Dell Wyse Management Suite

Version 2.1 High Availability Guide

June 2020
Rev. A00

Notes, cautions, and warnings

NOTE: A NOTE indicates important information that helps you make better use of your product.

CAUTION: A CAUTION indicates either potential damage to hardware or loss of data and tells you how to avoid the problem.

WARNING: A WARNING indicates a potential for property damage, personal injury, or death.

© 2020 Dell Inc. or its subsidiaries. All rights reserved. Dell, EMC, and other trademarks are trademarks of Dell Inc. or its subsidiaries. Other trademarks may be trademarks of their respective owners.

Contents

Chapter 1: Introduction..... 5
High availability overview.....5

Chapter 2: High availability architecture..... 6
System requirements for high availability.....6

Chapter 3: High availability on Windows Server 2012 R2/2016/2019..... 8
Creating clustered roles.....8

Chapter 4: Achieve high availability on Windows Server 2012 R2/2016/2019..... 11
Add failover cluster feature on Windows Server 2012 R2/2016/2019.....11
Create file share witness.....16
Configure cluster quorum settings.....17
Creating clustered roles.....20

Chapter 5: Achieve high availability for MySQL InnoDB..... 23
High availability with MySQL InnoDB 23
Install MySQL InnoDB database..... 23
Check MySQL InnoDB server instances..... 41
Create a cluster instance for MySQL InnoDB.....42
Add server instance to MySQL InnoDB cluster.....43
Configure MySQL Router.....44
Create database and users on MySQL InnoDB server..... 57

Chapter 6: Achieve high availability on MongoDB.....58
Install MongoDB 58
Create replica servers for MongoDB database..... 59
Create stratus user 59
Create database user 60
Create DBadmin user for MongoDB 60
Edit mongod.cfg file 60
Initiate replication on the servers..... 61

Chapter 7: Achieve high availability for Teradici devices..... 65
Install and configure HAProxy.....65

Chapter 8: Install Wyse Management Suite on Windows Server 2012 R2/2016/2019 67

Chapter 9: Post installation checks 73

Chapter 10: Upgrade Wyse Management Suite version 1.3 to 1.4.....74

Chapter 11: Upgrading from Wyse Management Suite version 2.x/2.1 to Wyse Management Suite version 3.x.....82

Chapter 12: Troubleshooting.....86

1

Introduction

Wyse Management Suite is the next generation management solution and enables you to configure, monitor, manage, and optimize your Dell Wyse thin clients. This helps you to deploy and manage thin clients on a high availability set-up with improved performance.

It offers advanced feature options such as cloud versus on-premises deployment, manage-from-anywhere by using a mobile application, and enhanced security such as BIOS configuration and port lockdown. Other features include device discovery and registration, asset and inventory management, configuration management, operating system and applications deployment, real-time commands, monitoring, alerts, reports, and troubleshooting of endpoints.

Wyse Management Suite version supports high availability and significantly minimizes the system downtime. The solution also protects the system from unplanned downtime and helps you to achieve the required availability to meet the business goals.

This guide describes the solution architecture and explains how to set up, configure, and maintain high availability clusters at the application and database level.

High availability overview

About this task

The high availability solution for Wyse Management Suite version includes the following sections:

Steps

1. Review the high availability requirements—see System requirements to set up high availability.
2. Deploy high availability on Microsoft Windows Server 2012 R2/2016/2019—see Deploy high availability on Windows Server 2012 R2/2016/2019.
3. Deploy high availability on MySQL InnoDB servers—see Deploy high availability on MySQL InnoDB.
4. Deploy high availability on MongoDB—see Deploy high availability on MongoDB .
5. Configure high availability proxy (for Teradici devices)—see Deploy high availability for Teradici servers.
6. Install Wyse Management version on Windows Server 2012 R2/2016/2019—see Install Wyse Management Suite on Windows Server 2012 R2/2016/2019.
7. Review the post installation checks—see Post installation checks.
8. Troubleshooting issues with workaround—see Troubleshooting.

2

High availability architecture

The Dell Wyse Management Suite architecture consists of Windows Server 2012 R2/2016/2019 with failover cluster enabled. The Windows cluster contains a main computer that supports other applications and ensures minimum downtime by harnessing the redundant. This is used for application failover for Tomcat, Memcache, MQTT services. MongoDB database cluster helps in the event of primary database failure the secondary database will take over. MySQL InnoDB database cluster has an inbuilt database clustering mechanism and secondary database will take over in case of primary read write database fail. Linux Server with HA Proxy is a load balancer and high availability server for EMSDK (Teradici) server. Local repository is created as part of the shared path that contains the applications, images, packages, and will not be part of the cluster set up.

NOTE: The high availability system requirements may change depending on the infrastructure at your work site.

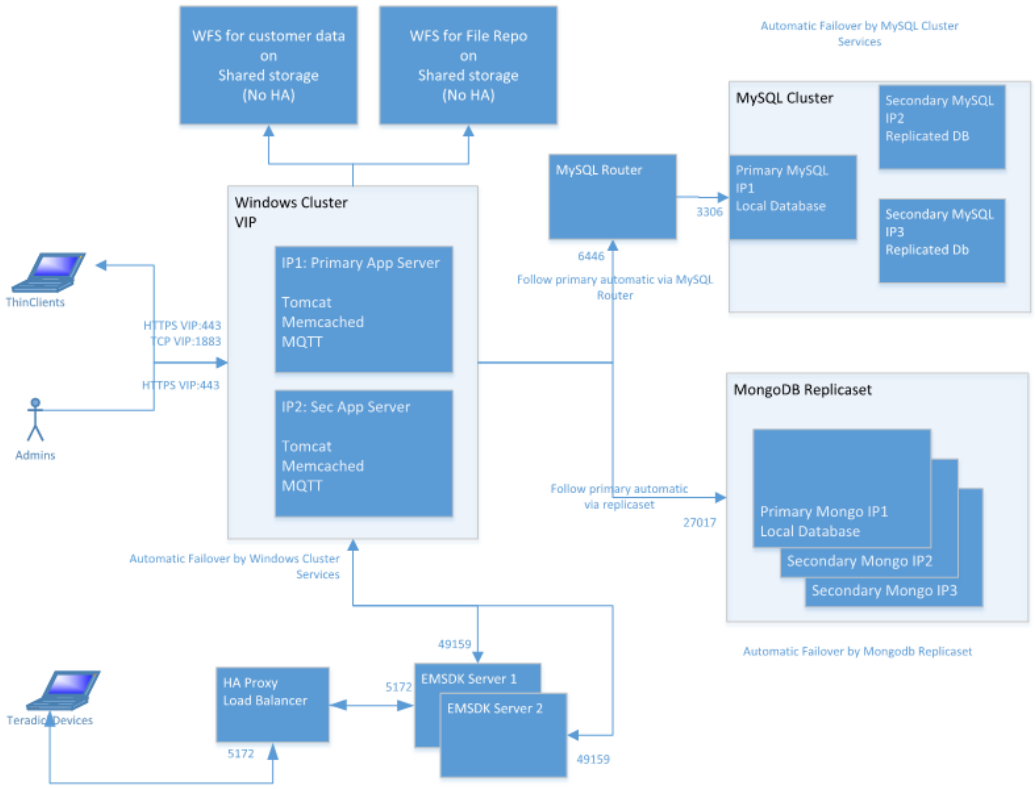

Figure 1. High availability architecture

System requirements for high availability

The table lists the minimum hardware and software requirement and supports up to 10,000 devices. Each instance of EMSDK can support a maximum of 5,000 devices. The deployment can be on individual servers or on a hypervisor environment, depending on the requirement.

The hardware and software requirements to set up high availability for Wyse Management Suite are:

Table 1. System requirements

Product	Port	Protocol	Description
Microsoft Windows Server 2012 R2/2016/2019	• Network communication ports: ○ UDP:3343 ○ TCP:3342 ○ UDP:137	• Minimum disk space—40 GB • Minimum number of systems—2 • Minimum memory (RAM)—8 GB • Minimum CPU requirements—4	Server where Wyse Management Suite is hosted. Supports English, French, Italian, German, and Spanish languages.
MySQL Cluster	• Network communication port—TCP:3306	• Minimum disk space—40 GB • Minimum number of systems—3 • Minimum memory (RAM)—8 GB • Minimum CPU requirements—4	Server in the high availability setup.
MySQL Router	• Network communication ports: ○ 6446 ○ 6447	• Minimum disk space—40 GB • Minimum number of systems—2 • Minimum memory (RAM)—8 GB • Minimum CPU requirements—4	Establishes communication in the high availability setup.
MongoDB	• Network communication port—TCP: 27017	• Minimum disk space—40 GB • Minimum number of systems—3 • Minimum memory (RAM)—8 GB • Minimum CPU requirements—4	Database
EMSDK	• Network communication port—TCP: 5172 • TCP 49159	• Minimum disk space—40 GB • Minimum number of systems—2 • Minimum memory (RAM)—8 GB • Minimum CPU requirements—4	Enterprise SDK server
HAProxy	• Network communication port—TCP: 5172	• Minimum disk space—40 GB • Minimum number of systems—1 • Minimum memory (RAM)—4 GB • Minimum CPU requirements—2	Load balancer in the high availability setup. Ubuntu version 12.04 and later.

NOTE: Ensure that you add the TCP ports 443, 8080 and 1883 to the firewall exception list during high availability setup.

3

High availability on Windows Server 2012 R2/2016/2019

A failover cluster is a group of independent systems that increases the availability and scalability of clustered roles. This feature supports multiple workloads running clusters on hardware or on virtual machines.

A failover cluster is a group of systems that are independent and increases the availability and scalability of clustered roles. The clustered servers are the nodes that are connected to one another as a network. If one or more of the cluster nodes fail, other nodes become active and prevents failover of the systems in the network. The clustered roles that are created during cluster setup monitor to verify that the systems are working in the clustered network. If any of the systems are not working, they are restarted or moved to another node.

The failover cluster network for high availability on Windows Server 2012 R2/2016/2019 contains two nodes, Node 1 and Node 2 that are configured on systems running Windows Server 2012 R2/2016/2019. In the failover cluster network, if Node 1 that is working as the primary node fails, Node 2 starts working automatically as the primary node. After Node 1 becomes active, it automatically becomes the secondary node. The systems have a shared storage space that is connected in a network.

NOTE: The IP address of the systems in the image is an example and varies for each system at your work place.

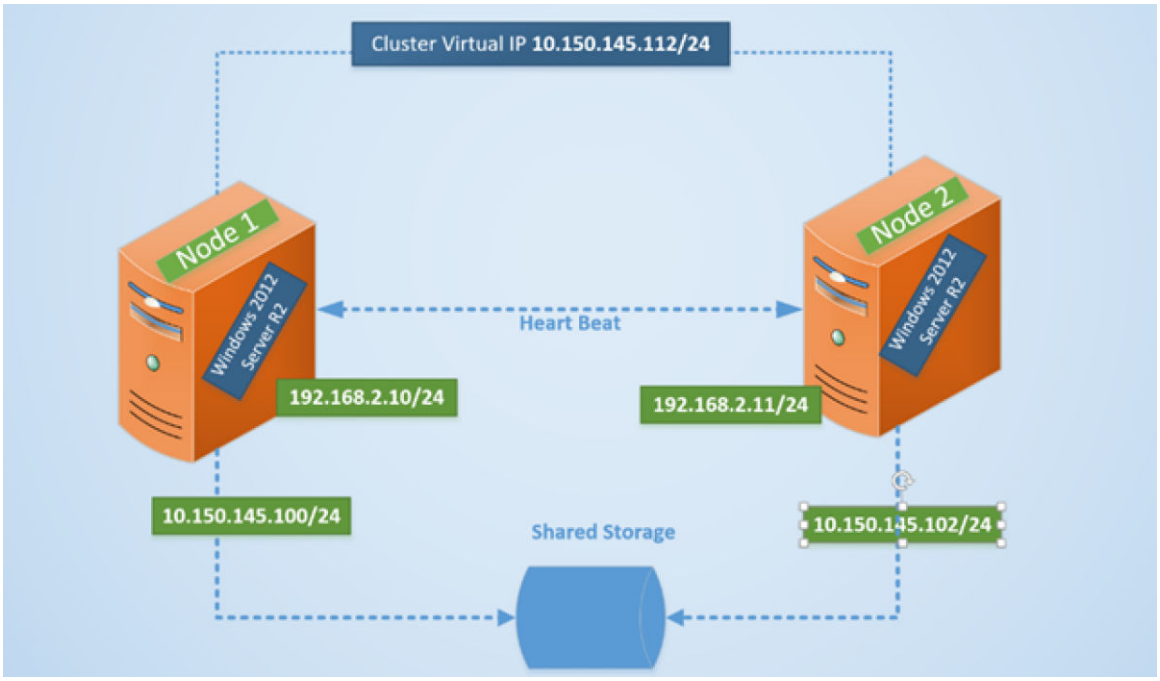

Figure 2. Failover cluster setup

Creating clustered roles

Prerequisites

After you create the failover cluster, you can create clustered roles to host cluster workloads. Ensure that Wyse Management Suite is installed on the servers and point to the remote database before you create clustered roles.

Steps

1. In Windows Server 2012, right-click the **Start** menu and then select **Server Manager** to launch the Server Manager dashboard
2. Click **Failover Cluster Manager** to launch the cluster manager.
3. Right-click **Roles** and then select **Configure Role** to display the **High Availability Wizard** screen.

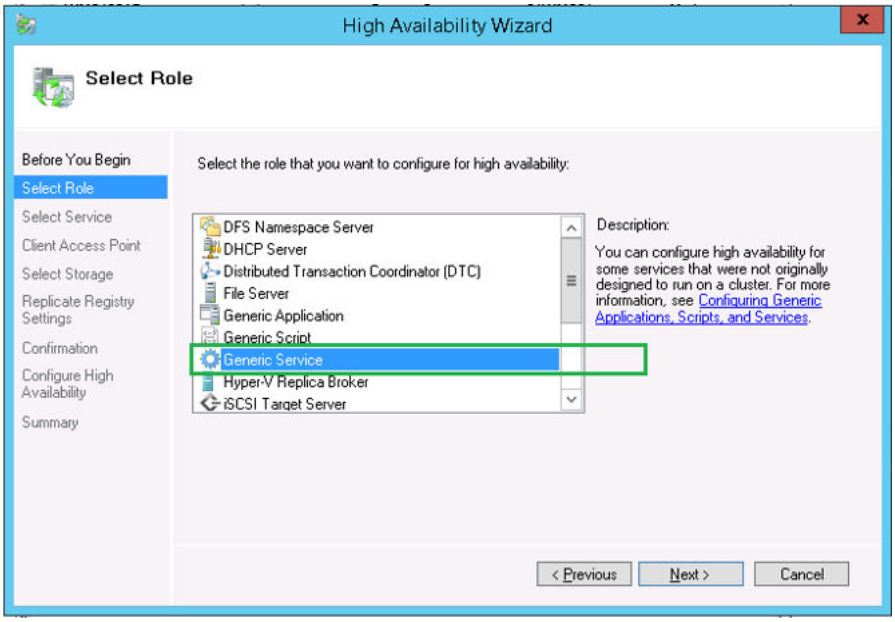

Figure 3. High availability wizard

4. Select **Generic Service** and then click **Next** to view the **Select Service** screen.

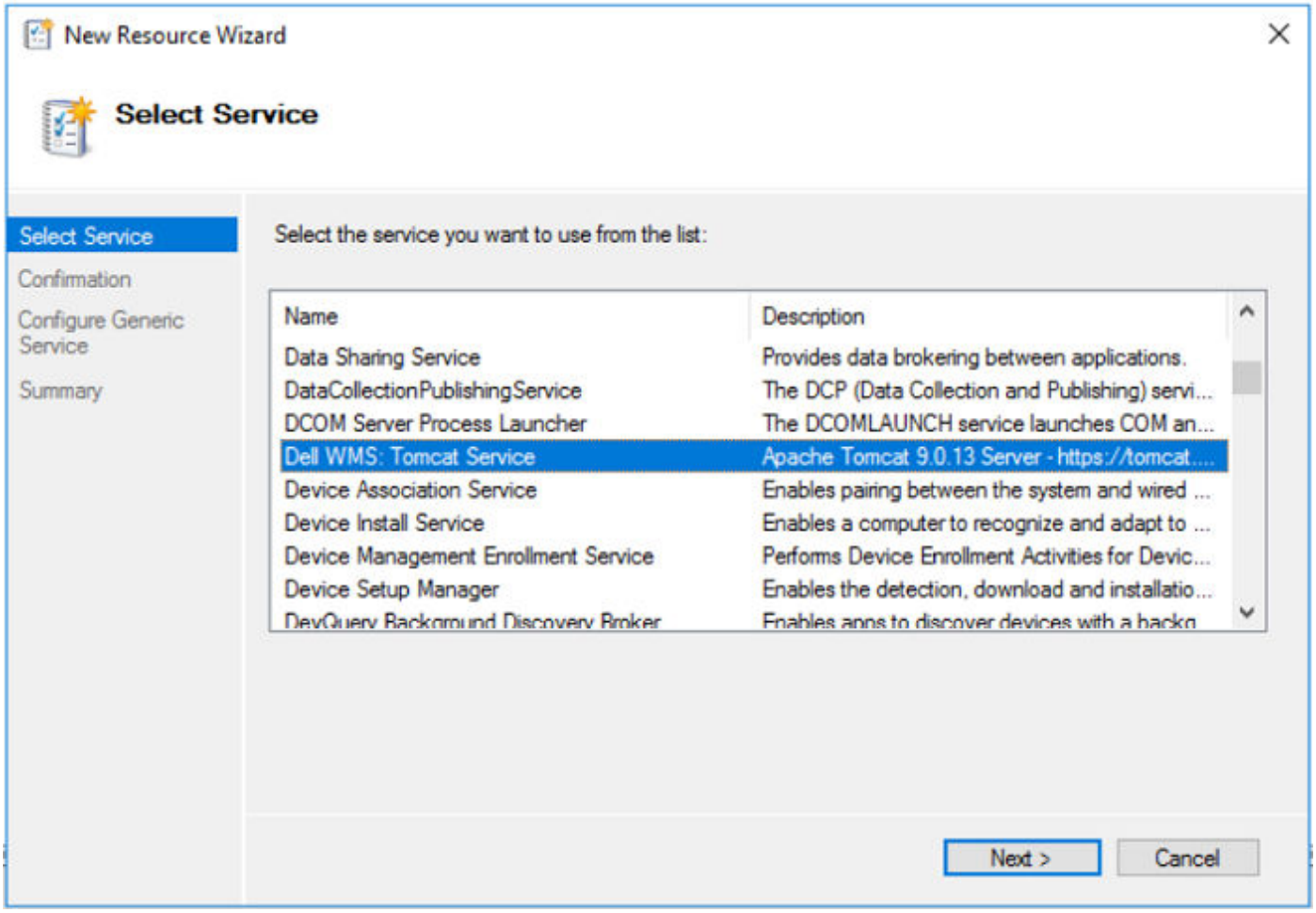

Figure 4. Select service

5. Select **Dell WMS: Tomcat Service** and then click **Next**.

NOTE: You can add the Wyse Management Suite related services to the cluster only after you install Wyse Management Suite.

The **High Availability Wizard** screen is displayed where you need to create the client access point and establish connectivity between the Windows server 2012 and Wyse Management Suite.

6. Type a network name in the **Name** field and then click **Next**. The **Confirmation** screen is displayed with the network name and IP address details of the server.

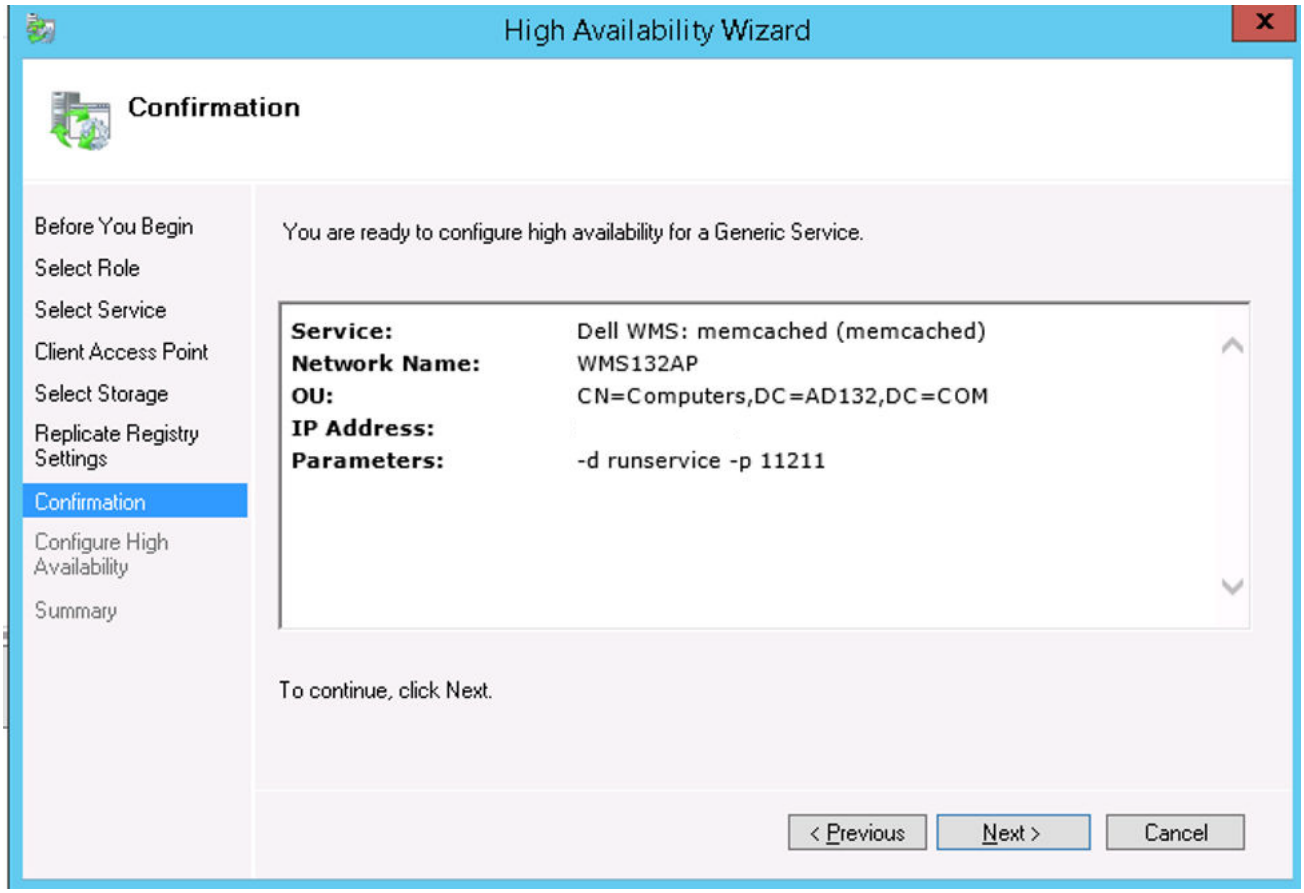

Figure 5. Confirmation

7. Click **Next** to complete the process.
8. To add other Wyse Management Suite services as part of the cluster, launch **Failover Cluster Manager**, and then go to **ActionsRoles** to display the network name that you have created.
9. Click on the network name, and go to **Add ResourceGeneric Service**.
10. Select the following services from the **New Resource Wizard** screen that needs to be added as part of the cluster:
    a. Dell WMS: MQTT Broker
    b. Dell WMS: memcached
11. Click **Next** to complete the task.
    The Wyse Management Suite services that have been added as part of the cluster are displayed with the status **Running**.

4

Achieve high availability on Windows Server 2012 R2/2016/2019

About this task

The following are the steps to achieve high availability on Windows Server 2012/2016/2019:

1. Add failover cluster feature on Windows Server 2012 R2/2016/2019—see Adding failover cluster feature on Windows Server 2012 R2/2016/2019.
2. Create file share witness—see Create file share witness.
3. Configure cluster Quorum—see Configure cluster Quorum.
4. Create clustered roles—see Create cluster roles.

Add failover cluster feature on Windows Server 2012 R2/2016/2019

About this task

To add the failover clustering feature on the Windows server 2012/2016/2019, do the following:

Steps

1. In Microsoft Windows Server 2012 R2/2016/2019, click **Start** to open the **Start** screen and then click **Server Manager** to launch the **Server Manager** dashboard.

   NOTE: Server Manager is a management console in Windows Server 2012 R2/2016/2019 that enables you to add server roles/features, manage, and deploy servers.

2. Click **Add roles and features** and select an option to configure the server based on your requirement from the **Add Roles and Feature Wizard** screen.

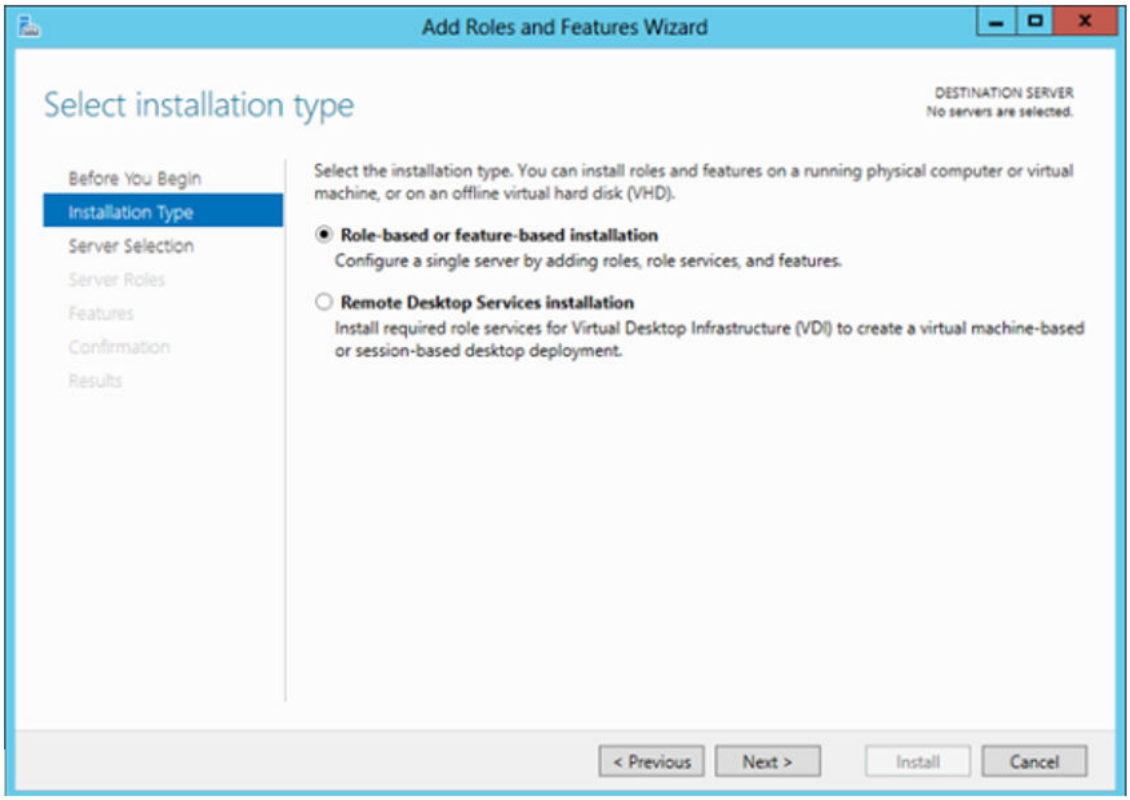

Figure 6. Role based selection

3. Click **Installation Type** and select **Role-based or Feature-based installation** and then click **Next** to view the list of servers in the **Select destination server** screen.

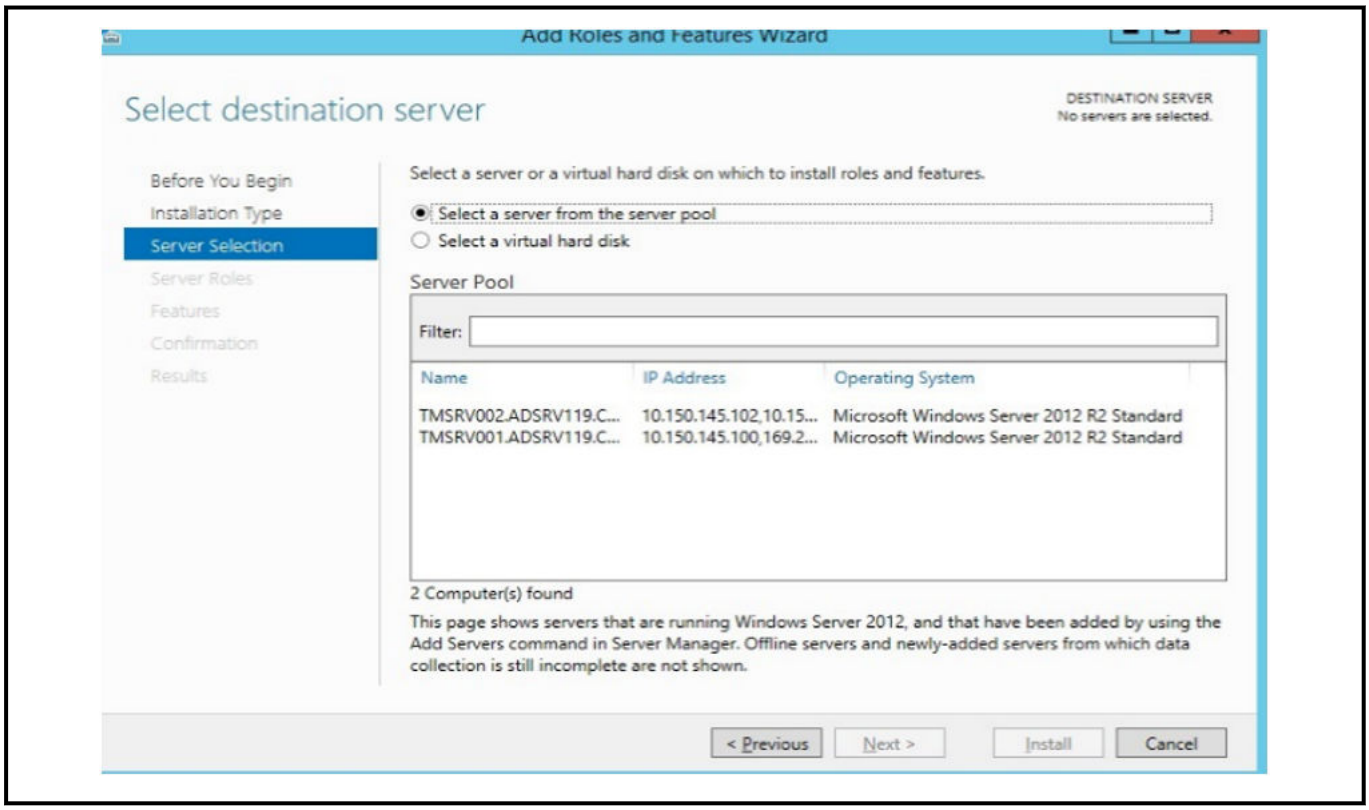

Figure 7. Select server destination

4. Select the server where you want to enable the failover cluster feature and then click **Next**.
5. Select **Failover Clustering** on the **Features** screen, and then click **Next**. After you enable the failover cluster on the servers, open the **Failover Cluster Manager** on the server at Node 1.
6. Click **Yes** to confirm installation, and enable the failover cluster feature on the selected server.
7. In the **Failover Cluster Manager** screen, click **Validate Configuration** to view the **Validate a Configuration Wizard** add the required servers or nodes to cluster.

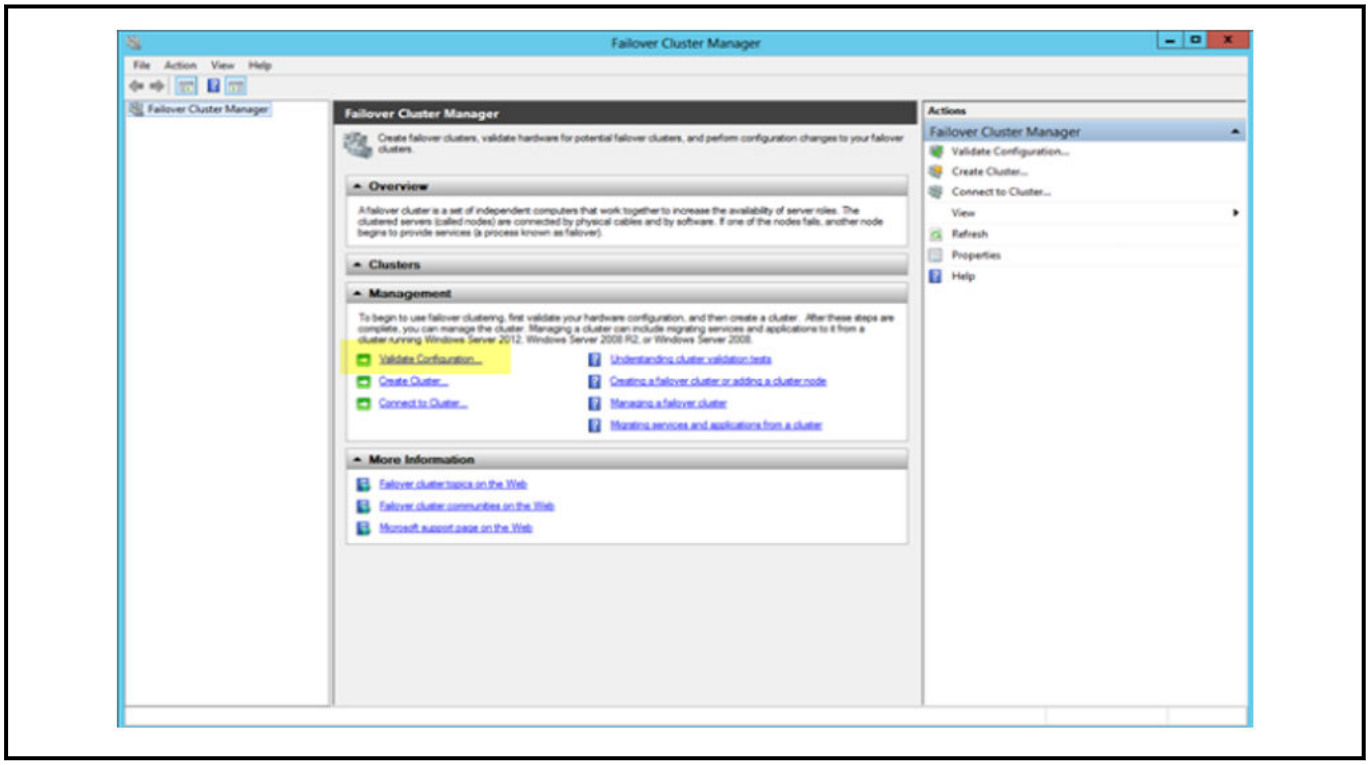

Figure 8. Failover cluster manager

8. Click **Select servers or cluster** and then click **Browse** to configure the servers.
9. Click**Next** and select **Run all tests** from the **Testing Options** screen.

