

Instrukcja do ćwiczenia

git-annex

Michał Liszcz Jakub Sawicki

27 maja 2016

Data wykonania	
Skład Grupy	
Ocena	

Podczas wykonywania ćwiczenia odznaczaj wykonane podpunkty!

Przed przystąpieniem do ćwiczenia sprawdź obecność i stan sprzętu. Wszelkie nieprawidłowości należy natychmiast zgłosić prowadzącemu.

- 1 Skonfiguruj dwie maszyny wirtualne centos6 z zainstalowanymi odpowiednimi paczkami. Maszyny niech będą dostępne dla siebie pod nazwami `hostA` i `hostB`.

```
docker-compose up
docker-compose exec --user git hostA /bin/bash
git@hostA:~$ ssh-keygen
git@hostA:~$ cat 'git' | ssh-copy-id git@hostB
docker-compose exec --user git hostB /bin/bash
git@hostB:~$ ssh-keygen
git@hostB:~$ cat 'git' | ssh-copy-id git@hostA
```

- 2 Utwórz na obu maszynach repozytoria `hostA-main` i `hostB-main` (gdzie prefiks oznacza maszynę, na której dane repozytorium powinno się znajdować). Remote'y maszyn powinny nawzajem na siebie wskazywać.

```
git@hostA:~$ git init hostA-main.git
git@hostA:~$ cd hostA-main.git
git@hostA:hostA-main.git$ git annex init 'hostA - main'

git@hostB:~$ git init hostB-main.git
git@hostB:~$ cd hostB-main.git
git@hostB:hostB-main.git$ git annex init 'hostB - main'
git@hostB:hostB-main.git$ git remote add hostA-main git@hostA:hostA-main.git
git@hostB:hostB-main.git$ git pull hostA-main master # nie zadziała dopoki nie ma c

git@hostA:hostA-main.git$ git remote add hostB-main git@hostB:hostB-main.git
```

- 3 W repozytorium `hostA-main` utwórz plik `important_file.txt` i dodaj go do indeksu `git-annex`. Uzyskaj dostęp do zawartości pliku na maszynie `hostB`. Zweryfikuj gdzie znajduje się aktualnie plik z `hostA` i `hostB`.

```

git@hostA:hostA-main.git$ echo "important big file" > important_file.txt
git@hostA:hostA-main.git$ git annex add important_file.txt
git@hostA:hostA-main.git$ git commit -m "important_file"

git@hostB:hostB-main.git$ git annex sync
git@hostB:hostB-main.git$ ll
total 4
lrwxrwxrwx 1 git git 178 May 19 10:18 important_file.txt -> .git/annex/objects/3J/Z
git@hostB:hostB-main.git$ cat important_file.txt
cat: important_file.txt: No such file or directory
git@hostB:hostB-main.git$ git annex get important_file.txt
git@hostB:hostB-main.git$ cat important_file.txt
important big file
git@hostB:hostB-main.git$ git annex whereis important_file.txt
whereis important_file.txt (2 copies)
    bad498fb-e968-467a-98a8-c59fd7800932 — hostA-main (hostA - main)
    f0c9f4ba-7550-40ce-9eb0-a325b991d306 — here (hostB - main)
ok

git@hostA:hostA-main.git$ git annex whereis important_file.txt
whereis important_file.txt (1 copy)
    bad498fb-e968-467a-98a8-c59fd7800932 — here (hostA - main)
ok
git@hostA:hostA-main.git$ git annex sync
git@hostA:hostA-main.git$ git annex whereis important_file.txt
whereis important_file.txt (2 copies)
    bad498fb-e968-467a-98a8-c59fd7800932 — here (hostA - main)
    f0c9f4ba-7550-40ce-9eb0-a325b991d306 — hostB-main (hostB - main)
ok

