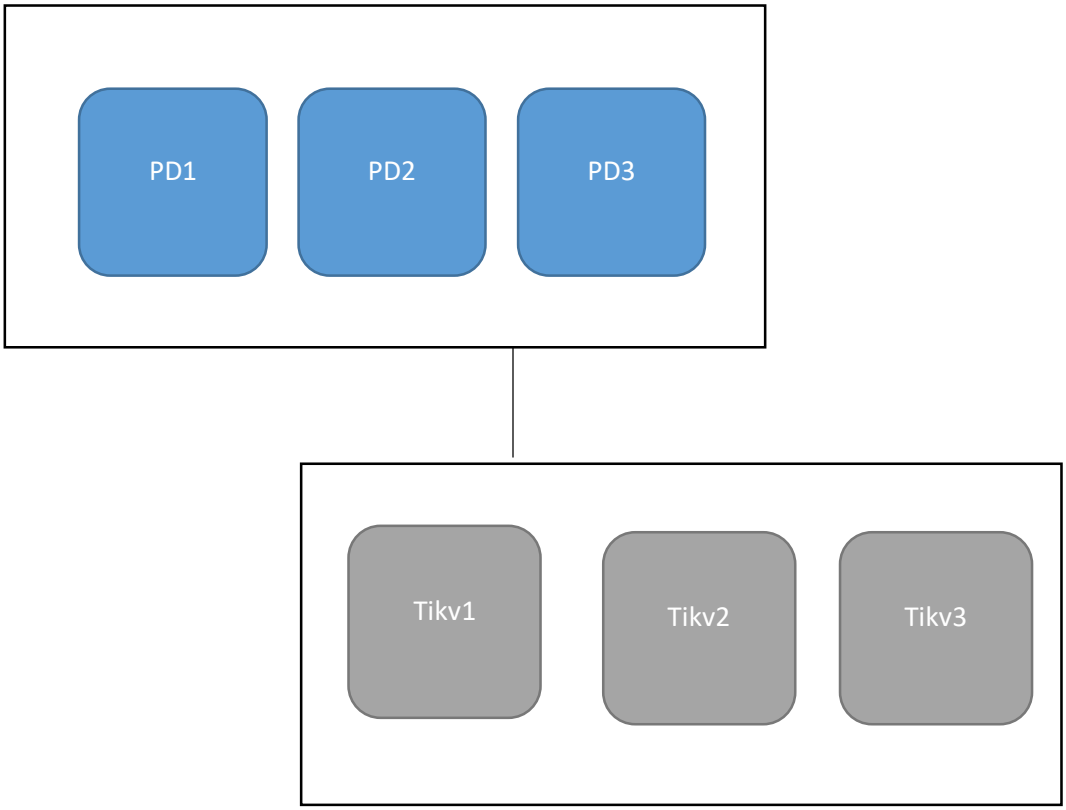
Laporan EAS BDT

0511640000042 Daniel Lumbantobing

1. Arsitektur TiDB



	Pd1	Pd2	Pd3	Tikv1	Tikv2	Tikv3
Plugin	PD, TiDB, node_export, Prometheus, Grafana	PD, node_export	PD, node_export	TiKV, node_export	TiKV, node_export	TiKV, node_export
OS	CentOS 7	CentOS 7	CentOS 7	CentOS 7	CentOS 7	CentOS 7
RAM	512 MB	512 MB	512 MB	512 MB	512 MB	512 MB
IP	192.168.16.42	192.168.16.43	192.168.16.44	192.168.16.45	192.168.16.46	192.168.16.47

2. Implementasi Pada Vagrant

i. Melakukan inisiasi Vagrant Init

ii. Menyiapkan Vagrantfile seperti berikut.

app.py | Vagrantfile

```
# -*- mode: ruby -*-
# vi: set ft=ruby :

Vagrant.configure("2") do |config|
    (1..3).each do |i|
      config.vm.define "pd#{i}" do |node|
        node.vm.hostname = "pd#{i}"

        # Gunakan CentOS 7 dari geerlingguy yang sudah dilengkapi VirtualBox Guest Addition
        node.vm.box = "geerlingguy/centos7"
        node.vm.box_version = "1.2.19"

        # Disable checking VirtualBox Guest Addition agar tidak compile ulang setiap restart
        node.vbguest.auto_update = false

        node.vm.network "private_network", ip: "192.168.16.#{41+i}"

        node.vm.provider "virtualbox" do |vb|
          vb.name = "pd#{i}"
          vb.gui = false
          vb.memory = "512"
        end

        node.vm.provision "shell", path: "provision/bootstrap.sh", privileged: false
      end
    end

    (1..3).each do |i|
      config.vm.define "tikv#{i}" do |node|
        node.vm.hostname = "tikv#{i}"

        # Gunakan CentOS 7 dari geerlingguy yang sudah dilengkapi VirtualBox Guest Addition
        node.vm.box = "geerlingguy/centos7"
        node.vm.box_version = "1.2.19"

        # Disable checking VirtualBox Guest Addition agar tidak compile ulang setiap restart
        node.vbguest.auto_update = false

        node.vm.network "private_network", ip: "192.168.16.#{44+i}"

        node.vm.provider "virtualbox" do |vb|
          vb.name = "tikv#{i}"
          vb.gui = false
          vb.memory = "512"
        end

        node.vm.provision "shell", path: "provision/bootstrap.sh", privileged: false
      end
    end
  end
```

iii. Menyiapkan file provision sebagai berikut.