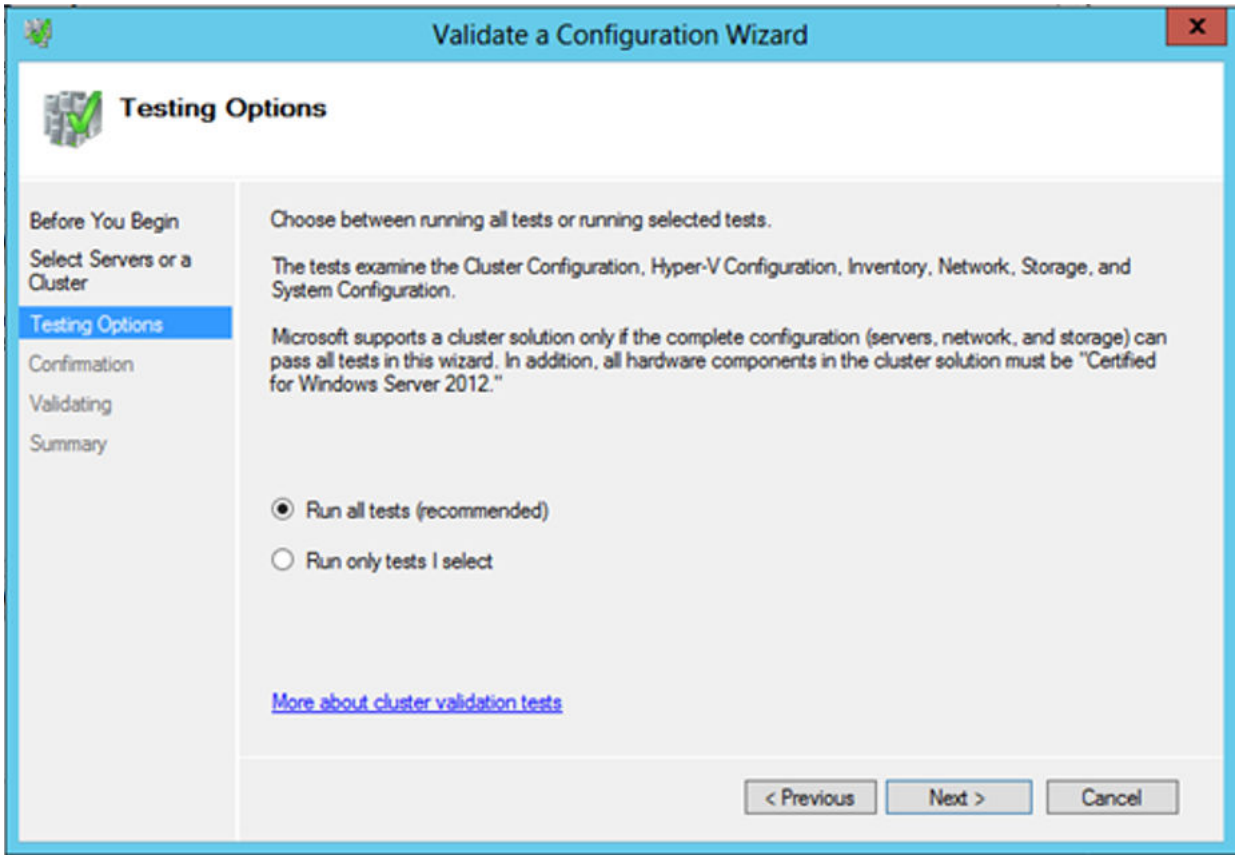

Figure 9. Testing options

10. Click **Next**. The **Confirmation** screen is displayed with the list of selected servers.

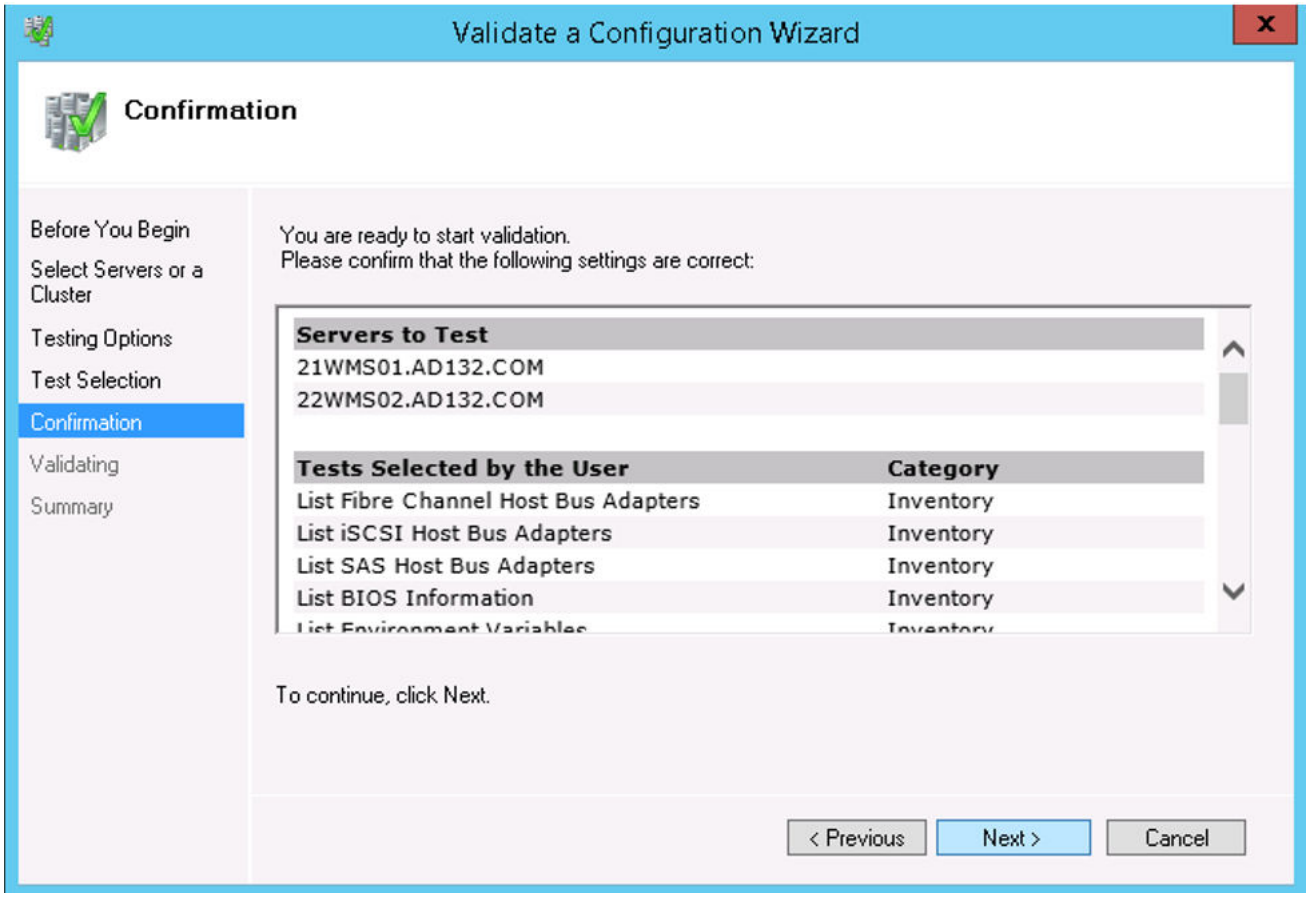

Figure 10. Confirmation

11. Click **Next**. The **Summary** screen is displayed with the failover cluster validation report.

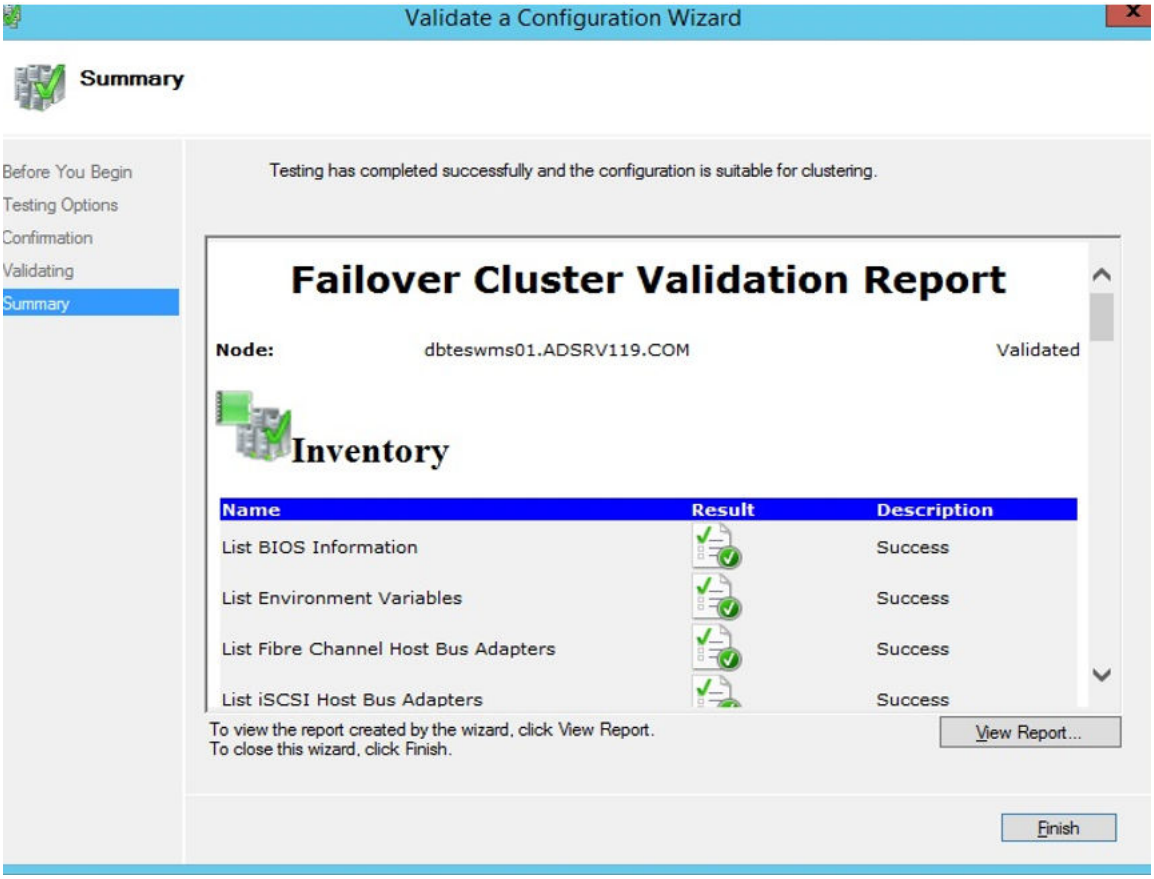

Figure 11. Test summary details

12. Click **View Report** to check the report. If the status is **Passed,** you can proceed with the next step. If the status is **Failed**, you must fix the errors before you proceed with the next step.

    NOTE: The Create Cluster Wizard screen is displayed if there are no validation errors.

13. Click **Next** and type a name for the cluster in the **Cluster Name** field and then select the IP address of the system.
14. Click **Next** and the **Confirmation** screen is displayed.
15. Click **Next** to create the cluster on all the selected clustered nodes and then click **View Report** to view the warning messages.
16. Click **Finish** to create the failover cluster.

Create file share witness

A file share witness is a basic file share that the cluster computer has read/write access. The file share must be on a separate Windows Server 2012 in the same domain where the cluster resides.

About this task

To create a file share witness, do the following:

Steps

1. In Microsoft Windows Server 2012, Right-click the **Start** Menu and then select **Server Manager** to launch the Server Manager dashboard
2. Click the **Server Manager** icon to access the server manager.
3. Go to **Files and Storage ServicesShares** and then click **Tasks**.
4. Click **New Share**. The **New Share Wizard** is displayed.
5. Click **Select Profile** to create a file share and then click **Next**.
6. On the **Share location** screen, select the server and share location for the file share and then click **Next**.
7. On the **Share Name** screen, type a name in the **Share name** field and then click **Next** until the **Confirmation** screen is displayed.

8. Click **Create** to create the file share and the **View results** screen is displayed with the status as **Completed** which indicates that the file share witness is created without any errors.
9. Click **Close** to exit.

Configure cluster quorum settings

The cluster configuration database, also called the quorum, contains details as to which server should be active at any given time in a cluster set-up.

About this task

To configure the cluster quorum settings, do the following:

Steps

1. In Microsoft Windows Server 2012, click **Start** to open the **Start** screen and then click **Server Manager** to launch the Server Manager dashboard.
2. Click the **Server Manager** icon to access the server manager and then click **Failover Cluster Manager** to launch the cluster manager.
3. Right-click the cluster node, and go to **More ActionsConfigure Cluster Quorum Settings** to display the **Configure Cluster Quorum Wizard** screen.
4. Click **Next**. Select **Select the quorum witness** from the **Select Quorum Configuration Option** screen.

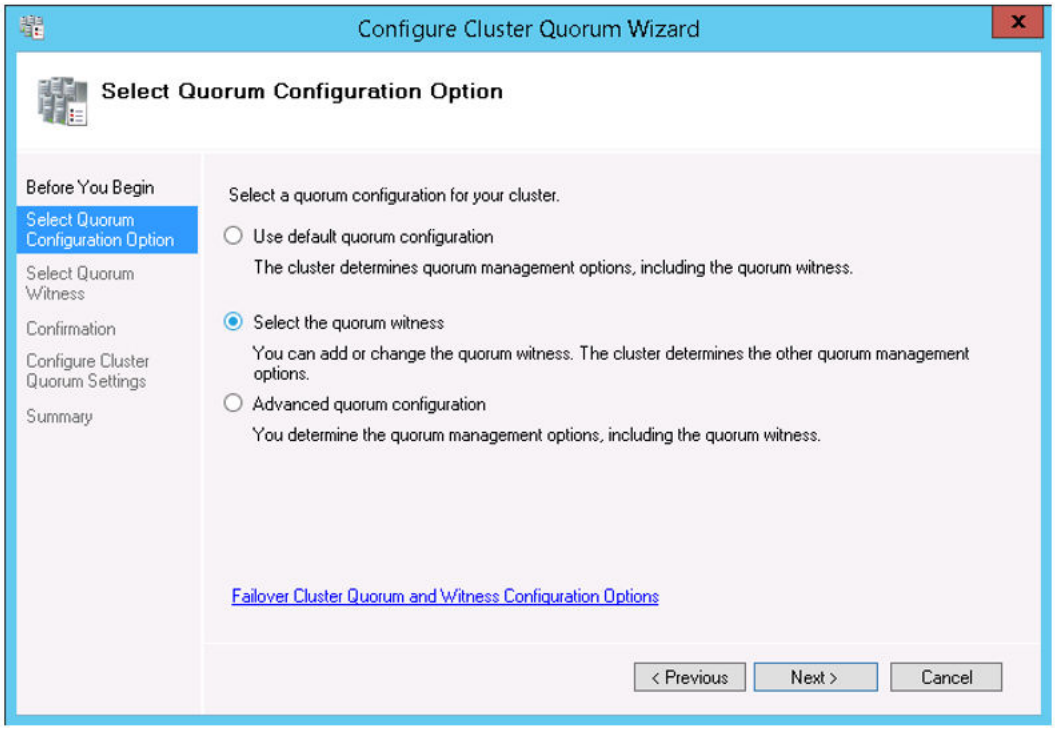

Figure 12. Quorum cluster wizard

5. Click **Next**. Select **All Nodes** from the **Select Voting Configuration** screen.

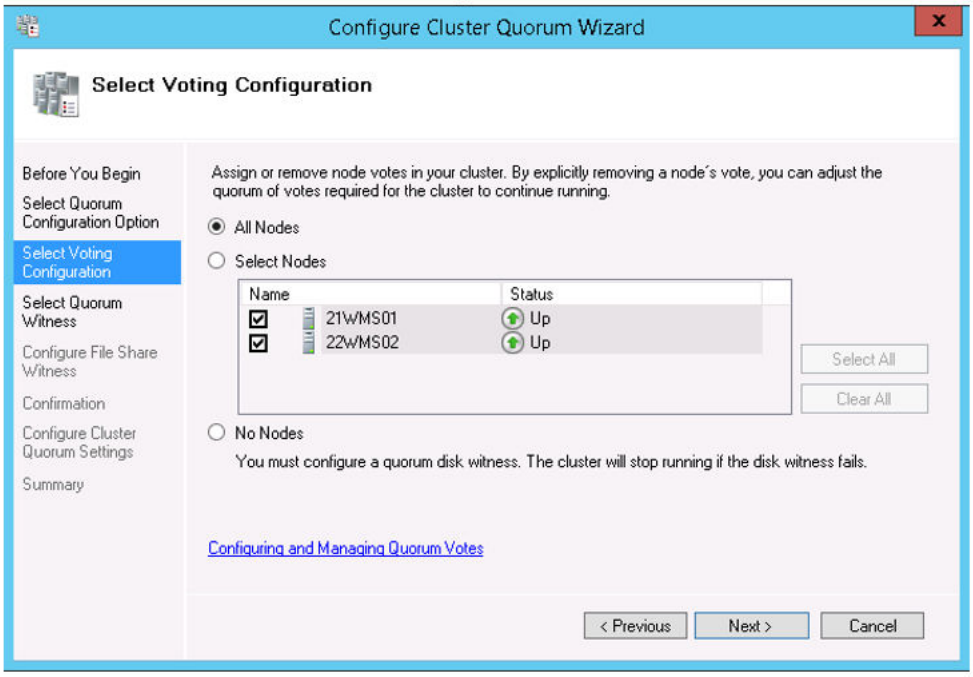

Figure 13. Select voting configuration

6. Click **Next** . Select **Configure a file share witness** from the **Select Quorum Witness** screen.
7. Click **Next** and then type the share path in the **File Share Path** field from the **Configure a file share witness** screen.

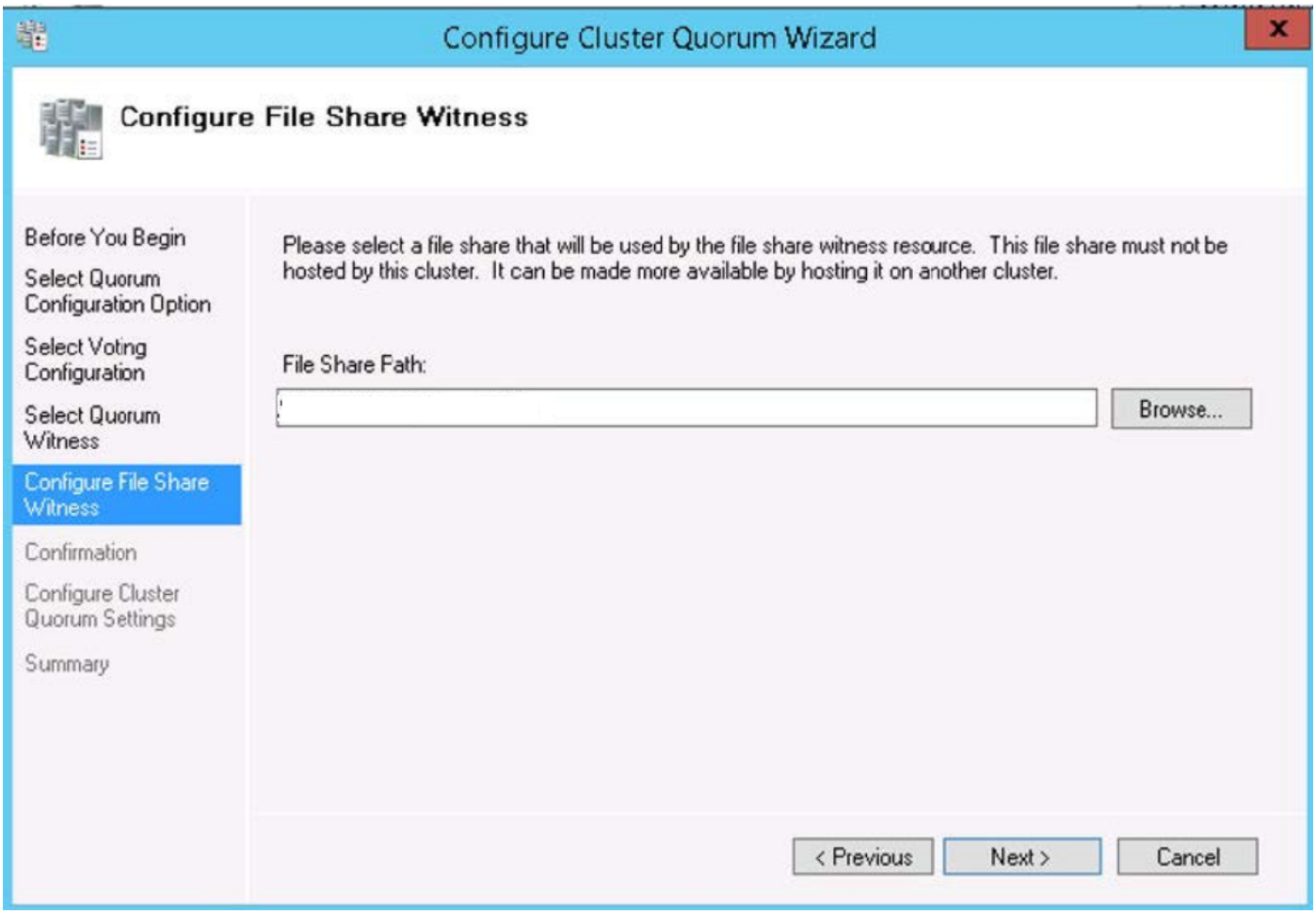

Figure 14. Configure file share witness

8. Click **Next** . The**Summary** screen is displayed with the configured quorum settings.

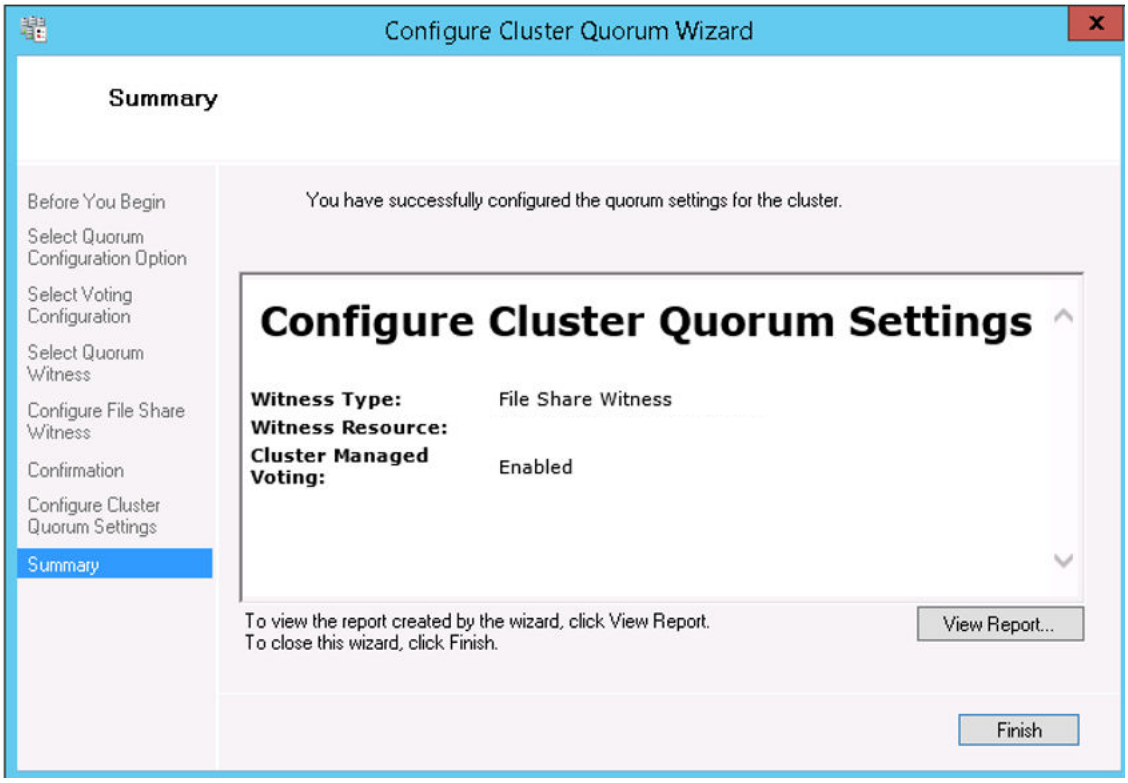

Figure 15. Summary of the quorum settings

9. Click **Finish** to complete the quorum settings.

Creating clustered roles

Prerequisites

After you create the failover cluster, you can create clustered roles to host cluster workloads. Ensure that Wyse Management Suite is installed on the servers and point to the remote database before you create clustered roles.

Steps

1. In Windows Server 2012, right-click the **Start** menu and then select **Server Manager** to launch the Server Manager dashboard
2. Click **Failover Cluster Manager** to launch the cluster manager.
3. Right-click **Roles** and then select **Configure Role** to display the **High Availability Wizard** screen.

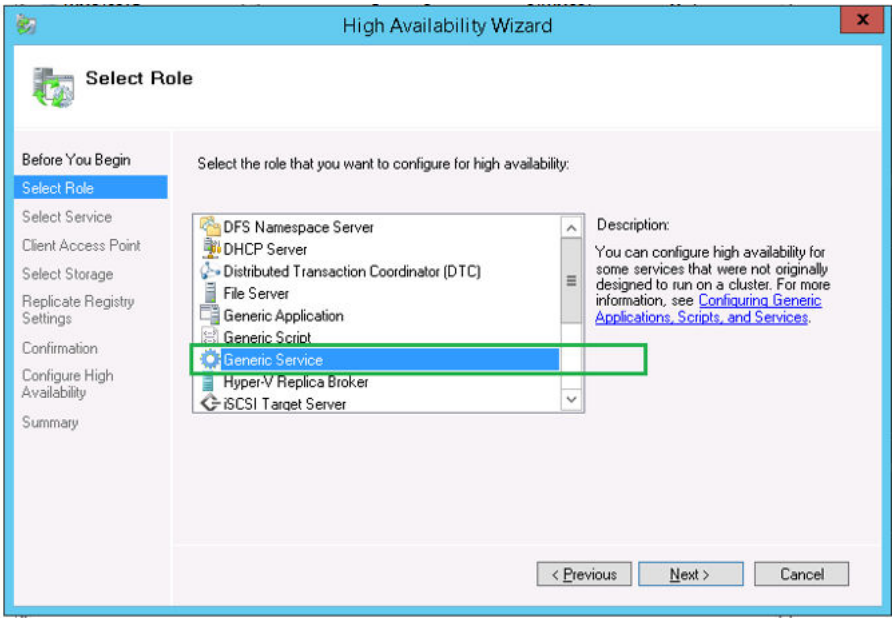

Figure 16. High availability wizard

4. Select **Generic Service** and then click **Next** to view the **Select Service** screen.

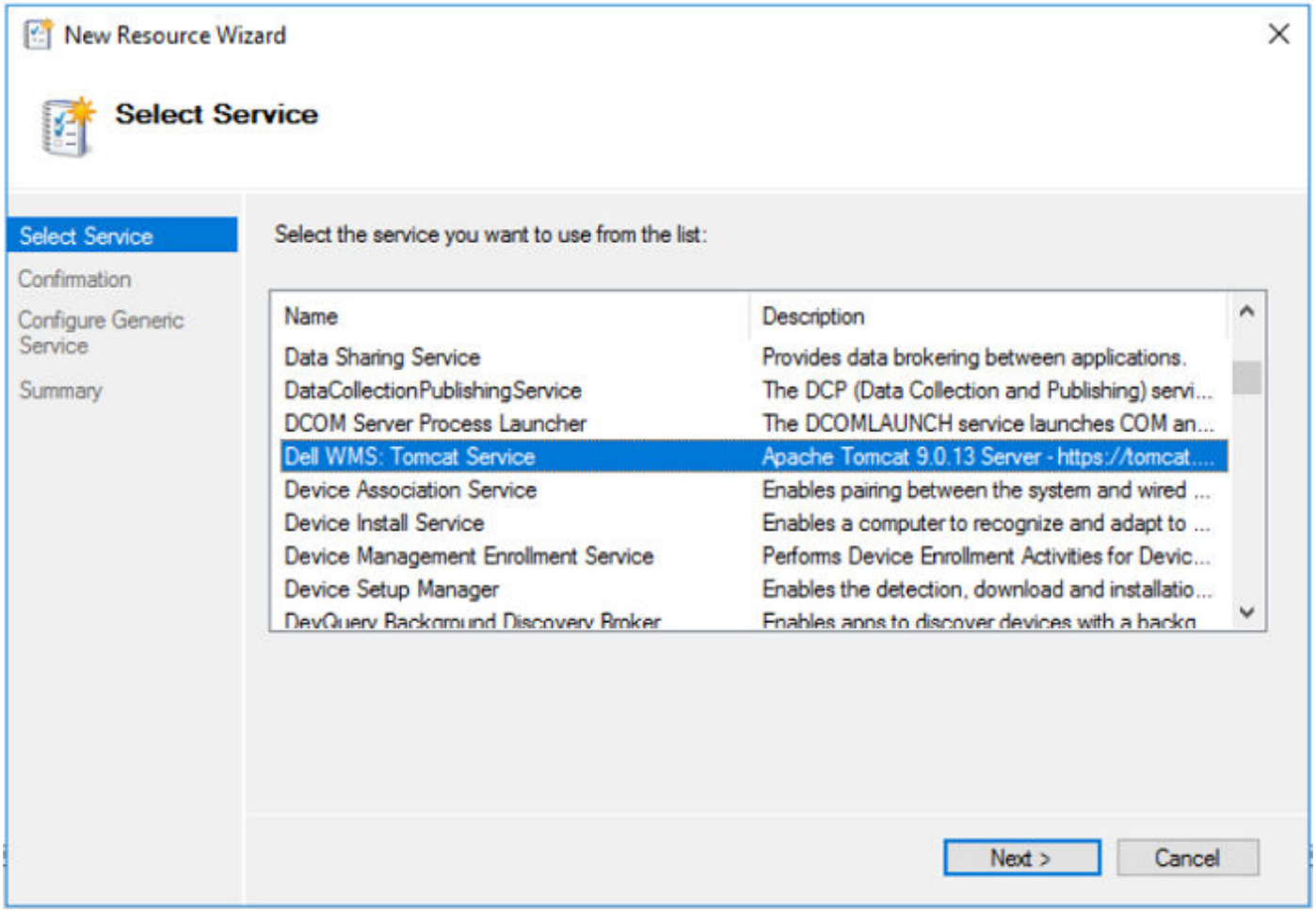

Figure 17. Select service

5. Select **Dell WMS: Tomcat Service** and then click **Next**.

NOTE: You can add the Wyse Management Suite related services to the cluster only after you install Wyse Management Suite.

The **High Availability Wizard** screen is displayed where you need to create the client access point and establish connectivity between the Windows server 2012 and Wyse Management Suite.

6. Type a network name in the **Name** field and then click **Next**. The **Confirmation** screen is displayed with the network name and IP address details of the server.

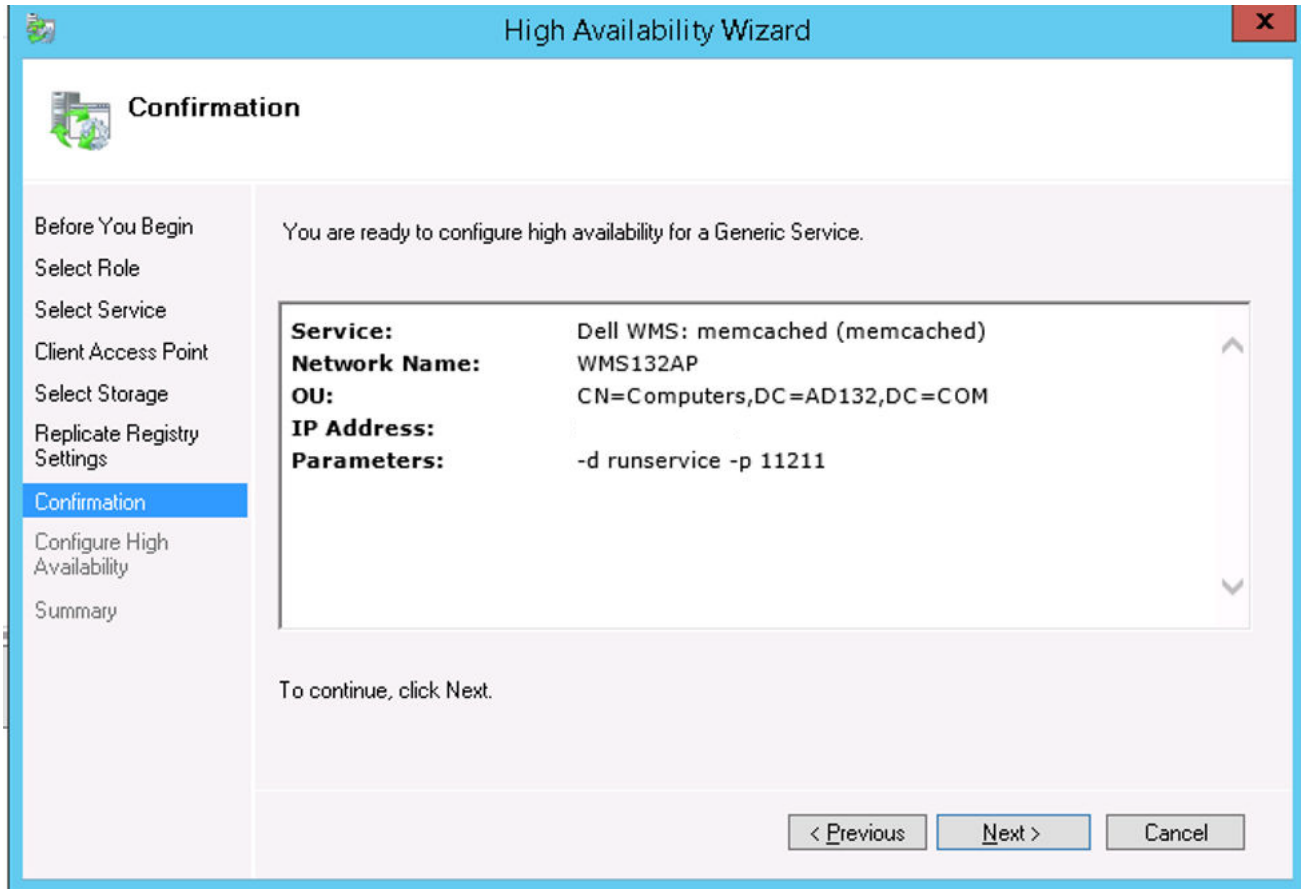

Figure 18. Confirmation

7. Click **Next** to complete the process.
8. To add other Wyse Management Suite services as part of the cluster, launch **Failover Cluster Manager**, and then go to **ActionsRoles** to display the network name that you have created.
9. Click on the network name, and go to **Add ResourceGeneric Service**.
10. Select the following services from the **New Resource Wizard** screen that needs to be added as part of the cluster:
    a. Dell WMS: MQTT Broker
    b. Dell WMS: memcached
11. Click **Next** to complete the task.
    The Wyse Management Suite services that have been added as part of the cluster are displayed with the status **Running**.

5

Achieve high availability for MySQL InnoDB

About this task

The following steps explain how to achieve high availability for MySQL InnoDB:

Steps

1. Check MySQL InnoDB server instance—see Create MySQL InnoDB cluster.
2. Add server or node to MySQL InnoDB—see Adding server or node to MySQL InnoDB cluster.
3. Create MySQL Router—see Configure MySQL Router.

High availability with MySQL InnoDB

The MySQL InnoDB cluster provides a complete high availability solution for MySQL. The client application is connected to the primary node by using the MySQL router. If the primary node fails, a secondary node is automatically promoted to the role of primary node, and the MySQL router routes the requests to the new primary node.

The components of the MySQL InnoDB cluster are:

- MySQL server
- MySQL router

Install MySQL InnoDB database

About this task

To install MySQL InnoDB database, do the following:

Steps

1. Double-click the MySQL installer.
   The **MySQL Installer** window is displayed.
2. On the **License Agreement** screen, read the license agreement, and click **Next**.
3. On the **Choosing a Setup Type** screen, click the **Custom** radio button, and click **Next**.

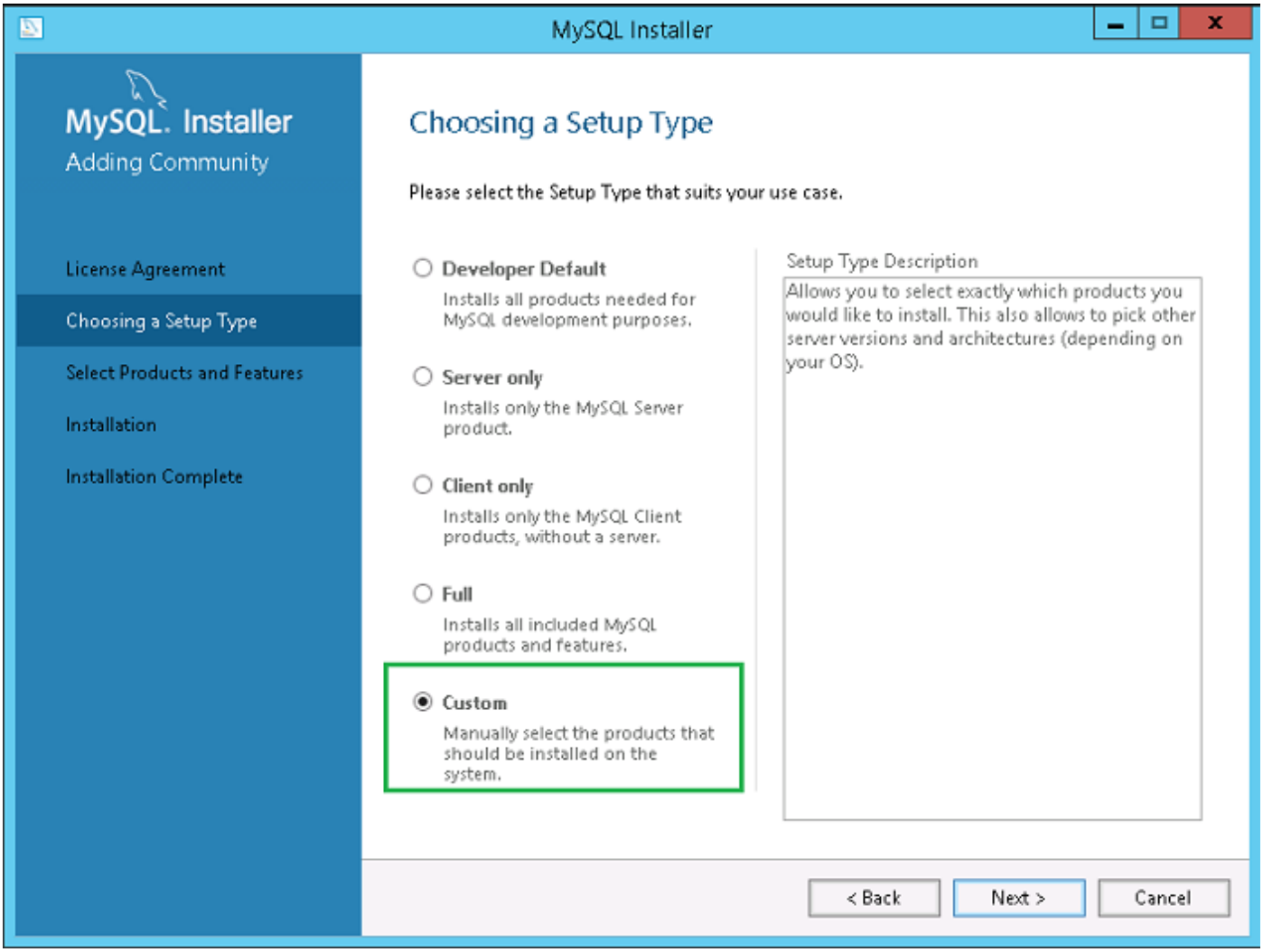

Figure 19. Setup type

4. On the **Select Products and Features** screen, select the MySQL Server, workbench, and shell components, and click **Next**.

