System den NTP-Server

nahtlos verwendet, um die Systemuhr mithilfe des Befehls **chronyc sources** zu synchronisieren. Verwenden Sie den Befehl **chronyc sources -v**, um zusätzlichen Erläuterungen zu der Ausgabe zu erhalten.

```
[root@host ~]# chronyc sources -v
210 Number of sources = 1

    .-- Source mode  '^' = server, '=' = peer, '#' = local clock.
    / .- Source state '*' = current synced, '+' = combined , '-' = not combined,
    | /   '?' = unreachable, 'x' = time may be in error, '~' = time too variable.
    ||                               .- xxxx [ yyyy ] +/- zzzz
    ||                               / xxxx = adjusted offset,
    ||           Log2(Polling interval) .-          | yyyy = measured offset,
    ||                               \          | zzzz = estimated error.
    ||                               |          |
MS Name/IP address      Stratum Poll Reach LastRx Last sample
=====
^* classroom.example.com      8     6    17    23   -497ns[-7000ns] +/-  956us
```

Das Zeichen \* im Feld **S** (Geberstatus) gibt an, dass der **classroom.example.com**-Server als Zeitgeber verwendet wurde, und dass es sich dabei um den NTP-Server handelt, mit dem das System derzeit synchronisiert ist.



#### Literaturhinweise

Manpages **timedatectl(1)**, **tzselect(8)**, **chronyd(8)**, **chrony.conf(5)** und **chronyc(1)**

#### NTP-Pool-Projekt

<http://www.pool.ntp.org/>

#### Zeitzonendatenbank

<http://www.iana.org/time-zones>

## ► Angeleitete Übung

# Verwalten der genauen Uhrzeit

In dieser Übung passen Sie die Zeitzone auf einem Server an und stellen sicher, dass die Systemuhr mit einer NTP-Zeitquelle synchronisiert wird.

## Ergebnisse

Es werden folgende Fähigkeiten vermittelt:

- Ändern Sie die Zeitzone auf einem Server.
- Konfigurieren Sie den Server so, dass seine Uhrzeit mit einer NTP-Zeitquelle synchronisiert wird.

## Bevor Sie Beginnen

Melden Sie sich als Benutzer **student** mit dem Passwort **student** bei **workstation** an.

Führen Sie auf **workstation** das Skript **lab log-maintain start** aus, um diese Übung zu beginnen. Dieses Skript stellt sicher, dass die Zeitsynchronisierung auf **servera** deaktiviert ist, um Ihnen die Möglichkeit zu geben, die Einstellungen im System manuell zu aktualisieren und die Zeitsynchronisation zu aktivieren.

```
[student@workstation ~]$ lab log-maintain start
```

- 1. Öffnen Sie auf **workstation** als **student** eine SSH-Sitzung zu **servera**.

```
[student@workstation ~]$ ssh student@servera
...output omitted...
[student@servera ~]$
```

- 2. In diesem Beispiel nehmen wir an, dass das System **servera** nach Haiti verlagert wird, daher müssen Sie die Zeitzone entsprechend aktualisieren. Nutzen Sie **sudo**, um die Rechte des **student**-Benutzers während der Ausführung des **timedatectl**-Befehls zum Aktualisieren der Zeitzone zu erhöhen. Verwenden Sie **student** bei Anforderung als Passwort.
- 2.1. Verwenden Sie den Befehl **tzselect**, um die entsprechende Zeitzone für Haiti zu bestimmen.

```
[student@servera ~]$ tzselect
Please identify a location so that time zone rules can be set correctly.
Please select a continent, ocean, "coord", or "TZ".
1) Africa
2) Americas
3) Antarctica
4) Asia
5) Atlantic Ocean
```

```
6) Australia
7) Europe
8) Indian Ocean
9) Pacific Ocean
10) coord - I want to use geographical coordinates.
11) TZ - I want to specify the time zone using the Posix TZ format.
#? 2
Please select a country whose clocks agree with yours.
1) Anguilla          19) Dominican Republic    37) Peru
2) Antigua & Barbuda 20) Ecuador           38) Puerto Rico
3) Argentina         21) El Salvador        39) St Barthelemy
4) Aruba             22) French Guiana      40) St Kitts & Nevis
5) Bahamas           23) Greenland         41) St Lucia
6) Barbados          24) Grenada           42) St Maarten (Dutch)
7) Belize             25) Guadeloupe        43) St Martin (French)
8) Bolivia            26) Guatemala         44) St Pierre & Miquelon
9) Brazil             27) Guyana            45) St Vincent
10) Canada            28) Haiti              46) Suriname
11) Caribbean NL     29) Honduras          47) Trinidad & Tobago
12) Cayman Islands   30) Jamaica           48) Turks & Caicos Is
13) Chile              31) Martinique        49) United States
14) Colombia          32) Mexico             50) Uruguay
15) Costa Rica        33) Montserrat       51) Venezuela
16) Cuba               34) Nicaragua         52) Virgin Islands (UK)
17) Curaçao           35) Panama            53) Virgin Islands (US)
18) Dominica          36) Paraguay
#? 28
The following information has been given:
```

Haiti

Therefore TZ='America/Port-au-Prince' will be used.

Selected time is now: Tue Feb 19 00:51:05 EST 2019.

Universal Time is now: Tue Feb 19 05:51:05 UTC 2019.

Is the above information OK?

- 1) Yes
  - 2) No
- #? 1

You can make this change permanent for yourself by appending the line

TZ='America/Port-au-Prince'; export TZ  
to the file '.profile' in your home directory; then log out and log in again.

Here is that TZ value again, this time on standard output so that you  
can use the /usr/bin/tzselect command in shell scripts:

America/Port-au-Prince

Beachten Sie, dass der vorangehende Befehl **tzselect** die entsprechende Zeitzone  
für Haiti anzeigen.

- 2.2. Verwenden Sie den Befehl **timedatectl**, um die Zeitzone auf **servera** auf  
**America/Port-au-Prince** zu aktualisieren.

```
[student@servera ~]$ sudo timedatectl set-timezone \
America/Port-au-Prince
[sudo] password for student: student
```

- 2.3. Verwenden Sie den Befehl **timedatectl**, um zu überprüfen, ob die Zeitzone auf **America/Port-au-Prince** aktualisiert wurde.

```
[student@servera ~]$ timedatectl
    Local time: Tue 2019-02-19 01:16:29 EST
    Universal time: Tue 2019-02-19 06:16:29 UTC
          RTC time: Tue 2019-02-19 06:16:29
             Time zone: America/Port-au-Prince (EST, -0500)
System clock synchronized: no
          NTP service: inactive
        RTC in local TZ: no
```

- 3. Konfigurieren Sie den **chronyd**-Service auf **servera** für die Synchronisierung der Systemzeit mit der NTP-Zeitzonenquelle **classroom.example.com**.

- 3.1. Bearbeiten Sie die Datei **/etc/chrony.conf** so, dass der Server **classroom.example.com** als NTP-Zeitzquelle festgelegt wird. Sie können den Befehl **sudo vim /etc/chrony.conf** verwenden, um die Konfigurationsdatei anzupassen. Die folgende Ausgabe zeigt die Konfigurationszeile, die Sie in der Konfigurationsdatei ergänzen müssen:

```
...output omitted...
server classroom.example.com iburst
...output omitted...
```

Die vorhergehende Zeile in der Konfigurationsdatei **/etc/chrony.conf** enthält die Option **iburst** zur Beschleunigung der initialen Zeitsynchronisation.

- 3.2. Verwenden Sie den Befehl **timedatectl** zum Einschalten der Zeitsynchronisation auf **servera**.

```
[student@servera ~]$ sudo timedatectl set-ntp yes
```

Der vorangehende Befehl **timedatectl** aktiviert den NTP-Server mit den geänderten Einstellungen in der **/etc/chrony.conf**-Konfigurationsdatei. Der vorangehende Befehl **timedatectl** kann entweder den Service **chronyd** oder **ntpd** aktivieren, basierend darauf, was derzeit auf dem System installiert ist.

- 4. Stellen Sie sicher, dass die Zeiteinstellungen auf **servera** aktuell für die Synchronisierung mit **classroom.example.com** als Zeitquelle in der Unterrichtsumgebung konfiguriert sind.
- 4.1. Verwenden Sie den Befehl **timedatectl**, um zu überprüfen, dass auf **servera** derzeit die Zeitsynchronisation aktiviert ist.

```
[student@servera ~]$ timedatectl
    Local time: Tue 2019-02-19 01:52:17 EST
    Universal time: Tue 2019-02-19 06:52:17 UTC
        RTC time: Tue 2019-02-19 06:52:17
      Time zone: America/Port-au-Prince (EST, -0500)
System clock synchronized: yes
          NTP service: active
    RTC in local TZ: no
```



### Anmerkung

Wenn die vorherige Ausgabe anzeigt, dass die Uhr nicht synchronisiert ist, warten Sie zwei Sekunden und führen Sie den Befehl **timedatectl** erneut aus. Es dauert einige Sekunden, bis die Zeiteinstellungen erfolgreich mit der Zeitquelle synchronisiert sind.

- 4.2. Verwenden Sie den Befehl **chronyc**, um zu überprüfen, ob das System **servera** derzeit seine Zeiteinstellungen mit **classroom.example.com** als Zeitquelle synchronisiert.

```
[student@servera ~]$ chronyc sources -v
210 Number of sources = 1

    .-- Source mode  '^' = server, '=' = peer, '#' = local clock.
    / .. Source state '*' = current synced, '+' = combined , '-' = not combined,
    | /   '?' = unreachable, 'x' = time may be in error, '~' = time too variable.
    ||                               .- xxxx [ yyyy ] +/- zzzz
    ||     Reachability register (octal) -. | xxxx = adjusted offset,
    ||     Log2(Polling interval) --. | yyyy = measured offset,
    ||                           \ | zzzz = estimated error.
    ||                           | |
MS Name/IP address          Stratum Poll Reach LastRx Last sample
=====
^* classroom.example.com      2   6   377   62  +105us[ +143us] +/-  14ms
```

Beachten Sie, dass in der vorhergehenden Ausgabe ein Sternchen (\*) im Feld Quellzustand (**S**) für die NTP-Zeitquelle **classroom.example.com** angezeigt wird. Das Sternchen zeigt an, dass die lokale Systemzeit derzeit erfolgreich mit der NTP-Zeitquelle synchronisiert wird.

- 4.3. Melden Sie sich von **servera** ab.

```
[student@servera ~]$ exit
logout
Connection to servera closed.
[student@workstation ~]$
```

## Beenden

Führen Sie auf **workstation** das Skript **lab log-maintain finish** aus, um diese Übung zu beenden. Dieses Skript stellt sicher, dass die ursprüngliche Zeitzone zusammen mit allen ursprünglichen Zeiteinstellungen auf **servera** wiederhergestellt wird.

```
[student@workstation ~]$ lab log-maintain finish
```

Hiermit ist die angeleitete Übung beendet.

## ► Praktische Übung

# Analysieren und Speichern von Protokollen

### Leistungscheckliste

In dieser Übung ändern Sie die Zeitzone auf einem vorhandenen Server und konfigurieren eine neue Protokolldatei für alle Ereignisse, die sich auf Authentifizierungsfehler beziehen.

### Ergebnisse

Es werden folgende Fähigkeiten vermittelt:

- Aktualisieren Sie die Zeitzone auf einem vorhandenen Server.
- Konfigurieren Sie eine neue Protokolldatei, um alle Nachrichten zu speichern, die sich auf Authentifizierungsfehler beziehen.

### Bevor Sie Beginnen

Melden Sie sich als Benutzer **student** mit dem Passwort **student** bei **workstation** an.

Führen Sie auf **workstation** das Skript **lab log-review start** aus, um diese Übung zu beginnen. Dieses Skript zeichnet die aktuelle Zeitzone vom System **serverb** auf und stellt sicher, dass die Umgebung korrekt eingerichtet ist.

```
[student@workstation ~]$ lab log-review start
```

1. Öffnen Sie auf **workstation** als **student** eine SSH-Sitzung zu **serverb**.
2. Nehmen Sie an, dass das System **serverb** nach Jamaica verlagert wird, daher müssen Sie die Zeitzone entsprechend aktualisieren. Erhöhen Sie mit **sudo** die Berechtigungen des Benutzers **student**, damit der Befehl **timedatectl** die Zeitzone aktualisieren kann. Verwenden Sie **student** bei Anforderung als Passwort.
3. Zeigen Sie die Ereignisprotokolle an, die in den letzten 30 Minuten auf **serverb** erfasst wurden.
4. Erstellen Sie die Datei **/etc/rsyslog.d/auth-errors.conf** mit der erforderlichen Konfiguration, damit der Service **rsyslog** Meldungen in Bezug auf Authentifizierungs- und Sicherheitsproblemen in die neue Datei **/var/log/auth-errors** schreibt. Verwenden Sie die Facility **authpriv** und die Priorität **alert** in der Konfigurationsdatei.

### Bewertung

Führen Sie auf **workstation** den Befehl **lab log-review grade** aus, um den Erfolg dieser Übung zu bestätigen.

```
[student@workstation ~]$ lab log-review grade
```

## Beenden

Führen Sie auf **workstation** das Skript **lab log-review finish** aus, um die praktische Übung abzuschließen. Dieses Skript stellt sicher, dass die ursprüngliche Zeitzone zusammen mit allen ursprünglichen Zeiteinstellungen auf **serverb** wiederhergestellt wird.

```
[student@workstation ~]$ lab log-review finish
```

Hiermit ist die angeleitete Übung beendet.

## ► Lösung

# Analysieren und Speichern von Protokollen

### Leistungscheckliste

In dieser Übung ändern Sie die Zeitzone auf einem vorhandenen Server und konfigurieren eine neue Protokolldatei für alle Ereignisse, die sich auf Authentifizierungsfehler beziehen.

### Ergebnisse

Es werden folgende Fähigkeiten vermittelt:

- Aktualisieren Sie die Zeitzone auf einem vorhandenen Server.
- Konfigurieren Sie eine neue Protokolldatei, um alle Nachrichten zu speichern, die sich auf Authentifizierungsfehler beziehen.

### Bevor Sie Beginnen

Melden Sie sich als Benutzer **student** mit dem Passwort **student** bei **workstation** an.

Führen Sie auf **workstation** das Skript **lab log-review start** aus, um diese Übung zu beginnen. Dieses Skript zeichnet die aktuelle Zeitzone vom System **serverb** auf und stellt sicher, dass die Umgebung korrekt eingerichtet ist.

```
[student@workstation ~]$ lab log-review start
```

1. Öffnen Sie auf **workstation** als **student** eine SSH-Sitzung zu **serverb**.

```
[student@workstation ~]$ ssh student@serverb
...output omitted...
[student@serverb ~]$
```

2. Nehmen Sie an, dass das System **serverb** nach Jamaica verlagert wird, daher müssen Sie die Zeitzone entsprechend aktualisieren. Erhöhen Sie mit **sudo** die Berechtigungen des Benutzers **student**, damit der Befehl **timedatectl** die Zeitzone aktualisieren kann. Verwenden Sie **student** bei Anforderung als Passwort.
  - 2.1. Zeigen Sie mit dem Befehl **timedatectl** die verfügbaren Zeitzonen an und ermitteln Sie die geeignete Zeitzone für Jamaika.

```
[student@serverb ~]$ timedatectl list-timezones | grep Jamaica
America/Jamaica
```

- 2.2. Legen Sie mit dem Befehl **timedatectl** die Zeitzone des Systems **serverb** auf **America/Jamaica** fest.

```
[student@serverb ~]$ sudo timedatectl set-timezone America/Jamaica  
[sudo] password for student:
```

- 2.3. Überprüfen Sie mit dem Befehl **timedatectl**, ob die Zeitzone erfolgreich auf **America/Jamaica** festgelegt wurde.

```
[student@serverb ~]$ timedatectl  
          Local time: Tue 2019-02-19 11:12:46 EST  
      Universal time: Tue 2019-02-19 16:12:46 UTC  
        RTC time: Tue 2019-02-19 16:12:45  
       Time zone: America/Jamaica (EST, -0500)  
System clock synchronized: yes  
    NTP service: active  
RTC in local TZ: no
```

3. Zeigen Sie die Ereignisprotokolle an, die in den letzten 30 Minuten auf **serverb** erfasst wurden.

- 3.1. Legen Sie mit dem Befehl **date** den Zeitrahmen fest, für den Journaleinträge angezeigt werden sollen.

```
[student@serverb ~]$ date  
Fri Feb 22 07:31:05 EST 2019  
[student@serverb ~]$ date -d "-30 minutes"  
Fri Feb 22 07:01:31 EST 2019
```

- 3.2. Zeigen Sie mit dem Befehl **journalctl** und den Optionen **--since** und **--until** die Protokollereignisse an, die in den letzten 30 Minuten auf **serverb** erfasst wurden. Um **journalctl** zu beenden, drücken Sie **q**.

```
[student@serverb ~]$ journalctl --since 07:01:00 --until 07:31:00  
...output omitted...  
Feb 22 07:24:28 serverb.lab.example.com systemd[1138]: Reached target Timers.  
Feb 22 07:24:28 serverb.lab.example.com systemd[1138]: Reached target Paths.  
Feb 22 07:24:28 serverb.lab.example.com systemd[1138]: Starting D-Bus User Message Bus Socket.  
Feb 22 07:24:28 serverb.lab.example.com systemd[1138]: Listening on D-Bus User Message Bus Socket.  
Feb 22 07:24:28 serverb.lab.example.com systemd[1138]: Reached target Sockets.  
Feb 22 07:24:28 serverb.lab.example.com systemd[1138]: Reached target Basic System.  
Feb 22 07:24:28 serverb.lab.example.com systemd[1138]: Reached target Default.  
Feb 22 07:24:28 serverb.lab.example.com systemd[1138]: Startup finished in 123ms.  
Feb 22 07:24:28 serverb.lab.example.com systemd[1]: Started User Manager for UID 1000.  
Feb 22 07:24:28 serverb.lab.example.com sshd[1134]: pam_unix(sshd:session): session opened for user student by (uid=0)  
Feb 22 07:26:56 serverb.lab.example.com systemd[1138]: Starting Mark boot as successful...
```

```
Feb 22 07:26:56 serverb.lab.example.com systemd[1138]: Started Mark boot as
successful.
lines 1-36/36 (END) q
[student@serverb ~]$
```

4. Erstellen Sie die Datei **/etc/rsyslog.d/auth-errors.conf** mit der erforderlichen Konfiguration, damit der Service **rsyslog** Meldungen in Bezug auf Authentifizierungs- und Sicherheitsproblemen in die neue Datei **/var/log/auth-errors** schreibt. Verwenden Sie die Facility **authpriv** und die Priorität **alert** in der Konfigurationsdatei.
  - 4.1. Erstellen Sie die Datei **/etc/rsyslog.d/auth-errors.conf**, um die neue Datei **/var/log/auth-errors** als Ziel für Meldungen anzugeben, die sich auf Authentifizierungs- und Sicherheitsprobleme beziehen. Sie können den Befehl **sudo vim /etc/rsyslog.d/auth-errors.conf** verwenden, um die Konfigurationsdatei zu erstellen.

```
authpriv.alert  /var/log/auth-errors
```

- 4.2. Starten Sie den Service **rsyslog** neu, damit die Änderungen in der Konfigurationsdatei wirksam werden.

```
[student@serverb ~]$ sudo systemctl restart rsyslog
```

- 4.3. Verwenden Sie den Befehl **logger**, um eine neue Protokollmeldung in die Datei **/var/log/auth-errors** zu schreiben. Verwenden Sie die Option **-p authpriv.alert**, um eine Protokollmeldung zu generieren, die sich auf Authentifizierungs- und Sicherheitsprobleme bezieht.

```
[student@serverb ~]$ logger -p authpriv.alert "Logging test authpriv.alert"
```

- 4.4. Überprüfen Sie mit dem Befehl **tail**, ob die Datei **/var/log/auth-errors** den Protokolleintrag mit der Meldung **Logging test authpriv.alert** enthält.

```
[student@serverb ~]$ sudo tail /var/log/auth-errors
Feb 19 11:56:07 serverb student[6038]: Logging test authpriv.alert
```

- 4.5. Melden Sie sich von **serverb** ab.

```
[student@serverb ~]$ exit
logout
Connection to serverb closed.
[student@workstation ~]$
```

## Bewertung

Führen Sie auf **workstation** den Befehl **lab log-review grade** aus, um den Erfolg dieser Übung zu bestätigen.

```
[student@workstation ~]$ lab log-review grade
```

## Beenden

Führen Sie auf **workstation** das Skript **lab log-review finish** aus, um die praktische Übung abzuschließen. Dieses Skript stellt sicher, dass die ursprüngliche Zeitzone zusammen mit allen ursprünglichen Zeiteinstellungen auf **serverb** wiederhergestellt wird.

```
[student@workstation ~]$ lab log-review finish
```

Hiermit ist die angeleitete Übung beendet.

# Zusammenfassung

---

In diesem Kapitel erhielten Sie folgende Informationen:

- Die Services **systemd-journald** und **rsyslog** erfassen und schreiben Protokollmeldungen in die entsprechenden Dateien.
- Das Verzeichnis **/var/log** enthält Protokolldateien.
- Die periodische Rotation von Protokolldateien verhindert, dass der Dateisystemspeicher voll wird.
- Die **systemd**-Journale sind temporär und bleiben nach dem Bootvorgang nicht bestehen.
- Der **chrony**-Service hilft, Zeiteinstellungen mit einer Zeitquelle zu synchronisieren.
- Die Zeitzone des Servers kann basierend auf seinem Standort aktualisiert werden.

## Kapitel 12

# Netzwerkmanagement

### Ziel

Konfigurieren von Netzwerkschnittstellen und Einstellungen auf Red Hat Enterprise Linux – Servern

### Ziele

- Beschreiben der grundlegenden Konzepte der Netzwerkadressierung und des Routings für einen Server
- Testen und Prüfen der aktuellen Netzwerkkonfiguration mit Befehlszeilenprogrammen
- Verwalten von Netzwerkeinstellungen und Geräten mit **nmcli**
- Ändern von Netzwerkeinstellungen durch Bearbeiten der Konfigurationsdateien
- Konfigurieren des statischen Hostnamens eines Servers und der Namensauflösung sowie Testen der Ergebnisse

### Abschnitte

- Beschreiben von Netzwerkkonzepten (mit Quiz)
- Validieren der Netzwerkkonfiguration (mit angeleiteter Übung)
- Konfigurieren von Netzwerken über die Befehlszeile (mit angeleiteter Übung)
- Bearbeiten der Netzwerkconfigurationsdateien (mit angeleiteter Übung)
- Konfigurieren von Hostnamen und Namensauflösung (mit angeleiteter Übung)

### Praktische Übung

Netzwerkmanagement

# Beschreiben von Netzwerkkonzepten

---

## Ziele

Nach Abschluss dieses Abschnittes sollten Sie die grundlegenden Konzepte der Netzwerkkonfiguration und des Routings für einen Server beschreiben können.

## TCP/IP-Netzwerkmodell

Das *TCP/IP-Netzwerkmodell* ist eine vereinfachte, vierstufige Gruppe von Abstraktionen, die beschreibt, wie verschiedene Protokolle zusammenarbeiten, damit Computer über das Internet Datenverkehr von einem Computer zum anderen senden können. Es wird in RFC 1122, *Requirements for Internet Hosts – Communication Layers*, spezifiziert. Die vier Schichten sind:

- **Anwendung**

Jede Anwendung verfügt über festgelegte Kommunikationsrichtlinien, sodass Clients und Server plattformübergreifend miteinander kommunizieren können. Zu den gängigen Protokollen zählen SSH (Fernanmeldung), HTTPS (sicheres Internet) NFS oder CIFS (Dateifreigabe) und SMTP (E-Mail-Übermittlung).

- **Transport**

TCP und UDP sind die Transportprotokolle. TCP ist ein zuverlässiges, verbindungsorientiertes Transportprotokoll, wohingegen UDP ein verbindungsloses *Datenpaket*-Protokoll ist. Anwendungsprotokolle verwenden TCP- oder UDP-Ports. Eine Liste der allgemein bekannten und registrierten Ports befindet sich in der Datei **/etc/services**.

Wenn ein Paket über das Netzwerk verschickt wird, bilden Service-Port und IP-Adresse einen *Socket*. Jedes Paket hat einen Quell-Socket und einen Ziel-Socket. Diese Informationen können zur Überwachung und zum Filtern verwendet werden.

- **Internet**

Daten werden über die Internet- bzw. Netzwerkschicht vom Quell-Host an den Ziel-Host übermittelt. Die Protokolle IPv4 und IPv6 sind Protokolle der Internetschicht. Jedem Host sind eine IP-Adresse und ein Präfix zugewiesen, mit denen die Netzwerkkonfiguration bestimmt werden kann. Router verbinden Netzwerke.

- **Link bzw. Datensicherung**

Die Datensicherungsschicht stellt die Verbindung zu physischen Medien her. Das kabelgebundene Ethernet (802.3) und das kabellose WLAN (802.11) sind die beiden am häufigsten verwendeten Netzwerktypen. Jedes physische Gerät besitzt eine Hardware-Adresse (MAC), mit der das Ziel der Pakete auf dem lokalen Netzwerksegment identifiziert werden kann.

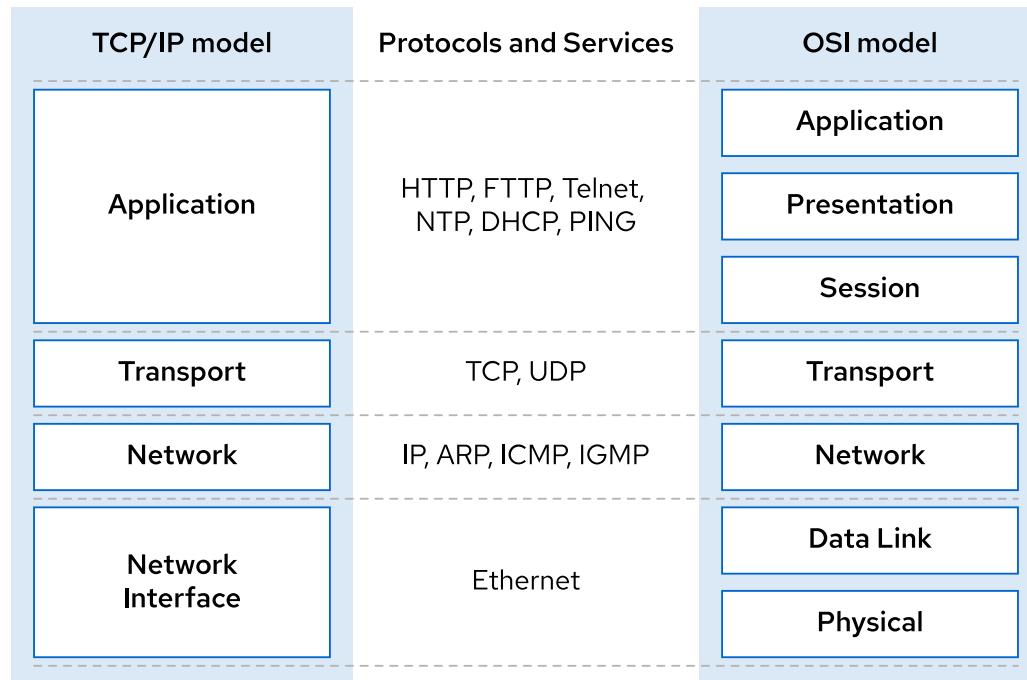


Abbildung 12.1: Vergleich der TCP/IP- und OSI-Netzwerkmodelle

## Beschreiben der Namen von Netzwerkschnittstellen

Jeder Netzwerkport eines Systems hat einen Namen, den Sie zum Konfigurieren und Identifizieren verwenden.

Ältere Versionen von Red Hat Enterprise Linux verwendeten Namen wie **eth0**, **eth1** und **eth2** für die einzelnen Netzwerkschnittstellen. Der Name **eth0** war der erste vom Betriebssystem erkannte Netzwerkport, **eth1** der zweite und so weiter. Wenn Geräte hinzugefügt und entfernt werden, kann der Mechanismus zum Erkennen und Benennen von Geräten jedoch ändern, welche Schnittstelle welchen Namen erhält. Darüber hinaus garantiert der PCIe-Standard nicht die Reihenfolge, in der PCIe-Geräte beim Booten erkannt werden. Dies kann zu unerwarteten Änderungen der Gerätennamen beim Geräte- oder Systemstart führen.

Neuere Versionen von Red Hat Enterprise Linux verwenden ein anderes Benennungssystem. Anstatt auf der Erkennungsreihenfolge zu basieren, werden die Namen der Netzwerkschnittstellen anhand von Informationen aus der Firmware, der PCI-Bustopologie und dem Typ des Netzwerkgeräts zugewiesen.

Die Namen der Netzwerkschnittstellen beginnen mit dem Schnittstellentyp:

- Ethernet-Schnittstellen beginnen mit **en**
- WLAN-Schnittstellen beginnen mit **wl**
- WWAN-Schnittstellen beginnen mit **ww**

Der Rest des Schnittstellennamens nach dem Typ basiert auf Informationen, die von der Serverfirmware bereitgestellt oder vom Standort des Geräts in der PCI-Topologie bestimmt werden.

- **oN** gibt an, dass dies ein integriertes Gerät ist, und die vom der Firmware des Servers bereitgestellte Indexnummer **N** für das Gerät. **eno1** ist daher das integrierte Ethernet-Gerät 1. Viele Server stellen diese Informationen nicht zur Verfügung.

## Kapitel 12 | Netzwerkmanagement

- **sN** gibt an, dass sich dieses Gerät im PCI-Hotplug-Steckplatz N befindet. **ens3** ist daher eine Ethernet-Karte im PCI-Hotplug-Steckplatz 3.
- **pMsN** gibt an, dass dies ein PCI-Gerät auf dem Bus M im Steckplatz N ist. **wlp4s0** ist daher eine WLAN-Karte auf PCI-Bus 4 in Steckplatz 0. Wenn die Karte ein Multifunktionsgerät ist (möglicherweise mit einer Ethernet-Karte mit mehreren Ports oder Geräten mit Ethernet und anderen Funktionen), könnte **fN** an den Gerätenamen angefügt sein. **enp0s1f0** ist daher Funktion 0 der Ethernet-Karte auf Bus 0 in Steckplatz 1. Möglicherweise gibt es auch eine zweite Schnittstelle mit dem Namen **enp0s1f1**. Das ist Funktion 1 des gleichen Geräts.

Permanente Benennung bedeutet, dass Sie, sobald Sie wissen, wie die Netzwerkschnittstelle im System heißt, auch wissen, dass sie sich später nicht mehr ändert. Der Nachteil ist, dass Sie nicht davon ausgehen können, dass ein System mit einer Schnittstelle diese Schnittstelle **eth0** benennt.

## IPv4-Netzwerke

IPv4 ist das primäre Netzwerkprotokoll, das heute im Internet verwendet wird. Sie sollten mindestens über ein grundlegendes Verständnis von IPv4-Netzwerken verfügen, um die Netzwerkkommunikation für Ihre Server verwalten zu können.

## IPv4-Adressen

Eine IPv4-Adresse ist eine 32-Bit-Zahl, die in der Regel im Dezimalformat in vier durch Punkte getrennten 8-Bit-Oktette mit Werten zwischen 0 und 255 dargestellt wird. Die Adresse ist in den *Netzwerkteil* und den *Hostteil* aufgeteilt. Für alle Hosts im gleichen Subnetz, die ohne Router direkt miteinander kommunizieren können, ist der Netzwerkteil identisch. Durch den Netzwerkteil wird das Subnetz identifiziert. Der Hostteil ist innerhalb eines Subnetzes nur einmal vorhanden, da er einen bestimmten Host eindeutig identifiziert.

Im modernen Internet ist die Größe eines IPv4-Subnetzes variabel. Um zu ermitteln, welcher Teil einer IPv4-Adresse der Netzwerk- und welcher der Hostteil ist, muss einem Administrator die dem Subnetz zugewiesene *Netzmaske* bekannt sein. Die Netzmaske gibt an, wie viele Bits der IPv4-Adresse zum Subnetz gehören. Je mehr Bits für den Hostteil zur Verfügung stehen, desto mehr Hosts kann das Subnetz enthalten.

Die niedrigste mögliche Adresse in einem Subnetz (Hostteil besteht im Binärformat nur aus Nullen) wird manchmal als *Netzwerkadresse* bezeichnet. Die höchste mögliche Adresse in einem Subnetz (Hostteil besteht im Binärformat nur aus Einsen) wird in IPv4 für Broadcast-Nachrichten verwendet und entsprechend als *Broadcast-Adresse* bezeichnet.

Netzmasken werden in zwei Formaten dargestellt. Die ältere Syntax einer Netzmaske, die für den Netzwerkteil 24 Bits verwendet, lautet **255.255.255.0**. In einer neueren, als CIDR-Notation bezeichneten Syntax wird das *Netzwerkpräfix /24* festgelegt. Beide Formate vermitteln dieselben Informationen, nämlich die Anzahl der führenden Bits in der IP-Adresse, welche die Netzwerkadresse bestimmen.

Die folgenden Beispiele beschreiben, wie die IP-Adresse, das Präfix (die Netzmaske), der Netzwerkteil und der Hostteil zusammenhängen.

**IP Address:**

$$172.17.5.3 = \textcolor{red}{10101100}.\textcolor{blue}{00010001}.\textcolor{blue}{00000101}.\textcolor{blue}{00000011}$$
**Netmask:**

$$255.255.0.0 = \textcolor{blue}{11111111}.\textcolor{blue}{11111111}.\textcolor{blue}{00000000}.\textcolor{blue}{00000000}$$

Prefix: /16  

$$\underbrace{\textcolor{blue}{10101100}.\textcolor{blue}{00010001}}_{\text{Network}} \underbrace{\textcolor{red}{00000101}.\textcolor{blue}{00000011}}_{\text{Host}}$$

**IP Address:**

$$192.168.5.3 = \textcolor{red}{11000000}.\textcolor{blue}{10101000}.\textcolor{blue}{00000101}.\textcolor{blue}{00000011}$$
**Netmask:**

$$255.255.255.0 = \textcolor{blue}{11111111}.\textcolor{blue}{11111111}.\textcolor{blue}{11111111}.\textcolor{blue}{00000000}$$

Prefix: /24  

$$\underbrace{\textcolor{blue}{11000000}.\textcolor{blue}{10101000}.\textcolor{blue}{00000101}}_{\text{Network}} \underbrace{\textcolor{red}{00000011}}_{\text{Host}}$$

Abbildung 12.2: IPv4-Adressen und Netzmasken

## Bestimmen der Netzwerkadresse für 192.168.1.107/24

|                   |                     |                                            |
|-------------------|---------------------|--------------------------------------------|
| Hostadresse       | 192.168.1.107       | <b>11000000.10101000.00000001.01101011</b> |
| Netzwerkpräfix    | /24 (255.255.255.0) | <b>11111111.11111111.11111111.00000000</b> |
| Netzwerkadresse   | 192.168.1.0         | <b>11000000.10101000.00000001.00000000</b> |
| Broadcast-Adresse | 192.168.1.255       | <b>11000000.10101000.00000001.11111111</b> |

## Bestimmen der Netzwerkadresse für 10.1.1.18/8

|                   |                |                                            |
|-------------------|----------------|--------------------------------------------|
| Hostadresse       | 10.1.1.18      | <b>00001010.00000001.00000001.00010010</b> |
| Netzwerkpräfix    | /8 (255.0.0.0) | <b>11111111.00000000.00000000.00000000</b> |
| Netzwerkadresse   | 10.0.0.0       | <b>00001010.00000000.00000000.00000000</b> |
| Broadcast-Adresse | 10.255.255.255 | <b>00001010.11111111.11111111.11111111</b> |

## Bestimmen der Netzwerkadresse für 172.16.181.23/19

|             |                |                                            |
|-------------|----------------|--------------------------------------------|
| Hostadresse | 172.168.181.23 | <b>10101100.10101000.10110101.00010111</b> |
|-------------|----------------|--------------------------------------------|

|                   |                     |                                            |
|-------------------|---------------------|--------------------------------------------|
| Netzwerkpräfix    | /19 (255.255.224.0) | <b>11111111.11111111.11100000.00000000</b> |
| Netzwerkadresse   | 172.168.160.0       | <b>10101100.10101000.10100000.00000000</b> |
| Broadcast-Adresse | 172.168.191.255     | <b>10101100.10101000.10111111.11111111</b> |

Die spezielle Adresse 127.0.0.1 verweist immer auf das lokale System („localhost“), und das Netzwerk 127.0.0.0/8 gehört zum lokalen System, sodass dieses über Netzwerkprotokolle mit sich selbst kommunizieren kann.

## IPv4-Routing

Sowohl bei IPv4 als auch bei IPv6 muss der Netzwerdatenverkehr von Host zu Host und von Netzwerk zu Netzwerk übertragen werden. Jeder Host verfügt über eine *Routingtabelle*, die festlegt, wie der Datenverkehr für bestimmte Netzwerke geroutet werden soll. Ein Eintrag der Routingtabelle nennt das Zielnetzwerk, die für den Datenverkehr zu verwendende Schnittstelle und die IP-Adresse des dazwischenliegenden Routers, über den die Nachricht an den Zielort übermittelt wird. Der Eintrag in der Routingtabelle, der mit dem Ziel des Netzwerdatenverkehrs übereinstimmt, wird für das Routing verwendet. Bei zwei übereinstimmenden Einträgen wird der Eintrag mit dem längsten Präfix verwendet.

Wenn der Netzwerkverkehr nicht mit einer spezifischeren Route übereinstimmt, enthält die Routingtabelle normalerweise einen Eintrag für eine *Standardroute* zum gesamten IPv4-Internet: 0.0.0.0/0. Diese Standardroute verweist auf einen Router in einem erreichbaren Subnetz (d. h. in einem Subnetz mit einer spezifischeren Route in der Routingtabelle des Hosts).

Wenn ein Router nicht an ihn adressierten Datenverkehr empfängt, wird dieser Datenverkehr nicht wie bei einem regulären Host ignoriert, sondern anhand der Routingtabelle *weitergeleitet*. Somit wird der Datenverkehr entweder direkt an den Zielhost gesendet (falls sich der Router im selben Subnetz befindet) oder an einen anderen Router weitergeleitet. Dieser Weiterleitungsprozess wird fortgesetzt, bis der Datenverkehr sein Ziel erreicht hat.

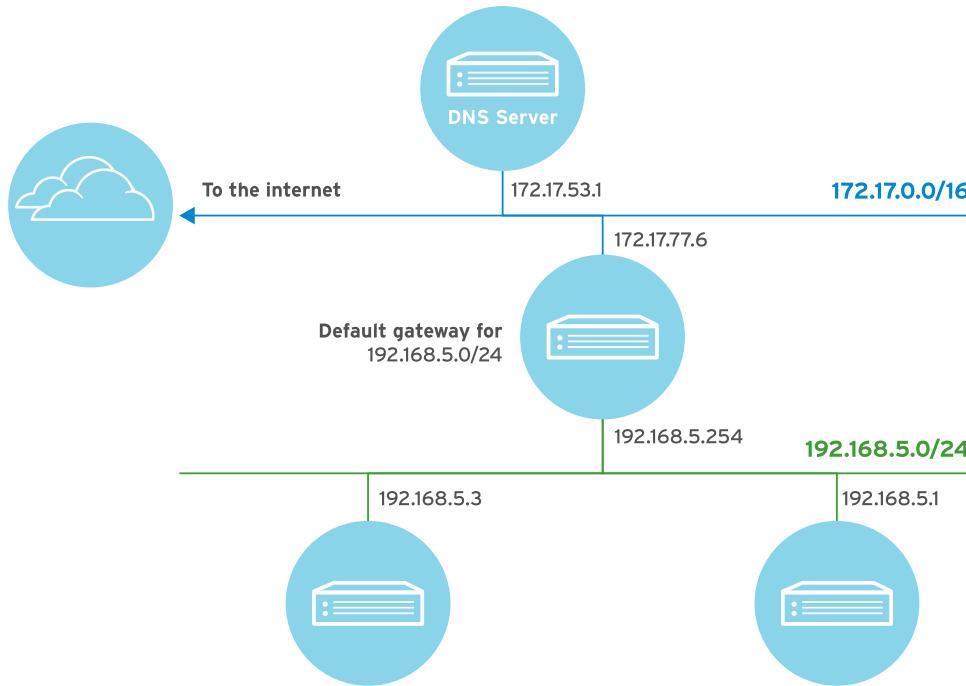


Abbildung 12.3: Beispiel einer Netzwerktopologie

### Beispiel einer Routingtabelle

| Ziel               | Schnittstelle | Router (falls benötigt) |
|--------------------|---------------|-------------------------|
| 192.0.2.0/24       | wlo1          |                         |
| 192.168.5.0/24     | enp3s0        |                         |
| 0.0.0.0 (Standard) | enp3s0        | 192.168.5.254           |

In diesem Beispiel wird der Datenverkehr von diesem Host an die IP-Adresse **192.0.2.102** direkt über die kabellose Schnittstelle **wlo1** ans Ziel übermittelt, da diese die größte Übereinstimmung mit der Route **192.0.2.0/24** hat. Datenverkehr an die IP-Adresse **192.168.5.3** wird direkt über die Ethernet-Schnittstelle **enp3s0** ans Ziel übermittelt, da diese die größte Übereinstimmung mit der Route **192.168.5.0/24** hat.

Datenverkehr an die IP-Adresse **10.2.24.1** wird über die Ethernet-Schnittstelle **enp3s0** an einen Router unter **192.168.5.254** übermittelt, der den Datenverkehr an seinen Zielort weiterleitet. Dieser Datenverkehr hat die größte Übereinstimmung mit der Route **0.0.0.0/0**, da es keine spezifischere Route in der Routingtabelle des Hosts gibt. Der Router verwendet seine eigene Routingtabelle, um zu bestimmen, wohin der Datenverkehr weitergeleitet werden soll.

## IPv4-Adress- und Routenkonfiguration

Ein Server kann seine IPv4-Netzwerkeinstellungen automatisch beim Start über einen *DHCP*-Server konfigurieren. Ein lokaler Client-Daemon fragt den Link nach einem Server und Netzwerkeinstellungen ab und ruft eine *Lease* ab, um diese Einstellungen für eine bestimmte

Zeitspanne zu verwenden. Wenn der Client nicht in regelmäßigen Abständen eine Verlängerung der Lease anfordert, können die Netzwerkkonfigurationseinstellungen verloren gehen.

Alternativ können Sie einen Server für die Verwendung einer *statischen* Netzwerkkonfiguration konfigurieren. In diesem Fall werden die Netzwerkeinstellungen aus lokalen Konfigurationsdateien gelesen. Sie müssen die korrekten Einstellungen von Ihrem Netzwerkadministrator erhalten und bei Bedarf manuell aktualisieren, um Konflikte mit anderen Servern zu vermeiden.

## IPv6-Netzwerke

IPv6 soll das Netzwerkprotokoll IPv4 letztendlich ersetzen. Sie müssen die Funktionsweise verstehen, da immer mehr Produktionssysteme die IPv6-Addressierung verwenden. Beispielsweise verwenden viele ISPs bereits IPv6 für interne Kommunikations- und Geräteverwaltungsnetzwerke, um seltene IPv4-Adressen für Kundenzwecke zu erhalten.

IPv6 kann in einem *Dual-Stack*-Modell auch parallel zu IPv4 verwendet werden. In dieser Konfiguration kann eine Netzwerkschnittstelle eine IPv6-Adresse oder -Adressen sowie IPv4-Adressen haben. Red Hat Enterprise Linux arbeitet standardmäßig in einem Dual-Stack-Modell.

## IPv6-Adressen

Eine IPv6-Adresse ist eine 128-Bit-Zahl, die normalerweise in Form von acht durch Doppelpunkte getrennte Gruppen von vier hexadezimalen Nibbles (halbe Byte) ausgedrückt wird. Jedes Nibble steht für vier Bit der IPv6-Adresse, das heißt jede Gruppe steht für 16 Bit der IPv6-Adresse.

```
2001:0db8:0000:0010:0000:0000:0000:0001
```

Um die Eingabe von IPv6-Adressen zu vereinfachen, können führende Nullen in einer durch Doppelpunkt getrennten Gruppe weggelassen werden. In jeder durch Doppelpunkt getrennten Gruppe muss jedoch mindestens eine Hexadezimalziffer geschrieben werden.

```
2001:db8:0:10:0:0:0:1
```

Da Adressen mit langen Nullfolgen gängig sind, können Gruppen von aufeinander folgenden Nullen mit *exakt einem* `::`-Block verbunden werden.

```
2001:db8:0:10::1
```

Beachten Sie, dass diesen Regeln zufolge `2001:db8::010:0:0:0:1` zwar eine weniger geeignete Schreibweise, jedoch eine gültige Darstellung der Beispieladresse ist. Für Administratoren, die sich mit IPv6 noch nicht gut auskennen, mag das verwirrend sein. Einige Tipps zum Schreiben durchgängig lesbarer Adressen:

- Führende Nullen in einer Gruppe unterdrücken.
- Adressen mit `::` so weit wie möglich kürzen.
- Wenn eine Adresse zwei aufeinander folgende Gruppen von Nullen enthält, die gleich lang sind, sollten für jede Gruppe die Nullgruppen ganz links zu `::` und die Gruppen ganz rechts zu `:0:` gekürzt werden.
- Kein Nullgruppe mit `::` kürzen (obwohl zulässig). Stattdessen `:0:` verwenden und `::` für aufeinander folgende Nullgruppen aufsparen.
- Für hexadezimale Zahlen immer Buchstaben in Kleinschreibung **a** bis **f** verwenden.



## **Wichtig**

Wenn Sie eine IPv6-Adresse durch einen TCP- oder UDP-Netzwerkport ergänzen, schließen Sie die IPv6-Adresse in eckige Klammern ein, sodass der Port nicht so aussieht, als wäre er ein Teil der Adresse.

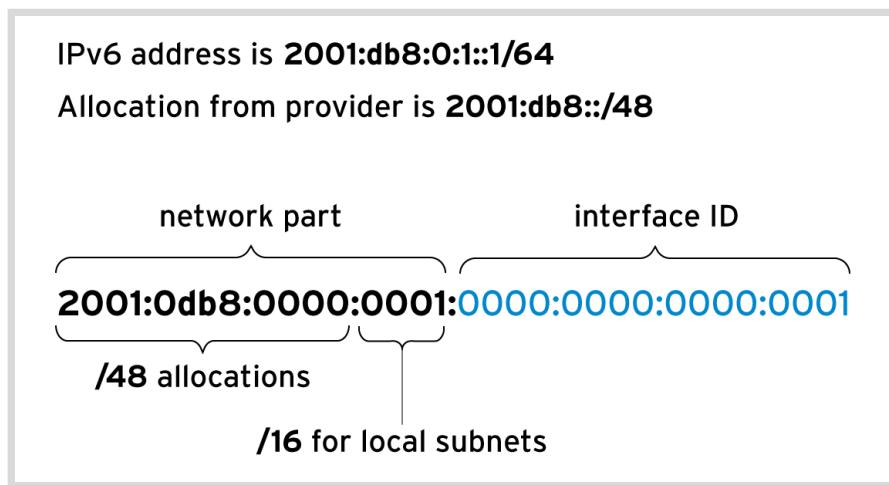
[2001:db8:0:10::1]:80

# IPv6-Subnetze

Normale IPv6-Unicast-Adressen bestehen aus den beiden Teilen *Netzwerkpräfix* und *Schnittstellen-ID*. Das Netzwerkpräfix identifiziert das Subnetz. Keine zwei Netzwerkschnittstellen im selben Subnetz dürfen dieselbe Schnittstellen-ID haben. Die Schnittstellen-ID identifiziert eine bestimmte Schnittstelle im Subnetz.

Im Gegensatz zu IPv4 hat IPv6 eine Standard-Subnetz-Maske, die für fast alle normalen Adressen verwendet wird, nämlich /64. In diesem Fall macht das Netzwerkpräfix die eine Hälfte der Adresse und die Schnittstellen-ID die andere Hälfte der Adresse aus. Das heißt, ein einziges Subnetz kann beliebig viele Hosts enthalten.

In der Regel wird der Netzwerk-Provider einer Organisation ein kürzeres Präfix zuweisen, beispielsweise /48. Dadurch bleibt ein Rest des Netzwerkteils für die Zuweisung von Subnetzen (immer mit der Länge /64) von diesem zugewiesenen Präfix übrig. Im Falle von /48 verbleiben 16 Bit für Subnetze (bis zu 65536 Subnetze).



**Abbildung 12.4: IPv6-Adressteile und Aufteilung in Subnetze**

## Gängige IPv6-Adressen und Netzwerke

| IPv6-Adresse oder Netzwerk | Zweck     | Beschreibung                                                                             |
|----------------------------|-----------|------------------------------------------------------------------------------------------|
| ::1/128                    | localhost | Die IPv6-Entsprechung zu <b>127.0.0.1/8</b> , eingerichtet in der Loopback-Schnittstelle |

| IPv6-Adress oder Netzwerk | Zweck                                 | Beschreibung                                                                                                                                                                                                                                                                                                                                                                                                                                                                            |
|---------------------------|---------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <code>::</code>           | Die nicht spezifizierte Adresse       | Die IPv6-Entsprechung zu <b>0.0.0.0</b> . Für Netzwerkservices kann diese Adresse angeben, dass alle konfigurierten IP-Adressen überwacht werden.                                                                                                                                                                                                                                                                                                                                       |
| <code>::/0</code>         | Die Standardroute (das IPv6-Internet) | Die IPv6-Entsprechung zu <b>0.0.0.0/0</b> . Die Standardroute in der Routingtabelle stimmt mit diesem Netzwerk überein. Der gesamte Verkehr, für den keine bessere Route vorhanden ist, wird an den Router für dieses Netzwerk gesendet.                                                                                                                                                                                                                                                |
| <code>2000::/3</code>     | Globale Unicast-Adressen              | „Normale“ IPv6-Adressen werden momentan von IANA aus diesem Bereich zugewiesen. Die entspricht allen Netzwerken von <b>2000::/16</b> bis <b>3fff::/16</b> .                                                                                                                                                                                                                                                                                                                             |
| <code>fd00::/8</code>     | Eindeutige lokale Adressen (RFC 4193) | IPv6 hat keine direkte Entsprechung zum privaten RFC 1918-Adressbereich. Diese Adresse kommt ihm jedoch sehr nahe. Sites können sich mit diesen Adressen selbst einen privaten routingfähigen IP-Adressbereich in der Organisation zuweisen. Diese Netzwerke können jedoch im globalen Internet nicht verwendet werden. Die Site muss <i>willkürlich</i> eine <b>/48-Zuteilung</b> von diesem Bereich auswählen, kann jedoch die Zuweisung normal in <b>/64</b> -Netzwerke unterteilen. |
| <code>fe80::/10</code>    | Link-local-Adressen                   | Jede IPv6-Schnittstelle konfiguriert automatisch eine <i>link-local</i> -Unicast-Adresse, die nur für den lokalen Link des <b>fe80::/64</b> -Netzwerks funktioniert.<br><br>Jedoch ist der gesamte <b>fe80::/10</b> -Bereich für die zukünftige Verwendung durch den lokalen Link reserviert. Dies wird im weiteren Verlauf des Kapitels ausführlich erläutert.                                                                                                                         |
| <code>ff00::/8</code>     | Multicast                             | Die IPv6-Entsprechung zu <b>224.0.0.0/4</b> . Multicast ermöglicht die Übertragung an mehrere Hosts gleichzeitig und ist besonders in IPv6 wichtig, da hier keine Broadcast-Adressen verwendet werden.                                                                                                                                                                                                                                                                                  |



### Wichtig

In der obigen Tabelle werden die Netzwerkadress-Zuordnungen aufgeführt, die für bestimmte Zwecke reserviert sind. Diese Zuordnungen können aus vielen verschiedenen Netzwerken bestehen. Zur Erinnerung: IPv6-Netzwerk, die von den globalen und link-local-Unicast-Bereichen zugewiesen werden, haben die Standard-Subnetz-Maske **/64**.

*Link-local-Adressen* in IPv6 sind Adressen, die nicht routingfähig sind. Sie dienen nur zur Kommunikation mit Hosts an einem bestimmten Netzwerk-Link. Jede Netzwerkschnittstelle im System wird im **fe80::/64**-Netzwerk automatisch mit einer link-local-Adresse konfiguriert. Um ihre Eindeutigkeit sicherzustellen, wird die Schnittstellen-ID der link-local-Adresse aus der Ethernet-Hardwareadresse der Netzwerkschnittstelle gebildet. In der Regel wird die 48-Bit-MAC-Adresse in eine 64-Bit-Schnittstellen-ID konvertiert, indem Bit 7 der MAC-Adresse invertiert und **ff:fe** zwischen die beiden mittleren Byte eingefügt wird.

- Netzwerkpräfix: **fe80::/64**
- MAC-Adresse: **00:11:22:aa:bb:cc**
- Link-local-Adresse: **fe80::211:22ff:fea:bbcc/64**

Die link-local-Adressen anderer Rechner können von anderen Hosts auf demselben Link wie normale Adressen verwendet werden. Da in jedem Link ein **fe80::/64**-Netzwerk vorhanden ist, ist es nicht möglich, die ausgehende Schnittstelle mit der Routingtabelle korrekt auszuwählen. Der für die Kommunikation mit einer link-local-Adresse zu verwendende Link muss am Ende der Adresse mit einem Scope-Bezeichner angegeben werden. Der Scope-Bezeichner beginnt mit einem %-Zeichen. Dann folgt der Name der Netzwerkschnittstelle.

Beispiel: Sie möchten die link-local-Adresse **fe80::211:22ff:fea:bbcc** mit **ping6** pingen und dazu den mit der Netzwerkschnittstelle **ens3** verbundenen Link verwenden. Die korrekte Syntax des Befehls lautet in diesem Fall:

```
[user@host ~]$ ping6 fe80::211:22ff:fea:bbcc%ens3
```



### Anmerkung

Scope-Bezeichner werden nur benötigt, wenn Adressen mit dem Bereich „link“ kontaktiert werden. Normale globale Adressen werden genauso verwendet wie in IPv4 und wählen ihre ausgehenden Schnittstellen in der Routingtabelle aus.

*Multicast* ermöglicht es einem System, Datenverkehr an eine spezielle IP-Adresse zu senden, die von mehreren Systemen empfangen wird. Es unterscheidet sich von Broadcast, da nur bestimmte Systeme im Netzwerk den Datenverkehr empfangen. Es unterscheidet sich auch von Broadcast in IPv4, da abhängig von der Konfiguration Ihrer Netzwerkrouter und -systeme möglicherweise ein Teil des Multicast-Datenverkehrs in andere Subnetze geroutet wird.

Multicast spielt in IPv6 eine wichtigere Rolle als in IPv4, da in IPv6 keine Broadcast-Adresse vorhanden ist. Eine wichtige Multicast-Adresse in IPv6 ist **ff02::1**, die link-local-Adresse **all-nodes**. Durch Pingen dieser Adresse wird der Datenverkehr an alle Knoten des Links gesendet. Link-scope-Multicast-Adressen (ab **ff02::/8**) müssen genau wie link-local-Adressen mit einem Scope-Bezeichner angegeben werden.

```
[user@host ~]$ ping6 ff02::1%ens3
PING ff02::1%ens3(ffff::1) 56 data bytes
64 bytes from fe80::211:22ff:feaa:bbcc: icmp_seq=1 ttl=64 time=0.072 ms
64 bytes from fe80::200:aaff:fe33:2211: icmp_seq=1 ttl=64 time=102 ms (DUP!)
64 bytes from fe80::bcd:efff:fea1:b2c3: icmp_seq=1 ttl=64 time=103 ms (DUP!)
64 bytes from fe80::211:22ff:feaa:bbcc: icmp_seq=2 ttl=64 time=0.079 ms
...output omitted...
```

## Konfiguration von IPv6-Adressen

Bei IPv4 gibt es zwei Konfigurationsmöglichkeiten für Adressen an Netzwerkschnittstellen. Netzwerkadressen können manuell durch den Administrator oder dynamisch mit DHCP vom Netzwerk konfiguriert werden. IPv6 unterstützt ebenfalls die manuelle Konfiguration und zwei Methoden der dynamischen Konfiguration, eine davon ist DHCPv6.

Schnittstellen-IDs für statische IPv6-Adressen können genau wie bei IPv4 beliebig ausgewählt werden. In IPv4 gibt es zwei Adressen im Netzwerk, die nicht verwendet werden konnten, die niedrigste Adresse im Subnetz und die höchste Adresse im Subnetz. In IPv6 sind die folgenden Schnittstellen-IDs reserviert und können nicht für normale Netzwerkadressen auf einem Host verwendet werden:

- Der von allen Routern im Link verwendete All-Zeros-Bezeichner **0000:0000:0000:0000** („Subnet Router Anycast“). (Für das Netzwerk **2001:db8::/64** wäre dies die Adresse **2001:db8::**.)
- Die Bezeichner **fdff:ffff:ffff:ff80** bis **fdff:ffff:ffff:ffff**.

Die Funktionsweise von DHCPv6 unterscheidet sich von der von DHCP für IPv4, da keine Broadcast-Adresse vorhanden ist. Im Wesentlichen sendet ein Host eine DHCPv6-Anforderung von der link-local-Adresse an den Port 547/UDP auf **ff02::1:2**. Dies ist die link-local-Multicast-Gruppe von **all-dhcp-Server**. Der DHCPv6-Server sendet anschließend eine Antwort mit den entsprechenden Informationen an Port 546/UDP an der link-local-Adresse des Clients.

Das *dhcp*-Paket in Red Hat Enterprise Linux 8 bietet Unterstützung für einen DHCPv6-Server.

Neben DHCPv6 unterstützt IPv6 auch eine zweite dynamische Konfigurationsmethode, die so genannte *Stateless Address Autoconfiguration (SLAAC)*. Normalerweise ruft der Host mit SLAAC seine Schnittstelle mit einer link-local-Adresse **fe80::/64** auf. Anschließend sendet er eine „Router Solicitation“ an **ff02::2**, die link-local-Multicast-Gruppe all-routers. Ein IPv6-Router am lokalen Link antwortet mit einem Netzwerkpräfix und möglicherweise anderen Informationen auf die link-local-Adresse des Hosts. Der Host verwendet dieses Netzwerkpräfix dann mit einer Schnittstellen-ID, die normalerweise auf dieselbe Weise erstellt wird wie die link-local-Adressen. Der Router sendet in regelmäßigen Abständen Multicast-Aktualisierungen („Router-Advertisements“), um die bereitgestellten Informationen zu bestätigen oder zu aktualisieren.

Mit dem *radvd*-Paket in Red Hat Enterprise Linux 8 kann ein IPv6-Router auf Red Hat Enterprise Linux -Basis SLAAC über Router-Advertisements bereitstellen.



### Wichtig

Darüber hinaus sieht die Konfiguration eines normalen Red Hat Enterprise Linux 8-Rechners, der IPv4-Adressen über DHCP abruft, auch die Verwendung von SLAAC für den Abruf von IPv6-Adressen vor. Dies kann dazu führen, dass Rechner unerwarteterweise IPv6-Adressen abrufen, wenn ein IPv6-Router in das Netzwerk eingebunden wird.

Einige IPv6-Bereitstellungen kombinieren SLAAC und DHCPv6. SLAAC stellt hierbei nur Informationen zu Netzwerkadressen und DHCPv6 andere Informationen bereit, beispielsweise die zu konfigurierenden DNS-Server und Suchdomains.

## Hostnamen und IP-Adressen

Es wäre unpraktisch, wenn Sie zum Kontaktieren Ihrer Server immer IP-Adressen verwenden müssten. Menschen arbeiten im Allgemeinen lieber mit Namen als mit langen und komplizierten Zahlenfolgen. Linux verfügt daher über eine Reihe von Mechanismen, um einen Hostnamen einer IP-Adresse zuzuordnen – zusammenfassend als *Namensauflösung* bezeichnet.

Eine Methode besteht darin, für jeden Namen einen statischen Eintrag in der Datei **/etc/hosts** auf jedem System festzulegen. Dazu müssen Sie die Kopie der Datei auf jedem Server manuell aktualisieren.

Bei den meisten Hosts können Sie die Adresse mittels eines Hostnamens (oder einen Hostnamen mittels einer Adresse) unter Verwendung des Netzwerkservice *Domain Name System (DNS)* suchen. DNS ist ein verteiltes Netzwerk von Servern, das Zuordnungen von Hostnamen zu IP-Adressen ermöglicht. Damit dieser Namensservice funktioniert, muss ein Host auf einen Nameserver verwiesen werden. Dieser Nameserver muss sich nicht zwangsläufig im selben Subnetz befinden; er muss jedoch vom Host erreicht werden können. Dies wird normalerweise über DHCP oder eine statische Einstellung in einer Datei mit dem Namen **/etc/resolv.conf** konfiguriert. In späteren Abschnitten dieses Kapitels wird beschrieben, wie die Namensauflösung konfiguriert wird.



## Literaturhinweise

Manpages **services(5)**, **ping(8)**, **biosdevname(1)** und **udev(7)**

Weitere Informationen finden Sie im Handbuch *Configuring and Managing Networking* unter

[https://access.redhat.com/documentation/en-us/red\\_hat\\_enterprise\\_linux/8/html-single/configuring\\_and\\_managing\\_networking/](https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux/8/html-single/configuring_and_managing_networking/)

### Grundlegendes zu den vorhersehbaren Netzwerkgerätenamen von systemd

<https://major.io/2015/08/21/understanding-systemds-predictable-network-device-names/>

Ausgewählte IETF RFC-Referenzen:

#### RFC 2460: Internet Protocol, Version 6 (IPv6) Specification

<http://tools.ietf.org/html/rfc2460>

#### RFC 4291: IP Version 6 Addressing Architecture

<http://tools.ietf.org/html/rfc4291>

#### RFC 5952: A Recommendation For IPv6 Address Text Representation

<http://tools.ietf.org/html/rfc5952>

#### RFC 4862: IPv6 Stateless Address Autoconfiguration

<http://tools.ietf.org/html/rfc4862>

#### RFC 3315: Dynamic Host Configuration Protocol for IPv6 (DHCPv6)

<http://tools.ietf.org/html/rfc3315>

#### RFC 3736: Stateless Dynamic Host Configuration Protocol (DHCP) Service for IPv6

<http://tools.ietf.org/html/rfc3736>

#### RFC 4193: Unique Local IPv6 Unicast Addresses

<http://tools.ietf.org/html/rfc4193>

## ► Quiz

# Beschreiben von Netzwerkkonzepten

Wählen Sie die richtige Antwort auf die folgenden Fragen aus:

- ▶ 1. Welche Zahl gibt die Größe einer IPv4-Adresse in Bit an?
  - a. 4
  - b. 8
  - c. 16
  - d. 32
  - e. 64
  - f. 128
  
- ▶ 2. Welcher Begriff bestimmt die Anzahl der führenden Bits in der IP-Adresse, welche die Netzwerkadresse bestimmen?
  - a. Netscope
  - b. Netzmaske
  - c. Subnetz
  - d. Multicast
  - e. netaddr
  - f. network
  
- ▶ 3. Welche Adresse stellt eine gültige IPv4-Host-Adresse im Netzwerk 192.168.1.0/24 dar?
  - a. 192.168.1.188
  - b. 192.168.1.0
  - c. 192.168.1.255
  - d. 192.168.1.256
  
- ▶ 4. Welche Zahl gibt die Größe einer IPv6-Adresse in Bit an?
  - a. 4
  - b. 8
  - c. 16
  - d. 32
  - e. 64
  - f. 128

► 5. Welche Adresse stellt keine gültige IPv6-Adresse dar?

- a. 2000:0000:0000:0000:0000:0000:0001
- b. 2::1
- c. ::
- d. ff02::1:0:0
- e. 2001:3::7:0:2
- f. 2001:db8::7::2
- g. 2000::1

► 6. Was ermöglicht es einem System, Datenverkehr an eine spezielle IP-Adresse zu senden, die von mehreren Systemen empfangen wird?

- a. Netscope
- b. Netzmaske
- c. Subnetz
- d. Multicast
- e. netaddr
- f. Netzwerk

## ► Lösung

# Beschreiben von Netzwerkkonzepten

Wählen Sie die richtige Antwort auf die folgenden Fragen aus:

► 1. Welche Zahl gibt die Größe einer IPv4-Adresse in Bit an?

- a. 4
- b. 8
- c. 16
- d. 32
- e. 64
- f. 128

► 2. Welcher Begriff bestimmt die Anzahl der führenden Bits in der IP-Adresse, welche die Netzwerkadresse bestimmen?

- a. Netscope
- b. Netzmaske
- c. Subnetz
- d. Multicast
- e. netaddr
- f. network

► 3. Welche Adresse stellt eine gültige IPv4-Host-Adresse im Netzwerk 192.168.1.0/24 dar?

- a. 192.168.1.188
- b. 192.168.1.0
- c. 192.168.1.255
- d. 192.168.1.256

► 4. Welche Zahl gibt die Größe einer IPv6-Adresse in Bit an?

- a. 4
- b. 8
- c. 16
- d. 32
- e. 64
- f. 128

► **5. Welche Adresse stellt keine gültige IPv6-Adresse dar?**

- a. 2000:0000:0000:0000:0000:0000:0001
- b. 2::1
- c. ::
- d. ff02::1:0:0
- e. 2001:3::7:0:2
- f. 2001:db8::7::2
- g. 2000::1

► **6. Was ermöglicht es einem System, Datenverkehr an eine spezielle IP-Adresse zu senden, die von mehreren Systemen empfangen wird?**

- a. Netscope
- b. Netzmaske
- c. Subnetz
- d. Multicast
- e. netaddr
- f. Netzwerk

# Validieren der Netzwerkkonfiguration

## Ziele

Nach Abschluss dieses Abschnitts sollten Sie in der Lage sein, die aktuelle Netzwerkkonfiguration mit Befehlszeilenprogrammen zu testen und zu prüfen.

## Erfassen von Netzwerkschnittstelleninformationen

### Identifizieren von Netzwerkschnittstellen

Der Befehl **ip link** listet alle auf Ihrem System verfügbaren Netzwerkschnittstellen auf:

```
[user@host ~]$ ip link show
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN mode DEFAULT
    group default qlen 1000
        link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
2: ens3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc mq state UP mode DEFAULT
    group default qlen 1000
        link/ether 52:54:00:00:0a brd ff:ff:ff:ff:ff:ff
3: ens4: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc mq state UP mode DEFAULT
    group default qlen 1000
        link/ether 52:54:00:00:1e brd ff:ff:ff:ff:ff:ff
```

Im obigen Beispiel verfügt der Server über drei Netzwerkschnittstellen: **lo** ist das Loopback-Gerät, das mit dem Server selbst verbunden ist, und zwei Ethernet-Schnittstellen, **ens3** und **ens4**.

Um jede Netzwerkschnittstelle korrekt zu konfigurieren, müssen Sie wissen, welche mit welchem Netzwerk verbunden ist. In vielen Fällen kennen Sie die MAC-Adresse der mit jedem Netzwerk verbundenen Schnittstelle, entweder weil sie physisch auf der Karte oder dem Server aufgedruckt ist oder weil es sich um einen virtuellen Rechner handelt, und Sie wissen, wie er konfiguriert ist. Die MAC-Adresse des Geräts wird für jede Schnittstelle nach **link/ether** aufgelistet. Damit wissen Sie, dass die Netzwerkkarte mit der MAC-Adresse **52:54:00:00:00:0a** die Netzwerkschnittstelle **ens3** ist.

## Anzeigen von IP-Adressen

Verwenden Sie den Befehl **ip**, um Informationen zu Geräten und Adressen anzuzeigen. Eine einzelne Netzwerkschnittstelle kann mehrere IPv4- oder IPv6-Adressen haben.

```
[user@host ~]$ ip addr show ens3
2: ens3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP
    qlen 1000
        ② link/ether 52:54:00:00:0b brd ff:ff:ff:ff:ff:ff
        ③ inet 192.0.2.2/24 brd 192.0.2.255 scope global ens3
            valid_lft forever preferred_lft forever
        ④ inet6 2001:db8:0:1:5054:ff:fe00:b/64 scope global
```

```
    valid_lft forever preferred_lft forever
⑤inet6 fe80::5054:ff:fe00:b/64 scope link
    valid_lft forever preferred_lft forever
```

- ① **UP** ist eine aktive Schnittstelle.
- ② In der Zeile **link/ether** ist die Hardware (MAC)-Adresse des Geräts angegeben.
- ③ Die Zeile **inet** zeigt eine IPv4-Adresse, die Länge des Netzwerkpräfixes und den Scope an.
- ④ Die Zeile **inet6** zeigt eine IPv6-Adresse, die Länge des Netzwerkpräfixes und den Scope an. Diese Adresse hat den Scope *global* und wird normalerweise verwendet.
- ⑤ Die Zeile **inet6** zeigt, dass sich die Schnittstelle auf eine IPv6-Adresse des Scopes *link* bezieht. Sie kann nur für die Kommunikation am lokalen Ethernet-Link verwendet werden.

## Anzeigen von Leistungsstatistiken

Mit dem Befehl **ip** können auch Statistiken zur Netzwerkleistung angezeigt werden. Für jede Netzwerkschnittstelle können Zähler verwendet werden, um Netzwerkprobleme zu erkennen. Die Zähler zeichnen Statistiken auf für Dinge wie die Anzahl der empfangenen (RX) und übertragenen (TX) Pakete, Paketfehler und die Anzahl der Pakete, die ignoriert (dropped) wurden.

```
[user@host ~]$ ip -s link show ens3
2: ens3: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc pfifo_fast state UP qlen 1000
    link/ether 52:54:00:00:00:0a brd ff:ff:ff:ff:ff:ff
        RX: bytes packets errors dropped overrun mcast
        269850    2931      0      0      0      0
        TX: bytes packets errors dropped carrier collsns
        300556    3250      0      0      0      0
```

## Prüfen der Konnektivität zwischen Hosts

Mit dem Befehl **ping** kann die Konnektivität geprüft werden. Der Befehl wird so lange ausgeführt, bis die Tastenkombination **Strg+c** gedrückt wird, außer es gibt Optionen zur Beschränkung der Anzahl der gesendeten Pakete.

```
[user@host ~]$ ping -c3 192.0.2.254
PING 192.0.2.1 (192.0.2.254) 56(84) bytes of data.
64 bytes from 192.0.2.254: icmp_seq=1 ttl=64 time=4.33 ms
64 bytes from 192.0.2.254: icmp_seq=2 ttl=64 time=3.48 ms
64 bytes from 192.0.2.254: icmp_seq=3 ttl=64 time=6.83 ms

--- 192.0.2.254 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2003ms
rtt min/avg/max/mdev = 3.485/4.885/6.837/1.424 ms
```

Der Befehl **ping6** ist die IPv6-Version von **ping** in Red Hat Enterprise Linux. Er kommuniziert über IPv6 und nimmt IPv6-Adressen an, funktioniert aber ansonsten wie **ping**.

```
[user@host ~]$ ping6 2001:db8:0:1::1
PING 2001:db8:0:1::1(2001:db8:0:1::1) 56 data bytes
64 bytes from 2001:db8:0:1::1: icmp_seq=1 ttl=64 time=18.4 ms
64 bytes from 2001:db8:0:1::1: icmp_seq=2 ttl=64 time=0.178 ms
64 bytes from 2001:db8:0:1::1: icmp_seq=3 ttl=64 time=0.180 ms
^C
```

```
-- 2001:db8:0:1::1 ping statistics --
3 packets transmitted, 3 received, 0% packet loss, time 2001ms
rtt min/avg/max/mdev = 0.178/6.272/18.458/8.616 ms
[user@host ~]$
```

Wenn Sie Link-local-Adressen und die Multicast-Gruppe link-local all-nodes (**ff02::1**) pingen, muss die zu verwendende Netzwerkschnittstelle explizit mit einem Bezeichner für die Scope-Zone angegeben werden (beispielsweise **ff02::1%ens3**). Fehlt dieser Bezeichner, wird der Fehler *connect::Invalid argument* angezeigt.

Das Pingen von **ff02::1** kann nützlich sein, um andere IPv6-Knoten auf dem lokalen Netzwerk zu finden.

```
[user@host ~]$ ping6 ff02::1%ens4
PING ff02::1%ens4(fe02::1) 56 data bytes
64 bytes from fe80::78cf:ffff:fed2:f97b: icmp_seq=1 ttl=64 time=22.7 ms
64 bytes from fe80::f482:dbff:fe25:6a9f: icmp_seq=1 ttl=64 time=30.1 ms (DUP!)
64 bytes from fe80::78cf:ffff:fed2:f97b: icmp_seq=2 ttl=64 time=0.183 ms
64 bytes from fe80::f482:dbff:fe25:6a9f: icmp_seq=2 ttl=64 time=0.231 ms (DUP!)
^C
--- ff02::1%ens4 ping statistics ---
2 packets transmitted, 2 received, +2 duplicates, 0% packet loss, time 1001ms
rtt min/avg/max/mdev = 0.183/13.320/30.158/13.374 ms
[user@host ~]$ ping6 -c 1 fe80::f482:dbff:fe25:6a9f%ens4
PING fe80::f482:dbff:fe25:6a9f%ens4(fe80::f482:dbff:fe25:6a9f) 56 data bytes
64 bytes from fe80::f482:dbff:fe25:6a9f: icmp_seq=1 ttl=64 time=22.9 ms

--- fe80::f482:dbff:fe25:6a9f%ens4 ping statistics ---
1 packets transmitted, 1 received, 0% packet loss, time 0ms
rtt min/avg/max/mdev = 22.903/22.903/22.903/0.000 ms
```

Bedenken Sie, dass IPv6-link-local-Adressen, genau wie normale Adressen, von anderen Hosts am selben Link verwendet werden können.

```
[user@host ~]$ ssh fe80::f482:dbff:fe25:6a9f%ens4
user@fe80::f482:dbff:fe25:6a9f%ens4's password:
Last login: Thu Jun  5 15:20:10 2014 from host.example.com
[user@server ~]$
```

## Fehlerbehebung für Routing

Das Netzwerk-Routing ist komplex und manchmal verhält sich der Datenverkehr nicht wie erwartet. Hier sind einige nützliche Diagnosewerkzeuge.

### Anzeigen der Routingtabelle

Zum Anzeigen von Routing-Informationen verwenden Sie den Befehl **ip** mit der Option **route**.

```
[user@host ~]$ ip route
default via 192.0.2.254 dev ens3 proto static metric 1024
192.0.2.0/24 dev ens3 proto kernel scope link src 192.0.2.2
10.0.0.0/8 dev ens4 proto kernel scope link src 10.0.0.11
```

Dies zeigt die IPv4-Routingtabelle an. Alle Pakete mit dem Netzwerk **10.0.0.0/8** als Ziel werden direkt über das Gerät **ens4** dorthin geschickt. Alle Pakete mit dem Netzwerk **192.0.2.0/24** als Ziel werden direkt über das Gerät **ens3** dorthin geschickt. Alle anderen Pakete werden über das Gerät **ens3** an den Standardrouter **192.0.2.254** geschickt.

