Linux Allgemein

- Arbeitsspeicher im Cache freigeben
- Zugriffsrechte für Dateien und Ordner
- Asciiquarium installieren
- E-Mail Benachrichtigung bei SSH Login
- Standard Boot Partition Ã¤ndern
- Debian Fehlermeldung: dpkg: warning: 'ldconfig' not found in PATH or not executable.
- Mit sendmail Mails verschicken
- Port von sendmail ändern
- Bootloader Hintergrund ändern
- Skript ausführbar machen
- SSH Root Login verbieten
- Belegte Ports in Linux ausgeben
- HTTPS Reverse Proxy mit Nginx konfigurieren
- Apache - Error 404 "Objekt nicht gefunden"
- CPU Informationen und Auslastung im Terminal anzeigen
- Nginx Reverse Proxy - WebSocket verliert die Verbindung
- Apache2 SSL Virtualhost SSL Konfiguration
- Suchen und Ersetzen in Linux

Arbeitsspeicher im Cache freigeben

Einleitung

Wenn du einen Linux Server länger in Betrieb hast, kann es sein, dass dein Monitoring System anschlägt und meldet das kein Arbeitsspeicher mehr frei ist. Der Arbeitsspeicher befindet sich dennoch nur im Cache.

> Durchführung geschieht auf eigene Gefahr!

Erklärung

Wenn ein Linux System arbeitet, schreibt das System häufig verwendete Dateien und Daten in den Cache. So muss nicht immer die Festplatte verwendet werden. Dieses ermöglicht eine höhere Geschwindigkeit des Servers.

Der Cache beeinträchtigt das Linux System aber nicht weiter, wenn das Linux System merkt, dass es mehr Arbeitsspeicher braucht, gibt es automatisch mehr Arbeitsspeicher frei.

Befehl

Mithilfe des folgenden Befehls wird der Cache wieder freigegeben. Der Server nimmt dann nur den Arbeitsspeicher, den er momentan braucht.

```
sync && echo 3 > /proc/sys/vm/drop_caches
```

Falls du sehen möchtest, wie sich die Auslastung des Arbeitsspeichers verÃ¤ndert, kannst du jeweils ein `free` ansetzen. Dieses zeigt an, wie viel Arbeitsspeicher der Server gesamt, benutzt und frei hat. Dieses wird dann wieder in den Hauptspeicher und SWAP aufgegliedert.

```
free && sync && echo 3 > /proc/sys/vm/drop_caches && free
```

Erklärung Befehl

Der `free` Befehl sorgt dafür, eine Rückmeldung über den freien, verwendeten und gesamten Arbeitsspeicher anzuzeigen.

`sync` schreibt die Cache-Dateien auf die Festplatte.

`echo 3 > /proc/sys/vm/drop_caches` schreibt die Zahl 3 in die Datei **drop_caches**, was zur Folge hat, dass der Cache wieder freigegeben wird.

Zugriffsrechte für Dateien und Ordner

Einleitung

In Linux herrscht ein striktes Berechtigungssystem. Wir können für jeden Ordner festlegen, wer welche Datei schreiben, lesen oder ausführen darf. Diese Konfiguration machen wir über das Terminal. Ansonsten können wir die Berechtigungen auch über die GUI setzen.

Zur Verwaltung der Berechtigungen verwenden wir den Befehl **chmod**. Mit diesen können wir Berechtigungen festlegen, verändern oder ganz entfernen.

Syntax von chmod

Wenn wir den Befehl **chmod** verwenden möchten, müssen wir erstmal die Syntax des Befehls nachvollziehen.

```
chmod [optionen] <maske> <datei>
```

Unter **maske** verstehen wir die Berechtigungsmaske. Die Zuteilung kann Symbolisch oder Numerisch umgesetzt werden.

Symbolische Zuteilung

Wenn wir die Berechtigungen über Symbole und Buchstaben ändern möchten, wird die Maske in 3 Teilbereiche aufgeteilt.

- Benutzerkategorie
- Operator
- Rechte

Die Rechte werden von der Benutzerkategorie immer durch einen Operator getrennt. Der Operator gibt an, ob die Rechte jeweils hinzugefügt, entfernt oder gesetzt werden.