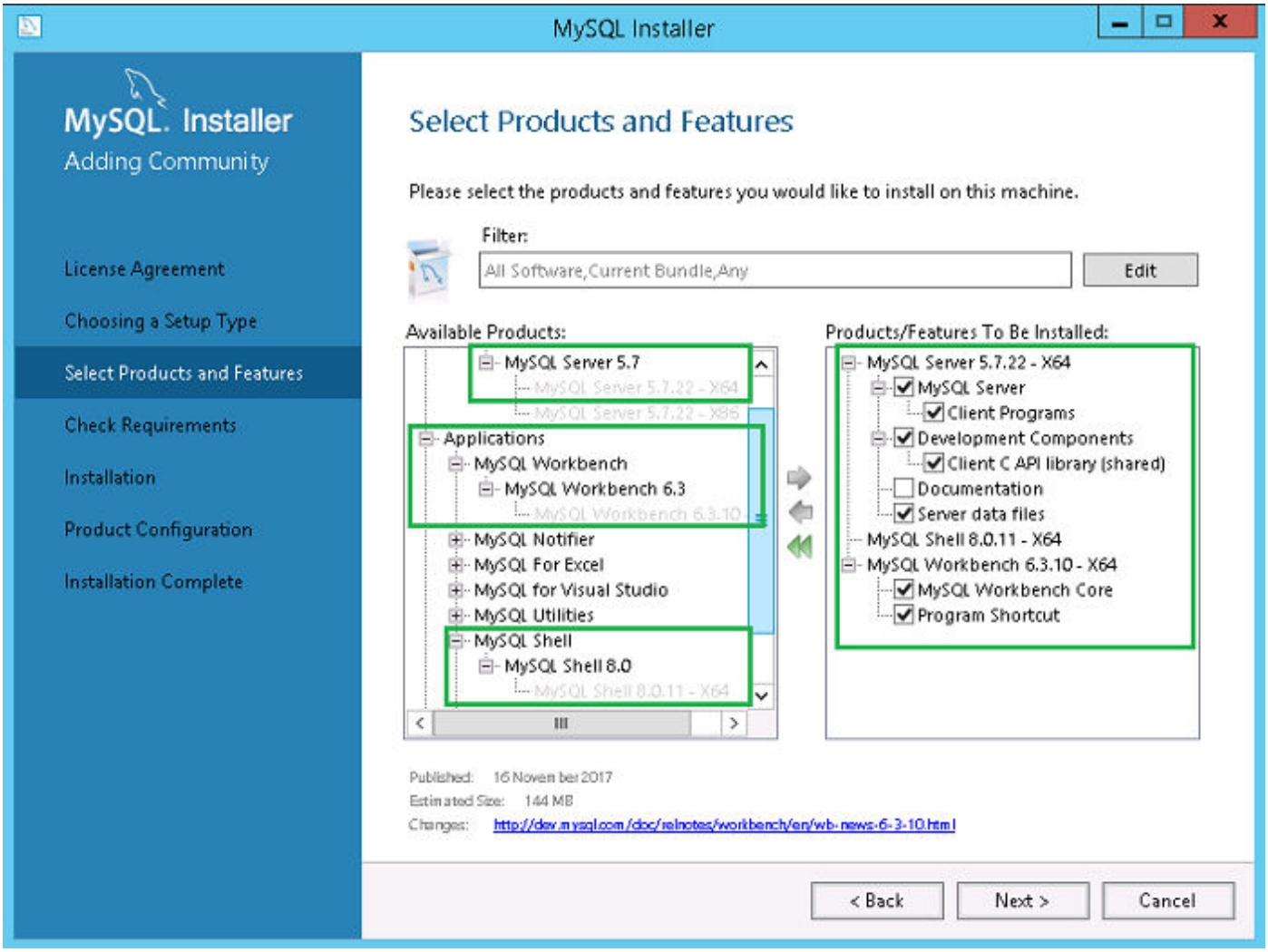

Figure 20. Products and features

5. On the **Check Requirements** screen, select the components, and click **Execute**.

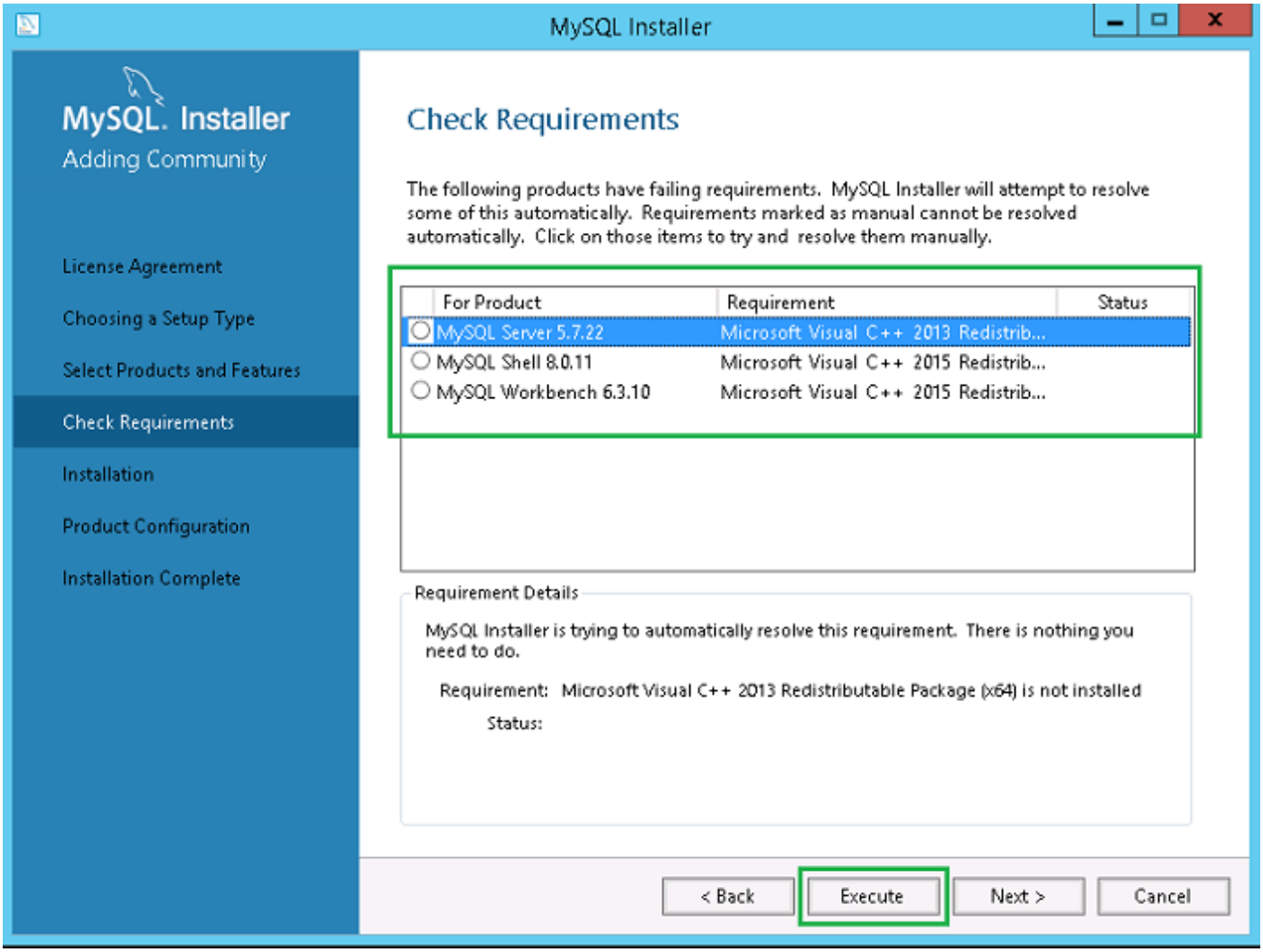

Figure 21. Requirements

6. Install the required components, and click **Next**.

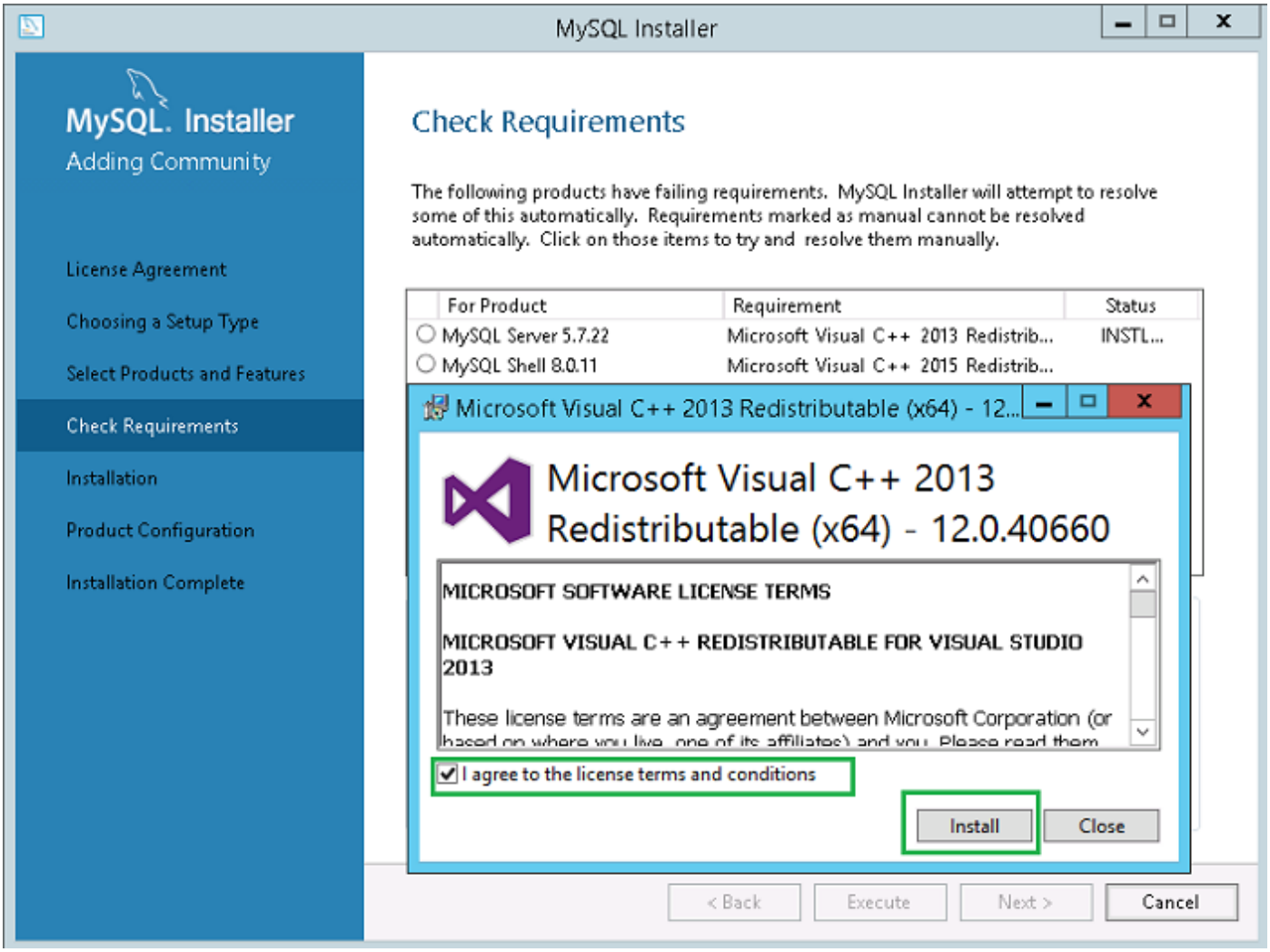

Figure 22. Components installation

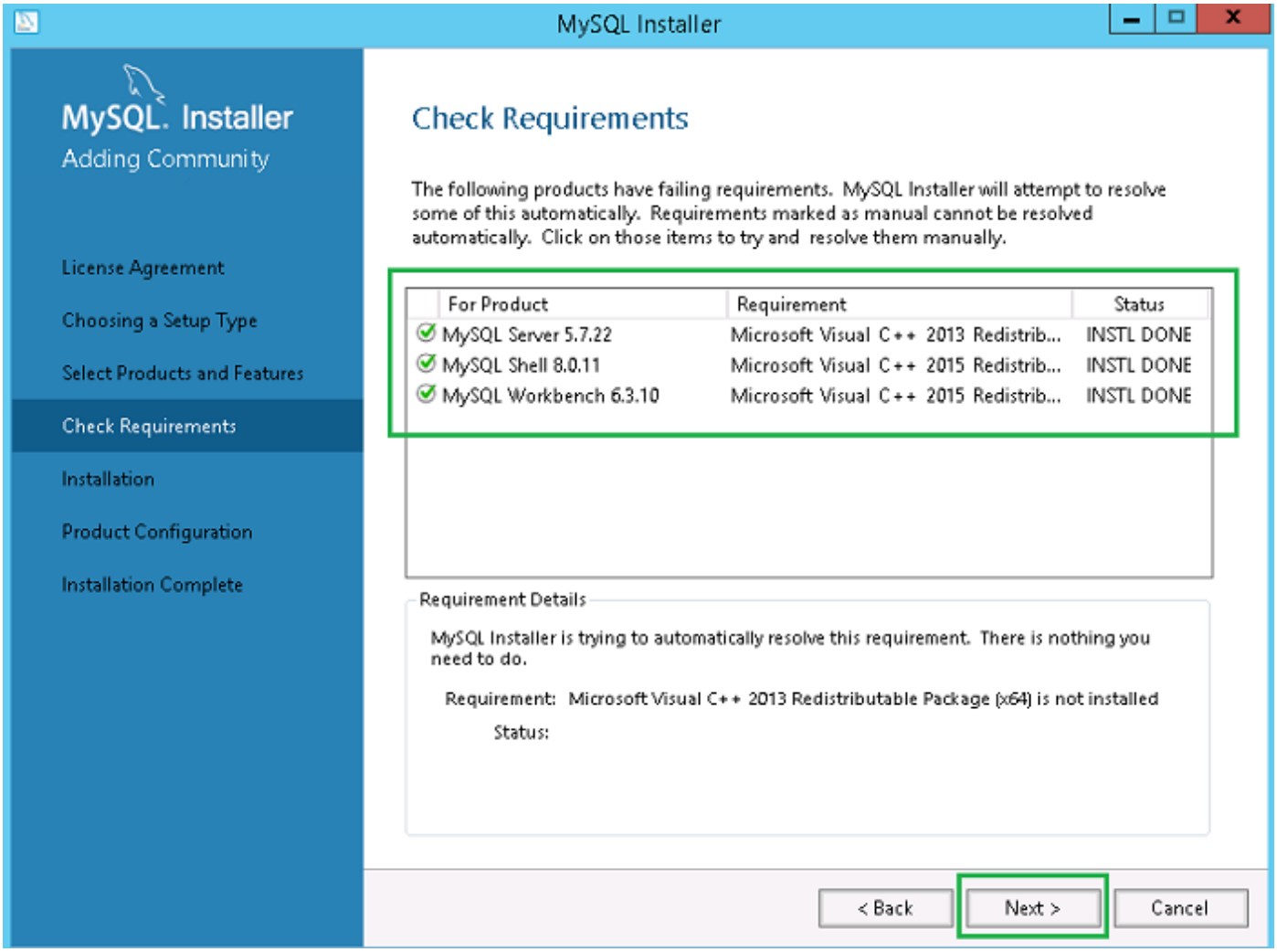

Figure 23. Requirements

7. On the **Installation** screen, click **Execute**.

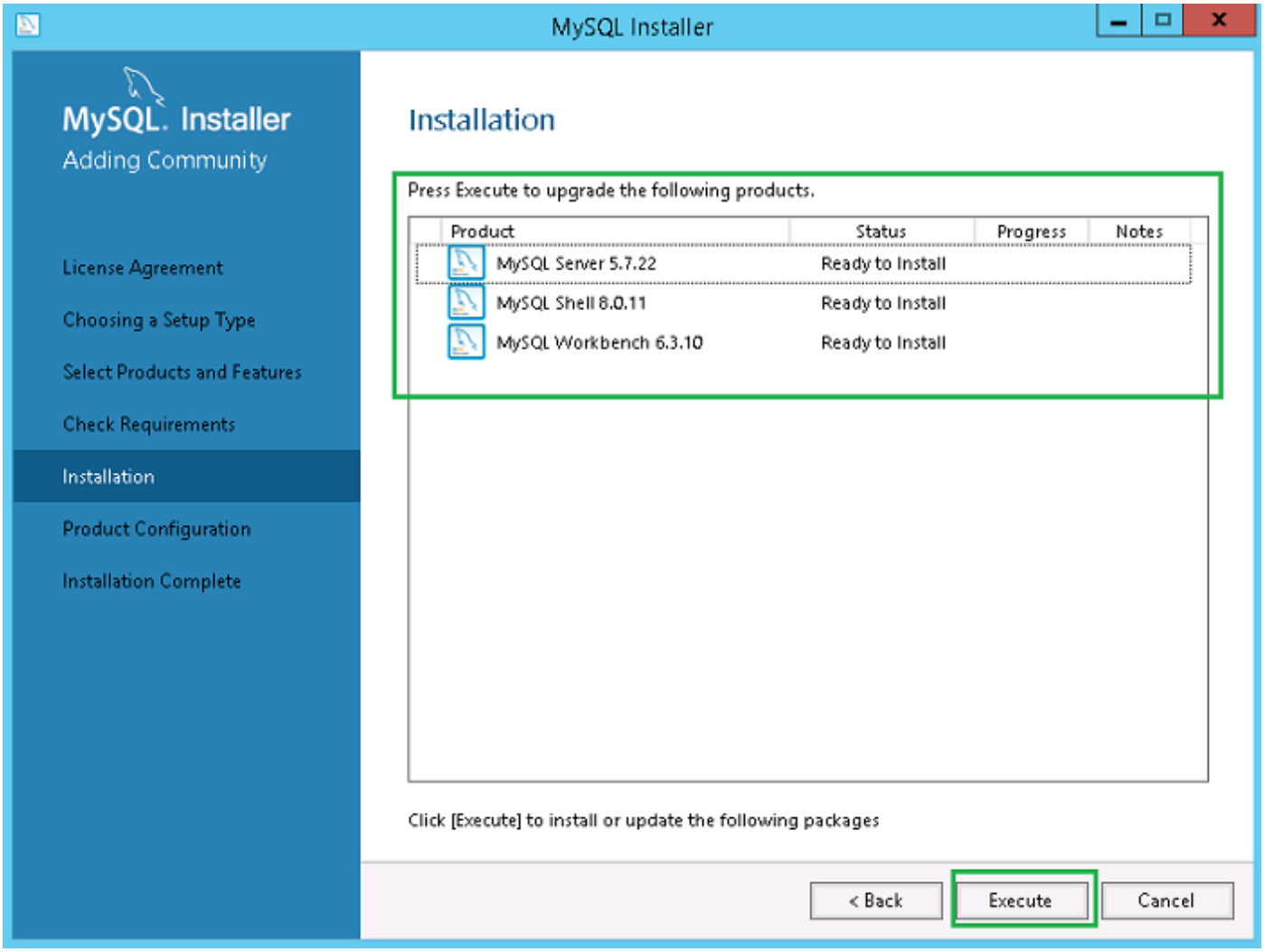

Figure 24. Installation

The MySQL server, workbench, and shell components are upgraded.

8. Click **Next**.

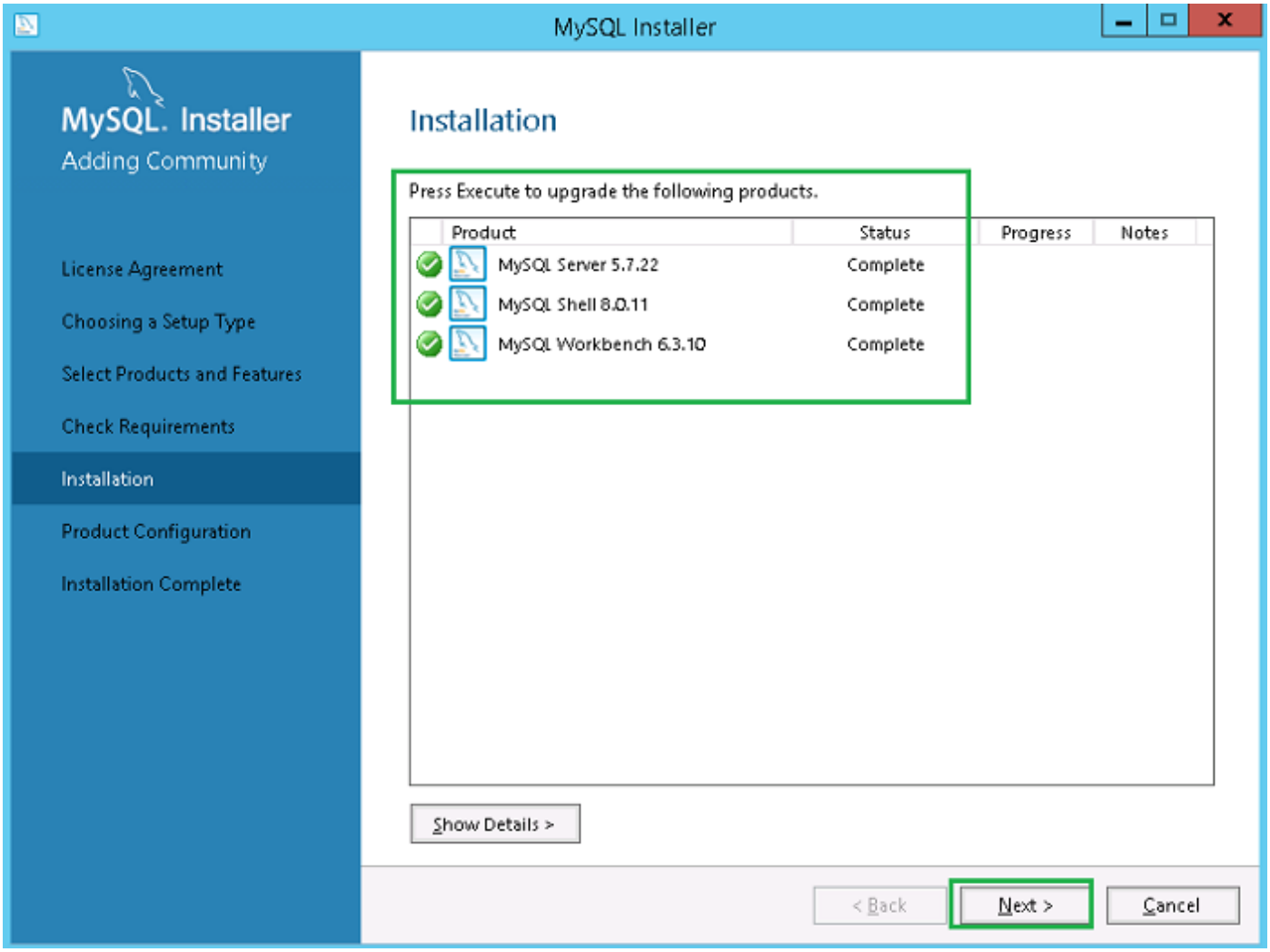

Figure 25. Installation

9. On the **Product Configuration** screen, the MySQL server component is displayed.

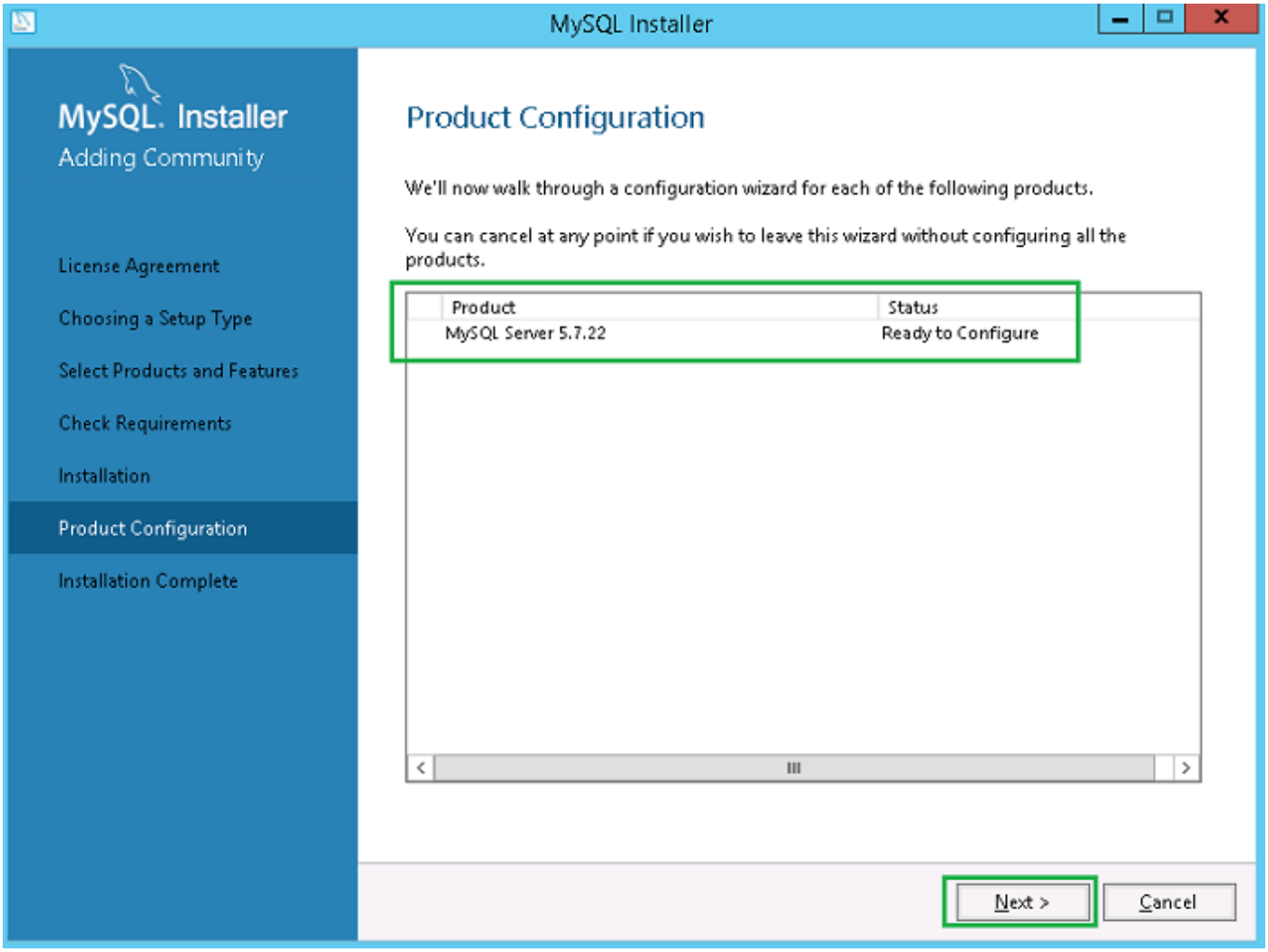

Figure 26. Product configuration

10. Click **Next** to configure the MySQL server component.
11. On the Group Replication screen, click the **Standalone MySQL Server / Classic MySQL Replication** radio button, and click **Next**.

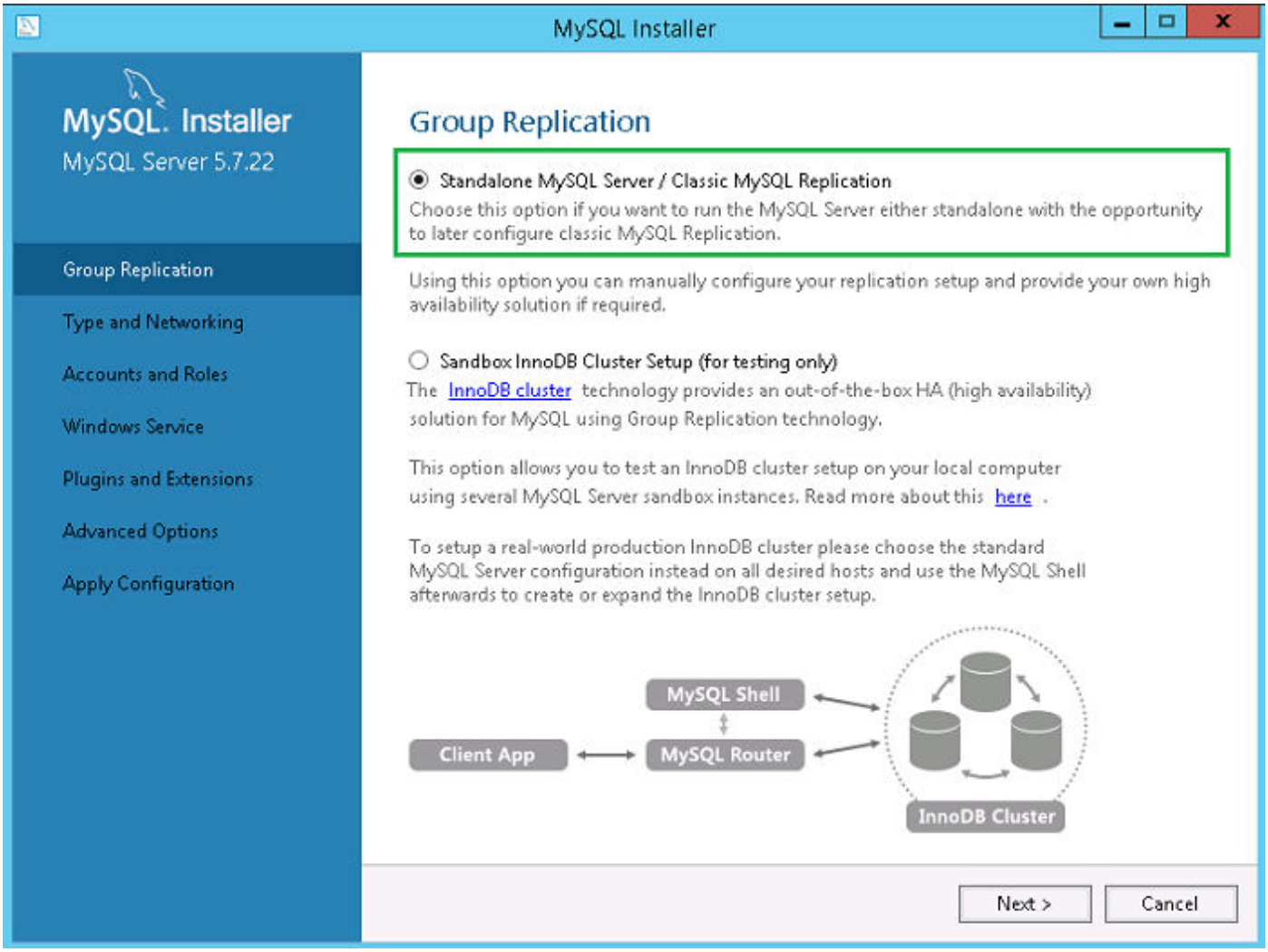

Figure 27. Group replication

12. On the **Type and Networking** screen, select the **Dedicated Computer** option from the **Config Type** drop-down list.

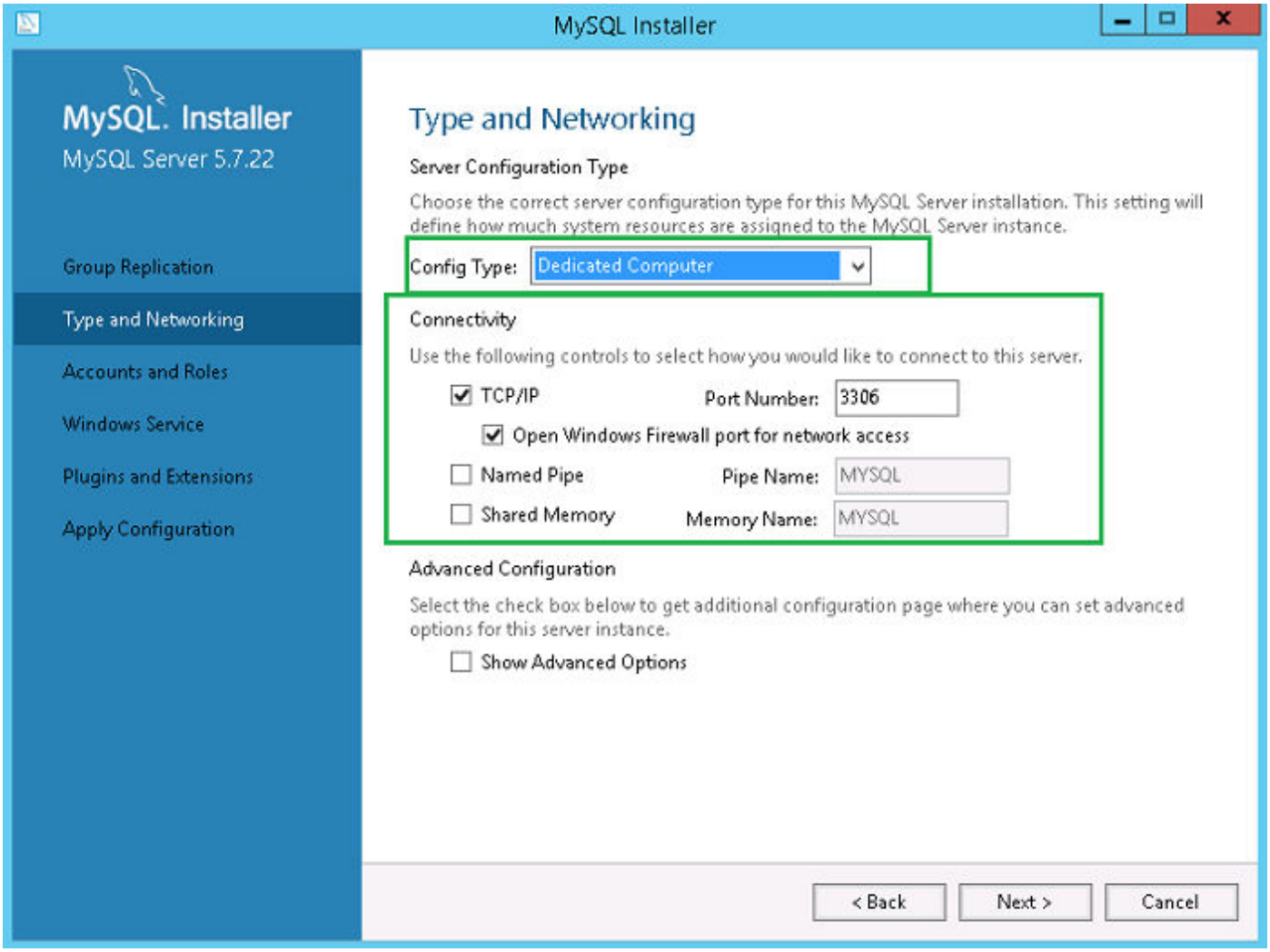

Figure 28. Type and networking

13. Select and configure the options in the **Connectivity** section, and click **Next**.
14. In the **Accounts and Roles** screen, enter the MySQL root password.
15. Click **Add User**.

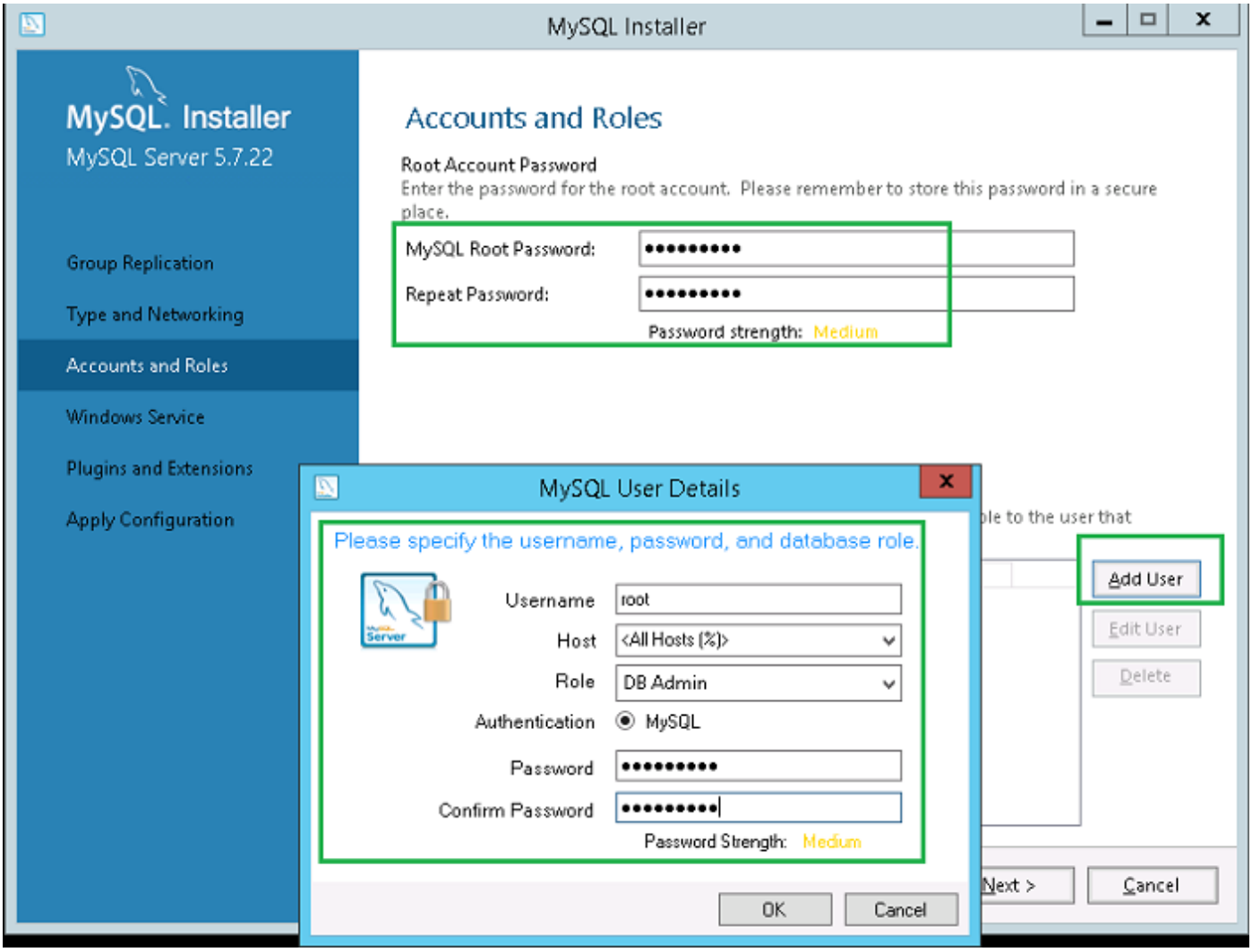

Figure 29. Add user

The **MySQL User Details** window is displayed.

16. Enter the credentials and click **Ok**.
    The newly added user account is displayed in the **MySQL User Accounts** section.

