Hortonworks Data Platform

Installing HDP Manually

(Feb 3, 2015)

docs.hortonworks.com

Hortonworks Data Platform : Installing HDP Manually

The Hortonworks Data Platform, powered by Apache Hadoop, is a massively scalable and 100% open source platform for storing, processing and analyzing large volumes of data. It is designed to deal with data from many sources and formats in a very quick, easy and cost-effective manner. The Hortonworks Data Platform consists of the essential set of Apache Hadoop projects including MapReduce, Hadoop Distributed File System (HDFS), HCatalog, Pig, Hive, HBase, ZooKeeper and Ambari. Hortonworks is the major contributor of code and patches to many of these projects. These projects have been integrated and tested as part of the Hortonworks Data Platform release process and installation and configuration tools have also been included.

Unlike other providers of platforms built using Apache Hadoop, Hortonworks contributes 100% of our code back to the Apache Software Foundation. The Hortonworks Data Platform is Apache-licensed and completely open source. We sell only expert technical support, training and partner-enablement services. All of our technology is, and will remain free and open source.

Please visit the Hortonworks Data Platform page for more information on Hortonworks technology. For more information on Hortonworks services, please visit either the Support or Training page. Feel free to Contact Us directly to discuss your specific needs.

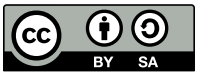


Table of Contents

List of Tables

1. Getting Ready to Install

This section describes the information and materials you need to get ready to install the Hortonworks Data Platform (HDP) on Windows.

Use the following instructions before you start deploying Hadoop using HDP installer:

- Understanding the HDP Components
- Meet Minimum System Requirements
- Prepare for Hadoop Installation

1.1. Understanding the HDP Components

The Hortonworks Data Platform consists of three layers.

- **Core Hadoop 2**: The basic components of Apache Hadoop version 2.x.
  - **Hadoop Distributed File System (HDFS)**: A special purpose file system designed to provide high-throughput access to data in a highly distributed environment.
  - **YARN**: A resource negotiator for managing high volume distributed data processing. Previously part of the first version of MapReduce.
  - **MapReduce 2 (MR2)**: A set of client libraries for computation using the MapReduce programming paradigm and a History Server for logging job and task information. Previously part of the first version of MapReduce.
- **Essential Hadoop**: A set of Apache components designed to ease working with Core Hadoop.
  - **Apache Pig**: A platform for creating higher level data flow programs that can be compiled into sequences of MapReduce programs, using Pig Latin, the platform's native language.
  - **Apache Hive**: A tool for creating higher level SQL-like queries using HiveQL, the tool's native language, that can be compiled into sequences of MapReduce programs.
  - **Apache HCatalog**: A metadata abstraction layer that insulates users and scripts from how and where data is physically stored.
  - **WebHCat (Templeton)**: A component that provides a set of REST-like APIs for HCatalog and related Hadoop components.
  - **Apache HBase**: A distributed, column-oriented database that provides the ability to access and manipulate data randomly in the context of the large blocks that make up HDFS.
  - **Apache ZooKeeper**: A centralized tool for providing services to highly distributed systems. ZooKeeper is necessary for HBase installations.

- **Supporting Components**: A set of components that allow you to monitor your Hadoop installation and to connect Hadoop with your larger compute environment.
  - **Apache Oozie**:A server based workflow engine optimized for running workflows that execute Hadoop jobs.
  - **Apache Sqoop**: A component that provides a mechanism for moving data between HDFS and external structured datastores. Can be integrated with Oozie workflows.
  - **Apache Flume**: A log aggregator. This component must be installed manually.
  - **Apache Mahout**: A scalable machine learning library that implements several different approaches to machine learning.
  - **Apache Knox**: A REST API gateway for interacting with Apache Hadoop clusters. The gateway provides a single access point for all REST interactions with Hadoop clusters.

For more information on the structure of the HDP, see Understanding the Hadoop Ecosystem.

While it is possible to deploy all of HDP on a single host, this is appropriate only for initial evaluation, see Cluster Planning Guide. In general you should use at least three hosts: one master host and two slaves.

1.2. Meet Minimum System Requirements

To run the Hortonworks Data Platform, your system must meet minimum requirements.

- Hardware Recommendations
- Operating System Requirements
- Software Requirements
- (Optional) Microsoft SQL Server Requirements

1.2.1. Hardware Recommendations

Although there is no single hardware requirement for installing HDP, there are some basic guidelines, see sample setups here: Hardware Recommendations for Apache Hadoop.

Note

When installing HDP, 1 GB of free space is required in on the system drive.

1.2.2. Operating Systems Requirements

The following operating systems are supported:

- Windows Server 2008 R2 (64-bit)
- Windows Server 2012 (64-bit)

1.2.3. Software Requirements

This section provides download locations and installation instructions for each software pre-requisite.

Table 1.1. HDFS Ports

Software	Version	Environment Variable	Description	Installation Notes
Python	2.7.X	PATH	Add the directory where Python is installed, following the instructions in this guide the path is C:\python.	Spaces in the path to the executable are not allowed. Do not install Python in the default location (Program Files), see install from PS CLI or Manually.
Java JDK	JDK 1.7.0 51	PATH	Add the directory where Java application is installed. For example C:\java\jdk1.7.0\bin	Spaces in the path to the executable are not allowed. Do not install Java in the default location (Program Files) see install from PS CLI or Manually
		JAVA_HOME	Create a new system variable for JAVA_HOME that points to the directory where the JDK is installed. For example C:\java\jdk1.7.0.	
Microsoft Visual C++	2010	PATH	Default location automatically added.	Install with default parameters, see PS CLI.
Microsoft .NET Framework	4.0	PATH	Default location automatically added.	Install with default parameters, see PS CLI.

1.2.4. (Optional) MS SQL Server for Hive and Oozie Database Instances

By default, Hive and Oozie use an embedded Derby database for its metastore. However you can also use Microsoft SQL server.

Note

For details on installing and configuring Microsoft SQL Server, see TechNet instructions, such as SQL Server 2012

- To use an external database for Hive and Oozie metastores, ensure that Microsoft SQL Server database is deployed and available in your environment and that your database administrator creates the following databases and users. You need the following details while configuring the HDP Installer:
  - For Hive, a SQL database instance:
    1. Create Hive database instance in SQL, and record the name such as *hive_dbname*.

2. Create Hive user on SQL and add them to the `sysadmin` role within SQL, and record the name and password such as *`hive_dbuser`*/*`hive_dbpasswd`*.

3. Set the security policy for SQL to use both SQL and Windows authentication, the default setting is Windows authentication only.

- For Oozie, a SQL database instance:

1. Create an Oozie database instance and record the name, such as *`oozie_dbname`*.

2. Create Oozie user on SQL and add them to the `sysadmin` role within SQL and record the user name and password, such as *`oozie_dbuser`*/*`oozie_dbpasswd`*.

3. Set the security policy for SQL to use both SQL and Windows authentication, the default setting is Windows authentication only.

Important

Before using SQL server for Hive or Oozie metastores, you must set up Microsoft SQL Server JDBC Driver after installing the components using the instructions provided here.

1.3. Prepare for Hadoop Installation

To deploy HDP across a cluster, you need to prepare your multi-node cluster deploy environment. Follow these steps to ensure each cluster node is prepared to be an HDP cluster node:

- Gather Host Information
- Configure Network Time Server
- Set Interfaces to IPv4 addresses Preferred
- (Optional) Create Hadoop user
- Enable Remote Powershell Script Execution
- Configure ports
- Install required Software

1.3.1. Gather Hadoop Cluster Information

To deploy your HDP installation, you need to collect the **Hostname** OR **IPv4 address** of each the following cluster component:

- Required Components:
  - NameNode and optional Secondary NameNode
  - ResourceManager

- Hive Server
- SlaveNode
- WebCat
- Client Host

- Optional Components:
  - ZooKeeper
  - HBase Master
  - Flume
  - Knox Gateway
  - Microsoft SQL Server configured with a Hive and Oozie database instance, system account names and passwords

Important

The installer fails if it cannot resolve the hostname of each cluster node. To determine the hostname for a particular cluster node, open the command line interface on that system and execute **hostname** and then **nslookup** *hostname* to verify that the name resolves to the correct IP address.

1.3.2. Configure Network Time Server

The clocks of all the nodes in your cluster must be able to synchronize with each other. To configure this for Windows Server, use the instructions provided here.

1.3.3. Set Interfaces to IPv4 Preferred

Configure all the Windows Server nodes in your cluster to use IPv4 addresses only. You can either disable IPv6, see How to disable IP version 6 or its specific components in Windows or set the preference to IPv4.

Ensure that the host FQDN resolves to an IPv4 address as follows:

1. Open a command prompt and verify that IPv4 is set to preferred:

```
ipconfig /all

Connection-specific DNS Suffix . . :
Description . . . . . . . . . . . : Intel(R) PRO/1000 MT Network
Connection Physical Address. . . . : XX-XX-XX-XX-XX
DHCP Enabled. . . .   . . . . . . . : No
Autoconfiguration Enabled .  . . . : Yes
IPv4 Address. . . . . . . . . . . : 10.0.0.2(Preferred)
Subnet Mask . . . . . . . . . . . : 255.255.255.0
Default Gateway . . . . . . . . . : 10.0.0.100
DNS Servers . . . . . . . . . . . : 10.10.0.101
NetBIOS over Tcpip. . . . . . . . : Enabled
```

2. Flush the DNS cache:

```
ipconfig /flushdns
```

3. Verify that the hostname of the system resolves to the correct IP address:

```
ping -a 10.0.0.2

Pinging win08r2-node1.HWXsupport.com 10.0.0.2 with 32 bytes of data:
Reply from 10.0.0.2: bytes=32 time<1ms TTL=128
Reply from 10.0.0.2: bytes=32 time<1ms TTL=128
Reply from 10.0.0.2: bytes=32 time<1ms TTL=128
```

1.3.4. (Optional) Create Hadoop user

HDP installer takes the following actions to create hadoop user for your environment:

- If the user hadoop does not exist, HDP installer automatically creates a local user with random password.
- If the user hadoop already exists, HDP installer will change the current password to a new random password. The random password is passed on the command line throughout the install process, then discarded. Administrator can change the password later, but it must be done both in the user configuration and in the service objects installed on each machine via Service Manager.

1.3.5. Enable Remote Powershell Script Execution

The MSI installation scripts and many utility scripts within HDP require remote execution of Powershell scripts are enabled on all nodes in the Hadoop cluster. For example, the scripts for starting and stopping the entire cluster with a single command that are provided with HDP requires remote scripting and trust to be enabled. Therefore, we strongly recommend that you complete the following three settings on every host in your cluster.

1.3.5.1. Enable Remote PS Execution for Nodes in a Workgroup

You can set these in Active Directory via Group Policies (for a Group including all hosts in your Hadoop cluster), or you can execute the given Powershell commands on every host in your cluster.

Important

Ensure that the Administrator account on the Windows Server node has a password. The remote scripting below will not work if the Administrator account has an empty password.

Enable remote scripting using Powershell commands

1. On each host in the cluster, execute the following commands in a Powershell window with "Run as Administrator" elevation:

```
Set-ExecutionPolicy "AllSigned"

Enable-PSRemoting

Set-item wsman:localhost\client\trustedhosts -value "Host1,Host2"
```

The last argument is a list of comma-separated hostnames in your cluster (for example, "`HadoopHost1`, `HadoopHost2`, `HadoopHost3`").

2. On each host in the cluster, execute the following commands in a Powershell window with "Run as Administrator" elevation:

```
winrm quickconfig
 winrm set winrm/config/client @{TrustedHosts="host1, host2, host3"}
```

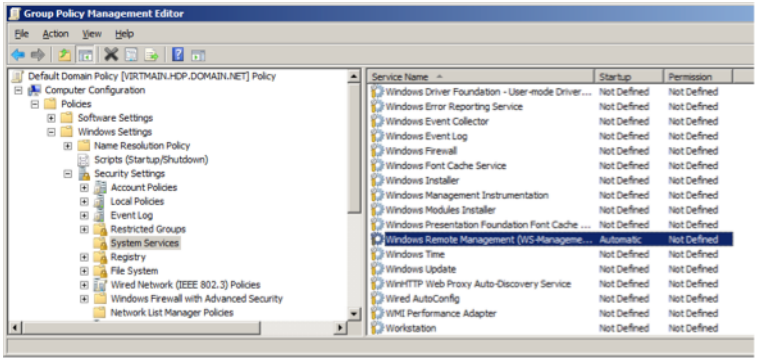
The last argument is a list of comma-separated hostnames in your cluster (for example, "`HadoopHost1`, `HadoopHost2`, `HadoopHost3`").

1.3.5.2. Enable networking configurations for Active Directory Domains

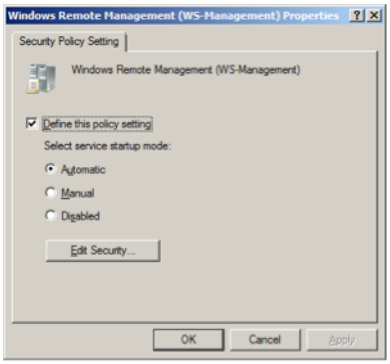
To enable remote scripting and to configure right domain policies for Windows Remote Management complete the following instructions on a domain controller machine (all actions are performed via **Group Policy Management\Default Domain Policy/Edit**):

1. Set the WinRM service to auto start.

   - Go to **Computer Configuration** -> **Policies** -> **Windows Settings** -> **Security Settings** -> **System Services** -> **Windows Remote Management (WS-Management)**.



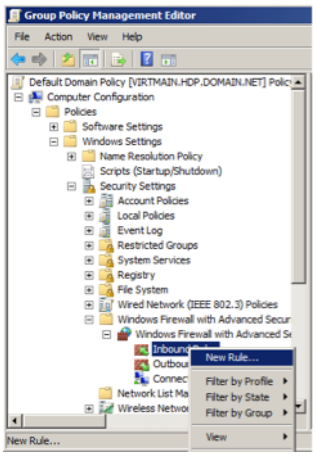
   - Set **Startup Mode** to **Automatic**.



