GNU Parallel Implementation of Freebayes

By Pu Zheng
2019.05.07

Setup a node

1. Create instances

   Create c4.xlarge instance with Ubuntu 16.04
   Instance should within a VPC with subnet mask 172.30.8.0/24
   Instance should within a subnet inside previous VPC at 172.30.8.0/28
   Allow auto assign public IP
   Instance should have security group with:
   Inbound:

type	Protocol	Port Range	Source	Description
Custom TCP Rule	TCP	10000 - 10100	0.0.0.0/0	
Custom TCP Rule	TCP	10000 - 10100	::/0	
SSH	TCP	22	0.0.0.0/0	
SSH	TCP	22	::/0	
NFS	TCP	2049	0.0.0.0/0	
NFS	TCP	2049	::/0	

Outbound:

Type	Protocol	Port Range	Destination	Description
All traffic	All	All	0.0.0.0/0	

(install packages and modify hosts can be done in batch)

1. Install packages

```
sudo apt-get update
sudo apt-get --assume-yes install parallel zlib1g-dev libbz2-dev liblzma-dev bamtools
sudo apt-get --assume-yes install make cmake gcc g++ awscli samtools nfs-common
git clone --recursive git://github.com/ekg/freebayes.git
cp /home/ubuntu/freebayes/vcflib/tabixpp/tabix.hpp /home/ubuntu/freebayes/vcflib/src/
cd freebayes
make
make install
```

2. Modify hosts file

```
sudo vim /etc/hosts
```

   Add the following:

```
172.30.8.8 ip-172-30-8-8 master
172.30.8.10 ip-172-30-8-10 node1
172.30.8.4 ip-172-30-8-6 node2
```

```
172.30.8.9 ip-172-30-8-9 node3
172.30.8.12 ip-172-30-8-12 node4
```

Add this command to remove some warnings:

```
sudo -- sh -c "echo 127.0.1.1 $(hostname) >> /etc/hosts"
```

Notice: now you can ssh to master by:

```
ssh freebayes@master
```

However, you still need to type in password. So we need to remove it.

3. Modify ssh

   Type in:

   ```
   sudo vim /etc/ssh/sshd_config
   ```

   Go to line:

   ```
   # Change to no to disable tunnelled clear text passwords
   PasswordAuthentication yes
   ```

   Make sure this is yes

   Update the configuration by:

   ```
   sudo service ssh restart
   ```

4. Add new user

   ```
   sudo adduser freebayes
   sudo adduser freebayes sudo
   ```

5. Create ssh-key to skip typing in password during ssh to master

   ```
   ssh-keygen (use default values by pressing many enters)
   ssh-copy-id freebayes@master
   ```

   Now you can ssh to master without typing in password!

6. Map NFS data folder (make sure nfs-common is already installed)

   ```
   su - freebayes
   mkdir data
   sudo mount -t nfs master:/home/freebayes/data /home/freebayes/data
   ```

Special setups for master node

1. SSH connection

   Password free SSH connection should be built from master to every worker node, therefore:

   ```
   ssh-keygen
   ssh-copy-id freebayes@node1
   ssh-copy-id freebayes@node2
   ssh-copy-id freebayes@node3
   ……
   ```

2. NFS server

   ```
   sudo apt-get install nfs-kernel-server
   ```

```
sudo -- sh -c "echo /home/freebayes/data \
*(rw,sync,no_root_squash,no_subtree_check)>> /etc/exports"
sudo exportfs -a
sudo service nfs-kernel-server restart
```

3. Download data

```
sudo wget
ftp://ftp.1000genomes.ebi.ac.uk/vol1/ftp/phase3/data/NA21141/alignment/NA21141.chrom20.ILLUMINA.bwa.GIH.low_coverage.20130415.bai
sudo wget
ftp://ftp.1000genomes.ebi.ac.uk/vol1/ftp/phase3/data/NA21141/alignment/NA21141.chrom20.ILLUMINA.bwa.GIH.low_coverage.20130415.bam
sudo wget
ftp://ftp.1000genomes.ebi.ac.uk/vol1/ftp/phase3/data/NA21141/alignment/NA21141.chrom20.ILLUMINA.bwa.GIH.low_coverage.20130415.bas
```

