

# Správa služeb

## 1. Instalace a konfigurace webového serveru

### 1.1. Instalace a spuštění webového serveru

1.1.1. Nainstalujte software webového serveru:

```
# yum install httpd mod_ssl
```

1.1.2. Povolte a spusťte službu „httpd“:

```
# systemctl enable --now httpd
Created symlink from /etc/systemd/system/multi-
user.target.wants/httpd.service to /usr/lib/systemd/system/httpd.service.
```

nebo

```
# systemctl enable httpd
# systemctl start httpd
```

1.1.3. Ověřte, zda je služba aktivní:

```
# systemctl is-active httpd
active
```

nebo

```
# systemctl status httpd
```

1.1.4. Případně (pro účely testování) vytvořte testovací stránku:

```
# echo "I am $(hostname)." > /var/www/html/index.html
```

### 1.2. Povolení komunikace webového serveru přes firewall

1.2.1. Upravte pravidla firewallu tak, aby umožňovala připojení na standardní porty webového serveru „80“ a „443“:

```
# firewall-cmd --add-service=http --add-service=https --permanent
```

```
success
```

1.2.2. Načtěte novou konfiguraci firewallu:

```
# firewall-cmd --reload  
success
```

1.2.3. Ověřte konfiguraci firewallu pro výchozí (public) zónu:

```
# firewall-cmd --list-all  
public (active)  
  target: default  
  icmp-block-inversion: no  
  interfaces: eth0  
  sources:  
  services: dhcpv6-client http https ssh  
  ports:  
  protocols:  
  masquerade: no  
  forward-ports:  
  source-ports:  
  icmp-blocks:  
  rich rules:
```

1.2.4. Případně zobrazte testovací stránku spuštěním následujícího příkazu z jiného počítače:

```
# curl <IP>  
I am Arnold.
```

(Použijte IP adresu webového serveru.)

## 1.3. Změna portu webového serveru

1.3.1. Změňte výchozí port v konfiguračním souboru webového serveru na „8090“:

```
# sed -i 's/Listen 80/Listen 8090/' /etc/httpd/conf/httpd.conf
```

1.3.2. Povolte port na firewallu:

```
# firewall-cmd --add-port=8090/tcp --permanent
```

1.3.3. Načtěte novou konfiguraci firewallu:

```
# firewall-cmd --reload
```

1.3.4. Povolte port v rámci SELinuxu (pokud se SELinux používá):

```
# semanage port -at http_port_t -p tcp 8090
```

1.3.5. Restartujte webový server:

```
# systemctl restart httpd
```

1.3.6. Ověřte, zda služba naslouchá na novém portu:

```
# netstat -an | grep :8090
tcp        0      0 0.0.0.0:8090          0.0.0.0:*           LISTEN
```

1.3.7. Případně zobrazte testovací stránku spuštěním následujícího příkazu z jiného počítače:

```
# curl <IP>:8090
I am Arnold.
```

(Použijte IP adresu webového serveru.)

---

## 2. Vytvoření vlastní systemd služby jako root

2.1. Vytvořte vlastní soubor jednotky služby systemd v adresáři „/etc/systemd/system“, která bude spouštět skript každých 5 minut:

```
# cat > /etc/systemd/system/swap_check.service

[Unit]
Description=Swap check service

[Service]
Type=oneshot
ExecStart=/usr/local/bin/swap_check.sh
StandardOutput=append:/var/log/swap_check.log
StandardError=append:/var/log/swap_check.log

[Install]
WantedBy=default.target

^D
```

(Soubory jednotek systemd v adresáři „/usr/lib/systemd/system“ jsou poskytovány softwarovými balíčky a jakékoli jejich změny mohou být během aktualizací softwaru přepsány.)

2.2. Vytvořte vlastní soubor jednotky časovače systemd v adresáři „/etc/systemd/system“, který

spustí odpovídající službu:

```
# cat > /etc/systemd/system/swap_check.timer

[Unit]
Description=Activate swap_check.service every 5 minutes

[Timer]
OnCalendar=*:0/5

[Install]
WantedBy=timers.target

^D
```

2.3. Vytvořte patřičný skript a soubor logu:

```
# cat > /usr/local/bin/swap_check.sh
#!/bin/bash

# Get total swap used
total_swap_used=$(free -h | awk '/Swap/ {print $3}')

# Print the result
echo "Date: $(date)"
echo "Total Swap Used: $total_swap_used"
echo

^D
```

```
# chmod +x /usr/local/bin/swap_check.sh
# touch /var/log/swap_check.log
```

2.4. Ověřte, že soubory vlastních systemd jednotek neobsahují žádné chyby:

```
# systemd-analyze verify /etc/systemd/system/swap_check.*
```

(Pokud příkaz nevrátí žádný výstup, soubory úspěšně prošly ověřením.)

