# Installing N-central in Microsoft® Azure

This document describes how to create an Azure-based N-central installation starting with an N-central Azure Virtual Machine image file. The process includes the following steps:

- upload an image to Azure

- instantiate an Azure VM based on this image

## Prerequisites

Ensure that you have:

- an Azure subscription

- a Windows-based PC loaded with:

    o PuTTY (or any Secure Shell client)

    o PowerShell with Azure extensions (if this is not done, there are some instructions included for setting this up)

**Important Notes**:

- N-central is certified to run on Azure. Performance and scalability testing will be complete by the end of 2015. Until that time, SolarWinds N-able recommends deployment on limited sized environments.

- Ensure they you have properly configured N-central before you begin. Refer to the N-central 10 System Requirements documented in the 10.0 SP1 release notes.

- SolarWinds N-able recommends that you plan your deployment with N-central Support. A license may need to be applied before you are able to log in to N-central.

- You must have basic knowledge about how to install and configure N-central. Refer to the *N-central Installation Guide*, version 10.0 SP1.

# Creating an Azure-based N-central Installation

**Note**: This article describes how to install an N-central image for deployment in Azure. The article assumes general understanding and knowledge of Azure including network and hardware troubleshooting.
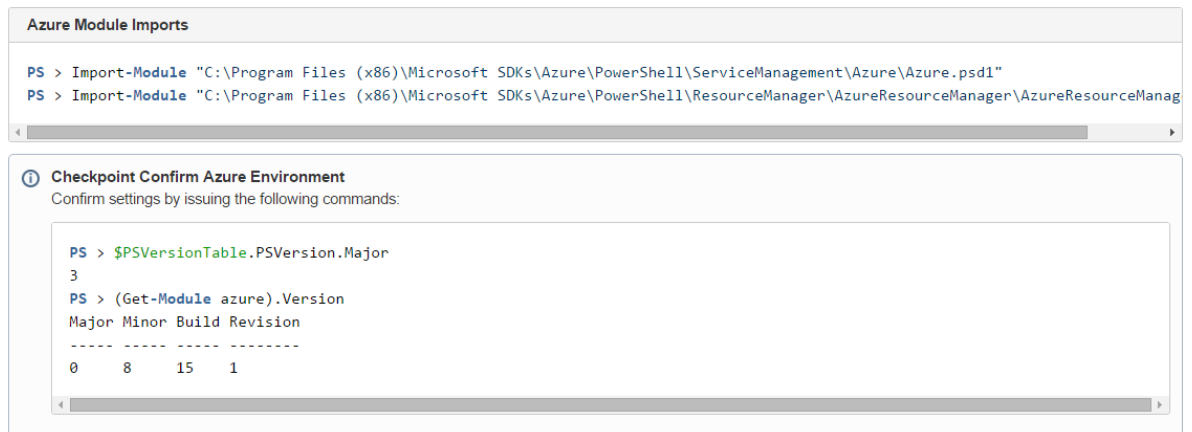
1. Use the VM install file image in NRC: **Software Downloads → N-central 10 SP1**.

2. Set up the PowerShell and Azure environment. PowerShell commands are used to perform the Azure steps required to convert and upload the virtual machine image file and create an Azure cloud VM based on the image. Each step below requires PowerShell with Azure extensions installed on the working PC. Background for the environment set up can be found here: https://azure.microsoft.com/en-us/documentation/articles/powershell-install-configure/

   **Notes:**
   - You must have Administrator privileges for the Azure environment.
   - Another method to launch PowerShell can be done via Start Menu > Right Click Azure PowerShell tile > Run as Administrator.
   - You must download the image file N-central-x.x.x.vhd.gz. Extract this file and you will use the .vhd file in your Azure environment.

   a. For the PowerShell console, the following imports add the required PowerShell Azure modules previously installed (see above).