4. Download reference

```
sudo wget
ftp://ftp.1000genomes.ebi.ac.uk/vol1/ftp/technical/reference/human_g1k_v37.fasta.gz
sudo wget
ftp://ftp.1000genomes.ebi.ac.uk/vol1/ftp/technical/reference/human_g1k_v37.fasta.fai
sudo gunzip human_g1k_v37.fasta.gz
awk 'BEGIN {RS=">"} /20 / {print ">"$0}' human_g1k_v37.fasta > chr20.fa
```

5. Create a hostfile so you can call sub-nodes in GNU parallel:
    a. Write the following info into a file called hostfile:

    ```
    # four cores in local node
    @mastergroup/4/:
    # four cores in worker node
    @workergroup+g1+g2+g4/4/freebayes@node1
    @workergroup+g2+g4/4/freebayes@node2
    @workergroup+g4/4/freebayes@node3
    @workergroup+g4/4/freebayes@node4
    ……
    ```

    b. Check corresponding host names in this hostfile:

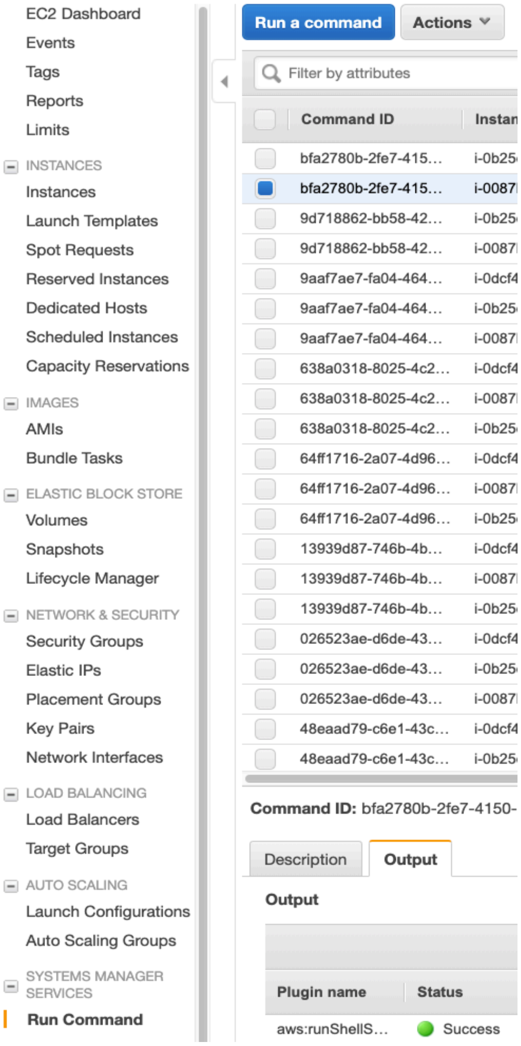
    ```
    parallel --nonall --slf hostfile hostname
    ```

    You should see output like this when its fully setup:

    ```
    ip-172-30-8-8
    ip-172-30-8-10
    ip-172-30-8-9
    ip-172-30-8-4
    ip-172-30-8-12
    ```