Benutzerkategorie	
u	Besitzer
g	Gruppe
o	Andere
a	All (Besitzer, Gruppe, Andere)

Operator	
+	Rechte hinzufügen
-	Rechte entfernen
=	Rechte neu setzen

Rechte	
r	Lesen
w	Schreiben
x	Auführen

Syntax Beispiele:

```
chmod a+rwx datei.txt

chmod +x script.sh

chmod u=rw,g=rw,o=r datei.txt
```

Numerische Zuteilung

Im Gegenteil zu der symbolischen Zuordnung können Berechtigungen auch über die numerische Zuteilung gesetzt werden. Dort wird eine dreistellige Zahl angegeben. Diese teilt mit, welche Berechtigungen, welche Ebene bekommt.

- **1. Zahl** = Besitzer
- **2. Zahl** = Gruppe
- **3. Zahl** = Andere

Die Rechte werden dann mit einer Nummer identifiziert. Die Zahl **751** gibt z.B. die Berechtigung für den Besitzer **Vollzugriff**, für die Gruppe **Lesen, Ausführen** und für andere auf **Nur Ausführen**.

Berechtigungen	
7	Vollzugriff
6	Lesen, Schreiben
5	Lesen, Ausführen
4	Nur Lesen
3	Schreiben, Ausführen
2	Nur Schreiben
1	Nur Ausführen
0	Keine Berechtigungen

Syntaxbeispiele:

```
chmod 777 datei.txt


chmod -R 700 /footer/topbar
```

Optionen

Um den Befehl jetzt noch intelligenter zu gestalten, gibt es die **Optionen**. Mit den Optionen können wir dem Befehl noch etwas mitgeben, wie er sich verhalten soll. Die Option geben wir mit einem **Bindestrich** vorne dran an, und wir schreiben dann die entsprechenden Buchstaben dahinter.

Optionen	
-c	hat die gleiche Funktion wie -**v**, gibt aber nur Rückmeldung wenn etwas geändert wird.
-f	Unterdrückt Fehlermeldungen
-R	Damit werden die Berechtigungen Rekursiv auf Unterverzeichnisse und Unterdateien geschrieben.
-v	Zeigt alles an was der Befehl gerade am System macht.

Asciiquarium installieren

Einleitung

Mit Asciiquarium haben wir unser eigenes Aquarium in unserem Terminal. Dieses Projekt hat keinen sinnhaften Grund, aber ist ganz witzig, es sich mal anzuschauen. Es sollte daher nicht in produktiv System verwendet werden, da dort auf eine minimale Installation gesetzt werden sollte, um wenig bis keine Einfalltore zu haben.

Installation

Zuerst müssen wir uns mit dem Terminal unseres Linux Server / Clients verbinden. Wir melden uns dort als *root* an, damit wir nachher die Befehle ohne **sudo** absetzen können.

Im ersten Abschnitt müssen wir **Term-Animation** installieren. Dazu müssen wir ein paar Befehle absetzen. Zuerst installieren wir das Paket **X**.

```
apt install libcurses-perl -y
```

Dann navigieren wir in das Verzeichnis **/tmp**, laden die benötigten Dateien herunter und entpacken diese. Im Anschluss navigieren wir in das Verzeichnis.

```
cd /tmp
wget http://search.cpan.org/CPAN/authors/id/K/KB/KBAUCOM/Term-Animation-2.6.tar.gz
tar -zxvf Term-Animation-2.6.tar.gz
cd Term-Animation-2.6
```

Als nächsten Schritt führen wir das **Makefile Perl Skript** aus und schließen dann im Abschluss die Installation ab.