Ergänzen Sie die Option **-6** zum Anzeigen der IPv6-Routingtabelle:

```
[user@host ~]$ ip -6 route
unreachable ::/96 dev lo metric 1024 error -101
unreachable ::ffff:0.0.0/96 dev lo metric 1024 error -101
2001:db8:0:1::/64 dev ens3 proto kernel metric 256
unreachable 2002:a00::/24 dev lo metric 1024 error -101
unreachable 2002:7f00::/24 dev lo metric 1024 error -101
unreachable 2002:a9fe::/32 dev lo metric 1024 error -101
unreachable 2002:ac10::/28 dev lo metric 1024 error -101
unreachable 2002:c0a8::/32 dev lo metric 1024 error -101
unreachable 2002:e000::/19 dev lo metric 1024 error -101
unreachable 3ffe:ffff::/32 dev lo metric 1024 error -101
fe80::/64 dev ens3 proto kernel metric 256
default via 2001:db8:0:1::ffff dev ens3 proto static metric 1024
```

Ignorieren Sie in diesem Beispiel die Routen, die nicht erreichbar sind und auf Netzwerke zeigen, die nicht verwendet werden. Es bleiben drei Routen:

1. Zum Netzwerk **2001:db8:0:1::/64** mit der **ens3**-Schnittstelle (die vermutlich eine Adresse auf diesem Netzwerk hat).
2. Das Netzwerk **fe80::/64** mit der **ens3**-Schnittstelle für die Link-local-Adresse. Auf einem System mit mehreren Schnittstellen gibt es eine Route zu **fe80::/64** von jeder Schnittstelle für jede link-local-Adresse.
3. Eine Standardroute zu allen Netzwerken im IPv6-Internet (das **::/0**-Netzwerk), die keine spezifischere Route im System haben. Die Route verläuft bei **2001:db8:0:1::ffff** durch den Router und ist mit dem **ens3**-Gerät erreichbar.

## Verfolgen von Datenverkehrs routen

Um den Pfad zu verfolgen, den der Netzwerksdatenverkehr nimmt, um einen Remote-Host über mehrere Router zu erreichen, verwenden Sie entweder **traceroute** oder **tracepath**.

Auf diese Weise können Sie feststellen, ob ein Problem bei einem Ihrer Router oder bei einem Zwischenprogramm vorliegt. Beide Befehle verwenden UDP-Pakete, um standardmäßig einen Pfad zu verfolgen. Viele Netzwerke blockieren jedoch Datenverkehr über UDP und ICMP. Mit dem Befehl **traceroute** kann der Pfad über UDP (Standard), ICMP (-I) oder TCP (-T) Pakete verfolgt werden. Normalerweise ist der Befehl **traceroute** jedoch nicht standardmäßig installiert.

```
[user@host ~]$ tracepath access.redhat.com
...output omitted...
4: 71-32-28-145.rcmt.qwest.net          48.853ms asymm 5
5: dcp-brdr-04.inet.qwest.net           100.732ms asymm 7
6: 206.111.0.153.ptr.us.xo.net         96.245ms asymm 7
7: 207.88.14.162.ptr.us.xo.net         85.270ms asymm 8
8: ae1d0.cir1.atlanta6-ga.us.xo.net    64.160ms asymm 7
9: 216.156.108.98.ptr.us.xo.net        108.652ms
```

```
10: bu-ether13.at1ngamq46w-bcr00.tbone.rr.com          107.286ms asymm 12
...output omitted...
```

Jede Zeile im Ergebnis des Befehls **tracepath** stellt einen Router oder einen *Hop* dar, den die Pakete auf ihrer Reise durchquert haben. Weitere Informationen wie das *Round Trip Timing (RTT)* oder Veränderungen an der Größe der *Maximum Transmission Unit (MTU)* werden bereitgestellt, falls sie verfügbar sind. Die Angabe **asymm** bedeutet, dass der Verkehr diesen Router erreicht hat und von diesem mit anderen (*asymmetrischen*) Routen zurückgegeben wurde. Die gezeigten Router sind diejenigen, die für den ausgehenden Datenverkehr verwendet werden, nicht für den Rückdatenverkehr.

Die Befehle **tracepath6** und **traceroute -6** entsprechen **tracepath** und **traceroute** für IPv6.

```
[user@host ~]$ tracepath6 2001:db8:0:2::451
1?: [LOCALHOST]                      0.091ms pmtu 1500
1:  2001:db8:0:1::ba                 0.214ms
2:  2001:db8:0:1::1                  0.512ms
3:  2001:db8:0:2::451                0.559ms reached
Resume: pmtu 1500 hops 3 back 3
```

## Fehlersuche bei Ports und Services

TCP-Services verwenden Sockets als Endpunkte für die Kommunikation und bestehen aus einer IP-Adresse, einem Protokoll und einer Portnummer. Services überwachen im Regelfall Standard-Ports, wohingegen Clients einen zufällig verfügbaren Port verwenden. Allgemein bekannte Benennungen für Standard-Ports sind in der Datei **/etc/services** aufgeführt.

Mit dem Befehl **ss** können Socket-Statistiken aufgerufen werden. Durch den Befehl **ss** soll das ältere Tool **netstat** ersetzt werden, das Teil des Pakets *net-tools* ist und einigen Systemadministratoren möglicherweise vertrauter ist. Es ist jedoch nicht immer installiert.

```
[user@host ~]$ ss -ta
State      Recv-Q Send-Q      Local Address:Port          Peer Address:Port
LISTEN      0      128              *:sunrpc               *:*
LISTEN      0      128              ①*:ssh                 *:*
LISTEN      0      100             ②127.0.0.1:smtp        *:*
LISTEN      0      128              *:36889               *:*
ESTAB       0      0                ③172.25.250.10:ssh    172.25.254.254:59392
LISTEN      0      128              :::sunrpc              :::*
LISTEN      0      128              ④:::ssh                :::*
LISTEN      0      100             ⑤::1:smtp              :::*
LISTEN      0      128              :::34946               :::*
```

- ➊ Der für SSH verwendete Port überwacht alle IPv4-Adressen. Mit dem Symbol „\*“ wird im Zusammenhang mit IPv4-Adressen oder Ports „all“ ausgedrückt.
- ➋ Der für SMTP verwendete Port hört auf der IPv4-Loopback-Schnittstelle **127.0.0.1**.
- ➌ Die hergestellte SSH-Verbindung befindet sich auf der Schnittstelle 172.25.250.10 und entspringt einem System mit der Adresse **172.25.254.254**.
- ➍ Der für SSH verwendete Port überwacht alle IPv6-Adressen. Mit der Syntax „::“ werden alle IPv6-Schnittstellen dargestellt.
- ➎ Der für SMTP verwendete Port überwacht die IPv6-Loopback-Schnittstelle ::1.

## Optionen für ss und netstat

| Option         | Beschreibung                                                                                                                                                                                                                                                                                                                                                                                                      |
|----------------|-------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>-n</b>      | Zeigt Zahlen anstelle von Namen für Schnittstellen und Ports an.                                                                                                                                                                                                                                                                                                                                                  |
| <b>-t</b>      | Zeigt TCP-Sockets an.                                                                                                                                                                                                                                                                                                                                                                                             |
| <b>-u</b>      | Zeigt UDP-Sockets an.                                                                                                                                                                                                                                                                                                                                                                                             |
| <b>-l</b>      | Zeigt nur abhörende Sockets an.                                                                                                                                                                                                                                                                                                                                                                                   |
| <b>-a</b>      | Zeigt alle (abhörende und etablierte) Sockets an.                                                                                                                                                                                                                                                                                                                                                                 |
| <b>-p</b>      | Zeigt den Prozess an, der die Sockets verwendet.                                                                                                                                                                                                                                                                                                                                                                  |
| <b>-A inet</b> | Zeigt aktive Verbindungen (jedoch keine abhörenden Sockets) für die <b>inet</b> -Adressenfamilie an. Das heißt, lokale UNIX-Domain-Sockets werden ignoriert.<br><br>Für <b>ss</b> werden IPv4 und IPv6-Verbindungen angezeigt.<br>Für <b>netstat</b> werden nur IPv4-Verbindungen angezeigt.<br>( <b>netstat -A inet6</b> zeigt IPv6-Verbindungen an und <b>netstat -46</b> zeigt IPv4 und IPv6 gleichzeitig an.) |



### Literaturhinweise

Manpages **ip-link(8)**, **ip-address(8)**, **ip-route(8)**, **ip(8)**, **ping(8)**, **tracepath(8)**, **traceroute(8)**, **ss(8)** und **netstat(8)**

Weitere Informationen finden Sie im Handbuch *Configuring and Managing Networking* unter

[https://access.redhat.com/documentation/en-us/red\\_hat\\_enterprise\\_linux/8/html-single/configuring\\_and\\_managing\\_networking/](https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux/8/html-single/configuring_and_managing_networking/)

## ► Angeleitete Übung

# Validieren der Netzwerkkonfiguration

In dieser Übung überprüfen Sie die Netzwerkkonfiguration eines Ihrer Server.

## Ergebnisse

Identifizieren Sie die aktuellen Netzwerkschnittstellen und die grundlegenden Netzwerkadressen.

## Bevor Sie Beginnen

Melden Sie sich als Benutzer **student** mit dem Passwort **student** bei **workstation** an.

Führen Sie auf **workstation** den Befehl **lab net-validate start** aus. Dieser Befehl führt ein Startskript aus, das ermittelt, ob der Host **servera** im Netzwerk erreichbar ist.

```
[student@workstation ~]$ lab net-validate start
```

- 1. Melden Sie sich mit dem Befehl **ssh** bei **servera** als Benutzer **student** an. Die Systeme sind für die Verwendung von SSH-Schlüsseln zur Authentifizierung und dem passwordlosen Zugriff auf **servera** konfiguriert.

```
[student@workstation ~]$ ssh student@servera  
...output omitted...  
[student@servera ~]$
```



### Wichtig

Die Namen der Netzwerkschnittstellen werden durch den Bustyp und die Erkennungsreihenfolge der Geräte während des Bootvorgangs bestimmt. Die Namen Ihrer Netzwerkschnittstellen variieren je nach Kursplattform und verwendeter Hardware.

Suchen Sie jetzt auf Ihrem System den Schnittstellennamen (wie **ens06** oder **en1p2**), der der Ethernet-Adresse **52:54:00:00:fa:0a** zugeordnet ist. Verwenden Sie diesen Schnittstellennamen, um den Platzhalter **enX** in dieser Übung zu ersetzen.

Suchen Sie den Netzwerkschnittstellennamen, der der Ethernet-Adresse **52:54:00:00:fa:0a** zugeordnet ist. Notieren oder merken Sie sich diesen Namen und ersetzen Sie damit den Platzhalter **enX** in den nachfolgenden Befehlen.

```
[student@servera ~]$ ip link
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default
    qlen 1000
        link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
2: enX: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group
    default qlen 1000
        link/ether 52:54:00:00:fa:0a brd ff:ff:ff:ff:ff:ff
```

- 3. Zeigen Sie die aktuelle IP-Adresse und die Netzmaske für alle Schnittstellen an.

```
[student@servera ~]$ ip addr
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default
    qlen 1000
        link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
            inet 127.0.0.1/8 scope host lo
                valid_lft forever preferred_lft forever
            inet6 ::1/128 scope host
                valid_lft forever preferred_lft forever
2: enX: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group
    default qlen 1000
        link/ether 52:54:00:00:fa:0a brd ff:ff:ff:ff:ff:ff
            inet 172.25.250.10/24 brd 172.25.250.255 scope global noprefixroute ens3
                valid_lft forever preferred_lft forever
            inet6 fe80::3059:5462:198:58b2/64 scope link noprefixroute
                valid_lft forever preferred_lft forever
```

- 4. Zeigen Sie die Statistik für die Schnittstelle **enX** an.

```
[student@servera ~]$ ip -s link show enX
2: enX: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP mode
    DEFAULT group default qlen 1000
        link/ether 52:54:00:00:fa:0a brd ff:ff:ff:ff:ff:ff
        RX: bytes packets errors dropped overrun mcast
            89014225 168251 0 154418 0 0
        TX: bytes packets errors dropped carrier collsns
            608808 6090 0 0 0 0
```

- 5. Zeigen Sie die Routing-Informationen an.

```
[student@servera ~]$ ip route
default via 172.25.250.254 dev enX proto static metric 100
172.25.250.0/24 dev enX proto kernel scope link src 172.25.250.10 metric 100
```

- 6. Stellen Sie sicher, dass der Router verfügbar ist.

```
[student@servera ~]$ ping -c3 172.25.250.254
PING 172.25.250.254 (172.25.250.254) 56(84) bytes of data.
64 bytes from 172.25.250.254: icmp_seq=1 ttl=64 time=0.196 ms
64 bytes from 172.25.250.254: icmp_seq=2 ttl=64 time=0.436 ms
64 bytes from 172.25.250.254: icmp_seq=3 ttl=64 time=0.361 ms
```

```
--- 172.25.250.254 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 49ms
rtt min/avg/max/mdev = 0.196/0.331/0.436/0.100 ms
```

- 7. Rufen Sie alle Hops zwischen dem lokalen System und **classroom.example.com** auf.

```
[student@servera ~]$ tracepath classroom.example.com
 1?: [LOCALHOST]                                pmtu 1500
 1:  workstation.lab.example.com                0.270ms
 1:  workstation.lab.example.com                0.167ms
 2:  classroom.example.com                     0.473ms reached
Resume: pmtu 1500 hops 2 back 2
```

- 8. Rufen Sie die abhörenden TCP-Sockets auf dem lokalen System auf.

```
[student@servera ~]$ ss -lt
State      Recv-Q Send-Q      Local Address:Port        Peer Address:Port
LISTEN      0      128          0.0.0.0:sunrpc        0.0.0.0:*
LISTEN      0      128          0.0.0.0:ssh           0.0.0.0:*
LISTEN      0      128          [::]:sunrpc          [::]:*
LISTEN      0      128          [::]:ssh             [::]:*
```

- 9. Beenden Sie **servera**.

```
[student@servera ~]$ exit
logout
Connection to servera closed.
[student@workstation ~]$
```

## Beenden

Führen Sie auf **workstation** das Skript **lab net-validate finish** aus, um diese Übung abzuschließen.

```
[student@workstation ~]$ lab net-validate finish
```

Hiermit ist die angeleitete Übung beendet.

# Konfigurieren von Netzwerken über die Befehlszeile

---

## Ziele

Nach Abschluss dieses Abschnitts sollten Sie Netzwerkeinstellungen und -geräte mit dem Befehl **nmcli** verwalten können.

## Beschreiben von NetworkManager-Konzepten

NetworkManager ist ein Daemon zur Überwachung und Verwaltung von Netzwerkeinstellungen. Zusätzlich zum Daemon gibt es ein Applet für das GNOME-Benachrichtigungsfeld, das Informationen über den Netzwerkstatus bereitstellt. Befehlszeilen- sowie grafische Tools kommunizieren mit NetworkManager und speichern Konfigurationsdateien im Verzeichnis **/etc/sysconfig/network-scripts**.

- Ein **Gerät** ist eine Netzwerkschnittstelle.
- Eine **Verbindung** ist eine Zusammenstellung von Einstellungen, die für ein Gerät konfiguriert werden können.
- Pro Gerät kann immer nur eine Verbindung gleichzeitig *aktiv* sein. Es können mehrere Verbindungen vorhanden sein, die von verschiedenen Geräten verwendet werden oder die Änderung von Konfigurationen für dasselbe Gerät ermöglichen. Wenn Sie die Netzwerkeinstellungen vorübergehend ändern müssen, können Sie ändern, welche Verbindung für ein Gerät aktiv ist, statt die Konfiguration einer Verbindung zu ändern. Beispielsweise kann ein Gerät für eine WLAN-Netzwerkschnittstelle auf einem Laptop verschiedene Verbindungen für das drahtlose Netzwerk am Arbeitsplatz und für das drahtlose Netzwerk zu Hause verwenden.
- Jede Verbindung wird durch einen *Namen* oder eine ID identifiziert.
- Mit dem Dienstprogramm **nmcli** können Sie Verbindungsdateien über die Befehlszeile erstellen und bearbeiten.

## Anzeigen von Netzwerkinformationen

Der Befehl **nmcli dev status** zeigt den Status aller Netzwerkgeräte an:

```
[user@host ~]$ nmcli dev status
DEVICE  TYPE      STATE       CONNECTION
eno1    ethernet  connected   eno1
ens3    ethernet  connected   static-ens3
eno2    ethernet  disconnected --
lo     loopback  unmanaged   --
```

Der Befehl **nmcli con show** zeigt eine Liste aller Verbindungen an. Um nur die aktiven Verbindungen aufzulisten, fügen Sie die Option **--active** hinzu.

```
[user@host ~]$ nmcli con show
NAME      UUID                                  TYPE      DEVICE
eno2      ff9f7d69-db83-4fed-9f32-939f8b5f81cd 802-3-ethernet --
```

```
static-ens3 72ca57a2-f780-40da-b146-99f71c431e2b 802-3-ethernet ens3
eno1       87b53c56-1f5d-4a29-a869-8a7bdaf56dfa 802-3-ethernet eno1
[user@host ~]$ nmcli con show --active
NAME          UUID                                  TYPE      DEVICE
static-ens3   72ca57a2-f780-40da-b146-99f71c431e2b 802-3-ethernet ens3
eno1         87b53c56-1f5d-4a29-a869-8a7bdaf56dfa 802-3-ethernet eno1
```

## Hinzufügen einer Netzwerkverbindung

Mit dem Befehl **nmcli con add** können Sie neue Netzwerkverbindungen hinzufügen. Bei den folgenden **nmcli con add**-Beispielbefehlen wird davon ausgegangen, dass der Name der hinzugefügten Netzwerkverbindung noch nicht verwendet wird.

Der folgende Befehl fügt eine neue Verbindung namens **eno2** für die Schnittstelle **eno2** hinzu, die IPv4-Netzwerkinformationen mit DHCP abruft und beim Start automatisch eine Verbindung herstellt. Durch Überwachen auf Router-Advertisements am lokalen Link ruft der Befehl außerdem IPv6-Netzwerkeinstellungen ab. Der Name der Konfigurationsdatei basiert auf dem Wert der Option **con-name**, **eno2**, und wird in der Datei **/etc/sysconfig/network-scripts/ifcfg-eno2** gespeichert.

```
[root@host ~]# nmcli con add con-name eno2 type ethernet ifname eno2
```

Im nächsten Beispiel wird die Verbindung **eno2** für das Gerät **eno2** mit einer statischen IPv4-Adresse erstellt, welche die IPv4-Adresse und das Netzwerkpräfix **192.168.0.5/24** und das Standard-Gateway **192.168.0.254** verwendet. Sie verbindet sich jedoch trotzdem automatisch beim Start und speichert die Konfiguration in derselben Datei. Aufgrund von Einschränkungen der Bildschirmgröße beenden Sie die erste Zeile mit einem Shell-Escape-Zeichen (\) und schließen Sie den Befehl in der nächsten Zeile ab.

```
[root@host ~]# nmcli con add con-name eno2 type ethernet ifname eno2 \
    ipv4.address 192.168.0.5/24 ipv4.gateway 192.168.0.254
```

Im letzten Beispiel wird die Verbindung **eno2** für das Gerät **eno2** mit statischen IPv6- und IPv4-Adressen erstellt, welche die IPv6-Adresse und das Netzwerkpräfix **2001:db8:0:1::c000:207/64** und das Standard-IPv6-Gateway **2001:db8:0:1::1** sowie die IPv4-Adresse und das Netzwerkpräfix **192.0.2.7/24** und das Standard-IPv4-Gateway **192.0.2.1** verwendet. Sie verbindet sich jedoch trotzdem automatisch beim Start und speichert die Konfiguration in **/etc/sysconfig/network-scripts/ifcfg-eno2**. Aufgrund von Einschränkungen der Bildschirmgröße beenden Sie die erste Zeile mit einem Shell-Escape-Zeichen (\) und schließen Sie den Befehl in der nächsten Zeile ab.

```
[root@host ~]# nmcli con add con-name eno2 type ethernet ifname eno2 \
    ipv6.address 2001:db8:0:1::c000:207/64 ipv6.gateway 2001:db8:0:1::1 \
    ipv4.address 192.0.2.7/24 ipv4.gateway 192.0.2.1
```

## Kontrollieren von Netzwerkverbindungen

Der Befehl **nmcli con up name** aktiviert die Verbindung *name* auf der Netzwerkschnittstelle, an die sie gebunden ist. Beachten Sie, dass der Befehl den Namen einer *Verbindung*, nicht den Namen der Netzwerkschnittstelle verwendet. Denken Sie daran, dass der Befehl **nmcli con show** die Namen aller verfügbaren Verbindungen auflistet.

```
[root@host ~]# nmcli con up static-ens3
```

Der Befehl **nmcli dev disconnect device** trennt das Netzwerkschnittstellengerät und fährt es herunter. Der Befehl kann mit **nmcli dev dis device** abgekürzt werden:

```
[root@host ~]# nmcli dev dis ens3
```



### Wichtig

Deaktivieren Sie mit **nmcli dev dis device** eine Netzwerkschnittstelle.

Der Befehl **nmcli con down name** ist in der Regel nicht die beste Möglichkeit, eine Netzwerkschnittstelle zu deaktivieren, da er die Verbindung beendet.

Standardmäßig werden jedoch die meisten vernetzten Systemverbindungen mit aktiviertem **autoconnect** konfiguriert. Dann wird die Verbindung aktiviert, sobald die Netzwerkschnittstelle verfügbar ist. Da die Netzwerkschnittstelle der Verbindung weiterhin verfügbar ist, fährt **nmcli con down name** die Schnittstelle herunter, aber NetworkManager fährt sie sofort wieder hoch, wenn die Verbindung nicht gänzlich von der Schnittstelle getrennt wird.

## Ändern von Netzwerkverbindungseinstellungen

NetworkManager-Verbindungen haben zwei Arten von Einstellungen. Es gibt *statische* Verbindungseigenschaften, die vom Administrator konfiguriert und in den Konfigurationsdateien in **/etc/sysconfig/network-scripts/ifcfg-\*** gespeichert werden. Darüber hinaus kann es *aktive* Verbindungsdaten geben, welche die Verbindung von einem DHCP-Server abrufen und nicht persistent gespeichert werden.

Um die aktuellen Einstellungen für eine Verbindung aufzulisten, führen Sie den Befehl **nmcli con show name** aus. Hierbei ist *name* der Name der Verbindung. Einstellungen in Kleinschreibung sind statische Eigenschaften, die der Administrator ändern kann. Einstellungen in Großbuchstaben sind aktive Einstellungen, die temporär für diese Instanz der Verbindung verwendet werden.

```
[root@host ~]# nmcli con show static-ens3
connection.id:                      static-ens3
connection.uuid:                     87b53c56-1f5d-4a29-a869-8a7bdaf56dfa
connection.interface-name:           --
connection.type:                     802-3-ethernet
connection.autoconnect:              yes
connection.timestamp:                1401803453
connection.read-only:                no
connection.permissions:
connection.zone:                    --
connection.master:                  --
connection.slave-type:               --
connection.secondaries:
connection.gateway-ping-timeout:    0
802-3-ethernet.port:                --
802-3-ethernet.speed:               0
802-3-ethernet.duplex:              --
802-3-ethernet.auto-negotiate:     yes
802-3-ethernet.mac-address:        CA:9D:E9:2A:CE:F0
```

```

802-3-ethernet.cloned-mac-address:      --
802-3-ethernet.mac-address-blacklist:
802-3-ethernet.mtu:                     auto
802-3-ethernet.s390-subchannels:
802-3-ethernet.s390-nettype:            --
802-3-ethernet.s390-options:
  ipv4.method:                          manual
  ipv4.dns:                            192.168.0.254
  ipv4.dns-search:                     example.com
  ipv4.addresses:                      { ip = 192.168.0.2/24, gw =
    192.168.0.254 }
  ipv4.routes:
  ipv4.ignore-auto-routes:             no
  ipv4.ignore-auto-dns:               no
  ipv4.dhcp-client-id:                --
  ipv4.dhcp-send-hostname:            yes
  ipv4.dhcp-hostname:                 --
  ipv4.never-default:                 no
  ipv4.may-fail:                      yes
  ipv6.method:                        manual
  ipv6.dns:                           2001:4860:4860::8888
  ipv6.dns-search:                   example.com
  ipv6.addresses:                    { ip = 2001:db8:0:1::7/64, gw =
    2001:db8:0:1::1 }
  ipv6.routes:
  ipv6.ignore-auto-routes:           no
  ipv6.ignore-auto-dns:              no
  ipv6.never-default:                no
  ipv6.may-fail:                     yes
  ipv6.ip6-privacy:                  -1 (unknown)
  ipv6.dhcp-hostname:                --
...output omitted...

```

Mit dem Befehl **nmcli con mod name** können die Einstellungen für eine Verbindung geändert werden. Die Änderungen werden auch in der Datei **/etc/sysconfig/network-scripts/ifcfg-name** für die Verbindung gespeichert. Verfügbare Einstellungen sind in der Manpage **nm-settings(5)** dokumentiert.

So legen Sie die IPv4-Adresse für die Verbindung **static-ens3** auf **192.0.2.2/24** und das Standard-Gateway auf **192.0.2.254** fest:

```
[root@host ~]# nmcli con mod static-ens3 ipv4.address 192.0.2.2/24 \
  ipv4.gateway 192.0.2.254
```

So legen Sie die IPv6-Adresse für die Verbindung **static-ens3** auf **2001:db8:0:1::a00:1/64** und das Standard-Gateway auf **2001:db8:0:1::1** fest:

```
[root@host ~]# nmcli con mod static-ens3 ipv6.address 2001:db8:0:1::a00:1/64 \
  ipv6.gateway 2001:db8:0:1::1
```



### Wichtig

Wenn eine Verbindung, die ihre IPv4-Informationen bisher über einen DHCPv4-Server bezieht, nach einer Änderung dieselben Informationen nun von statischen Konfigurationsdateien abruft, sollte auch die Einstellung **ipv4.method** von **auto** in **manual** geändert werden.

Wenn eine Verbindung, die ihre IPv6-Informationen bisher über SLAAC oder einen DHCPv6-Server bezieht, nach einer Änderung dieselben Informationen nur von statischen Konfigurationsdateien abruft, sollte entsprechend die Einstellung **ipv6.method** von **auto** oder **dhcp** in **manual** geändert werden.

Andernfalls kann es passieren, dass die Verbindung nach ihrer Aktivierung hängt, nicht erfolgreich abgeschlossen wird oder neben der statischen Adresse eine IPv4-Adresse von DHCP oder eine IPv6-Adresse von DHCPv6 oder SLAAC bezieht.

Eine Reihe von Einstellungen kann mehrere Werte haben. Es ist möglich, der Liste spezifische Werte hinzuzufügen oder Werte zu entfernen. Hierfür wird vor den Einstellungsname ein **+**- oder **--**-Symbol gesetzt.

## Löschen einer Netzwerkverbindung

Der Befehl **nmcli con del name** löscht die Verbindung *name* aus dem System. Dabei wird sie vom Gerät getrennt. Die Datei **/etc/sysconfig/network-scripts/ifcfg-name** wird entfernt.

```
[root@host ~]# nmcli con del static-ens3
```

## Wer kann Netzwerkeinstellungen ändern?

Der Benutzer **root** kann alle erforderlichen Änderungen an der Netzwerkkonfiguration mit **nmcli** vornehmen.

Reguläre Benutzer, die bei der lokalen Konsole angemeldet sind, können jedoch auch viele Netzwerkkonfigurationsänderungen am System vornehmen. Sie müssen sich über die Systemtastatur entweder bei einer textbasierten virtuellen Konsole oder bei der grafischen Desktopumgebung anmelden, um diese Steuerung zu erhalten. Die Logik dahinter ist, dass jemand, der physisch an der Konsole des Computers anwesend ist, ihn wahrscheinlich als Workstation oder Laptop verwendet und er möglicherweise drahtlose oder drahtgebundene Netzwerkschnittstellen nach Belieben konfigurieren, aktivieren und deaktivieren muss. Wenn das System allerdings ein Server im Rechenzentrum ist, sollten im Allgemeinen die einzigen Benutzer, die sich lokal an dem Rechner selbst anmelden, Administratoren sein.

Reguläre Benutzer, die sich mit **ssh** anmelden, haben keinen Zugriff, um Netzwerkberechtigungen zu ändern, ohne **root** zu werden.

Sie können mit dem Befehl **nmcli gen permissions** Ihre aktuellen Berechtigungen anzeigen.

## Übersicht der Befehle

In der folgenden Tabelle sind die in diesem Abschnitt erläuterten wichtigsten **nmcli**-Befehle aufgeführt.

| Befehl                                    | Zweck                                                                                     |
|-------------------------------------------|-------------------------------------------------------------------------------------------|
| <b>nmcli dev status</b>                   | Zeigt den NetworkManager-Status aller Netzwerkschnittstellen an.                          |
| <b>nmcli con show</b>                     | Zeigt alle Verbindungen an.                                                               |
| <b>nmcli con show <i>name</i></b>         | Listet die aktuellen Einstellungen für die Verbindung <i>name</i> auf.                    |
| <b>nmcli con add con-name <i>name</i></b> | Fügt die neue Verbindung <i>name</i> hinzu.                                               |
| <b>nmcli con mod <i>name</i></b>          | Ändert die Verbindung <i>name</i> .                                                       |
| <b>nmcli con reload</b>                   | Lädt die Konfigurationsdateien neu (sinnvoll, wenn sie manuell bearbeitet wurden).        |
| <b>nmcli con up <i>name</i></b>           | Aktiviert die Verbindung <i>name</i> .                                                    |
| <b>nmcli dev dis <i>dev</i></b>           | Deaktiviert und trennt die aktuelle Verbindung auf der Netzwerkschnittstelle <i>dev</i> . |
| <b>nmcli con del <i>name</i></b>          | Löscht die Verbindung <i>name</i> und ihre Konfigurationsdatei.                           |



#### Literaturhinweise

Manpages **NetworkManager(8)**, **nmcli(1)**, **nmcli-examples(5)**, **nm-settings(5)**, **hostnamectl(1)**, **resolv.conf(5)**, **hostname(5)**, **ip(8)** und **ip-address(8)**

## ► Angeleitete Übung

# Konfigurieren von Netzwerken über die Befehlszeile

In dieser Übung konfigurieren Sie Netzwerkeinstellungen mit **nmcli**.

## Ergebnisse

Sie sollten nun die Konfiguration eines Systems von DHCP in statisch ändern können.

## Bevor Sie Beginnen

Melden Sie sich als Benutzer **student** mit dem Passwort **student** bei **workstation** an.

Führen Sie auf **workstation** den Befehl **lab net-configure start** aus. Dieser Befehl führt ein Startskript aus, das ermittelt, ob der Host **servera** im Netzwerk erreichbar ist.

```
[student@workstation ~]$ lab net-configure start
```



### Anmerkung

Wenn Sie vom Befehl **sudo** dazu aufgefordert werden, das Passwort für **student** einzugeben, dann geben Sie **student** ein.

- 1. Melden Sie sich mit dem Befehl **ssh** bei **servera** als Benutzer **student** an. Die Systeme sind für die Verwendung von SSH-Schlüsseln zur Authentifizierung konfiguriert. Daher ist kein Passwort erforderlich, um sich auf **servera** anzumelden.

```
[student@workstation ~]$ ssh student@servera
...output omitted...
[student@servera ~]$
```

- 2. Suchen Sie die Namen der Netzwerkschnittstellen.



### Wichtig

Die Namen der Netzwerkschnittstellen werden durch den Bustyp und die Erkennungsreihenfolge der Geräte während des Bootvorgangs bestimmt. Die Namen Ihrer Netzwerkschnittstellen variieren je nach Kursplattform und verwendeter Hardware.

Suchen Sie jetzt auf Ihrem System den Schnittstellennamen (wie **ens06** oder **en1p2**), der der Ethernet-Adresse **52:54:00:00:fa:0a** zugeordnet ist. Verwenden Sie diesen Schnittstellennamen, um den Platzhalter **enX** in dieser Übung zu ersetzen.

Suchen Sie den Netzwerkschnittstellennamen, der der Ethernet-Adresse **52:54:00:00:fa:0a** zugeordnet ist. Notieren oder merken Sie sich diesen Namen und ersetzen Sie damit den Platzhalter **enX** in den nachfolgenden Befehlen.

```
[student@servera ~]$ ip link
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default
    qlen 1000
        link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
2: enX: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default
    qlen 1000
        link/ether 52:54:00:00:fa:0a brd ff:ff:ff:ff:ff:ff
```

► 3. Zeigen Sie die Netzwerkeinstellungen mit **nmcli** an.

3.1. Zeigen Sie alle Verbindungen an.

```
[student@servera ~]$ nmcli con show
NAME           UUID                                  TYPE      DEVICE
Wired connection 1  03da038a-3257-4722-a478-53055cc90128  ethernet  enX
```

3.2. Zeigen Sie nur die aktive Verbindung an.

Der Name Ihrer Netzwerkschnittstelle sollte unter **DEVICE** angezeigt werden und der Name der Verbindung, die für dieses Gerät aktiv ist, ist in derselben Zeile unter **NAME** aufgeführt. Diese Übung setzt voraus, dass die aktive Verbindung **Wired connection 1** ist.

Falls der Name der aktiven Verbindung abweicht, verwenden Sie diesen anstelle von **Wired connection 1** für den Rest dieser Übung.

```
[student@servera ~]$ nmcli con show --active
NAME           UUID                                  TYPE      DEVICE
Wired connection 1  03da038a-3257-4722-a478-53055cc90128  ethernet  enX
```

3.3. Zeigen Sie alle Konfigurationseinstellungen für die aktive Verbindung an.

```
[student@servera ~]$ nmcli con show "Wired connection 1"
connection.id:            Wired connection 1
connection.uuid:          03da038a-3257-4722-a478-53055cc90128
connection.stable-id:     --
connection.type:          802-3-ethernet
connection.interface-name: --
connection.autoconnect:   yes
...output omitted...
ipv4.method:              manual
ipv4.dns:                 172.25.250.254
ipv4.dns-search:          lab.example.com,example.com
ipv4.dns-options:         ""
ipv4.dns-priority:        0
ipv4.addresses:           172.25.250.10/24
ipv4.gateway:             172.25.250.254
...output omitted...
GENERAL.NAME:             Wired connection 1
```

```

GENERAL.UUID:          03da038a-3257-4722-a478-53055cc90128
GENERAL.DEVICES:      enx
GENERAL.STATE:        activated
GENERAL.DEFAULT:      yes
GENERAL.DEFAULT6:     no
GENERAL.SPEC-OBJECT:  --
GENERAL.VPN:          no
GENERAL.DBUS-PATH:    /org/freedesktop/NetworkManager/ActiveConnection/1
GENERAL.CON-PATH:     /org/freedesktop/NetworkManager/Settings/1
GENERAL.ZONE:         --
GENERAL.MASTER-PATH:  --
IP4.ADDRESS[1]:       172.25.250.10/24
IP4.GATEWAY:          172.25.250.254
IP4.ROUTE[1]:         dst = 172.25.250.0/24, nh = 0.0.0.0, mt = 100
IP4.ROUTE[2]:         dst = 0.0.0.0/0, nh = 172.25.250.254, mt = 100
IP4.DNS[1]:           172.25.250.254
IP6.ADDRESS[1]:       fe80::3059:5462:198:58b2/64
IP6.GATEWAY:          --
IP6.ROUTE[1]:         dst = fe80::/64, nh = ::, mt = 100
IP6.ROUTE[2]:         dst = ff00::/8, nh = ::, mt = 256, table=255

```

Drücken Sie **q**, um den Befehl zu beenden.

### 3.4. Zeigen Sie den Gerätestatus an.

```
[student@servera ~]$ nmcli dev status
DEVICE  TYPE      STATE      CONNECTION
enX     ethernet  connected  Wired connection 1
lo      loopback  unmanaged  --
```

### 3.5. Zeigen Sie die Einstellungen für das Gerät **enX** an.

```
[student@servera ~]$ nmcli dev show enX
GENERAL.DEVICE:          enx
GENERAL.TYPE:            ethernet
GENERAL.HWADDR:          52:54:00:00:FA:0A
GENERAL.MTU:              1500
GENERAL.STATE:            100 (connected)
GENERAL.CONNECTION:       Wired connection 1
GENERAL.CON-PATH:         /org/freedesktop/NetworkManager/ActiveConnection/1
WIRED-PROPERTIES.CARRIER: on
IP4.ADDRESS[1]:           172.25.250.10/24
IP4.GATEWAY:              172.25.250.254
IP4.ROUTE[1]:             dst = 172.25.250.0/24, nh = 0.0.0.0, mt = 100
IP4.ROUTE[2]:             dst = 0.0.0.0/0, nh = 172.25.250.254, mt = 100
IP4.DNS[1]:               172.25.250.254
IP6.ADDRESS[1]:           fe80::3059:5462:198:58b2/64
IP6.GATEWAY:              --
IP6.ROUTE[1]:             dst = fe80::/64, nh = ::, mt = 100
IP6.ROUTE[2]:             dst = ff00::/8, nh = ::, mt = 256, table=255
```

- ▶ 4. Erstellen Sie eine statische Verbindung mit derselben IPv4-Adresse, demselben Netzwerkpräfix und demselben Standard-Gateway. Nennen Sie die neue Verbindung *static-addr*.

**Warnung**

Da der Zugriff auf Ihr System über die primäre Netzwerkverbindung erfolgt, kann die Einstellung falscher Werte dazu führen, dass Ihr Rechner unerreichbar wird. Wenn dies eintritt, verwenden Sie die Schaltfläche **Reset**, die sich über der ehemals grafischen Anzeige Ihres Rechners befindet, und versuchen Sie es erneut.

```
[student@servera ~]$ sudo nmcli con add con-name "static-addr" ifname enx \
type ethernet ipv4.method manual \
ipv4.address 172.25.250.10/24 ipv4.gateway 172.25.250.254
Connection 'static-addr' (15aa3901-555d-40cb-94c6-cea6f9151634) successfully
added.
```

- 5. Modifizieren Sie die neue Verbindung, um die DNS-Einstellungen hinzuzufügen.

```
[student@servera ~]$ sudo nmcli con mod "static-addr" ipv4.dns 172.25.250.254
```

- 6. Rufen Sie die neue Verbindung auf, und aktivieren Sie sie.

- 6.1. Rufen Sie alle Verbindungen auf.

```
[student@servera ~]$ nmcli con show
NAME           UUID                                  TYPE      DEVICE
Wired connection 1  03da038a-3257-4722-a478-53055cc90128  ethernet  enX
static-addr      15aa3901-555d-40cb-94c6-cea6f9151634  ethernet  --
```

- 6.2. Zeigen Sie die aktive Verbindung an.

```
[student@servera ~]$ nmcli con show --active
NAME           UUID                                  TYPE      DEVICE
Wired connection 1  03da038a-3257-4722-a478-53055cc90128  ethernet  enX
```

- 6.3. Aktivieren Sie die neue Verbindung **static-addr**.

```
[student@servera ~]$ sudo nmcli con up "static-addr"
Connection successfully activated (D-Bus active path: /org/freedesktop/
NetworkManager/ActiveConnection/2)
```

- 6.4. Prüfen Sie die neue aktive Verbindung.

```
[student@servera ~]$ nmcli con show --active
NAME           UUID                                  TYPE      DEVICE
static-addr      15aa3901-555d-40cb-94c6-cea6f9151634  ethernet  enx
```

- 7. Konfigurieren Sie die Originalverbindung so, dass sie nicht beim Booten des Systems ausgeführt wird, und verifizieren Sie, dass die statische Verbindung beim Booten des Systems verwendet wird.

**Kapitel 12 |** Netzwerkmanagement

- 7.1. Deaktivieren Sie den automatischen Verbindungsauftakt der Verbindung beim Systemstart.

```
[student@servera ~]$ sudo nmcli con mod "Wired connection 1" \
connection.autoconnect no
```

- 7.2. Starten Sie das System neu.

```
[student@servera ~]$ sudo systemctl reboot
Connection to servera closed by remote host.
Connection to servera closed.
[student@workstation ~]$
```

- 7.3. Zeigen Sie die aktive Verbindung an.

```
[student@workstation ~]$ ssh student@servera
...output omitted...
[student@servera ~]$ nmcli con show --active
NAME           UUID                                  TYPE      DEVICE
static-addr   15aa3901-555d-40cb-94c6-cea6f9151634  ethernet  enX
```

► **8.** Testen Sie die Verbindung mit den neuen Netzwerkadressen.

- 8.1. Verifizieren Sie die IP-Adresse.

```
[student@servera ~]$ ip addr show enX
2: enX: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group
default qlen 1000
    link/ether 52:54:00:00:fa:0a brd ff:ff:ff:ff:ff:ff
    inet 172.25.250.10/24 brd 172.25.250.255 scope global noprefixroute enX
        valid_lft forever preferred_lft forever
    inet6 fe80::6556:cdd9:ce15:1484/64 scope link noprefixroute
        valid_lft forever preferred_lft forever
```

- 8.2. Verifizieren Sie das Standard-Gateway.

```
[student@servera ~]$ ip route
default via 172.25.250.254 dev enX proto static metric 100
172.25.250.0/24 dev enX proto kernel scope link src 172.25.250.10 metric 100
```

- 8.3. Pingen Sie die DNS-Adresse.

```
[student@servera ~]$ ping -c3 172.25.250.254
PING 172.25.250.254 (172.25.250.254) 56(84) bytes of data.
64 bytes from 172.25.250.254: icmp_seq=1 ttl=64 time=0.225 ms
64 bytes from 172.25.250.254: icmp_seq=2 ttl=64 time=0.314 ms
64 bytes from 172.25.250.254: icmp_seq=3 ttl=64 time=0.472 ms

--- 172.25.250.254 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 46ms
rtt min/avg/max/mdev = 0.225/0.337/0.472/0.102 ms
```

8.4. Beenden Sie **servera**.

```
[student@servera ~]$ exit  
logout  
Connection to servera closed.  
[student@workstation ~]$
```

## Beenden

Führen Sie auf **workstation** das Skript **lab net-configure finish** aus, um diese Übung abzuschließen.

```
[student@workstation ~]$ lab net-configure finish
```

Hiermit ist die angeleitete Übung beendet.

# Bearbeiten der Netzwerkkonfigurationsdateien

---

## Ziele

Nach Abschluss dieses Abschnitts sollten Sie die Netzwerkkonfiguration durch Bearbeiten der Konfigurationsdateien modifizieren können.

## Beschreiben der Verbindungskonfigurationsdateien

Standardmäßig werden Änderungen, die mit `nmcli con mod name` vorgenommen wurden, automatisch in `/etc/sysconfig/network-scripts/ifcfg-name` gespeichert. Die Datei kann auch manuell mit einem Texteditor bearbeitet werden. Führen Sie anschließend `nmcli con reload` aus, sodass NetworkManager die Konfigurationsänderungen liest.

Aus Gründen der Rückwärtskompatibilität haben die in dieser Datei gespeicherten Anweisungen einen anderen Namen und eine andere Syntax als die `nm-settings(5)`-Namen. In der folgenden Tabelle sind einige der wichtigen Einstellungsnamen `ifcfg-*`-Anweisungen zugeordnet.

### Vergleich von nm-Einstellungen und ifcfg--Anweisungen

| <code>nmcli con mod</code>               | <code>ifcfg-* file</code>                   | <code>Auswirkung</code>                                                                                                                                                                                                                                                                                               |
|------------------------------------------|---------------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <code>ipv4.method manual</code>          | <code>BOOTPROTO=none</code>                 | Statisch konfigurierte IPv4-Adressen.                                                                                                                                                                                                                                                                                 |
| <code>ipv4.method auto</code>            | <code>BOOTPROTO=dhcp</code>                 | Sucht nach Konfigurationseinstellungen von einem DHCPv4-Server. Wenn statische Adressen festgelegt sind, werden sie erst abgerufen, wenn Informationen vom DHCPv4 eingehen.                                                                                                                                           |
| <code>ipv4.addresses 192.0.2.1/24</code> | <code>IPADDR=192.0.2.1<br/>PREFIX=24</code> | Legt die statische IPv4-Adresse und das Netzwerkpräfix fest. Wenn mehr als eine Adresse für die Verbindung festgelegt ist, wird die zweite Adresse durch die Direktiven <code>IPADDR1</code> und <code>PREFIX1</code> , die dritte durch die Direktiven <code>IPADDR2</code> und <code>PREFIX2</code> usw. definiert. |
| <code>ipv4.gateway 192.0.2.254</code>    | <code>GATEWAY=192.0.2.254</code>            | Legt das Standard-Gateway fest.                                                                                                                                                                                                                                                                                       |
| <code>ipv4.dns 8.8.8.8</code>            | <code>DNS1=8.8.8.8</code>                   | Bearbeitet <code>/etc/resolv.conf</code> für die Verwendung dieses <code>nameserver</code> .                                                                                                                                                                                                                          |

| <b>nmcli con mod</b>                 | <b>ifcfg-* file</b>                     | <b>Auswirkung</b>                                                                                                                                                                                                                                                                       |
|--------------------------------------|-----------------------------------------|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>ipv4.dns-search example.com</b>   | <b>DOMAIN=example.com</b>               | Bearbeitet <b>/etc/resolv.conf</b> für die Verwendung dieser Domain in der Anweisung <b>search</b> .                                                                                                                                                                                    |
| <b>ipv4.ignore-auto-dns true</b>     | <b>PEERDNS=no</b>                       | Ignoriert DNS-Serverinformationen vom DHCP-Server.                                                                                                                                                                                                                                      |
| <b>ipv6.method manual</b>            | <b>IPV6_AUTOCONF=no</b>                 | Statisch konfigurierte IPv6-Adressen.                                                                                                                                                                                                                                                   |
| <b>ipv6.method auto</b>              | <b>IPV6_AUTOCONF=yes</b>                | Konfiguriert Netzwerkeinstellungen mit SLAAC von Router-Advertisements.                                                                                                                                                                                                                 |
| <b>ipv6.method dhcp</b>              | <b>IPV6_AUTOCONF=no<br/>DHCPV6C=yes</b> | Konfiguriert Netzwerkeinstellungen mit DHCPv6, nicht mit SLAAC.                                                                                                                                                                                                                         |
| <b>ipv6.addresses 2001:db8::a/64</b> | <b>IPV6ADDR=2001:db8::a/64</b>          | Legt die statische IPv6-Adresse und das Netzwerkpräfix fest. Wenn mehrere Adressen für die Verbindung eingerichtet sind, verwendet <b>IPV6ADDR_SECONDARIES</b> eine mit doppelten Anführungsstrichen gekennzeichnete Liste von mit Leerzeichen getrennten Adressen-/Präfixdefinitionen. |
| <b>ipv6.gateway 2001:db8::1</b>      | <b>IPV6_DEFAULTGW=2001::1</b>           | Legt das Standard-Gateway fest.                                                                                                                                                                                                                                                         |
| <b>ipv6.dns fde2:6494:1e09:2::d</b>  | <b>DNS1=fde2:6494:...:d</b>             | Bearbeitet <b>/etc/resolv.conf</b> für die Verwendung dieses Nameserver. Identisch mit IPv4.                                                                                                                                                                                            |
| <b>ipv6.dns-search example.com</b>   | <b>IPV6_DOMAIN=example.co</b>           | Bearbeitet <b>/etc/resolv.conf</b> für die Verwendung dieser Domain in der Anweisung <b>search</b> .                                                                                                                                                                                    |
| <b>ipv6.ignore-auto-dns true</b>     | <b>IPV6_PEERDNS=no</b>                  | Ignoriert DNS-Serverinformationen vom DHCP-Server.                                                                                                                                                                                                                                      |
| <b>connection.autoconnect yes</b>    | <b>ONBOOT=yes</b>                       | Diese Verbindung beim Boot-Vorgang automatisch aktivieren.                                                                                                                                                                                                                              |
| <b>connection.id ens3</b>            | <b>NAME=ens3</b>                        | Der Name dieser Verbindung.                                                                                                                                                                                                                                                             |

| <b>nmcli con mod</b>                  | <b>ifcfg-* file</b> | <b>Auswirkung</b>                                                                |
|---------------------------------------|---------------------|----------------------------------------------------------------------------------|
| <b>connection.interface-name ens3</b> | <b>DEVICE=ens3</b>  | Die Verbindung ist an die Netzwerkschnittstelle mit diesem Namen gebunden.       |
| <b>802-3-ethernet.mac-address ...</b> | <b>HWADDR=...</b>   | Die Verbindung ist an die Netzwerkschnittstelle mit dieser MAC-Adresse gebunden. |

## Ändern der Netzwerkkonfiguration

Sie können das Netzwerk auch direkt durch Bearbeiten von Verbindungskonfigurationsdateien konfigurieren. Über diese Verbindungskonfigurationsdateien werden die Softwareschnittstellen für die einzelnen Netzwerkgeräte gesteuert. Üblicherweise tragen die Dateien den Namen **/etc/sysconfig/network-scripts/ifcfg-name**, wobei *name* für den Namen des Geräts oder der Verbindung steht, das/die über die jeweilige Konfigurationsdatei gesteuert wird. In der folgenden Tabelle finden Sie Standardvariablen, die in den Konfigurationsdateien für die statische oder dynamische IPv4-Konfiguration verwendet werden.

### IPv4-Konfigurationsoptionen für die Datei ifcfg

| <b>Statisch</b>       | <b>Dynamisch</b>      | <b>Beide:</b>      |
|-----------------------|-----------------------|--------------------|
| <b>BOOTPROTO=none</b> | <b>BOOTPROTO=dhcp</b> | <b>DEVICE=ens3</b> |

Bei den statischen Einstellungen enden die Variablen für IP-Adresse, Präfix und Gateway mit einer Zahl. Dadurch können der Schnittstelle mehrere Wertesätze zugewiesen werden. Die DNS-Variable verfügt außerdem über eine Zahl, mit der die Suchreihenfolge festgelegt wird, wenn mehrere Server angegeben werden.

Nachdem Sie die Konfigurationsdatei verändert haben, führen Sie **nmcli con reload** aus, damit NetworkManager die Konfigurationsänderungen einliest. Die Schnittstelle muss danach trotzdem neu gestartet werden, damit die Änderungen vorgenommen werden.

```
[root@host ~]# nmcli con reload
[root@host ~]# nmcli con down "static-ens3"
[root@host ~]# nmcli con up "static-ens3"
```



### Literaturhinweise

Manpage **nmcli(1)**

Weitere Informationen finden Sie im Handbuch *Configuring and Managing Networking* unter

[https://access.redhat.com/documentation/en-us/red\\_hat\\_enterprise\\_linux/8/html-single/configuring\\_and\\_managing\\_networking/](https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux/8/html-single/configuring_and_managing_networking/)

## ► Angeleitete Übung

# Bearbeiten der Netzwerkkonfigurationsdateien

In dieser Übung ändern Sie die Netzwerkkonfigurationsdateien manuell und stellen sicher, dass die neuen Einstellungen wirksam werden.

## Ergebnisse

Sie sollten eine zusätzliche Netzwerkadresse für jedes System hinzufügen können.

## Bevor Sie Beginnen

Melden Sie sich als Benutzer **student** mit dem Passwort **student** bei **workstation** an.

Führen Sie auf **workstation** den Befehl **lab net-edit start** aus. Dieser Befehl führt ein Startskript aus, das ermittelt, ob die Hosts **servera** und **serverb** im Netzwerk erreichbar sind.

```
[student@workstation ~]$ lab net-edit start
```

- ▶ 1. Melden Sie sich mit dem Befehl **ssh** bei **servera** als Benutzer **student** an. Die Systeme sind für die Verwendung von SSH-Schlüsseln zur Authentifizierung konfiguriert. Daher ist kein Passwort erforderlich, um sich auf **servera** anzumelden.

```
[student@workstation ~]$ ssh student@servera
...output omitted...
[student@servera ~]$
```

- ▶ 2. Suchen Sie die Namen der Netzwerkschnittstellen.



### Wichtig

Die Namen der Netzwerkschnittstellen werden durch den Bustyp und die Erkennungsreihenfolge der Geräte während des Bootvorgangs bestimmt. Die Namen Ihrer Netzwerkschnittstellen variieren je nach Kursplattform und verwendeter Hardware.

Suchen Sie jetzt auf Ihrem System den Schnittstellennamen (wie **ens06** oder **en1p2**), der der Ethernet-Adresse **52:54:00:00:fa:0a** zugeordnet ist. Verwenden Sie diesen Schnittstellennamen, um den Platzhalter **enX** in dieser Übung zu ersetzen.

Suchen Sie den Netzwerkschnittstellennamen, der der Ethernet-Adresse **52:54:00:00:fa:0a** zugeordnet ist. Notieren oder merken Sie sich diesen Namen und ersetzen Sie damit den Platzhalter **enX** in den nachfolgenden Befehlen. Die aktive

Verbindung wird auch **Wired connection 1** genannt (und wird daher von der Datei **/etc/sysconfig/network-scripts/ifcfg-Wired\_connection\_1** verwaltet).

Wenn Sie in diesem Kapitel bereits Übungen durchgeführt haben und dies für Ihr System zutrifft, sollte dies auch für diese Übung zutreffen.

```
[student@servera ~]$ ip link
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default
    qlen 1000
        link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
2: enX: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group
    default qlen 1000
        link/ether 52:54:00:00:fa:0a brd ff:ff:ff:ff:ff:ff
[student@servera ~]$ nmcli con show --active
NAME           UUID                                  TYPE      DEVICE
Wired connection 1  03da038a-3257-4722-a478-53055cc90128  ethernet  enX
[student@servera ~]$ ls \
/etc/sysconfig/network-scripts/ifcfg-Wired_connection_1
/etc/sysconfig/network-scripts/ifcfg-Wired_connection_1
```

- 3. Wechseln Sie auf **servera** zum Benutzer **root** und bearbeiten Sie dann die Datei **/etc/sysconfig/network-scripts/ifcfg-Wired\_connection\_1**, um die zusätzliche Adresse **10.0.1.1/24** hinzuzufügen.
- 3.1. Verwenden Sie den Befehl **sudo -i**, um zum **root**-Benutzer zu wechseln. Das Passwort für den Benutzer **student** lautet **student**.

```
[student@servera ~]$ sudo -i
[sudo] password for student: student
[root@servera ~]#
```

- 3.2. Fügen Sie der Datei einen Eintrag zur Festlegung der IPv4-Adresse hinzu.

```
[root@servera ~]# echo \
"IPADDR1=10.0.1.1" >> /etc/sysconfig/network-scripts/ifcfg-Wired_connection_1
```

- 3.3. Fügen Sie der Datei einen Eintrag zur Festlegung des Netzwerkpräfixes hinzu.

```
[root@servera ~]# echo \
"PREFIX1=24" >> /etc/sysconfig/network-scripts/ifcfg-Wired_connection_1
```

- 4. Aktivieren Sie die neue Adresse.

- 4.1. Laden Sie die Konfigurationsänderungen erneut.

```
[root@servera ~]# nmcli con reload
```

- 4.2. Starten Sie die Verbindung mit den neuen Einstellungen neu.

```
[root@servera ~]# nmcli con up "Wired connection 1"
Connection successfully activated (D-Bus active path: /org/freedesktop/
NetworkManager/ActiveConnection/3)
```

- 5. Verifizieren Sie die neue IP-Adresse.

```
[root@servera ~]# ip addr show enX
2: enX: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group
    default qlen 1000
        link/ether 52:54:00:00:fa:0a brd ff:ff:ff:ff:ff:ff
        inet 172.25.250.10/24 brd 172.25.250.255 scope global noprefixroute enX
            valid_lft forever preferred_lft forever
        inet 10.0.1.1/24 brd 10.0.1.255 scope global noprefixroute enX
            valid_lft forever preferred_lft forever
        inet6 fe80::4bf3:e1d9:3076:f8d7/64 scope link noprefixroute
            valid_lft forever preferred_lft forever
```

- 6. Beenden Sie **servera** und kehren Sie als Benutzer **student** zu **workstation** zurück.

```
[root@servera ~]# exit
logout
[student@servera ~]$ exit
logout
Connection to servera closed.
[student@workstation ~]$
```

- 7. Bearbeiten Sie auf **serverb** die Datei **/etc/sysconfig/network-scripts/ifcfg-Wired\_connection\_1**, um die zusätzliche Adresse **10.0.1.2/24** hinzuzufügen. Dann laden Sie die neue Konfiguration.

- 7.1. Melden Sie sich auf **workstation** mit dem Befehl **ssh** bei **serverb** als Benutzer **student** an.

```
[student@workstation ~]$ ssh student@serverb
...output omitted...
[student@serverb ~]$
```

- 7.2. Verwenden Sie den Befehl **sudo -i**, um zum **root**-Benutzer zu wechseln. Das Passwort für den Benutzer **student** lautet **student**.

```
[student@serverb ~]$ sudo -i
[sudo] password for student: student
[root@serverb ~]#
```

- 7.3. Fügen Sie der Datei **ifcfg-Wired\_connection\_1** Einträge zur Festlegung der zweiten IPv4-Adresse und des Netzwerkpräfixes hinzu.

```
[root@serverb ~]# echo \
"IPADDR1=10.0.1.2" >> /etc/sysconfig/network-scripts/ifcfg-Wired_connection_1
[root@serverb ~]# echo \
"PREFIX1=24" >> /etc/sysconfig/network-scripts/ifcfg-Wired_connection_1
```

- 7.4. Laden Sie die Konfigurationsänderungen erneut.

```
[root@serverb ~]# nmcli con reload
```

7.5. Starten Sie die Verbindung mit den neuen Einstellungen.

```
[root@serverb ~]# nmcli con up "Wired connection 1"
Connection successfully activated (D-Bus active path: /org/freedesktop/
NetworkManager/ActiveConnection/4)
```

7.6. Verifizieren Sie die neue IP-Adresse.

```
[root@serverb ~]# ip addr show enX
2: enX: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group
    default qlen 1000
        link/ether 52:54:00:00:fa:0b brd ff:ff:ff:ff:ff:ff
        inet 172.25.250.11/24 brd 172.25.250.255 scope global noprefixroute enX
            valid_lft forever preferred_lft forever
        inet 10.0.1.2/24 brd 10.0.1.255 scope global noprefixroute enX
            valid_lft forever preferred_lft forever
        inet6 fe80::74c:3476:4113:463f/64 scope link noprefixroute
            valid_lft forever preferred_lft forever
```

► 8. Testen Sie die Verbindung mit den neuen Netzwerkadressen.

8.1. Pingen Sie von **serverb** aus die neue Adresse von **servera**.

```
[root@serverb ~]# ping -c3 10.0.1.1
PING 10.0.1.1 (10.0.1.1) 56(84) bytes of data.
64 bytes from 10.0.1.1: icmp_seq=1 ttl=64 time=0.342 ms
64 bytes from 10.0.1.1: icmp_seq=2 ttl=64 time=0.188 ms
64 bytes from 10.0.1.1: icmp_seq=3 ttl=64 time=0.317 ms

--- 10.0.1.1 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 35ms
rtt min/avg/max/mdev = 0.188/0.282/0.342/0.068 ms
```

8.2. Beenden Sie **serverb** und kehren Sie zu **workstation** zurück.

```
[root@serverb ~]# exit
logout
[student@serverb ~]$ exit
logout
Connection to serverb closed.
[student@workstation ~]$
```

8.3. Greifen Sie auf **workstation** mit dem Befehl **ssh** auf **servera** als Benutzer **student** zu, um die neue Adresse von **serverb** zu pingen.

```
[student@workstation ~]$ ssh student@servera ping -c3 10.0.1.2
PING 10.0.1.2 (10.0.1.2) 56(84) bytes of data.
64 bytes from 10.0.1.2: icmp_seq=1 ttl=64 time=0.269 ms
64 bytes from 10.0.1.2: icmp_seq=2 ttl=64 time=0.338 ms
64 bytes from 10.0.1.2: icmp_seq=3 ttl=64 time=0.361 ms
```

```
--- 10.0.1.2 ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 48ms
rtt min/avg/max/mdev = 0.269/0.322/0.361/0.044 ms
```

## Beenden

Führen Sie auf **workstation** das Skript **lab net-edit finish** aus, um diese Übung abzuschließen.

```
[student@workstation ~]$ lab net-edit finish
```

Hiermit ist die angeleitete Übung beendet.

# Konfigurieren von Hostnamen und Namensauflösung

## Ziele

Nach Abschluss dieses Abschnitts sollten Sie den statischen Hostnamen eines Servers und die Namensauflösung konfigurieren und testen können.

## Ändern des Systemhostnamens

Der Befehl **hostname** ruft den vollständigen, qualifizierten Hostnamen des Systems auf oder modifiziert diesen temporär.

```
[root@host ~]# hostname  
host.example.com
```

In der Datei **/etc/hostname** kann ein statischer Hostname festgelegt werden. Mit dem Befehl **hostnamectl** kann sowohl die Datei modifiziert als auch der Status des vollständigen, qualifizierten Hostnamens abgerufen werden. Wenn diese Datei nicht existiert, wird der Hostname durch eine umgekehrte DNS-Abfrage festgelegt, sobald der Schnittstelle eine IP-Adresse zugewiesen wurde.

```
[root@host ~]# hostnamectl set-hostname host.example.com  
[root@host ~]# hostnamectl status  
  Static hostname: host.example.com  
    Icon name: computer-vm  
      Chassis: vm  
    Machine ID: f874df04639f474cb0a9881041f4f7d4  
      Boot ID: 6a0abe03ef0b4a97bcb8afb7b281e4d3  
Virtualization: kvm  
Operating System: Red Hat Enterprise Linux 8.2 (Ootpa)  
  CPE OS Name: cpe:/o:redhat:enterprise_linux:8.2:GA  
    Kernel: Linux 4.18.0-193.el8.x86_64  
  Architecture: x86-64  
[root@host ~]# cat /etc/hostname  
host.example.com
```



### Wichtig

Bei Red Hat Enterprise Linux 7 und neueren Versionen wird der statische Hostname in **/etc/hostname** gespeichert. Bei Red Hat Enterprise Linux 6 und älteren Versionen wird der Hostname als Variable in **/etc/sysconfig/network** gespeichert.

## Konfigurieren der Namensauflösung

Der *Stub-Resolver* wird zum Umwandeln von Hostnamen in IP-Adressen und umgekehrt verwendet. Anhand der Konfiguration von wird festgelegt, wo die Datei **/etc/nsswitch.conf** gesucht werden soll. Standardmäßig wird zuerst der Inhalt der Datei **/etc/hosts** überprüft.

```
[root@host ~]# cat /etc/hosts
127.0.0.1      localhost localhost.localdomain localhost4 localhost4.localdomain4
::1            localhost localhost.localdomain localhost6 localhost6.localdomain6

172.25.254.254 classroom.example.com
172.25.254.254 content.example.com
```

Mit dem Befehl **getent hosts hostname** kann die Hostnamensauflösung durch Verwendung der Datei **/etc/hosts** getestet werden.

Ist in der Datei **/etc/hosts** kein passender Eintrag vorhanden, versucht der Stub-Resolver standardmäßig den Hostnamen anhand des DNS-Namenservers abzurufen. Die Ausführung dieser Abfrage wird über die Datei **/etc/resolv.conf** gesteuert:

- **search**: Liste der abzufragenden Domain-Namen mit kurzem Hostnamen. Diese Einstellung und **domain** sollten nicht in derselben Datei verwendet werden. Ist dies dennoch der Fall, erhält die letzte Instanz den Vorrang. Nähere Informationen hierzu finden Sie in **resolv.conf(5)**.
- **nameserver**: IP-Adresse des abzufragenden Nameservers. Es können bis zu drei Nameserver-Anweisungen angegeben werden, damit bei Ausfällen Backups zur Verfügung stehen.

```
[root@host ~]# cat /etc/resolv.conf
# Generated by NetworkManager
domain example.com
search example.com
nameserver 172.25.254.254
```

NetworkManager aktualisiert die Datei **/etc/resolv.conf** mittels DNS-Einstellungen in den Verbindungskonfigurationsdateien. Verwenden Sie den Befehl **nmcli** zum Modifizieren der Verbindungen.

```
[root@host ~]# nmcli con mod ID ipv4.dns IP
[root@host ~]# nmcli con down ID
[root@host ~]# nmcli con up ID
[root@host ~]# cat /etc/sysconfig/network-scripts/ifcfg-ID
...output omitted...
DNS1=8.8.8.8
...output omitted...
```

Standardmäßig ersetzt **nmcli con mod ID ipv4.dns IP** alle vorhergehenden DNS-Einstellungen mit der zur Verfügung gestellten neuen IP-Liste. Durch das Symbol **+** oder **-** vor dem Argument **ipv4.dns** wird ein einzelner Eintrag entweder hinzugefügt oder entfernt.

```
[root@host ~]# nmcli con mod ID +ipv4.dns IP
```

So fügen Sie den DNS-Server mit der IPv6-IP-Adresse **2001:4860:4860::8888** in die Liste der Nameserver ein, die mit der Verbindung **static-ens3** verwendet werden sollen:

```
[root@host ~]# nmcli con mod static-ens3 +ip6.dns 2001:4860:4860::8888
```



### Anmerkung

Statische IPv4 und IPv6 DNS-Einstellungen enden alle als **nameserver**-Anweisungen in **/etc/resolv.conf**. Sie sollten sicherstellen, dass (auf einem Dual-Stack-System) mindestens ein IPv4-erreichbarer Nameserver vorhanden ist. Im Idealfall sollten mindestens ein IPv4- und ein zweiter IPv6-Nameserver vorhanden sein, falls Sie Probleme mit Ihrem IPv4- oder IPv6-Netzwerk haben.

## Testen der DNS-Namensaufflösung

Mit dem Befehl **host HOSTNAME** kann die DNS-Serverkonnektivität geprüft werden.

```
[root@host ~]# host classroom.example.com
classroom.example.com has address 172.25.254.254
[root@host ~]# host 172.25.254.254
254.25.25.172.in-addr.arpa domain name pointer classroom.example.com.
```



### Wichtig

Die Datei **/etc/resolv.conf** wird von DHCP automatisch beim Starten von Schnittstellen umgeschrieben, sofern in den relevanten Schnittstellenkonfigurationsdateien nicht **PEERDNS=no** festgelegt wurde. Stellen Sie dies mit dem **nmcli**-Befehl ein.

```
[root@host ~]# nmcli con mod "static-ens3" ipv4.ignore-auto-dns yes
```



### Literaturhinweise

Manpages **nmcli(1)**, **hostnamectl(1)**, **hosts(5)**, **getent(1)**, **host(1)** und **resolv.conf(5)**

Weitere Informationen finden Sie im Handbuch *Configuring and Managing Networking* unter  
[https://access.redhat.com/documentation/en-us/red\\_hat\\_enterprise\\_linux/8/html-single/configuring\\_and\\_managing\\_networking/](https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux/8/html-single/configuring_and_managing_networking/)

## ► Angeleitete Übung

# Konfigurieren von Hostnamen und Namensauflösung

In dieser Übung konfigurieren Sie manuell den statischen Hostnamen des Systems, die `/etc/hosts`-Datei und die DNS-Namensauflösung.

## Ergebnisse

Sie können nun einen benutzerdefinierten Hostnamen festlegen und die Einstellungen für die Namensauflösung konfigurieren.

## Bevor Sie Beginnen

Melden Sie sich als Benutzer **student** mit dem Passwort **student** bei **workstation** an.

Führen Sie auf **workstation** den Befehl **lab net-hostnames start** aus. Dieser Befehl führt ein Startskript aus, das ermittelt, ob der Host **servera** im Netzwerk erreichbar ist.

```
[student@workstation ~]$ lab net-hostnames start
```

- ▶ 1. Melden Sie sich mit dem Befehl **ssh** bei **servera** als Benutzer **student** an. Die Systeme sind für die Verwendung von SSH-Schlüsseln zur Authentifizierung konfiguriert. Daher ist kein Passwort erforderlich, um sich auf **servera** anzumelden.

```
[student@workstation ~]$ ssh student@servera
...output omitted...
[student@servera ~]$
```

- ▶ 2. Rufen Sie die aktuellen Einstellungen für den Hostnamen auf.

- 2.1. Rufen Sie den aktuellen Hostnamen auf.

```
[student@servera ~]$ hostname
servera.lab.example.com
```

- 2.2. Rufen Sie den Status des Hostnamens auf.

```
[student@servera ~]$ hostnamectl status
  Static hostname: n/a
  Transient hostname: servera.lab.example.com
            Icon name: computer-vm
            Chassis: vm
      Machine ID: f874df04639f474cb0a9881041f4f7d4
        Boot ID: 22ae5279f57049678eda547bdb39a19d
  Virtualization: kvm
Operating System: Red Hat Enterprise Linux 8.2 (Ootpa)
```

```
CPE OS Name: cpe:/o:redhat:enterprise_linux:8.2:GA  
Kernel: Linux 4.18.0-193.el8.x86_64  
Architecture: x86-64
```

Notieren Sie sich den temporären Hostnamen, der über DHCP oder mDNS abgerufen wurde.

- 3. Geben Sie einen statischen Hostnamen an, der dem aktuellen, vorübergehenden Hostnamen gleicht.

- 3.1. Ändern Sie den Hostnamen und die Hostnamen-Konfigurationsdatei.

```
[student@servera ~]$ sudo hostnamectl set-hostname \  
servera.lab.example.com  
[sudo] password for student: student  
[student@servera ~]$
```

- 3.2. Sehen Sie sich den Inhalt der Datei **/etc/hostname** an, die den Hostnamen beim Netzwerkstart angibt.

```
servera.lab.example.com
```

- 3.3. Rufen Sie den Status des Hostnamens auf.

```
[student@servera ~]$ hostnamectl status  
Static hostname: servera.lab.example.com  
Icon name: computer-vm  
Chassis: vm  
Machine ID: f874df04639f474cb0a9881041f4f7d4  
Boot ID: 22ae5279f57049678eda547bdb39a19d  
Virtualization: kvm  
Operating System: Red Hat Enterprise Linux 8.2 (Ootpa)  
CPE OS Name: cpe:/o:redhat:enterprise_linux:8.2:GA  
Kernel: Linux 4.18.0-193.el8.x86_64  
Architecture: x86-64
```

Beachten Sie, dass der vorübergehende Hostname nicht angezeigt wird, nachdem ein statischer Hostname konfiguriert wurde.

- 4. Ändern Sie den Hostnamen temporär ab.

- 4.1. Ändern Sie den Hostnamen.

```
[student@servera ~]$ sudo hostname testname
```

- 4.2. Rufen Sie den aktuellen Hostnamen auf.

```
[student@servera ~]$ hostname  
testname
```

- 4.3. Sehen Sie sich den Inhalt der Datei **/etc/hostname** an, die den Hostnamen beim Netzwerkstart angibt.

```
servera.lab.example.com
```

4.4. Rebooten Sie das System.

```
[student@servera ~]$ sudo systemctl reboot  
Connection to servera closed by remote host.  
Connection to servera closed.  
[student@workstation ~]$
```

4.5. Melden Sie sich von **workstation** aus bei **servera** als Benutzer **student** an.

```
[student@workstation ~]$ ssh student@servera  
...output omitted...  
[student@servera ~]$
```

4.6. Rufen Sie den aktuellen Hostnamen auf.

```
[student@servera ~]$ hostname  
servera.lab.example.com
```

► 5. Fügen Sie einen lokalen Spitznamen für den Kursraum-Server hinzu.

5.1. Rufen Sie die IP-Adresse für classroom.example.com auf.

```
[student@servera ~]$ host classroom.example.com  
classroom.example.com has address 172.25.254.254
```

5.2. Ändern **/etc/hosts** so, dass mit dem zusätzlichen Namen von **class** auf die IP-Adresse 172.25.254.254 zugegriffen werden kann.

```
127.0.0.1 localhost localhost.localdomain localhost4 localhost4.localdomain4  
::1 localhost localhost.localdomain localhost6 localhost6.localdomain6  
  
172.25.254.254 classroom.example.com classroom class  
172.25.254.254 content.example.com content  
...content omitted...
```

5.3. Rufen Sie die IP-Adresse für **class** auf.

```
[student@servera ~]$ host class  
Host class not found: 2(SERVFAIL)  
[student@servera ~]$ getent hosts class  
172.25.254.254 classroom.example.com class
```

5.4. Pingen Sie **class**.

```
[student@servera ~]$ ping -c3 class  
PING classroom.example.com (172.25.254.254) 56(84) bytes of data.  
64 bytes from classroom.example.com (172.25.254.254): icmp_seq=1 ttl=64 time=0.397  
ms
```

```
64 bytes from classroom.example.com (172.25.254.254): icmp_seq=2 ttl=64 time=0.447
ms
64 bytes from classroom.example.com (172.25.254.254): icmp_seq=3 ttl=64 time=0.470
ms

--- classroom.example.com ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 2000ms
rtt min/avg/max/mdev = 0.397/0.438/0.470/0.030 ms
```

### 5.5. Beenden Sie **servera**.

```
[student@servera ~]$ exit
logout
Connection to servera closed.
[student@workstation ~]$
```

## Beenden

Führen Sie auf **workstation** das Skript **lab net-hostnames finish** aus, um diese Übung abzuschließen.

```
[student@workstation ~]$ lab net-hostnames finish
```

Hiermit ist die angeleitete Übung beendet.

## ► Praktische Übung

# Netzwerkmanagement

### Leistungscheckliste

In dieser Übung konfigurieren Sie die Netzwerkeinstellungen für einen Red Hat Enterprise Linux-Server.

### Ergebnisse

Sie sollten in der Lage sein, zwei statische IPv4-Adressen für die primäre Netzwerkschnittstelle zu konfigurieren.

### Bevor Sie Beginnen

Melden Sie sich als Benutzer **student** mit dem Passwort **student** bei **workstation** an.

Führen Sie auf der **workstation** den Befehl **lab net-review start** aus. Dieser Befehl führt ein Startskript aus, das ermittelt, ob der Host **serverb** im Netzwerk erreichbar ist.

```
[student@workstation ~]$ lab net-review start
```

1. Melden Sie sich mit dem Befehl **ssh** bei **serverb** als **student** an. Die Systeme sind für die Verwendung von SSH-Schlüsseln zur Authentifizierung konfiguriert. Daher ist kein Passwort erforderlich, um sich bei **servera** anzumelden.
2. Verwenden Sie den Befehl **sudo -i**, um zum **root**-Benutzer zu wechseln. Verwenden Sie bei Aufforderung **student** als Passwort.
3. Erstellen Sie mittels der Einstellungen in der Tabelle eine neue Verbindung mit einer statischen Netzwerkverbindung.

| Parameter          | Einstellung                                                                                         |
|--------------------|-----------------------------------------------------------------------------------------------------|
| Verbindungsname    | lab                                                                                                 |
| Schnittstellenname | enX (kann variieren, verwenden Sie die Schnittstelle mit der MAC-Adresse <b>52:54:00:00:fa:0b</b> ) |
| IP-Adresse         | 172.25.250.11/24                                                                                    |
| Gateway-Adresse    | 172.25.250.254                                                                                      |
| DNS-Adresse        | 172.25.250.254                                                                                      |

4. Konfigurieren Sie die neue Verbindung so, dass sie automatisch gestartet wird. Es sollen keine anderen Verbindungen automatisch gestartet werden.
5. Ändern Sie die neue Verbindung so, dass sie ebenfalls die Adresse 10.0.1.1/24 verwendet.
6. Konfigurieren Sie die Datei **hosts** so, dass 10.0.1.1 als **privat** referenziert werden kann.