bootstrap.sh

```
# Referensi:
# https://pingcap.com/docs/stable/how-to/deploy/from-tarball/testing-environment/

# Update the repositories
# sudo yum update -y

# Copy open files limit configuration
sudo cp /vagrant/config/tidb.conf /etc/security/limits.d/

# Enable max open file
sudo sysctl -w fs.file-max=1000000

# Copy atau download TiDB binary dari http://download.pingcap.org/tidb-v3.0-linux-amd64.tar.gz
cp /vagrant/installer/tidb-v3.0-linux-amd64.tar.gz .

# Extract TiDB binary
tar -xzf tidb-v3.0-linux-amd64.tar.gz

# Install MariaDB to get MySQL client
sudo yum -y install mariadb

# Install Git
sudo yum -y install git

# Install nano text editor
sudo yum -y install nano

#Install node exporter
wget https://github.com/prometheus/node_exporter/releases/download/v0.15.2/node_exporter-0.15.2.linux-amd64.tar.gz

#extract node exporter
tar -xzf node_exporter-0.15.2.linux-amd64.tar.gz
```

iv. Melalkukan perintah vagrant up

v. Mengkonfigurasi TiDb Pada masing – masing node :

a. Untuk seluruh PD :

```
./bin/pd-server --name=pd1 \
        --data-dir=pd \
        --client-urls="http://192.168.16.XX:2379" \
        --peer-urls="http://192.168.16.XX:2380" \
        --initial-
cluster="pd1=http://192.168.16.42:2380,pd2=http://192.168.16.43:2380,p
d3=http://192.168.16.44:2380" \
        --log-file=pd.log &

ctt: XX diganti sesuai IP PD
```

b. Untuk seluruh Tikv

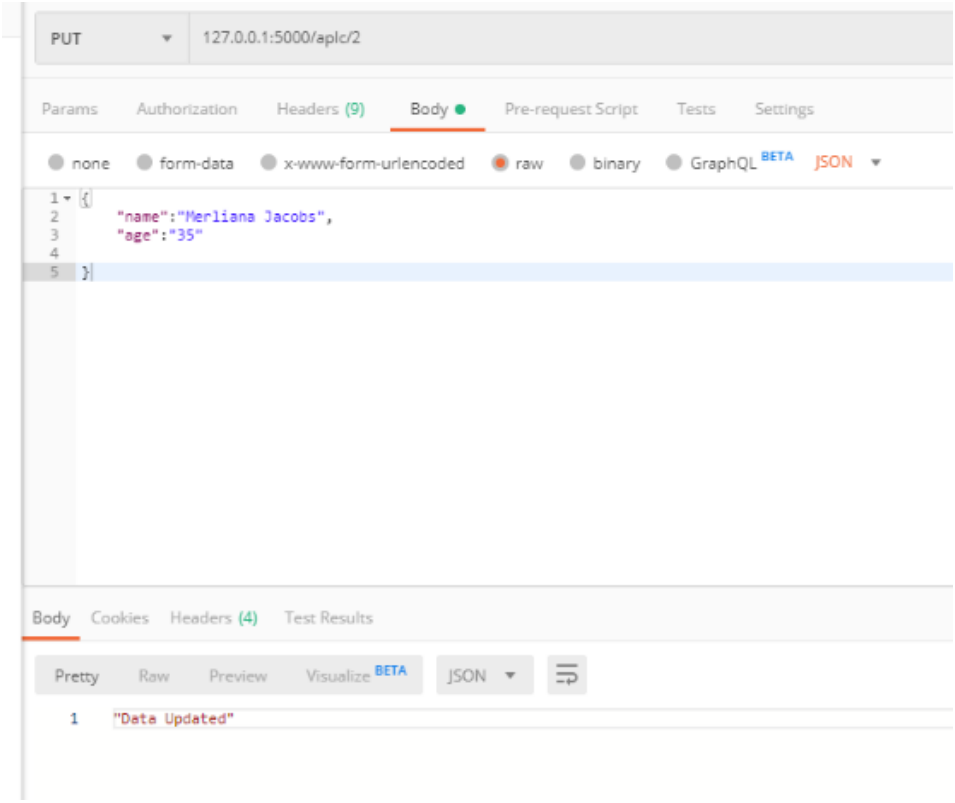
```
./bin/tikv-server --pd="192.168.16.42:2379,192.168.16.43:2379.192.168.16.44:2379"
\
        --addr="192.168.16.XX:20160" \
        --data-dir=tikv \
        --log-file=tikv.log &

ctt: XX diganti sesuai IP masing – masing Tikv
```

vi. Kembali pada pd1, masukan perintah berikut.

