**Task 4.** Partitioning/sharding and sort key

Joe Black has heard about MongoDB can be scaled vertically and horizontally. He would like to see the illustration on the horizontal scaling with partitioning/sharding features in MongoDB.

From the past data of the shopping cart, it showed that most customers did not return for shopping (i.e. most customers have one shopping card only). But recently, many customers are return customers and frequently shopping with ShoppingWorld (i.e. most customers have many shopping carts). MongoDB provides the sharding (horizontal scaling) feature which is to distribute data across multiple machines. These features support the deployment of large data sets with higher throughout which may help the performance of the queries.

For showing the features of partitioning to Joe Black, use the MongoDB installed on your local computer to complete this task. The reason is MongoDB Atlas webservice requires to upgrade from the free subscription before you can create replica sets for partitioning data using the partition key.

In general, you are provided with the VMWare Windows Servers with the MongoDB installed to show these features instead. Once when you have started the VMWare server, you need to make sure that the Windows domain controller is able to communicate with the member server. (Note: Instruction are given in the class exercise.)

Run the insert queries which you have saved in the previous tasks **(Part 2 Task 3 a, b, c & d)** to insert the document in the collections for the database. Do not worry about the validator.

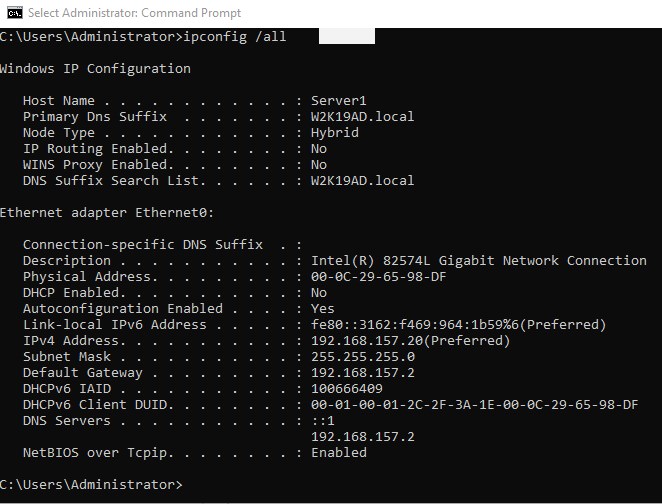
You are required to configure the MongoDB instances in the VM Windows servers (Server1 & Server2) as shown in the diagram below so that the MongoDB router would be able to route transactions into the two shards. Without using 9 separate physical computers, use the following port numbers to simulate the mongod instances.

A diagram of a server

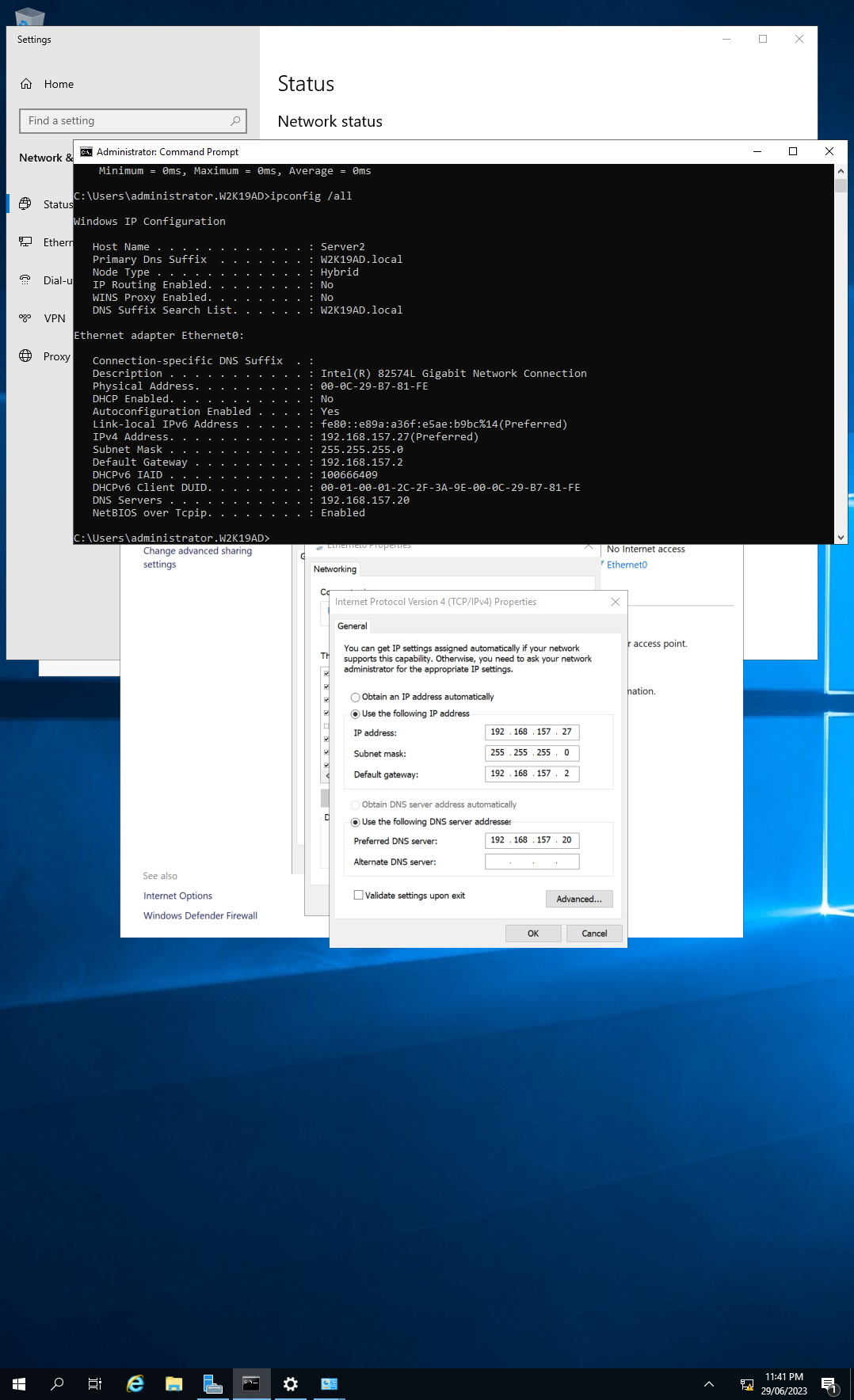
Description automatically generated with low confidence

1. Setup a config server, two shards and a router
2. Given the VMWare Windows servers, MongoDB replica setup script and config files in the MongoDB Assignment (ShoppingWorld) - Part 3 Student Files, configure the VMWare servers so that they are communicating using the fixed IP address as shown in the diagram above. Submit screen shots with the two server IP Addresses. Note: you must use the given IP addresses above since servers in practice must have fixed IP addresses..

