Install Infinite i on Microsoft Azure

Setup guide for a virtual machine running Windows Server

May 2021

Infinite Corporation

Microsoft

Contents

1 Introduction **3**

1.1 About this guide 3

1.2 Prerequisites 4

2 Set up a Windows Server VM **4**

2.1 Install .NET Framework 3.5 5

2.2 Install Cygwin 6

2.3 Install Visual Studio 2015 Community Edition 11

2.4 Install optional components 13

2.5 Set up working directories 13

3 Install Infinite i Console **15**

3.1 Configure Infinite i Console 17

3.2 Configure Windows Firewall 20

4 Install Infinite i **24**

4.1 Configure system environmental variables 28

4.2 Test the compilers 34

4.3 Restore libraries provided by Infinite Corporation 36

5 Create the database on Azure **37**

6 *Optional*. Set up Infinite i database connection **40**

6.1 Set the SQL Server security mode 41

6.2 Create the SQL Server database 42

6.3 Initialize the database 44

6.4 Create an ODBC DSN 45

6.5 *Optional.* SQL Server redundancy and high availability 48

7 *Optional*. Install Infinite Cloud **49**

© 2021 Microsoft Corporation. All rights reserved. This guide is for informational purposes only. MICROSOFT MAKES NO WARRANTIES, EXPRESS, IMPLIED OR STATUTORY, AS TO THE INFORMATION IN THIS DOCUMENT. This document is provided "as is." Information and views expressed in this document, including URL and other internet website references, may change without notice. You bear the risk of using it. This document does not provide you with any legal rights to any intellectual property in any Microsoft product.

1 Introduction

This guide walks through the steps to set up and install Infinite i on Microsoft Azure. Infinite i is a suite of compilers, utilities, and operating system services that you can use to migrate RPG, COBOL, and CL application programs developed for the IBM iSeries (AS/400) environment. Without rewriting functionally rich applications, you can move them to the open architecture platform of Azure and begin benefitting from the scale and flexibility of the cloud.

Applications are migrated at the source code level, recompiled, and executed on Azure Virtual Machines running either Linux or Windows Server. Using Infinite i utilities, compilers, and services, you can run RPG and COBOL applications on Azure as if they were in the AS/400. The Infinite i runtime includes the commands of the AS/400 system and the database. For example, you can work with Programming Development Manager (PDM), Source Entry Utility (SEU), Data File Utility (DFU), and Screen Design Aid (SDA), where you can write screens (display files), physical and logical files, source code, and execute and debug programs.

Infinite i makes it easy to maintain RPG and COBOL applications in development and production environments that are very similar to the AS/400, down to the menus, command line, and green screen user interface. The DB2 database is replicated on Azure using an internal Infinite i database, or it can be converted to any relational database engine, including Azure SQL Database, Oracle, and MySQL. Infinite i also supports Linux, but this guide steps through the process for Windows Server.

1.1 About this guide

Infinite Corporation has been a development partner with Microsoft for decades, and we worked closely together in creating this guide. It describes and illustrates the steps to:

- Set up a virtual machine (VM) running Windows Server, where you install and run the Infinite i Console used to connect to Infinite i rehosting.
- Set up an Open Database Connectivity (ODBC) interface and create a SQL Server database on a VM.
- Generate the Infinite i system, which includes a complete replication of the AS/400 database (DB2/400).
- Install and configure Infinite i.

1.2 Prerequisites

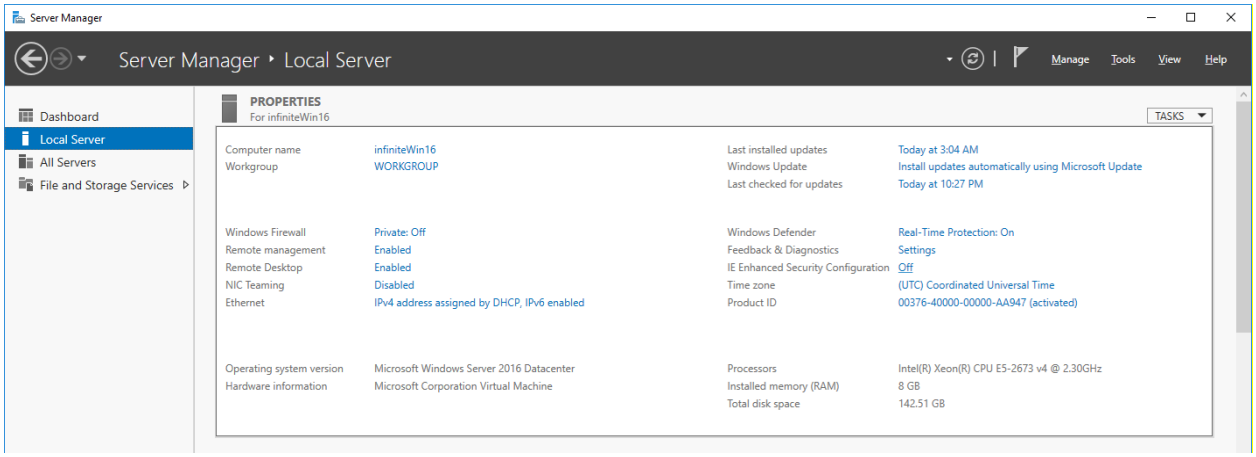
This guide assumes you have access to the following:

- A complete licensed set of RPG, COBOL, and CL source code to run Infinite i.
- An Azure account. You can install Infinite i using the smallest VM size in Azure.
- *Optional*. A relational database. This guide describes how to install a Microsoft SQL Server database and set up a data source name (DSN) for Infinite i. Applications use this name to request a connection to an ODBC data source. We recommend this step for the availability it adds. With SQL Server, you can also replicate a database across Azure regions.
- *Optional*. Apache Tomcat web server. If you want, you can use Tomcat with Infinite Cloud to implement a web UI for Infinite i and the user applications. For more information about this option, refer to *Infinite Cloud for Windows Documentation & Installation Instructions*.

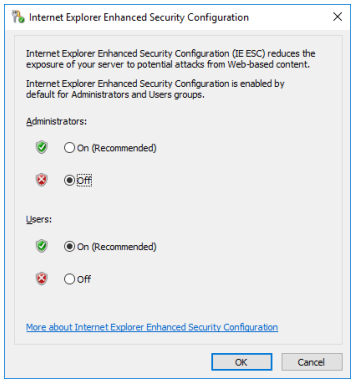
2 Set up a Windows Server VM

Start the setup process from the VM that you already created on Azure.

1. Go to Azure portal, sign on, and connect to the VM running Windows Server that you already set up for Infinite i.
2. Sign in to Azure portal and sign in to the Windows Server VM using your Administrator credentials.
3. Go to **Start** > **Settings** > **Update & Security**.
4. Install all Windows Updates, then check **Download with Manual Updates**.
5. Open Server Manager. In the left menu, click **Local Server**, then click the **IE Enhanced Security Configuration** option.



6. When prompted, in the **Enhanced Security Configuration** options, under **Administrators**, select **Off**, then click **OK**.

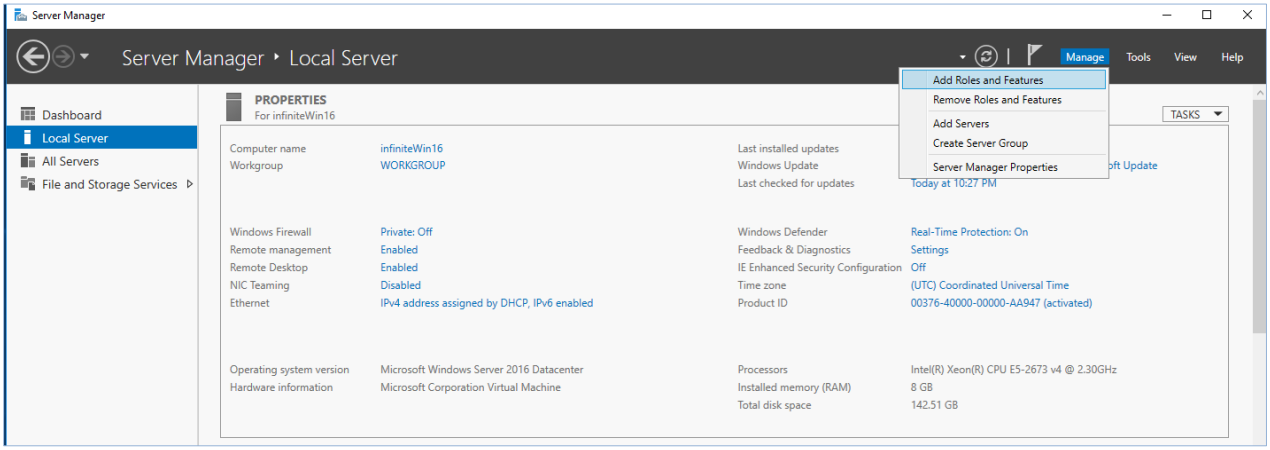


7. Keep Server Manager open for the next step.

2.1 Install .NET Framework 3.5

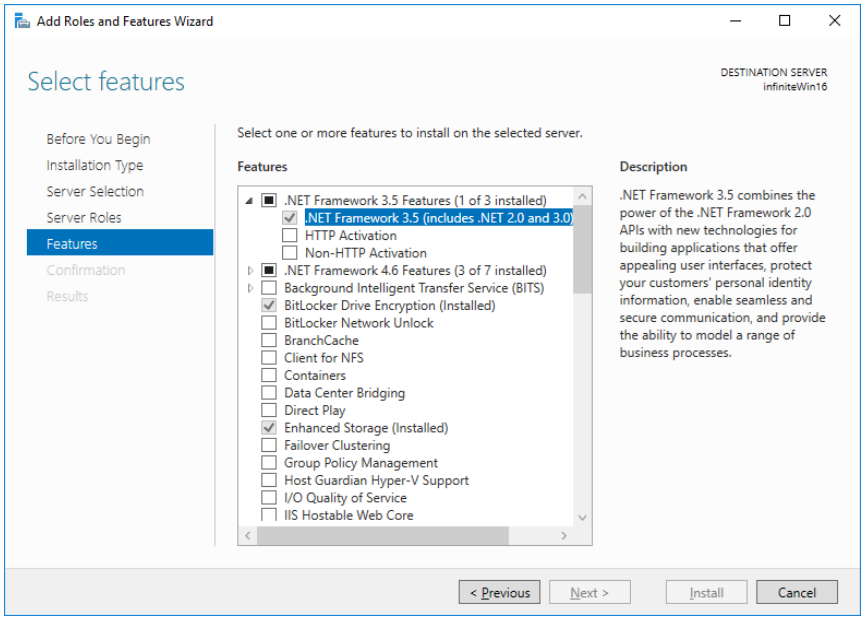
This setup process requires .NET Framework 3.5, a technology that supports building and running Windows apps and web services. If you don't have it already, install it as follows.

1. On the **Local Server** page in Server Manager, click **Manage** (in the header bar) and choose **Add Roles and Features**.



2. Click **Next** until you get to **Select features**, then click **.NET Framework 3.5 Features**.

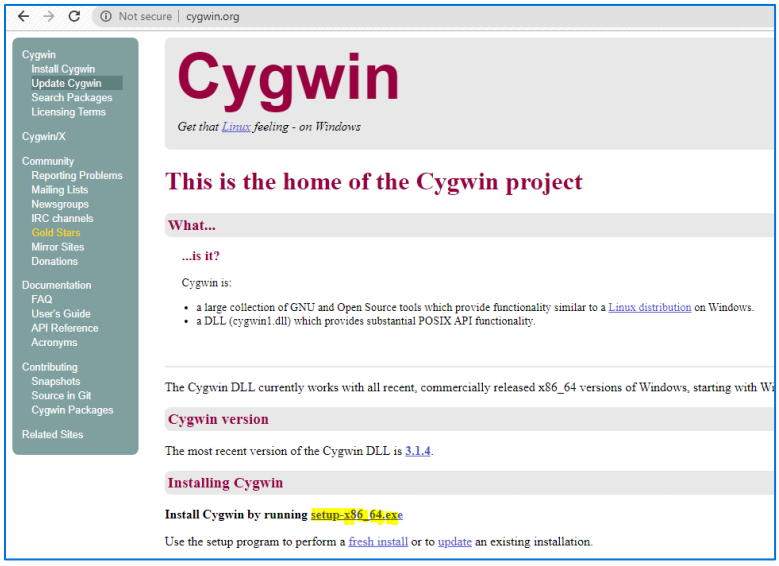
3. Click **Next**, then click **Install**.



2.2 Install Cygwin

This guide recommends using Cygwin tools to create a POSIX-compatible programming and runtime environment that runs natively on Windows Server. These tools are available at no cost as part of GNU software (gcc, gas, ld, and so on) and are covered by the standard X11 license.[1]

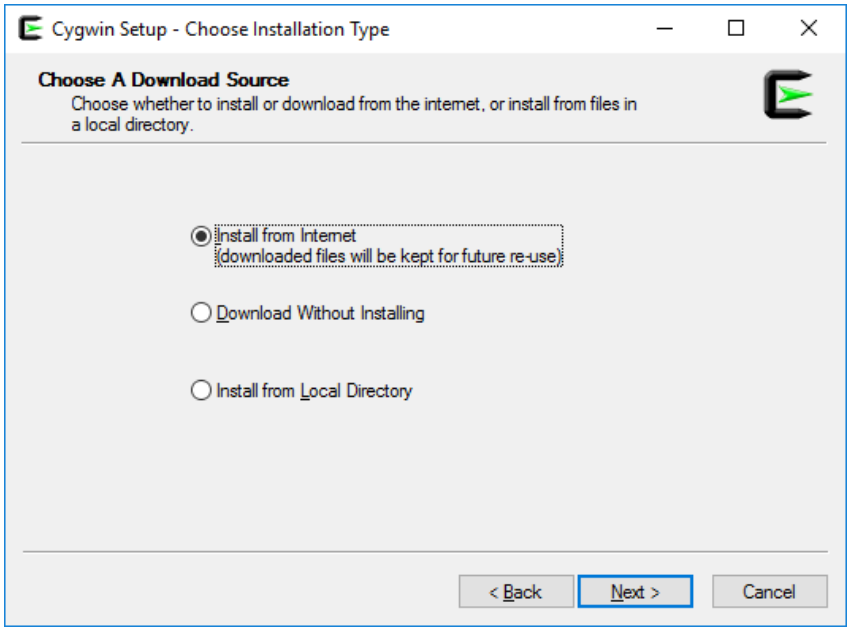
1. Go to https://cygwin.org/ and download **setup-x84_64.exe**.



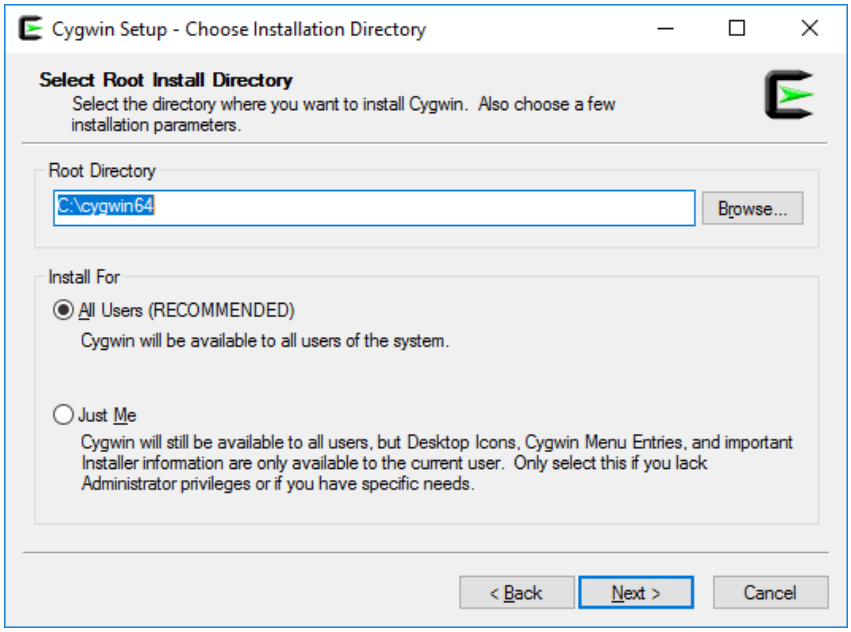
[1] For more licensing information, see "Are the Cygwin tools free software?" in *Cygwin Overview*. It recommends reading the copyright section of the FAQ for more information on how the GNU GPL may affect your use of these tools.

2. Run **setup-x84_64.exe** to start Cygwin Setup.

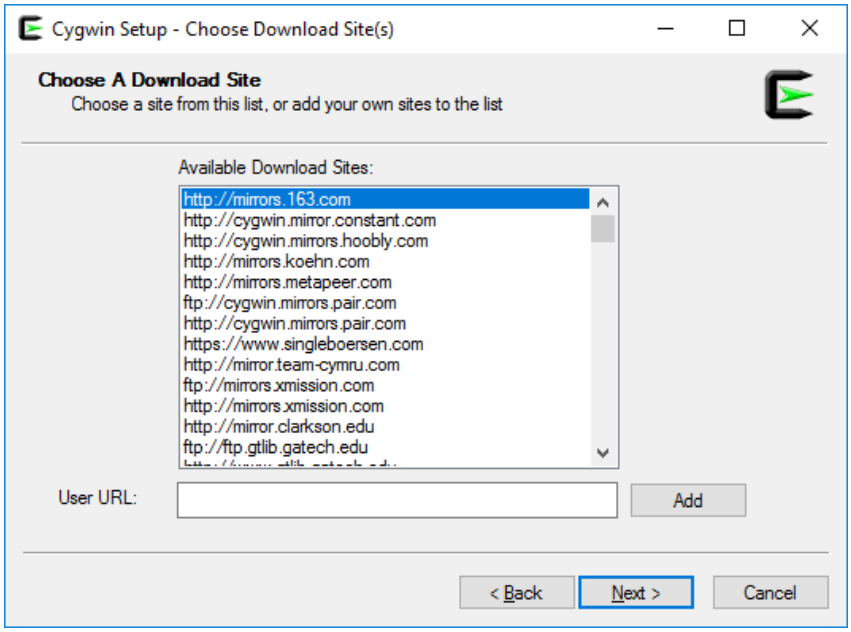
3. For **Choose a Download Source**, select **Install from Internet** and click **Next**.