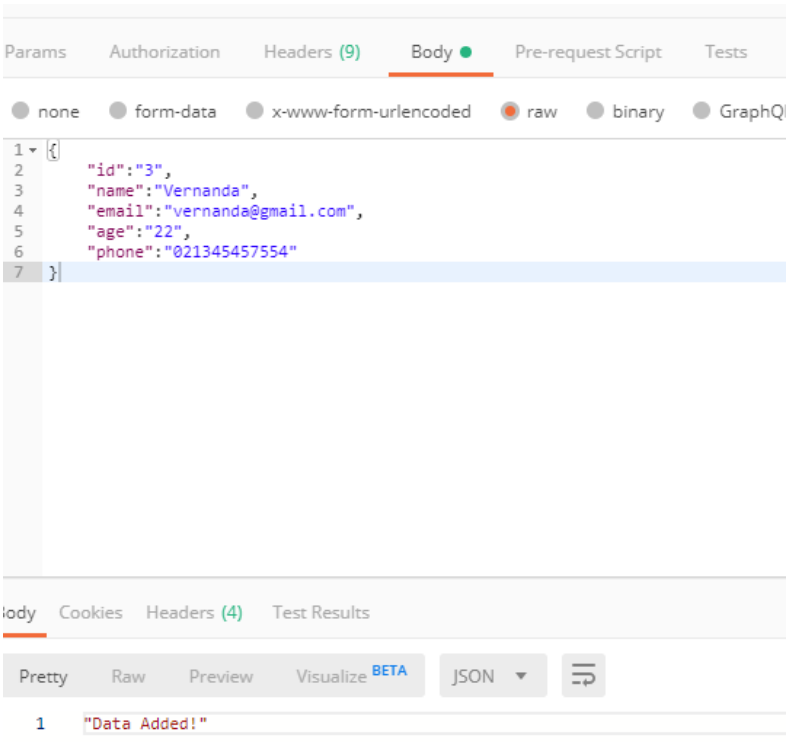
```
./bin/tidb-server --store=tikv \
                  --path="192.168.16.42:2379" \
                  --log-file=tidb.log &
```

3. Aplikasi CRUD (API)

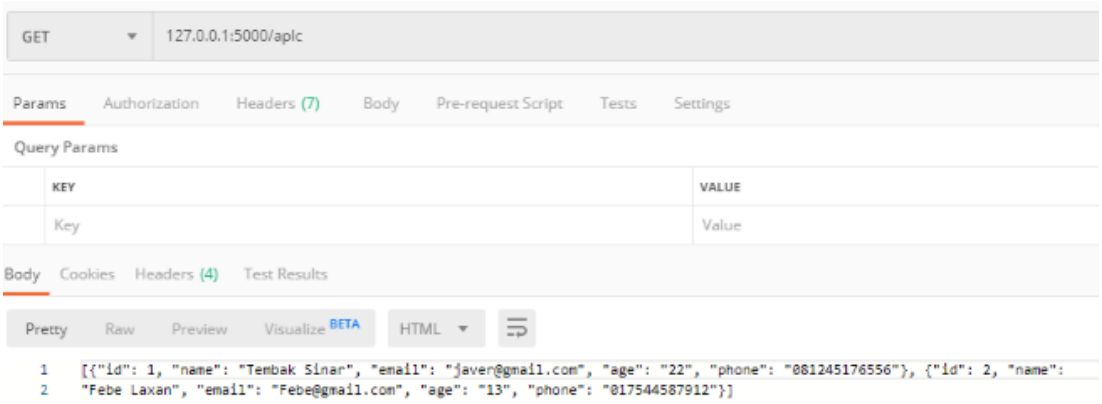
a. **Update Data**



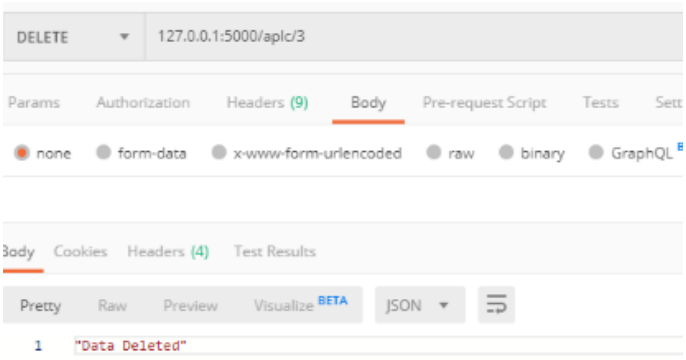
b. **Create**


Params
Authorization
Headers (9)
Body
Pre-request Script
Tests
none
form-data
x-www-form-urlencoded
raw
binary
GraphQ
{
"id":"3",
"name":"Vernanda",
"email":"vernanda@gmail.com",
"age":"22",
"phone":"021345457554"
}
Cookies
Headers (4)
Test Results
Pretty
Raw
Preview
Visualize BETA
JSON
"Data Added!"

