

Muhammad Adistya Azhar

05111640000103

Basis Data Terdistribusi

1. Implementasi Arsitektur Sistem Basis Data Terdistribusi

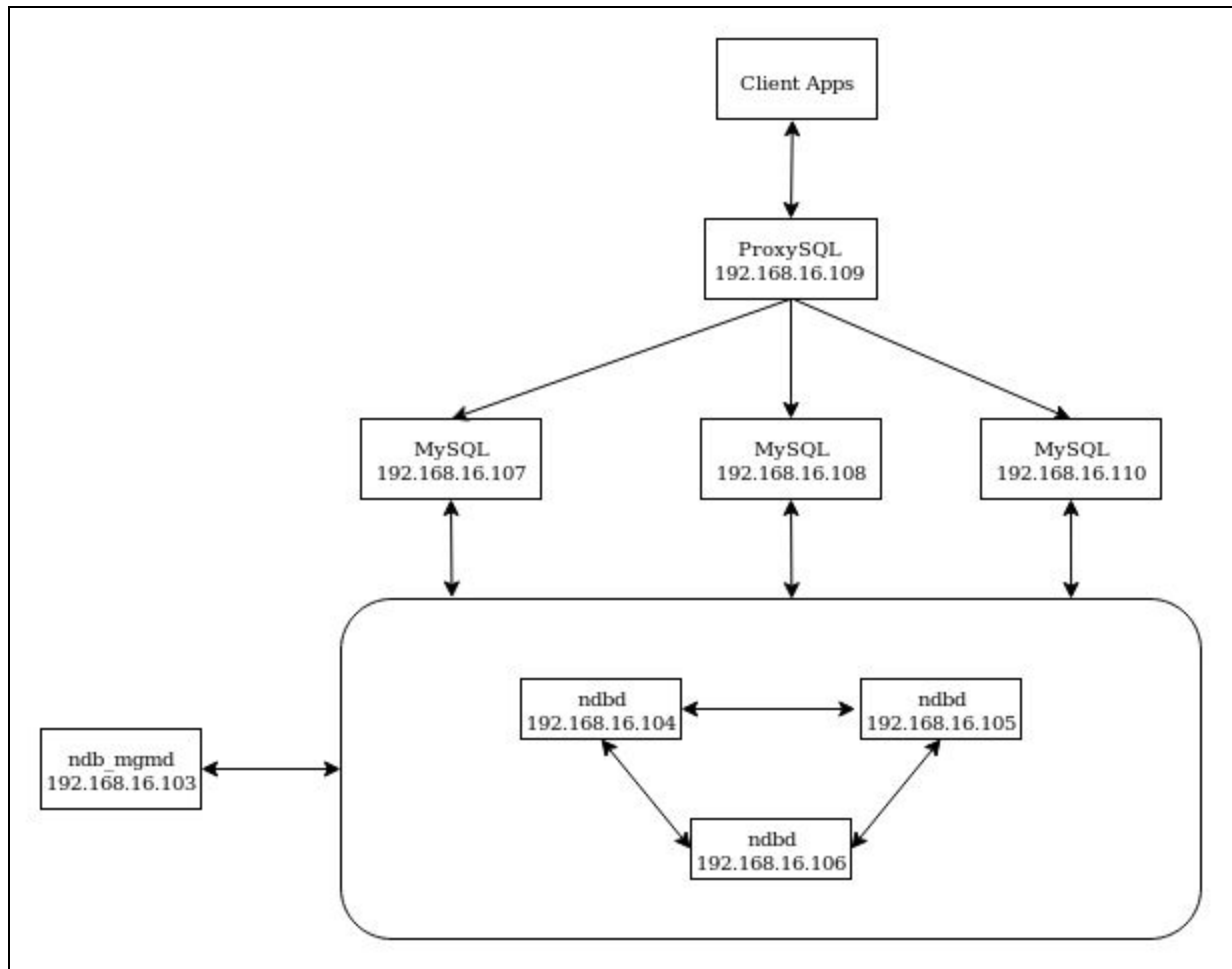
A. Spesifikasi Server

Arsitektur terdiri dari 8 server, masing-masing server memiliki spesifikasi sebagai berikut:

- 512 MB RAM
- Ubuntu 16.04

Pembagian IP server sebagai berikut:

- NDB Management server:
 - 192.168.16.103
- NDB Data server:
 - 192.168.16.104, 192.168.16.105, dan 192.168.16.106
- MySQL Server:
 - 192.168.16.107, 192.168.16.108, dan 192.168.16.110
- ProxySQL:
 - 192.168.16.109



Ilustrasi arsitektur MySQL NDB Cluster

B. Proses Konfigurasi dan Instalasi

1. Instalasi dan konfigurasi cluster manager

- Pada server 192.168.16.103 jalankan command berikut untuk download file .deb cluster manager:

```
wget  
https://dev.mysql.com/get/Downloads/MySQL-Cluster-7.6/mysql-cluster-community-management-server_7.6.12-1ubuntu16.04_amd64.deb
```

- Install .deb yang telah didownload:

```
sudo dpkg -i  
mysql-cluster-community-management-server_7.6.12-1ubuntu16.04_amd64.d  
eb
```

- Konfigurasi cluster manager:

```
sudo mkdir /var/lib/mysql-cluster  
sudo nano /var/lib/mysql-cluster/config.ini
```

- File /var/lib/mysql-cluster/config.ini, pada file ini terdapat pembagian IP NDB Data server, IP NDB Management server, dan MySQL server.

```
[ndbd default]  
NoOfReplicas=3  
  
[ndb_mgmd]  
hostname=192.168.16.103  
datadir=/var/lib/mysql-cluster  
  
[ndbd]  
hostname=192.168.16.104  
NodeId=2  
datadir=/usr/local/mysql/data  
  
[ndbd]  
hostname=192.168.16.105  
NodeId=3  
datadir=/usr/local/mysql/data  
  
[ndbd]  
hostname=192.168.16.106  
NodeId=4  
datadir=/usr/local/mysql/data  
  
[mysqld]  
hostname=192.168.16.107  
  
[mysqld]  
hostname=192.168.16.108  
  
[mysqld]  
hostname=192.168.16.110
```

- Buat systemd service

```
sudo nano /etc/systemd/system/ndb_mgmd.service
```

- File /etc/systemd/system/ndb_mgmd.service

```
[Unit]
Description=MySQL NDB Cluster Management Server
After=network.target auditd.service

[Service]
Type=forking
ExecStart=/usr/sbin/ndb_mgmd -f /var/lib/mysql-cluster/config.ini
ExecReload=/bin/kill -HUP $MAINPID
KillMode=process
Restart=on-failure

[Install]
WantedBy=multi-user.target
```

- Jalankan ndb_mgmd service

```
sudo systemctl daemon-reload
sudo systemctl enable ndb_mgmd
sudo systemctl start ndb_mgmd
sudo systemctl status ndb_mgmd
```

2. Instalasi dan konfigurasi NDB Data server
- Download dan install NDB Data server

```
wget
https://dev.mysql.com/get/Downloads/MySQL-Cluster-7.6/mysql-cluster-community-data-node\_7.6.12-1ubuntu16.04\_amd64.deb

Sudo apt update
sudo apt install libclass-methodmaker-perl
sudo dpkg -i
MySQL-Cluster-7.6/mysql-cluster-community-data-node_7.6.12-1ubuntu16.04_amd64.deb
```

- File konfigurasi NDB Data:

```
sudo nano /etc/my.cnf
sudo mkdir -p /usr/local/mysql/data
```

