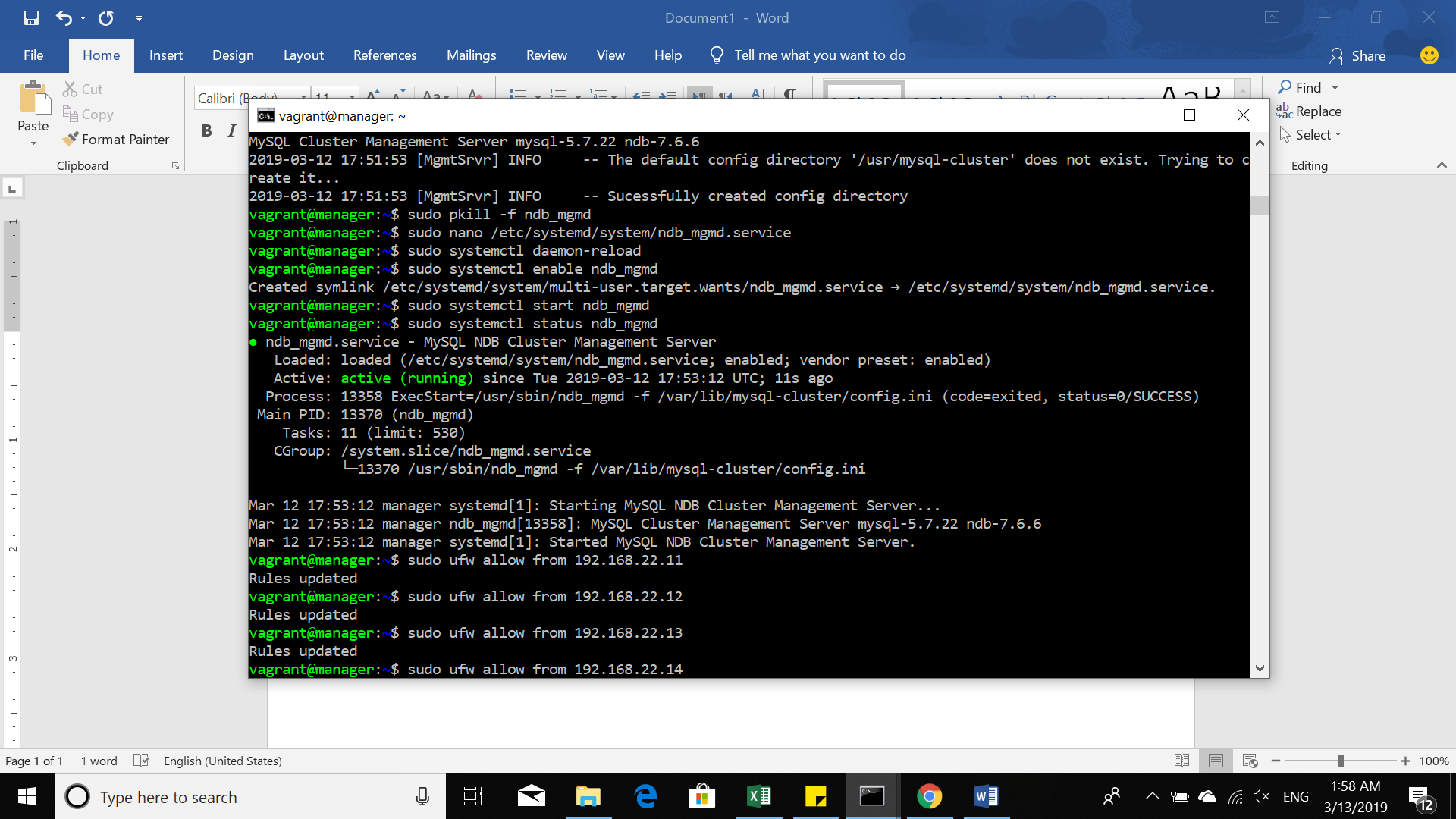
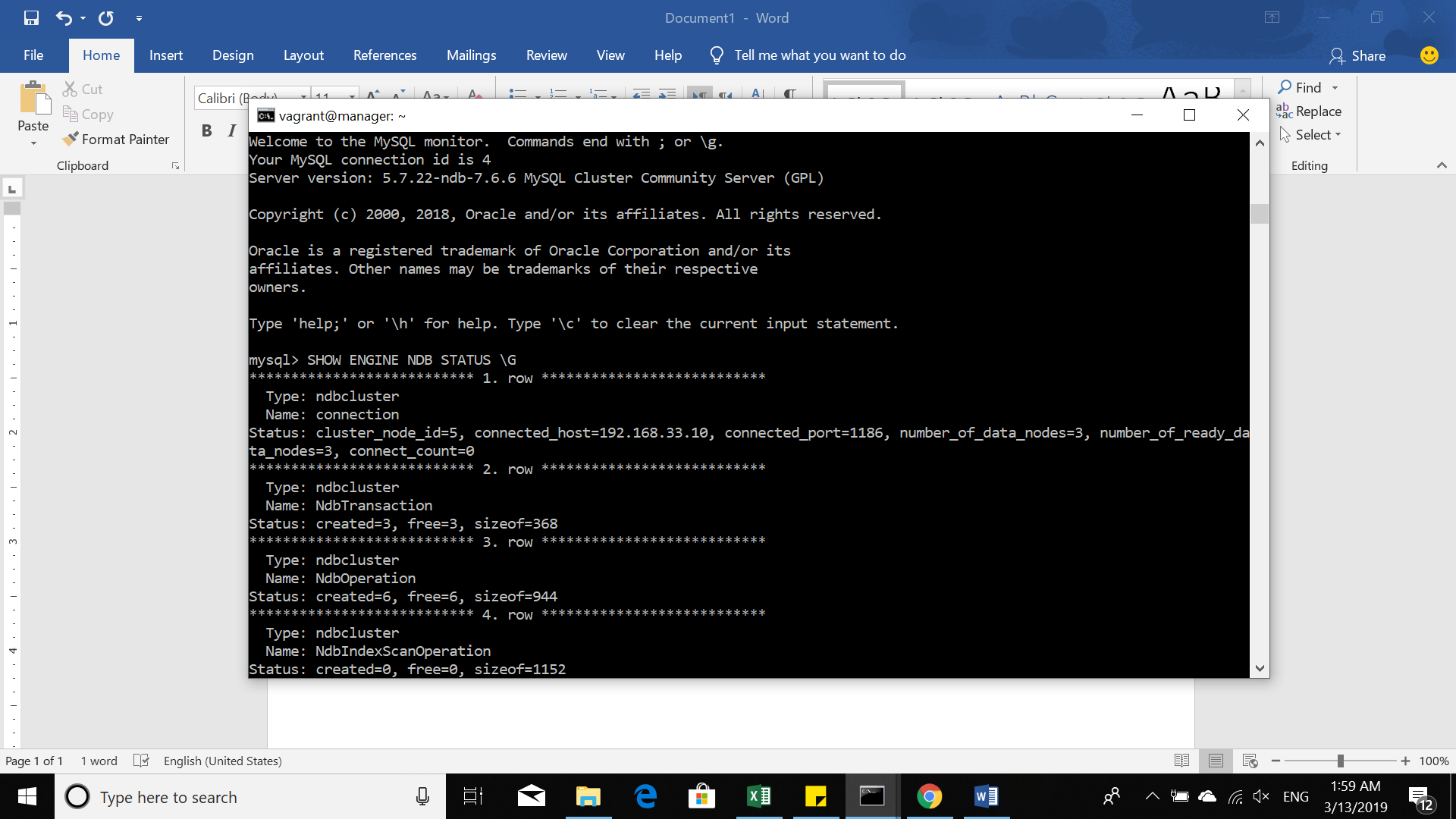
