Harmonogram zadań – Ubuntu 24.04 Desktop i Server

Cel ogólny lekcji: Nauczenie się planowania zadań przy pomocy narzędzi cron i at.

Cele szczegółowe lekcji:

- Nauczy się, jak skonfigurować i korzystać z narzędzia at w celu zaplanowania zadania na konkretną godzinę.
- Pozna, jak korzystać z narzędzia cron w celu zaplanowania zadań cyklicznych.
- Nauczy się konfigurować zadania crontab dla użytkownika oraz dla konta root.
- Nauczy się, jak przeglądać zaplanowane zadania i usuwać je z listy.
- Będzie w stanie potwierdzić, czy zaplanowane zadania działają poprawnie, przeglądając odpowiednie pliki z logami.

1. Harmonogram zadań narzędziem at

Zainstaluj wymagane pakiety:

```
sudo apt install at finger
```

Sprawdź, czy usługa atd działa:

```
systemctl status atd.service
```

Jeśli usługa nie działa, uruchom ją:

```
systemctl start atd.service
```

Zaplanowanie zadania na konkretną godzinę (np. za 3 minuty):

```
at hh:mm
```

```
finger >> /var/log/syslog
```

Zakończ edytor Ctrl+D.

Sprawdź zaplanowane zadania:

```
atq (lub at -l)
```

Widzimy, że tylko zadanie zaplanowane na 12:00 jest nadal na liście

(zgłoszenie nr1)

Usuń zadanie:

```
atrm job_number
```

2. Harmonogramowanie zadań przy pomocy cron

Sprawdź status usługi cron:

```
systemctl status cron.service
```

Edytuj crontab (domyślnie otwiera się w nano):

```
crontab -e
```

Dodaj zadanie np. co minutę:

```
* * * * * finger >> ~/users.log
```

Sprawdź bieżący czas:

```
date
```

Zapisz godzine i minute.

Sprawdź bieżący dzień tygodnia:

```
date +%u
```

Wynik:

- 1 = poniedziałek
- 2 = wtorek
- 3 = środa
- 4 = czwartek
- 5 = piątek
- 6 = sobota
- 7 = niedziela

Dodaj wpis: [minuta] [godzina] * * [numer_dnia_tygodnia] tar czvf /tmp/etc.tgz /etc

- Zamiast 0 14 * * 2 wpisz godzinę i minutę kilka minut później niż aktualny czas oraz numer aktualnego dnia tygodnia (np. jeśli jest 12:10 w wtorek, wpisz 15 12 * * 2).

```
0 14 * * 2 tar czvf /tmp/etc.tgz /etc
```

Zapisz plik w nano: Ctrl+O, Enter, Ctrl+X.

Wyjaśnienie składni:

- - 0 → minuta (0)
- - 14 → godzina (14:00)
- - * → każdy dzień miesiąca
- - * → każdy miesiąc
- - 2 → dzień tygodnia (wtorek, gdzie 0=niedziela, 1=poniedziałek, 2=wtorek)
- - tar czvf /tmp/etc.tgz /etc → polecenie tworzące archiwum katalogu /etc w /tmp

Sprawdź zadania:

```
crontab -l
```

- Poczekaj do ustawionej godziny i sprawdź efekt poleceniem:

```
ls -lh /tmp/etc.tgz
```

(zgłoszenie nr2)

Usuń crontab:

```
crontab -r
```

Zamykamy wszystkie otwarte okna.

Ważne dla Ubuntu 24.04:

Sprawdź, czy cron działa: systemctl status cron.service Jeśli nie działa:

```
sudo systemctl start cron.service
```

```
sudo systemctl enable cron.service
```

- **Nazwy skryptów:** muszą być zgodne z zasadami run-parts → bez kropek i rozszerzeń (np. my_hourly_script.sh jest OK, ale my.hourly.script.sh NIE).
- **Harmonogram:** Skrypty w cron.hourly są uruchamiane przez run-parts na podstawie wpisu w /etc/crontab: 17 * * * * root cd / && run-parts --report /etc/cron.hourly (czyli około 17 minuty każdej godziny).

3. Skrypty w katalogach cron.x

Kolejne miejsce związane z tematem to katalogi, w których znajdują się skrypty uruchamiane w określonych okresach. Sprawdź, czy są to katalogi:

- /etc/cron.daily – codziennie
- /etc/cron.hourly – co godzinę
- /etc/cron.weekly – raz w tygodniu
- /etc/cron.monthly – raz w miesiącu