Server1:



Server2:



1. Setup the sharded cluster in the following procedures:

* In the VMWare servers, both MongoDB has been installed locally in Server1 and Server2.

Server1:

A screenshot of a computer

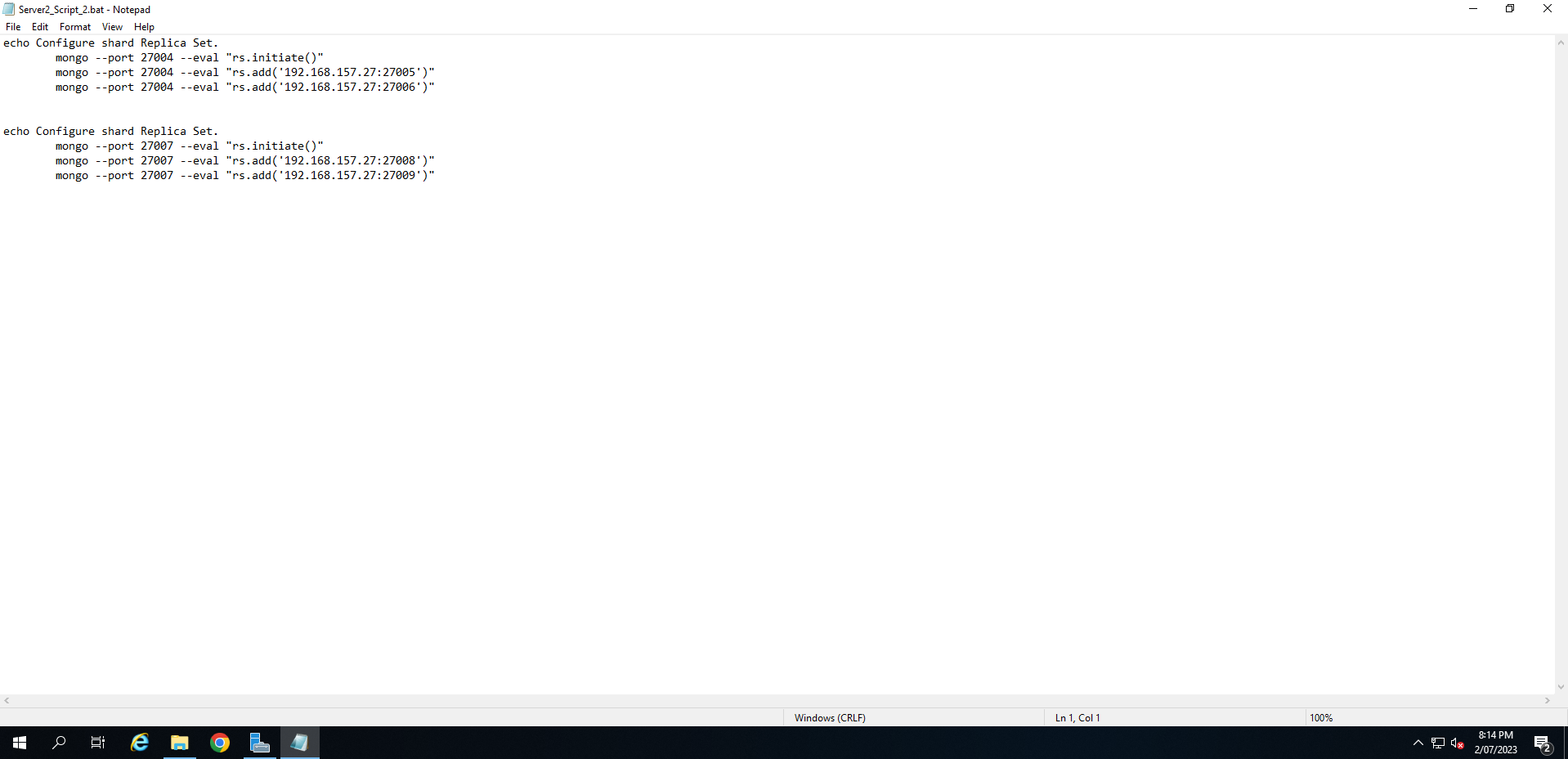
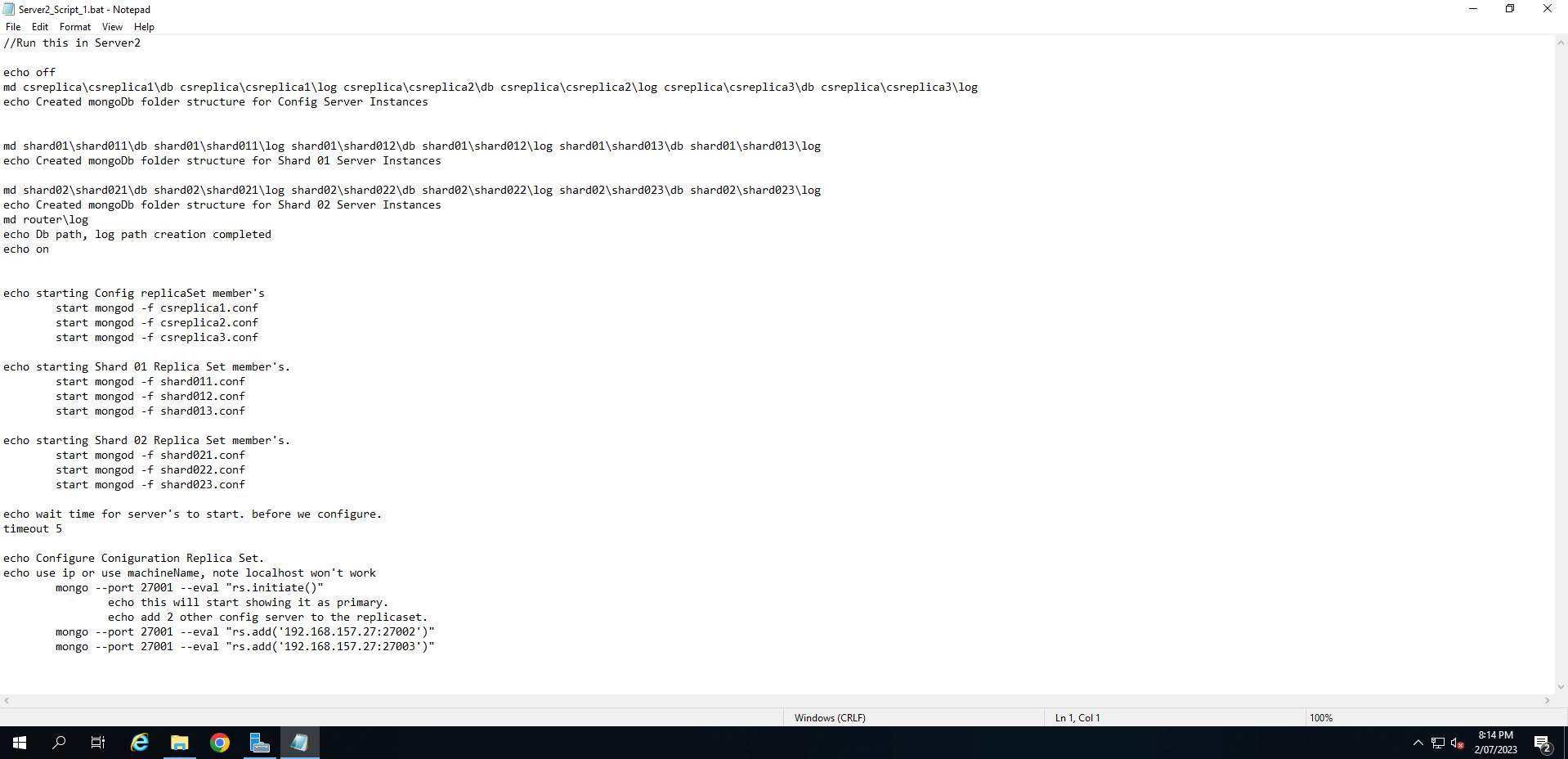