7. Starten Sie das System neu.
8. Verifizieren Sie auf **workstation** mit dem **ping** Befehl, dass **serverb** initialisiert ist.

## Bewertung

Führen Sie auf **workstation** das Skript **lab net-review grade** aus, um den Erfolg dieser praktischen Übung zu bestätigen.

```
[student@workstation ~]$ lab net-review grade
```

## Beenden

Führen Sie auf **workstation** das Skript **lab net-review finish** aus, um diese praktische Übung abzuschließen.

```
[student@workstation ~]$ lab net-review finish
```

Damit ist die praktische Übung abgeschlossen.

## ► Lösung

# Netzwerkmanagement

### Leistungscheckliste

In dieser Übung konfigurieren Sie die Netzwerkeinstellungen für einen Red Hat Enterprise Linux-Server.

### Ergebnisse

Sie sollten in der Lage sein, zwei statische IPv4-Adressen für die primäre Netzwerkschnittstelle zu konfigurieren.

### Bevor Sie Beginnen

Melden Sie sich als Benutzer **student** mit dem Passwort **student** bei **workstation** an.

Führen Sie auf der **workstation** den Befehl **lab net-review start** aus. Dieser Befehl führt ein Startskript aus, das ermittelt, ob der Host **serverb** im Netzwerk erreichbar ist.

```
[student@workstation ~]$ lab net-review start
```

1. Melden Sie sich mit dem Befehl **ssh** bei **serverb** als **student** an. Die Systeme sind für die Verwendung von SSH-Schlüsseln zur Authentifizierung konfiguriert. Daher ist kein Passwort erforderlich, um sich bei **servera** anzumelden.

```
[student@workstation ~]$ ssh student@serverb  
...output omitted...  
[student@serverb ~]$
```

2. Verwenden Sie den Befehl **sudo -i**, um zum **root**-Benutzer zu wechseln. Verwenden Sie bei Aufforderung **student** als Passwort.

```
[student@serverb ~]$ sudo -i  
[sudo] password for student: student  
[root@serverb ~]#
```

3. Erstellen Sie mittels der Einstellungen in der Tabelle eine neue Verbindung mit einer statischen Netzwerkverbindung.

| Parameter          | Einstellung                                                                                         |
|--------------------|-----------------------------------------------------------------------------------------------------|
| Verbindungsname    | lab                                                                                                 |
| Schnittstellenname | enX (kann variieren, verwenden Sie die Schnittstelle mit der MAC-Adresse <b>52:54:00:00:fa:0b</b> ) |
| IP-Adresse         | 172.25.250.11/24                                                                                    |
| Gateway-Adresse    | 172.25.250.254                                                                                      |
| DNS-Adresse        | 172.25.250.254                                                                                      |

Ermitteln Sie den Schnittstellennamen und den Namen der aktuell aktiven Verbindung. Die Lösung geht davon aus, dass der Schnittstellenname **enX** und der Verbindungsname **Wired connection 1** lauten.

```
[root@serverb ~]# ip link
1: lo: <LOOPBACK,UP,LOWER_UP> mtu 65536 qdisc noqueue state UNKNOWN group default
    qlen 1000
        link/loopback 00:00:00:00:00:00 brd 00:00:00:00:00:00
2: enX: <BROADCAST,MULTICAST,UP,LOWER_UP> mtu 1500 qdisc fq_codel state UP group default
    qlen 1000
        link/ether 52:54:00:00:fa:0b brd ff:ff:ff:ff:ff:ff
[root@serverb ~]# nmcli con show --active
NAME           UUID                                  TYPE      DEVICE
Wired connection 1  03da038a-3257-4722-a478-53055cc90128  ethernet  enX
```

Erstellen Sie das neue **lab**-Verbindungsprofil basierend auf den Informationen in der Tabelle, die in der Anleitung beschrieben ist. Ordnen Sie das Profil dem Namen Ihrer Netzwerkschnittstelle zu, der in der Ausgabe des vorherigen **ip link**-Befehls aufgeführt ist.

```
[root@serverb ~]# nmcli con add con-name lab iface enX type ethernet \
    ipv4.method manual \
    ipv4.address 172.25.250.11/24 ipv4.gateway 172.25.250.254
[root@serverb ~]# nmcli con mod "lab" ipv4.dns 172.25.250.254
```

- Konfigurieren Sie die neue Verbindung so, dass sie automatisch gestartet wird. Es sollen keine anderen Verbindungen automatisch gestartet werden.

```
[root@serverb ~]# nmcli con mod "lab" connection.autoconnect yes
[root@serverb ~]# nmcli con mod "Wired connection 1" connection.autoconnect no
```

5. Ändern Sie die neue Verbindung so, dass sie ebenfalls die Adresse 10.0.1.1/24 verwendet.

```
[root@serverb ~]# nmcli con mod "lab" +ipv4.addresses 10.0.1.1/24
```

Alternativ:

```
[root@serverb ~]# echo "IPADDR1=10.0.1.1" \
>> /etc/sysconfig/network-scripts/ifcfg-lab
[root@serverb ~]# echo "PREFIX1=24" >> /etc/sysconfig/network-scripts/ifcfg-lab
```

6. Konfigurieren Sie die Datei **hosts** so, dass 10.0.1.1 als **privat** referenziert werden kann.

```
[root@serverb ~]# echo "10.0.1.1 private" >> /etc/hosts
```

7. Starten Sie das System neu.

```
[root@serverb ~]# systemctl reboot
Connection to serverb closed by remote host.
Connection to serverb closed.
[student@workstation ~]$
```

8. Verifizieren Sie auf **workstation** mit dem **ping** Befehl, dass **serverb** initialisiert ist.

```
[student@workstation ~]$ ping -c3 serverb
PING serverb.lab.example.com (172.25.250.11) 56(84) bytes of data.
64 bytes from serverb.lab.example.com (172.25.250.11): icmp_seq=1 ttl=64
time=0.478 ms
64 bytes from serverb.lab.example.com (172.25.250.11): icmp_seq=2 ttl=64
time=0.504 ms
64 bytes from serverb.lab.example.com (172.25.250.11): icmp_seq=3 ttl=64
time=0.513 ms

--- serverb.lab.example.com ping statistics ---
3 packets transmitted, 3 received, 0% packet loss, time 78ms
rtt min/avg/max/mdev = 0.478/0.498/0.513/0.023 ms
[student@workstation ~]$
```

## Bewertung

Führen Sie auf **workstation** das Skript **lab net-review grade** aus, um den Erfolg dieser praktischen Übung zu bestätigen.

```
[student@workstation ~]$ lab net-review grade
```

## Beenden

Führen Sie auf **workstation** das Skript **lab net-review finish** aus, um diese praktische Übung abzuschließen.

```
[student@workstation ~]$ lab net-review finish
```

Damit ist die praktische Übung abgeschlossen.

# Zusammenfassung

---

In diesem Kapitel erhielten Sie folgende Informationen:

- Das TCP/IP-Netzwerkmodell ist eine vereinfachte, vierstufige Gruppe von Abstraktionen, die beschreibt, wie verschiedene Protokolle zusammenarbeiten, damit Computer über das Internet Datenverkehr von einem Computer zum anderen senden können.
- IPv4 ist das primäre Netzwerkprotokoll, das heute im Internet verwendet wird. IPv6 soll das Netzwerkprotokoll IPv4 letztendlich ersetzen. Standardmäßig arbeitet Red Hat Enterprise Linux im Dual-Stack-Modus, wobei beide Protokolle parallel verwendet werden.
- NetworkManager ist ein Daemon zur Überwachung und Verwaltung der Netzwerkkonfiguration.
- Der Befehl **nmcli** ist ein Befehlszeilentool zur Konfiguration von Netzwerkeinstellungen mit NetworkManager.
- Der statische Hostname des Systems wird in der Datei **/etc/hostname** gespeichert. Der Befehl **hostnamectl** dient zum Ändern oder Anzeigen des Status des Hostnamens des Systems und der zugehörigen Einstellungen. Der Befehl **hostname** ruft den Hostnamen des Systems auf oder modifiziert diesen temporär.

## Kapitel 13

# Archivieren und Übertragen von Dateien

### Ziel

Archivieren Sie Dateien und kopieren Sie sie zwischen Systemen.

### Ziele

- Archivieren Sie Dateien und Verzeichnisse in einer komprimierten Datei mit tar und extrahieren Sie den Inhalt eines vorhandenen tar-Archivs.
- Übertragen Sie Dateien mit SSH zu oder von einem Remote-System.
- Synchronisieren Sie die Inhalte einer lokalen Datei oder eines Verzeichnisses mit einer Kopie auf einem Remote-Server.

### Abschnitte

- Verwalten komprimierter tar-Archive (und angeleitete Übung)
- Sicheres Übertragen von Dateien zwischen Systemen (und angeleitete Übung)
- Synchronisieren von Dateien zwischen Systemen (und angeleitete Übung)

### Praktische Übung

Archivieren und Übertragen von Dateien

# Verwalten von komprimierten tar-Archiven

## Ziele

Nach Abschluss dieses Abschnitts sollten Sie Dateien und Verzeichnisse mit **tar** in einer komprimierten Datei archivieren und den Inhalt eines vorhandenen **tar**-Archivs extrahieren können.

## Befehl tar

Das Archivieren und Komprimieren von Dateien ist nützlich bei der Erstellung von Datensicherungen und um Daten in einem Netzwerk zu übertragen. Einer der ältesten und meistverwendeten Befehle zum Erstellen von und Arbeiten mit Sicherungsarchiven ist **tar**.

Mit **tar** können Benutzer viele große Dateien in einer einzelnen Datei (Archiv) zusammenfassen. Ein **tar**-Archiv ist eine strukturierte Folge von Dateidaten mit Metadaten zu jeder Datei und einem Index, sodass einzelne Dateien extrahiert werden können. Das Archiv kann mithilfe der Komprimierungsmethoden **gzip**, **bzip2** oder **xz** komprimiert werden.

Mit dem Befehl **tar** kann auch der Inhalt von Archiven aufgelistet oder die Dateien auf dem aktuellen System extrahiert werden.

## Ausgewählte tar-Optionen

**tar**-Befehlsoptionen werden in Operationen unterteilt (die Aktion, die Sie ausführen möchten): Allgemeine Optionen und Komprimierungsoptionen. In der folgenden Tabelle werden allgemeine Optionen, lange Versionen von Optionen und ihre Beschreibung angezeigt:

### Übersicht über tar-Operationen

| Option               | Beschreibung                                |
|----------------------|---------------------------------------------|
| <b>-c, --create</b>  | Neues Archiv erstellen.                     |
| <b>-x, --extract</b> | Aus vorhandenem Archiv extrahieren.         |
| <b>-t, --list</b>    | Inhaltsverzeichnis eines Archivs auflisten. |

### Ausgewählte, allgemeine tar-Optionen

| Option               | Beschreibung                                                                                            |
|----------------------|---------------------------------------------------------------------------------------------------------|
| <b>-v, --verbose</b> | Verbose. Zeigt, welche Dateien archiviert oder extrahiert werden.                                       |
| <b>-f, --file=</b>   | File name. Auf diese Option muss der Dateiname des zu verwendenden oder zu erstellenden Archivs folgen. |

| Option                            | Beschreibung                                                                                                                                    |
|-----------------------------------|-------------------------------------------------------------------------------------------------------------------------------------------------|
| <b>-p, --preserve-permissions</b> | Berechtigungen für Dateien und Verzeichnisse beim Extrahieren eines Archivs beibehalten, ohne die <b>Aufhebung der Maskierung</b> zu entfernen. |

### Übersicht über tar Komprimierungsoptionen

| Option             | Beschreibung                                                                                                                        |
|--------------------|-------------------------------------------------------------------------------------------------------------------------------------|
| <b>-z, --gzip</b>  | Komprimierung gzip verwenden ( <b>.tar.gz</b> ).                                                                                    |
| <b>-j, --bzip2</b> | Komprimierung bzip2 verwenden ( <b>.tar.bz2</b> ). bzip2 erzielt in der Regel eine bessere Komprimierungsrate als gzip.             |
| <b>-J, --xz</b>    | Komprimierung xz verwenden ( <b>.tar.xz</b> ). Die Komprimierung xz erzielt in der Regel eine bessere Komprimierungsrate als bzip2. |

## Auflisten der Optionen des Befehls tar

Der Befehl **tar** erwartet eine der drei folgenden Optionen:

- Verwenden Sie die Option **-c** oder **--create** zum Erstellen eines Archivs.
- Verwenden Sie die Option **-t** oder **--list**, um die Inhalte eines Archivs aufzulisten.
- Verwenden Sie die Option **-x** oder **--extract**, um ein Archiv zu extrahieren.

Andere häufig verwendete Optionen sind:

- Verwenden Sie die Option **-f** oder **--file=** mit einem Dateinamen als Argument des zu verwendenden Archivs.
- Verwenden Sie die Option **-v** oder **--verbose** für die Ausführlichkeit. Dies ist hilfreich, um zu prüfen, welche Dateien zum Archiv hinzugefügt oder aus dem Archiv extrahiert werden.



### Anmerkung

Der Befehl **tar** unterstützt einen dritten, alten Optionsstil, der die einbuchstabigen Standardoptionen ohne führendes **-** verwendet. Dieser Stil wird immer noch häufig verwendet und Sie treffen möglicherweise auf diese Syntax, wenn Sie mit Anweisungen oder Befehlen anderer Personen arbeiten. Im Befehl **info tar 'old options'** wird detailliert erläutert, wie sich dies von normalen Kurzoptionen unterscheidet.

Sie können alte Optionen vorerst ignorieren und sich auf die Standardsyntax für kurze und lange Optionen konzentrieren.

## Archivieren von Dateien und Verzeichnissen

Die erste Option, die bei der Erstellung eines neuen Archivs verwendet wird, ist die Option **c**, gefolgt von der Option **f**. Danach folgen ein Leerzeichen, dann der Name des zu erstellenden Archivs und schließlich die Liste der Dateien und Verzeichnisse, die dem Archiv hinzugefügt werden sollten. Das Archiv wird im aktuellen Verzeichnis erstellt, sofern nicht anders angegeben.

**Warnung**

Bevor Sie ein tar-Archiv erstellen, prüfen Sie, ob sich kein weiteres Archiv mit dem gleichen Namen in dem Verzeichnis befindet, in dem das neue Archiv erstellt werden soll. Der Befehl **tar** überschreibt ein vorhandenes Archiv ohne Warnung.

Der folgende Befehl erstellt ein Archiv namens **archive.tar** mit den Inhalten von **file1**, **file2** und **file3** im Benutzerverzeichnis.

```
[user@host ~]$ tar -cf archive.tar file1 file2 file3
[user@host ~]$ ls archive.tar
archive.tar
```

Der obige Befehl **tar** kann auch mit den Optionen in der langen Version ausgeführt werden.

```
[user@host ~]$ tar --file=archive.tar --create file1 file2 file3
```

**Anmerkung**

Wenn Dateien nach absoluten Pfadnamen archiviert werden, wird der vorangestellte `/` standardmäßig aus dem Dateinamen entfernt. Das Entfernen des vorangestellten `/` des Pfads hilft Benutzern, beim Extrahieren des Archivs das Überschreiben wichtiger Dateien zu vermeiden. Der Befehl **tar** extrahiert Dateien relativ zum aktuellen Arbeitsverzeichnis.

Damit tar die ausgewählten Dateien archivieren kann, ist es obligatorisch, dass der Benutzer, der den Befehl **tar** ausführt, die Dateien lesen kann. Das Erstellen eines neuen Archivs des Ordners `/etc` und seines kompletten Inhalts erfordert **root**-Berechtigungen, da nur der Benutzer **root** alle Dateien lesen darf, die im Verzeichnis `/etc` enthalten sind. Ein unprivilegierter Benutzer könnte ein Archiv des Verzeichnisses `/etc` erstellen, aber das Archiv lässt Dateien aus, die keine Leseberechtigungen für den Benutzer enthalten, und es lässt Verzeichnisse aus, die keine Lese- und Ausführberechtigungen für den Benutzer enthalten.

So erstellen Sie als **root**-Benutzer das tar-Archiv `/root/etc.tar` mit dem Verzeichnis `/etc` als Inhalt:

```
[root@host ~]# tar -cf /root/etc.tar /etc
tar: Removing leading `/' from member names
[root@host ~]#
```

**Wichtig**

Einige erweiterte Berechtigungen, die in diesem Kurs nicht behandelt werden, wie z. B. ACLs und SELinux-Kontexte, werden nicht automatisch in einem **tar**-Archiv gespeichert. Verwenden Sie beim Erstellen eines Archivs die Option `--xattrs`, um diese erweiterten Attribute im tar-Archiv zu speichern.

## Auflisten der Inhalte von Archiven

Die Option **t** weist **tar** an, den Inhalt (Inhaltsverzeichnis, daher **t**) des Archivs aufzulisten. Verwenden Sie die Option **f** mit dem Namen des abzufragenden Archivs. Beispiel:

```
[root@host ~]# tar -tf /root/etc.tar
etc/
etc/fstab
etc/crypttab
etc/mtab
...output omitted...
```

## Extrahieren von Dateien aus Archiven

Ein tar-Archiv sollte in der Regel in ein leeres Verzeichnis extrahiert werden, um sicherzustellen, dass keine vorhandenen Dateien überschrieben werden. Wenn der **root**-Benutzer ein Archiv extrahiert, behält der Befehl **tar** den ursprünglichen Benutzer und den Gruppeneigentümer der Dateien bei. Wenn ein normaler Benutzer Dateien mit **tar** extrahiert, ist der Dateieigentümer der Benutzer, der die Dateien aus dem Archiv extrahiert.

Führen Sie den folgenden Befehl aus, um Dateien aus dem **/root/etc.tar**-Archiv im Verzeichnis **/root/etcbackup** wiederherzustellen:

```
[root@host ~]# mkdir /root/etcbackup
[root@host ~]# cd /root/etcbackup
[root@host etcbackup]# tar -tf /root/etc.tar
etc/
etc/fstab
etc/crypttab
etc/mtab
...output omitted...
[root@host etcbackup]# tar -xf /root/etc.tar
```

Standardmäßig wird beim Extrahieren von Dateien aus einem Archiv die **Aufhebung der Maskierung** aus den Berechtigungen des Archivinhalts entfernt. Um die Berechtigungen einer archivierten Datei beizubehalten, muss beim Extrahieren eines Archivs die Option **p** verwendet werden.

In diesem Beispiel wird das Archiv namens **/root/myscripts.tar** unter Beibehaltung der Berechtigungen der extrahierten Dateien in das Verzeichnis **/root/scripts** extrahiert:

```
[root@host ~]# mkdir /root/scripts
[root@host ~]# cd /root/scripts
[root@host scripts]# tar -xpf /root/myscripts.tar
```

## Erstellen eines komprimierten Archivs

Der Befehl **tar** unterstützt drei Komprimierungsmethoden. Es gibt drei verschiedene Komprimierungsmethoden, die vom Befehl **tar** unterstützt werden. Die **gzip**-Komprimierung ist die älteste und schnellste Methode und sie ist am weitesten über Distributionen und Plattformen hinweg verbreitet. Die **bzip2**-Komprimierung erzeugt im Vergleich zu **gzip** kleinere Archivdateien, ist aber weniger weit verbreitet als **gzip**, während die **xz**-Komprimierungsmethode

relativ neu ist, aber in der Regel die beste Komprimierungsrate unter den verfügbaren Methoden bietet.



### Anmerkung

Die Effektivität jedes Komprimierungsalgorithmus hängt von der Art der zu komprimierenden Daten ab. Bereits komprimierte Datendateien, wie komprimierte Bildformate oder RPM-Dateien, erbringen in der Regel niedrige Komprimierungsraten.

Es empfiehlt sich, ein einziges Verzeichnis der obersten Ebene zu verwenden, das andere Verzeichnisse und Dateien enthalten kann, um die geordnete Extrahierung der Dateien zu vereinfachen.

Verwenden Sie eine der folgenden Optionen, um ein komprimiertes tar-Archiv zu erstellen:

- **-z** oder **--gzip** für die **gzip**-Komprimierung (**filename.tar.gz** oder **filename.tgz**)
- **-j** oder **--bzip2** für die **bzip2**-Komprimierung (**filename.tar.bz2**)
- **-J** oder **-xz** für die **xz**-Komprimierung (**filename.tar.xz**)

So erstellen Sie ein mit **gzip** komprimiertes Archiv namens **/root/etcbackup.tar.gz** mit dem Inhalt aus dem Verzeichnis **/etc** auf dem **Host**:

```
[root@host ~]# tar -czf /root/etcbackup.tar.gz /etc
tar: Removing leading `/' from member names
```

So erstellen Sie ein mit **bzip2** komprimiertes Archiv namens **/root/logbackup.tar.bz2** mit dem Inhalt aus dem Verzeichnis **/var/log** auf dem **Host**:

```
[root@host ~]$ tar -cjf /root/logbackup.tar.bz2 /var/log
tar: Removing leading `/' from member names
```

So erstellen Sie ein mit **xz** komprimiertes Archiv namens **/root/sshconfig.tar.xz** mit dem Inhalt aus dem Verzeichnis **/etc/ssh** auf dem **Host**:

```
[root@host ~]$ tar -cJf /root/sshconfig.tar.xz /etc/ssh
tar: Removing leading `/' from member names
```

Überprüfen Sie nach dem Erstellen eines Archivs dessen Inhalt mithilfe der **tf**-Optionen. Es ist nicht obligatorisch, die Option für den Komprimierungsagenten zu verwenden, wenn der Inhalt einer komprimierten Archivdatei aufgelistet wird. Verwenden Sie z. B. den folgenden Befehl, um den in der Datei **/root/etcbackup.tar.gz** archivierten Inhalt aufzulisten, der die **gzip**-Komprimierung verwendet:

```
[root@host ~]# tar -tf /root/etcbackup.tar.gz /etc
etc/
etc/fstab
etc/crypttab
etc/mtab
...output omitted...
```

## Extrahieren eines komprimierten Archivs

Als ersten Schritt zum Extrahieren eines komprimierten **tar**-Archivs bestimmen Sie, wohin die archivierten Dateien extrahiert werden sollen. Dann erstellen Sie das Zielverzeichnis und wechseln dorthin. Der Befehl **tar** ermittelt die verwendete Komprimierung. In der Regel ist es nicht erforderlich, dieselbe Komprimierungsmethode wie bei der Erstellung des Archivs zu verwenden. Es ist zulässig, die Komprimierungsmethode zum Befehl **tar** hinzuzufügen. In diesem Fall muss die richtige Option für den Dekomprimierungstyp verwendet werden. Andernfalls führt tar aufgrund des Dekomprimierungstyps zu einem Fehler, der in den Optionen angegeben wurde und nicht mit dem Dekomprimierungstyp der Datei übereinstimmt.

So extrahieren Sie den Inhalt eines mit **gzip** komprimierten Archiv namens **/root/etcbackup.tar.gz** in das Verzeichnis **/tmp/etcbackup**:

```
[root@host ~]# mkdir /tmp/etcbackup
[root@host ~]# cd /tmp/etcbackup
[root@host etcbackup]# tar -tf /root/etcbackup.tar.gz
etc/
etc/fstab
etc/crypttab
etc/mtab
...output omitted...
[root@host etcbackup]# tar -xzf /root/etcbackup.tar.gz
```

So extrahieren Sie den Inhalt eines mit **bzip2** komprimierten Archiv namens **/root/logbackup.tar.bz2** in das Verzeichnis **/tmp/logbackup**:

```
[root@host ~]# mkdir /tmp/logbackup
[root@host ~]# cd /tmp/logbackup
[root@host logbackup]# tar -tf /root/logbackup.tar.bz2
var/log/
var/log/lastlog
var/log/README
var/log/private/
var/log/wtmp
var/log/btmp
...output omitted...
[root@host logbackup]# tar -xjf /root/logbackup.tar.bz2
```

So extrahieren Sie den Inhalt eines mit **xz** komprimierten Archiv namens **/root/sshbackup.tar.xz** in das Verzeichnis **/tmp/sshbackup**:

```
[root@host ~]$ mkdir /tmp/sshbackup
[root@host ~]# cd /tmp/sshbackup
[root@host sshbackup]# tar -tf /root/sshbackup.tar.xz
etc/ssh/
etc/ssh/moduli
etc/ssh/ssh_config
etc/ssh/ssh_config.d/
etc/ssh/ssh_config.d/05-redhat.conf
etc/ssh/sshd_config
...output omitted...
[root@host sshbackup]# tar -xJf /root/sshbackup.tar.xz
```

Das Auflisten eines komprimierten tar-Archivs funktioniert genauso wie das Auflisten eines unkomprimierten **tar**-Archivs.



### Anmerkung

Außerdem können Sie die Befehle **gzip**, **bzip2** und **xz** auch unabhängig voneinander zur Komprimierung einzelner Dateien verwenden. So ergibt der Befehl **gzip etc.tar** beispielsweise die komprimierte Datei **etc.tar.gz**, während der Befehl **bzip2 abc.tar** die komprimierte Datei **abc.tar.bz2** und der Befehl **xz myarchive.tar** die komprimierte Datei **myarchive.tar.xz** ergibt.

Die entsprechenden Befehle zum Dekomprimieren lauten **gunzip**, **bunzip2** und **unxz**. So ergibt der Befehl **gunzip /tmp/etc.tar.gz** beispielsweise die unkomprimierte tar-Datei **etc.tar**, während der Befehl **bunzip2 abc.tar.bz2** die unkomprimierte tar-Datei **abc.tar** und der Befehl **unxz myarchive.tar.xz** die unkomprimierte tar-Datei **myarchive.tar** ergibt.



### Literaturhinweise

Manpages **tar(1)**, **gzip(1)**, **gunzip(1)**, **bzip2(1)**, **bunzip2(1)**, **xz(1)**, **unxz(1)**

## ► Angeleitete Übung

# Verwalten von komprimierten tar-Archiven

In dieser Übung erstellen Sie Archivdateien und extrahieren deren Inhalt mit dem tar-Befehl.

## Ergebnisse

Sie sollten in der Lage sein, eine Verzeichnisstruktur zu archivieren und den Archivinhalt an einem anderen Speicherort zu extrahieren.

## Bevor Sie Beginnen

Melden Sie sich als Benutzer **student** mit dem Passwort **student** bei **workstation** an.

Führen Sie auf **workstation** den Befehl **lab archive-manage start** aus. Dieser Befehl führt ein Startskript aus, das ermittelt, ob der Host **servera** im Netzwerk erreichbar ist. Das Skript stellt außerdem sicher, dass die in der Übung zu erstellende Datei und das zu erstellende Verzeichnis nicht auf **servera** vorhanden sind.

```
[student@workstation ~]$ lab archive-manage start
```

- 1. Melden Sie sich mit dem Befehl **ssh** bei **servera** als Benutzer **student** an.

```
[student@workstation ~]$ ssh student@servera
...output omitted...
[student@servera ~]$
```

- 2. Wechseln Sie zum **root**-Benutzer, da nur der **root**-Benutzer auf sämtliche Inhalte im Verzeichnis **/etc** zugreifen kann.

```
[student@servera ~]$ su -
Password: redhat
[root@servera ~]#
```

- 3. Verwenden Sie den Befehl **tar** mit den **-czf**-Optionen, um ein Archiv des Verzeichnisses **/etc** mit **gzip**-Komprimierung zu erstellen. Speichern Sie die Archivdatei als **/tmp/etc.tar.gz**.

```
[root@servera ~]# tar -czf /tmp/etc.tar.gz /etc
tar: Removing leading `/' from member names
[root@servera ~]#
```

- 4. Verwenden Sie den Befehl **tar** mit den **-tzf**-Optionen, um sicherzustellen, dass das **etc.tar.gz**-Archiv die Dateien aus dem Verzeichnis **/etc** enthält.

```
[root@servera ~]# tar -tzf /tmp/etc.tar.gz
etc/
etc/mtab
etc/fstab
etc/crypttab
etc/resolv.conf
...output omitted...
```

- 5. Erstellen Sie auf **servera** ein Verzeichnis mit dem Namen **/backuptest**. Überprüfen Sie, dass die Sicherungsdatei **etc.tar.gz** ein gültiges Archiv ist, indem Sie die Datei in das Verzeichnis **/backuptest** dekomprimieren.

5.1. Erstellen Sie das Verzeichnis **/backuptest**.

```
[root@servera ~]# mkdir /backuptest
```

5.2. Wechseln Sie in das Verzeichnis **/backuptest**.

```
[root@servera ~]# cd /backuptest
[root@servera backuptest]#
```

5.3. Listen Sie vor dem Extrahieren den Inhalt des **etc.tar.gz**-Archivs auf.

```
[root@servera backuptest]# tar -tzf /tmp/etc.tar.gz
etc/
etc/mtab
etc/fstab
etc/crypttab
etc/resolv.conf
...output omitted...
```

5.4. Extrahieren Sie das Archiv **/tmp/etc.tar.gz** in das Verzeichnis **/backuptest**.

```
[root@servera backuptest]# tar -xzf /tmp/etc.tar.gz
[root@servera backuptest]#
```

5.5. Listen Sie den Inhalt des Verzeichnisses **/backuptest** auf. Überprüfen Sie, ob das Verzeichnis die Dateien aus dem Verzeichnis **/etc** enthält.

```
[root@servera backuptest]# ls -l
total 12
drwxr-xr-x. 95 root root 8192 Feb  8 10:16 etc
[root@servera backuptest]# cd etc
[root@servera etc]# ls -l
total 1204
-rw-r--r--.  1 root root      16 Jan 16 23:41 adjtime
-rw-r--r--.  1 root root    1518 Sep 10 17:21 aliases
drwxr-xr-x.  2 root root     169 Feb  4 21:58 alternatives
-rw-r--r--.  1 root root     541 Oct  2 21:01 anacrontab
...output omitted...
```

► 6. Beenden Sie **servera**.

```
[root@servera backuptest]# exit  
logout  
[student@servera ~]$ exit  
logout  
Connection to servera closed.  
[student@workstation]$
```

## Beenden

Führen Sie auf **workstation** das Skript **lab archive-manage finish** aus, um diese Übung zu beenden.

```
[student@workstation ~]$ lab archive-manage finish
```

Hiermit ist die angeleitete Übung beendet.

# Sicheres Übertragen von Dateien zwischen Systemen

---

## Ziele

Nach Abschluss dieses Abschnitts sollten Sie in der Lage sein, Dateien sicher an ein bzw. von einem Remote-System mit SSH zu übertragen.

## Übertragen von Dateien mit Secure Copy

OpenSSH ist hilfreich, um Shell-Befehle sicher auf Remote-Systemen auszuführen. Der Secure Copy-Befehl **scp**, der Teil der OpenSSH-Suite ist, kopiert Dateien von einem Remote-System auf das lokale System oder vom lokalen System auf ein Remote-System. Der Befehl verwendet den SSH-Server für die Authentifizierung und verschlüsselt Daten bei der Übertragung.

Sie können einen Remote-Speicherort für die Quelle oder das Ziel der zu kopierenden Dateien angeben. Das Format des Remote-Speicherorts sollte in der Form **[user@]host :/path** angegeben werden. Der **user@**-Teil des Arguments ist optional. Wenn er fehlt, wird Ihr aktueller lokaler Benutzername verwendet. Wenn Sie den Befehl ausführen, authentifiziert sich Ihr **scp**-Client beim Remote-SSH-Server, genau wie **ssh**, mit der schlüsselbasierten Authentifizierung oder per Aufforderung zur Eingabe Ihres Passworts.

Das folgende Beispiel zeigt, wie die lokalen Dateien **/etc/yum.conf** und **/etc/hosts** auf **host** in das Benuterverzeichnis von **remoteuser** auf dem Remote-System **remotehost** kopiert werden:

```
[user@host ~]$ scp /etc/yum.conf /etc/hosts remoteuser@remotehost:/home/remoteuser
remoteuser@remotehost's password: password
yum.conf                                100%   813      0.8KB/s  00:00
hosts                                    100%   227      0.2KB/s  00:00
```

Sie können eine Datei auch in die andere Richtung, von einem Remote-System auf das lokale Dateisystem, kopieren. In diesem Beispiel wird die Datei **/etc/hostname** auf **remotehost** in das lokale Verzeichnis **/home/user** kopiert. Der Befehl **scp** authentifiziert sich bei **remotehost** als Benutzer **remoteuser**.

```
[user@host ~]$ scp remoteuser@remotehost:/etc/hostname /home/user
remoteuser@remotehost's password: password
hostname                               100%    22      0.0KB/s  00:00
```

Mit der Option **-r** können Sie eine komplette Verzeichnisstruktur rekursiv kopieren. Im folgenden Beispiel wird das Remote-Verzeichnis **/var/log** auf **remotehost** rekursiv in das lokale Verzeichnis **/tmp/** auf **host** kopiert. Sie müssen als **root** eine Verbindung zum Remote-System herstellen, um sicherzustellen, dass Sie alle Dateien auf dem Remote-Verzeichnis **/var/log** lesen können.

```
[user@host ~]$ scp -r root@remoteuser:/var/log /tmp
root@remotehost's password: password
...output omitted...
```

## Übertragen von Dateien mit dem Secure File Transfer Program

Verwenden Sie den Befehl **sftp**, um Dateien interaktiv von einem SSH-Server mit dem Secure File Transfer Program hoch- oder herunterzuladen. Eine Sitzung mit dem Befehl **sftp** verwendet den sicheren Authentifizierungsmechanismus und die verschlüsselte Datenübertragung zum und vom SSH-Server.

Wie der Befehl **scp** verwendet der Befehl **sftp** ebenfalls **[user@]host**, um Zielsystem und Benutzername zu identifizieren. Wenn Sie keinen Benutzer angeben, versucht der Befehl, sich mit Ihrem lokalen Benutzernamen als Remote-Benutzernamen anzumelden. Sie erhalten dann eine **sftp>**-Eingabeaufforderung.

```
[user@host ~]$ sftp remoteuser@remotehost  
remoteuser@remotehost's password: password  
Connected to remotehost.  
sftp>
```

Die interaktive **sftp**-Sitzung akzeptiert mehrere Befehle, die auf die gleiche Weise im Remote-Dateisystem wie im lokalen Dateisystem funktionieren, z. B. **ls**, **cd**, **mkdir**, **rmdir** und **pwd**. Der Befehl **put** lädt eine Datei auf das Remote-System hoch. Der Befehl **get** lädt eine Datei von Remote-System herunter. Mit dem Befehl **exit** wird die **sftp**-Sitzung beendet.

So laden Sie die Datei **/etc/hosts** auf dem lokalen System in das neu erstellte Verzeichnis **/home/remoteuser/hostbackup** auf **remotehost** hoch. Die **sftp**-Sitzung geht davon aus, dass auf den Befehl **put** immer eine Datei auf dem lokalen Dateisystem folgt. Der Anfangspunkt ist das Benuterverzeichnis des verbindenden Benutzers, in diesem Fall **/home/remoteuser**:

```
sftp> mkdir hostbackup  
sftp> cd hostbackup  
sftp> put /etc/hosts  
Uploading /etc/hosts to /home/remoteuser/hostbackup/hosts  
/etc/hosts  100%  227      0.2KB/s  00:00  
sftp>
```

Um **/etc/yum.conf** vom Remote-Host in das aktuelle Verzeichnis im lokalen System herunterzuladen, führen Sie den Befehl **get /etc/yum.conf** aus, und beenden Sie die **sftp**-Sitzung mit dem Befehl **exit**.

```
sftp> get /etc/yum.conf  
Fetching /etc/yum.conf to yum.conf  
/etc/yum.conf  100%  813      0.8KB/s  00:00  
sftp> exit  
[user@host ~]$
```



### Literaturhinweise

Manpages von **scp(1)** und **sftp(1)**

## ► Angeleitete Übung

# Sicheres Übertragen von Dateien zwischen Systemen

In dieser Übung kopieren Sie Dateien von einem Remote-System mithilfe von **scp** in ein lokales Verzeichnis.

## Ergebnisse

Sie sollten in der Lage sein, Dateien von einem Remote-Host in ein Verzeichnis auf dem lokalen Rechner kopieren zu können.

## Bevor Sie Beginnen

Melden Sie sich als Benutzer **student** mit dem Passwort **student** bei **workstation** an.

Führen Sie auf **workstation** den Befehl **lab archive-transfer start** aus. Dieser Befehl führt ein Startskript aus, das ermittelt, ob die Hosts **servera** und **serverb** im Netzwerk erreichbar sind. Das Skript stellt außerdem sicher, dass die in der Übung zu erstellende Datei und das zu erstellende Verzeichnis nicht auf **servera** vorhanden sind.

```
[student@workstation ~]$ lab archive-transfer start
```

- 1. Melden Sie sich mit dem Befehl **ssh** bei **servera** als Benutzer **student** an.

```
[student@workstation ~]$ ssh student@servera  
...output omitted...  
[student@servera ~]$
```

- 2. Mit dem Befehl **scp** kopieren Sie das Verzeichnis **/etc/ssh** von **serverb** in das Verzeichnis **/home/student/serverbackup** auf **servera**.

- 2.1. Erstellen Sie auf **servera** ein Verzeichnis mit dem Namen **/home/student/serverbackup**.

```
[student@servera ~]$ mkdir ~/serverbackup
```

- 2.2. Mit dem Befehl **scp** kopieren Sie das Verzeichnis **/etc/ssh** von **serverb** rekursiv in das Verzeichnis **/home/student/serverbackup** auf **servera**. Geben Sie das Passwort **redhat** ein, wenn Sie dazu aufgefordert werden. Beachten Sie, dass nur der **root**-Benutzer den gesamten Inhalt im Verzeichnis **/etc/ssh** lesen kann.

```
[student@servera ~]$ scp -r root@serverb:/etc/ssh ~/serverbackup  
The authenticity of host 'serverb (172.25.250.11)' can't be established.  
ECDSA key fingerprint is SHA256:qaS0PToLrq1C02XGk1A0iY7CaP7aPKimerDoaUkv720.  
Are you sure you want to continue connecting (yes/no)? yes
```

```
Warning: Permanently added 'serverb,172.25.250.11' (ECDSA) to the list of known hosts.
root@serverb's password: redhat
moduli                                100%  550KB  57.9MB/s  00:00
ssh_config                             100% 1727    1.4MB/s  00:00
05-redhat.conf                         100%  690    1.6MB/s  00:00
01-training.conf                        100%   36    80.5KB/s  00:00
ssh_host_ed25519_key                   100%  387    1.2MB/s  00:00
ssh_host_ed25519_key.pub                100%   82    268.1KB/s  00:00
ssh_host_ecdsa_key                     100%  492    1.5MB/s  00:00
ssh_host_ecdsa_key.pub                 100%  162    538.7KB/s  00:00
ssh_host_rsa_key                       100% 1799    4.9MB/s  00:00
ssh_host_rsa_key.pub                  100%  382    1.2MB/s  00:00
sshd_config                            100% 4469    9.5MB/s  00:00
```

- 2.3. Überprüfen Sie, ob das Verzeichnis **/etc/ssh** von **serverb** in das Verzeichnis **/home/student/serverbackup** auf **servera** kopiert wird.

```
[student@servera ~]$ ls -lR ~/serverbackup
/home/student/serverbackup:
total 0
drwxr-xr-x. 3 student student 245 Feb 11 18:35 ssh

/home/student/serverbackup/ssh:
total 588
-rw-r--r--. 1 student student 563386 Feb 11 18:35 moduli
-rw-r--r--. 1 student student 1727 Feb 11 18:35 ssh_config
drwxr-xr-x. 2 student student 28 Feb 11 18:35 ssh_config.d
-rw-----. 1 student student 4469 Feb 11 18:35 sshd_config
-rw-r-----. 1 student student 492 Feb 11 18:35 ssh_host_ecdsa_key
-rw-r--r--. 1 student student 162 Feb 11 18:35 ssh_host_ecdsa_key.pub
-rw-r-----. 1 student student 387 Feb 11 18:35 ssh_host_ed25519_key
-rw-r--r--. 1 student student 82 Feb 11 18:35 ssh_host_ed25519_key.pub
-rw-r-----. 1 student student 1799 Feb 11 18:35 ssh_host_rsa_key
-rw-r--r--. 1 student student 382 Feb 11 18:35 ssh_host_rsa_key.pub

/home/student/serverbackup/ssh/ssh_config.d:
total 8
-rw-r--r--. 1 student student 36 Feb 11 18:35 01-training.conf
-rw-r--r--. 1 student student 690 Feb 11 18:35 05-redhat.conf
```

### ► 3. Beenden Sie **servera**.

```
[student@servera ~]$ exit
logout
Connection to servera closed.
[student@workstation]$
```

## Beenden

Führen Sie auf **workstation** das Skript **lab archive-transfer finish** aus, um diese Übung zu beenden.

```
[student@workstation ~]$ lab archive-transfer finish
```

Hiermit ist die angeleitete Übung beendet.

# Sicheres Synchronisieren von Dateien zwischen Systemen

## Ziele

Nach Abschluss dieses Abschnitts sollten Sie in der Lage sein, den Inhalt einer lokalen Datei oder eines Verzeichnisses effizient und sicher mit einer Kopie auf einem Remote-Server zu synchronisieren.

## Synchronisieren von Dateien und Verzeichnissen mit rsync

Der Befehl **rsync** stellt eine weitere Möglichkeit dar, um Dateien sicher von einem System auf ein anderes zu kopieren. Das Tool verwendet einen Algorithmus, der die kopierte Datenmenge minimiert, indem nur die geänderten Teile der Dateien synchronisiert werden. Der Befehl unterscheidet sich von **scp** dahingehend, dass bei zwei ähnlichen Dateien oder Verzeichnissen auf zwei Servern der Befehl **rsync** nur die Unterschiede zwischen den Dateisystemen kopiert, während **scp** alles kopieren würde.

Ein Vorteil von **rsync** besteht darin, dass Dateien sicher und effizient zwischen einem lokalen System und einem Remote-System kopiert werden können. Die erste Synchronisierung eines Verzeichnisses dauert etwa so lange wie das Kopieren. Bei allen nachfolgenden Synchronisierungen müssen jedoch nur noch die Unterschiede über das Netzwerk kopiert werden, wodurch Aktualisierungen erheblich beschleunigt werden.

Eine wichtige Option von **rsync** ist die Option **-n**, mit der ein Probelauf ausgeführt wird. Ein Probelauf simuliert, was geschieht, wenn der Befehl ausgeführt wird. Der Probelauf zeigt die Änderungen, die **rsync** ausführen würde, wenn der Befehl normal ausgeführt wird. Führen Sie einen Probelauf vor dem eigentlichen **rsync**-Vorgang durch, um sicherzustellen, dass keine wichtigen Dateien überschrieben oder gelöscht werden.

Zwei häufige Optionen beim Synchronisieren mit **rsync** sind **-v** und **-a**.

Die Option **-v** oder **--verbose** bietet eine detailliertere Ausgabe. Dies ist nützlich für die Fehlerbehebung und zum Anzeigen des Live-Fortschritts.

Die Option **-a** oder **--archive** aktiviert den „Archivmodus“. Dies ermöglicht das rekursive Kopieren und aktiviert eine Vielzahl nützlicher Optionen, bei denen die meisten Eigenschaften der Dateien erhalten bleiben. Der Archivmodus entspricht dem Festlegen der folgenden Optionen:

### Mit rsync -a aktivierte Optionen (Archivmodus)

| Option                 | Beschreibung                                                    |
|------------------------|-----------------------------------------------------------------|
| <b>-r, --recursive</b> | zum rekursiven Synchronisieren der gesamten Verzeichnisstruktur |
| <b>-l, --links</b>     | zum Synchronisieren von symbolischen Links                      |
| <b>-p, --perms</b>     | zum Beibehalten von Berechtigungen                              |

| Option               | Beschreibung                                |
|----------------------|---------------------------------------------|
| <b>-t, --times</b>   | zum Beibehalten von Zeitstempeln            |
| <b>-g, --group</b>   | zum Beibehalten der Gruppeneigentümerschaft |
| <b>-o, --owner</b>   | zum Beibehalten des Eigentümers der Dateien |
| <b>-D, --devices</b> | zum Synchronisieren von Gerätedateien       |

Im Archivmodus werden keine Hardlinks beibehalten, da dies die Synchronisierung erheblich beschleunigen kann. Wenn Sie auch Hardlinks beibehalten möchten, fügen Sie die Option **-H** hinzu.



### Anmerkung

Wenn Sie erweiterte Berechtigungen verwenden, benötigen Sie möglicherweise zwei zusätzliche Optionen:

- **-A** zum Beibehalten von ACLs
- **-X** zum Beibehalten von SELinux-Kontexten

Sie können mit **rsync** den Inhalt einer lokalen Datei oder eines lokalen Verzeichnisses mit einer Datei oder einem Verzeichnis auf einem Remote-Rechner synchronisieren, wobei einer der beiden Rechner als Quelle verwendet wird. Sie können auch den Inhalt zweier lokaler Dateien oder Verzeichnisse synchronisieren.

So synchronisieren Sie beispielsweise den Inhalt des Verzeichnisses **/var/log** mit dem Verzeichnis **/tmp**:

```
[user@host ~]$ su -
Password: password
[root@host ~]# rsync -av /var/log /tmp
receiving incremental file list
log/
log/README
log/boot.log
...output omitted...
log/tuned/tuned.log

sent 11,592,423 bytes received 779 bytes 23,186,404.00 bytes/sec
total size is 11,586,755 speedup is 1.00
[user@host ~]$ ls /tmp
log  ssh-RLjDdarkKiW1
[user@host ~]$
```

Ein Schrägstrich am Ende des Quellverzeichnisses synchronisiert den Inhalt dieses Verzeichnisses, ohne das Unterverzeichnis im Zielverzeichnis neu zu erstellen. In diesem Beispiel wird das Verzeichnis **log** *nicht* im Verzeichnis **/tmp** erstellt, sondern es wird nur der Inhalt von **/var/log** mit **/tmp** synchronisiert.

```
[root@host ~]# rsync -av /var/log/ /tmp
sending incremental file list
./
README
boot.log
...output omitted...
tuned/tuned.log

sent 11,592,389 bytes received 778 bytes 23,186,334.00 bytes/sec
total size is 11,586,755 speedup is 1.00
[root@host ~]# ls /tmp
anaconda          dnf.rpm.log-20190318  private
audit             dnf.rpm.log-20190324  qemu-ga
boot.log          dnf.rpm.log-20190331  README
...output omitted...
```



### Wichtig

Bei der Eingabe des Quellverzeichnisses für den Befehl **rsync** ist es entscheidend, ob der Verzeichnisname einen nachgestellten Schrägstrich enthält. Dadurch wird bestimmt, ob das **Verzeichnis** oder nur der **Verzeichnisinhalt** mit dem Ziel synchronisiert wird.

Die Bash-**Tab**-Vervollständigung fügt automatisch einen nachgestellten Schrägstrich an Verzeichnisnamen an.

Wie die Befehle **scp** und **sftp** gibt **rsync** Remote-Speicherorte im Format **[user@]host:/path** an. Der Remote-Speicherort kann entweder das Quell- oder das Zielsystem sein, aber einer der beiden Rechner muss lokal sein.

Um die Dateieigentümerschaft beizubehalten, müssen Sie auf dem Zielsystem **root** sein. Wenn das Ziel remote ist, authentifizieren Sie sich als **root**. Wenn das Ziel lokal ist, müssen Sie **rsync** als **root** ausführen.

In diesem Beispiel wird das lokale Verzeichnis **/var/log** mit dem Verzeichnis **/tmp** auf dem **remotehost**-System synchronisiert:

```
[root@host ~]# rsync -av /var/log remotehost:/tmp
root@remotehost's password: password
receiving incremental file list
log/
log/README
log/boot.log
...output omitted...
sent 9,783 bytes received 290,576 bytes 85,816.86 bytes/sec
total size is 11,585,690 speedup is 38.57
```

Auf dieselbe Weise kann das Remote-Verzeichnis **/var/log** auf **remotehost** mit dem lokalen Verzeichnis **/tmp** auf **host** synchronisiert werden:

```
[root@host ~]# rsync -av remotehost:/var/log /tmp
root@remotehost's password: password
receiving incremental file list
log/boot.log
log/dnf.librepo.log
log/dnf.log
...output omitted...

sent 9,783 bytes received 290,576 bytes 85,816.86 bytes/sec
total size is 11,585,690 speedup is 38.57
```



### Literaturhinweise

Manpage **rsync(1)**

## ► Angeleitete Übung

# Sicheres Synchronisieren von Dateien zwischen Systemen

In dieser Übung synchronisieren Sie den Inhalt eines lokalen Verzeichnisses mit einer Kopie auf einem Remote-Server, indem Sie **rsync** über SSH verwenden.

## Ergebnisse

Sie sollten den Befehl **rsync** zum Synchronisieren des Inhalts eines lokalen Verzeichnisses mit einer Kopie auf einem Remote-Server verwenden.

## Bevor Sie Beginnen

Melden Sie sich als Benutzer **student** mit dem Passwort **student** bei **workstation** an.

Führen Sie auf **workstation** den Befehl **lab archive-sync start** aus. Dieser Befehl führt ein Startskript aus, das ermittelt, ob die Hosts **servera** und **serverb** im Netzwerk erreichbar sind. Das Skript stellt außerdem sicher, dass die in der Übung zu erstellende Datei und das zu erstellende Verzeichnis nicht auf **servera** vorhanden sind.

```
[student@workstation ~]$ lab archive-sync start
```

- 1. Melden Sie sich mit dem Befehl **ssh** bei **servera** als Benutzer **student** an.

```
[student@workstation ~]$ ssh student@servera
...output omitted...
[student@servera ~]$
```

- 2. Erstellen Sie ein Verzeichnis mit dem Namen **/home/student/serverlogs** auf **servera**. Verwenden Sie den Befehl **rsync** zum sicheren Erstellen einer ersten Kopie der Verzeichnisstruktur **/var/log** auf **serverb** im Verzeichnis **/home/student/serverlogs** auf **servera**.

- 2.1. Erstellen Sie auf **servera** das Zielverzeichnis mit dem Namen **/home/student/serverlogs**, um die von **serverb** synchronisierten Protokolldateien zu speichern.

```
[student@servera ~]$ mkdir ~/serverlogs
```

- 2.2. Mit dem Befehl **rsync** synchronisieren Sie die Verzeichnisstruktur **/var/log** auf **serverb** mit dem Verzeichnis **/home/student/serverlogs** auf **servera**. Beachten Sie, dass nur der **root**-Benutzer den gesamten Inhalt im Verzeichnis **/var/log** auf **serverb** lesen kann. Bei der ersten Synchronisierung werden alle Dateien übertragen.

```
[student@servera ~]$ rsync -av root@serverb:/var/log ~/serverlogs
root@serverb's password: redhat
receiving incremental file list
log/
log/README
log/boot.log
...output omitted...
log/tuned/tuned.log

sent 992 bytes received 13,775,064 bytes 2,119,393.23 bytes/sec
total size is 13,768,109 speedup is 1.00
```

- 3. Führen Sie als **root**-Benutzer auf **serverb** den Befehl **logger "Log files synchronized"** aus, um einen neuen Eintrag in der Protokolldatei **/var/log/messages** anzulegen, aus dem hervorgeht, wann die letzte Synchronisierung erfolgte.

```
[student@servera ~]$ ssh root@serverb 'logger "Log files synchronized"'
Password: redhat
[student@servera ~]$
```

- 4. Mit dem Befehl **rsync** synchronisieren Sie die Verzeichnisstruktur **/var/log** auf **serverb** sicher mit dem Verzeichnis **/home/student/serverlogs** auf **servera**. Beachten Sie, dass dieses Mal nur die geänderten Protokolldateien übertragen werden.

```
[student@servera ~]$ rsync -av root@serverb:/var/log ~/serverlogs
root@serverb's password: redhat
receiving incremental file list
log/messages
log/secure
log/audit/audit.log

sent 3,496 bytes received 27,243 bytes 8,782.57 bytes/sec
total size is 11,502,695 speedup is 374.21
```

- 5. Beenden Sie **servera**.

```
[student@servera ~]$ exit
logout
Connection to servera closed.
[student@workstation]$
```

## Beenden

Führen Sie auf **workstation** das Skript **lab archive-sync finish** aus, um diese Übung zu beenden.

```
[student@workstation ~]$ lab archive-sync finish
```

Hiermit ist die angeleitete Übung beendet.

## ► Praktische Übung

# Archivieren und Übertragen von Dateien

### Leistungscheckliste

In dieser praktischen Übung archivieren und sichern Sie die Inhalte von Verzeichnissen mit den Befehlen **tar**, **rsync** und **scp**.

### Ergebnisse

Es werden folgende Fähigkeiten vermittelt:

- Synchronisieren eines Remote-Verzeichnisses in ein lokales Verzeichnis.
- Erstellen eines Archivs mit dem Inhalt eines synchronisierten Verzeichnisses.
- Sicherer Kopieren eines Archivs auf einen Remote-Host.
- Extrahieren eines Archivs.

### Bevor Sie Beginnen

Melden Sie sich als Benutzer **student** mit dem Passwort **student** bei **workstation** an.

Führen Sie auf **workstation** den Befehl **lab archive-review start** aus. Dieser Befehl führt ein Startskript aus, das ermittelt, ob die Hosts **servera** und **serverb** im Netzwerk erreichbar sind. Das Skript stellt außerdem sicher, dass die in der Übung zu erstellenden Dateien und Verzeichnisse nicht auf **serverb** und **workstation** vorhanden sind.

```
[student@workstation ~]$ lab archive-review start
```

1. Synchronisieren Sie auf **serverb** die Verzeichnisstruktur **/etc** von **servera** mit dem Verzeichnis **/configsync**.
2. Erstellen Sie mit der **gzip**-Komprimierung ein Archiv namens **configfile-backup-servera.tar.gz** mit den Inhalten des Verzeichnisses **/configsync**.
3. Kopieren Sie die Archivdatei **/root/configfile-backup-servera.tar.gz** sicher von **serverb** in das Verzeichnis **/home/student** auf **workstation** (als Benutzer **student** mit dem Passwort **student**).
4. Extrahieren Sie auf **workstation** den Inhalt des Archivs **/home/student/configfile-backup-servera.tar.gz** in das Verzeichnis **/tmp/savedconfig/**.
5. Kehren Sie auf **workstation** zum Benutzerverzeichnis **student** zurück.

```
[student@workstation savedconfig]$ cd
```

### Bewertung

Führen Sie auf **workstation** das Skript **lab archive-review grade** aus, um den Erfolg dieser praktischen Übung zu bestätigen.

```
[student@workstation ~]$ lab archive-review grade
```

## Beenden

Führen Sie auf **workstation** das Skript **lab archive-review finish** aus, um diese Übung zu beenden.

```
[student@workstation ~]$ lab archive-review finish
```

Damit ist die praktische Übung abgeschlossen.

## ► Lösung

# Archivieren und Übertragen von Dateien

### Leistungscheckliste

In dieser praktischen Übung archivieren und sichern Sie die Inhalte von Verzeichnissen mit den Befehlen **tar**, **rsync** und **scp**.

### Ergebnisse

Es werden folgende Fähigkeiten vermittelt:

- Synchronisieren eines Remote-Verzeichnisses in ein lokales Verzeichnis.
- Erstellen eines Archivs mit dem Inhalt eines synchronisierten Verzeichnisses.
- Sicherer Kopieren eines Archivs auf einen Remote-Host.
- Extrahieren eines Archivs.

### Bevor Sie Beginnen

Melden Sie sich als Benutzer **student** mit dem Passwort **student** bei **workstation** an.

Führen Sie auf **workstation** den Befehl **lab archive-review start** aus. Dieser Befehl führt ein Startskript aus, das ermittelt, ob die Hosts **servera** und **serverb** im Netzwerk erreichbar sind. Das Skript stellt außerdem sicher, dass die in der Übung zu erstellenden Dateien und Verzeichnisse nicht auf **serverb** und **workstation** vorhanden sind.

```
[student@workstation ~]$ lab archive-review start
```

1. Synchronisieren Sie auf **serverb** die Verzeichnisstruktur **/etc** von **servera** mit dem Verzeichnis **/configsync**.

- 1.1. Melden Sie sich mit dem Befehl **ssh** bei **serverb** als **student** an.

```
[student@workstation ~]$ ssh student@serverb
...output omitted...
[student@serverb ~]$
```

- 1.2. Verwenden Sie den Befehl **su**, um zum **root**-Benutzer zu wechseln, da zum Erstellen des Verzeichnisses **/configsync** Superuser-Berechtigungen erforderlich sind. In späteren Schritten werden Sie die in der Verzeichnisstruktur **/etc** vorhandenen Dateien archivieren, deren Eigentümer der **root**-Benutzer ist. Dies erfordert ebenfalls Superuser-Berechtigungen.

```
[student@serverb ~]$ su -
Password: redhat
[root@serverb ~]#
```

**Kapitel 13 |** Archivieren und Übertragen von Dateien

- 1.3. Erstellen Sie das Verzeichnis **/configsync**, in dem die synchronisierten Konfigurationsdateien von **servera** gespeichert werden sollen.

```
[root@serverb ~]# mkdir /configsync
```

- 1.4. Mit dem Befehl **rsync** synchronisieren Sie die Verzeichnisstruktur **/etc** von **servera** mit dem Verzeichnis **/configsync** auf **serverb**.

Beachten Sie, dass nur der **root**-Benutzer den gesamten Inhalt im Verzeichnis **/etc** auf **servera** lesen kann.

```
[root@serverb ~]# rsync -av root@servera:/etc /configsync
root@servera's password: redhat
receiving incremental file list
/etc/
/etc/.pwd.lock
...output omitted...
/etc/yum/protected.d -> ../dnf/protected.d
/etc/yum/vars -> ../dnf/vars

sent 10,958 bytes received 21,665,987 bytes 3,334,914.62 bytes/sec
total size is 21,615,767 speedup is 1.00
```

2. Erstellen Sie mit der **gzip**-Komprimierung ein Archiv namens **configfile-backup-servera.tar.gz** mit den Inhalten des Verzeichnisses **/configsync**.

- 2.1. Verwenden Sie den Befehl **tar** mit den **-czf**-Optionen, um ein mit **gzip** komprimiertes Archiv zu erstellen.

```
[root@serverb ~]# tar -czf configfile-backup-servera.tar.gz /configsync
tar: Removing leading `/' from member names
[root@serverb ~]#
```

- 2.2. Verwenden Sie den Befehl **tar** mit den **-tzf**-Optionen, um den Inhalt des Archivs **configfile-backup-servera.tar.gz** aufzulisten.

```
[root@serverb ~]# tar -tzf configfile-backup-servera.tar.gz
...output omitted...
configsync/etc/vimrc
configsync/etc/wgetrc
configsync/etc/xattr.conf
```

3. Kopieren Sie die Archivdatei **/root/configfile-backup-servera.tar.gz** sicher von **serverb** in das Verzeichnis **/home/student** auf **workstation** (als Benutzer **student** mit dem Passwort **student**).

```
[root@serverb ~]# scp ~/configfile-backup-servera.tar.gz \
student@workstation:/home/student
...output omitted...
student@workstation's password: student
configfile-backup-servera.tar.gz          100% 5110KB  64.5MB/s   00:00
```

4. Extrahieren Sie auf **workstation** den Inhalt des Archivs **/home/student/configfile-backup-servera.tar.gz** in das Verzeichnis **/tmp/savedconfig/**.

- 4.1. Beenden Sie **serverb**.

```
[root@serverb ~]# exit  
logout  
[student@serverb ~]$ exit  
logout  
Connection to serverb closed.  
[student@workstation]$
```

- 4.2. Erstellen Sie das Verzeichnis **/tmp/savedconfig**, in das der Inhalt des Archivs **/home/student/configfile-backup-servera.tar.gz** extrahiert wird.

```
[student@workstation ~]$ mkdir /tmp/savedconfig
```

- 4.3. Wechseln Sie in das Verzeichnis **/tmp/savedconfig**.

```
[student@workstation ~]$ cd /tmp/savedconfig  
[student@workstation savedconfig]$
```

- 4.4. Verwenden Sie den Befehl **tar** mit den **-tzf**-Optionen, um den Inhalt des Archivs **configfile-backup-servera.tar.gz** aufzulisten.

```
[student@workstation savedconfig]$ tar -tzf ~/configfile-backup-servera.tar.gz  
...output omitted...  
configsSync/etc/vimrc  
configsSync/etc/wgetrc  
configsSync/etc/xattr.conf
```

- 4.5. Verwenden Sie den Befehl **tar** mit den **-xzf**-Optionen, um den Inhalt des Archivs **/home/student/configfile-backup-servera.tar.gz** in das Verzeichnis **/tmp/savedconfig/** zu extrahieren.

```
[student@workstation savedconfig]$ tar -xzf ~/configfile-backup-servera.tar.gz  
[student@workstation savedconfig]$
```

- 4.6. Listen Sie die Verzeichnisstruktur auf, um zu verifizieren, ob das Verzeichnis Dateien aus dem Verzeichnis **/etc** enthält.

```
[student@workstation savedconfig]$ ls -lR  
. :  
total 0  
drwxr-xr-x. 3 student student 17 Feb 13 10:13 configsSync  
  
.configsSync:  
total 12  
drwxr-xr-x. 95 student student 8192 Feb 13 09:41 etc  
  
.configsSync/etc:
```

## Kapitel 13 | Archivieren und Übertragen von Dateien

```
total 1212
-rw-r--r--. 1 student student      16 Jan 16 23:41 adjtime
-rw-r--r--. 1 student student    1518 Sep 10 17:21 aliases
drwxr-xr-x. 2 student student     169 Feb  4 21:58 alternatives
...output omitted...
```

5. Kehren Sie auf **workstation** zum Benutzerverzeichnis **student** zurück.

```
[student@workstation savedconfig]$ cd
```

## Bewertung

Führen Sie auf **workstation** das Skript **lab archive-review grade** aus, um den Erfolg dieser praktischen Übung zu bestätigen.

```
[student@workstation ~]$ lab archive-review grade
```

## Beenden

Führen Sie auf **workstation** das Skript **lab archive-review finish** aus, um diese Übung zu beenden.

```
[student@workstation ~]$ lab archive-review finish
```

Damit ist die praktische Übung abgeschlossen.

# Zusammenfassung

---

In diesem Kapitel erhielten Sie folgende Informationen:

- Der Befehl **tar** erstellt eine Archivdatei aus einer Reihe von Dateien und Verzeichnissen, extrahiert Dateien aus dem Archiv und listet den Inhalt eines Archivs auf.
- Der Befehl **tar** bietet eine Reihe verschiedener Komprimierungsmethoden, um die Archivgröße zu verringern.
- Der **SSH**-Service stellt nicht nur eine sichere Remote-Shell bereit, sondern bietet auch die Befehle **scp** und **sftp** als sichere Wege zum Übertragen von Dateien von und zu einem Remote-System, das den **SSH**-Server ausführt.
- Der Befehl **rsync** synchronisiert Dateien sicher und effizient zwischen zwei Verzeichnissen, von denen sich eines auf einem Remote-System befinden kann.



## Kapitel 14

# Installieren und Aktualisieren von Softwarepaketem

### Ziel

Laden, installieren, aktualisieren und verwalten Sie Softwarepaketem von Red Hat und YUM-Paket-Repositorys.

### Ziele

- Registrieren eines Systems bei Ihrem Red Hat-Benutzerkonto und Zuweisen von Berechtigungen für Software-Updates und Support-Services mit Red Hat Subscription Management
- Erläutern, wie Software als RPM-Pakete bereitgestellt wird, und Untersuchen der auf dem System installierten Pakete mit YUM und RPM
- Suchen, Installieren und Aktualisieren der Softwarepaketem mit dem **yum**-Befehl
- Aktivieren und Deaktivieren der YUM-Repositorys von Red Hat oder Drittanbietern über einen Server
- Erläutern, wie Module die Installation bestimmter Softwareversionen ermöglichen, Auflisten, Aktivieren und Wechseln von Modul-Streams sowie Installieren und Aktualisieren von Paketen aus einem Modul

### Abschnitte

- Registrieren von Systemen für den Red Hat Support (und Test)
- Erläutern und Untersuchen von RPM-Softwarepaketem (und Test)
- Installieren und Aktualisieren von Softwarepaketem mit YUM (und angeleitete Übung)
- Aktivieren von YUM-Software-Repositorys (und angeleitete Übung)
- Verwalten von Paketmodul-Streams (und angeleitete Übung)

## Praktische Übung

Installieren und Aktualisieren von Softwarepaketen

# Registrieren von Systemen für den Red Hat Support

## Ziele

Nach Abschluss dieses Abschnitts sollten Sie in der Lage sein, ein System bei Ihrem Red Hat-Benutzerkonto zu registrieren und ihm Berechtigungen für Software-Updates und Support-Services mit Red Hat Subscription Management zuzuweisen.

## Red Hat-Subskriptionsverwaltung

Red Hat Subscription Management bietet Tools, mit deren Hilfe Rechnern Berechtigungen für Produktsubskriptionen erteilt werden können, damit Administratoren Updates für Softwarepakete erhalten und Informationen zu Supportverträgen und Subskriptionen nachverfolgen können, die von ihren Systemen verwendet werden. Standardtools wie PackageKit und **yum** können Softwarepakete und Updates durch ein von Red Hat bereitgestelltes Netzwerk zur Verteilung von Inhalten abrufen.

Mit den Tools von Red Hat Subscription Management können vier grundlegende Aufgaben durchgeführt werden:

- **Registrieren** eines Systems, um es einem Red Hat-Benutzerkonto zuzuordnen. Auf diese Weise kann der Subscription Manager eine gezielte Bestandsaufnahme des Systems erstellen. Wird es nicht mehr verwendet, so kann die Registrierung eines Systems wieder aufgehoben werden.
- **Erstellen einer Subskription** für ein System, um ihm Updates für ausgewählte Red Hat-Produkte zugänglich zu machen. Subskriptionen sind jeweils bestimmte Supportstufen, Ablaufdaten und Standard-Repositorys zugeordnet. Die Tools können entweder für eine automatische Zuordnung oder zur Auswahl einer bestimmten Berechtigung genutzt werden. Wenn sich die Anforderungen ändern, können Subskriptionen entfernt werden.
- **Aktivieren von Repositorys** für die Bereitstellung von Softwarepaketen. Für jede Subskription sind standardmäßig mehrere Repositorys aktiviert, aber andere Repositorys wie Updates oder Quellcode können nach Bedarf aktiviert oder deaktiviert werden.
- **Prüfen und Verfolgen** der Berechtigungen, die verfügbar sind oder genutzt werden. Informationen zu Subskriptionen können lokal auf einem bestimmten System oder für ein Benutzerkonto entweder auf der Seite „Subscriptions“ des Red Hat Customer Portal oder im *Subscription Asset Manager (SAM)* angezeigt werden.

## Registrieren eines Systems

Es gibt verschiedene Möglichkeiten, ein System bei Red Hat Customer Portal zu registrieren. Es gibt eine grafische Oberfläche, auf die Sie mit einer GNOME-Anwendung oder über den Web Console-Service zugreifen können. Außerdem ist ein Befehlszeilentool verfügbar.

Um ein System bei der GNOME-Anwendung zu registrieren, starten Sie Red Hat Subscription Manager, indem Sie **Activities** auswählen. Geben Sie *subscription* in das Feld **Type to search...** ein und klicken Sie auf **Red Hat Subscription Manager**. Geben Sie das entsprechende Passwort ein, wenn Sie zur Authentifizierung aufgefordert werden. Dadurch wird das folgende Fenster **Subscriptions** geöffnet:

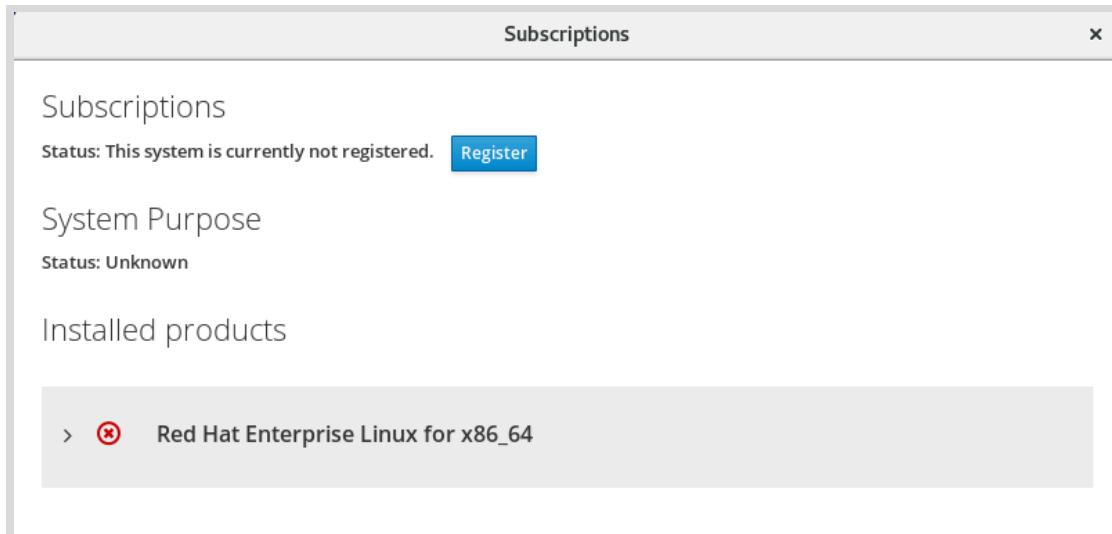


Abbildung 14.1: Hauptfenster von Red Hat Subscription Manager

Klicken Sie zum Registrieren des Systems im Fenster **Subscriptions** auf die Schaltfläche **Register**. Daraufhin wird folgendes Dialogfeld angezeigt:

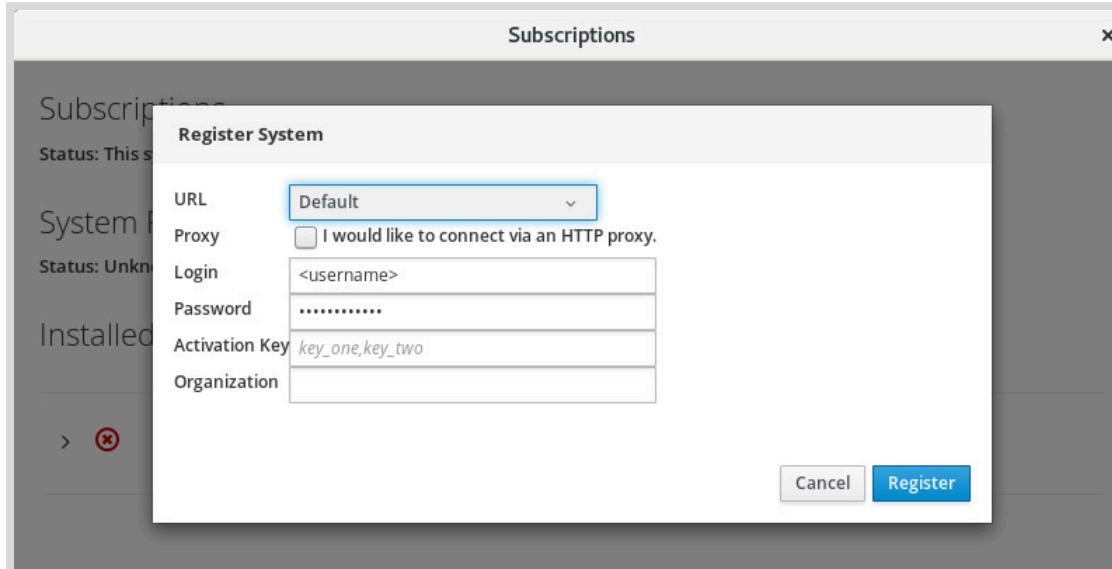


Abbildung 14.2: Dialogfeld des Red Hat Subscription Manager zur Angabe von Leistungsort und Kontoinformationen

In diesem Dialogfeld wird ein System bei einem Subskriptionsserver registriert. Der Server wird standardmäßig beim Red Hat Customer Portal registriert. Geben Sie in **Login** den Anmeldenamen und in **Password** das Passwort für das Red Hat Customer Portal-Benutzerkonto an, bei dem das System registriert werden soll, und klicken Sie auf die Schaltfläche **Register**.

Bei der Registrierung wird dem System automatisch eine Subskription beigefügt, sofern eine verfügbar ist.

Nach der Registrierung des Systems und Zuordnung einer Subskription schließen Sie das Fenster **Subscriptions**. Das System ist jetzt ordnungsgemäß für eine Subskription registriert und kann über Red Hat Updates empfangen oder neue Software installieren.

## Registrierung über die Befehlszeile

Mit dem Befehl **subscription-manager**(8) können Sie ein System auch ohne Verwendung einer grafischen Umgebung registrieren. Der Befehl **subscription-manager** ordnet ein System automatisch den am besten zum System passenden und kompatiblen Subskriptionen zu.

- Registrieren eines Systems bei einem Red Hat-Benutzerkonto:

```
[user@host ~]$ subscription-manager register --username=yourusername \
--password=yourpassword
```

- Anzeige aller verfügbaren Subskriptionen:

```
[user@host ~]$ subscription-manager list --available | less
```

- Automatische Zuordnung einer Subskription:

```
[user@host ~]$ subscription-manager attach --auto
```

- Alternativ können Sie eine Subskription aus einem bestimmten Pool aus der Liste der verfügbaren Subskriptionen anhängen:

```
[user@host ~]$ subscription-manager attach --pool=poolID
```

- Anzeige verbrauchter Subskriptionen:

```
[user@host ~]$ subscription-manager list --consumed
```

- Aufheben der Registrierung eines Systems:

```
[user@host ~]$ subscription-manager unregister
```



### Anmerkung

**subscription-manager** kann zudem auch in Kombination mit *Aktivierungsschlüsseln* verwendet werden, was das Registrieren und Zuordnen vordefinierter Subskriptionen ohne Angabe eines Benutzernamens oder Passworts gestattet. Diese Registrierungsmethode kann für automatische Installationen und Bereitstellungen sehr hilfreich sein. Aktivierungsschlüssel werden häufig durch einen Subskriptionsverwaltungs-Service, etwa den Subscription Asset Manager oder Red Hat Satellite, am Standort ausgegeben und werden in diesem Kurs nicht im Detail besprochen.