Za pocą poleceń:

ls /etc/cron*

ls /etc/ -l |grep cron

```
ubuntu@ubuntu2204:~$ sudo -s
root@ubuntu2204:/home/ubuntu# ls /etc/cron*
/etc/crontab

/etc/cron.d:
anacron  e2scrub_all  sysstat

/etc/cron.daily:
0anacron  apport  apt-compat  dpkg  logrotate  man-db  sysstat

/etc/cron.hourly:

/etc/cron.monthly:
0anacron

/etc/cron.weekly:
0anacron  man-db

/etc/cron.yearly:
root@ubuntu2204:/home/ubuntu# ls /etc/ -l |grep cron
-rw-r--r--  1 root                root                  335 kwi  8  2024 anacrontab
drwxr-xr-x  2 root                root                 4096 kwi 24  2024 cron.d
drwxr-xr-x  2 root                root                 4096 kwi 24  2024 cron.daily
drwxr-xr-x  2 root                root                 4096 kwi 24  2024 cron.hourly
drwxr-xr-x  2 root                root                 4096 kwi 24  2024 cron.monthly
-rw-r--r--  1 root                root                 1136 mar 30  2024 crontab
drwxr-xr-x  2 root                root                 4096 kwi 24  2024 cron.weekly
drwxr-xr-x  2 root                root                 4096 kwi 24  2024 cron.yearly
```

(zgłoszenie nr3)

Przykład dodania skryptu (nie obowiązkowy element ćwiczenia):

nano /etc/cron.daily/my_daily_script.sh

#!/bin/bash

echo 'Codzienny skrypt'

date

```
chmod +x /etc/cron.daily/my_daily_script.sh
```

4. Katalog cron.daily

w systemie Ubuntu zawiera skrypty, które są automatycznie wykonywane codziennie przez demon cron. Możesz umieścić własne skrypty w tym katalogu, a zostaną one uruchamiane raz dziennie zgodnie z harmonogramem cron. Oto przykład korzystania z katalogu cron.daily:

1. Otwórz terminal w systemie Ubuntu.
2. Przejdź do katalogu cron.daily za pomocą polecenia:

```
cd /etc/cron.daily
```

3. Utwórz nowy skrypt bash, na przykład my_daily_script.sh, za pomocą swojego ulubionego edytora tekstowego:

```
nano my_daily_script.sh
```

4. Wpisz treść skryptu, na przykład:

```
#!/bin/bash

echo "To jest mój codzienny skrypt!"

date

# Dodaj inne polecenia, które mają zostać wykonane codziennie
```

Skrypt może zawierać dowolne polecenia, które chcesz wykonać codziennie.

5. Zapisz i zamknij plik.
6. Upewnij się, że skrypt ma odpowiednie uprawnienia do wykonania:

```
chmod +x my_daily_script.sh
```

7. Skrypt my_daily_script.sh zostanie automatycznie wykonany codziennie przez demon cron, o ile masz zainstalowany i uruchomiony daemon cron na swoim systemie.

Należy pamiętać, że skrypty w katalogu cron.daily muszą mieć odpowiednie uprawnienia do wykonania, czyli muszą być wykonywalne. Ważne jest również, aby treść skryptu była poprawnie napisana i testowana, aby uniknąć nieoczekiwanych efektów w systemie.

Uwagi dla Ubuntu 24.04:

- Demon cron musi być aktywny: sprawdzisz to poleceniem:
- Skrypty w katalogu cron.daily są uruchamiane przez run-parts, więc:
  - Nazwa skryptu **nie może zawierać kropek ani rozszerzeń** (np. my_daily_script.sh jest akceptowalna, ale my.daily.script.sh nie).
- Harmonogram uruchamiania skryptów w cron.daily jest kontrolowany przez plik /etc/crontab lub usługę anacron (domyślnie około 06:25 rano).

```
cat my_daily_script.sh
```

(zgłoszenie nr4)

5. Katalog /etc/cron.hourly

w systemie Ubuntu zawiera skrypty, które są automatycznie wykonywane co godzinę przez demon cron. Możesz umieścić własne skrypty w tym katalogu, a zostaną one uruchamiane raz na godzinę zgodnie z harmonogramem cron.

Oto przykład korzystania z katalogu /etc/cron.hourly:

1. Otwórz terminal w systemie Ubuntu.
2. Przejdź do katalogu /etc/cron.hourly za pomocą polecenia:

```
cd /etc/cron.hourly
```

3. Utwórz nowy skrypt bash, na przykład my_hourly_script.sh, za pomocą swojego ulubionego edytora tekstowego:

```
nano my_hourly_script.sh
```

4. Wpisz treść skryptu, na przykład:

```
#!/bin/bash

echo "To jest mój skrypt, który wykonuje się co godzinę!"

date

# Dodaj inne polecenia, które mają zostać wykonane co godzinę
```

Skrypt może zawierać dowolne polecenia, które chcesz wykonywać co godzinę.

5. Zapisz i zamknij plik.

6. Upewnij się, że skrypt ma odpowiednie uprawnienia do wykonania:

```
chmod +x my_hourly_script.sh
```

7. Skrypt my_hourly_script.sh zostanie automatycznie wykonany co godzinę przez demon cron, o ile masz zainstalowany i uruchomiony daemon cron na swoim systemie.