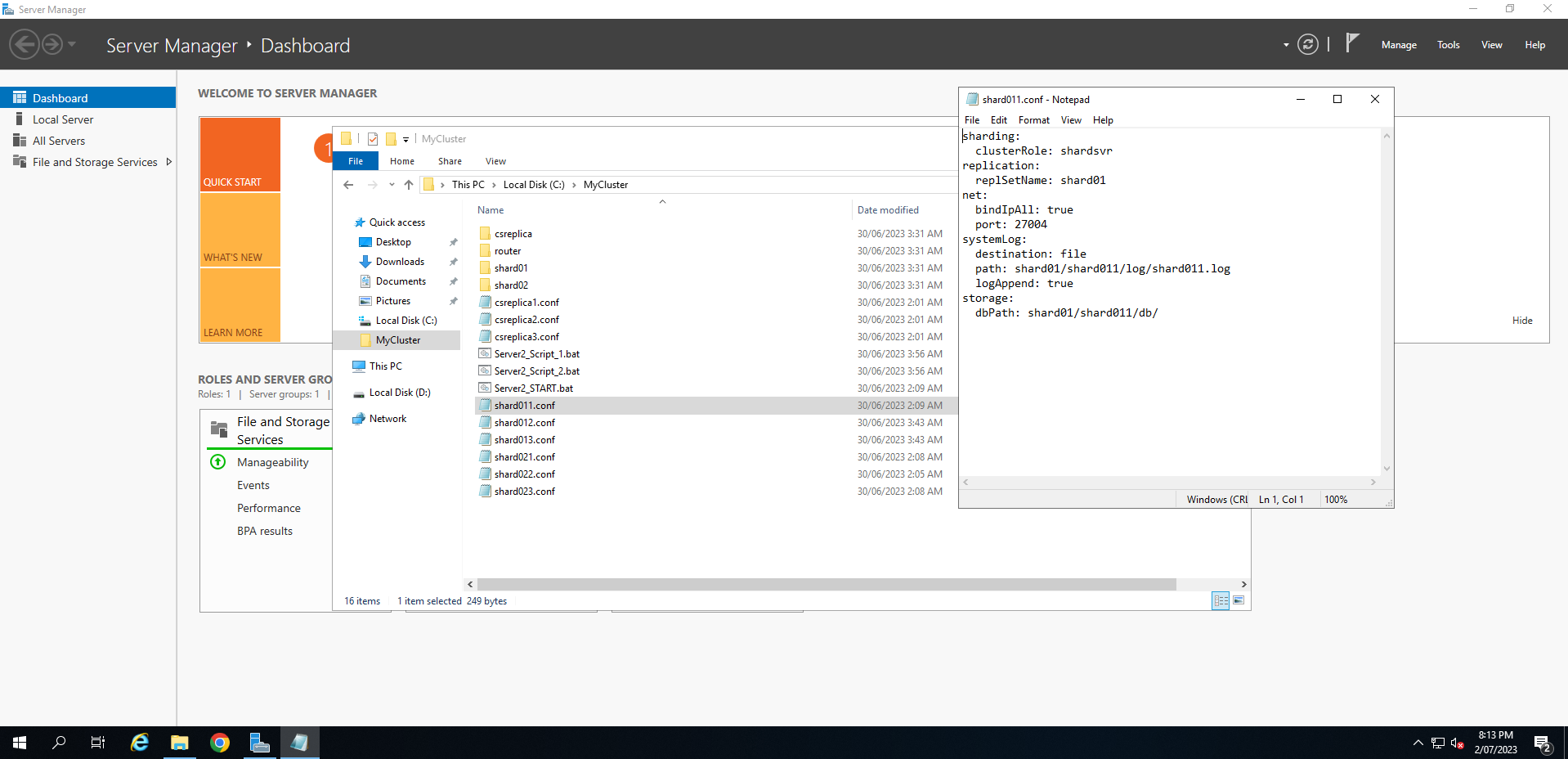
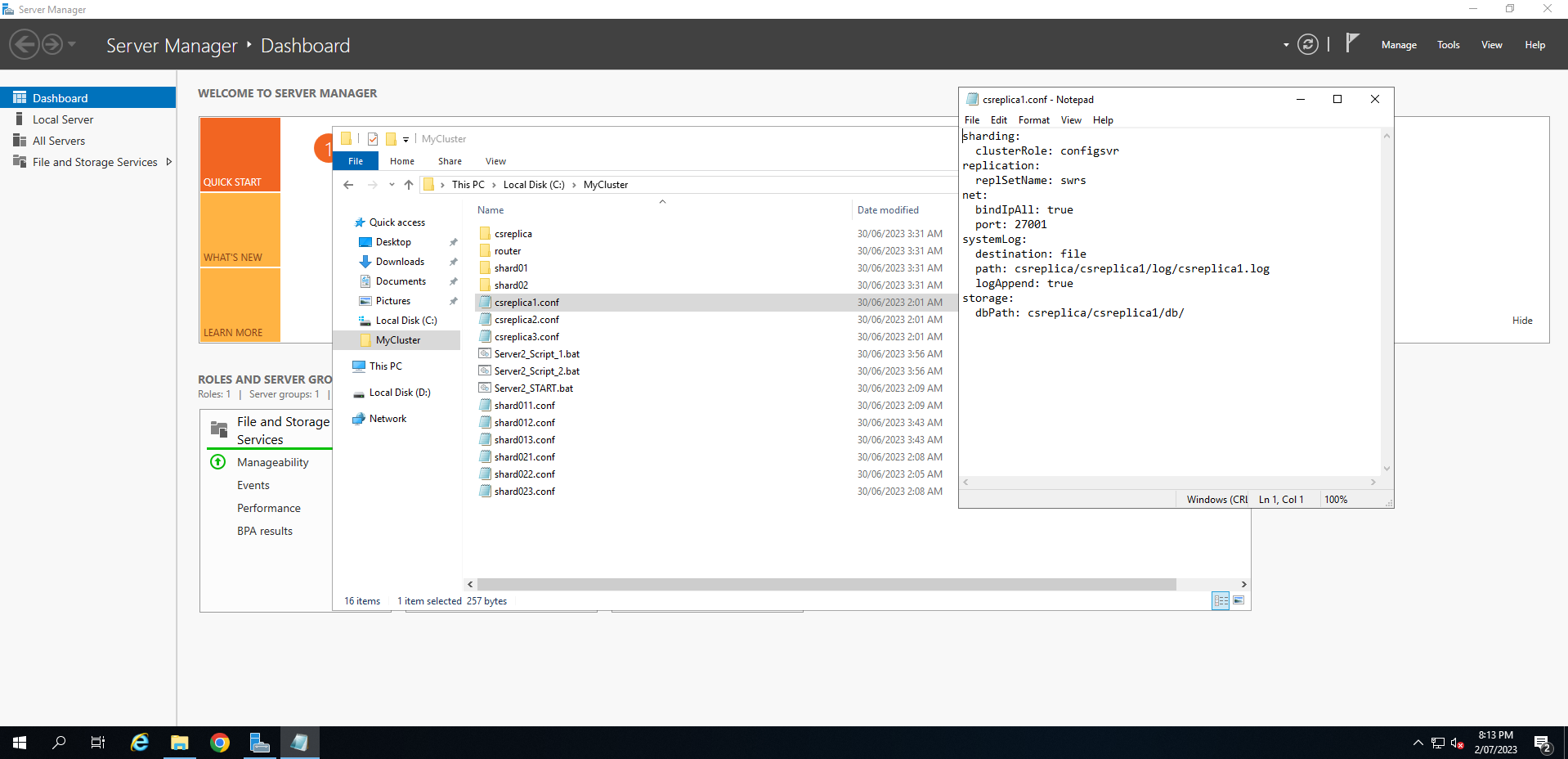
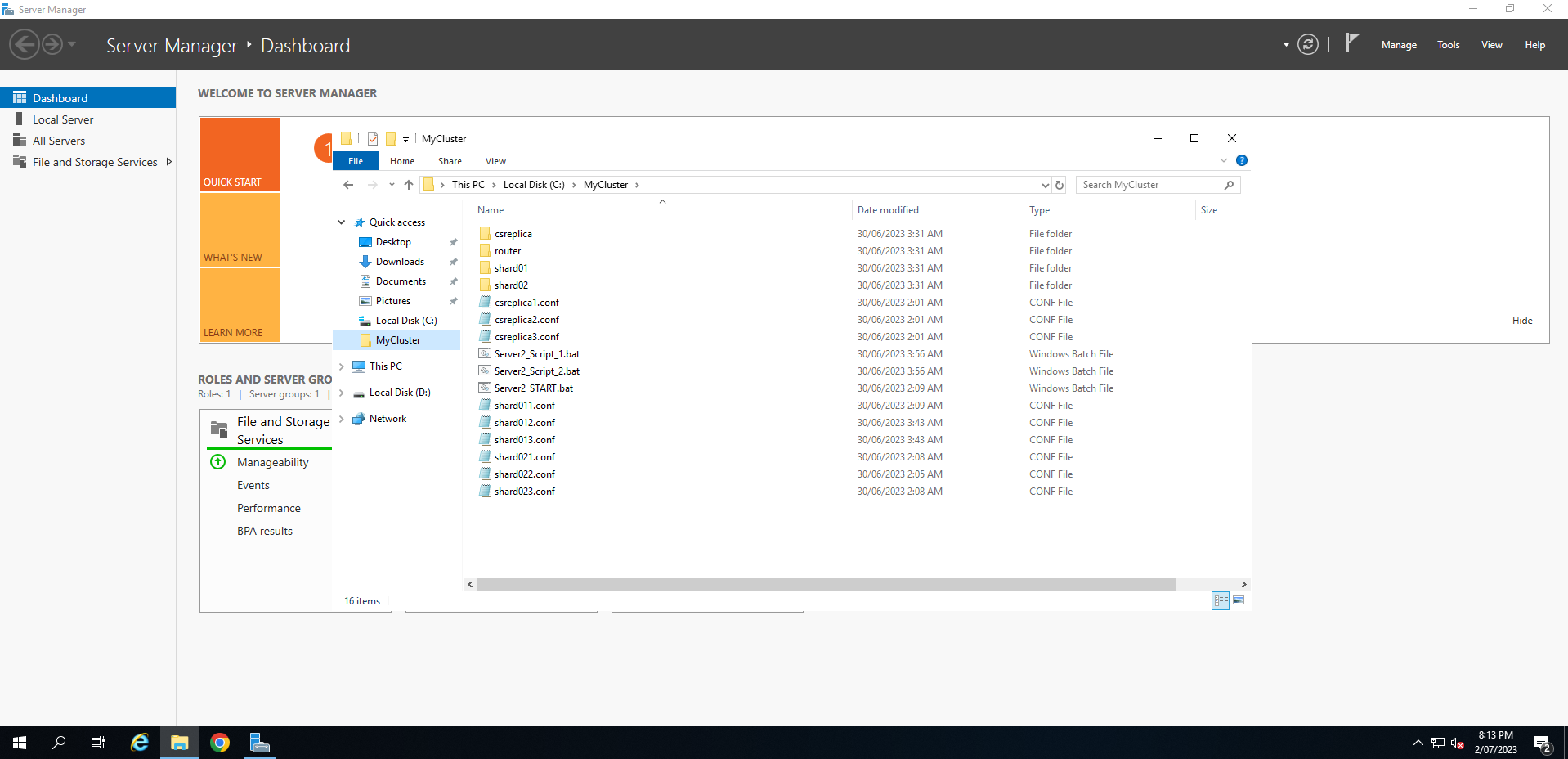
Description automatically generated