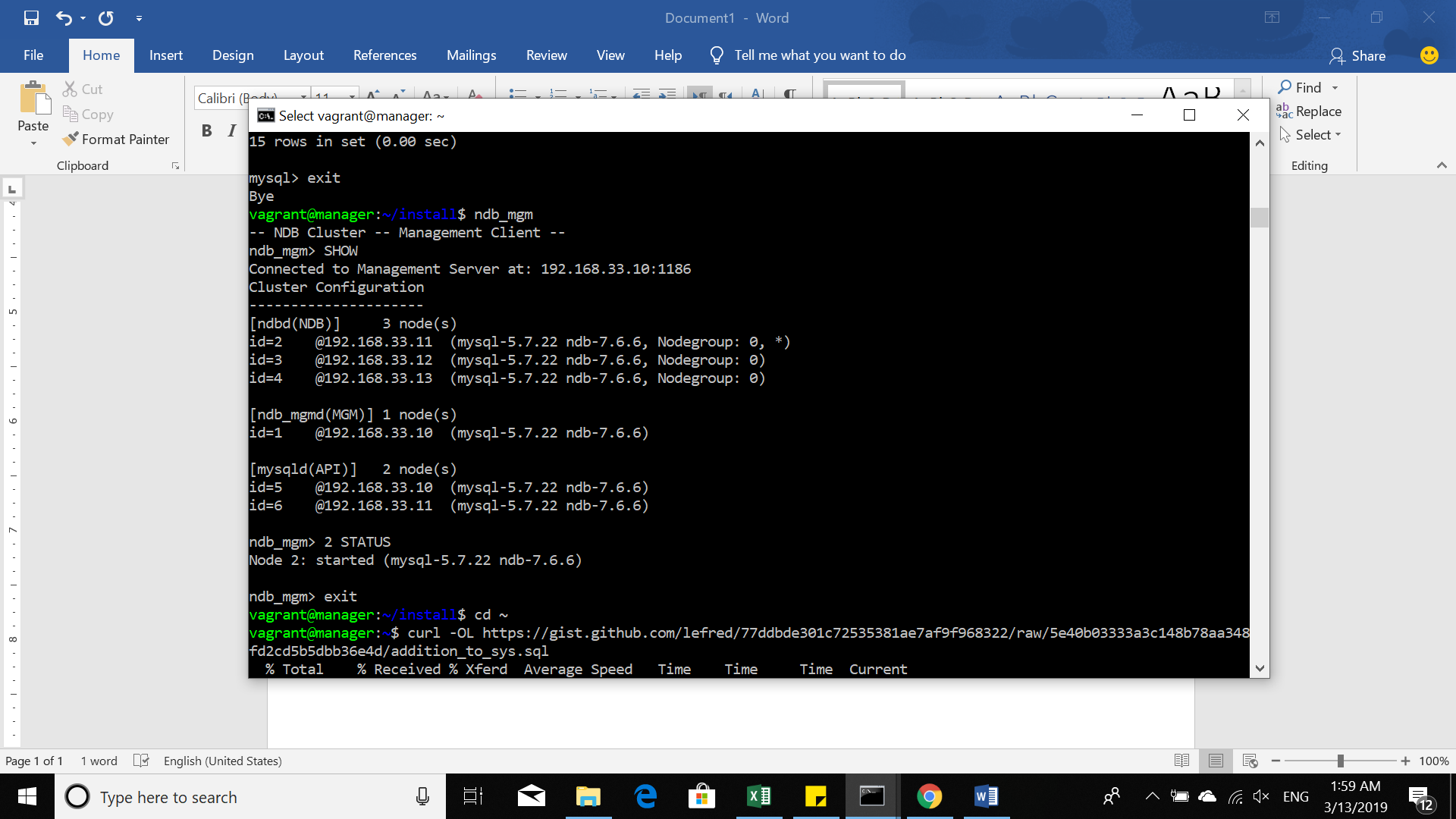
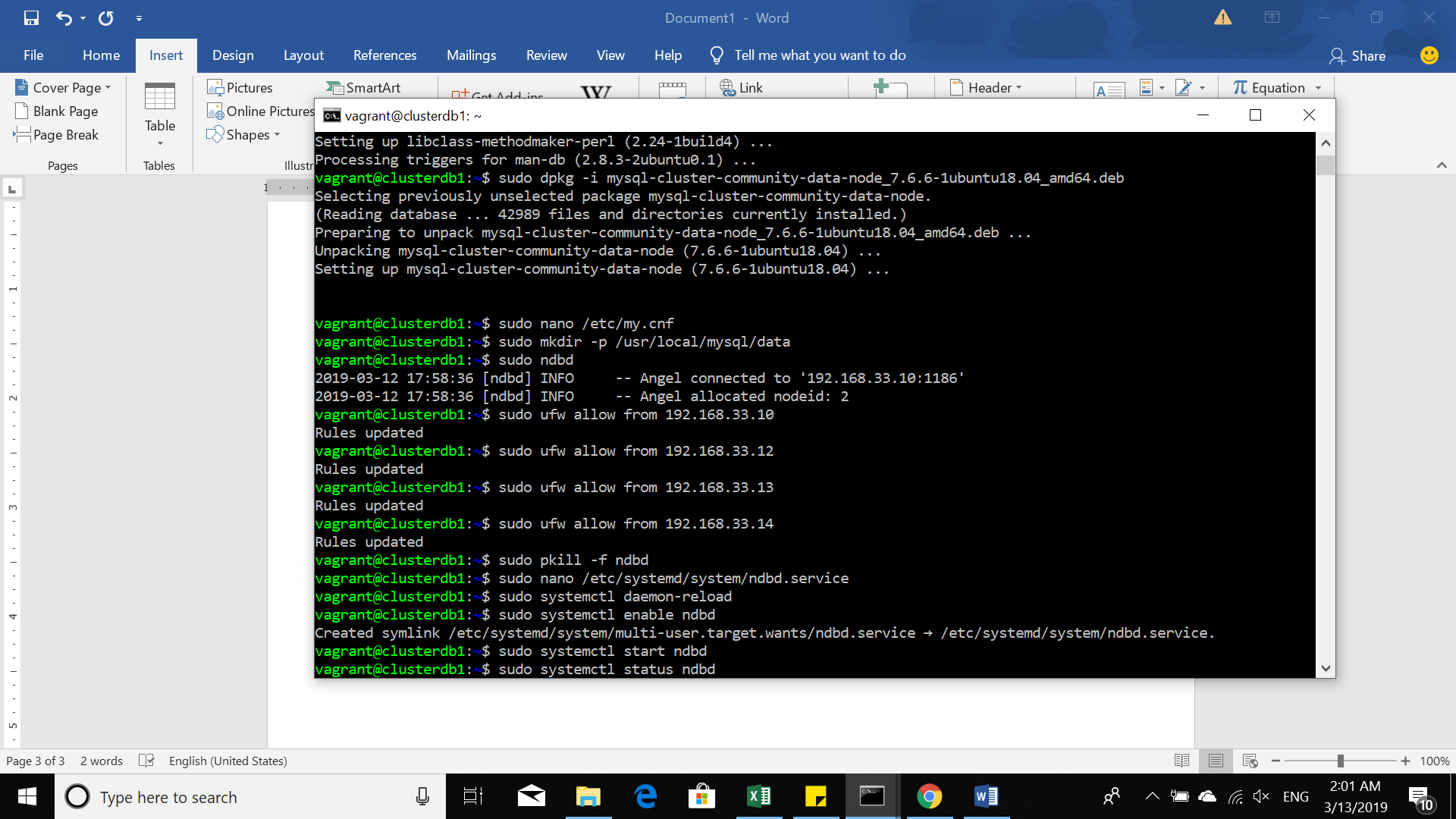
**MANAGER**