4. For **Root Directory**, enter **C:\cygwin64** and click **Next**.

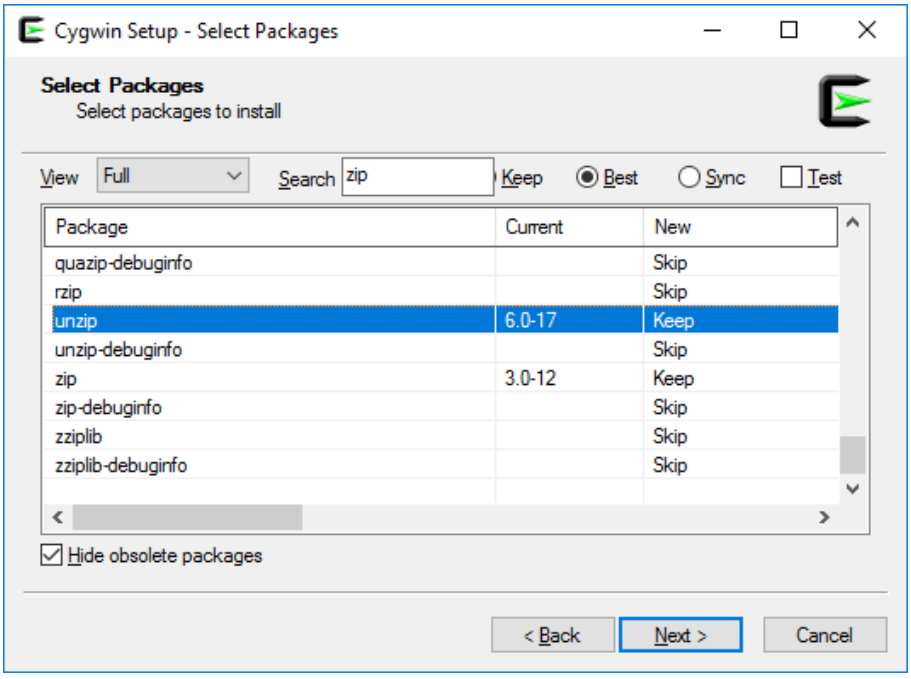


5. For **Available Download Sites**, select the first option on the list and click **Next**.

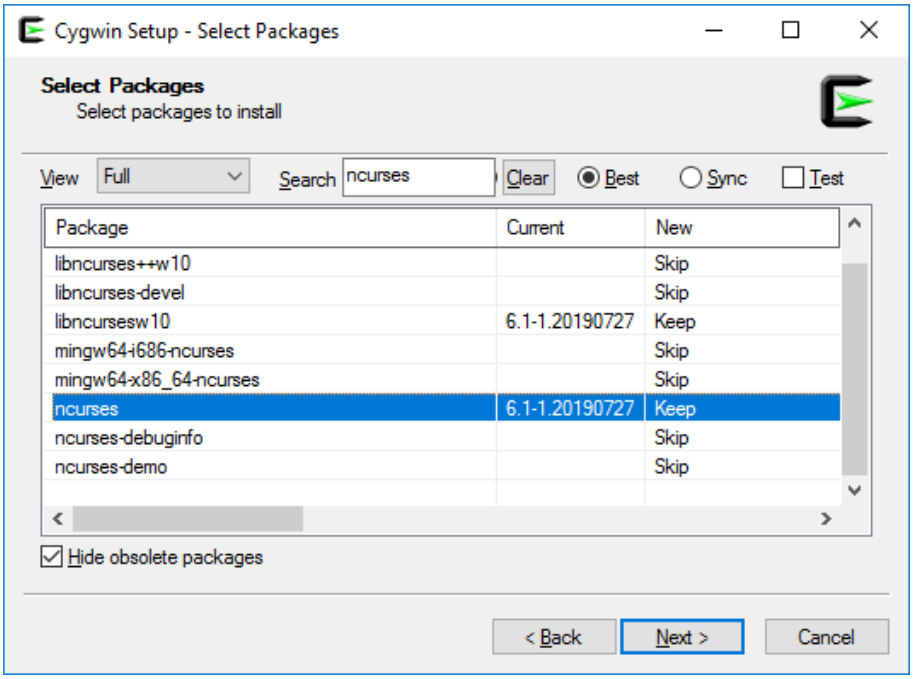


6. For **Select Packages**, set **View** to **Full**, then select the following packages:

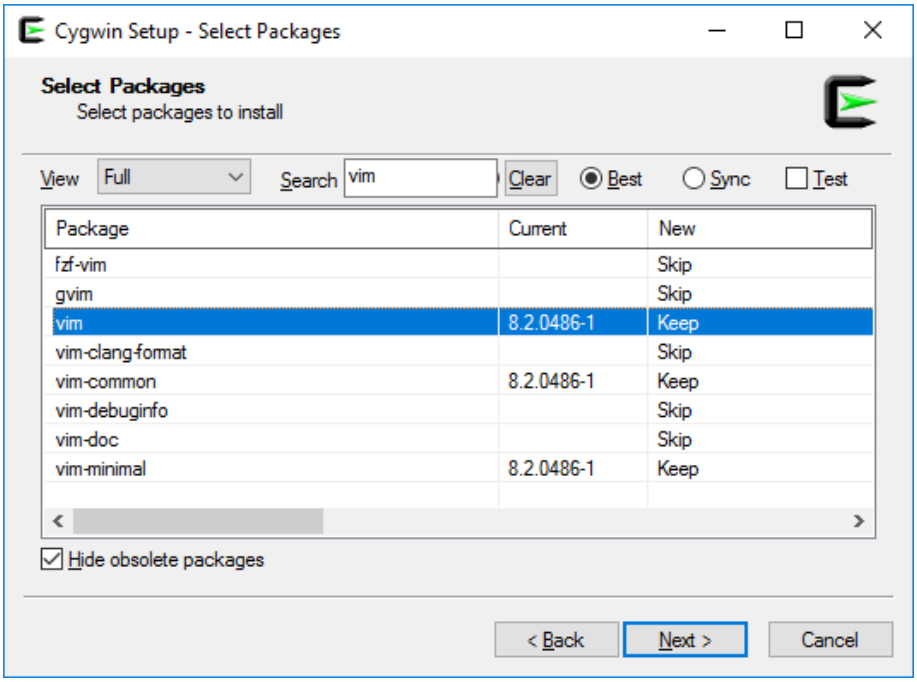
   a. In the Search box, type **zip**, then choose **unzip** from the list of archives and click **Next**.



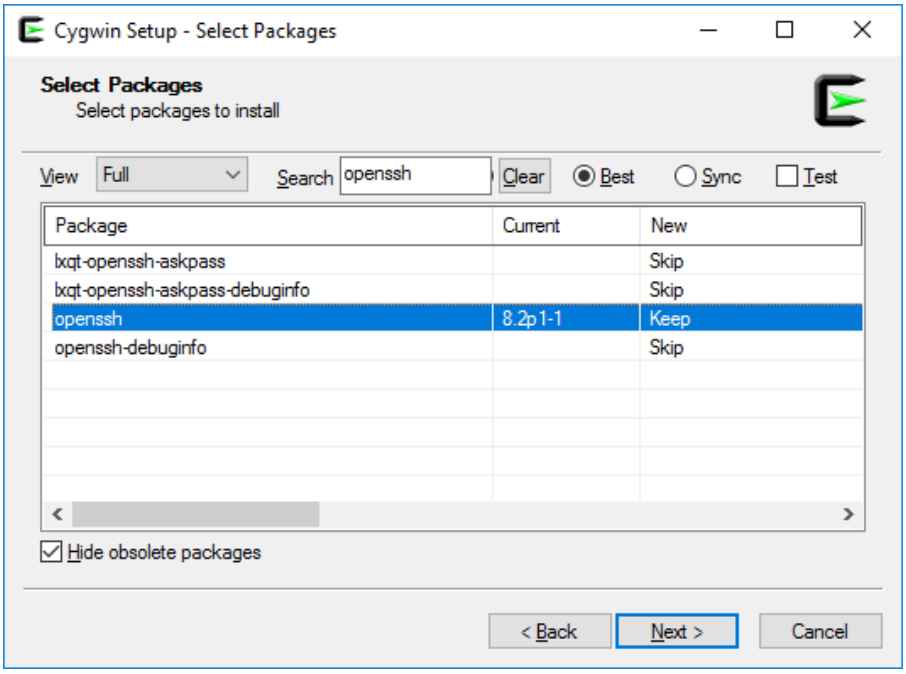
b. In the Search box, type **ncurses**, then choose **ncurses** from the list of bases and click **Next**.



c. In the Search box, type **vim**, then choose **vim** from the list of editors and click **Next**.

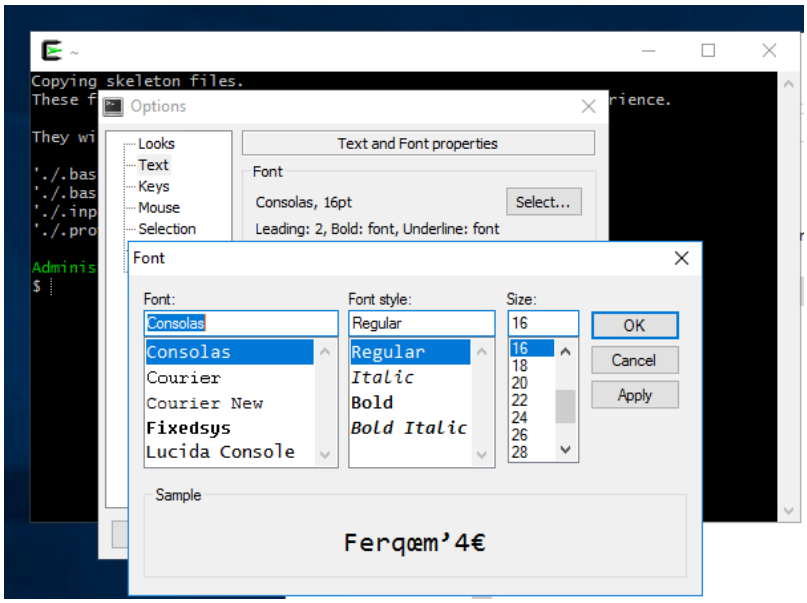


d. In the Search box, type **openssh**, then choose **openssh** from the list and click **Next**.



7. Continue to install Cygwin.

8. When installation is finished, right-click the top of the Cygwin terminal and click **Options** > **Font**. We recommend using 16 point Consolas font for the command prompt.



9. Open the Cygwin command prompt and edit .bash_profile using the command:

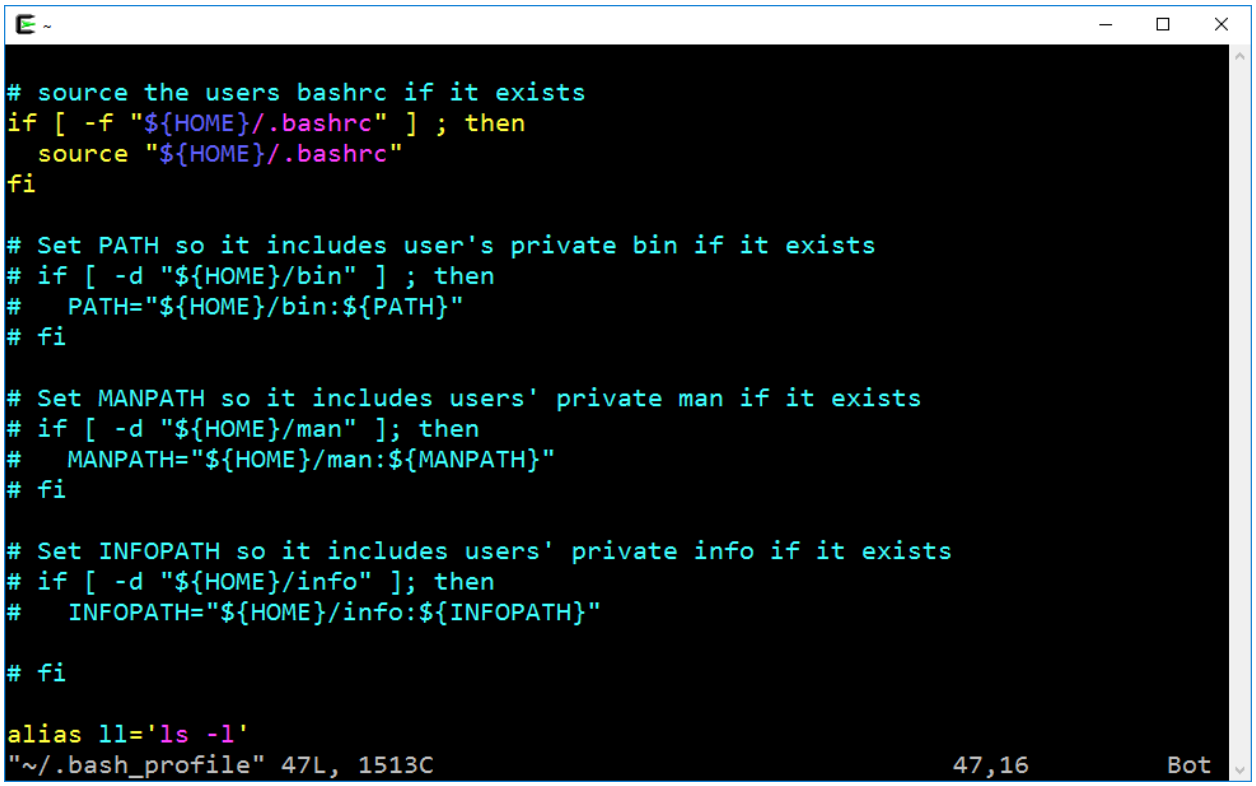
```
vi /home/Administrator/.bash_profile
```

10. Press **Shift+G** to go to the end of the file.

11. Press **Shift+O** to open a new line in insert mode, then add the following line:

```
alias ll='ls -l'
```

12. Press the Esc key to exit insert mode, then press **Shift+Z+Z** (the **Z** key twice) to save changes.



2.3 Install Visual Studio 2015 Community Edition

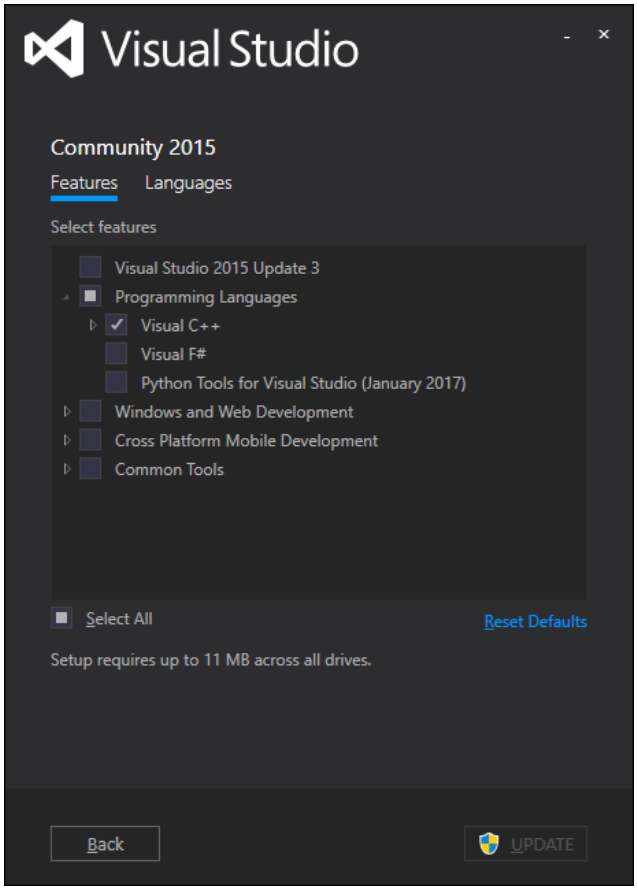
The next step installs Visual Studio 2015 Community Edition, which is used later to compile some of the executables. The C compiler from Visual Studio is used for that.

1. Run the vs_community.exe file that is provided.

> **Note:** If you already have the later version of Visual Studio 2019 Community Edition, you can use it, but these steps have been tested only with the earlier version provided.

2. In the installation options, choose **Custom** and uncheck all the options by clicking **Select All**.

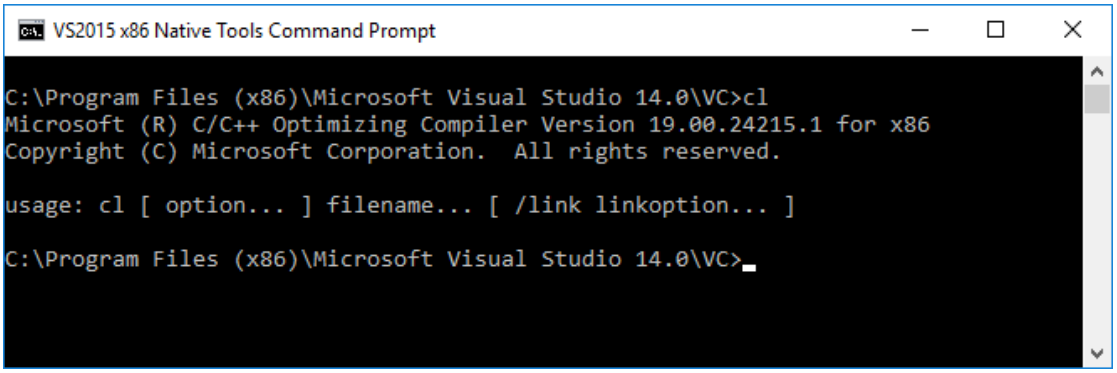
3. Click **Programming Languages** and check **Visual C++** (and no other options).



After the installation finishes, you may see the following message when you start Visual Studio Community for the first time:

```
License: 30 day trial (for evaluation purposes only) Your evaluation period has ended.
```

If so, go to C:\Program Files (or wherever you installed Visual Studio). In the Visual Studio 2015 folder, click **VS2015 x86 Native Tools Command Prompt**. Type **cl** and press **Enter**. The following output appears:



2.4 Install optional components

We recommend installing the Google Chrome browser, the Notepad++ source code editor, and the FileZilla FTP tool.

These tools are free to download and install. In Windows environments, use of Notepad++ is governed by GPL License. FileZilla is distributed under the terms of the GNU General Public License.

Download and install as follows:

- For Google Chrome, go to https://www.google.com/chrome.
- For Notepad++, go to https://notepad-plus-plus.org/.

  Tip: After installation, we recommend clicking **Settings > Style Configurator**. Check **Enable global font** and **Enable global font size** and set **Font size** to **14**.
- For FileZilla, go to https://filezilla-project.org/.

  Tip: The FileZilla client also supports FTP over TLS (FTPS) and SFTP.

2.5 Set up working directories

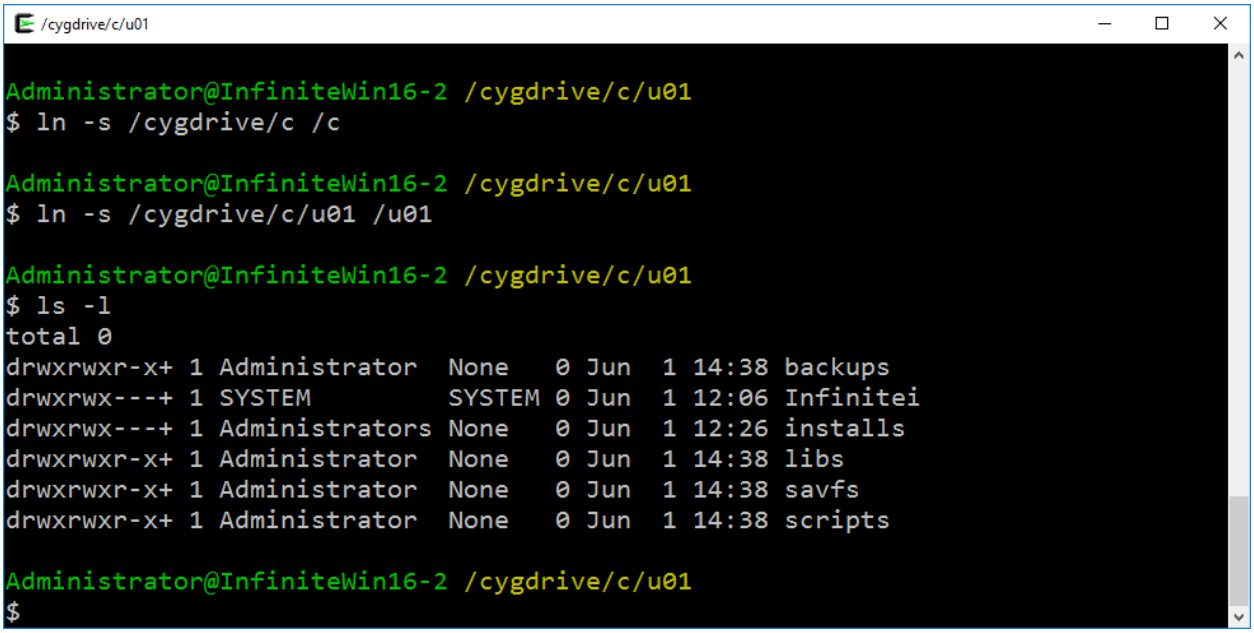
This step creates working directories using Cygwin.