Server2:

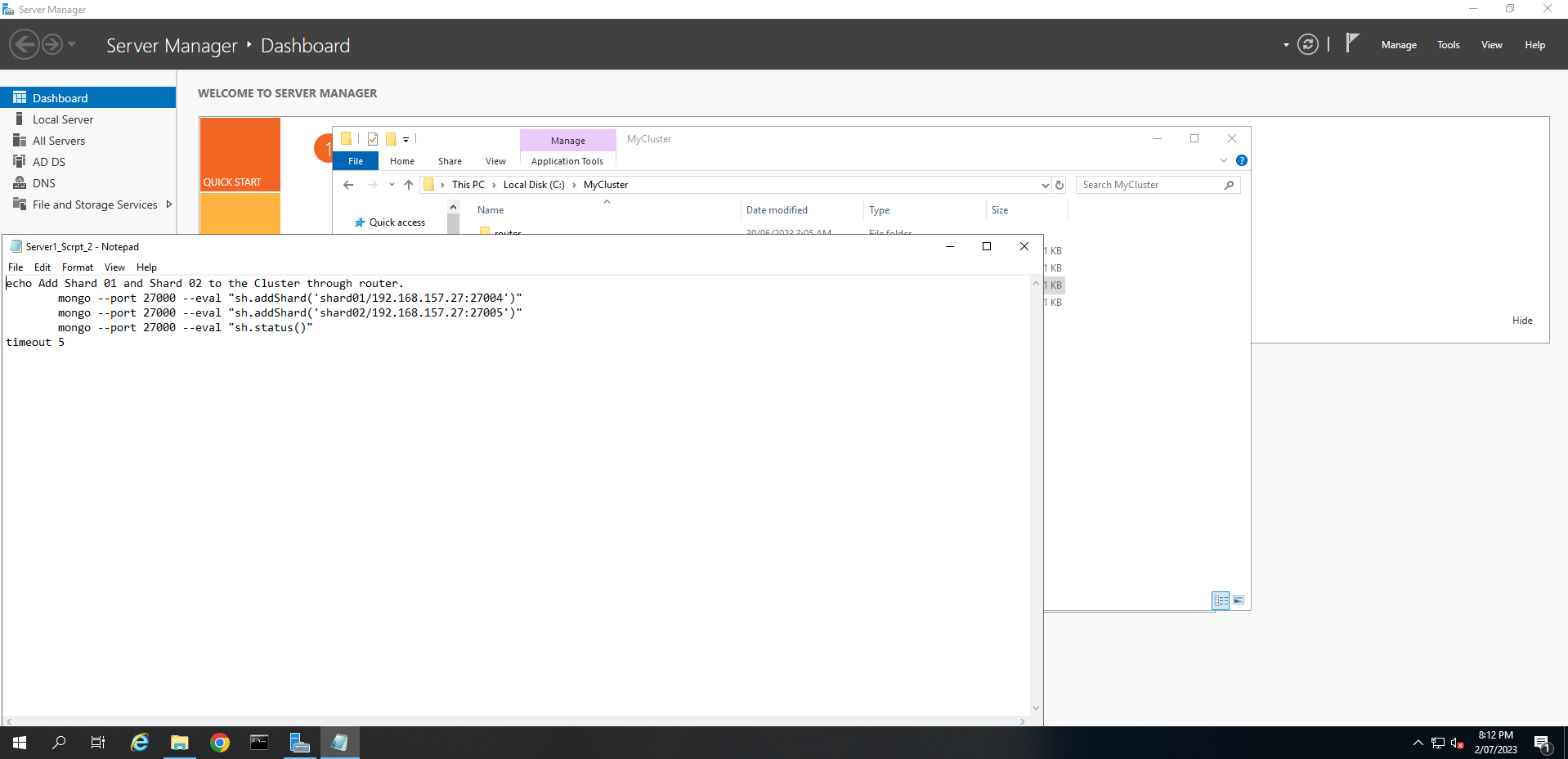
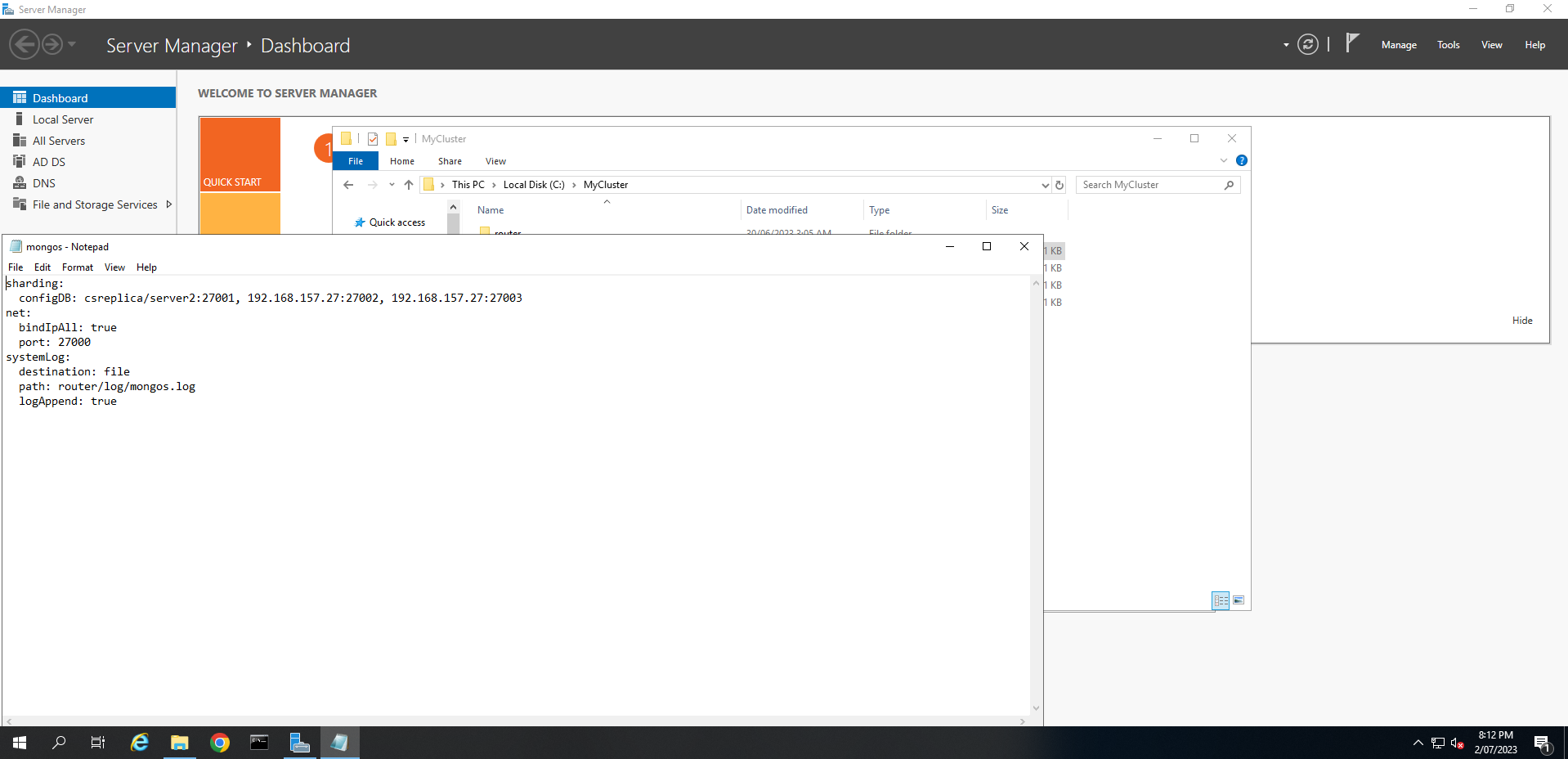