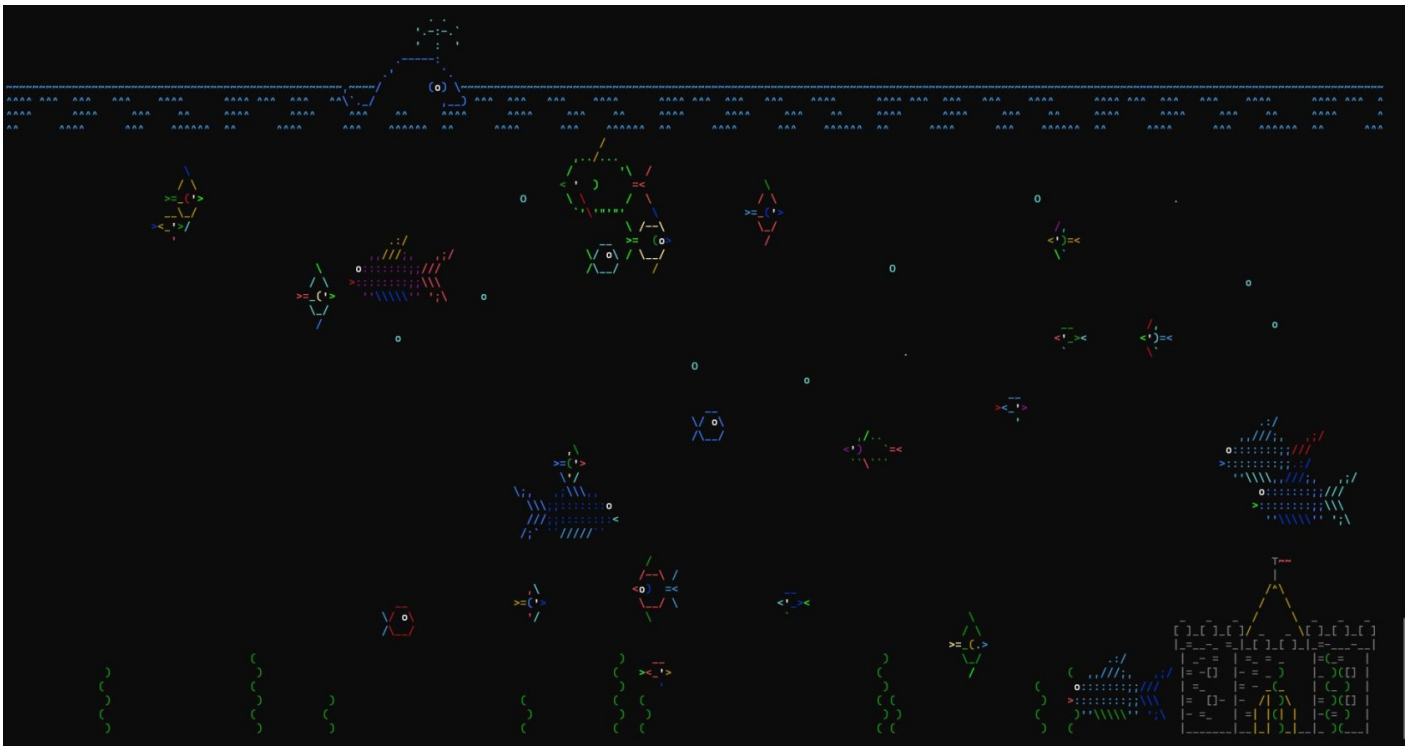
```
perl Makefile.PL && make && make test
make install
```

Jetzt kümmern wir uns darum, dass wir **Asciiquarium** installieren. Dazu wechseln wir wieder in das **/tmp** Verzeichnis und laden das Perl Skript herunter und entpacken dieses wieder, kopieren es in das **/usr/local/bin** Verzeichnis und passen die Berechtigungen an, damit das Skript ausführbar ist.

```
cd /tmp
wget --no-check-certificate http://www.robobunny.com/projects/asciiquarium/asciiquarium.tar.gz
```

```
tar -zxvf asciiquarium.tar.gz
cp asciiquarium /usr/local/bin/
chmod 0755 /usr/local/bin/asciiquarium
```

Wenn wir jetzt unser Aquarium begutachten wollen, starten wir es mit dem Befehl **asciiquarium**.



E-Mail Benachrichtigung bei SSH Login

Einleitung

Wenn wir einen Linux Server absichern wollen, gehört eine stetige Kontrolle aus Sicherheitsgründen auch dazu. Hier konfigurieren wir eine E-Mail Benachrichtigung, die abgesendet wird, sobald sich jemand per SSH auf dem Server einloggt.

Durchführung

Zuerst müssen wir das Paket **s-nail** installieren.

```
sudo apt install s-nail -y
```

Im zweiten Schritt erstellen wir ein Skript, welches, ausgeführt wird, sobald sich jemand auf dem Server einloggt.

```
nano /opt/skripte/ssh-login.sh
```

Dort fügen wir folgenden Inhalt ein, das Skript kann natürlich auch gerne angepasst werden.

```
#!/bin/bash
echo "------------------------------------------------------------------"
echo "Login auf $(hostname) am $(date +%Y-%m-%d) um $(date +%H:%M)"
echo "Benutzer: $USER"
echo "------------------------------------------------------------------"
pinky
```

Jetzt verändern wir die Datei **/etc/profile** und fügen dort den Aufrufer des Skriptes hinzu

```
/opt/skripte/ssh-login.sh | mailx -s "SSH Login auf <server>" <empfÃ¤nger>@<domain>
```

Als Letztes passen wir noch die Berechtigungen des Skriptes an.

```
sudo chmod 755 /opt/skripte/ssh-login.sh
```

Die E-Mails werden jetzt bei einer Anmeldung automatisch an die entsprechende E-Mail-Adresse versendet.