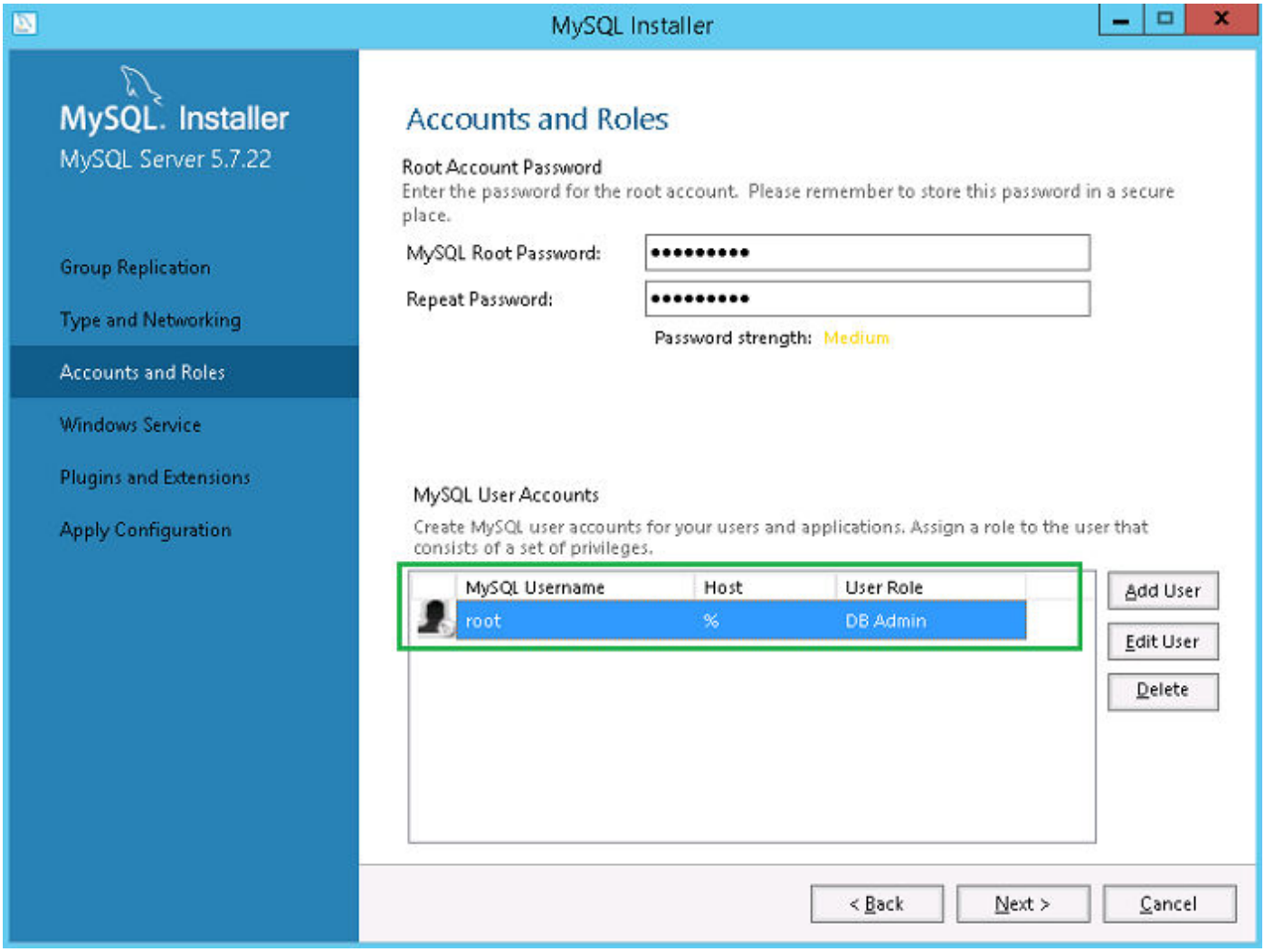

Figure 30. Accounts and roles

17. Click **Next**.
18. On the **Windows Service** screen, enter the MySQL Windows service name, and click **Next**.

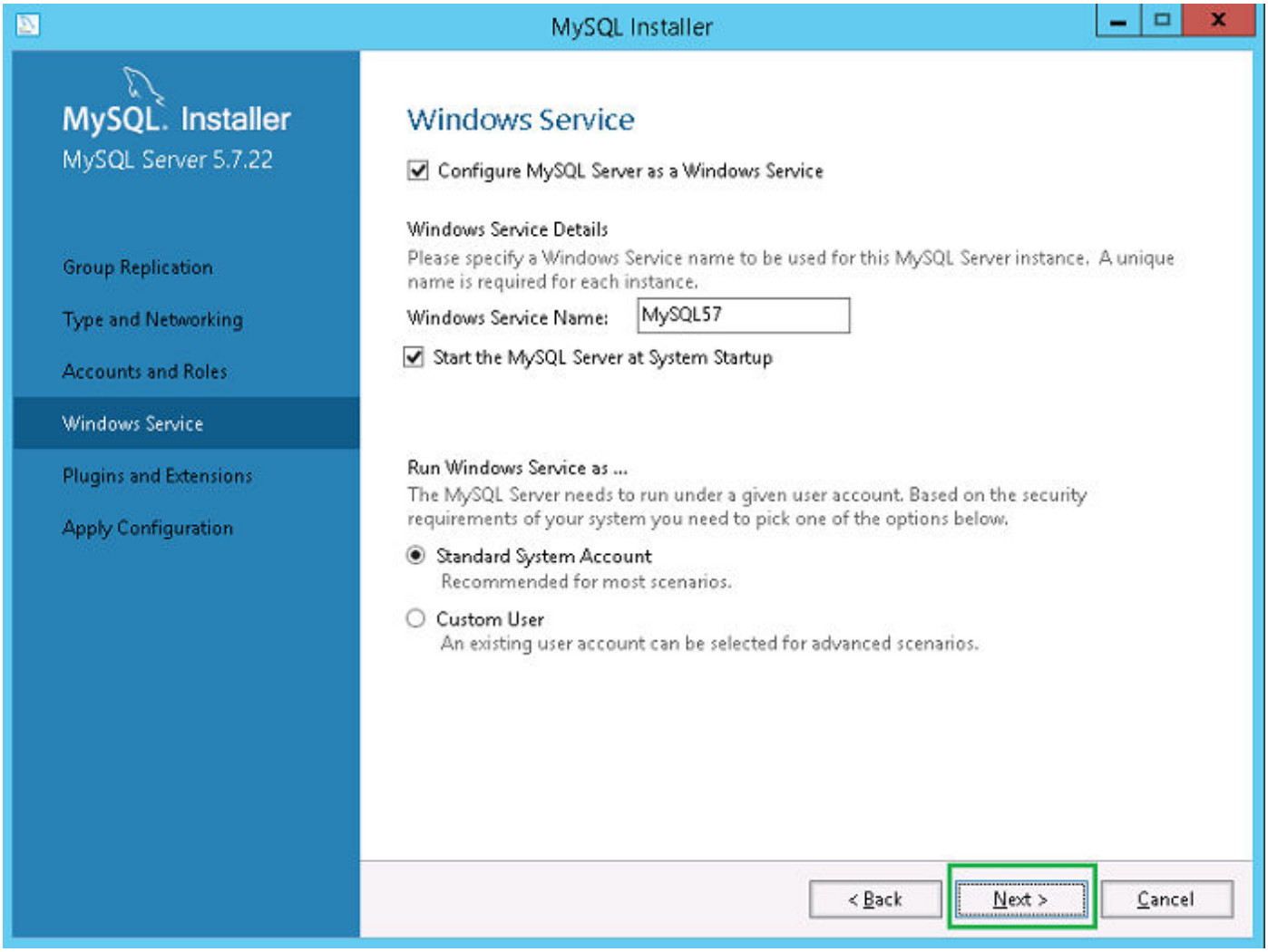

Figure 31. Windows service

19. On the **Plugins and Extensions** screen, click **Next**.

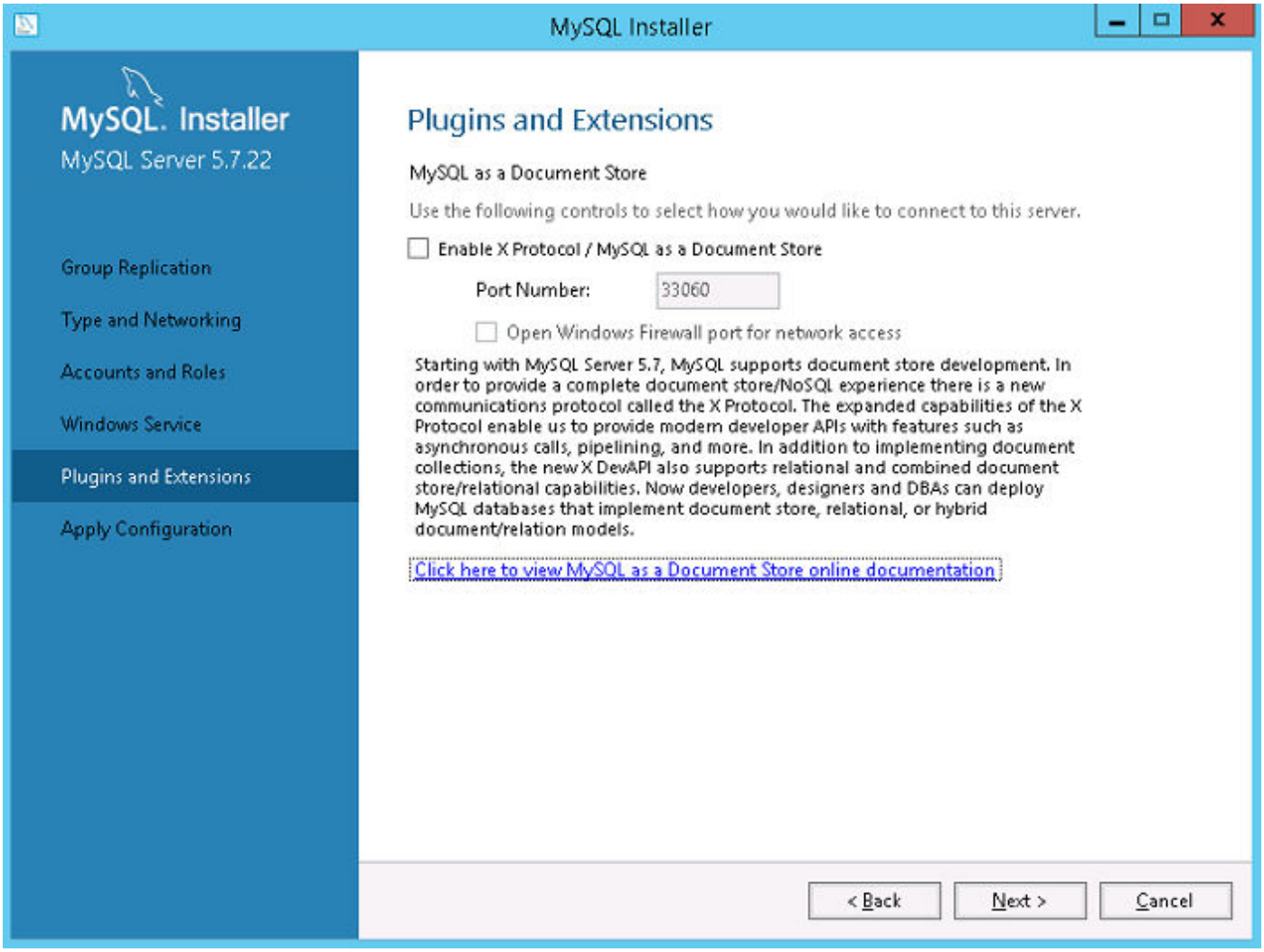

Figure 32. Plugins and extensions

20. On the **Apply Configuration** screen, click **Execute**.
    The configurations are applied to the MySQL component.

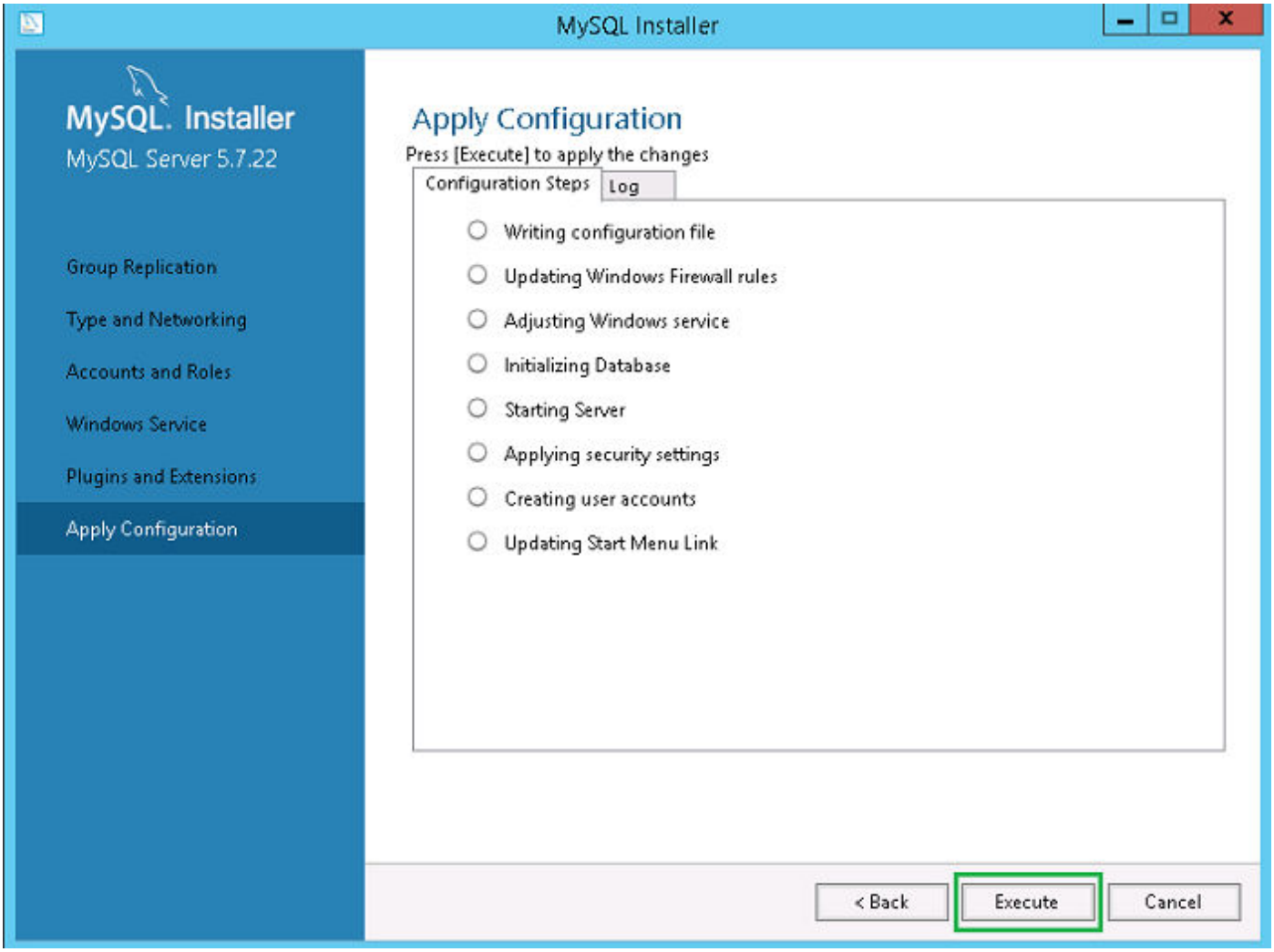

Figure 33. Apply configurations

21. Click **Finish**.

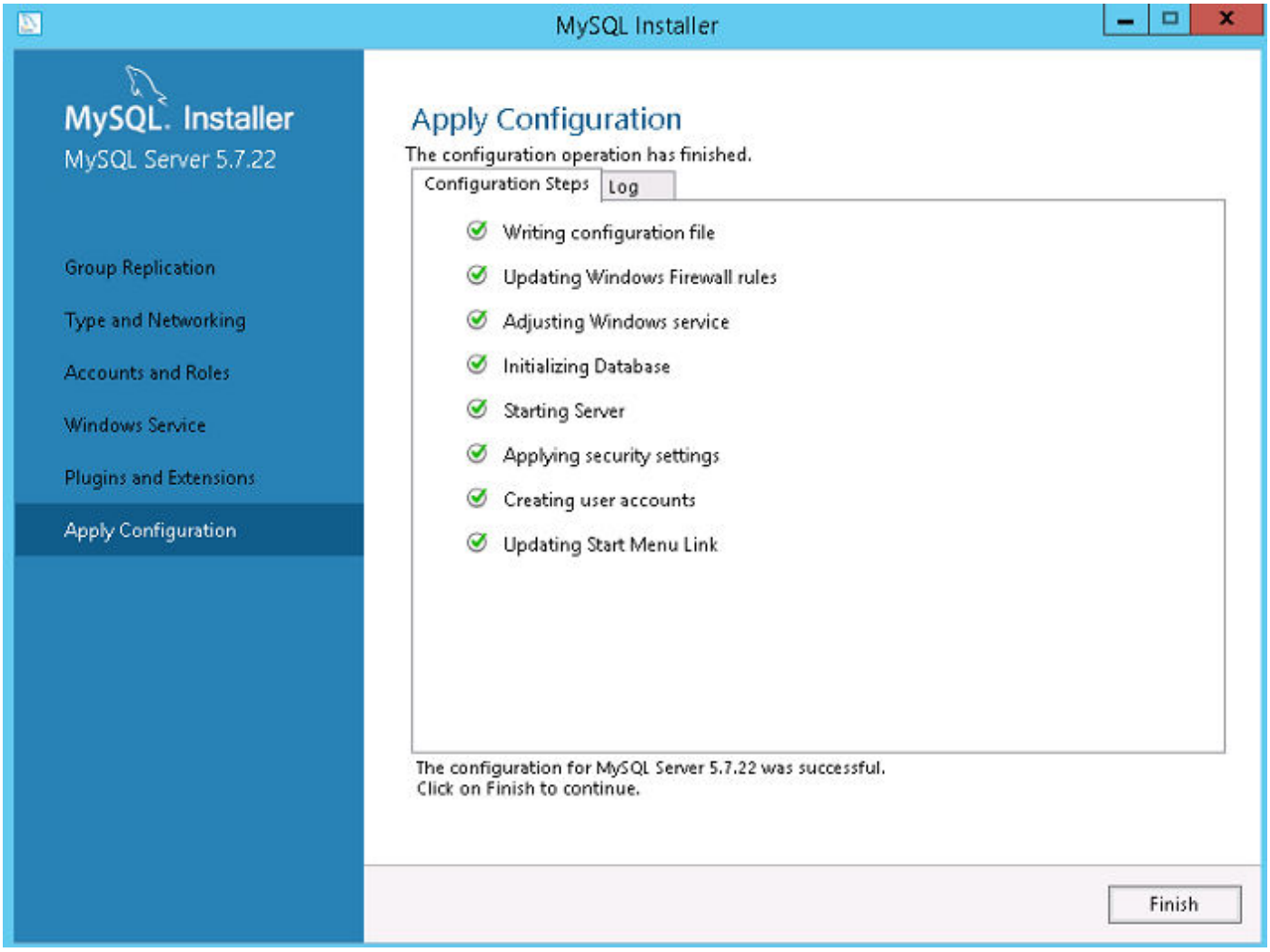

Figure 34. Apply configurations

22. On the **Product Configuration** screen, click **Next**.

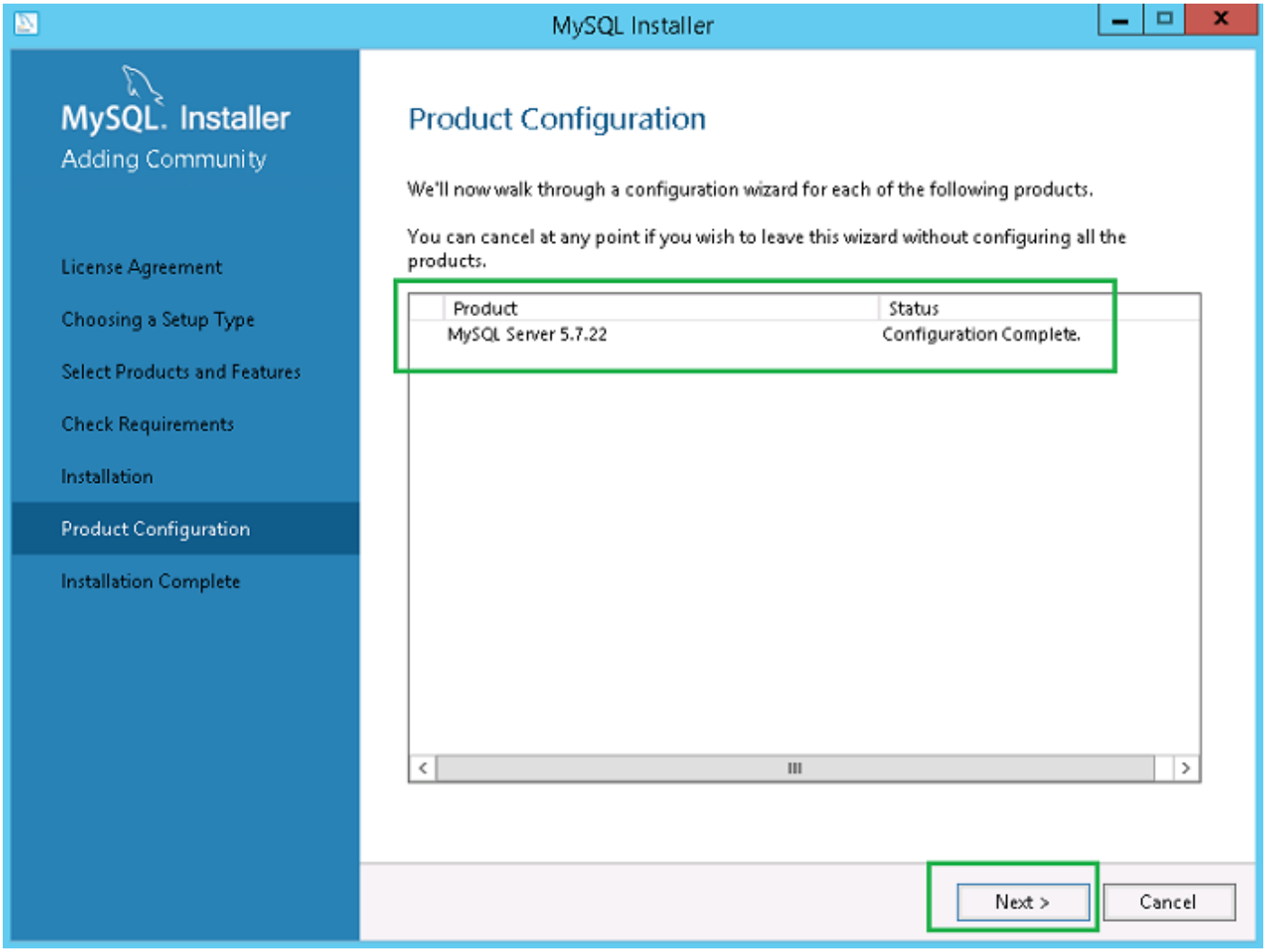

Figure 35. Product configuration

23. On the **Installation Complete** screen, click **Finish**.

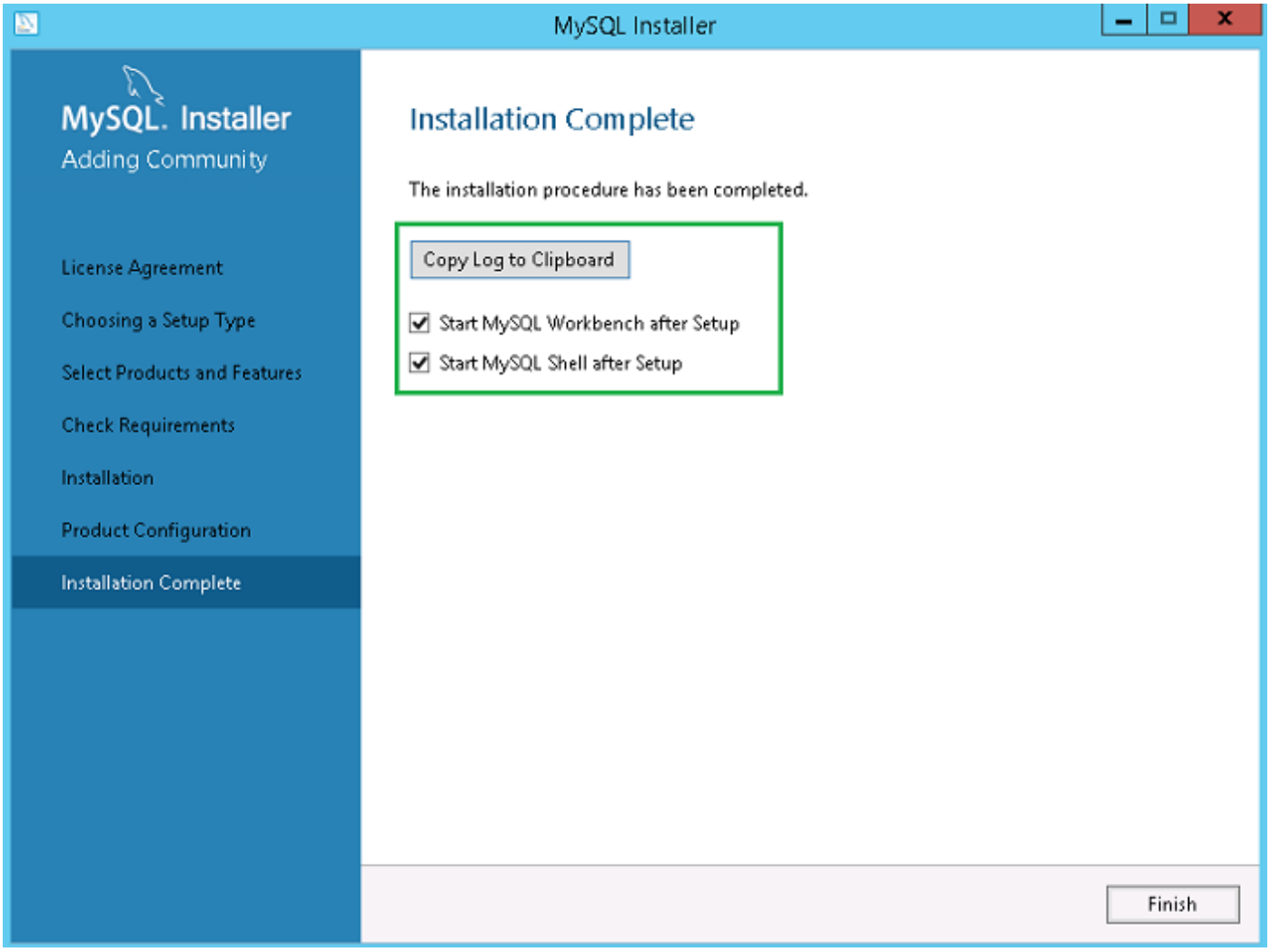

Figure 36. Installation complete

Next steps

Follow the procedure to install and configure MySQL server in all the three servers of the MySQL cluster.

NOTE: To set up the environment as per the high availability setup, see dev.mysql.com.

Check MySQL InnoDB server instances

About this task

Before you add MySQL InnoDB to the cluster setup, verify that MySQL InnoDB is created as per the cluster requirements.

You must login as **root** user to run the commands and restart the system each time you run a set of commands.

Run the following commands to verify that the MySQL InnoDB server instance meets the configured cluster requirements:

NOTE: The IP Address is different for each system that is used at your work place and the following commands are used only as an example.

Steps

- To check that the MySQL InnoDB is created as per the requirements, run the following commands at the command prompt:
- `mysql-js> dba.checkInstanceConfiguration('root@IP Address1')`
- `mysql-js> dba.checkInstanceConfiguration('root@IP Address2')`

- `mysql-js> dba.checkInstanceConfiguration('root@IP Address3')`

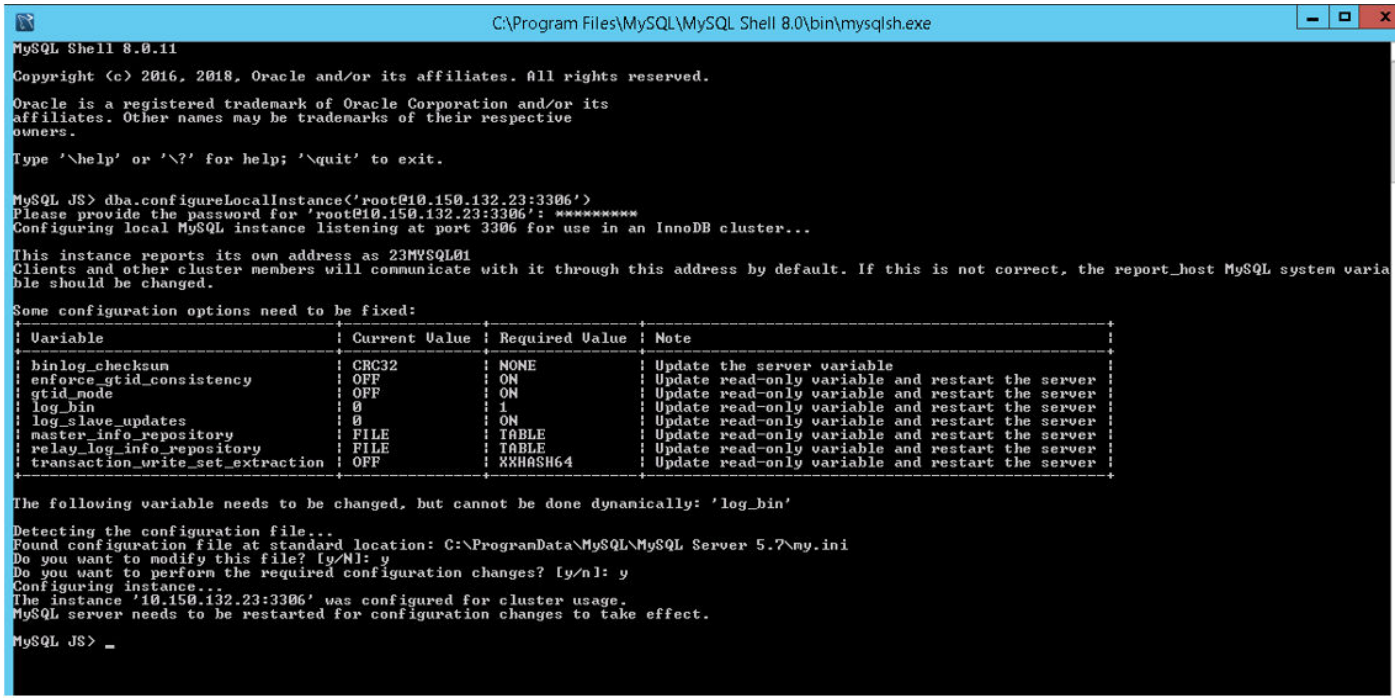

Figure 37. MySQL command prompt

To check that the MySQL InnoDB is created on all the three cluster nodes, run the following commands at the command prompt:

- `mysql-js> dba.checkInstanceConfiguration('root@IPAddress1:3306')`
- `mysql-js> dba.checkInstanceConfiguration('root@IPAddress2:3306')`
- `mysql-js> dba.checkInstanceConfiguration('root@IPAddress3:3306')`

The instance "IPAddress:3306" is valid for InnoDB cluster usage; 'Status': 'ok' message is displayed.

Create a cluster instance for MySQL InnoDB

Prerequisites

After you have installed MySQL InnoDB instance on the servers, create a cluster instance.

About this task

To create a cluster for MySQL InnoDB, do the following:

Steps

1. Login as administrator user from the command prompt. This user account should have administrative privileges. For example, **DBadmin**. The following screen shows an example of logging in as root user.

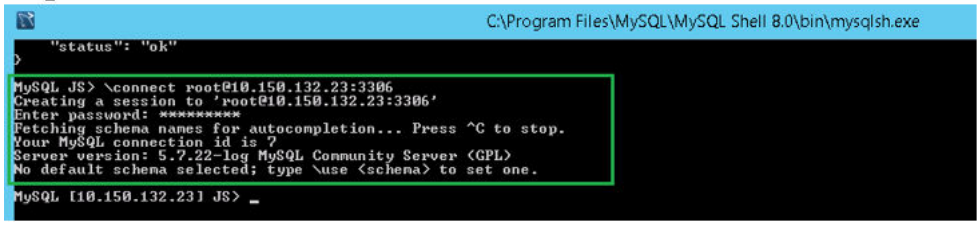

Figure 38. Login prompt

2. Run the following command to create a cluster with a unique name. For example, **MySQLCluster**.
   `MySql JS> var cluster = dba.createCluster('MySQLCluster')`

3. Run the following command to check the status of the cluster.
   `MySql JS>Cluster.status()`

   The status of the created cluster is displayed as **ONLINE** which indicates that the cluster is created successfully.

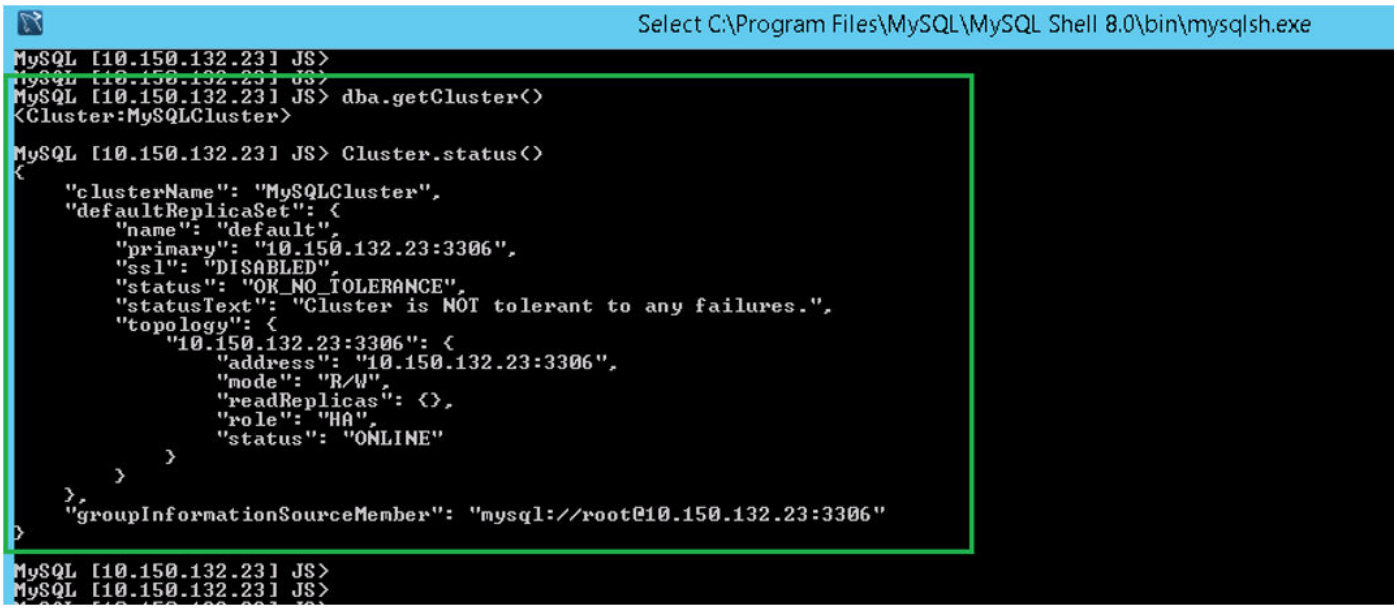

Figure 39. Confirmation screen

Add server instance to MySQL InnoDB cluster

Prerequisites

- Before you add servers or nodes to the clusters, change the server id to either 2 or 3 in the `my.conf` file in the secondary MySQL servers at `C:\ProgramData\MySQL\MySQL Server 5.7`.
- Only the primary MySQL server must have server ID as 1. The server ID should be unique across the SQL cluster.

About this task

You must add server instance to the MySQL InnoDB cluster as primary or secondary.