## Berechtigungszertifikate

Eine Berechtigung ist eine Subskription, die einem System zugeordnet wurde. Digitale Zertifikate dienen zum Speichern aktueller Berechtigungsinformationen auf dem lokalen System. Nach der Registrierung werden Berechtigungszertifikate im Verzeichnis **/etc/pki** und seinen Unterverzeichnissen gespeichert.

- **/etc/pki/product** enthält Zertifikate, die die auf dem System installierten Red -Produkte angeben.

- **/etc/pki/consumer** enthält Zertifikate, die das Red Hat-Benutzerkonto identifizieren, bei dem das System registriert ist
- **/etc/pki/entitlement** enthält Zertifikate, die die dem System zugeordneten Subskriptionen angeben.

Die Zertifikate können direkt mit dem Dienstprogramm **rct** angezeigt werden, aber die **subscription-manager**-Tools bieten einfachere Möglichkeiten, um die dem System zugeordneten Subskriptionen zu prüfen.



#### Literaturhinweise

Manpages **subscription-manager**(8) und **rct**(8)

#### Erste Schritte mit Red Hat Subscription Management

<https://access.redhat.com/site/articles/433903>

## ► Quiz

# Registrieren von Systemen für den Red Hat Support

Wählen Sie die richtige Antwort auf die folgenden Fragen aus:

- ▶ 1. Mit welchem Befehl können Sie ein System auch ohne Verwendung einer grafischen Umgebung registrieren?
  - a. **rct**
  - b. **subscription-manager**
  - c. **rpm**
  - d. **yum**
- ▶ 2. Mit welchem GUI-Tool wird ein System registriert und abonniert?
  - a. PackageKit
  - b. **gpk-application**
  - c. Red Hat Subscription Manager
  - d. **gnome-software**
- ▶ 3. Welche Aufgabe(n) kann (können) mit den Tools von Red Hat Hat Subscription Management durchgeführt werden?
  - a. Registrieren eines Systems
  - b. Abonnieren eines Systems
  - c. Aktivieren von Repositorys
  - d. Überprüfen und Verfolgen von Berechtigungen
  - e. Alle aufgeführten Optionen.

## ► Lösung

# Registrieren von Systemen für den Red Hat Support

Wählen Sie die richtige Antwort auf die folgenden Fragen aus:

- ▶ 1. Mit welchem Befehl können Sie ein System auch ohne Verwendung einer grafischen Umgebung registrieren?
  - a. rct
  - b. **subscription-manager**
  - c. rpm
  - d. yum
- ▶ 2. Mit welchem GUI-Tool wird ein System registriert und abonniert?
  - a. PackageKit
  - b. gpk-application
  - c. Red Hat Subscription Manager
  - d. gnome-software
- ▶ 3. Welche Aufgabe(n) kann (können) mit den Tools von Red Hat Subscription Management durchgeführt werden?
  - a. Registrieren eines Systems
  - b. Abonnieren eines Systems
  - c. Aktivieren von Repositorys
  - d. Überprüfen und Verfolgen von Berechtigungen
  - e. Alle aufgeführten Optionen.

# Erläutern und Untersuchen von RPM-Softwarepaketen

---

## Ziele

Nach Abschluss dieses Abschnitts sollten Sie in der Lage sein zu erläutern, wie Software als RPM-Pakete bereitgestellt wird. Zudem sollten Sie die auf dem System installierten Pakete mit YUM und RPM untersuchen können.

## Software-Pakete und RPM

Der RPM Package Manager, der ursprünglich von Red Hat entwickelt wurde, bietet eine Standardmethode zum Packen von Software für die Distribution. Das Verwalten von Software in Form von *RPM-Paketen* ist wesentlich einfacher als das Arbeiten mit Software, die einfach aus einem Archiv in ein Dateisystem extrahiert wurde. Bei Verwendung von RPM-Paketen können Administratoren nachvollziehen, welche Dateien von dem Softwarepaket installiert wurden, welche Dateien bei der Deinstallation des Pakets entfernt werden müssen und ob vor der Installation eines Pakets alle erforderliche Unterstützungspakete vorhanden sind. Informationen zu installierten Paketen werden in jedem System in einer lokalen RPM-Datenbank gespeichert. Sämtliche Red Hat-Software für Red Hat Enterprise Linux wird als RPM-Paket bereitgestellt.

Die Namen der RPM-Paketdateien bestehen aus vier Elementen (sowie dem Suffix **.rpm**): **name-version-release.architecture**:

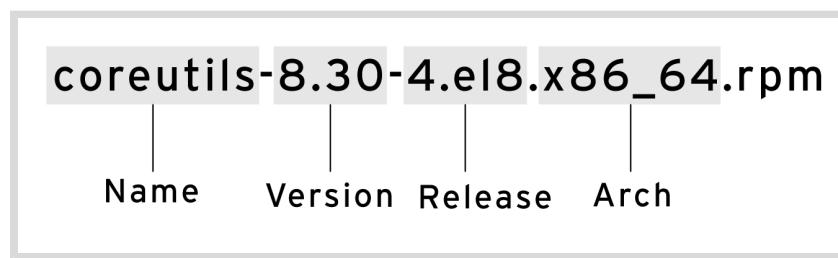


Abbildung 14.3: Elemente von RPM-Dateinamen

- NAME ist ein Wort oder mehrere Wörter zur Beschreibung des Inhalts (coreutils).
- VERSION ist die Versionsnummer der ursprünglichen Software (8.30).
- RELEASE ist die Release-Nummer des Pakets auf Grundlage dieser Version und wird durch den Verpackenden festgelegt, der nicht mit dem ursprünglichen Software-Entwickler identisch sein muss (4.el8).
- ARCH ist die Prozessorarchitektur, auf der das Paket ausgeführt werden soll. **noarch** zeigt an, dass die Inhalte dieses Pakets nicht architekturspezifisch sind (im Gegensatz zu **x86\_64** für 64-Bit, **aarch64** für 64-Bit ARM usw.).

Für die Installation von Paketen aus Repositorys ist lediglich der Paketname erforderlich. Wenn mehrere Versionen vorhanden sind, wird das Paket mit der höheren Versionsnummer installiert. Wenn mehrere Releases einer einzelnen Version vorhanden sind, wird das Paket mit der höheren Release-Nummer installiert.

Jedes RPM-Paket ist ein spezielles Archiv, das aus drei Komponenten besteht:

- Die durch das Paket zu installierenden Dateien.

## Kapitel 14 | Installieren und Aktualisieren von Softwarepaketen

- Informationen zum Paket (Metadaten), etwa Name, Version, Release, Architektur; eine Zusammenfassung und Beschreibung des Pakets; ob die Installation anderer Pakete erforderlich ist; Lizenzinformationen; ein Änderungsprotokoll für das Paket und andere Details.
- Skripts, die bei der Installation, Aktualisierung oder Entfernung dieses Pakets oder anderer Pakete ausgeführt werden können.

Normalerweise signieren Softwareanbieter RPM-Pakete digital mit GPG-Schlüsseln (Red Hat signiert alle veröffentlichten Pakete digital). Das RPM-System überprüft die Paketintegrität, indem es bestätigt, dass das Paket mit dem entsprechenden GPG-Schlüssel signiert wurde. Das RPM-System lehnt die Installation eines Pakets ab, wenn die GPG-Signatur nicht übereinstimmt.

## Aktualisieren von Software mit RPM-Paketen

Red Hat generiert ein komplettes RPM-Paket zum Aktualisieren von Software. Ein Administrator, der dieses Paket installiert, erhält nur die aktuellste Version des Pakets. Bei Red Hat ist es nicht erforderlich, dass ältere Pakete installiert und dann gepatcht werden. Um Software zu aktualisieren, entfernt RPM die ältere Version des Pakets und installiert dann die neue Version. Bei Updates werden normalerweise Konfigurationsdateien beibehalten. Der Paketersteller der neuen Version definiert jedoch das genaue Verhalten.

In den meisten Fällen kann immer nur eine Version oder Release eines Pakets installiert sein. Wenn ein Paket jedoch so aufgebaut ist, dass keine Konflikte zwischen Dateinamen bestehen, können mehrere Versionen installiert werden. Das wichtigste Beispiel hierfür ist das Paket **kernel1**. Da ein neuer Kernel nur durch Starten mit diesem Kernel getestet werden kann, ist das Paket absichtlich so ausgelegt, dass mehrere Versionen gleichzeitig installiert werden können. Falls der neue Kernel nicht bootet, ist der alte Kernel weiterhin verfügbar und kann gestartet werden.

## Überprüfen von RPM-Paketen

Beim Dienstprogramm **rpm** handelt es sich um ein systemnahe Werkzeug, mit dem Informationen zum Inhalt von Paketdateien und installierten Paketen ermittelt werden können. Standardmäßig werden Informationen aus der lokalen Datenbank der installierten Pakete abgerufen. Sie können jedoch mit der Option **-p** angeben, dass Sie Informationen über eine heruntergeladene Paketdatei erhalten möchten. Dies ist hilfreich, um den Inhalt der Paketdatei vor der Installation zu überprüfen.

Die allgemeine Abfrageform ist:

- **rpm -q [select-options] [query-options]**

### RPM-Abfragen: Allgemeine Informationen zu installierten Paketen

- **rpm -qa**: Listet alle installierten Pakete auf
- **rpm -qf FILENAME**: Ermittelt, welches Paket FILENAME bereitstellt

```
[user@host ~]$ rpm -qf /etc/yum.repos.d  
redhat-release-8.0-0.39.el8.x86_64
```

### RPM-Abfragen: Informationen über bestimmte Pakete

- **rpm -q**: Listet auf, welche Version des Pakets aktuell installiert ist

```
[user@host ~]$ rpm -q yum  
yum-4.0.9.2-4.el8.noarch
```

- **rpm -qi**: Zeigt detaillierte Informationen zu einem Paket an
- **rpm -ql**: Listet die durch das Paket installierten Dateien auf

```
[user@host ~]$ rpm -ql yum  
/etc/yum.conf  
/etc/yum/pluginconf.d  
/etc/yum/protected.d  
/etc/yum/vars  
/usr/bin/yum  
/usr/share/man/man1/yum-aliases.1.gz  
/usr/share/man/man5/yum.conf.5.gz  
/usr/share/man/man8/yum-shell.8.gz  
/usr/share/man/man8/yum.8.gz
```

- **rpm -qc**: Listet nur die durch das Paket installierten Konfigurationsdateien auf

```
[user@host ~]$ rpm -qc openssh-clients  
/etc/ssh/ssh_config  
/etc/ssh/ssh_config.d/05-redhat.conf
```

- **rpm -qd**: Listet nur die durch das Paket installierten Dokumentationsdateien auf

```
[user@host ~]$ rpm -qd openssh-clients  
/usr/share/man/man1/scp.1.gz  
/usr/share/man/man1/sftp.1.gz  
/usr/share/man/man1/ssh-add.1.gz  
/usr/share/man/man1/ssh-agent.1.gz  
/usr/share/man/man1/ssh-copy-id.1.gz  
/usr/share/man/man1/ssh-keyscan.1.gz  
/usr/share/man/man1/ssh.1.gz  
/usr/share/man/man5/ssh_config.5.gz  
/usr/share/man/man8/ssh-pkcs11-helper.8.gz
```

- **rpm -q --scripts**: Listet Shell-Skripts auf, die vor oder nach der Installation oder Deinstallation des Pakets ausgeführt werden

```
[user@host ~]$ rpm -q --scripts openssh-server  
preinstall scriptlet (using /bin/sh):  
getent group sshd >/dev/null || groupadd -g 74 -r sshd || :  
getent passwd sshd >/dev/null || \  
    useradd -c "Privilege-separated SSH" -u 74 -g sshd \  
    -s /sbin/nologin -r -d /var/empty/sshd sshd 2> /dev/null || :  
postinstall scriptlet (using /bin/sh):  
  
if [ $1 -eq 1 ] ; then  
    # Initial installation  
    /usr/bin/systemctl preset sshd.service sshd.socket >/dev/null 2>&1 || :
```

```

fi
preuninstall scriptlet (using /bin/sh):

if [ $1 -eq 0 ] ; then
    # Package removal, not upgrade
    /usr/bin/systemctl --no-reload disable sshd.service sshd.socket > /dev/
null 2>&1 || :
    /usr/bin/systemctl stop sshd.service sshd.socket > /dev/null 2>&1 || :
fi
postuninstall scriptlet (using /bin/sh):

/usr/bin/systemctl daemon-reload >/dev/null 2>&1 || :
if [ $1 -ge 1 ] ; then
    # Package upgrade, not uninstall
    /usr/bin/systemctl try-restart sshd.service >/dev/null 2>&1 || :
fi

```

- **rpm -q --changelog**: Listet die Änderungsinformationen für das Paket auf

```
[user@host ~]$ rpm -q --changelog audit
* Wed Jan 09 2019 Steve Grubb <sgrubb@redhat.com> 3.0-0.10.20180831git0047a6c
resolves: rhbz#1655270] Message "audit: backlog limit exceeded" reported
- Fix annobin failure

* Fri Dec 07 2018 Steve Grubb <sgrubb@redhat.com> 3.0-0.8.20180831git0047a6c
resolves: rhbz#1639745 - build requires go-toolset-7 which is not available
resolves: rhbz#1643567 - service auditd stop exits prematurely
resolves: rhbz#1616428 - Update git snapshot of audit package
- Remove static libs subpackage
...output omitted...
```

Abfragen von lokalen Paketdateien:

```
[user@host ~]$ ls -l wonderwidgets-1.0-4.x86_64.rpm
-rw-rw-r--. 1 user user 257 Mar 13 20:06 wonderwidgets-1.0-4.x86_64.rpm
[user@host ~]$ rpm -qlp wonderwidgets-1.0-4.x86_64.rpm
/etc/wonderwidgets.conf
/usr/bin/wonderwidgets
/usr/share/doc/wonderwidgets-1.0
/usr/share/doc/wonderwidgets-1.0/README.txt
```

## Installieren von RPM-Paketen

Mit dem Befehl **rpm** können Sie auch ein RPM-Paket installieren, das Sie in Ihr lokales Verzeichnis heruntergeladen haben.

```
[root@host ~]# rpm -ivh wonderwidgets-1.0-4.x86_64.rpm
Verifying...  #####[100%]
Preparing...   #####[100%]
Updating / installing...
  1:wonderwidgets-1.0-4                                     #####[100%]
[root@host ~]#
```

Im nächsten Abschnitt dieses Kapitels wird jedoch ein leistungsfähigeres Tool, **yum**, zum Verwalten der RPM-Installation und -Updates über die Befehlszeile beschrieben.



### Warnung

Seien Sie bei der Installation von Drittanbieterpaketem vorsichtig, nicht nur wegen der Software, die möglicherweise installiert wird, sondern auch, da das RPM-Paket beliebige Skripts enthalten kann, die im Rahmen des Installationsvorgangs als **root**-Benutzer ausgeführt werden.



### Anmerkung

Sie können Dateien aus einer RPM-Paketdatei extrahieren, ohne das Paket zu installieren. Das Dienstprogramm **rpm2cpio** kann den Inhalt des RPM an ein spezielles Archivierungstool namens **cpio** übergeben, das alle Dateien oder einzelne Dateien extrahieren kann.

Geben Sie die Ausgabe von **rpm2cpio PACKAGEFILE.rpm** über eine Pipe an **cpio -id** weiter, um alle im RPM-Paket gespeicherten Dateien zu extrahieren. Unterverzeichnis-Baumstrukturen werden nach Bedarf relativ zum aktuellen Arbeitsverzeichnis erstellt.

```
[user@host tmp-extract]$ rpm2cpio wonderwidgets-1.0-4.x86_64.rpm | cpio -id
```

Einzelne Dateien werden unter Angabe des Dateipfads extrahiert:

```
[user@host ~]$ rpm2cpio wonderwidgets-1.0-4.x86_64.rpm | cpio -id "*txt"
11 blocks
[user@host ~]$ ls -l usr/share/doc/wonderwidgets-1.0/
total 4
-rw-r--r--. 1 user user 76 Feb 13 19:27 README.txt
```

## Übersicht über RPM-Abfragebefehle

Installierte Pakete können mit dem Befehl **rpm** direkt abgefragt werden. Fügen Sie die Option **-q** hinzu, um eine Paketdatei vor der Installation abzufragen.

| Befehl              | Aufgabe                                                   |
|---------------------|-----------------------------------------------------------|
| <b>rpm -qa</b>      | Listet alle aktuell installierten RPM-Pakete auf          |
| <b>rpm -q NAME</b>  | Zeigt die auf dem System installierte Version von NAME an |
| <b>rpm -qi NAME</b> | Zeigt detaillierte Informationen zu einem Paket an        |
| <b>rpm -ql NAME</b> | Listet alle in einem Paket enthaltenen Dateien auf        |

| Befehl                         | Aufgabe                                                                                                                |
|--------------------------------|------------------------------------------------------------------------------------------------------------------------|
| <b>rpm -qc NAME</b>            | Listet die in einem Paket enthaltenen Konfigurationsdateien auf                                                        |
| <b>rpm -qd NAME</b>            | Listet die in einem Paket enthaltenen Dokumentationsdateien auf                                                        |
| <b>rpm -q --changelog NAME</b> | Zeigt eine kurze Zusammenfassung der Gründe für eine neue Paketversion an                                              |
| <b>rpm -q --scripts NAME</b>   | Zeigt die Shell-Skripts an, die bei der Installation, Aktualisierung oder Deinstallation von Paketen ausgeführt werden |



### Literaturhinweise

Manpages **rpm(8)**, **rpm2cpio(8)**, **cpio(1)** und **rpmkeys(8)**

## ► Angeleitete Übung

# Erläutern und Untersuchen von RPM-Softwarepaketen

In dieser Übung sammeln Sie Informationen zu einem Paket von einem Drittanbieter, extrahieren daraus Dateien zur Überprüfung und installieren es anschließend auf einem Server.

## Ergebnisse

Sie sollten in der Lage sein, ein Paket zu installieren, das nicht über Software-Repositories auf einem Server bereitgestellt wird.

## Bevor Sie Beginnen

Melden Sie sich als Benutzer **student** mit dem Passwort **student** bei **workstation** an.

Führen Sie auf der **workstation** den Befehl **lab software-rpm start** aus. Dieses Skript führt ein Startskript aus, das ermittelt, ob der Host **servera** im Netzwerk erreichbar ist. Das Skript lädt auch das Paket *rhcса-script-1.0.0-1.noarch.rpm* in das Verzeichnis **/home/student** auf **servera**.

```
[student@workstation ~]$ lab software-rpm start
```

- 1. Melden Sie sich mit dem Befehl **ssh** bei **servera** als Benutzer **student** an.

```
[student@workstation ~]$ ssh student@servera
...output omitted...
[student@servera ~]$
```

- 2. Zeigen Sie für das Paket *rhcса-script-1.0.0-1.noarch.rpm* die Paketinformationen an und listen Sie enthaltene Dateien auf. Zeigen Sie auch das Skript an, das ausgeführt wird, wenn das Paket installiert oder deinstalliert wird.

- 2.1. Zeigen Sie Informationen für das Paket *rhcса-script-1.0.0-1.noarch.rpm* an.

```
[student@servera ~]$ rpm -q -p rhcsа-script-1.0.0-1.noarch.rpm -i
Name      : rhcsа-script
Version   : 1.0.0
Release   : 1
Architecture: noarch
Install Date: (not installed)
Group     : System
Size      : 1056
License   : GPL
Signature : (none)
Source RPM : rhcsа-script-1.0.0-1.src.rpm
Build Date : Wed 06 Mar 2019 03:59:46 PM IST
```

**Kapitel 14 |** Installieren und Aktualisieren von Softwarepaketen

```
Build Host : foundation0.ilt.example.com
Relocations : (not relocatable)
Packager : Snehangshu Karmakar
URL : http://example.com
Summary : RHCSA Practice Script
Description :
A RHCSA practice script.
The package changes the motd.
```

- 2.2. Listen Sie für das Paket *rhcsa-script-1.0.0-1.noarch.rpm* die enthaltenen Dateien auf.

```
[student@servera ~]$ rpm -q -p rhcsa-script-1.0.0-1.noarch.rpm -l
/opt/rhcsa-script/mymotd
```

- 2.3. Zeigen Sie das Skript an, das ausgeführt wird, wenn das Paket *rhcsa-script-1.0.0-1.noarch.rpm* installiert oder deinstalliert wird.

```
[student@servera ~]$ rpm -q -p rhcsa-script-1.0.0-1.noarch.rpm --scripts
preinstall scriptlet (using /bin/sh):
if [ "$1" == "2" ]; then
    if [ -e /etc/motd.orig ]; then
        mv -f /etc/motd.orig /etc/motd
    fi
fi
postinstall scriptlet (using /bin/sh):
...output omitted...
```

- 3. Extrahieren Sie den Inhalt des Pakets *rhcsa-script-1.0.0-1.noarch.rpm* im Verzeichnis **/home/student**.

- 3.1. Verwenden Sie die Befehle **rpm2cpio** und **cpio -tv**, um die Dateien im Paket *rhcsa-script-1.0.0-1.noarch.rpm* aufzulisten.

```
[student@servera ~]$ rpm2cpio rhcsa-script-1.0.0-1.noarch.rpm | cpio -tv
-rw-r--r-- 1 root      root     1056 Mar  6 15:59 ./opt/rhcsa-script/mymotd
3 blocks
```

- 3.2. Extrahieren Sie alle Dateien des Pakets *rhcsa-script-1.0.0-1.noarch.rpm* im Verzeichnis **/home/student**. Verwenden Sie die Befehle **rpm2cpio** und **cpio -idv**, um die Dateien zu extrahieren und die Hauptverzeichnisse zu erstellen, wenn dies im ausführlichen Modus erforderlich ist.

```
[student@servera ~]$ rpm2cpio rhcsa-script-1.0.0-1.noarch.rpm | cpio -idv
./opt/rhcsa-script/mymotd
3 blocks
```

- 3.3. Überprüfen Sie die extrahierten Dateien im Verzeichnis **/home/student/opt**.

```
[student@servera ~]$ ls -lR opt
opt:
total 0
drwxrwxr-x. 2 student student 20 Mar 7 14:44 rhcsa-script

opt/rhcsa-script:
total 4
-rw-r--r--. 1 student student 1056 Mar 7 14:44 mymotd
```

- 4. Installieren Sie das Paket *rhcsa-script-1.0.0-1.noarch.rpm*. Mit dem Befehl **sudo** können Sie Superuser-Berechtigungen zum Installieren des Pakets erhalten.

- 4.1. Mit dem Befehl **sudo rpm -ivh** können Sie das RPM-Paket *rhcsa-script-1.0.0-1.noarch.rpm* installieren.

```
[student@servera ~]$ sudo rpm -ivh rhcsa-script-1.0.0-1.noarch.rpm
[sudo] password for student: student
Verifying... ################################ [100%]
Preparing... ################################ [100%]
Updating / installing...
 1:rhcsa-script-1.0.0-1 ################################ [100%]
[student@servera ~]$
```

- 4.2. Mit dem Befehl **rpm** können Sie überprüfen, ob das Paket installiert ist.

```
[student@servera ~]$ rpm -q rhcsa-script
rhcsa-script-1.0.0-1.noarch
```

- 5. Beenden Sie **servera**.

```
[student@servera ~]$ exit
logout
Connection to servera closed.
[student@workstation ~]$
```

## Beenden

Führen Sie auf **workstation** das Skript **lab software-rpm finish** aus, um diese Übung zu beenden. Dieses Skript entfernt alle während der Übung auf **servera** installierten Pakete.

```
[student@workstation ~]$ lab software-rpm finish
```

Hiermit ist die angeleitete Übung beendet.

# Installieren und Aktualisieren von Softwarepaketen mit YUM

---

## Ziele

Nach Abschluss dieses Abschnitts sollten Sie in der Lage sein, Softwarepakte mit dem Befehl **yum** zu suchen, zu installieren und zu aktualisieren.

## Verwalten von Softwarepaketen mit Yum

Mit dem Low-Level-Befehl **rpm** können Pakete installiert werden. Er funktioniert jedoch nicht mit Paket-Repositorys und löst keine Abhängigkeiten aus mehreren Quellen automatisch auf.

Yum stellt ein besseres System für die Verwaltung von RPM-basierten Softwareinstallationen und -Updates dar. Mit dem Befehl **yum** können Sie Softwarepakte und deren Abhängigkeiten installieren, aktualisieren, entfernen und Informationen dazu abrufen. Sie können einen Verlauf der durchgeführten Transaktionen abrufen und mit mehreren Software-Repositorys von Red Hat und Drittanbietern arbeiten.

### Suchen von Software mit Yum

- **yum help** zeigt die Syntax an.
- **yum list** zeigt installierte und verfügbare Pakete an.

```
[user@host ~]$ yum list 'http*'
Available Packages
http-parser.i686          2.8.0-2.e18          rhel8-appstream
http-parser.x86_64          2.8.0-2.e18          rhel8-appstream
httpcomponents-client.noarch 4.5.5-4.module+el8+2452+b359bfcd rhel8-appstream
httpcomponents-core.noarch   4.4.10-3.module+el8+2452+b359bfcd rhel8-appstream
httpd.x86_64                2.4.37-7.module+el8+2443+605475b7 rhel8-appstream
httpd-devel.x86_64          2.4.37-7.module+el8+2443+605475b7 rhel8-appstream
httpd-filesystem.noarch     2.4.37-7.module+el8+2443+605475b7 rhel8-appstream
httpd-manual.noarch         2.4.37-7.module+el8+2443+605475b7 rhel8-appstream
httpd-tools.x86_64          2.4.37-7.module+el8+2443+605475b7 rhel8-appstream
```

- **yum search KEYWORD** führt Pakete nur nach Schlüsselbegriffen auf, die in den Namens- und Zusammenfassungsfeldern gefunden werden.

Wenn Sie nach Paketen mit dem Begriff „web server“ in ihren Namens- und Beschreibungsfeldern suchen möchten, verwenden Sie **search all**:

```
[user@host ~]$ yum search all 'web server'
=====
Summary & Description Matched: web server =====
pcp-pmda-weblog.x86_64 : Performance Co-Pilot (PCP) metrics from web server logs
nginx.x86_64 : A high performance web server and reverse proxy server
=====
Summary Matched: web server =====
libcurl.x86_64 : A library for getting files from web servers
libcurl.i686 : A library for getting files from web servers
libcurl.x86_64 : A library for getting files from web servers
```

```
===== Description Matched: web server =====
httpd.x86_64 : Apache HTTP Server
git-instaweb.x86_64 : Repository browser in gitweb
...output omitted...
```

- **yum info PACKAGE NAME** gibt detaillierte Informationen über ein Paket einschließlich des für die Installation erforderlichen Speicherplatzes zurück.

So rufen Sie Informationen über den Apache HTTP Server ab:

```
[user@host ~]$ yum info httpd
Available Packages
Name        : httpd
Version     : 2.4.37
Release    : 7.module+el8+2443+605475b7
Arch       : x86_64
Size        : 1.4 M
Source      : httpd-2.4.37-7.module+el8+2443+605475b7.src.rpm
Repo        : rhel8-appstream
Summary     : Apache HTTP Server
URL         : https://httpd.apache.org/
License     : ASL 2.0
Description  : The Apache HTTP Server is a powerful, efficient, and extensible
               : web server.
```

- **yum provides PATHNAME** zeigt Pakete an, die zum angegebenen Pfadnamen passen (enthält oft Platzhalterzeichen).

Zur Suche von Paketen, die das Verzeichnis **/var/www/html** bereitstellen, verwenden Sie:

```
[user@host ~]$ yum provides /var/www/html
httpd-filesystem-2.4.37-7.module+el8+2443+605475b7.noarch : The basic directory
layout for the Apache HTTP server
Repo        : rhel8-appstream
Matched from:
Filename   : /var/www/html
```

## Installieren und Verwalten von Software mit yum

- **yum install PACKAGE NAME** ruft Softwarepakete (einschließlich Abhängigkeiten) ab und installiert sie.

```
[user@host ~]$ yum install httpd
Dependencies resolved.
=====
 Package          Arch      Version       Repository      Size
=====
 Installing:
  httpd           x86_64    2.4.37-7.module...
  Installing dependencies:
   apr             x86_64    1.6.3-8.el8      rhel8-appstream  125 k
   apr-util        x86_64    1.6.1-6.el8      rhel8-appstream  105 k
...output omitted...
Transaction Summary
```

```
=====
Install 9 Packages

Total download size: 2.0 M
Installed size: 5.4 M
Is this ok [y/N]: y
Downloading Packages:
(1/9): apr-util-bdb-1.6.1-6.el8.x86_64.rpm           464 kB/s | 25 kB   00:00
(2/9): apr-1.6.3-8.el8.x86_64.rpm                     1.9 MB/s | 125 kB  00:00
(3/9): apr-util-1.6.1-6.el8.x86_64.rpm               1.3 MB/s | 105 kB  00:00
...output omitted...
Total   8.6 MB/s | 2.0 MB  00:00

Running transaction check
Transaction check succeeded.

Running transaction test
Transaction test succeeded.

Running transaction
  Preparing          :                           1/1
  Installing        : apr-1.6.3-8.el8.x86_64      1/9
  Running scriptlet: apr-1.6.3-8.el8.x86_64      1/9
  Installing        : apr-util-bdb-1.6.1-6.el8.x86_64 2/9
...output omitted...
Installed:
  httpd-2.4.37-7.module+el8+2443+605475b7.x86_64 apr-util-bdb-1.6.1-6.el8.x86_64
  apr-util-openssl-1.6.1-6.el8.x86_64             apr-1.6.3-8.el8.x86_64
...output omitted...
Complete!
```

- **yum update PACKAGE NAME** ruft eine neuere Version des angegebenen Pakets samt den zugehörigen Abhängigkeiten ab und installiert sie. Im Allgemeinen wird versucht, vorhandene Konfigurationsdateien beizubehalten; in manchen Fällen werden sie allerdings umbenannt, wenn das Paketprogramm davon ausgeht, dass die alten Dateien nach dem Update nicht mehr funktionieren. Wird PACKAGE NAME nicht angegeben, werden alle relevanten Updates installiert.

```
[user@host ~]$ sudo yum update
```

Da ein neuer Kernel nur durch Starten mit diesem Kernel getestet werden kann, ist das Paket absichtlich so ausgelegt, dass mehrere Versionen gleichzeitig installiert werden können. Falls der neue Kernel nicht bootet, ist der alte Kernel weiterhin verfügbar. Die Verwendung von **yum update kernel** führt tatsächlich zu einer *Installation* des neuen Kernels. Die Konfigurationsdateien enthalten eine Liste von Paketen, die *immer installiert* werden sollen, auch wenn der Administrator ein Update anfordert.



### Anmerkung

Mit **yum list kernel** führen Sie alle installierbaren und verfügbaren Kernel auf. Den aktuell ausgeführten Kernel zeigen Sie mit dem Befehl **uname** an. Mit der Option **-r** werden nur Version und Release des Kernels angezeigt, während mit der Option **-a** Kernel-Release und Zusatzinformationen angezeigt werden.

```
[user@host ~]$ yum list kernel
Installed Packages
kernel.x86_64          4.18.0-60.el8      @anaconda
kernel.x86_64          4.18.0-67.el8      @rhel-8-for-x86_64-baseos-htb-rpms
[user@host ~]$ uname -r
4.18.0-60.el8.x86_64
[user@host ~]$ uname -a
Linux host.lab.example.com 4.18.0-60.el8.x86_64 #1 SMP Fri Jan 11 19:08:11 UTC
2019 x86_64 x86_64 x86_64 GNU/Linux
```

- **yum remove PACKAGE NAME** entfernt ein installiertes Softwarepaket samt allen unterstützten Paketen.

```
[user@host ~]$ sudo yum remove httpd
```



### Warnung

Der Befehl **yum remove** entfernt die aufgeführten Pakete *und alle Pakete, die das zu entfernende Paket erfordern* (und Pakete, die wiederum diese Pakete benötigen usw.). Da es auf diese Weise zu unerwarteten Löschvorgängen bei Paketen kommen kann, sollten Sie die Liste der zu entfernenden Pakete stets sorgfältig prüfen.

## Installieren und Verwalten von Softwaregruppen mit yum

- Bei **yum** gibt es auch das Konzept der *Komponentengruppen*. Dabei handelt es sich um Sammlungen verwandter Software, die für einen bestimmten Zweck gemeinsam installiert werden. In Red Hat Enterprise Linux 8 gibt es zwei Arten von Gruppen. Reguläre Gruppen sind Sammlungen von Paketen. *Umgebungsgruppen* sind Sammlungen regulärer Gruppen. Die von einer Gruppe bereitgestellten Pakete oder Gruppen können **obligatorisch** (müssen also bei Installation der Gruppe installiert werden), **standardmäßig** (normalerweise bei der Installation der Gruppe installiert) oder **optional** sein (werden bei Installation der Gruppe nur auf spezifische Nachfrage installiert).

Wie bei **yum list** zeigt der Befehl **yum group list** die Namen installierter und verfügbarer Gruppen an.

```
[user@host ~]$ yum group list
Available Environment Groups:
  Server with GUI
  Minimal Install
  Server
  ...output omitted...
Available Groups:
```

```
Container Management
.NET Core Development
RPM Development Tools
...output omitted...
```

Manche Gruppen werden normalerweise im Rahmen von Umgebungsgruppen installiert und sind standardmäßig versteckt. Listen Sie diese ausgeblendeten Gruppen mit dem Befehl **yum group list hidden** auf.

- **yum group info** zeigt Informationen zu einer Gruppe an. Sie enthalten eine Liste obligatorischer, standardmäßiger und optionaler Paketnamen.

```
[user@host ~]$ yum group info "RPM Development Tools"
Group: RPM Development Tools
Description: These tools include core development tools such rpmbuild.
Mandatory Packages:
    redhat-rpm-config
    rpm-build
Default Packages:
    rpmdevtools
Optional Packages:
    rpmlint
```

- **yum group install** installiert eine Gruppe, die ihre obligatorischen und standardmäßigen Pakete sowie Pakete installiert, von denen diese abhängig sind.

```
[user@host ~]$ sudo yum group install "RPM Development Tools"
...output omitted...
Installing Groups:
  RPM Development Tools

Transaction Summary
=====
Install 64 Packages

Total download size: 21 M
Installed size: 62 M
Is this ok [y/N]: y
...output omitted...
```



### Wichtig

Das Verhalten von Yum-Gruppen hat sich ab Red Hat Enterprise Linux 7 geändert. In RHEL 7 und späteren Versionen werden Gruppen als *Objekte* behandelt und vom System verfolgt. Wenn eine installierte Gruppe aktualisiert wird und sie über das Yum-Repository mit neuen obligatorischen oder standardmäßigen Paketen ergänzt wurde, werden diese neuen Pakete bei der Aktualisierung installiert.

In RHEL 6 und früheren Versionen gilt eine Gruppe als installiert, wenn alle obligatorischen Pakete installiert wurden oder wenn sie keine obligatorischen Pakete enthielt oder wenn standardmäßige oder optionale Pakete installiert sind. Ab RHEL 7 gilt eine Gruppe *nur* dann als installiert, wenn sie mit **yum group install** installiert wurde. Mit dem Befehl **yum group mark install GROUPNAME** können Sie eine Gruppe als installiert kennzeichnen und alle noch fehlenden Pakete und ihre Abhängigkeiten werden beim nächsten Update installiert.

Schlussendlich gab es in RHEL 6 und früheren Versionen nicht die Zweiwort-Form der **yum group**-Befehle. Mit anderen Worten: In RHEL 6 war der Befehl **yum grouplist** vorhanden, aber in RHEL 7 und RHEL 8 war der entsprechende Befehl **yum group list** nicht verfügbar.

### Anzeige des Transaktionsverlaufs

- Alle Transaktionen, mit denen Software installiert und entfernt wird, werden in **/var/log/dnf.rpm.log** protokolliert.

```
[user@host ~]$ tail -5 /var/log/dnf.rpm.log
2019-02-26T18:27:00Z SUBDEBUG Installed: rpm-build-4.14.2-9.el8.x86_64
2019-02-26T18:27:01Z SUBDEBUG Installed: rpm-build-4.14.2-9.el8.x86_64
2019-02-26T18:27:01Z SUBDEBUG Installed: rpmdevtools-8.10-7.el8.noarch
2019-02-26T18:27:01Z SUBDEBUG Installed: rpmdevtools-8.10-7.el8.noarch
2019-02-26T18:38:40Z INFO --- logging initialized ---
```

- yum history** zeigt eine Zusammenfassung der Transaktionen zum Installieren und Entfernen an.

| [user@host ~]\$ sudo yum history |                          |                  |           |         |
|----------------------------------|--------------------------|------------------|-----------|---------|
| ID                               | Command line             | Date and time    | Action(s) | Altered |
| 7                                | group install RPM Develo | 2019-02-26 13:26 | Install   | 65      |
| 6                                | update kernel            | 2019-02-26 11:41 | Install   | 4       |
| 5                                | install httpd            | 2019-02-25 14:31 | Install   | 9       |
| 4                                | -y install @base firewal | 2019-02-04 11:27 | Install   | 127 EE  |
| 3                                | -C -y remove firewalld - | 2019-01-16 13:12 | Removed   | 11 EE   |
| 2                                | -C -y remove linux-firmw | 2019-01-16 13:12 | Removed   | 1       |
| 1                                |                          | 2019-01-16 13:05 | Install   | 447 EE  |

- Die Option **history undo** kehrt eine Transaktion um.

```
[user@host ~]$ sudo yum history undo 5
Undoing transaction 7, from Tue 26 Feb 2019 10:40:32 AM EST
Install apr-1.6.3-8.el8.x86_64 @rhel8-appstream
Install apr-util-1.6.1-6.el8.x86_64 @rhel8-appstream
Install apr-util-bdb-1.6.1-6.el8.x86_64 @rhel8-appstream
Install apr-util-openssl-1.6.1-6.el8.x86_64 @rhel8-appstream
Install httpd-2.4.37-7.module+el8+2443+605475b7.x86_64 @rhel8-appstream
...output omitted...
```

## Übersicht der Yum-Befehle

Pakete können nach Namen oder nach Paketgruppen gesucht, installiert, aktualisiert und entfernt werden.

| Aufgabe:                                                         | Befehl:                            |
|------------------------------------------------------------------|------------------------------------|
| Auflistung aller installierten und verfügbaren Pakete nach Namen | <b>yum list [NAME-PATTERN]</b>     |
| Auflistung aller installierten und verfügbaren Gruppen           | <b>yum group list</b>              |
| Suche nach Paketen nach Schlüsselbegriffen                       | <b>yum search KEYWORD</b>          |
| Anzeige der Detailangaben zu einem Paket                         | <b>yum info PACKAGE_NAME</b>       |
| Installation eines Pakets                                        | <b>yum install PACKAGE_NAME</b>    |
| Installation einer Paketgruppe                                   | <b>yum group install GROUPNAME</b> |
| Aktualisierung aller Pakete                                      | <b>yum update</b>                  |
| Entfernung eines Pakets                                          | <b>yum remove PACKAGE_NAME</b>     |
| Anzeige des Transaktionsverlaufs                                 | <b>yum history</b>                 |



### Literaturhinweise

Manpages **yum(1)** und **yum.conf(5)**

Weitere Informationen finden Sie im Kapitel *Managing software packages* im Handbuch *Red Hat Enterprise Linux 8 Configuring basic system settings* unter [https://access.redhat.com/documentation/en-us/red\\_hat\\_enterprise\\_linux/8/html-single/configuring\\_basic\\_system\\_settings/index#managing-software-packages\\_configuring-basic-system-settings](https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux/8/html-single/configuring_basic_system_settings/index#managing-software-packages_configuring-basic-system-settings)

## ► Angeleitete Übung

# Installieren und Aktualisieren von Softwarepaketen mit YUM

In dieser Übung werden Sie Pakete und Paketgruppen installieren und entfernen.

## Ergebnisse

Sie sollten in der Lage sein, Pakete mit Abhängigkeiten zu installieren und zu entfernen.

## Bevor Sie Beginnen

Melden Sie sich als Benutzer **student** mit dem Passwort **student** bei **workstation** an.

Führen Sie auf der **workstation** den Befehl **lab software-yum start** aus. Dieser Befehl führt ein Startskript aus, das ermittelt, ob der Host **servera** im Netzwerk erreichbar ist.

```
[student@workstation ~]$ lab software-yum start
```

- 1. Melden Sie sich mit dem Befehl **ssh** bei **servera** als Benutzer **student** an. Die Systeme sind für die Verwendung von SSH-Schlüsseln zur Authentifizierung konfiguriert. Daher ist kein Passwort erforderlich, um sich auf **servera** anzumelden.

```
[student@workstation ~]$ ssh student@servera
...output omitted...
[student@servera ~]$
```

- 2. Verwenden Sie den Befehl **sudo -i**, um an der Eingabeaufforderung der Shell zum **root**-Benutzer zu wechseln.

```
[student@servera ~]$ sudo -i
Password: student
[root@servera ~]#
```

- 3. Suchen Sie nach einem bestimmten Paket.

- 3.1. Versuchen Sie, den Befehl **guile** auszuführen. Sie stellen fest, dass dieser nicht installiert ist.

```
[root@servera ~]# guile
-bash: guile: command not found
```

- 3.2. Verwenden Sie den Befehl **yum search**, um nach Paketen zu suchen, die **guile** als Teil ihres Namens oder ihrer Zusammenfassung aufweisen.

```
[root@servera ~]# yum search guile
=====
Name Exactly Matched: guile =====
guile.i686 : A GNU implementation of Scheme for application extensibility
guile.x86_64 : A GNU implementation of Scheme for application extensibility
```

- 3.3. Verwenden Sie den Befehl **yum info**, um weitere Informationen zum Paket **guile** zu erhalten.

```
[root@servera ~]# yum info guile
Available Packages
Name        : guile
Epoch       : 5
Version     : 2.0.14
Release     : 7.el8
...output omitted...
```

- 4. Verwenden Sie den Befehl **yum install**, um das Paket **guile** zu installieren.

```
[root@servera ~]# yum install guile
...output omitted...
Dependencies resolved.

=====
Package      Arch    Version          Repository           Size
=====
Installing:
 guile       x86_64  5:2.0.14-7.el8   rhel-8.2-for-x86_64-appstream-rpms 3.5 M
Installing dependencies:
 gc          x86_64  7.6.4-3.el8      rhel-8.2-for-x86_64-appstream-rpms 109 k
 libatomic_ops x86_64  7.6.2-3.el8      rhel-8.2-for-x86_64-appstream-rpms 38 k
 libtool-ltdl x86_64  2.4.6-25.el8     rhel-8.2-for-x86_64-baseos-rpms   58 k

Transaction Summary
=====
Install 4 Packages

Total download size: 3.7 M
Installed size: 12 M
Is this ok [y/N]: y
...output omitted...
Complete!
```

- 5. Entfernen Sie Pakete.

- 5.1. Verwenden Sie den Befehl **yum remove**, um das Paket **guile** zu entfernen, aber antworten Sie mit **no**, wenn Sie dazu aufgefordert werden. Wie viele Pakete würden entfernt?

```
[root@servera ~]# yum remove guile
...output omitted...
Dependencies resolved.

=====
```

```

Package      Arch    Version      Repository      Size
=====
Removing:
  guile       x86_64  5:2.0.14-7.el8   @rhel-8.2-for-x86_64-appstream-rpms  12 M
Removing unused dependencies:
  gc          x86_64  7.6.4-3.el8     @rhel-8.2-for-x86_64-appstream-rpms 221 k
  libatomic_ops x86_64  7.6.2-3.el8   @rhel-8.2-for-x86_64-appstream-rpms  75 k
  libtool-ltdl x86_64  2.4.6-25.el8  @rhel-8.2-for-x86_64-baseos-rpms     69 k

Transaction Summary
=====
Remove 4 Packages

Freed space: 12 M
Is this ok [y/N]: n
Operation aborted.

```

- 5.2. Verwenden Sie den Befehl **yum remove**, um das Paket **gc** zu entfernen, aber antworten Sie mit **no**, wenn Sie dazu aufgefordert werden. Wie viele Pakete würden entfernt?

```

[root@servera ~]# yum remove gc
...output omitted...
Dependencies resolved.
=====
Package      Arch    Version      Repository      Size
=====
Removing:
  gc          x86_64  7.6.4-3.el8     @rhel-8.2-for-x86_64-appstream-rpms 221 k
Removing dependent packages:
  guile       x86_64  5:2.0.14-7.el8   @rhel-8.2-for-x86_64-appstream-rpms  12 M
Removing unused dependencies:
  libatomic_ops x86_64  7.6.2-3.el8   @rhel-8.2-for-x86_64-appstream-rpms  75 k
  libtool-ltdl x86_64  2.4.6-25.el8  @rhel-8.2-for-x86_64-baseos-rpms     69 k

Transaction Summary
=====
Remove 4 Packages

Freed space: 12 M
Is this ok [y/N]: n
Operation aborted.

```

- 6. Sammeln Sie Informationen zur Komponentengruppe „Security Tools“ und installieren Sie sie auf **servera**.

- 6.1. Verwenden Sie den Befehl **yum group list**, um alle verfügbaren Komponentengruppen aufzulisten.

```
[root@servera ~]# yum group list
```

**Kapitel 14** | Installieren und Aktualisieren von Softwarepaketen

- 6.2. Verwenden Sie den Befehl **yum group info**, um weitere Informationen zur Komponentengruppe **Security Tools** zu erhalten, einschließlich einer Liste der darin enthaltenen Pakete.

```
[root@servera ~]# yum group info "Security Tools"
Group: Security Tools
Description: Security tools for integrity and trust verification.
Default Packages:
  scap-security-guide
Optional Packages:
  aide
  hmaccalc
  openscap
  openscap-engine-sce
  openscap-utils
  scap-security-guide-doc
  scap-workbench
  tpm-quote-tools
  tpm-tools
  tpm2-tools
  trousers
  udica
```

- 6.3. Verwenden Sie den Befehl **yum group install**, um die Komponentengruppe **Security Tools** zu installieren.

```
[root@servera ~]# yum group install "Security Tools"
Dependencies resolved.
=====
 Package           Arch   Version        Repository      Size
 =====
Installing group/module packages:
  scap-security-guide noarch 0.1.48-7.el8 rhel-8-for-x86_64-appstream-rpms 6.9 M
Installing dependencies:
  GConf2            x86_64 3.2.6-22.el8 rhel-8-for-x86_64-appstream-rpms 1.0 M
...output omitted...

Transaction Summary
=====
Install  6 Packages

Total download size: 12 M
Installed size: 247 M
Is this ok [y/N]: y
...output omitted...
Installed:
  GConf2-3.2.6-22.el8.x86_64          libxslt-1.1.32-4.el8.x86_64
  openscap-1.3.2-6.el8.x86_64         openscap-scanner-1.3.2-6.el8.x86_64
  scap-security-guide-0.1.48-7.el8.noarch  xml-common-0.6.3-50.el8.noarch

Complete!
```

- 7. Machen Sie sich mit dem Verlauf und den Rückgängig-Optionen von **yum** vertraut.

- 7.1. Verwenden Sie den Befehl **yum history**, um den letzten **yum**-Verlauf anzuzeigen.

```
[root@servera ~]# yum history
ID      | Command line           | Date and time   | Action(s)    | Altered
-----
6 | group install Security T | 2019-02-26 17:11 | Install      | 7
5 | install guile          | 2019-05-26 17:05 | Install      | 4
4 | -y install @base firewal | 2019-02-04 11:27 | Install      | 127 EE
3 | -C -y remove firewalld - | 2019-01-16 13:12 | Removed      | 11 EE
2 | -C -y remove linux-firmw | 2019-01-16 13:12 | Removed      | 1
1 |                           | 2019-01-16 13:05 | Install      | 447 EE
```

Auf Ihrem System ist der Verlauf wahrscheinlich anders.

- 7.2. Verwenden Sie den Befehl **yum history info**, um zu bestätigen, dass es sich bei der letzten Transaktion um die Gruppeninstallation handelt. Ersetzen Sie im folgenden Befehl die Transaktions-ID durch die ID aus dem vorherigen Schritt.

```
[root@servera ~]# yum history info 6
Transaction ID : 6
Begin time      : Tue 26 Feb 2019 05:11:25 PM EST
Begin rpmdb     : 563:bf48c46156982a78e290795400482694072f5ebb
End time        : Tue 26 Feb 2019 05:11:33 PM EST (8 seconds)
End rpmdb       : 623:bf25b424ccf451dd0a6e674fb48e497e66636203
User            : Student User <student>
Return-Code     : Success
Releasever     : 8
Command Line   : group install Security Tools
Packages Altered:
  Install libxslt-1.1.32-4.el8.x86_64      @rhel-8.2-for-x86_64-baseos-rpms
  Install xml-common-0.6.3-50.el8.noarch    @rhel-8.2-for-x86_64-baseos-rpms
...output omitted...
```

- 7.3. Verwenden Sie den Befehl **yum history undo**, um die Pakete zu entfernen, die installiert wurden, als das Paket **guile** installiert wurde. Suchen Sie auf Ihrem System die korrekte Transaktions-ID in der Ausgabe des **yum history**-Befehls und verwenden Sie dann diese ID im folgenden Befehl.

```
[root@servera ~]# yum history undo 5
```

- 8. Melden Sie sich beim System **servera** ab.

```
[root@servera ~]# exit
logout
[student@servera ~]$ exit
Connection to servera closed.
[student@workstation ~]$
```

## Beenden

Führen Sie auf **workstation** das Skript **lab software-yum finish** aus, um diese Übung abzuschließen.

```
[student@workstation ~]$ lab software-yum finish
```

Hiermit ist die angeleitete Übung beendet.

# Aktivieren von YUM-Software-Repositories

---

## Ziele

Nach Abschluss dieses Abschnitts sollten Sie in der Lage sein, die Verwendung von YUM-Repositories von Red Hat (oder Drittanbietern) über einen Server zu aktivieren und zu deaktivieren.

## Aktivieren von Red Hat-Software-Reposotories

Beim Registrieren eines Systems beim Subskriptionsverwaltungs-Service wird der Zugriff auf Software-Repositories automatisch auf Grundlage der zugeordneten Subskriptionen konfiguriert. So zeigen Sie alle verfügbaren Repositories an:

```
[user@host ~]$ yum repolist all
...output omitted...
rhel-8-for-x86_64-appstream-debug-rpms    ... AppStream (Debug RPMs)  disabled
rhel-8-for-x86_64-appstream-rpms          ... AppStream (RPMs)        enabled:
5,045
rhel-8-for-x86_64-appstream-source-rpms   ... AppStream (Source RPMs) disabled
rhel-8-for-x86_64-baseos-debug-rpms       ... BaseOS (Debug RPMs)   enabled:
2,270
rhel-8-for-x86_64-baseos-rpms            ... BaseOS (RPMs)         enabled:
1,963
...output omitted...
```

Mit dem Befehl **yum config-manager** können Sie Repositories aktivieren oder deaktivieren. Um ein Repository zu aktivieren, legt der Befehl den Parameter **enabled** auf **1** fest. Der folgende Befehl aktiviert z. B. das Repository **rhel-8-server-debug-rpms**:

```
[user@host ~]$ yum config-manager --enable rhel-8-server-debug-rpms
Updating Subscription Management repositories.
=====
repo: rhel-8-for-x86_64-baseos-debug-rpms
[relabel-8-for-x86_64-baseos-debug-rpms]
bandwidth = 0
baseurl = [https://cdn.redhat.com/content/dist/rhel8/8/x86_64/baseos/debug]
cachedir = /var/cache/dnf
cost = 1000
deltarpm = 1
deltarpm_percentage = 75
enabled = 1
...output omitted...
```

Nicht von Red Hat stammende Quellen stellen Software über Repositories von Drittanbietern bereit, auf die mit dem Befehl **yum** über eine Website, einen FTP-Server oder das lokale Dateisystem zugegriffen werden kann. Adobe stellt beispielsweise einen Teil seiner Software für Linux über ein YUM-Repository bereit. In einem Red Hat-Kursraum hostet der Kursraumserver `content.example.com` YUM-Repositories.

## Kapitel 14 | Installieren und Aktualisieren von Softwarepaketen

Erstellen Sie zum Aktivieren der Unterstützung für neue Repositorys von Drittanbieter eine Datei im Verzeichnis **/etc/yum.repos.d/**. Repository-Konfigurationsdateien müssen mit der **.repo**-Erweiterung enden. Die Repository-Definition enthält die URL des Repositorys, einen Namen, ob die Paketsignaturen mit GPG geprüft werden sollen und wenn ja, die URL zu dem vertrauenswürdigen GPG-Schlüssel.

## Erstellen von YUM-Repositories

Erstellen Sie YUM-Repositorys mit dem Befehl **yum config-manager**. Der folgende Befehl erstellt eine Datei mit dem Namen **/etc/yum.repos.d/d1.fedoraproject.org\_pub\_epel\_8\_Everything\_x86\_64.repo** mit der angezeigten Ausgabe.

```
[user@host ~]$ yum config-manager \
--add-repo="https://dl.fedoraproject.org/pub/epel/8/Everything/x86_64/"
Adding repo from: https://dl.fedoraproject.org/pub/epel/8/Everything/x86_64/

[d1.fedoraproject.org_pub_epel_8_Everything_x86_64_]
name=created by yum config-manager from https://dl.fedoraproject.org/pub/epel/8/
Everything/x86_64/
baseurl=https://dl.fedoraproject.org/pub/epel/8/Everything/x86_64/
enabled=1
```

Ändern Sie diese Datei, um benutzerdefinierte Werte und den Speicherort eines GPG-Schlüssels anzugeben. Schlüssel werden an verschiedenen Speicherorten am Remote-Repository-Standort gespeichert, z. B. <http://dl.fedoraproject.org/pub/epel/RPM-GPG-KEY-EPEL-8>. Administratoren sollten den Schlüssel vorzugsweise in eine lokale Datei herunterladen, anstatt **yum** das Abrufen des Schlüssels aus einer externen Quelle zu gestatten. Beispiel:

```
[EPEL]
name=EPEL 8
baseurl=https://dl.fedoraproject.org/pub/epel/8/Everything/x86_64/
enabled=1
gpgcheck=1
gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-EPEL-8
```

## RPM-Konfigurationspakete für lokale Repositorys

Manche Repositorys enthalten eine Konfigurationsdatei und den öffentlichen GPG-Schlüssel als Bestandteil eines RPM-Pakets, das mithilfe des Befehls **yum localinstall** heruntergeladen und installiert werden kann. Ein Beispiel hierfür ist das Freiwilligenprojekt EPEL (Extra Packages for Enterprise Linux), das nicht von Red Hat unterstützt, aber mit Red Hat Enterprise Linux kompatible Software bereitstellt.

Der folgende Befehl installiert das Red Hat Enterprise Linux 8 EPEL-Repository-Paket:

```
[user@host ~]$ rpm --import \
http://dl.fedoraproject.org/pub/epel/RPM-GPG-KEY-EPEL-8
[user@host ~]$ yum install \
https://dl.fedoraproject.org/pub/epel/epel-release-latest-8.noarch.rpm
```

Konfigurationsdateien enthalten oft mehrere Repository-Verweise in einer einzigen Datei. Jeder Repository-Verweis beginnt mit einem Namen (ein Wort) in eckigen Klammern.

```
[user@host ~]$ cat /etc/yum.repos.d/epel.repo
[epel]
name=Extra Packages for Enterprise Linux $releasever - $basearch
#baseurl=https://download.fedoraproject.org/pub/epel/$releasever/Everything/
$basearch
metalink=https://mirrors.fedoraproject.org/metalink?repo=epel-$releasever&arch=
$basearch&infra=$infra&content=$contentdir
enabled=1
gpgcheck=1
gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-EPEL-8

[epel-debuginfo]
name=Extra Packages for Enterprise Linux $releasever - $basearch - Debug
#baseurl=https://download.fedoraproject.org/pub/epel/$releasever/Everything/
$basearch/debug
metalink=https://mirrors.fedoraproject.org/metalink?repo=epel-debug-
$releasever&arch=$basearch&infra=$infra&content=$contentdir
enabled=0
gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-EPEL-8
gpgcheck=1

[epel-source]
name=Extra Packages for Enterprise Linux $releasever - $basearch - Source
#baseurl=https://download.fedoraproject.org/pub/epel/$releasever/Everything/SRPMS
metalink=https://mirrors.fedoraproject.org/metalink?repo=epel-source-
$releasever&arch=$basearch&infra=$infra&content=$contentdir
enabled=0
gpgkey=file:///etc/pki/rpm-gpg/RPM-GPG-KEY-EPEL-8
gpgcheck=1
```

Um ein Repository zu definieren, es aber nicht standardmäßig zu durchsuchen, fügen Sie den Parameter **enabled=0** ein. Repositorys können mit dem Befehl **yum config-manager** dauerhaft oder mit den Befehlsoptionen **--enablerepo=PATTERN** und **--disablerepo=PATTERN** des Befehls **yum** vorübergehend aktiviert und deaktiviert werden.



### Warnung

Installieren Sie vor der Installation signierter Pakete den RPM-GPG-Schlüssel. Dadurch wird überprüft, ob die Pakete zu einem importierten Schlüssel gehören. Ansonsten tritt bei dem Befehl **yum** aufgrund eines fehlenden Schlüssels ein Fehler auf. Mit der Option **--nogpgcheck** kann das Fehlen von GPG-Schlüsseln ignoriert werden. Dies könnte allerdings die Installation gefälschter oder unsicherer Pakete auf dem System ermöglichen und so eventuell seine Sicherheit beeinträchtigen.



### Literaturhinweise

Manpages **yum(1)**, **yum.conf(5)** und **yum-config-manager(1)**

Weitere Informationen finden Sie im Kapitel *Managing software repositories* im Handbuch *Red Hat Enterprise Linux 8 Configuring basic system settings* unter [https://access.redhat.com/documentation/en-us/red\\_hat\\_enterprise\\_linux/8/html-single/configuring\\_basic\\_system\\_settings/index#managing-software-repositories\\_managing-software-packages](https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux/8/html-single/configuring_basic_system_settings/index#managing-software-repositories_managing-software-packages)

## ► Angeleitete Übung

# Aktivieren von YUM-Software-Repositorys

In dieser Übung konfigurieren Sie Ihren Server so, dass er Pakete aus einem Remote-YUM-Repository abruft. Dann aktualisieren oder installieren Sie ein Paket aus diesem Repository.

## Ergebnisse

Sie sollten in der Lage sein, ein System so zu konfigurieren, dass es Software-Updates von einem Kursraumserver erhält und das System für die Verwendung der neuesten Pakete aktualisiert.

## Bevor Sie Beginnen

Melden Sie sich als Benutzer **student** mit dem Passwort **student** bei **workstation** an.

Führen Sie auf der **workstation** den Befehl **lab software-repo start** aus. Dieser Befehl führt ein Startskript aus, das ermittelt, ob der Host **servera** im Netzwerk erreichbar ist. Das Skript stellt zudem sicher, dass das Paket **yum** installiert ist.

```
[student@workstation ~]$ lab software-repo start
```

- 1. Melden Sie sich mit dem Befehl **ssh** bei **servera** als Benutzer **student** an.

```
[student@workstation ~]$ ssh student@servera
...output omitted...
[student@servera ~]$
```

- 2. Verwenden Sie den Befehl **sudo -i**, um an der Eingabeaufforderung der Shell zu **root** zu wechseln.

```
[student@servera ~]$ sudo -i
[sudo] password for student: student
[root@servera ~]#
```

- 3. Konfigurieren Sie Software-Repositories auf **servera** so, dass sie benutzerdefinierte Pakete und Updates über die folgende URL erhalten:

- Unter [http://content.example.com/rhel8.2/x86\\_64/rhcsa-practice/rht](http://content.example.com/rhel8.2/x86_64/rhcsa-practice/rht) bereitgestellte benutzerdefinierte Pakete
  - Unter [http://content.example.com/rhel8.2/x86\\_64/rhcsa-practice/errata](http://content.example.com/rhel8.2/x86_64/rhcsa-practice/errata) bereitgestellte Updates zu den benutzerdefinierten Paketen
- 3.1. Verwenden Sie **yum config-manager**, um das Repository mit den benutzerdefinierten Paketen hinzuzufügen.

```
[root@servera ~]# yum config-manager \
--add-repo "http://content.example.com/rhel8.2/x86_64/rhcsa-practice/rht"
Adding repo from: http://content.example.com/rhel8.2/x86_64/rhcsa-practice/rht
```

- 3.2. Überprüfen Sie die Software-Repository-Datei, die mit dem vorherigen Befehl im Verzeichnis **/etc/yum.repos.d** erstellt wurde. Verwenden Sie den Befehl **vim**, um die Datei zu bearbeiten und den Parameter *gpgcheck=0* hinzuzufügen, damit die GPG-Schlüsselprüfung für das Repository deaktiviert wird.

```
[root@servera ~]# vim \
/etc/yum.repos.d/content.example.com_rhel8.2_x86_64_rhcsa-practice_rht.repo
[content.example.com_rhel8.2_x86_64_rhcsa-practice_rht]
name=created by dnf config-manager from http://content.example.com/rhel8.2/x86_64/
rhcsa-practice/rht
baseurl=http://content.example.com/rhel8.2/x86_64/rhcsa-practice/rht
enabled=1
gpgcheck=0
```

- 3.3. Erstellen Sie die Datei **/etc/yum.repos.d/errata.repo**, um das Updates-Repository mit dem folgenden Inhalt zu aktivieren:

```
[rht-updates]
name=rht updates
baseurl=http://content.example.com/rhel8.2/x86_64/rhcsa-practice/errata
enabled=1
gpgcheck=0
```

- 3.4. Verwenden Sie den Befehl **yum repolist all**, um alle Repositorys auf dem System aufzulisten:

```
[root@servera ~]# yum repolist all
repo id                                repo name      status
content.example.com_rhel8.2_x86_64_rhcsa-practice_rht created by .... enabled
rht-updates                               rht updates   enabled
...output omitted...
```

- 4. Deaktivieren Sie das Software-Repository **rht-updates** und installieren Sie das Paket **rht-system**.

- 4.1. Deaktivieren Sie mit **yum config-manager --disable** das Repository **rht-updates**.

```
[root@servera ~]# yum config-manager --disable rht-updates
```

- 4.2. Führen Sie das Paket **rht-system** auf, und installieren Sie es dann:

```
[root@servera ~]# yum list rht-system
Available Packages
rht-system.noarch 1.0.0-1 content.example.com_rhel8.2_x86_64_rhcsa-practice_rht
[root@servera ~]# yum install rht-system
```

```
Dependencies resolved.
=====
 Package           Arch    Version      Repository      Size
=====
Installing:
 rht-system       noarch  1.0.0-1     content..._rht   3.7 k
 ...output omitted...
Is this ok [y/N]: y
...output omitted...
Installed:
 rht-system-1.0.0-1.noarch
Complete!
```

- 4.3. Überprüfen Sie, ob das Paket *rht-system* installiert ist, und notieren Sie sich die Versionsnummer des Pakets.

```
[root@servera ~]# yum list rht-system
Installed Packages
rht-system.noarch 1.0.0-1 @content.example.com_rhel8.2_x86_64_rhcsa-practice_rht
```

- 5. Aktivieren Sie das Software-Repository **rht-updates** und aktualisieren Sie alle relevanten Softwarepakete.

- 5.1. Verwenden Sie **yum config-manager --enable**, um das Repository **rht-updates** zu aktivieren.

```
[root@servera ~]# yum config-manager --enable rht-updates
```

- 5.2. Verwenden Sie den Befehl **yum update** zum Aktualisieren aller Softwarepakete auf **servera**.

```
[root@servera ~]# yum update
Dependencies resolved.
=====
 Package           Arch    Version      Repository      Size
=====
Upgrading:
 rht-system       x86_64  1.0.0-2.el7  rht-updates   3.9 k
...output omitted...
Is this ok [y/N]: y
...output omitted...
Complete!
```

- 5.3. Überprüfen Sie, ob ein Upgrade des Pakets *rht-system* durchgeführt wurde, und notieren Sie sich die Versionsnummer des Pakets.

```
[root@servera ~]# yum list rht-system
Installed Packages
rht-system.noarch 1.0.0-2.el7          @rht-updates
```

- 6. Beenden Sie **servera**.

```
[root@servera ~]# exit  
logout  
[student@servera ~]$ exit  
logout  
Connection to servera closed.  
[student@workstation ~]$
```

## Beenden

Führen Sie auf **workstation** das Skript **lab software-repo finish** aus, um diese Übung abzuschließen. Dieses Skript entfernt alle während der Übung auf **servera** installierten Software-Repositorys und Pakete.

```
[student@workstation ~]$ lab software-repo finish
```

Hiermit ist die angeleitete Übung beendet.

# Verwalten von Paketmodul-Streams

---

## Ziele

Am Ende dieses Abschnitts sollten Sie zu Folgendem in der Lage sein:

- Erläuterung, wie Module die Installation bestimmter Softwareversionen ermöglichen.
- Vorgehensweise beim Auflisten, Aktivieren und Wechseln von Modul-Streams.
- Installieren und Aktualisieren von Paketen eines Moduls.

## Einführung in Application Stream

Red Hat Enterprise Linux 8.0 führt das Konzept „Application Stream“ ein. Mehrere Versionen der mit der Distribution gelieferten Benutzerbereichskomponenten werden jetzt gleichzeitig bereitgestellt. Sie werden möglicherweise häufiger aktualisiert als die zentralen Betriebssystempakete. Dies bietet Ihnen mehr Flexibilität beim Anpassen von Red Hat Enterprise Linux ohne Beeinträchtigung der zugrunde liegenden Stabilität der Plattform oder bestimmter Bereitstellungen.

Die Verwaltung alternativer Versionen des Softwarepaketes einer Anwendung und der zugehörigen Pakete bedeutete traditionell, dass für die einzelnen Versionen unterschiedliche Repositorys verwaltet wurden. Dies führte für Entwickler, die die neueste Version einer Anwendung verwenden wollten, und für Administratoren, die die stabilste Version der Anwendung verwenden wollten, zu einer Situation, die nur schwer zu verwalten war. Dieser Vorgang wird in Red Hat Enterprise Linux 8 mithilfe einer neuen Technologie namens *Modularität* vereinfacht. Durch die Modularität kann ein einzelnes Repository mehrere Versionen des Anwendungspakets und seiner Abhängigkeiten hosten.

Red Hat Enterprise Linux 8-Inhalte werden über zwei zentrale Software-Repositorys verteilt: *BaseOS* und *Application Stream (AppStream)*.

## BaseOS

Das BaseOS-Repository stellt den zentralen Inhalt des Betriebssystems für Red Hat Enterprise Linux als RPM-Pakete bereit. BaseOS-Komponenten verfügen über einen Lebenszyklus, der mit Inhalten vorheriger Red Hat Enterprise Linux-Versionen identisch ist.

## Anwendungsstream

Das Application Stream-Repository stellt Inhalte mit unterschiedlichen Lebenszyklen sowohl als Module sowie auch als herkömmliche Pakete bereit. Das Application Stream-Repository enthält erforderliche Teile des Systems sowie eine Vielzahl von Anwendungen, die zuvor als Teil der Red Hat Software Collections und anderer Produkte und Programme verfügbar waren.



### Wichtig

Sowohl BaseOS als auch AppStream sind erforderliche Komponenten eines Red Hat Enterprise Linux 8-Systems.

## Kapitel 14 | Installieren und Aktualisieren von Softwarepaketen

Das Application Stream-Repository enthält zwei Arten von Inhalten: *Module* und traditionelle RPM-Pakete. Ein Modul beschreibt einen Satz von zusammengehörenden RPM-Paketen. Module können mehrere Streams enthalten, um mehrere Versionen von Anwendungen für die Installation verfügbar zu machen. Durch Aktivieren eines Modul-Streams erhält das System Zugriff auf die RPM-Pakete innerhalb dieses Modul-Streams.

## Module

Ein Modul ist ein Satz von RPM-Paketen, die einen konsistenten, zusammengehörigen Satz bilden. In der Regel handelt es sich dabei um eine bestimmte Version einer Softwareanwendung oder Programmiersprache. Ein typisches Modul kann Pakete mit einer Anwendung, Pakete mit den spezifischen Abhängigkeitsbibliotheken der Anwendung, Pakete mit Anwendungsdokumentation und Pakete mit Hilfsprogrammen enthalten.

## Modul-Streams

Jedes Modul kann einen oder mehrere Modul-Streams aufweisen, die verschiedene Versionen des Inhalts enthalten. Jeder der Streams empfängt unabhängig voneinander entsprechende Updates. Stellen Sie sich den Modul-Stream als virtuelles Repository im physischen Application Stream-Repository vor.

Für jedes Modul kann nur einer seiner Streams aktiviert werden und seine Pakete bereitstellen.

## Modulprofile

Jedes Modul kann über ein oder mehrere Profile verfügen. Ein Profil entspricht einer Liste bestimmter Pakete, die zusammen für einen bestimmten Anwendungsfall installiert werden, z. B. für Server, Clients, Entwicklung, minimale Installation oder andere.

Durch das Installieren eines bestimmten Modulprofils wird lediglich eine bestimmte Gruppe von Paketen aus dem Modul-Stream installiert. Sie können Pakete anschließend wie gewohnt installieren oder deinstallieren. Wenn Sie kein Profil angeben, installiert das Modul sein *Standardprofil*.

## Verwalten von Modulen mit Yum

Die neu in Red Hat Enterprise Linux 8 enthaltene Yum Version 4 unterstützt die neuen modularen Funktionen von Application Stream.

Zur Verarbeitung des modularen Inhalts wurde der Befehl **yum module** hinzugefügt. Ansonsten arbeitet **yum** mit Modulen ähnlich wie mit regulären Paketen.

## Auflisten von Modulen

Verwenden Sie **yum module list**, um eine Liste der verfügbaren Module anzuzeigen:

```
[user@host ~]$ yum module list
Red Hat Enterprise Linux 8 for x86_64 - AppStream (RPMs)
Name           Stream      Profiles   Summary
389-ds         1.4        default    389 Directory Server (base)
ant            1.10 [d]    common    [d] Java build tool
container-tools 1.0 [d]    common    [d] Common tools and dependencies for
                                container runtimes
```

```
...output omitted...
Hint: [d]efault, [e]nabled, [x]disabled, [i]nstalled
```



### Anmerkung

Verwenden Sie den *Hint* (Hinweis) am Ende der Ausgabe zur Ermittlung, welche Streams und Profile aktiviert, deaktiviert, installiert und als Standard festgelegt sind.

So listen Sie die Modul-Streams für ein bestimmtes Modul auf und rufen deren Status ab:

```
[user@host ~]$ yum module list perl
Red Hat Enterprise Linux 8 for x86_64 - AppStream (RPMS)
Name Stream Profiles Summary
perl 5.24 common [d], minimal Practical Extraction and Report Language
perl 5.26 [d] common [d], minimal Practical Extraction and Report Language
```

So zeigen Sie Details eines Moduls an:

```
[user@host ~]$ yum module info perl
Name : perl
Stream : 5.24
Version : 820190207164249
Context : ee766497
Profiles : common [d], minimal
Default profiles : common
Repo : rhel-8-for-x86_64-appstream-rpms
Summary : Practical Extraction and Report Language
...output omitted...
Artifacts : perl-4:5.24.4-403.module+el8+2770+c759b41a.x86_64
            : perl-Algorithm-Diff-0:1.1903-9.module+el8+2464+d274aed1.noarch
            : perl-Archive-Tar-0:2.30-1.module+el8+2464+d274aed1.noarch
...output omitted...
```



### Anmerkung

Ohne Angabe eines Modul-Streams zeigt **yum module info** die Liste der Pakete an, die vom Standardprofil eines Moduls mit dem Standard-Stream installiert wurden. Verwenden Sie das Format *modulename:stream*, um einen bestimmten Modul-Stream anzuzeigen. Fügen Sie die Option **--profile** hinzu, um Informationen zu installierten Paketen für jedes der Profile des Moduls anzuzeigen. Beispiel:

```
[user@host ~]$ yum module info --profile perl:5.24
```

## Aktivieren von Modul-Streams und Installieren von Modulen

Modul-Streams müssen aktiviert sein, um deren Modul installieren zu können. Um diesen Vorgang beim Installieren eines Moduls zu vereinfachen, aktiviert das Modul bei Bedarf seinen Modul-

Stream-Modul-Streams können manuell mit **yum module enable** und durch die Angabe des Namens des Modul-Streams aktiviert werden.

**Wichtig**

Für ein bestimmtes Modul kann nur ein Modul-Stream aktiviert werden. Durch Aktivieren eines zusätzlichen Modul-Streams wird der ursprüngliche Modul-Stream deaktiviert.

Installieren Sie ein Modul mit standardmäßigem Stream und Standardprofilen:

```
[user@host ~]$ sudo yum module install perl
Dependencies resolved.
=====
Package           Arch    Version      Repository          Size
=====
Installing group/module packages:
perl              x86_64  4:5.26.3-416.el8
   rhel-8-for-x86_64-appstream-htb-rpms 72 k
Installing dependencies:
...output omitted...
Running transaction
  Preparing          :   1/1
  Installing        : perl-Exporter-5.72-396.el8.noarch             1/155
  Installing        : perl-Carp-1.42-396.el8.noarch                2/155
...output omitted...
Installed:
  perl-4:5.26.3-416.el8.x86_64
  perl-Encode-Locale-1.05-9.el8.noarch
...output omitted...
Complete!
```

**Anmerkung**

Dieselben Ergebnisse können durch Ausführen von **yum install @perl** erzielt werden. Die @-Notation informiert **yum** darüber, dass es sich bei dem Argument um einen Modulnamen anstelle eines Paketnamens handelt.

So überprüfen Sie den Status des Modul-Streams und das installierte Profil:

```
[user@host ~]$ yum module list perl
Red Hat Enterprise Linux 8 for x86_64 - AppStream (RPMs)
Name   Stream      Profiles          Summary
perl   5.24        common, minimal  Practical Extraction and Report Language
perl   5.26 [d][e]  common [i], minimal  Practical Extraction and Report Language

Hint: [d]efault, [e]nabled, [x]disabled, [i]nstalled
```

## Entfernen von Modulen und Deaktivieren von Modul-Streams

Beim Entfernen eines Moduls werden alle Pakete entfernt, die von Profilen des aktuell aktivierten Modul-Streams installiert wurden, sowie alle weiteren von diesen Paketen abhängigen Pakete und

Module. Von diesem Modul-Stream installierte Pakete, die in keinem seiner Profile aufgeführt sind, bleiben auf dem System installiert und können manuell entfernt werden.



### Warnung

Das Entfernen von Modulen und das Wechseln von Modul-Streams können etwas schwierig sein. Das Umschalten des für ein Modul aktivierten Streams entspricht dem Zurücksetzen des aktuellen Streams und dem Aktivieren des neuen Streams. Installierte Pakete werden nicht automatisch geändert. Sie müssen das manuell ausführen.