1. Open the Cygwin command prompt and run the following commands:

```
mkdir /cygdrive/c/u01
cd /cygdrive/c/u01
mkdir backups installs libs savfs scripts
```

2. To create links to C: and u01 in Cygwin, run the following commands:

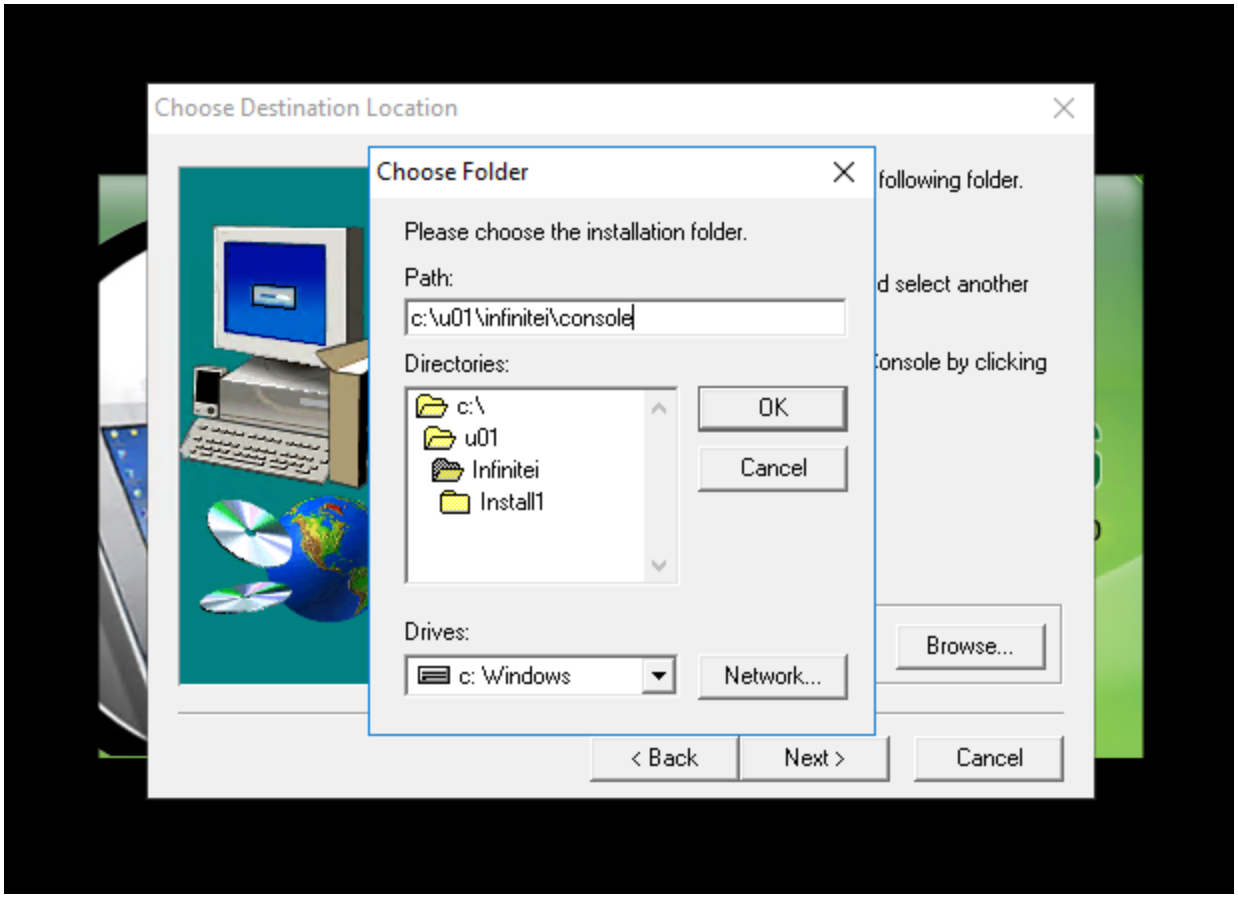
```
ln -s /cygdrive/c /c
ln -s /cygdrive/c/u01 /u01
```


/cygdrive/c/u01
Administrator@InfiniteWin16-2 /cygdrive/c/u01
$ ln -s /cygdrive/c /c
Administrator@InfiniteWin16-2 /cygdrive/c/u01
$ ln -s /cygdrive/c/u01 /u01
Administrator@InfiniteWin16-2 /cygdrive/c/u01
$ ls -l
total 0
drwxrwxr-x+ 1 Administrator None 0 Jun 1 14:38 backups
drwxrwx---+ 1 SYSTEM SYSTEM 0 Jun 1 12:06 Infinitei
drwxrwx---+ 1 Administrators None 0 Jun 1 12:26 installs
drwxrwxr-x+ 1 Administrator None 0 Jun 1 14:38 libs
drwxrwxr-x+ 1 Administrator None 0 Jun 1 14:38 savfs
drwxrwxr-x+ 1 Administrator None 0 Jun 1 14:38 scripts
Administrator@InfiniteWin16-2 /cygdrive/c/u01
$

3 Install Infinite i Console

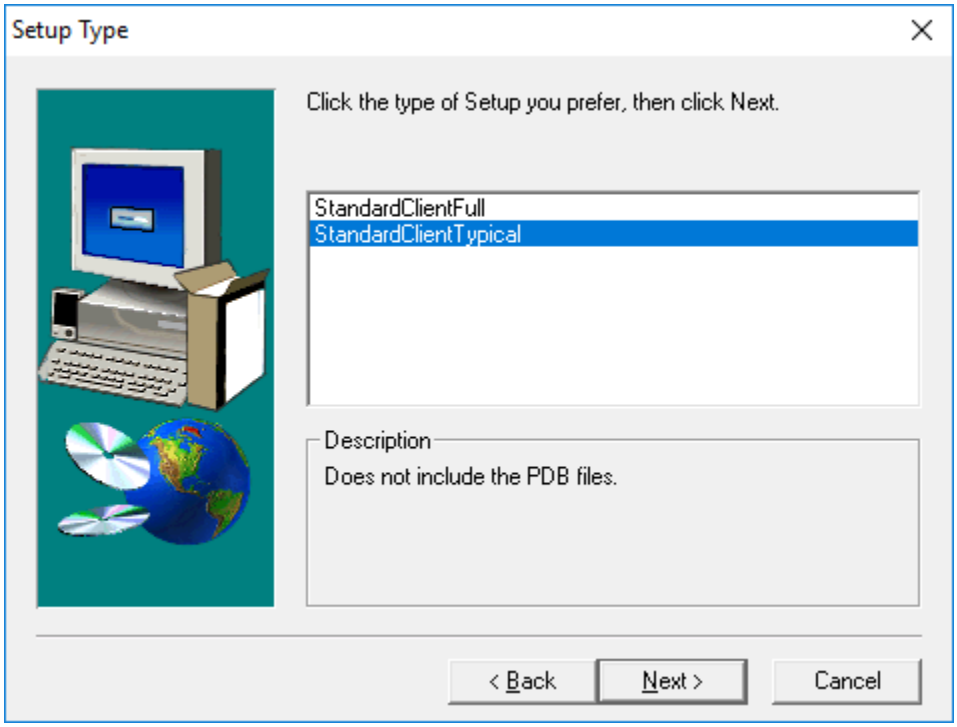
To install the Infinite i Console, you use the credentials provided when you validated your license. The console is how you connect to Infinite i rehosting, where you can work with files and programs, perform user tasks, and more, including migrating applications.

1. Expand the ISeriesConsole.zip file that's provided. Or download Infinite i Console from **Host: sftp.infinitecorporation.com**
2. For **User** and **Pass**, type the credentials provided by Infinite i.
3. For **File**, type **installs\ISeriesConsole.zip**.
4. Place the downloaded file in **C:\u01\installs**.
5. Run setup and click the option to choose the destination location. For **Path**, type **C:\u01\infinitei\console** and click **Next**.

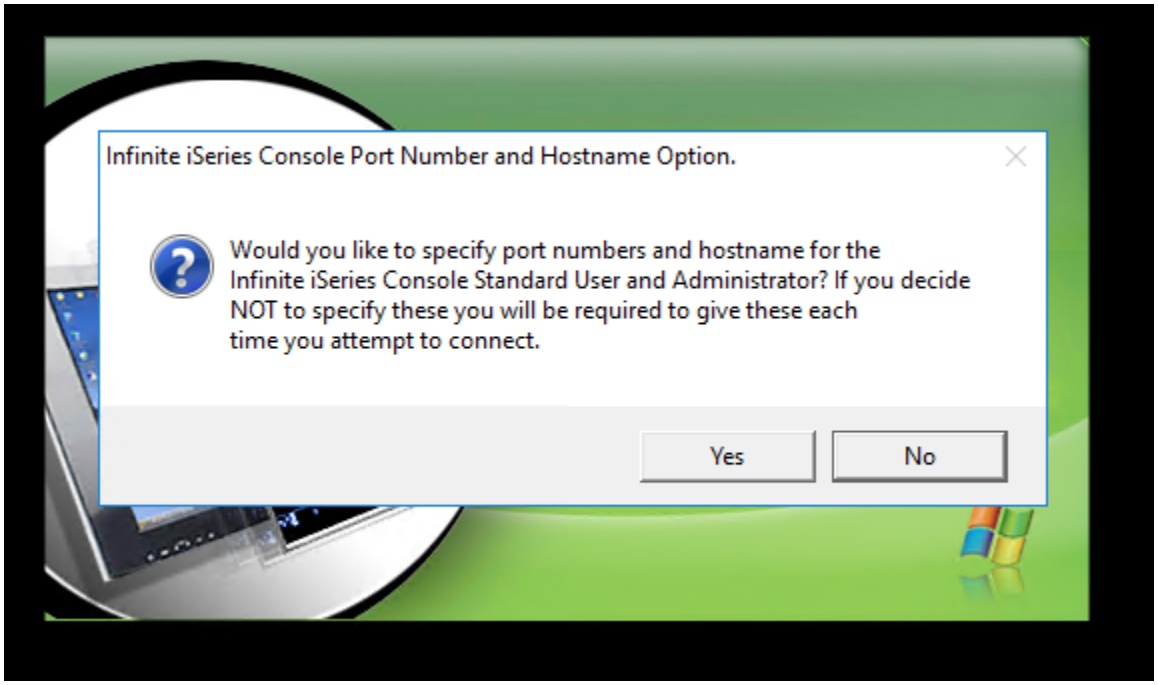


6. Follow the prompts in the setup wizard. Accept the destination location you specified and click **Next**.

7. In **Setup Type**, click **StandardClientTypical** and then click **Next**.



8. When prompted to specify port numbers and hostnames, click **Yes**.

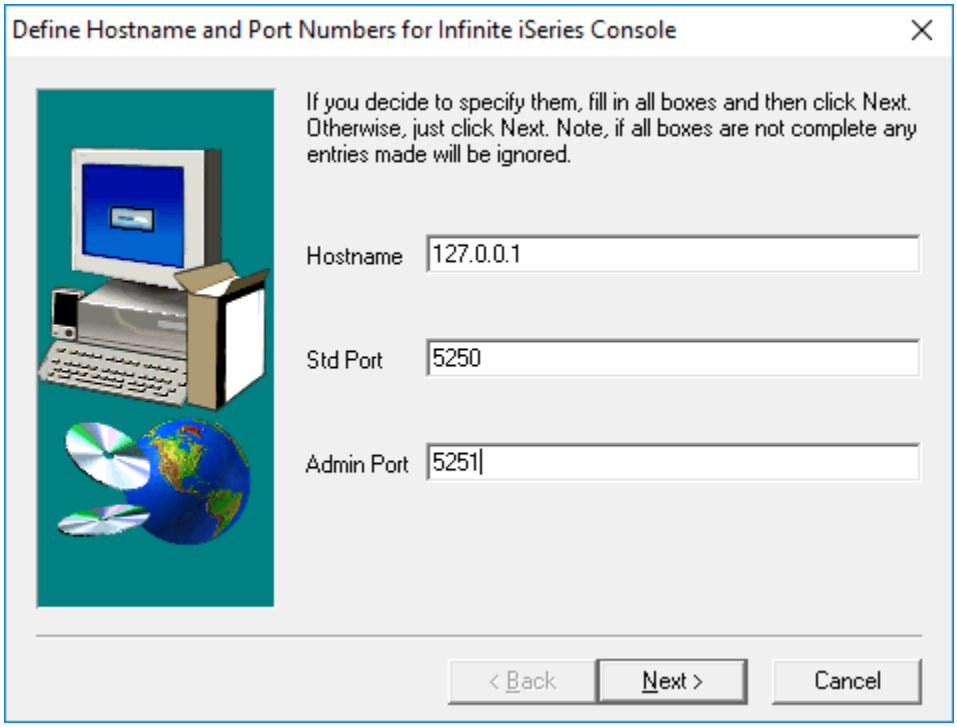


9. Use the following settings:

   Hostname: 127.0.0.1

   Std Port: 5250

   Admin Port: 5251

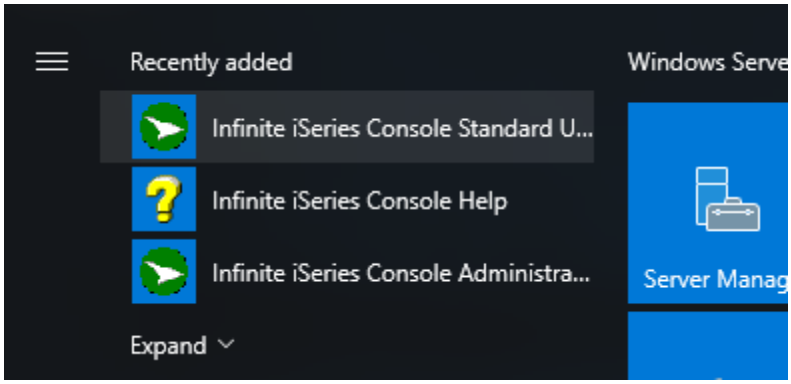


10. Click **Next** and follow the prompts to complete the setup.

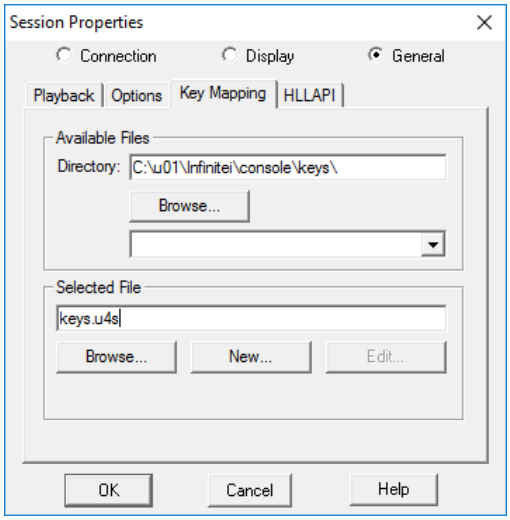
3.1 Configure Infinite i Console

This step configures the function keys on your keyboard to work like the function keys in AS/400.

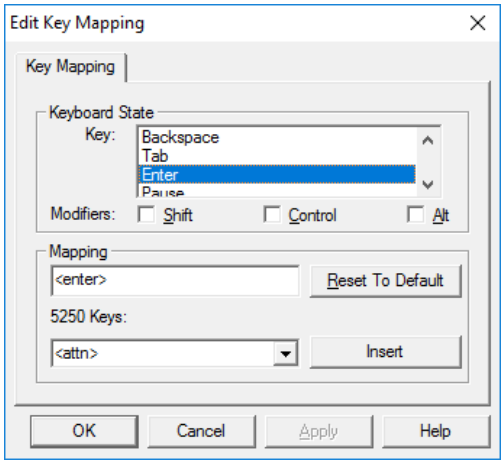
1. Click **Start** and locate the recently added Infinite iSeries Console program group. Expand the list to display all the groups.
2. Right-click **Infinite iSeries Console Standard** and click **Run as administrator**. Ignore any connection error messages.



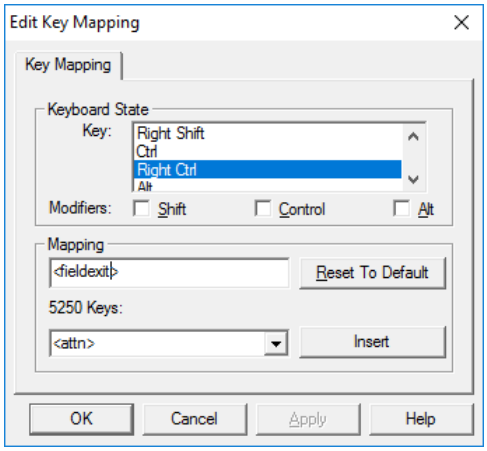
3. Click **Session** > **Properties** > **General** > **Key Mapping**.
4. For **Directory**, type **C:\u01\infinitei\console\keys\,** and for **Selected File**, type **keyx.u4s,** and then click **New**.



5. For **Key**, click **Enter**, then in **Mapping**, change the value to **<enter>**.

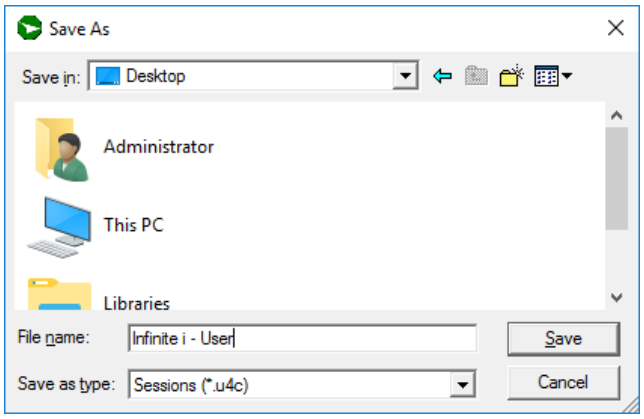


6. For **Key**, scroll down and click **Right Ctrl**, then in **Mapping**, change the value to **<fieldexit>**.



7. Click **OK** to exit **Edit Key Mapping**, then click **OK** to close **Session Properties**.

8. Click **File** > **Save As** to save the session properties. For **Save in**, specify **Desktop**. For **File name**, type **Infinite i - User.u4c** and click **Save**.

