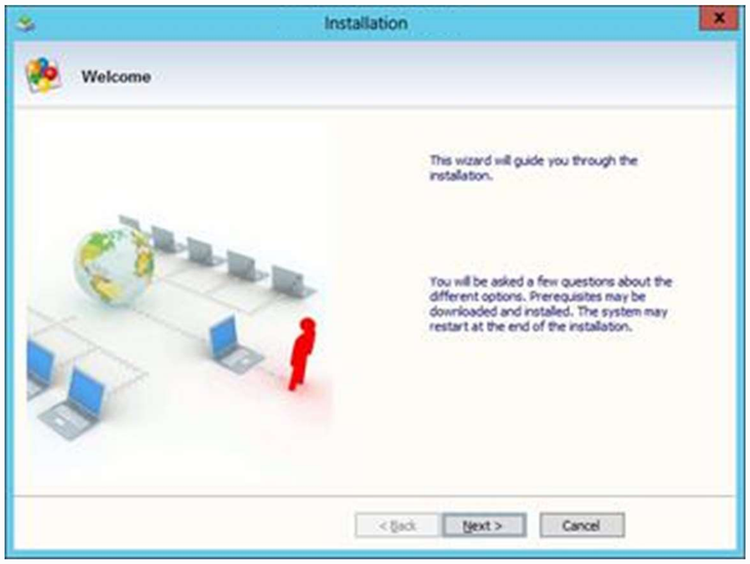
Setup the Second Node (Worker Node #2)

There are two options to setup the second node.

Option 1 – GUI Setup

The first way is to repeat the exact same steps for setting up the first worker node. If it is possible to sit in front of the console or the Remote Desktop Console and click through the installation wizard, this is very straight forward to install the second worker nodes.



This wizard will guide you through the installation.

You will be asked a few questions about the different options. Prerequisites may be downloaded and installed. The system may restart at the end of the installation.

Note

If GUI installer is all you need, you can skip the Options 2 section. Most of the small deployment (< 5000 users) can use the GUI installer because it is fast and simple. However, if you need to setup many worker nodes for big deployment, you may prefer command scripts as shown in Option 2 below.

Option 2 – Command Line Script Setup

There is also a second option to clone the first node to the second node. The second option is completely scriptable. In case you need to prepare 10+ worker nodes, the second option may be easier compared to sitting in front of the Graphic User Interface and clicking through the UI keys and buttons.

To explain scripting, we will break up the script into steps. Once you understand what each step does, you can put all the command lines together into a single script file.

For scripting we use PowerShell, so all the commands below will be executed inside PowerShell.

13.1 – Preparation on the Worker Node #1

We will clone the program files and the registry entry. This step will export the registry and save them as registry files first. Also, we will save the SSL certificate here as well. This is a one-time preparation step.

(The bold text with character shading indicate script commands to run.)

On worker node #1, first we will make a directory to store the registry file.

```
mkdir c:depot
```

After that, we will change directory into the folder.

```
cd c:depot
```

We will export the 64-bit of the registry key.

```
reg export HKEY_LOCAL_MACHINESOFTWAREDatabox Databox64.reg /reg:64
```

```
The operation completed successfully.
```

We will export the 32-bit of the registry key.

```
reg export HKEY_LOCAL_MACHINESOFTWAREDatabox Databox32.reg /reg:32
```

The operation completed successfully.

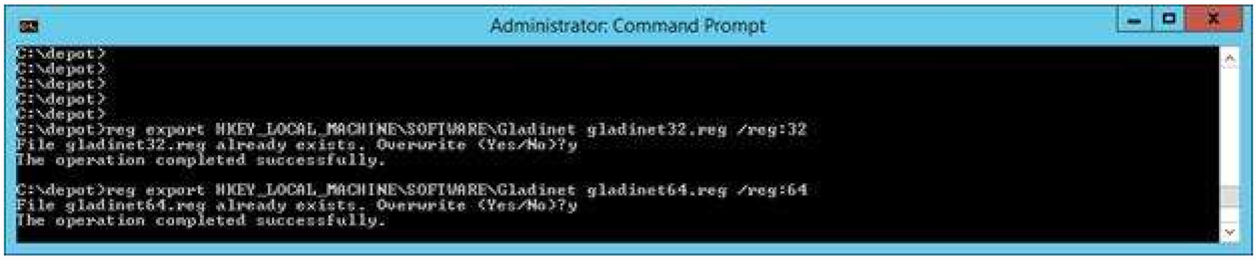
Now, we will verify the two registry files exported into our depot directory.