Es wird nicht empfohlen, einen anderen als den derzeit installierten Modul-Stream direkt zu installieren, da während der Installation möglicherweise Upgrade-Skripts ausgeführt werden, die Probleme mit dem ursprünglichen Modul-Stream verursachen. Dies kann zu Datenverlust oder anderen Konfigurationsproblemen führen.

Gehen Sie mit Bedacht vor.

So entfernen Sie ein installiertes Modul:

```
[user@host ~]$ sudo yum module remove perl
Dependencies resolved.
=====
Package           ArchVersion      Repository
Size
=====
Removing:
perl              x86_64:5.26.3-416.el8    @rhel-8.0-for-x86_64-
appstream-rpms 0
Removing unused dependencies:
...output omitted...
Running transaction
Preparing          :                               1/1
Erasing            : perl-4:5.26.3-416.el8.x86_64          1/155
Erasing            : perl-CPAN-2.18-397.el8.noarch        2/155
...output omitted...
Removed:
perl-4:5.26.3-416.el8.x86_64
dwz-0.12-9.el8.x86_64
...output omitted...
Complete!
```

Nachdem das Modul entfernt wurde, ist der Modul-Stream weiterhin aktiviert. So überprüfen Sie, ob der Modul-Stream noch aktiviert ist:

```
[user@host ~]$ yum module list perl
Red Hat Enterprise Linux 8 for x86_64 - AppStream (RPMs)
Name Stream Profiles Summary
perl 5.24 common [d], minimal Practical Extraction and Report Language
perl 5.26 [d][e] common [d], minimal Practical Extraction and Report Language

Hint: [d]efault, [e]nabled, [x]disabled, [i]nstalled
```

So deaktivieren Sie den Modul-Stream:

```
[user@host ~]$ sudo yum module disable perl
...output omitted...
Dependencies resolved.

=====
 Package          Arch      Version       Repository      Size
=====
Disabling module streams:
 perl            5.26
Is this ok [y/N]: y
Complete!
```

## Wechseln der Modul-Streams

Das Wechseln von Modul-Streams erfordert im Allgemeinen ein Upgrade oder Downgrade des Inhalts auf eine andere Version.

Um einen sauberen Wechsel zu gewährleisten, sollten Sie zuerst die vom Modul-Stream bereitgestellten Module entfernen. Dadurch werden alle Pakete entfernt, die von den Profilen des Moduls installiert wurden, sowie alle Module und Pakete, von denen diese Pakete abhängig sind.

Um die vom Modul installierten Pakete aufzulisten, ist im folgenden Beispiel das Modul `postgresql:9.6` installiert:

```
[user@host ~]$ sudo yum module info postgresql | grep module+el8 | \
sed 's/.*/ //g;s/\n/ /g' | xargs yum list installed
Installed Packages
postgresql.x86_64           9.6.10-1.module+el8+2470+d1bafa0e   @rhel-8.0-for-
x86_64-appstream-rpms
postgresql-server.x86_64      9.6.10-1.module+el8+2470+d1bafa0e   @rhel-8.0-for-
x86_64-appstream-rpms
```

Entfernen Sie die im vorherigen Befehl aufgelisteten Pakete. Markieren Sie die zu deinstallierenden Modulprofile.

```
[user@host ~]$ sudo yum module remove postgresql
...output omitted...
Is this ok [y/N]: y
...output omitted...
Removed:
  postgresql-server-9.6.10-1.module+el8+2470+d1bafa0e.x86_64
  libpq-10.5-1.el8.x86_64  postgresql-9.6.10-1.module+el8+2470+d1bafa0e.x86_64
Complete
```

Setzen Sie nach dem Entfernen der Modulprofile den Modul-Stream zurück. Verwenden Sie den Befehl **yum module reset**, um den Modul-Stream zurückzusetzen.

```
[user@host ~]$ sudo yum module reset postgresql
=====
Package      Arch       Version   Repository   Size
=====
Resetting module streams:
postgresql          9.6

Transaction Summary
=====

Is this ok [y/N]: y
Complete!
```

So aktivieren Sie einen anderen Modul-Stream und installieren das Modul:

```
[user@host ~]$ sudo yum module install postgresql:10
```

Der neue Modul-Stream wird aktiviert und der aktuelle Stream deaktiviert. Möglicherweise muss für Pakete des vorherigen Modul-Streams ein Upgrade oder Downgrade ausgeführt werden, die nicht im neuen Profil aufgelistet werden. Führen Sie diese Aufgabe bei Bedarf mit **yum distro-sync** aus. Es können auch Pakete des vorherigen Modul-Streams vorhanden sein, die installiert bleiben. Entfernen Sie diese mit **yum remove**.



### Literaturhinweise

Weitere Informationen finden Sie im Kapitel *Using AppStream* im Handbuch *Red Hat Enterprise Linux 8 Installing, managing, and removing user space components* unter [https://access.redhat.com/documentation/en-us/red\\_hat\\_enterprise\\_linux/8/html-single/installing\\_managing\\_and\\_removing\\_user-space\\_components/index#using-appstream\\_using-appstream](https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux/8/html-single/installing_managing_and_removing_user-space_components/index#using-appstream_using-appstream)

### Modularität

<https://docs.fedoraproject.org/en-US/modularity/>

## ► Angeleitete Übung

# Verwalten von Paketmodul-Streams

In dieser Übung listen Sie die verfügbaren Module auf, aktivieren einen bestimmten Modul-Stream und installieren Pakete aus diesem Stream.

## Ergebnisse

Es werden folgende Fähigkeiten vermittelt:

- Auflisten der installierten Module und Überprüfen der Informationen eines Moduls.
- Aktivieren und Installieren eines Moduls über einen Stream.
- Wechseln zu einem bestimmten Modul-Stream.
- Entfernen und Deaktivieren eines Moduls.

## Bevor Sie Beginnen

Melden Sie sich als Benutzer **student** mit dem Passwort **student** bei **workstation** an.

Führen Sie auf der **workstation** den Befehl **lab software-module start** aus. Dieser Befehl führt ein Startskript aus, das ermittelt, ob der Host **servera** im Netzwerk erreichbar ist. Das Skript stellt außerdem sicher, dass die erforderlichen Software-Repositories verfügbar sind. Es installiert außerdem das Modul *postgresql:9.6*.

```
[student@workstation ~]$ lab software-module start
```

- 1. Melden Sie sich mit dem Befehl **ssh** bei **servera** als Benutzer **student** an.

```
[student@workstation ~]$ ssh student@servera
...output omitted...
[student@servera ~]$
```

- 2. Wechseln Sie über die Eingabeaufforderung der Shell zu **root**.

```
[student@servera ~]$ sudo -i
[sudo] password for student: student
[root@servera ~]#
```

- 3. Listen Sie die verfügbaren Module, Streams und installierten Module auf. Überprüfen Sie die Informationen für das Modul *python36*.

- 3.1. Verwenden Sie den Befehl **yum module list**, um die verfügbaren Module und Streams aufzulisten.

```
[root@servera ~]# yum module list
Red Hat Enterprise Linux 8.2 AppStream (dvd)
Name      Stream     Profiles          Summary
...output omitted...
python27   2.7 [d]    common [d]       Python ... version 2.7
python36   3.6 [d][e]  build, common [d] Python ... version 3.6
python38   3.8 [d]    build, common [d] Python ... version 3.8
...output omitted...
Hint: [d]efault, [e]nabled, [x]disabled, [i]nstalled
```

- 3.2. Verwenden Sie den Befehl **yum module list --installed**, um die installierten Module und Streams aufzulisten.

```
[root@servera ~]# yum module list --installed
Red Hat Enterprise Linux 8.2 AppStream (dvd)
Name      Stream     Profiles          Summary
postgresql 9.6 [e]  client, server [d] [i] PostgreSQL server and client ...

Hint: [d]efault, [e]nabled, [x]disabled, [i]nstalled
```

- 3.3. Verwenden Sie den Befehl **yum module info**, um die Details des Moduls *python36* zu überprüfen.

```
[root@servera ~]# yum module info python36
Name           : python36
Stream         : 3.6 [d][e][a]
Version        : 8010020190724083915
Context        : a920e634
Architecture   : x86_64
Profiles       : build, common [d]
Default profiles: common
Repo           : rhel-8.2-for-x86_64-appstream-rpms
Summary         : Python programming language, version 3.6
...output omitted...
Hint: [d]efault, [e]nabled, [x]disabled, [i]nstalled, [a]ctive]
```

- 4. Installieren Sie das Modul *python36* aus dem Stream **3.6** und dem **allgemeinen** Profil. Überprüfen Sie den aktuellen Status des Moduls.
- 4.1. Verwenden Sie den Befehl **yum module install**, um das Modul *python36* zu installieren. Verwenden Sie die Syntax **name:stream/profil**, um das Modul *python36* aus dem Stream **3.6** und dem **allgemeinen** Profil zu installieren.



### Anmerkung

Sie können **/profil** weglassen, um das Standardprofil zu verwenden. Wenn Sie **:stream** weglassen, wird der standardmäßige Stream verwendet.

```
[root@servera ~]# yum module install python36:3.6/common
...output omitted...
Is this ok [y/N]: y
...output omitted...
Complete!
```

4.2. Überprüfen Sie den aktuellen Status des Moduls *python36*.

```
[root@servera ~]# yum module list python36
Red Hat Enterprise Linux 8.2 AppStream (dvd)
Name      Stream      Profiles          Summary
python36   3.6 [d][e]    build, common [d] [i]  Python ... version 3.6

Hint: [d]efault, [e]nabled, [x]disabled, [i]nstalled
```

- 5. Wechseln Sie über das Modul *postgresql* des **Server**-Profils zur Verwendung des Streams **10**.

5.1. Verwenden Sie den Befehl **yum module list**, um das Modul *postgresql* und den Stream aufzulisten. Beachten Sie, dass derzeit der Modul-Stream *postgresql:9.6* installiert ist.

```
[root@servera ~]# yum module list postgresql
Red Hat Enterprise Linux 8.2 AppStream (dvd)
Name      Stream      Profiles          Summary
postgresql 9.6 [e]    client, server [d] [i] PostgreSQL server and client ...
postgresql 10 [d]    client, server [d]    PostgreSQL server and client ...
postgresql 12         client, server [d]    PostgreSQL server and client ...

Hint: [d]efault, [e]nabled, [x]disabled, [i]nstalled
```

5.2. Entfernen und deaktivieren Sie den Modul-Stream *postgresql* zusammen mit allen vom Profil installierten Paketen.

```
[root@servera ~]# yum module remove postgresql
...output omitted...
Is this ok [y/N]: y
...output omitted...
Removed:
  postgresql-server-9.6.10-1.module+el8+2470+d1bafa0e.x86_64
  libpq-10.5-1.el8.x86_64  postgresql-9.6.10-1.module+el8+2470+d1bafa0e.x86_64
Complete
```

5.3. Setzen Sie das Modul *postgresql* und seine Streams zurück.

```
[root@servera ~]# yum module reset postgresql
=====
Package      Arch      Version      Repository      Size
=====
Resetting modules:
```

```
postgresql
```

```
Transaction Summary
```

```
=====  
Is this ok [y/N]: y
```

```
Complete!
```

- 5.4. Wechseln Sie mit dem Befehl **yum module install** zum Modul-Stream *postgresql:10*.

```
[root@servera ~]# yum module install postgresql:10  
...output omitted...  
Is this ok [y/N]: y  
...output omitted...  
Complete!
```

- 5.5. Überprüfen Sie, ob das Modul *postgresql* zum Stream **10** gewechselt ist.

```
[root@servera ~]# yum module list postgresql  
Red Hat Enterprise Linux 8.2 AppStream (dvd)  
Name      Stream   Profiles          Summary  
postgresql  9.6      client, server [d]  PostgreSQL server and client ...  
postgresql  10 [d][e] client, server [d] [i] PostgreSQL server and client ...  
postgresql  12      client, server [d]  PostgreSQL server and client ...  
  
Hint: [d]efault, [e]nabled, [x]disabled, [i]nstalled
```

- 6. Entfernen und deaktivieren Sie den Modul-Stream *postgresql* zusammen mit allen vom Profil installierten Paketen.

- 6.1. Verwenden Sie den Befehl **yum module remove**, um das Modul *postgresql* zu entfernen. Der Befehl entfernt auch alle von diesem Modul installierten Pakete.

```
[root@servera ~]# yum module remove postgresql  
...output omitted...  
Is this ok [y/N]: y  
...output omitted...  
Complete!
```

- 6.2. Deaktivieren Sie den Modul-Stream *postgresql*.

```
[root@servera ~]# yum module disable postgresql  
...output omitted...  
Is this ok [y/N]: y  
...output omitted...  
Complete!
```

- 6.3. Stellen Sie sicher, dass der Modul-Stream *postgresql* entfernt und deaktiviert wird.

```
[root@servera ~]# yum module list postgresql
Red Hat Enterprise Linux 8.2 AppStream (dvd)
Name      Stream   Profiles          Summary
postgresql 9.6 [x]  client, server [d] PostgreSQL server and client ...
postgresql 10 [d][x] client, server [d] PostgreSQL server and client ...
postgresql 12 [x]  client, server [d] PostgreSQL server and client ...

Hint: [d]efault, [e]nabled, [x]disabled, [i]nstalled
```

► 7. Beenden Sie **servera**.

```
[root@servera ~]# exit
logout
[student@servera ~]$ exit
logout
Connection to servera closed.
[student@workstation ~]$
```

## Beenden

Führen Sie auf **workstation** das Skript **lab software-module finish** aus, um diese Übung abzuschließen. Dieses Skript entfernt alle während der Übung auf **servera** installierten Module.

```
[student@workstation ~]$ lab software-module finish
```

Hiermit ist die angeleitete Übung beendet.

## ► Praktische Übung

# Installieren und Aktualisieren von Softwarepaketen

### Leistungscheckliste

In dieser Übung verwalten Sie Software-Repositorys und Modul-Streams und installieren und aktualisieren Pakete von diesen Repositorys und Streams.

### Ergebnisse

Es werden folgende Fähigkeiten vermittelt:

- Verwalten von Software-Repositorys und Modul-Streams.
- Installieren und Aktualisieren von Paketen aus Repositorys und Streams.
- Installieren eines RPM-Pakets.

### Bevor Sie Beginnen

Melden Sie sich als Benutzer **student** mit dem Passwort **student** bei **workstation** an.

Führen Sie auf **workstation** den Befehl **lab software-review start** aus. Dieses Skript stellt sicher, dass **serverb** verfügbar ist. Außerdem lädt es alle für die praktische Übung erforderlichen Pakete herunter.

```
[student@workstation ~]$ lab software-review start
```

1. Konfigurieren Sie auf **serverb** ein Software-Repository, um Updates zu erhalten. Nennen Sie das Repository **errata** und konfigurieren Sie das Repository in der Datei **/etc/yum.repos.d/errata.repo**. Es sollte auf [http://content.example.com/rhel8.2/x86\\_64/rhcsa-practice/errata](http://content.example.com/rhel8.2/x86_64/rhcsa-practice/errata) zugreifen. Aktivieren Sie nicht GPG-Signaturen.
2. Installieren Sie auf **serverb** das neue Paket **xsane-gimp** und das Modul **Apache HTTP Server** aus dem Stream **2.4** und dem **allgemeinen** Profil.
3. Aus Sicherheitsgründen sollte **serverb** nicht in der Lage sein zu drucken. Dazu entfernen Sie das Paket **cups**. Beenden Sie das **root**-Konto.
4. Das Startskript lädt auch das Paket **rhcsa-script-1.0.0-1.noarch.rpm** in das Verzeichnis **/home/student** auf **servera** herunter.  
Bestätigen Sie, dass das Paket **rhcsa-script-1.0.0-1.noarch.rpm** auf **serverb** verfügbar ist. Installieren Sie das Paket. Sie benötigen Superuser-Berechtigungen für die Installation des Pakets. Prüfen Sie, ob das Paket installiert wurde. Beenden Sie **serverb**.

### Bewertung

Führen Sie auf **workstation** das Skript **lab software-review grade** aus, um den Erfolg dieser praktischen Übung zu bestätigen.

```
[student@workstation ~]$ lab software-review grade
```

## Beenden

Führen Sie auf **workstation** das Skript **lab software-review finish** aus, um diese Übung zu beenden. Dieses Skript entfernt das Repository und die während dieser Übung erstellten Pakete.

```
[student@workstation ~]$ lab software-review finish
```

Damit ist die praktische Übung abgeschlossen.

## ► Lösung

# Installieren und Aktualisieren von Softwarepaketen

### Leistungscheckliste

In dieser Übung verwalten Sie Software-Repositories und Modul-Streams und installieren und aktualisieren Pakete von diesen Repositorys und Streams.

### Ergebnisse

Es werden folgende Fähigkeiten vermittelt:

- Verwalten von Software-Repositories und Modul-Streams.
- Installieren und Aktualisieren von Paketen aus Repositorys und Streams.
- Installieren eines RPM-Pakets.

### Bevor Sie Beginnen

Melden Sie sich als Benutzer **student** mit dem Passwort **student** bei **workstation** an.

Führen Sie auf **workstation** den Befehl **lab software-review start** aus. Dieses Skript stellt sicher, dass **serverb** verfügbar ist. Außerdem lädt es alle für die praktische Übung erforderlichen Pakete herunter.

```
[student@workstation ~]$ lab software-review start
```

1. Konfigurieren Sie auf **serverb** ein Software-Repository, um Updates zu erhalten. Nennen Sie das Repository **errata** und konfigurieren Sie das Repository in der Datei **/etc/yum.repos.d/errata.repo**. Es sollte auf [http://content.example.com/rhel8.2/x86\\_64/rhcsa-practice/errata](http://content.example.com/rhel8.2/x86_64/rhcsa-practice/errata) zugreifen. Aktivieren Sie nicht GPG-Signaturen.
  - 1.1. Melden Sie sich auf **workstation** mit dem Befehl **ssh** bei **serverb** als Benutzer **student** an.

```
[student@workstation ~]$ ssh student@serverb
...output omitted...
[student@serverb ~]$
```

- 1.2. Verwenden Sie den Befehl **sudo -i**, um zum **root**-Benutzer zu wechseln.

```
[student@serverb ~]$ sudo -i
[sudo] password for student: student
[root@serverb ~]#
```

- 1.3. Erstellen Sie die Datei **/etc/yum.repos.d/errata.repo** mit dem folgenden Inhalt:

```
[errata]
name=Red Hat Updates
baseurl=http://content.example.com/rhel8.2/x86_64/rhcsa-practice/errata
enabled=1
gpgcheck=0
```

2. Installieren Sie auf **serverb** das neue Paket *xsane-gimp* und das Modul *Apache HTTP Server* aus dem Stream **2.4** und dem **allgemeinen** Profil.
  - 2.1. Verwenden Sie den Befehl **yum list**, um die verfügbaren Pakete für *xsane-gimp* aufzulisten.

```
[root@serverb ~]# yum list xsane-gimp
Last metadata expiration check: 0:24:30 ago on Thu 07 Mar 2019 03:50:55 PM CET.
Available Packages
xsane-gimp.x86_64      0.999-30.el8      rhel-8.2-for-x86_64-appstream-rpms
```

- 2.2. Installieren Sie die neueste Version des Pakets *xsane-gimp* mit dem Befehl **yum install**.

```
[root@serverb ~]# yum install xsane-gimp
...output omitted...
Install 59 Packages

Total download size: 53 M
Installed size: 217 M
Is this ok [y/N]: y
...output omitted...
Complete!
[root@serverb ~]#
```

- 2.3. Führen Sie die verfügbaren Module und Streams auf. Suchen Sie nach dem Modul *httpd*. Verwenden Sie den Befehl **yum install**, um das Modul *httpd* mit dem Stream **2.4** und dem **allgemeinen** Profil zu installieren.

```
[student@serverb ~]$ yum module list
Name      Stream      Profiles          Summary
...output omitted...
httpd     2.4 [d]    common [d], devel, minimal   Apache HTTP Server
...output omitted...
[root@serverb ~]# yum module install httpd:2.4/common
Install 10 Packages

Total download size: 2.1 M
Installed size: 5.7 M
Is this ok [y/N]: y
...output omitted...
Complete!
[root@serverb ~]#
```

3. Aus Sicherheitsgründen sollte **serverb** nicht in der Lage sein zu drucken. Dazu entfernen Sie das Paket *cups*. Beenden Sie das **root**-Konto.

- 3.1. Verwenden Sie den Befehl **yum list**, um das installierte Paket *cups* aufzulisten.

```
[root@serverb ~]# yum list cups
Installed Packages
cups.x86_64           1:2.2.6-33.el8          @rhel-8.2-for-x86_64-appstream-rpms
[root@serverb ~]#
```

- 3.2. Verwenden Sie den Befehl **yum remove**, um das Paket *cups* zu entfernen.

```
[root@serverb ~]# yum remove cups.x86_64
...output omitted...
Remove 9 Packages

Freed space: 11 M
Is this ok [y/N]: y
...output omitted...
Complete!
```

- 3.3. Beenden Sie das **root**-Konto.

```
[root@serverb ~]# exit
[student@serverb ~]$
```

4. Das Startskript lädt auch das Paket *rhcsa-script-1.0.0-1.noarch.rpm* in das Verzeichnis **/home/student** auf **servera** herunter.

Bestätigen Sie, dass das Paket *rhcsa-script-1.0.0-1.noarch.rpm* auf **serverb** verfügbar ist. Installieren Sie das Paket. Sie benötigen Superuser-Berechtigungen für die Installation des Pakets. Prüfen Sie, ob das Paket installiert wurde. Beenden Sie **serverb**.

- 4.1. Verwenden Sie den Befehl **rpm** zur Bestätigung, dass das Paket *rhcsa-script-1.0.0-1.noarch.rpm* auf **serverb** verfügbar ist, indem Sie die Paketinformationen anzeigen.

```
[student@serverb ~]$ rpm -q -p rhcsa-script-1.0.0-1.noarch.rpm -i
Name        : rhcsa-script
Version     : 1.0.0
Release     : 1
Architecture: noarch
Install Date: (not installed)
Group       : System
Size        : 1056
License     : GPL
Signature   : (none)
Source RPM  : rhcsa-script-1.0.0-1.src.rpm
Build Date  : Wed 06 Mar 2019 11:29:46 AM CET
Build Host  : foundation0.ilt.example.com
Relocations : (not relocatable)
Packager    : Snehangshu Karmakar
URL         : http://example.com
Summary     : RHCSA Practice Script
```

```
Description :  
A RHCSA practice script.  
The package changes the motd.
```

- 4.2. Mit dem Befehl **sudo yum localinstall** können Sie das Paket *rhcsa-script-1.0.0-1.noarch.rpm* installieren. Das Passwort lautet **student**.

```
[student@serverb ~]$ sudo yum localinstall \  
rhcsa-script-1.0.0-1.noarch.rpm  
[sudo] password for student: student  
Last metadata expiration check: 1:31:22 ago on Thu 07 Mar 2019 03:50:55 PM CET.  
Dependencies resolved.  
=====  
 Package           Arch    Version      Repository      Size  
=====  
Installing:  
 rhcsa-script     noarch  1.0.0-1       @commandline      7.6 k  
  
Transaction Summary  
=====  
Install 1 Package  
  
Total size: 7.6 k  
Installed size: 1.0 k  
Is this ok [y/N]: y  
Downloading Packages:  
Running transaction check  
Transaction check succeeded.  
Running transaction test  
Transaction test succeeded.  
Running transaction  
Preparing          :                                1/1  
Running scriptlet: rhcsa-script-1.0.0-1.noarch      1/1  
Installing        : rhcsa-script-1.0.0-1.noarch      1/1  
Running scriptlet: rhcsa-script-1.0.0-1.noarch      1/1  
Verifying         : rhcsa-script-1.0.0-1.noarch      1/1  
  
Installed:  
 rhcsa-script-1.0.0-1.noarch  
  
Complete!
```

- 4.3. Mit dem Befehl **rpm** können Sie überprüfen, ob das Paket installiert ist.

```
[student@serverb ~]$ rpm -q rhcsa-script  
rhcsa-script-1.0.0-1.noarch  
[student@serverb ~]$
```

- 4.4. Beenden Sie **serverb**.

```
[student@serverb ~]$ exit  
logout  
Connection to serverb closed.  
[student@workstation ~]$
```

## Bewertung

Führen Sie auf **workstation** das Skript **lab software-review grade** aus, um den Erfolg dieser praktischen Übung zu bestätigen.

```
[student@workstation ~]$ lab software-review grade
```

## Beenden

Führen Sie auf **workstation** das Skript **lab software-review finish** aus, um diese Übung zu beenden. Dieses Skript entfernt das Repository und die während dieser Übung erstellten Pakete.

```
[student@workstation ~]$ lab software-review finish
```

Damit ist die praktische Übung abgeschlossen.

# Zusammenfassung

---

In diesem Kapitel erhielten Sie folgende Informationen:

- Red Hat Subscription Management bietet Tools, um Rechnern Berechtigungen für Produktsubskriptionen zu erteilen, Updates für Softwarepakete zu erhalten und Informationen zu Supportverträgen und Subskriptionen nachzuverfolgen, die von den Systemen verwendet werden.
- Software wird als RPM-Pakete bereitgestellt, mit denen sich Software auf einfache Weise auf dem System installieren, aktualisieren und deinstallieren lässt.
- Mit dem Befehl **rpm** kann eine lokale Datenbank abgefragt werden, um Informationen zum Inhalt von installierten Paketen bereitzustellen und heruntergeladene Paketdateien zu installieren.
- **yum** ist ein leistungsstarkes Befehlszeilentool, mit dem Softwarepakete installiert, aktualisiert, entfernt und abgefragt werden können.
- Red Hat Enterprise Linux 8 verwendet „Application Streams“, um ein einzelnes Repository zum Hosten mehrerer Versionen von Anwendungspaketen und deren Abhängigkeiten bereitzustellen.

## Kapitel 15

# Zugriff auf Linux-Dateisysteme

### Ziel

Zugreifen auf sowie Prüfen und Verwenden von vorhandenen Dateisystemen auf an einen Linux-Server angeschlossenen Storage.

### Ziele

- Erläutern, was ein Blockgerät ist, Interpretieren der Dateinamen von Speichergeräten und Identifizieren des vom Dateisystem für ein bestimmtes Verzeichnis oder eine bestimmte Datei verwendeten Speichergeräts.
- Zugreifen auf Dateisysteme durch Anhängen an ein Verzeichnis in der Dateisystemhierarchie.
- Suchen nach Dateien in gemounteten Dateisystemen mit den Befehlen **find** und **locate**.

### Abschnitte

- Identifizieren von Dateisystemen und Geräten (mit Quiz)
- Mounten und Unmounten von Dateisystemen (mit angeleiteter Übung)
- Suchen von Dateien im System (mit angeleiteter Übung)

### Praktische Übung

Zugriff auf Linux-Dateisysteme

# Identifizieren von Dateisystemen und Geräten

---

## Ziele

Nach Abschluss dieses Abschnitts sollten Sie in der Lage sein, ein Verzeichnis in der Dateisystemhierarchie zu identifizieren und zu erkennen, auf welchem Gerät es gespeichert ist.

## Konzepte zur Speicherverwaltung

Der Zugriff auf Dateien auf einem Linux-Server erfolgt über die Dateisystemhierarchie, einen einzelnen invertierten Verzeichnisbaum. Diese Dateisystemhierarchie setzt sich aus *Dateisysteme* zusammen, die von den für Ihr System verfügbaren Speichergeräten bereitgestellt werden. Jedes Dateisystem ist ein Speichergerät, das zum Speichern von Dateien formatiert wurde.

In gewissem Sinne stellt die Linux-Dateisystemhierarchie eine Sammlung von Dateisystemen auf separaten Speichergeräten so dar, als ob es sich um einen Satz von Dateien auf einem riesigen Speichergerät handelt, auf dem Sie navigieren können. In den meisten Fällen müssen Sie nicht wissen, auf welchem Speichergerät sich eine bestimmte Datei befindet, Sie müssen nur das Verzeichnis kennen, in dem sich diese Datei befindet.

Manchmal kann es jedoch wichtig sein. Möglicherweise müssen Sie ermitteln, wie voll ein Speichergerät ist und welche Verzeichnisse in der Dateisystemhierarchie betroffen sind. Eventuell sind in den Protokollen eines Speichergeräts Fehler enthalten und Sie müssen wissen, welche Dateisysteme gefährdet sind. Vielleicht möchten Sie nur einen Hardlink zwischen zwei Dateien erstellen und Sie müssen wissen, ob sie sich im selben Dateisystem befinden, um festzustellen, ob dies möglich ist.

## Dateisysteme und Mount-Punkte

Um den Inhalt eines Dateisystems in der Dateisystemhierarchie verfügbar zu machen, muss es in einem leeren Verzeichnis *gemountet* werden. Dieses Verzeichnis heißt *Mount-Punkt*. Wenn Sie dieses Verzeichnis mit **ls** auflisten, wird der Inhalt des gemounteten Dateisystems angezeigt und Sie können wie üblich auf diese Dateien zugreifen und sie verwenden. Viele Dateisysteme werden beim Booten automatisch gemountet.

Wenn Sie nur mit Microsoft Windows-Laufwerksbuchstaben gearbeitet haben, ist dies ein grundlegend anderes Konzept. Es ähnelt in gewisser Weise der Funktion für über NTFS gemountete Ordner.

## Dateisysteme, Speicher und Blockgeräte

Der Low-Level-Zugriff auf Speichergeräte unter Linux erfolgt über einen speziellen Dateityp namens *Blockgerät*. Diese Blockgeräte müssen mit einem Dateisystem formatiert werden, bevor sie gemountet werden können.

Blockgerätedateien werden im Verzeichnis **/dev** zusammen mit anderen Gerätedateien gespeichert. Gerätedateien werden vom Betriebssystem automatisch erstellt. In Red Hat Enterprise Linux wird das erste erkannte SATA/PATA-, SAS-, SCSI- oder USB-Laufwerk als **/dev/sda**, die zweite als **/dev/sdb** usw. bezeichnet. Diese Namen stellen das gesamte Laufwerk dar.

Andere Speichertypen haben andere Benennungsarten.

## Benennung von Blockgeräten

| Typ des Geräts                                                            | Gerätebenennungsmuster                                    |
|---------------------------------------------------------------------------|-----------------------------------------------------------|
| Über SATA/SAS/USB angeschlossener Speicher                                | <code>/dev/sda</code> , <code>/dev/sdb</code> ...         |
| <b>virtio-blk</b> paravirtualisierter Speicher (einige virtuelle Rechner) | <code>/dev/vda</code> , <code>/dev/vdb</code> ...         |
| Über NVMe angeschlossener Speicher (viele SSDs)                           | <code>/dev/nvme0</code> , <code>/dev/nvme1</code> ...     |
| SD/MMC/eMMC-Speicher (SD-Karten)                                          | <code>/dev/mmcblk0</code> , <code>/dev/mmcblk1</code> ... |



### Anmerkung

Viele virtuelle Rechner verwenden den neueren paravirtualisierten Speicher **virtio-scsi**, dessen Benennung im Stil `/dev/sd*` erfolgt.

## Laufwerkspartitionen

Normalerweise verwenden Sie nicht das gesamte Speichergerät als ein Dateisystem. Speichergeräte sind in der Regel in kleinere Teile aufgeteilt, die als *Partitionen* bezeichnet werden.

Mit Partitionen teilen Sie Laufwerke in mehrere Bereiche auf: Die verschiedenen Partitionen können jeweils mit unterschiedlichen Dateisystemen formatiert oder für unterschiedliche Zwecke verwendet werden. So kann beispielsweise eine Partition Benutzerverzeichnisse enthalten, während eine andere Systemdaten und Protokolle beinhalten kann. Füllt ein Benutzer die Benutzerverzeichnispartition vollständig mit Daten, ist in der Systempartition möglicherweise weiterhin Kapazität verfügbar.

Partitionen sind eigenständige Blockgeräte. Auf über SATA angeschlossenem Speicher ist die erste Partition auf dem ersten Laufwerk `/dev/sda1`. Die dritte Partition auf dem zweiten Laufwerk ist `/dev/sdb3` usw. Paravirtualisierte Speichergeräte haben ein ähnliches Benennungssystem.

Die Benennung der Partitionen bei über NVMe angeschlossenen SSD-Geräten ist unterschiedlich. In diesem Fall ist die erste Partition auf dem ersten Laufwerk `/dev/nvme0p1`. Die dritte Partition auf dem zweiten Laufwerk ist `/dev/nvme1p3` usw. SD- oder MMC-Karten haben ein ähnliches Benennungssystem.

Eine lange Auflistung der Geräteliste `/dev/sda1` auf `host` gibt den speziellen Dateityp als **b** an, was für Blockgerät steht:

```
[user@host ~]$ ls -l /dev/sda1
brw-rw----. 1 root disk 8, 1 Feb 22 08:00 /dev/sda1
```

## Logische Volumes

Eine weitere Möglichkeit für die Organisation von Datenträgern und Partitionen ist das *Logical Volume Management* (LVM) (logische Datenträgerverwaltung). Mit LVM können ein oder mehr Blockgeräte in einen Storage Pool mit der Bezeichnung *Volume-Gruppe* zusammengeführt

werden. Der Speicherplatz in der Volume-Gruppe wird dann an ein oder mehrere *logische Volumes* verteilt; dies sind die funktionalen Entsprechungen einer Partition auf einer physischen Festplatte.

Das LVM-System weist Volume-Gruppen und logischen Volumes bei der Erstellung Namen zu. LVM erstellt ein Verzeichnis in **/dev**, das dem Gruppennamen entspricht, und erstellt dann in diesem neuen Verzeichnis einen symbolischen Link mit demselben Namen wie das logische Volume. Diese logische Volume-Datei kann dann gemountet werden. Beispiel: Wenn eine Volume-Gruppe den Namen **myvg** hat und das logische Volume in dieser Gruppe **mylv** heißt, lautet der vollständige Pfadname zur Gerätedatei des logischen Volumes **/dev/myvg/mylv**.



### Anmerkung

Die oben erwähnte Form des Gerätenamens bei logischen Volumes wird als symbolischer Link zu der tatsächlichen Gerätedatei implementiert, die für den Zugriff verwendet wird. Dieser kann zwischen den Bootvorgängen variieren. Es gibt eine andere Form von Gerätenamen für logische Volumes, die mit Dateien in **/dev/mapper** verknüpft sind, die häufig verwendet werden und die auch symbolische Links zur eigentlichen Gerätedatei sind.

## Prüfen von Dateisystemen

Um einen Überblick über die lokalen und Remote-Dateisystemgeräte und die verfügbare freie Speicherkapazität zu erhalten, führen Sie den Befehl **df** aus. Wenn der Befehl **df** ohne Argumente ausgeführt wird, werden die insgesamt vorhandene Datenträgerkapazität, die belegte Datenträgerkapazität, die freie Datenträgerkapazität und der verwendete Prozentsatz der insgesamt vorhandenen Datenträgerkapazität für alle gemounteten standardmäßigen Dateisysteme angegeben. Der Bericht bezieht sich sowohl auf lokale als auch auf Remote-Dateisysteme.

Das folgende Beispiel zeigt die Dateisysteme und Mount-Punkte auf **host** an.

```
[user@host ~]$ df
Filesystem      1K-blocks    Used Available Use% Mounted on
/devtmpfs        912584       0   912584   0% /dev
tmpfs           936516       0   936516   0% /dev/shm
tmpfs           936516   16812   919704   2% /run
tmpfs           936516       0   936516   0% /sys/fs/cgroup
/dev/vda3      8377344 1411332   6966012  17% /
/dev/vda1      1038336 169896   868440  17% /boot
tmpfs          187300       0   187300   0% /run/user/1000
```

Die Partitionierung des Systems **host** zeigt zwei physische Dateisysteme, die auf **/** und **/boot** gemountet sind. Dies ist für virtuelle Rechner gängig. Die Geräte **tmpfs** und **devtmpfs** sind Dateisysteme im Systemspeicher. Alle in **tmpfs** oder **devtmpfs** geschriebenen Dateien sind nach einem Systemneustart verschwunden.

Für eine verbesserte Lesbarkeit der Ausgabegrößen stehen zwei für den Menschen lesbare Optionen zur Verfügung: **-h** oder **-H**. Der Unterschied zwischen diesen Optionen ist, dass **-h** Berichte in KiB ( $2^{10}$ ), MiB ( $2^{20}$ ) oder GiB ( $2^{30}$ ) ausgibt, während die Option **-H** Berichte in SI-Einheiten ausgibt: KB ( $10^3$ ), MB ( $10^6$ ), GB ( $10^9$ ) usw. Festplattenhersteller geben in der Werbung für ihre Produkte in der Regel SI-Einheiten an.

Zeigen Sie einen Bericht zu den Dateisystemen auf dem System **host** an, in dem alle Einheiten in ein für den Menschen lesbaren Format konvertiert wurden:

```
[user@host ~]$ df -h
Filesystem      Size  Used Avail Use% Mounted on
devtmpfs        892M    0  892M   0% /dev
tmpfs          915M    0  915M   0% /dev/shm
tmpfs          915M   17M  899M   2% /run
tmpfs          915M    0  915M   0% /sys/fs/cgroup
/dev/vda3       8.0G  1.4G  6.7G  17% /
/dev/vda1     1014M 166M  849M  17% /boot
tmpfs         183M    0  183M   0% /run/user/1000
```

Um detailliertere Informationen über den von einer bestimmten Verzeichnisstruktur belegten Speicherplatz zu erhalten, verwenden Sie den Befehl **du**. Für den Befehl **du** stehen die Optionen **-h** und **-H** zur Konvertierung der Ausgabe in ein für Menschen lesbaren Format zur Verfügung. Mit dem Befehl **du** wird die Größe aller Dateien in der aktuellen Verzeichnisstruktur rekursiv angezeigt.

Zeigen Sie einen Bericht zur Datenträgerbelegung für das Verzeichnis **/usr/share** auf **host** an:

```
[root@host ~]# du /usr/share
...output omitted...
176 /usr/share/sm智ntools
184 /usr/share/nano
8 /usr/share/cmake/bash-completion
8 /usr/share/cmake
356676 /usr/share
```

Zeigen Sie einen Bericht in einem für Menschen lesbaren Format zur Datenträgerbelegung für das Verzeichnis **/usr/share** auf **host** an:

```
[root@host ~]# du -h /var/log
...output omitted...
176K /usr/share/sm智ntools
184K /usr/share/nano
8.0K /usr/share/cmake/bash-completion
8.0K /usr/share/cmake
369M /usr/share
```



### Literaturhinweise

Manpages **df(1)** und **du(1)**

## ► Quiz

# Identifizieren von Dateisystemen und Geräten

Wählen Sie die richtigen Antworten auf die folgenden Fragen aus:

- ▶ 1. Wie heißt die Gerätedatei eines gesamten SATA-Laufwerks im Verzeichnis /dev?
  - a. /dev/vda
  - b. /dev/sda1
  - c. /dev/sda
  - d. /dev/vg\_install/lv\_home
  
- ▶ 2. Wählen Sie den Namen der Gerätedatei für die dritte Partition auf dem zweiten SATA-Laufwerk aus.
  - a. /dev/vda2
  - b. /dev/sda3
  - c. /dev/sdb2
  - d. /dev/sdb3
  
- ▶ 3. Wie lautet der Name der Gerätedatei des gesamten zweiten virtio-blk-Laufwerks, das an einen virtuellen Rechner angeschlossen ist?
  - a. /dev/vda2
  - b. /dev/sda2
  - c. /dev/vdb2
  - d. /dev/vdb
  
- ▶ 4. Wählen Sie den richtigen Namen der Gerätedatei für die dritte Partition auf dem zweiten virtio-blk-Laufwerk aus, das an einen virtuellen Rechner angeschlossen ist.
  - a. /dev/vda3
  - b. /dev/sda3
  - c. /dev/vdb3
  - d. /dev/vda3
  
- ▶ 5. Welcher Befehl bietet einen Überblick über die Mount-Punkte des Dateisystems und die verfügbare freie Speicherkapazität in SI-Einheiten?
  - a. df
  - b. df -H
  - c. df -h
  - d. du -h

## ► Lösung

# Identifizieren von Dateisystemen und Geräten

Wählen Sie die richtigen Antworten auf die folgenden Fragen aus:

- ▶ 1. Wie heißt die Gerätedatei eines gesamten SATA-Laufwerks im Verzeichnis /dev?
  - a. /dev/vda
  - b. /dev/sda1
  - c. /dev/sda
  - d. /dev/vg\_install/lv\_home
  
- ▶ 2. Wählen Sie den Namen der Gerätedatei für die dritte Partition auf dem zweiten SATA-Laufwerk aus.
  - a. /dev/vda2
  - b. /dev/sda3
  - c. /dev/sdb2
  - d. /dev/sdb3
  
- ▶ 3. Wie lautet der Name der Gerätedatei des gesamten zweiten virtio-blk-Laufwerks, das an einen virtuellen Rechner angeschlossen ist?
  - a. /dev/vda2
  - b. /dev/sda2
  - c. /dev/vdb2
  - d. /dev/vdb
  
- ▶ 4. Wählen Sie den richtigen Namen der Gerätedatei für die dritte Partition auf dem zweiten virtio-blk-Laufwerk aus, das an einen virtuellen Rechner angeschlossen ist.
  - a. /dev/vda3
  - b. /dev/sda3
  - c. /dev/vdb3
  - d. /dev/vda3
  
- ▶ 5. Welcher Befehl bietet einen Überblick über die Mount-Punkte des Dateisystems und die verfügbare freie Speicherkapazität in SI-Einheiten?
  - a. df
  - b. df -H
  - c. df -h
  - d. du -h

# Mounten und Unmounten von Dateisystemen

---

## Ziele

Nach Abschluss dieses Abschnitts sollten Sie auf den Inhalt von Dateisystemen zugreifen können, indem Sie Dateisysteme in der Dateisystemhierarchie hinzufügen und daraus entfernen.

## Manuelles Mounten von Dateisystemen

Ein auf einem Wechseldatenträger befindliches Dateisystem muss manuell gemountet werden, um darauf zugreifen zu können. Mit dem Befehl **mount** kann der Benutzer **root** ein Dateisystem manuell mounten. Mit dem ersten Argument des Befehls **mount** wird das zu mountende Dateisystem angegeben. Das zweite Argument gibt das Verzeichnis an, das als Mount-Punkt in der Dateisystemhierarchie verwendet werden soll.

Es gibt zwei gebräuchliche Methoden, um das Dateisystem auf einer Laufwerkspartition für den Befehl **mount** anzugeben:

- Mit dem Namen der Gerätedatei in **/dev**, das das Dateisystem enthält.
- Mit der *UUID* des Dateisystems, einer universell eindeutigen Kennung.

Das Mounten eines Gerätes ist relativ einfach. Sie müssen das Gerät identifizieren, das Sie mounten möchten, sicherstellen, dass der Mount-Punkt vorhanden ist, und das Gerät am Mount-Punkt mounten.

## Identifizieren des Blockgeräts

Ein Hotplug-fähiges Speichergerät, wie Festplattenlaufwerke (HDD), Solid-State-Festplatten (SSD) in einem Server-Caddy oder ein USB-Speichergeräte, könnte bei jedem Anschluss an ein System an einen anderen Port angeschlossen werden.

Verwenden Sie den Befehl **lsblk**, um die Details eines angegebenen Blockgeräts oder aller verfügbaren Geräte aufzulisten.

```
[root@host ~]# lsblk
NAME      MAJ:MIN RM  SIZE RO TYPE MOUNTPOINT
vda        253:0   0   12G  0 disk
└─vda1    253:1   0     1G  0 part /boot
└─vda2    253:2   0     1G  0 part [SWAP]
└─vda3    253:3   0   11G  0 part /
vdb        253:16  0   64G  0 disk
└─vdb1    253:17  0   64G  0 part
```

Wenn Sie wissen, dass Sie gerade ein 64 GB großes Speichergerät mit einer Partition hinzugefügt haben, dann können Sie anhand der vorherigen Ausgabe abschätzen, dass **/dev/vdb1** die Partition ist, die Sie mounten möchten.

## Mounten nach Blockgerätename

Im folgenden Beispiel wird das Dateisystem in der Partition **/dev/vdb1** im Verzeichnis **/mnt/data** gemountet.

```
[root@host ~]# mount /dev/vdb1 /mnt/data
```

Um ein Dateisystem zu mounten, muss das Zielverzeichnis bereits vorhanden sein. Das Verzeichnis **/mnt** ist standardmäßig vorhanden und dient als temporärer Mount-Punkt.

Sie können das Verzeichnis **/mnt** verwenden oder besser noch ein Unterverzeichnis von **/mnt** als temporären Mount-Punkt erstellen, es sei denn, Sie haben einen guten Grund, einen bestimmten Speicherort in der Dateisystemhierarchie als Mount-Punkt zu verwenden.



### Wichtig

Wenn das Verzeichnis, das als Mount-Punkt dient, nicht leer ist, kann auf Dateien, die in dieses Verzeichnis vor dem Mounten des Dateisystems kopiert wurden, erst wieder nach dem Unmounten des Dateisystems zugegriffen werden.

Dieser Ansatz funktioniert kurzfristig gut. Die Reihenfolge, in der das Betriebssystem Laufwerke erkennt, kann sich jedoch ändern, wenn dem System Geräte hinzugefügt oder daraus entfernt werden. Dadurch wird der Gerätename geändert, der dem jeweiligen Speichergerät zugeordnet ist. Ein besserer Ansatz wäre das Mounten anhand bestimmter in das Dateisystem integrierter Merkmale.

## Mounten nach Dateisystem-UUID

Die UUID ist eine dauerhafte Kennung, die einem Dateisystem zugeordnet ist. Dabei handelt es sich um eine sehr lange hexadezimale Zahl, die als universell eindeutige Kennung fungiert. Diese UUID ist Teil des Dateisystems und verändert sich nur, wenn das Dateisystem erneut erstellt wird.

Der Befehl **lsblk -fp** listet den vollständigen Pfad des Geräts mit den UUIDs und Mount-Punkten sowie dem Typ des Dateisystems in der Partition auf. Wenn das Dateisystem nicht gemountet wird, ist der Mount-Punkt leer.

```
[root@host ~]# lsblk -fp
  NAME   FSTYPE LABEL UUID                                     MOUNTPOINT
/dev/vda
└─/dev/vda1 xfs   23ea8803-a396-494a-8e95-1538a53b821c /boot
└─/dev/vda2 swap   cdf61ded-534c-4bd6-b458-cab18b1a72ea [SWAP]
└─/dev/vda3 xfs   44330f15-2f9d-4745-ae2e-20844f22762d /
/dev/vdb
└─/dev/vdb1 xfs   46f543fd-78c9-4526-a857-244811be2d88
```

Mounten Sie das Dateisystem anhand der UUID des Dateisystems.

```
[root@host ~]# mount UUID="46f543fd-78c9-4526-a857-244811be2d88" /mnt/data
```

## Automatisches Mounten von Wechseldatenträgern

Wenn Sie angemeldet sind und die grafische Desktopumgebung verwenden, wird beim Einlegen eines Wechseldatenträgers dieser automatisch gemountet.

Der Wechseldatenträger wird in **/run/media/USERNAME/LABEL** gemountet. **USERNAME** ist dabei der Name des Benutzers, der bei der grafischen Umgebung angemeldet ist, und **LABEL** ist

eine Kennung, meist der Name, der dem Dateisystem bei der Erstellung zugewiesen wurde, falls verfügbar.

Bevor Sie das Gerät entfernen, sollten Sie es manuell ummounten.

## Unmounten von Dateisystemen

Beim Herunterfahren und Booten werden alle Dateisysteme automatisch geunmountet.

Im Rahmen dieses Prozesses werden alle im Arbeitsspeicher zwischengespeicherten Dateisystemdaten auf das Speichergerät geleert, sodass die Dateisystemdaten nicht beschädigt werden.



### Warnung

Dateisystemdaten werden häufig im Arbeitsspeicher zwischengespeichert. Um zu vermeiden, dass Daten auf dem Datenträger beschädigt werden, müssen Sie die Wechseldatenträger daher vor dem Entfernen ummounten. Beim Unmounten werden die Daten vor der Freigabe des Datenträgers synchronisiert, um die Datenintegrität sicherzustellen.

Zum Unmounten eines Dateisystems muss für den Befehl **umount** der Mount-Punkt als Argument angegeben werden.

```
[root@host ~]# umount /mnt/data
```

Das Unmounten ist nicht möglich, wenn das gemountete Dateisystem verwendet wird. Für eine erfolgreiche Ausführung des Befehls **umount** muss der Zugriff aller Prozesse auf Daten unter dem Mount-Punkt eingestellt werden.

Im folgenden Beispiel schlägt **umount** fehl, weil das Dateisystem verwendet wird (die Shell verwendet **/mnt/data** als aktuelles Arbeitsverzeichnis), und es wird eine Fehlermeldung generiert.

```
[root@host ~]# cd /mnt/data
[root@host data]# umount /mnt/data
umount: /mnt/data: target is busy.
```

Mit dem Befehl **lsof** werden alle offenen Dateien und der Prozess aufgeführt, der im angegebenen Verzeichnis darauf zugreift. Dies ist hilfreich, um festzustellen, von welchen Prozessen das erfolgreiche Unmounten des Dateisystems derzeit verhindert wird.

```
[root@host data]# lsof /mnt/data
COMMAND PID USER FD TYPE DEVICE SIZE/OFF NODE NAME
bash    1593 root cwd DIR 253,17      6  128 /mnt/data
lsof    2532 root cwd DIR 253,17     19  128 /mnt/data
lsof    2533 root cwd DIR 253,17     19  128 /mnt/data
```

Nachdem die Prozesse ermittelt wurden, kann eine Maßnahme ergriffen werden, wie zum Beispiel das Warten auf die Beendigung des Prozesses oder das Senden eines **SIGTERM**- oder **SIGKILL**-Signals an den Prozess. In diesem Fall reicht es aus, das aktuelle Arbeitsverzeichnis in ein Verzeichnis außerhalb des Mount-Punktes zu ändern.

```
[root@host data]# cd  
[root@host ~]# umount /mnt/data
```



### Anmerkung

Ein häufiger Grund für Fehler beim Unmounten von Dateisystemen ist, dass eine Bash-Shell den Mount-Punkt oder ein Unterverzeichnis als aktuelles Arbeitsverzeichnis verwendet. Verwenden Sie den Befehl **cd** zum Verlassen des Dateisystems, um dieses Problem zu beheben.



### Literaturhinweise

Manpages **lsblk(8)**, **mount(8)**, **umount(8)** und **lsof(8)**

## ► Angeleitete Übung

# Mounten und Unmounten von Dateisystemen

In dieser Übung üben Sie das Mounten und Unmounten von Dateisystemen.

### Ergebnisse

Sie sollten in der Lage sein, ein neues Dateisystem zu identifizieren, an einem bestimmten Mount-Punkt zu mounten und es anschließend zu ummounten.

### Bevor Sie Beginnen

Melden Sie sich als Benutzer **student** mit dem Passwort **student** bei **workstation** an.

Führen Sie auf **workstation** den Befehl **lab fs-mount start** aus. Dieser Befehl führt ein Startskript aus, das ermittelt, ob der Host **servera** im Netzwerk erreichbar ist. Das Skript erstellt auch eine Partition auf der zweiten Festplatte, die an **servera** angeschlossen ist.

```
[student@workstation ~]$ lab fs-mount start
```

- 1. Melden Sie sich mit dem Befehl **ssh** bei **servera** als Benutzer **student** an.

```
[student@workstation ~]$ ssh student@servera  
...output omitted...  
[student@servera ~]$
```

- 2. Der zweiten Festplatte (**/dev/vdb**) auf **servera** wurde eine neue Partition mit einem Dateisystem hinzugefügt. Mounten Sie die neu verfügbare Partition anhand der UUID am neu erstellten Mount-Punkt **/mnt/newspace**.
- 2.1. Verwenden Sie den Befehl **sudo -i**, um zu **root** zu wechseln, da nur der Benutzer **root** ein Gerät manuell mounten kann.

```
[student@servera ~]$ sudo -i  
[sudo] password for student: student  
[root@servera ~]#
```

- 2.2. Erstellen Sie das Verzeichnis **/mnt/newspace**.

```
[root@servera ~]# mkdir /mnt/newspace
```

- 2.3. Verwenden Sie den Befehl **lsblk** mit der Option **-fp**, um die UUID des Geräts **/dev/vdb1** zu ermitteln.

```
[root@servera ~]# lsblk -fp /dev/vdb
NAME      FSTYPE LABEL UUID                                     MOUNTPOINT
/dev/vdb
└─/dev/vdb1 xfs    a04c511a-b805-4ec2-981f-42d190fc9a65
```

- 2.4. Mounten Sie das Dateisystem mithilfe der UUID im Verzeichnis **/mnt/newspace**. Ersetzen Sie die UUID durch die des Datenträgers **/dev/vdb1** aus der vorherigen Befehlausgabe.

```
[root@servera ~]# mount \
UUID="a04c511a-b805-4ec2-981f-42d190fc9a65" /mnt/newspace
```

- 2.5. Stellen Sie sicher, dass das Gerät **/dev/vdb1** im Verzeichnis **/mnt/newspace** gemountet wurde.

```
[root@servera ~]# lsblk -fp /dev/vdb
NAME      FSTYPE LABEL UUID                                     MOUNTPOINT
/dev/vdb
└─/dev/vdb1 xfs    a04c511a-b805-4ec2-981f-42d190fc9a65 /mnt/newspace
```

- 3. Wechseln Sie in das Verzeichnis **/mnt/newspace**, und erstellen Sie ein neues Verzeichnis **/mnt/newspace/newdir** mit einer leeren Datei **/mnt/newspace/newdir/newfile**.

- 3.1. Wechseln Sie in das Verzeichnis **/mnt/newspace**.

```
[root@servera ~]# cd /mnt/newspace
```

- 3.2. Erstellen Sie das neue Verzeichnis **/mnt/newspace/newdir**.

```
[root@servera newspace]# mkdir newdir
```

- 3.3. Erstellen Sie eine neue leere Datei **/mnt/newspace/newdir/newfile**.

```
[root@servera newspace]# touch newdir/newfile
```

- 4. Unmounten Sie das im Verzeichnis **/mnt/newspace** gemountete Dateisystem.

- 4.1. Verwenden Sie den Befehl **umount**, um **/mnt/newspace** zu ummounten, während das aktuelle Verzeichnis in der Shell weiterhin **/mnt/newspace** ist. Der Befehl **umount** kann das Gerät nicht ummounten.

```
[root@servera newspace]# umount /mnt/newspace
umount: /mnt/newspace: target is busy.
```

- 4.2. Wechseln Sie das aktuelle Verzeichnis in der Shell zu **/root**.

```
[root@servera newspace]# cd
[root@servera ~]#
```

4.3. Jetzt können Sie **/mnt/newspace** erfolgreich ummounten.

```
[root@servera ~]# umount /mnt/newspace
```

► 5. Beenden Sie **servera**.

```
[root@servera ~]# exit  
logout  
[student@servera ~]$ exit  
logout  
Connection to servera closed.  
[student@workstation]$
```

## Beenden

Führen Sie auf **workstation** das Skript **lab fs-mount finish** aus, um diese Übung zu beenden.

```
[student@workstation ~]$ lab fs-mount finish
```

Hiermit ist die angeleitete Übung beendet.

# Suchen von Dateien im System

## Ziele

Nach Abschluss dieses Abschnitts sollten Sie in der Lage sein, mit den Befehlen **find** und **locate** in gemounteten Dateisystemen nach Dateien zu suchen.

## Suchen nach Dateien

Ein Systemadministrator benötigt Tools, um im Dateisystem nach Dateien zu suchen, die bestimmten Kriterien entsprechen. In diesem Kapitel werden zwei Befehle erläutert, mit denen Dateien im Dateisystem gesucht werden können.

- Mit dem Befehl **locate** wird in einem vorab generierten Index nach Dateinamen oder Pfadnamen gesucht und die Ergebnisse werden sofort zurückgegeben.
- Mit dem Befehl **find** wird nach Dateien in Echtzeit gesucht, in dem die Dateisystemhierarchie durchlaufen wird.

## Suchen von Dateien nach Namen

Der Befehl **locate** sucht Dateien basierend auf dem Namen oder Pfad der Datei. Dieser Befehl ist schnell, weil er diese Informationen in der Datenbank **mlocate** sucht. Diese Datenbank wird jedoch nicht in Echtzeit aktualisiert und muss regelmäßig aktualisiert werden, damit die Ergebnisse korrekt sind. Dies bedeutet auch, dass **locate** keine Dateien findet, die seit der letzten Aktualisierung der Datenbank erstellt wurden.

Die Datenbank **locate** wird automatisch täglich aktualisiert. Der Benutzer **root** kann jedoch jederzeit mit dem Befehl **updatedb** eine sofortige Aktualisierung erzwingen.

```
[root@host ~]# updatedb
```

Der Befehl **locate** beschränkt Ergebnisse für unprivilegierte Benutzer. Um den resultierenden Dateinamen sehen zu können, muss der Benutzer über Suchberechtigung für das Verzeichnis verfügen, in dem sich die Datei befindet.

Suchen Sie in Verzeichnisstrukturen, für die **user** auf **host** über Leseberechtigungen verfügt, nach Dateien, deren Name oder Pfad den Bestandteil **passwd** enthält.

```
[user@host ~]$ locate passwd
/etc/passwd
/etc/passwd-
/etc/pam.d/passwd
/etc/security/opasswd
/usr/bin/gpasswd
/usr/bin/grub2-mkpasswd-pbkdf2
/usr/bin/lpasswd
/usr/bin/passwd
...output omitted...
```

## Kapitel 15 | Zugriff auf Linux-Dateisysteme

Es werden auch dann Ergebnisse zurückgegeben, wenn der Dateiname oder Pfad nur teilweise mit der Suchabfrage übereinstimmt.

```
[root@host ~]# locate image
/etc/selinux/targeted-contexts/virtual_image_context
/usr/bin/grub2-mkimage
/usr/lib/sysimage
/usr/lib/dracut/dracut.conf.d/02-generic-image.conf
/usr/lib/firewalld/services/ovirt-imageio.xml
/usr/lib/grub/i386-pc/lnxboot.image
...output omitted...
```

Mit der Option **-i** wird eine Suche durchgeführt, bei der die Groß- und Kleinschreibung nicht beachtet wird. Bei Verwendung dieser Option stimmen alle möglichen Kombinationen aus Groß- und Kleinbuchstaben mit der Suchabfrage überein.

```
[user@host ~]$ locate -i messages
...output omitted...
/usr/share/vim/vim80/lang/zh_TW/LC_MESSAGES
/usr/share/vim/vim80/lang/zh_TW/LC_MESSAGES/vim.mo
/usr/share/vim/vim80/lang/zh_TW.UTF-8/LC_MESSAGES
/usr/share/vim/vim80/lang/zh_TW.UTF-8/LC_MESSAGES/vim.mo
/usr/share/vim/vim80/syntax/messages.vim
/usr/share/vim/vim80/syntax/msmessages.vim
/var/log/messages
```

Mit der Option **-n** wird die Anzahl der vom Befehl **locate** zurückgegebenen Suchergebnisse eingeschränkt. Mit dem folgenden Beispiel wird die Anzahl der von **locate** zurückgegebenen Suchergebnisse auf die ersten fünf Übereinstimmungen beschränkt:

```
[user@host ~]$ locate -n 5 snow.png
/usr/share/icons/HighContrast/16x16/status/weather-snow.png
/usr/share/icons/HighContrast/22x22/status/weather-snow.png
/usr/share/icons/HighContrast/24x24/status/weather-snow.png
/usr/share/icons/HighContrast/256x256/status/weather-snow.png
/usr/share/icons/HighContrast/32x32/status/weather-snow.png
```

## Suchen nach Dateien in Echtzeit

Der Befehl **find** sucht Dateien durch Ausführen einer Echtzeitsuche in der Dateisystemhierarchie. Er ist langsamer als **locate**, aber genauer. Er kann auch nach Dateien basierend auf anderen Kriterien als dem Dateinamen suchen, z. B. den Berechtigungen der Datei, dem Dateityp, ihrer Größe oder der Änderungszeit.

Der Befehl **find** sucht Dateien im Dateisystem anhand des Benutzerkontos, das die Suche ausgeführt hat. Der Benutzer, der den Befehl **find** aufruft, benötigt zur Prüfung des Inhalts eines Verzeichnisses Lese- und Ausführungsrechte für dieses.

Das erste Argument für den Befehl **find** ist das zu durchsuchende Verzeichnis. Wird kein Verzeichnisargument angegeben, beginnt **find** mit der Suche im aktuellen Verzeichnis und sucht ausgehend von hier in allen Unterverzeichnissen nach Übereinstimmungen.

Um anhand des Dateinamens nach Dateien zu suchen, verwenden Sie die Option **-name FILENAME**. Mit dieser Option gibt **find** den Pfad zu Dateien zurück, die genau mit *FILENAME* übereinstimmen. Um beispielsweise nach Dateien mit dem Namen **sshd\_config** zu suchen und dabei im Verzeichnis **/** zu beginnen, führen Sie den folgenden Befehl aus:

```
[root@host ~]# find / -name sshd_config  
/etc/ssh/sshd_config
```



### Anmerkung

Beim Befehl **find** verwenden die Optionen mit vollständigen Wörtern einen einzelnen Bindestrich und die Optionen folgen auf das Pfadnamenargument; dies ist anders als bei den meisten anderen Linux-Befehlen.

Für die Suche nach einem Dateinamen und die Rückgabe aller Ergebnisse mit teilweisen Übereinstimmungen stehen Platzhalterzeichen zur Verfügung. Bei Verwendung von Platzhalterzeichen ist es wichtig, den zu suchenden Dateinamen anzugeben, um zu verhindern, dass die Platzhalter vom Terminal interpretiert werden.

Im folgenden Beispiel wird beginnend im Verzeichnis **/** nach Dateien gesucht, die die Endung **.txt** aufweisen:

```
[root@host ~]# find / -name '*.*.txt'  
/etc/pki/nssdb/pkcs11.txt  
/etc/brltty/brl-lt-all.txt  
/etc/brltty/brl-mb-all.txt  
/etc/brltty/brl-md-all.txt  
/etc/brltty/brl-mn-all.txt  
...output omitted...
```

Um im Verzeichnis **/etc/** nach Dateien zu suchen, die das Wort **pass** an beliebiger Stellen im Namen auf **host** enthalten, führen Sie den folgenden Befehl aus:

```
[root@host ~]# find /etc -name '*pass*'  
/etc/security/opasswd  
/etc/pam.d/passwd  
/etc/pam.d/password-auth  
/etc/passwd-  
/etc/passwd  
/etc/authselect/password-auth
```

Verwenden Sie für eine Suche nach einem bestimmten Dateinamen ohne Unterscheidung nach Groß- und Kleinschreibung die Option **-iname** gefolgt vom Dateinamen. Um ohne Unterscheidung nach Groß- und Kleinschreibung nach Dateien zu suchen, die den Text **messages** im Namen im Verzeichnis **/** auf **host** enthalten, führen Sie den folgenden Befehl aus:

```
[root@host ~]# find / -iname '*messages*'  
...output omitted...  
/usr/share/vim/vim80/lang/zh_CN.UTF-8/LC_MESSAGES  
/usr/share/vim/vim80/lang/zh_CN.cp936/LC_MESSAGES  
/usr/share/vim/vim80/lang/zh_TW/LC_MESSAGES  
/usr/share/vim/vim80/lang/zh_TW.UTF-8/LC_MESSAGES  
/usr/share/vim/vim80/syntax/messages.vim  
/usr/share/vim/vim80/syntax/msmessages.vim
```

## Suchen nach Dateien anhand von Besitzer oder Berechtigung

Mit dem Befehl **find** kann anhand des Besitzers oder von Berechtigungen nach Dateien gesucht werden. Hilfreiche Optionen bei der Suche nach Besitzer sind **-user** und **-group**, die nach Name suchen, sowie **-uid** und **-gid**, die nach ID suchen.

Suchen Sie nach Dateien mit dem Besitzer **user** im Verzeichnis **/home/user** auf **host**.

```
[user@host ~]$ find -user user  
.  
./.bash_logout  
./.bash_profile  
./.bashrc  
./.bash_history
```

Suchen Sie nach Dateien im Besitz der Gruppe **user** im Verzeichnis **/home/user** auf **host**.

```
[user@host ~]$ find -group user  
.  
./.bash_logout  
./.bash_profile  
./.bashrc  
./.bash_history
```

Suchen Sie nach Dateien im Besitz der Benutzer-ID **1000** im Verzeichnis **/home/user** auf **host**.

```
[user@host ~]$ find -uid 1000  
.  
./.bash_logout  
./.bash_profile  
./.bashrc  
./.bash_history
```

Suchen Sie nach Dateien im Besitz der Gruppen-ID **1000** im Verzeichnis **/home/user** auf **host**.

```
[user@host ~]$ find -gid 1000  
.  
./.bash_logout  
./.bash_profile  
./.bashrc  
./.bash_history
```

Die Optionen **-user** und **-group** können zusammen verwendet werden, um nach Dateien zu suchen, bei denen sich der Dateibesitzer und der Gruppenbesitzer unterscheiden. Das folgende Beispiel listet Dateien auf, deren Besitzer der Benutzer **root** ist und die zur Gruppe **mail** gehören.

```
[root@host ~]# find / -user root -group mail  
/var/spool/mail  
...output omitted...
```

Die Option **-perm** wird für die Suche nach Dateien mit einem bestimmten Satz von Berechtigungen verwendet. Berechtigungen können als Oktalwerte mit der Kombination von **4**, **2** und **1** für Lesen, Schreiben und Ausführen definiert werden. Berechtigungen kann ein **/**- oder **--** Zeichen vorangestellt werden.

Eine numerische Berechtigung, der ein **/** vorangestellt ist, findet Dateien, die über mindestens ein Bit (Benutzer, Gruppe oder Andere) dieses Berechtigungssatzes verfügen. Bei einer Datei mit den Berechtigungen **r--r--r--** wird keine Übereinstimmung mit **/222** festgestellt, bei einer Datei mit **rw-r--r--** jedoch schon. Ein **--**-Zeichen vor einer Berechtigung bedeutet, dass für alle drei Felder dieses Bit gesetzt sein muss, also keines der gerade gegebenen Beispiele einen Treffer ergeben würde, **rw-rw-rw->** hingegen schon.

Das nächste Beispiel ist etwas komplexer. Der folgende Befehl findet jede Datei, für die der Benutzer über Lese-, Schreib- und Ausführungsberechtigungen verfügt, Mitglieder der Gruppe über Lese- und Schreibberechtigungen und andere nur schreibgeschützten Zugriff haben:

```
[root@host ~]# find /home -perm 764
```

So finden Sie Dateien, für die der Benutzer zumindest über Schreib- und Ausführendberechtigungen verfügt *und* die Gruppe zumindest Schreibberechtigung hat *und* andere zumindest Lesezugriff haben:

```
[root@host ~]# find /home -perm -324
```

So finden Sie Dateien, für die der Benutzer zumindest über Leseberechtigung verfügt *oder* die Gruppe zumindest die Leseberechtigung hat *oder* andere zumindest Schreibzugriff haben:

```
[root@host ~]# find /home -perm /442
```

Bei Verwendung mit **/** oder **-** verhält sich der Wert **0** wie ein Platzhalter, da er die Bedeutung *eine Berechtigung für mindestens nichts* hat.

Führen Sie zum Suchen nach Dateien im Verzeichnis **/home/user**, für die andere zumindest über Lesezugriff auf **host** verfügen, den folgenden Befehl aus:

```
[user@host ~]$ find -perm -004
```

Suchen Sie alle Dateien im Verzeichnis **/home/user**, für die *other* Schreibberechtigungen auf **host** hat.

```
[user@host ~]$ find -perm -002
```

## Suchen nach Dateien anhand der Größe

Mit dem Befehl **find** können mit der Option **-size**, gefolgt von einem numerischen Wert und der Einheit, Dateien einer bestimmten Größe gesucht werden. Verwenden Sie die folgende Liste als Einheiten für die Option **-size**:

- **k** für Kilobyte
- **M** für Megabyte
- **G** für Gigabyte

Das folgende Beispiel zeigt, wie Sie nach Dateien mit einer Größe von 10 Megabyte (aufgerundet) suchen.

```
[user@host ~]$ find -size 10M
```

So suchen Sie nach Dateien mit einer Größe von *mehr als* 10 Gigabyte.

```
[user@host ~]$ find -size +10G
```

So listen Sie alle Dateien mit einer Größe von *weniger als* 10 Kilobyte auf.

```
[user@host ~]$ find -size -10k
```



### Wichtig

Mit der Option **-size** für Einheitenmodifizierer werden alle Werte zu ganzen Einheiten gerundet. So werden beispielsweise mit dem Befehl **find -size 1M** Dateien mit einer Größe kleiner als 1 MB angezeigt, da alle Dateien auf 1 MB aufgerundet werden.

## Suchen nach Dateien anhand der Änderungszeit

Mit der Option **-mmin** gefolgt von der Zeit in Minuten werden alle Dateien gesucht, deren Inhalt vor **n** Minuten geändert wurde. Der Zeitstempel der Datei wird immer abgerundet. Bei Verwendung mit Bereichen werden auch Bruchwerte unterstützt (**+n** und **-n**).

Führen Sie zum Suchen nach allen Dateien, deren Inhalt vor 120 Minuten auf **host** geändert wurde, den folgenden Befehl aus:

```
[root@host ~]# find / -mmin 120
```

Mit dem Modifizierer **+** vor der Minutenanzahl wird nach allen Dateien in **/** gesucht, die vor mehr als **n** Minuten geändert wurden. In diesem Beispiel werden Dateien aufgelistet, die vor mehr als 200 Minuten geändert wurden.

```
[root@host ~]# find / -mmin +200
```

Mit dem Modifizierer **-** wird die Suche dahingehend geändert, dass alle Dateien im Verzeichnis **/** gesucht werden, die vor weniger als **n** Minuten geändert wurden. In diesem Beispiel werden Dateien aufgelistet, die vor weniger als 150 Minuten geändert wurden.

```
[root@host ~]# find / -mmin -150
```

## Suchen nach Dateien anhand des Dateityps

Mit der Option **-type** im Befehl **find** wird der Suchbereich auf einen bestimmten Dateityp eingeschränkt. Verwenden Sie die folgende Liste, um die erforderlichen Flags zum Einschränken des Suchbereichs zu übergeben:

- **f** für normale Datei
- **d** für Verzeichnis
- **l** für Softlink
- **b** für Blockgerät

Suchen Sie nach allen Verzeichnissen im Verzeichnis **/etc** auf **host**:

```
[root@host ~]# find /etc -type d  
/etc  
/etc/tmpfiles.d  
/etc/systemd  
/etc/systemd/system  
/etc/systemd/system/getty.target.wants  
...output omitted...
```

Suchen Sie nach allen Softlinks auf **host**:

```
[root@host ~]# find / -type l
```

Erzeugen Sie eine Liste aller Blockgeräte im Verzeichnis **/dev** auf **host**:

```
[root@host ~]# find /dev -type b  
/dev/vda1  
/dev/vda
```

Mit der Option **-links** gefolgt von einer Zahl werden alle Dateien gesucht, die über eine bestimmte Anzahl von Hardlinks verfügen. Dem Wert kann ein **+**-Modifizierer vorangestellt werden, um nach Dateien mit einer höheren Anzahl als der angegebenen Anzahl von Hardlinks zu suchen. Steht der Zahl ein **--**-Modifizierer voran, wird die Suche auf alle Dateien mit einer kleineren Anzahl von Hardlinks als angegeben beschränkt.

Suchen Sie nach allen Standarddateien mit mehr als einem Hardlink auf **host**:

```
[root@host ~]# find / -type f -links +1
```



### Literaturhinweise

Manpages **locate(1)**, **updatedb(8)** und **find(1)**

## ► Angeleitete Übung

# Suchen von Dateien im System

In dieser Übung suchen Sie mithilfe der Befehle **find** und **locate** nach bestimmten Dateien in gemounteten Dateisystemen.

## Ergebnisse

Sie sollten Dateien mithilfe der Befehle **find** und **locate** suchen können.

## Bevor Sie Beginnen

Melden Sie sich als Benutzer **student** mit dem Passwort **student** bei **workstation** an.

Führen Sie auf **workstation** den Befehl **lab fs-locate start** aus. Dieser Befehl führt ein Startskript aus, das ermittelt, ob der Host **servera** im Netzwerk erreichbar ist.

```
[student@workstation ~]$ lab fs-locate start
```

- 1. Melden Sie sich mit dem Befehl **ssh** bei **servera** als Benutzer **student** an.

```
[student@workstation ~]$ ssh student@servera  
...output omitted...  
[student@servera ~]$
```

- 2. Verwenden Sie den Befehl **locate** zum Suchen nach Dateien auf **servera**.

- 2.1. Auch wenn die Datenbank **locate** jeden Tag automatisch aktualisiert wird, sollten Sie sicherstellen, dass die Datenbank auf dem neuesten Stand ist, indem Sie manuell ein Update auf **servera** starten. Verwenden Sie den Befehl **sudo updatedb** zum Aktualisieren der vom Befehl **locate** verwendeten Datenbank.

```
[student@servera ~]$ sudo updatedb  
[sudo] password for student: student  
[student@servera ~]$
```

- 2.2. Suchen Sie die Konfigurationsdatei **logrotate.conf**.

```
[student@servera ~]$ locate logrotate.conf  
/etc/logrotate.conf  
/usr/share/man/man5/logrotate.conf.5.gz
```

- 2.3. Suchen Sie die Konfigurationsdatei **networkmanager.conf**, ohne die Groß- und Kleinschreibung zu berücksichtigen.

```
[student@servera ~]$ locate -i networkmanager.conf  
/etc/NetworkManager/NetworkManager.conf  
/etc/dbus-1/system.d/org.freedesktop.NetworkManager.conf  
/usr/share/man/man5/NetworkManager.conf.5.gz
```

- 3. Führen Sie mit dem Befehl **find** Echtzeitsuchen auf **servera** gemäß den folgenden Anforderungen aus:
- Suchen Sie alle Dateien im Verzeichnis **/var/lib**, deren Besitzer der Benutzer **chrony** ist.
  - Listen Sie alle Dateien im Verzeichnis **/var** auf, die im Besitz des Benutzers **root** und der Gruppe **mail** sind.
  - Listen Sie alle Dateien im Verzeichnis **/usr/bin** auf, die eine Dateigröße von mehr als 50 KB haben.
  - Suchen Sie alle Dateien im Verzeichnis **/home/student**, die in den letzten 120 Minuten nicht geändert wurden.
  - Listen Sie alle Blockgerätedateien im Verzeichnis **/dev** auf.
- 3.1. Suchen Sie mithilfe des Befehls **find** nach allen Dateien im Verzeichnis **/var/lib**, deren Besitzer der Benutzer **chrony** ist. Verwenden Sie den Befehl **sudo**, da die Dateien im Verzeichnis **/var/lib** im Besitz von **root** sind.

```
[student@servera ~]$ sudo find /var/lib -user chrony  
[sudo] password for student: student  
/var/lib/chrony  
/var/lib/chrony/drift
```

- 3.2. Listen Sie alle Dateien im Verzeichnis **/var** auf, die im Besitz des Benutzers **root** sind und zur Gruppe **mail** gehören.

```
[student@servera ~]$ sudo find /var -user root -group mail  
/var/spool/mail
```

- 3.3. Listen Sie alle Dateien im Verzeichnis **/usr/bin** auf, die eine Dateigröße von mehr als 50 KB haben.

```
[student@servera ~]$ find /usr/bin -size +50k  
/usr/bin/iconv  
/usr/bin/locale  
/usr/bin/localeddef  
/usr/bin/cmp  
...output omitted...
```

- 3.4. Suchen Sie alle Dateien im Verzeichnis **/home/student**, die in den letzten 120 Minuten nicht geändert wurden.

```
[student@servera ~]$ find /home/student -mmin +120  
/home/student/.bash_logout  
/home/student/.bash_profile  
/home/student/.bashrc  
...output omitted...
```

3.5. Listen Sie alle Blockgerätedateien im Verzeichnis **/dev** auf.

```
[student@servera ~]$ find /dev -type b
/dev/vdd
/dev/vdc
/dev/vdb
/dev/vda3
/dev/vda2
/dev/vda1
/dev/vda
```

► 4. Beenden Sie **servera**.

```
[student@servera ~]$ exit
logout
Connection to servera closed.
[student@workstation]$
```

## Beenden

Führen Sie auf **workstation** das Skript **lab fs-locate finish** aus, um diese Übung zu beenden.

```
[student@workstation ~]$ lab fs-locate finish
```

Hiermit ist die angeleitete Übung beendet.