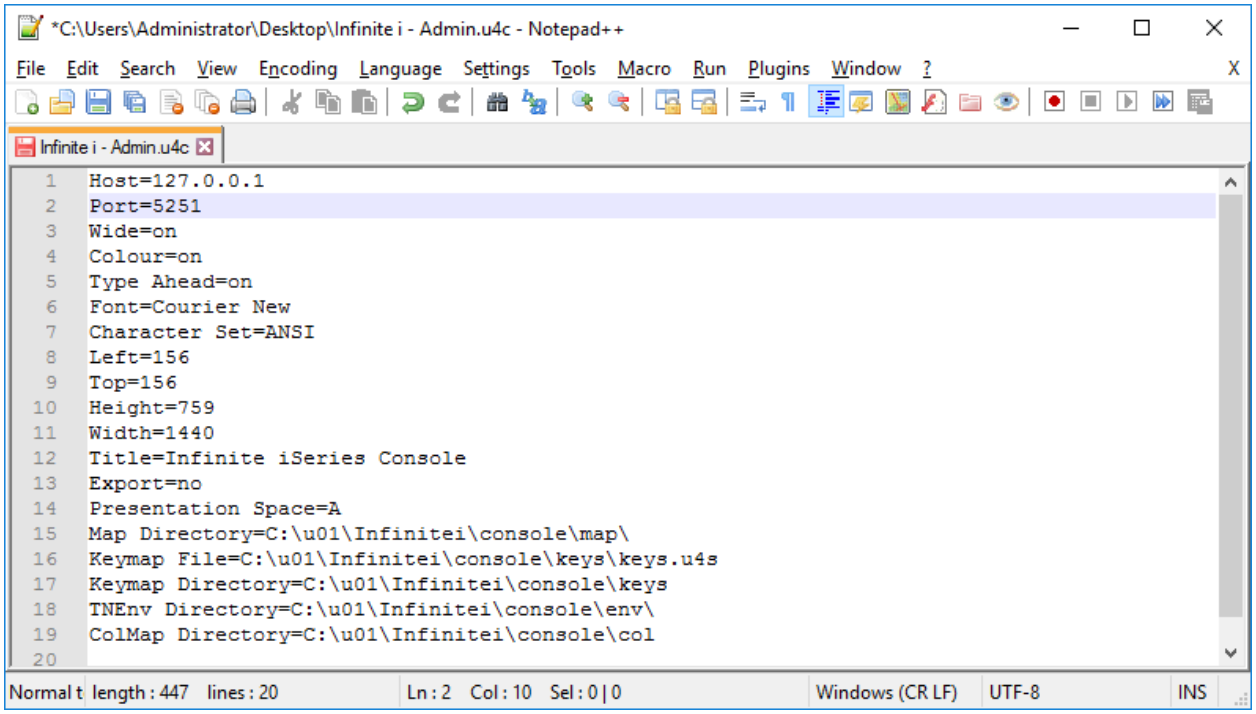


9. On the desktop, copy the **Infinite i - User.u4c** icon, then paste it to create a duplicate.

10. Rename the copy **Infinite i - Admin.u4c**.

11. Right-click the **Infinite i - Admin.u4c** icon and choose **Open with Notepad++**.

12. Change **Port=** to **5251.**



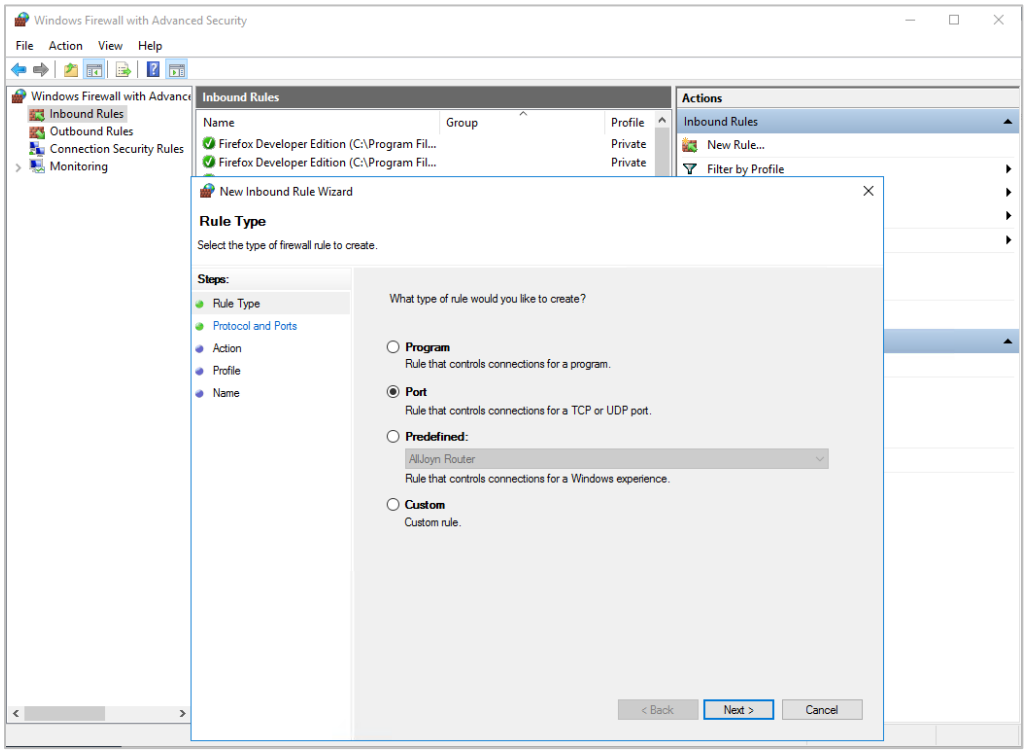
13. Save your changes and exit Notepad++.

3.2 Configure Windows Firewall

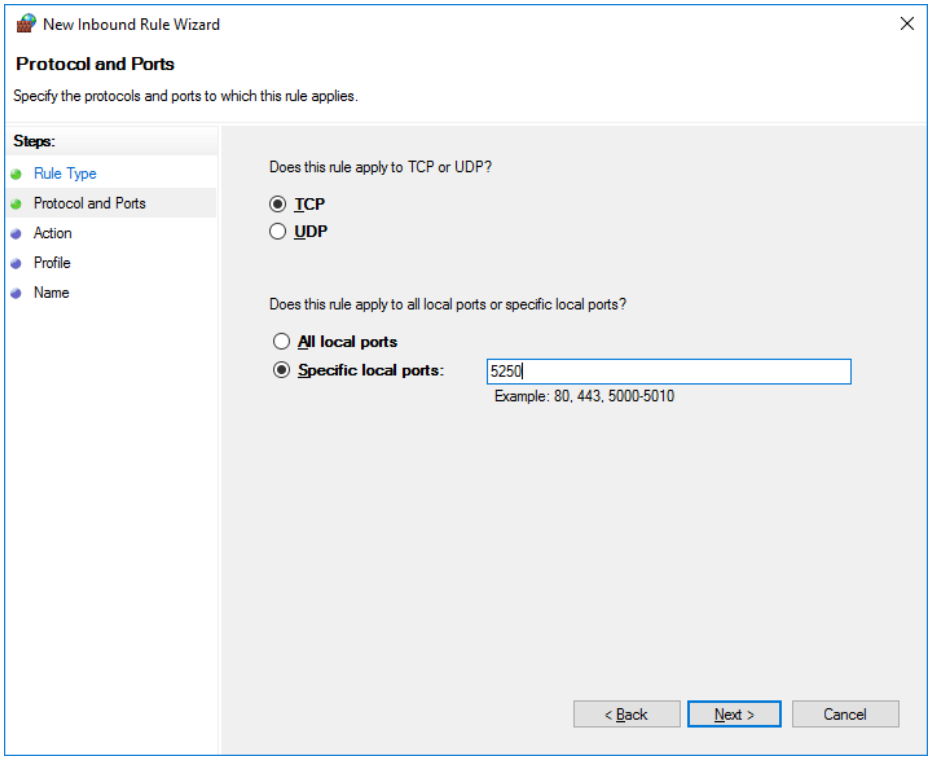
This step configures the rules for incoming network connections and packets to allow Infinite i to use TCP port 5250, SMB port 139, and internet port 8080.

1. Open **Control Panel** and go to **System and security** > **Windows Defender Firewall** > **Advanced Settings**.
2. Click **Inbound Rules**, then click **New Rule**.

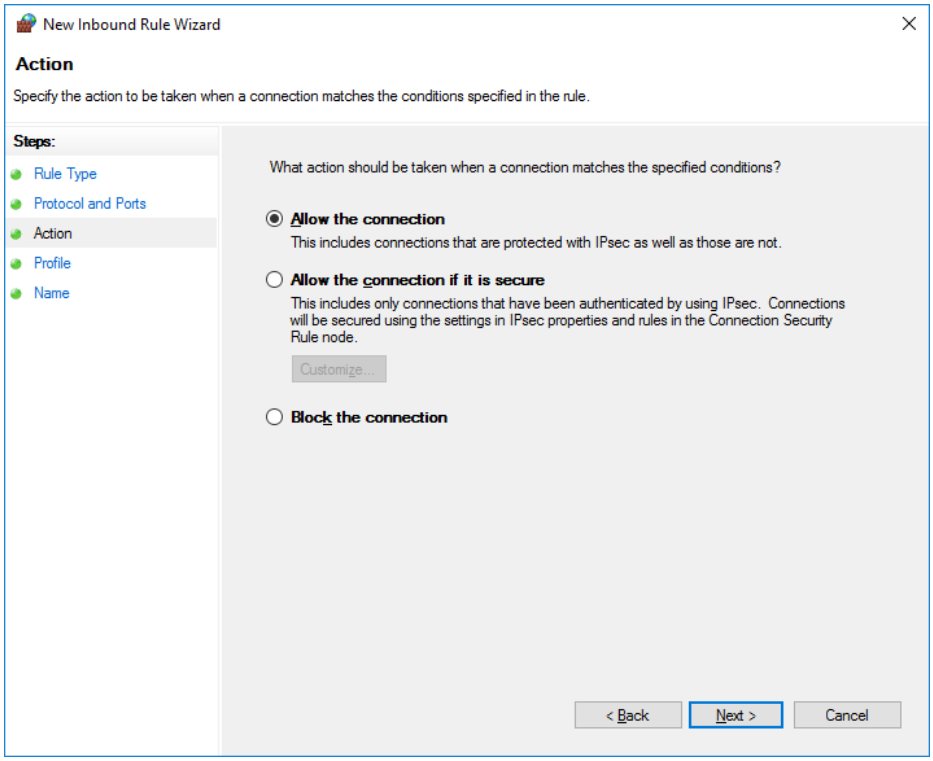
3. In **Rule Type**, click **Port**, and then click **Next**.



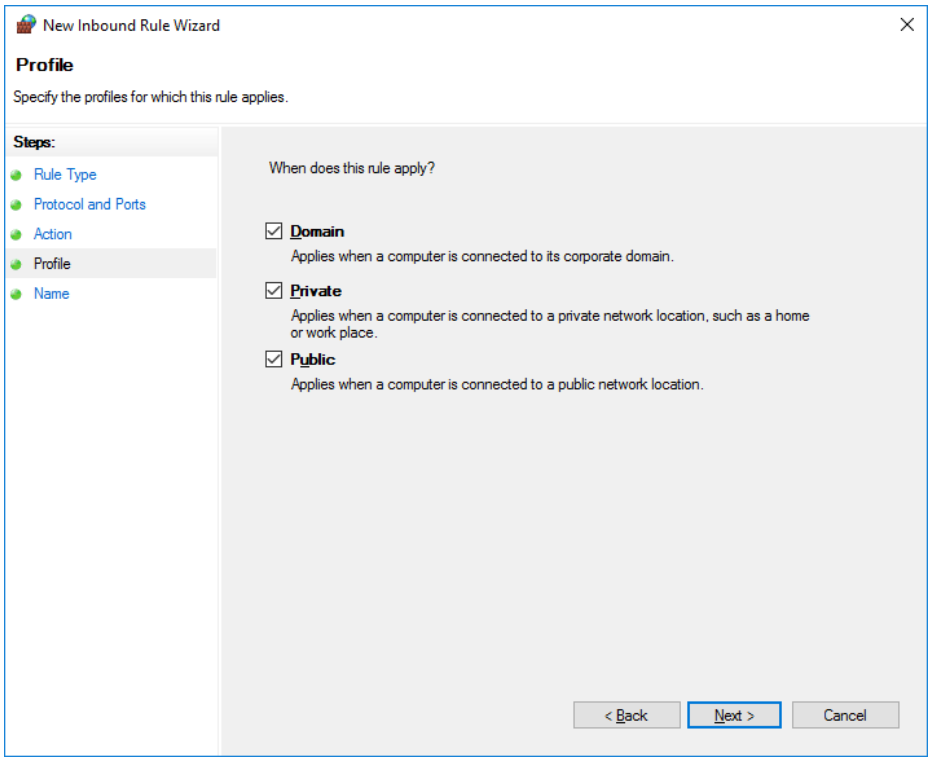
4. In **Protocol and Ports**, select **Specific local ports**, type **5250** and click **Next**.



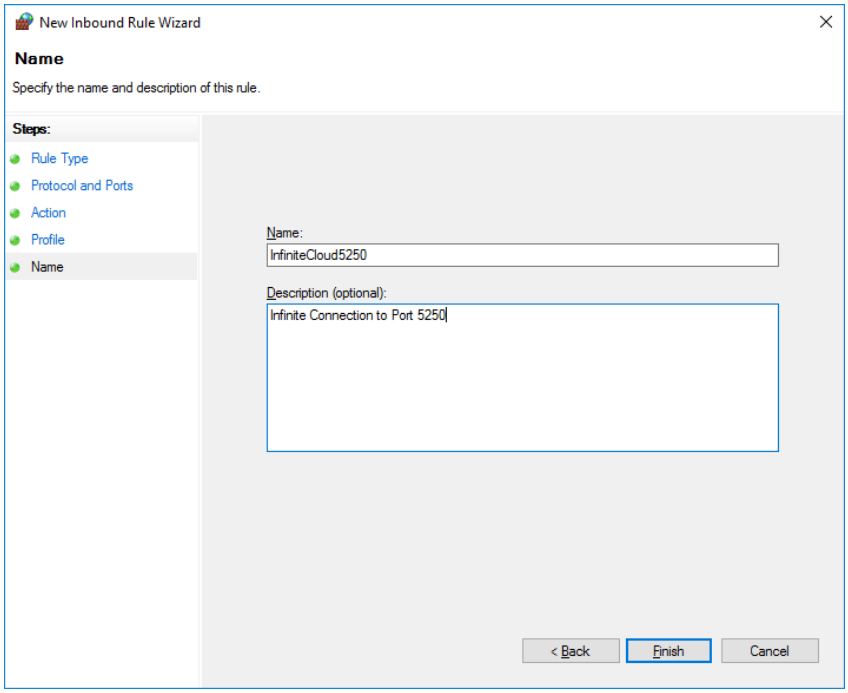
5. In **Action**, click **Allow the connection,** and then click **Next**.



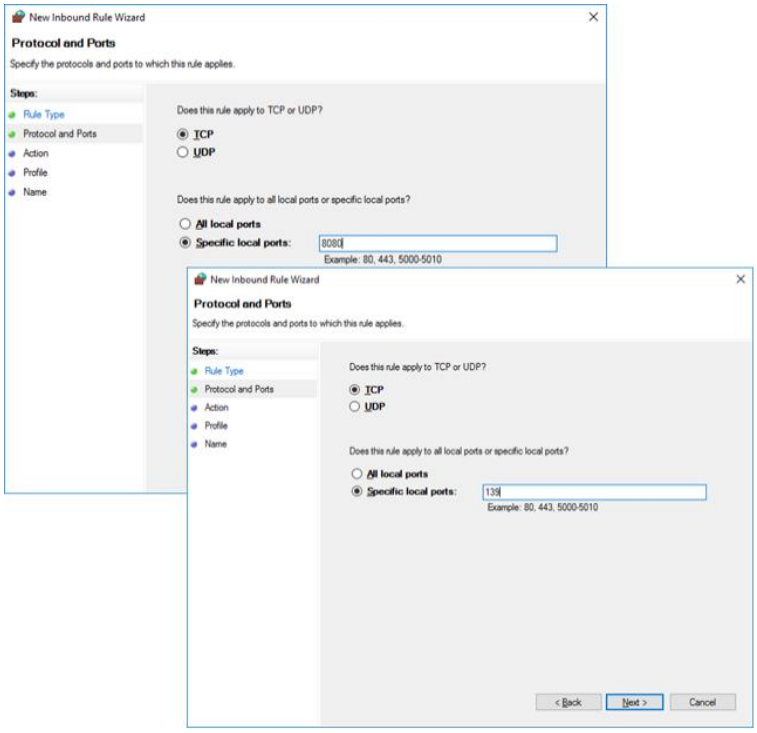
6. In **Profile**, accept the default options and click **Next**.



7. In **Name**, type **InfiniteCloud5250**, and add a description if you want. This makes the rule easier to locate later. Click **Finish**. The new rule appears in **Inbound Rules**.



8. Repeat the previous steps to add two more rules that apply to specific local ports. Add a rule for port **8080** named **InfiniteCloud8080** and a rule for Port **139** named **InfiniteCloud139**.

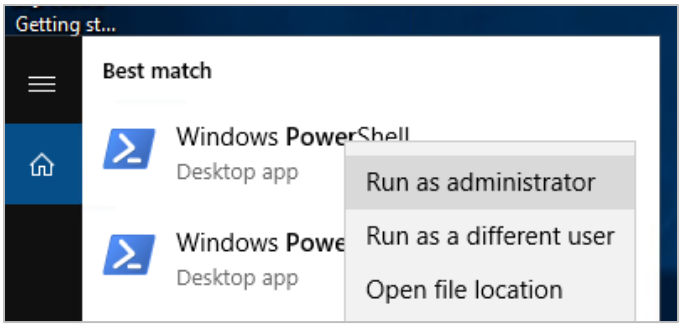


9. Close the firewall settings.

4 Install Infinite i

The next step is to install the Infinite i client. You need the user credentials that Infinite i provided with your software license.

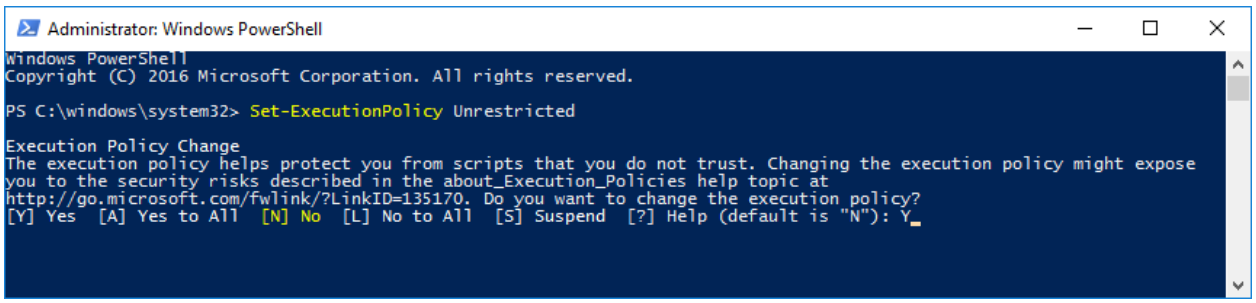
1. Run the Infinite-V13.16.00.msi file that was provided, or download it from **Host: sftp.infinitecorporation.com**
2. For **User** and **Pass**, type the user credentials that you received.
3. For **File**, type **installs\Infinitei-V13.00.03.msi**.
4. Place the downloaded file in **C:\u01\installs**.
5. Right-click **Windows PowerShell** and click **Run as administrator**.



