

Instrukcja do ćwiczenia

git-annex

Michał Liszcz Jakub Sawicki

31 maja 2016

wersja dla prowadzącego

- 1 Skonfiguruj dwie maszyny wirtualne **centos6** z zainstalowanymi następującymi paczkami (git-annex dostępny jest w repozytorium EPEL ¹, pozostałe są w standardowych repozytoriach):

- `git >= 1.7.1`
- `git-annex >= 3.20120522`

Dodaj w pliku `/etc/hosts` wpisy, by maszyny były dostępne pod nazwami `hostA` i `hostB`.

```
yum upgrade ca-certificates
```

```
wget http://download.fedoraproject.org/pub/epel/6/x86_64/ \
    epel-release-6-8.noarch.rpm
rpm -ivh epel-release-6-8.noarch.rpm
```

- 2 Na obu maszynach skonfiguruj serwer SSH, utwórz użytkownika `git` oraz utwórz dla niego parę kluczy (`ssh-keygen`) Wymień klucze publiczne między maszynami (`ssh-copy-id`) tak, by było możliwe logowanie przy ich pomocy.

Sprawdź czy z każdej maszyny możesz zalogować się na drugą z nich:

```
git@hostA:~$ ssh git@hostB
git@hostB:~$ ssh git@hostA
```

- 3 Utwórz na obu maszynach repozytoria `hostA-main` i `hostB-main` (gdzie prefiks oznacza maszynę, na której dane repozytorium powinno się znajdować). w każdym z repozytoriów zainicjalizuj git-annex i dodaj drugie repozytorium jako remote.

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

...

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

¹CentOS6.2 dostępny w laboratorium nie posiada go w repozytorium. Konieczne jest ściągnięcie i zainstalowanie paczki `epel-release-6-8.noarch`. Pomocne może być także uaktualnienie certyfikatów: `yum upgrade ca-certificates`.

- 4 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**.
- 5 Zmodyfikuj zawartość pliku w **hostB-main** i zweryfikuj jak rozprzestrzeniają się zmiany.
- 6 Utwórz kolejne repozytorium **hostB-backup**. Powinno to być repozytorium, które przecho-
wywane będzie na dysku backupowym **/dev/sdb** (utwórz odpowiednią partycję i zamontuj
w **/mnt/backup**).

```
hostB$ fdisk /dev/sdb -> utworzenie nowej partycji
hostB$ mkfs.ext4 /dev/sdb1
hostB$ mkdir /mnt/backup
hostB$ mount /dev/sdb1 /mnt/backup
```

- 7 Utwórz plik **plik.bin** o wielkości 500MB w **hostB-main**. Zbadaj szybkość transferu przy
kopiowaniu pliku do **hostA-main** (przez SSH) i do **hostB-media** (kopia lokalna).

	prędkość transferu [MB/s]
przez SSH	
kopia lokalna	

Przykładowe wyniki:

```
[root@lab429-08 hostB-main.git]# git annex copy --to hostA-main plik.bin
copy plik.bin (checking hostA-main...) (to hostA-main...) SHA256-s512000000--4c98cf638799a07eb85872e7f5f5d
512000000 100\% 45.98MB/s 0:00:10 (xfer#1, to-check=0/1)
```

```
sent 512062644 bytes received 31 bytes 48767873.81 bytes/sec
total size is 512000000 speedup is 1.00
ok
(Recording state in git...)
```

```
[root@lab429-08 hostB-main.git]# git annex copy --to hostB-backup plik.bin
copy plik.bin (to hostB-backup...) SHA256-s512000000--4c98cf638799a07eb85872e7f5f5d
512000000 100\% 115.10MB/s 0:00:04 (xfer#1, to-check=0/1)
```

```
sent 512062644 bytes received 31 bytes 113791705.56 bytes/sec
total size is 512000000 speedup is 1.00
ok
(Recording state in git...)
```

- 8 Zbadaj działanie flagi **numcopies**. Zwróć uwagę na to, że wersja git-annex dostępna w
repozytorium centos6 nie implementuje jeszcze polecenia **numcopies**.
 - Ustaw flagę **numcopies** na 2.
 - Następnie utwórz w wybranym repozytorium bardzo ważny plik (**my_data.txt**) i sko-
piuj go do jednego z pozostałych repozytoriów, tak by w systemie były dwie kopie.
 - Spróbuj porzucić plik lokalnie i obserwuj wyniki.
- 9 Umieść plik z poprzedniego punktu w dwóch zdalnych repozytoriach i porzuć lokalną kopię.
Spróbuj pobrać ją z flagą **get -auto**. Zasymuluj awarię jednego ze zdalnych repozytoriów
a następnie usuń je z indeksu git-annex. Sprawdź ile kopii pliku istnieje w systemie. Czy
teraz możesz pobrać go do lokalnego repozytorium z użyciem flagi **-auto**?

```
[root@lab429-08 hostB-main.git]# git annex get --auto plik.bin
get plik.bin (from hostB-backup...) SHA256-s512000000--4c98cf638799a07eb85872e7f5f5d8
512000000 100\% 56.26MB/s 0:00:08 (xfer#1, to-check=0/1)

sent 512062644 bytes received 31 bytes 53901334.21 bytes/sec
total size is 512000000 speedup is 1.00
ok
(Recording state in git...)
```

- 10 Zasymuluj zmianę którejś z kopii pliku `plik.bin`. Upewnij się, że posiadasz co najmniej dwie kopie i zmodyfikuj zawartość pliku bezpośrednio w systemie plików. Wykorzystaj `git-annex` do wykrycia niepoprawnej kopii, jej usunięcia z systemu i odtworzenia poprawnej kopii.

```
hostB-main.git$ git annex fsck --from hostB-backup plik.bin
fsck plik.bin Bad file size (7 B larger); dropped from hostB-backup
```

```
Only 1 of 2 trustworthy copies exist of plik.bin
Back it up with git-annex copy.
failed
(Recording state in git...)
git-annex: fsck: 1 failed
```

```
hostB-main.git$ git annex get --auto plik.bin
get plik.bin (from hostA-main...) SHA256-s512000000--4c98cf638799a07eb85872e7f5f5d8
512000000 100\% 45.54MB/s 0:00:10 (xfer#1, to-check=0/1)

sent 30 bytes received 512062649 bytes 44527189.48 bytes/sec
total size is 512000000 speedup is 1.00
ok
(Recording state in git...)
```

- 11 **Zadanie dodatkowe.** Załóż konto w serwisie <https://gitlab.com> ². Gitlab wspiera tylko autentykację przy użyciu klucza publicznego. Dodaj w panelu ustawień konta klucz wygenerowany na początku ćwiczenia. Utwórz repozytorium i dodaj je jako remote do istniejącego repozytorium. Zsynchronizuj status git-annex. Zbadaj szybkość transferu 50MB pliku do Gitlab.com, do drugiej maszyny w sieci lokalnej i do repozytorium na tej samej maszynie.

	prędkość transferu [MB/s]
przez SSH	
kopia lokalna	
Gitlab.com	

²Gitlab.com oferuje 10GB darmowego miejsca na przechowywanie plików z użyciem systemu git-annex