   ```
   PS > Import-Module "C:\Program Files (x86)\Microsoft
   SDKs\Azure\PowerShell\ServiceManagement\Azure\Azure.psd1"

   PS > Import-Module "C:\Program Files (x86)\Microsoft
   SDKs\Azure\PowerShell\ResourceManager\AzureResourceManager\AzureResourceManager.psd1"
   ```

   **Azure Module Imports**
   ```
   PS > Import-Module "C:\Program Files (x86)\Microsoft SDKs\Azure\PowerShell\ServiceManagement\Azure\Azure.psd1"
   PS > Import-Module "C:\Program Files (x86)\Microsoft SDKs\Azure\PowerShell\ResourceManager\AzureResourceManager\AzureResourceManag
   ```

   ⓘ **Checkpoint Confirm Azure Environment**
   Confirm settings by issuing the following commands:
   ```
   PS > $PSVersionTable.PSVersion.Major
   3
   PS > (Get-Module azure).Version
   Major Minor Build Revision
   ----- ----- ----- --------
   0     8     15    1
   ```

   b. Connect the console to the Azure subscription.

   **Note:** Alternative Method → Download the encrypted credentials so you don't have to input credentials as a variable every time you launch PowerShell (however this may pose a security risk).

   - Get-AzurePublishSettingsFile. In the Azure login you can download the credential file.

- Import-AzurePublishSettingsFile "C:\file.publishsettings" and use the "Get-AzureSubscription".

```
Azure Subscription

PS > Add-AzureAccount  #Pops an Azure credentials window
PS > $sub = Get-AzureSubscription -ExtendedDetails
PS > Select-AzureSubscription -SubscriptionId $sub.SubscriptionId -Current
```

c. There must be a network set up on the Azure account. If there isn't, refer to the steps in the following Azure document:
https://azure.microsoft.com/en-us/documentation/articles/virtual-networks-create-vnet/

- The name length is between 8 and 24 characters long.
- The name must be unique and cannot be renamed.

```
Set up Storage

PS > $location = "central US"    # use Azure region desired
PS > $StoreAcctName = "ncentralstorage"
PS > $StoreAcct = New-AzureStorageAccount -StorageAccountName $StoreAcctName -Location $location -Type "Standard_LRS"
PS > $StoreKey = (Get-AzureStorageKey -StorageAccountName $StoreAcctName).Primary
PS > $StoreContext = New-AzureStorageContext -StorageAccountName $StoreAcctName -StorageAccountKey $StoreKey
PS > $StoreContainerName = "ncentralstoragecon"
PS > $StoreContainer = New-AzureStorageContainer -Name $StoreContainerName -Permission Container -Context $StoreContext
```

```
ⓘ Checkpoint Storage Endpoint
  Issue the following and note the Blob End Point:

    PS > Write-Output $StoreContainer
    Blob End Point: https://ncentralstorage.blob.core.windows.net/
```

The Storage type must support page blobs (VM storage) Premium Storage:

- Local Redundant
- Geo-Redundant
- Premium Locally Redundant
  (Only available on the DS-series VMs)