C:depot>dir

Directory of C:depot

07/19/2014 12:20 AM 762 Databox32.reg

07/19/2014 12:20 AM 546 Databox64.reg



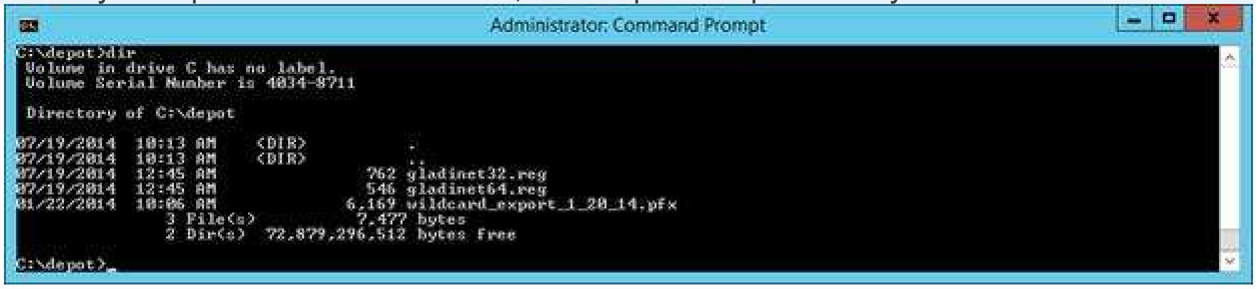
```
C:\depot>
C:\depot>
C:\depot>
C:\depot>
C:\depot>
C:\depot>reg export HKEY_LOCAL_MACHINE\SOFTWARE\Gladinet gladinet32.reg /reg:32
File gladinet32.reg already exists. Overwrite (Yes/No)?y
The operation completed successfully.

C:\depot>reg export HKEY_LOCAL_MACHINE\SOFTWARE\Gladinet gladinet64.reg /reg:64
File gladinet64.reg already exists. Overwrite (Yes/No)?y
The operation completed successfully.
```

Now copy the SSL certificate file into this folder as well

After copying in the SSL certificate in the depot directory, the directory structure will look like this:

Note

When you export the SSL certificate, also import the private key.



```
C:\depot>dir
 Volume in drive C has no label.
 Volume Serial Number is 4034-8711

 Directory of C:\depot

07/19/2014  10:13 AM    <DIR>          .
07/19/2014  10:13 AM    <DIR>          ..
07/19/2014  12:45 AM               762 gladinet32.reg
07/19/2014  12:45 AM               546 gladinet64.reg
01/22/2014  10:06 AM             6,169 wildcard_export_1_20_14.pfx
               3 File(s)          7,477 bytes
               2 Dir(s)  72,879,296,512 bytes free

C:\depot>_
```

We will also need to get the SSL certificate's Thumbprint for later use by using a PowerShell Command. The thumbprint string will be used in script to import SSL certificate to IIS automatically. For example, the thumbprint below will be used later.

```
E88BC131D403B9EF5AA44DC520D3BABDB93E5886

PS C:> Get-ChildItem -path cert:LocalMachineMy

Directory: Microsoft.PowerShell.SecurityCertificate::LocalMachineMy

Thumbprint Subject

E88BC131D403B9EF5AA44DC520D3BABDB93E5886 CN=*.Databox.com,
OU=COMODO SSL Wildcard, OU=Domain Control Validated

E50A764700D92D717E8CE21DCF99CAB5F5280198 CN=localhost
```

Clone Program Files

Now, on worker node #2, we will copy c:Program Files (x86)Databox from worker node #1 to worker node #2.

We will go to the worker node2's administrative command prompt with PowerShell.

First change directory to the c:Program Files (x86)directory.

cd "C:Program Files (x86)"

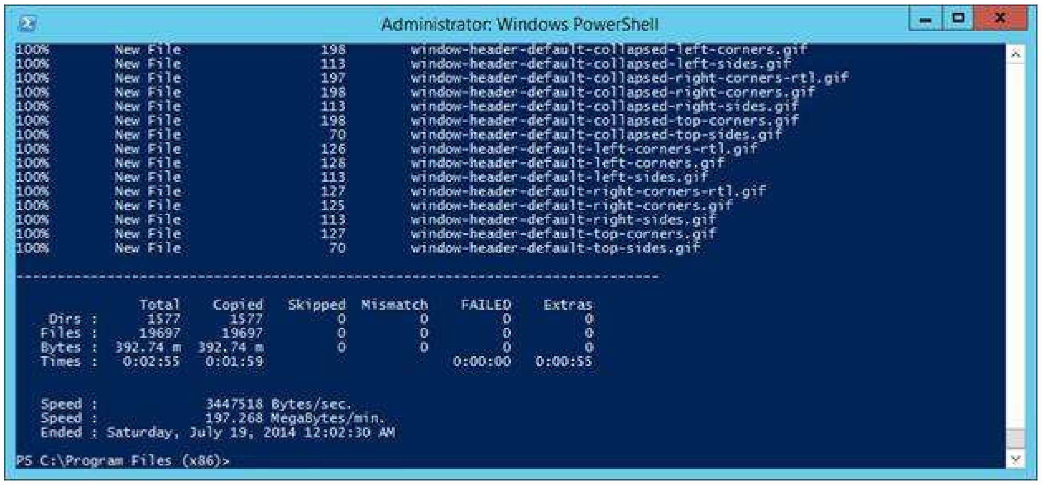
Second, we will map a network drive to the worker node #1.

net use z: \node1.tsys.Databox.comc$

Third, we will copy over the "Databox" folder.

robocopy "z:Program Files (x86)Databox" "Databox" /s

When it is done, the PowerShell Window looks like this:



Administrator: Windows PowerShell

```
100%        New File                198        window-header-default-collapsed-left-corners.gif
100%        New File                113        window-header-default-collapsed-left-sides.gif
100%        New File                197        window-header-default-collapsed-right-corners-rtl.gif
100%        New File                198        window-header-default-collapsed-right-corners.gif
100%        New File                113        window-header-default-collapsed-right-sides.gif
100%        New File                198        window-header-default-collapsed-top-corners.gif
100%        New File                 70        window-header-default-collapsed-top-sides.gif
100%        New File                126        window-header-default-left-corners-rtl.gif
100%        New File                128        window-header-default-left-corners.gif
100%        New File                113        window-header-default-left-sides.gif
100%        New File                127        window-header-default-right-corners-rtl.gif
100%        New File                125        window-header-default-right-corners.gif
100%        New File                113        window-header-default-right-sides.gif
100%        New File                127        window-header-default-top-corners.gif
100%        New File                 70        window-header-default-top-sides.gif

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

               Total    Copied   Skipped  Mismatch    FAILED    Extras
    Dirs :      1577      1577         0         0         0         0
   Files :     19697     19697         0         0         0         0
   Bytes :  392.74 m  392.74 m         0         0         0         0
   Times :   0:02:55   0:01:59                       0:00:00   0:00:55


   Speed :             3447518 Bytes/sec.
   Speed :             197.268 MegaBytes/min.
   Ended : Saturday, July 19, 2014 12:02:30 AM

PS C:\Program Files (x86)>
```

Clone the Registry

Now we will copy over the registry file from worker node #1.

On worker node #2, first we will create a folder to receive the files.

C:> **mkdir c:depot>dir**

After that, we will change directory into the folder.

C:> **cd c:depot**

Now, we will copy over the registry files (the z: drive is already mapped to worker node #1's c$ share)

C:depot>copy z:depot*.*

z:depotDatabox32.reg

z:depotDatabox64.reg

2 file(s) copied.

Now, import the registry files. 32-bit first

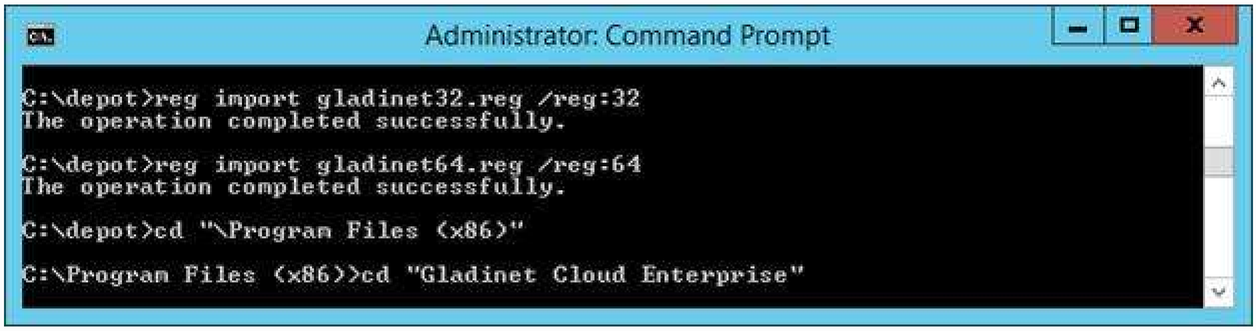
C:depot> **reg import Databox32.reg /reg:32**

The operation completed successfully.

Now import the registry file. 64-bit second

C:depot> **reg import Databox64.reg /reg:64**

The operation completed successfully.



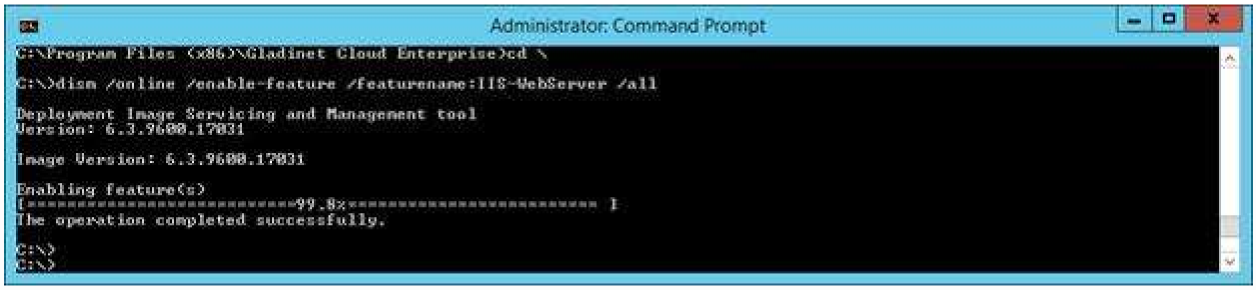
```
C:\depot>reg import gladinet32.reg /reg:32
The operation completed successfully.

C:\depot>reg import gladinet64.reg /reg:64
The operation completed successfully.

C:\depot>cd "\Program Files (x86)"

C:\Program Files (x86)>cd "Gladinet Cloud Enterprise"
```

Command Line Setup for the IIS ASP.NET and WCF applications.

Next, we need to make sure IIS WebServer role is enabled, WCF is activated and so on.