c. **Read**

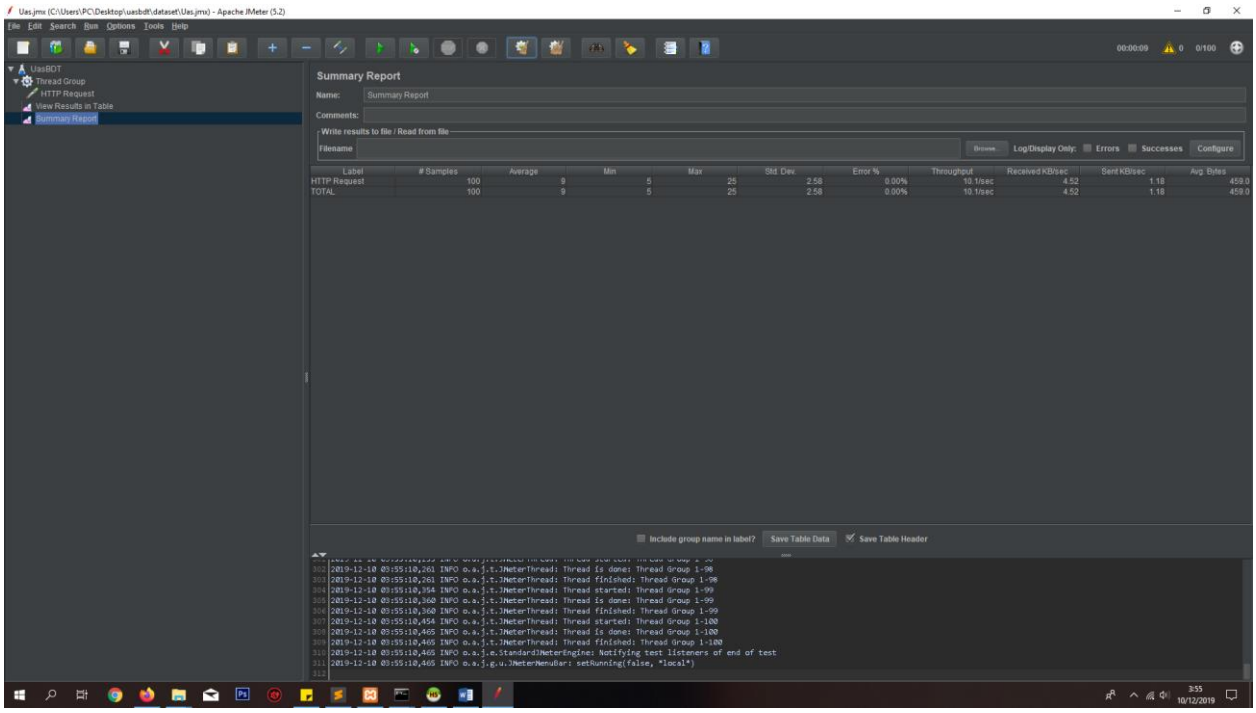


d. **DELETE**

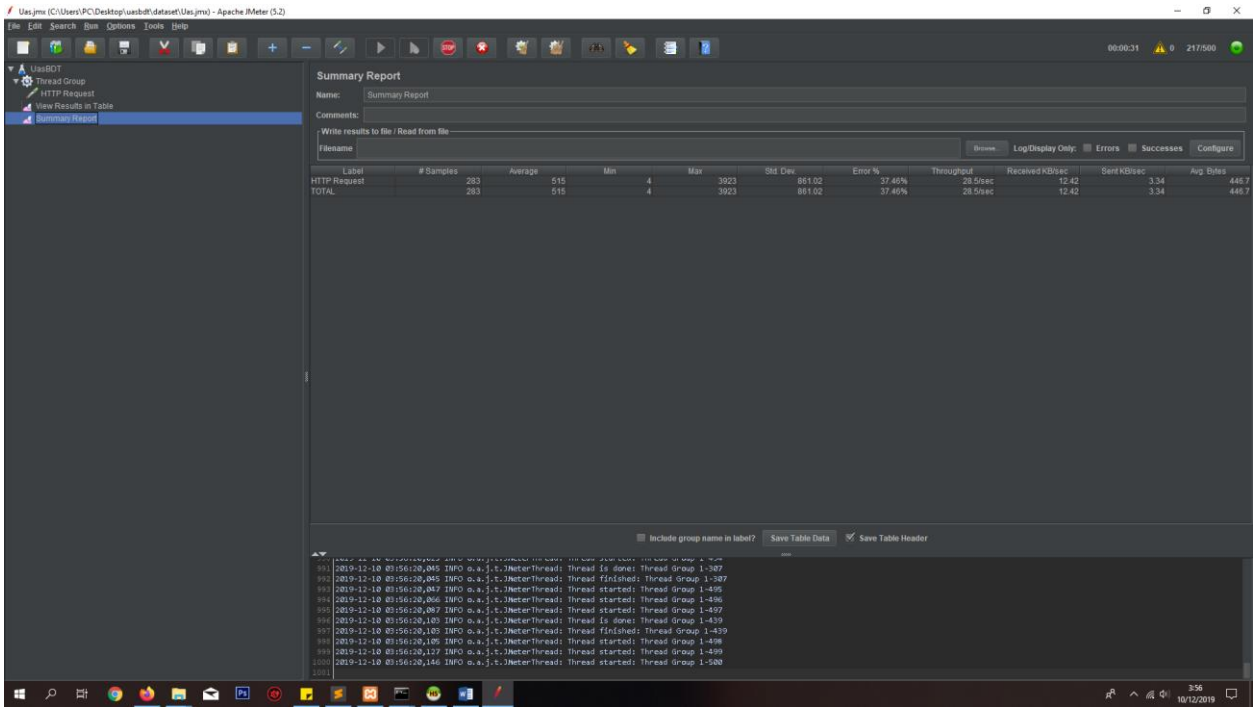


4. Testing dengan JMeter

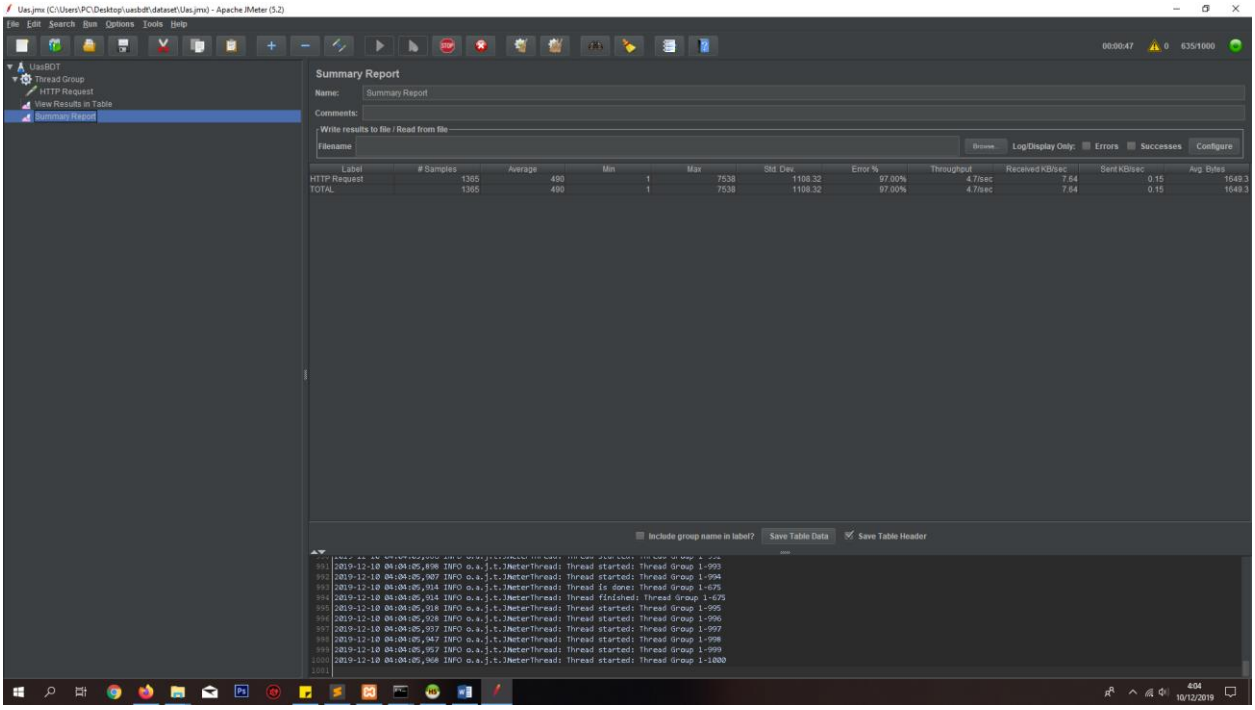
a. 100 User


Uas.jmx (C:\Users\PC\Desktop\uasbdt\dataset\Uas.jmx) - Apache JMeter (5.2)
File Edit Search Run Options Tools Help
UasBDT
Thread Group
HTTP Request
View Results in Table
Summary Report
Summary Report
Name: Summary Report
Comments:
Write results to file / Read from file
Filename
Browse... Log/Display Only: Errors Successes Configure
Label # Samples Average Min Max Std. Dev. Error % Throughput Received KB/sec Sent KB/sec Avg. Bytes
HTTP Request 100 9 5 25 2.58 0.00% 10.1/sec 4.52 1.18 459.0
TOTAL 100 9 5 25 2.58 0.00% 10.1/sec 4.52 1.18 459.0
Include group name in label? Save Table Data Save Table Header
2019-12-10 03:55:10,261 INFO o.a.j.t.JMeterThread: Thread is done: Thread Group 1-98
2019-12-10 03:55:10,261 INFO o.a.j.t.JMeterThread: Thread finished: Thread Group 1-98
2019-12-10 03:55:10,354 INFO o.a.j.t.JMeterThread: Thread started: Thread Group 1-99
2019-12-10 03:55:10,360 INFO o.a.j.t.JMeterThread: Thread is done: Thread Group 1-99
2019-12-10 03:55:10,360 INFO o.a.j.t.JMeterThread: Thread finished: Thread Group 1-99
2019-12-10 03:55:10,454 INFO o.a.j.t.JMeterThread: Thread started: Thread Group 1-100
2019-12-10 03:55:10,465 INFO o.a.j.t.JMeterThread: Thread is done: Thread Group 1-100
2019-12-10 03:55:10,465 INFO o.a.j.t.JMeterThread: Thread finished: Thread Group 1-100
2019-12-10 03:55:10,465 INFO o.a.j.e.StandardJMeterEngine: Notifying test listeners of end of test
2019-12-10 03:55:10,465 INFO o.a.j.g.u.JMeterMenuBar: setRunning(false, *local*)
3:55
10/12/2019