Do the following to add a server instance to the MySQL InnoDB cluster:

1. Log in as **DB Admin** user from the command prompt on the primary server.
2. Run the following command to add a server instance to the MySQL InnoDB cluster:

   ```
   cluster.addInstance('root@IPAddress2:3306')
   ```

   ```
   cluster.addInstance('root@IPAddress3:3306')
   ```

   NOTE: The IP address and the port numbers are only examples and varies based on the system that you are using at your work place.
3. Run the following command to check the status of the server instance:

   ```
   cluster.status()
   ```

NOTE:

- **If the server IDs are same in all the nodes, and if you try to add instances in the Cluster, the error message Server_ID is already in used by the peer node, Result<Runtime Error> is displayed.**
- **All the nodes should display the status as ONLINE which indicates that the nodes have been added successfully to the MySQL InnoDB cluster setup.**

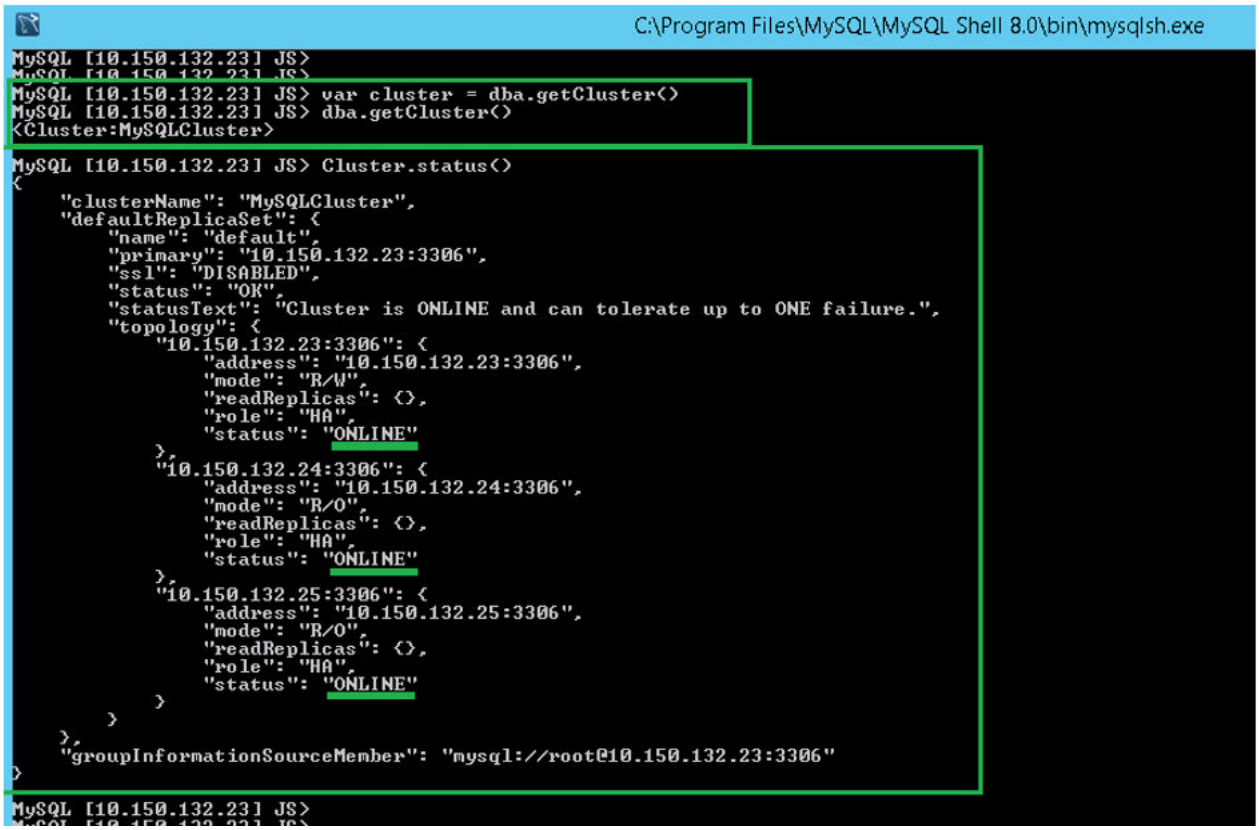

Figure 40. Cluster status

Configure MySQL Router

Prerequisites

MySQL Router establishes communication network between Wyse Management Suite and MySQL InnoDB.

About this task

To install MySQL Router, do the following:

Steps

1. Log in to the Windows Server 2012/2016 to install MySQL Router. For more information, see MySQL Router Installation
2. Select **MySQL Router** from the **Select Products and Features** screen and then click **Next** .

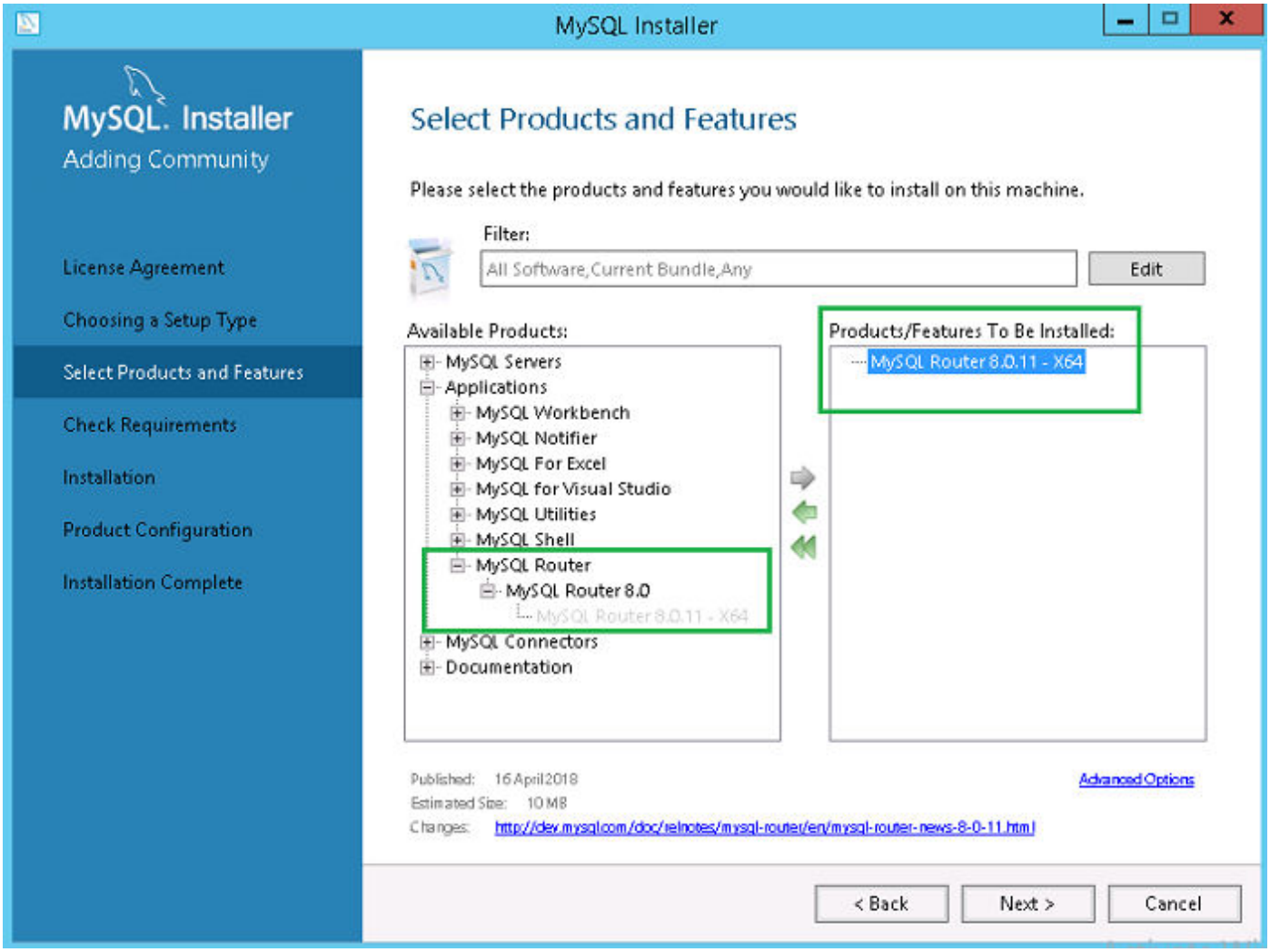

Figure 41. Select products and features

3. On the **Check Requirements** screen, click **Execute**.

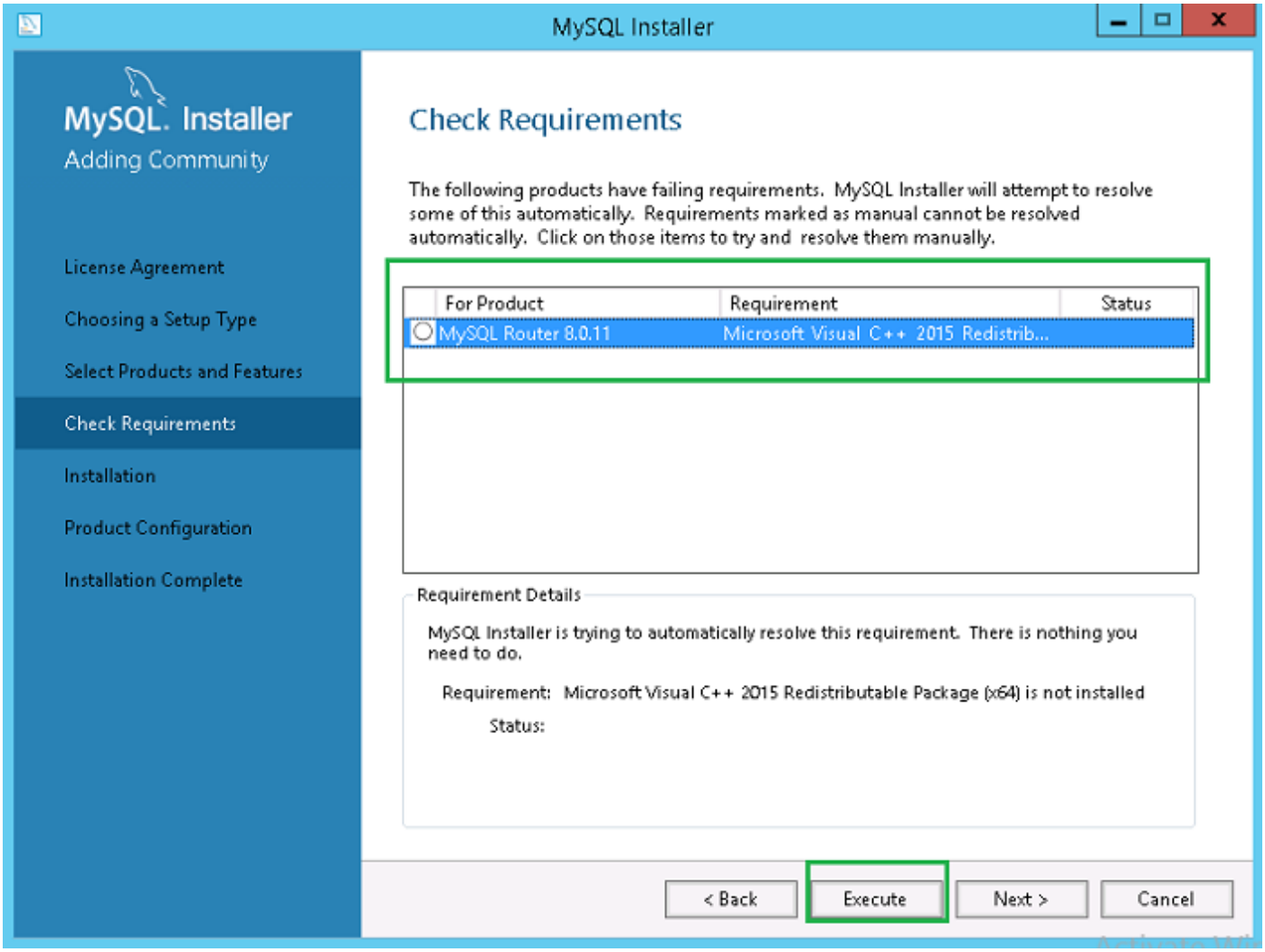

Figure 42. Check requirements

4. Install the required components, and click **Next**.

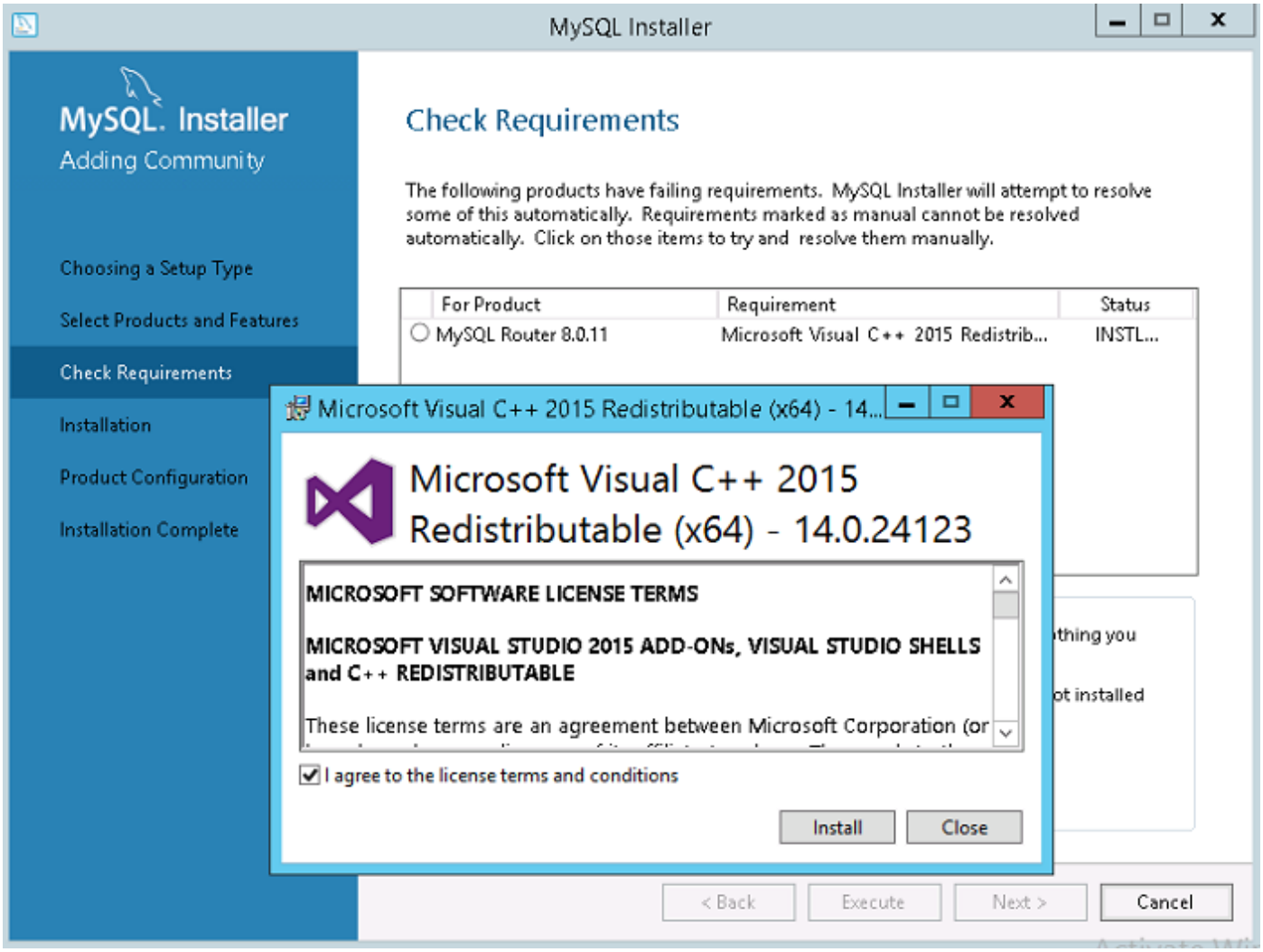

Figure 43. Components install

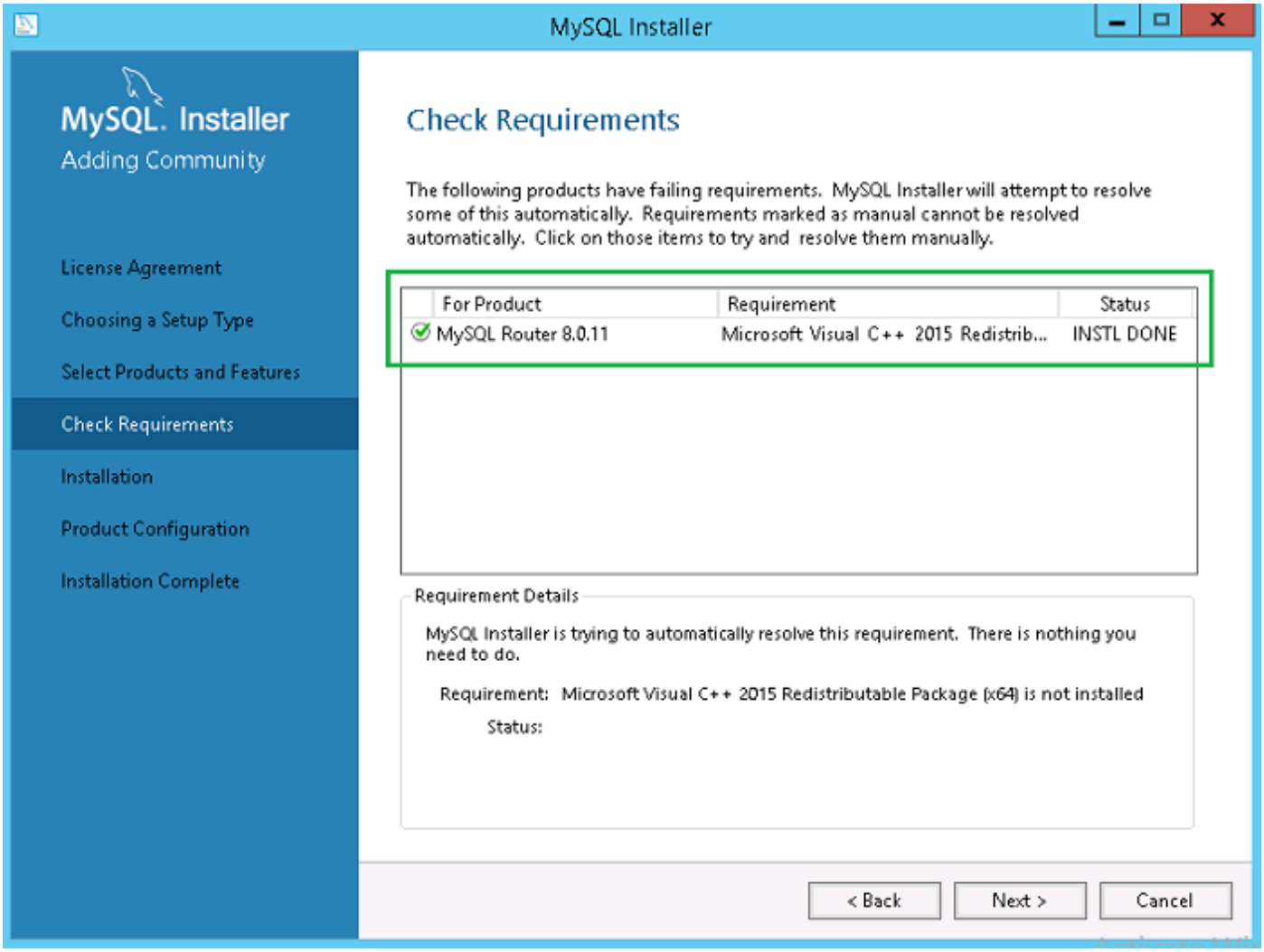

Figure 44. Check requirements

5. On the **Installation** screen, click **Execute**.

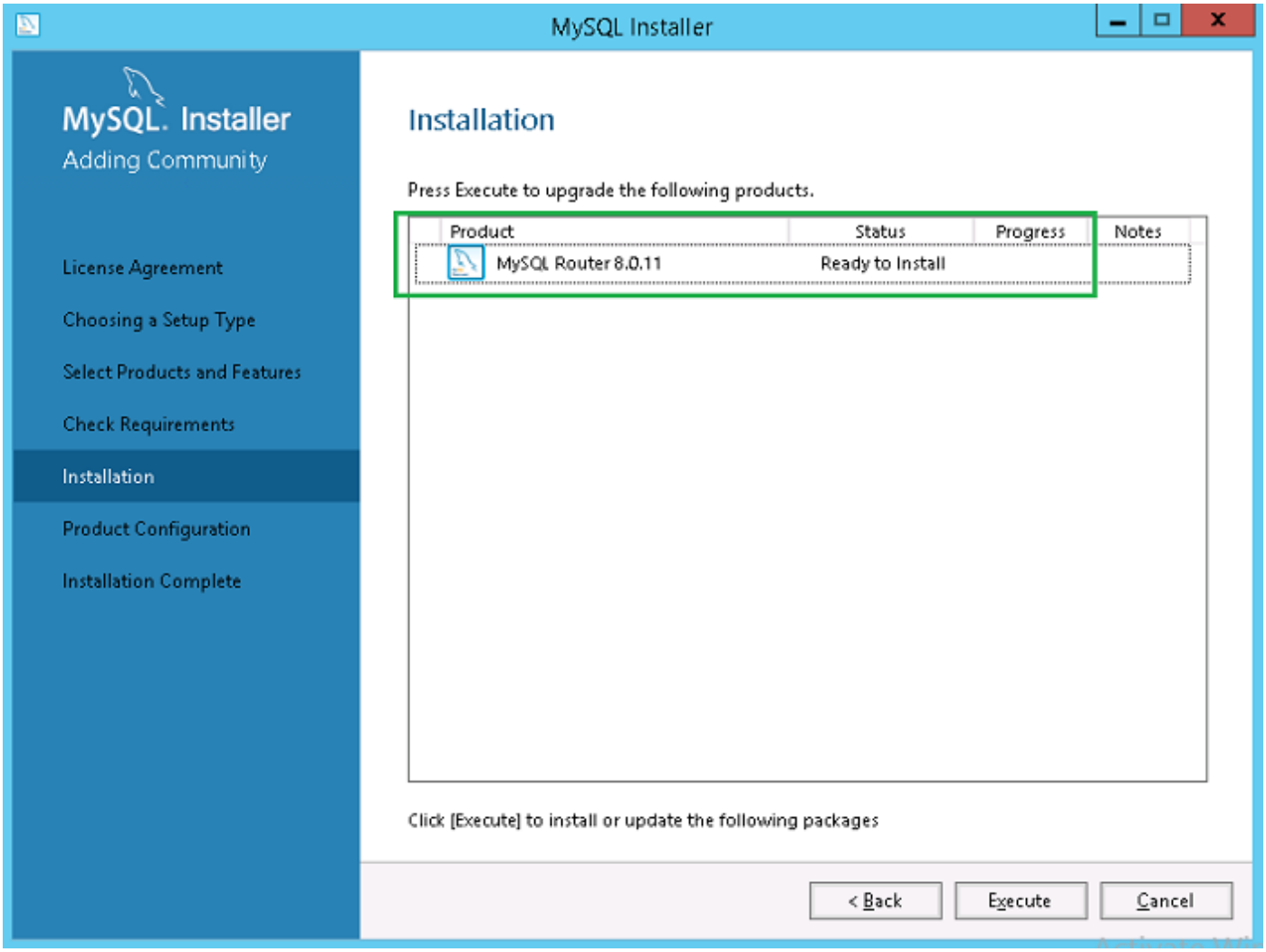

Figure 45. Installation

MySQL router component is upgraded.

6. Click **Next**.

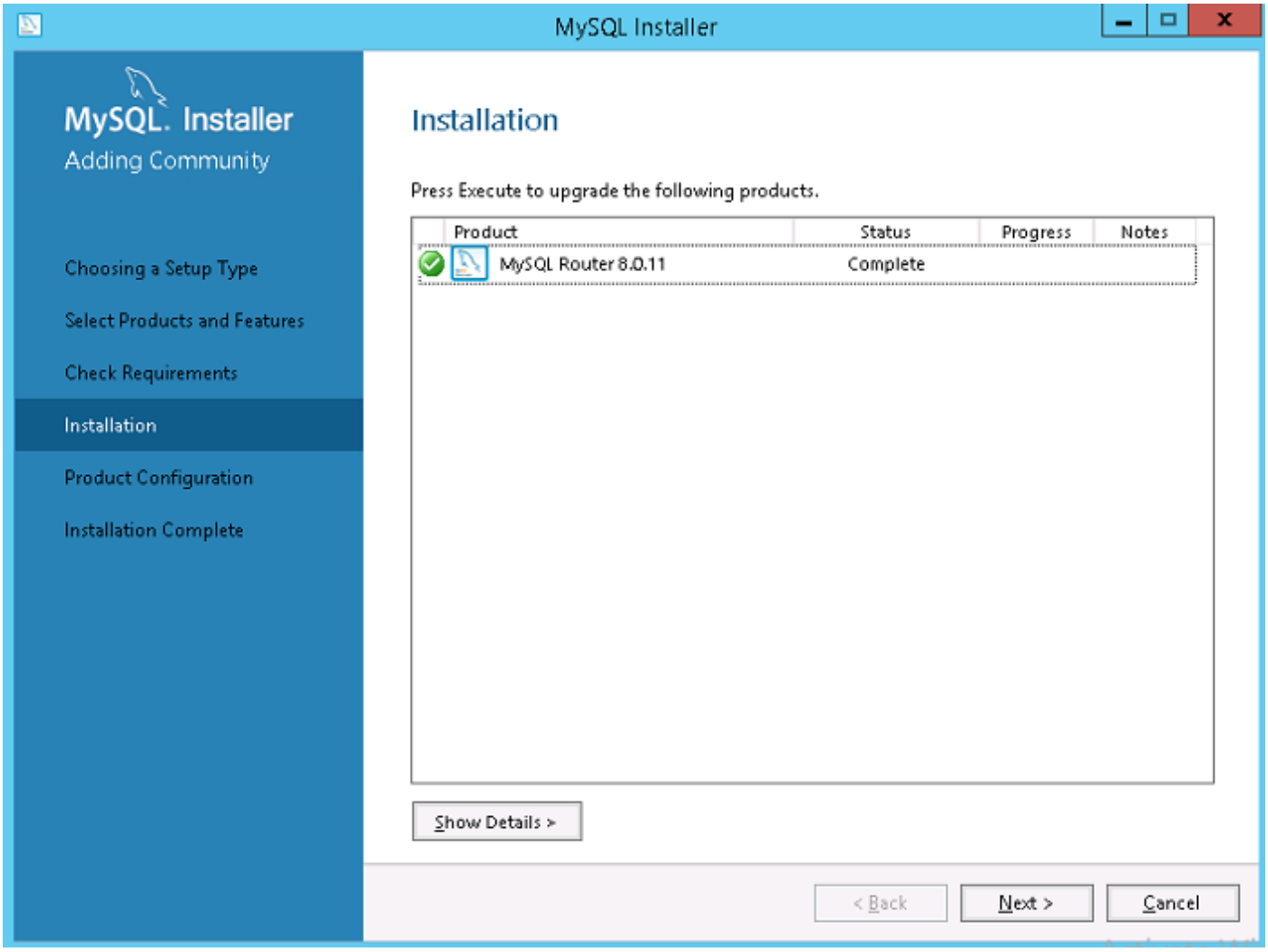

Figure 46. Installation

7. On the **Product Configuration** screen, the MySQL router component is displayed.

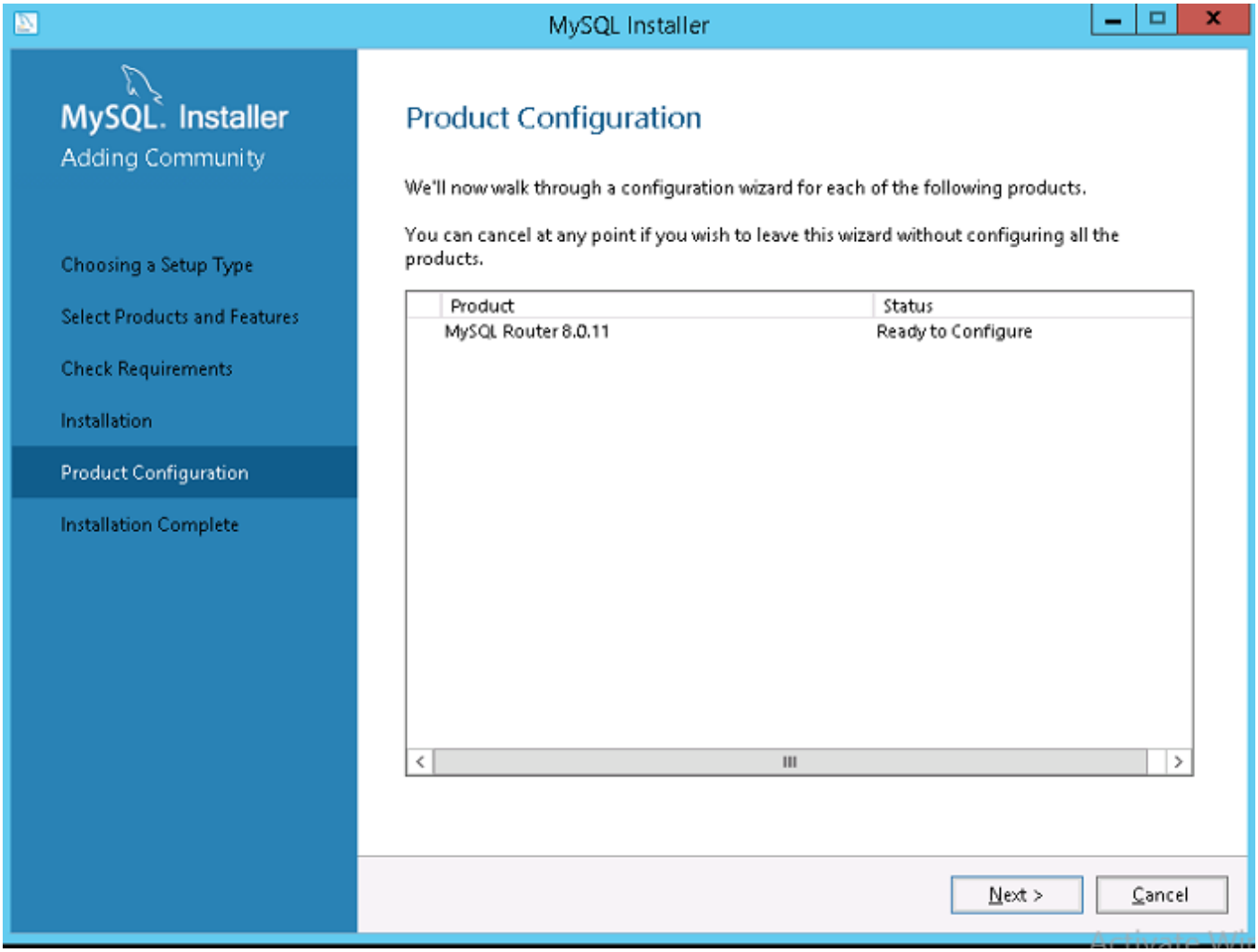

Figure 47. Product configuration

8. Click **Next** to configure the MySQL router component.
9. On the **MySQL Router Configuration** screen, enter the hostname, port number, management user, and password.