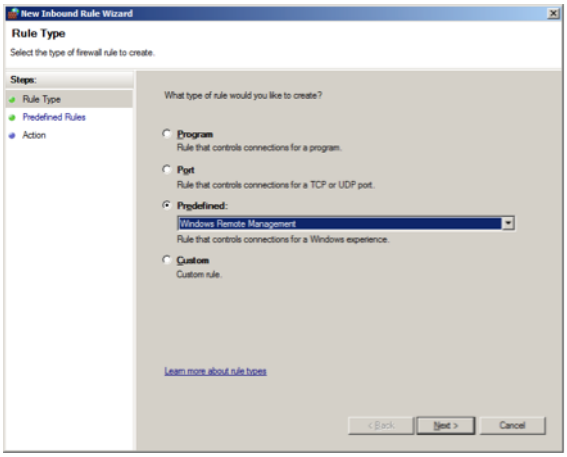
2. Add firewall exceptions to allow the service to communicate.

   - Go to **Computer Configuration** -> **Policies** -> **Windows Settings** -> **Security Settings** -> **Windows Firewall with Advanced Security** .

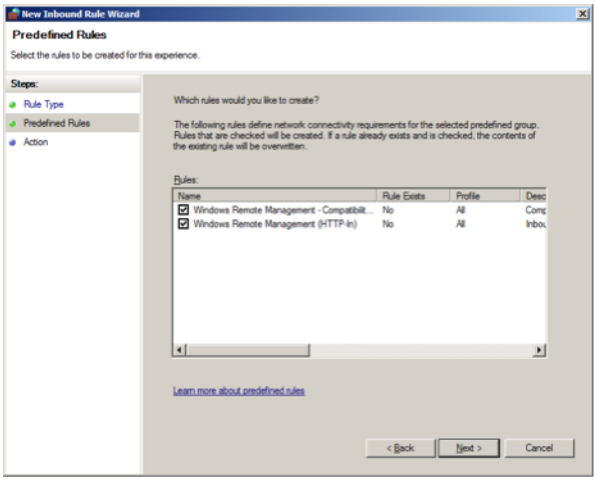
- Right click on **Windows Firewall with Advanced Security** to create a new Inbound Rule.



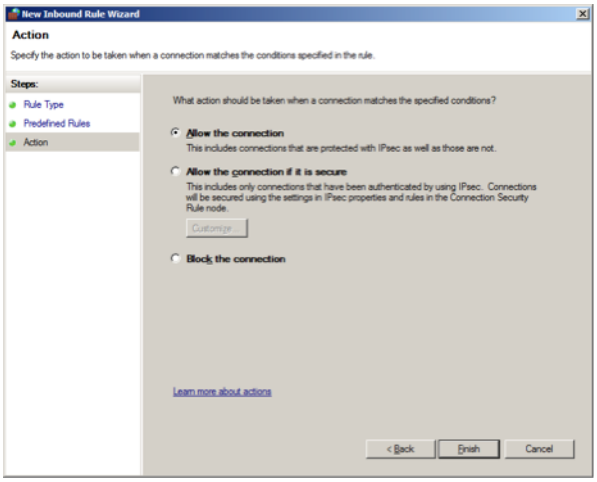
- Select the type of rule as **Predefined** as **Windows Remote Management** .



The Predefined rule will automatically create two rules as shown below:



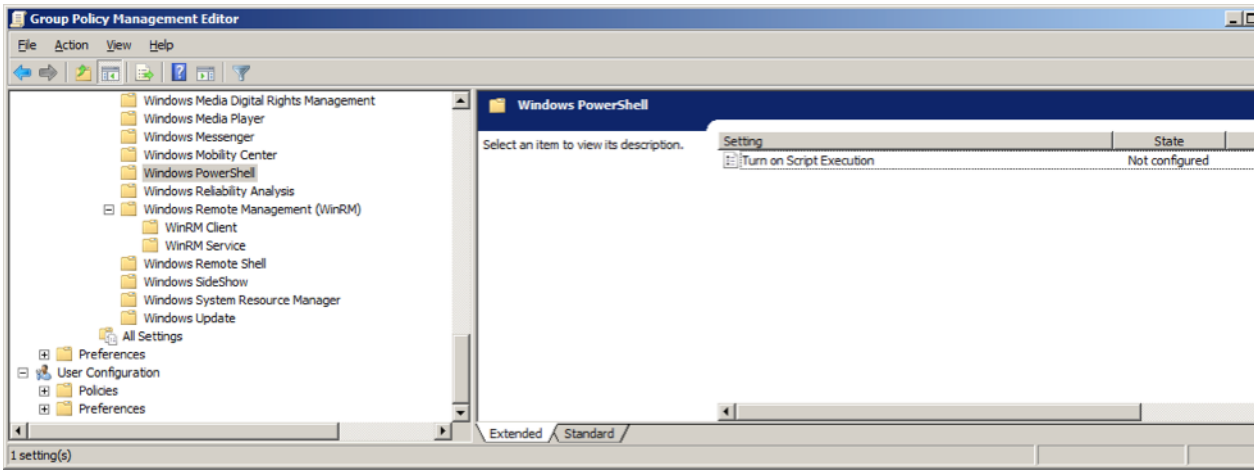
- Configure the **Action** as **Allow the connection** and click **Finish**.



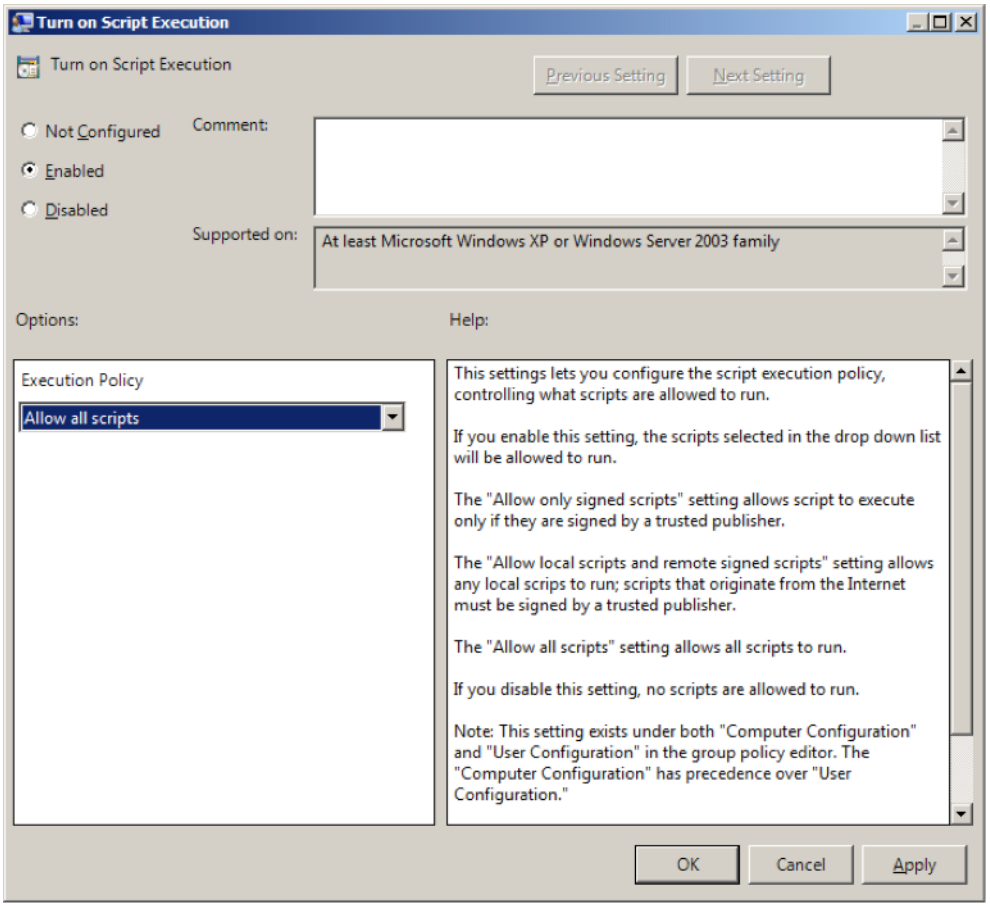
3. Set script execution policy.

- Go to **Computer Configuration** -> **Policies** -> **Administrative Templates** -> **Windows Components** -> **Windows PowerShell** .

- Enable **Script Execution** .

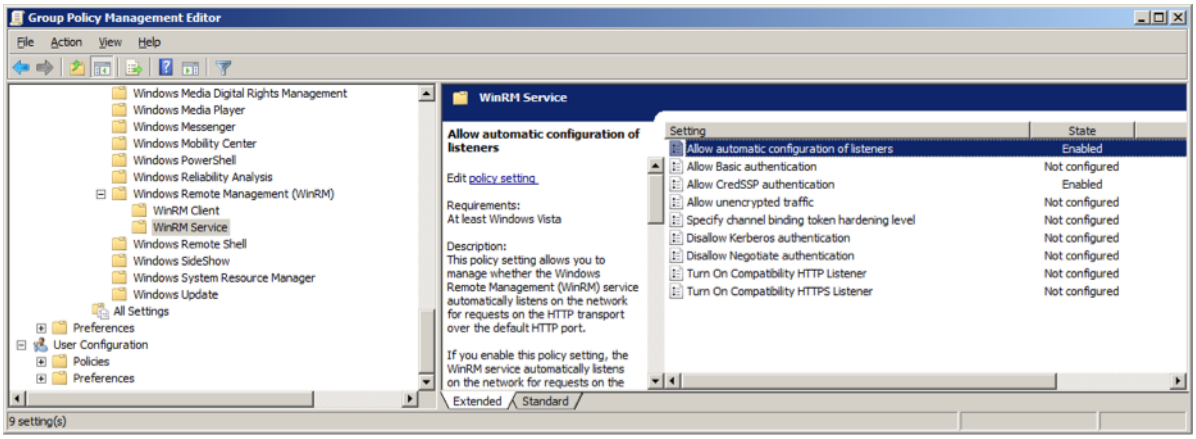


- Set Execution Policy to **Allow all scripts**.



4. Setup WinRM service.

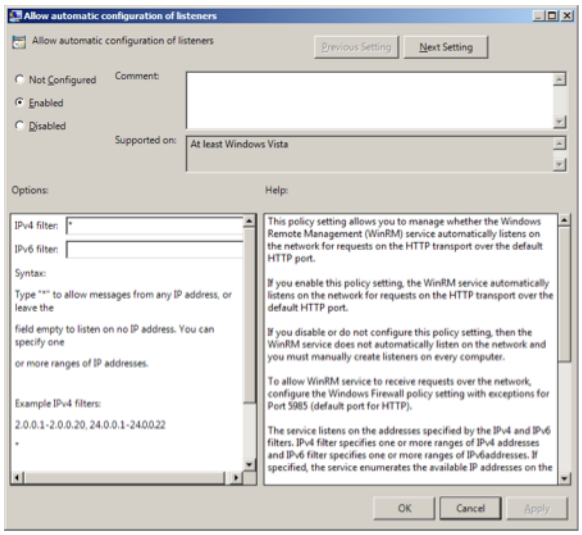
   - Go to **Computer Configuration** -> **Policies** -> **Administrative Templates** -> **Windows Components** -> **Windows Remote Management (WinRM)** -> **WinRM Service**.



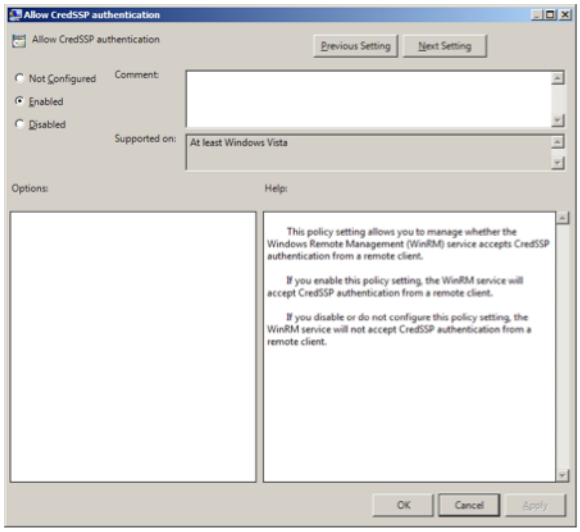
   - Create a WinRM listener.

     a. To allow automatic configuration of listeners, select **Enabled**.

b. Set **IPv4 filter** to * (all addresses or specify range)

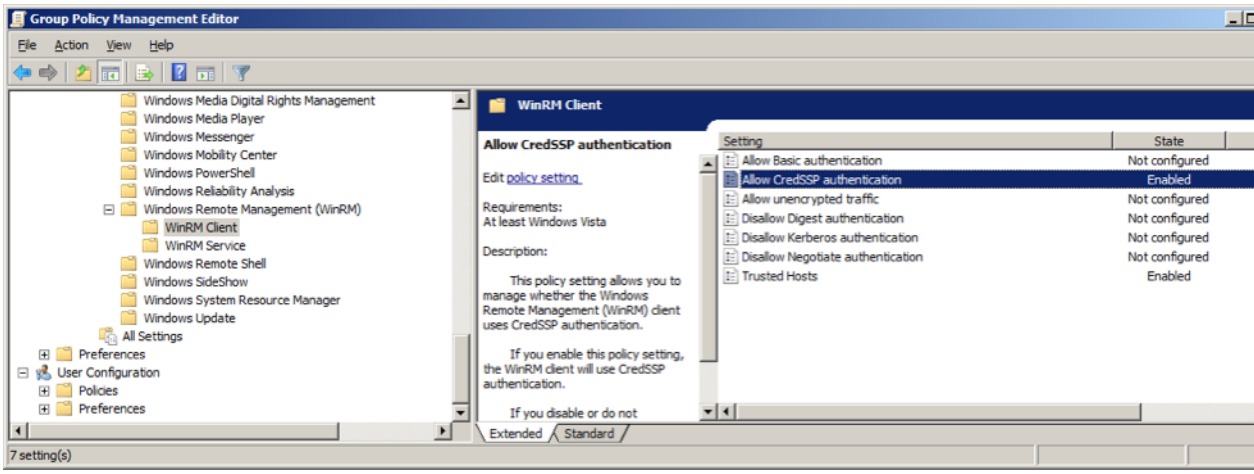


c. Allow CredSSP authentication and click OK.