Standard Boot Partition Ã¤ndern

Einleitung

In diesem Beitrag gehe ich drauf ein, wie wir beim GRUB Bootloader die Standard-Partition, die zum Booten verwendet wird, ändern können. Durch die Änderung würde das System, wenn es keine andere Eingabe erhält, automatisch das entsprechende Betriebssystem starten.

Boot Reihenfolge ändern

Backup erstellen

Im ersten Schritt erstellen wir ein Backup unserer derzeitigen GRUB Konfiguration. Dies empfiehlt sich, da Änderungen am Bootloader auch zu Problemen führen kann. Und so können wir jederzeit wieder zurück auf die vorherige Konfiguration.

Um ein Backup zu erstellen, führe den folgenden Befehl aus:

```
sudo cp /etc/default/grub /etc/default/grub.bak
```

Ps: Wir können natürlich auch einen anderen Zielnamen überlegen, falls wir z.B. schon eine Datei mit dem Namen haben, welche nicht überschrieben werden soll.

Backup kontrollieren

Um sicherzustellen, dass der Backup-Vorgang erfolgreich durchgeführt wurde, führen wir den folgenden Befehl aus, um uns den Dateiinhalt der Datei anzeigen zu lassen.

```
cat /etc/default/grub.bak
```

Änderungen speichern

Um einmal die Änderungen zu registrieren, führen wir den folgenden Befehl aus:

```
sudo update-grub
```

Damit wird noch einmal die aktuelle GRUB Konfiguration in den Bootloader geschrieben. Jetzt werden wir aber die Änderungen durchführen und werden dementsprechend den Befehl nachher

nochmal ausführen.

Änderungen vornehmen

Um die Änderungen an der Datei vorzunehmen, müssen wir mit einem Editor unserer Wahl die GRUB Konfigurationsdatei öffnen. Ich verwende dafür den Terminal Editor **nano**.

```
sudo nano /etc/default/grub
```

Hier müssen wir nur die Zahl hinter GRUB_DEFAULT= ändern. Hier geben wir die Position des Boot-Eintrags ein, welchen wir verwenden möchten. Die Aufzählung beginnt bei 0. Sprich: Die erste Position ist 0, die zweite Position ist 1, und immer so weiter.

Sobald wir die Änderungen durchgeführt haben, müssen wir wieder den GRUB Bootloader aktualisieren. Dabei wird dann die aktuelle Konfigurationsdatei eingelesen.

```
sudo update-grub
```

Debian Fehlermeldung: dpkg: warning: 'ldconfig' not found in PATH or not executable.

Einleitung

Letztens bin ich bei der Installation eines Linux Servers auf folgendes Problem gestoßen.
Beim Installieren von Paketen bekam ich die oben angegebenen Fehlermeldung:

> dpkg: warning: 'ldconfig' not found in PATH or not executable

In diesem Beitrag will ich kurz zeigen, wie wir das Problem beseitigen können.

Problem beseitigen

Im ersten Schritt müssen wir uns per **SSH** auf unseren Linux Server schalten, oder anderweitig den **Shell-Zugriff** erreichen. Im Anschluss geben wir die nachstehende Befehle nacheinander ein:

```
nano /root/.bashrc
```

Dort fügen wir in der letzten Zeile der Datei den folgenden Code ein und speichern die Datei:

```
export PATH=/sbin:/bin:/usr/bin:/usr/sbin:/usr/local/sbin:/usr/local/bin
```

Als letzten Schritt geben wir den folgenden Befehl ein:

```
. /root/.bashrc
```

Jetzt sollte es möglich sein, die Pakete wieder zu installieren!

Mit sendmail Mails verschicken

Einleitung

Es ist möglich, mit Linux Mails direkt aus der Kommandozeile zu versenden. Administratoren oder Programmierer schicken sich damit häufig Statusmeldungen oder andere Nachrichten. Auf vielen Webseiten wird häufig dazu die PHP Funktion **sendmail** verwendet. Dazu brauchst du einen E-Mail-Account bei einem Provider deiner Wahl. Wenn der Mail-Server im gleichen Netz wie der Webserver steht, und keine Kommunikation über das Internet erfolgen muss, kann bei richtiger Kommunikation ohne einen E-Mail-Account, eine E-Mail versendet werden.

> sendmail gilt als veraltet und sollte dementsprechend mit Bedacht verwendet werden.