b. 500 User


Uas.jmx (C:\Users\PC\Desktop\uasbdt\dataset\Uas.jmx) - Apache JMeter (5.2)
File Edit Search Run Options Tools Help
00:00:31 0 217/500
UasBDT
Thread Group
HTTP Request
View Results in Table
Summary Report
Summary Report
Name: Summary Report
Comments:
Write results to file / Read from file
Filename
Browse... Log/Display Only: Errors Successes Configure
Label # Samples Average Min Max Std. Dev. Error % Throughput Received KB/sec Sent KB/sec Avg. Bytes
HTTP Request 283 515 4 3923 861.02 37.46% 28.5/sec 12.42 3.34 446.7
TOTAL 283 515 4 3923 861.02 37.46% 28.5/sec 12.42 3.34 446.7
Include group name in label? Save Table Data Save Table Header
2019-12-10 03:56:20,045 INFO o.a.j.t.JMeterThread: Thread is done: Thread Group 1-307
2019-12-10 03:56:20,045 INFO o.a.j.t.JMeterThread: Thread finished: Thread Group 1-307
2019-12-10 03:56:20,087 INFO o.a.j.t.JMeterThread: Thread started: Thread Group 1-495
2019-12-10 03:56:20,066 INFO o.a.j.t.JMeterThread: Thread started: Thread Group 1-496
2019-12-10 03:56:20,087 INFO o.a.j.t.JMeterThread: Thread started: Thread Group 1-497
2019-12-10 03:56:20,103 INFO o.a.j.t.JMeterThread: Thread is done: Thread Group 1-439
2019-12-10 03:56:20,103 INFO o.a.j.t.JMeterThread: Thread finished: Thread Group 1-439
2019-12-10 03:56:20,105 INFO o.a.j.t.JMeterThread: Thread started: Thread Group 1-498
2019-12-10 03:56:20,127 INFO o.a.j.t.JMeterThread: Thread started: Thread Group 1-499
2019-12-10 03:56:20,146 INFO o.a.j.t.JMeterThread: Thread started: Thread Group 1-500
3:56
10/12/2019

c. 1000 User



5. Testing Sysbench

a. Instalasi Sysbench

i. Masuk ke Pd1 dan masukan perintah berikut.

```
curl -s
https://packagecloud.io/install/repositories/akopytov/sysbench/script.rpm.sh |
sudo bash
sudo yum -y install sysbench
```

ii. Clone repo tidb bench

```
git clone https://github.com/pingcap/tidb-bench.git
cd tidb-bench/sysbench
```

b. Pengujian

i. Modifikasi file config dengan mengubah database sesuai yang digunakan

ii. Melakukan persiapan dengan `./run.sh point_select prepare 100`

iii. Lalu run dengan `./run.sh point_select run 100`

c. Hasil

i. Dengan 3 PD menyala

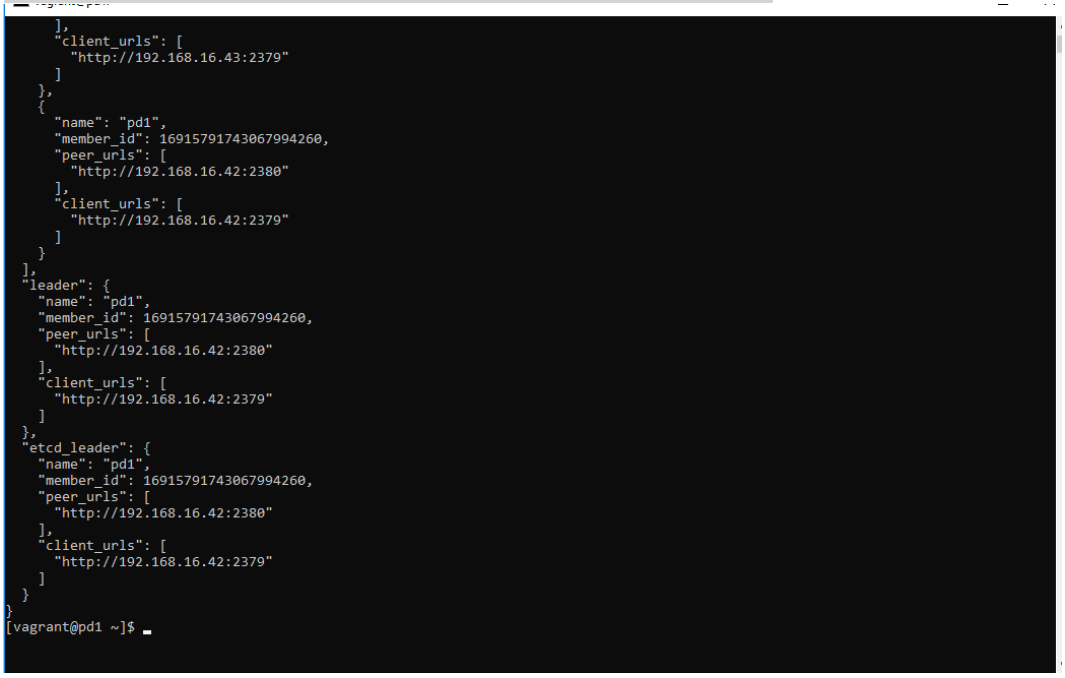
```
  GNU nano 2.3.1                    File: point_select_run_100.log

[ 230s ] thds: 100 tps: 6977.11 qps: 6977.11 (r/w/o: 6977.11/0.00/0.00) lat (ms,95%): 26.20 err/s: 0.00 reconn/s: 0.00
[ 240s ] thds: 100 tps: 6915.06 qps: 6915.06 (r/w/o: 6915.06/0.00/0.00) lat (ms,95%): 27.17 err/s: 0.00 reconn/s: 0.00
[ 250s ] thds: 100 tps: 7016.12 qps: 7016.12 (r/w/o: 7016.12/0.00/0.00) lat (ms,95%): 26.20 err/s: 0.00 reconn/s: 0.00
[ 260s ] thds: 100 tps: 7215.90 qps: 7215.90 (r/w/o: 7215.90/0.00/0.00) lat (ms,95%): 25.28 err/s: 0.00 reconn/s: 0.00
[ 270s ] thds: 100 tps: 7387.87 qps: 7387.87 (r/w/o: 7387.87/0.00/0.00) lat (ms,95%): 24.38 err/s: 0.00 reconn/s: 0.00
[ 280s ] thds: 100 tps: 7209.81 qps: 7209.81 (r/w/o: 7209.81/0.00/0.00) lat (ms,95%): 25.74 err/s: 0.00 reconn/s: 0.00
[ 290s ] thds: 100 tps: 6715.75 qps: 6715.75 (r/w/o: 6715.75/0.00/0.00) lat (ms,95%): 26.68 err/s: 0.00 reconn/s: 0.00
[ 300s ] thds: 100 tps: 7095.04 qps: 7095.04 (r/w/o: 7095.04/0.00/0.00) lat (ms,95%): 25.74 err/s: 0.00 reconn/s: 0.00
SQL statistics:
    queries performed:
        read:                            2151779
        write:                           0
        other:                           0
        total:                           2151779
    transactions:                        2151779 (7171.53 per sec.)
    queries:                             2151779 (7171.53 per sec.)
    ignored errors:                      0      (0.00 per sec.)
    reconnects:                          0      (0.00 per sec.)

General statistics:
    total time:                          300.0432s
    total number of events:              2151779

Latency (ms):
         min:                                    0.52
         avg:                                   13.94
         max:                                  106.65
         95th percentile:                       25.74
         sum:                                29997496.09

Threads fairness:
    events (avg/stddev):           21517.7900/58.16
    execution time (avg/stddev):   299.9750/0.01
```