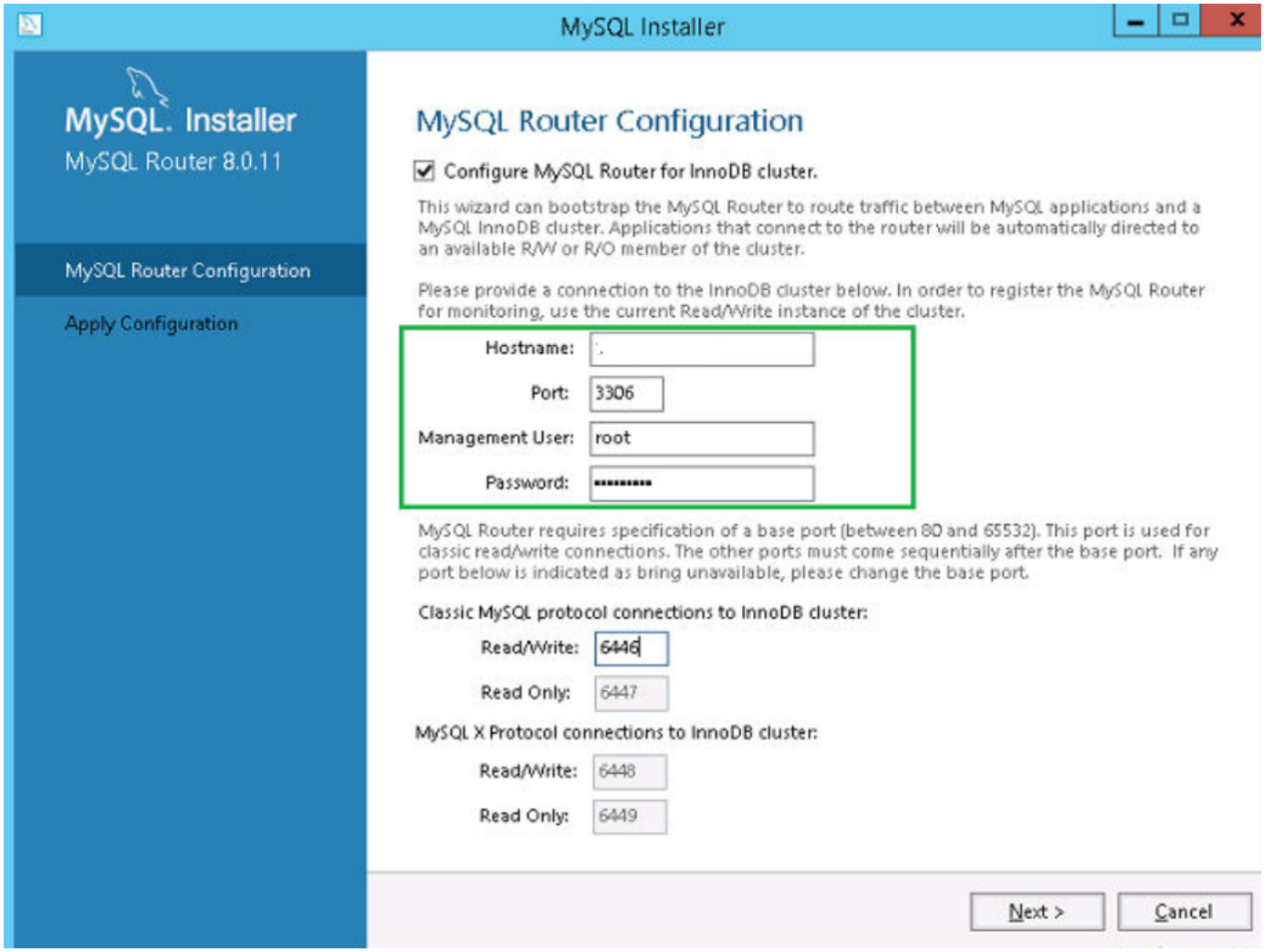

Figure 48. MySQL Router Configuration

10. On the **Apply Configuration** screen, click **Execute**.

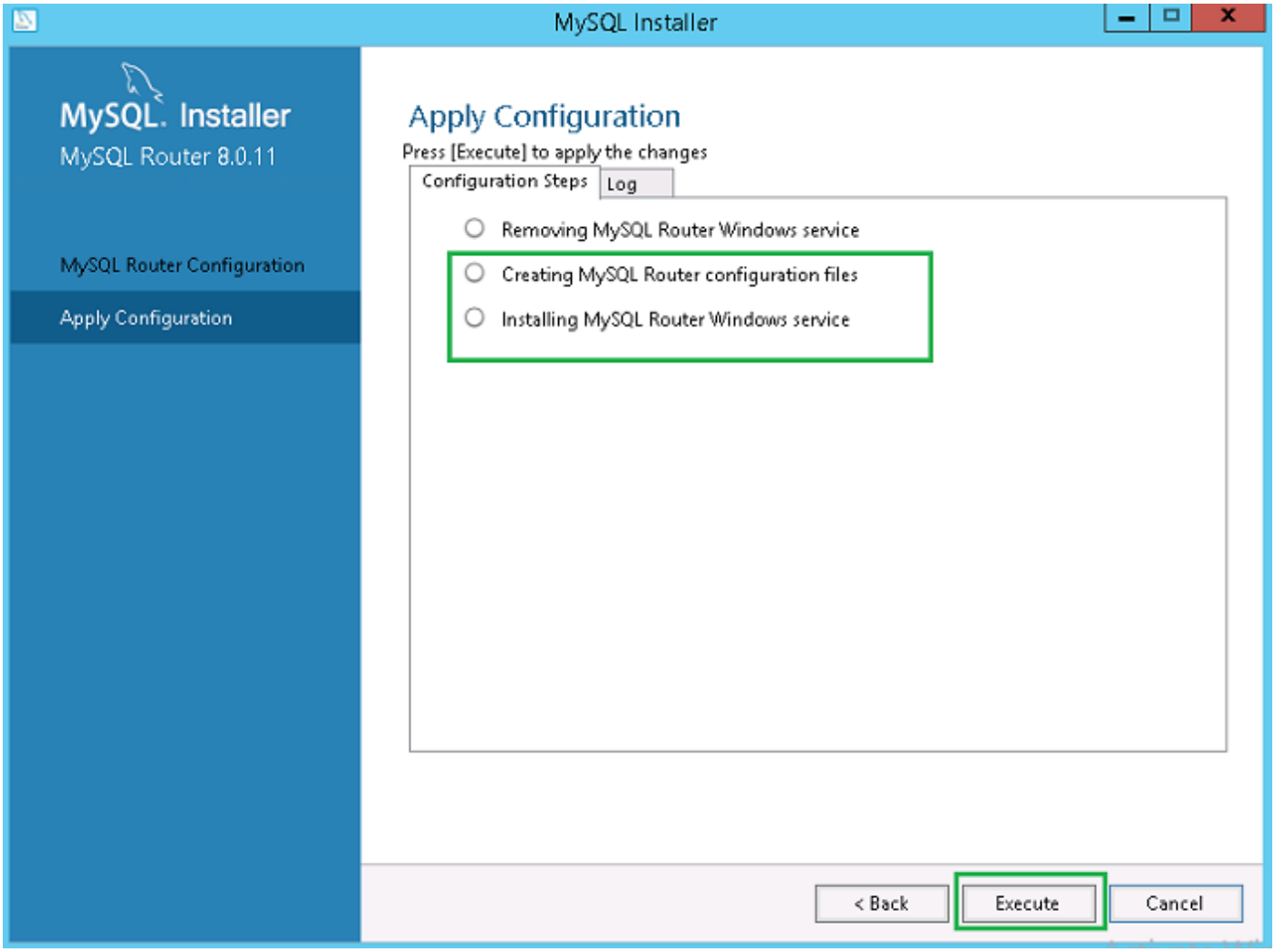

Figure 49. Apply configuration

11. Click **Finish**.

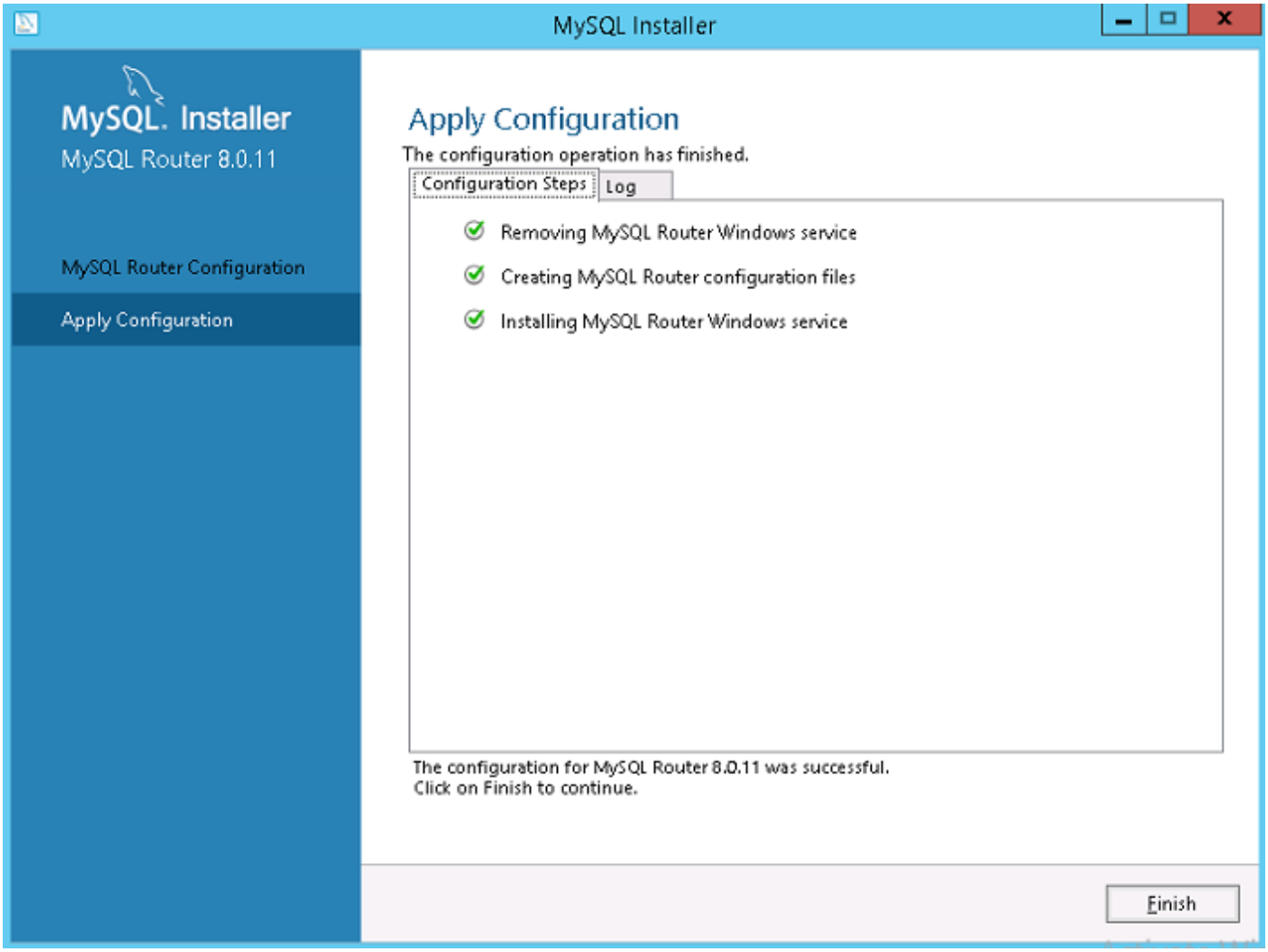

Figure 50. Apply configurations

12. On the **Product Configuration** screen, click **Next**.

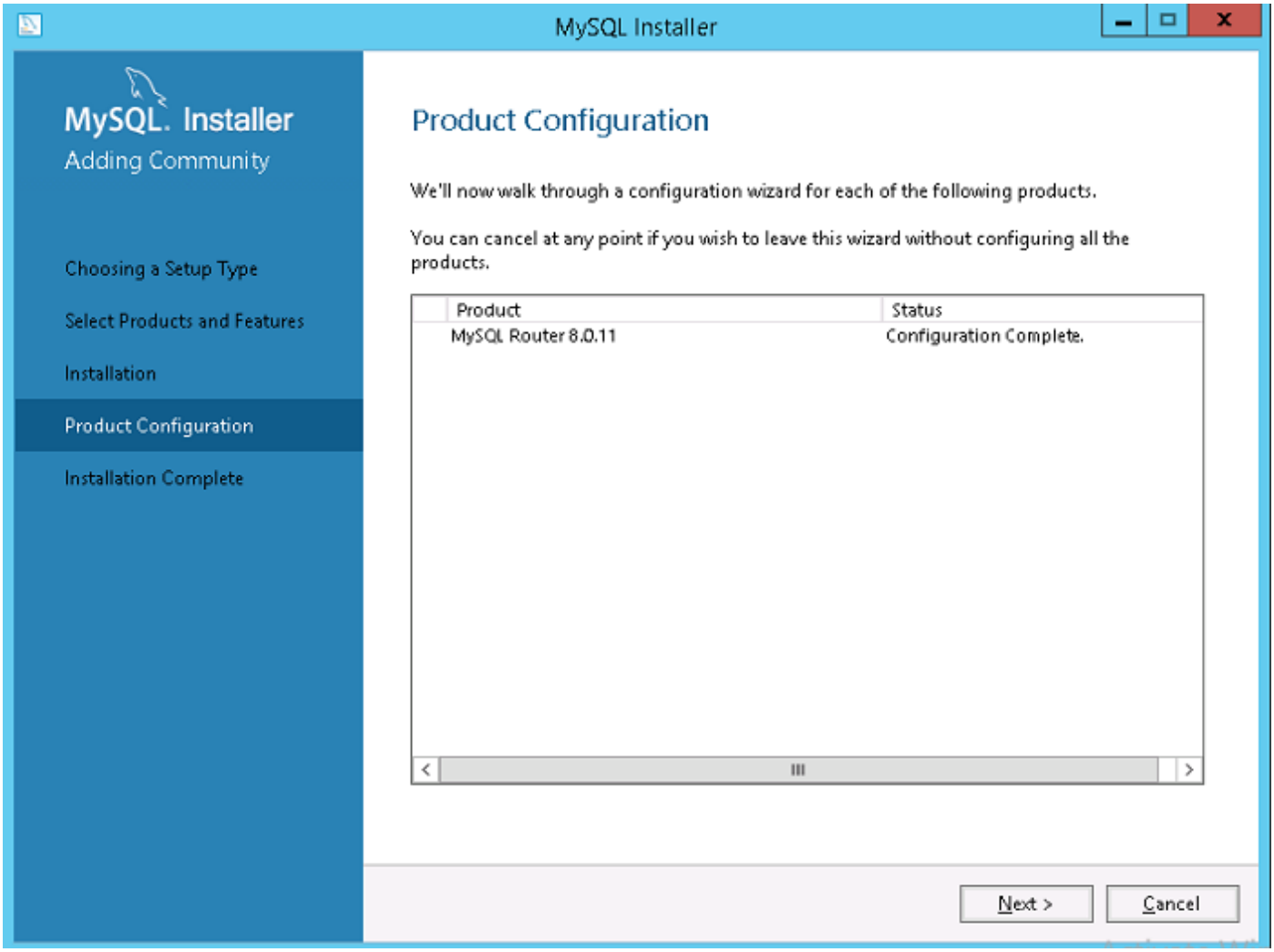

Figure 51. Product configuration

The **Installation Complete** message is displayed.

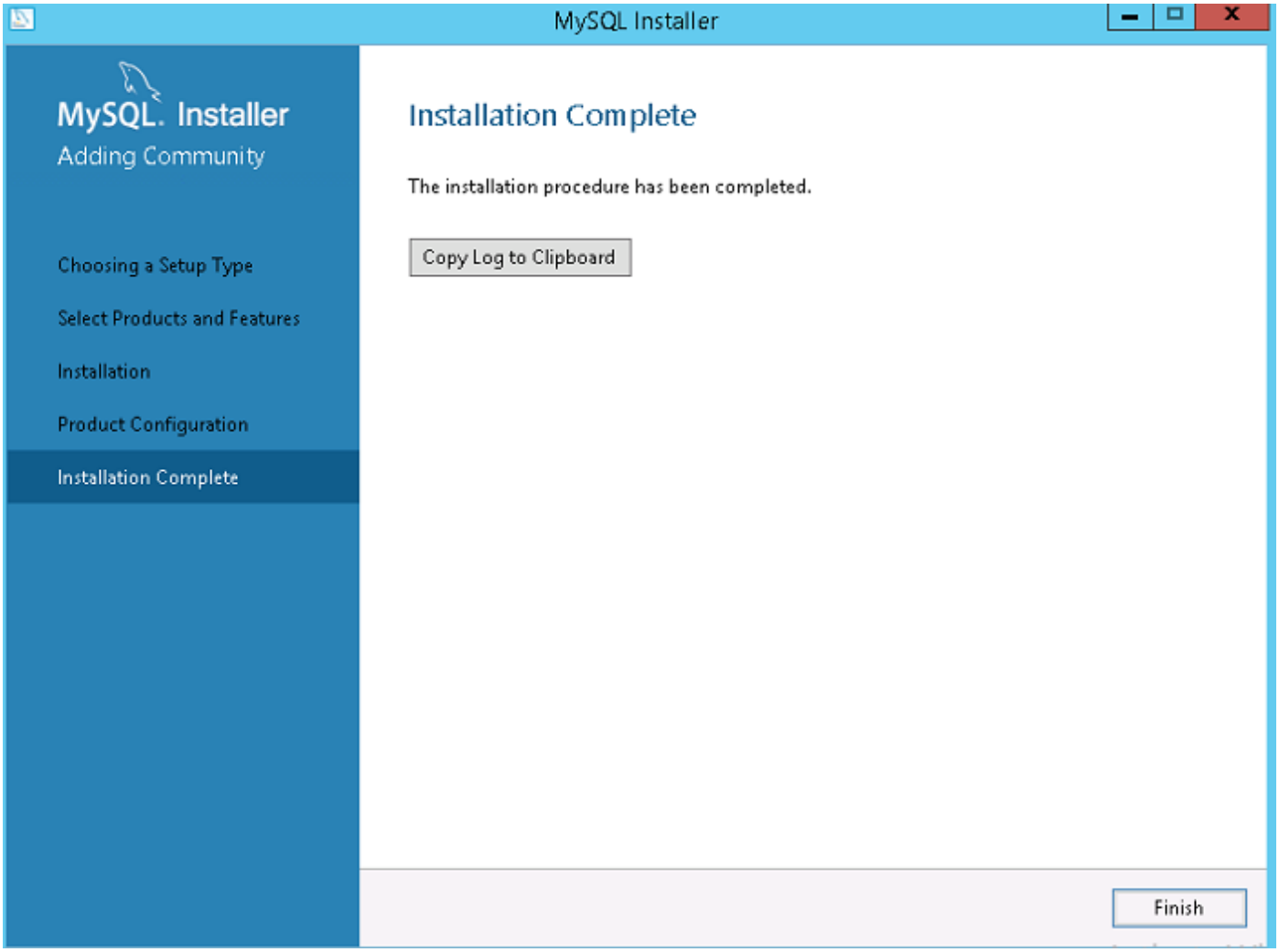

Figure 52. Installation complete

13. Click **Finish**.
14. Browse to `\ProgramData\MySQL\MySQL Router` directory, and open the file `mysqlrouter.conf` to check that the bootstrap property with all the configured MySQL servers are part of cluster setup.

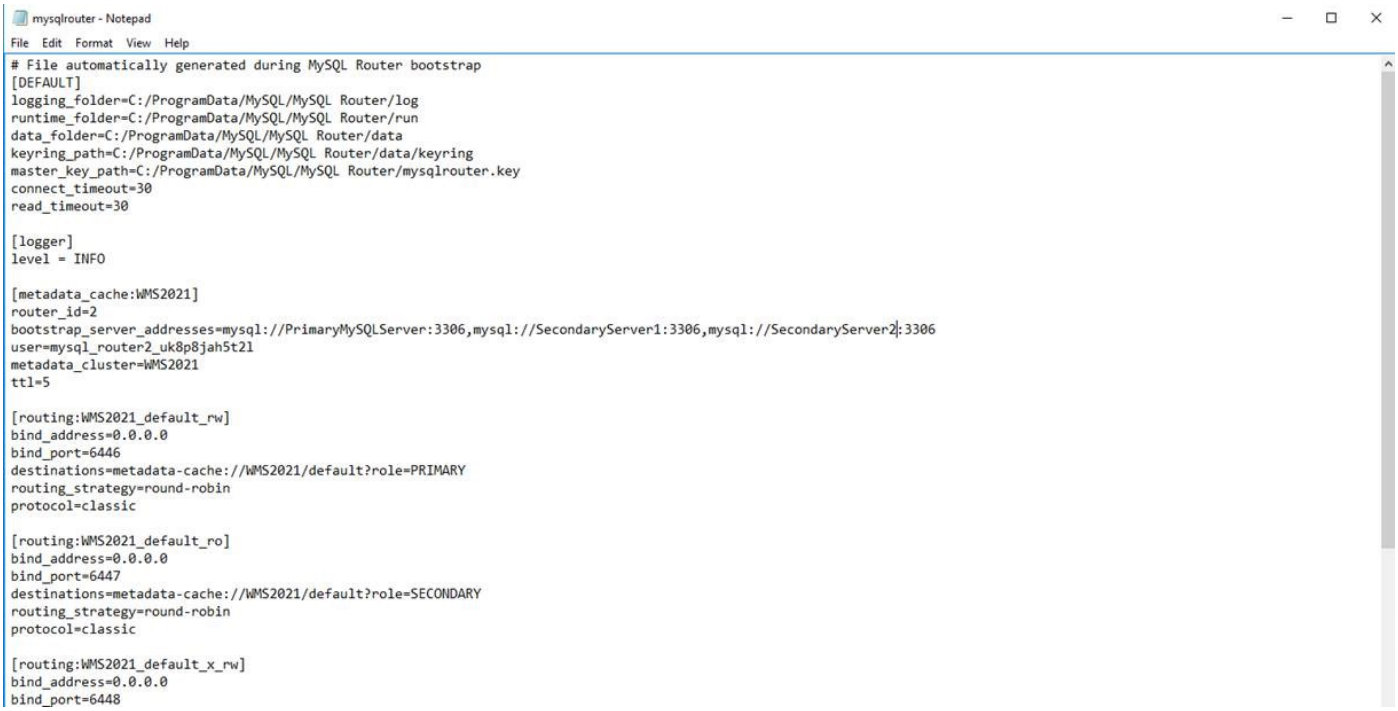

Figure 53. Bootstrap server address

Create database and users on MySQL InnoDB server

You must create the database and user accounts with administrator privileges on MySQL InnoDB server.

About this task

To create database on MySQL InnoDB server, run the following SQL commands:

```
Create Database stratus DEFAULT CHARACTER SET utf8 DEFAULT COLLATE utf8_unicode_ci;
CREATE USER 'STRATUS'@'LOCALHOST';
CREATE USER 'STRATUS'@'IP ADDRESS';
SET PASSWORD FOR 'STRATUS'@'LOCALHOST' = PASSWORD <db_password>;
SET PASSWORD FOR 'STRATUS'@ <IP_Address> = PASSWORD <db_password>;
GRANT ALL PRIVILEGES ON *.* TO 'STRATUS'@<IP_Address> IDENTIFIED BY <db_password> WITH GRANT
OPTION;
GRANT ALL PRIVILEGES ON *.* TO 'STRATUS'@'LOCALHOST' IDENTIFIED BY <db_password> WITH GRANT
OPTION;
```

NOTE: Instead of IP Address, you can type the Wildcard for Network /Subnet or Multiple Single host entry where Wyse Management Suite application server will be installed.

6

Achieve high availability on MongoDB

About this task

The following steps explain how to achieve high availability on MongoDB:

Steps

1. Install MongoDB—see Installing MongoDB.
2. Create replica servers—see Creating Replica servers.
3. Create Stratus users—see Creating Stratus user account.
4. Create root user—see Creating root user for MongoDB.
5. Edit MongoDB configuration file—see Editing MongoDB configuration file.

Install MongoDB

About this task

To install MongoDB on all the three nodes, do the following:

NOTE: For information on installing MongoDB see—Install MongoDB

Steps

1. Copy the MongoDB installation files on a system.
2. Create two folders `Data\log` and `data\db` on a secondary drive other than `Drive C`.

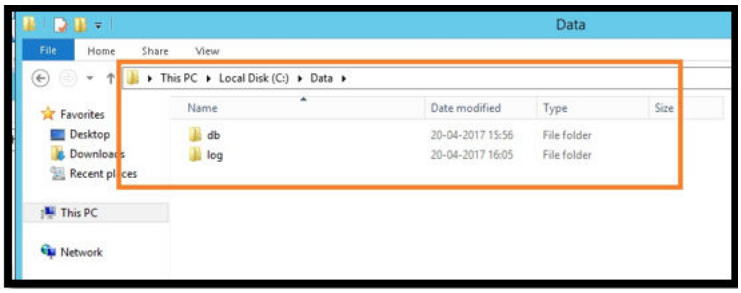

Figure 54. Data files

3. Go to the folder where you have copied the MongoDB installation files, and create a file `mongod.cfg` from the command prompt.

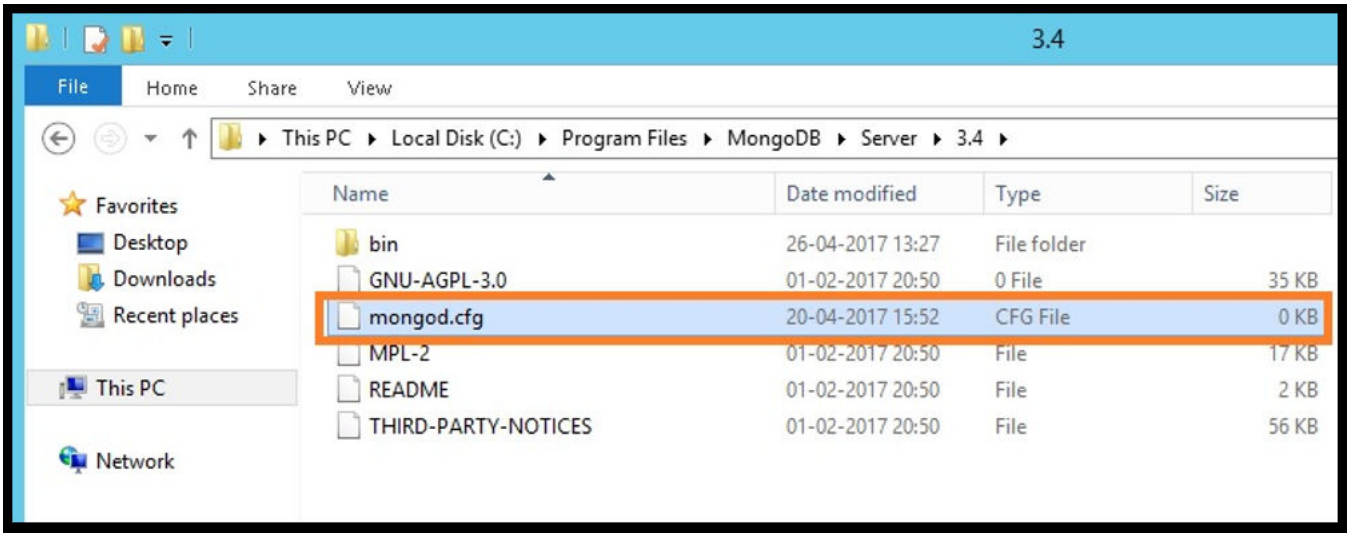

Figure 55. mongod.cfg file

4. Open the `mongod.cfg` file in a text editor, and add:

```
systemLog:
    destination: file
    path: c:\data\log\mongod.log
storage:
    dbPath: c:\data\db
```

5. Save the file.
6. Open command prompt.
7. Run the following command to start the MongoDB service:
   a. `C:\MongoDB\bin>.\mongod.exe --config c:\Mongodb\mongod.cfg --install`
   b. `C:\MongoDB\bin>net start mongodb`

   The message **MongoDB service is starting** is displayed.
8. Change the working directory to `\MongoDB\bin`.
9. Run `Mongo.exe` at the command prompt to complete the MongoDB installation.

Create replica servers for MongoDB database

You must create replica servers to avoid any system failures. The replica servers should have the capacity to store multiple distributed read operations.

For more information to create replica servers, see Deploy a Replica Server Set at docs.mongodb.com/manual.

Create stratus user

Create an user, for example, StratusUser using the Wyse Management Suite to access MongoDB.

NOTE: The stratus user and password are examples and can be created using a different name and password at your work place.

Run the following command to create the StratusUser:

```
db.createUser({
user: "stratus",
pwd: <db_password>,
roles: [ { role: "userAdminAnyDatabase", db: "admin" },
{ role: "dbAdminAnyDatabase", db: "admin" },
{ role: "readWriteAnyDatabase", db: "admin" },
{ role: "dbOwner", db: "DBUser" }]})
```

Create database user

Create an user, for example, DBUser using the Wyse Management Suite to access MongoDB.

NOTE: The database user and password are examples and can be created using a different name and password at your work place.

Run the following command to create the DBUser:

```
db.createUser({
user: "DBUser",
pwd: <db_password>,
roles: [ { role: "userAdminAnyDatabase", db: "admin" },
{ role: "dbAdminAnyDatabase", db: "admin" },
{ role: "readWriteAnyDatabase", db: "admin" },
{ role: "dbOwner", db: "DBUser" }]
})
```

Create DBadmin user for MongoDB

Login to the MongoDB using the user account created in the previous section. The DBadmin user is created with the administrative privileges.

Run the following command to create the DBadmin user:

```
mongo -uDBUser -pPassword admin
use admin
db.createUser(  {
user: "DBadmin",
pwd: <DBadmin user password>,
roles: [ { role: "DBadmin", db: "admin" } ]
})
```

Edit mongod.cfg file

You must edit the `mongod.cfg` file to enable the security for the MongoDB database.

1. Login to MongoDB as root user that you have already created and run the following command:

   ```
   mongo -uroot -<root password> admin
   ```

2. Go to `\data\bin\mongod.cfg` directory, and open `mongod.cfg` file in a text editor.
3. Edit `mongod.cfg` file as shown in the following command:

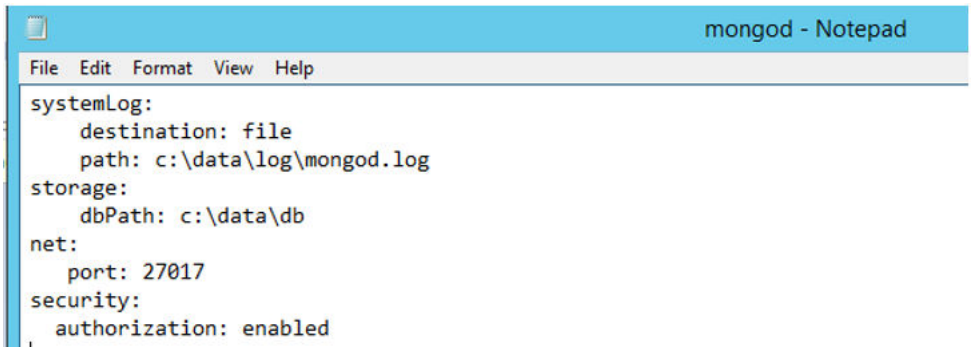

Figure 56. Edit mongod.cfg

```
systemLog:
destination: file
path: c:\data\log\mongod.log
storage:
dbPath: c:\data\db\Mongo
```

```
net:
bindIp: x.x.x.x, 0.0.0.0
port: 27017
security:
authorization: enabled
```

NOTE: The port numbers will change depending on the system at the work place.

4. Save `mongod.cfg` and exit.

Initiate replication on the servers

Ensure that you disable firewall on Windows and stop Tomcat servers if they are running.

1. Login to MongoDB as root user that you have already created and run the following command:

```
mongo -uroot -<root password> admin
```

2. Go to `\data\bin\mongod.cfg` directory, and open `mongod.cfg` file in a text editor.
3. Add the following three lines in the `mongod.cfg` file:

```
systemLog:
    destination: file
    path: c:\data\log\mongod.log
storage:
    dbPath: c:\data\db\Mongo
net:
    bindIp: X.X.X.X, 0.0.0.0
    port: 27017
security:
  authorization: enabled
  keyFile: c:\data\log\mongod.key.txt
replication:
   replSetName: wms
```

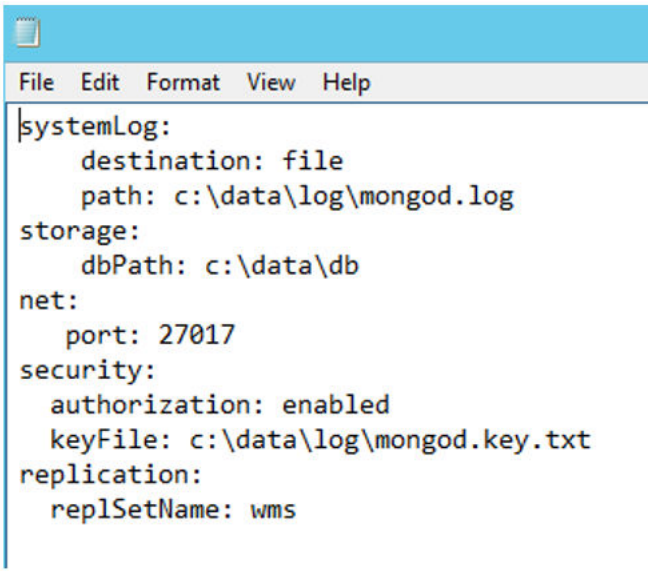

Figure 57. Enabling security

4. Create `mongod.key.txt` file and copy on all the three servers.

NOTE: Ensure that the `mongod.key.txt` file content or key is the same in all the three servers.

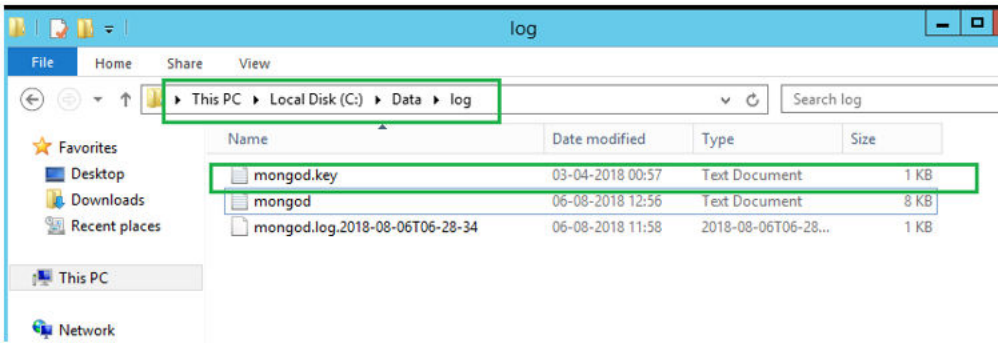

Figure 58. Copy the mongod key file

5. After you copy the file, stop the mongod service by running the following command:

   ```
   net stop mongodb
   ```

6. Start the mongod service by running the following command:

   ```
   net start mongodb
   ```

7. Repeat the steps from 1 to 6 in all the three nodes of MongoDB servers.
8. Initiate replication on the primary node of the MongoDB cluster logging in using DBadmin user and then run the following command:

   ```
   rs.initiate();
   ```

   ```
   C:\Mongo\bin>mongo.exe -u root -p x` admin
   MongoDB shell version v4.2.1
   connecting to: mongodb://127.0.0.1:27017/admin?
   compressors=disabled&gssapiServiceName=mongodb
   Implicit session: session { "id" : UUID("952f322c-1eb4-46c4-9b5e-bd536e2c1e7e") }
   MongoDB server version: 4.2.1
   MongoDB Enterprise > use admin
   switched to db admin
   MongoDB Enterprise >
   MongoDB Enterprise >
   MongoDB Enterprise > rs.initiate();
   {
   "info2" : "no configuration specified. Using a default configuration for the set",
   "me" : "10.150.132.37:27017",
   "ok" : 1
   }
   ```

9. Check the replication status by running the following command:

   ```
   rs.status();
   ```

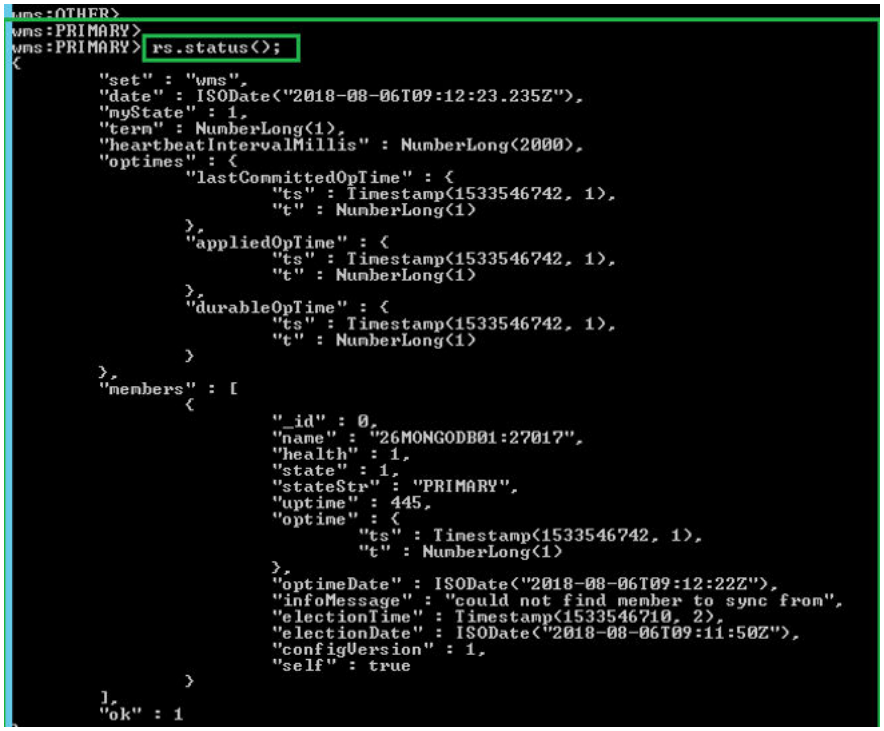

Figure 59. Replication status

10. Start mongod service and add the secondary nodes to the second and third nodes in the MongoDB cluster:

```
rs.add("IPAddress2:27017")
```

```
rs.add("IPAddress3:27017")
```

```
MongoDB Enterprise wms20:PRIMARY> rs.add("10.150.132.36:27017")
{
"ok" : 1,
"$clusterTime" : {
"clusterTime" : Timestamp(1579600528, 1),
"signature" : {
"hash" : BinData(0,"8N3uoZ5khebgbY+PsFxJZvMaI1g="),
"keyId" : NumberLong("6784332217662308354")
}
},
"operationTime" : Timestamp(1579600528, 1)
}
```

NOTE: The port numbers will differ based on the systems at your network and systems.

11. After you add the nodes in the MongoDB cluster, check the replication status by running the following commands for the primary and secondary nodes:

```
rs.status();
```

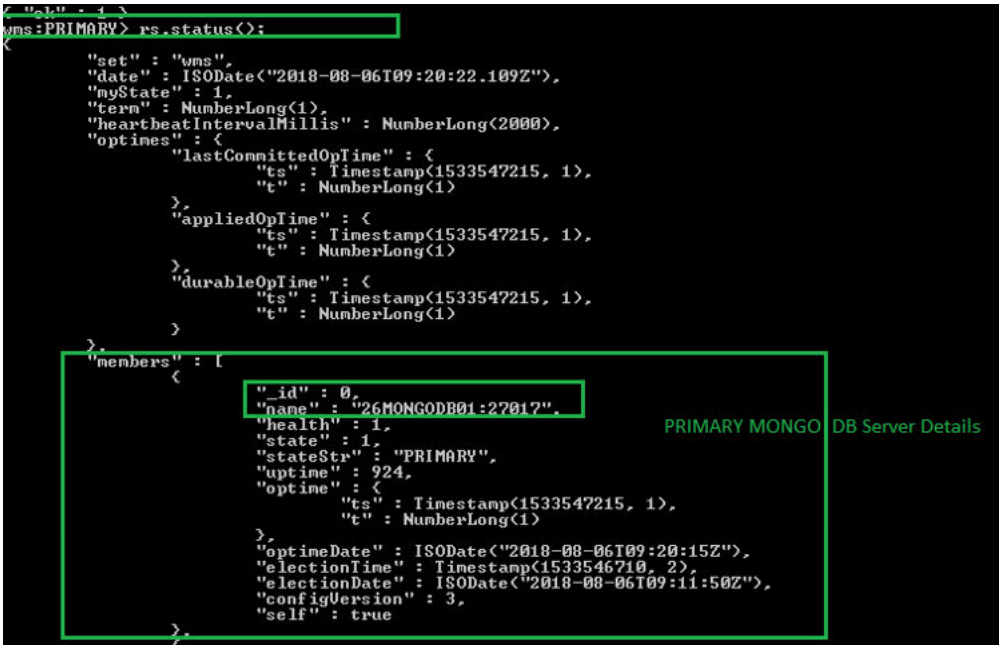

Figure 60. Status in primary server

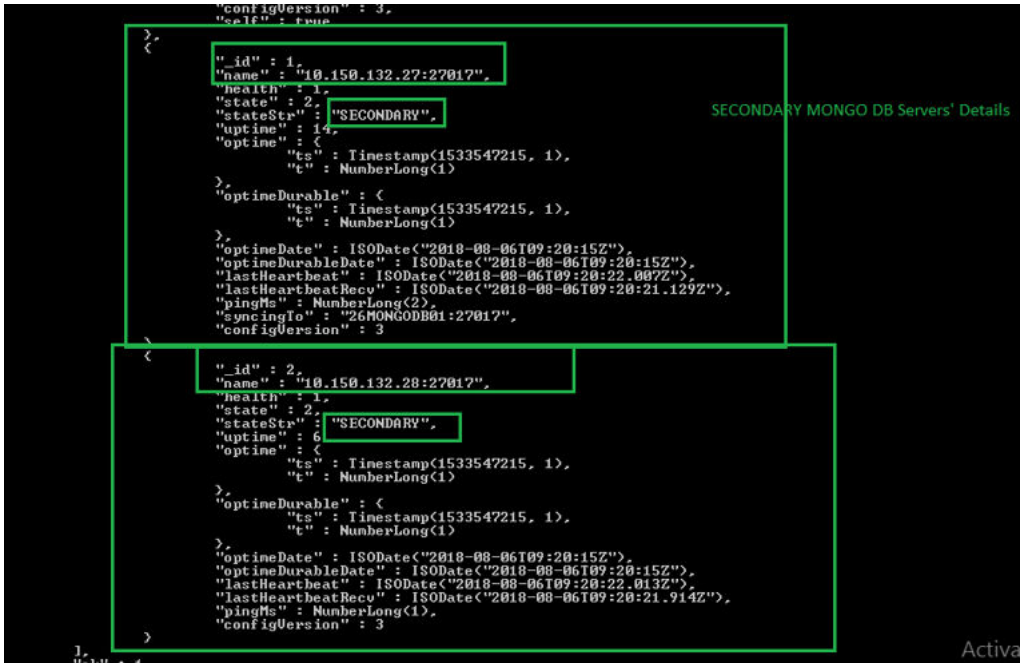

Figure 61. Secondary server status

7

Achieve high availability for Teradici devices

Wyse Management Suite uses the HAProxy hosted on the Ubuntu server 16.04.1 LTS to perform load balancing between the EMSDK servers. HAProxy is a load balancer proxy that can also provide high availability based on how it is configured. It is a popular open source software for TCP/HTTP Load Balancer, and proxy solution which runs on Linux operating system. The most common use is to improve the performance and reliability of a server environment by distributing the workload across multiple servers.

About this task

The following points explains how to achieve high availability for Teradici devices using HAProxy on Linux operating system:

- There will be only one instance of Teradici server as part of high availability with Wyse Management Suite.
- Teradici device support requires installation of EMSDK. EMSDK is a software component provided by Teradici that is integrated into Wyse Management Suite. Wyse Management Suite Installer installs EMSDK can be installed on Wyse Management Suite server or on a separate server. You need minimum of two instances of EMSDK to support more than 5000 devices, and all EMSDK servers should be on remote servers.
- Only one instance of EMSDK can be installer per server.
- Teradici Device support requires a PRO license.
- High availability of Teradici will be provided through HAProxy.
- If Teradici server goes down, device will reconnect automatically to the next available EMSDK server.

Install and configure HAProxy

About this task

HAProxy which is the load balancer for ThreadX 5x devices is configured on Ubuntu Linux version 16.04.1 with HAproxy version 1.6.

Do the following to install and configure HAProxy on Ubuntu Linux system:

1. Log in to Ubuntu system using the user credentials used during the installation of Ubuntu operating system.
2. Run the following commands to install HAProxy

   ```
   sudo apt-get install software-properties-common
   sudo add-apt-repository ppa:vbernat/haproxy-1.6
   sudo apt-get update
   sudo apt-get install haproxy
   ```
3. Run the following command to take backup of the original configuration:

   ```
   sudo cp /etc/haproxy/haproxy.cfg /etc/haproxy/ haproxy.cfg.original
   ```
4. Edit the HAProxy configuration file in a suitable text editor by running the following commands:

   ```
   sudo nano /etc/haproxy/haproxy.cfg
   ```

   Add the following entries in the configuration file:

   ```
   Global section: Maxconn <maximum number of connections>
   Frontend tcp-in: bind :5172
   Back end servers: server :5172
   maxconn <maximum number of connections per Teradici device proxy server>
   ```

   NOTE: Administrator must add additional back end servers beyond the total number of client's capacity to have seamless failover.
5. Save the changes to the `haproxy.cfg` file by typing CTRL+O.