Voraussetzungen

Um jetzt E-Mails mit sendmail über das Internet zu versenden, benötigst du folgende Informationen zu deinem E-Mail-Account:

- SMTP-Adresse
- SMTP-Port
- Login Daten (Benutzername und Kennwort)

E-Mail versenden

Wenn du E-Mails versenden möchtest, kannst du das über unterschiedliche Möglichkeiten machen. Alle gehen direkt von der Kommandozeile aus. Du musst also keine GUI oder sonstige Web Oberfläche öffnen.

Du benötigst, um E-Mails zu versenden das Paket **ssmtp**. Dieses können wir einfach nachinstallieren.

```
sudo apt install ssmtp -y
```

E-Mail nur mit Betreff

```
echo "Subject: Test E-Mail" | sendmail mail@pc-wiki.de
```

E-Mail aus Datei lesen

Zuerst legen wir eine Datei an, in dem sich der E-Mail Inhalt befindet.

```
nano email.txt
```

In dieser Textdatei fügst du den Text ein, den du gerne versenden möchtest. Den Betreff, E-Mail-Adressen und den Nachrichtentext kannst du natürlich gerne ändern. Wichtig ist, dass die Struktur so bestehen bleibt.

```
Cc: mail@phil-un.de
Subject: E-Mail aus Datei
From: server@pc-wiki.de
Content-Type: text/html; charset="utf8"

<html>
<body>
<div style="
background-color:
#abcdef; width: 300px;
height: 300px;
">
</div>
  <h1>Status Meldung</h1>
  <p>Die E-Mails werden erfolgreich vom Server versendet!</p>
</body>
</html>
```

Zum Schluss müssen wir nur noch die E-Mail versenden. Dort kannst du den Empfänger natürlich wieder anpassen.

```
sendmail mail@pc-wiki.de < mail.txt
```

E-Mail über STMP Server versenden

Wenn du E-Mails über den SMTP Server versenden möchtest, um z.B. E-Mails über einen E-Mailserver zu versenden. Musst du die Logindaten in einer Konfigurationsdatei angeben. Öffne zuerst die Konfigurationsdatei

```
sudo nano /etc/ssmtp/ssmtp.conf
```

Dort fügst du folgenden Code ein. Die Logindaten sowie den Mail-Server musst du noch anpassen.

```
UseSTARTTLS=YES
root=server@phil-un.de
mailhub=mail.server.de:587
AuthUser=<server>
AuthPass=<Pa$$w0rd>
```

Und nun versendest du eine E-Mail mit dem folgenden Befehl

```
ssmtp mail@pc-wiki.de < mail.txt
```

> Wenn eine Fehlermeldung erscheint, kannst du diese nutzen um den Fehler zu finden.
> **Beispiel:** `ssmtp: Authorization failed`

Port von sendmail ändern

Einleitung

Manchmal stößt man auf folgendes Problem: Auf einem Server läuft ein Mailserver, z.B. in einem Docker Container und dazu möchte man mit **sendmail** ggf. Mails versenden. Dazu müssen wir den Port von **sendmail** verändern.

Anwendung

Zuerst verbinden wir uns mit unserem Server, damit wir Konsolenzugriff haben. Dort öffnen wir mit einem Editor unserer Wahl die Konfigurationsdatei.

```
sudo nano /etc/mail/sendmail.mc
```

Dort suchen wir nach folgenden Zeilen:

```
dnl DAEMON_OPTIONS(`Family=inet6, Name=MTA-v6, Port=smtp, Addr=::1')dnl
DAEMON_OPTIONS(`Family=inet,  Name=MTA-v4, Port=smtp, Addr=127.0.0.1')dnl
```

Dort ersetzt du **smtp** durch die Port Nummer deiner Wahl.

```
dnl DAEMON_OPTIONS(`Family=inet6, Name=MTA-v6, Port=25000, Addr=::1')dnl
DAEMON_OPTIONS(`Family=inet,  Name=MTA-v4, Port=25000, Addr=127.0.0.1')dnl
```

Als Nächstes lässt du dir die Konfigurationsdatei neu erstellen.

```
sudo m4 /etc/mail/sendmail.mc > /etc/mail/sendmail.cf
```

Und als Letztes startest du sendmail neu, dann sollte dein sendmail jetzt den neuen Port verwenden.

```
sudo systemctl restart sendmail
```

Bootloader Hintergrund ändern

Einleitung

In diesem Beitrag gehe ich kurz drauf ein, wie wir im GRUB Bootloader das Hintergrundbild des Bootloaders ändern können.