- File /etc/my.cnf, berisi connection string NDB Management server

```
[mysql_cluster]
ndb-connectstring=192.168.16.103
```

- Systemd ndb service

```
sudo nano /etc/systemd/system/ndbd.service
```

```
[Unit]
Description=MySQL NDB Data Node Daemon
After=network.target auditd.service

[Service]
Type=forking
ExecStart=/usr/sbin/ndbd
ExecReload=/bin/kill -HUP $MAINPID
KillMode=process
Restart=on-failure

[Install]
WantedBy=multi-user.target
```

```
sudo systemctl daemon-reload
sudo systemctl enable ndbd
sudo systemctl start ndbd
sudo systemctl status ndbd
```

3. Instalasi dan konfigurasi MySQL server dan client
- Download dan install MySQL server

```
wget
https://dev.mysql.com/get/Downloads/MySQL-Cluster-7.6/mysql-cluster\_7.6.12-1ubuntu16.04\_amd64.deb-bundle.tar

mkdir install
tar -xvf mysql-cluster_7.6.12-1ubuntu16.04_amd64.deb-bundle.tar -C install/

cd install

sudo apt update
sudo apt install libaio1 libmecab2

sudo dpkg -i mysql-common_7.6.12-1ubuntu16.04_amd64.deb
```

```

sudo dpkg -i mysql-cluster-community-client_7.6.12-1ubuntu16.04_amd64.deb
sudo dpkg -i mysql-client_7.6.12-1ubuntu16.04_amd64.deb
sudo dpkg -i
mysql-cluster-community-server_7.6.12-1ubuntu16.04_amd64.deb

sudo dpkg -i mysql-server_7.6.12-1ubuntu16.04_amd64.deb

```

- Konfigurasi file /etc/mysql/my.cnf

```

!includedir /etc/mysql/conf.d/
!includedir /etc/mysql/mysql.conf.d/

[mysqld]
# Options for mysqld process:
ndbcluster          # run NDB storage engine
default-storage-engine=NDBCLUSTER

[mysql_cluster]
# Options for NDB Cluster processes:
ndb-connectstring=192.168.16.103 # location of management server

```

4. Melihat status cluster

- Terdapat 3 NDB Data server yang terkoneksi

```

mysql> SHOW ENGINE NDB STATUS \G
***** 1. row *****
Type: ndbcluster
Name: connection
Status: cluster_node_id=6, connected_host=192.168.16.103, connected_port=1186,
number of data nodes=3, number of ready data nodes=3, connect count=0
***** 2. row *****

```

- Melalui client ndb_mgm, terdapat 3 NDB Data server yang terkoneksi, 1 NDB Management server, dan 2 MySQL server.

```

vagrant@node107:~$ ndb_mgm
-- NDB Cluster -- Management Client --
ndb_mgm> SHOW
Connected to Management Server at: 192.168.16.103:1186
Cluster Configuration
-----
[ndbd(NDB)] 3 node(s)
id=2 @192.168.16.104 (mysql-5.7.22 ndb-7.6.6, Nodegroup: 0, *)
id=3 @192.168.16.105 (mysql-5.7.22 ndb-7.6.6, Nodegroup: 0)
id=4 @192.168.16.106 (mysql-5.7.22 ndb-7.6.6, Nodegroup: 0)

[ndb_mgmd(MGM)] 1 node(s)
id=1 @192.168.16.103 (mysql-5.7.28 ndb-7.6.12)

[mysqld(API)] 2 node(s)
id=5 @192.168.16.107 (mysql-5.7.28 ndb-7.6.12)
id=6 @192.168.16.108 (mysql-5.7.28 ndb-7.6.12)

```

5. Instalasi dan konfigurasi ProxySQL

- Download dan install file ProxySQL

```

curl -OL
https://github.com/sysown/proxysql/releases/download/v1.4.4/proxysql_1.4.4-u
buntu16_amd64.deb

sudo dpkg -i proxysql_*

sudo apt-get update
sudo apt-get install mysql-client

```

- Jalankan ProxySQL

```

sudo systemctl start proxysql

```

- Pada MySQL server, jalankan command berikut untuk setting MySQL user yang digunakan oleh ProxySQL untuk monitoring server

```

curl -OL
https://gist.github.com/lefred/77ddbde301c72535381ae7af9f968322/raw/5e40b
03333a3c148b78aa348fd2cd5b5dbb36e4d/addition_to_sys.sql

mysql -u root -p < addition_to_sys.sql
mysql -u root -p

```

```
mysql> CREATE USER 'monitor'@'%' IDENTIFIED BY 'monitorpassword';
mysql> GRANT SELECT on sys.* to 'monitor'@'%';
mysql> FLUSH PRIVILEGES;
```

- Konfigurasi monitoring ProxySQL

```
ProxySQLAdmin> UPDATE global_variables SET variable_value='monitor'
WHERE variable_name='mysql-monitor_username';
ProxySQLAdmin> UPDATE global_variables SET
variable_value='monitorpassword' WHERE
variable_name='mysql-monitor_password';

ProxySQLAdmin> LOAD MYSQL VARIABLES TO RUNTIME;
ProxySQLAdmin> SAVE MYSQL VARIABLES TO DISK;
```

- Menambahkan MySQL server ke ProxySQL server pool

```
ProxySQLAdmin> INSERT INTO mysql_group_replication_hostgroups
(writer_hostgroup, backup_writer_hostgroup, reader_hostgroup,
offline_hostgroup, active, max_writers, writer_is_also_reader,
max_transactions_behind) VALUES (2, 4, 3, 1, 1, 3, 1, 100);

ProxySQLAdmin> INSERT INTO mysql_servers(hostgroup_id, hostname, port)
VALUES (2, '192.168.16.107', 3306);
ProxySQLAdmin> INSERT INTO mysql_servers(hostgroup_id, hostname, port)
VALUES (2, '192.168.16.108', 3306);
ProxySQLAdmin> INSERT INTO mysql_servers(hostgroup_id, hostname, port)
VALUES (2, '192.168.16.110', 3306);

LOAD MYSQL SERVERS TO RUNTIME;
SAVE MYSQL SERVERS TO DISK;
```

- Membuat MySQL user di MySQL server

```
mysql> CREATE USER 'adis'@'%' IDENTIFIED BY 'password';
mysql> GRANT ALL PRIVILEGES on playground.* to 'adis'@'%';
mysql> FLUSH PRIVILEGES;
```

- Menambahkan MySQL user yang baru dibuat ke ProxySQL

```
ProxySQLAdmin> INSERT INTO mysql_users(username, password,
default_hostgroup) VALUES ('playgrounduser', 'playgroundpassword', 2);
```



```
ProxySQLAdmin> LOAD MYSQL USERS TO RUNTIME;  
ProxySQLAdmin> SAVE MYSQL USERS TO DISK;
```

2. Pemanfaatan basis data terdistribusi dalam aplikasi

a. Import struktur database aplikasi CRUD

- Source code mengarahkan ke IP ProxySQL menggunakan database credentials yang sudah dibuat

```
services.AddDbContext<BlogsDbContext>(options =>  
options.UseMySQL("Server=192.168.16.109;Port=6033;Database=adis_blog;U  
ser=adis;Password=password;" ));
```

dotnet ef database update

```
mysql> use adis_blog;  
Database changed  
mysql> show tables;  
+-----+  
| Tables_in_adis_blog |  
+-----+  
| _EFMigrationsHistory |  
| comments              |  
| favorite_posts        |  
| followings            |  
| post_tag              |  
| posts                 |  
| tags                  |  
| users                 |  
+-----+  
8 rows in set (0.01 sec)
```

Hasil import struktur database

b. Cara menggunakan aplikasi

- Register

Username

Password

[Register Account](#)

- Login

Login

[Login](#)

- Create post

Title

erat molestie hendrerit.