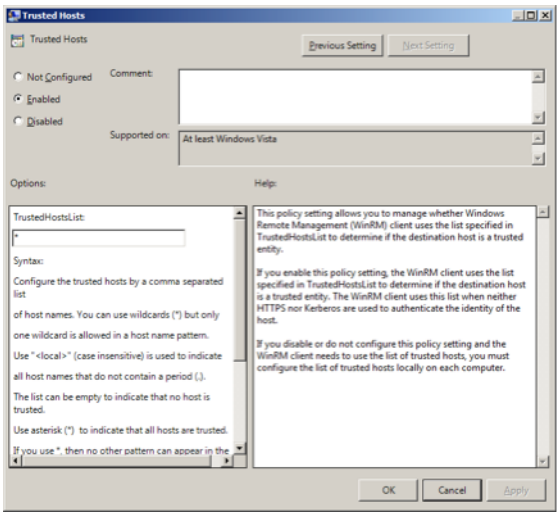


5. Setup WinRM client.

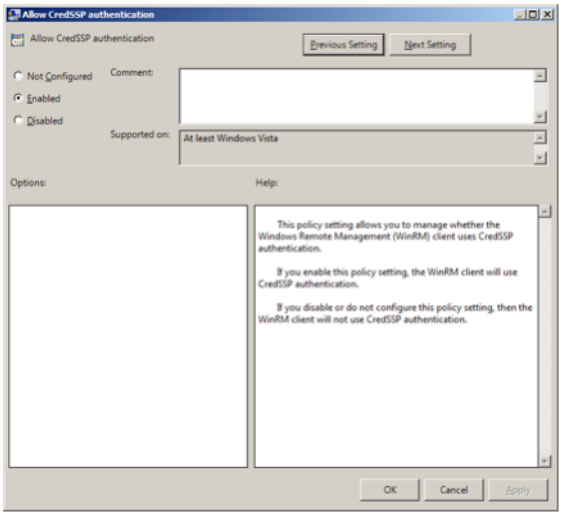
- Go to **Computer Configuration** -> **Policies** -> **Administrative Templates** -> **Windows Components** -> **Windows Remote Management (WinRM)** -> **WinRM Client**.



- Configure the trusted host list (the IP addresses of the computers that can initiate connections to the WinRM service). To do this, set **TrustedHostsList** to * (all addresses or specify range).

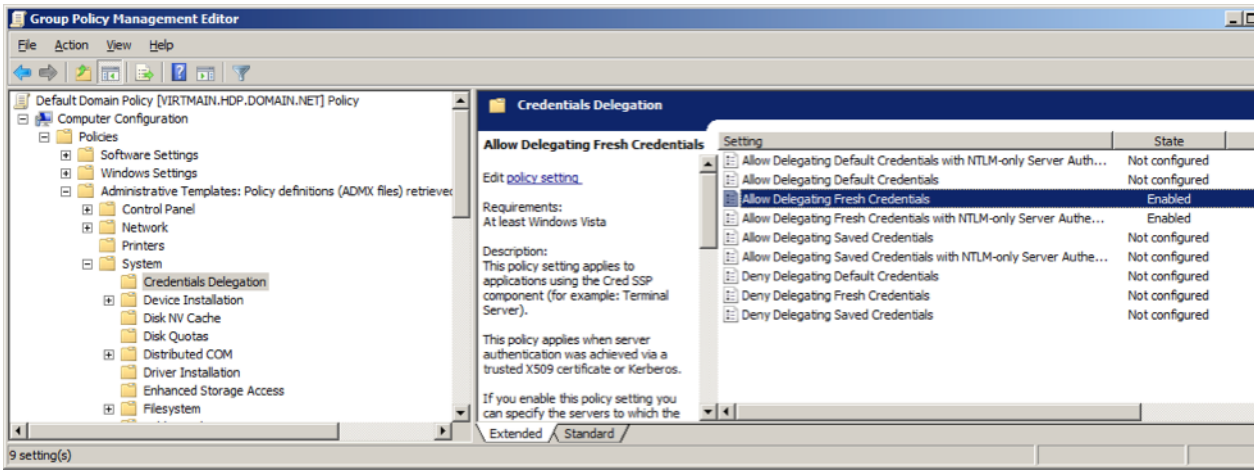


- Allow CredSSP authentication and click OK.



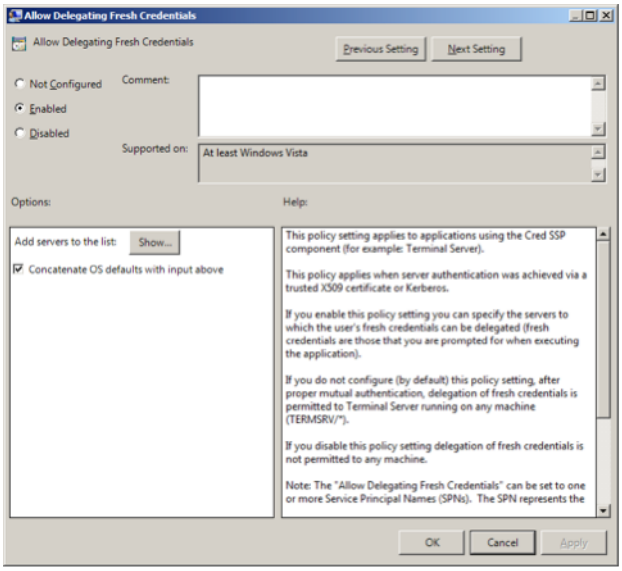
6. Enable credentials delegation.

   - Go to **Computer Configuration** -> **Policies** -> **Administrative Templates** -> **System** -> **Credentials Delegation**.

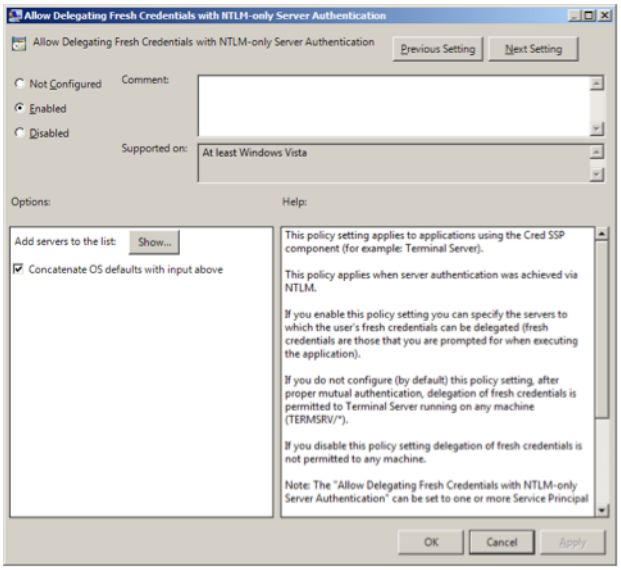


   - Select **Enabled** to allow delegation fresh credentials.

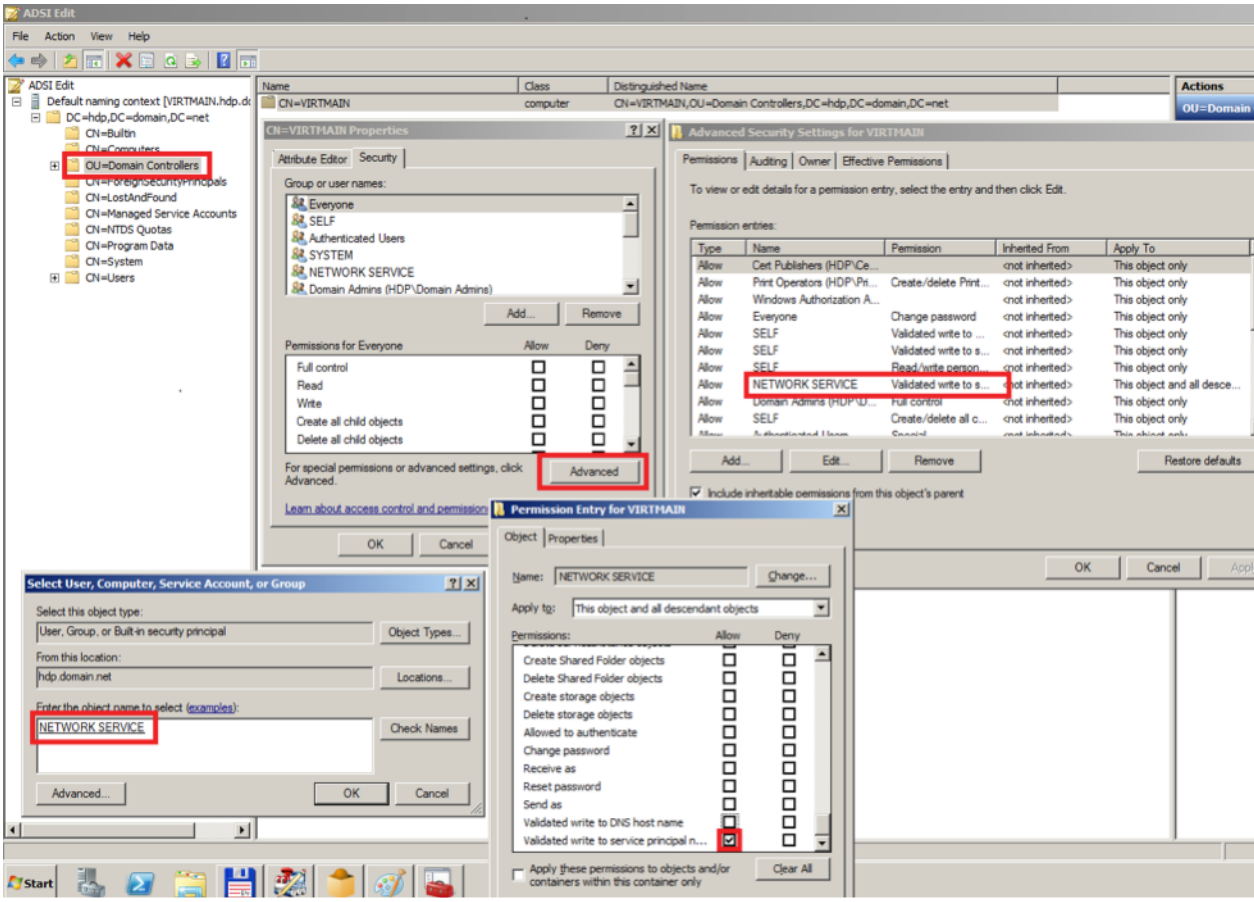
   - Under **Options** click on **Show**. Set **WSMAN** to * (all addresses or specify range). Click on **Next Setting**.



- Select **Enabled** to allow delegation fresh credentials with NTLM-only server authentication.
- Under **Options** click on **Show**. Set **WSMAN** to * (all addresses or specify range). Click on **Finish**.



7. Enable creating WSMAN SPN.
   - Go to **Start**-> **Run**. In the dialog box, type `ADSIEdit.msc` and click **Enter**.
   - Expand **OU=Domain Controllers** menu item and select **CN=domain controller hostname**. Go to **Properties** -> **Security** -> **Advanced** -> **Add**.
   - Enter **NETWORK SERVICE**, click **Check Names**, then **Ok**. In the Permission Entry select **Validated write to service principal name**. Click **Allow** and **OK** to save your changes.



8. Restart WinRM service and update policies.

   - On the domain controller machine, execute the following commands in PowerShell:

     ```
     Restart-Service WinRM
     ```

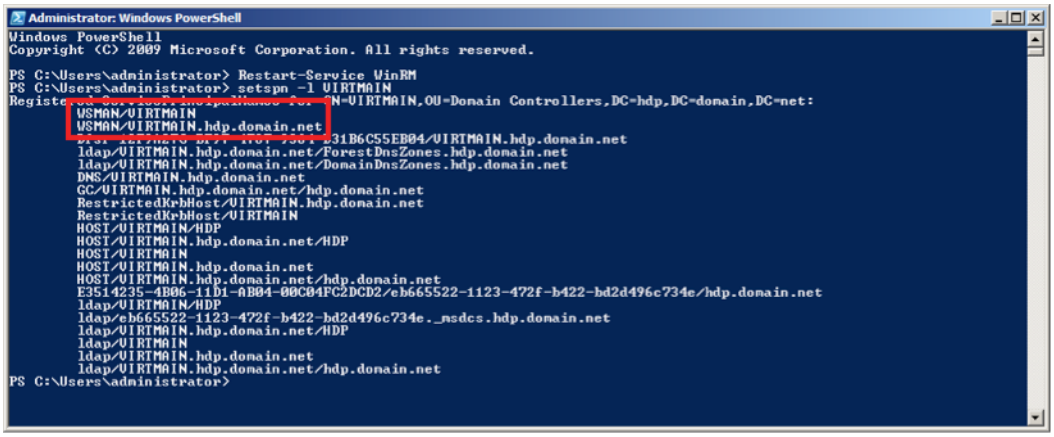
   - On other hosts in domain, execute the following commands:

     ```
     gpupdate /force
     ```

   - Ensure that SPN-s WSMAN is created for your environment. Execute the following command on your domain controller machine:

     ```
     setspn -l $Domain_Controller_Hostname
     ```

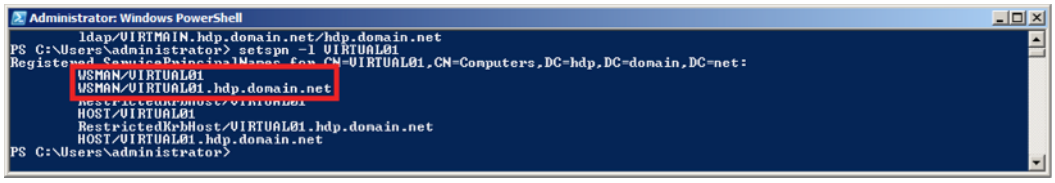
     You should see output similar to the following:



9. Check the WSMAN SPN on other host in domain. Execute the following command on any one of your host machines:

```
setspn -l $Domain_Controller_Hostname
```

You should see output similar to the following:



1.3.6. Configure ports

HDP uses multiple ports for communication with clients and between service components. To enable HDP communication, open the specific ports that HDP uses.

To open specific ports only, you can set the access rules in Windows.