Bild ändern

Dazu müssen wir einfach ein Bild, welches wir uns heruntergeladen haben, in ein Verzeichnis kopieren / verschieben. Diesen Pfad müssen wir uns jetzt merken oder aufschreiben.

```
sudo cp index.jpeg /boot/grub/
```

Im Anschluss öffnen wir wieder die GRUB Konfigurationsdatei und fügen folgenden Code mit unserem Pfad hinzu.

```
GRUB_BACKGROUND="/boot/grub/index.jpeg"
```

Am Schluss müssen wir jetzt nur die GRUB Konfiguration wieder aktualisieren.

```
sudo update-grub
```

Skript ausführbar machen

Einleitung

In Linux kannst du grundlegend fremde Skripte nicht starten. Diese musst du dann quasi erst freigeben, dann können die Skripte ausgeführt werden.

Skript freigeben

Wenn du ein Skript ausführbar machen möchtest, wechselst du in das Verzeichnis, in dem sich das Skript befindet und setzt dann den Namen der Datei an das Ende. Dann kannst du das Skript ohne Probleme ausführen.

```
sudo chmod +x ./<name-des-skripts>
```

SSH Root Login verbieten

Einleitung

Damit wir einen sicheren Linux Server haben, können wir den Root Zugang über SSH auf unserem Server deaktivieren. Wenn Server angegriffen werden, wird zuerst immer der **root** Benutzer verwendet, da dieser auf jedem System eingerichtet ist und volle Berechtigungen hat.

Login deaktivieren

Um den Root Login zu deaktivieren, müssen wir die Konfigurationsdatei des SSH Services editieren. Dazu navigieren wir in das Verzeichnis vom SSH Dienst.

```
cd /etc/ssh
```

Dort öffnen wir mit dem Editor unserer Wahl die Konfigurationsdatei.

```
nano sshd_config
```

Dort müssen wir nach **PermitRootLogin** suchen. Wir müssen dort eine Raute vor den Eintrag hinzufügen, damit dieser aus kommentiert ist.

Dann starten wir den SSH Server neu.

```
sudo systemctl restart ssh
```

Jetzt ist kein Login als Root mehr möglich!

> Stelle vorher sicher das ein weiterer Benutzer angelegt ist, mit dem man sich weiterhin am Server anmelden kann.

Belegte Ports in Linux ausgeben

Einleitung

In diesem Beitrag beschreibe ich kurz, wie wir unter Linux uns ausgeben lassen können, welche Software oder Dienst einen bestimmten Port belegt. Dies ist wichtig, wenn, wie z. B.Starten einen neuen Dienst installieren möchten und wir beim Starten die Meldung erhalten, dass der Port belegt ist. Damit können wir dann feststellen, was die Ursache dafür ist.

Ursache herausfinden

Um die Ursache herauszufinden, warum ein Port schon verwendet ist, müssen wir nur das **Terminal / Konsole** öffnen und den folgenden Befehl eingeben. Dabei müssen wir nur `<Port>` durch die **Port Nummer** ersetzen. Wir erhalten dann eine Liste aller Anwendungen, die diesen Port nutzen.

```
ss -tunelp | grep <port>
```

HTTPS Reverse Proxy mit Nginx konfigurieren

Einleitung

Sobald wir **Lokal Webdienste** erstellen, möchten wir diese vielleicht auch über ein **TLS Zertifikat** verschlüsseln. Dazu verwenden wir **selbst signierte Zertifikate** und einen **Nginx Webserver**. Dadurch ist es möglich, dass wir unsere **Web-Dienste** über **HTTPS** erreichbar machen können.

Wie wir **TLS Zertifikate** erstellen können, wird hier genauer erklärt.

Nginx installieren

Im ersten Schritt müssen wir auf unserem **Debian Server** das Paket **Nginx** installieren. Dazu verwenden wir folgenden Befehl:

```
apt update && apt upgrade -y && apt install nginx -y
```

Nginx Konfiguration anpassen

Jetzt erstellen wir die **Nginx Konfigurationsdatei** einmal neu. Im Anschluss öffnen wir die **Konfigurationsdatei** und fügen den nachstehenden Inhalt in die Datei ein.

```
rm /etc/nginx/sites-enabled/default
nano /etc/nginx/sites-enabled/default
```

```
server {
  listen 80;
  server_name host.name;
  return 301 https://die.domain$request_uri;
 }

server {
  listen 443 ssl;
  server_name host.name;
  ssl_certificate  /pfad/zum/zertifikat.csr;
```

```
    ssl_certificate_key  /pfad/zum/zertifikat.key;
    ssl_prefer_server_ciphers on;

    location / {
        proxy_pass http://localhost:<port>;

        proxy_set_header        Host $host;
        proxy_set_header        X-Real-IP $remote_addr;
        proxy_set_header        X-Forwarded-For $proxy_add_x_forwarded_for;
        proxy_set_header        X-Forwarded-Proto $scheme;
    }
}
```