Body

File Edit View Insert Format Tools

↶ ↷ **B** *I*

≡ ≡ ≡

<>

Nulla rinibus magna quis mauris egestas vestibulum a non elit. Nullam id ligula sagittis, egestas neque at, mollis tellus. Nulla posuere ante dolor, vitae faucibus ante laoreet nec. Proin non leo elementum, venenatis libero eu, tempor nulla. In feugiat, nulla vel ultrices consequat, felis elit vestibulum ipsum, id vehicula ligula leo id elit. Aliquam vel velit hendrerit, gravida risus ut, aliquam nisl. Pellentesque non nulla ornare, sodales quam non, sagittis mauris.

Cras suscipit est eget dapibus venenatis. Integer nibh ligula, porta in cursus ut, tempus quis ante. Class aptent taciti sociosqu ad litora torquent per conubia nostra, per inceptos himenaeos. Maecenas sit amet orci tempus, varius massa vel, consequat mauris. Duis ut mauris malesuada, pharetra metus sed, rutrum sapien. Pellentesque lacus arcu, pretium varius maximus non, ornare quis erat. Curabitur volutpat diam vel maximus interdum. Proin erat dui, tempor vitae hendrerit sit amet rutrum eget egestas. Vivamus sit amet euismod lectus. Nam vel id eros. Donec

Lifestyle x

x | v

Browse...

No file selected.

Submit

- Comment post

Comments

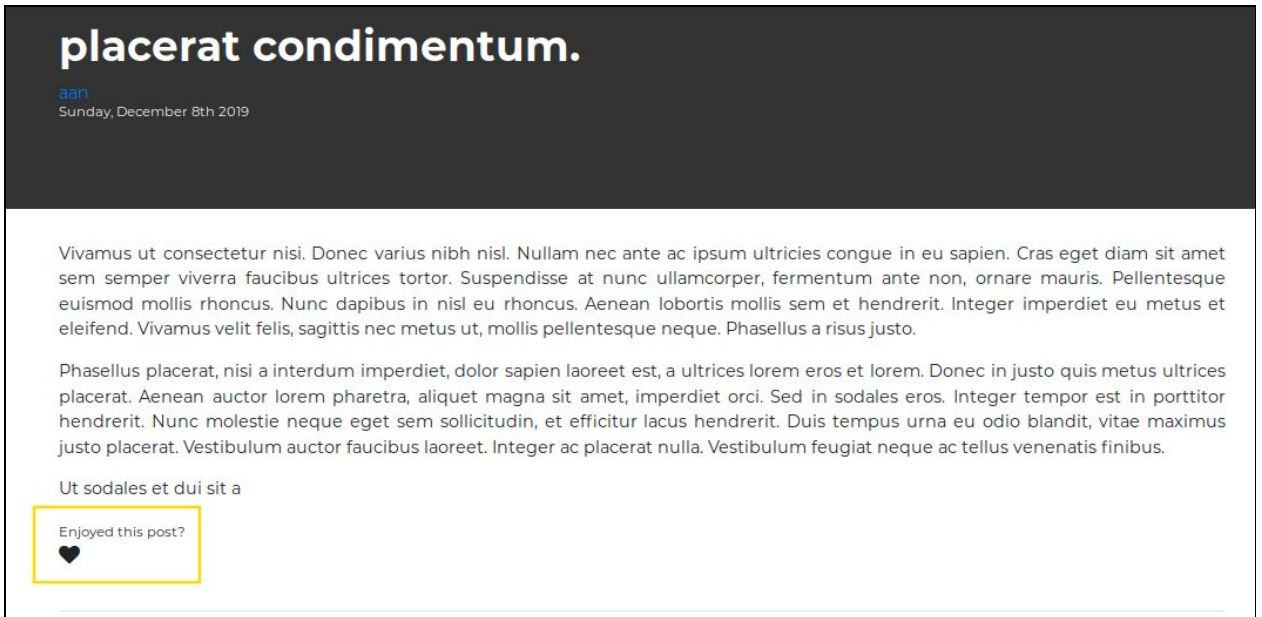
ultrices felis fermentum eget. Maecenas sed justo velit. Donec et eros ex. Vestibulum quis venenatis tellus, at aliquam tellus. Morbi sodales leo erat.

Post Comment

- Follow/ unfollow user



- Like post



- All posts

placerat condimentum.

Vivamus ut consectetur nisi. Donec varius nibh nisl. Nullam nec ante ac ipsum ultricies congue in eu...

[Read more ...](#)

Aenean ut magna at justo varius laoreet.

. Donec laoreet facilisis ipsum, et fermentum leo ultrices et. Nulla sem nisl, dignissim sed neque ...

[Read more ...](#)

In nunc nisi, viverra nec nunc at, accumsan posuere mauris. Nulla id nisl ipsum.

Ut a venenatis metus. Suspendisse sodales et ante vitae ullamcorper. Suspendisse potenti. Ut nec nun...

[Read more ...](#)

erat molestie hendrerit.

Nulla finibus magna quis mauris egestas vestibulum a non elit. Nullam id ligula sagittis, egestas ne...

[Read more ...](#)

Applications that use MySQL can employ standard APIs to

It is worth taking into account that Cluster nodes do not make use of the MySQL privilege system whe...

[Read more ...](#)

- [View comments](#)

ultrices felis fermentum eget. Maecenas sed justo velit. Donec et eros ex. Vestibulum quis venenatis tellus, at aliquam tellus. Morbi sodales leo erat.

Sunday, December 8th 2019

Sed et tortor scelerisque, vehicula nibh nec, tempor mi. Vestibulum ante ipsum primis in faucibus orci luctus et ultrices posuere cubilia Curae;

Sunday, December 8th 2019

. Etiam convallis suscipit dui, in feugiat justo ultricies ut. Mauris massa arcu, interdum tempor augue nec, porttitor pretium sem. Maecenas semper nisi in enim mollis, sed vulputate ante tempus.

Sunday, December 8th 2019

- [View post](#)

Aenean ut magna at justo varius laoreet.

aan

Sunday, December 8th 2019

Donec laoreet facilisis ipsum, et fermentum leo ultrices et. Nulla sem nisl, dignissim sed neque eu, tincidunt pellentesque augue. Nam semper, ligula vel posuere viverra, libero arcu luctus augue, a facilisis urna risus ac nibh. Proin at volutpat purus. Sed quis libero sed eros porta pellentesque. Suspendisse ut luctus eros, id interdum eros. Nam in enim sagittis, hendrerit ex nec, tempor odio. Sed non nisl viverra, ultrices ligula id, laoreet lorem. Morbi sit amet odio ac lectus scelerisque finibus ut sed neque. Mauris bibendum orci sed nunc condimentum egestas. Curabitur venenatis justo ac tristique bibendum. Sed nec orci sem.

3. Uji performa aplikasi dan basis data

a. Aplikasi JMeter

i. 100 users selama 10 detik

Ketika hit endpoint `/api/posts` menghasilkan rata-rata sample time 30 ms.

Seluruh request mendapatkan status 200 OK.