ii. Dengan 2 PD menyala

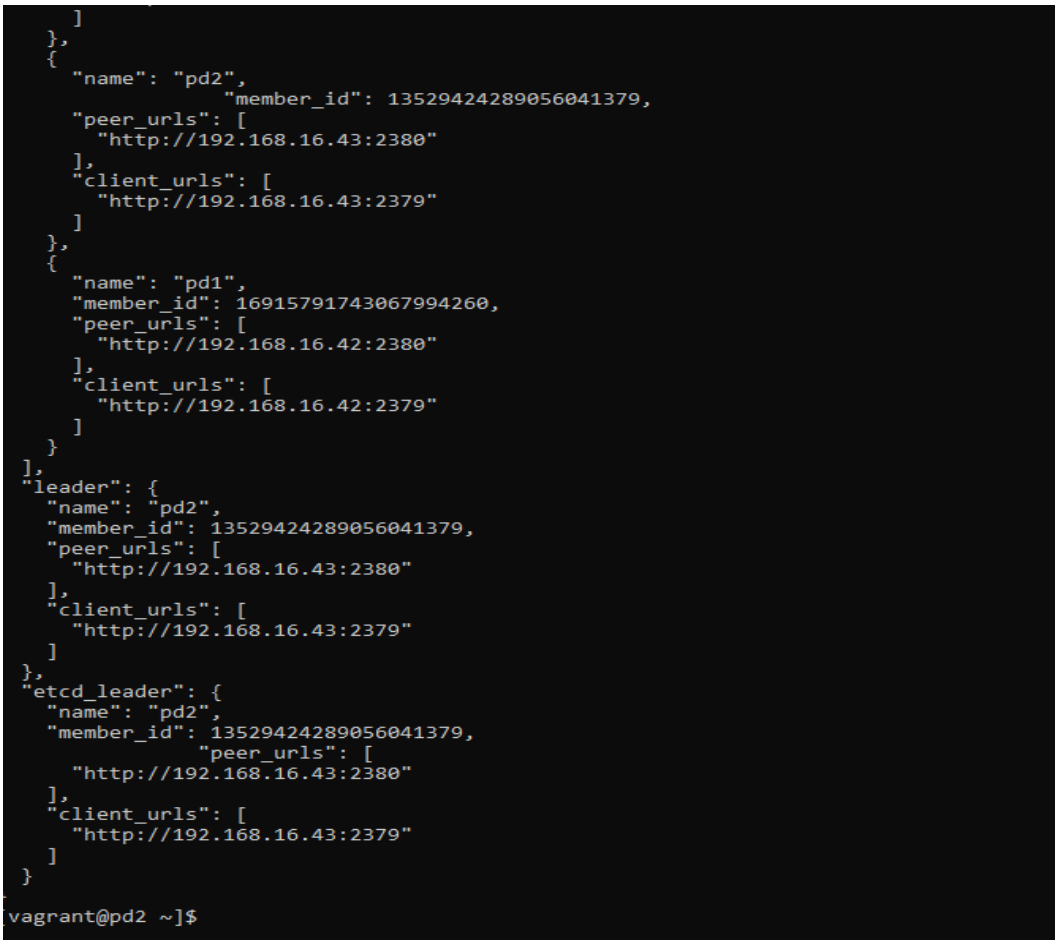
```
250s ] thds: 100 tps: 7416.31 qps: 7416.31 (r/w/o: 7416.31/0.00/0.00) lat (ms,95%): 24.83 err/s: 0.00 reconn/s: 0.00
260s ] thds: 100 tps: 7342.18 qps: 7342.18 (r/w/o: 7342.18/0.00/0.00) lat (ms,95%): 24.83 err/s: 0.00 reconn/s: 0.00
270s ] thds: 100 tps: 7206.89 qps: 7206.89 (r/w/o: 7206.89/0.00/0.00) lat (ms,95%): 25.28 err/s: 0.00 reconn/s: 0.00
280s ] thds: 100 tps: 7172.20 qps: 7172.20 (r/w/o: 7172.20/0.00/0.00) lat (ms,95%): 24.83 err/s: 0.00 reconn/s: 0.00
290s ] thds: 100 tps: 7309.71 qps: 7309.71 (r/w/o: 7309.71/0.00/0.00) lat (ms,95%): 25.28 err/s: 0.00 reconn/s: 0.00
300s ] thds: 100 tps: 7335.28 qps: 7335.28 (r/w/o: 7335.28/0.00/0.00) lat (ms,95%): 25.28 err/s: 0.00 reconn/s: 0.00
QL statistics:
    queries performed:
        read:                            2203326
        write:                           0
        other:                           0
        total:                           2203326
    transactions:                        2203326 (7343.46 per sec.)
    queries:                             2203326 (7343.46 per sec.)
    ignored errors:                      0      (0.00 per sec.)
    reconnects:                          0      (0.00 per sec.)

eneral statistics:
    total time:                          300.0376s
    total number of events:              2203326

atency (ms):
         min:                                    0.48
         avg:                                   13.61
         max:                                  105.09
         95th percentile:                       24.83
         sum:                                29996886.03

hreads fairness:
    events (avg/stddev):           22033.2600/60.91
    execution time (avg/stddev):   299.9689/0.02
```