Sobald wir diesen Inhalt in die Datei eingefügt haben, brauchen wir jetzt nur noch einmal den **Nginx Dienst** neu zu starten.

```
systemctl restart nginx
```

Sobald der Dienst neu gestartet ist, überprüfen wir, ob der Dienst ordnungsgemäß gestartet ist. Dies können wir mit dem nachstehenden Befehl überprüfen.

```
systemctl status nginx
```

```
* nginx.service - A high performance web server and a reverse proxy server
     Loaded: loaded (/lib/systemd/system/nginx.service; enabled; vendor preset: enabled)
     Active: active (running) since Thu 2022-10-27 05:47:07 UTC; 46min ago
       Docs: man:nginx(8)
    Process: 10700 ExecStartPre=/usr/sbin/nginx -t -q -g daemon on; master_process on; (code=exited, status=0/SUCCESS)
    Process: 10702 ExecStart=/usr/sbin/nginx -g daemon on; master_process on; (code=exited, status=0/SUCCESS)
   Main PID: 10703 (nginx)
      Tasks: 2 (limit: 19031)
     Memory: 5.7M
        CPU: 244ms
     CGroup: /system.slice/nginx.service
             |-10703 nginx: master process /usr/sbin/nginx -g daemon on; master_process on;
             `-10704 nginx: worker process
```

Apache - Error 404 "Objekt nicht gefunden"

Einleitung

In diesem Beitrag erläutere ich kurz, wie wir das Problem mit dem **Apache Fehler 404 "Objekt nicht gefunden"**, beheben können. Das Problem kann mehrere Ursachen habe, aber diese, die mir aktuell untergekommen sind, zähle ich hier auf.

Fehler beheben

1. Berechtigungen falsch gesetzt

Ein Fehler kann es sein, dass es nicht möglich ist aufgrund fehlender Berechtigungen die Webserver Dateien anzuzeigen. Zu Testzwecken können wir ruhig mal die Berechtigung auf **777** setzen. In Produktivumgebungen sollten die Berechtigungen so gesetzt werden, dass nur die Berechtigungen eingeräumt werden, welche benötigt werden.

Zu den, wie oben angegebenen Testzwecken kann man es mit diesen Berechtigungen mal ausprobieren, ob es dann funktioniert.

```
chmod 777 -R /pfad/zum/webroot
```

Wir können dann auch noch den Besitzer der Dateien / Ordner ändern. Apache verwendet Standard-gemäß den Benutzer *www-data*. Diesem Benutzer werden wir gleich die Besitzerrechte des Ordners verpassen. Dieser kann dann alles mit dem Verzeichnis anstellen.

```
chown -R www-data:www-data /pfad/zum/webroot
```

2. Fehlende .htaccess Datei

Ein kleiner, aber dennoch tragischer Fehler kann eine fehlende .htaccess Datei sein. Manchmal kann es bei einem Kopiervorgang vorkommen, dass wir vergessen, die **.htaccess** Datei mitzukopieren. Falls diese fehlt, kann der Webserver manchmal Webserver Dateien nicht öffnen oder PHP-Skripte ausführen.

Die **.htaccess** Datei enthält Anweisungen für den Apache-Webserver. Ohne diese Datei kann ein Apache Webserver z.B. nicht mit *Redirects* umgehen.

CPU Informationen und Auslastung im Terminal anzeigen

Einleitung

In diesem Beitrag erkläre ich kurz, wie wir in einem Linux Terminal einsehen können, welche CPU in unserem System verbaut ist, und wie die Auslastung der entsprechenden CPU ist.

CPU-Informationen anzeigen

Damit wir die CPU Informationen anzeigt bekommen, müssen wir den folgenden Befehl eingeben.

```
sudo cat /proc/cpuinfo | less
```

Wir erhalten jetzt die CPU Informationen pro Kern. Das heißt, entsprechend wie viele Kerne wir haben, bekommen wir so oft die Ausgabe in das Konsolenfenster ausgegeben. Mit der Taste **Q** schließen wir die Ansicht.