Enable IIS-WebServer first

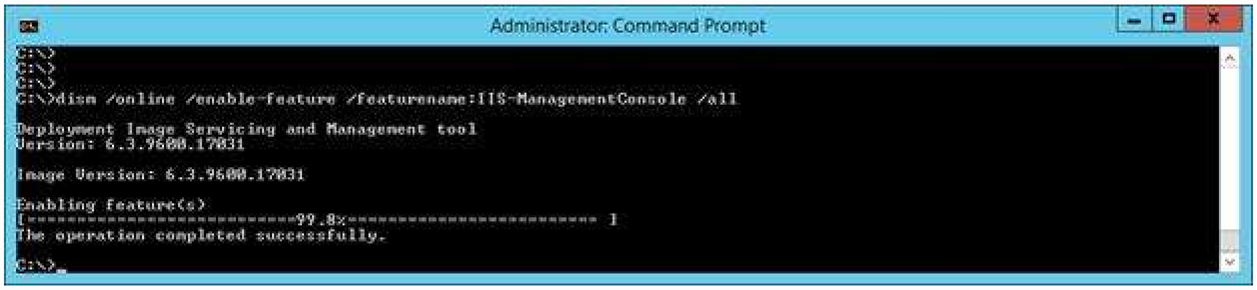
dism /online /enable-feature /featurename:IIS-WebServer /all



```
C:\Program Files (x86)\Gladinet Cloud Enterprise>cd \

C:\>dism /online /enable-feature /featurename:IIS-WebServer /all

Deployment Image Servicing and Management tool
Version: 6.3.9600.17031

Image Version: 6.3.9600.17031

Enabling feature(s)
[===========================99.8%========================= ]
The operation completed successfully.

C:\>
C:\>
```

Enable IIS Management Console

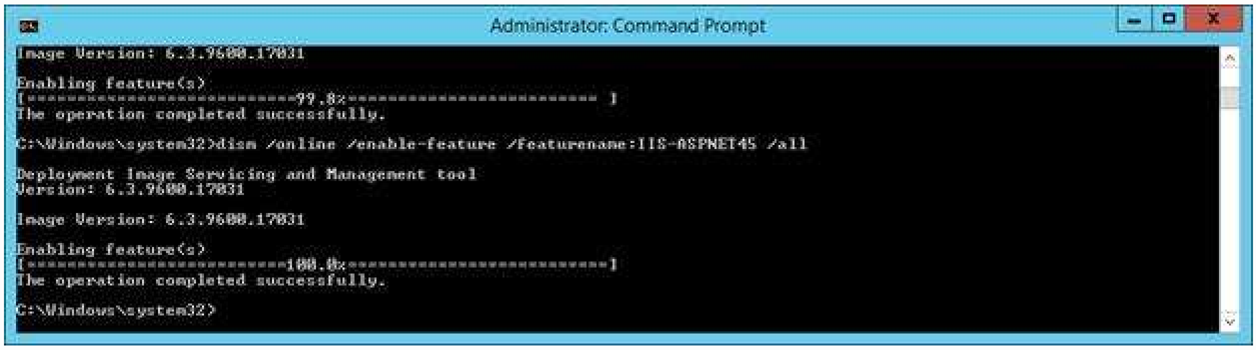
dism /online /enable-feature /featurename:IIS-ManagementConsole /all



```
C:\>
C:\>
C:\>
C:\>dism /online /enable-feature /featurename:IIS-ManagementConsole /all

Deployment Image Servicing and Management tool
Version: 6.3.9600.17031

Image Version: 6.3.9600.17031

Enabling feature(s)
[===========================99.8%========================= ]
The operation completed successfully.

C:\>_
```

Enable ASP.NET45

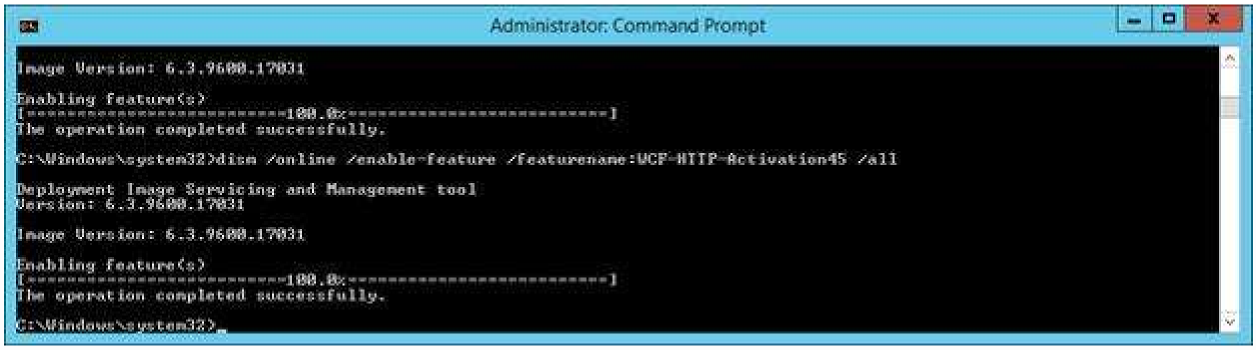
dism /online /enable-feature /featurename:IIS-ASPNET45 /all



```
Image Version: 6.3.9600.17031

Enabling feature(s)
[===========================99.8%========================= ]
The operation completed successfully.

C:\Windows\system32>dism /online /enable-feature /featurename:IIS-ASPNET45 /all

Deployment Image Servicing and Management tool
Version: 6.3.9600.17031

Image Version: 6.3.9600.17031

Enabling feature(s)
[==========================100.0%==========================]
The operation completed successfully.

C:\Windows\system32>
```