Sample #	Start Time	Thread Name	Label	Sample Time(...)	Status	Bytes	Sent Bytes	Latency	Connect Time...
1	17:20:27.576	Thread Grou...	HTTP Request	12	✓	1718	135	11	0
2	17:20:27.677	Thread Grou...	HTTP Request	33	✓	1718	135	33	1
3	17:20:27.773	Thread Grou...	HTTP Request	27	✓	1718	135	26	1
4	17:20:27.872	Thread Grou...	HTTP Request	27	✓	1718	135	27	1
5	17:20:27.975	Thread Grou...	HTTP Request	31	✓	1718	135	31	1
6	17:20:28.076	Thread Grou...	HTTP Request	36	✓	1718	135	35	2
7	17:20:28.175	Thread Grou...	HTTP Request	27	✓	1718	135	27	2
8	17:20:28.275	Thread Grou...	HTTP Request	36	✓	1718	135	36	4
9	17:20:28.376	Thread Grou...	HTTP Request	28	✓	1718	135	28	1
10	17:20:28.476	Thread Grou...	HTTP Request	27	✓	1718	135	27	1
11	17:20:28.576	Thread Grou...	HTTP Request	27	✓	1718	135	27	2
12	17:20:28.682	Thread Grou...	HTTP Request	31	✓	1718	135	30	2
13	17:20:28.771	Thread Grou...	HTTP Request	41	✓	1718	135	39	1
14	17:20:28.873	Thread Grou...	HTTP Request	12	✓	1718	135	12	0
15	17:20:28.971	Thread Grou...	HTTP Request	27	✓	1718	135	27	0
16	17:20:29.075	Thread Grou...	HTTP Request	38	✓	1718	135	38	2
17	17:20:29.172	Thread Grou...	HTTP Request	43	✓	1718	135	43	2
18	17:20:29.274	Thread Grou...	HTTP Request	16	✓	1718	135	16	0
19	17:20:29.372	Thread Grou...	HTTP Request	25	✓	1718	135	25	1
20	17:20:29.471	Thread Grou...	HTTP Request	37	✓	1718	135	37	2
21	17:20:29.573	Thread Grou...	HTTP Request	35	✓	1718	135	35	2
22	17:20:29.672	Thread Grou...	HTTP Request	11	✓	1718	135	10	0
23	17:20:29.772	Thread Grou...	HTTP Request	28	✓	1718	135	28	1
24	17:20:29.872	Thread Grou...	HTTP Request	25	✓	1718	135	25	1
25	17:20:29.972	Thread Grou...	HTTP Request	31	✓	1718	135	30	1
26	17:20:30.072	Thread Grou...	HTTP Request	44	✓	1718	135	44	2
27	17:20:30.174	Thread Grou...	HTTP Request	29	✓	1718	135	29	2
28	17:20:30.271	Thread Grou...	HTTP Request	25	✓	1718	135	25	1
29	17:20:30.372	Thread Grou...	HTTP Request	28	✓	1718	135	28	1
30	17:20:30.472	Thread Grou...	HTTP Request	34	✓	1718	135	34	2

☐ Scroll automatically?
 ☐ Child samples?
 No of Samples 100
 Latest Sample 29
 Average 30
 Deviation 21

- ii. 500 users selama 10 detik
 Ketika hit endpoint /api/posts menghasilkan rata-rata sample time 15 ms.
 Seluruh request mendapatkan status 200 OK.

Sample #	Start Time	Thread Name	Label	Sample Time(ms)	Status	Bytes	Sent Bytes	Latency	Connect ...
1	17:31:01.536	Thread Grou...	HTTP Request	23	✓	1718	135	23	0
2	17:31:01.556	Thread Grou...	HTTP Request	13	✓	1718	135	13	0
3	17:31:01.577	Thread Grou...	HTTP Request	16	✓	1718	135	16	1
4	17:31:01.597	Thread Grou...	HTTP Request	12	✓	1718	135	12	1
5	17:31:01.627	Thread Grou...	HTTP Request	8	✓	1718	135	8	0
6	17:31:01.638	Thread Grou...	HTTP Request	14	✓	1718	135	14	1
7	17:31:01.665	Thread Grou...	HTTP Request	8	✓	1718	135	8	1
8	17:31:01.678	Thread Grou...	HTTP Request	8	✓	1718	135	8	1
9	17:31:01.707	Thread Grou...	HTTP Request	8	✓	1718	135	8	0
10	17:31:01.716	Thread Grou...	HTTP Request	7	✓	1718	135	7	1
11	17:31:01.738	Thread Grou...	HTTP Request	6	✓	1718	135	6	0
12	17:31:01.756	Thread Grou...	HTTP Request	6	✓	1718	135	6	0
13	17:31:01.776	Thread Grou...	HTTP Request	9	✓	1718	135	9	0
14	17:31:01.796	Thread Grou...	HTTP Request	8	✓	1718	135	8	0
15	17:31:01.816	Thread Grou...	HTTP Request	9	✓	1718	135	9	0
16	17:31:01.836	Thread Grou...	HTTP Request	10	✓	1718	135	10	1
17	17:31:01.856	Thread Grou...	HTTP Request	33	✓	1718	135	33	1
18	17:31:01.877	Thread Grou...	HTTP Request	30	✓	1718	135	29	0
19	17:31:01.897	Thread Grou...	HTTP Request	23	✓	1718	135	22	1
20	17:31:01.917	Thread Grou...	HTTP Request	28	✓	1718	135	28	1
21	17:31:01.936	Thread Grou...	HTTP Request	17	✓	1718	135	17	1
22	17:31:01.956	Thread Grou...	HTTP Request	10	✓	1718	135	10	0
23	17:31:01.976	Thread Grou...	HTTP Request	10	✓	1718	135	10	0
24	17:31:01.996	Thread Grou...	HTTP Request	14	✓	1718	135	14	0
25	17:31:02.016	Thread Grou...	HTTP Request	7	✓	1718	135	7	0
26	17:31:02.036	Thread Grou...	HTTP Request	10	✓	1718	135	10	1
27	17:31:02.056	Thread Grou...	HTTP Request	8	✓	1718	135	8	0
28	17:31:02.076	Thread Grou...	HTTP Request	7	✓	1718	135	7	0
29	17:31:02.096	Thread Grou...	HTTP Request	7	✓	1718	135	7	1
30	17:31:02.116	Thread Grou...	HTTP Request	7	✓	1718	135	6	1

☐ Scroll automatically?
 ☐ Child samples?
 No of Samples 500
 Latest Sample 8
 Average 15
 Deviation 12

- iii. 1000 users selama 10 detik
 Ketika hit endpoint /api/posts menghasilkan rata-rata sample time 18 ms.
 Seluruh request mendapatkan status 200 OK.