The following text is a sample HAProxy configuration file:

```
global
        log /dev/log    local0
        log /dev/log    local1 notice
        chroot /var/lib/haproxy
        daemon
        #maxconn is maximum allowed connections
        maxconn 60000
defaults
        log     global
        mode    tcp
        timeout connect 5000ms
        timeout client  50000ms
        timeout server  50000ms
        errorfile 400 /etc/haproxy/errors/400.http
        errorfile 403 /etc/haproxy/errors/403.http
        errorfile 408 /etc/haproxy/errors/408.http
        errorfile 500 /etc/haproxy/errors/500.http
        errorfile 502 /etc/haproxy/errors/502.http
        errorfile 503 /etc/haproxy/errors/503.http
        errorfile 504 /etc/haproxy/errors/504.http

frontend fe_teradici_5172
  bind :5172
  mode tcp
  backlog 4096
  maxconn 70000
  default_backend be_teradici_5172

backend be_teradici_5172
  mode  tcp
  option log-health-checks
  option tcplog
  balance leastconn
  server emsdk1 :5172 check server emsdk2 5172 check :  timeout queue 5s timeout server
86400s
  option srvtcpka

#frontend fe_teradici_5172
#replace IP with IP of your Linux proxy machine bind Eg:  10.150.105.119:5172

#default_backend servers

#backend servers
#Add your multiple back end windows machine ip with 5172 as port
# maxconn represents number of connection- replace 10 with limit #(below 20000)
# "server1" "server2" are just names and not keywords

#server server1 10.150.105.121:5172 maxconn 20000 check
#server server2 10.150.105.124:5172 maxconn 20000 check
```

6. Validate the HAProxy configuration by running the following command:

   `sudo haproxy -f /etc/haproxy/haproxy.cfg -c`

   If the confiration is valid, the message `Configuration is Valid` is displayed.
7. Restart HAProxy servce by running the following command:

   `Sudo service haproxy restart`
8. Stop HAProxy by running the following command:

   `serviceSudo service haproxy stop`

8

Install Wyse Management Suite on Windows Server 2012 R2/2016/2019

Prerequisites

Ensure that the following servers are configured before installation of Wyse Management Suite application:

- Windows Fail over Cluster on Two Nodes
- MongoDB Server Running with replica set
- MySQL Server InnoDB Cluster up running
- MySQL Router installed on the two Nodes
- Install Visual C++ 2015 or 2017 Redistributable package (x64) or later versions. Wyse Management Suite installer requires VCRUNTIME140.dll file to connect with MongoDB replica set or stand-alone setup with version 4.2.1.

About this task

Installation of Wyse Management Suite 1.3 or higher on both the Nodes in Windows Cluster

Steps

1. Launch the Wyse Management Suite installer.

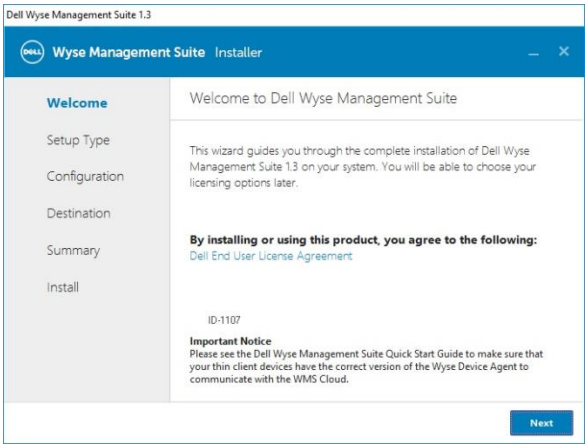

Figure 62. Welcome screen

2. Select Custom type installation.

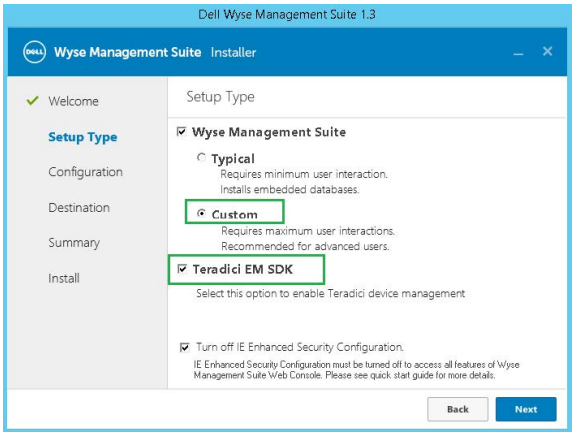

Figure 63. Setup type

3. Select the External Remote Mongo database option (MongoDB Cluster with Replica set created). Ensure to provide the remote primary Mongo DB server information and port number; and Mongo DB username and password.

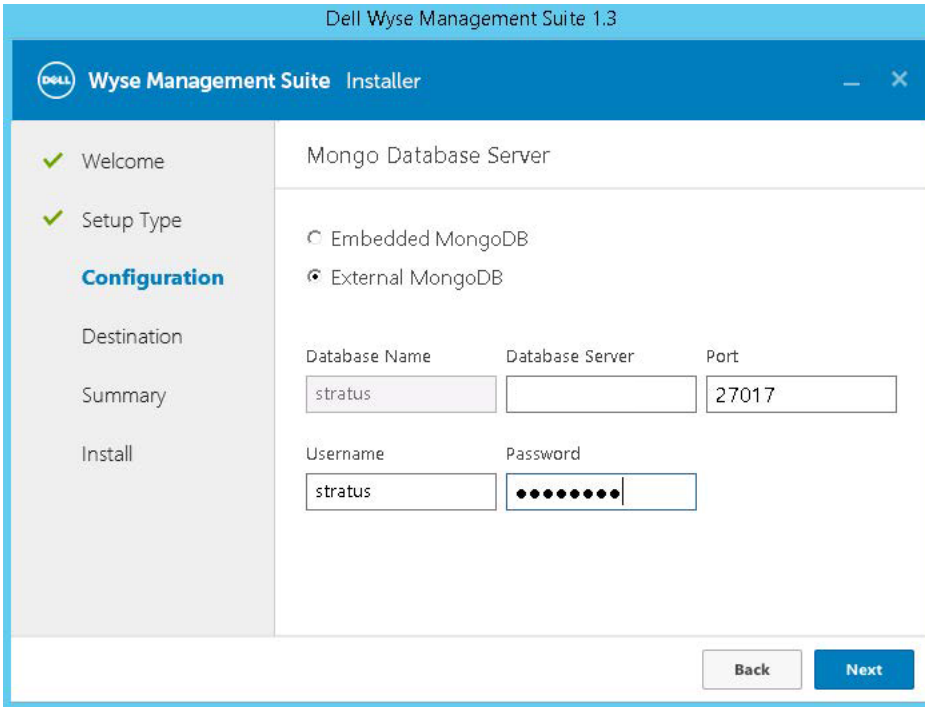

Figure 64. Configuration

4. Select the **External MariaDB** option for MySQL. Provide MySQL router address (Local Host if it is installed on Wyse Management Suite server node) in the **External Maria DB Server** fields with the port number(Default 6446). You must type the MySQL database user account information that was created initially.

   NOTE: Ensure that the "Stratus" Database is created and "DB User" account(stratus) with appropriate Privileges is created on MySQL server.

   The following commands are to be started in the Primary Node or R/W MySQL DB Server:

   a. Open command prompt with Admin mode, go to "C:\Program Files\MariaDB 10.0\bin>" and start command, "C:\Program Files \MariaDB 10.0\bin>mysql.exe -u root –p"
   b. Provide the root password which was created during My SQL server installation to log in into DB server.

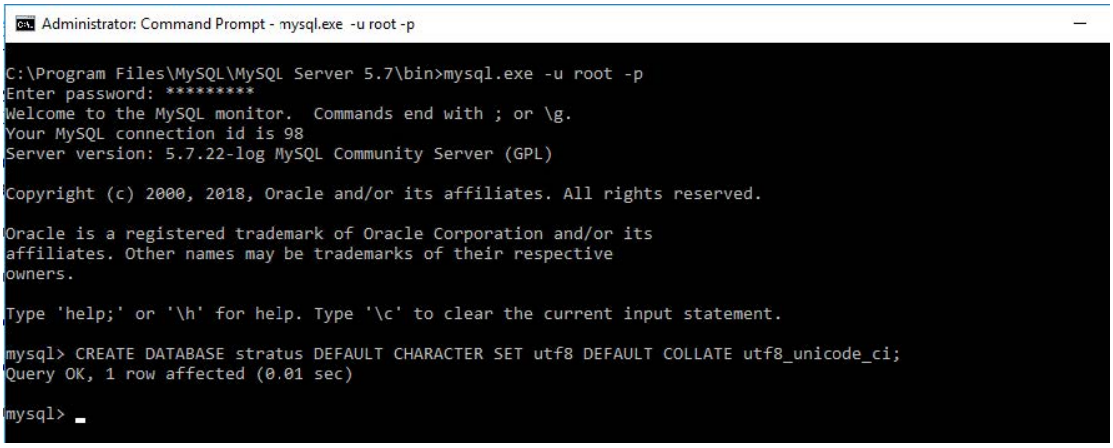

Figure 65. Root password

c. Execute the command, `CREATE DATABASE stratus DEFAULT CHARACTER SET utf8 DEFAULT COLLATE utf8_unicode_ci` to create DB.

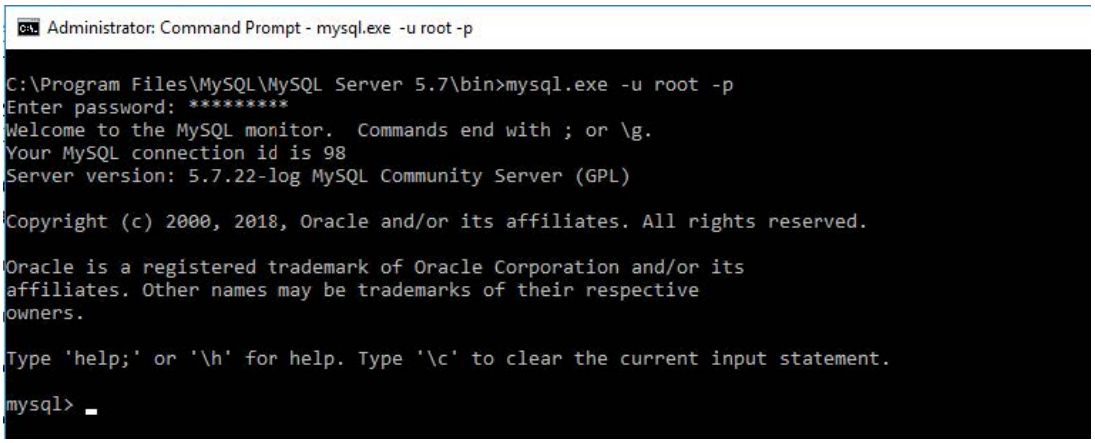

Figure 66. Database command

d. Execute following commands to create and Stratus User account and privileges:

- Create user 'stratus'@'localhost'
- Create user 'stratus'@'10.150.132.21'
- Set password for 'stratus'@'localhost' = password ('PASSWORD')
- Set password for 'stratus'@'IP ADDRESS'= password ('PASSWORD')
- Grant all privileges on *.* to 'stratus'@'IP ADDRESS' identified by 'PASSWORD' with grant option.
- Grant all privileges on *.* to 'stratus'@'localhost' identified by 'PASSWORD' with grant option.

e. Provide MySQL router information in the External Maria DB Server fields with port number and MySQL DB user account information.

NOTE: The above commands can be started through the MySQL workbench for creating users and privileges with wildcards.

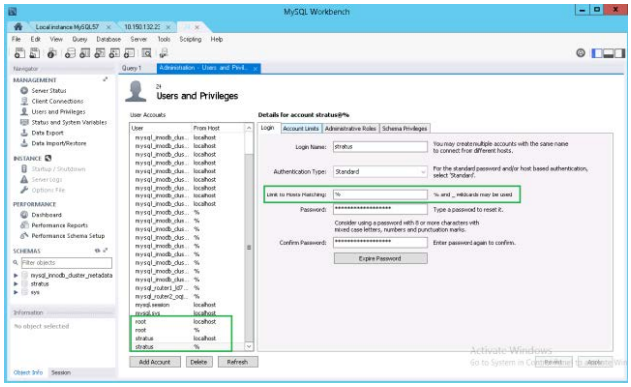

Figure 67. MySQL workbench

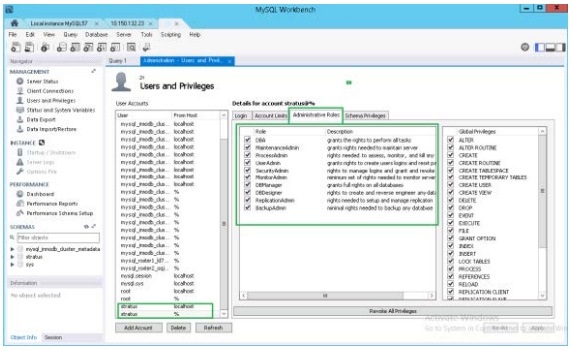

Figure 68. MySQL workbench

5. Provide ports information for Wyse Management Suite related Services in “Port Selection” window.

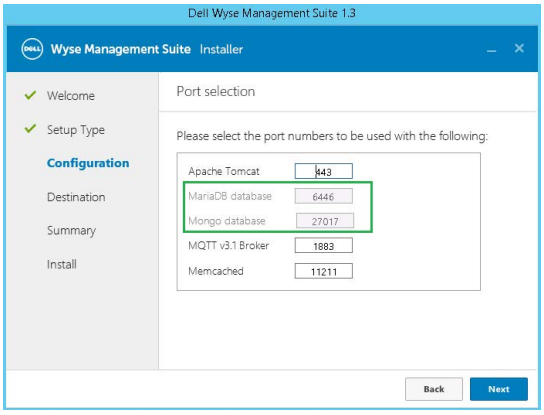

Figure 69. Configuration

6. Provide administrator credentials and email address information.

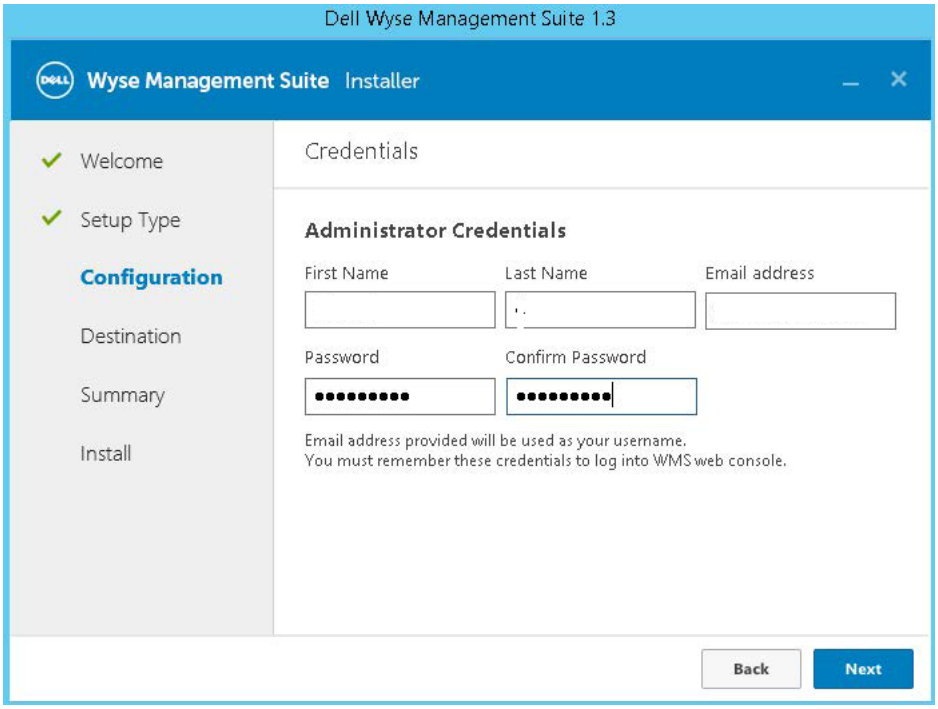

Figure 70. Configuration

7. Provide Teradici EM SDK Port information and CIFS User Account information.

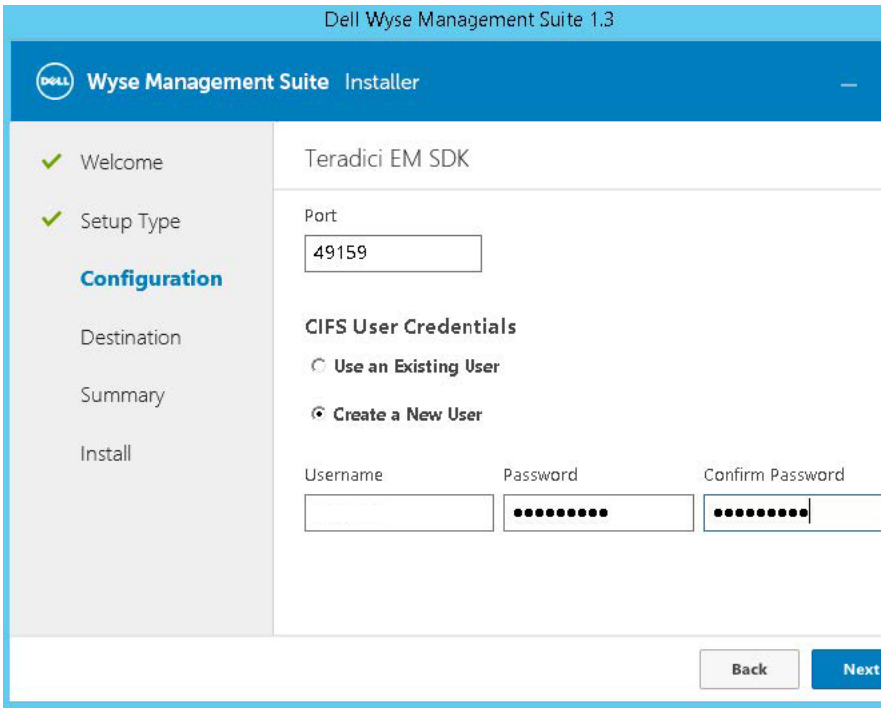

Figure 71. Teradici EM SDK

8. Provide 'Destination Installation folder path' and 'Shared UNC path' for Local repository.

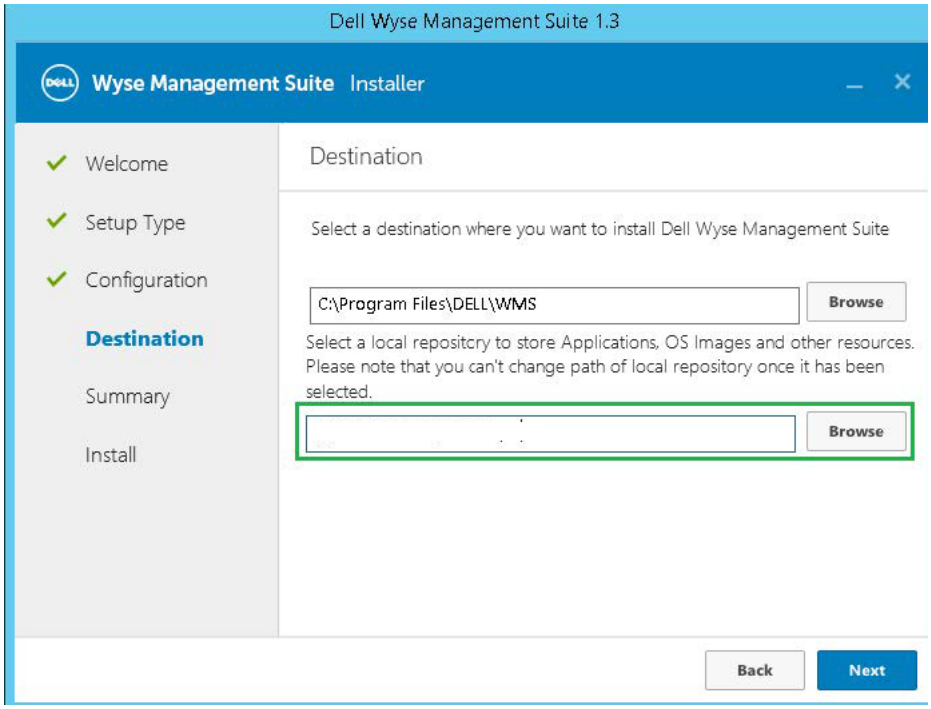

Figure 72. Destination

9. Recheck the Installation Summary information before we proceed with the Wyse Management Suite installation.
10. Complete the Installation on both the nodes.

    Type the Destination Installation folder path and Shared UNC path for the local repository and then click **Next**. The message **The installation was successful** is displayed.

    NOTE: The shared UNC path should be kept out of both the Windows Server where Wyse Management Suite application is installed. Before you install Wyse Management Suite application on Node 2, ensure to delete the 'Data' folder present in the Wyse Management Suite Local Repository; which was created during installation on Node 1. After 'Data' folder is deleted from the shared UNC WMS Local Repository path, you can install Wyse Management Suite Application in the Node 2 of the Windows Cluster.

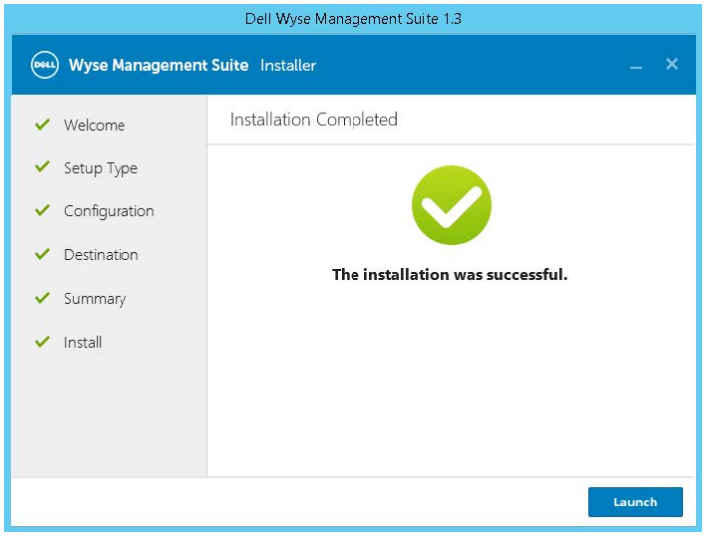

Figure 73. Installation successful

9

Post installation checks

About this task

Do the following to check the high availability for Wyse Management Suite:

- Launch the Wyse Management Suite administrator portal and check whether you can log in using the web interface.
- Edit the `bootstrap.properties` file in the Tomcat server under the `\Dell\WMS\Tomcat-9\webapps\ccm-web\WEB-INF\classes` folder for MongoDB as follows:

```
mongodb.seedList = MongoDBServer1_IP:27017, MongoDBServer2_IP:27017,
MongoDBServer3_IP:27017
```

Do the following to make changes in the MongoDB and MySQL DB tables:

1. Log in to Mongo DB using Robo 3T and update **Windows Cluster Virtual IP/Hostname of Access Point** values in the **bootstrapProperties** table with the following attributes:
   - Stratusapp.server.url
   - Stratus.external.mqtt.url
   - Memcached. Servers
   - Mqtt.server.url
2. Update the MySQL tables and restart the Tomcat on both the nodes. Manually update `mysql` database table to retain the `ServerIp` in the `ServersInCluster` table to be active by running the following command:

   ```
   Update serversInCluster set ServerIp = '<VIP address of Windows Cluster>';
   ```

   NOTE: Ensure that there is only one record in `serversInCluster` table and if there are more than one record, delete the excess records.

   ```
   Update queuelock set IpInLock = '<VIP address of Windows Cluster>';
   ```
3. Connect the FQDN address of the access point to the **Memcached** registry on both nodes of the high availability setup using the following paths:
   - Registry path—HKLM\SYSTEM\CurrentControlSet\Services\Memcached\
   - Image path—C:\Program Files\DELL\WMS\memcached\memcached.exe -d runservice -p 11211-l <FQDN of Access Point> -U 0

10

Upgrade Wyse Management Suite version 1.3 to 1.4

Prerequisites

- Ensure that the `mongodb.seedList` value in the `bootstrap.properties` file includes backslash character (\) in the list of Mongo database servers. The `bootstrap.properties` file is at `Tomcat-9\webapps\ccm-web\WEB-INF\classes`, `mongodb.seedList = MongoDBServer1_IP\:27017, MongoDBServer2_IP\:27017, MongoDBServer3_IP \:27017`.

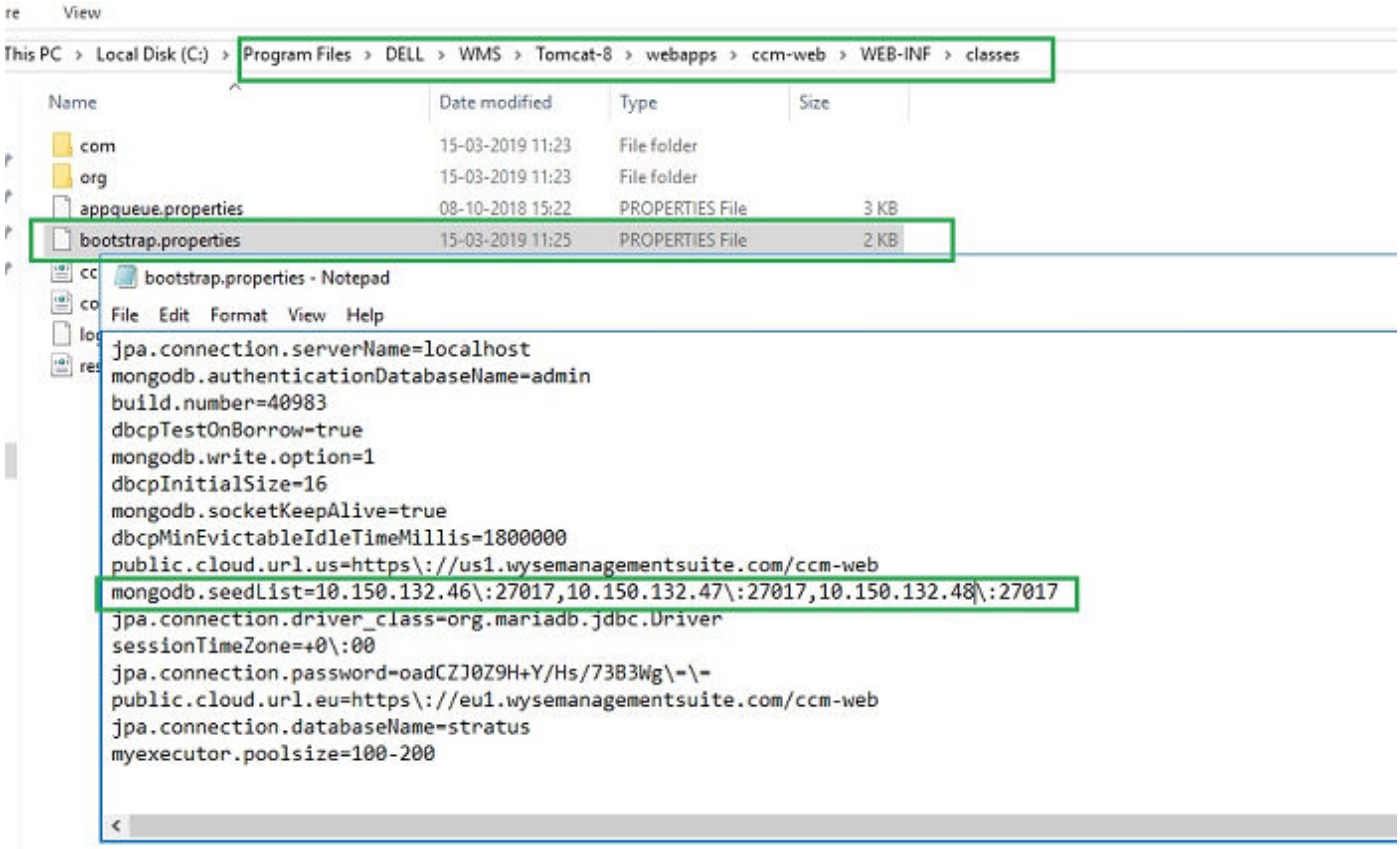

Figure 74. Prerequisite

- Ensure that the primary (active) Mongo database server with read and write access is the first entry in the `mongodb.seedList`. This is because the installer uses only the first entry as the primary server in the MongoDB cluster.

About this task

To upgrade Wyse Management Suite from version 1.4 to 2.0, do the following:

Steps

1. Double-click the Wyse Management Suite 1.4 installer package.
2. On the **Welcome** screen, read the license agreement and click **Next**.