Enable WCF

dism /online /enable-feature /featurename:WCF-HTTP-Activation45 /all

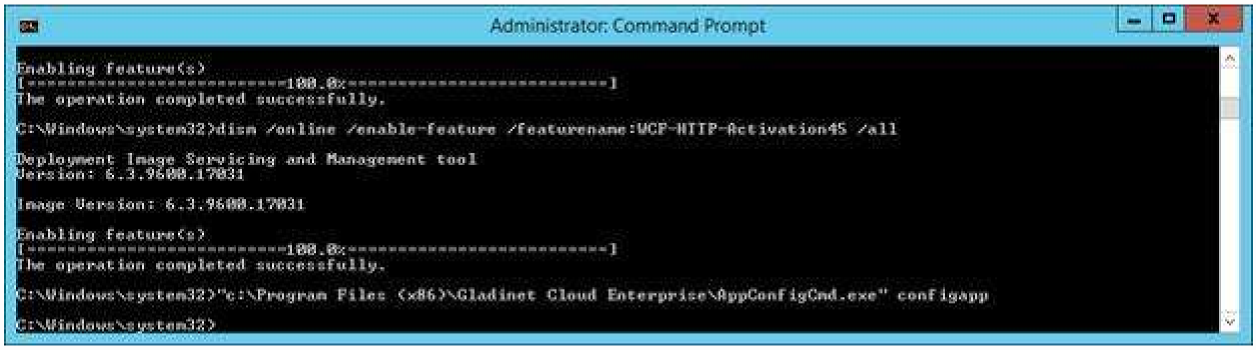


```
Image Version: 6.3.9600.17031

Enabling feature(s)
[==========================100.0%==========================]
The operation completed successfully.

C:\Windows\system32>dism /online /enable-feature /featurename:WCF-HTTP-Activation45 /all

Deployment Image Servicing and Management tool
Version: 6.3.9600.17031

Image Version: 6.3.9600.17031

Enabling feature(s)
[==========================100.0%==========================]
The operation completed successfully.

C:\Windows\system32>_
```

Setup Web Applications

Run the following command to set it up.

& “C:Program Files (x86)DataboxAppConfigCmd.exe” configapp



```
Enabling feature(s)
[==========================100.0%==========================]
The operation completed successfully.

C:\Windows\system32>dism /online /enable-feature /featurename:WCF-HTTP-Activation45 /all

Deployment Image Servicing and Management tool
Version: 6.3.9600.17031

Image Version: 6.3.9600.17031

Enabling feature(s)
[==========================100.0%==========================]
The operation completed successfully.

C:\Windows\system32>"c:\Program Files (x86)\Gladinet Cloud Enterprise\AppConfigCmd.exe" configapp

C:\Windows\system32>
```

Import SSL Certificate

First we copy over the SSL certificate from node1

cd c:depot

***copy /y z:depot**.pfx**

Then we import the certificate, replace the <password> with the real password of the PFX file.

certutil -p <password> -importPFX wildcard_export_1_20_14.pfx

After that, we will use PowerShell command to set it up in IIS. First we need to import the WebAdministration module.

PS C:> **Import-Module WebAdministration**

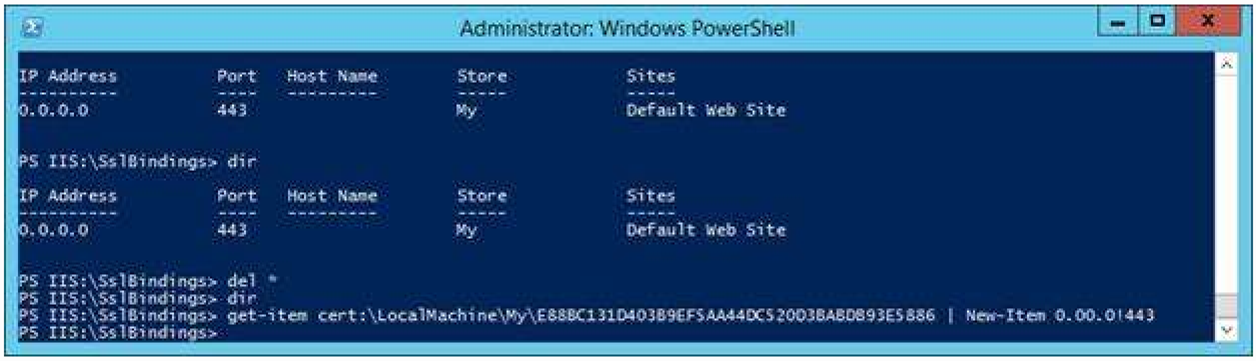
After that, we will go to the SslBindings

PS C:> **cd IIS:SslBindings**

PS IIS:SslBindings> ***del ***

Now, we use the certificate and apply it to IIS. In the example below, the E88…886 thumbprint part matches the thumbprint string we had earlier on worker node #1.