Należy pamiętać, że skrypty w katalogu /etc/cron.hourly muszą mieć odpowiednie uprawnienia do wykonania, czyli muszą być wykonywalne. Ważne jest również, aby treść skryptu była poprawnie napisana i testowana, aby uniknąć nieoczekiwanych efektów w systemie.

```
cat my_hourly_script.sh
```

(zgłoszenie nr5)

6. Katalog cron.monthly

w systemie Ubuntu zawiera skrypty, które są automatycznie wykonywane raz na miesiąc przez demon cron. Możesz umieścić własne skrypty w tym katalogu, a zostaną one uruchamiane raz na miesiąc zgodnie z harmonogramem cron.

Oto przykład korzystania z katalogu cron.monthly:

1. Otwórz terminal w systemie Ubuntu.
2. Przejdź do katalogu cron.monthly za pomocą polecenia:

```
cd /etc/cron.monthly
```

3. Utwórz nowy skrypt bash, na przykład my_monthly_script.sh, za pomocą swojego ulubionego edytora tekstowego:

```
nano my_monthly_script.sh
```

4. Wpisz treść skryptu, na przykład:

```
#!/bin/bash

echo "To jest mój skrypt, który wykonuje się raz na miesiąc!"

date

# Dodaj inne polecenia, które mają zostać wykonane raz na miesiąc
```

Skrypt może zawierać dowolne polecenia, które chcesz wykonywać raz na miesiąc.

5. Zapisz i zamknij plik.
6. Upewnij się, że skrypt ma odpowiednie uprawnienia do wykonania:

```
chmod +x my_monthly_script.sh
```

7. Skrypt my_monthly_script.sh zostanie automatycznie wykonany raz na miesiąc przez demon cron, o ile masz zainstalowany i uruchomiony demon cron na swoim systemie.

Należy pamiętać, że skrypty w katalogu cron.monthly muszą mieć odpowiednie uprawnienia do wykonania, czyli muszą być wykonywalne. Ważne jest również, aby treść skryptu była poprawnie napisana i testowana, aby uniknąć nieoczekiwanych efektów w systemie.

```
cat my_monthly_script.sh
```

(zgłoszenie nr6)

7. Różnice w stosunku do starszych wersji Ubuntu

- Zarządzanie usługami: dawniej /etc/init.d/cron status, teraz systemctl status cron.service.
- Domyślny edytor crontab: dawniej vi, teraz nano.
- Obsługa poleceń systemowych: systemctl zastąpił init.d.

Wnioski z ćwiczenia:

1. at jest narzędziem do jednorazowego planowania zadań na określoną godzinę i minutę, co jest przydatne do zadań punktualnych.
2. cron służy do planowania zadań cyklicznych (np. co minutę, godzinę, w określony dzień tygodnia lub miesiąca).
3. Konfiguracja i zarządzanie różnią się:
   - at wymaga podania konkretnej daty i godziny,
   - cron używa specjalnego formatu w pliku crontab.
4. Zadania cron można definiować jako zwykły użytkownik lub root – ważne ze względu na dostęp do zasobów systemowych.
5. Zarządzanie zadaniami:
   - atq lub at -l → lista zadań at,
   - atrm → usuwanie zadań at,

- crontab -r → usunięcie całego pliku crontab.

6. Katalogi cron.x (cron.daily, cron.hourly, cron.weekly, cron.monthly) pozwalają na automatyczne wykonywanie skryptów z określoną częstotliwością.
7. Skrypty w katalogach cron.x muszą być wykonywalne (chmod +x skrypt.sh) i zgodne z zasadami run-parts (bez kropek w nazwie).
8. Sprawdzenie działania crona jest kluczowe:

systemctl status cron.service

sudo systemctl start cron.service

sudo systemctl enable cron.service

9. Harmonogram w Ubuntu 24.04:
   - cron.hourly uruchamiany około 17 minuty każdej godziny (wg /etc/crontab),
   - cron.daily zwykle rano (np. 06:25).
10. Monitorowanie logów (tail -F ~/users.log) pomaga potwierdzić poprawność działań.
11. Przy planowaniu zadań cyklicznych należy zwracać uwagę na specyfikację czasu i daty, aby uniknąć konfliktów.
12. Dokładność i ostrożność w tworzeniu i zarządzaniu zadaniami są kluczowe, aby uniknąć problemów i zapewnić poprawne działanie.

Po sprawdzeniu przez prowadzącego przywróć pierwszą migawkę.

Podsumowanie:

Po wykonaniu wszystkich czynności z powyższej instrukcji przeczytaj ponownie z zrozumieniem cel ogólny i cele szczegółowe, które znajdują się na pierwszej stronie instrukcji. Jeżeli one zostały niezrealizowane to powtarzaj wykonie tej instrukcji w szkole lub/i w domu do momentu zrealizowania.