## ► Praktische Übung

# Zugriff auf Linux-Dateisysteme

### Leistungscheckliste

In dieser praktischen Übung mounten Sie ein lokales Dateisystem und suchen bestimmte Dateien in diesem Dateisystem.

### Ergebnisse

Es werden folgende Fähigkeiten vermittelt:

- Mounten eines Dateisystems
- Generieren eines Berichts zur Datenträgerbelegung
- Suchen nach Dateien im lokalen Dateisystem

### Bevor Sie Beginnen

Melden Sie sich als Benutzer **student** mit dem Passwort **student** bei **workstation** an.

Führen Sie auf **workstation** den Befehl **lab fs-review start** aus. Dieser Befehl führt ein Startskript aus, das ermittelt, ob der Host **serverb** im Netzwerk erreichbar ist. Das Skript erstellt auch eine Partition auf dem zweiten Laufwerk, das an **serverb** angeschlossen ist.

```
[student@workstation ~]$ lab fs-review start
```

1. Identifizieren Sie auf **serverb** als **root** die UUID für **/dev/vdb1** und mounten Sie **/dev/vdb1** anhand der UUID im Verzeichnis **/mnt/freespace**.
2. Generieren Sie einen Bericht zur Datenträgerbelegung für das Verzeichnis **/usr/share**, und speichern Sie das Ergebnis in der Datei **/mnt/freespace/results.txt**.
3. Verwenden Sie den Befehl **locate**, um alle **rsyslog.conf**-Konfigurationsdateien zu finden, und speichern Sie das Ergebnis in der Datei **/mnt/freespace/search1.txt**.
4. Speichern Sie das Suchergebnis aller Dateien im Verzeichnis **/usr/share**, die größer als 50 MB und kleiner als 100 MB sind, in der Datei **/mnt/freespace/search2.txt**.
5. Beenden Sie **serverb**.

### Bewertung

Führen Sie auf **workstation** das Skript **lab fs-review grade** aus, um den Erfolg dieser praktischen Übung zu bestätigen.

```
[student@workstation ~]$ lab fs-review grade
```

## Beenden

Führen Sie auf **workstation** das Skript **lab fs-review finish** aus, um diese Übung zu beenden.

```
[student@workstation ~]$ lab fs-review finish
```

Damit ist die praktische Übung abgeschlossen.

## ► Lösung

# Zugriff auf Linux-Dateisysteme

### Leistungscheckliste

In dieser praktischen Übung mounten Sie ein lokales Dateisystem und suchen bestimmte Dateien in diesem Dateisystem.

### Ergebnisse

Es werden folgende Fähigkeiten vermittelt:

- Mounten eines Dateisystems
- Generieren eines Berichts zur Datenträgerbelegung
- Suchen nach Dateien im lokalen Dateisystem

### Bevor Sie Beginnen

Melden Sie sich als Benutzer **student** mit dem Passwort **student** bei **workstation** an.

Führen Sie auf **workstation** den Befehl **lab fs-review start** aus. Dieser Befehl führt ein Startskript aus, das ermittelt, ob der Host **serverb** im Netzwerk erreichbar ist. Das Skript erstellt auch eine Partition auf dem zweiten Laufwerk, das an **serverb** angeschlossen ist.

```
[student@workstation ~]$ lab fs-review start
```

1. Identifizieren Sie auf **serverb** als **root** die UUID für **/dev/vdb1** und mounten Sie **/dev/vdb1** anhand der UUID im Verzeichnis **/mnt/freespace**.

- 1.1. Melden Sie sich mit dem Befehl **ssh** bei **serverb** als **student** an.

```
[student@workstation ~]$ ssh student@serverb
...output omitted...
[student@serverb ~]$
```

- 1.2. Verwenden Sie den Befehl **su -**, um zum Benutzer **root** zu wechseln.

```
[student@serverb ~]$ su -
Password: redhat
[root@serverb ~]#
```

- 1.3. Verwenden Sie den Befehl **lsblk** zur Ermittlung der UUID des Geräts **/dev/vdb1**.

```
[root@serverb ~]# lsblk -fp /dev/vdb
NAME      FSTYPE LABEL UUID                                     MOUNTPOINT
/dev/vdb
└─/dev/vdb1 xfs    a04c511a-b805-4ec2-981f-42d190fc9a65
```

- 1.4. Erstellen Sie das Verzeichnis **/mnt/freespace**.

```
[root@serverb ~]# mkdir /mnt/freespace
```

- 1.5. Mounten Sie das Gerät **/dev/vdb1** mithilfe der UUID im Verzeichnis **/mnt/freespace**.

```
[root@serverb ~]# mount UUID="a04c511a-b805-4ec2-981f-42d190fc9a65" /mnt/freespace
```

- 1.6. Stellen Sie sicher, dass das Gerät **/dev/vdb1** im Verzeichnis **/mnt/freespace** gemountet wurde.

```
[root@serverb ~]# lsblk -fp /dev/vdb1
NAME      FSTYPE LABEL UUID                                     MOUNTPOINT
/dev/vdb
└─/dev/vdb1 xfs    a04c511a-b805-4ec2-981f-42d190fc9a65 /mnt/freespace
```

2. Generieren Sie einen Bericht zur Datenträgerbelegung für das Verzeichnis **/usr/share**, und speichern Sie das Ergebnis in der Datei **/mnt/freespace/results.txt**.

```
[root@serverb ~]# du /usr/share > /mnt/freespace/results.txt
```

3. Verwenden Sie den Befehl **locate**, um alle **rsyslog.conf**-Konfigurationsdateien zu finden, und speichern Sie das Ergebnis in der Datei **/mnt/freespace/search1.txt**.

- 3.1. Verwenden Sie den Befehl **updatedb**, um die von **locate** verwendete Datenbank zu aktualisieren.

```
[root@serverb ~]# updatedb
```

- 3.2. Suchen Sie die **rsyslog.conf**-Konfigurationsdateien, und speichern Sie das Ergebnis in der Datei **/mnt/freespace/search1.txt**.

```
[root@serverb ~]# locate rsyslog.conf > /mnt/freespace/search1.txt
```

4. Speichern Sie das Suchergebnis aller Dateien im Verzeichnis **/usr/share**, die größer als 50 MB und kleiner als 100 MB sind, in der Datei **/mnt/freespace/search2.txt**.

```
[root@serverb ~]# find /usr/share -size +50M -size -100M > \
/mnt/freespace/search2.txt
```

5. Beenden Sie **serverb**.

```
[root@serverb ~]$ exit  
logout  
[student@serverb ~]$ exit  
logout  
Connection to serverb closed.  
[student@workstation]$
```

## Bewertung

Führen Sie auf **workstation** das Skript **lab fs-review grade** aus, um den Erfolg dieser praktischen Übung zu bestätigen.

```
[student@workstation ~]$ lab fs-review grade
```

## Beenden

Führen Sie auf **workstation** das Skript **lab fs-review finish** aus, um diese Übung zu beenden.

```
[student@workstation ~]$ lab fs-review finish
```

Damit ist die praktische Übung abgeschlossen.

# Zusammenfassung

---

In diesem Kapitel erhielten Sie folgende Informationen:

- Speichergeräte werden durch einen bestimmten Dateityp mit der Bezeichnung "Blockgerät" dargestellt.
- Der Befehl **df** gibt die insgesamt vorhandene Datenträgerkapazität, die belegte Datenträgerkapazität und die freie Datenträgerkapazität für alle gemounteten standardmäßigen Dateisysteme aus.
- Mit dem Befehl **mount** kann der Benutzer **root** ein Dateisystem manuell mounten.
- Alle Prozesse müssen den Zugriff auf den Mount-Punkt beenden, um das Gerät erfolgreich zu ummounten.
- Die Wechseldatenträger werden bei Verwendung der grafischen Umgebung im Verzeichnis **/run/media** gemountet.
- Mit dem Befehl **find** wird eine Echtzeitsuche in lokalen Dateisystemen ausgeführt, um Dateien anhand von Suchkriterien zu finden.

## Kapitel 16

# Analysieren von Servern und Erhalten von Unterstützung

### Ziel

Untersuchen und Lösen von Problemen in der webbasierten Managementoberfläche, Erhalten von Unterstützung bei der Lösung von Problemen von Red Hat.

### Ziele

- Aktivieren der Web Console-Managementoberfläche zur Remote-Verwaltung und -Überwachung der Leistung eines Red Hat Enterprise Linux-Servers.
- Beschreiben wichtiger Ressourcen, die über das Red Hat Customer Portal verfügbar sind, und Suchen nach Informationen in der Dokumentation und der Knowledgebase von Red Hat.
- Überprüfen von Servern auf Probleme, Beheben oder Lösen der Probleme und Bestätigen der Lösung durch Red Hat Insights.

### Abschnitte

- Analysieren und Verwalten von Remote-Servern (mit angeleiteter Übung)
- Erhalten von Hilfe im Red Hat Customer Portal (mit angeleiteter Übung)
- Erkennen und Lösen von Problemen mit Red Hat Insights (mit Test)

# Analysieren und Verwalten von Remote-Servern

---

## Ziele

Nach Abschluss dieses Abschnitts sollten Sie in der Lage sein, die Web Console-Managementoberfläche zur Remote-Verwaltung und -Überwachung der Leistung eines Red Hat Enterprise Linux-Servers zu aktivieren.

## Beschreiben von Web Console

Web Console ist eine webbasierte Management-Benutzeroberfläche für Red Hat Enterprise Linux 8, die für die Verwaltung und Überwachung Ihrer Server entwickelt wurde. Sie basiert auf dem Open Source-Service „Cockpit“.

Mit Web Console können Sie Systemprotokolle überwachen und Diagramme der Systemleistung anzeigen. Darüber hinaus können Sie Ihren Webbrowser verwenden, um Einstellungen mithilfe von grafischen Tools in der Web Console-Benutzeroberfläche zu ändern; dazu gehört auch eine voll funktionsfähige interaktive Terminalsitzung.

## Aktivieren von Web Console

Web Console wird von Red Hat Enterprise Linux 8 standardmäßig in allen Installationsvarianten installiert, mit Ausnahme der minimalen Installation. Verwenden Sie den folgenden Befehl, um Web Console zu installieren:

```
[user@host ~]$ sudo yum install cockpit
```

Aktivieren und starten Sie den Service **cockpit.socket**, der einen Webserver ausführt. Dieser Schritt ist erforderlich, wenn Sie die Verbindung zum System über die Web-Benutzeroberfläche herstellen müssen.

```
[user@host ~]$ sudo systemctl enable --now cockpit.socket
Created symlink /etc/systemd/system/systemd.wants/cockpit.socket → /usr/
lib/systemd/system/cockpit.socket.
```

Wenn Sie ein benutzerdefiniertes Firewall-Profil verwenden, müssen Sie den Service **cockpit** zu **firewalld** hinzufügen, um den Port 9090 in der Firewall zu öffnen:

```
[user@host ~]$ sudo firewall-cmd --add-service=cockpit --permanent
success
[user@host ~]$ sudo firewall-cmd --reload
success
```

## Anmelden bei Web Console

Web Console stellt einen eigenen Webserver zur Verfügung. Starten Sie Firefox, um sich bei Web Console anzumelden. Sie können sich mit dem Benutzernamen und dem Passwort eines beliebigen lokalen Benutzerkontos beim System anmelden, einschließlich des Benutzers **root**.

Öffnen Sie `https://servername:9090` in Ihrem Webbrowser, wobei `servername` der Hostname oder die IP-Adresse Ihres Servers ist. Die Verbindung wird durch eine TLS-Sitzung geschützt. Das System wird standardmäßig mit einem selbstsignierten TLS-Zertifikat installiert und wenn Sie Ihren Webbrowser zum ersten Mal verbinden, wird wahrscheinlich eine Sicherheitswarnung angezeigt. Die Manpage `cockpit-ws(8)` enthält Anweisungen zum Ersetzen des TLS-Zertifikats durch ein ordnungsgemäß signiertes Zertifikat.

Geben Sie im Anmeldebildschirm Ihren Benutzernamen und Ihr Passwort ein.

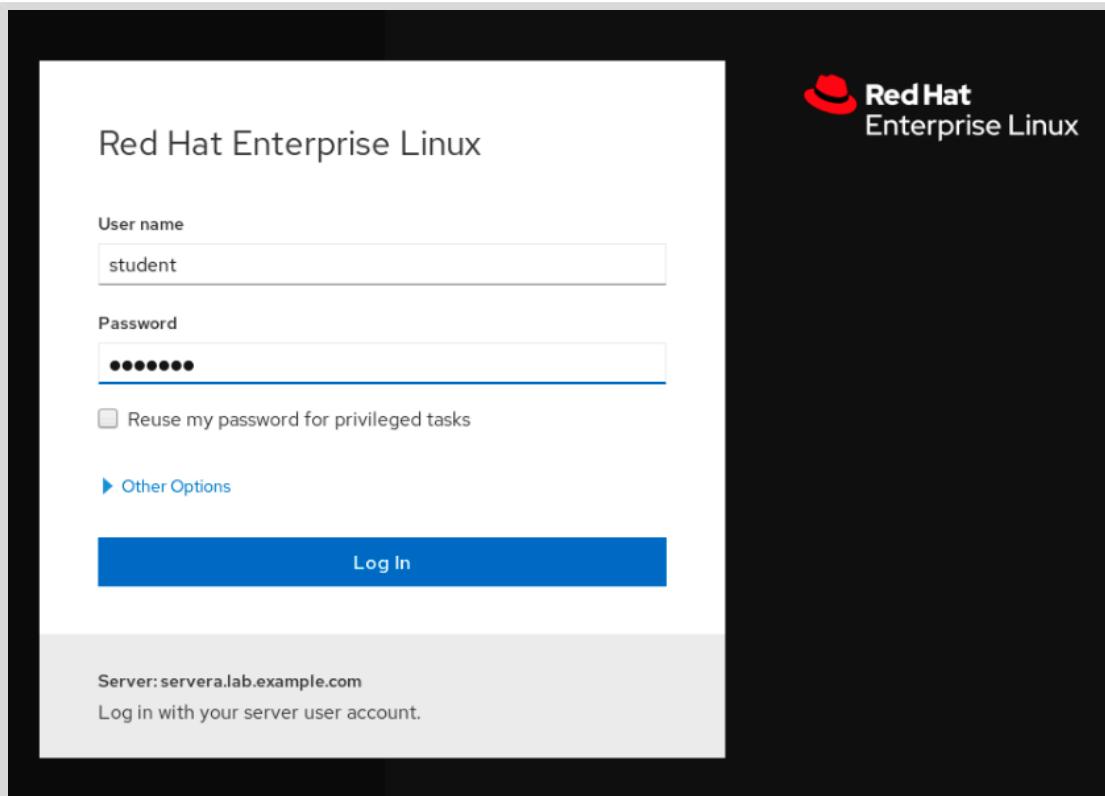


Abbildung 16.1: Der Web Console-Anmeldebildschirm

Klicken Sie optional auf die Option **Reuse my password for privileged tasks**. Auf diese Weise können Sie Befehle mit sudo-Berechtigungen ausführen. So können Sie beispielsweise Systeminformationen ändern oder neue Benutzerkonten konfigurieren.

Klicken Sie auf **Log In**.

Web Console zeigt den Benutzernamen auf der rechten Seite der Titelleiste an. Wenn Sie die Option **Reuse my password for privileged tasks** aktivieren, wird links neben dem Benutzernamen das Symbol **Privileged** angezeigt.

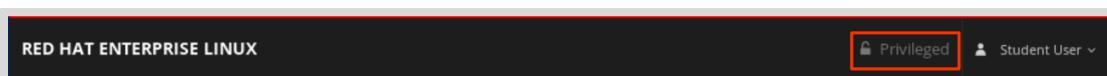
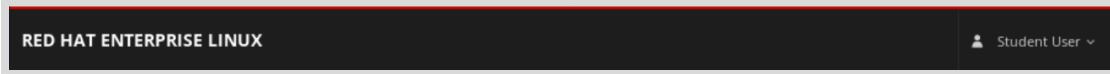


Abbildung 16.2: Titelleiste eines berechtigten Benutzers

Wenn Sie als nicht berechtigter Benutzer angemeldet sind, wird das Symbol **Privileged** nicht angezeigt.



## Ändern von Passwörtern

Privilegierte und unprivilegierte Benutzer können ihre eigenen Passwörter ändern, während sie bei Web Console angemeldet sind. Klicken Sie in der Navigationsleiste auf **Accounts**. Klicken Sie auf die Bezeichnung Ihres Benutzerkontos, um die Kontodetailseite zu öffnen.

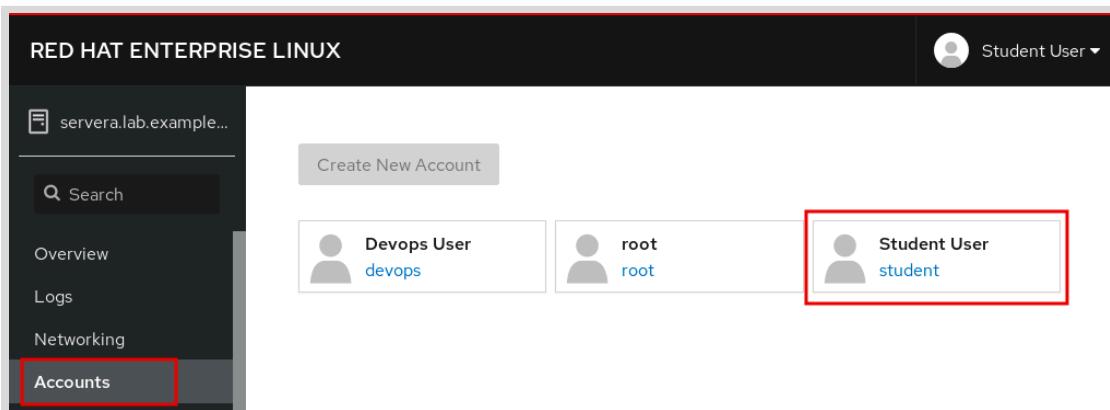


Abbildung 16.4: Anzeigen von Benutzerkonten

Als nicht berechtigter Benutzer können Sie lediglich Ihr Passwort festlegen oder zurücksetzen und öffentliche SSH-Schlüssel verwalten. Um Ihr Passwort festzulegen oder zurückzusetzen, klicken Sie auf **Set Password**.

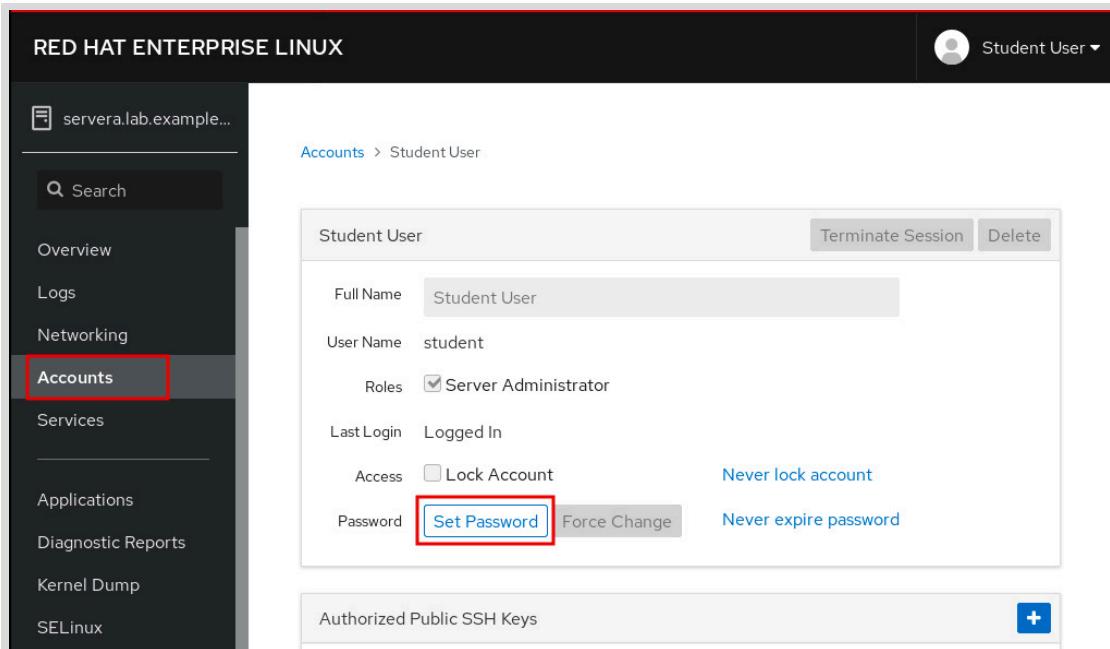


Abbildung 16.5: Benutzerkontodetails

Geben Sie Ihre Daten in die Felder **Old Password**, **New Password** und **Confirm New Password** ein. Klicken Sie auf **Set**, um das neue Passwort zu aktivieren.

The screenshot shows a 'Set Password' dialog box. It contains three input fields: 'Old Password', 'New Password', and 'Confirm New Password', all of which are highlighted with a red box. Below these fields is a horizontal progress bar. At the bottom right of the dialog are two buttons: 'Cancel' and 'Set', with 'Set' being highlighted with a red box.

Abbildung 16.6: Festlegen und Zurücksetzen von Passwörtern

## Fehlerbehebung mit Web Console

Web Console ist ein leistungsfähiges Tool zur Fehlerbehebung. Sie können grundlegende Systemstatistiken in Echtzeit überwachen, Systemprotokolle prüfen und schnell zu einer Terminalsitzung in Web Console wechseln, um weitere Informationen von der Befehlszeilenschnittstelle zu sammeln.

### Überwachen von Systemstatistiken in Echtzeit

Klicken Sie in der Navigationsleiste auf **Overview**, um Informationen über das System anzuzeigen, z. B. den Hardwaretyp, das Betriebssystem, den Hostnamen usw. Wenn Sie als nicht berechtigter Benutzer angemeldet sind, können Sie zwar alle Informationen sehen, dürfen jedoch keine Werte ändern. In der folgenden Abbildung wird ein Teil der **Übersichtsseite** angezeigt.

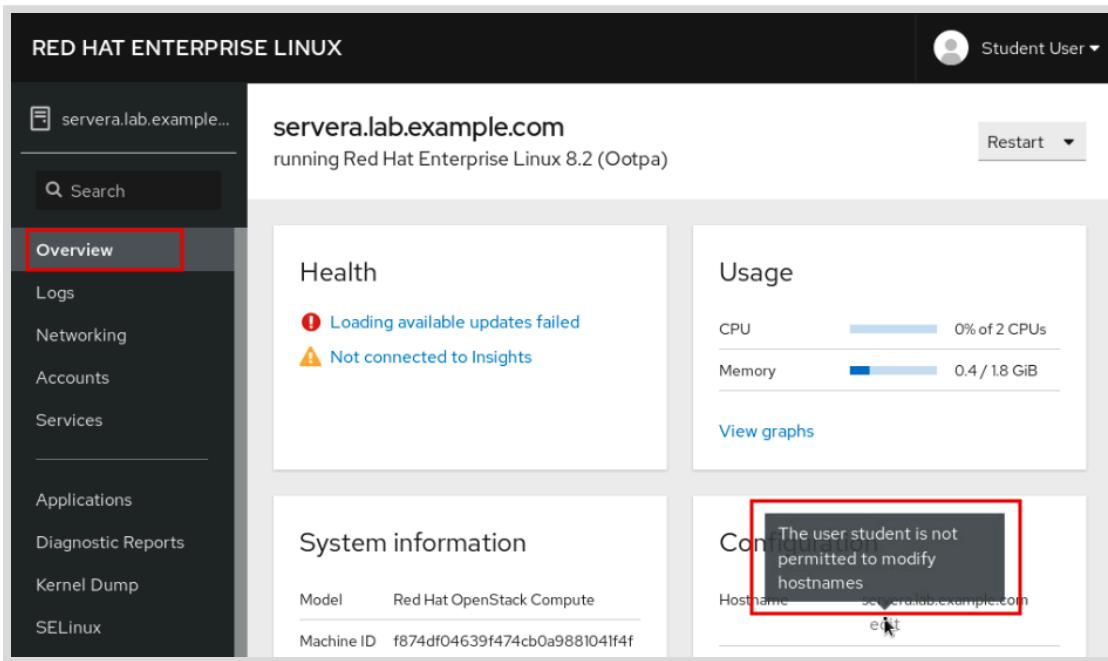


Abbildung 16.7: Übersichtsseite eines nicht privilegierten Benutzers

Klicken Sie auf der Seite **Overview** auf **View graphs**, um Diagramme der aktuellen Systemleistung für CPU-Aktivität, Speicherauslastung, Disk-E/A und Netzwerknutzung anzuzeigen.

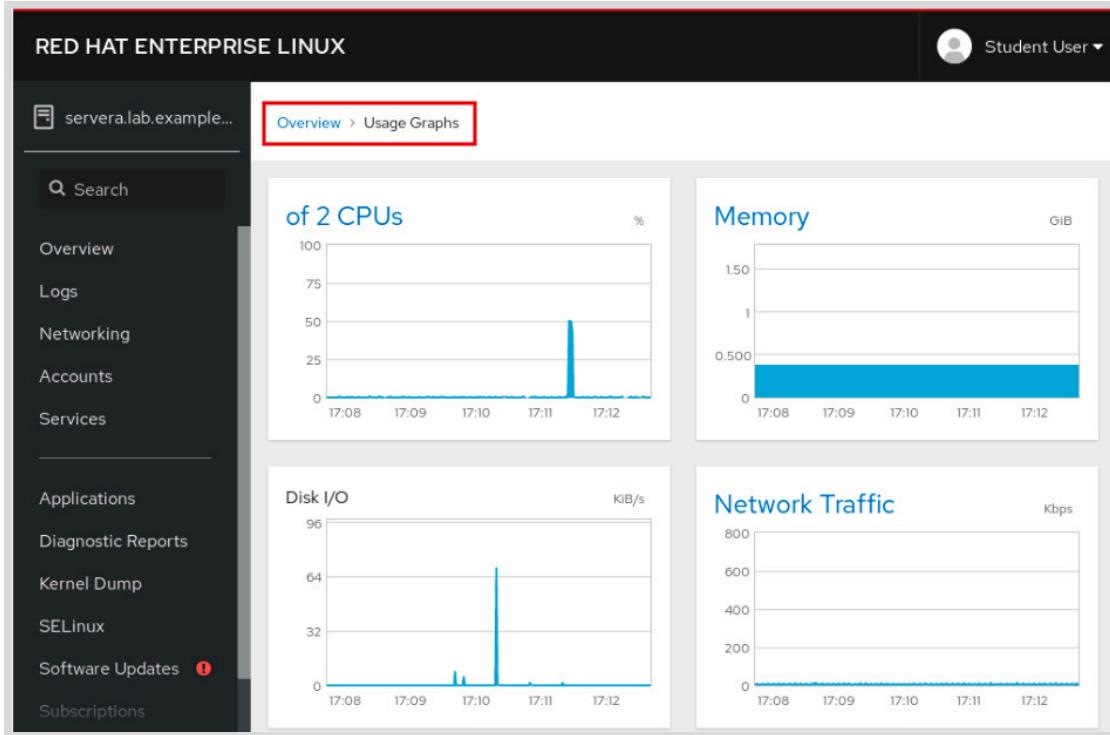


Abbildung 16.8: System-Leistungskennzahlen eines nicht berechtigten Benutzers

## Prüfen und Filtern von Syslog-Ereignissen

Über **Logs** in der Navigationsleiste können Sie auf Analysetools für die Systemprotokolle zugreifen. Sie können die Menüs auf der Seite verwenden, um Protokollmeldungen basierend auf einem Protokollierungszeitraum und/oder einem Schweregrad zu filtern. Web Console verwendet das aktuelle Datum als Standard, aber Sie können auf das Menü für das Datum klicken und einen Datumsbereich auswählen. Entsprechend enthält das Menü **Severity** Optionen von **Everything** bis hin zu spezifischeren Schweregraden wie **Alert and above**, **Debug and above** usw.

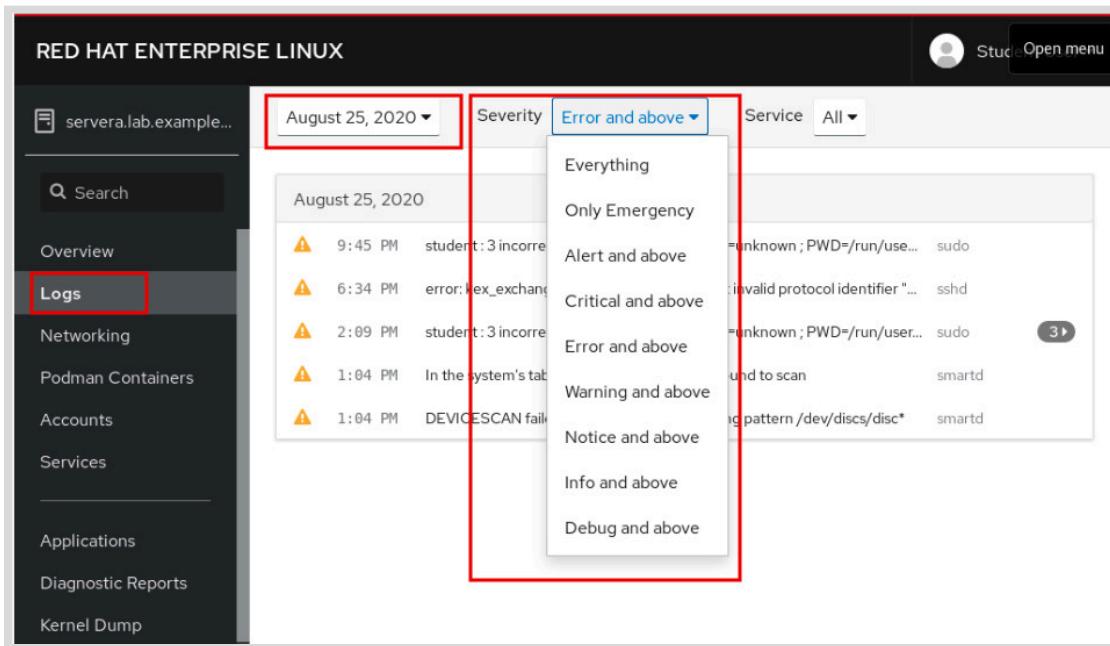


Abbildung 16.9: Auswahlmöglichkeiten für den Protokollschweregrad

## Kapitel 16 | Analysieren von Servern und Erhalten von Unterstützung

Klicken Sie auf eine Zeile, um Details zum Protokollbericht anzuzeigen. Im folgenden Beispiel enthält die erste Zeile einen Eintrag zu einer **sudo**-Protokollmeldung.

The screenshot shows the Cockpit web interface for Red Hat Enterprise Linux. The left sidebar has a dark theme with white text. The 'Logs' option is highlighted with a red box. The main area displays log entries for August 26, 2020. One entry is selected, and its content is shown in a larger box: "student : 3 incorrect password attempts; TTY=unknown; PWD=/run/user/1000; USER=root; COMMAND=/bin/cockpit-bridge --privileged". The word "sudo" in this line is also highlighted with a red box.

Abbildung 16.10: Auswahl eines Protokolleintrags

Das folgende Beispiel zeigt die Details, die angezeigt werden, wenn Sie auf die Zeile **sudo** klicken. Zu den Details des Berichts gehören der ausgewählte Protokolleintrag (**sudo**), Datum, Uhrzeit, Priorität und Syslog-Komponente des Protokolleintrags, der Hostname des Systems, das die Protokollmeldung ausgegeben hat, und vieles mehr.

The screenshot shows the Cockpit web interface for Red Hat Enterprise Linux. The left sidebar has a dark theme with white text. The 'Logs' option is highlighted with a red box. The main area shows a detailed view of a log entry for 'student'. The entry is titled 'Entry at 2020-08-26 07:41:47'. The log line is: "student : 3 incorrect password attempts; TTY=unknown; PWD=/run/user/1000; USER=root; COMMAND=/bin/cockpit-bridge --privileged". Below this, various log fields are listed: PRIORITY (1), SYSLOG\_FACILITY (10), SYSLOG\_IDENTIFIER (sudo), \_AUDIT\_LOGINUID (1000), \_AUDIT\_SESSION (3), \_BOOT\_ID (d8e459d758dc41c099c917b21e9e51ed), \_CAP\_EFFECTIVE (3fffffff), \_COMM (sudo), \_GID (1000), \_HOSTNAME (servera.lab.example.com), and \_MACHINE\_ID (f874df04639f474cb0a9881041f4f7d4). The word "sudo" in the log line and in the field names is highlighted with a red box.

Abbildung 16.11: Details eines Protokolleintrags

## Ausführen von Befehlen in einer Terminalsitzung

Über **Terminal** in der Navigationsleiste können Sie auf eine voll funktionsfähige Terminalsitzung innerhalb der Web Console-Benutzeroberfläche zugreifen. Auf diese Weise können Sie beliebige Befehle ausführen, um das System zu verwalten, mit ihm zu arbeiten und Aufgaben auszuführen, die von den anderen von Web Console bereitgestellten Tools nicht unterstützt werden.

Die folgende Abbildung zeigt Beispiele für gängige Befehle, die zum Sammeln zusätzlicher Informationen verwendet werden. Indem Sie den Inhalt des Verzeichnisses **/var/log** auflisten, werden Ihnen Protokolldateien in Erinnerung gebracht, die wertvolle Informationen enthalten könnten. Der Befehl **id** liefert kurze Informationen wie z. B. die Gruppenmitgliedschaft, die ggf. bei der Behebung von Problemen im Zusammenhang mit Dateizugriffsbeschränkungen helfen können. Der Befehl **ps au** bietet eine schnelle Übersicht über die im Terminal ausgeführten Prozesse und den mit dem Prozess verbundenen Benutzer.

```
[student@servera ~]$ ls /var/log
anaconda          dnf.librepo.log      private
audit              dnf.librepo.log-20200826  qemu-ga
boot.log          dnf.log             rhsm
boot.log-20200820 dnf.rpm.log       samba
boot.log-20200821 firewalld          secure
boot.log-20200824 hawkey.log        secure-20200826
boot.log-20200825 hawkey.log-20200826 spooler
btmp              insights-client    spooler-20200826
chrony            lastlog           sssd
cloud-init.log    maillog           tuned
cloud-init-output.log maillog-20200826 wtmp
cron              messages          wtmp
cron-20200826     messages-20200826
[student@servera ~]$ id
uid=1000(student) gid=1000(student) groups=1000(student),10(wheel) context=unconfined_u:unconfined_r:unconfined_t:s0-s0:c0.c1023
[student@servera ~]$ ps au
USER      PID %CPU %MEM   VSZ   RSS TTY      STAT START   TIME COMMAND
root      1085  0.0  0.1 225432  1952  ttyS0   Ss+  07:27  0:00 /sbin/agetty
root      1086  0.0  0.0 225792  1728  tty1    Ss+  07:27  0:00 /sbin/agetty
student   2700  0.0  0.2 234104  5180  pts/0    Ss   07:48  0:00 /bin/bash
student   2723  0.0  0.2 267612  4020  pts/0    R+   07:48  0:00 ps au
[student@servera ~]$
```

Abbildung 16.12: Fehlerbehebung in Terminalsitzung für nicht berechtigten Benutzer

## Erstellen von Diagnoseberichten

Ein Diagnosebericht ist eine Sammlung von Konfigurationsdetails, Systeminformationen und Diagnoseinformationen von einem Red Hat Enterprise Linux-System. Die im vollständigen Bericht gesammelten Daten umfassen Systemprotokolle und Debugging-Informationen, die zur Problembearbeitung verwendet werden können.

Melden Sie sich als berechtigter Benutzer bei Web Console an. Klicken Sie in der Navigationsleiste auf **Diagnostic Reports**, um die Seite zu öffnen, auf der diese Berichte erstellt werden. Klicken Sie auf **Create Report**, um einen neuen Diagnosebericht zu generieren.

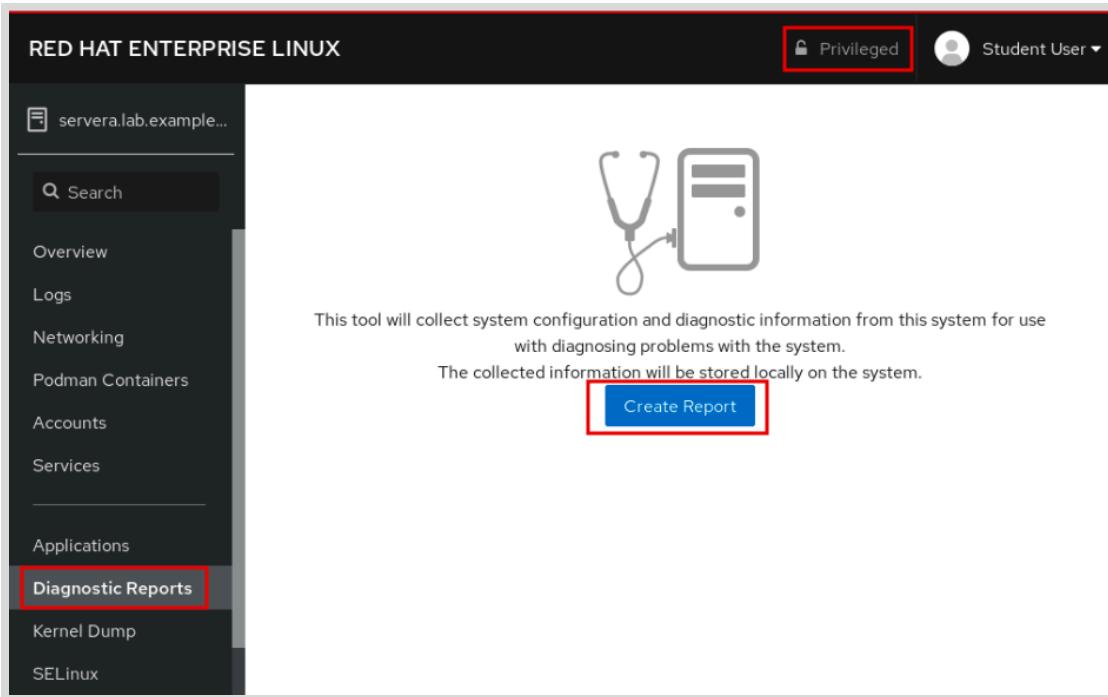


Abbildung 16.13: Erstellen eines Diagnoseberichts

In der Benutzeroberfläche wird **Done!** angezeigt, wenn der Bericht fertiggestellt ist. Klicken Sie auf **Download report**, um den Bericht zu speichern.

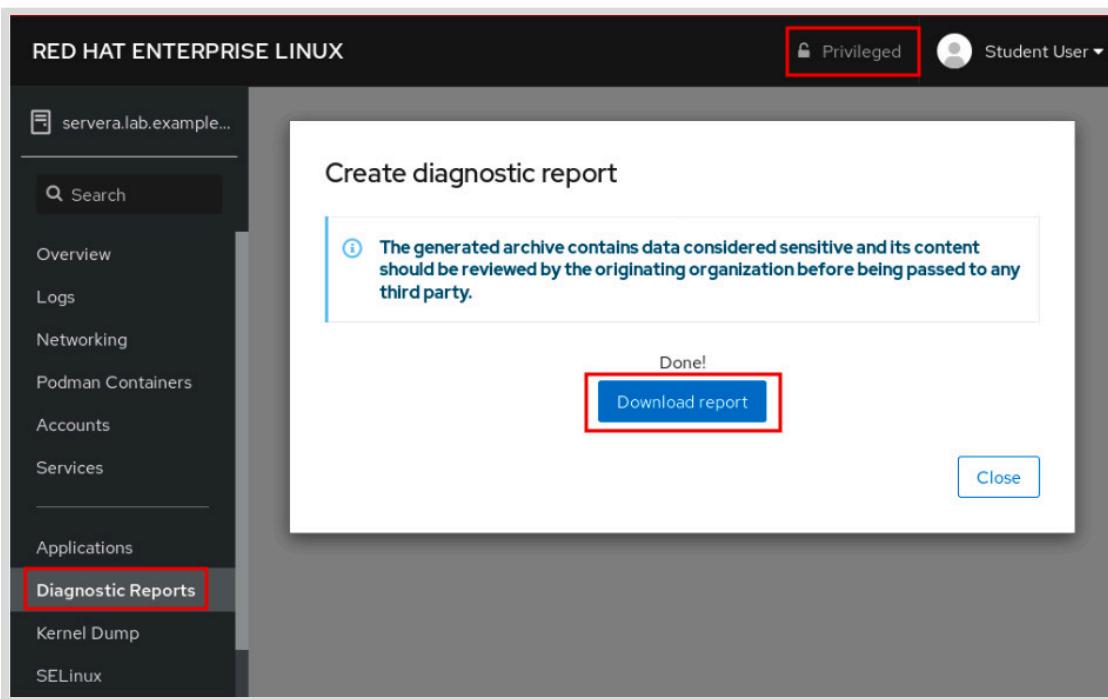


Abbildung 16.14: Herunterladen eines fertigen Berichts

Klicken Sie auf **Save File**, um die Datei zu speichern und den Vorgang abzuschließen.

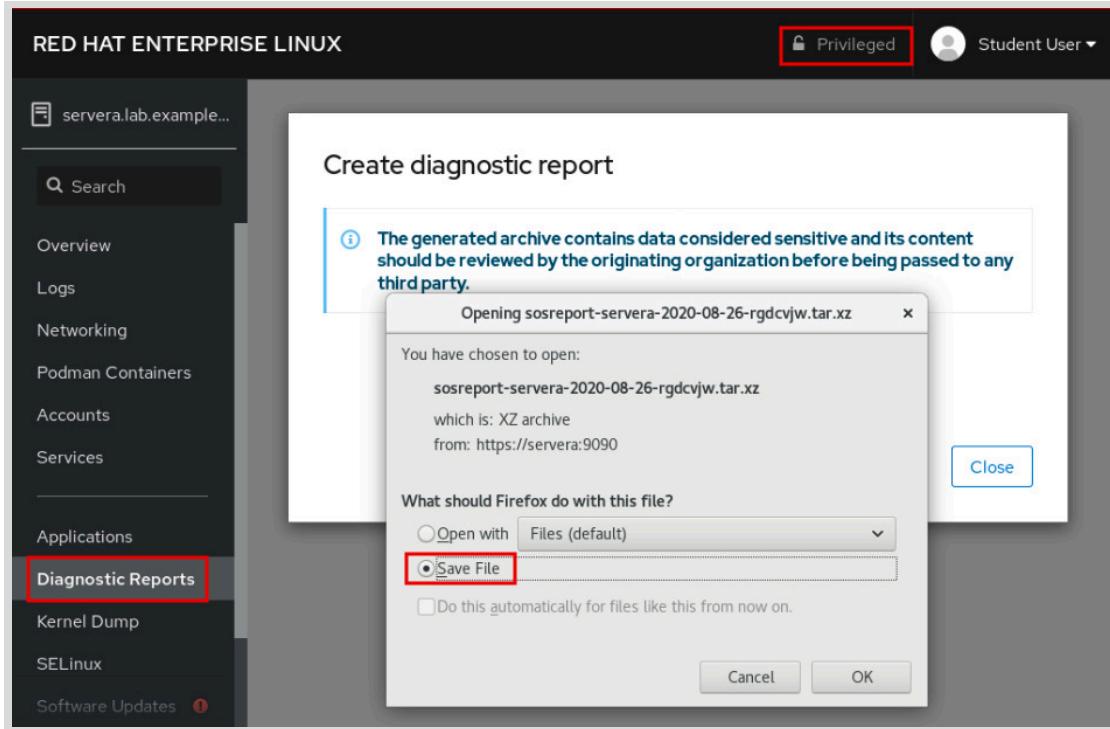


Abbildung 16.15: Speichern eines Diagnoseberichts

Der fertige Bericht wird im Verzeichnis **Downloads** auf dem System gespeichert, das den Webbrowser hostet, mit dem auf Web Console zugegriffen wird. In diesem Beispiel ist der Host **workstation**.

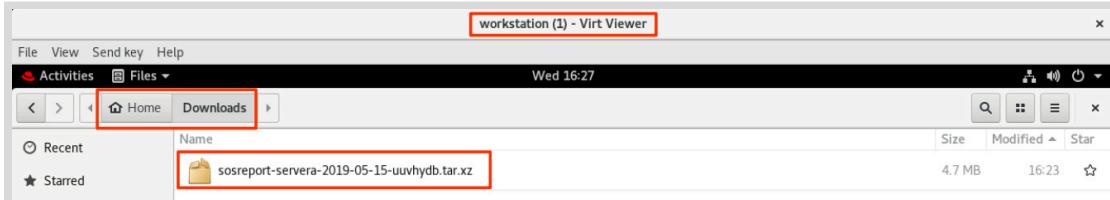


Abbildung 16.16: Zugreifen auf einen fertigen Bericht

## Verwalten von Systemservices mit Web Console

Als berechtigter Benutzer können Sie in Web Console Systemservices beenden, starten, aktivieren und neu starten. Darüber hinaus können Sie Netzwerkschnittstellen konfigurieren, Firewall-Services konfigurieren, Benutzerkonten verwalten und vieles mehr. Die folgenden Abbildungen zeigen gängige Beispiele für die Verwendung der Managementtools von Web Console.

### Systemenergieoptionen

Mit Web Console können Sie das System neu starten oder herunterfahren. Melden Sie sich als berechtigter Benutzer bei Web Console an. Klicken Sie in der Navigationsleiste auf **Overview**, um auf die Systemenergioptionen zuzugreifen.

Wählen Sie die gewünschte Option im Menü oben rechts aus, um ein System entweder neu zu starten oder herunterzufahren.

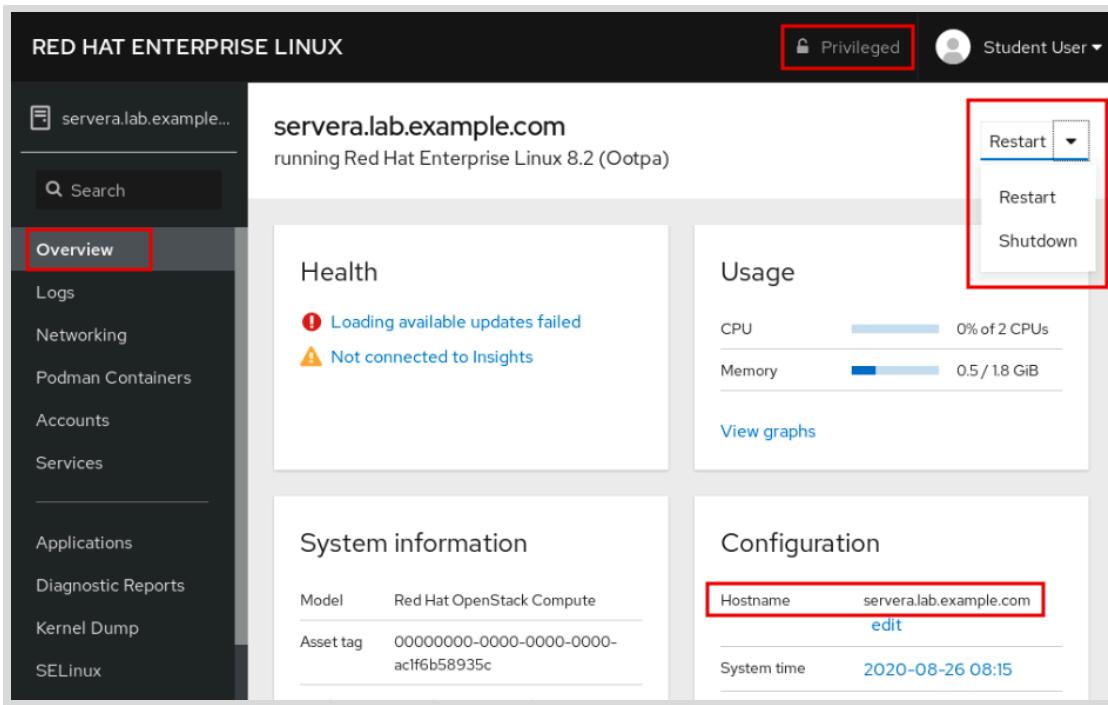


Abbildung 16.17: Systemenergieoptionen

## Steuern der aktiven Systemservices

Sie können Services mit grafischen Tools in Web Console starten, aktivieren, deaktivieren und beenden. Klicken Sie in der Navigationsleiste auf **Services**, um auf die Services-Startseite von Web Console zuzugreifen. Um Services zu verwalten, klicken Sie oben auf der Services-Startseite auf **System Services**. Suchen Sie über die Suchleiste oder scrollen Sie durch die Seite, um den Service auszuwählen, den Sie verwalten möchten.

Wählen Sie im folgenden Beispiel die Zeile **chronyd.service** aus, um die Servicemanagementseite zu öffnen.

The screenshot shows the Red Hat Enterprise Linux system services interface. The left sidebar has a 'Services' tab selected, which is highlighted with a red box. The main area shows a table of services with columns: Name, Description, Status, and Type. The 'chronyd' service is highlighted with a red box in the table.

| Service            | Description                                  | Status           | Type     |
|--------------------|----------------------------------------------|------------------|----------|
| arp-ethers         | Load static arp entries                      | inactive (dead)  | Disabled |
| atd                | Job spooling tools                           | active (running) | Enabled  |
| auditd             | Security Auditing Service                    | active (running) | Enabled  |
| auth-rpcgss-module | Kernel Module supporting RPCSEC_GSS          | inactive (dead)  | Static   |
| autovt@            | autovt@.service Template                     |                  |          |
| blk-availability   | Availability of block devices                | inactive (dead)  | Disabled |
| chrony-dnssrv@     | chrony-dnssrv@.service Template              |                  |          |
| chrony-wait        | Wait for chrony to synchronize system clock  | inactive (dead)  | Disabled |
| <b>chronyd</b>     | <b>NTP client/server</b>                     | active (running) | Enabled  |
| cloud-config       | Apply the settings specified in cloud-config | inactive (dead)  | Disabled |

Abbildung 16.18: Services: Startansicht

Klicken Sie auf **Stop**, **Restart** bzw. **Disallow running (mask)**, um den Service wie gewünscht zu verwalten. In dieser Ansicht wird der Service bereits ausgeführt. Weitere Informationen zum Service erhalten Sie durch Klicken auf einen der hervorgehobenen Links oder indem Sie durch die Serviceprotokolle scrollen, die unterhalb des Servicemanagementschnitts angezeigt werden.

The screenshot shows the Red Hat Enterprise Linux services details interface for the 'chronyd.service'. The left sidebar has a 'Services' tab selected. The main area shows detailed information for the 'chronyd.service'. A context menu is open over the service status button, with options: 'Restart', 'Stop', and 'Disallow running (mask)'. The 'Service Logs' button is also visible at the bottom.

**Services > chronyd.service**

**NTP client/server**

Status: Running (Active since August 2021)  
Automatically starts

Path: /usr/lib/systemd/system/chronyd.service

Requires: .mount, system.slice, sysinit.target

Wanted By: multi-user.target

Conflicts: systemd-timesyncd.service, shutdown.target, ntpd.service

Before: shutdown.target, multi-user.target

After: basic.target, tmp.mount, system.slice, sntp.service, .mount, sysinit.target, ntpd.service, systemd-tmpfiles-setup.service, systemd-journald.socket, ntpdate.service

Service Logs

Abbildung 16.19: Services: Servicedetails und Management-Benutzeroberfläche

## Konfigurieren von Netzwerkschnittstellen und der Firewall

Klicken Sie in der Navigationsleiste auf **Networking**, um die Firewall-Regeln und Netzwerkschnittstellen zu verwalten. Das folgende Beispiel zeigt, wie Sie Informationen zu Netzwerkschnittstellen sammeln und verwalten.

The screenshot shows the Red Hat Enterprise Linux interface for managing networking. On the left, a sidebar lists options like Overview, Logs, Networking (which is selected and highlighted with a red box), Podman Containers, Accounts, Services, Applications, Diagnostic Reports, Kernel Dump, and SELinux. The main area displays two line graphs: 'Sending' (Kbps) and 'Receiving' (Kbps) over time (08:28 to 08:32). Below the graphs, the 'Firewall' section shows '1 Active Zone' with a toggle switch set to 'On'. The 'Interfaces' section lists 'eth0' with IP address '172.25.250.10/24, 172.25.250.99/24', sending speed '2.53 Kbps', and receiving speed '1.06 Kbps'. At the bottom is a 'Networking Logs' section. A red box highlights the 'Networking' button in the sidebar and the 'Interfaces' section.

Abbildung 16.20: Networking: Startansicht

Klicken Sie im Abschnitt **Interfaces** auf den Namen der gewünschten Schnittstelle, um auf die Managementseite zuzugreifen. In diesem Beispiel ist die Schnittstelle **eth0** ausgewählt.

This screenshot shows the 'Interfaces' section of the networking interface. It includes a header row with columns for 'Name', 'IP Address', 'Sending', and 'Receiving'. Below is a table entry for 'eth0' with the IP address '172.25.250.10/24, 172.25.250.100/24' and speeds of '3.84 Kbps' and '1.58 Kbps'. A red box highlights the 'eth0' entry in the table.

Abbildung 16.21: Networking: Interfaces

Im oberen Teil der Managementseite wird die Netzwerddatenverkehrsaktivität für das ausgewählte Gerät angezeigt. Scrollen Sie nach unten, um Konfigurationseinstellungen und Managementoptionen anzuzeigen.

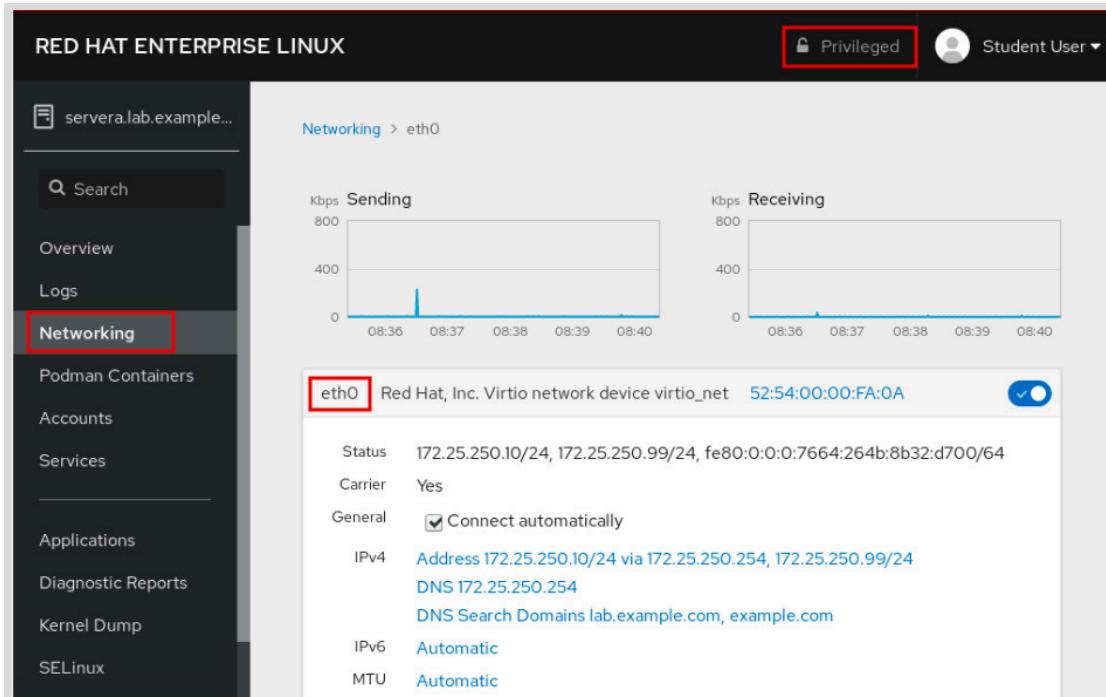


Abbildung 16.22: Networking: Schnittstellendetails

Klicken Sie auf die hervorgehobenen Links für die gewünschte Konfiguration, um Konfigurationsoptionen zu ändern oder zu einer Schnittstelle hinzuzufügen. In diesem Beispiel werden für den Link **IPv4** eine einzelne IP-Adresse und Netzmaske, **172.25.250.10/24**, für die Netzwerkschnittstelle **eth0** angezeigt. Um der Netzwerkschnittstelle **eth0** eine zusätzliche IP-Adresse hinzuzufügen, klicken Sie auf den hervorgehobenen Link.



Abbildung 16.23: Networking: Konfigurationsabschnitt für eth0

Klicken Sie auf das Pluszeichen (+) auf der rechten Seite der Listenauswahl **Manual**, um eine zusätzliche IP-Adresse hinzuzufügen. Geben Sie eine IP-Adresse und eine Netzwerkmaske in die entsprechenden Felder ein. Klicken Sie auf **Apply**, um die neuen Einstellungen zu aktivieren.

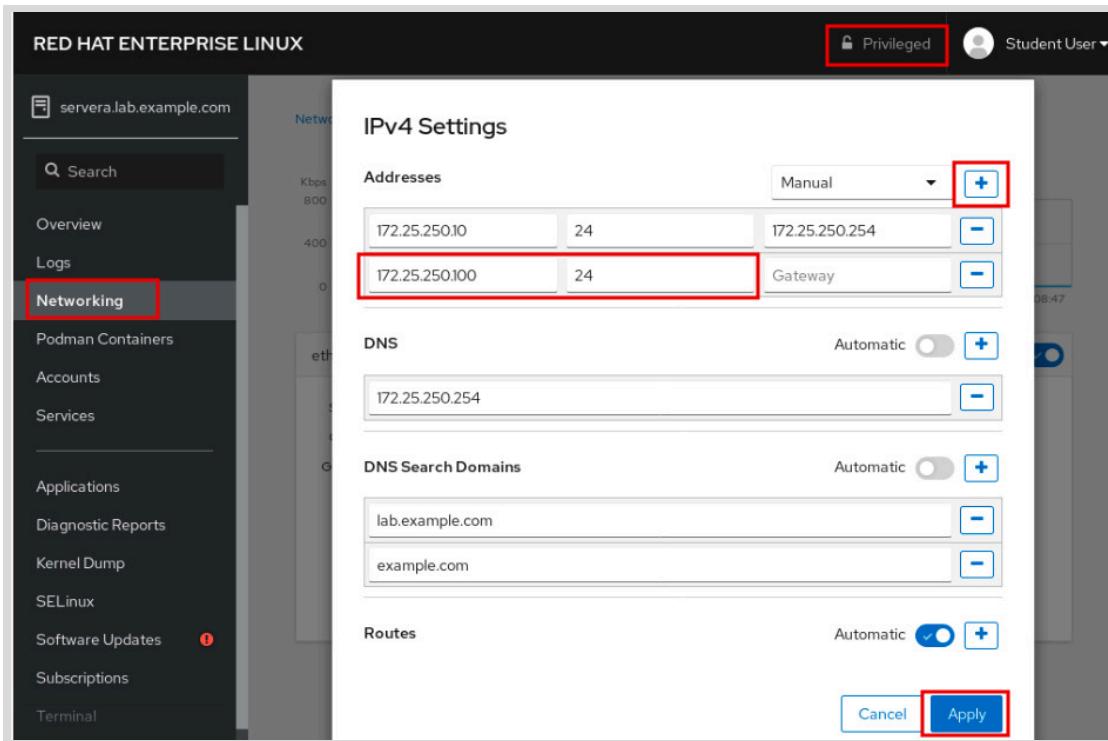


Abbildung 16.24: Hinzufügen einer IP-Adresse zu einer vorhandenen Schnittstelle

Daraufhin wird automatisch wieder die Managementseite der Schnittstelle angezeigt, auf der Sie die neue IP-Adresse bestätigen können.

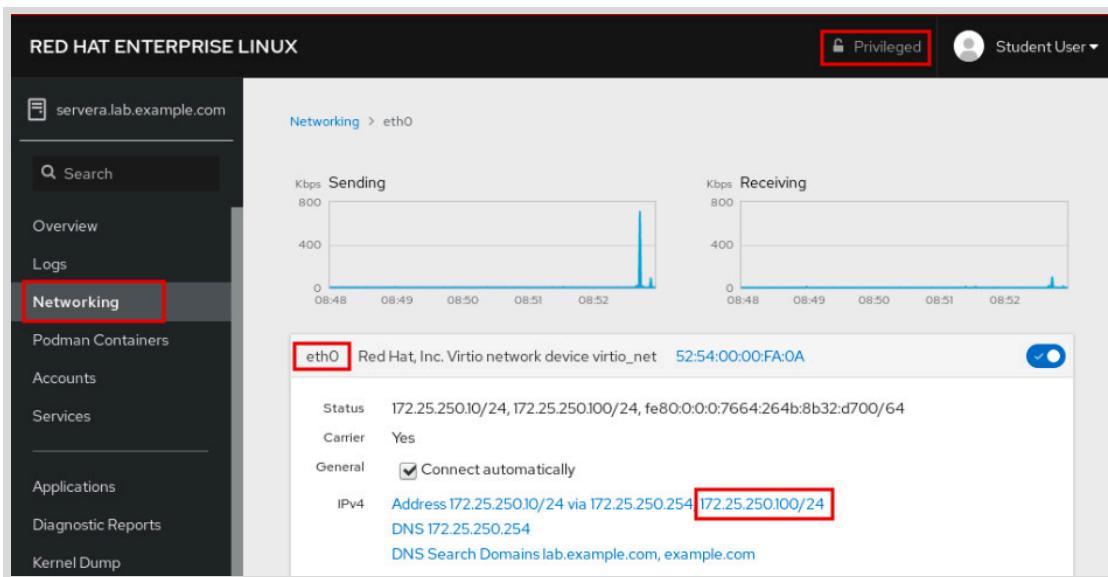


Abbildung 16.25: Bestätigen der neuen IP-Adresse

## Verwalten von Benutzerkonten

Als berechtigter Benutzer können Sie neue Benutzerkonten in Web Console erstellen. Klicken Sie in der Navigationsleiste auf **Accounts**, um vorhandene Benutzerkonten anzuzeigen. Klicken Sie auf **Create New Account**, um die Managementseite für Benutzerkonten zu öffnen.

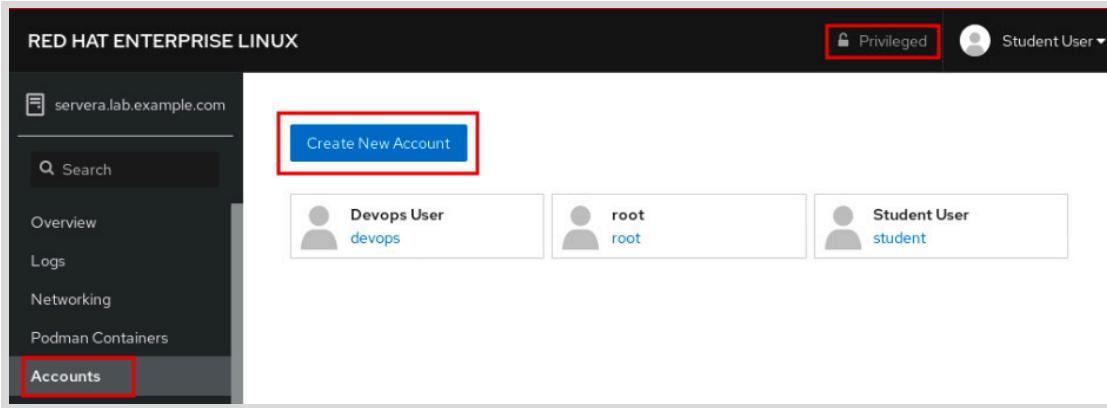


Abbildung 16.26: Vorhandene Benutzerkonten

Geben Sie die Informationen für das neue Benutzerkonto ein und klicken Sie dann auf **Create**.

This is a 'Create New Account' dialog box. It contains fields for 'Full Name' (New User), 'User Name' (nuser), 'Password' (a series of black dots), 'Confirm' (a series of black dots), and an 'Access' section with a 'Lock Account' checkbox (unchecked). At the bottom are 'Cancel' and 'Create' buttons, with 'Create' being highlighted by a red box.

Abbildung 16.27: Erstellen eines neuen Benutzerkontos

Daraufhin wird automatisch wieder die Managementseite für Benutzerkonten angezeigt, auf der Sie das neue Benutzerkonto bestätigen können.

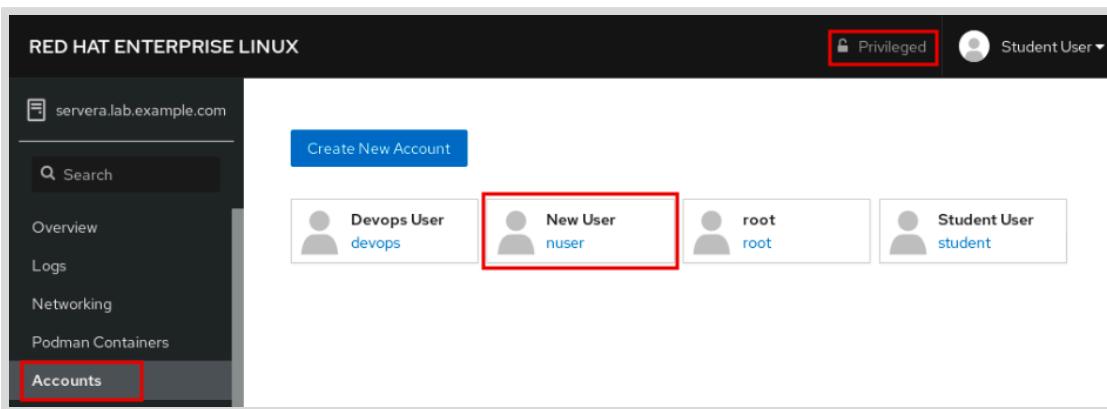


Abbildung 16.28: Managementseite für Benutzerkonto



### Literaturhinweise

Manpages **cockpit(1)**, **cockpit-ws(8)** und **cockpit.conf(5)**

Weitere Informationen finden Sie unter *Managing systems using Web Console* in der Anleitung zur Verwendung von Cockpit zur Verwaltung von Systemen in *Red Hat Enterprise Linux 8* unter

[https://access.redhat.com/documentation/en-us/red\\_hat\\_enterprise\\_linux/8/html-single/managing\\_systems\\_using\\_the\\_web\\_console/](https://access.redhat.com/documentation/en-us/red_hat_enterprise_linux/8/html-single/managing_systems_using_the_web_console/)

## ► Angeleitete Übung

# Analysieren und Verwalten von Remote-Servern

In dieser Übung aktivieren Sie Web Console auf einem Server und greifen darauf zu, um den Server zu verwalten und Probleme zu diagnostizieren und zu beheben.

## Ergebnisse

Sie sollten Web Console verwenden können, um grundlegende Systemfunktionen zu überwachen, Protokolldateien zu untersuchen, Benutzerkonten zu erstellen und auf das Terminal zuzugreifen.

## Bevor Sie Beginnen

Melden Sie sich als Benutzer **student** mit dem Passwort **student** bei **workstation** an.

Führen Sie auf **workstation** den Befehl **lab support-cockpit start** aus. Dieser Befehl führt ein Startskript aus, um zu ermitteln, ob die Hosts **servera** und **serverb** im Netzwerk erreichbar sind, und ggf. Änderungen vorzunehmen.

```
[student@workstation ~]$ lab support-cockpit start
```

- 1. Melden Sie sich mit dem Befehl **ssh** bei **servera** als Benutzer **student** an. Die Systeme sind für die Verwendung von SSH-Schlüsseln zur Authentifizierung konfiguriert. Daher ist kein Passwort erforderlich, um sich bei **servera** anzumelden.

```
[student@workstation ~]$ ssh student@servera
Activate the web console with: systemctl enable --now cockpit.socket

[student@servera ~]$
```

- 2. Web Console ist bereits auf dem System installiert, aber noch nicht aktiviert. Aktivieren und starten Sie den Service **cockpit**.
- 2.1. Führen Sie den Befehl **systemctl enable --now cockpit.socket** aus, um den Web Console-Service zu aktivieren. Führen Sie den Befehl **sudo** aus, um Superuser-Berechtigungen abzurufen, und geben Sie als Passwort **student** ein, wenn Sie dazu aufgefordert werden.

```
[student@servera ~]$ sudo systemctl enable --now cockpit.socket
[sudo] password for student:
Created symlink /etc/systemd/system/sockets.target.wants/cockpit.socket → /usr/
lib/systemd/system/cockpit.socket.
```

- 3. Öffnen Sie auf **workstation** Firefox und melden Sie sich bei der Web Console-Benutzeroberfläche auf **servera.lab.example.com** an. Melden Sie sich als der Benutzer **student** mit dem Passwort **student** an.
- 3.1. Öffnen Sie Firefox und navigieren Sie zu `https://servera.lab.example.com:9090`.
  - 3.2. Akzeptieren Sie das selbstsignierte Zertifikat, indem Sie es als Ausnahme hinzufügen.
  - 3.3. Deaktivieren Sie das Kontrollkästchen **Reuse my password for privileged tasks**.
  - 3.4. Melden Sie sich als der Benutzer **student** mit dem Passwort **student** an.  
Sie sind jetzt als normaler Benutzer mit minimalen Berechtigungen angemeldet.

- 4. Verifizieren Sie Ihre aktuelle Autorisierung innerhalb der Web Console-Benutzeroberfläche.
- 4.1. Klicken Sie in der Navigationsleiste auf **Terminal**, um auf das Terminal zuzugreifen. Es wird eine Terminalsitzung mit dem bereits angemeldeten Benutzer **student** geöffnet. Bestätigen Sie mithilfe des Befehls **id**, dass die Befehlausführung im eingebetteten Terminal funktioniert.

```
[student@servera ~]$ id  
uid=1000(student) gid=1000(student) groups=1000(student),10(wheel)  
context=unconfined_u:unconfined_r:unconfined_t:s0
```

- 4.2. Klicken Sie zum Verwalten von Benutzern in der linken Navigationsleiste auf **Accounts**. Bewegen Sie den Mauszeiger über die Schaltfläche **Create New Account** in der oberen linken Ecke. Beachten Sie, dass der Benutzer **student** nicht zur Erstellung neuer Benutzerkonten berechtigt ist.
  - 4.3. Klicken Sie auf den Link **Student User**. Auf der Kontodetailseite des Benutzers **student** sehen Sie, dass der Benutzer nur berechtigt ist, ein neues Passwort festzulegen oder autorisierte öffentliche SSH-Schlüssel hinzuzufügen.
  - 4.4. Klicken Sie in der rechten oberen Ecke auf **Student User → Log Out**.
- 5. Greifen Sie mit Administratorrechten auf Web Console zu.
- 5.1. Melden Sie sich erneut als Benutzer **student** mit dem Passwort **student** bei der Web Console-Benutzeroberfläche an. Aktivieren Sie dieses Mal jedoch das Kontrollkästchen **Reuse my password for privileged tasks**.
  - 5.2. Um den Administratorzugriff zu verifizieren, vergewissern Sie sich, dass die Bezeichnung **Privileged** neben dem Benutzerkontonamen **Student User** rechts oben in der Web Console-Benutzeroberfläche angezeigt wird.

- 6. Klicken Sie in der linken Navigationsleiste auf **Overview**, um die Systemstatistiken zu untersuchen.
- Auf dieser Seite werden verschiedene grundlegende Betriebssystemstatistiken angezeigt, beispielsweise die aktuelle Auslastung, die Disk-Nutzung, die Disk-I/O und der Netzwerkdatenverkehr.

- 7. Klicken Sie in der linken Navigationsleiste auf **Logs**, um die Systemprotokolle zu überprüfen.

Auf dieser Seite werden die **systemd**-Systemprotokolle angezeigt. Mit den Schaltflächen in der linken oberen Ecke können Sie die Anzeige der Protokolleinträge nach Datum und Schweregrad der Protokolle ändern.

- 7.1. Klicken Sie auf die Liste **Severity** und wählen Sie **Everything** aus.
- 7.2. Klicken Sie je nach dem aktuellen Tag des Monats auf einen beliebigen Protokolleintrag in der Liste. Daraufhin wird eine Detailseite für den Protokolleintrag mit zusätzlichen Informationen zum Ereignis geöffnet, z. B. dem Hostnamen, dem SELinux-Kontext oder der PID-Nummer des Prozesses, dem der Eintrag entspricht.

- 8. Fügen Sie einem vorhandenen Netzwerkschnittstellengerät eine zweite IP-Adresse hinzu.

- 8.1. Klicken Sie in der linken Navigationsleiste auf **Networking**.

Diese Seite zeigt Details der aktuellen Netzwerkkonfiguration für **servera** sowie Echtzeit-Netzwerkstatistiken, Firewall-Konfiguration und Protokolleinträge für das Netzwerk an.

- 8.2. Scrollen Sie nach unten zum Abschnitt **Interfaces** und klicken Sie auf die Zeile für den Netzwerkschnittstellennamen.

Eine Detailseite mit Echtzeit-Netzwerkstatistiken sowie der aktuellen Konfiguration dieser Netzwerkschnittstelle wird angezeigt.