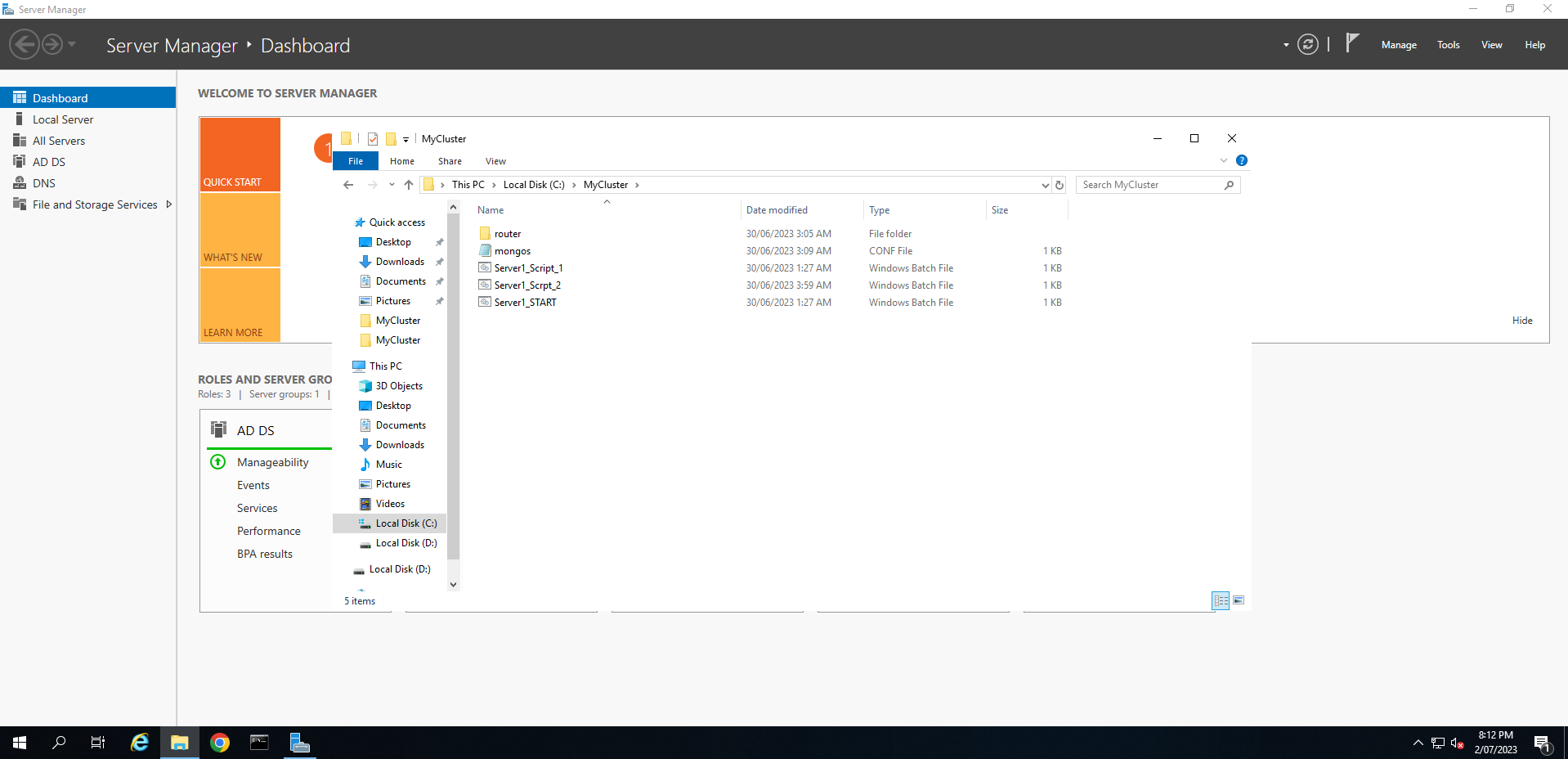
A screenshot of a computer