**Note:** The previous step provides you with a storage container. To change the storage container, use the command shown in the example below:

```
ActiveDirectoryUserId      : hdevadoss@swiengineering.onmicrosoft.com
Account                    : Microsoft.Azure.Common.Authentication.Models.AzureAccount
Certificate                :
AccountAdminLiveEmailId     : dev-grp-labmanagement@solarwinds.com
CurrentCoreCount           : 24
CurrentHostedServices      : 13
CurrentDnsServers          : 0
CurrentLocalNetworkSites   : 0
CurrentVirtualNetworkSites : 4
CurrentStorageAccounts     : 15
MaxCoreCount               : 10000
MaxDnsServers              : 20
MaxHostedServices          : 200
MaxLocalNetworkSites       : 20
MaxVirtualNetworkSites     : 50
MaxStorageAccounts         : 100
ServiceAdminLiveEmailId    : dev-grp-labmanagement@solarwinds.com
SubscriptionRealName       : RDOps Azure Cloud
SubscriptionStatus         : Active
OperationDescription       :
OperationId                :
OperationStatus            :
ServiceEndpoint            : https://management.core.windows.net/
ResourceManagerEndpoint    : https://management.azure.com/
GalleryEndpoint            :
SubscriptionId             : 47f8e67f-7f14-4b12-9316-b8916b54c9df
SubscriptionName           : RDOps Azure Cloud
Environment                :
SupportedModes             : AzureServiceManagement,AzureResourceManager
DefaultAccount             :
Accounts                   : {hdevadoss@swiengineering.onmicrosoft.com}
IsDefault                  : True
IsCurrent                  : True
CurrentStorageAccountName  : harincentralstorage5
TenantId                   :
```

```
PS C:\> Set-AzureSubscription -CurrentStorageAccountName $StoreAcctName -SubscriptionName "RDOps Azure Cloud"
```

```
ActiveDirectoryUserId      : hdevadoss@swiengineering.onmicrosoft.com
Account                    : Microsoft.Azure.Common.Authentication.Models.AzureAccount
Certificate                :
AccountAdminLiveEmailId     : dev-grp-labmanagement@solarwinds.com
CurrentCoreCount           : 24
CurrentHostedServices      : 13
CurrentDnsServers          : 0
CurrentLocalNetworkSites   : 0
CurrentVirtualNetworkSites : 4
CurrentStorageAccounts     : 16
MaxCoreCount               : 10000
MaxDnsServers              : 20
MaxHostedServices          : 200
MaxLocalNetworkSites       : 20
MaxVirtualNetworkSites     : 50
MaxStorageAccounts         : 100
ServiceAdminLiveEmailId    : dev-grp-labmanagement@solarwinds.com
SubscriptionRealName       : RDOps Azure Cloud
SubscriptionStatus         : Active
OperationDescription       :
OperationId                :
OperationStatus            :
ServiceEndpoint            : https://management.core.windows.net/
ResourceManagerEndpoint    : https://management.azure.com/
GalleryEndpoint            :
SubscriptionId             : 47f8e67f-7f14-4b12-9316-b8916b54c9df
SubscriptionName           : RDOps Azure Cloud
Environment                :
SupportedModes             : AzureServiceManagement,AzureResourceManager
DefaultAccount             :
Accounts                   : {hdevadoss@swiengineering.onmicrosoft.com}
IsDefault                  : True
IsCurrent                  : True
CurrentStorageAccountName  : harincentralstorage6
TenantId                   :
```

Afterwards, re-run this command:

**PS > $sub = Get-AzureSubscription -ExtendedDetails**

3. Upload the VHD file to Azure and register it as an OS disk. This allows it to be used for VM creation in Azure.

    **Note**: The upload URL is comprised of the ***Blob End Point*** concatenated with the folder name.

    **Helpful Tip**: GUI apps will cause uploads to upload the entire 200GB image block-by-block. Microsoft recommends using PowerShell to upload data blocks without empty spaces.

```
VHD Upload to Azure

PS > $DiskName = "ncdisk"
PS > $UploadURL = "https://ncentralstorage.blob.core.windows.net/uploads/ncentralVM"
PS > Add-AzureVhd -LocalFilePath $vhdPath -Destination $UploadURL -NumberOfUploaderThreads 5
PS > Add-AzureDisk -DiskName $DiskName -MediaLocation $UploadURL -Label $DiskName -OS Linux
```

4. Create the Azure service (unit of deployment) and VM.

```
Azure VM Creation

PS > $ServName = "ncentralserv0"  # the service name forms part of the URL for accessing the VM
PS > $vmName = "ncentralvm0"
PS > New-AzureService -ServiceName $ServName -Location $location
PS > $myVM = New-AzureVMConfig -DiskName $DiskName -Name $vmName -InstanceSize Large
PS > Set-AzureSubnet -SubnetNames $Subnet -VM $myVM
PS > New-AzureVM -ServiceName $ServName -VNetName $Network -VMs $myVM
```

**Note:** In order for the use of $Subnet and $Network variables noted in the example above you will have to define them:

    **Example:**        $network = "Group SWIENG-Dev-CentralUS SWIENG-DevNet-CentralUS"
                            $subnet = "subnet0"

The commands to create the VM can be used as written: (Set-AzureSubnet, New-AzureVM).