For example, the following command will open up port 80 in the active Windows Firewall:

```
netsh advfirewall firewall add rule name=AllowRPCCommunication dir=in action=
allow protocol=TCP localport=80
```

For example, the following command will open up ports 49152-65535 in the active Windows Firewall:

```
netsh advfirewall firewall add rule name=AllowRPCCommunication dir=in action=
allow protocol=TCP localport=49152-65535
```

The tables below specify which ports must be opened for which ecosystem components to communicate with each other.

Make sure that appropriate ports are opened before you install HDP.

HDFS Ports: The following table lists the default ports used by the various HDFS services.

Table 1.2. HDFS Ports

Service	Servers	Default Ports Used	Protocol	Description	Need End User Access?	Configuration Parame
NameNode WebUI	Master Nodes (NameNode and any back-up NameNodes)	50070	http	Web UI to look at current status of HDFS, explore file system	Yes (Typically admins, Dev/ Support teams)	`dfs.http.address`
NameNode metadata service		8020/9000	IPC	File system metadata operations	Yes (All clients who directly need to interact with the HDFS)	Embedded in URI spec
DataNode	All Slave Nodes	50075	http	DataNode WebUI to access the status, logs etc.	Yes (Typically admins, Dev/ Support teams)	`dfs.datanode.htt`
		50010		Data transfer		`dfs.datanode.add`
		50020	IPC	Metadata operations	No	`dfs.datanode.ipc`
Secondary NameNode	Secondary NameNode and any backup Secondanry NameNode	50090	http	Checkpoint for NameNode metadata	No	`dfs.secondary.ht`

YARN Ports: The following table lists the default ports used by the various YARN services.

Table 1.3. YARN Ports

Service	Servers	Default Ports Used	Protocol	Description	Need End User Access?	Configuration Parameters
Resource Manager WebUI	Master Nodes (Resource Manager and any back-up Resource Manager node)	8088	http	Web UI for Resource Manager	Yes	`yarn.resourcemanager.webapp.address`
	Master Nodes (Resource Manager and any back-up Resource Manager node)	8090	https	Web UI for Resource Manager	Yes	`yarn.resourcemanager.webapp.https.a`
Resource Manager	Master Nodes (Resource Manager Node)	8032	IPC	For application submissions	Yes (All clients who need to submit the YARN applications including Hive, Hive server, Pig)	Embedded in URI specified by `yarn.resourcemanager.address`
Resource Manager Admin Interface	Master Nodes (Resource Manager and any back-up Resource Manager node)	8033		Administrative interface	Yes (Typically admins and support teams)	`yarn.resourcemanager.admin.address`
Resource Manager Scheduler	Master Nodes (Resource Manager and any back-up Resource Manager node)	8031	http	Resource Manager Interface	Yes (Typically admins, Dev/ Support teams)	`yarn.resourcemanager.scheduler.addr`
NodeManager Web UI	All Slave Nodes	50060	http		Yes (Typically admins, Dev/ Support teams)	`yarn.nodemanager.webapp.address`

Hive Ports: The following table lists the default ports used by the Hive services.

Table 1.4. Hive Ports

Service	Servers	Default Ports Used	Protocol	Description	Need End User Access?	Configuration Parameters
HiveServer2	HiveServer2 machine (Usually a utility machine)	10001	thrift	Service for programmatically (Thrift/ JDBC) connecting to Hive	Yes	`ENV Variable HIVE_PORT`
Hive Server	Hive Server machine (Usually a utility machine)	10000	thrift	Service for programmatically (Thrift/ JDBC) connecting to Hive	Yes (Clients who need to connect to Hive either programatically or through UI SQL tools that use JDBC)	`ENV Variable HIVE_PORT`
Hive Metastore		9083	thrift	Service for programmatically (Thrift/ JDBC) connecting to Hive metadata	Yes (Clients that run Hive, Pig and potentially M/R jobs that use HCatalog)	`hive.metastore.uris`

WebHcat Port: The following table lists the default port used by the WebHCat service.

Table 1.5. WebHCat Port

Service	Servers	Default Ports Used	Protocol	Description	Need End User Access?	Configuration Parameters
WebHCat Server	Any utility machine	50111	http	Web API on top of HCatalog and other Hadoop services	Yes	`templeton.port`

Table 1.6. HBase Ports

Service	Servers	Default Ports Used	Protocol	Description	Need End User Access?	Configuration Parameters
HMaster	Master Nodes (HBase Master Node and any back-up HBase Master node)	60000			Yes	`hbase.master.port`

Service	Servers	Default Ports Used	Protocol	Description	Need End User Access?	Configuration Parameters
HMaster Info Web UI	Master Nodes (HBase master Node and back up HBase Master node if any)	60010	http	The port for the HBase-Master web UI. Set to -1 if you do not want the info server to run.	Yes	`hbase.master.info.port`
Region Server	All Slave Nodes	60020			Yes (Typically admins, dev/ support teams)	`hbase.regionserver.port`
Region Server	All Slave Nodes	60030	http		Yes (Typically admins, dev/ support teams)	`hbase.regionserver.info.port`
ZooKeeper	All ZooKeeper Nodes	2888		Port used by ZooKeeper peers to talk to each other. See here for more information.	No	`hbase.zookeeper.peerport`
ZooKeeper	All ZooKeeper Nodes	3888		Port used by ZooKeeper peers to talk to each other. See here for more information.		`hbase.zookeeper.leaderport`
		2181		Property from ZooKeeper's config `zoo.cfg`. The port at which the clients will connect.		`hbase.zookeeper.property.clientPort`

1.3.7. Install Required Software

On each node in the cluster the following software must be installed:

- Microsoft .NET Framework v4.0, see Installing .NET with PS CLI

- Microsoft Visual C++ 2010 only, see Installing Visual C++ Distributable Package from PS CLI
- (Optional)
- Java version: 1.7.0_51, see Installing Java JDK from PS CLI or Manually installing Java JDK
- Python v2.7 or higher, see Installing Python from PS CLI or Manually installing Python

1.3.7.1. Installing Required Software using Powershell CLI

Identify a workspace directory that will have all the software installation files. In the powershell instructions in this section, $env:WORKSPACE refers to the full path of the workspace directory using an environment variable, for example:

```
setx WORKSPACE "C:\workspace" /m
```

After setting the environment variable using setx from the command prompt, you must restart the powershell cli. When using a script you may want to set Workspace as a standard PS variable to avoid having to restart powershell.

Ensure that you install the following software on every host machine in your cluster:

- **Python 2.7.X**

  Use the following instructions to manually install Python in your local environment:

  1. Download Python from here to the workspace directory.
  2. Install Python and update the PATH environment variable. Using Administrator privileges. From the Powershell window, execute the following commands as `Administrator` user:

     ```
     $key = "HKLM:\SYSTEM\CurrentControlSet\Control\Session Manager\
     Environment”
     $currentPath =  (Get-ItemProperty -Path $key -name Path).Path + ';'
     $pythonDir = "C:\Python\"

     msiexec  /qn /norestart /l* $env:WORKSPACE\python_install.log /i
      $env:WORKSPACE\python-2_7_5_amd64.msi TARGETDIR=$pythonDir ALLUSERS=1
     setx PATH "$currentPath$pythonDir" /m
     ```

     where *WORKSPACE* is an environment variable for the directory path where the installer is located.

     Important

     Ensure the downloaded Python MSI name matches `python-2_7_5_amd64.msi`. If not, change the above command to match the MSI file name.

- **Microsoft Visual C++ 2010 Redistributable Package (64-bit)**

  1. Use the instructions provided here to download Microsoft Visual C++ 2010 Redistributable Package (64-bit) to the workspace directory.
  2. Execute the following command from Powershell with Administrator privileges:

```
& "$env:WORKSPACE\vcredist_x64.exe" /q /norestart /log "$env:WORKSPACE\
C_2010_install.log"
```

- **Microsoft .NET framework 4.0**

  1. Use the instructions provided here to download Microsoft .NET framework 4.0 to the workspace directory.

  2. Execute the following command from Powershell with Administrator privileges:

```
& "$env:WORKSPACE\NDP451-KB2858728-x86-x64-AllOS-ENU.exe" /q /norestart /
log "$env:WORKSPACE\NET-install_log.htm"
```

- **JDK version 7**

  Use the instructions provided below to manually install JDK to the workspace directory:

  1. Check the version. From a command shell or Powershell window, type:

```
java -version
```

Note

Uninstall the Java package if the JDK version is less than v1.6 update 31.

  2. Go to Oracle Java SE Downloads page and download the JDK installer to the workspace directory.

  3. From Powershell with Administrator privileges, execute the following commands:

```
$key = "HKLM:\SYSTEM\CurrentControlSet\Control\Session Manager\
Environment”
$currentPath =  (Get-ItemProperty -Path $key -name Path).Path + ';'
$javaDir = "C:\java\jdk1.7.0_51\"

& "$env:WORKSPACE\jdk-7u51-windows-x64.exe" /qn /norestart /log
 "$env:WORKSPACE\jdk-install.log"  INSTALLDIR="C:\java" ALLUSERS=1
setx JAVA_HOME "$javaDir" /m
setx PATH "$currentPath$javaDir\bin" /m
```

  where *WORKSPACE* is an environment variable for the directory path where the installer is located and *C:\java\jdk1.7.0_51* is the path where java will be installed. Ensure that no whitespace characters are present in the installation directory's path. For example, C:\Program Files is not allowed.

  4. Verify your installation and that the Java application is in your Path environment variable. From a command shell or Powershell window, type:

```
java -version
java version "1.7.0_51"
Java(TM) SE Runtime Environment (build 1.7.0_51-b18)
Java HotSpot(TM) 64-Bit Server VM (build 24.51-b03, mixed mode)
```

1.3.7.2. Installing the Required Software Manually

This section explains how to manually install the following software:

- **Microsoft Visual C++ 2010 Redistributable Package (64 bit)**: Download and install using the defaults.
- **Microsoft .NET framework 4.0**: Download and install using the defaults.
- **Java JDK**
- **Python**

To manually install Oracle Java JDK:

1. Download the Oracle JDK and install to a directory that contains no whitespace in the path, such as `C:\Java`.
2. Open the **Control Panel -> System** pane and click on **Advanced system settings**.
3. Click **Advanced**.
4. Click **Environment Variables**.
5. Add a system environment variable, `JAVA_HOME`:
   a. Under **System variables**, click **New**.
   b. Enter the **Variable Name** as `JAVA_HOME`.
   c. Enter the **Value** as the installation path for the Java Development Kit, such as `C:\Java\jdk1.7.0_51`.
   d. Click **OK**.
   e. To validate the setting, open a DOS cli and type:

```
echo %JAVA_HOME%
C:\Java\jdk1.7.0_45\
```

   The path to the Java installation is returned.
6. Update the `PATH` environment variable. Using Administrator privileges:
   a. Under **System Variables**, find `PATH` and click **Edit**.
   b. After the last entry in the Path value, enter a semi-colon and the installation path to the JDK, such as `;C:\Java\jdk1.7.0_51\bin`..
   c. Click **OK**.
   d. To validate the setting, open a DOS cli and type:

```
Java -version
java version "1.7.0"
...
```

   The Java version and details is returned.
7. Click **OK** to close the Environment Variables dialog box.

To manually install Python:

1. Download Python from here and install to a directory that contains no whitespace in the path, such as `C:\Python`.

2. Update the PATH environment variable. Using Administrator privileges:

   a. Open the **Control Panel -> System** pane and click on the **Advanced system settings** link.

   b. Click on the **Advanced** tab.

   c. Click the **Environment Variables** button.

   d. Under **System Variables**, find PATH and click **Edit**.

   e. After the last entry in the Path value, enter a semi-colon and the installation path to the Python installation directory, such as

      `;C:\Python27`.

   f. Click **OK** twice to close the Environment Variables dialog box.

   g. To validate your settings, from a command shell or Powershell window, type:

```
python -V
Python 2.7.6
```

2. Defining Hadoop Cluster Properties

The Hortonworks Data Platform consists of multiple components that are installed across the cluster. The cluster properties file specifies the directory locations and node host names for each of the components. The installer checks the hostname against the properties file to determine which services to install when you run the installer.

Use one of the following methods to modify the cluster properties file:

- Option I - Use the HDP Setup Interface to generate a cluster properties file for GUI use or export a generated `clusterproperties.txt` for a CLI installation. Recommended for first-time users and single-node installations.
- Option II - Manually Define Cluster Properties to manually create a `clusterproperties.txt` file, if you are familiar with your systems and HDP requirements.

2.1. Downloading the HDP Installer

Download the HDP Installation zip, and extract the files. The zip contains the following files:

- HDP MSI installer
- Sample `clusterproperties.txt` file
- Compression files:
  - `hadoop-lzo-0.4.19.2.1.10.0-2299`
  - `gplcompression.dll`

Optional, when implementing HDFS compression download the LZO compression DLL from here.