6. On the command line, run:

```
Set-ExecutionPolicy Unrestricted
```

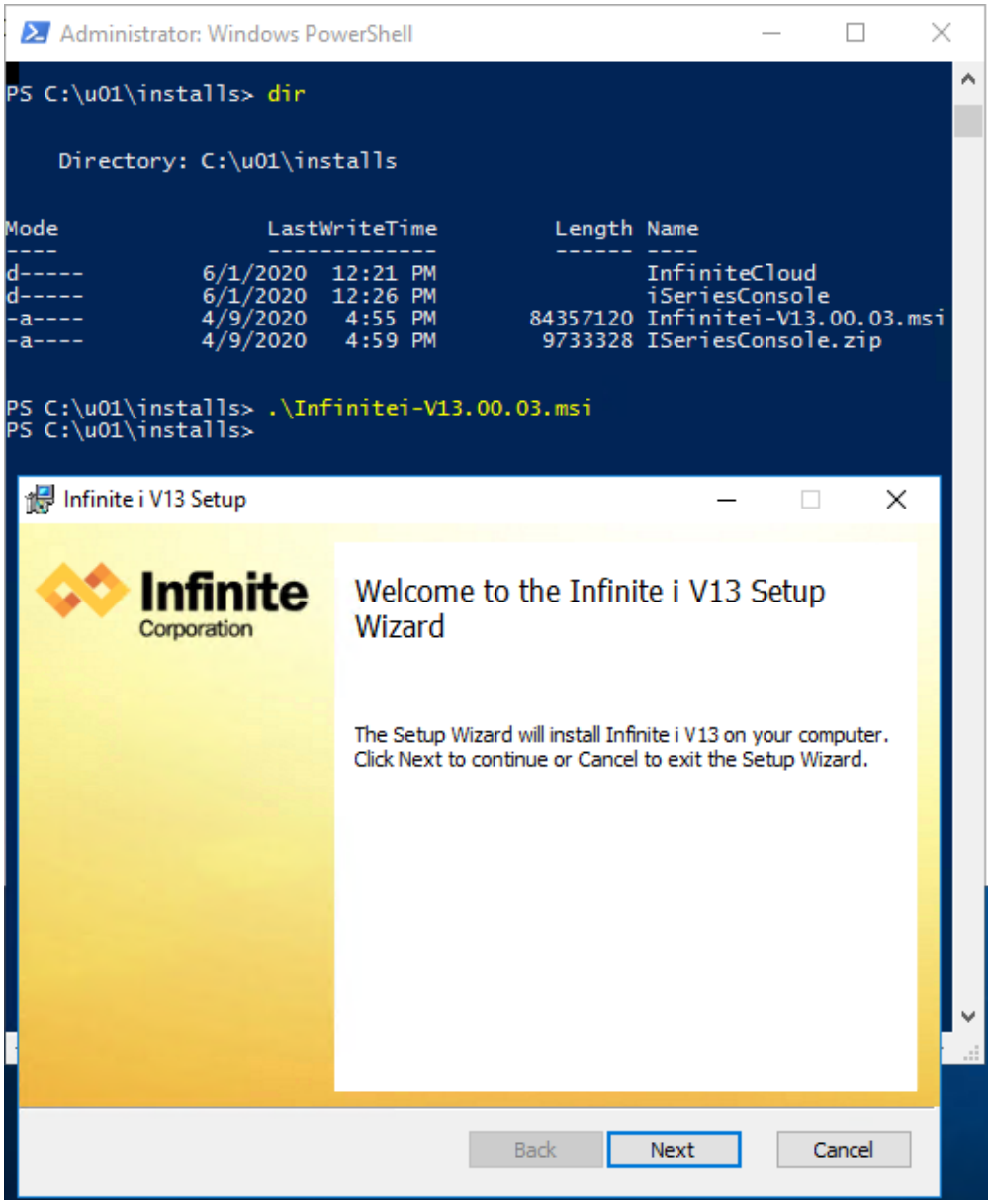
7. When prompted, type **Y** and press **Enter**.



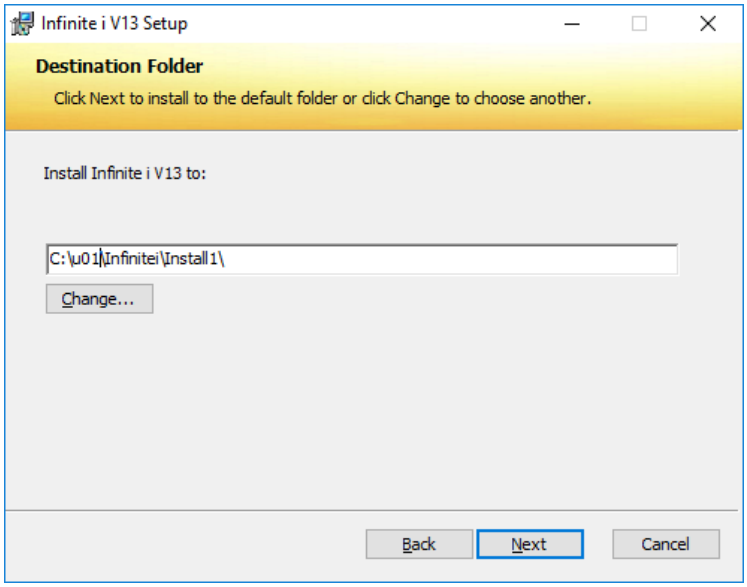
8. On the command line, run the following two commands:

```
cd c:\u01\installs

.\Infinitei-V13.00.03.msi
```

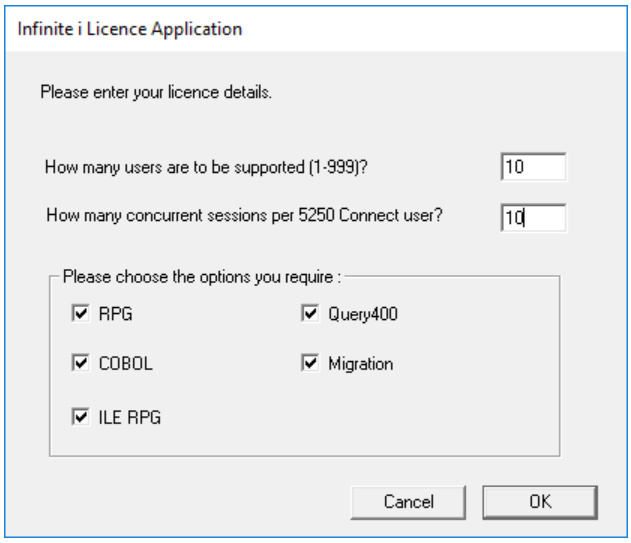
The Infinite i Setup Wizard starts.


Administrator: Windows PowerShell
PS C:\u01\installs> dir
Directory: C:\u01\installs
Mode LastWriteTime Length Name
d----- 6/1/2020 12:21 PM InfiniteCloud
d----- 6/1/2020 12:26 PM iSeriesConsole
-a---- 4/9/2020 4:55 PM 84357120 Infinitei-V13.00.03.msi
-a---- 4/9/2020 4:59 PM 9733328 ISeriesConsole.zip
PS C:\u01\installs> .\Infinitei-V13.00.03.msi
PS C:\u01\installs>
Infinite i V13 Setup
Infinite
Corporation
Welcome to the Infinite i V13 Setup Wizard
The Setup Wizard will install Infinite i V13 on your computer.
Click Next to continue or Cancel to exit the Setup Wizard.
Back
Next
Cancel

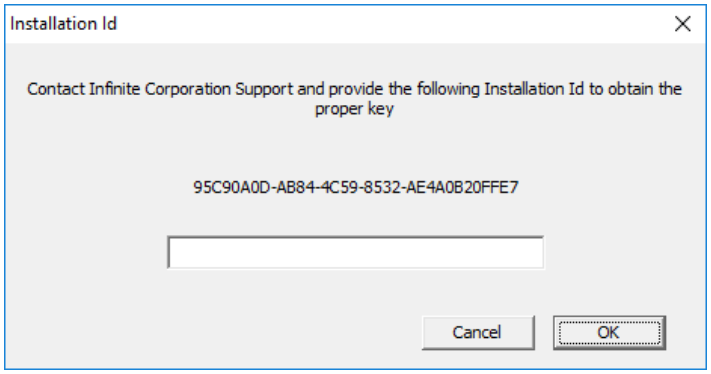
9. In **Destination Folder**, type **c:\u01\infinitei\install1** and click **Next**.



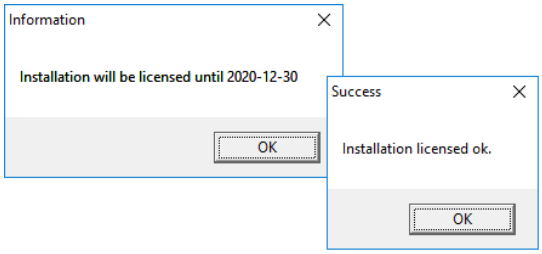
10. In **Infinite i License Application**, type the number of users and sessions, and check the options for your license based on the information provided by Infinite Corporation, then click **OK**.



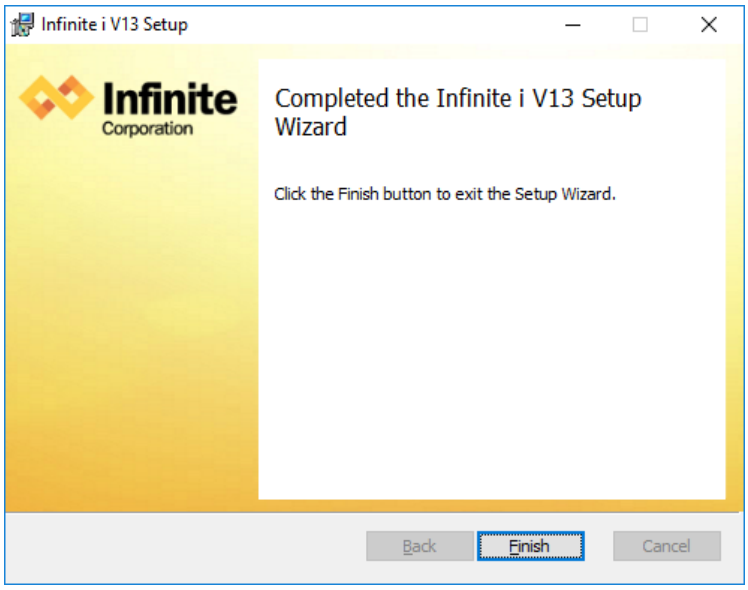
11. In **Installation Id**, copy the installation ID that appears and email it to Infinite Corporation (support@infinitecorporation.com). Please copy your technical support representative.



12. When you receive the GUID from Infinite Corporation, enter it in the input box and click **OK** to continue the installation process.



13. Click **Finish** to exit the Setup Wizard and continue with the next steps.



4.1 Configure system environmental variables

This step defines the system paths that the software requires.

1. Edit the **c:\infinitei\install1\bin\u4attach** file and replace it with the following content:

```
@echo off
set U4DFLTDIR=C:\u01\infinitei\install1
set U4SYSID=install1
set INCLUDE=%INCLUDE%;C:\u01\infinitei\install1\include;
set LIB=%LIB%;C:\u01\infinitei\install1\lib;
set PATH=%PATH%;C:\u01\infinitei\install1\lib;C:\u01\infinitei\install1\bin;
C:\u01\infinitei\Install1\dlls;C:\u01\infinitei\install1\sysdlls;
set _NT_SYMBOL_PATH=C:\u01\Infinitei\Install1\pdbs
```

2. Edit the **C:\u01\infinitei\install1\etc\u4env** file and add the following entries:

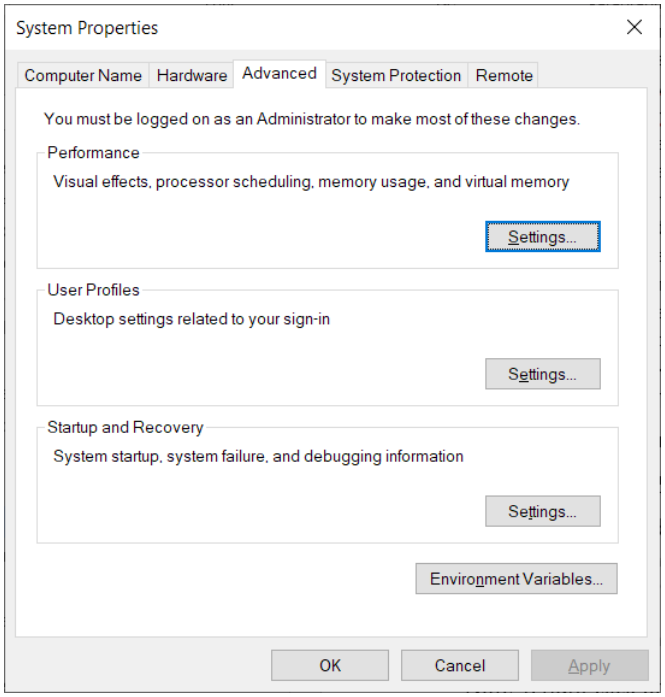
```
File c:\u01\infinitei\install1\etc\u4env:
VSINSTALLDIR=C:\Program Files (x86)\Microsoft Visual Studio 14.0
SDKINSTALLDIR=C:\Program Files (x86)\Microsoft SDKs\Windows\v7.0A
U4LIBDIR=C:\u01\libs
U4ORDMEMSIZE=4
U4SORTMAXFD=8192
ISAMFILES=250
ISAMOPENS=250
MSS_HOME=MSSQLSVR
U4DBCUSER=sa
U4DBCPASSWD=sa_password@infiniteiDSN

Note: The lines highlighted apply only with the implementation of SQL Server.

DSN configured on ODBC Data Source Administrator (64-bit):
DSN name: infiniteiDSN
SQL Server: SQLSERVER_name
Default Database: infiniteidb
Username: sa
Password: sa_password
Microsoft SQL Server ODBC Driver Version 06.03.9600
Data Source Name: infiniteiDSN
Data Source Description: infiniteidb
Server: SQLSERVER_name
Database: infiniteidb
Language: (Default)
Translate Character Data: Yes
Log Long Running Queries: No
Log Driver Statistics: No
Use Regional Settings: No
```

```
Prepared Statements Option: Drop temporary procedures on disconnect
Use Failover Server: No
Use ANSI Quoted Identifiers: Yes
Use ANSI Null, Paddings and Warnings: Yes
Data Encryption: No
```

3. Edit the **C:\u01\infinitei\install1\etc\U4SYS.sdc** file and add the following:

```
QMAXDBMEMSZ=16
```

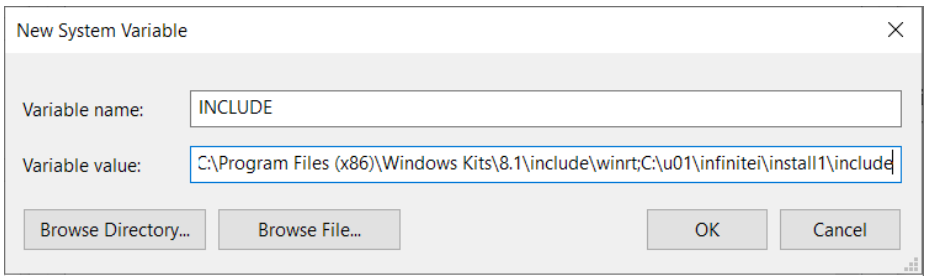
4. To edit the system environment variables, go to **Start** > **System** > **Advanced System Settings**.



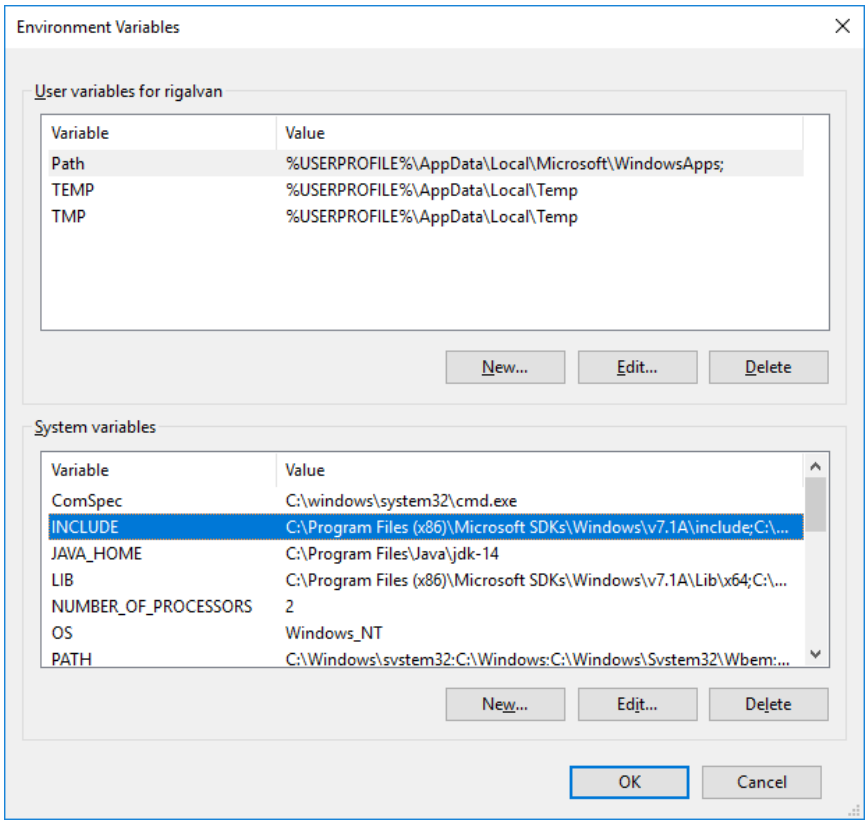
5. Click **Environment Variables**. Under **System Variables**, click **New.** In **New System Variable**, add the following system environment variables, separated by a semi-colon (;).

> **Note:** Do not leave spaces between path entries.