**Note:** InstanceSize should be the Azure Instance Size you selected for your region or storage type. For example: "Standard_DS2"**.**

**Azure VM Image Sizes**

| # of devices | Azure instance size  [Virtualized Azure] | Optimal performance for up to … |
|---|---|---|
| 1000 or less | Basic A2 | 1,000 Agents |
| 1,000 – 3,000 | Standard A4 (8 Cores, 14 GB memory) | 3,000 Agents |
| 3,000 – 6,000 | Standard DS4 (8 Cores, 28 GB memory) | 6,000 Agents |

**Checkpoint VM Creation**

Query the status of the new VM and allow a few minutes for the VM to appear in the cloud. Note the *IpAddress* and *DNSName* variables for reference when connecting to this server.

The following is an example of the typical output:

```
PS > Get-AzureVM -ServiceName $ServName -Name $vmName

DeploymentName              : ncentralserv0
Name                        : ncentralvm0
Label                       :
VM                          : Microsoft.WindowsAzure.Commands.ServiceManagement.Model.PersistentVM
InstanceStatus              : ReadyRole
IpAddress                   : 172.16.0.4
InstanceStateDetails        :
PowerState                  : Started
InstanceErrorCode           :
InstanceFaultDomain         : 0
InstanceName                : ncentralvm0
InstanceUpgradeDomain       : 0
InstanceSize                : Large
HostName                    :
AvailabilitySetName         :
DNSName                     : http://ncentralserv0.cloudapp.net/
Status                      : ReadyRole
GuestAgentStatus            : Microsoft.WindowsAzure.Commands.ServiceManagement.Model.GuestAgentStatus
ResourceExtensionStatusList : {}
PublicIPAddress             :
PublicIPName                :
NetworkInterfaces           : {}
VirtualNetworkName          : Group SWIENG-Dev-CentralUS SWIENG-DevNet-CentralUS
ServiceName                 : ncentralserv0
OperationDescription        : Get-AzureVM
OperationId                 : cd77b2c331f3ce81a0dbb6e0e74417e2
OperationStatus             : OK
```

5.  Configure the endpoints for the VM:

```
Azure VM Endpoints

PS > $myVM = Get-AzureVM -ServiceName $ServName -Name $vmName
PS > Add-AzureEndpoint -VM $myVM -Name "SSH" -Protocol "tcp" -PublicPort 22 -LocalPort 22 | Update-AzureVM
PS > Add-AzureEndpoint -VM $myVM -Name "HTTPS" -Protocol "tcp" -PublicPort 443 -LocalPort 443 | Update-AzureVM
PS > Add-AzureEndpoint -VM $myVM -Name "HTTP" -Protocol "tcp" -PublicPort 80 -LocalPort 80 | Update-AzureVM
PS > Add-AzureEndpoint -VM $myVM -Name "N-Able" -Protocol "tcp" -PublicPort 10000 -LocalPort 10000 | Update-AzureVM
```

**Note:**

- Set a reserved IP address for the VM. This reserves the VIP assignment if the VM powers off and de-allocates.
- Set an instance IP. This IP is the IP you want to use publically and to set in N-central. It **does not** use the Azure load balancer and allows full traffic to flow, including ICMP, which is not allowed using the VIP assignment/public DNS:

  Instance level IP:
  Get-AzureVM -ServiceName "fvc-cloud" -Name "fvc-noc" | Set-AzurePublicIP -PublicIPName instanceIp1 | Update-AzureVM