PS IIS:SslBindings> get-item cert:LocalMachineMyE88BC131D403B9EF5AA44DC520D3BABDB93E5886 | New-Item 0.00.0!443

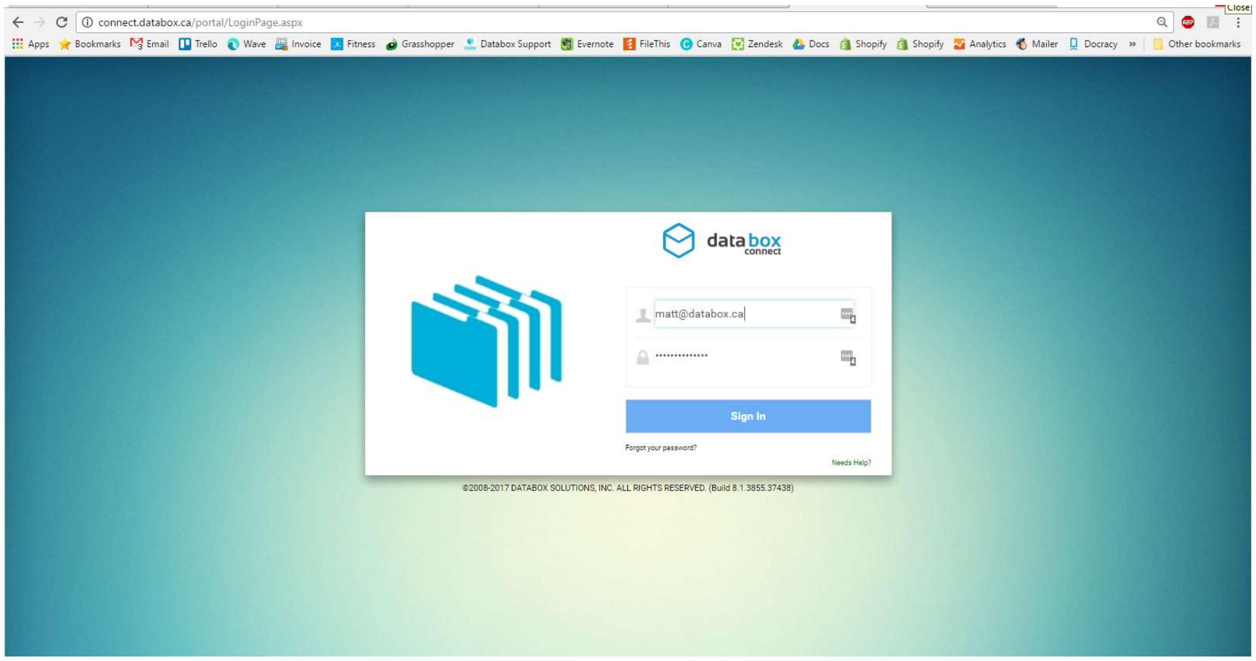

Administrator: Windows PowerShell

```
IP Address         Port   Host Name        Store            Sites
----------         ----   ---------        -----            -----
0.0.0.0            443                     My               Default Web Site


PS IIS:\SslBindings> dir

IP Address         Port   Host Name        Store            Sites
----------         ----   ---------        -----            -----
0.0.0.0            443                     My               Default Web Site


PS IIS:\SslBindings> del *
PS IIS:\SslBindings> dir
PS IIS:\SslBindings> get-item cert:\LocalMachine\My\E88BC131D403B9EF5AA44DC520D3BABDB93E5886 | New-Item 0.00.0!443
PS IIS:\SslBindings>
```

Now, Worker Node #2 will be ready after a reboot. Reboot now.

Sanity Check Worker Node #2

After worker node #2 is up, you can use the http://localhost on worker node to check that the worker node #2 is running.



matt@databox.ca

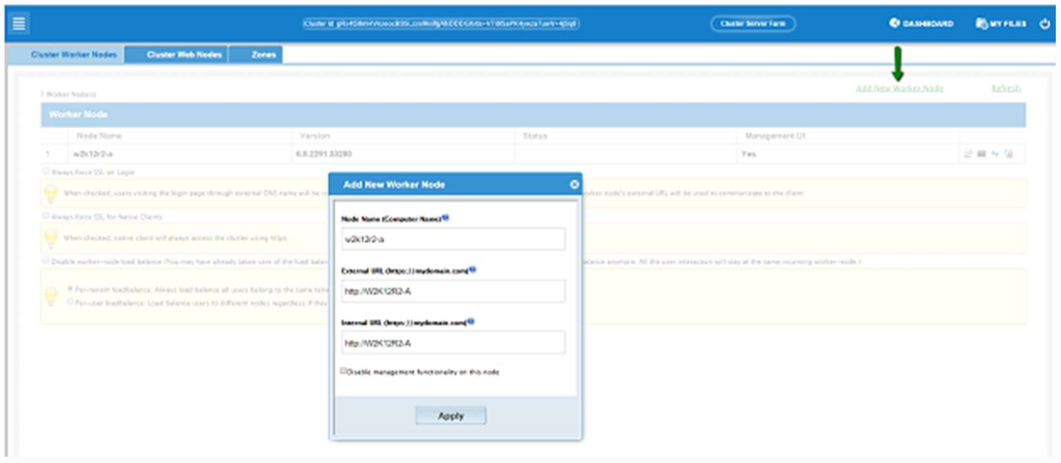
©2008-2017 DATABOX SOLUTIONS, INC. ALL RIGHTS RESERVED. (Build 8.1.3855.37438)

Add the worker node 2 to the cluster

While we are doing sanity check for worker node #2, we can also register it to the cluster by log in as the Master Administrator. Go to the “Cluster Worker Nodes” page and do “Add New Worker Node”.