2.5. Načtěte novou konfiguraci systemd, aby rozpoznal nově vytvořené soubory jednotek:

```
# systemctl daemon-reload
```

2.6. Spusťte jednotku časovače, která spouští odpovídající službu:

```
# systemctl start swap_check.timer
```

## 2.7. Povolte jednotku časovače po spuštění systému:

```
# systemctl enable swap_check.timer
Created symlink /etc/systemd/system/timers.target.wants/swap_check.timer →
/etc/systemd/system/swap_check.timer.
```

## 2.8. Zkontrolujte stav jednotky časovače:

```
# systemctl status swap_check.timer
● swap_check.timer - Activate swap_check.service every 5 minutes
   Loaded: loaded (/etc/systemd/system/swap_check.timer; enabled; preset:
disabled)
   Active: active (waiting) since Tue 2024-01-30 23:22:13 CET; 12s ago
   Trigger: Tue 2024-01-30 23:25:00 CET; 2min 33s left
   Triggers: ● swap_check.service

Jan 30 23:22:13 arnold systemd[1]: Started swap_check.timer - Activate
swap_check.service every 5 minutes.
```

## 2.9. Zkontrolujte stav jednotky služby:

```
# systemctl status swap_check.service
○ swap_check.service - Swap check service
   Loaded: loaded (/etc/systemd/system/swap_check.service; disabled;
preset: disabled)
   Active: inactive (dead) since Tue 2024-01-30 23:22:16 CET; 20s ago
   Duration: 2.089s
   TriggeredBy: ● swap_check.timer
   Process: 109496 ExecStart=/usr/local/bin/swap_check.sh (code=exited,
status=0/SUCCESS)
   Main PID: 109496 (code=exited, status=0/SUCCESS)
   CPU: 2.525s

Jan 30 23:22:13 arnold systemd[1]: Starting swap_check.service - Swap check
service...
Jan 30 23:22:16 arnold systemd[1]: swap_check.service: Deactivated
successfully.
Jan 30 23:22:16 arnold systemd[1]: Finished swap_check.service - Swap check
service.
Jan 30 23:22:16 arnold systemd[1]: swap_check.service: Consumed 2.525s CPU
time.
```

## 2.10. Zobrazte logy jednotky časovače:

```
# journalctl -u swap_check.timer
```

## 2.11. Zobrazte logy jednotky služby:

```
# journalctl -u swap_check.service
```

2.12. Zobrazte aktualizovaný výstup služby:

```
# tail -f /var/log/swap_check.log
```

### 3. Vytvoření vlastní systemd služby jako běžný uživatel

3.1. Přihlaste se přímo jako konkrétní uživatel (nepoužívejte příkaz „su“ nebo „sudo“).

3.2. Vytvořte konfigurační systemd adresář uživatele (pokud ještě neexistuje):

```
$ mkdir -p ~/.config/systemd/user
```

3.3. Vytvořte vlastní soubor jednotky služby systemd pro ukázkovou aplikaci, např. „myweb.service“:

```
$ cat > ~/.config/systemd/user/myweb.service
```

```
[Unit]
Description=My sample application

[Service]
WorkingDirectory=/home/dookie/myweb/html
ExecStart=/usr/bin/python3 -m http.server 8080
Restart=on-failure

[Install]
WantedBy=default.target

^D
```

3.4. Vytvořte patřičný adresář a soubor:

```
$ mkdir -p ~/myweb/html
$ echo "Hello World" > ~/myweb/html/index.html
```

3.5. Ověřte, že soubor vlastní jednotky služby systemd neobsahuje žádné chyby:

```
$ systemd-analyze --user verify ~/.config/systemd/user/myweb.service
```

(Pokud příkaz nevrátí žádný výstup, soubor úspěšně prošel ověřením.)

3.6. Načtěte novou konfiguraci systemd, aby rozpoznal nově vytvořený soubor jednotky:

```
$ systemctl --user daemon-reload
```

### 3.7. Spusťte službu:

```
$ systemctl --user start myweb
```

### 3.8. Povolte službu:

```
$ systemctl --user enable myweb  
Created symlink  
/home/testuser/.config/systemd/user/default.target.wants/myweb.service →  
/home/testuser/.config/systemd/user/myweb.service.
```

### 3.9. Zkontrolujte stav služby:

```
$ systemctl --user status myweb  
● myweb.service - My sample application  
   Loaded: loaded (/home/testuser/.config/systemd/user/myweb.service;  
disabled; preset: disabled)  
   Active: active (running) since Tue 2024-01-30 09:17:13 CET; 14s ago  
     Main PID: 75650 (python3)  
       Tasks: 1 (limit: 9226)  
      Memory: 11.7M  
         CPU: 56ms  
       CGroup:  
/user.slice/user-1001.slice/user@1001.service/app.slice/myweb.service  
└─75650 /usr/bin/python3 -m http.server 8080  
  
Jan 30 09:17:13 arnold systemd[75548]: Started myweb.service - My sample  
application.
```

### 3.10. Povolte spuštění služby nezávisle na relaci aktivního uživatele:

```
$ loginctl enable-linger $USER
```

(systemd služby běžného uživatele se implicitně ukončí po odhlášení uživatele.)

### 3.11. Případně ověřte konfiguraci:

```
$ loginctl show-user $USER | grep -i linger  
Linger=yes
```

### 3.12. Zobrazte logy služby:

```
$ journalctl --user -u myweb
```

### 3.13. Ověřte, že webová služba funguje na portu 8080:

```
$ curl http://localhost:8080  
Hello World
```

From:

<https://prompt.cz/> - **Prompt.cz**

Permanent link:

<https://prompt.cz/sprava-sluzeb>

Last update: **2024/02/04 21:37**