6. After the VM is created in Azure, the Azure portal monitors the VM state, as shown below.
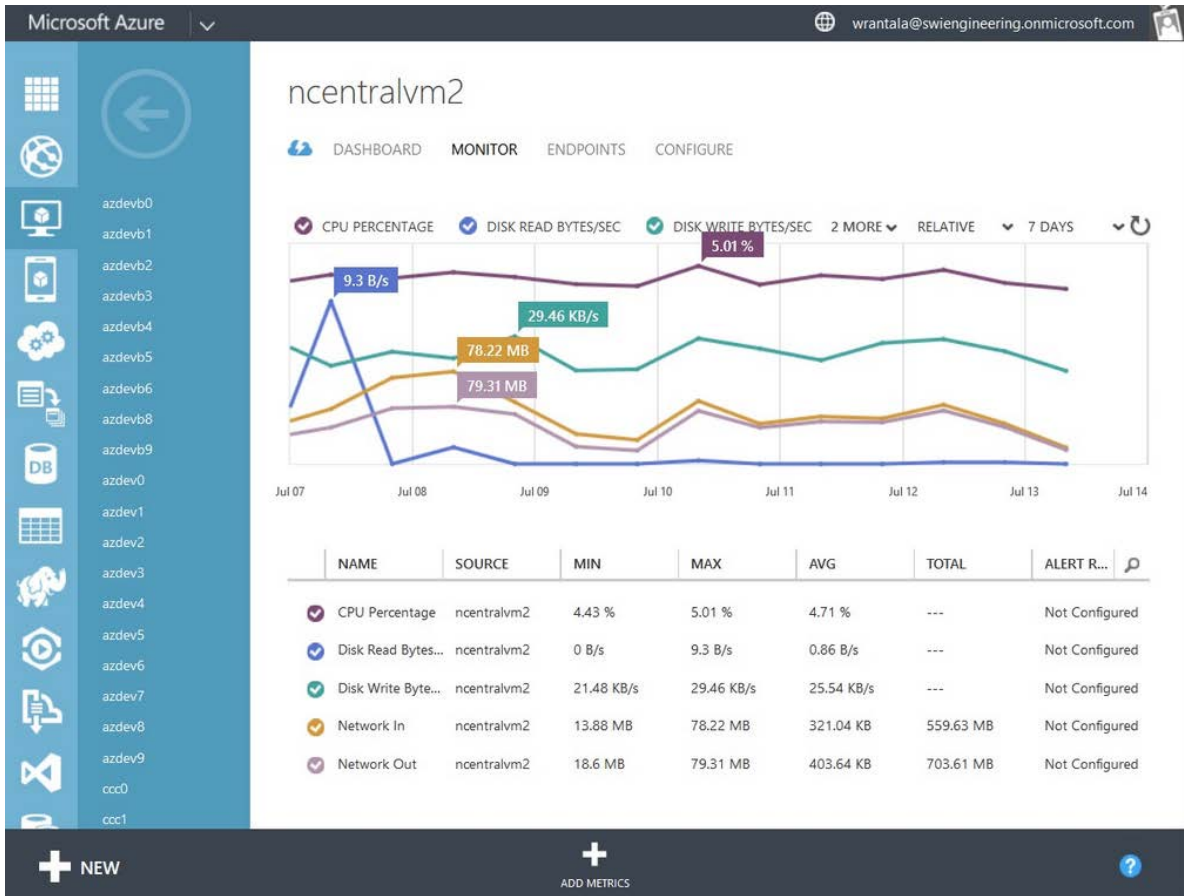


7. After you have everything set up in Azure, activate the N-central Server.

8. You must also set the Appliance Setting in your N-central instance for the probe/agent communication and to propagate any changes made to your Azure storage account.

   To make the required change in N-central, navigate to **Administration → Defaults → Appliance Settings** and add the Azure VM URL produced in Step 5 (specifically from the DNSName variable).