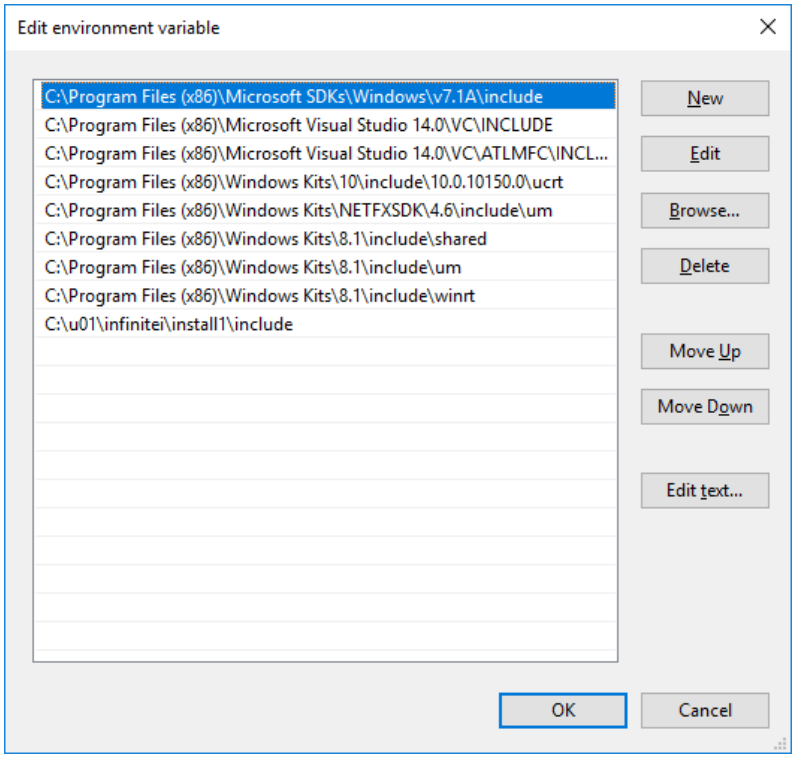
```
INCLUDE=
C:\Program Files (x86)\Microsoft SDKs\Windows\v7.1A\include;C:\Program Files
(x86)\Microsoft Visual Studio 14.0\VC\INCLUDE;C:\Program Files (x86)\Microsoft
Visual Studio 14.0\VC\ATLMFC\INCLUDE;C:\Program Files (x86)\Windows
Kits\10\include\10.0.10150.0\ucrt;C:\Program Files (x86)\Windows
Kits\NETFXSDK\4.6\include\um;C:\Program Files (x86)\Windows
Kits\8.1\include\shared;C:\Program Files (x86)\Windows
Kits\8.1\include\um;C:\Program Files (x86)\Windows
Kits\8.1\include\winrt;C:\u01\infinitei\install1\include
```



6. Click **OK** to return to **Environment Variables**.



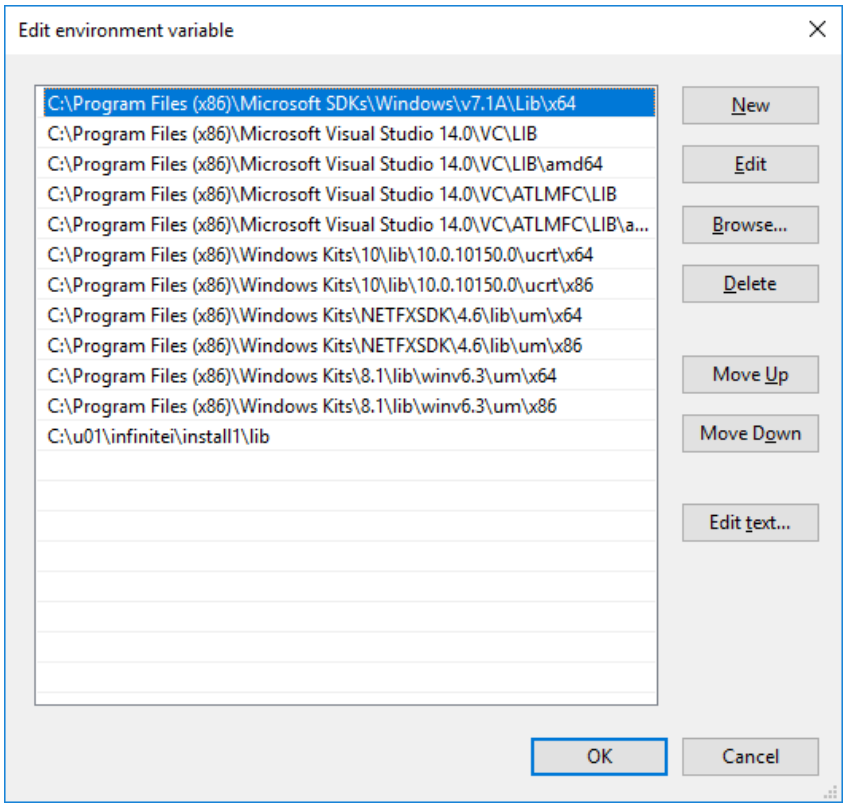
7. To verify the variable, double-click the INCLUDE line. The following screen is displayed:



8. To add the LIB variables, click **New**. In **New System Variables**, follow the same procedure as before to add the following system environment variables, separated by a semi-colon (;).

```
LIB=
C:\Program Files (x86)\Microsoft SDKs\Windows\v7.1A\Lib\x64;C:\Program Files
(x86)\Microsoft Visual Studio 14.0\VC\LIB;C:\Program Files (x86)\Microsoft Visual
Studio 14.0\VC\LIB\amd64;C:\Program Files (x86)\Microsoft Visual Studio
14.0\VC\ATLMFC\LIB;C:\Program Files (x86)\Microsoft Visual Studio
14.0\VC\ATLMFC\LIB\amd64;C:\Program Files (x86)\Windows
Kits\10\lib\10.0.10150.0\ucrt\x64;C:\Program Files (x86)\Windows
Kits\10\lib\10.0.10150.0\ucrt\x86;C:\Program Files (x86)\Windows
Kits\NETFXSDK\4.6\lib\um\x64;C:\Program Files (x86)\Windows
Kits\NETFXSDK\4.6\lib\um\x86;C:\Program Files (x86)\Windows
Kits\8.1\lib\winv6.3\um\x64;C:\Program Files (x86)\Windows
Kits\8.1\lib\winv6.3\um\x86;C:\u01\infinitei\install1\lib
```

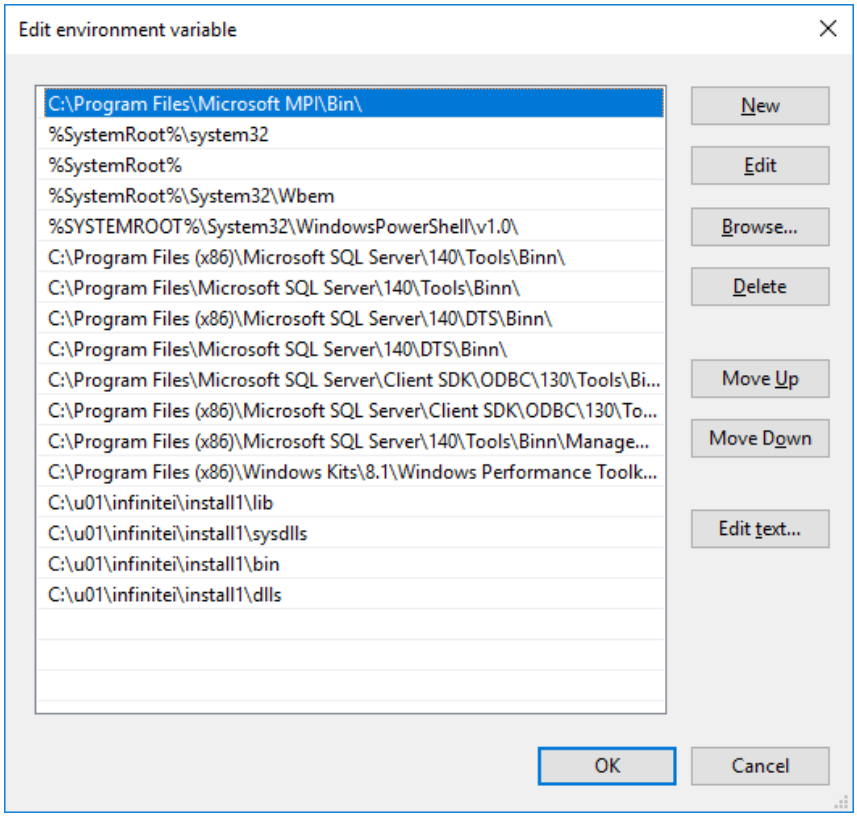
9. Click **OK**.



10. To add the PATH variables, click **New.** In **New System Variable**, follow the same procedure as before to add the following system environment variables, separated by a semi-colon (;).

```
PATH=

C:\Windows\system32;C:\Windows;C:\Windows\System32\Wbem;C:\Windows\System32\Windo
wsPowerShell\v1.0\;C:\Program Files (x86)\Microsoft Visual Studio
14.0\Common7\IDE\PrivateAssemblies\;C:\Program Files (x86)\Windows
Kits\8.1\Windows Performance
Toolkit\;C:\u01\infinitei\install1\lib;C:\u01\infinitei\install1\sysdlls;C:\u01\i
nfinitei\install1\bin;C:\u01\infinitei\install1\dlls
```

11. Click **OK**.



12. Click **OK** to close the **Environment Variable** settings.

> **Note:** You can also query the registry for the variables using the SET command from the command line.

13. Use the **Infinite i - Admin.u4c** icon on the desktop to start Infinite i.
14. On the **Sign On** screen, select **F10=Alternative Signon** and sign in using your Windows Administrator credentials.
15. Go to **System Administration Menu** > **Stop Admin Daemon**.
16. Press **Page Down** until no more new messages are shown, then press **Enter**.
17. Go to **Administrative Tools** > **Services and Stop** > **Start the Infinite i Windows service INFINITEISERV**.
18. Go to **Generate System** and press **Enter**, then in the **System Generation** screen, press **Enter**. The following message appears:

```
System definition file created for system U4SYS" press ENTER
```

19. Go to **Start Admin Daemon** and press **Enter**. Press **Page Down** until no more new messages are shown and press **Enter**.
20. Press **F3**, go to **Sign off** and press **Enter**.
21. Close the Infinite Console window.
22. To make testing easier, create a desktop shortcut named **U4** with the following target:

```
%windir%\System32\cmd.exe /K "prompt
$n:\$g&&c:\u01\infinitei\install1\bin\u4attach.bat&&c:\cygwin64\cygwin.bat"
```

23. Test the shortcut by double-clicking the **U4** shortcut and running the **u4lslib** command. A list of libraries appears.
24. On the Infinite i Console, use the **Infinite i - User** icon to sign in to Infinite i.
25. On the **Sign On** screen, select **F10=Alternative Signon** and sign in using your Windows Administrator credentials.

4.2 Test the compilers

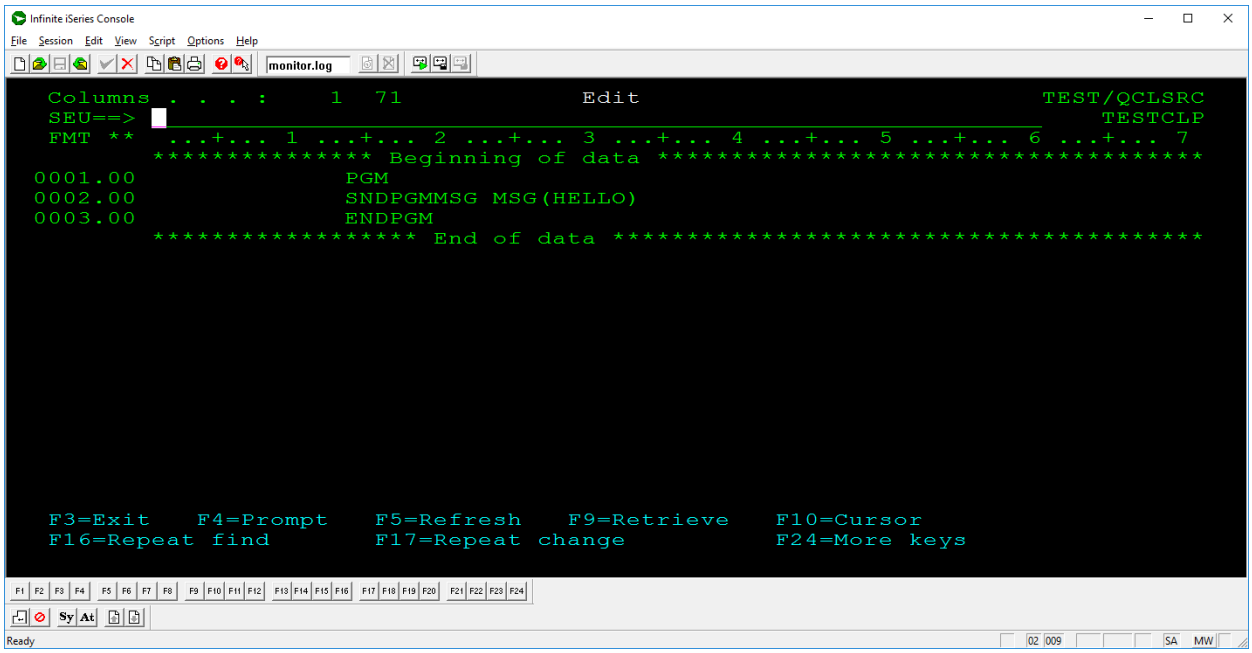
In this step, you create and compile a program to test the compilers.

1. In Infinite i, start SEU.
2. Test the compilers by creating and compiling any program:

```
CRTLIB TEST
CRTSRCPF FILE(TEST/QCLSRC)
STRSEU SRCFILE(TEST/QCLSRC) SRCMBR(TESTCLP) TYPE(CLP)
```

3. Add the following commands:

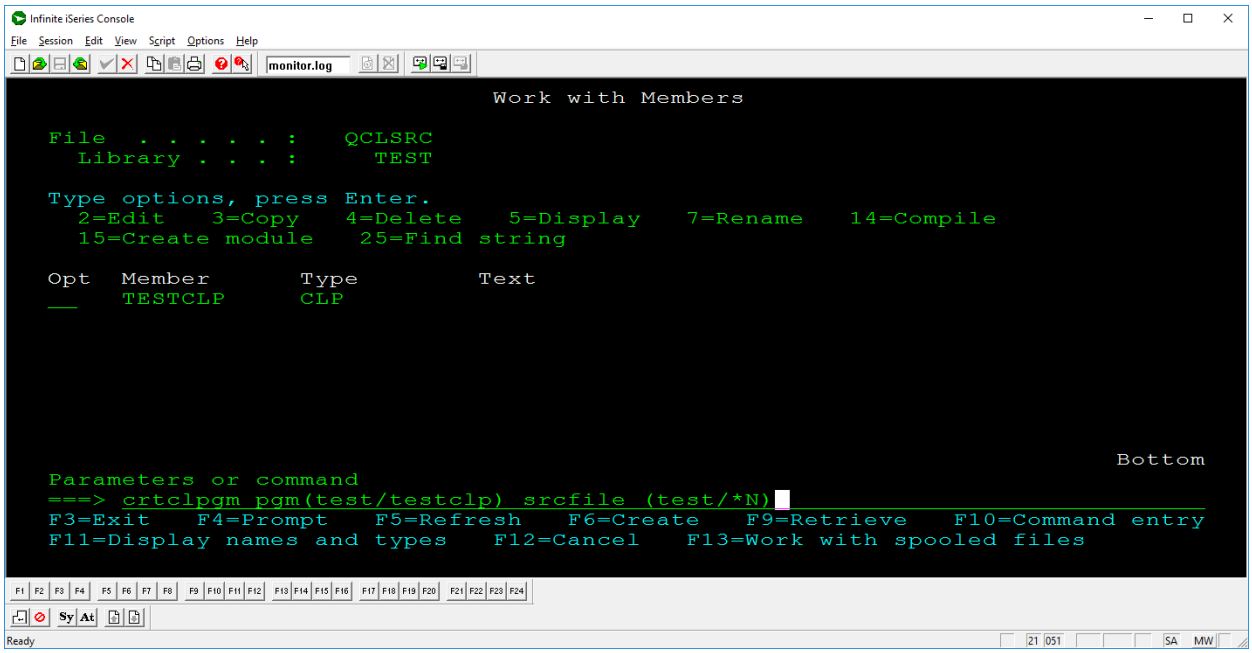
```
0000.01 PGM
0000.02 SNDPGMMSG MSG(HELLO)
0000.03 ENDPGM
```



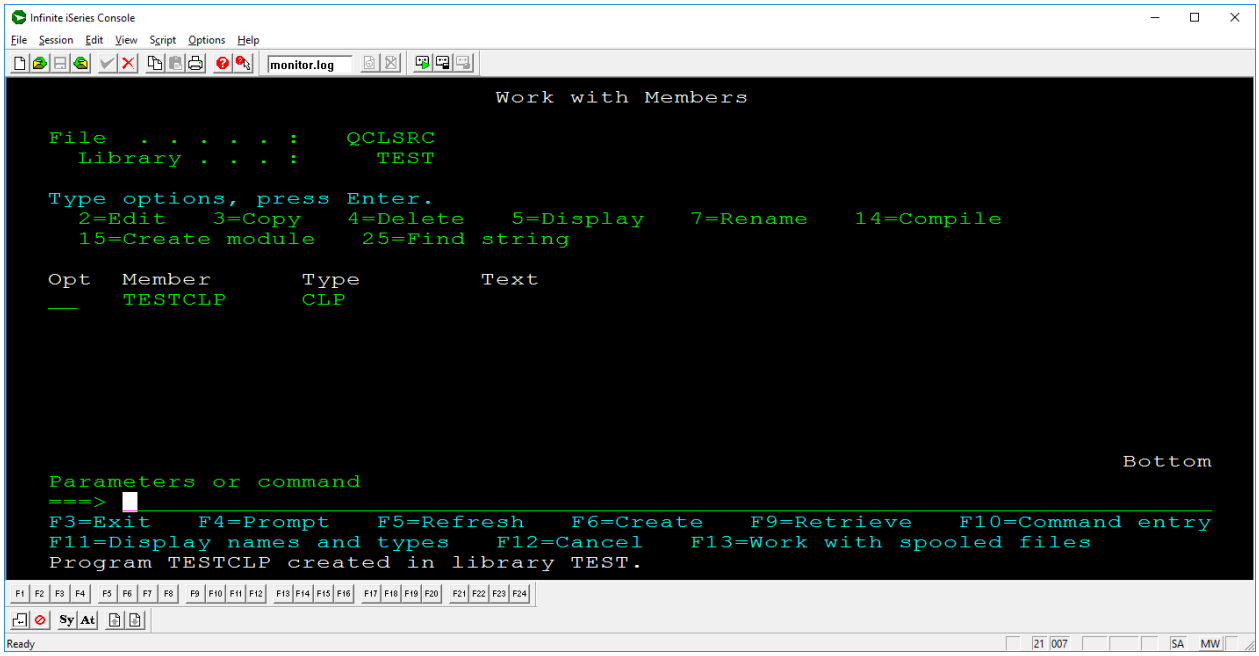
4. Press **F3** to save the source and exit SEU.

5. At the prompt, type:

```
===>crtclpgm pgm(test/testclp) srcfile(test/*n)
```



The output shows that the TESTCLIP program was created in the TEST library.



4.3 Restore libraries provided by Infinite Corporation

In this step, you copy the library backup files (*.u4s) provided by Infinite Corporation to the c:\u01\backups folder.