Batch Install packages

1. Go to run command



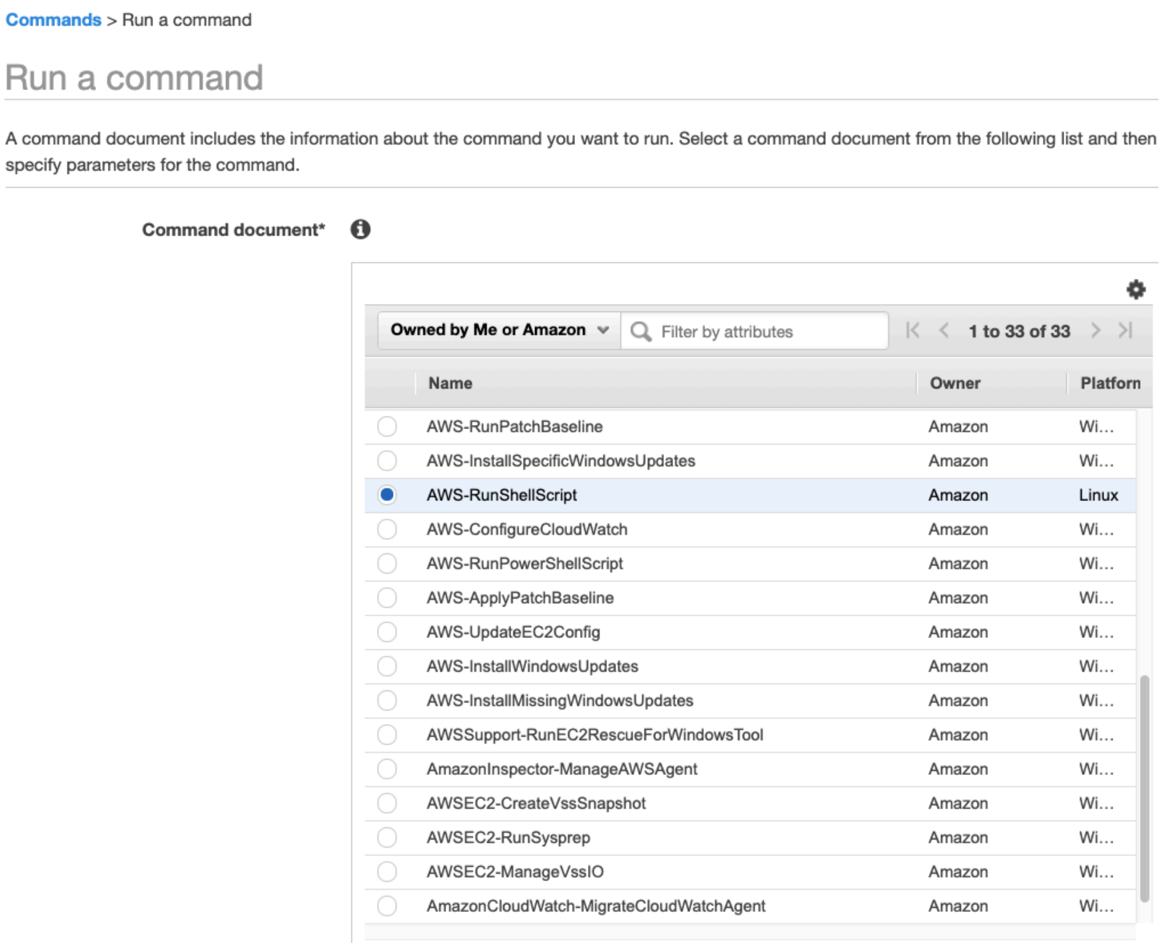
□	Command ID	Instan
□	bfa2780b-2fe7-415…	i-0b25
☑	bfa2780b-2fe7-415…	i-0087
□	9d718862-bb58-42…	i-0b25
□	9d718862-bb58-42…	i-0087
□	9aaf7ae7-fa04-464…	i-0dcf4
□	9aaf7ae7-fa04-464…	i-0b25
□	9aaf7ae7-fa04-464…	i-0087
□	638a0318-8025-4c2…	i-0dcf4
□	638a0318-8025-4c2…	i-0087
□	638a0318-8025-4c2…	i-0b25
□	64ff1716-2a07-4d96…	i-0dcf4
□	64ff1716-2a07-4d96…	i-0087
□	64ff1716-2a07-4d96…	i-0b25
□	13939d87-746b-4b…	i-0dcf4
□	13939d87-746b-4b…	i-0087
□	13939d87-746b-4b…	i-0b25
□	026523ae-d6de-43…	i-0dcf4
□	026523ae-d6de-43…	i-0b25
□	026523ae-d6de-43…	i-0087
□	48eaad79-c6e1-43c…	i-0dcf4
□	48eaad79-c6e1-43c…	i-0b25

Command ID: bfa2780b-2fe7-4150-

Output

Plugin name	Status
aws:runShellS…	● Success

2. Select command document to be "AWS-RunShellScript"



Commands > Run a command

Run a command

A command document includes the information about the command you want to run. Select a command document from the following list and then specify parameters for the command.