   The following is an example of this change:
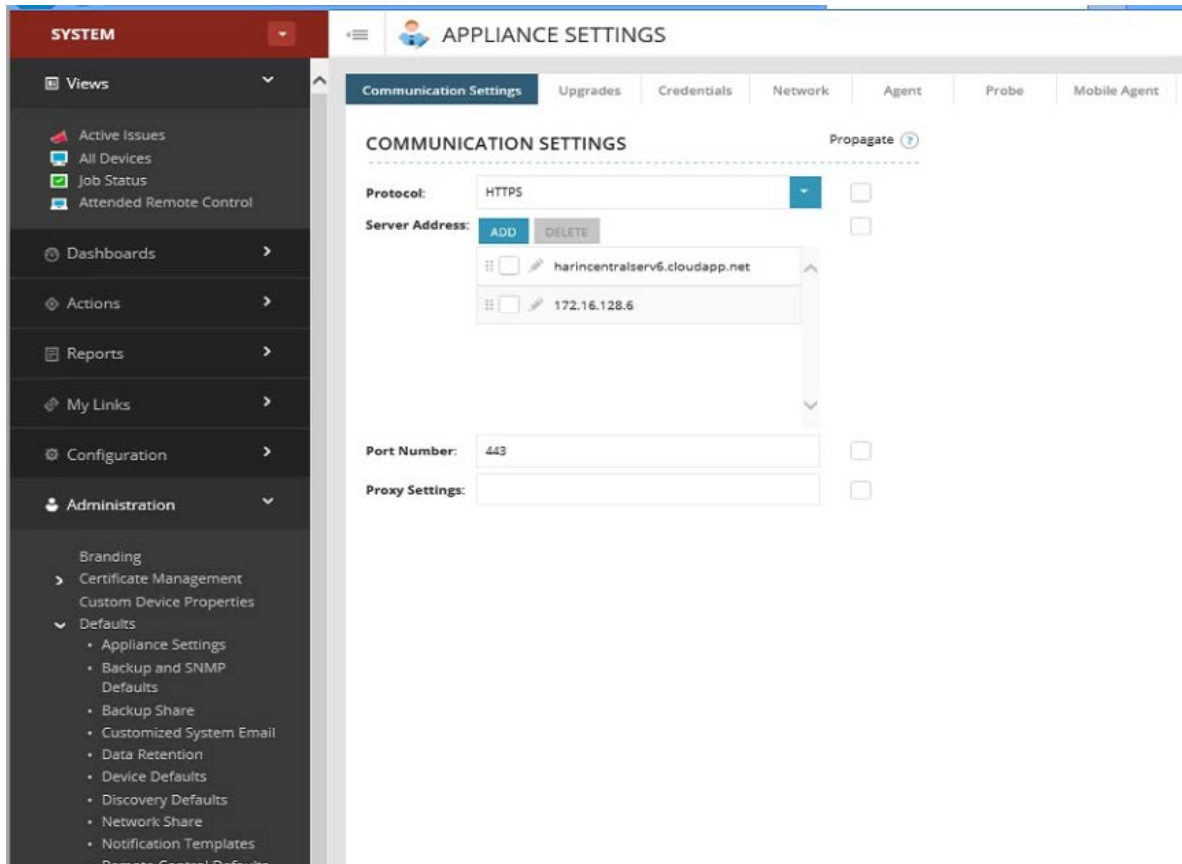
**Note:** If the VM is powered off in Azure, it will be unallocated. Powering it back up will cause N-central to require re-activation.

## Licensing and Support Information

### Agent/Probe Installation Software

N-central 10.x uses the 7-Zip file archiver for installing agents and probes. 7-Zip is free software redistributed under the terms of the GNU Lesser General Public License as published by the Free Software Foundation. For more information, see http://www.7-zip.org.

### Customer Support

Toll Free: 1-866-302-4689
Web Page: www.n-able.com
Technical Support Self-Service Portal: http://www.n-able.com/support/

### Feedback

SolarWinds N-able (formerly known as N-able Technologies) is a market driven organization that places importance on customer, partner and alliance feedback. All feedback is welcome at the following email address: n-able-feedback@solarwinds.com.

### About SolarWinds N-able

SolarWinds N-able is the global leader in remote monitoring and management software for managed service providers and IT departments. SolarWinds N-able's award-winning N-central platform and complementary toolsets, backed by best-in-class business and technical services, are proven to reduce IT support costs, improve network performance and increase productivity through the proactive monitoring, management and optimization of IP-enabled devices and IT infrastructure. SolarWinds N-able is 100% channel-friendly and maintains operations in North America, the U.K., the Netherlands and Australia.