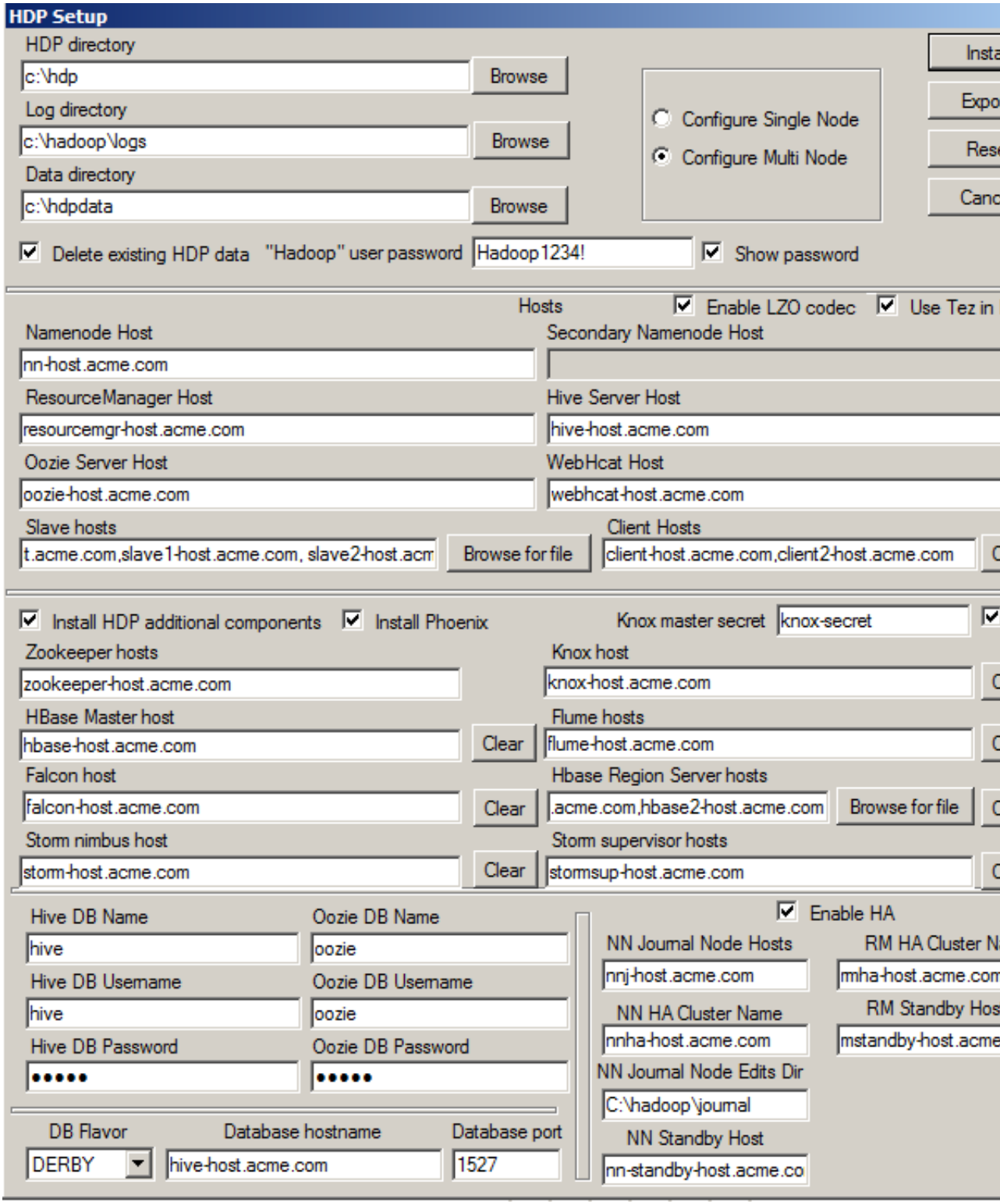
2.2. Using the HDP Setup Interface

You can define the cluster properties using the HDP Setup form. After you set the cluster property fields, you can then either export the configuration and use it deploy HDP from the command line, or you can complete the form and .

1. Open the command prompt and enter the following command:

```
runas /user:adminstrator msiexec /i "hdp-2.1.10.0.winpkg.msi"
 MSIUSEREALADMINDETECTION=1
```

   The HDP Setup form displays.



2. Choose the type of deployment by selecting:

- **Configure Single Node**: To install all cluster nodes on the current host; the hostname fields are pre-populated with the name of the current computer, see Quick Start Guide for Single Node Installation.
- **Configure Multi Node**: To create a property file for cluster deployment or to manually install a node (or subset of nodes) on the current computer.

3. Set the fields in the required components:

Configuration Values for HDP Setup form

Configuration Property Name	Description	Exampl
HDP directory	HDP installation directory.	d:\hdp
Log directory	HDP's operational logs are written to this directory on each cluster host. Ensure that you have sufficient disk space for storing these log files.	d:\had
Data Directory	HDP data will be stored in this directory on each cluster node. You can add multiple comma-separated data locations for multiple data directories.	d:\hdp
Enable LZO codec	Use LZO compression for HDP.	Selected
Use Tez in Hive	Install Tez on the Hive host.	Selected
NameNode Host	The FQDN for the cluster node that will run the NameNode master service.	NAMENO
Secondary NameNode Host[a]	The FQDN for the cluster node that will run the Secondary NameNode master service.	SECOND
ResourceManager Host	The FQDN for the cluster node that will run the YARN Resource Manager master service.	RESOUR
Hive Server Host	The FQDN for the cluster node that will run the Hive Server master service.	HIVE_S
Oozie Server Host	The FQDN for the cluster node that will run the Oozie Server master service.	OOZIE_
WebHcat Host	The FQDN for the cluster node that will run the WebHCat master service.	WEBHCA
Slave hosts	A comma-separated list of FQDN for those cluster nodes that will run the DataNode and TaskTracker services.	slave1
Clients Hosts	A comma-separated list of FQDN for those cluster nodes that will store JARs and other job related files.	client
ZooKeeper hosts	A comma-separated list of FQDN for those cluster nodes that will run the ZooKeeper hosts.	ZOOKEE

[a]Not applicable with HA.

4. Click install optional components, and complete the following fields:

Configuration Property Name	Description	Exampl
Install Phoenix	Installs Phoenix on the HBase Server.	Selected
Install Knox	Installs Knox Gateway.	Selected
Knox Master secret	Enter the password for starting and stopping the gateway.	knox-s
HBase Master host	The FQDN for the cluster node that will run the HBase master.	HBASE-
Falcon host	The FQDN for the cluster node that will run Falcon.	falcon

Configuration Property Name	Description	Exampl
Storm nimbus host	The FQDN for the cluster node that will run the Storm Nimbus master service.	storm-
Knox host	The FQDN for the cluster node that will run Knox.	knox-h
Flume hosts	A comma-separated list of FQDN for those cluster nodes that will run the Flume service.	FLUME_ FLUME_
HBase Region Server hosts	A comma-separated list of FQDN for those cluster nodes that will run the HBase Region Server services.	slave1
Hive DB Name	Database for Hive metastore. If using SQL Server, ensure that you create the database on the SQL Server instance.	hivedb
Storm supervisor hosts	A comma-separated list of FQDN for those cluster nodes that will run the Storm Supervisors.	storm-

5. Enter the database information for Hive and Oozie as follows:

Configuration Property Name	Description	Exampl
Hive DB Username	User account credentials for Hive metastore database instance. Ensure that this user account has appropriate permissions.	hive_u
Hive DB Password		hive_p
Oozie DB Name	Database for Oozie metastore. If using SQL Server, ensure that you create the database on the SQL Server instance.	oozied
Oozie DB Username	User account credentials for Oozie metastore database instance. Ensure that this user account has appropriate permissions.	oozie_
Oozie DB Password		oozie_
DB Flavor	Database type for Hive and Oozie metastores (allowed databases are SQL Server and Derby). To use default embedded Derby instance, set the value of this property to `derby`. To use an existing SQL Server instance as the metastore DB, set the value as `mssql`.	mssql
Database Hostname	FQDN for the node where the metastore database service is installed. If using SQL Server, set the value to your SQL Server hostname. If using Derby for Hive metastore, set the value to HIVE_SERVER_HOST.	sqlser
Database port	This is an optional property required only if you are using SQL Server for Hive and Oozie metastores. By default, the database port is set to `1433`.	1433

6. To ensure that a multi-node cluster remains available, you should configure and enable High Availability. Configuring High Availability includes defining the locations and names of hosts in a cluster that are available to act as JournalNodes and the Resource Manager along with specifying a standby NameNode to fall back on in the event that the primary NameNode fails.

   To configure NameNode High Availability, select the **Enable Namenode HA** check box, then enter values in the following fields:

   High Availability Configuration Values for MSI Installer

Property	Description	Example Value
Enable HA	Whether to deploy a highly available NameNode or not.	Selected
NN Journal Node Hosts	A comma-separated list of FQDN for those cluster nodes that will run the JournalNode processes.	`journalnode1.acme.com`, `journalnode2.acme.com`, `journalnode3.acme.com`

Property	Description	Example Value
NN HA Cluster Name	This name is used for both configuration and authority component of absolute HDFS paths in the cluster.	`hdp2-ha-acme.com`
NN Journal Node Edits Directory	This is the absolute path on the JournalNode machines where the edits and other local state used by the JournalNodes (JNs) are stored. You can only use a single path for this configuration.	`d:\hadoop\journal`
NN Standby Namenode Host	The host for the standby NameNode.	STANDBY_NAMENODE.acme.com
RM Cluster Name	Logical name for the HA Resource Manager cluster.	`rmha-cluster`
RM Standby Host	The host for the standby Resource Manager.	STANDBY-resourcemgr.acme.com

Note

To Enable High Availability, you must run several commands while starting cluster services.

7. To continue with the GUI installation process, select **Install**.

Note

If you make a configuration mistake and want to clear fields, select **Reset** to clear all fields and begin again.

8. To export your HDP Setup configuration as a cluster properties text file and switch to the CLI installation process, select **Export**.

Note

Selecting Export stops the GUI installation process and produces the `clusterproperties.txt` file based on your GUI fields. Verify that all information in the fields are accurate before proceeding.

2.3. Manually Creating a Cluster Properties File

Use the following instructions to manually configure the cluster properties file for deploying HDP from the command-line or in a script:

1. Create a `clusterproperties.txt` file or use the sample clusterproperties.txt file extracted from the HDP Installation zip file.

2. Add the properties to the `clusterproperties.txt` file as described in the table below:

Important

- All properties in the `clusterproperties.txt` file must be separated by a newline character.

- Directory paths cannot contain whitespace characters.

For example, `C:\Program Files\Hadoop` is an invalid directory path for HDP.

- Use Fully Qualified Domain Names (FQDN) for specifying the network host name for each cluster host. The FQDN is a DNS name that uniquely identifies the computer on the network. By default, it is a concatenation of the host name, the primary DNS suffix, and a period.
- When specifying the host lists in the `clusterproperties.txt` file, if the hosts are multi-homed or have multiple NIC cards, make sure that each name or IP address by which you specify the hosts is the preferred name or IP address by which the hosts can communicate among themselves. In other words, these should be the addresses used internal to the cluster, not those used for addressing cluster nodes from outside the cluster.
- To Enable NameNode HA, you must include the HA properties and exclude the SECONDARY_NAMENODE_HOST definition.

Configuration Values for MSI Installer

Configuration Property Name	Description	Exampl
HDP_LOG_DIR	HDP's operational logs are written to this directory on each cluster host. Ensure that you have sufficient disk space for storing these log files.	`d:\had`
HDP_DATA_DIR	HDP data will be stored in this directory on each cluster node. You can add multiple comma-separated data locations for multiple data directories.	`d:\hdp`
NAMENODE_HOST	The FQDN for the cluster node that will run the NameNode master service.	`NAMENO`
SECONDARY_NAMENODE_HOST	The FQDN for the cluster node that will run the Secondary NameNode master service.	`SECOND`
RESOURCEMANAGER_HOST	The FQDN for the cluster node that will run the YARN Resource Manager master service.	`RESOUR`
HIVE_SERVER_HOST	The FQDN for the cluster node that will run the Hive Server master service.	`HIVE-S`
OOZIE_SERVER_HOST	The FQDN for the cluster node that will run the Oozie Server master service.	`OOZIE-`
WEBHCAT_HOST	The FQDN for the cluster node that will run the WebHCat master service.	`WEBHCA`
FLUME_HOSTS	A comma-separated list of FQDN for those cluster nodes that will run the Flume service.	`FLUME-` `FLUME-`
HBASE_MASTER	The FQDN for the cluster node that will run the HBase master.	`HBASE-`
HBASE_REGIONSERVERS	A comma-separated list of FQDN for those cluster nodes that will run the HBase Region Server services.	`slave1`
SLAVE_HOSTS	A comma-separated list of FQDN for those cluster nodes that will run the DataNode and TaskTracker services.	`slave1`
ZOOKEEPER_HOSTS	A comma-separated list of FQDN for those cluster nodes that will run the ZooKeeper hosts.	`ZOOKEE`
FALCON_HOSTS	A comma-separated list of FQDN for those cluster nodes that will run the Falcon hosts.	`falcon`
KNOX_HOST	The FQDN of the Knox Gateway host.	`KNOX-H`
IS_TEZ	Install the Tez component on Hive host.	`YES or`

Configuration Property Name	Description	Example
IS_PHOENIX	Installs Phoenix on the HBase hosts.	YES or
ENABLE_LZO	Enables the LZO codec for compression in HBase cells.	YES or
DB_FLAVOR	Database type for Hive and Oozie metastores (allowed databases are SQL Server and Derby). To use default embedded Derby instance, set the value of this property to `derby`. To use an existing SQL Server instance as the metastore DB, set the value as `mssql`.	`mssql`
DB_HOSTNAME	FQDN for the node where the metastore database service is installed. If using SQL Server, set the value to your SQL Server hostname. If using Derby for Hive metastore, set the value to HIVE_SERVER_HOST.	`sqlser`
DB_PORT	This is an optional property required only if you are using SQL Server for Hive and Oozie metastores. By default, the database port is set to `1433`.	1433
HIVE_DB_NAME	Database for Hive metastore. If using SQL Server, ensure that you create the database on the SQL Server instance.	`hivedb`
HIVE_DB_USERNAME	User account credentials for Hive metastore database instance. Ensure that this user account has appropriate permissions.	`hive_u`
HIVE_DB_PASSWORD		`hive_p`
OOZIE_DB_NAME	Database for Oozie metastore. If using SQL Server, ensure that you create the database on the SQL Server instance.	`oozied`
OOZIE_DB_USERNAME	User account credentials for Oozie metastore database instance. Ensure that this user account has appropriate permissions.	`oozie_`
OOZIE_DB_PASSWORD		`oozie_`

The following snapshot illustrates a sample `clusterproperties.txt` file:

```
#Log directory
HDP_LOG_DIR=c:\hadoop\logs

#Data directory
HDP_DATA_DIR=c:\hdpdata

#hosts
NAMENODE_HOST=nn-host.acme.com
SECONDARY_NAMENODE_HOST=sec-nn-host.acme.com
RESOURCEMANAGER_HOST=resourcemgr-host.acme.com
HIVE_SERVER_HOST=hive-host.acme.com
OOZIE_SERVER_HOST=oozie-host.acme.com
WEBHCAT_HOST=webhcat-host.acme.com
SLAVE_HOSTS=slave-host.acme.com,slave1-host.acme.com, slave2-host.acme.com
ZOOKEEPER_HOSTS=zookeeper-host.acme.com
CLIENT_HOSTS=client-host.acme.com,client2-host.acme.com
IS_TEZ=yes
ENABLE_LZO=yes
HBASE_MASTER=hbase-host.acme.com
HBASE_REGIONSERVERS=hbase-host.acme.com,hbase2-host.acme.com
FLUME_HOSTS=flume-host.acme.com
FALCON_HOST=falcon-host.acme.com
KNOX_HOST=knox-host.acme.com
STORM_NIMBUS=storm-host.acme.com
STORM_SUPERVISORS=stormsup-host.acme.com
IS_PHOENIX=yes

#Database host
DB_FLAVOR=DERBY
DB_HOSTNAME=hive-host.acme.com
DB_PORT=1527
```

```
#Hive properties
HIVE_DB_NAME=hive
HIVE_DB_USERNAME=hive
HIVE_DB_PASSWORD=hive3

#Oozie properties
OOZIE_DB_NAME=oozie
OOZIE_DB_USERNAME=oozie
OOZIE_DB_PASSWORD=oozie
```