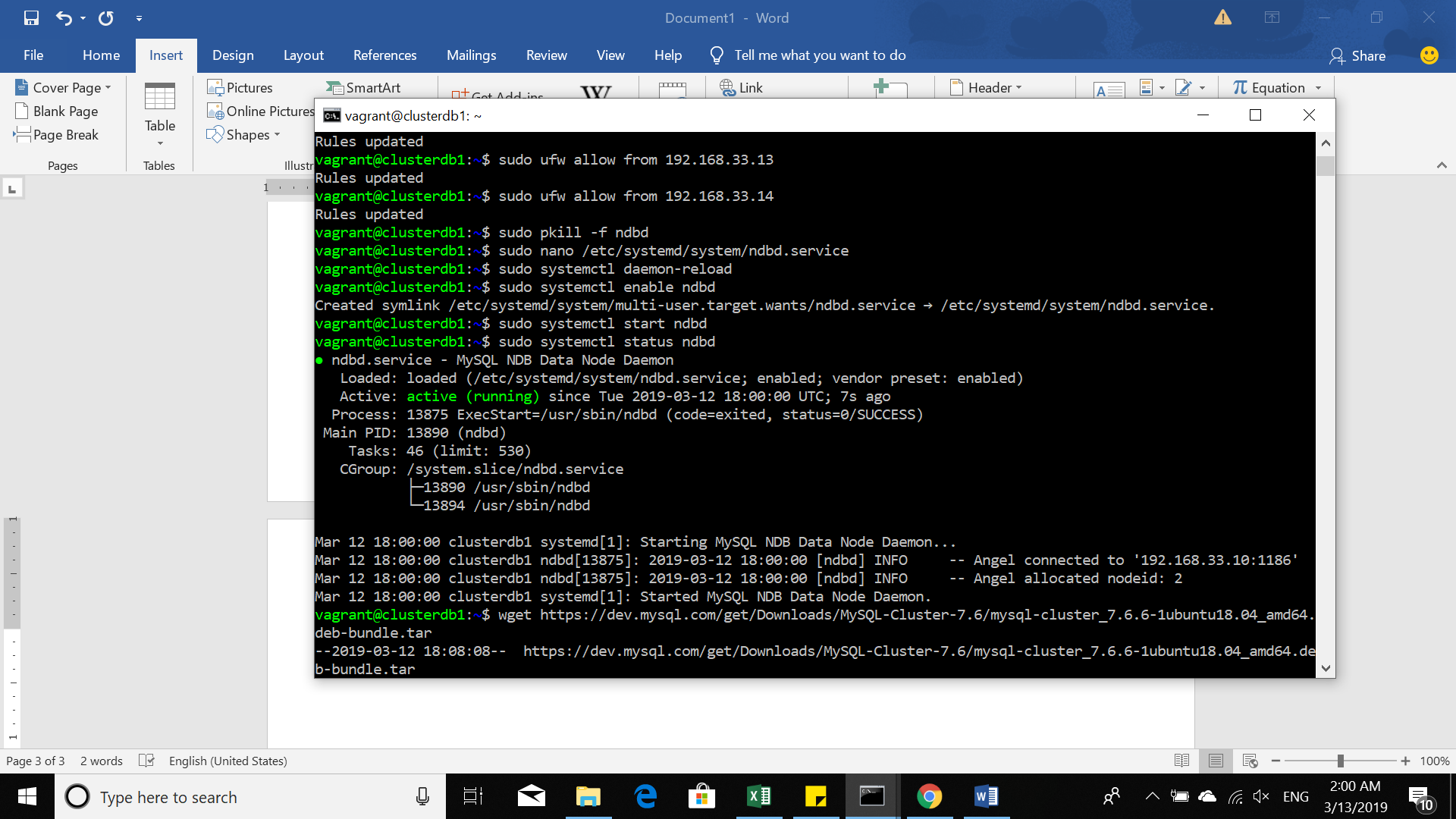


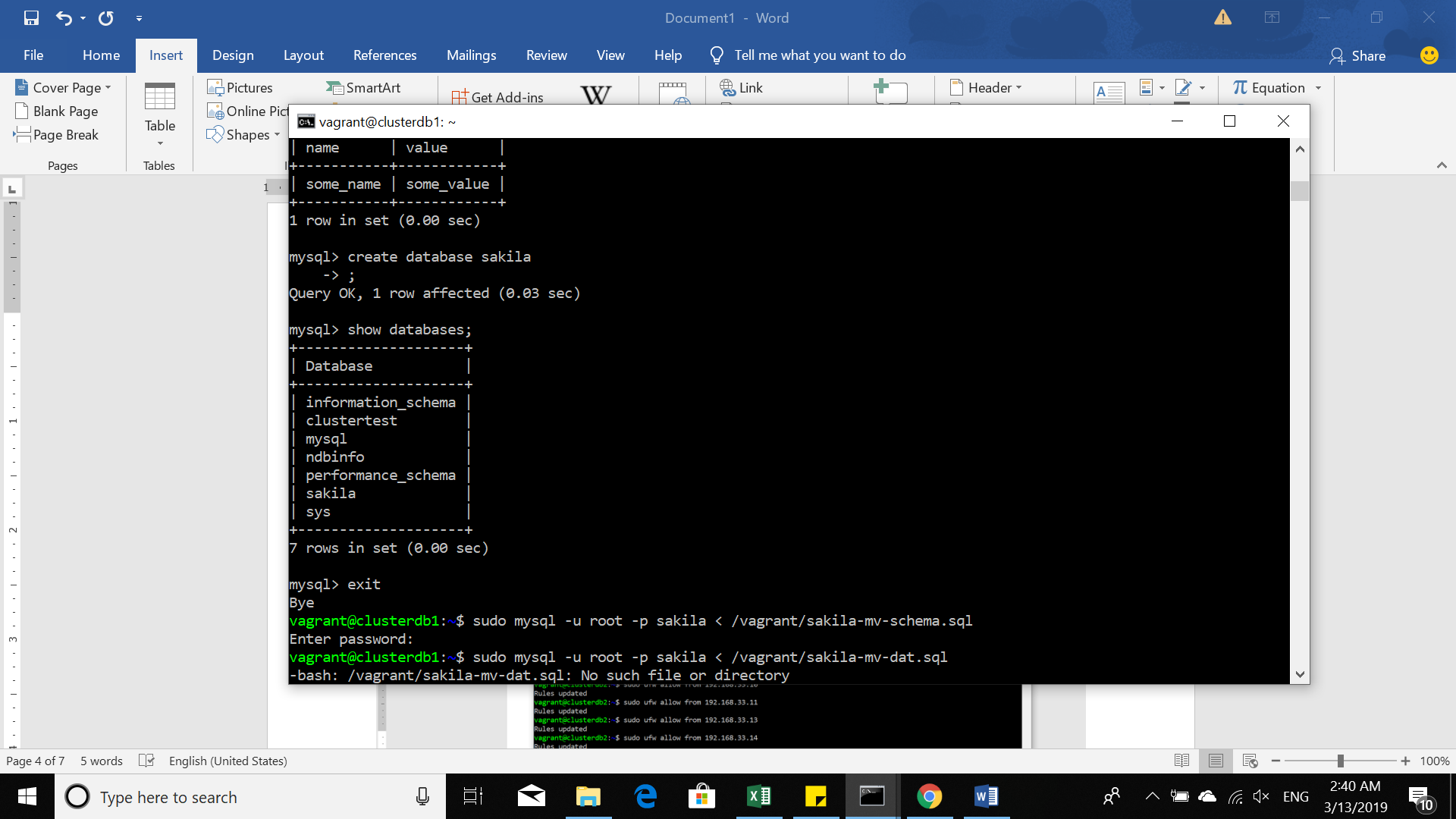


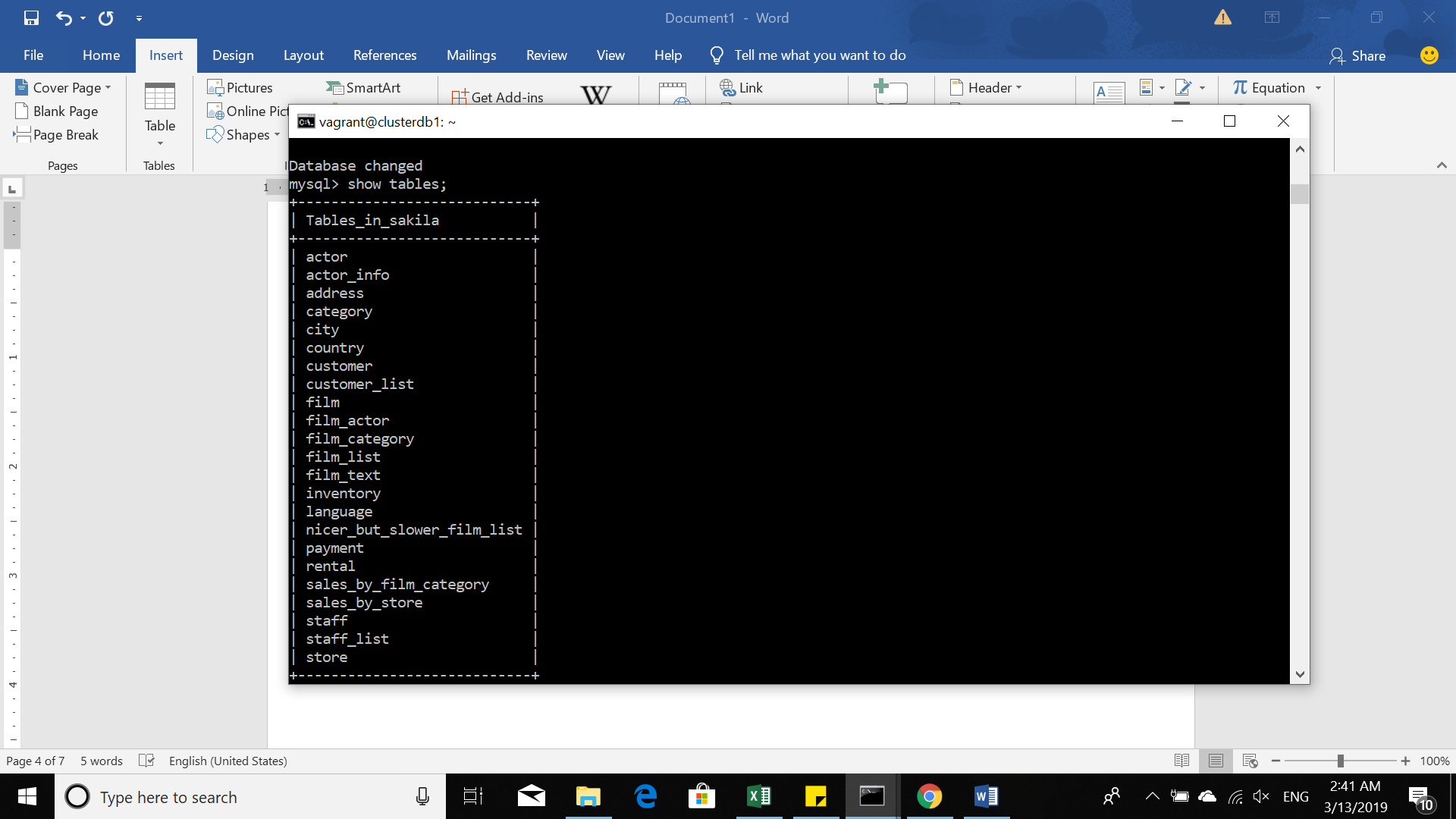


**DB1**

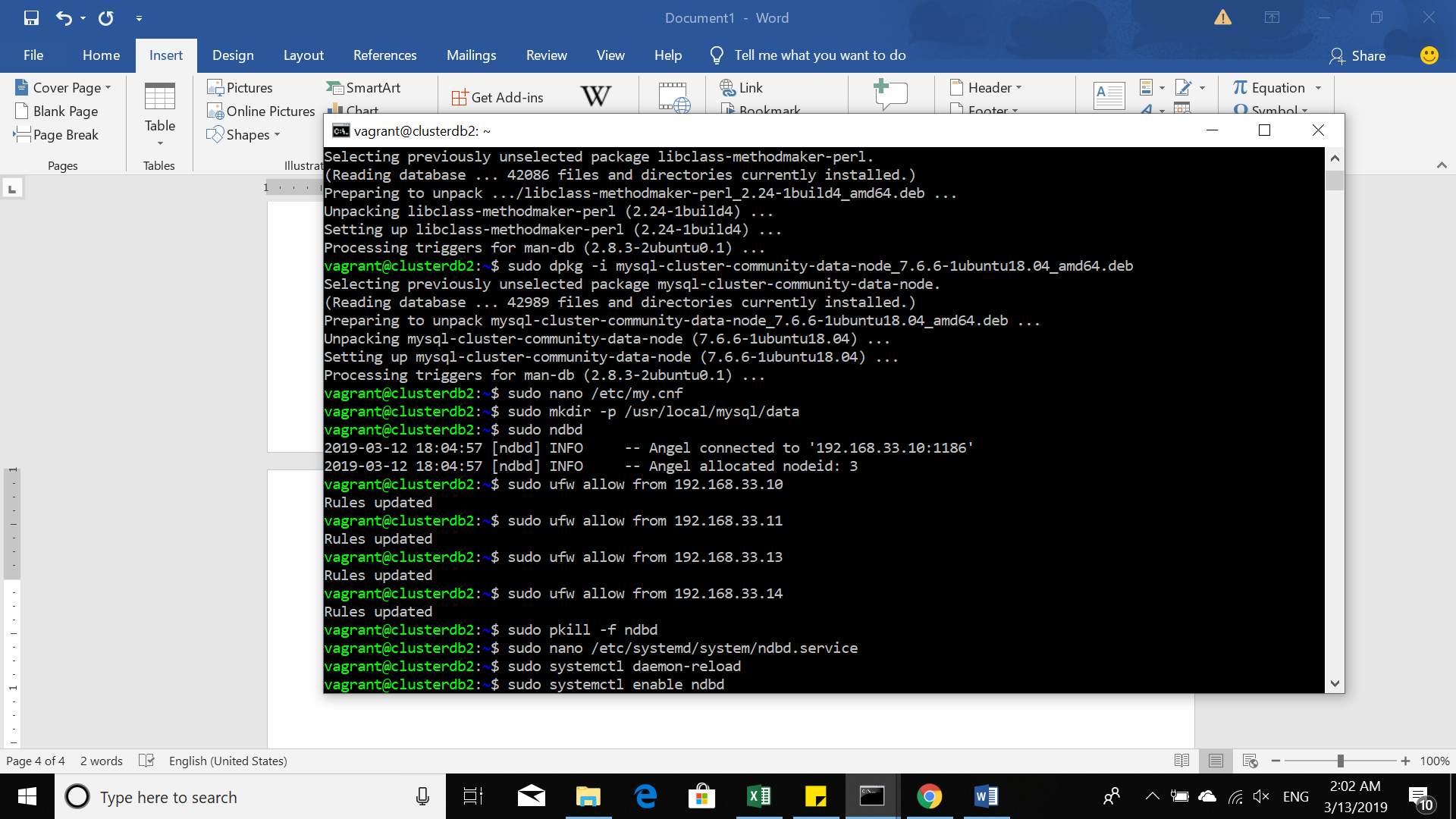


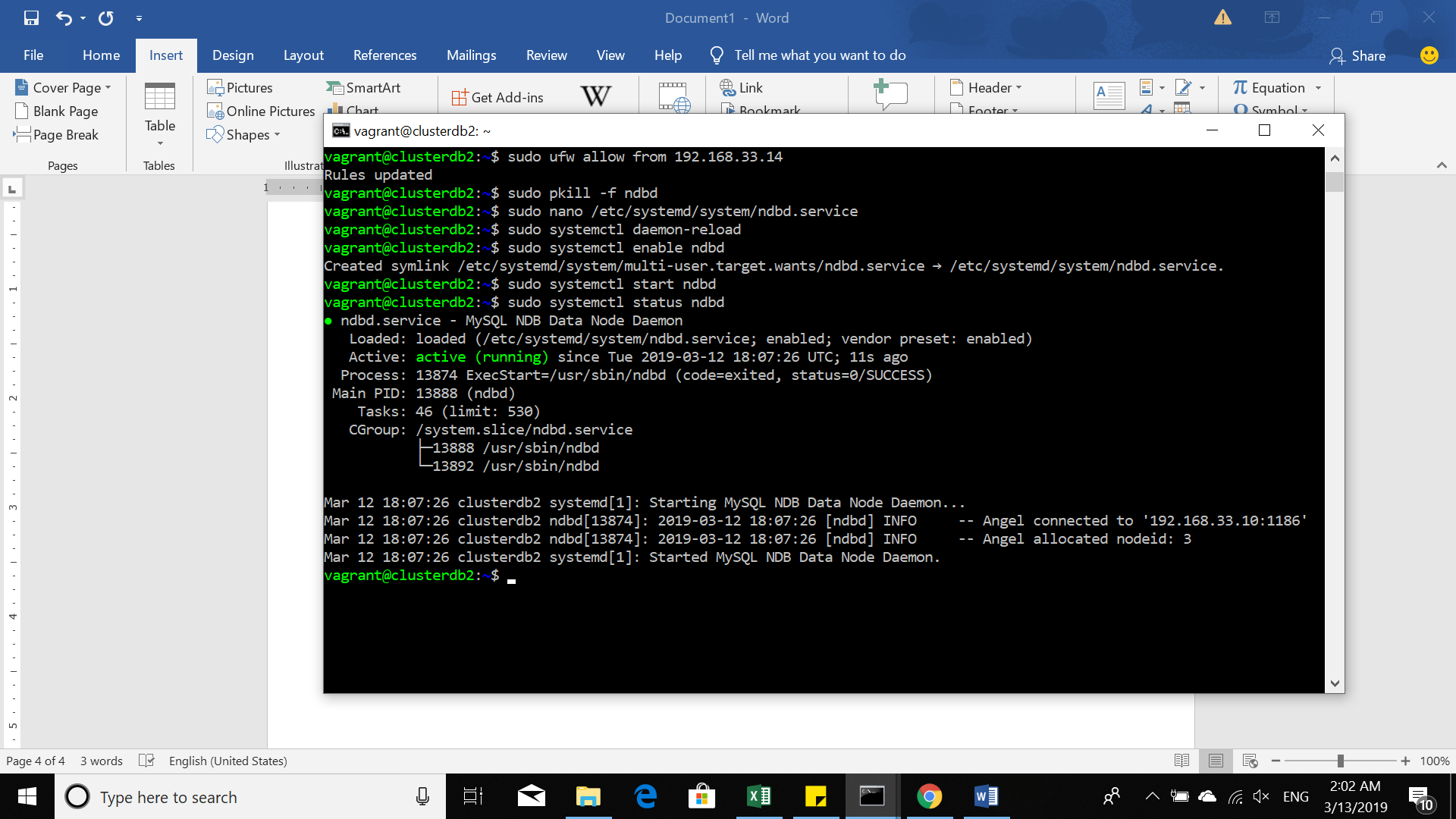




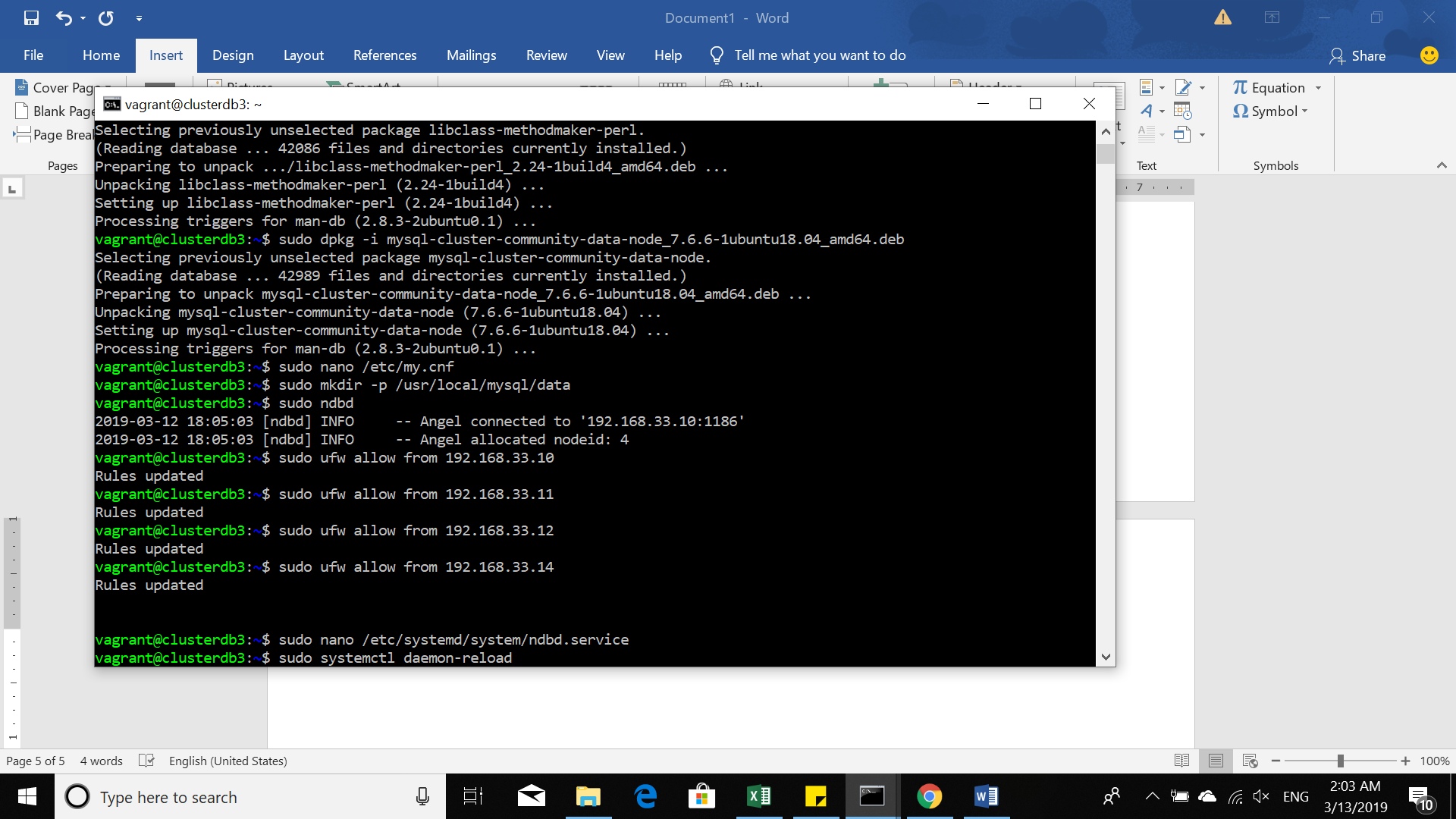