1. Open the U4 command prompt and run:

```
cd to /u01/backups
```

2. If needed, unzip the files using the **unzip *.zip** command.

3. *Optional*. Archive the original zip files using these commands:

```
mkdir zip
mv *.zip zip/
```

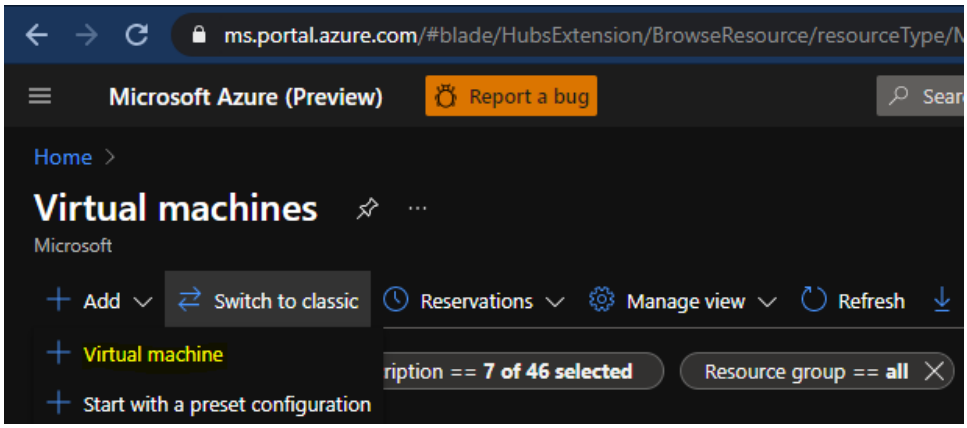
4. To restore the libraries, execute the following command:

```
for l in `ls /u01/backups//*.u4s | cut -d/ -f5 | grep -v "^Q"` ; do echo
Restoring library
$l...; u4go -x "U4RSTLIB PATH('c:\\u01\\backups\\${l}')" ; done;
```

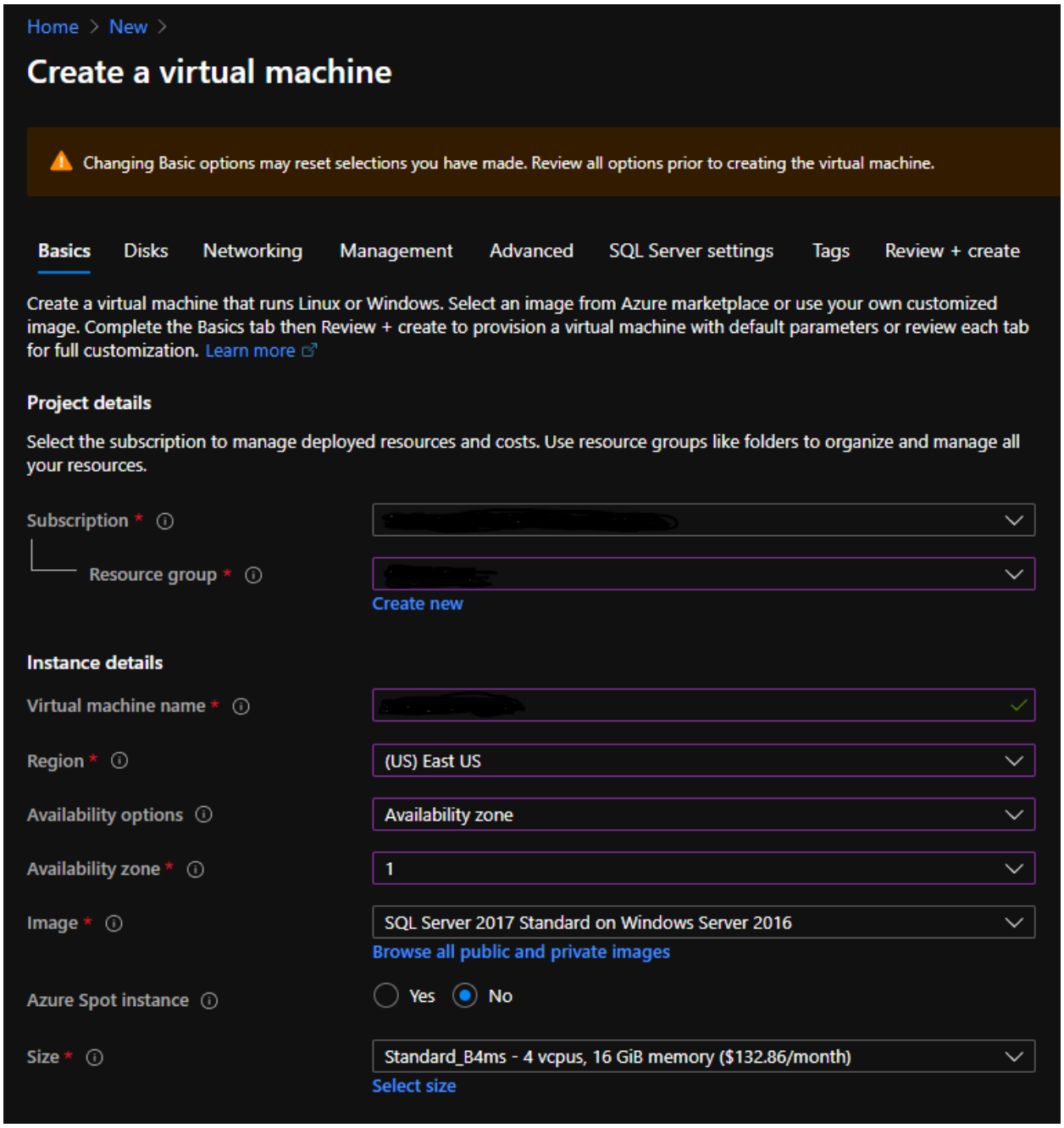
5 Create the database on Azure

In this step, you work in Azure portal to create a virtual machine (VM) with high availability, availability zones, and an image of SQL Server 2017 Standard on Windows Server 2016. This VM is for the production environment.

1. Sign in to Azure portal (azure.portal.com), go to **Virtual Machines**, and select **Add Virtual Machine**.

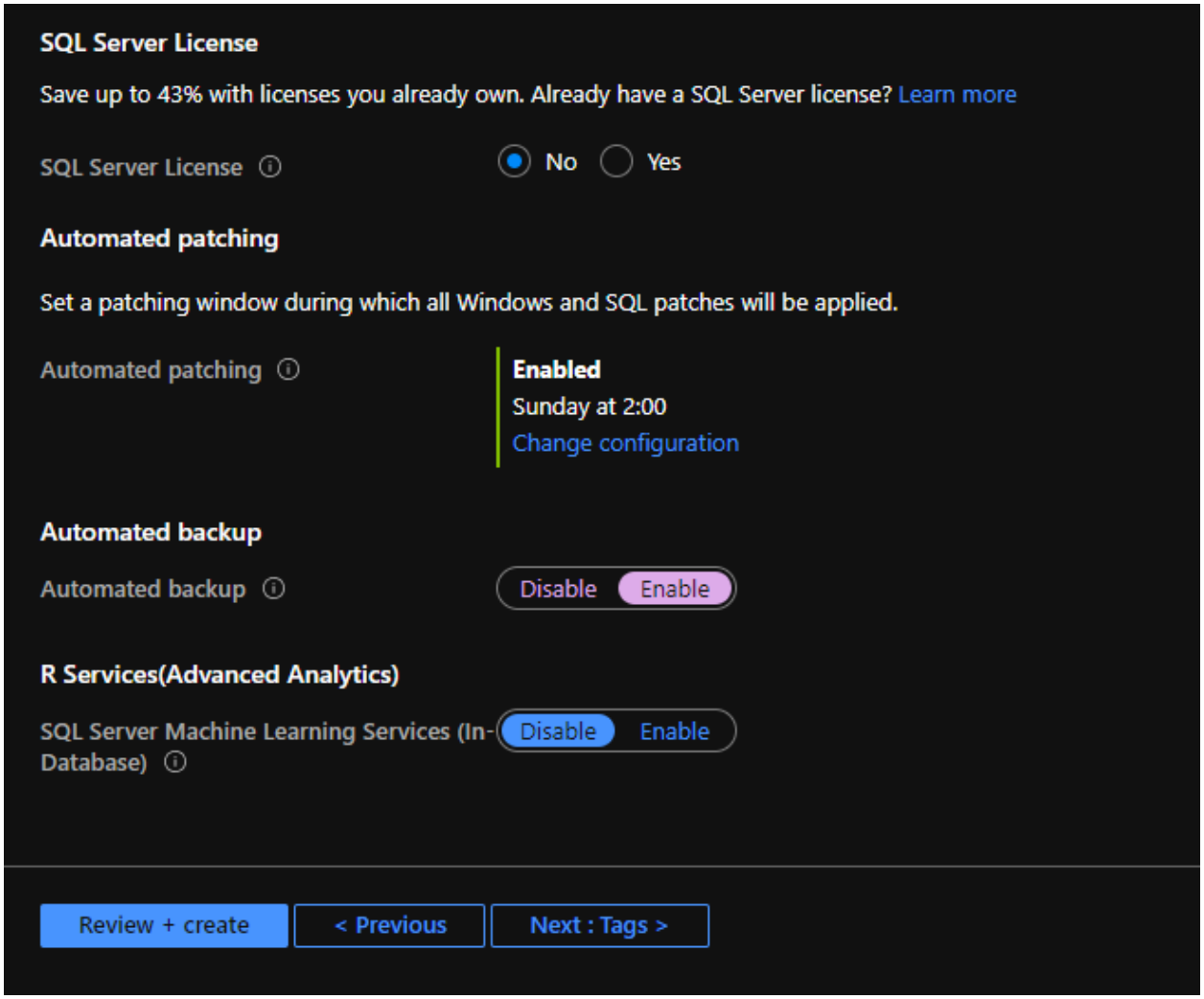


2. Enter your subscription details and choose a VM size based on your needs. The following example selects the Standard_B4ms VM.



3. Accept the default settings on the **Disks**, **Network**, **Management**, and **Advanced** tabs.

4. On the **SQL Server Settings** tab, set **Automated Backup** to **Enable**. This setting supports disaster recovery. For details, see [Automated Backup v2 for Azure virtual machines](#).



5. Complete the setup steps to create the VM with SQL Server.

6 *Optional*. Set up Infinite i database connection

The next step connects Infinite i to SQL Server. This step is optional because Infinite i includes DB2 emulation with its own file implementation. It also provides the option to convert the physical and logical files into a relational database, such as SQL Server, Oracle, and MySQL in Azure. You don't have to change the RPG, COBOL, or CL source code.

To use SQL Server, you must edit the Infinite i **u4env** configuration file. You did this in section 4.1, "Configure system environmental variables," when you edited the **u4env** configuration file to include the following environment variables:

C:\u01\infinitei\install1\etc\u4env file:

```
File c:\u01\infinitei\install1\etc\u4env:
VSINSTALLDIR=C:\Program Files (x86)\Microsoft Visual Studio 14.0
SDKINSTALLDIR=C:\Program Files (x86)\Microsoft SDKs\Windows\v7.0A
U4LIBDIR=C:\u01\libs
U4ORDMEMSIZE=4
U4SORTMAXFD=8192
ISAMFILES=250
ISAMOPENS=250
MSS_HOME=MSSQLSVR
U4DBCUSER=sa
U4DBCPASSWD=sa_password@infiniteiDSN
```

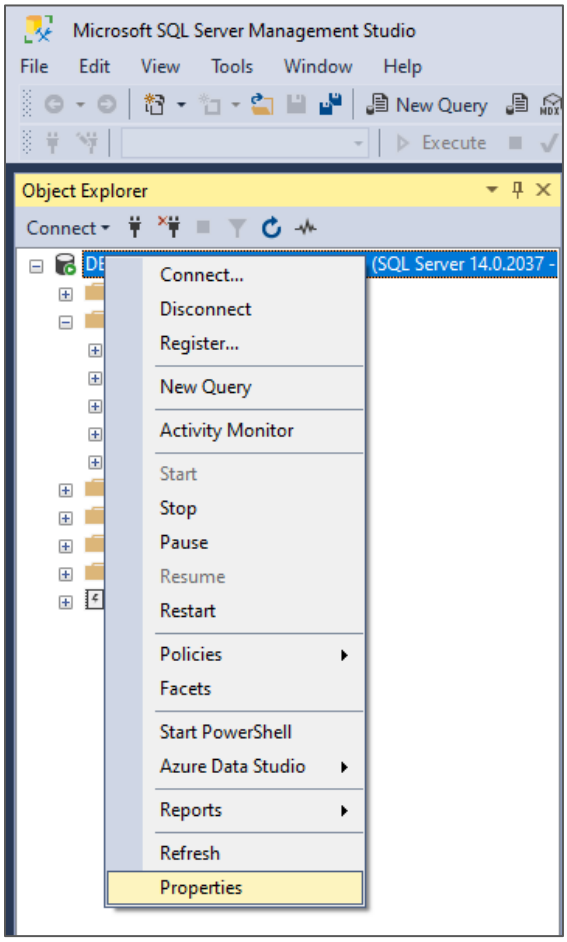
The highlighted lines are part of the configuration needed to use SQL Server with Infinite i.

The remaining steps in this section tell you how to set up SQL Server using SQL Server Management Studio, a visual tool for executing queries and stored procedures.

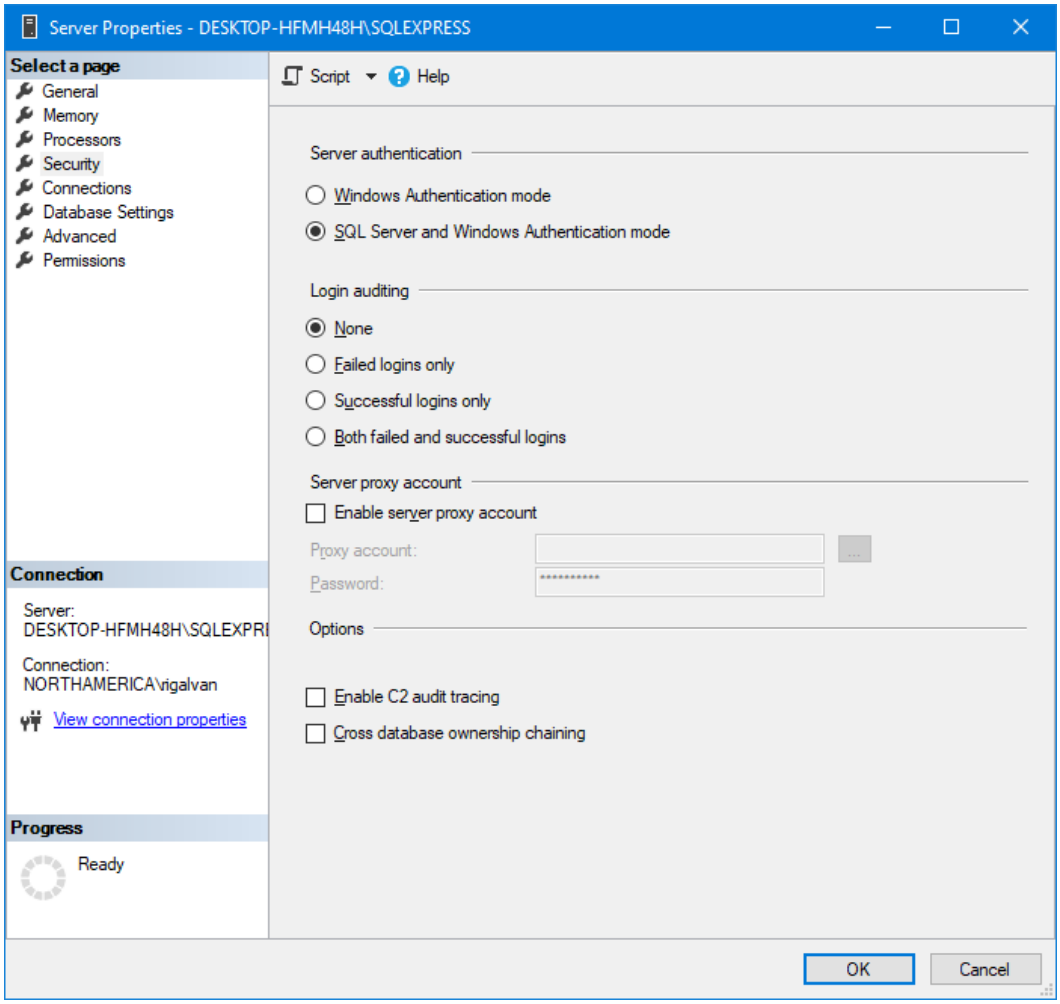
6.1 Set the SQL Server security mode

In this step, you use Management Studio to set up the login identity for the person or process that is connecting to an instance of SQL Server.

1. Start Management Studio and open **Object Explorer**.
2. Right-click the server instance in which you want to create the new login and select **Properties**.



3. In **Select a page**, right-click **Security** and select the following options:
   a. For **Server authentication**, select **SQL Server and Windows Authentication mode**.
   b. For **Login auditing**, choose **None**.

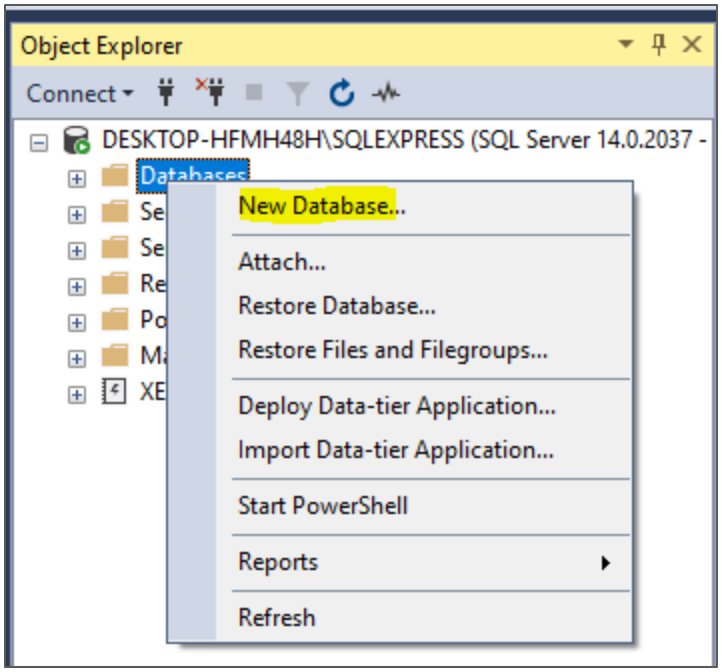


4. Click **OK**.
5. To create the login, see Create a Login in the SQL Server documentation.

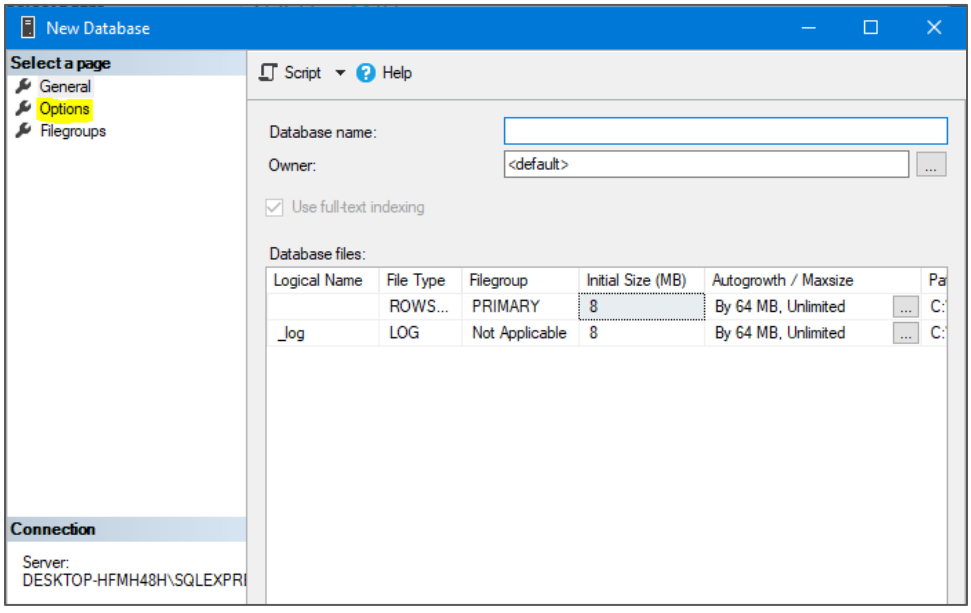
6.2 Create the SQL Server database

You can use Management Studio to create a new SQL Server database.