Sample #	Start Time	Thread Name	Label	Sample Time(ms)	Status	Bytes	Sent Bytes	Latency	Connect ...
1	17:34:07.463	Thread Grou...	HTTP Request	9		1718	135	9	1
2	17:34:07.473	Thread Grou...	HTTP Request	13		1718	135	13	0
3	17:34:07.483	Thread Grou...	HTTP Request	9		1718	135	9	0
4	17:34:07.494	Thread Grou...	HTTP Request	8		1718	135	8	0
5	17:34:07.503	Thread Grou...	HTTP Request	7		1718	135	7	0
6	17:34:07.513	Thread Grou...	HTTP Request	6		1718	135	6	0
7	17:34:07.524	Thread Grou...	HTTP Request	8		1718	135	7	0
8	17:34:07.534	Thread Grou...	HTTP Request	9		1718	135	9	0
9	17:34:07.543	Thread Grou...	HTTP Request	8		1718	135	8	0
10	17:34:07.553	Thread Grou...	HTTP Request	8		1718	135	8	0
11	17:34:07.563	Thread Grou...	HTTP Request	8		1718	135	8	1
12	17:34:07.580	Thread Grou...	HTTP Request	12		1718	135	12	0
13	17:34:07.584	Thread Grou...	HTTP Request	10		1718	135	10	0
14	17:34:07.593	Thread Grou...	HTTP Request	7		1718	135	7	0
15	17:34:07.603	Thread Grou...	HTTP Request	6		1718	135	6	0
16	17:34:07.612	Thread Grou...	HTTP Request	8		1718	135	8	0
17	17:34:07.622	Thread Grou...	HTTP Request	6		1718	135	6	0
18	17:34:07.632	Thread Grou...	HTTP Request	7		1718	135	7	0
19	17:34:07.642	Thread Grou...	HTTP Request	10		1718	135	10	0
20	17:34:07.653	Thread Grou...	HTTP Request	7		1718	135	7	0
21	17:34:07.662	Thread Grou...	HTTP Request	7		1718	135	7	0
22	17:34:07.673	Thread Grou...	HTTP Request	31		1718	135	31	0
23	17:34:07.686	Thread Grou...	HTTP Request	54		1718	135	49	1
24	17:34:07.704	Thread Grou...	HTTP Request	45		1718	135	45	0
25	17:34:07.714	Thread Grou...	HTTP Request	37		1718	135	37	0
26	17:34:07.694	Thread Grou...	HTTP Request	61		1718	135	61	0
27	17:34:07.724	Thread Grou...	HTTP Request	34		1718	135	34	0
28	17:34:07.734	Thread Grou...	HTTP Request	36		1718	135	35	0
29	17:34:07.743	Thread Grou...	HTTP Request	27		1718	135	27	0
30	17:34:07.753	Thread Grou...	HTTP Request	20		1718	135	20	0

☐ Scroll automatically?
 ☐ Child samples?
 No of Samples 1000
 Latest Sample 6
 Average 18
 Deviation 20

b. Basis Data Sysbench

i. Prepare data

```
sysbench ./oltp_read_write.lua --threads=5
--mysql-host=192.168.16.109 --mysql-user=adis
--mysql-password=password --mysql-port=6033 --tables=20
--table-size=10000000 --db-driver=mysql
--mysql_storage_engine=ndbcluster prepare
```

ii. Uji coba server 192.168.16.107

```
sysbench ./oltp_read_write.lua --threads=16 --events=0 --time=15
--mysql-host=192.168.16.107 --mysql-user=adis
--mysql-password=password --mysql-port=3306 --tables=20
--delete_inserts=10 --index_updates=10 --non_index_updates=10
--table-size=10000000 --db-ps-mode=disable --report-interval=1
--db-driver=mysql --mysql_storage_engine=ndbcluster run
```

```
Initializing worker threads...
Threads started!

[ 1s ] thds: 16 tps: 33.94 qps: 2549.34 (r/w/o: 672.77/449.18/1427.39) lat (ms,95%): 411.96 err/s: 0.00 reconn/s: 0.00
[ 2s ] thds: 16 tps: 47.01 qps: 2557.59 (r/w/o: 660.15/464.11/1433.33) lat (ms,95%): 369.77 err/s: 0.00 reconn/s: 0.00
[ 3s ] thds: 16 tps: 44.00 qps: 2412.00 (r/w/o: 635.00/433.00/1344.00) lat (ms,95%): 383.33 err/s: 0.00 reconn/s: 0.00
[ 4s ] thds: 16 tps: 47.00 qps: 2420.99 (r/w/o: 635.00/488.00/1297.99) lat (ms,95%): 419.45 err/s: 0.00 reconn/s: 0.00
[ 5s ] thds: 16 tps: 45.00 qps: 2500.02 (r/w/o: 594.00/489.00/1417.01) lat (ms,95%): 390.30 err/s: 0.00 reconn/s: 0.00
[ 6s ] thds: 16 tps: 36.00 qps: 2152.03 (r/w/o: 548.01/380.01/1224.02) lat (ms,95%): 458.96 err/s: 0.00 reconn/s: 0.00
[ 7s ] thds: 16 tps: 46.00 qps: 2495.05 (r/w/o: 632.01/484.01/1379.02) lat (ms,95%): 434.83 err/s: 0.00 reconn/s: 0.00
[ 8s ] thds: 16 tps: 44.00 qps: 2517.87 (r/w/o: 611.97/488.97/1416.92) lat (ms,95%): 369.77 err/s: 0.00 reconn/s: 0.00
[ 9s ] thds: 16 tps: 43.00 qps: 2403.08 (r/w/o: 570.02/445.01/1388.04) lat (ms,95%): 411.96 err/s: 0.00 reconn/s: 0.00
[10s ] thds: 16 tps: 42.00 qps: 2527.15 (r/w/o: 642.04/472.03/1413.08) lat (ms,95%): 376.49 err/s: 0.00 reconn/s: 0.00
[11s ] thds: 16 tps: 45.00 qps: 2327.79 (r/w/o: 604.94/430.96/1291.88) lat (ms,95%): 397.39 err/s: 0.00 reconn/s: 0.00
[12s ] thds: 16 tps: 34.00 qps: 1860.04 (r/w/o: 461.01/343.01/1056.02) lat (ms,95%): 569.67 err/s: 0.00 reconn/s: 0.00
[13s ] thds: 16 tps: 33.00 qps: 2023.89 (r/w/o: 501.97/382.98/1138.94) lat (ms,95%): 511.33 err/s: 0.00 reconn/s: 0.00
[14s ] thds: 16 tps: 47.00 qps: 2504.13 (r/w/o: 632.03/479.02/1393.07) lat (ms,95%): 404.61 err/s: 0.00 reconn/s: 0.00
[15s ] thds: 16 tps: 44.00 qps: 2447.98 (r/w/o: 615.99/472.00/1359.99) lat (ms,95%): 376.49 err/s: 0.00 reconn/s: 0.00

SQL statistics:
  queries performed:
    read:          9058
    write:         6852
    other:         20322
    total:        36232
  transactions:    647 (42.42 per sec.)
  queries:         36232 (2375.67 per sec.)
  ignored errors:  0 (0.00 per sec.)
  reconnects:      0 (0.00 per sec.)

General statistics:
  total time:      15.2499s
  total number of events: 647

Latency (ms):
  min:            273.27
  avg:            375.42
  max:            595.83
  95th percentile: 467.30
  sum:            242899.13

Threads fairness:
  events (avg/stddev): 40.4375/0.50
  execution time (avg/stddev): 15.1812/0.06
```