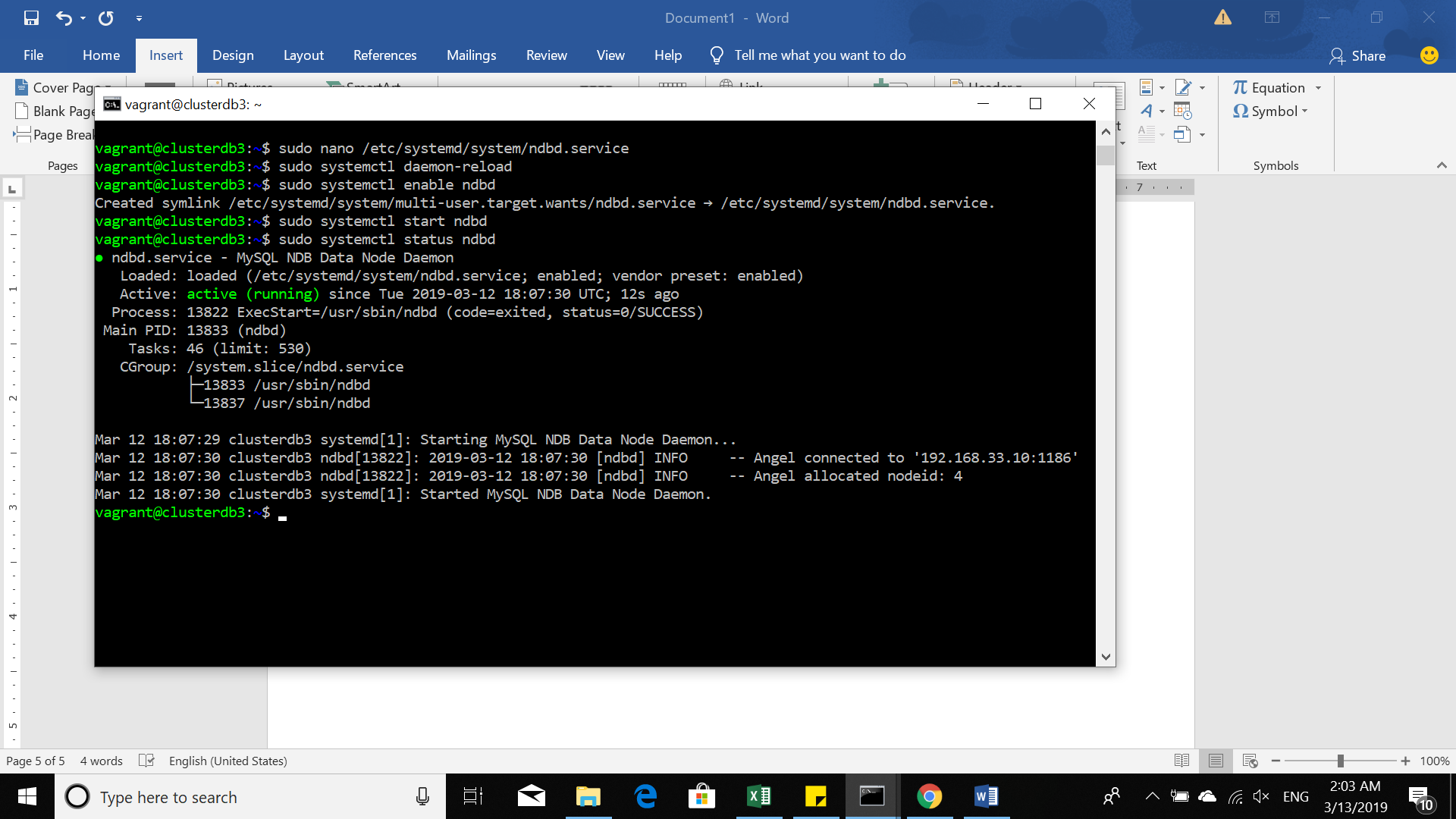
**DB2**



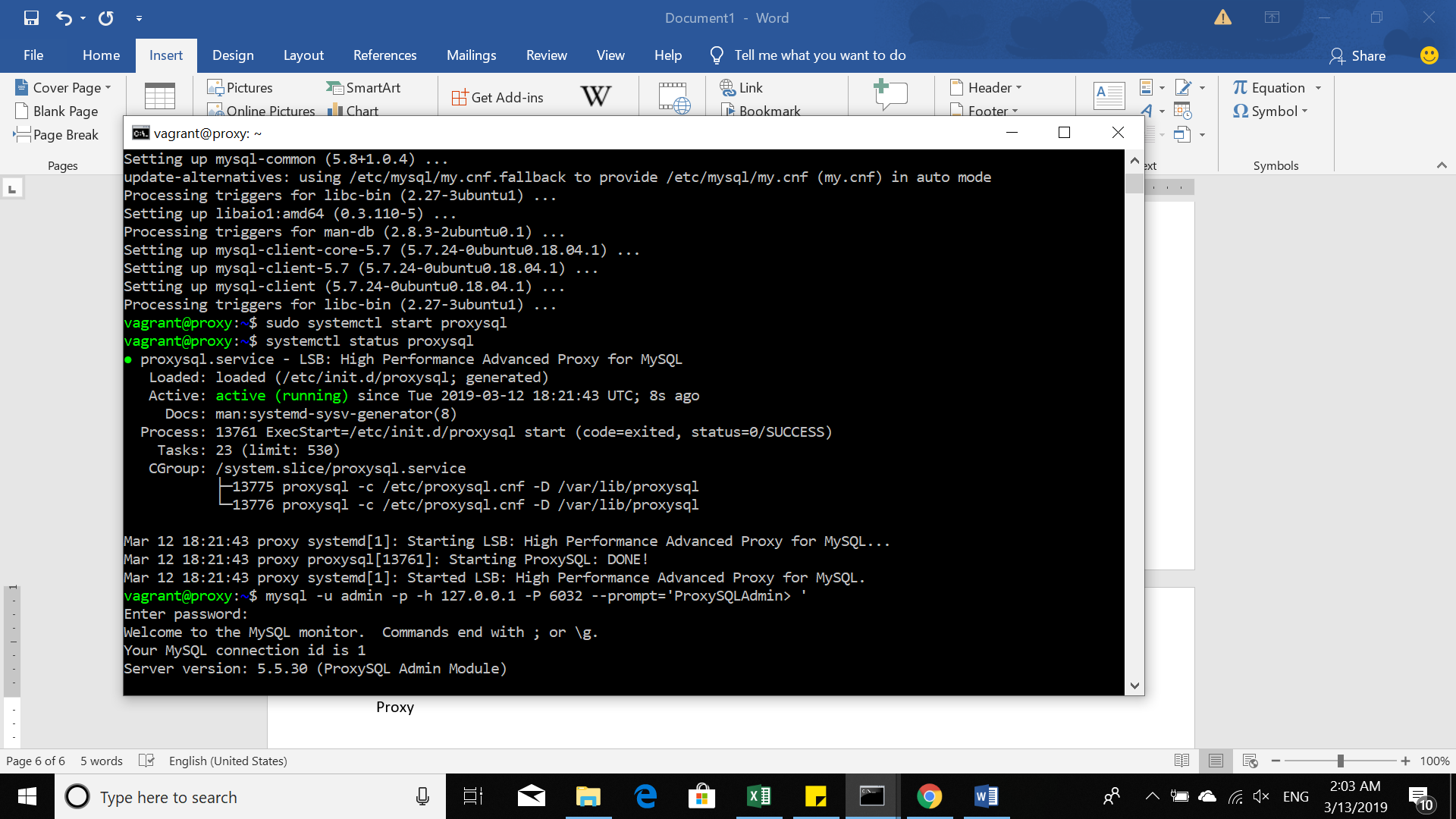


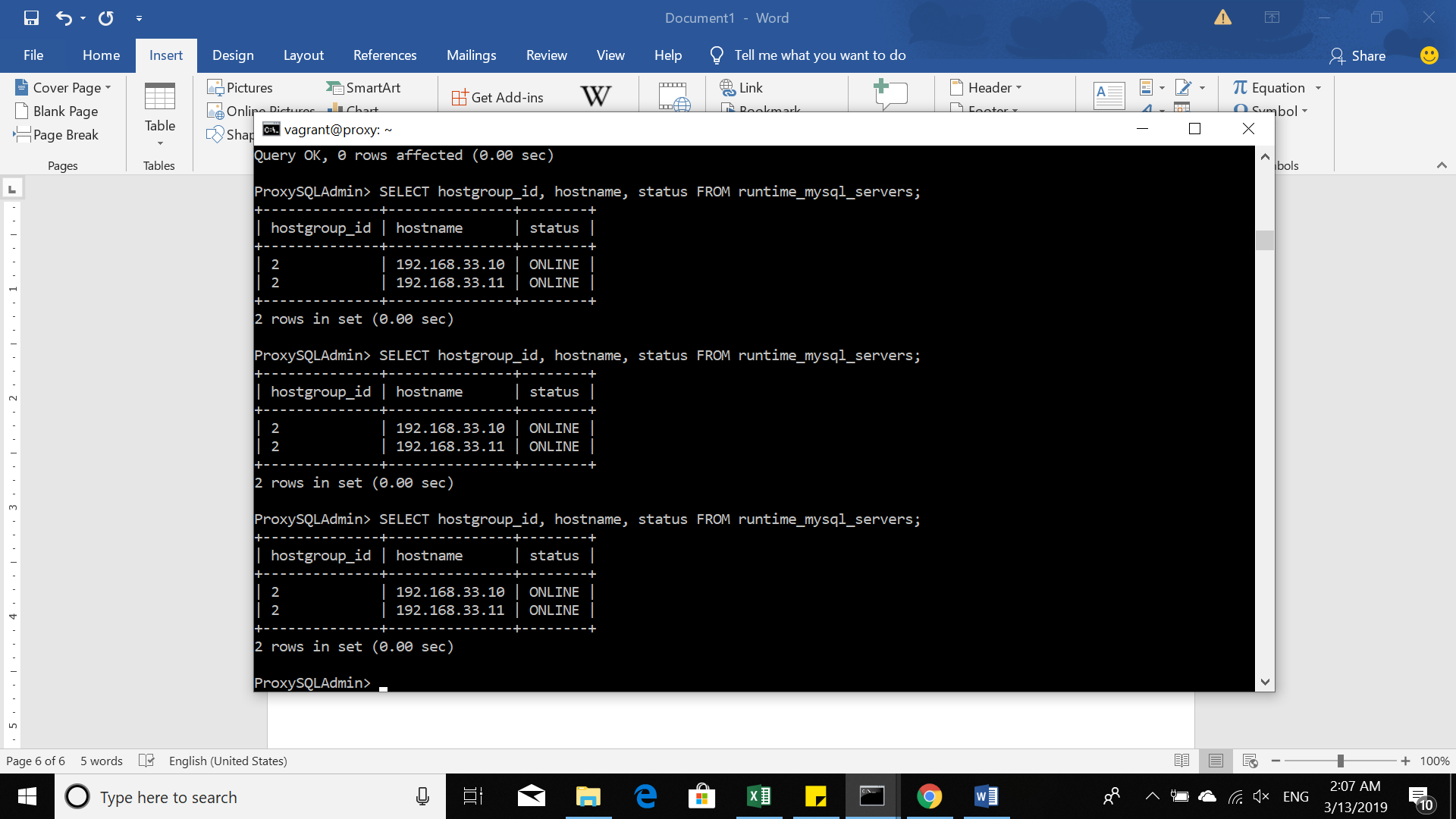
**DB3**





**Proxy**





Implementasi MYSQL Cluster :

1. **Yang dibutuhkan**

* Vagrant
* Virtualbox
* MySQL
* MySQL data Sample

1. **Model Arsitektur**

|  |  |  |
| --- | --- | --- |
| IP Address | Hostname | Deskripsi |
| 192.168.33.10 | Manager | Node Manager |
| 192.168.33.11 | CLusterdb1 | Node1 |
| 192.168.33.12 | Clusterdb2 | Node2 |
| 192.168.33.13 | Clusterdb3 | Node3 |
| 192.168.33.14 | Proxy | Load Balancer |

1. Instalasi
2. Tahap awal mengestrak file mysql-cluster.zip
3. Melakukan konfigurasi pada Vagrantfile.
4. Melakukan vagrant up pada file yang mysql-cluster
5. Melakukan vagrant ssh di setiap clusternya