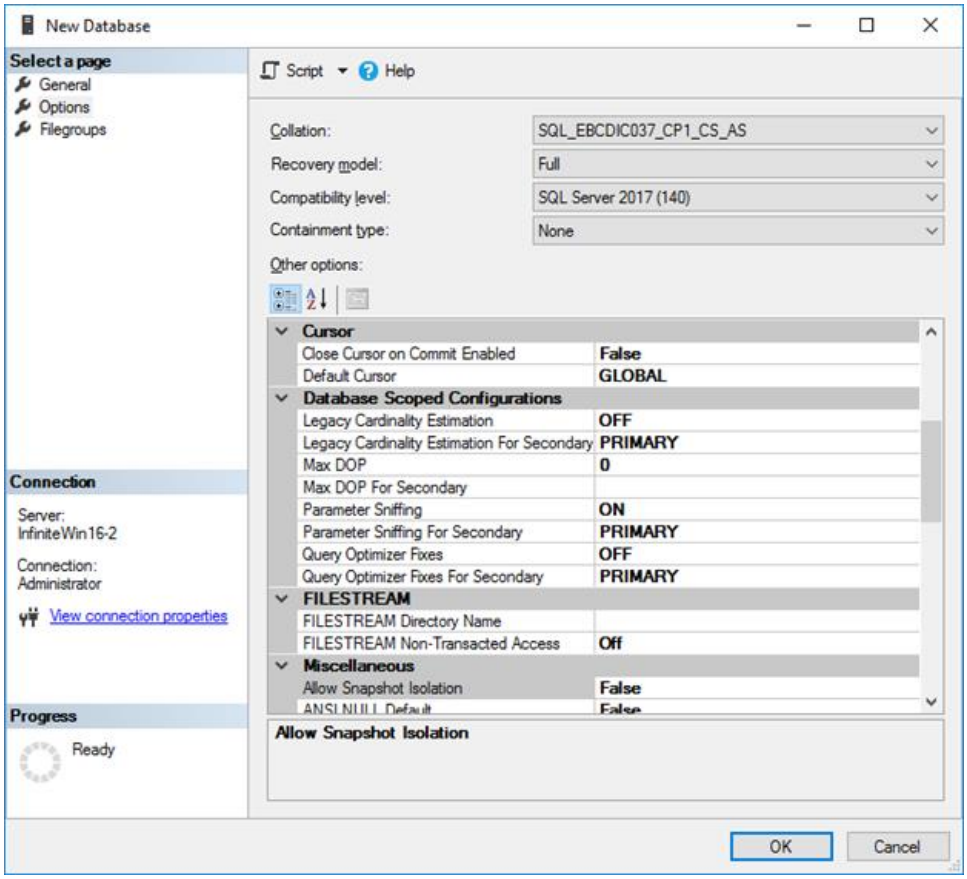
1. Start Management Studio and sign in using your Windows or SQL Server credentials.
2. In **Object Explorer**, right-click **Databases** and select **New Database**.



3. In **Database name**, type a name such as *Infiniteidb*. In **Owner**, accept the default or select an owner.



4. In **Select a page**, click **Options**.
5. If you want to use the EBCDIC data format as in the AS/400, in **Collation**, select **SQL_EBCDIC037_CP1_CS_AS**. Otherwise, accept the default and click **OK**.



6.3 Initialize the database

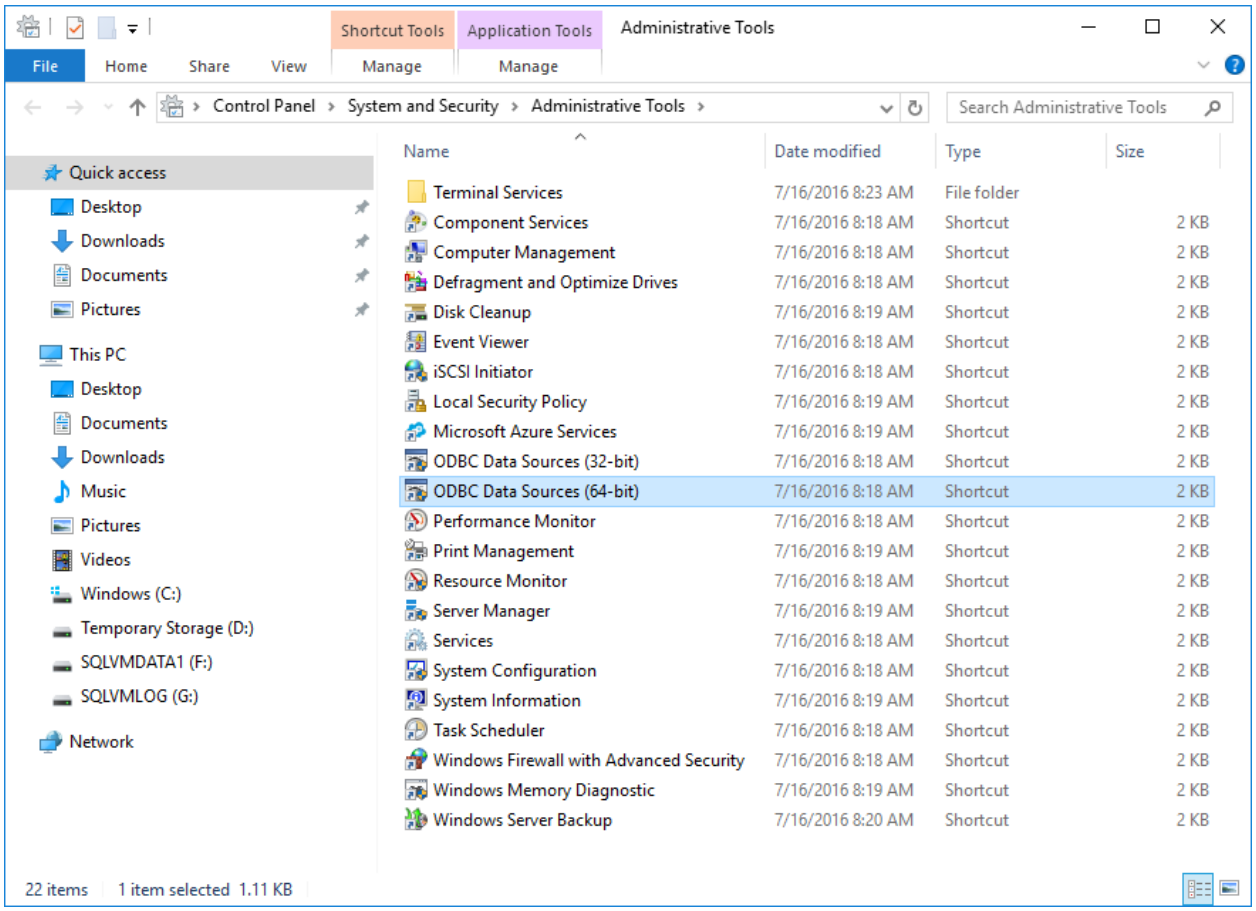
This step uses Management Studio, a visual tool for executing queries and stored procedures.

1. Sign in to Management Studio using the password you just created.
2. Select the **create_mapping_mss** file from **c:\u01\infinitei\install1\sql**.
3. In the **create_mapping_mss** file, press **Ctrl+E** to execute a query.
4. Repeat the procedure with the U4MSSLOCK2000.SQL file located in the same folder.
5. To execute the new procedure, press **Ctrl+E**.
6. Close Management Studio.

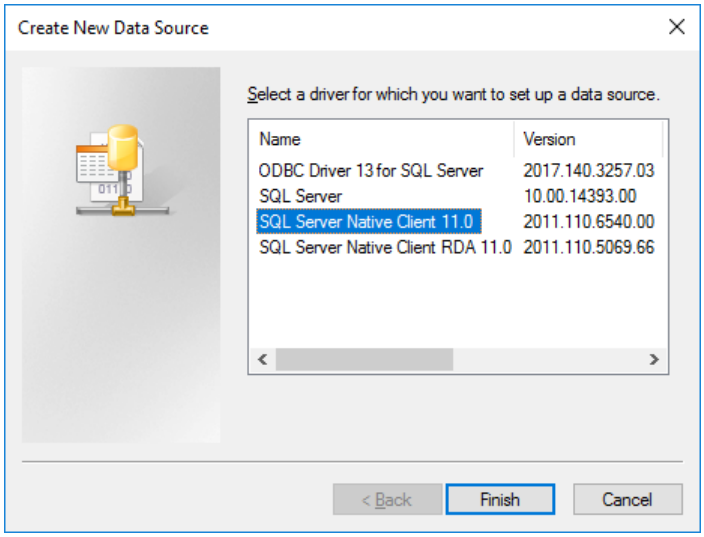
6.4 Create an ODBC DSN

This step describes how to create a system data source name (DSN) for the Infinite i setup. A DSN is a saved configuration that describes a database connection to be used by an ODBC driver.

1. Open **Control Panel** and go to **System and Security** > **Administrative Tools**.
2. Double-click **ODBC Data Sources (64-bit)**.



3. On the **System DSN** tab, click **Add**.
4. In **Create a New Data Source**, select **SQL Server Native Client 11.0** and click **Finish**.

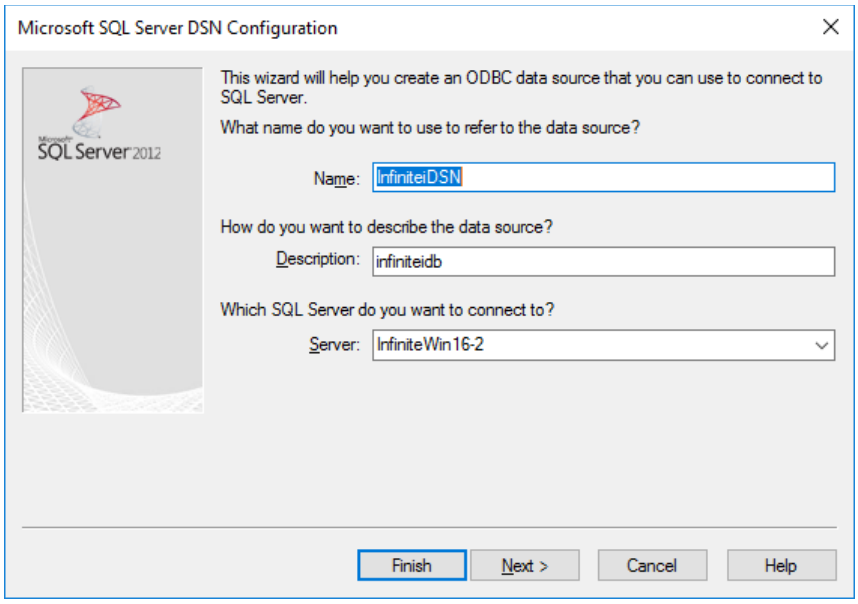


5. In the wizard, use the following values for the DSN of the server associated with the SQL Server engine, then click **Next**:

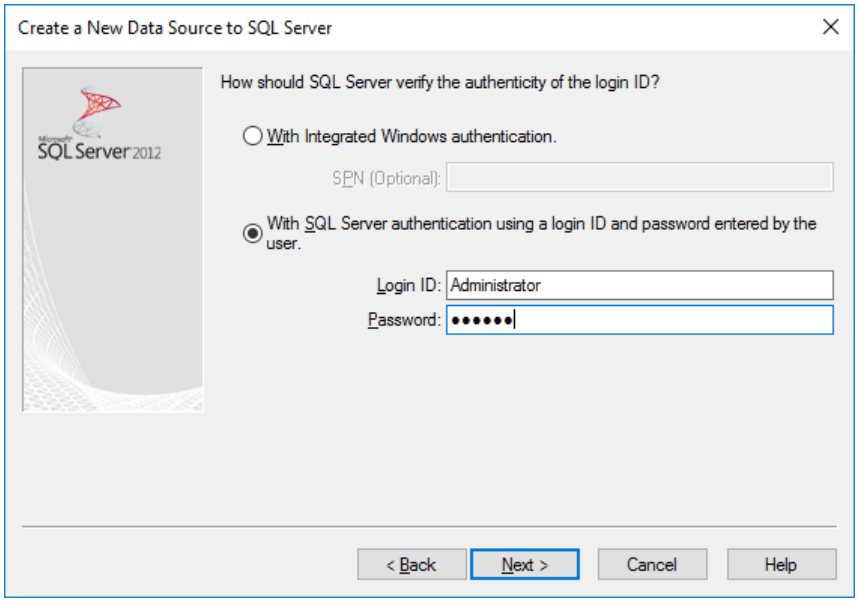
   Name: **InfiniteiDSN**

   Description: **infiniteidb**

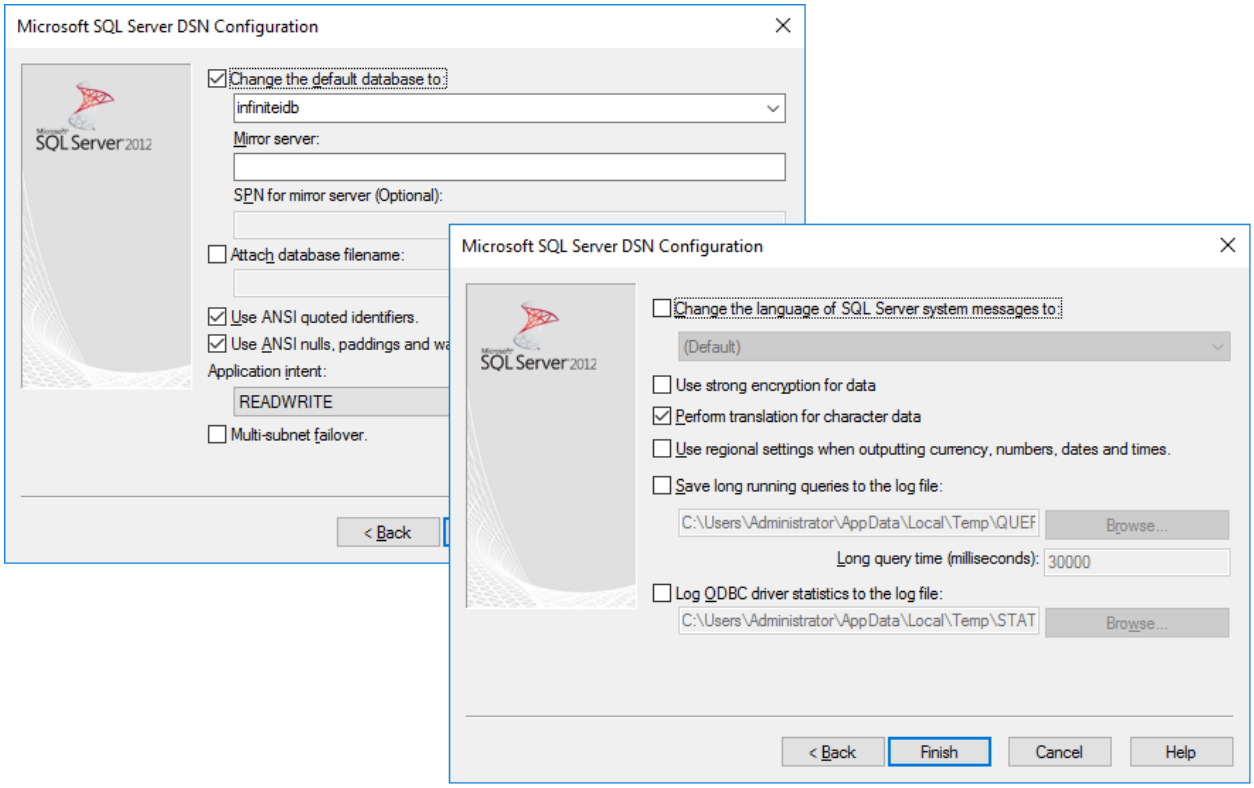
   Server: **InfiniteWin 16-2**



6. For **Login ID** and **Password**, type the credentials of the Administrator user for SQL Server authentication, then click **Next**.

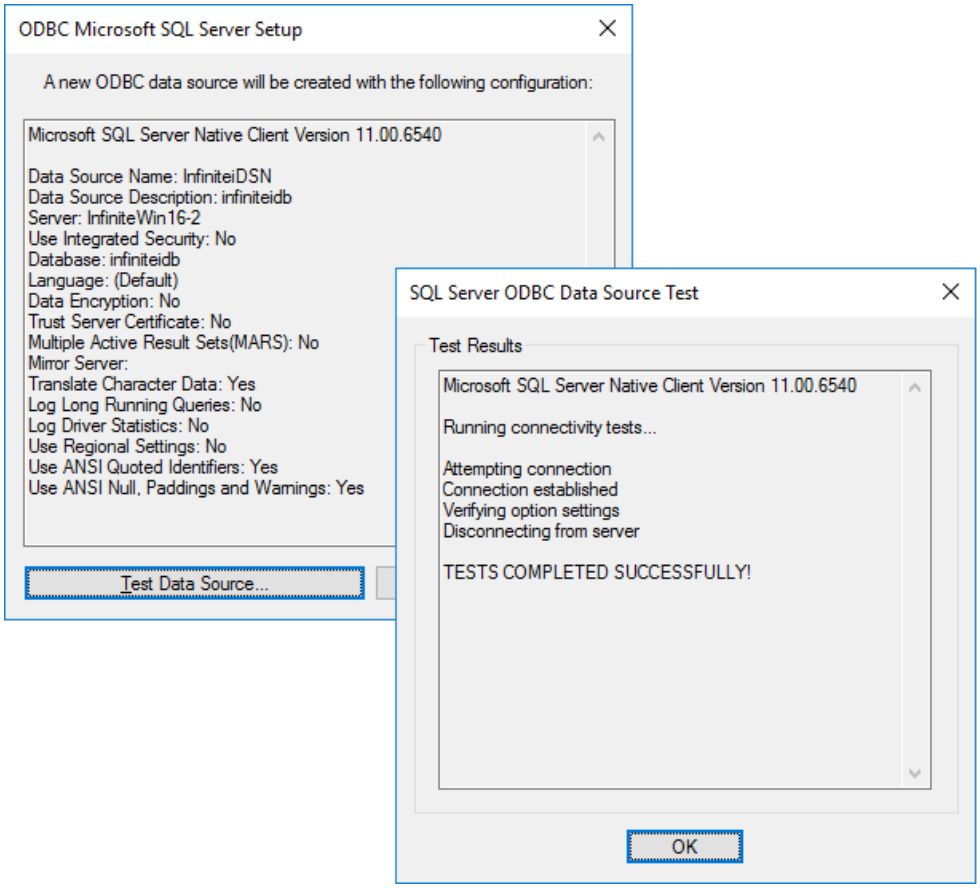


7. Check **Change the default database to** and type **infiniteidb** as the default database. Click **Next**, then click **Next** again.



8. Click **Finish**.

9. Click **Test Data Source**. If the connection works, a message appears as shown.



10. To finish, click **OK** to return to **New Database**, and then click **Finish**.

6.5 *Optional.* SQL Server redundancy and high availability

When you create a virtual machine, you have a choice in how to increase SQL Server redundancy and availability. You can:

- Place the VMs in the same availability set. The VMs within an availability set do not share resources and are not updated at the same time. When you place VMs in the same availability set, they get protection from datacenter outages caused by equipment failure or from updates.

- Place the VMs in different Availability Zones. Zones protect against the failure of an entire datacenter. Each Zone represents a set of datacenters within a region, so an outage in one datacenter cannot take all of your VMs offline.

VMs can be in one or the other, not both. See also Always-On availability group on SQL Server on Azure VMs in the SQL Server documentation.

7 *Optional*. Install Infinite Cloud

If you want to modernize the user interface, you can install Infinite Cloud, a sophisticated recognition engine that reads System I green screens and dynamically renders them in point-and-click graphical screens. The new screens are represented as XML-usable objects that J2EE or .NET applications can execute in a web browser. Infinite Cloud is written in Eclipse and can use any Eclipse-based plug-in.

- For more information, see the Infinite Cloud website.
- For complete installation steps, refer to *Infinite Cloud for Windows Documentation & Installation Instructions*.