iii. Uji coba server 192.168.16.108

```
sysbench ./oltp_read_write.lua --threads=16 --events=0 --time=15
--mysql-host=192.168.16.108 --mysql-user=adis
--mysql-password=password --mysql-port=3306 --tables=20
--delete_inserts=10 --index_updates=10 --non_index_updates=10
--table-size=10000000 --db-ps-mode=disable --report-interval=1
--db-driver=mysql --mysql_storage_engine=ndbcluster run
```

```

=6033 --tables=20 --delete_inserts=10 --index_updates=10 --non_index_updates=10 --table-s
Initializing worker threads...
Threads started!

[ 1s ] thds: 16 tps: 31.94 qps: 2441.07 (r/w/o: 670.64/402.19/1368.24) lat (ms,95%): 397.39 err/s: 0.00 reconn/s: 0.00
[ 2s ] thds: 16 tps: 43.02 qps: 2045.96 (r/w/o: 517.24/416.19/1112.52) lat (ms,95%): 539.71 err/s: 0.00 reconn/s: 0.00
[ 3s ] thds: 16 tps: 43.00 qps: 2551.96 (r/w/o: 612.99/479.99/1458.98) lat (ms,95%): 539.71 err/s: 0.00 reconn/s: 0.00
[ 4s ] thds: 16 tps: 42.00 qps: 2486.08 (r/w/o: 648.02/441.01/1397.05) lat (ms,95%): 390.30 err/s: 0.00 reconn/s: 0.00
[ 5s ] thds: 16 tps: 46.99 qps: 2510.62 (r/w/o: 645.90/466.93/1397.79) lat (ms,95%): 376.49 err/s: 0.00 reconn/s: 0.00
[ 6s ] thds: 16 tps: 45.01 qps: 2465.31 (r/w/o: 637.08/493.06/1335.17) lat (ms,95%): 383.33 err/s: 0.00 reconn/s: 0.00
[ 7s ] thds: 16 tps: 46.00 qps: 2523.00 (r/w/o: 632.00/478.00/1413.00) lat (ms,95%): 369.77 err/s: 0.00 reconn/s: 0.00
[ 8s ] thds: 16 tps: 45.95 qps: 2532.27 (r/w/o: 595.36/493.47/1443.45) lat (ms,95%): 376.49 err/s: 0.00 reconn/s: 0.00
[ 9s ] thds: 16 tps: 42.04 qps: 2518.61 (r/w/o: 616.64/468.49/1433.49) lat (ms,95%): 376.49 err/s: 0.00 reconn/s: 0.00
[ 10s ] thds: 16 tps: 44.00 qps: 2618.06 (r/w/o: 653.02/504.01/1461.03) lat (ms,95%): 363.18 err/s: 0.00 reconn/s: 0.00
[ 11s ] thds: 16 tps: 47.00 qps: 2592.10 (r/w/o: 651.02/471.02/1470.05) lat (ms,95%): 363.18 err/s: 0.00 reconn/s: 0.00
[ 12s ] thds: 16 tps: 46.00 qps: 2581.94 (r/w/o: 655.99/481.99/1443.97) lat (ms,95%): 369.77 err/s: 0.00 reconn/s: 0.00
[ 13s ] thds: 16 tps: 47.00 qps: 2613.01 (r/w/o: 667.00/495.00/1451.00) lat (ms,95%): 363.18 err/s: 0.00 reconn/s: 0.00
[ 14s ] thds: 16 tps: 48.00 qps: 2558.90 (r/w/o: 659.97/489.98/1408.94) lat (ms,95%): 363.18 err/s: 0.00 reconn/s: 0.00
[ 15s ] thds: 16 tps: 31.00 qps: 1621.87 (r/w/o: 416.97/327.97/876.93) lat (ms,95%): 493.24 err/s: 0.00 reconn/s: 0.00

SQL statistics:
  queries performed:
    read:          9310
    write:         7071
    other:         20859
    total:        37240
  transactions:    665 (43.49 per sec.)
  queries:         37240 (2435.29 per sec.)
  ignored errors:  0 (0.00 per sec.)
  reconnects:     0 (0.00 per sec.)

General statistics:
  total time:      OUTPUT: DERRUGCONSOLE 15.2902s
  total number of events: 665
  sql: command not found: solargraph

Latency (ms):
  min:            320.24
  avg:            366.14
  max:            576.34
  95th percentile: 493.24
  sum:            243485.39

Threads fairness:
  events (avg/stddev): 41.5625/0.50
  execution time (avg/stddev): 15.2178/0.06

```

iv. Uji coba server 192.168.16.110

```

sysbench ./oltp_read_write.lua --threads=16 --events=0 --time=15
--mysql-host=192.168.16.110 --mysql-user=adis
--mysql-password=password --mysql-port=3306 --tables=20
--delete_inserts=10 --index_updates=10 --non_index_updates=10
--table-size=10000000 --db-ps-mode=disable --report-interval=1
--db-driver=mysql --mysql_storage_engine=ndbcluster run

```



```
Initializing worker threads...
Threads started!

[ 1s ] thds: 16 tps: 15.97 qps: 1457.97 (r/w/o: 447.07/169.65/841.25) lat (ms,95%): 787.74 err/s: 0.00 reconn/s: 0.00
[ 2s ] thds: 16 tps: 48.03 qps: 2523.32 (r/w/o: 670.35/474.25/1378.72) lat (ms,95%): 376.49 err/s: 0.00 reconn/s: 0.00
[ 3s ] thds: 16 tps: 47.00 qps: 2535.95 (r/w/o: 633.99/485.99/1415.97) lat (ms,95%): 376.49 err/s: 0.00 reconn/s: 0.00
[ 4s ] thds: 16 tps: 33.00 qps: 1910.08 (r/w/o: 472.02/324.01/1114.05) lat (ms,95%): 569.67 err/s: 0.00 reconn/s: 0.00
[ 5s ] thds: 16 tps: 34.00 qps: 2095.05 (r/w/o: 466.01/387.01/1242.03) lat (ms,95%): 442.73 err/s: 0.00 reconn/s: 0.00
[ 6s ] thds: 16 tps: 44.00 qps: 2271.94 (r/w/o: 610.98/410.99/1249.97) lat (ms,95%): 475.79 err/s: 0.00 reconn/s: 0.00
[ 7s ] thds: 16 tps: 44.00 qps: 2503.90 (r/w/o: 608.98/464.98/1429.95) lat (ms,95%): 397.39 err/s: 0.00 reconn/s: 0.00
[ 8s ] thds: 16 tps: 44.00 qps: 2459.12 (r/w/o: 612.03/443.02/1404.07) lat (ms,95%): 404.61 err/s: 0.00 reconn/s: 0.00
[ 9s ] thds: 16 tps: 48.00 qps: 2635.95 (r/w/o: 667.99/468.99/1498.97) lat (ms,95%): 356.70 err/s: 0.00 reconn/s: 0.00
[10s ] thds: 16 tps: 47.00 qps: 2603.92 (r/w/o: 644.98/474.99/1483.96) lat (ms,95%): 356.70 err/s: 0.00 reconn/s: 0.00
[11s ] thds: 16 tps: 44.00 qps: 2492.10 (r/w/o: 613.02/453.02/1426.06) lat (ms,95%): 376.49 err/s: 0.00 reconn/s: 0.00
[12s ] thds: 16 tps: 42.00 qps: 2468.96 (r/w/o: 613.99/442.99/1411.98) lat (ms,95%): 383.33 err/s: 0.00 reconn/s: 0.00
[13s ] thds: 16 tps: 45.00 qps: 2576.07 (r/w/o: 640.02/466.01/1470.04) lat (ms,95%): 376.49 err/s: 0.00 reconn/s: 0.00
[14s ] thds: 16 tps: 46.97 qps: 2596.36 (r/w/o: 651.59/470.70/1474.07) lat (ms,95%): 363.18 err/s: 0.00 reconn/s: 0.00
[15s ] thds: 16 tps: 45.03 qps: 2592.50 (r/w/o: 658.38/467.27/1466.85) lat (ms,95%): 369.77 err/s: 0.00 reconn/s: 0.00

SQL statistics:
  queries performed:
    read:          9016
    write:         6518
    other:         20530
    total:        36064
  transactions:    644 (42.49 per sec.)
  queries:         36064 (2379.35 per sec.)
  ignored errors:  0 (0.00 per sec.)
  reconnects:      0 (0.00 per sec.)

General statistics:
  total time:      15.1557s
  total number of events: 644

Latency (ms):
  min:            231.55
  avg:            374.78
  max:            788.86
  95th percentile: 520.62
  sum:            241360.95

Threads fairness:
  events (avg/stddev): 40.2500/0.75
  execution time (avg/stddev): 15.0851/0.05
```