2.4. Configure High Availability

To ensure that a multi-node cluster remains available, configure and enable High Availability. Configuring High Availability includes defining locations and names of hosts in a cluster that are available to act as journal nodes and a standby name node in the event that the primary namenode fails. To configure High Availability, add the following properties to `clusterproperties.txt` and set values as follows:

Configuring High Availability in a Windows-based Cluster

Property	Description	Example Value	Mandatory/Optional
HA	Whether to deploy a highly available NameNode or not.	yes or no	Optional
NN_HA_JOURNALNODE_HOSTS	A comma-separated list of FQDN for those cluster nodes that will run the JournalNode processes.	`journalnode1.acme.com, journalnode2.acme.com, journalnode3.acme.com`	Optional
NN_HA_CLUSTER_NAME	This name is used for both configuration and authority component of absolute HDFS paths in the cluster.	`hdp2-ha`	Optional
NN_HA_JOURNALNODE_EDITS_DIR	The path on the JournalNode machines where the edits and other local state used by the JournalNodes (JNs) are stored. You can only use a single path for this configuration.	`d:\hadoop\journal`	Optional
NN_HA_STANDBY_NAMENODE_HOST	The host for the standby NameNode.	STANDBY_NAMENODE.acme.com	Optional
RM_HA_CLUSTER_NAME	A logical name for the Resource Manager cluster.	`HA Resource Manager`	Optional
RM_HA_STANDBY_RESOURCEMANAGER_HOST	The FQDN of the standby resource manager host.	`rm-standby-host.acme.com`	Optional

To Enable High Availability, you must run several commands while starting cluster services.

3. Quick Start Guide for Single Node HDP Installation

Use the following instructions to deploy HDP on a single node Windows Server machine:

1. On the host, complete all the prerequisites, see the following sections in Getting Ready to Install:
   - Supported operating system
   - Dependent software and environment variable settings, including Java, .Net, and Python
   - Open ports required for HDP operation

   Note

   Before installation you must set an environment variable for JAVA_HOME. Do not install Java in a location that has spaces in the path name.

2. Prepare the single node machine.
   a. Configure firewall.

      HDP uses multiple ports for communication with clients and between service components.

      If your corporate policies require maintaining per server firewall, you must enable the ports listed here. Use the following command to open these ports:

      ```
      netsh advfirewall firewall add rule name=AllowRPCCommunication dir=in
       action=allow protocol=TCP localport=$PORT_NUMBER
      ```

      - For example, the following command will open up port 80 in the active Windows Firewall:

        ```
        netsh advfirewall firewall add rule name=AllowRPCCommunication dir=in
         action=allow protocol=TCP localport=80
        ```

      - For example, the following command will open ports all ports from 49152 to 65535. in the active Windows Firewall:

        ```
        netsh advfirewall firewall add rule name=AllowRPCCommunication dir=in
         action=allow protocol=TCP localport=49152-65535
        ```

      If your networks security policies allow you open all the ports, use the following instructions to disable Windows Firewall: http://technet.microsoft.com/en-us/library/cc766337(v=ws.10).aspx

3. Install and start HDP.
   a. Download the HDP for Windows MSI file from: https://public-repo-1.hortonworks.com/HDP-Win/2.1/2.1.10.0/hdp-2.1.10.0.zip .

b. Open a command prompt as Administrator:

```
runas /user:administrator "cmd /C msiexec /lv c:\hdplog.txt /
i $PATH_to_MSI_file  MSIUSEREALADMINDETECTION=1"
```

c. Run the MSI installer command. If you are installing on Windows Server 2012, use this method to open the installer:

where the `$PATH_to_MSI_file` parameter should be modified to match the location of the downloaded MSI file.

The following example illustrates the command to launch the installer:

```
runas /user:administrator "cmd /C msiexec /lv c:\hdplog.txt /i C:\
MSI_INSTALL\hdp-2.1.10.0.winpkg.msi  MSIUSEREALADMINDETECTION=1"
```

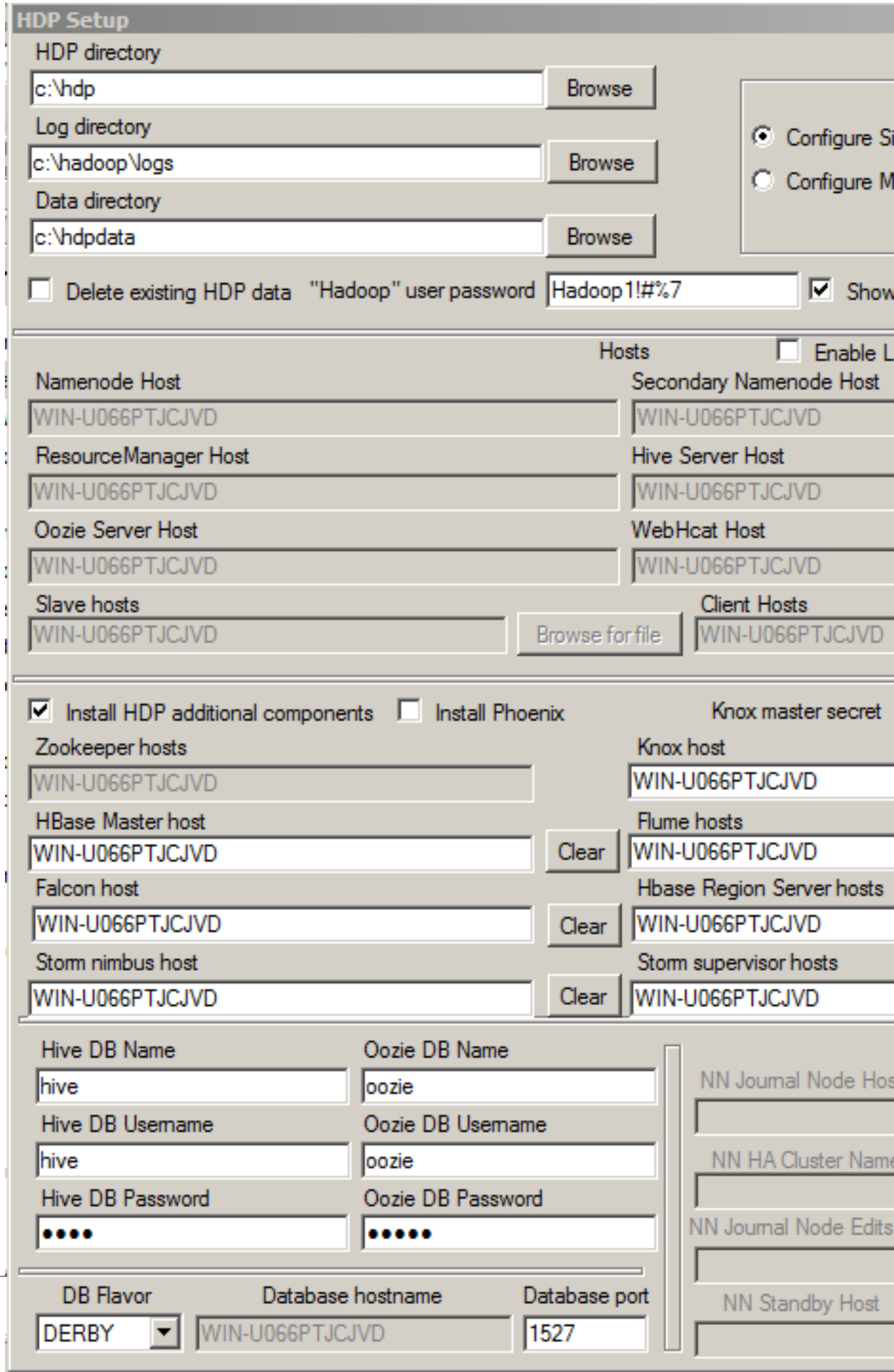
d. The HDP Setup window appears pre-populated with the host name of the server, as well as default installation parameters.

You must specify the following parameters:

- **Hadoop User Password**: Enter that password for the Hadoop super user (the administrative user). This password enables you to log in as the administrative user and perform administrative actions. Password requirements are controlled by Windows, and typically require that the password include a combination of uppercase and lowercase letters, digits, and special characters.
- **Hive and Oozie DB Names, Usernames, and Passwords**: Set the DB (database) name, user name, and password for the Hive and Oozie metastores. You can use the boxes at the lower left of the HDP Setup window ("Hive DB Name", "Hive DB Username", etc.) to specify these parameters.
- **DB Flavor**: Select DERBY to use an embedded database for the single-node HDP installation.

You can optionally configure the following parameters (for a detailed description of each option, see Defining Cluster Properties:

- **HDP Directory**: The directory in which HDP will be installed. The default installation directory is `c:\hdp`.
- **Log Directory**: The directory for the HDP service logs. The default location is `c:\hadoop\logs`.
- **Data Directory**: The directory for user data for each HDP service. The default location is `c:\hdpdata`.
- **Delete Existing HDP Data**: Selecting this check box removes any existing data from prior HDP installs. This ensures that HDFS starts with a formatted file system. For a single node installation, it is recommended that you select this option to start with a freshly formatted HDFS.
- **Install HDP Additional Components**: Select this check box to install ZooKeeper, Flume, Storm, Knox or HBase as HDP services deployed to the single node server.


HDP Setup
HDP directory
c:\hdp
Browse
Log directory
c:\hadoop\logs
Browse
Data directory
c:\hdpdata
Browse
Configure Si
Configure M
Delete existing HDP data
"Hadoop" user password
Hadoop1!#%7
Show
Hosts
Enable L
Namenode Host
WIN-U066PTJCJVD
Secondary Namenode Host
WIN-U066PTJCJVD
ResourceManager Host
WIN-U066PTJCJVD
Hive Server Host
WIN-U066PTJCJVD
Oozie Server Host
WIN-U066PTJCJVD
WebHcat Host
WIN-U066PTJCJVD
Slave hosts
WIN-U066PTJCJVD
Browse for file
Client Hosts
WIN-U066PTJCJVD
Install HDP additional components
Install Phoenix
Knox master secret
Zookeeper hosts
WIN-U066PTJCJVD
Knox host
WIN-U066PTJCJVD
HBase Master host
WIN-U066PTJCJVD
Clear
Flume hosts
WIN-U066PTJCJVD
Falcon host
WIN-U066PTJCJVD
Clear
Hbase Region Server hosts
WIN-U066PTJCJVD
Storm nimbus host
WIN-U066PTJCJVD
Clear
Storm supervisor hosts
WIN-U066PTJCJVD
Hive DB Name
hive
Oozie DB Name
oozie
Hive DB Username
hive
Oozie DB Username
oozie
Hive DB Password
Oozie DB Password
NN Journal Node Hos
NN HA Cluster Name
NN Journal Node Edits
NN Standby Host
DB Flavor
DERBY
Database hostname
WIN-U066PTJCJVD
Database port
1527

Note

When deploying HDP with the LZO compression enabled, put the following three files in the same directory as the HDP for Windows Installer (and the cluster.properties file):

- `hadoop-lzo-0.4.19.2.1.10.0-2060` from the HDP for Windows Installation zip.
- `gplcompression.dll` from the HDP for Windows Installation zip.
- `lzo2.dll` LZO compression DLL downloaded from here.

e. When you have finished setting the installation parameters, click **Install** to install HDP.

Note

The **Export** button on the HDP Setup window to exports the configuration information for use in a CLI/script-driven deployment. Clicking **Export** stops the installation and creates a clusterproperties.txt file that contains the configuration information specified in the fields on the HDP Setup window.

The HDP Setup window closes, and a progress indicator displays while the installer is running. The installation may take several minutes. Also, the time remaining estimate may be inaccurate.

A confirmation message displays when the installation is complete.

Note

If you did not select the "Delete existing HDP data"check box, and you are reinstalling Hadoop the HDFS file system must be formatted. To format the HDFS file system, open the Hadoop Command Line shortcut on the Windows desktop, then run the following command:

```
%HADOOP_HOME%\bin\hadoop namenode -format
```

f. Start all HDP services on the single machine.

In a command prompt, navigate to the HDP install directory. This is the "HDP directory" setting you specified in the HDP Setup window.

Run the following command from the HDP install directory:

```
%HADOOP_NODE%\start_local_hdp_services
```

g. Validate the install by running the full suite of smoke tests:

i. Create a smoketest user directory in HDFS:

```
%HADOOP_HOME%\bin\hadoop -mkdir -p /user/smoketest
%HADOOP_HOME%\bin\hadoop dfs -chown -R smoketest
```

ii. Run the provided smoke tests as the hadoop user or create a smoketest user in HDFS:

```
runas /user:hadoop "cmd /K %HADOOP_HOME%\Run-SmokeTests.cmd"
```

iii. Run as the smoketest user to verify that the HDP services work as expected:

```
runas /user:smoketest "cmd /K %HADOOP_HOME%\Run-SmokeTests.cmd"
```

4. Deploying Multi-node HDP Cluster

This section explains the different options you can use when deploying a multinode Hadoop Cluster from the command line or in a script. When installing from the command line, the Hadoop setup script parses the Cluster Properties file and determines which services to install based on the system hostname where the installation is running.

Use one the following option to deploy HDP:

- Option I: Central push install using corporate standard procedures
- Option II: Central push install using provided script
- Option III: Manual Install one node at a time

4.1. HDP MSI Installer Properties

This section explains the HDP MSI installer command line options and HDP public properties to use when installing a multi-node Hadoop Cluster.