```

- 4 Zmodyfikuj zawartość pliku w `hostB-main` i zweryfikuj jak rozprzestrzeniają się zmiany.

```

git@hostB:hostB-main.git$ echo " modified" >> important_file.txt
bash: important_file.txt: Permission denied
git@hostB:hostB-main.git$ git annex unlock important_file.txt
git@hostB:hostB-main.git$ ll
total 4
-rw-r--r-- 1 git git 20 May 19 10:55 important_file.txt
git@hostB:hostB-main.git$ echo " modified" >> important_file.txt
git@hostB:hostB-main.git$ git commit important_file.txt -m "Modified important_file"
git@hostB:hostB-main.git$ ll
total 4
lrwxrwxrwx 1 git git 178 May 19 10:55 important_file.txt -> .git/annex/objects/wG/2
git@hostB:hostB-main.git$ git annex whereis important_file.txt
whereis important_file.txt (1 copy)
    f0c9f4ba-7550-40ce-9eb0-a325b991d306 — here (hostB - main)
ok

git@hostA:hostA-main.git$ git annex whereis important_file.txt
whereis important_file.txt (2 copies)
    bad498fb-e968-467a-98a8-c59fd7800932 — here (hostA - main)
    f0c9f4ba-7550-40ce-9eb0-a325b991d306 — hostB-main (hostB - main)
ok
git@hostA:hostA-main.git$ git annex sync

```

```
git@hostA:hostA-main.git$ git annex whereis important_file.txt
whereis important_file.txt (1 copy)
    f0c9f4ba-7550-40ce-9eb0-a325b991d306 — hostB-main (hostB - main)
ok
```

- 5 Utwórz kolejne repozytorium **hostB-media**. Może to być repozytorium, które przechowywane będzie na przykład na napędzie USB.

```
git@hostB:~$ git init hostB-media.git
git@hostB:~$ cd hostB-media.git
git@hostB:hostB-media.git$ git annex init 'hostB - media'
git@hostB:hostB-media.git$ git remote add hostB-main /home/git/hostB-main.git
git@hostB:hostB-media.git$ git pull hostB-main master
git@hostB:hostB-media.git$ git annex sync

git@hostB:hostB-main.git$ git remote add hostB-media /home/git/hostB-media.git
git@hostB:hostB-main.git$ git annex sync
```

- 6 Utwórz plik o wielkości 500MB w hostB-main. Zbadaj szybkość transferu do hostA-main i hostB-media.

```
git@hostB:hostB-main.git$ dd if=/dev/zero of=large.bin bs=1024 count=500000
git@hostB:hostB-main.git$ git annex add large.bin
git@hostB:hostB-main.git$ git commit -m "Added large.bin"
git@hostB:hostB-main.git$ git annex copy large.bin hostB-media
copy large.bin (to hostB-media...) ok
(Recording state in git...)
git@hostB:hostB-main.git$ git annex copy large.bin hostA-main
copy large.bin (checking hostA-main...) (to hostA-main...)
SHA256-s512000000--4c98cf638799a07eb85872e7f5f5d8d661c09e6e15af02f3655ed9250cae13b0
    512000000 100%    83.22MB/s    0:00:05 (xfer#1, to-check=0/1)

sent 512062644 bytes  received 31 bytes  78778873.08 bytes/sec
total size is 512000000  speedup is 1.00
ok
```

- 7 Zbadaj działanie flagi **numcopies**. Zwróć uwagę na to, że wersja git-annex dostępna w repozytorium centos6 nie implementuje jeszcze polecenia **numcopies**.

```
git@hostB:hostB-main.git$ git config annex.numcopies 2 # konfig sie nie propaguje p
git@hostB:hostB-main.git$ git annex copy important_file.txt --to hostA-main
git@hostB:hostB-main.git$ git annex drop important_file.txt
drop important_file.txt (checking hostA-main...) (unsafe)
    Could only verify the existence of 1 out of 2 necessary copies

    No other repository is known to contain the file.

    (Use --force to override this check, or adjust annex.numcopies.)
failed
git-annex: drop: 1 failed
```