**Langkah-langkah membuat multi-Node MYSQL Cluster** :

**Langkah I**: Melakukan instalasi dan konfigurasi pada Cluster Manager

**Langkah II** : Instalasi **ndb\_mgmd** menggunakan **dpkg**:

$ sudo dpkg -i mysql-cluster-community-management-server\_7.6.6 1ubuntu18.04\_amd64.de

**Langkah III** : Membuat directory **/var/lib/mysql-cluster**

$ sudo mkdir /var/lib/mysql-cluster

**Langkah IV :** Membuat dan mengedit konfigurasi pada text editor

$ sudo nano /var/lib/mysql-cluster/config.ini

**Disesuaikan dengan yang telah kita tentukan**

[ndbd default]

# Options affecting ndbd processes on all data nodes:

NoOfReplicas=2 # Number of replicas

[ndb\_mgmd]

# Management process options:

hostname=198.51.100.2 # Hostname of the manager

datadir=/var/lib/mysql-cluster # Directory for the log files

[ndbd]

hostname=198.51.100.0 # Hostname/IP of the first data node

NodeId=2 # Node ID for this data node

datadir=/usr/local/mysql/data # Remote directory for the data files

[ndbd]

hostname=198.51.100.1 # Hostname/IP of the second data node

NodeId=3 # Node ID for this data node

datadir=/usr/local/mysql/data # Remote directory for the data files

[mysqld]

# SQL node options:

hostname=198.51.100.2 # In our case the MySQL server/client is on the same Droplet as the cluster manager

**Langkah 6 :** Start Manager dengan menggunakan **ndb\_mgmd**

$ sudo ndb\_mgmd -f /var/lib/mysql-cluster/config.ini

**Hasil keluaran (output):**

MySQL Cluster Management Server mysql-5.7.22 ndb-7.6.6

2018-07-25 21:48:39 [MgmtSrvr] INFO -- The default config directory '/usr/mysql-cluster' does not exist. Trying to create it...

2018-07-25 21:48:39 [MgmtSrvr] INFO -- Successfully created config directory

**Langkah 8 :** Start Cluster Management server

$ sudo pkill -f ndb\_mgmd

**Langkah 9** : Membuka kembali file

$ sudo nano /etc/systemd/system/ndb\_mgmd.service

/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

**Langkah 10** : Mereload kembali sistem konfigurasi manager dengan menggunakan

**daemon-reload:**

$ sudo systemctl daemon-reload

**Langkah 11 :** Melakukan enable service MYSQL Cluster Manager start

$ sudo systemctl enable ndb\_mgmd

**Langkah 12** : Melakukan Start kembali

$ sudo systemctl start ndb\_mgmd

**Langkah 13 :** Melakukan verify pada NDB Cluster Management service yang sedang dijalankan

$ sudo systemctl status ndb\_mgmd

Hasil (output):

ndb\_mgmd.service - MySQL NDB Cluster Management Server

Loaded: loaded (/etc/systemd/system/ndb\_mgmd.service; enabled; vendor preset: enabled)

Active: active (running) since Thu 2018-07-26 21:23:37 UTC; 3s ago

Process: 11184 ExecStart=/usr/sbin/ndb\_mgmd -f /var/lib/mysql-cluster/config.ini (code=exited, status=0/SUCCESS)

Main PID: 11193 (ndb\_mgmd)

Tasks: 11 (limit: 4915)

CGroup: /system.slice/ndb\_mgmd.service

└─11193 /usr/sbin/ndb\_mgmd -f /var/lib/mysql-cluster/config.ini

**Langkah 14 :**

* $ sudo ufw allow from 198.51.100.0
* $ sudo ufw allow from 198.51.100.1

**Hasil(output):**

Rule added

**Konfigurasi dan menjalankan MYSQL Server dan Client**

* $cd ~
* $ wget https://dev.mysql.com/get/Downloads/MySQL-Cluster-7.6/mysql-cluster\_7.6.6-1ubuntu18.04\_amd64.deb-bundle.tar

**Pertama, install pada directory**

$ mkdir install

**Extract ke dalam directory**

$ tar -xvf mysql-cluster\_7.6.6-1ubuntu18.04\_amd64.deb-bundle.tar -C install/

**Pindah folder dan extract kembali MySQL Cluster**

$ cd install

**Install MySQL**

* $ sudo dpkg -i mysql-common\_7.6.6-1ubuntu18.04\_amd64.deb
* $ sudo dpkg -i mysql-cluster-community-client\_7.6.6-1ubuntu18.04\_amd64.deb
* $ sudo dpkg -i mysql-client\_7.6.6-1ubuntu18.04\_amd64.deb
* $ sudo dpkg -i mysql-cluster-community-server\_7.6.6-1ubuntu18.04\_amd64.deb

**Kemudian install MySQL dengan menggunakan** dpkg:

$ sudo dpkg -i mysql-server\_7.6.6-1ubuntu18.04\_amd64.deb

**Membuka file editor:**

$ sudo nano /etc/mysql/my.cnf

**Maka, hasil(output):**

/etc/mysql/my.cnf

# Copyright (c) 2015, 2016, Oracle and/or its affiliates. All rights reserved.

#

# This program is free software; you can redistribute it and/or modify

# it under the terms of the GNU General Public License as published by

# the Free Software Foundation; version 2 of the License.

#

# This program is distributed in the hope that it will be useful,

# but WITHOUT ANY WARRANTY; without even the implied warranty of

# MERCHANTABILITY or FITNESS FOR A PARTICULAR PURPOSE. See the

# GNU General Public License for more details.

#

# You should have received a copy of the GNU General Public License

# along with this program; if not, write to the Free Software

# Foundation, Inc., 51 Franklin St, Fifth Floor, Boston, MA 02110-1301 USA

#

# The MySQL Cluster Community Server configuration file.

#

# For explanations see

# http://dev.mysql.com/doc/mysql/en/server-system-variables.html

# \* IMPORTANT: Additional settings that can override those from this file!

# The files must end with '.cnf', otherwise they'll be ignored.

#

!includedir /etc/mysql/conf.d/

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

**Konfigurasi:**

/etc/mysql/my.cnf

. . .

[mysqld]

# Options for mysqld process:

ndbcluster # run NDB storage engine

[mysql\_cluster]

# Options for NDB Cluster processes:

ndb-connectstring=198.51.100.2 # location of management server

**Restart MySQL server :**

$ sudo systemctl enable mysql

$ sudo systemctl restart mysql

**Langkah selanjutnya : Verifikasi MySQL Cluster Installaton**

$ mysql -u root -p

**Hasil(output):**

Welcome to the MySQL monitor. Commands end with ; or \g.

Your MySQL connection id is 3

Server version: 5.7.22-ndb-7.6.6 MySQL Cluster Community Server (GPL)

Copyright (c) 2000, 2018, Oracle and/or its affiliates. All rights reserved.

Oracle is a registered trademark of Oracle Corporation and/or its

affiliates. Other names may be trademarks of their respective

owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.

mysql>

**Pada MySQL client, menjalankan perintah :**

Mysql> SHOW ENGINE NDB STATUS \G

**Hasil(output):**

Output

\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\* 1. row \*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*

Type: ndbcluster

Name: connection

Status: cluster\_node\_id=4, connected\_host=198.51.100.2, connected\_port=1186, number\_of\_data\_nodes=2, number\_of\_ready\_data\_nodes=2, connect\_count=0

. . .