The Node Name will match the Host Name. The external URL will match the virtual IP or the DNS name for the load balanced DNS name. The internal URL will match the internal IP address or internal DNS name.

You can also repeat the same validation methods for the setup as we have done for worker node #1.



Add New Worker Node

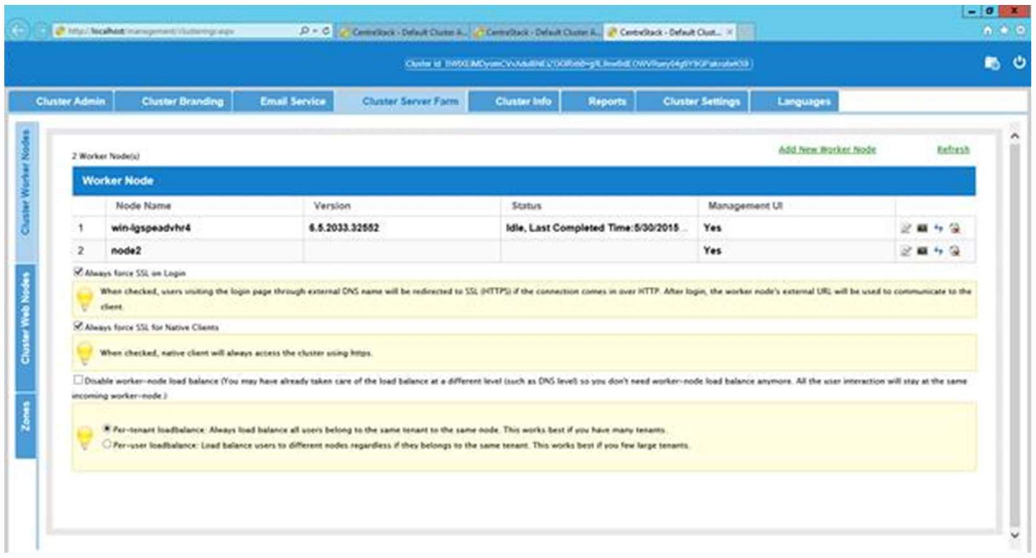
Node Name (Computer Name): w2k12r2-a

External URL (https://mydomain.com): http://W2K12R2-A

Internal URL (https://mydomain.com): http://W2K12R2-A

☐ Disable management functionality on this node

After it is added.



2 Worker Node(s)

Worker Node

	Node Name	Version	Status	Management UI
1	win-lgspeadvhr4	6.5.2033.32552	Idle, Last Completed Time:5/30/2015 ...	Yes
2	node2			Yes

☑ Always force SSL on Login

When checked, users visiting the login page through external DNS name will be redirected to SSL (HTTPS) if the connection comes in over HTTP. After login, the worker node's external URL will be used to communicate to the client.

☑ Always force SSL for Native Clients

When checked, native client will always access the cluster using https.

☐ Disable worker-node load balance (You may have already taken care of the load balance at a different level (such as DNS level) so you don't need worker-node load balance anymore. All the user interaction will stay at the same incoming worker-node.)

[✓ Per-tenant loadbalance: Always load balance all users belong to the same tenant to the same node. This works best if you have many tenants.]

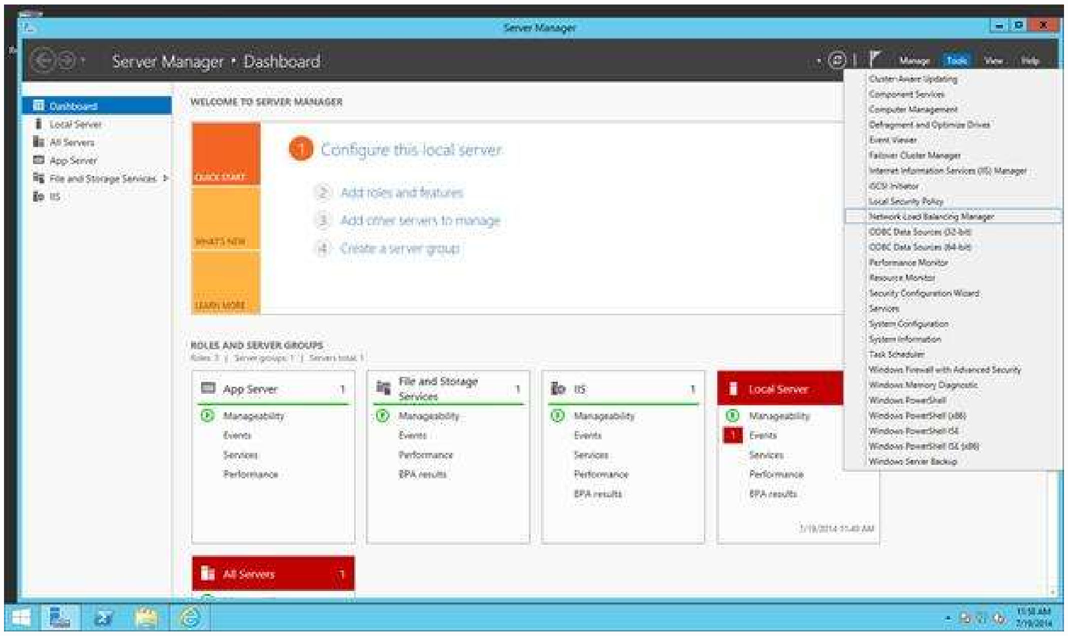
Per-user loadbalance: Load balance users to different nodes regardless if they belongs to the same tenant. This works best if you few large tenants.