c. Uji fail-over

- i. Menghentikan salah satu NDB server ditengah-tengah load test endpoint /api/posts menggunakan JMeter
Meskipun salah satu NDB server dimatikan, pengambilan data melalui endpoint /api/posts tetap dapat dilakukan dengan jumlah 1000 user tanpa error.

```

Cluster Configuration
-----
[ndbd(NDB)] 3 node(s)
id=2 @192.168.16.104 (mysql-5.7.22 ndb-7.6.6, No
degroup: 0, *)
id=3 @192.168.16.105 (mysql-5.7.22 ndb-7.6.6, No
degroup: 0)
id=4 @192.168.16.106 (mysql-5.7.22 ndb-7.6.6, No
degroup: 0)

[ndb_mgmd(MGM)] 1 node(s)
id=1 @192.168.16.103 (mysql-5.7.28 ndb-7.6.12)

[mysqld(API)] 3 node(s)
id=5 @192.168.16.107 (mysql-5.7.28 ndb-7.6.12)
id=6 @192.168.16.108 (mysql-5.7.28 ndb-7.6.12)
id=7 @192.168.16.110 (mysql-5.7.28 ndb-7.6.12)

ndb_mgm> 3 stop
Node 3: Node shutdown initiated
Node 3 has shutdown.

ndb_mgm> Node 3: Node shutdown completed.

```

Sample #	Start Time	Thread Name	Label	Sample Time(ms)	Status	Bytes	Sent Bytes	Latency	Connect time...
1	23:46:15.556	Thread Grou...	HTTP Request	18	✓	1718	128	18	0
2	23:46:15.605	Thread Grou...	HTTP Request	7	✓	1718	128	7	1
3	23:46:15.655	Thread Grou...	HTTP Request	7	✓	1718	128	7	0
4	23:46:15.704	Thread Grou...	HTTP Request	7	✓	1718	128	7	0
5	23:46:15.755	Thread Grou...	HTTP Request	8	✓	1718	128	8	0
6	23:46:15.805	Thread Grou...	HTTP Request	107	✓	1718	128	107	1
7	23:46:15.855	Thread Grou...	HTTP Request	66	✓	1718	128	66	0
8	23:46:15.905	Thread Grou...	HTTP Request	24	✓	1718	128	23	1
9	23:46:15.955	Thread Grou...	HTTP Request	21	✓	1718	128	21	0
10	23:46:16.005	Thread Grou...	HTTP Request	12	✓	1718	128	12	1
11	23:46:16.055	Thread Grou...	HTTP Request	17	✓	1718	128	17	0
12	23:46:16.104	Thread Grou...	HTTP Request	13	✓	1718	128	13	0
13	23:46:16.154	Thread Grou...	HTTP Request	7	✓	1718	128	7	0
14	23:46:16.204	Thread Grou...	HTTP Request	7	✓	1718	128	7	0
15	23:46:16.254	Thread Grou...	HTTP Request	8	✓	1718	128	8	0
16	23:46:16.304	Thread Grou...	HTTP Request	9	✓	1718	128	9	1
17	23:46:16.354	Thread Grou...	HTTP Request	10	✓	1718	128	10	1
18	23:46:16.405	Thread Grou...	HTTP Request	20	✓	1718	128	20	0
19	23:46:16.454	Thread Grou...	HTTP Request	19	✓	1718	128	19	1
20	23:46:16.504	Thread Grou...	HTTP Request	25	✓	1718	128	25	1
21	23:46:16.555	Thread Grou...	HTTP Request	33	✓	1718	128	32	0
22	23:46:16.605	Thread Grou...	HTTP Request	27	✓	1718	128	27	0
23	23:46:16.655	Thread Grou...	HTTP Request	15	✓	1718	128	15	0
24	23:46:16.704	Thread Grou...	HTTP Request	11	✓	1718	128	11	0
25	23:46:16.754	Thread Grou...	HTTP Request	12	✓	1718	128	12	0
26	23:46:16.804	Thread Grou...	HTTP Request	22	✓	1718	128	22	1
27	23:46:16.854	Thread Grou...	HTTP Request	22	✓	1718	128	22	1
28	23:46:16.905	Thread Grou...	HTTP Request	28	✓	1718	128	28	1
29	23:46:16.954	Thread Grou...	HTTP Request	23	✓	1718	128	23	1
30	23:46:17.004	Thread Grou...	HTTP Request	21	✓	1718	128	21	0
31	23:46:17.055	Thread Grou...	HTTP Request	37	✓	1718	128	37	1
32	23:46:17.104	Thread Grou...	HTTP Request	20	✓	1718	128	20	1
33	23:46:17.154	Thread Grou...	HTTP Request	21	✓	1718	128	20	1
34	23:46:17.204	Thread Grou...	HTTP Request	24	✓	1718	128	24	1
35	23:46:17.255	Thread Grou...	HTTP Request	27	✓	1718	128	26	0

Scroll automatically? Child samples? No of Samples 1000 Latest Sample 22 Average 21 Deviation 9

- ii. Menghentikan salah satu MySQL server ditengah-tengah load test endpoint /api/posts menggunakan JMeter
- Ketika salah satu MySQL server dimatikan, ada 28 dari 1000 request yang menghasilkan error. Ini disebabkan karena ProxySQL masih menggunakan host MySQL yang telah dimatikan. Setelah ProxySQL mengetahui host tersebut tidak bisa dikontak, maka ProxySQL menggunakan host lain yang ada di pool.

```

* Management: https://landscape.canonical.com
* Support: https://ubuntu.com/advantage

110 packages can be updated.
68 updates are security updates.

Last login: Sat Dec 7 14:30:06 2019 from 10.0.2.2
vagrant@node107:~$ sudo service mysql stop
vagrant@node107:~$