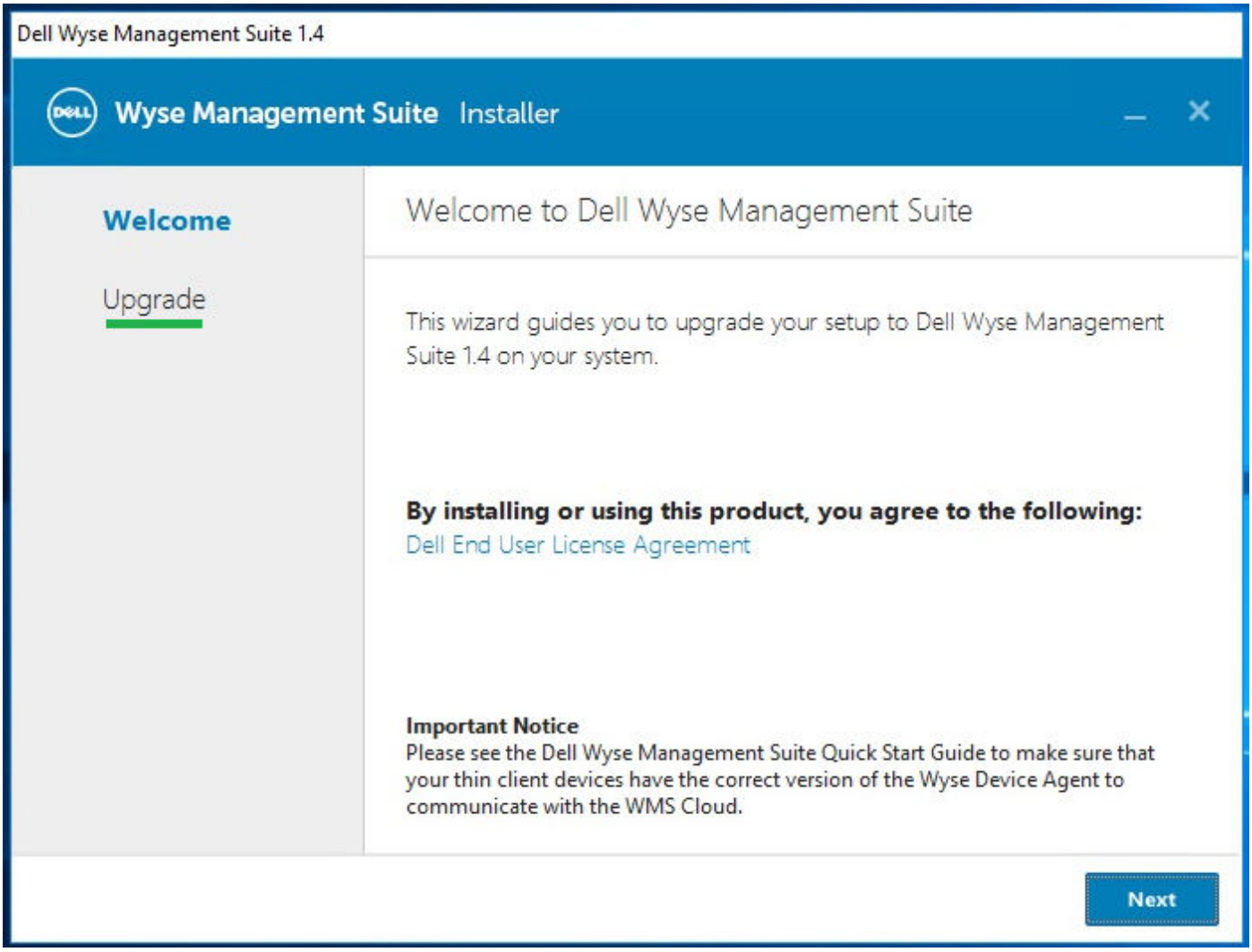

Figure 75. Welcome screen

3. On the **Upgrade** page, click **Next** to upgrade Wyse Management Suite .

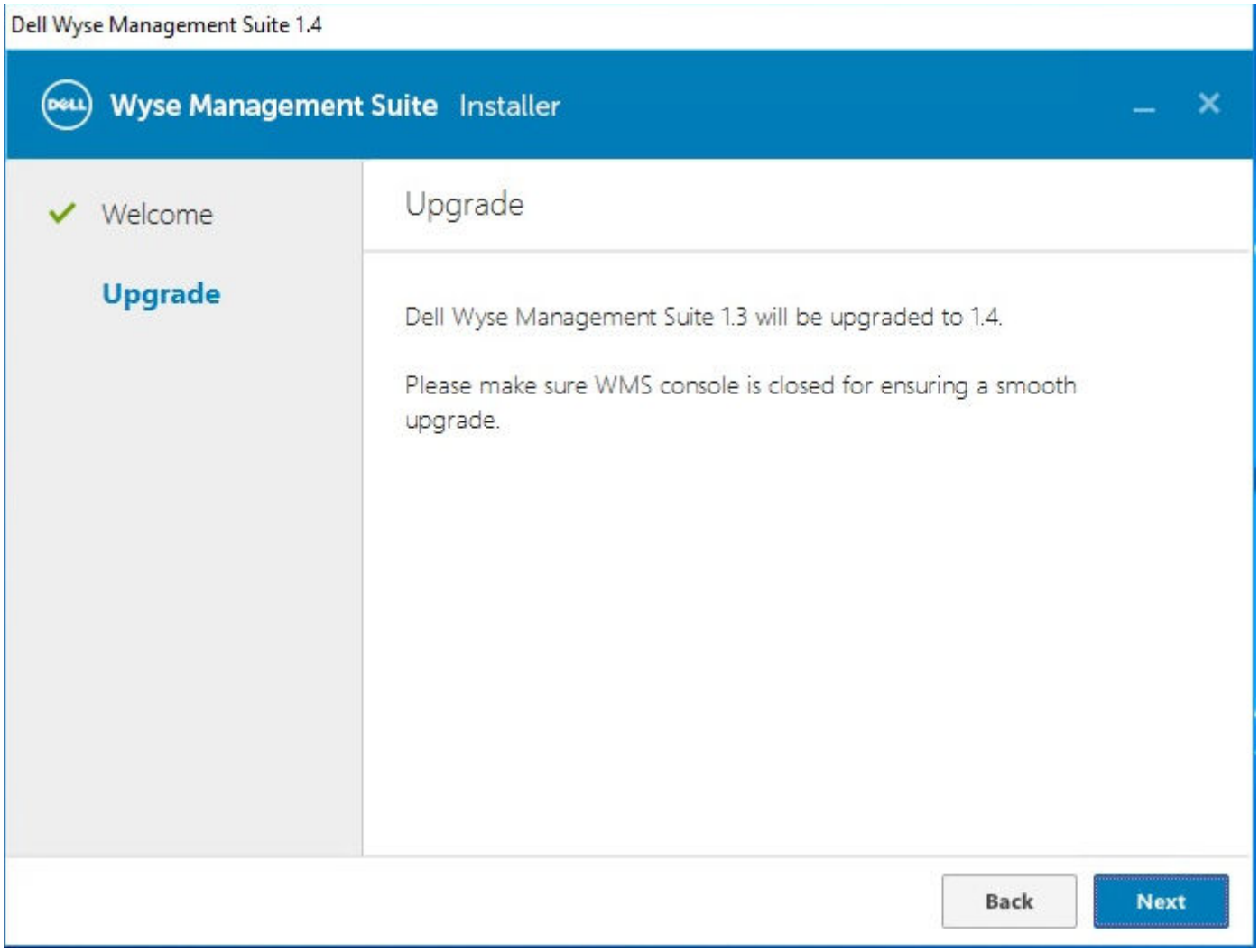

Figure 76. Upgrade

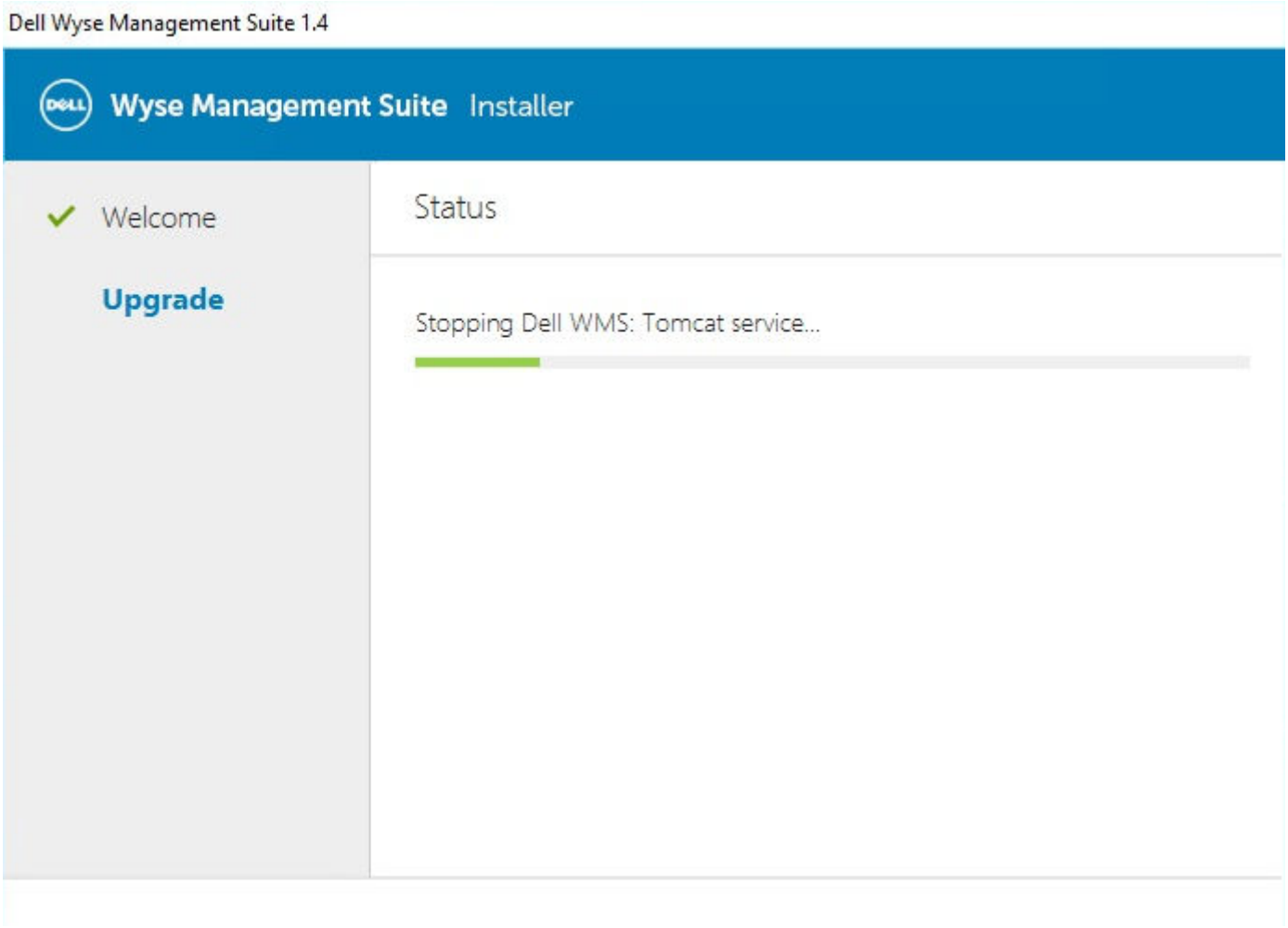

Figure 77. Upgrade

4. Click **Launch** to open the Wyse Management Suite web console.

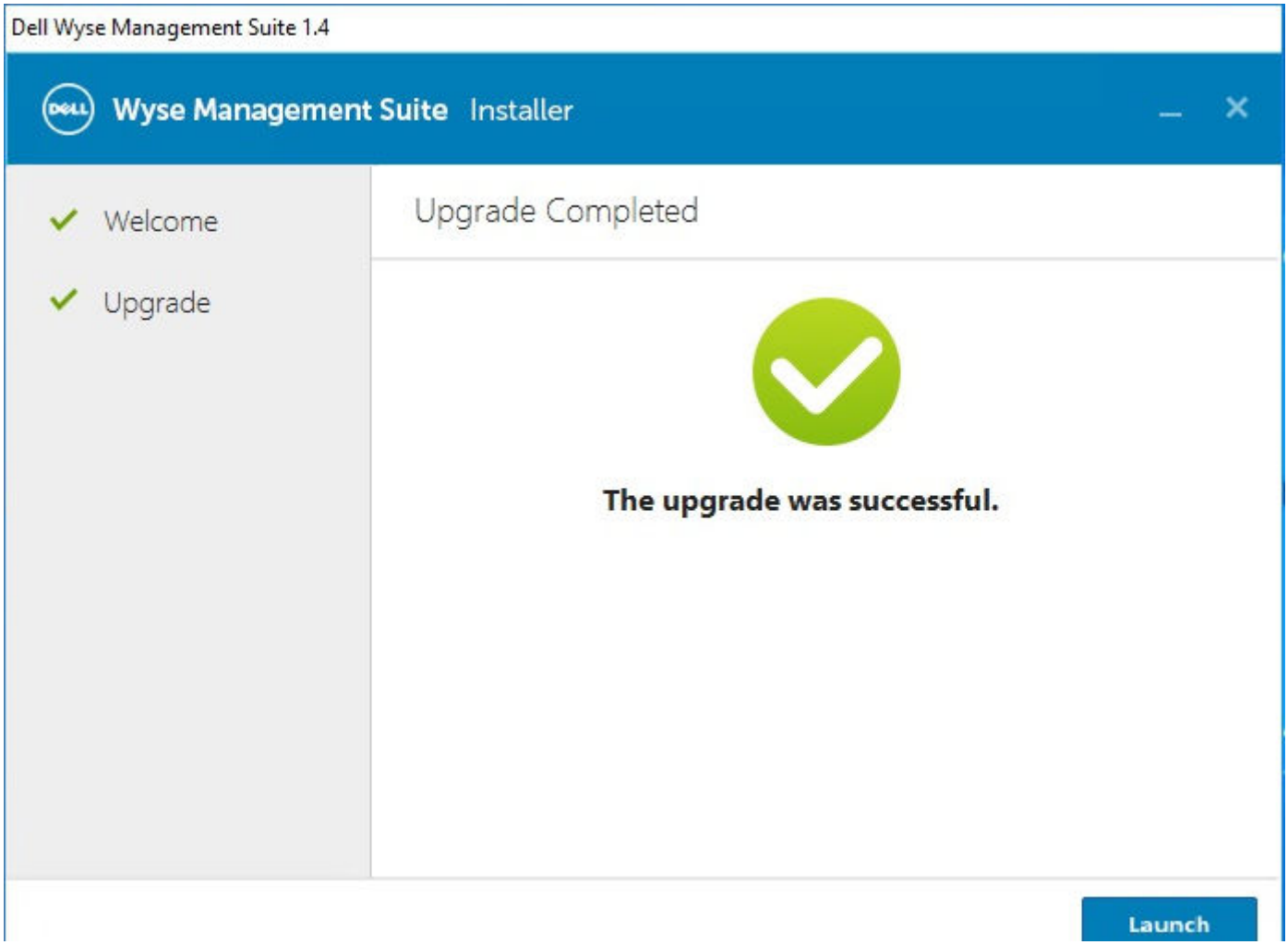

Figure 78. Launch

Next steps

- Ensure that Tomcat-8 folder and subfolders are deleted, and Tomcat-9 folder and subfolders are created. Also, do the following:
  - Ensure that `Tomcat-9\webapps\ccm-web\WEB-INF\classes` folders and subfolders are created.
  - Ensure that Tomcat-9 service is added, and Tomcat-9 service is running.
  - Ensure that the `bootstrap.properties` file is copied from `Tomcat-8\ webapps\ccm-web\WEB-INF\classes` folder to `Tomcat-9\webapps\ccm-web\WEB-INF\classes folder`.
  - Ensure that the `mongodb.seedList` value in the `bootstrap.properties` file includes backslash character (\) in the list of Mongo database servers. The `bootstrap.properties` file is at `Tomcat-8\webapps\ccm-web\WEB-INF\classes`, `mongodb.seedList = MongoDBServer1_IP\:27017,MongoDBServer2_IP\:27017, MongoDBServer3_IP \:27017`.
  - Ensure that the primary and secondary MongoDB servers entries are present in the `mongodb.seedList`.
- In the Windows Fail-over Cluster, if the status of the access point is down due to the unavailability of the Tomcat 8 service, do the following:
  1. Go to **Failover Cluster Manager** > **Cluster** > **Roles** > **Access Point**.
  2. Check the status of the Wyse management Suite related services, roles, and access point.

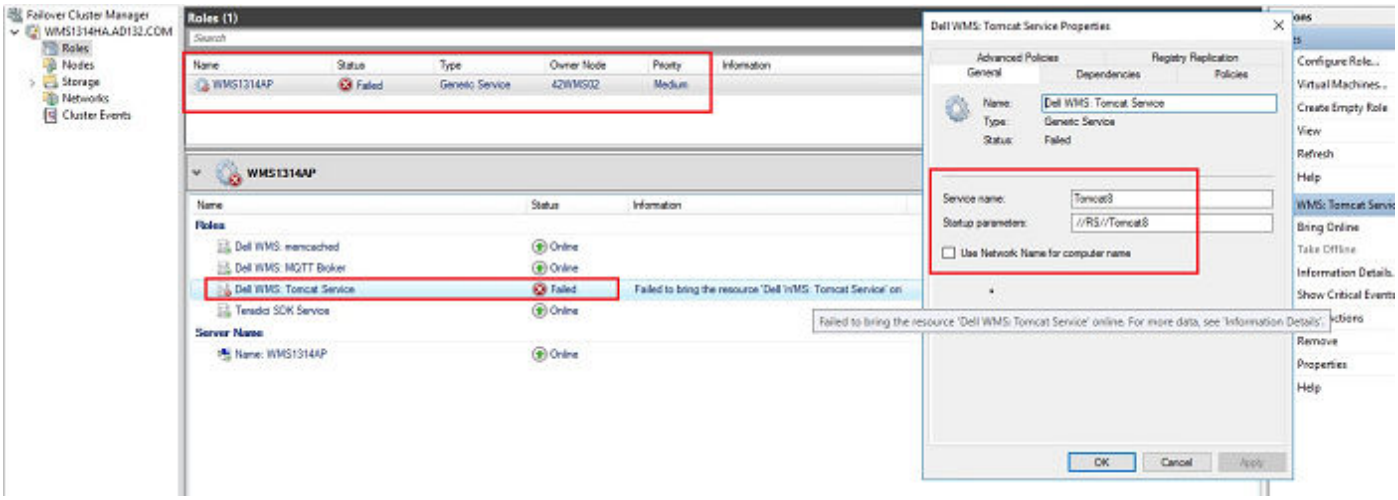

Figure 79. Access point

3. Check the version of the Tomcat service. If the version of the Tomcat service is 8, you must manually remove Tomcat-8 and add Tomcat-9 service into the Access Point. This is because, when you upgrade Wyse Management Suite 1.4 to WMS 2.0, Tomcat-8 service is replaced with Tomcat-9.
4. Right-click the Tomcat-8 service, and then click **Remove**.

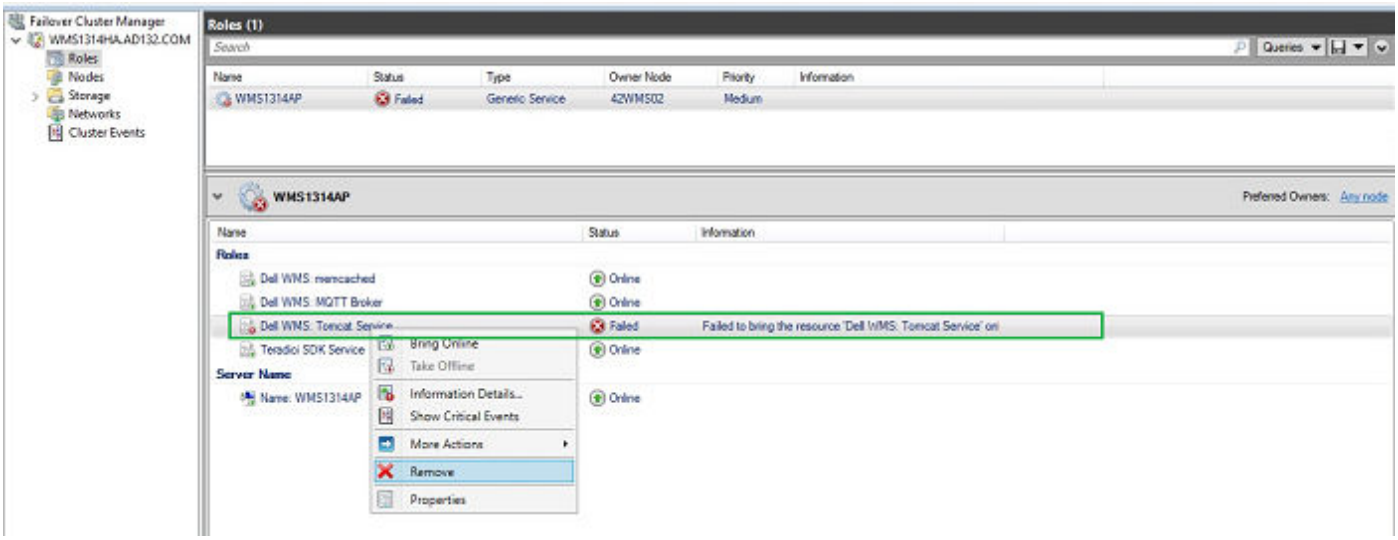

Figure 80. Tomcat service removal

5. Add the Tomcat-9 service to the access point.

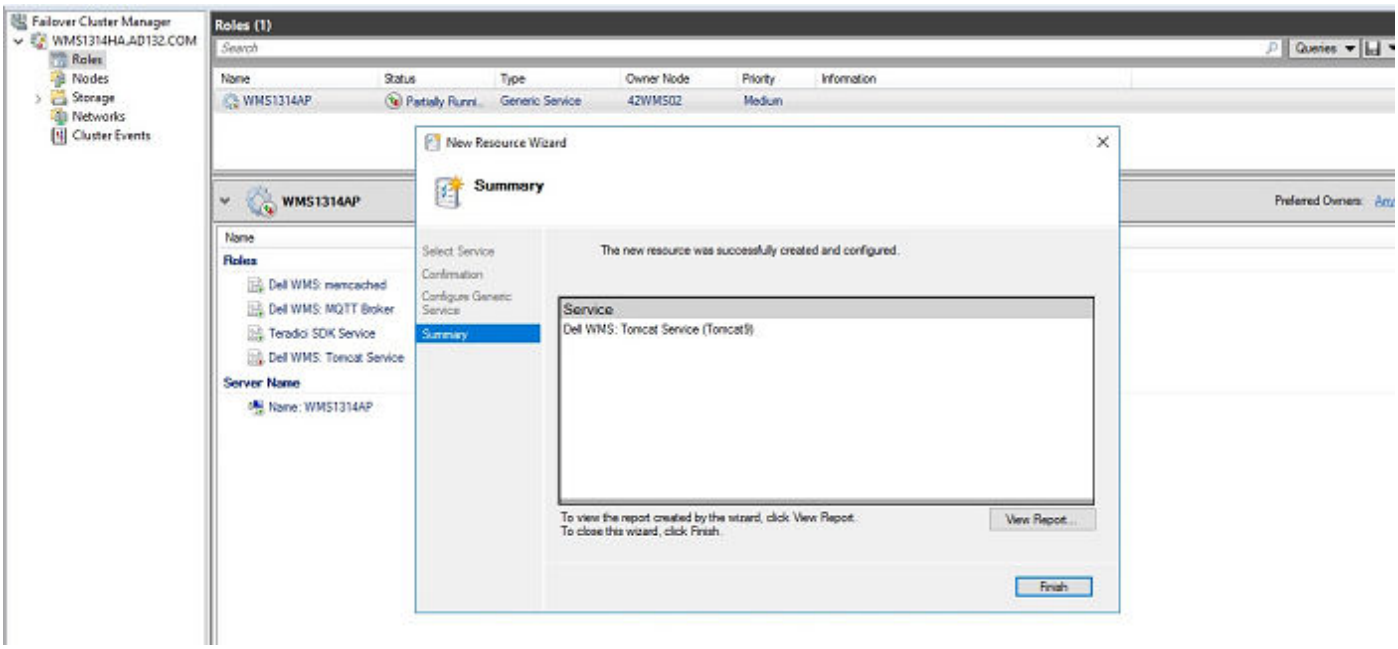

Figure 81. Tomcat-9 service

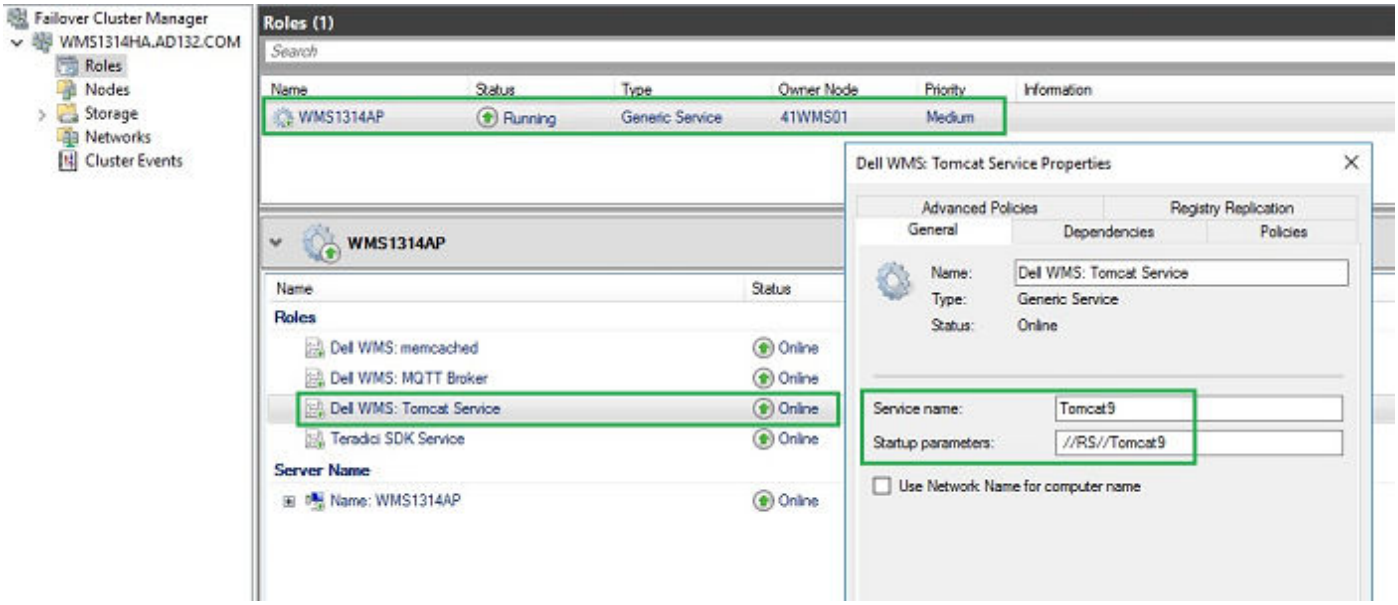

Figure 82. Tomcat 9 service

6. Bind the FQDN address of the access point of High Availability to the Memcached registry on both nodes of the High Availability setup using the command

```
Registry Path: HKLM\SYSTEM\CurrentControlSet\Services\Memcached\
“ImagePath” = “C:\Program Files\DELL\WMS\memcached\memcached.exe" -d runservice -p -I
11211 WMS1314AP.AD132.COM -U 0”
```

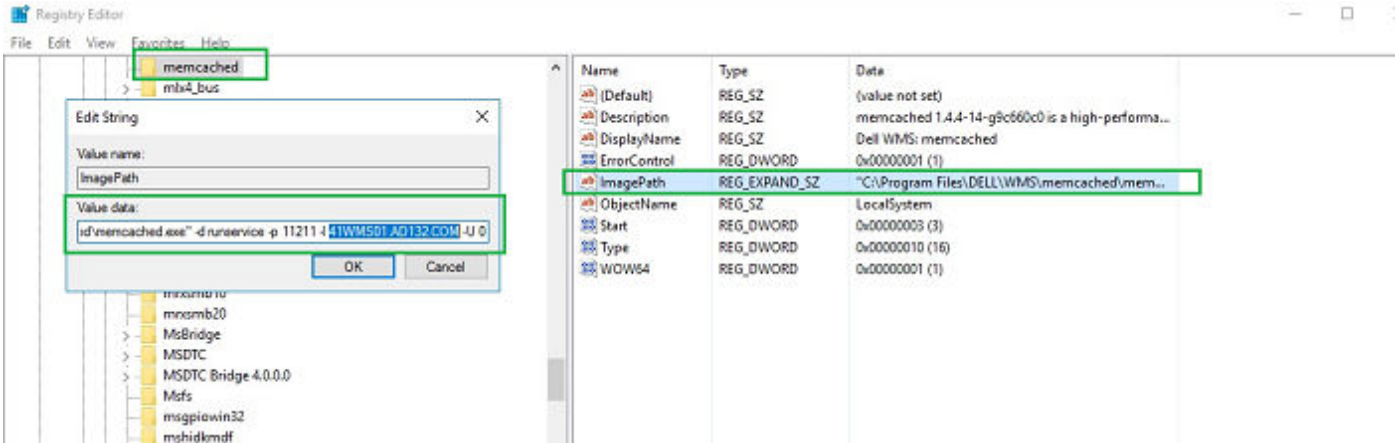

Figure 83. Memcached data

11

Upgrading from Wyse Management Suite version 2.x/2.1 to Wyse Management Suite version 3.x

Prerequisites

Ensure to perform the following tasks before upgrading to Wyse Management Suite version 3.x.

- Set the policy of the resources(tomcat,memcache,mqtt) in the access point to "if resource fails, Do not restart" though default policy “if resource fails, attempt restart on current node ” is recommended, for failover scenario it does not allow the product to upgrade.

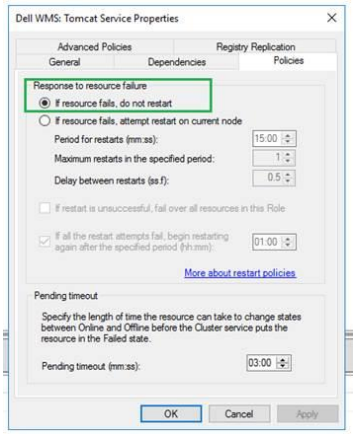

Figure 84. Tomcat Service Properties

- Upgrade the MongoDB Replica Set from 3.4.1 to 4.2.1; path of Mongo DB Upgrade is 3.4.1 >> 3.6 >> 4.0 >> 4.2.1. This is because from Wyse Management Suite 2.0 version onwards we are supporting MongoDB version 4.2.1 due to scheme changes made to support RAPTOR 9.0 devices.

  1. Upgrading Replica Set from 3.4.1 to 3.6—see https://docs.mongodb.com/manual/release-notes/3.6-upgrade-replica-set/.
  2. Upgrading Replica Set from 3.6 to 4.0.13—see https://docs.mongodb.com/manual/release-notes/4.0-upgrade-replica-set/.
  3. Upgrading Replica Set from 4.0 to 4.2.1—see https://docs.mongodb.com/manual/release-notes/4.0-upgrade-replica-set/.

- The primary MongoDB server must be the first entry in the 'mongodb.seedList' value in 'bootstrap.properties' file under

  " Tomcat-9\webapps\ccm-web\WEB-INF\classes ".

- The MS Services Control Panel "services.msc" and any Wyse Management Suite related Files and Folder must be closed.
- Install Visual C++ 2015 or 2017 Redistributable package (x64) or later versions. Wyse Management Suite installer requires VCRUNTIME140.dll file to connect with MongoDB replica set or stand-alone setup with version 4.2.1.

Steps

1. Double-click the Wyse Management Suite 2.x installer package.
2. On the Welcome screen, read the license agreement and click **Next**.

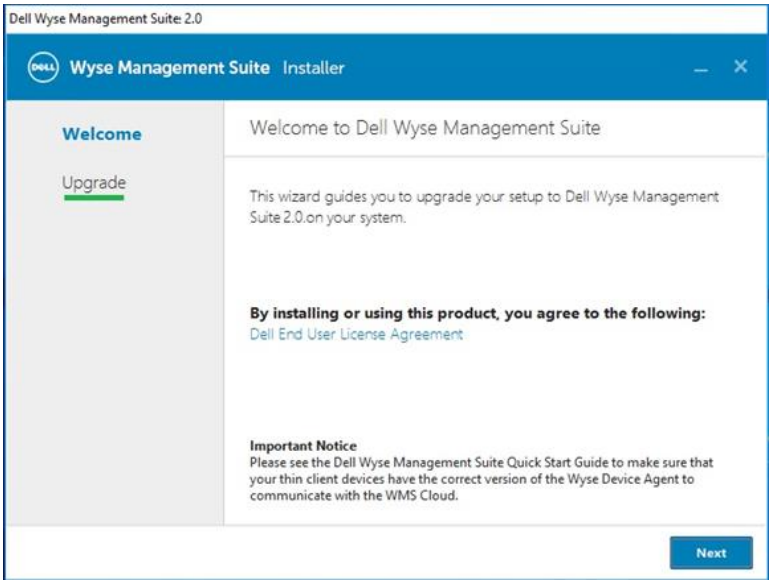

Figure 85. Upgrade

3. Clear the **Bind Memcached to 127.0.0.1** check box.

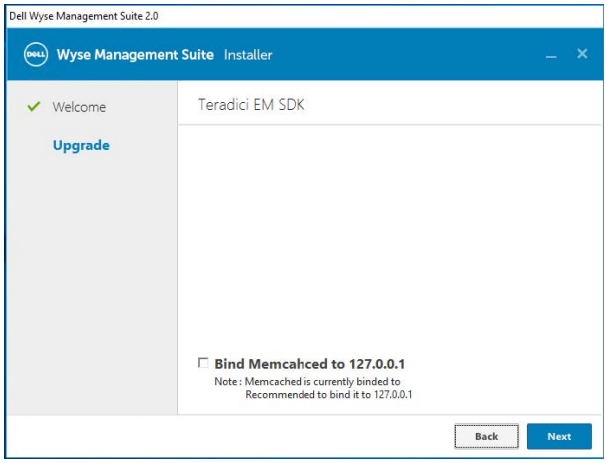

Figure 86. Teradici EM SDK

4. Click **Next**.

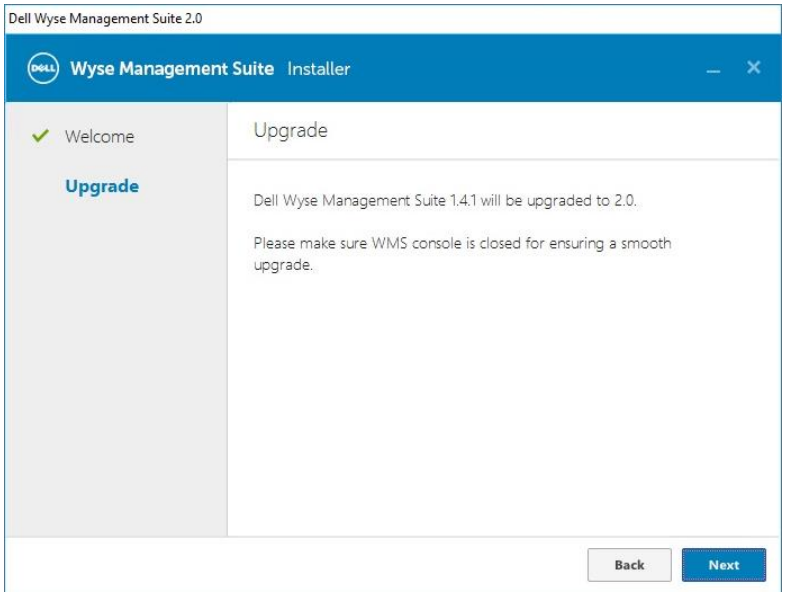

Figure 87. Upgrade

5. In the **Dell Wyse Management Suite - InstallShield Wizard** window, click **Yes**.

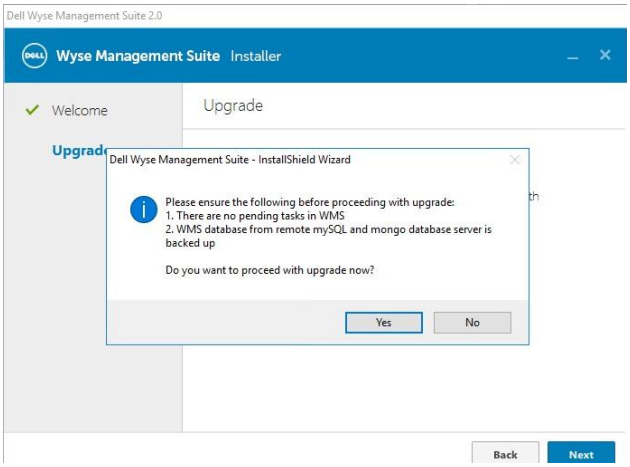

Figure 88. Upgrade

Wait for the installation to complete.

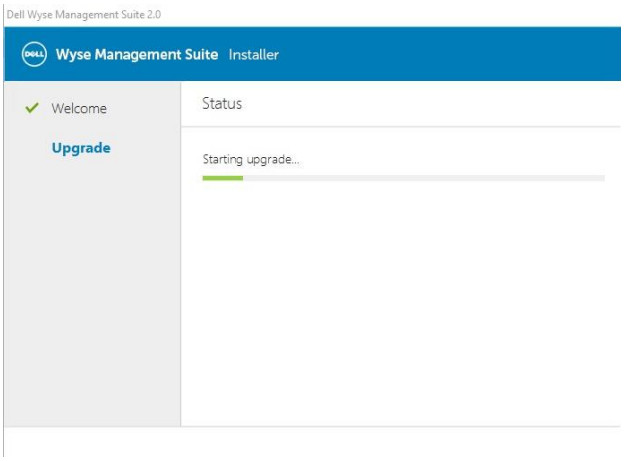

Figure 89. Upgrade

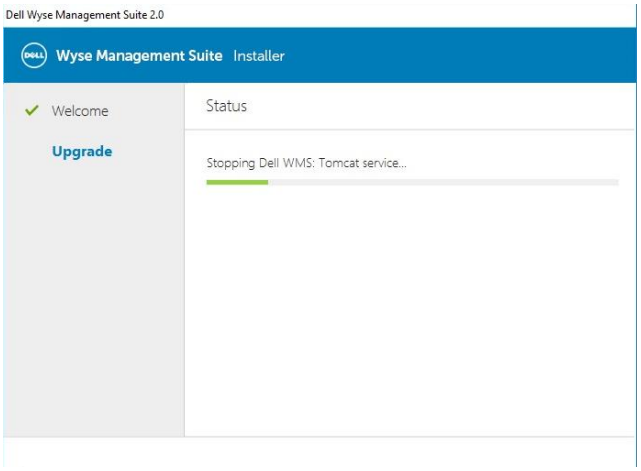

Figure 90. Upgrade

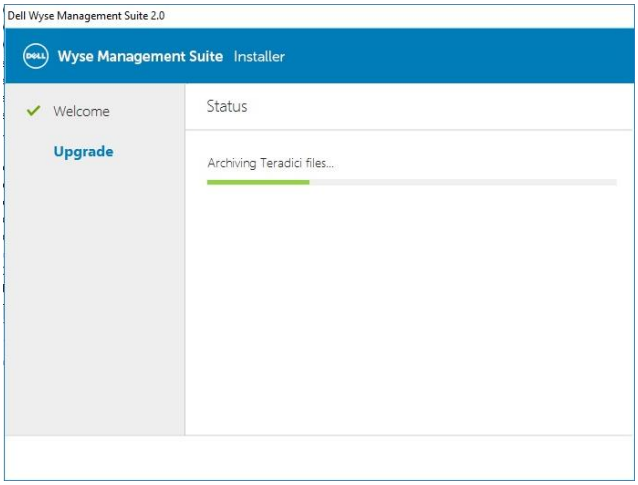

Figure 91. Upgrade

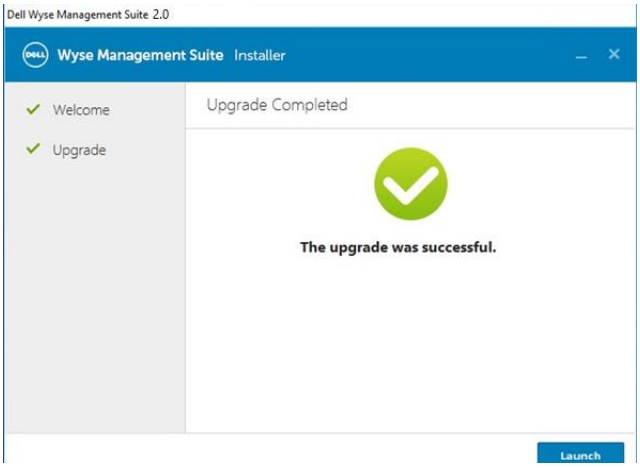

Figure 92. Upgrade

Post upgrade from Wyse Management Suite version1.4/1.4.1 to Wyse Management Suite version 2.0

Ensure to change back the Access Point's service policy setting of the resources to default "if resource fails, attempt restart on current node " configuration.

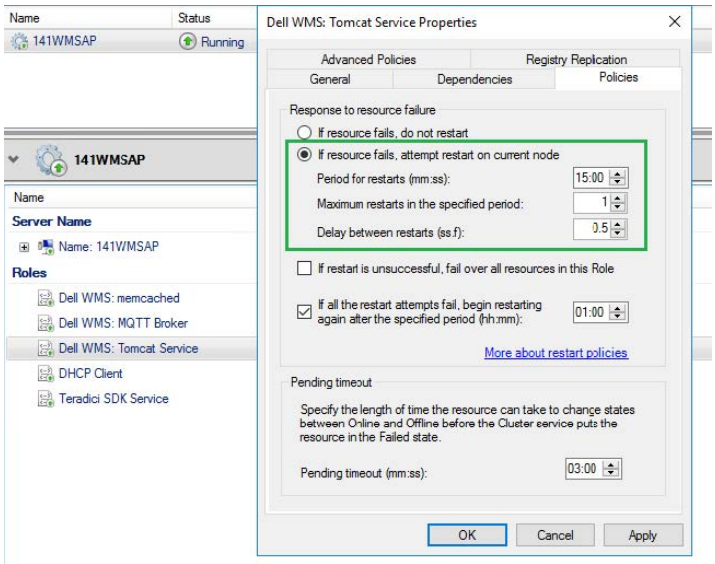

Figure 93. Access point

12

Troubleshooting

About this task

This section provides troubleshooting information for Wyse Management Suite version 1.x for the cluster set up.

- Problem: Where is the Wyse Management Suite log file located to check server installation issues.

  Workaround: The log file is in the `%temp% WMSInstall.log`folder.
- Problem: Where is the Tomcat service related log file located to check the application related issues.

  Workaround: If any of the node/server in the cluster does not work and fails to be part of the MySQL cluster do the following:

  1. Reboot the cluster node and run the command `var cluster = dba.rebootClusterFromCompleteOutage();` in the shell prompt.
  2. Reconfigure the local instance using the command `dba.configureLocalInstance('root@Server_IPAddress:3306')`.
  3. Add the instnce back to the cluster using the command `cluster.addInstance('root@Server_IPAddress:3306')`.
- Problem: If any of the server or node in the cluster stops working and is not part of the MySQL InnoDB cluster.

  Workaround: Perform the following steps at the command prompt:

```
var cluster = dba.rebootClusterFromCompleteOutage(); #Reboot the cluster instance
dba.configureLocalInstance('root@Server_IPAddress:3306') #Reconfigure the local instance
cluster.addInstance('root@Server_IPAddress:3306')#Add the cluster instance back to the
network
My-SQL JS> cluster.rejoinInstance("root@Server_IPAddress")
```

- Problem: If the server IDs are same in all the nodes, and if we try adding instances in the Cluster, an error message **ERROR: Error joining instance to cluster** is displayed.

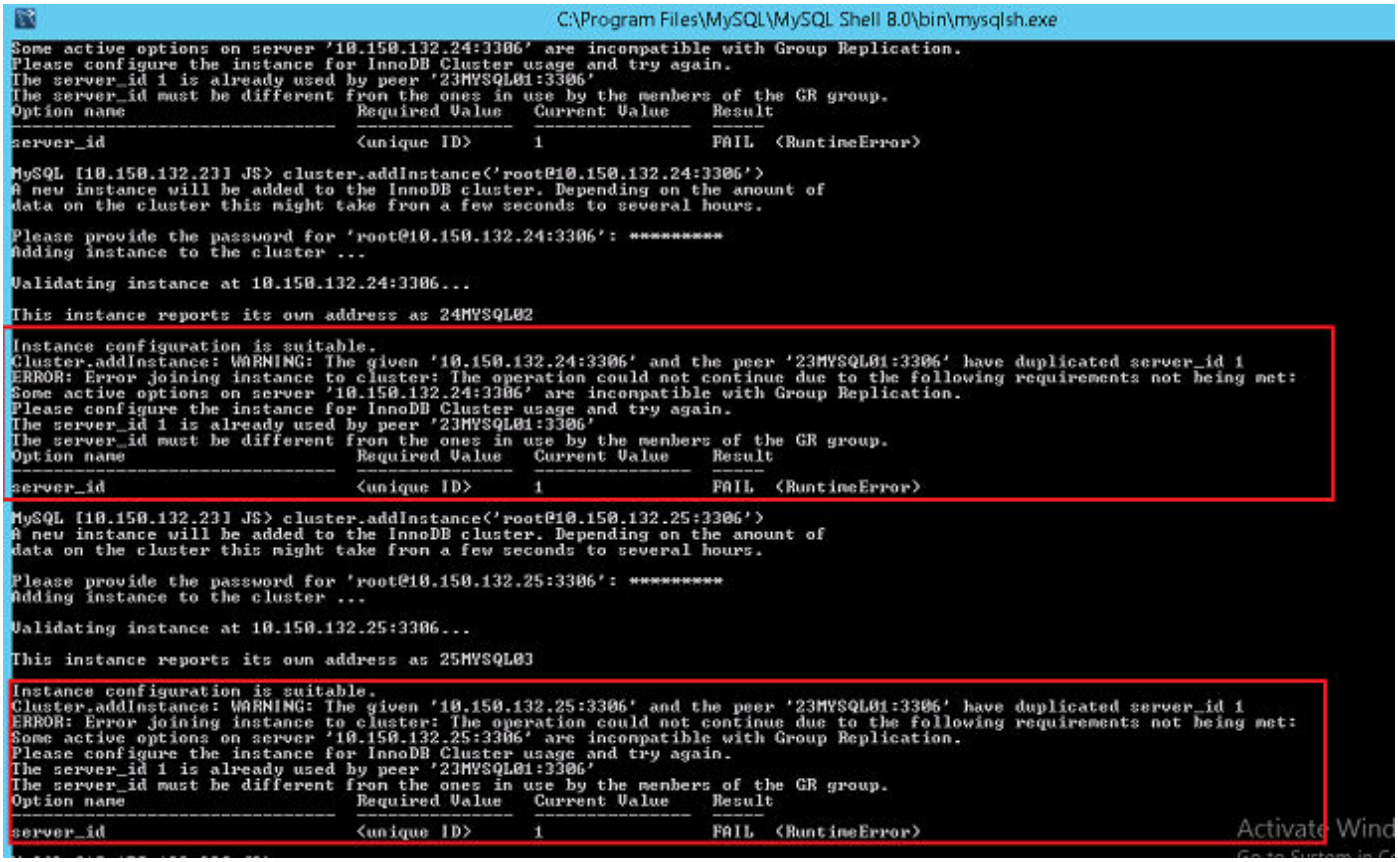

Figure 94. Error message

Workaround: Change the server ID entries in the `my.conf` file located in the `\ProgramData\MySQL\MySQL Server 5.7` directory.

Figure 95. change server ID