The format of the HDP MSI Installer command is:

```
msiexec /qn /lv "log_file" /i "msi_file" MSIUSEREALADMINDETECTION=1 HDP_DIR=
"install_dir" HDP_LAYOUT="cluster_properties_file" HDP_USER_PASSWORD=
"password" DESTROY_DATA="YES_OR_NO" HDP="YES_OR_NO" FLUME="YES_or_NO" HBASE=
"YES_or_NO" KNOX="YES_or_NO" KNOX_MASTER_SECRET="secret" FALCON="YES_or_NO"
 STORM="YES_or_NO"
```

where:

- `msiexec /qn /lv "log_file" /i "msi_file MSIUSEREALADMINDETECTION=1"` is the standard installer options recommended by Hortonworks:
  - `/qn` (quiet, no UI) suppresses the HDP Setup Window. Use `/qb` (quiet basic) to suppress the HDP Setup and show a progress bar.
  - `/lv "log_file"` (log verbose) creates a verbose installation log with the name you specified; if only a file name is provided it is created in the directory where the msiexec was launched.
  - `/i "msi_file"` points to the HDP Installer file, we recommend specifying the absolute path.
  - `MSIUSEREALADMINDETECTION=1` ensures that the user running the installer has true administrator permissions.

  For more information on standard msiexec options, enter `msiexec /?` in a command prompt.
- HDP public properties, that is everything following the last option `/i "msi_file"` in the command line format above, are described in the following table. These public properties are passed by msiexec to the Hadoop setup script:

Table 4.1. HDP Public Properties

Property	Value	Value Defined in Cluster Properties file	Description
DESTROY_DATA	Yes or No	none	Yes removes previous HDP data and formats the NameNode. No leaves the previous data and does not format the NameNode.
HDP_USER_PASSWORD	*password*	none	Password defined when creating the Hadoop user. Note that if the password does not meet your password policy standards the installation will fail.
HDP_DIR	*install_dir*	none	Absolute path to the Hadoop root directory where HDP components are installed.
HDP_LAYOUT	*clusterproperties_full_path*	none	Defines the absolute path to the Cluster Properties file. Note that relative paths are not supported and the path may not contain spaces. For example, `C:\MSI_Install\clusterproperties.txt` .
HDP	Yes or No	ZOOKEEPER_HOSTS	Setting this to Yes instructs the MSI to install the optional HDP components, such as Flume, HBase, Knox, Falcon and Storm. When enabled, you must specify the components on the commandline. For example: HDP="YES" KNOX="YES" KNOX_SECRET="secret" FALCON="NO" HBASE="YES" FLUME="NO" STORM="NO". Excluding the optional components from the command line causes the installation to fail.
FLUME	Yes or No	FLUME_HOSTS	Includes the installation of Flume components on the hosts matching the name defined in the cluster properties file.
HBASE	Yes or No	HBASE_MASTER and HBASE_REGIONSERVERS	Includes the installation of HBase components on the hosts matching the name defined in the cluster properties file.
KNOX	Yes or No	KNOX_HOST	Includes the installation of Knox Gateway on the host matching the name defined in the cluster properties file. When yes the KNOX_SECRET must also be specified as a parameter.
KNOX_MASTER_SECRET	*secret*	none	Specified only when KNOX="yes". The master secret to protect Knox security components, such as SSL certificates.
FALCON	Yes or No	FALCON_HOSTS	Includes the installation of the Falcon components on the host matching the name defined in the cluster properties file.
STORM	Yes or No	STORM_NIMBUS and STORM_SUPERVISORS	Includes the installation of the Storm components on the host matching the name defined in the cluster properties file.

For the optional HDP Components, specify the same property, such as HDP=yes FLUME=no HBASE=yes KNOX=no FALCON=no STORM=no, on the command line of every cluster node. If you are not installing any optional components specify HDP=no. Components are only installed if the hostname matches a value in the cluster properties file.

To install a basic cluster with HBase, use the following command on every node:

```
msiexec /qn /i  D:\MSI_Install\hdp-2.1.10.0.winpkg.msi /lv D:\MSI_Install\hdp.
log MSIUSEREALADMINDETECTION=1 HDP_LAYOUT=D:\MSI_Install\cluster.properties
 HDP_DIR=D:\hdp DESTROY_DATA=yes HDP_USER_PASSWORD=#TestUser123! HDP=yes KNOX=
no FALCON=no STORM=no HBase=yes STORM=no FLUME=no
```

Important

Use the same `clusterproperties.txt` file on every node in the cluster. When LZO compression is enabled you must also put the following three files i the same directory:

- `hadoop-lzo-0.4.19.2.1.10.0-2296` from the HDP for Windows Installation zip.
- `gplcompression.dll` from the HDP for Windows Installation zip.
- `lzo2.dll` LZO compression DLL downloaded from here.

4.2. Option I - Central Push Install Using A Deployment Service

Many Windows Data Centers have standard corporate procedures for performing centralized push-install of software packages to hundreds or thousands of computers at the same time. In general, these same procedures also allow a centralized push-install of HDP to a Hadoop cluster.

If your Data Center already has such procedures in place, then follow this simple checklist:

1. Identify and configure the hosts for the Hadoop cluster nodes.
2. On the host nodes, complete all the prerequisites, see the following sections in Preparing the Environment:
   - Supported operating system
   - Dependent software and environment variable settings
   - Enable Powershell Remote scripting and set cluster nodes as trusted hosts
   - Resolvable host names, and static IPv4 addresses
   - Open ports required for HDP operation

Note

Before installation you must set an environment variable for JAVA_HOME. Do not install Java in a location that has spaces in the path name.

3. Download the HDP Windows Installation package from here, which includes a sample `clusterproperties.txt` file, hadoop-lzo jar, and gplcompression dll file.
4. Create Cluster Properties file using your host information, see Define Cluster Properties.

Important

Nodes in the cluster communicate with each other using the host name or IP address defined in the cluster properties file. For multi-homed systems

and systems with more than one NIC, ensure that the preferred name or IP address is specified in the Cluster Properties file.

5. Using your standard procedures to push both the HDP Installer MSI and the custom `clusterproperties.txt` file to each node in the cluster.

6. Continuing to use your standard procedures to remotely execute the installation with the `msiexec` command documented in section Understanding the HDP MSI Installer Properties.

Note

The HDP Installer unpacks the MSI contents to `%SystemDrive%\HadoopInstallFiles`. A detailed installation log is located at `%SystemDrive%\HadoopInstallFiles\HadoopSetupTools\hdp-2.1.10.0.winpkg.install`. This folder is required to uninstall HDP, do not remove it.

7. Examine the return results and/or logs from your standard procedures to ensure that all nodes were successfully installed.

After the installation completes, you must configure and start the Hadoop services.

4.3. Option II - Central HDP Install Using the Push Install HDP Script

Hortonworks provides a powershell script (`push_install_hdp.ps1` included in the resources directory of the installer zip) that installs HDP one system at a time on all hosts defined in the cluster properties file. Use this script to deploy HDP to a small test clusters. The script does not require shared storage, it copies the installation files to the target using the Windows Administrative Share. Ensure that the Admin Share is enabled on all cluster hosts, and that the Administrator account executing the script has the privileges to write to the cluster hosts.

To use the Push Install HDP script:

1. On the host nodes, complete all the prerequisites. In Preparing the Environment see the following sections :

   - Supported operating system
   - Dependent software and environment variable settings
   - Enable Powershell Remote scripting and set cluster nodes as trusted hosts
   - Resolvable host names, and static IPv4 addresses
   - Open ports required for HDP operation

Note

Before installation you must set an environment variable for JAVA_HOME. Do not install Java in a location that has spaces in the path name.

2. Additionally on each host node you must also do the following:

   - Enable the Administrative Share:

     ```
     netsh firewall set service type remoteadmin enabled
     ```

   - Create the a target directory to which the installer can copy the files used for the installation:

     ```
     mkdir D:\MSI_Install
     ```

3. Download the HDP Windows Installation package from here, which includes a sample `clusterproperties.txt` file, hadoop-lzo jar, and gplcompression dll file.

4. Define your Cluster Properties and save to a file, see

Important

Nodes in the cluster communicate with each other using the host name or IP address defined in the cluster properties file. For multi-homed systems and systems with more than one NIC, ensure that the preferred name or IP address is specified in the Cluster Properties file.

5. Copy the HDP MSI Installer, your custom `clusterproperties.txt`, and the `push_install_hdp.ps1` files to the source directory on the master install node (the host from which you are running the push install).

6. Determine the MSI command line parameters, see Understanding the HDP MSI Installer Properties

7. On the master install node, open a command prompt with run as Administrator, and enter following:

   ```
   cd source_path
   powershell -File push_install_hdp.ps1 source_path  destination_path
     clusterproperties_file  files_list skip msiexec_command -parallel
   ```

   where:

   - *source_path*: Absolute path to the installation files. This directory must contain the HDP MSI and the `clusterproperties.txt` file, as well as any other files the installer will push to the cluster nodes. For example , `D:\MSI_Install`.

   - *destination_path*: Absolute path to an existing directory on the target cluster nodes. All nodes must have this directory. The installer copies the *files_list* from the *source_path* to the *destination_path*. This path is specified as a local path on the target host. For example, `D:\MSI_Install`.

   - *clusterproperties_file*: Name of the custom Cluster Properties file. For example, `clusterproperties.txt`. (Do NOT include the path to the file.)

- *files_list*: Comma-delimited list of file names that the installer copies from the *$source_path* to all cluster hosts. The list must contain both the Cluster Property and HDP Installer file names. For example, `hdp-2.1.10.0.winpkg.msi,cluster.properties`. The list can not contain spaces. Ensure that all the listed files are in the *$source_path*.

Tip

When deploying HDP with the LZO compression enabled, put the following three files in the same directory as the HDP for Windows Installer (and the cluster.properties file) and include them in the file list:

- `hadoop-lzo-0.4.19.2.1.10.0-2296` from the HDP for Windows Installation zip.
- `gplcompression.dll` from the HDP for Windows Installation zip.
- `lzo2.dll` LZO compression DLL downloaded from here.

- *msiexec_command*: Complete installation command that the script executes on the target nodes, see Understanding the HDP MSI Installer Properties.

The installer script will return error messages or successful completion results to the Install Master host. These messages will be printed out at the end of the script execution. Examine these return results to ensure that all nodes were successfully installed.

Note

On each node, the HDP Installer unpacks the MSI contents to `%SystemDrive%\HadoopInstallFiles`. A detailed installation log is located at `%SystemDrive%\HadoopInstallFiles\HadoopSetupTools\hdp-2.1.10.0.winpkg.install`. This folder is required to uninstall HDP, do not remove it.

4.4. Option III - Installing HDP from the Command-line

Note

Before installation you must set an environment variable for `JAVA_HOME`. Do not install Java in a location that has spaces in the path name.

Use the following instructions to install a single Hadoop Cluster node from the command line using a Cluster Properties file:

1. On the host nodes, complete all the prerequisites. In Preparing the Environment see the following sections :
   - Supported operating system
   - Dependent software and environment variable settings

- Enable Powershell Remote scripting and set cluster nodes as trusted hosts
- Resolvable host names, and static IPv4 addresses
- Open ports required for HDP operation

2. Download the HDP Windows Installation package from here, which includes a sample `clusterproperties.txt` file, hadoop-lzo jar, and gplcompression dll file.

3. Optionally, download the LZO compression DLL from here.

4. Create a Cluster Properties file using your host information, see Define Cluster Properties.

Important

Nodes in the cluster communicate with each other using the host name or IP address defined in the cluster properties file. For multi-homed systems (systems that can be access internally and externally) and systems with more than one NIC, ensure that the preferred name or IP address is specified in the Cluster Properties file.

5. Place the MSI and custom `clusterproperties.txt` file in a local subdirectory on the host. Only the Hadoop Services that match the system's hostname in the cluster properties file will get installed.

6. **(Optional)**: When installing HDP with HDFS compression enabled, put the following three files in the same directory as the HDP for Windows Installer and the `cluster.properties` file:

   - hadoop-lzo-0.4.19.2.1.10.0-2296 from the HDP for Windows Installation zip.
   - gplcompression.dll from the HDP for Windows Installation zip.
   - `lzo2.dll` download from here.

   Open a command prompt with the runas Administrator option, and execute the following command:

```
msiexec /qn /i  "msi_file_name" /lv "log_file_name"
 MSIUSEREALADMINDETECTION=1 HDP_USER_PASSWORD="Password" HDP_LAYOUT=
"cluster_properties_file" HDP_DIR="install_dir" DESTROY_DATA=yes HDP=yes
 KNOX=no FALCON=no STORM=no
```

   See Understanding the HDP MSI Installer Properties for a detailed description of the command line options.

The HDP Installer unpacks the MSI contents to `%SystemDrive%\HadoopInstallFiles`. A detailed installation log is located at `%SystemDrive%\HadoopInstallFiles\HadoopSetupTools\hdp-2.1.10.0.winpkg.install`. This folder is required to uninstall HDP, do not remove it.

Note

If you did not select the "Delete existing HDP data"check box, and you are reinstalling Hadoop the HDFS file system must be formatted.

To format the HDFS file system, open the Hadoop Command Line shortcut on the Windows desktop, then run the following command:

```
runas /user:hadoop "cmd /K %HADOOP_HOME%\bin\hadoop namenode -
format"
```

5. Configure HDP Components and Services

After installing HDP components, you must update the following component settings or install additional software:

Use one of the following methods to modify the cluster properties file:

- Enabling HDP Services
- Configure Hive when Metastore DB is in a Named Instance (MS SQL Only)
- Configure MapReduce on HDFS
- Configure MapReduce on HDFS
- Configure HBase on HDFS
- Configure Hive on HDFS
- Set up Tez for Hive
- Configure Garbage Collector for NameNode
- Configure (Optional) Install Microsoft SQL Server JDBC Driver
- Starting HDP Services

5.1. Enabling HDP Services

By default the following HDP services are disabled, to allow these services to start and stop using the Start Local or Remote HDP script you must enable them:

- Apache Hadoop `falcon`
- Apache Hadoop `flumeagent`
- Apache Hadoop `rest`
- Apache Hadoop `thrift` or Apache Hadoop `thrift2`

Once enabled, these services will start and stop using the Start Local Services or Start Remote Services scripts.