Erweiterte CPU-Informationen anzeigen

Wenn wir erweiterte CPU-Informationen angezeigt bekommen möchten, müssen wir den Befehl `lscpu` verwenden. Dieser gibt in der Konsole eine erweiterte Ansicht über die CPU Informationen dar.

```
sudo lscpu
```

CPU Auslastung anzeigen

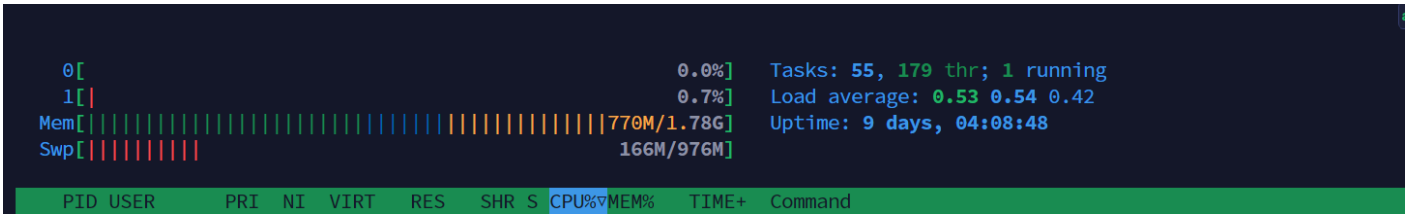
Um die CPU Auslastung unseres Linux Servers uns anzeigen zu lassen, können wir das Tool **htop** verwenden. Dieses kann man schnell über den Paketmanager nachinstallieren.

Installation für **Debian / Ubuntu** sieht wie folgt aus:

```
sudo apt install htop -y
```

Nach der erfolgreichen Installation können wir durch den Befehl `sudo htop` das Tool öffnen. Wir erhalten dann eine detaillierte Übersicht über die verfügbaren Kerne und deren einzelnen Auslastung.

```
sudo htop
```



Nginx Reverse Proxy - WebSocket verliert die Verbindung

Einleitung

Konfigurieren wir unseren **Nginx Webserver** so, dass dieser als **Reverse Proxy** arbeitet, können wir bei einigen **Web-Anwendungen** das Problem haben, dass unser Client die Verbindung zum **WebSocket** verliert.

Konfiguration

Um dieses Problem zu beseitigen, müssen wir nur lediglich unsere **Nginx-Konfiguration** anpassen.

```
    location / {
        proxy_pass http://localhost:8080;

        proxy_http_version      1.1;
        proxy_set_header        Upgrade $http_upgrade;
        proxy_set_header        Connection "upgrade";
        proxy_set_header        Host $host;
        proxy_set_header        X-Real-IP $remote_addr;
        proxy_set_header        X-Forwarded-For $proxy_add_x_forwarded_for;
        proxy_set_header        X-Forwarded-Proto $scheme;
    }
```

Im Anschluss starten wir nur noch den **Nginx Webserver** neu. Danach sollte der **WebSocket** die Verbindung offen halten.

Apache2 SSL Virtualhost SSL Konfiguration

Einleitung

In diesem Beitrag befindet sich eine Version des **Apache2 Virtualhost** für die SSL-Verbindung.

Konfigurationsdatei

```
  <VirtualHost *:80>
    ServerName FQDN
    ServerAdmin webmaster@localhost
    DocumentRoot /var/www/html
    <Directory />
          Options FollowSymLinks
          AllowOverride None
    </Directory>
    ErrorLog  /error.log
    LogLevel warn
    CustomLog  /access.log combined
    ServerSignature On
</VirtualHost>

  <VirtualHost *:443>
    ServerName FQDN
    ServerAdmin webmaster@localhost
    DocumentRoot /var/www/html

    <Directory />
          Options FollowSymLinks
          AllowOverride None
    </Directory>
    ErrorLog  /error.log
    LogLevel warn
    CustomLog  /access.log combined
```

```
    ServerSignature On
        SSLEngine on
        SSLCertificateFile /cert/cert.crt
        SSLCertificateKeyFile /cert/cert.key
</VirtualHost>
```

Suchen und Ersetzen in Linux

Einleitung

In diesem kurzen Artikel geht es kurz darum, wie wir unter Linux **Suchen und Ersetzen** in einer Datei über die *CLI (Commandline Interface)* durchführen können. Dadurch können wir Inhalte in einer Text-Datei durch einen anderen Inhalt überschreiben.

Durchführung

Um den Inhalt zu überschreiben, müssen wir den folgenden Befehl verwenden:

```
sed -i 's/alter Text/neuer Text/g' datei.txt
```

> **Info:** Ersetze `alter Text` durch den Text den du ersetzen möchtest. Ersetze `neuer Text` den Text ein welchen du einfügen möchtest. Im Anschluss passt du `datei.txt` an, auf die Datei die du bearbeiten möchtest.

Sobald wir den Befehl ausgeführt haben, werden die Werte in der Datei überschrieben.