```

Sample #	Start Time	Thread Name	Label	Sample Time(ms)	Status	Bytes	Sent Bytes	Latency	Connect
1	23:50:08.288	Thread Grou...	HTTP Request	103	✗	37889	128	86	86
2	23:50:08.390	Thread Grou...	HTTP Request	29	✗	37889	128	15	15
3	23:50:08.438	Thread Grou...	HTTP Request	54	✗	37889	128	27	27
5	23:50:08.488	Thread Grou...	HTTP Request	48	✗	37889	128	29	29
8	23:50:08.593	Thread Grou...	HTTP Request	85	✗	37889	128	71	71
12	23:50:08.837	Thread Grou...	HTTP Request	43	✗	37608	128	30	30
14	23:50:08.938	Thread Grou...	HTTP Request	45	✗	37889	128	21	21
15	23:50:08.988	Thread Grou...	HTTP Request	64	✗	37608	128	48	48
16	23:50:09.088	Thread Grou...	HTTP Request	37	✗	37889	128	19	19
17	23:50:09.138	Thread Grou...	HTTP Request	29	✗	37889	128	17	17
18	23:50:09.188	Thread Grou...	HTTP Request	30	✗	37889	128	18	18
19	23:50:09.240	Thread Grou...	HTTP Request	81	✗	37889	128	58	58
20	23:50:09.289	Thread Grou...	HTTP Request	46	✗	37889	128	33	33
21	23:50:09.339	Thread Grou...	HTTP Request	27	✗	37889	128	14	14
22	23:50:09.390	Thread Grou...	HTTP Request	26	✗	37889	128	13	13
23	23:50:09.439	Thread Grou...	HTTP Request	25	✗	37889	128	13	13
24	23:50:09.539	Thread Grou...	HTTP Request	105	✗	37889	128	61	61
25	23:50:09.638	Thread Grou...	HTTP Request	49	✗	37889	128	27	27
26	23:50:09.738	Thread Grou...	HTTP Request	57	✗	37889	128	30	30
31	23:50:09.788	Thread Grou...	HTTP Request	84	✗	37608	128	76	76
32	23:50:09.838	Thread Grou...	HTTP Request	41	✗	37889	128	25	25
33	23:50:09.888	Thread Grou...	HTTP Request	25	✗	37889	128	14	14
34	23:50:09.938	Thread Grou...	HTTP Request	27	✗	37608	128	19	19
38	23:50:10.088	Thread Grou...	HTTP Request	82	✗	37889	128	56	56
39	23:50:10.188	Thread Grou...	HTTP Request	32	✗	37608	128	22	22
43	23:50:10.291	Thread Grou...	HTTP Request	118	✗	37889	128	91	91
44	23:50:10.437	Thread Grou...	HTTP Request	35	✗	37889	128	19	19
48	23:50:10.588	Thread Grou...	HTTP Request	108	✗	37889	128	44	44
4	23:50:08.338	Thread Grou...	HTTP Request	183	✓	1718	128	183	183
6	23:50:08.538	Thread Grou...	HTTP Request	15	✓	1718	128	15	15
7	23:50:08.639	Thread Grou...	HTTP Request	24	✓	1718	128	24	24
9	23:50:08.688	Thread Grou...	HTTP Request	16	✓	1718	128	16	16
10	23:50:08.738	Thread Grou...	HTTP Request	14	✓	1718	128	14	14
11	23:50:08.788	Thread Grou...	HTTP Request	12	✓	1718	128	12	12

Scroll automatically? Child samples? No of Samples 1000 Latest Sample 39 Average 33 Deviation 3

- Tambahkan monitoring dashboard menggunakan Grafana
 - Install dan konfigurasi MySQL exporter
 - Pada masing-masing MySQL server, jalankan command berikut

```
wget
https://github.com/prometheus/mysqlld\_exporter/releases/download/v0.11.0/my
sqlld\_exporter-0.11.0.linux-amd64.tar.gz

tar xvzf /vagrant/mysqlld_exporter-0.11.0.linux-amd64.tar.gz
cd mysqlld_exporter-0.11.0.linux-amd64/

sudo mv mysqlld_exporter /usr/local/bin/
```

- Buat file konfigurasi

```
sudo nano /etc/.exporter.cnf
```

- Isi file /etc/.exporter.cnf sebagai berikut, nilai host mengikuti IP address masing-masing server

```
[client]
host=192.168.16.107
user=adis
password=password
```

- Buat MySQL exporter menjadi service, pada masing-masing server

```
sudo nano /lib/systemd/system/mysql_exporter.service
```

- File /lib/systemd/system/mysql_exporter.service

```
[Unit]
Description=MySQL Exporter

[Service]
Type=simple
Restart=always
ExecStart=/usr/local/bin/mysqlld_exporter \
--config.my-cnf /etc/.exporter.cnf \
--collect.auto_increment.columns \
--collect.binlog_size \
--collect.engine_innodb_status \
```

```
--collect.global_status \  
--web.listen-address=192.168.16.107:9104
```

```
[Install]  
WantedBy=multi-user.target
```

- Jalankan service MySQL exporter

```
sudo systemctl daemon-reload  
sudo systemctl start mysql_exporter.service  
sudo systemctl status mysql_exporter.service
```

- b. Install dan konfigurasi Prometheus
- Buat direktori Prometheus dan user Prometheus

```
sudo useradd --no-create-home --shell /bin/false prometheus  
sudo mkdir /etc/prometheus  
sudo mkdir /var/lib/prometheus  
sudo chown prometheus:prometheus /etc/prometheus  
sudo chown prometheus:prometheus /var/lib/prometheus
```

- Download dan extract file Prometheus

```
curl -LO  
https://github.com/prometheus/prometheus/releases/download/v2.3.2/prometh  
eus-2.3.2.linux-amd64.tar.gz  
tar -xvf prometheus-2.3.2.linux-amd64.tar.gz  
mv prometheus-2.3.2.linux-amd64 prometheus-files
```

- Copy file Prometheus dan ubah ownership

```
sudo cp prometheus-files/prometheus /usr/local/bin/  
sudo cp prometheus-files/promtool /usr/local/bin/  
sudo chown prometheus:prometheus /usr/local/bin/prometheus  
sudo chown prometheus:prometheus /usr/local/bin/promtool  
  
sudo cp -r prometheus-files/consoles /etc/prometheus  
sudo cp -r prometheus-files/console_libraries /etc/prometheus  
sudo chown -R prometheus:prometheus /etc/prometheus/consoles  
sudo chown -R prometheus:prometheus /etc/prometheus/console_libraries
```

- Konfigurasi Prometheus

```
sudo nano /etc/prometheus/prometheus.yml
```

- File /etc/prometheus/prometheus.yml

```
global:
  scrape_interval: 10s

scrape_configs:
  - job_name: 'prometheus'
    scrape_interval: 5s
    static_configs:
      - targets: ['localhost:9090']

  - job_name: 'mysql'
    scrape_interval: 5s
    static_configs:
      - targets:
        ['192.168.16.107:9104','192.168.16.108:9104','192.168.16.110:9104']
```

- Ubah ownership dan jalankan service prometheus

```
sudo chown prometheus:prometheus /etc/prometheus/prometheus.yml

sudo systemctl start prometheus
```

- Akses localhost:9090 melalui web browser untuk melihat dashboard Prometheus

Prometheus Alerts Graph Status ▾ Help

☐ Enable query history [Try experimental React UI](#)

Expression (press Shift+Enter for newlines)

Execute - insert metric at cursor - ▾

Graph Console

◀ Moment ▶

Element	Value
no data	

[Remove Graph](#)


Add Graph

c. Install dan konfigurasi Grafana

- Download Grafana


```
sudo apt-get install -y apt-transport-https
sudo apt-get install -y software-properties-common wget
sudo add-apt-repository "deb https://packages.grafana.com/oss/deb stable main"
sudo apt-get update
sudo apt-get install grafana
```

- Akses localhost:9090 untuk membuka dashboard Grafana. Tambahkan datasource baru, pilih Prometheus.


 Add data source
Choose a data source type

Filter by name or type [Cancel](#)

Time series databases

 Prometheus
Open source time series database & alerting

- Isi <http://localhost:9090> pada field URL

 **Data Sources / Prometheus-1**
Type: Prometheus

Settings

Dashboards

Name

Prometheus-1

Default

☐

HTTP

URL

http://localhost:9090

Whitelisted Cookies

Add Name

Add

Auth

Basic auth	<input type="checkbox"/>	With Credentials	<input type="checkbox"/>
TLS Client Auth	<input type="checkbox"/>	With CA Cert	<input type="checkbox"/>
Skip TLS Verify	<input type="checkbox"/>		
Forward OAuth Identity	<input type="checkbox"/>		

Scrape Interval

Query timeout

60s

HTTP Method

Choose

Misc

Custom query parameters


Example: max_source_resolution=5m&timeout=10

Save & Test

Delete

Back

- Isi kode 7362 untuk MySQL dashboard



Import

Import dashboard from file or Grafana.com

Upload .json file

Grafana.com Dashboard

Paste Grafana.com dashboard url or id

7362

⚠ Dashboard not found

Or paste JSON

Load

- Tunggu beberapa saat agar statistik pada dashboard muncul