- 8.3. Klicken Sie auf den Link **Address 172.25.250.10/24 via 172.25.250.254**.

Ein Fenster **IPv4 Settings** wird geöffnet, in dem Sie die Konfiguration der Netzwerkschnittstelle ändern können.

- 8.4. Klicken Sie im Fenster **IPv4 Settings** neben **Manual** auf das Pluszeichen (+).

- 8.5. Geben Sie in das Textfeld **Address 172.25.250.99** als zweite IP-Adresse ein.

- 8.6. Geben Sie im Textfeld **Prefix length or Netmask** als Netzmaskenwert **24** ein.

- 8.7. Klicken Sie auf **Apply**, um die neue Netzwerkkonfiguration zu speichern.

Die neue Konfiguration wird sofort übernommen. Die neue IP-Adresse wird in der Zeile **IPv4** angezeigt.

- 9. Erstellen Sie ein neues Benutzerkonto.

- 9.1. Klicken Sie in der linken Navigationsleiste auf **Accounts**.

- 9.2. Klicken Sie auf **Create New Account**.

- 9.3. Fügen Sie im Fenster **Create New Account** die folgenden Details hinzu:

| Feld       | Wert      |
|------------|-----------|
| Full Name  | manager1  |
| User Name  | manager1  |
| Password   | redh@tl23 |
| Bestätigen | redh@tl23 |

9.4. Klicken Sie auf **Create**.

- 10. Rufen Sie innerhalb von Web Console eine Terminal-Sitzung auf, um den Benutzer **manager1** zur Gruppe **wheel** hinzuzufügen.

10.1. Klicken Sie in der linken Navigationsleiste auf **Terminal**.

10.2. Zeigen Sie mithilfe des Befehls **id manager1** die Gruppenmitgliedschaft des Benutzers **manager1** an.

```
[student@servera ~]$ id manager1  
uid=1001(manager1) gid=1001(manager1) groups=1001(manager1)  
[student@servera ~]$
```

10.3. Verwenden Sie den Befehl **sudo usermod -aG wheel manager1**, um **manager1** zur Gruppe **wheel** hinzuzufügen.

```
[student@servera ~]$ sudo usermod -aG wheel manager1  
[sudo] password for student: student  
[student@servera ~]$
```

10.4. Verifizieren Sie mit dem Befehl **id manager1**, ob **manager1** ein Mitglied der Gruppe **wheel** ist.

```
[student@servera ~]$ id manager1  
uid=1001(manager1) gid=1001(manager1) groups=1001(manager1),10(wheel)  
[student@servera ~]$
```

- 11. Aktivieren und starten Sie den Service „Kernel process accounting“ (**psacct**).

11.1. Klicken Sie in der linken Navigationsleiste auf **Services**.

11.2. Suchen Sie nach dem Service **Kernel process accounting**. Klicken Sie auf den Service-Link. Auf einer Detailseite wird der Servicestatus als „deaktiviert“ angezeigt.

11.3. Klicken Sie auf die Schaltfläche **Start und Enable** neben dem Servicenamen.

11.4. Der Service ist jetzt aktiviert und gestartet.

- 12. Melden Sie sich von der Web Console-Benutzeroberfläche ab.

- 13. Melden Sie sich von **servera** ab.

```
[student@servera ~]$ exit  
[student@workstation ~]$
```

## Beenden

Führen Sie auf **workstation** das Skript **lab support-cockpit finish** aus, um diese Übung abzuschließen.

```
[student@workstation ~]$ lab support-cockpit finish
```

Hiermit ist die angeleitete Übung beendet.

# Erhalten von Hilfe im Red Hat Customer Portal

---

## Ziele

Nach Abschluss dieses Abschnitts sollten Sie in der Lage sein, wichtige Ressourcen zu beschreiben, die über das Red Hat Customer Portal verfügbar sind, und mit ihrer Hilfe nach Informationen in der Dokumentation und der Knowledgebase von Red Hat zu suchen.

## Zugreifen auf Support-Ressourcen im Red Hat Customer Portal

Das Red Hat Customer Portal (<https://access.redhat.com>) bietet Kunden Zugriff auf Dokumentation, Downloads, Tools und technisches Fachwissen. In der Knowledgebase können Kunden nach Lösungen, häufig gestellten Fragen (FAQs) und Artikeln suchen. Das Customer Portal bietet Ihnen folgende Möglichkeiten:

- Zugreifen auf offizielle Produktdokumentation
- Einreichen und Verwalten von Support-Tickets
- Verwalten von Software-Subskriptionen und Berechtigungen
- Herunterladen von Software-Downloads, -Updates und -Bewertungen
- Nutzen von Tools zur Optimierung der Konfiguration Ihrer Systeme

Auf einige Bereiche der Site kann jeder Besucher zugreifen, andere Bereiche sind dagegen nur für Kunden mit aktiven Subskriptionen zugänglich. Hilfe zum Zugriff auf das Customer Portal finden Sie hier: <https://access.redhat.com/help/>

## Orientieren im Customer Portal

Sie können über einen Webbrower auf das Red Hat Customer Portal zugreifen. In diesem Abschnitt wird die Tour durch das Customer Portal vorgestellt. Sie finden die Tour hier: <https://access.redhat.com/start>

Die Tour ist ein sehr nützliches Tool, um zu erfahren, was das Portal zu bieten hat und wie Sie Ihre Red Hat-Subskription optimal nutzen können. Sobald Sie sich beim Red Hat Customer Portal angemeldet haben, klicken Sie auf **Tour the Customer Portal**.

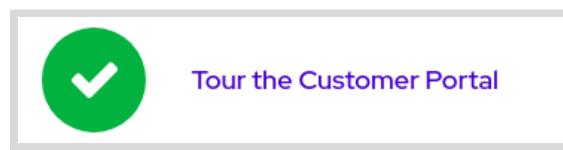


Abbildung 16.29: Tour durch das Customer Portal

Das Fenster **WELCOME TO THE RED HAT CUSTOMER PORTAL** wird mit zwei Optionen geöffnet: **CLOSE** und **NEXT**. Klicken Sie auf **NEXT**, um die Tour zu beginnen. Dies ist das erste einer Reihe von Fenstern, in denen verschiedene Teile der Benutzeroberfläche vorgestellt werden.

## Die obere Navigationsleiste

Die ersten drei Stationen der Tour durch das Customer Portal finden Sie in der oberen Navigationsleiste der Red Hat Customer Portal-Website:



Abbildung 16.30: Obere Navigationsleiste

**Subscriptions** öffnet eine neue Seite, auf der Sie Ihre registrierten Systeme und die Verwendung Ihrer Subskriptionen und Berechtigungen verwalten können. Sie enthält Informationen zu geltenden Errata und ermöglicht Ihnen die Erstellung von *Aktivierungsschlüsseln*, die Sie bei der Registrierung von Systemen verwenden können, um sicherzustellen, dass die Systeme Berechtigungen aus den richtigen Subskriptionen erhalten. Wenn Sie Teil einer Organisation sind, kann Ihr Organisationsadministrator Ihren Zugriff auf diese Seite einschränken.

**Downloads** öffnet eine neue Seite, auf der Sie auf Ihre Produktdownloads zugreifen und Evaluierungsberechtigungen für Produkte anfordern können, für die Sie keine Berechtigungen haben.

**Support Cases** öffnet eine neue Seite, auf der Sie Support-Tickets über das Case Management-System erstellen, verfolgen und verwalten können, sofern Ihre Organisation diese Zugriffsebene autorisiert hat.

Ihr Name ist der Titel für das **Benutzermenü**, in dem Sie Ihr Benutzerkonto, Benutzerkonten, deren Organisationsadministrator Sie sind, Ihr persönliches Profil sowie Optionen für E-Mail-Benachrichtigungen über verfügbare neue Inhalte verwalten können.

Das Globussymbol öffnet das Menü **Select Your Language**, um Ihre Sprachpräferenzen für das Customer Portal festzulegen.

## Themenmenüs

Unter der oberen Navigationsleiste auf der Hauptseite des Customer Portal befinden sich Menüs, über die Sie zu vier Hauptkategorien der auf der Website verfügbaren Ressourcen navigieren können.

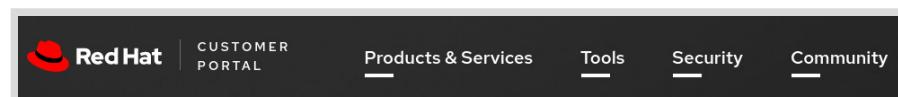


Abbildung 16.31: Ressourcenmenüs

**Products & Services** bietet Zugriff auf die *Produkt-Hubs*. Dies sind Seiten, über die auf produktsspezifische Auswertungen, Übersichten, Leitfäden für erste Schritte und andere Produkt-Supportinformationen zugegriffen werden kann. Außerdem können Sie auf Dokumentation für Red Hat-Produkte, Direkt-Links zur Knowledgebase mit Support-Artikeln sowie Informationen zu Support-Richtlinien und Möglichkeiten zur Kontaktaufnahme mit dem Red Hat-Support zugreifen.

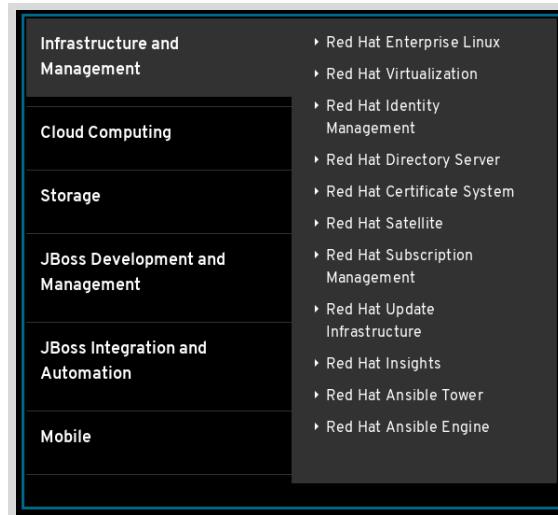


Abbildung 16.32: Products &amp; Services

Das Menü **Tools** enthält Links zu Tools für die erfolgreiche Nutzung von Red Hat-Produkten. Der Abschnitt „Solution Engine“ bietet Ihnen eine effiziente Möglichkeit, auf Basis des Produkts schnell nach Lösungen für Ihre Probleme zu suchen und ein Support-Ticket zu öffnen, wenn Sie keine zufriedenstellende Lösung finden. Der Abschnitt „Customer Portal Labs“ stellt eine Sammlung von webbasierten Anwendungen und Tools zur Verfügung, mit denen Sie die Leistung verbessern, Probleme diagnostizieren, Sicherheitsprobleme erkennen und Ihre Konfigurationen optimieren können. Mit dem Tool Product Life Cycle Checker können Sie beispielsweise ein bestimmtes Produkt auswählen und seinen Support-Lebenszyklus-Zeitplan anzeigen. Ein anderes Tool, der Rescue Mode Assistant, hilft Ihnen beim Zurücksetzen des root-Passwort eines Systems, beim Generieren von Diagnoseberichten oder beim Beheben von Problemen mit Dateisystemen beim Booten. Es gibt jedoch noch viele andere Tools auf dieser Website.

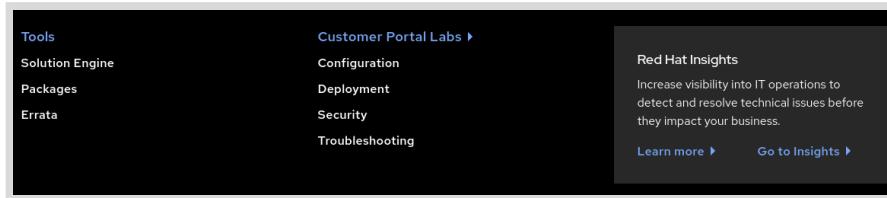


Abbildung 16.33: Menü „Tools“ in Customer Portal

Über den Abschnitt „Security“ können Sie auf das *Red Hat Product Security Center* unter <https://access.redhat.com/security/> zugreifen. Dieser Abschnitt enthält auch Informationen zu bekannten Sicherheitsproblemen, bietet Zugriff auf die Red Hat CVE-Datenbank, den Security-Kanal des Red Hat-Blogs und Ressourcen zum Sicherheitsreaktionsprozess von Red Hat sowie dazu, wie wir Probleme bewerten und lösen.

Der Abschnitt „Community“ schließlich ist ein Ort, an dem Red Hat-Experten, -Kunden und -Partner miteinander kommunizieren und zusammenarbeiten können. Hier finden Sie Diskussionsforen, Blogs und Informationen zu bevorstehenden Veranstaltungen in Ihrer Nähe.



### Anmerkung

Sie sollten die gesamte Tour unter Erste Schritte mit Red Hat [<https://access.redhat.com/start>] absolvieren, einschließlich der Abschnitte zur Personalisierung Ihrer Customer Portal-Oberfläche und zur Erkundung der Vorteile Ihrer Red Hat-Subskription, um alle Aspekte des Customer Portal kennen zu lernen. Sie benötigen mindestens eine aktive Subskription in Ihrem Customer Portal-Benutzerkonto, um auf diese Seite zuzugreifen.

## Durchsuchen der Knowledgebase mit dem Red Hat Support Tool

Das Red Hat Support Tool-Dienstprogramm, **redhat-support-tool**, bietet eine textbasierte Schnittstelle, mit der Sie Artikel in der Knowledgebase und Support-Tickets im Customer Portal über die Befehlszeile Ihres Systems durchsuchen können. Das Tool hat keine grafische Benutzeroberfläche und benötigt einen Internetzugang, da es in das Red Hat Customer Portal interagiert. Führen Sie den Befehl **redhat-support-tool** über eine beliebige Terminal- oder SSH-Verbindung aus.

Der Befehl **redhat-support-tool** kann in einem interaktiven Modus verwendet oder als Befehl mit Optionen und Argumenten aufgerufen werden. Die Syntax des Tools ist bei beiden Methoden identisch. Standardmäßig wird das Programm im interaktiven Modus gestartet. Verwenden Sie den Sub-Befehl **help**, um alle verfügbaren Befehle anzuzeigen. Der interaktive Modus unterstützt die Tab-Vervollständigung und das Aufrufen von Programmen in der übergeordneten Shell.

```
[user@host ~]$ redhat-support-tool
Welcome to the Red Hat Support Tool.
Command (? for help):
```

Beim ersten Aufruf von **redhat-support-tool** wird eine Aufforderung zur Eingabe von Anmeldeinformationen des Red Hat Customer Portal-Subskribenten angezeigt. Damit diese Angaben nicht wiederholt eingegeben werden müssen, können die Kontoinformationen mit dem Tool im Benutzerverzeichnis gespeichert werden (`~/.redhat-support-tool/redhat-support-tool.conf`). Wenn alle Befehle über ein bestimmtes Red Hat Customer Portal-Benutzerkonto aufgerufen werden, kann die Option **--global** Kontoinformationen zusammen mit anderen systemweiten Konfigurationseinstellungen in `/etc/redhat-support-tool.conf` speichern. Die Tool-Konfigurationseinstellungen werden mit dem **config**-Befehl des Tools geändert.

Mit dem Befehl **redhat-support-tool** können Subskribenten Inhalte der Knowledgebase über das Red Hat Customer Portal suchen und anzeigen. Die Knowledgebase lässt Schlüsselwortsuchen zu, ähnlich wie der Befehl **man**. Sie können Fehlercodes, Syntax aus Protokolldateien oder eine beliebige Kombination aus Schlüsselwörtern eingeben, um eine Liste mit relevanten Lösungsdokumenten zu erhalten.

Nachfolgend sind eine Erstkonfiguration und eine einfache Suche dargestellt:

```
[user@host ~]$ redhat-support-tool
Welcome to the Red Hat Support Tool.
Command (? for help): search How to manage system entitlements with subscription-
manager
Please enter your RHN user ID: subscriber
Save the user ID in /home/student/.redhat-support-tool/redhat-support-tool.conf
(y/n): y
Please enter the password for subscriber: password
Save the password for subscriber in /home/student/.redhat-support-tool/redhat-
support-tool.conf (y/n): y
```

Nachdem der Benutzer die erforderlichen Konfigurationseinstellungen an der Eingabeaufforderung angegeben hat, fährt das Tool mit der ursprünglichen Suchanfrage fort:

```
Type the number of the solution to view or 'e' to return to the previous menu.
1 [ 253273:VER] How to register and subscribe a system to the Red Hat Customer
    Portal using Red Hat Subscription-Manager
2 [ 265523:VER] Enabling or disabling a repository using Red Hat Subscription
    Management
3 [ 100423:VER] Why does subscription-manager list return: "No Installed
    Products found" ?
...output omitted...
Select a Solution: 1
```

Wenn Sie Artikel Nummer 1 wie oben auswählen, werden Sie aufgefordert, den zu lesenden Abschnitt des Dokuments auszuwählen. Drücken Sie schließlich die Taste **Q**, um den aktuellen Abschnitt zu verlassen, oder drücken Sie die Taste wiederholt, um den Befehl **redhat-support-tool** zu beenden.

```
Select a Solution: 1

Type the number of the section to view or 'e' to return to the previous menu.
1 Title
2 Issue
3 Environment
4 Resolution
5 Display all sections
End of options.
Section: 1

Title
=====
How to register and subscribe a system to the Red Hat Customer Portal using Red
    Hat Subscription-Manager
URL:      https://access.redhat.com/solutions/253273
Created On: None
Modified On: 2017-11-29T15:33:51Z

(END) q
Section:
Section: q

Select a Solution: q
```

```
Command (? for help): q
[user@hosts ~]#
```

## Zugreifen auf Knowledgebase-Artikel anhand der Dokumenten-ID

Sie können direkt nach Online-Artikeln suchen, indem Sie den **kb**-Befehl des Tools mit der Knowledgebase-Dokumenten-ID verwenden. Ein zurückgegebenes Dokument wird ohne Seitenumbruch auf dem Bildschirm angezeigt. Sie können es jedoch in eine Datei umleiten, um es zu speichern, und **less** verwenden, um es bildschirmweise durchzuscrollen.

```
[user@host ~]$ redhat-support-tool kb 253273

Title
=====
How to register and subscribe a system to the Red Hat Customer Portal using Red
Hat Subscription-Manager
URL:      https://access.redhat.com/solutions/253273
Created On: None
Modified On: 2017-11-29T15:33:51Z

Issue
=====
* How to register a new `Red Hat Enterprise Linux` system to the Customer Portal
  using `Red Hat Subscription-Manager`
...output omitted...
```

## Verwalten von Support-Tickets mit dem Red Hat Support Tool

Eine Produktsubskription bietet u. a. den Vorteil, dass Sie über das Red Hat Customer Portal auf technischen Support zugreifen können. Je nach Support-Level des Systems für die Subskription können Sie Red Hat über Online-Tools oder per Telefon kontaktieren. Ausführliche Informationen finden Sie hier: [https://access.redhat.com/site/support/policy/support\\_process](https://access.redhat.com/site/support/policy/support_process)

### Vorbereiten von Fehlerberichten

Sammeln Sie die relevanten Informationen für Ihren Fehlerbericht, bevor Sie sich an den Support von Red Hat wenden.

*Definieren Sie das Problem.* Das Problem und die damit verbundenen Symptome sollten anschaulich beschrieben werden können. Beschreiben Sie alles so genau wie möglich. Geben Sie die Schritte an, mit denen das Problem reproduziert werden kann.

*Sammeln Sie Hintergrundinformationen.* Welches Produkt und welche Version sind betroffen? Halten Sie relevante Diagnoseinformationen bereit. Diese können die Ausgabe von **sosreport** einschließen (dieses Thema wird später in diesem Abschnitt erläutert). Bei Kernel-Problemen kann dies **kdump**-Absturzabbilder des Systems oder ein digitales Foto der Kernel-Rückverfolgung einschließen, die auf dem Bildschirm des abgestürzten Systems angezeigt werden.

*Ermitteln Sie den Schweregrad.* Red Hat verwendet vier Schweregrade zur Einordnung von Problemen. Nach einer Meldung von Problemen mit dem Schweregrad *Urgent* und

*High* sollte das entsprechende lokale Support-Center angerufen werden (siehe <https://access.redhat.com/site/support/contact/technicalSupport>).

| Schweregrad                   | Beschreibung                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                                  |
|-------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|
| <i>Urgent</i> (Schweregrad 1) | Ein Problem, das die Verwendung der Software in einer Produktionsumgebung erheblich beeinträchtigt. Dies beinhaltet den Verlust von Produktionsdaten oder fehlerhafte Produktionssysteme. Durch das Problem können Unternehmensprozesse nicht mehr durchgeführt werden, und es existiert kein Workaround.                                                                                                                                                                                                                                                     |
| <i>High</i> (Schweregrad 2)   | Ein Problem, bei dem die Software weiterhin verwendet werden kann, ihr Einsatz in einer Produktionsumgebung jedoch stark eingeschränkt ist. Durch das Problem werden die Unternehmensprozesse stark beeinträchtigt, und es existiert kein Workaround.                                                                                                                                                                                                                                                                                                         |
| <i>Medium</i> (Schweregrad 3) | Ein Problem, das den teilweise, aber nicht kritischen Ausfall der Software in einer Produktions- oder Entwicklungsumgebung zur Folge hat. Die Produktionsumgebungen im Unternehmen werden nur geringfügig bis mittelschwer beeinträchtigt. Das Unternehmen ist mittels eines Workarounds weiterhin dazu in der Lage, das tägliche Geschäft abzuwickeln. Bei Entwicklungsumgebungen führt die Situation zu Problemen beim Übergang Ihres Projekts in die Produktionsphase.                                                                                     |
| <i>Low</i> (Schweregrad 4)    | Eine Frage zur allgemeinen Handhabung, das Einsenden eines Dokumentierungsfehlers oder Vorschläge für zukünftige Produktverbesserungen oder -modifikationen. Diese Probleme verursachen bei Produktionsumgebungen wenige bis keine Einschränkungen in Bezug auf Ihr Unternehmen oder die Leistung bzw. Funktionalität Ihres Systems. Die Entwicklungsumgebungen in Ihrem Unternehmen werden nur geringfügig bis mittelschwer beeinträchtigt, und Ihr Unternehmen ist mittels eines Workarounds weiterhin dazu in der Lage, das tägliche Geschäft abzuwickeln. |

## Verwalten von Fehlerberichten mit redhat-support-tool

Sie können Red Hat Support-Tickets mit **redhat-support-tool** erstellen, anzeigen, ändern und schließen. Wenn Support-Tickets den Status **opened** oder **maintained** aufweisen, können Benutzer Dateien oder Dokumentation anhängen, z. B. Diagnoseberichte (`sosreport`). Die Dateien werden mit dem Tool hochgeladen und an Tickets angehängt.

Ticketdetails, einschließlich Produktnamen, Version, Zusammenfassung, Beschreibung, Schweregrad und Ticketgruppe, können mithilfe von Befehlsoptionen zugewiesen werden. Alternativ können Sie die erforderlichen Informationen gemäß den Eingabeaufforderungen des Tools angeben. Im folgenden Beispiel wird ein neues Ticket geöffnet. Die Optionen **--product** und **--version** werden angegeben.

```
[user@host ~]$ redhat-support-tool
Welcome to the Red Hat Support Tool.
Command (? for help): opencase --product="Red Hat Enterprise Linux" --
version="7.0"
Please enter a summary (or 'q' to exit): System fails to run without power
Please enter a description (Ctrl-D on an empty line when complete):
When the server is unplugged, the operating system fails to continue.
```

```
1 Urgent
2 High
3 Normal
4 Low
Please select a severity (or 'q' to exit): 4
Would you like to assign a case group to this case (y/N)? N
Would see if there is a solution to this problem before opening a support case?
(y/N) N
-----
Support case 01034421 has successfully been opened.
```

Wenn die Optionen **--product** und **--version** nicht angegeben werden, stellt der Befehl **redhat-support-tool** eine Liste mit Auswahlmöglichkeiten für diese Optionen bereit.

```
[user@host ~]$ redhat-support-tool
Welcome to the Red Hat Support Tool.
Command (? for help): opencase
Do you want to use the default product - "Red Hat Enterprise Linux" (y/N)?: y
...output omitted...
29 7.4
30 7.5
31 7.6
32 8.0 Beta
Please select a version (or 'q' to exit): 32
Please enter a summary (or 'q' to exit): yum fails to install apache
Please enter a description (Ctrl-D on an empty line when complete):
yum cannot find correct repo
1 Urgent
2 High
3 Normal
4 Low
Please select a severity (or 'q' to exit): 4
Would you like to use the default (Ungrouped Case) Case Group (y/N)? : y
Would you like to see if there's a solution to this problem before opening a
support case? (y/N) N
-----
Support case 010355678 has successfully been opened.
```

## Anhängen von Diagnoseinformationen an ein Support-Ticket

Das Einbeziehen von Diagnoseinformationen kann zu einer schnelleren Lösung führen. Hängen Sie den **sosreport** an, wenn das Ticket geöffnet wird. Mit dem **sosreport**-Befehl wird ein komprimiertes tar-Archiv generiert, das im aktiven System gesammelte Diagnoseinformationen enthält. Wenn bereits ein Archiv erstellt wurde, fordert **redhat-support-tool** Sie auf, dieses anzugeben:

```
Please attach a SoS report to support case 01034421. Create a SoS report as
the root user and execute the following command to attach the SoS report
directly to the case:
redhat-support-tool addattachment -c 01034421 path to sosreport

Would you like to attach a file to 01034421 at this time? (y/N) N
Command (? for help):
```

Wenn kein aktueller SoS-Bericht vorhanden ist, kann ein Administrator später einen erstellen und anhängen. Verwenden Sie den Befehl **redhat-support-tool addattachment**, um den Bericht anzuhängen.

Support-Tickets können auch vom Subskribenten angezeigt, geändert und geschlossen werden:

```
Command (? for help): listcases

Type the number of the case to view or 'e' to return to the previous menu.
1 [Waiting on Red Hat] System fails to run without power
No more cases to display
Select a Case: 1

Type the number of the section to view or 'e' to return to the previous menu.
1 Case Details
2 Modify Case
3 Description
4 Recommendations
5 Get Attachment
6 Add Attachment
7 Add Comment
End of options.
Option: q

Select a Case: q

Command (? for help):q

[user@host ~]$ redhat-support-tool modifycase --status=Closed 01034421
Successfully updated case 01034421
[user@host ~]$
```

Das Red Hat Support Tool bietet erweiterte Diagnose- und Analysefunktionen für Anwendungen. Mithilfe von Kernel-Absturzabbild-Hauptdateien kann **redhat-support-tool** eine Rückverfolgung erstellen und extrahieren. Eine Rückverfolgung ist ein Bericht der aktiven Stack-Frames zum Zeitpunkt eines Absturzabbilds und ermöglicht eine Vor-Ort-Diagnose. Eine der Optionen des Befehls **redhat-support-tool** ist das Öffnen eines Support-Tickets.

Das Tool stellt auch eine Analyse der Protokolldatei bereit. Mit dem **analyze**-Befehl des Tools können viele Arten von Protokolldateien, u. a. Betriebssystem, JBoss, Python, Tomcat und oVirt, analysiert werden, um Symptome von Problemen zu erkennen. Die Protokolldateien können anschließend einzeln angezeigt und analysiert werden können. Durch die Bereitstellung von im Voraus bearbeiteten Analysen – anstelle von Rohdaten wie Absturzabbildern oder Protokolldateien – können Support-Tickets schneller geöffnet und den Technikern zur Verfügung gestellt werden.

## Beitreten zu Red Hat Developer

Eine weitere nützliche Ressource, die von Red Hat bereitgestellt wird, ist Red Hat Developer. Dieses auf <https://developer.redhat.com> gehostete Programm bietet Subskriptionsberechtigungen für Red Hat Software für Entwicklungszwecke sowie Dokumentation und Premium-Bücher von unseren Experten für Mikroservices, serverloses Computing, Kubernetes und Linux. Ein Blog, Links zu Informationen über bevorstehende Veranstaltungen und Schulungen sowie weitere Hilfsmittel und Links zum Red Hat Customer Portal sind ebenfalls verfügbar.

Die Registrierung ist kostenlos und kann unter <https://developer.redhat.com/register> vorgenommen werden.



### Literaturhinweise

Manpage **sosreport**<sup>(1)</sup>

#### **Red Hat Access: Red Hat Support Tool**

<https://access.redhat.com/site/articles/445443>

#### **Red Hat Support Tool First Use**

<https://access.redhat.com/site/videos/534293>

#### **Kontaktaufnahme mit dem technischen Support von Red Hat**

[https://access.redhat.com/site/support/policy/support\\_process/](https://access.redhat.com/site/support/policy/support_process/)

#### **Hilfe – Red Hat Customer Portal**

<https://access.redhat.com/site/help/>

## ► Angeleitete Übung

# Erhalten von Hilfe im Red Hat Customer Portal

In dieser Übung erstellen Sie mithilfe von Web Console einen Diagnosebericht.

### Ergebnisse

Sie sollten in der Lage sein, einen Diagnosebericht mithilfe von Web Console zu erstellen, der als Teil eines Support-Tickets an das Red Hat Customer Portal gesendet werden kann.

### Bevor Sie Beginnen

Melden Sie sich als Benutzer **student** mit dem Passwort **student** bei **workstation** an.

Führen Sie auf **workstation** den Befehl **lab support-portal start** aus. Dieser Befehl führt ein Startskript aus, das ermittelt, ob **servera** im Netzwerk erreichbar ist. Außerdem startet und aktiviert er Web Console auf **servera**.

```
[student@workstation ~]$ lab support-portal start
```

- 1. Melden Sie sich auf **workstation** mit dem Befehl **ssh** bei **servera** als Benutzer **student** an.

```
[student@workstation ~]$ ssh student@servera
Web console: https://servera.lab.example.com:9090/ or https://172.25.250.10:9090/
[student@servera ~]$
```

- 2. Überprüfen Sie mit dem Befehl **systemctl**, ob der Service **cockpit** ausgeführt wird. Geben Sie **student** ein, wenn Sie zur Passworteingabe aufgefordert werden.

```
[student@servera ~]$ systemctl status cockpit.socket
● cockpit.socket - Cockpit Web Service Socket
  Loaded: loaded (/usr/lib/systemd/system/cockpit.socket; enabled; vendor preset: disabled)
  Active: active (listening) since Thu 2019-05-16 10:32:33 IST; 4min 37s ago
    Docs: man:cockpit-ws(8)
   Listen: [::]:9090 (Stream)
  Process: 676 ExecStartPost=/bin/ln -snf active.motd /run/cockpit/motd (code=exited, status=0/SUCCESS)
  Process: 668 ExecStartPost=/usr/share/cockpit/motd/update-motd localhost (code=exited, status=0/SUCCESS)
    Tasks: 0 (limit: 11405)
   Memory: 1.5M
      CGroup: /system.slice/cockpit.socket
      ...output omitted...
```

- 3. Melden Sie sich von **servera** ab.

```
[student@servera ~]$ exit  
[student@workstation ~]$
```

- 4. Öffnen Sie *Firefox* auf **workstation** und melden Sie sich als Benutzer **root** mit dem Passwort **redhat** bei der Web Console-Benutzeroberfläche an, die auf **servera.lab.example.com** ausgeführt wird.
- 4.1. Öffnen Sie *Firefox* und rufen Sie die Adresse **https://servera.lab.example.com:9090** auf.
  - 4.2. Akzeptieren Sie bei der entsprechenden Aufforderung das selbstsignierte Zertifikat, indem Sie es als Ausnahme hinzufügen.
  - 4.3. Melden Sie sich als Benutzer **root** mit dem Passwort **redhat** an. Sie sind jetzt als berechtigter Benutzer angemeldet, was zum Erstellen eines Diagnoseberichts erforderlich ist.
  - 4.4. Klicken Sie in der linken Navigationsleiste auf **Diagnostic Reports**. Klicken Sie auf **Create Report**. Die Erstellung des Berichts dauert einige Minuten.
- 5. Wenn der Bericht fertig ist, klicken Sie auf **Download report**. Speichern Sie die Datei.
- 5.1. Klicken Sie auf die Schaltfläche **Download report** und dann auf die Schaltfläche **Save File**.
  - 5.2. Klicken Sie auf die Schaltfläche **Close**.
  - 5.3. Melden Sie sich von der Web Console-Benutzeroberfläche ab.

## Beenden

Führen Sie auf **workstation** das Skript **lab support-portal finish** aus, um diese Übung zu beenden.

```
[student@workstation ~]$ lab support-portal finish
```

Hiermit ist die angeleitete Übung beendet.

# Erkennen und Lösen von Problemen mit Red Hat Insights

## Ziele

Nach Abschluss dieses Abschnitts sollten Sie Red Hat Insights verwenden können, um Server auf Probleme zu überprüfen, diese zu beheben oder zu lösen und zu bestätigen, dass die Lösung funktioniert.

## Einführung in Red Hat Insights

Red Hat Insights ist ein Predictive Analytics-Tool, mit dem Sie Bedrohungen für die Sicherheit, Leistung, Verfügbarkeit und Stabilität von Systemen, auf denen Red Hat-Produkte in Ihrer Infrastruktur ausgeführt werden, erkennen und beheben können. Insights wird als Software-as-a-Service-(SaaS-)Produkt bereitgestellt, sodass Sie es schnell bereitstellen und skalieren können, ohne dass zusätzliche Infrastrukturanforderungen erfüllt werden müssen. Außerdem können Sie sofort von den neuesten Empfehlungen und Updates von Red Hat profitieren, die für Ihre bereitgestellten Systeme spezifisch sind.

Red Hat aktualisiert die von Insights verwendete Knowledgebase regelmäßig basierend auf gängigen Support-Risiken, Sicherheitslücken, als fehlerhaft bekannten Konfigurationen und anderen von Red Hat identifizierten Problemen. Maßnahmen zur Minderung oder Behebung dieser Probleme werden von Red Hat validiert und verifiziert. Auf diese Weise können Sie Probleme proaktiv identifizieren, priorisieren und lösen, bevor sie zu einem größeren Problem werden.

Für jedes erkannte Problem stellt Insights Schätzungen des dadurch entstehenden Risikos und Empfehlungen zur Minderung oder Behebung des Problems bereit. Diese Empfehlungen stellen möglicherweise Materialien wie Ansible Playbooks oder visuell lesbare Schritt-für-Schritt-Anweisungen zur Behebung des Problems bereit.

Die Empfehlungen von Insights werden auf jedes für den Service registrierte System individuell zugeschnitten. Sie installieren jedes Client-System mit einem Agenten, der Metadaten über die Laufzeitkonfiguration des Systems sammelt. Diese Daten sind eine Teilmenge dessen, was Sie dem Red Hat Support möglicherweise mit dem Befehl **sosreport** zur Verfügung stellen, um ein Support-Ticket zu lösen. Sie können die Daten, die Clients senden, beschränken oder ausblenden. Je nachdem, was Sie einschränken, können einige der Analyseregeln möglicherweise nicht mehr ausgeführt werden.

Fast sofort nachdem Sie einen Server registriert haben und die erste Synchronisierung der Systemmetadaten abgeschlossen ist, sollten Ihr Server und etwaige Empfehlungen für ihn in der Insights-Konsole im Red Hat Customer Portal angezeigt werden.

Derzeit bietet Insights Predictive Analytics und Empfehlungen für die folgenden Red Hat-Produkte:

- Red Hat Enterprise Linux 6.4 und höher
- Red Hat Virtualization 4 und höher
- Red Hat OpenShift Container Platform
- Red Hat OpenStack Platform 7 und höher

## Beschreiben der Insights-Architektur

Wenn Sie ein System mit Insights registrieren, sendet es sofort Metadaten über die aktuelle Konfiguration an die Insights-Plattform. Nach der Registrierung aktualisiert das System die für Insights bereitgestellten Metadaten regelmäßig. Das System sendet die Metadaten mit TLS-Verschlüsselung, um die Metadaten bei der Übertragung zu schützen.

Wenn Insights die Daten empfängt, analysiert es die Daten und zeigt das Ergebnis in der Web Console von Insights unter <https://cloud.redhat.com/insights> an.

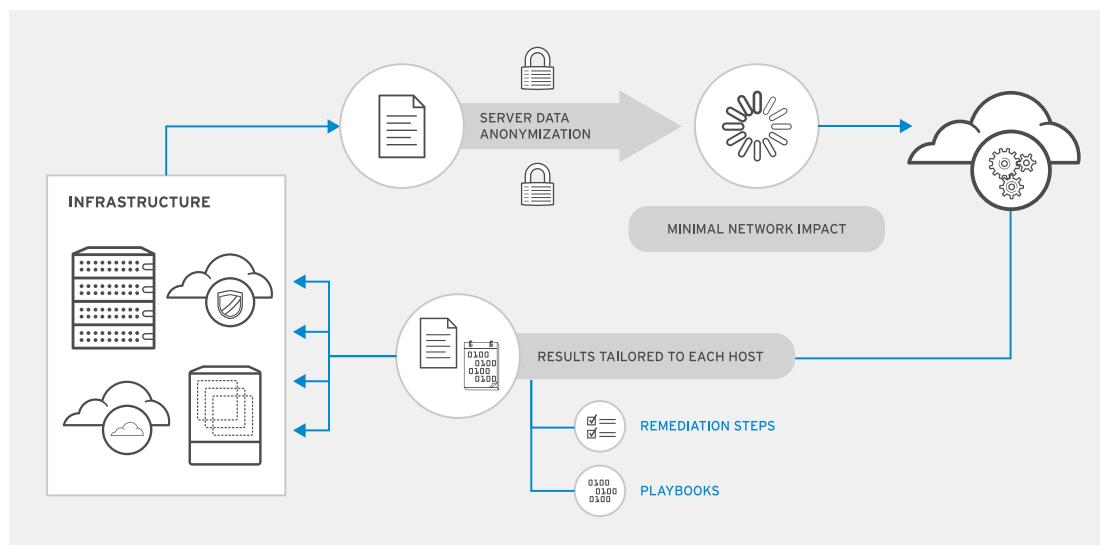


Abbildung 16.34: Basisarchitektur von Insights

## Installieren von Insights-Clients

Insights ist in Red Hat Enterprise Linux 8 als Teil der Subskription enthalten. Ältere Versionen von Red Hat Enterprise Linux-Servern benötigen zur Installation das Paket *insights-client* auf dem System.



### Wichtig

Das Paket *insights-client* ersetzt das ältere Paket *redhat-access-insights* ab Red Hat Enterprise Linux 7.5.

Wenn Ihr System über den Customer Portal Subscription Management-Service für Softwareberechtigungen registriert ist, können Sie Insights mit nur einem Befehl aktivieren. Führen Sie den Befehl **insights-client --register** aus, um das System zu registrieren.

```
[root@host ~]# insights-client --register
```

Der Insights-Client aktualisiert regelmäßig die für Insights bereitgestellten Metadaten. Sie können jederzeit den Befehl **insights-client** ausführen, um die Metadaten des Clients zu aktualisieren.

```
[root@host ~]# insights-client
Starting to collect Insights data for host.example.com
Uploading Insights data.
Successfully uploaded report for host.example.com.
View details about this system on cloud.redhat.com:
https://cloud.redhat.com/insights/inventory/dc480efd-4782-417e-a496-cb33e23642f0
```

## Registrieren eines RHEL-Systems bei Insights

Der allgemeine Prozess zum Registrieren eines RHEL-Servers bei Insights sieht folgendermaßen aus:

- Registrieren Sie das System interaktiv mit dem Red Hat Subscription Management-Service.

```
[root@host ~]# subscription-manager register --auto-attach
```

- Stellen Sie sicher, dass das Paket `insights-client` auf dem System installiert ist. In RHEL 7 ist dieses Paket im Kanal **rhel-7-server-rpms** enthalten.



### Anmerkung

Dieser Schritt ist bei Red Hat Enterprise Linux 8-Systemen nicht erforderlich.

```
[root@host ~]# yum install insights-client
```

- Führen Sie den Befehl `insights-client --register` aus, um das System beim Insights-Service zu registrieren und die ersten Systemmetadaten hochzuladen.

```
[root@host ~]# insights-client --register
```

- Vergewissern Sie sich, dass das System unter **Inventory** in der Insights Web Console unter <https://cloud.redhat.com/insights> angezeigt wird.

The screenshot shows the Red Hat Insights Cloud-Portal interface. The left sidebar has a dark background with white text and icons. It includes links for Red Hat Insights, Dashboard, Advisor, Vulnerability, Compliance, Patch, Drift, Policies, Inventory (which is highlighted with a blue bar), Remediations, Register Systems, and Support. The main content area is titled 'Inventory'. At the top, there is a search bar with a dropdown for 'Name' and a 'Filter by name' input field, along with a magnifying glass icon and a 'Delete' button. Below the search bar are filter buttons for 'Status' (Fresh, Stale), 'Source', 'Insights', and a 'Clear filters' link. A table lists four system entries: 'host.example.com' (last seen 32 minutes ago), 'utility.example.com' (40 minutes ago), 'rhel8.local' (8 hours ago), and 'rhel8.phoenix.home.lan' (8 hours ago). Each entry has a checkbox, a 'Tags' column with a shield icon and '0' count, and a 'Last seen' column with a downward arrow. To the right of each entry are three vertical dots. The bottom of the table shows a page number '1 - 5 of 5' and navigation arrows.

Abbildung 16.35: Insights Inventory im Cloud-Portal

## Navigieren in der Insights-Konsole

Insights bietet eine Reihe von Services, auf die Sie über die Web Console unter <https://cloud.redhat.com/insights> zugreifen können.

## Ermitteln von Konfigurationsproblemen mit dem Advisor-Service

Der Advisor-Service meldet Konfigurationsprobleme, die sich auf Ihre Systeme auswirken. Sie greifen über das Menü **Advisor → Recommendations** auf den Service zu.

The screenshot shows the Red Hat Advisor recommendations interface. It displays three items:

- Incident:** Decreased stability and/or performance due to filesystem over 95% capacity and available space is less than 100MB. Status: Enabled. Added: 5 months ago. Total risk: Important. Risk of change: Low. System: 1. Ansible: No.
- Traffic occurs or services are allowed unexpectedly when firewall zone drifting is enabled.** Status: Enabled. Added: 5 months ago. Total risk: Moderate. Risk of change: Moderate. System: 2. Ansible: No.
- Decreased security: Yum GPG verification disabled (third-party repos)** Status: Enabled. Added: 3 years ago. Total risk: Low. Risk of change: Very Low. System: 1. Ansible: No.

Abbildung 16.36: Empfehlungen des Advisor-Services

Insights bietet für jedes Problem zusätzliche Informationen, mit deren Hilfe Sie das Problem verstehen, Maßnahmen zu dessen Behebung priorisieren, die verfügbaren Minderungs- oder Abhilfemaßnahmen ermitteln und die Lösung mit einem Ansible Playbook automatisieren können. Insights bietet zudem Links zu Knowledgebase-Artikeln im Customer Portal.

The screenshot shows the Red Hat Advisor details interface for the first item. It displays the following information:

- Incident:** Decreased stability and/or performance due to filesystem over 95% capacity and available space is less than 100MB. Status: Enabled. Added: 5 months ago. Total risk: Important. Risk of change: Low. System: 1. Ansible: No.
- Description:** File systems nearing full capacity or inode usage can cause performance issues because blocks must be used from different block groups. Besides, file systems at or exceeding capacity will have stability issues because applications will no longer be able to write to the file system.
- Knowledgebase article:** A link to a knowledgebase article.
- View the affected system:** A link to view the affected system.
- Total risk:** The total risk of this remediation is **important**, based on the combination of likelihood and impact to remediate. Critical likelihood (red bar) and Medium impact (yellow bar).
- Risk of change:** The risk of change is **low**, because the change does not require that a system be taken offline. System reboot is **not required**.

Abbildung 16.37: Details zu einem Problem

Der Advisor-Service bewertet das Risiko, das ein Problem für Ihr System darstellt, in zwei Kategorien.

**Total risk**

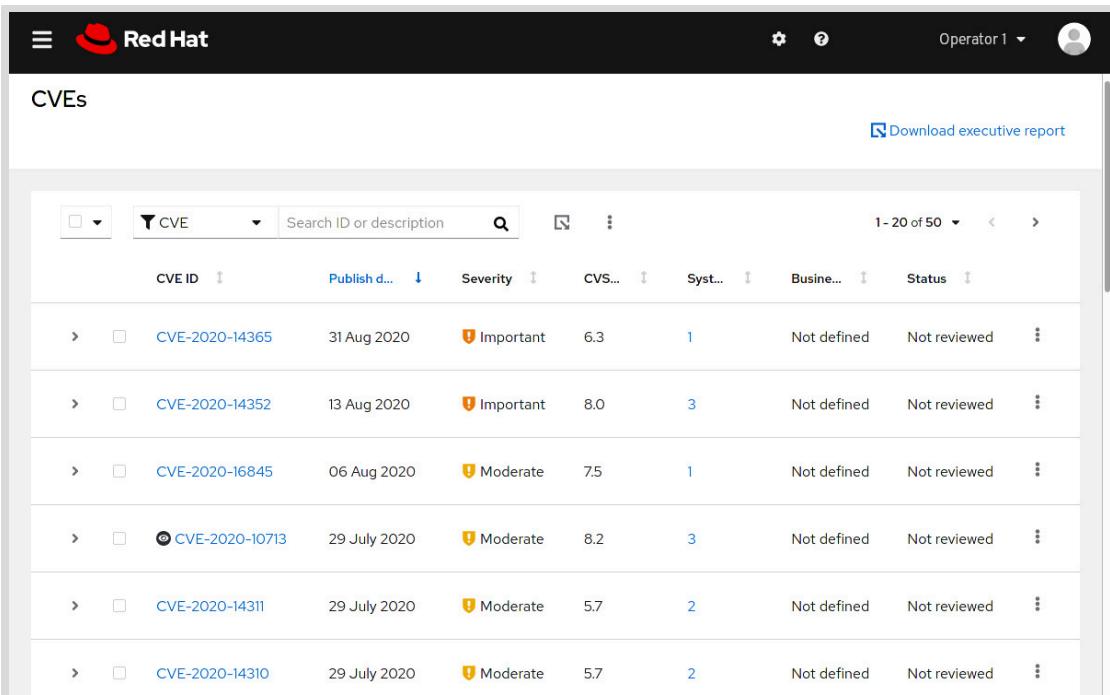
Zeigt die Auswirkungen des Problems auf Ihr System an.

**Risk of change**

Zeigt die Auswirkungen der Wiederherstellungsmaßnahme auf Ihr System an. So müssen Sie zum Beispiel das System möglicherweise neu starten.

**Bewerten der Sicherheit mit dem Vulnerability Service**

Der Vulnerability Service meldet *Common Vulnerabilities and Exposures (CVEs)*, die sich auf Ihre Systeme auswirken. Sie greifen über das Menü **Vulnerability** → **CVEs** auf den Service zu.



The screenshot shows the Red Hat Vulnerability Service interface. At the top, there's a navigation bar with the Red Hat logo, user information (Operator 1), and a download link for an executive report. Below the header, the page title is "CVEs". The main content is a table listing six CVE entries. The columns are: CVE ID, Publish d..., Severity, CVS..., Syst..., Busine..., and Status. The data is as follows:

| CVE ID                         | Publish d... | Severity  | CVS... | Syst... | Busine...   | Status       |
|--------------------------------|--------------|-----------|--------|---------|-------------|--------------|
| <a href="#">CVE-2020-14365</a> | 31 Aug 2020  | Important | 6.3    | 1       | Not defined | Not reviewed |
| <a href="#">CVE-2020-14352</a> | 13 Aug 2020  | Important | 8.0    | 3       | Not defined | Not reviewed |
| <a href="#">CVE-2020-16845</a> | 06 Aug 2020  | Moderate  | 7.5    | 1       | Not defined | Not reviewed |
| <a href="#">CVE-2020-10713</a> | 29 July 2020 | Moderate  | 8.2    | 3       | Not defined | Not reviewed |
| <a href="#">CVE-2020-14311</a> | 29 July 2020 | Moderate  | 5.7    | 2       | Not defined | Not reviewed |
| <a href="#">CVE-2020-14310</a> | 29 July 2020 | Moderate  | 5.7    | 2       | Not defined | Not reviewed |

Abbildung 16.38: Bericht des Vulnerability Service

Für jedes CVE bietet Insights zusätzliche Informationen und listet die betroffenen Systeme auf. Sie können auf **Remediate** klicken, um ein Ansible Playbook für die Wiederherstellung zu erstellen.

The screenshot shows the Red Hat Insights interface. On the left, a sidebar menu includes 'Red Hat Insights' (selected), 'Dashboard', 'Advisor', 'Vulnerability' (selected), 'CVEs' (selected), 'Systems', 'Compliance', 'Patch', 'Drift', 'Policies', 'Inventory', and 'Remediations'. The main content area is titled 'CVE-2020-14352'. It shows a brief description of a directory traversal vulnerability in librepo, mentioning it failed to sanitize paths in remote repository metadata. An attacker could copy files outside the destination directory via path traversal. The highest threat is to users of untrusted third-party repositories. The 'Business risk' is 'Not defined' and 'Status' is 'Not reviewed'. A prominent orange box indicates 'Important severity' with a warning icon. The 'CVSS 3.0 base score' is 8.0. Below this, the 'CVSS 3.0 vector' is listed as CVSS:3.1/AV:N/AC:L/PR:L/UI:R/S:U/C:H/I:H/A:H. A table titled 'Exposed systems' lists one system: 'host.example.com' with status 'Not reviewed' and last seen '1 hour ago'. A 'Remediate' button is available for this system.

Abbildung 16.39: Details einer CVE

## Analysieren der Compliance mit dem Compliance-Service

Der Compliance-Service analysiert Ihre Systeme und meldet deren Compliance-Stufe einer OpenSCAP-Richtlinie. Das OpenSCAP-Projekt implementiert Tools, um die Compliance eines Systems anhand einer Reihe von Regeln zu überprüfen. Insights bietet Regeln, mit denen Ihre Systeme anhand verschiedener Richtlinien bewertet werden, z. B. den *Payment Card Industry Data Security Standard (PCI DSS)*.

## Aktualisieren von Paketen mit dem Patch-Service

Der Patch-Service listet die Red Hat-Produktempfehlungen auf, die für Ihre Systeme gelten. Er kann auch ein Ansible Playbook generieren, das Sie ausführen können, um die RPM-Pakete zu aktualisieren, die den entsprechenden Empfehlungen zugeordnet sind. Um auf die Liste der Empfehlungen für ein bestimmtes System zuzugreifen, verwenden Sie das Menü **Patch** → **Systems**. Klicken Sie auf **Apply all applicable advisories**, damit ein System das Ansible Playbook generiert.

Abbildung 16.40: Patchen eines Systems

## Vergleichen von Systemen mit dem Drift-Service

Mit dem Drift-Service können Sie Systeme oder einen Systemverlauf vergleichen. Dieser Service kann Ihnen bei der Fehlerbehebung für ein System helfen, indem dieses System mit einem ähnlichen System oder einem früheren Systemstatus verglichen wird. Sie greifen über das Menü **Drift → Comparison** auf den Service zu.

Abbildung 16.41: Vergleichen eines Systemverlaufs

Der vorangegangene Screenshot zeigt, wie Insights das gleiche System zu zwei unterschiedlichen Zeiten vergleicht.

## Auslösen von Alarmen mit dem Policies-Service

Mit dem Policies-Service erstellen Sie Regeln zum Überwachen der Systeme und Senden von Warnmeldungen, wenn ein System Ihre Regeln nicht erfüllt. Insights bewertet die Regeln jedes Mal, wenn ein System seine Metadaten synchronisiert. Sie greifen über das Menü **Policies** auf den Service zu.

The screenshot shows the Red Hat Insights interface with the 'Policies' menu selected. A single policy is listed:

| Name   | Trigger actions | Last triggered |
|--------|-----------------|----------------|
| No GCC | Send Email      | Never          |

**Description:** Send an alert email when the gcc package is installed.  
Last updated 18 Sep 2020 | Created 18 Sep 2020

**Conditions:** facts.installed\_packages contains ['gcc']  
**Trigger actions:** Send Email

Abbildung 16.42: Details einer benutzerdefinierten Regel

## Zugriff auf das Inventory und die Remediation Playbooks und Überwachen von Subskriptionen

Die Seite **Inventory** enthält eine Liste der Systeme, die Sie bei Red Hat Insights registriert haben. In der Spalte **Last seen** wird der Zeitpunkt der letzten Metadatenaktualisierung für jedes System angezeigt. Wenn Sie auf einen Systemnamen klicken, können Sie dessen Details überprüfen und direkt auf die Advisor-, Vulnerability-, Compliance- und Patch-Services für dieses System zugreifen.

Auf der Seite **Remediations** werden alle Ansible Playbooks aufgeführt, die Sie für die Wiederherstellung erstellt haben. Sie können die Playbooks von dieser Seite herunterladen.

Über die Seite **Subscription Watch** können Sie die Nutzung Ihrer Red Hat-Subskriptionen überwachen.



## Literaturhinweise

Manpages **insights-client(8)** und **insights-client.conf(5)**

Weitere Informationen zu Red Hat Insights finden Sie in der *Produktdokumentation zu Red Hat Insights* unter  
[https://access.redhat.com/documentation/en-us/red\\_hat\\_insights](https://access.redhat.com/documentation/en-us/red_hat_insights)

Weitere Informationen über das Ausschließen von Daten, die von Insights gesammelt wurden, finden Sie in den Kapiteln *Red Hat Insights client data obfuscation* und *Red Hat Insights client data redaction* im *Client Configuration Guide for Red Hat Insights* unter  
[https://access.redhat.com/documentation/en-us/red\\_hat\\_insights/2020-04/html-single/client\\_configuration\\_guide\\_for\\_red\\_hat\\_insights/index](https://access.redhat.com/documentation/en-us/red_hat_insights/2020-04/html-single/client_configuration_guide_for_red_hat_insights/index)

Informationen zu den von Red Hat Insights gesammelten Daten finden Sie hier:  
**Von Red Hat Insights gesammelte Systeminformationen**  
<https://access.redhat.com/articles/1598863>

## ► Quiz

# Erkennen und Lösen von Problemen mit Red Hat Insights

Wählen Sie die richtigen Antworten auf die folgenden Fragen aus:

► 1. In welcher Reihenfolge treten die folgenden Ereignisse beim Verwalten eines Red Hat Enterprise Linux-Systems mit Red Hat Insights auf?

1. Red Hat Insights analysiert Systemmetadaten, um zu ermitteln, welche Probleme und Empfehlungen zutreffen.
  2. Der Insights-Client lädt Systemmetadaten zum Red Hat Insights-Service hoch.
  3. Der Administrator sieht sich die empfohlenen Maßnahmen im Red Hat Insights Customer Portal an.
  4. Insights sammelt Systemmetadaten zum Red Hat Enterprise Linux-System.
- a. 1, 2, 3, 4
  - b. 4, 2, 1, 3
  - c. 4, 2, 3, 1
  - d. 4, 1, 2, 3

► 2. Mit welchem Befehl wird ein Client bei Red Hat Insights registriert?

- a. `insights-client --register`
- b. `insights-client --no-upload`
- c. `subscription-manager register`
- d. `insights-client --unregister`

► 3. Auf welcher Seite der Red Hat Insights-Konsole können Sie ein Ansible Playbook erstellen, um die RPM-Pakete auf einem System zu aktualisieren?

- a. **Advisor** → **Recommendations**
- b. **Vulnerability** → **Systems**
- c. **Patch** → **Systems**
- d. **Fehlerbehebung**

## ► Lösung

# Erkennen und Lösen von Problemen mit Red Hat Insights

Wählen Sie die richtigen Antworten auf die folgenden Fragen aus:

► 1. In welcher Reihenfolge treten die folgenden Ereignisse beim Verwalten eines Red Hat Enterprise Linux-Systems mit Red Hat Insights auf?

1. Red Hat Insights analysiert Systemmetadaten, um zu ermitteln, welche Probleme und Empfehlungen zutreffen.
  2. Der Insights-Client lädt Systemmetadaten zum Red Hat Insights-Service hoch.
  3. Der Administrator sieht sich die empfohlenen Maßnahmen im Red Hat Insights Customer Portal an.
  4. Insights sammelt Systemmetadaten zum Red Hat Enterprise Linux-System.
- a. 1, 2, 3, 4  
b. 4, 2, 1, 3  
c. 4, 2, 3, 1  
d. 4, 1, 2, 3

► 2. Mit welchem Befehl wird ein Client bei Red Hat Insights registriert?

- a. `insights-client --register`
- b. `insights-client --no-upload`
- c. `subscription-manager register`
- d. `insights-client --unregister`

► 3. Auf welcher Seite der Red Hat Insights-Konsole können Sie ein Ansible Playbook erstellen, um die RPM-Pakete auf einem System zu aktualisieren?

- a. Advisor → Recommendations
- b. Vulnerability → Systems
- c. Patch → Systems
- d. Fehlerbehebung

# Zusammenfassung

---

In diesem Kapitel erhielten Sie folgende Informationen:

- Web Console ist eine webbasierte Management-Benutzeroberfläche für Ihren Server, die auf dem Open Source-Service Cockpit basiert.
- Web Console enthält Diagramme der Systemleistung, grafische Tools zum Verwalten der Systemkonfiguration und zum Überprüfen von Protokollen sowie eine interaktive Terminalsschnittstelle.
- Über das Red Hat Customer Portal haben Sie Zugriff auf Dokumentation, Downloads, Optimierungstools, Support-Ticket-Management sowie Subskriptions- und Berechtigungsmanagement für Ihre Red Hat-Produkte.
- **redhat-support-tool** ist ein Befehlszeilentool zum Abfragen der Knowledgebase und zum Arbeiten mit Support-Tickets über die Befehlszeile des Servers.
- Red Hat Insights ist ein SaaS-basiertes Predictive Analytics-Tool, mit dem Sie Bedrohungen für die Sicherheit, Leistung, Verfügbarkeit und Stabilität Ihrer Systems erkennen und beheben können.



## Kapitel 17

# Ausführliche Wiederholung

### Ziel

Wiederholen von Aufgaben aus *Red Hat System Administration I*

### Zielsetzungen

- Wiederholen von Aufgaben aus *Red Hat System Administration I*

### Abschnitte

- Ausführliche Wiederholung

### Praktische Übung

- Praktische Übung: Verwalten von Dateien über die Befehlszeile
- Praktische Übung: Verwalten von Benutzern und Gruppen, Berechtigungen und Prozessen
- Praktische Übung: Konfigurieren und Verwalten eines Servers
- Praktische Übung: Verwalten von Netzwerken
- Praktische Übung: Mounten von Dateisystemen und Suchen von Dateien

# Ausführliche Wiederholung

---

## Ziele

In diesem Abschnitt werden die in *Red Hat System Administration I* vermittelten Informationen und Kenntnisse wiederholt und aufgefrischt.

## Wiederholung von Red Hat System Administration I

Bevor Teilnehmer mit der ausführlichen Wiederholung für diesen Kurs beginnen, sollten sie mit den in den jeweiligen Kapiteln behandelten Themen vertraut sein.

Für zusätzliche Übungen stehen Teilnehmern auch die vorherigen Kapitel dieses Lehrbuchs zur Verfügung.

### Kapitel 1, Erste Schritte mit Red Hat Enterprise Linux

Beschreiben und Definieren von Open Source, Linux, Linux-Distributionen und Red Hat Enterprise Linux

- Definieren und Erläutern des Zwecks von Linux, Open Source, Linux-Distributionen und Red Hat Enterprise Linux

### Kapitel 2, Zugreifen auf die Befehlszeile

Anmelden bei einem Linux-System und Ausführen einfacher Befehle über die Shell

- Anmelden bei einem Linux-System auf einer lokalen Textkonsole und Ausführen einfacher Befehle über die Shell
- Anmelden mit der GNOME 3-Desktopumgebung bei einem Linux-System und Ausführen von Befehlen an einer Shell-Eingabeaufforderung in einem Terminalprogramm
- Zeitsparende Verwendung von Tab-Vervollständigung, Befehlsverlauf und Tastenkombinationen für die Befehlszeilenbearbeitung zum Ausführen von Befehlen in der Bash-Shell

### Kapitel 3, Verwalten von Dateien über die Befehlszeile

Kopieren, Verschieben, Erstellen, Löschen und Organisieren von Dateien über die Bash-Shell-Eingabeaufforderung

## Kapitel 17 | Ausführliche Wiederholung

- Beschreiben, wie Linux Dateien organisiert, und Erläutern des Zwecks verschiedener Verzeichnisse in der Dateisystemhierarchie
- Angeben des Speicherorts von Dateien relativ zum aktuellen Arbeitsverzeichnis und nach absolutem Speicherort, Bestimmen und Ändern Ihres Arbeitsverzeichnisses sowie Auflisten des Inhalts von Verzeichnissen
- Erstellen, Kopieren, Verschieben und Entfernen von Dateien und Verzeichnissen
- Festlegen, dass mehrere Dateinamen die gleiche Datei referenzieren, unter Verwendung von Hardlinks und symbolischen Verknüpfungen (oder „Softlinks“)
- Effizientes Ausführen von Befehlen, die sich auf viele Dateien auswirken, unter Verwendung der Mustervergleichsfunktionen der Bash-Shell

## **Kapitel 4, Abrufen von Hilfe in Red Hat Enterprise Linux**

Beheben von Problemen durch die Verwendung von lokalen Hilfesystemen

- Finden von Informationen auf den Handbuchseiten des lokalen Linux-Systems
- Finden von Informationen in der lokalen Dokumentation in GNU Info

## **Kapitel 5, Erstellen, Anzeigen und Bearbeiten von Textdateien**

Erstellen, Anzeigen und Bearbeiten von Textdateien über die Befehlsausgabe oder in einem Editor

- Speichern der Befehlsausgabe oder Fehler in einer Datei mit Shell-Umleitung und Verarbeiten der Befehlsausgabe über mehrere Befehlszeilenprogramme mit Pipes
- Erstellen und Bearbeiten von Textdateien mit dem **vim**-Editor
- Verwenden von Shell-Variablen zur Ausführung von Befehlen und Bearbeiten von Bash-Startskripts zur Festlegung von Shell- und Umgebungsvariablen, um das Verhalten der Shell und von in der Shell ausgeführten Programmen zu ändern

## **Kapitel 6, Verwalten lokaler Benutzer und Gruppen**

Erstellen, Verwalten und Löschen lokaler Benutzer und Gruppen und Verwalten lokaler Passwortrichtlinien

- Beschreiben des Zwecks von Benutzern und Gruppen auf einem Linux-System
- Wechseln zum Superuser-Konto, um ein Linux-System zu verwalten, und anderen Benutzern mit dem Befehl **sudo** Zugriff als Superuser zu gewähren
- Erstellen, Ändern und Löschen lokal definierter Benutzerkonten
- Erstellen, Ändern und Löschen lokal definierter Gruppenkonten
- Festlegen einer Passwortverwaltungsrichtlinie für Benutzer und manuelles Sperren und Entsperren von Benutzerkonten

## **Kapitel 7, Steuern des Zugriffs auf Dateien**

Einrichten von Linux-Dateisystemberechtigungen für Dateien und Interpretieren der Sicherheitseffekte verschiedener Berechtigungseinstellungen

- Auflisten der Dateisystemberechtigungen für Dateien und Verzeichnisse und Interpretieren der Auswirkungen dieser Berechtigungen auf den Zugriff von Benutzern und Gruppen

## Kapitel 17 | Ausführliche Wiederholung

- Ändern der Berechtigungen und Eigentümerschaft von Dateien mit Befehlszeilertools
- Steuern der Standardberechtigungen neuer, von Benutzern erstellter Dateien, Erläutern der Auswirkungen besonderer Berechtigungen und Verwenden spezieller Berechtigungen und Standardberechtigungen zum Festlegen des Gruppeneigentümers von in einem bestimmten Verzeichnis erstellten Dateien

## Kapitel 8, Überwachen und Verwalten von Linux-Prozessen

Evaluieren und Steuern von auf einem Red Hat Enterprise Linux-System ausgeführten Prozessen

- Abrufen von Informationen zu auf dem System ausgeführten Programmen zum Ermitteln und Steuern von Status, Ressourcennutzung und Eigentümerschaft
- Verwalten mehrerer, von derselben Terminalsitzung gestarteter Prozesse mit der Bash-Jobsteuerung
- Steuern und Beenden von nicht mit der Shell verbundenen Prozessen und erzwungenes Beenden von Benutzersitzungen und -prozessen
- Beschreiben der durchschnittlichen Systemauslastung und Ermitteln von Prozessen mit hohem Ressourcenverbrauch auf einem Server

## Kapitel 9, Steuern von Services und Daemons

Steuern und Überwachen von Netzwerkservices und System-Daemons mit Systemd

- Auflisten der System-Daemons und Netzwerkservices, die von **systemd**-Service- und Socket-Units gestartet wurden
- Steuern von System-Daemons und Netzwerkservices mit **systemctl**

## Kapitel 10, Konfigurieren und Sichern von SSH

Konfigurieren von sicherem Befehlszeilenservice auf Remote-Systemen mit OpenSSH

- Anmelden bei einem Remote-System und Ausführen von Befehlen mit **ssh**
- Konfigurieren der schlüsselbasierten Authentifizierung für ein Benutzerkonto zur sicheren und passwortlosen Anmeldung bei Remote-Systemen
- Einschränken der direkten Anmeldung als root und Deaktivieren der passwortbasierten Authentifizierung für den OpenSSH-Service

## Kapitel 11, Analysieren und Speichern von Protokollen

Suchen und Auswerten von Systemprotokolldateien zur Fehlerbehebung.

- Beschreiben der grundlegenden Protokollierungsarchitektur, die von Red Hat Enterprise Linux zum Aufzeichnen von Ereignissen verwendet wird.
- Interpretieren von Ereignissen in relevanten Systemprotokolldateien zur Fehlerbehebung oder Prüfung des Systemstatus.
- Suchen und Interpretieren von Einträgen im Systemjournal zur Fehlerbehebung oder Prüfung des Systemstatus.
- Konfigurieren des Systemjournals, sodass die Aufzeichnung von Ereignissen beim Neubooten eines Servers erhalten bleibt.

- Beibehalten der genauen Zeitsynchronisierung mithilfe von NTP und konfigurieren der Zeitzone, um korrekte Zeitstempel für Ereignisse zu gewährleisten, die vom Systemjournal und den Protokollen aufgezeichnet werden.

## **Kapitel 12, Netzwerkmanagement**

Konfigurieren von Netzwerkschnittstellen und Einstellungen auf Red Hat Enterprise Linux - Servern

- Beschreiben der grundlegenden Konzepte der Netzwerkadressierung und des Routings für einen Server
- Testen und Prüfen der aktuellen Netzwerkkonfiguration mit Befehlszeilenprogrammen
- Verwalten von Netzwerkeinstellungen und Geräten mit **nmcli**
- Ändern von Netzwerkeinstellungen durch Bearbeiten der Konfigurationsdateien
- Konfigurieren des statischen Hostnamens eines Servers und der Namensauflösung sowie Testen der Ergebnisse

## **Kapitel 13, Archivieren und Übertragen von Dateien**

Archivieren Sie Dateien und kopieren Sie sie zwischen Systemen.

- Archivieren Sie Dateien und Verzeichnisse in einer komprimierten Datei mit tar und extrahieren Sie den Inhalt eines vorhandenen tar-Archivs.
- Übertragen Sie Dateien mit SSH zu oder von einem Remote-System.
- Synchronisieren Sie die Inhalte einer lokalen Datei oder eines Verzeichnisses mit einer Kopie auf einem Remote-Server.

## **Kapitel 14, Installieren und Aktualisieren von Softwarepaketen**

Laden, installieren, aktualisieren und verwalten Sie Softwarepakete von Red Hat und YUM-Paket-Repositorys.

- Registrieren eines Systems bei Ihrem Red Hat-Benutzerkonto und Zuweisen von Berechtigungen für Software-Updates und Support-Services mit Red Hat Subscription Management
- Erläutern, wie Software als RPM-Pakete bereitgestellt wird, und Untersuchen der auf dem System installierten Pakete mit YUM und RPM
- Suchen, Installieren und Aktualisieren der Softwarepakete mit dem **yum**-Befehl
- Aktivieren und Deaktivieren der YUM-Repositorys von Red Hat oder Drittanbietern über einen Server
- Erläutern, wie Module die Installation bestimmter Softwareversionen ermöglichen, Auflisten, Aktivieren und Wechseln von Modul-Streams sowie Installieren und Aktualisieren von Paketen aus einem Modul

## **Kapitel 15, Zugriff auf Linux-Dateisysteme**

Zugreifen auf sowie Prüfen und Verwenden von vorhandenen Dateisystemen auf an einen Linux-Server angeschlossenen Storage.

## Kapitel 17 | Ausführliche Wiederholung

- Erläutern, was ein Blockgerät ist, Interpretieren der Dateinamen von Speichergeräten und Identifizieren des vom Dateisystem für ein bestimmtes Verzeichnis oder eine bestimmte Datei verwendeten Speichergeräts.
- Zugreifen auf Dateisysteme durch Anhängen an ein Verzeichnis in der Dateisystemhierarchie.
- Suchen nach Dateien in gemounteten Dateisystemen mit den Befehlen **find** und **locate**.

## Kapitel 16, Analysieren von Servern und Erhalten von Unterstützung

Untersuchen und Lösen von Problemen in der webbasierten Managementoberfläche, Erhalten von Unterstützung bei der Lösung von Problemen von Red Hat.

- Aktivieren der Web Console-Managementoberfläche zur Remote-Verwaltung und - Überwachung der Leistung eines Red Hat Enterprise Linux-Servers.
- Beschreiben wichtiger Ressourcen, die über das Red Hat Customer Portal verfügbar sind, und Suchen nach Informationen in der Dokumentation und der Knowledgebase von Red Hat.
- Überprüfen von Servern auf Probleme, Beheben oder Lösen der Probleme und Bestätigen der Lösung durch Red Hat Insights.

## ► Praktische Übung

# Verwalten von Dateien über die Befehlszeile

In dieser Überprüfung verwalten Sie Dateien, leiten eine bestimmte Gruppe von Zeilen aus einer Textdatei in eine andere Datei um und bearbeiten die Textdateien.

### Ergebnisse

Sie sollten über die folgenden Fähigkeiten verfügen:

- Verwalten von Dateien über die Befehlszeile
- Anzeigen einer bestimmten Anzahl von Zeilen aus Textdateien und Umleiten der Ausgabe in eine andere Datei
- Bearbeiten von Textdateien

### Bevor Sie Beginnen

Kopieren Sie alle Dateien oder Arbeiten, die Sie behalten möchten, vor dem Zurücksetzen auf andere Systeme. Setzen Sie jetzt die Systeme **workstation**, **servera** und **serverb** zurück. Warten Sie, bis die Systeme **workstation**, **servera** und **serverb** gestartet sind.

Melden Sie sich als Benutzer **student** mit dem Passwort **student** bei **workstation** an.

Führen Sie **lab rhcsa-rh124-review1 start** auf **workstation** aus, um die ausführliche Überprüfung zu beginnen. Dieses Skript erstellt die erforderlichen Dateien, um die Umgebung korrekt einzurichten.

```
[student@workstation ~]$ lab rhcsa-rh124-review1 start
```

### Anweisungen

Führen Sie die folgenden Aufgaben auf **serverb** aus, um die Übung abzuschließen.

- Erstellen Sie ein neues Verzeichnis namens **/home/student/grading**.
- Erstellen Sie drei leere Dateien im Verzeichnis **/home/student/grading: grade1**, **grade2** und **grade3**.
- Erfassen Sie die ersten fünf Zeilen der Datei **/home/student/bin/manage-files** in der Datei **/home/student/grading/manage-files.txt**.
- Hängen Sie die letzten drei Zeilen von **/home/student/bin/manage-files** an die Datei **/home/student/grading/manage-files.txt** an. Sie dürfen den in der Datei **/home/student/grading/manage-files.txt** bereits vorhandenen Text nicht überschreiben.
- Kopieren Sie **/home/student/grading/manage-files.txt** nach **/home/student/grading/manage-files-copy.txt**.
- Bearbeiten Sie die Datei **/home/student/grading/manage-files-copy.txt** so, dass sie zwei aufeinander folgende Zeilen mit dem Text **Test JJ** enthält.

## Kapitel 17 | Ausführliche Wiederholung

- Bearbeiten Sie die Datei **/home/student/grading/manage-files-copy.txt** so, dass die Textzeile **Test HH** nicht in der Datei vorhanden ist.
- Bearbeiten Sie die Datei **/home/student/grading/manage-files-copy.txt** so, dass die Zeile **A new line** zwischen den Zeilen **Test BB** und **Test CC** vorhanden ist.
- Erstellen Sie einen Hardlink namens **/home/student/grading/grade1** zur Datei **/home/student/hardlink**. Sie müssen dies tun, nachdem Sie die leere Datei **/home/student/grading/grade1** wie oben beschrieben erstellt haben.
- Erstellen Sie einen Softlink namens **/home/student/grading/grade2** zur Datei **/home/student/softlink**.
- Speichern Sie die Ausgabe eines Befehls, der den Inhalt des Verzeichnisses **/boot** auflistet, in der Datei **/home/student/grading/longlisting.txt**. Die Ausgabe sollte eine „lange Auflistung“ sein, die Dateiberechtigungen, Besitzer und Gruppenbesitzer, Größe und Änderungsdatum jeder Datei enthält.

## Bewertung

Führen Sie auf **workstation** den Befehl **lab rhcsa-rh124-review1 grade** aus, um den Erfolg dieser Übung zu bestätigen.

```
[student@workstation ~]$ lab rhcsa-rh124-review1 grade
```

## Beenden

Führen Sie **lab rhcsa-rh124-review1 finish** auf **workstation** aus, um die ausführliche Überprüfung abzuschließen. Dieses Skript löscht die Dateien und Verzeichnisse, die beim Start der ausführlichen Überprüfung erstellt wurden, und stellt sicher, dass die Umgebung auf **serverb** sauber ist.

```
[student@workstation ~]$ lab rhcsa-rh124-review1 finish
```

Damit ist die ausführliche Überprüfung abgeschlossen.

## ► Lösung

# Verwalten von Dateien über die Befehlszeile

In dieser Überprüfung verwalten Sie Dateien, leiten eine bestimmte Gruppe von Zeilen aus einer Textdatei in eine andere Datei um und bearbeiten die Textdateien.

## Ergebnisse

Sie sollten über die folgenden Fähigkeiten verfügen:

- Verwalten von Dateien über die Befehlszeile
- Anzeigen einer bestimmten Anzahl von Zeilen aus Textdateien und Umleiten der Ausgabe in eine andere Datei
- Bearbeiten von Textdateien

## Bevor Sie Beginnen

Kopieren Sie alle Dateien oder Arbeiten, die Sie behalten möchten, vor dem Zurücksetzen auf andere Systeme. Setzen Sie jetzt die Systeme **workstation**, **servera** und **serverb** zurück. Warten Sie, bis die Systeme **workstation**, **servera** und **serverb** gestartet sind.