Description automatically generated

* With the given Server 2 scripts files, copy the MyCluster folder into the C: drive of the VM Sever2.***Note: You must change all folder names used in the scripts, config file names used, IP addresses used to line up with the diagram above.***



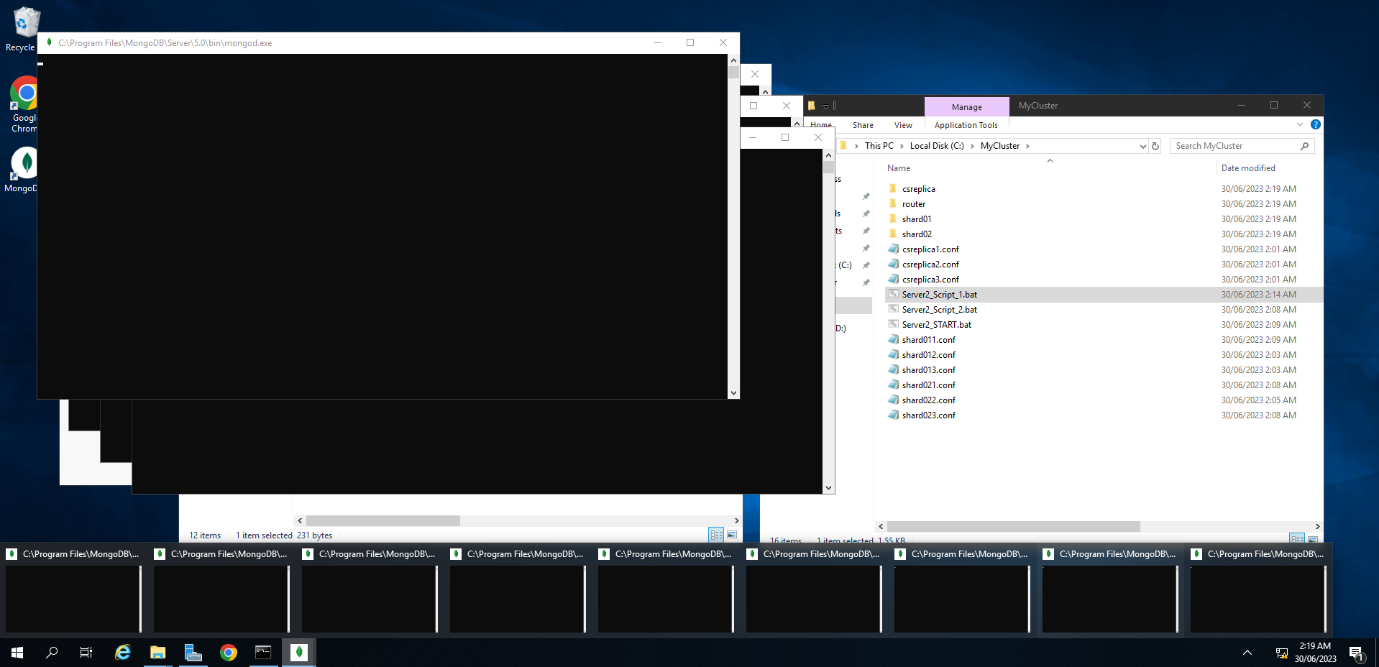
* With the given Server 1 scripts files, copy the MyCluster folder into the C: driver of the VM Server1. ***Note: You must change all folder names used in the scripts, config file names used, IP addresses used to line up with the diagram above****.*



* In Server2, run Server2\_Script1 script in the MyCluster folder.
* In Server2, run Server2\_Script1 script in the MyCluster folder.
* It sets up the folder structures for data and log files for the config server and shard01 and shard02 server instances.
* It also configures the replica set for the config server and shard01 and shard02.
* It initiates the config server replica set with one primary and two secondary members
* In Server1, run the Server1\_Script1 script.
* It configures the mongo router (mongos)
* In Sever 2, run Server2\_Script2 script in the MyCluster folder.
* It initiates the shard01 and shard 02 replica set. Each with one primary and two secondary members.
* Run commands to connect to the router port no. (> mongo --port xxxxxxxx) and show the replica status. Capture screen shots of the replica sets status.
* In Server1, run the Server1\_Script2 & Server1\_Start scripts.
* It connects the mongos client and add the shards to the mongos router.
* Run mongos commands to connect to the router port no (> **mongos** --port xxxxxxxx) and show the sharding status.
* Enable the authentication of the mongoDB in the Server1 and Server2 by creating a super user.
* use admin
* create a super user named “myAdmin” with password “password”
* allocate the role “userAdminAnyDatabase” db: “admin” and also with the role of readWriteAnyDatabase
* Write commands to enable sharding for the databases. (\*\*note; need to connect to the port no of the router first > **mongo --port 27000**).
* Show the shard status. Capture screen shot showing the sharding of the shoppingCartDB is enabled.

For submission:

* Screen shot of the Server2 with the 9 mongod instances started.