Add the worker node #2 to the Load Balancer.

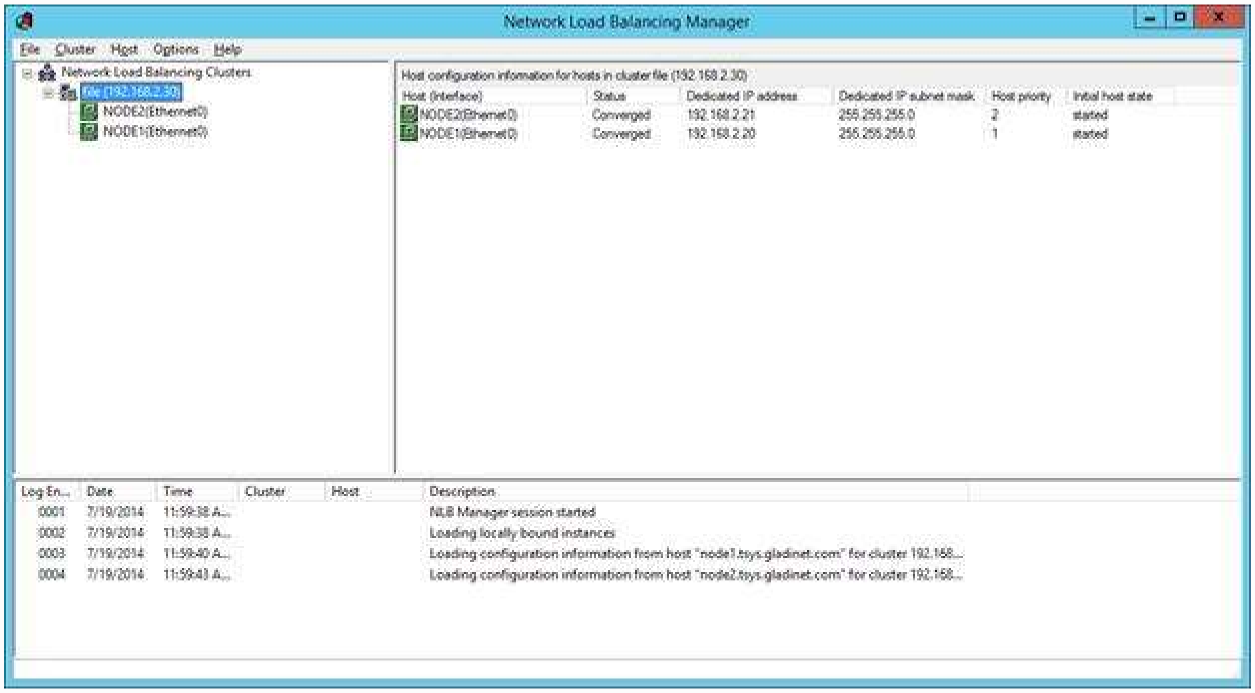
This step depends on the kind of load balancer you have.

Skip this if you are not using Windows 2012 NLB.

In the case of Windows 2012 R2 Network Load Balancer, You will add it in the “Network Load Balancing Management”.



After adding the Node2, it looks like this.



Host configuration information for hosts in cluster file (192.168.2.30)

Host (Interface)	Status	Dedicated IP address	Dedicated IP subnet mask	Host priority	Initial host state
NODE2(Ethernet0)	Converged	192.168.2.21	255.255.255.0	2	started
NODE1(Ethernet0)	Converged	192.168.2.20	255.255.255.0	1	started

```
Log En...  Date       Time        Cluster  Host  Description
0001       7/19/2014  11:59:38 A...              NLB Manager session started
0002       7/19/2014  11:59:38 A...              Loading locally bound instances
0003       7/19/2014  11:59:40 A...              Loading configuration information from host "node1.tsys.gladinet.com" for cluster 192.168...
0004       7/19/2014  11:59:43 A...              Loading configuration information from host "node2.tsys.gladinet.com" for cluster 192.168...
```

Pre-Compile ASP.NET Pages

It is not required to pre-compile ASP.NET pages because ASP.NET does just-in-time compile so the first user hitting a specific ASP.NET page will get the page compiled.

However, for best performance, we can precompile the ASP.NET pages.

PS C:WindowsMicrosoft.NETframework64v4.0.30319> **.aspnet_compiler.exe /v portal | Out-Null**

PS C:WindowsMicrosoft.NETframework64v4.0.30319> **.aspnet_compiler.exe /v management | Out-Null**