Melden Sie sich als Benutzer **student** mit dem Passwort **student** bei **workstation** an.

Führen Sie **lab rhcsa-rh124-review1 start** auf **workstation** aus, um die ausführliche Überprüfung zu beginnen. Dieses Skript erstellt die erforderlichen Dateien, um die Umgebung korrekt einzurichten.

```
[student@workstation ~]$ lab rhcsa-rh124-review1 start
```

## Anweisungen

Führen Sie die folgenden Aufgaben auf **serverb** aus, um die Übung abzuschließen.

- Erstellen Sie ein neues Verzeichnis namens **/home/student/grading**.
- Erstellen Sie drei leere Dateien im Verzeichnis **/home/student/grading: grade1**, **grade2** und **grade3**.
- Erfassen Sie die ersten fünf Zeilen der Datei **/home/student/bin/manage-files** in der Datei **/home/student/grading/manage-files.txt**.
- Hängen Sie die letzten drei Zeilen von **/home/student/bin/manage-files** an die Datei **/home/student/grading/manage-files.txt** an. Sie dürfen den in der Datei **/home/student/grading/manage-files.txt** bereits vorhandenen Text nicht überschreiben.
- Kopieren Sie **/home/student/grading/manage-files.txt** nach **/home/student/grading/manage-files-copy.txt**.
- Bearbeiten Sie die Datei **/home/student/grading/manage-files-copy.txt** so, dass sie zwei aufeinander folgende Zeilen mit dem Text **Test JJ** enthält.

**Kapitel 17 |** Ausführliche Wiederholung

- Bearbeiten Sie die Datei **/home/student/grading/manage-files-copy.txt** so, dass die Textzeile **Test HH** nicht in der Datei vorhanden ist.
- Bearbeiten Sie die Datei **/home/student/grading/manage-files-copy.txt** so, dass die Zeile **A new line** zwischen den Zeilen **Test BB** und **Test CC** vorhanden ist.
- Erstellen Sie einen Hardlink namens **/home/student/grading/grade1** zur Datei **/home/student/hardlink**. Sie müssen dies tun, nachdem Sie die leere Datei **/home/student/grading/grade1** wie oben beschrieben erstellt haben.
- Erstellen Sie einen Softlink namens **/home/student/grading/grade2** zur Datei **/home/student/softlink**.
- Speichern Sie die Ausgabe eines Befehls, der den Inhalt des Verzeichnisses **/boot** auflistet, in der Datei **/home/student/grading/longlisting.txt**. Die Ausgabe sollte eine „lange Auflistung“ sein, die Dateiberechtigungen, Besitzer und Gruppenbesitzer, Größe und Änderungsdatum jeder Datei enthält.

**1.** Erstellen Sie ein neues Verzeichnis namens **/home/student/grading**.1.1. Öffnen Sie auf **workstation** als **student** eine SSH-Sitzung zu **serverb**.

```
[student@workstation ~]$ ssh student@serverb  
...output omitted...  
[student@serverb ~]$
```

1.2. Erstellen Sie mit dem Befehl **mkdir** das Verzeichnis **/home/student/grading**.

```
[student@serverb ~]$ mkdir grading
```

Da Sie den vorhergehenden Befehl im Benutzerverzeichnis von **student** ausgeführt haben, haben Sie beim Erstellen des Verzeichnisses **grading** nicht den absoluten Pfad dazu angegeben.

**2.** Erstellen Sie drei leere Dateien im Verzeichnis **/home/student/grading**: **grade1**, **grade2** und **grade3**.2.1. Erstellen Sie mit dem Befehl **touch** die leeren Dateien **grade1**, **grade2** und **grade3** im Verzeichnis **/home/student/grading**. Wenden Sie die Shell-Funktion zur Klammererweiterung an, um alle drei Dateien mit einem einzigen **touch**-Befehl zu erstellen.

```
[student@serverb ~]$ touch grading/grade{1,2,3}
```

2.2. Überprüfen Sie mithilfe des Befehls **ls**, ob die Dateien **grade1**, **grade2** und **grade3** im Verzeichnis **/home/student/grading** vorhanden sind.

```
[student@serverb ~]$ ls grading/  
grade1 grade2 grade3
```

**3.** Erfassen Sie die ersten fünf Zeilen der Datei **/home/student/bin/manage-files** in der Datei **/home/student/grading/manage-files.txt**.

- 3.1. Verwenden Sie den Befehl **head**, um die ersten fünf Zeilen der Datei **/home/student/bin/manage-files** anzuzeigen und die Ausgabe in die Datei **/home/student/grading/manage-files.txt** umzuleiten.

```
[student@serverb ~]$ head -5 bin/manage-files > grading/manage-files.txt
```

Der vorhergehende Befehl verwendet das Symbol für einfache Umleitung (**>**), um die Befehlausgabe in **/home/student/grading/manage-files.txt** zu speichern, wodurch alle vorhandene Inhalte in der Datei überschrieben werden.

- 3.2. Überprüfen Sie, ob die Datei **/home/student/grading/manage-files.txt** den folgenden Text enthält.

```
Test AA  
Test BB  
Test CC  
Test DD  
Test EE
```

4. Hängen Sie die letzten drei Zeilen von **/home/student/bin/manage-files** an die Datei **/home/student/grading/manage-files.txt** an. Sie dürfen den in der Datei **/home/student/grading/manage-files.txt** bereits vorhandenen Text nicht überschreiben.

- 4.1. Verwenden Sie den Befehl **tail**, um die letzten drei Zeilen der Datei **/home/student/bin/manage-files** anzuzeigen und die Ausgabe an **/home/student/grading/manage-files.txt** anzuhängen.

```
[student@serverb ~]$ tail -3 bin/manage-files >> grading/manage-files.txt
```

Der vorhergehende Befehl verwendet das Symbol für doppelte Umleitung (**>>**), um die Ausgabe an **/home/student/grading/manage-files.txt** anzuhängen, wodurch die vorhandenen Inhalte in der Datei beibehalten werden.

- 4.2. Überprüfen Sie, ob die Datei **/home/student/grading/manage-files.txt** den folgenden Text enthält.

```
Test AA  
Test BB  
Test CC  
Test DD  
Test EE  
Test HH  
Test II  
Test JJ
```

5. Kopieren Sie die Datei **/home/student/grading/manage-files.txt** nach **/home/student/grading/manage-files-copy.txt**.

- 5.1. Navigieren Sie mit dem Befehl **cd** zum Verzeichnis **/home/student/grading**.

```
[student@serverb ~]$ cd grading/  
[student@serverb grading]$
```

**Kapitel 17 |** Ausführliche Wiederholung

- 5.2. Verwenden Sie den Befehl **cp**, um die Datei **/home/student/grading/manage-files.txt** nach **/home/student/grading/manage-files-copy.txt** zu kopieren.

```
[student@serverb grading]$ cp manage-files.txt manage-files-copy.txt
```

- 5.3. Navigieren Sie zurück zum Benutzerverzeichnis des Benutzers **student**.

```
[student@serverb grading]$ cd  
[student@serverb ~]$
```

6. Bearbeiten Sie die Datei **/home/student/grading/manage-files-copy.txt** so, dass sie zwei aufeinander folgende Zeilen mit dem Text **Test JJ** enthält.

- 6.1. Öffnen Sie die Datei **/home/student/grading/manage-files-copy.txt** im **vim**-Texteditor.

```
[student@serverb ~]$ vim grading/manage-files-copy.txt
```

- 6.2. Scrollen Sie im Befehlsmodus in **vim** nach unten zu der Zeile mit der Textzeile **Test JJ**. Drücken Sie zweimal die Taste **y** auf der Tastatur, um die Textzeile zu kopieren, und drücken Sie die Taste **p**, um die Textzeile unter dem Cursor einzufügen. Geben Sie **:wq** ein, um die Änderungen zu speichern und **vim** zu beenden. Überprüfen Sie, ob die Datei **/home/student/grading/manage-files-copy.txt** den folgenden Text enthält.

```
Test AA  
Test BB  
Test CC  
Test DD  
Test EE  
Test HH  
Test II  
Test JJ  
Test JJ
```

Beachten Sie, dass der vorstehende Inhalt zwei Kopien der Textzeile **Test JJ** enthält.

7. Bearbeiten Sie die Datei **/home/student/grading/manage-files-copy.txt** so, dass die Textzeile **Test HH** nicht in der Datei vorhanden ist.

- 7.1. Öffnen Sie die Datei **/home/student/grading/manage-files-copy.txt** im **vim**-Texteditor.

```
[student@serverb ~]$ vim grading/manage-files-copy.txt
```

- 7.2. Scrollen Sie im Befehlsmodus in **vim** nach unten zu der Zeile mit der Textzeile **Test HH**. Drücken Sie zweimal die Taste **d** auf der Tastatur, um die Textzeile zu löschen. Geben Sie **:wq** ein, um die Änderungen zu speichern und **vim** zu beenden. Überprüfen Sie, ob die Datei **/home/student/grading/manage-files-copy.txt** den folgenden Text enthält.

```
Test AA  
Test BB  
Test CC  
Test DD  
Test EE  
Test II  
Test JJ  
Test JJ
```

Beachten Sie, dass der vorstehende Inhalt die Textzeile **Test HH** nicht enthält.

8. Bearbeiten Sie die Datei **/home/student/grading/manage-files-copy.txt** so, dass die Zeile **A new line** zwischen den Zeilen **Test BB** und **Test CC** vorhanden ist.

- 8.1. Öffnen Sie die Datei **/home/student/grading/manage-files-copy.txt** im **vim**-Texteditor.

```
[student@serverb ~]$ vim grading/manage-files-copy.txt
```

- 8.2. Scrollen Sie im Befehlsmodus in **vim** nach unten zu der Zeile mit der Textzeile **Test CC**. Drücken Sie die Taste **i** auf der Tastatur, um in den Einfügemodus zu wechseln; der Cursor muss dabei am Anfang der Textzeile **Test CC** bleiben. Drücken Sie im Einfügemodus die **Eingabetaste** auf der Tastatur, um eine leere Zeile über dem Cursor einzufügen. Navigieren Sie mit der Nach-oben-Pfeiltaste zu der leeren Zeile, und erstellen Sie die Textzeile **A new line**. Drücken Sie die **Esc**-Taste auf der Tastatur, um zum Befehlsmodus zurückzukehren. Geben Sie **:wq** ein, um die Änderungen zu speichern und **vim** zu beenden. Überprüfen Sie, ob die Datei **/home/student/grading/manage-files-copy.txt** den folgenden Text enthält.

```
Test AA  
Test BB  
A new line  
Test CC  
Test DD  
Test EE  
Test II  
Test JJ  
Test JJ
```

Beachten Sie, dass der vorstehende Inhalt die Textzeile **A new line** enthält.

9. Erstellen Sie einen Hardlink namens **/home/student/grading/grade1** zur Datei **/home/student/hardlink**.

- 9.1. Verwenden Sie den Befehl **ln**, um den Hardlink namens **/home/student/hardlink** zur Datei **/home/student/grading/grade1** zu erstellen. Sie müssen dies tun, nachdem Sie die leere Datei **/home/student/grading/grade1** wie oben beschrieben erstellt haben.

```
[student@serverb ~]$ ln grading/grade1 hardlink
```

- 9.2. Zeigen Sie mit dem Befehl **ls -l** die Linkanzahl der Datei **/home/student/grading/grade1** an.

## Kapitel 17 | Ausführliche Wiederholung

```
[student@serverb ~]$ ls -l grading/grade1  
-rw-rw-r--. 2 student student 0 Mar 6 16:45 grading/grade1
```

10. Erstellen Sie einen Softlink namens **/home/student/grading/grade2** zur Datei **/home/student/softlink**.

- 10.1. Verwenden Sie den Befehl **ln -s**, um den Softlink namens **/home/student/softlink** zur Datei **/home/student/grading/grade2** zu erstellen.

```
[student@serverb ~]$ ln -s grading/grade2 softlink
```

- 10.2. Führen Sie den Befehl **ls -l** aus, um die Eigenschaften des Softlinks **/home/student/softlink** anzuzeigen.

```
[student@serverb ~]$ ls -l softlink  
lrwxrwxrwx. 1 student student 14 Mar 6 17:58 softlink -> grading/grade2
```

11. Speichern Sie die Ausgabe eines Befehls, der den Inhalt des Verzeichnisses **/boot** auflistet, in der Datei **/home/student/grading/longlisting.txt**. Die Ausgabe sollte eine „lange Auflistung“ sein, die Dateiberechtigungen, Besitzer und Gruppenbesitzer, Größe und Änderungsdatum jeder Datei enthält.

- 11.1. Führen Sie den Befehl **ls -l** aus, um den Inhalt des Verzeichnisses **/boot** als „lange Auflistung“ anzuzeigen und die Ausgabe in die Datei **/home/student/grading/longlisting.txt** umzuleiten.

```
[student@serverb ~]$ ls -l /boot > grading/longlisting.txt
```

- 11.2. Melden Sie sich von **serverb** ab.

```
[student@serverb ~]$ exit  
logout  
Connection to serverb closed.
```

## Bewertung

Führen Sie auf **workstation** den Befehl **lab rhcsa-rh124-review1 grade** aus, um den Erfolg dieser Übung zu bestätigen.

```
[student@workstation ~]$ lab rhcsa-rh124-review1 grade
```

## Beenden

Führen Sie **lab rhcsa-rh124-review1 finish** auf **workstation** aus, um die ausführliche Überprüfung abzuschließen. Dieses Skript löscht die Dateien und Verzeichnisse, die beim Start der ausführlichen Überprüfung erstellt wurden, und stellt sicher, dass die Umgebung auf **serverb** sauber ist.

```
[student@workstation ~]$ lab rhcsa-rh124-review1 finish
```

Damit ist die ausführliche Überprüfung abgeschlossen.

## ► Praktische Übung

# Verwalten von Benutzern und Gruppen, Berechtigungen und Prozessen

In dieser Überprüfung verwalten Sie Benutzer- und Gruppenkonten, legen Berechtigungen für Dateien und Verzeichnisse fest und verwalten Prozesse.

### Ergebnisse

Sie sollten über die folgenden Fähigkeiten verfügen:

- Verwalten von Benutzern und Gruppen
- Festlegen von Berechtigungen für Dateien und Verzeichnisse
- Entfernen von Prozessen, die zu viel CPU beanspruchen

### Bevor Sie Beginnen

Melden Sie sich als Benutzer **student** mit dem Passwort **student** bei **workstation** an.

Führen Sie **lab rhcsa-rh124-review2 start** auf **workstation** aus, um mit der ausführlichen Überprüfung zu beginnen. Dieses Skript führt einen Prozess aus, der die maximalen CPU-Ressourcen beansprucht, und erstellt die erforderlichen Dateien zur korrekten Einrichtung der Umgebung.

```
[student@workstation ~]$ lab rhcsa-rh124-review2 start
```

### Anweisungen

Führen Sie die folgenden Aufgaben auf **serverb** aus, um die Übung abzuschließen.

- Beenden Sie den Prozess, der derzeit die meiste CPU-Zeit beansprucht.
- Erstellen Sie eine neue Gruppe namens **database** mit der GID **50000**.
- Erstellen Sie einen neuen Benutzer namens **dbuser1**, der die Gruppe **database** als eine seiner sekundären Gruppen verwendet. Das Passwort des Benutzers **dbuser1** sollte anfänglich auf **redhat** festgelegt sein. Konfigurieren Sie den Benutzer **dbuser1** so, dass bei der ersten Anmeldung die Änderung des Passworts erzwungen wird. Der Benutzer **dbuser1** sollte sein Passwort **10** Tage nach dem Tag der Passwortänderung erneut ändern können. Das Passwort von **dbuser1** sollte **30** Tage nach dem letzten Tag der Passwortänderung ablaufen.
- Konfigurieren Sie den Benutzer **dbuser1** so, dass er **sudo** zum Ausführen jedes Befehls als Superuser verwenden kann.
- Konfigurieren Sie für den Benutzer **dbuser1** die standardmäßige Umask **007**.
- Die Berechtigungen für **/home/student/grading/review2** sollten den Gruppenmitgliedern von **database** und dem Benutzer **student** erlauben, auf das Verzeichnis zuzugreifen und Inhalte darin zu erstellen. Alle anderen Benutzer sollten über Lese- und Ausführungsberechtigungen für das Verzeichnis verfügen. Stellen

Sie außerdem sicher, dass Benutzer nur ihre eigenen Dateien aus **/home/student/grading/review2** löschen dürfen, nicht jedoch Dateien im Besitz anderer Benutzer.

## Bewertung

Führen Sie auf **workstation** den Befehl **lab rhcsa-rh124-review2 grade** aus, um den Erfolg dieser Übung zu bestätigen.

```
[student@workstation ~]$ lab rhcsa-rh124-review2 grade
```

## Beenden

Führen Sie **lab rhcsa-rh124-review2 finish** auf **workstation** aus, um die ausführliche Überprüfung abzuschließen. Dieses Skript beendet den Prozess, löscht die Dateien und Verzeichnisse, die beim Start der ausführlichen Überprüfung erstellt wurden, und stellt sicher, dass die Umgebung auf **serverb** sauber ist.

```
[student@workstation ~]$ lab rhcsa-rh124-review2 finish
```

Damit ist die ausführliche Überprüfung abgeschlossen.

## ► Lösung

# Verwalten von Benutzern und Gruppen, Berechtigungen und Prozessen

In dieser Überprüfung verwalten Sie Benutzer- und Gruppenkonten, legen Berechtigungen für Dateien und Verzeichnisse fest und verwalten Prozesse.

## Ergebnisse

Sie sollten über die folgenden Fähigkeiten verfügen:

- Verwalten von Benutzern und Gruppen
- Festlegen von Berechtigungen für Dateien und Verzeichnisse
- Entfernen von Prozessen, die zu viel CPU beanspruchen

## Bevor Sie Beginnen

Melden Sie sich als Benutzer **student** mit dem Passwort **student** bei **workstation** an.

Führen Sie **lab rhcsa-rh124-review2 start** auf **workstation** aus, um mit der ausführlichen Überprüfung zu beginnen. Dieses Skript führt einen Prozess aus, der die maximalen CPU-Ressourcen beansprucht, und erstellt die erforderlichen Dateien zur korrekten Einrichtung der Umgebung.

```
[student@workstation ~]$ lab rhcsa-rh124-review2 start
```

## Anweisungen

Führen Sie die folgenden Aufgaben auf **serverb** aus, um die Übung abzuschließen.

- Beenden Sie den Prozess, der derzeit die meiste CPU-Zeit beansprucht.
- Erstellen Sie eine neue Gruppe namens **database** mit der GID **50000**.
- Erstellen Sie einen neuen Benutzer namens **dbuser1**, der die Gruppe **database** als eine seiner sekundären Gruppen verwendet. Das Passwort des Benutzers **dbuser1** sollte anfänglich auf **redhat** festgelegt sein. Konfigurieren Sie den Benutzer **dbuser1** so, dass bei der ersten Anmeldung die Änderung des Passworts erzwungen wird. Der Benutzer **dbuser1** sollte sein Passwort **10** Tage nach dem Tag der Passwortänderung erneut ändern können. Das Passwort von **dbuser1** sollte **30** Tage nach dem letzten Tag der Passwortänderung ablaufen.
- Konfigurieren Sie den Benutzer **dbuser1** so, dass er **sudo** zum Ausführen jedes Befehls als Superuser verwenden kann.
- Konfigurieren Sie für den Benutzer **dbuser1** die standardmäßige Umask **007**.
- Die Berechtigungen für **/home/student/grading/review2** sollten den Gruppenmitgliedern von **database** und dem Benutzer **student** erlauben, auf das Verzeichnis zuzugreifen und Inhalte darin zu erstellen. Alle anderen Benutzer sollten über Lese- und Ausführungsberechtigungen für das Verzeichnis verfügen. Stellen

Sie außerdem sicher, dass Benutzer nur ihre eigenen Dateien aus **/home/student/grading/review2** löschen dürfen, nicht jedoch Dateien im Besitz anderer Benutzer.

1. Beenden Sie den Prozess, der derzeit die meiste CPU-Zeit beansprucht.
  - 1.1. Öffnen Sie auf **workstation** als **student** eine SSH-Sitzung zu **serverb**.

```
[student@workstation ~]$ ssh student@serverb  
...output omitted...  
[student@serverb ~]$
```

- 1.2. Verwenden Sie den Befehl **top**, um den Echtzeit-Systemstatus anzuzeigen.

```
[student@serverb ~]$ top
```

- 1.3. Achten Sie in der interaktiven Schnittstelle von **top** auf die Spalte **%CPU** und bestätigen Sie, dass ein Prozess namens **dd** darin aufgeführt wird, der die meisten CPU-Ressourcen beansprucht.

```
...output omitted...  
PID USER PR NI VIRT RES SHR S %CPU %MEM TIME+ COMMAND  
2303 student 20 0 217048 944 876 R 99.7 0.1 100:11.64 dd  
...output omitted...
```

Beachten Sie den Prozess **dd** mit der PID **2303** in der vorhergehenden Ausgabe, der den Großteil der CPU-Ressourcen beansprucht: **99,7 %**. Die PID und der Prozentsatz des CPU-Ressourcenverbrauchs können in Ihrem System variieren.

- 1.4. Geben Sie in der interaktiven Schnittstelle von **top k** ein, um den Prozess **dd** mit der PID **2303** zu beenden, den Sie im vorherigen Schritt ermittelt haben. Wenn die in der Eingabeaufforderung angezeigte Standard-PID mit der des Prozesses übereinstimmt, der die Mehrheit der CPU-Ressourcen beansprucht, drücken Sie die **Eingabetaste** auf der Tastatur. Wenn sie nicht übereinstimmt, geben Sie die PID interaktiv an.

```
...output omitted...  
PID to signal/kill [default pid = 2303] Enter  
...output omitted...
```

- 1.5. Verwenden Sie das Standardsignal **SIGTERM**, um den Prozess zu beenden.

```
...output omitted...  
Send pid 2833 signal [15/sigterm] Enter  
...output omitted...
```

- 1.6. Drücken Sie in der interaktiven Schnittstelle die Taste **q** auf der Tastatur, um **top** zu beenden.
2. Erstellen Sie eine neue Gruppe namens **database** mit der GID **50000**.
  - 2.1. Wechseln Sie zum Benutzer **root**.

```
[student@serverb ~]$ sudo su -  
[sudo] password for student: student  
[root@serverb ~]#
```

- 2.2. Erstellen Sie mithilfe des Befehls **groupadd** eine neue Gruppe namens **database** mit der GID **50000**.

```
[root@serverb ~]# groupadd -g 50000 database
```

3. Erstellen Sie einen neuen Benutzer namens **dbuser1**, der die Gruppe **database** als eine seiner sekundären Gruppen verwendet. Legen Sie das anfängliche Passwort des Benutzers **dbuser1** auf **redhat** fest. Konfigurieren Sie den Benutzer **dbuser1** so, dass bei der ersten Anmeldung die Änderung des Passworts erzwungen wird. Der Benutzer **dbuser1** sollte sein Passwort **10** Tage nach dem letzten Tag der Passwortänderung erneut ändern können. Das Passwort von **dbuser1** sollte **30** Tage nach dem letzten Tag der Passwortänderung ablaufen.

- 3.1. Erstellen Sie mithilfe des Befehls **useradd** einen neuen Benutzer namens **dbuser1**, der die Gruppe **database** als eine seiner sekundären Gruppen verwendet.

```
[root@serverb ~]# useradd -G database dbuser1
```

- 3.2. Verwenden Sie den Befehl **passwd**, um das Passwort von **dbuser1** auf **redhat** festzulegen.

```
[root@serverb ~]# passwd dbuser1  
Changing password for user dbuser1.  
New password: redhat  
BAD PASSWORD: The password is shorter than 8 characters  
Retype new password: redhat  
passwd: all authentication tokens updated successfully.
```

- 3.3. Verwenden Sie den Befehl **chage**, um den Benutzer **dbuser1** bei der ersten Anmeldung zur Änderung seines Passworts zu zwingen.

```
[root@serverb ~]# chage -d 0 dbuser1
```

- 3.4. Verwenden Sie den Befehl **chage**, um das Mindestalter des Passworts von **dbuser1** auf **10** Tage festzulegen.

```
[root@serverb ~]# chage -m 10 dbuser1
```

- 3.5. Verwenden Sie den Befehl **chage**, um das Höchstalter des Passworts von **dbuser1** auf **30** Tage festzulegen.

```
[root@serverb ~]# chage -M 30 dbuser1
```

4. Erstellen Sie die Datei **/etc/sudoers.d/dbuser1**, um den **dbuser1** so zu konfigurieren, dass der Benutzer **sudo** verwenden kann, um jeden Befehl als Superuser auszuführen. Sie

können den Befehl **vim /etc/sudoers.d/dbuser1** verwenden, um die Datei zu erstellen. Die Datei **/etc/sudoers.d/dbuser1** sollte folgenden Inhalt haben.

4.1.

```
dbuser1 ALL=(ALL) ALL
```

5. Konfigurieren Sie für den Benutzer **dbuser1** die standardmäßige Umask **007**.

5.1. Wechseln Sie zum Benutzer **dbuser1**.

```
[root@serverb ~]# su - dbuser1  
[dbuser1@serverb ~]$
```

5.2. Hängen Sie die Zeile **umask 007** an die Dateien **/home/dbuser1/.bash\_profile** und **/home/dbuser1/.bashrc** an.

```
[dbuser1@serverb ~]$ echo "umask 007" >> .bash_profile  
[dbuser1@serverb ~]$ echo "umask 007" >> .bashrc
```

5.3. Beenden Sie die Shell des Benutzers **dbuser1**.

```
[dbuser1@serverb ~]$ exit  
logout  
[root@serverb ~]#
```

6. Erstellen Sie ein neues Verzeichnis namens **/home/student/grading/review2** mit **student** und **database** als Besitzer bzw. Besitzergruppe. Konfigurieren Sie die Berechtigungen für dieses Verzeichnis so, dass alle neuen Dateien in diesem Verzeichnis **database** als Besitzergruppe erben, unabhängig vom erstellenden Benutzer. Die Berechtigungen für **/home/student/grading/review2** sollten den Gruppenmitgliedern von **database** und dem Benutzer **student** erlauben, auf das Verzeichnis zuzugreifen und Inhalte darin zu erstellen. Alle anderen Benutzer sollten über Lese- und Ausführungsberechtigungen für das Verzeichnis verfügen. Stellen Sie außerdem sicher, dass Benutzer nur ihre eigenen Dateien aus **/home/student/grading/review2** löschen dürfen, nicht jedoch Dateien im Besitzt anderer Benutzer.

6.1. Erstellen Sie mit dem Befehl **mkdir** das Verzeichnis **/home/student/grading/review2**.

```
[root@serverb ~]# mkdir /home/student/grading/review2
```

6.2. Verwenden Sie in **/home/student/grading/review2** den Befehl **chown**, um **student** und **database** als Besitzer bzw. Besitzergruppe festzulegen.

```
[root@serverb ~]# chown student:database /home/student/grading/review2
```

6.3. Verwenden Sie den Befehl **chmod**, um die spezielle Berechtigung SetGID auf **/home/student/grading/review2** anzuwenden.

```
[root@serverb ~]# chmod g+s /home/student/grading/review2
```

## Kapitel 17 | Ausführliche Wiederholung

- 6.4. Verwenden Sie den Befehl **chmod**, um den Berechtigungsmodus **775** auf **/home/student/grading/review2** anzuwenden.

```
[root@serverb ~]# chmod 775 /home/student/grading/review2
```

- 6.5. Verwenden Sie den Befehl **chmod**, um die spezielle Berechtigung stickybit auf **/home/student/grading/review2** anzuwenden.

```
[root@serverb ~]# chmod o+t /home/student/grading/review2
```

- 6.6. Beenden Sie die Shell des Benutzers **root**.

```
[root@serverb ~]# exit  
logout  
[student@serverb ~]$
```

- 6.7. Melden Sie sich von **serverb** ab.

```
[student@serverb ~]$ exit  
logout  
Connection to serverb closed.  
[student@workstation ~]$
```

## Bewertung

Führen Sie auf **workstation** den Befehl **lab rhcsa-rh124-review2 grade** aus, um den Erfolg dieser Übung zu bestätigen.

```
[student@workstation ~]$ lab rhcsa-rh124-review2 grade
```

## Beenden

Führen Sie **lab rhcsa-rh124-review2 finish** auf **workstation** aus, um die ausführliche Überprüfung abzuschließen. Dieses Skript beendet den Prozess, löscht die Dateien und Verzeichnisse, die beim Start der ausführlichen Überprüfung erstellt wurden, und stellt sicher, dass die Umgebung auf **serverb** sauber ist.

```
[student@workstation ~]$ lab rhcsa-rh124-review2 finish
```

Damit ist die ausführliche Überprüfung abgeschlossen.

## ► Praktische Übung

# Konfigurieren und Verwalten eines Servers

In dieser Überprüfung konfigurieren, sichern und verwenden Sie den SSH-Service, um auf einen Remote-Computer zuzugreifen, konfigurieren den Service **rsyslog**, archivieren lokale Dateien, übertragen lokale Dateien auf den Remote-Computer und verwalten Pakete mit **yum**.

### Ergebnisse

Sie sollten über die folgenden Fähigkeiten verfügen:

- Erstellen eines neuen SSH-Schlüsselpaars
- Deaktivieren von SSH-Anmeldungen als Benutzer **root**
- Deaktivieren von SSH-Anmeldungen mit Passwort
- Aktualisieren der Zeitzone eines Servers
- Installieren von Pakete und Paketmodulen mit **yum**
- Archivieren lokaler Dateien zu Backup-Zwecken
- Übertragen lokaler Dateien auf einen Remote-Computer

### Bevor Sie Beginnen

Melden Sie sich als Benutzer **student** mit dem Passwort **student** bei **workstation** an.

Führen Sie **lab rhcsa-rh124-review3 start** auf **workstation** aus, um die ausführliche Überprüfung zu beginnen. Dieses Skript erstellt die erforderlichen Dateien, um die Umgebung korrekt einzurichten.

```
[student@workstation ~]$ lab rhcsa-rh124-review3 start
```

### Anweisungen

Führen Sie die folgenden Aufgaben auf **serverb** aus, um die Übung abzuschließen.

- Generieren Sie SSH-Schlüssel für den Benutzer **student** auf **serverb**. Schützen Sie den Private Key nicht mit einer Passphrase. Die Dateien für privaten und öffentlichen Schlüssel sollten **/home/student/.ssh/review3\_key** beziehungsweise **/home/student/.ssh/review3\_key.pub** genannt werden.
- Konfigurieren Sie auf **servera** den Benutzer **student** so, dass Anmeldungen akzeptiert werden, die mit dem für **student** auf **serverb** erstellten SSH-Schlüsselpaar authentifiziert wurden. Der Benutzer **student** auf **serverb** sollte in der Lage sein, sich mittels SSH ohne Eingabe eines Passworts bei **servera** anzumelden.
- Konfigurieren Sie den Service **sshd** auf **serverb**, um zu verhindern, dass sich Benutzer über SSH als **root** anmelden.

## Kapitel 17 | Ausführliche Wiederholung

- Konfigurieren Sie den Service **sshd** auf **serverb**, um zu verhindern, dass sich Benutzer mit ihren Passwörtern anmelden. Benutzer sollten weiterhin Anmeldungen mit einem SSH-Schlüsselpaar authentifizieren können.
- Erstellen Sie ein tar-Archiv namens **/tmp/log.tar**, das den Inhalt von **/var/log** auf **serverb** enthält. Übertragen Sie das tar-Archiv remote in das Verzeichnis **/tmp** auf **servera**; authentifizieren Sie sich dabei als **student** und verwenden Sie den privaten Schlüssel des Benutzers **student** aus dem SSH-Schlüsselpaar.
- Konfigurieren Sie den Service **rsyslog** auf **serverb**, um alle empfangenen Meldungen mit der Prioritätsstufe **debug** oder höher in der Datei **/var/log/grading-debug** zu protokollieren. Diese Konfiguration sollte in einer **/etc/rsyslog.d/grading-debug.conf**-Datei festgelegt werden, die Sie erstellen müssen.
- Installieren Sie das im BaseOS-Repository verfügbare zsh-Paket auf **serverb**.
- Aktivieren Sie den Standard-Modul-Stream für das Modul **python36** und installieren Sie alle bereitgestellten Pakete aus diesem Stream auf **serverb**.
- Legen Sie die Zeitzone von **serverb** auf **Asia/Kolkata** fest.

## Bewertung

Führen Sie auf **workstation** den Befehl **lab rhcsa-rh124-review3 grade** aus, um den Erfolg dieser Übung zu bestätigen.

```
[student@workstation ~]$ lab rhcsa-rh124-review3 grade
```

## Beenden

Führen Sie **lab rhcsa-rh124-review3 finish** auf **workstation** aus, um die ausführliche Überprüfung abzuschließen. Dieses Skript löscht die Dateien und Verzeichnisse, die beim Start der ausführlichen Überprüfung erstellt wurden, und stellt sicher, dass die Umgebung auf **serverb** sauber ist.

```
[student@workstation ~]$ lab rhcsa-rh124-review3 finish
```

Damit ist die ausführliche Überprüfung abgeschlossen.

## ► Lösung

# Konfigurieren und Verwalten eines Servers

In dieser Überprüfung konfigurieren, sichern und verwenden Sie den SSH-Service, um auf einen Remote-Computer zuzugreifen, konfigurieren den Service **rsyslog**, archivieren lokale Dateien, übertragen lokale Dateien auf den Remote-Computer und verwalten Pakete mit **yum**.

## Ergebnisse

Sie sollten über die folgenden Fähigkeiten verfügen:

- Erstellen eines neuen SSH-Schlüsselpaars
- Deaktivieren von SSH-Anmeldungen als Benutzer **root**
- Deaktivieren von SSH-Anmeldungen mit Passwort
- Aktualisieren der Zeitzone eines Servers
- Installieren von Pakete und Paketmodulen mit **yum**
- Archivieren lokaler Dateien zu Backup-Zwecken
- Übertragen lokaler Dateien auf einen Remote-Computer

## Bevor Sie Beginnen

Melden Sie sich als Benutzer **student** mit dem Passwort **student** bei **workstation** an.

Führen Sie **lab rhcsa-rh124-review3 start** auf **workstation** aus, um die ausführliche Überprüfung zu beginnen. Dieses Skript erstellt die erforderlichen Dateien, um die Umgebung korrekt einzurichten.

```
[student@workstation ~]$ lab rhcsa-rh124-review3 start
```

## Anweisungen

Führen Sie die folgenden Aufgaben auf **serverb** aus, um die Übung abzuschließen.

- Generieren Sie SSH-Schlüssel für den Benutzer **student** auf **serverb**. Schützen Sie den Private Key nicht mit einer Passphrase. Die Dateien für privaten und öffentlichen Schlüssel sollten **/home/student/.ssh/review3\_key** beziehungsweise **/home/student/.ssh/review3\_key.pub** genannt werden.
- Konfigurieren Sie auf **servera** den Benutzer **student** so, dass Anmeldungen akzeptiert werden, die mit dem für **student** auf **serverb** erstellten SSH-Schlüsselpaar authentifiziert wurden. Der Benutzer **student** auf **serverb** sollte in der Lage sein, sich mittels SSH ohne Eingabe eines Passworts bei **servera** anzumelden.
- Konfigurieren Sie den Service **sshd** auf **serverb**, um zu verhindern, dass sich Benutzer über SSH als **root** anmelden.

**Kapitel 17 |** Ausführliche Wiederholung

- Konfigurieren Sie den Service **sshd** auf **serverb**, um zu verhindern, dass sich Benutzer mit ihren Passwörtern anmelden. Benutzer sollten weiterhin Anmeldungen mit einem SSH-Schlüsselpaar authentifizieren können.
- Erstellen Sie ein tar-Archiv namens **/tmp/log.tar**, das den Inhalt von **/var/log** auf **serverb** enthält. Übertragen Sie das tar-Archiv remote in das Verzeichnis **/tmp** auf **servera**; authentifizieren Sie sich dabei als **student** und verwenden Sie den privaten Schlüssel des Benutzers **student** aus dem SSH-Schlüsselpaar.
- Konfigurieren Sie den Service **rsyslog** auf **serverb**, um alle empfangenen Meldungen mit der Prioritätsstufe **debug** oder höher in der Datei **/var/log/grading-debug** zu protokollieren. Diese Konfiguration sollte in einer **/etc/rsyslog.d/grading-debug.conf**-Datei festgelegt werden, die Sie erstellen müssen.
- Installieren Sie das im BaseOS-Repository verfügbare zsh-Paket auf **serverb**.
- Aktivieren Sie den Standard-Modul-Stream für das Modul **python36** und installieren Sie alle bereitgestellten Pakete aus diesem Stream auf **serverb**.
- Legen Sie die Zeitzone von **serverb** auf **Asia/Kolkata** fest.

1. Generieren Sie SSH-Schlüssel für den Benutzer **student** auf **serverb**. Schützen Sie den Private Key nicht mit einer Passphrase. Die Dateien für privaten und öffentlichen Schlüssel sollten **/home/student/.ssh/review3\_key** beziehungsweise **/home/student/.ssh/review3\_key.pub** genannt werden.

- 1.1. Öffnen Sie auf **workstation** als **student** eine SSH-Sitzung zu **serverb**.

```
[student@workstation ~]$ ssh student@serverb
...output omitted...
[student@serverb ~]$
```

- 1.2. Generieren Sie mit dem Befehl **ssh-keygen** die SSH-Schlüssel für den Benutzer **student**.

```
[student@serverb ~]$ ssh-keygen
Generating public/private rsa key pair.
Enter file in which to save the key (/home/student/.ssh/id_rsa): /home/
student/.ssh/review3_key
Enter passphrase (empty for no passphrase): Enter
Enter same passphrase again: Enter
Your identification has been saved in /home/student/.ssh/review3_key.
Your public key has been saved in /home/student/.ssh/review3_key.pub.
The key fingerprint is:
SHA256:Uqefehw+vRfm94fQZDz/6IfNYSLK/OpiQ4n6lrKIbY student@serverb.lab.example.com
The key's randomart image is:
+---[RSA 2048]---+
|           |
|           . |
|       . . . |
|       . o . = |
|       . S . * ..|
```

```
|       . . .B +..|  
|.o . o . =o+ 0.o |  
|+= . + ..X o * .o|  
| Eoo .o.+.+o=.+=|  
+---[SHA256]-----+
```

2. Konfigurieren Sie auf **servera** den Benutzer **student** so, dass Anmeldungen akzeptiert werden, die mit dem für **student** auf **serverb** erstellten SSH-Schlüsselpaar authentifiziert wurden. Der Benutzer **student** auf **serverb** sollte in der Lage sein, sich mittels SSH ohne Eingabe eines Passworts bei **servera** anzumelden.
- 2.1. Verwenden Sie den Befehl **ssh-copy-id**, um den öffentlichen Schlüssel **/home/student/.ssh/review3\_key.pub** von **servera** nach **serverb** zu exportieren.

```
[student@servera ~]$ ssh-copy-id -i .ssh/review3_key.pub student@serverb  
/usr/bin/ssh-copy-id: INFO: Source of key(s) to be installed: ".ssh/review3.pub"  
/usr/bin/ssh-copy-id: INFO: attempting to log in with the new key(s), to filter  
out any that are already installed  
/usr/bin/ssh-copy-id: INFO: 1 key(s) remain to be installed -- if you are prompted  
now it is to install the new keys  
student@servera's password: student  
  
Number of key(s) added: 1  
  
Now try logging into the machine, with: "ssh 'student@servera'"  
and check to make sure that only the key(s) you wanted were added.
```

- 2.2. Verwenden Sie den Befehl **ssh**, um zu bestätigen, dass Sie sich von **serverb** als **student** mit dem privaten SSH-Schlüssel **/home/student/.ssh/review3\_key** bei **servera** anmelden können, ohne nach dem Passwort gefragt zu werden.

```
[student@serverb ~]$ ssh -i .ssh/review3_key student@servera  
...output omitted...  
[student@servera ~]$
```

- 2.3. Melden Sie sich von **servera** ab.

```
[student@servera ~]$ exit  
logout  
Connection to servera closed.  
[student@serverb ~]$
```

3. Konfigurieren Sie den Service **sshd** auf **serverb**, um zu verhindern, dass sich Benutzer mit SSH als **root** anmelden.
- 3.1. Legen Sie den Parameter **PermitRootLogin** in **/etc/ssh/sshd\_config** auf **no** fest. Sie können den Befehl **sudo vim /etc/ssh/sshd\_config** verwenden, um die Konfigurationsdatei zu bearbeiten.
- 3.2. Laden Sie den Service **sshd** neu.

```
[student@serverb ~]$ sudo systemctl reload sshd.service
```

4. Konfigurieren Sie den Service **sshd** auf **serverb**, um zu verhindern, dass sich Benutzer mit ihren Passwörtern anmelden. Benutzer sollten weiterhin Anmeldungen mit ihrem privaten Schlüssel des SSH-Schlüsselpaares authentifizieren können.
- 4.1. Legen Sie den Parameter **PasswordAuthentication** in **/etc/ssh/sshd\_config** auf **no** fest. Sie können den Befehl **sudo vim /etc/ssh/sshd\_config** verwenden, um die Konfigurationsdatei zu bearbeiten.
  - 4.2. Verwenden Sie den Befehl **sudo systemctl**, um den Service **sshd** neu zu laden.

```
[student@serverb ~]$ sudo systemctl reload sshd.service
```

5. Erstellen Sie ein tar-Archiv namens **/tmp/log.tar**, das den Inhalt von **/var/log** auf **serverb** enthält. Übertragen Sie das tar-Archiv remote in das Verzeichnis **/tmp** auf **servera**; authentifizieren Sie sich dabei als **student** und verwenden Sie **/home/student/.ssh/review3\_key** als privaten Schlüssel des Benutzers **student** aus dem SSH-Schlüsselpaar.
- 5.1. Verwenden Sie den Befehl **sudo tar**, um als Superuser ein Archiv namens **/tmp/log.tar** mit dem Inhalt von **/var/log** zu erstellen.

```
[student@serverb ~]$ sudo tar -cvf /tmp/log.tar /var/log  
[sudo] password for student: student  
...output omitted...
```

- 5.2. Verwenden Sie den Befehl **scp**, um die Archivdatei **/tmp/log.tar** remote in das Verzeichnis **/tmp** auf **servera** zu übertragen. Geben Sie **/home/student/.ssh/review3\_key** als privaten Schlüssel des SSH-Schlüsselpaares an.
- ```
[student@serverb ~]$ scp -i .ssh/review3_key /tmp/log.tar student@servera:/tmp  
log.tar          100%   14MB  57.4MB/s  00:00
```
6. Konfigurieren Sie den Service **rsyslog** auf **serverb**, um alle empfangenen Meldungen mit der Prioritätsstufe **debug** oder höher in der Datei **/var/log/grading-debug** zu protokollieren. Diese Konfiguration sollte in einer **/etc/rsyslog.d/grading-debug.conf**-Datei festgelegt werden, die Sie erstellen müssen.

- 6.1. Erstellen Sie die Datei **/etc/rsyslog.d/grading-debug.conf** mit dem folgenden Inhalt. Sie können den Befehl **sudo vim /etc/rsyslog.d/grading-debug.conf** verwenden, um die Datei zu erstellen.

```
* .debug /var/log/grading-debug
```

- 6.2. Führen Sie den Befehl **sudo systemctl** aus, um den Service **rsyslog** neu zu starten.

```
[student@serverb ~]$ sudo systemctl restart rsyslog.service
```

- 6.3. Verwenden Sie den Befehl **logger**, um die Protokollmeldung **Debug Testing** mit der Priorität **debug** zu generieren.

```
[student@serverb ~]$ logger -p debug Debug Testing
```

- 6.4. Vergewissern Sie sich, dass die Protokollmeldung **Debug Testing** in der Datei /var/log/grading-debug gespeichert wird.

```
[student@serverb ~]$ sudo tail /var/log/grading-debug
...output omitted...
Mar 12 09:55:23 serverb student[32383]: Debug Testing
```

7. Installieren Sie das im BaseOS-Repository verfügbare zsh-Paket mithilfe des Befehls **sudo yum** auf **serverb**.

7.1.

```
[student@serverb ~]$ sudo yum install zsh
...output omitted...
Is this ok [y/N]: y
...output omitted...
Installed:
  zsh-5.5.1-6.el8.x86_64
Complete!
```

8. Verwenden Sie den Befehl **yum**, um den Standard-Modul-Stream für das Modul **python36** zu aktivieren, und installieren Sie alle bereitgestellten Pakete aus diesem Stream auf **serverb**.

8.1.

```
[student@serverb ~]$ sudo yum module install python36
...output omitted...
Is this ok [y/N]: y
...output omitted...
Installed:
  python36-3.6.6-18.module+el8+2339+1a6691f8.x86_64           python3-
  pip-9.0.3-13.el8.noarch

Complete!
```

9. Legen Sie die Zeitzone von **serverb** auf **Asia/Kolkata** fest.

- 9.1. Verwenden Sie den Befehl **sudo timedatectl**, um die Zeitzone von **serverb** auf **Asia/Kolkata** festzulegen.

```
[student@serverb ~]$ sudo timedatectl set-timezone Asia/Kolkata
```

- 9.2. Melden Sie sich von **serverb** ab.

```
[student@serverb ~]$ exit
logout
Connection to serverb closed.
[student@workstation ~]$
```

## Bewertung

Führen Sie auf **workstation** den Befehl **lab rhcsa-rh124-review3 grade** aus, um den Erfolg dieser Übung zu bestätigen.

```
[student@workstation ~]$ lab rhcsa-rh124-review3 grade
```

## Beenden

Führen Sie **lab rhcsa-rh124-review3 finish** auf **workstation** aus, um die ausführliche Überprüfung abzuschließen. Dieses Skript löscht die Dateien und Verzeichnisse, die beim Start der ausführlichen Überprüfung erstellt wurden, und stellt sicher, dass die Umgebung auf **serverb** sauber ist.

```
[student@workstation ~]$ lab rhcsa-rh124-review3 finish
```

Damit ist die ausführliche Überprüfung abgeschlossen.

## ► Praktische Übung

# Verwalten von Netzwerken

In dieser Überprüfung konfigurieren und testen Sie die Netzwerkverbindung.

### Ergebnisse

Sie sollten über die folgenden Fähigkeiten verfügen:

- Konfigurieren der Netzwerkeinstellungen
- Testen der Netzwerkverbindung
- Festlegen eines statischen Hostnamens für das System
- Verwenden von lokal auflösbaren kanonische Hostnamen zum Herstellen einer Verbindung zu Systemen

### Bevor Sie Beginnen

Melden Sie sich als Benutzer **student** mit dem Passwort **student** bei **workstation** an.

Führen Sie **lab rhcsa-rh124-review4 start** auf **workstation** aus, um die ausführliche Überprüfung zu beginnen. Dieses Skript erstellt die erforderlichen Dateien, um die Umgebung korrekt einzurichten.

```
[student@workstation ~]$ lab rhcsa-rh124-review4 start
```

### Anweisungen

Führen Sie die folgenden Aufgaben auf **serverb** aus, um die Übung abzuschließen.



#### Warnung

Es gilt als nützliche Praxis, Netzwerkänderungen von der Serverkonsole aus vorzunehmen, entweder lokal oder über Hardware für den Remote-Konsolenzugriff. Wenn Sie mit **ssh** Netzwerkeinstellungen anpassen, kann ein fehlerhafter Befehl dazu führen, dass sich Ihre Sitzung aufhängt oder gesperrt wird. Korrekturen an der Netzwerkkonfiguration müssen dann über die Konsole vorgenommen werden.

Klicken Sie auf der Webseite, die Ihre Laborumgebung steuert, auf die Schaltfläche **OPEN CONSOLE** für **serverb**. In Ihrem Browser wird eine Registerkarte mit der Konsolensitzung für **serverb** geöffnet. Melden Sie sich an der Eingabeaufforderung als Benutzer **student** an.

- Ermitteln Sie den Namen der Ethernet-Schnittstelle und deren aktives Verbindungsprofil auf **serverb**.
- Erstellen Sie auf **serverb** ein neues Verbindungsprofil namens **static** für die verfügbare Ethernet-Schnittstelle, das Netzwerkeinstellungen statisch festlegt und kein DHCP verwendet. Verwenden Sie die Einstellungen in der folgenden Tabelle.

IPv4-Adresse	172.25.250.111
Netzmaske	255.255.255.0
Gateway	172.25.250.254
DNS-Server	172.25.250.254

Richten Sie die Ethernet-Schnittstelle des Servers so ein, dass die aktualisierten Netzwerkeinstellungen aus der obigen Tabelle verwendet werden.

- Stellen Sie sicher, dass der Hostname von **serverb** statisch auf **server-review4.lab4.example.com** festgelegt ist.
- Legen Sie auf **serverb** den kanonischen Hostnamen **client-review4** für die IPv4-Adresse **172. 25. 250. 10** des Hosts **servera.lab.example.com** fest.
- Konfigurieren Sie die zusätzliche IPv4-Adresse **172. 25. 250. 211** mit der Netzmaske **255.255.255.0** auf derselben Schnittstelle von **serverb**, die die vorhandenen statischen Netzwerkeinstellungen aufweist. Entfernen Sie die vorhandene IPv4-Adresse nicht. Stellen Sie sicher, dass **serverb** auf alle Adressen antwortet, wenn die statisch auf seiner Schnittstelle konfigurierte Verbindung aktiv ist.
- Stellen Sie auf **serverb** die ursprünglichen Netzwerkeinstellungen wieder her, indem Sie die ursprüngliche Netzwerkverbindung aktivieren und die manuell erstellte Netzwerkverbindung **static** deaktivieren.

## Bewertung

Führen Sie auf **workstation** den Befehl **lab rhcsa-rh124-review4 grade** aus, um den Erfolg dieser Übung zu bestätigen.

```
[student@workstation ~]$ lab rhcsa-rh124-review4 grade
```

## Beenden

Führen Sie **lab rhcsa-rh124-review4 finish** auf **workstation** aus, um die ausführliche Überprüfung abzuschließen. Dieses Skript löscht die Dateien und Verzeichnisse, die beim Start der ausführlichen Überprüfung erstellt wurden, und stellt sicher, dass die Umgebung auf **serverb** sauber ist.

```
[student@workstation ~]$ lab rhcsa-rh124-review4 finish
```

Damit ist die ausführliche Überprüfung abgeschlossen.

## ► Lösung

# Verwalten von Netzwerken

In dieser Überprüfung konfigurieren und testen Sie die Netzwerkverbindung.

## Ergebnisse

Sie sollten über die folgenden Fähigkeiten verfügen:

- Konfigurieren der Netzwerkeinstellungen
- Testen der Netzwerkverbindung
- Festlegen eines statischen Hostnamens für das System
- Verwenden von lokal auflösbaren kanonische Hostnamen zum Herstellen einer Verbindung zu Systemen

## Bevor Sie Beginnen

Melden Sie sich als Benutzer **student** mit dem Passwort **student** bei **workstation** an.

Führen Sie **lab rhcsa-rh124-review4 start** auf **workstation** aus, um die ausführliche Überprüfung zu beginnen. Dieses Skript erstellt die erforderlichen Dateien, um die Umgebung korrekt einzurichten.

```
[student@workstation ~]$ lab rhcsa-rh124-review4 start
```

## Anweisungen

Führen Sie die folgenden Aufgaben auf **serverb** aus, um die Übung abzuschließen.



### Warnung

Es gilt als nützliche Praxis, Netzwerkänderungen von der Serverkonsole aus vorzunehmen, entweder lokal oder über Hardware für den Remote-Konsolenzugriff. Wenn Sie mit **ssh** Netzwerkeinstellungen anpassen, kann ein fehlerhafter Befehl dazu führen, dass sich Ihre Sitzung aufhängt oder gesperrt wird. Korrekturen an der Netzwerkkonfiguration müssen dann über die Konsole vorgenommen werden.

Klicken Sie auf der Webseite, die Ihre Laborumgebung steuert, auf die Schaltfläche **OPEN CONSOLE** für **serverb**. In Ihrem Browser wird eine Registerkarte mit der Konsolensitzung für **serverb** geöffnet. Melden Sie sich an der Eingabeaufforderung als Benutzer **student** an.

- Ermitteln Sie den Namen der Ethernet-Schnittstelle und deren aktives Verbindungsprofil auf **serverb**.
- Erstellen Sie auf **serverb** ein neues Verbindungsprofil namens **static** für die verfügbare Ethernet-Schnittstelle, das Netzwerkeinstellungen statisch festlegt und kein DHCP verwendet. Verwenden Sie die Einstellungen in der folgenden Tabelle.

IPv4-Adresse	172.25.250.111
Netzmaske	255.255.255.0
Gateway	172.25.250.254
DNS-Server	172.25.250.254

Richten Sie die Ethernet-Schnittstelle des Servers so ein, dass die aktualisierten Netzwerkeinstellungen aus der obigen Tabelle verwendet werden.

- Stellen Sie sicher, dass der Hostname von **serverb** statisch auf **server-review4.lab4.example.com** festgelegt ist.
- Legen Sie auf **serverb** den kanonischen Hostnamen **client-review4** für die IPv4-Adresse **172.25.250.10** des Hosts **servera.lab.example.com** fest.
- Konfigurieren Sie die zusätzliche IPv4-Adresse **172.25.250.211** mit der Netzmaske **255.255.255.0** auf derselben Schnittstelle von **serverb**, die die vorhandenen statischen Netzwerkeinstellungen aufweist. Entfernen Sie die vorhandene IPv4-Adresse nicht. Stellen Sie sicher, dass **serverb** auf alle Adressen antwortet, wenn die statisch auf seiner Schnittstelle konfigurierte Verbindung aktiv ist.
- Stellen Sie auf **serverb** die ursprünglichen Netzwerkeinstellungen wieder her, indem Sie die ursprüngliche Netzwerkverbindung aktivieren und die manuell erstellte Netzwerkverbindung **static** deaktivieren.

1. Melden Sie sich über die Konsole als **student** bei **serverb** lokal an.
  - 1.1. Klicken Sie auf der Webseite, die Ihre Laborumgebung steuert, auf die Schaltfläche **OPEN CONSOLE** für **serverb**. In Ihrem Browser wird eine Registerkarte mit der Konsolensitzung für **serverb** geöffnet. Melden Sie sich an der Eingabeaufforderung als Benutzer **student** an.  
Es gilt als nützliche Praxis, Netzwerkänderungen von der Serverkonsole aus vorzunehmen, entweder lokal oder über Hardware für den Remote-Konsolenzugriff. Wenn Sie mit **ssh** Netzwerkeinstellungen anpassen, kann ein fehlerhafter Befehl dazu führen, dass sich Ihre Sitzung aufhängt oder gesperrt wird. Korrekturen an der Netzwerkkonfiguration müssen dann über die Konsole vorgenommen werden.
2. Ermitteln Sie den Namen der Ethernet-Schnittstelle auf **serverb** und den Namen des verwendeten Verbindungsprofils.
  - 2.1. In diesem Beispiel ist **enX** der Name der Ethernet-Schnittstelle. Der Name des Verbindungsprofils lautet **Wired connection 1**. Erstellen Sie das Verbindungsprofil **static** für diese Schnittstelle.
  - 2.2. Der Name der Netzwerkschnittstelle und des ursprünglichen Verbindungsprofils können auf Ihrem **serverb** abweichen. Verwenden Sie den von Ihrem System angezeigten Namen, um den Platzhalternamen **enX** in den Schritten dieser Lösung zu ersetzen.
3. Erstellen Sie auf **serverb** ein neues Verbindungsprofil namens **static** für die verfügbare Ethernet-Schnittstelle. Legen Sie die Netzwerkeinstellungen statisch fest, sodass DHCP nicht verwendet wird. Die Einstellungen sollten auf der folgenden Tabelle basieren:

IPv4-Adresse	172.25.250.111
Netzmaske	255.255.255.0
Gateway	172.25.250.254
DNS-Server	172.25.250.254

Die Ethernet-Schnittstelle des Servers **serverb** sollte die aktualisierten Netzwerkeinstellungen aus der vorherigen Tabelle verwenden.

- 3.1. Erstellen Sie mit **nmcli** die Verbindung **static** mit den angegebenen Netzwerkeinstellungen.

```
[student@serverb ~]$ sudo nmcli connection add con-name static type ethernet \
  iface enX ipv4.addresses '172.25.250.111/24' ipv4.gateway '172.25.250.254' \
  ipv4.dns '172.25.250.254' ipv4.method manual
[sudo] password for student: student
Connection 'static' (ac8620e6-b77e-499f-9931-118b8b015807) successfully added.
```

- 3.2. Verwenden Sie den Befehl **nmcli**, um die neuen Verbindungseinstellungen zu aktivieren.

```
[student@serverb ~]$ sudo nmcli connection up static
```

4. Legen Sie mit dem Befehl **hostnamectl** den **serverb**-Hostnamen auf **server-review4.lab4.example.com** fest. Überprüfen Sie den neuen Hostnamen.

- 4.1.

```
[student@serverb ~]$ sudo hostnamectl set-hostname server-review4.lab4.example.com
[sudo] password for student: student
[student@serverb ~]$ hostname
server-review4.lab4.example.com
```

5. Bearbeiten Sie auf **serverb** die Datei **/etc/hosts**, um den kanonischen Hostnamen **client-review4** für die IPv4-Adresse **172.25.250.10** des Hosts **servera.lab.example.com** festzulegen.

- 5.1. Bearbeiten Sie die Datei **/etc/hosts**, um **client-review4** als Namen für die IPv4-Adresse **172.25.250.10** hinzuzufügen.

```
172.25.250.10 servera.lab.example.com servera client-review4
```

- 5.2. Verifizieren Sie mithilfe des Befehls **ping**, ob Sie **172.25.250.10** unter Verwendung des kanonischen Hostnamens **client-review4** erreichen können.

```
[student@serverb ~]$ ping -c2 client-review4
PING servera.lab.example.com (172.25.250.10) 56(84) bytes of data.
64 bytes from servera.lab.example.com (172.25.250.10): icmp_seq=1 ttl=64
time=0.259 ms
64 bytes from servera.lab.example.com (172.25.250.10): icmp_seq=2 ttl=64
time=0.391 ms

--- servera.lab.example.com ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 33ms
rtt min/avg/max/mdev = 0.259/0.325/0.391/0.066 ms
```

6. Ändern Sie das Verbindungsprofil **static** so, dass die zusätzliche IPv4-Adresse **172.25.250.211** mit der Netzmaske **255.255.255.0** auf derselben **serverb**-Schnittstelle mit den vorhandenen statischen Einstellungen konfiguriert wird. Entfernen Sie die vorhandene IPv4-Adresse nicht. Überprüfen Sie, ob **serverb** auf alle Adressen reagiert, wenn das geänderte Verbindungsprofil aktiv ist.

- 6.1. Verwenden Sie den Befehl **nmcli**, um die neue IP-Adresse hinzuzufügen.

```
[student@serverb ~]$ sudo nmcli connection modify static \
+ipv4.addresses '172.25.250.211/24'
```

- 6.2. Verwenden Sie den Befehl **nmcli**, um die neue IP-Adresse zu aktivieren.

```
[student@serverb ~]$ sudo nmcli connection up static
...output omitted...
```

- 6.3. Führen Sie auf **workstation** den Befehl **ping** aus, um zu überprüfen, ob die IPv4-Adresse **172.25.250.211** erreicht werden kann.

```
[student@workstation ~]$ ping -c2 172.25.250.211
PING 172.25.250.211 (172.25.250.211) 56(84) bytes of data.
64 bytes from 172.25.250.211: icmp_seq=1 ttl=64 time=0.246 ms
64 bytes from 172.25.250.211: icmp_seq=2 ttl=64 time=0.296 ms

--- 172.25.250.211 ping statistics ---
2 packets transmitted, 2 received, 0% packet loss, time 50ms
rtt min/avg/max/mdev = 0.246/0.271/0.296/0.025 ms
```

7. Stellen Sie auf **serverb** die ursprünglichen Einstellungen wieder her, indem Sie die ursprüngliche Netzwerkverbindung aktivieren.

- 7.1. Kehren Sie zur Konsole zurück und aktivieren Sie mit dem Befehl **nmcli** das ursprüngliche Netzwerkprofil.

```
[student@serverb ~]$ sudo nmcli connection up "Wired connection 1"
...output omitted...
```

Der Name des ursprünglichen Verbindungsprofils kann auf Ihrem **serverb** abweichen. Ersetzen Sie den in dieser Lösung gezeigten Namen durch den Namen auf Ihrem System. Suchen Sie den Namen mit **nmcli connection show**.

- 7.2. Öffnen Sie von **workstation** aus eine SSH-Sitzung zu **serverb** als **student**, um zu überprüfen, ob die ursprünglichen Netzwerkeinstellungen erfolgreich aktiviert wurden.

```
[student@workstation ~]$ ssh student@serverb  
...output omitted...  
[student@server-review4 ~]$
```

- 7.3. Melden Sie sich von **serverb** ab und beenden Sie alle bis auf ein Terminal auf **workstation**.

```
[student@server-review4 ~]$ exit  
logout  
Connection to serverb closed.  
[student@workstation ~]$
```

## Bewertung

Führen Sie auf **workstation** den Befehl **lab rhcsa-rh124-review4 grade** aus, um den Erfolg dieser Übung zu bestätigen.

```
[student@workstation ~]$ lab rhcsa-rh124-review4 grade
```

## Beenden

Führen Sie **lab rhcsa-rh124-review4 finish** auf **workstation** aus, um die ausführliche Überprüfung abzuschließen. Dieses Skript löscht die Dateien und Verzeichnisse, die beim Start der ausführlichen Überprüfung erstellt wurden, und stellt sicher, dass die Umgebung auf **serverb** sauber ist.

```
[student@workstation ~]$ lab rhcsa-rh124-review4 finish
```

Damit ist die ausführliche Überprüfung abgeschlossen.

## ► Praktische Übung

# Mounten von Dateisystemen und Suchen von Dateien

In dieser Überprüfung mounten Sie ein Dateisystem und suchen Dateien anhand verschiedener Kriterien.

### Ergebnisse

Sie sollten über die folgenden Fähigkeiten verfügen:

- Mounten eines vorhandenen Dateisystems
- Suchen nach Dateien anhand von Dateiname, Berechtigungen und Größe

### Bevor Sie Beginnen

Melden Sie sich als Benutzer **student** mit dem Passwort **student** bei **workstation** an.

Führen Sie **lab rhcsa-rh124-review5 start** auf **workstation** aus, um mit der ausführlichen Überprüfung zu beginnen. Dieses Skript erstellt das benötigte Dateisystem sowie die erforderlichen Benutzer- und Gruppenkonten.

```
[student@workstation ~]$ lab rhcsa-rh124-review5 start
```

### Anweisungen

Führen Sie die folgenden Aufgaben auf **serverb** aus, um die Übung abzuschließen.

- Auf **serverb** ist ein Blockgerät mit dem Dateisystem **XFS** vorhanden, das jedoch noch nicht gemountet wurde. Ermitteln Sie das Blockgerät und mounten Sie es im Verzeichnis **/review5-disk**. Erstellen Sie das Verzeichnis **/review5-disk**, falls erforderlich.
- Suchen Sie auf **serverb** die Datei namens **review5-path**. Erstellen Sie eine Datei namens **/review5-disk/review5-path.txt**, die eine einzelne Zeile mit dem absoluten Pfad zur Datei **review5** enthält.
- Suchen Sie auf **serverb** alle Dateien mit **contractor1** und **contractor** als Besitzer bzw. Besitzergruppe. Die Dateien müssen außerdem die oktalen Berechtigungen **640** haben. Speichern Sie die Liste dieser Dateien in **/review5-disk/review5-perms.txt**.
- Suchen Sie auf **serverb** alle Dateien mit einer Größe von 100 Bytes. Speichern Sie die absoluten Pfade dieser Dateien in **/review5-disk/review5-size.txt**.

### Bewertung

Führen Sie auf **workstation** den Befehl **lab rhcsa-rh124-review5 grade** aus, um den Erfolg dieser Übung zu bestätigen.

```
[student@workstation ~]$ lab rhcsa-rh124-review5 grade
```

## Beenden

Führen Sie **lab rhcsa-rh124-review5 finish** auf **workstation** aus, um die ausführliche Überprüfung abzuschließen. Dieses Skript löscht das Dateisystem sowie die Benutzer- und Gruppenkonten, die beim Start der ausführlichen Überprüfung erstellt wurden, und stellt sicher, dass die Umgebung auf **serverb** sauber ist.

```
[student@workstation ~]$ lab rhcsa-rh124-review5 finish
```

Damit ist die ausführliche Überprüfung abgeschlossen.

## ► Lösung

# Mounten von Dateisystemen und Suchen von Dateien

In dieser Überprüfung mounten Sie ein Dateisystem und suchen Dateien anhand verschiedener Kriterien.

## Ergebnisse

Sie sollten über die folgenden Fähigkeiten verfügen:

- Mounten eines vorhandenen Dateisystems
- Suchen nach Dateien anhand von Dateiname, Berechtigungen und Größe

## Bevor Sie Beginnen

Melden Sie sich als Benutzer **student** mit dem Passwort **student** bei **workstation** an.

Führen Sie **lab rhcsa-rh124-review5 start** auf **workstation** aus, um mit der ausführlichen Überprüfung zu beginnen. Dieses Skript erstellt das benötigte Dateisystem sowie die erforderlichen Benutzer- und Gruppenkonten.

```
[student@workstation ~]$ lab rhcsa-rh124-review5 start
```

## Anweisungen

Führen Sie die folgenden Aufgaben auf **serverb** aus, um die Übung abzuschließen.

- Auf **serverb** ist ein Blockgerät mit dem Dateisystem **XFS** vorhanden, das jedoch noch nicht gemountet wurde. Ermitteln Sie das Blockgerät und mounten Sie es im Verzeichnis **/review5-disk**. Erstellen Sie das Verzeichnis **/review5-disk**, falls erforderlich.
- Suchen Sie auf **serverb** die Datei namens **review5-path**. Erstellen Sie eine Datei namens **/review5-disk/review5-path.txt**, die eine einzelne Zeile mit dem absoluten Pfad zur Datei **review5** enthält.
- Suchen Sie auf **serverb** alle Dateien mit **contractor1** und **contractor** als Besitzer bzw. Besitzergruppe. Die Dateien müssen außerdem die oktalen Berechtigungen **640** haben. Speichern Sie die Liste dieser Dateien in **/review5-disk/review5-perms.txt**.
- Suchen Sie auf **serverb** alle Dateien mit einer Größe von 100 Bytes. Speichern Sie die absoluten Pfade dieser Dateien in **/review5-disk/review5-size.txt**.

1. Mounten Sie auf **serverb** das im Leerlauf befindliche Blockgerät mit dem Dateisystem **XFS** im Verzeichnis **/review5-disk**.
  - 1.1. Öffnen Sie auf **workstation** als **student** eine SSH-Sitzung zu **serverb**.

```
[student@workstation ~]$ ssh student@serverb  
...output omitted...  
[student@serverb ~]$
```

- 1.2. Verwenden Sie den Befehl **lsblk -fs**, um das im Leerlauf befindliche Blockgerät mit dem Dateisystem **XFS** zu ermitteln.

```
[student@serverb ~]$ lsblk -fs  
NAME FSTYPE LABEL UUID MOUNTPOINT  
...output omitted...  
vdb1 xfs 3d97c5ef-23e7-4c1c-a9be-d5c475b3d0d5  
└vdb  
...output omitted...
```

Die vorhergehenden Ausgabe zeigt, dass das Blockgerät **vdb1** das Dateisystem **XFS** enthält, das in keinem Verzeichnis gemountet ist.

- 1.3. Erstellen Sie mit dem Befehl **sudo mkdir** das Verzeichnis **/review5-disk** als Superuser. Wenn der Befehl **sudo** Sie zur Eingabe eines Passworts auffordert, geben Sie das Passwort **student** ein.

```
[student@serverb ~]$ sudo mkdir /review5-disk  
[sudo] password for student: student
```

- 1.4. Verwenden Sie den Befehl **sudo mount**, um das Blockgerät **vdb1** als Superuser im Verzeichnis **/review5-disk** zu mounten.

```
[student@serverb ~]$ sudo mount /dev/vdb1 /review5-disk
```

- 1.5. Verifizieren Sie, dass das Blockgerät **vdb1** erfolgreich im Verzeichnis **/review5-disk** gemountet wurde.

```
[student@serverb ~]$ df -Th  
Filesystem Type Size Used Avail Use% Mounted on  
...output omitted...  
/dev/vdb1 xfs 2.0G 47M 2.0G 3% /review5-disk  
...output omitted...
```

2. Suchen Sie auf **serverb** die Datei namens **review5-path**. Halten Sie deren absoluten Pfad in der Textdatei **/review5-disk/review5-path.txt** fest.
- 2.1. Suchen Sie mit dem Befehl **find** die Datei namens **review5-path**. Leiten Sie alle Fehler des Befehls **find** an **/dev/null** um. Diese Umleitung ermöglicht es Ihnen, alle Fehler aus der Ausgabe des Befehls **find** zu verwerfen.

```
[student@serverb ~]$ find / -iname review5-path 2>/dev/null  
/var/tmp/review5-path
```

Notieren Sie sich den absoluten Pfad zur Datei **review5-path** aus der vorhergehenden Ausgabe.

- 2.2. Erstellen Sie die Textdatei **/review5-disk/review5-path.txt**. Halten Sie den absoluten Pfad zur Datei **review5-path**, den Sie im vorhergehenden Schritt ermittelt haben, in der Textdatei **/review5-disk/review5-path.txt** fest. Sie können den Befehl **sudo vim /review5-disk/review5-path.txt** verwenden, um die Textdatei zu erstellen. Geben Sie **:wq!** aus dem Befehlsmodus in **vim** ein, um die Änderungen zu speichern und die Datei zu schließen. Die folgende Ausgabe zeigt den Inhalt der Textdatei **/review5-disk/review5-path.txt**.

```
/var/tmp/review5-path
```

3. Suchen Sie auf **serverb** alle Dateien mit **contractor1** und **contractor** als Besitzer bzw. Besitzergruppe. Die Dateien müssen außerdem die oktalen Berechtigungen **640** haben. Halten Sie die absoluten Pfade zu allen diesen Dateien in der Textdatei **/review5-disk/review5-perms.txt** fest.
- 3.1. Verwenden Sie die Optionen **-user**, **-group** und **-perm** mit dem Befehl **find**, um alle Dateien zu finden, die über den Besitzer **contractor1**, die Besitzergruppe **contractor** und die oktalen Berechtigungen **640** verfügen. Leiten Sie alle Fehler des Befehls **find** an **/dev/null** um.

```
[student@serverb ~]$ find / -user contractor1 \
-group contractor \
-perm 640 2>/dev/null
/usr/share/review5-perms
```

Notieren Sie sich den absoluten Pfad zur Datei **review5-perms** aus der vorhergehenden Ausgabe. Nur die Datei **/usr/share/review5-perms** erfüllt die Kriterien des vorhergehenden **find**-Befehls.

- 3.2. Erstellen Sie die Textdatei **/review5-disk/review5-perms.txt**. Halten Sie den absoluten Pfad zu der einzigen Datei (**review5-perms**), die den Besitzer **contractor1**, die Besitzergruppe **contractor** und die oktalen Berechtigungen **640** aufweist, die im vorhergehenden Schritt ermittelt wurden, in der Textdatei **/review5-disk/review5-perms.txt** fest. Sie können den Befehl **sudo vim /review5-disk/review5-perms.txt** verwenden, um die Textdatei zu erstellen. Geben Sie **:wq!** aus dem Befehlsmodus in **vim** ein, um die Änderungen zu speichern und die Datei zu schließen. Die folgende Ausgabe zeigt den Inhalt der Textdatei **/review5-disk/review5-perms.txt**.

```
/usr/share/review5-perms
```

4. Suchen Sie auf **serverb** alle Dateien mit einer Größe von 100 Bytes. Halten Sie die absoluten Pfade zu allen diesen Dateien in **/review5-disk/review5-size.txt** fest.
- 4.1. Verwenden Sie die Option **-size** mit dem Befehl **find**, um alle Dateien mit einer Größe von 100 Bytes zu finden. Leiten Sie alle Fehler des Befehls **find** an **/dev/null** um.

```
[student@serverb ~]$ find / -size 100c 2>/dev/null
/dev/disk
/run/initramfs
/etc/lvm
/etc/audit
/etc/sos.conf
```

```
/usr/lib/python3.6/site-packages/dnf/conf  
/usr/lib/python3.6/site-packages/ptyprocess  
/usr/share/licenses/ethtool/LICENSE  
/usr/share/doc/libuser  
/usr/share/doc/python3-cryptography/docs/x509  
/usr/share/doc/python3-jinja2/ext  
/usr/share/doc/plymouth/AUTHORS  
/usr/share/vim/vim80/macros/maze/main.aap  
/usr/libexec/plymouth  
/opt/review5-size
```

Die vorhergehende Ausgabe kann auf Ihrem System anders aussehen, abhängig von der Anzahl der Dateien mit einer Größe von 100 Bytes in Ihrem System. Notieren Sie sich die absoluten Pfade zu allen Dateien aus der vorhergehenden Ausgabe.

- 4.2. Erstellen Sie die Textdatei **/review5-disk/review5-size.txt**. Halten Sie die absoluten Pfade zu allen Dateien mit einer Größe von 100 Bytes, die Sie im vorhergehenden Schritt ermittelt haben, in der Textdatei **/review5-disk/review5-size.txt** fest. Sie können den Befehl **sudo vim /review5-disk/review5-size.txt** verwenden, um die Textdatei zu erstellen. Geben Sie **:wq!** aus dem Befehlsmodus in **vim** ein, um die Änderungen zu speichern und die Datei zu schließen. Die Textdatei **/review5-disk/review5-size.txt** sollte neben anderen Pfaden den absoluten Pfad zur Datei **review5-size** enthalten.

```
...output omitted...  
/opt/review5-size  
...output omitted...
```

- 4.3. Melden Sie sich von **serverb** ab.

```
[student@serverb ~]$ exit  
logout  
Connection to serverb closed.  
[student@workstation ~]$
```

## Bewertung

Führen Sie auf **workstation** den Befehl **lab rhcsa-rh124-review5 grade** aus, um den Erfolg dieser Übung zu bestätigen.

```
[student@workstation ~]$ lab rhcsa-rh124-review5 grade
```

## Beenden

Führen Sie **lab rhcsa-rh124-review5 finish** auf **workstation** aus, um die ausführliche Überprüfung abzuschließen. Dieses Skript löscht das Dateisystem sowie die Benutzer- und Gruppenkonten, die beim Start der ausführlichen Überprüfung erstellt wurden, und stellt sicher, dass die Umgebung auf **serverb** sauber ist.

```
[student@workstation ~]$ lab rhcsa-rh124-review5 finish
```

Damit ist die ausführliche Überprüfung abgeschlossen.