**Membuka console Cluster Management , dan menggunakan command « ndb\_mgm »**

$ ndb\_mgm

**Hasil(Output) :**

Output

-- NDB Cluster -- Management Client --

ndb\_mgm>

**Melakukan command pada concole yaitu “SHOW” dan “ENTER”**

Ndb\_mgm> SHOW

**Hasil(output):**

**Melakukan command “STATUS”**

Ndb\_mgm> 2 STATUS

Output

Connected to Management Server at: 198.51.100.2:1186

Cluster Configuration

---------------------

[ndbd(NDB)] 2 node(s)

id=2 @198.51.100.0 (mysql-5.7.22 ndb-7.6.6, Nodegroup: 0, \*)

id=3 @198.51.100.1 (mysql-5.7.22 ndb-7.6.6, Nodegroup: 0)

[ndb\_mgmd(MGM)] 1 node(s)

id=1 @198.51.100.2 (mysql-5.7.22 ndb-7.6.6)

[mysqld(API)] 1 node(s)

id=4 @198.51.100.2 (mysql-5.7.22 ndb-7.6.6)

**Hasil(output):**

Output

Node 2: started (mysql-5.7.22 ndb-7.6.6)

**Langkah berikutnya : Inserting Data ke MySQL Cluster**

**Mengcreate database “clustertest” dengan command:**

Mysql> CREATE DATABASE clustertest;

**Kemudian membuat new database:**

Mysql> USE clustertest;

**Create table yang sederhana dengan membuat “test\_table”**

Mysql> CREATE TABLE test\_table (name VARCHAR(20), value VARCHAR(20)) ENGINE=ndbcluster;

**Melakukan start inserting data dengan menggunakan SQL query:**

Mysql> INSERT INTO test\_table (name,value) VALUES('some\_name','some\_value');

**Kemudian, verifikasi data yang telah di insert, dengan mengselect query**

Mysql> SELECT \* FROM test\_table;

**#Langkah-langkah Instalasi ProxySQL**

**Langkah I :** Download instalasi ProxySQL, Jalankan command pada server Proxy untuk mendownload

$ cd /tmp

$ curl -OL https://github.com/sysown/proxysql/releases/download/v1.4.4/proxysql\_1.4.4-ubuntu16\_amd64.deb

**Langkah II :** Install package yang sudah di download

$ sudo dpkg -i proxysql\_\*

**Untuk menghubungkan Service API dengan ProxySQL maka diperlukan instalasi mysql client**

$ sudo apt-get update

$ sudo apt-get install mysql-client

**Kemudian mengaktifkan layanan ProxySQL**

$ sudo systemctl start proxysql

**Setelah itu di cek apakah sudah berhasil berjalan dengan baik atau belum**

$ sudo systemctl status proxysql

**\* Setting password Admin ProxySQL, secara default awal password nya adalah admin**

$ mysql -u admin -p -h 127.0.0.1 -P 6032 --prompt='ProxySQLAdmin> '

**Kemudian juga mengganti password, ganti pasword sesuai yang diinginkan**

**#Langkah-langkah Konfigurasi Monitoring di Service API**

ProxySQLAdmin > UPDATE global\_variables SET variable\_value='admin:passwordbaru' WHERE variable\_name='admin-admin\_credentials';

ProxySQLAdmin > LOAD ADMIN VARIABLES TO RUNTIME;

ProxySQLAdmin > SAVE ADMIN VARIABLES TO DISK;

**Tahap awal yaitu, Download file sql addition yang sudah disiapkan dengan command pada service API.**

$ curl -OL https://gist.github.com/lefred/77ddbde301c72535381ae7af9f968322/raw/5e40b03333a3c148b78aa348fd2cd5b5dbb36e4d/addition\_to\_sys.sql

**Kemudian, membuat konfigurasi dalam file sql kedalam service**

$ mysql -u root -p < addition\_to\_sys.sql

**Setelah itu, login ke MySQL**

$ mysql -u root -p

**Membuat user dengan nama monitor**

CREATE USER 'monitor'@'%' IDENTIFIED BY 'monitorpassword';

GRANT SELECT on sys.\* to 'monitor'@'%';

FLUSH PRIVILEGES;

**Melakukan step diatas pada semua service API**

**#Langkah-langkah Konfigurasi Monitoring pada ProxySQL**

ProxySQLAdmin > UPDATE global\_variables SET variable\_value='monitor' WHERE variable\_name='mysql-monitor\_username';

ProxySQLAdmin > LOAD MYSQL VARIABLES TO RUNTIME;

ProxySQLAdmin > SAVE MYSQL VARIABLES TO DISK;

**Kemudian, daftarkan service API kedalam ProxySQL**

**Setelah itu di cek kembali, apakah Service API sudah terhubung dengan ProxtSQL**

ProxySQLAdmin > SELECT hostgroup\_id, hostname, status FROM runtime\_mysql\_servers;

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, '192.168.33.12', 3306);

ProxySQLAdmin > INSERT INTO mysql\_servers(hostgroup\_id, hostname, port) VALUES (2, '192.168.33.13', 3306);

ProxySQLAdmin > LOAD MYSQL SERVERS TO RUNTIME;

ProxySQLAdmin > SAVE MYSQL SERVERS TO DISK;

**#Membuat User MySQL pada Service API untuk dapat diakses melalui ProxySQL**

**Tahap Awal nya yaitu login(masuk) pada server service API**

service1 > mysql -u root -p

mysql > CREATE USER 'mysqlcluster'@'%' IDENTIFIED BY 'vagrant';

mysql > GRANT ALL PRIVILEGES on clustertest.\* to 'mysqlcluster'@'%';

mysql > FLUSH PRIVILEGES;

mysql > exit;

**#Membuat User MySQL pada ProxySQL agar dapat diakses melalui aplikasi**

ProxySQLAdmin > INSERT INTO mysql\_users(username, password, default\_hostgroup) VALUES ('mysqlcluster', 'vagrant', 2);

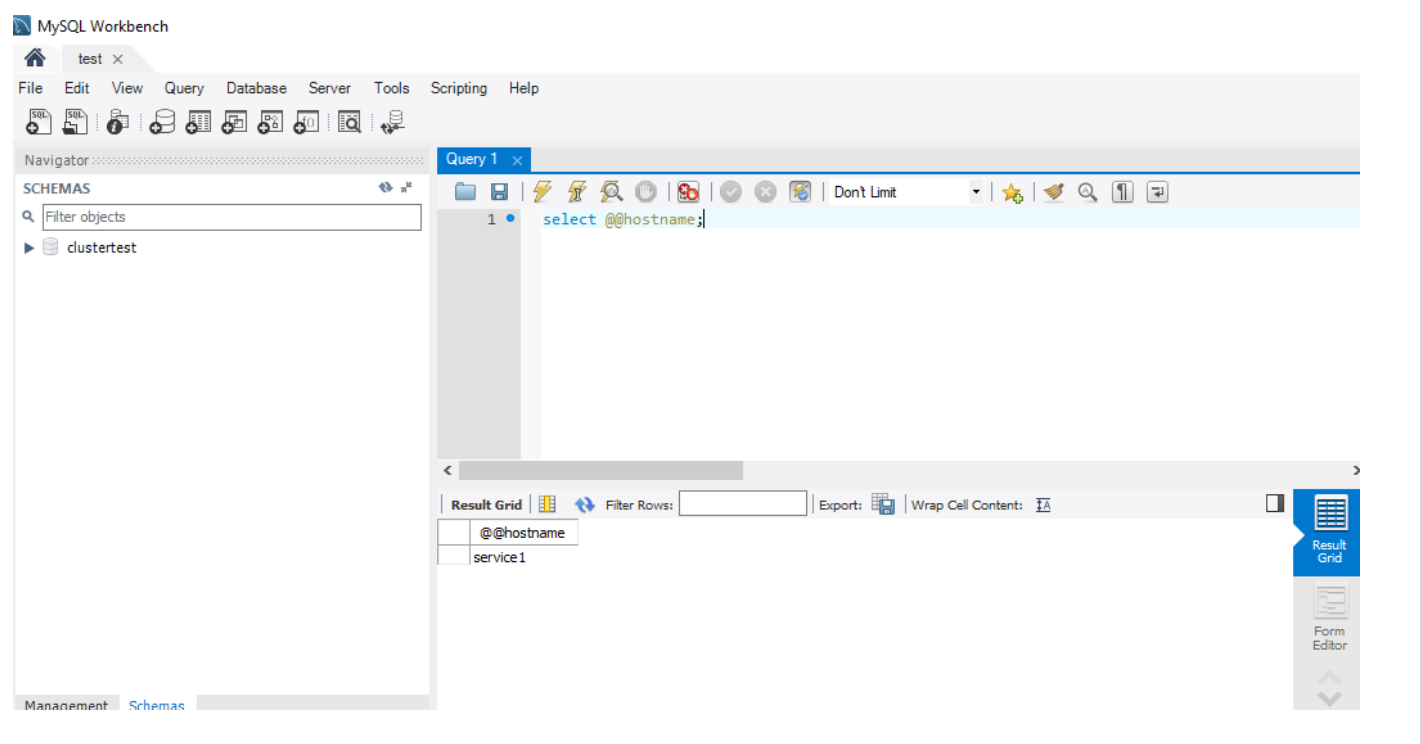
ProxySQLAdmin > LOAD MYSQL USERS TO RUNTIME;

ProxySQLAdmin > SAVE MYSQL USERS TO DISK;

**#Melakukan percobaan dengan menggunakan Aplikasi**

Select @@hostname

**Hasil (output):**

****