Command document*

	Name	Owner	Platform
○	AWS-RunPatchBaseline	Amazon	Wi...
○	AWS-InstallSpecificWindowsUpdates	Amazon	Wi...
●	AWS-RunShellScript	Amazon	Linux
○	AWS-ConfigureCloudWatch	Amazon	Wi...
○	AWS-RunPowerShellScript	Amazon	Wi...
○	AWS-ApplyPatchBaseline	Amazon	Wi...
○	AWS-UpdateEC2Config	Amazon	Wi...
○	AWS-InstallWindowsUpdates	Amazon	Wi...
○	AWS-InstallMissingWindowsUpdates	Amazon	Wi...
○	AWSSupport-RunEC2RescueForWindowsTool	Amazon	Wi...
○	AmazonInspector-ManageAWSAgent	Amazon	Wi...
○	AWSEC2-CreateVssSnapshot	Amazon	Wi...
○	AWSEC2-RunSysprep	Amazon	Wi...
○	AWSEC2-ManageVssIO	Amazon	Wi...
○	AmazonCloudWatch-MigrateCloudWatchAgent	Amazon	Wi...

3. Put commands into command box

```
sudo apt-get update
sudo apt-get --assume-yes install parallel zlib1g-dev libbz2-dev liblzma-dev bamtools
sudo apt-get --assume-yes install make cmake gcc g++ awscli samtools nfs-common
cd /home/ubuntu
git clone --recursive git://github.com/ekg/freebayes.git
cp /home/ubuntu/freebayes/vcflib/tabixpp/tabix.hpp /home/ubuntu/freebayes/vcflib/src/
cd freebayes
make
sudo make install
sudo -- sh -c "echo 127.0.1.1 $(hostname) >> /etc/hosts"
sudo -- sh -c "echo 172.30.8.8 ip-172-30-8-8 master >> /etc/hosts"
sudo -- sh -c "echo 172.30.8.10 ip-172-30-8-10 node1>> /etc/hosts"
sudo -- sh -c "echo 172.30.8.4 ip-172-30-8-4 node2 >> /etc/hosts"
sudo -- sh -c "echo 172.30.8.9 ip-172-30-8-9 node3 >> /etc/hosts"
sudo -- sh -c "echo 172.30.8.12 ip-172-30-8-12 node4 >> /etc/hosts"
sudo -- sh -c "echo 172.30.8.13 ip-172-30-8-13 nc1>> /etc/hosts"
sudo -- sh -c "echo 172.30.8.14 ip-172-30-8-14 nc2 >> /etc/hosts"
sudo -- sh -c "echo 172.30.8.7 ip-172-30-8-7 nc3 >> /etc/hosts"
sudo -- sh -c "echo 172.30.8.11 ip-172-30-8-11 nc4 >> /etc/hosts
```

4. Run, if there is a green light with success, the commands are correctly processed.

Run freebayes

1. Generate input region file by Splice index file a python script in Freebayes:

   ```
   Python ~/freebayes/scripts/fasta_generate_regions.py /home/freebayes/data/chr20.fa.fai
   1000000 > chr20_splice_1000000.fai
   ```

2. Shell scripts:
   a. First level wrapper to splice bam file and feed bam and a specific region into one freebayes: run_freebayes.sh
   b. GNU parallel of the first bam, which allows other inputs from GNU parallel: freebayes-gnu.sh
3. Data: /home/freebayes/data/NA21141.chrom20.ILLUMINA.bwa.GIH.low_coverage.20130415.bam

4. Run single-thread freebayes:

   ```
   time freebayes -f /home/freebayes/data/chr20.fa -v
   /home/freebayes/data/results/NA21141_chr20_serial.vcf
   /home/freebayes/data/NA21141.chrom20.ILLUMINA.bwa.GIH.low_coverage.20130415.bam
   ```

5. Run old freebayes-parallel:

   ```
   time ./freebayes-parallel /home/freebayes/chr20_splice_1000000.fai 1 -f
   /home/freebayes/data/chr20.fa -v /home/freebayes/data/results/NA21141_chr20_old_c1.vcf
   /home/freebayes/data/NA21141.chrom20.ILLUMINA.bwa.GIH.low_coverage.20130415.bam
   ```

6. Run GNU freebayes parallel:

   ```
   time ./freebayes-gnu chr20_splice_1000000.fai 4 --sshloginfile hostfile -S @workergroup >
   /home/freebayes/data/results/NA21141_chr20_n8c4.vcf
   ```

References:

[1] Garrison, Erik, and Gabor Marth. "Haplotype-based variant detection from short-read sequencing." *arXiv preprint arXiv:1207.3907* (2012).

[2] GNU Parallel: https://github.com/mmstick/parallel