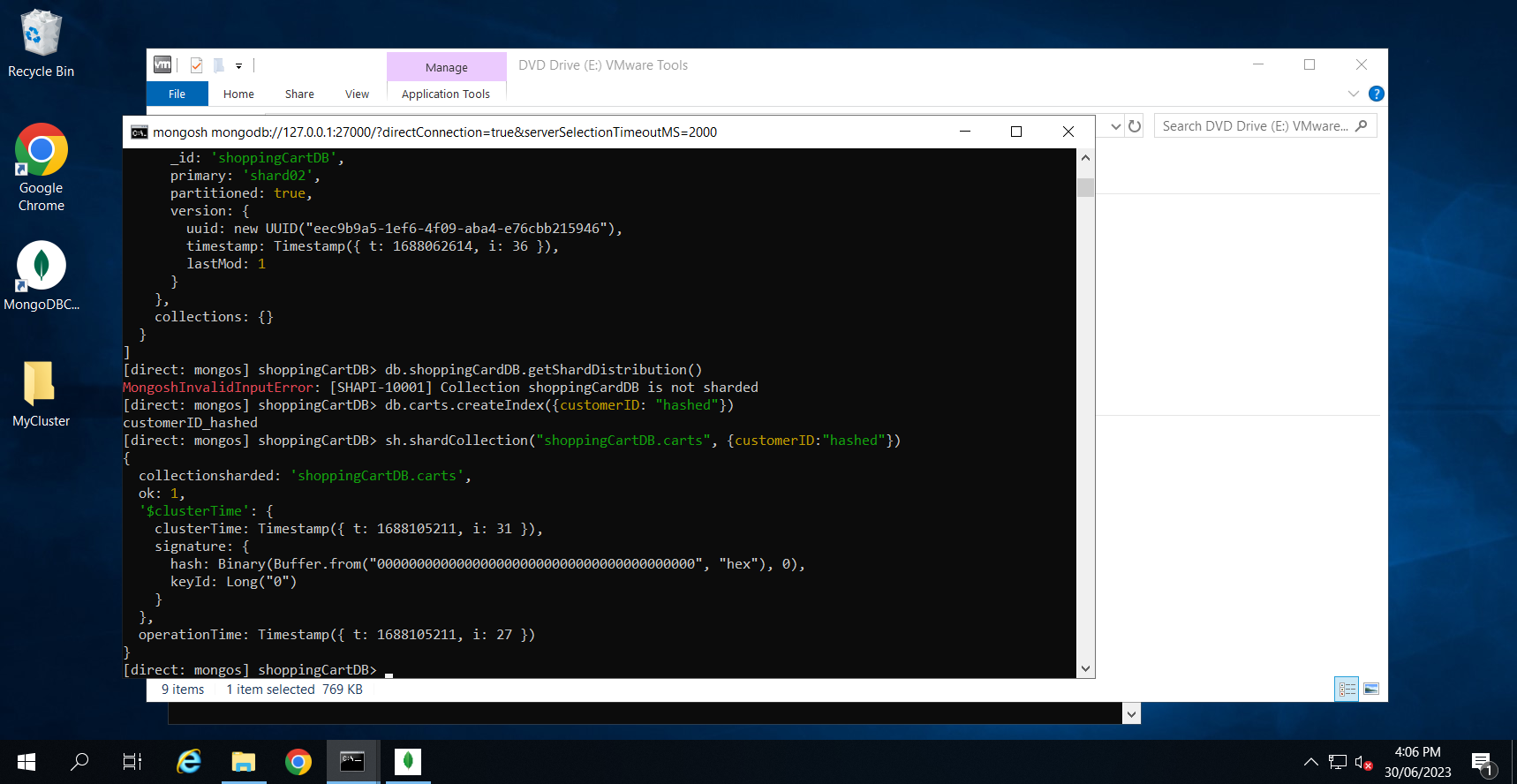


* Screen shots of the Server 2 with one of the replicas set with 1 primary and at least one of the secondary members

A screen shot of a computer

Description automatically generated with medium confidence

* Screen shot of Server2 showing the sharding of the shoppingCartDB is enabled.



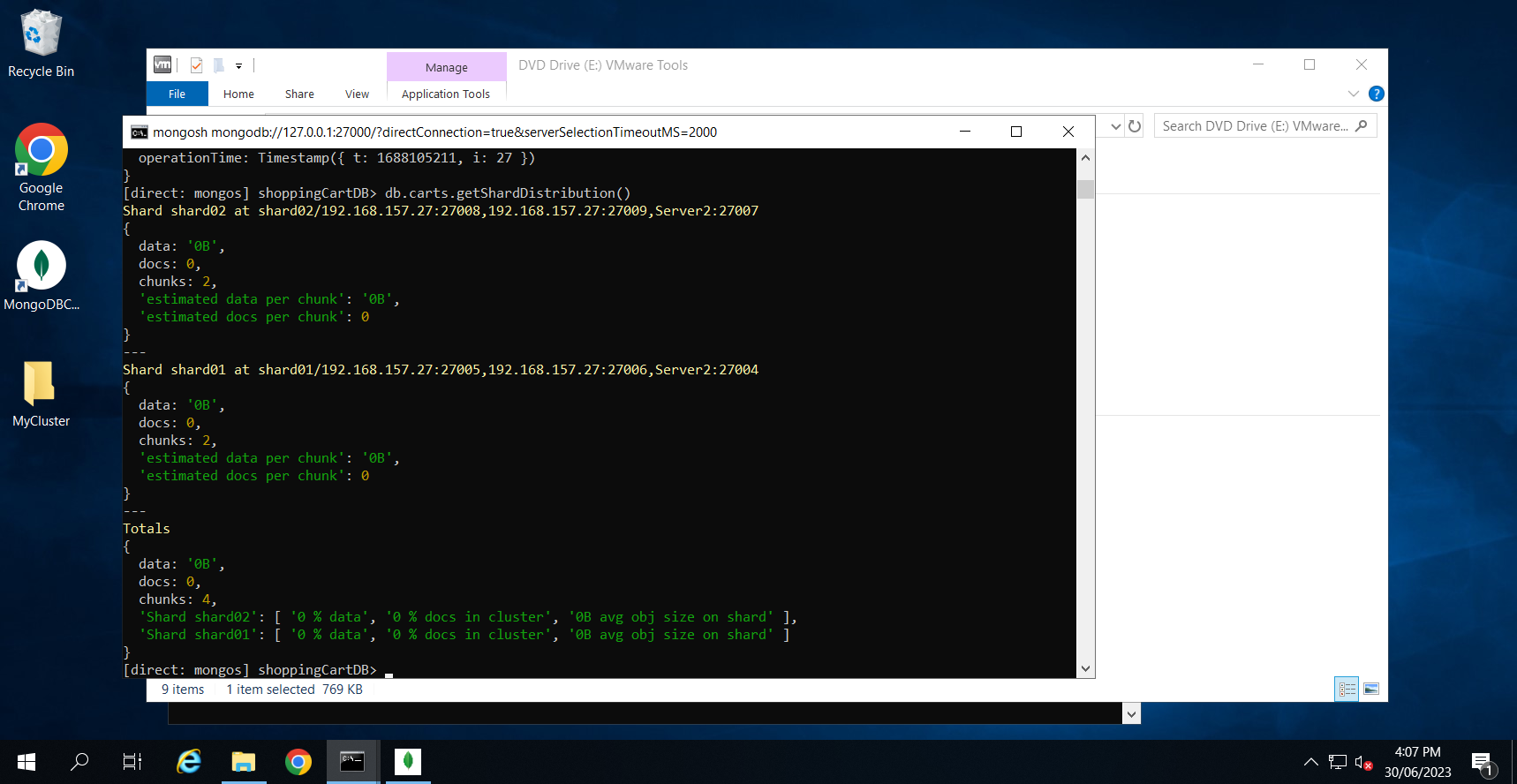
1. ShoppingWorld realised that customers always enquire their “shopping cart” information by providing their customer id when making enquiries. Recommend a shard key for the carts collection that is suitable for sharding. Explain why you choose that field(s) as the shard key. (approx. 20 words)

The chosen shard key for the carts collection is customerID as that is the field that is used the most.

1. In Server 1, run commands to do the following:

* Create a hashed index for the field to be act as the partition key (i.e. sort key) of the collection.
* Define a shard collection with the partition key.
* Capture a screen shot to show the shard collection distribution.
* Enable the balancer for the collections.

For submission, give the commands being used and provide one screen shot showing no distribution (i.e. 0% ) of records between the two shards yet.



1. In Server1, use the MongoDB Compass in your **windows host machine** to connect to the router.

( e.g. mongodb://router ip: router port).

Use a for loop to insert 500 new documents into the carts collection where the cartID and customerID as the following:

i.e. cartID as cart1, cart2, cart3, …….cart500

customerID as customer1, customer2, customer3 …. , customer500

The document structure can be simplified ads the following:

{

cartID: “cartID” + i,

cartdateTime: new Date(),

customerID: “customerID” + i,

cart\_Items: [ { itemID: 2,

qtyOrdered: 2}]}

1. Since there are 500 records inserted through the router, run a command to show the distribution of the carts documents are partitioned between the two shards. Submit a screen shot of the command used and the output with the % distributed between the two shards.