6. Uji fail-over

a. Melakukan check leader dengan perintah
`curl http://192.168.16.21:2379/pd/api/v1/members`



b. Karena leader ada pada pd1, maka kita mematikan pd1(vagrant halt) dan cek di pd lainya.



7. Monitoring dengan Grafana

a. Install Prometheus, Grafana dan Exporter

i. Tuliskan kode berikut untuk setiap node

```
cd node_exporter-0.15.2.linux-amd64
./node_exporter --web.listen-address=":9100" \
   --log.level="info" &
```

ii. Kembali ke node pertama dan tuliskan kode berikut.

```
wget
https://github.com/prometheus/prometheus/releases/download/v2.2.1/prometheus-
2.2.1.linux-amd64.tar.gz
    wget https://dl.grafana.com/oss/release/grafana-6.5.1.linux-amd64.tar.gz

    tar -xzf prometheus-2.2.1.linux-amd64.tar.gz
    tar -zxf grafana-6.5.1.linux-amd64.tar.gz
```

iii. Edit prometheus.yml menjadi

```
global:
  scrape_interval:     15s  # By default, scrape targets every 15 seconds.
  evaluation_interval: 15s  # By default, scrape targets every 15 seconds.
  # scrape_timeout is set to the global default value (10s).
  external_labels:
    cluster: 'test-cluster'
    monitor: "prometheus"

scrape_configs:
  - job_name: 'overwritten-nodes'
    honor_labels: true  # Do not overwrite job & instance labels.
    static_configs:
    - targets:
      - '192.168.16.42:9100'
      - '192.168.16.43:9100'
      - '192.168.16.44:9100'
      - '192.168.16.45:9100'
      - '192.168.16.46:9100'
      - '192.168.16.47:9100'

  - job_name: 'tidb'
    honor_labels: true  # Do not overwrite job & instance labels.
    static_configs:
    - targets:
      - '192.168.16.42:10080'

  - job_name: 'pd'
    honor_labels: true  # Do not overwrite job & instance labels.
    static_configs:
    - targets:
      - '192.168.16.42:2379'
      - '192.168.16.43:2379'
      - '192.168.16.44:2379'

  - job_name: 'tikv'
    honor_labels: true  # Do not overwrite job & instance labels.
    static_configs:
    - targets:
      - '192.168.16.45:20180'
      - '192.168.16.46:20180'
      - '192.168.16.47:20180'
```

iv. Menjalanakan Prometheus dengan perintah berikut.

```
cd ~
cd prometheus-2.2.1.linux-amd64
./prometheus \
   --config.file="./prometheus.yml" \
   --web.listen-address=":9090" \
   --web.external-url="http://192.168.16.42:9090/" \
   --web.enable-admin-api \
   --log.level="info" \
   --storage.tsdb.path="./data.metrics" \
   --storage.tsdb.retention="15d" &
```

v. Menambahkan grafana.ini

```
cd .. && cd grafana-6.5.1
   nano conf/grafana.ini
   [paths]
   data = ./data
   logs = ./data/log
   plugins = ./data/plugins
   [server]
   http_port = 3000
   domain = 192.168.16.42
   [database]
   [session]
   [analytics]
   check_for_updates = true
   [security]
   admin_user = admin
   admin_password = admin
   [snapshots]
   [users]
   [auth.anonymous]
   [auth.basic]
   [auth.ldap]
   [smtp]
   [emails]
   [log]
   mode = file
   [log.console]
   [log.file]
   level = info
   format = text
   [log.syslog]
   [event_publisher]
   [dashboards.json]
   enabled = false
   path = ./data/dashboards
   [metrics]
   [grafana_net]
   url = https://grafana.net
```

vi. Menjalankan grafana

```
./bin/grafana-server \
    --config="./conf/grafana.ini" &
```

b. Mengkonfigurasi Grafana

i. Masuk ke grafana dengan browser. url 192.168.16.42:3000. Memasukan user dan password “admin”

ii. Pada data source tambahan Prometheus dengan url 192.168.16.42 dan port 9090

iii. Mengimport dashboard grafana dari https://github.com/pingcap/tidb-ansible/tree/master/scripts dengan add json file.

iv. Berikut adalah hasil import dan monitoring dari tidb.json