1. Enable Thrift on a cluster node:

```
sc config thrift start= demand
```

Note

In test environments you may want to enable thrift2 instead. HDP 2.1 includes both Thrift and Thrift2, these services use the same port and cannot

run at the same time. Currently, Thrift2 is Alpha software and should only be used in test environments.

2. Enable Falcon:

```
sc config falcon start= demand
```

3. Enable the Flume Agent:

```
sc config flumeagent start= demand
```

4. (Optional) To allow access to the cluster through the Knox Gateway, enable Rest on a cluster node:

```
sc config rest start= demand
```

5.2. Configure Hive when Metastore DB is in a Named Instance (MS SQL Only)

When using MS SQL for the Hive metadata store and the Hive database is not in the default instance (that is it is in a named instance), you must configure the connection string after the installation completes as follows:

1. On the Hive host, open the hive-site.xml in a text editor.

2. Add the instance name to the property of the connection URL:

```
<property>
  <name>javax.jdo.option.ConnectionURL</name>
  <value>jdbc:sqlserver://$sql-host/$instance-name:port/$hive_db;create=
true</value>
  <description>JDBC connect string for a JDBC metastore</description>
</property>
```

   where the value contains the following environment specific information:

   - *$sql-host*: SQL server host name
   - *$instance-name*: the name of the instance that the Hive database is in
   - *$hive_db*: the name of the Hive database

3. Save the changes to `hive-site.xml`.

4. Finish configuring Hive as described in the following section before restarting the Apache Hadoop Hive service.

5.3. Configure MapReduce on HDFS

To use MapReduce, in HDFS make the MapReduce history folder, tmp, application logs, and a Yarn folders and then set permissions to the folders.

```
%HADOOP_HOME%\bin\hadoop.cmd dfs -mkdir -p /mapred/history/done /mapred/
history/done_intermediate
%HADOOP_HOME%\bin\hadoop.cmd dfs -chmod -R 1777 /mapred/history/
done_intermediate
%HADOOP_HOME%\bin\hadoop.cmd dfs -chmod 770 /mapred/history/done
%HADOOP_HOME%\bin\hadoop.cmd dfs -chown -R hadoop:hadoopUsers /mapred
%HADOOP_HOME%\bin\hadoop.cmd dfs -chmod 755 /mapred /mapred/history
%HADOOP_HOME%\bin\hadoop.cmd dfs -mkdir /tmp
%HADOOP_HOME%\bin\hadoop.cmd dfs -chmod 777 /tmp
%HADOOP_HOME%\bin\hadoop.cmd dfs -mkdir /app-logs
%HADOOP_HOME%\bin\hadoop.cmd dfs -chown hadoop:hadoopUsers /app-logs
%HADOOP_HOME%\bin\hadoop.cmd dfs -chmod 1777 /app-logs
%HADOOP_HOME%\bin\hadoop.cmd dfs -mkdir -p /yarn /yarn/generic-history/
%HADOOP_HOME%\bin\hadoop.cmd dfs -chmod -R 700 /yarn
%HADOOP_HOME%\bin\hadoop.cmd dfs -chown -R hadoop:hadoop /yarn
```

5.4. Configure HBase on HDFS

To use HBase make a HBase user and application data folder and then set permissions on the folders.

```
%HADOOP_HOME%\bin\hadoop.cmd dfs -mkdir -p /apps/hbase/data
%HADOOP_HOME%\bin\hadoop.cmd dfs -chown hadoop:hadoop /apps/hbase/data
%HADOOP_HOME%\bin\hadoop.cmd dfs -chown hadoop:hadoop /apps/hbase/data/..
%HADOOP_HOME%\bin\hadoop.cmd dfs -mkdir -p /user/hbase
%HADOOP_HOME%\bin\hadoop.cmd dfs -chown hadoop:hadoop /user/hbase
```

5.5. Configure Hive on HDFS

To use Hive, in HDFS create the Hive warehouse directory, the Hive and WebHcat user directories directory, and the WebHCat application folder. And then set permissions on the directory to allow all users access:

1. Open the command prompt with the Hadoop user account:

   ```
   runas /user:hadoop cmd
   ```

2. Make a user directory for hive and the hive warehouse directory as follows:

   ```
   %HADOOP_HOME%\bin\hadoop.cmd dfs -mkdir -p /user/hive /hive/warehouse
   ```

3. Make a user and application directory for WebHcat as follows:

   ```
   %HADOOP_HOME%\bin\hadoop.cmd dfs -mkdir -p /user/hcat
   %HADOOP_HOME%\bin\hadoop.cmd dfs -mkdir -p /apps/webhcat
   ```

4. Change the owner and permissions as follows:

   ```
   %HADOOP_HOME%\bin\hadoop.cmd dfs -chown hadoop:hadoop /user/hive
   %HADOOP_HOME%\bin\hadoop.cmd dfs -chmod -R 755 /user/hive
   %HADOOP_HOME%\bin\hadoop.cmd dfs -chown -R hadoop:users  /hive/warehouse
   %HADOOP_HOME%\bin\hadoop.cmd dfs -chown -R hadoop:hadoop /user/hcat
   %HADOOP_HOME%\bin\hadoop.cmd dfs -chmod -R 777 /hive/warehouse
   %HADOOP_HOME%\bin\hadoop.cmd dfs -chown -R hadoop:users /apps/webhcat
   %HADOOP_HOME%\bin\hadoop.cmd dfs -chmod -R 755 /apps/webhcat
   ```

5.6. Set up Tez for Hive

If your installation specified to use Tez for Hive, in the `cluster.properties` IS_TEZ=yes, after deployment perform the following steps as the hadoop user "hadoop":

1. Open the command prompt with the hadoop account:

```
runas /user:hadoop cmd
```

2. Make a Tez application directory in HDFS:

```
%HADOOP_HOME%\bin\hadoop.cmd fs -mkdir /apps/tez
```

3. Allow all users read and write access:

```
%HADOOP_HOME%\bin\hadoop.cmd fs -chmod -R 755 /apps/tez
```

4. Change the owner of the file to hadoop:

```
%HADOOP_HOME%\bin\hadoop.cmd fs -chown -R hadoop:users /apps/tez
```

5. Copy the Tez home directory on the local machine into the HDFS `/apps/tez` directory:

```
%HADOOP_HOME%\bin\hadoop.cmd fs -put %TEZ_HOME%\* /apps/tez
```

6. Remove the Tez configuration directory from the HDFS Tez application directory:

```
%HADOOP_HOME%\bin\hadoop.cmd fs -rm -r -skipTrash /apps/tez/conf
```

7. Ensure that the following properties are set in the `%HIVE_HOME%\conf\hive-site.xml`:

Table 5.1. Hive site configuration for Tez

Property	Default Value	Description
hive.auto.convert.join.noconditionaltask	true	Specifies whether Hive optimizes converting common JOIN statements into MAPJOIN statements. JOIN statements are converted if this property is enabled and the sum of size for n-1 of the tables/ partitions for an n-way join is smaller than the size specified with the hive.auto.convert.join.noconditionaltask.size property.
hive.auto.convert.join.noconditionaltask.size	10000000 (10 MB)	Specifies the size used to calculate whether Hive converts a JOIN statement into a MAPJOIN statement. The configuration property is ignored unless hive.auto.convert.join.noconditionaltask is enabled.
hive.optimize.reducededuplication.min.reducer	4	Specifies the minimum reducer parallelism threshold to meet before merging two MapReduce jobs. However, combining a mapreduce job with parallelism 100 with a mapreduce job with parallelism 1 may

Property	Default Value	Description
		negatively impact query performance even with the reduced number of jobs. The optimization is disabled if the number of reducers is less than the specified value.
hive.tez.container.size	-1	By default, Tez uses the java options from map tasks. Use this property to override that value. Assigned value must match value specified for mapreduce.map.child.java.opts.
hive.tez.java.opts	N/A	Set to the same value as `mapreduce.map.java.opts`.

Note

Adjust the settings above to your environment where appropriate; the `hive-default.xml.template` contains examples of the properties.

Verify the install succeeded by running smoke tests for tez and hive.

5.7. Configure Garbage Collector for NameNode

These steps enable logging for Garbage Collector. By default the Garbage Collector logging is disabled.

To enable GC logging on NameNode:

1. Open the Hadoop Environment script, `%HADOOP_HOME%\etc\hadoop\hadoop-env.cmd`.

2. Prepend the following text in the HADOOP_NAMENODE_OPTS definition:

   ```
   -Xloggc:%HADOOP_LOG_DIR%/gc-namenode.log -verbose:gc -XX:+PrintGCDetails -
   XX:+PrintGCTimeStamps -XX:+PrintGCDateStamps
   ```

   For example:

   ```
   set HADOOP_NAMENODE_OPTS=-Xloggc:%HADOOP_LOG_DIR%/gc-namenode.log -
   verbose:gc -XX:+PrintGCDetails -XX:+PrintGCTimeStamps -XX:+PrintGCDateStamps
    -Dhadoop.security.logger=%HADOOP_SECURITY_LOGGER% -Dhdfs.audit.logger=
   %HDFS_AUDIT_LOGGER% %HADOOP_NAMENODE_OPTS%
   ```

3. Run the following command to recreate the NameNode service XML:

   ```
   %HADOOP_HOME%\bin\hdfs.cmd --service namenode > %HADOOP_HOME%\bin\namenode.
   xml
   ```

4. Verify that the NameNode Service XML was updated.

5. Restart the NameNode service.

The NameNode start up configuration is changed to enable GC logging.

5.8. (Optional) Install Microsoft SQL Server JDBC Driver

If you are using MS SQL Server for Hive and Oozie metastores, you must install the MS SQL Server JDBC driver after installing Hive or Oozie.

1. Download the SQL JDBC JAR file sqljdbc_3.0.1301.101_enu.exe.
2. Run the downloaded file.

   (By default, the SQL JDBC driver file will be extracted at `C:\Users\Administrator\Downloads\Microsoft SQL Server JDBC Driver 3.0`.)
3. Copy and paste the `C:\Users\Administrator\Downloads\Microsoft SQL Server JDBC Driver 3.0\sqljdbc_3.0\enu\sqljdbc4.jar` file to *`$HIVE_HOME`*`/lib` (where *`$HIVE_HOME`* can be set to `D:\hadoop\hive-0.9.0`).

5.9. Starting HDP Services

1. Start HDP Services:

   a. On the Master Nodes start the local services as follows:

   ```
   %HADOOP_NODE%\start_local_hdp_services.cmd
   ```

   Wait for the Master Node services to start up before continuing.

   b. On any Master Node, start all slave node services as follows:

   ```
   %HADOOP_NODE%\start_remote_hdp_services.cmd
   ```

   c. On the Knox Gateway:

   ```
   %HADOOP_NODE%\start_local_hdp_services.cmd
   ```

2. Smoke test your installation using the instructions provided in the Validate the Install section.

6. Validate the Installation

After the HDP Cluster installation is completed, additional configuration is required before you can validate.

6.1. Run Smoke Test

After starting the HDP Services, run the smoke tests to validate the installation:

1. Create the Smoke Test user account:

```
net user /add smoketestuser myp@ssw0rd123!
```

2. Create a smoketest user directory in HDFS if one does not already exist:

```
%HADOOP_HOME%\bin\hadoop -mkdir -p /user/smoketest
%HADOOP_HOME%\bin\hadoop dfs -chown -R smoketest
```

3. On a cluster node, open a command prompt and execute the smoke test command script as shown below:

```
%HADOOP_NODE_INSTALL_ROOT%\Run-SmokeTests.cmd
```

The smoke tests validate the installed functionality by executing a set of tests for each HDP component. You must run these tests as the hadoop user or Create a User

Note

It is recommended to re-install HDP, if you see installation failures for any HDP component.

7. Upgrade HDP Manually

This document provides instructions on upgrading an HDP Windows cluster from HDP 1.3 or 2.0 to HDP 2.1. This is an "in-place" upgrade, where your user data and metadata does not need to be moved during the upgrade process but services must be stopped and re-installed. All the instructions in this section use the MS-DOS command prompt.

- Getting Ready to Upgrade
- Backing Up Critical HDFS Metadata
- Backing Up Your Configuration Files
- Stopping Running HDP 1.3 Services
- Uninstalling HDP 1.3 on All Nodes
- Preparing the HDP 2.0 Cluster Layout
- Prepare the Metastore Databases
- Installing HDP 2.0 and Maintaining Your Prior Data
- Upgrading HDFS Metadata
- Upgrading HBase
- Upgrading Oozie
- Starting HDP 2.0 Services
- Validating Your Data
- Verifying that HDP 2.0 Services are Working
- Finalize Upgrade
- Troubleshooting

Warning

These upgrade instructions only apply to HDP clusters that are not configured for high availability.

7.1. Getting Ready to Upgrade

To prepare for upgrade, you gather the following information:

- **Cluster Properties**: Save a copy of the `cluster.properties` file from the HDP installation directory, for example `c:\hdp\cluster.properties` to configure new HDP components.

- **File required for uninstall**: Confirm that the Uninstallation packages are available on each node in the cluster. The uninstallation package is in `C:\HadoopInstallFiles`. Without these packages, the Uninstaller cannot remove the HDP packages on each node.
- **HDP data directory**: Identify where user data and metadata is being stored by HDFS and MapReduce. These directories are retained during the upgrade process.

  The data directory is defined in the cluster.properties file. For example:

  ```
  #Data directory
  HDP_DATA_DIR=c:\hdp_data
  ```

- **New HDP components**: Identify the hosts for the new service components and ensure that these hosts meet all the prerequisites. The new components are Falcon, Storm, and Knox.
- **Hadoop user password**: You must run some of the upgrade and configuration steps with the hadoop user, therefore you must know the users password.

Note

If you do not know the password for the `hdfs` user, then reset the password to a known password and continue. For example, run the following command to change the password:

```
net user hdfs NewPassword123!
```

7.2. Backing up critical HDFS metadata

Back up the following critical data before attempting an upgrade. On the node that hosts the NameNode, open the **Hadoop Command line** shortcut that opens a command window in the Hadoop directory. Run the following commands:

1. Open the command prompt using the Hadoop user account and go to the HDFS home directory:

   ```
   runas /user:hdfs "cmd /K cd %HDFS_HOME%"
   ```

   Where %HDFS_HOME% is the HDFS home directory, for example `c:\hdp`.

2. Run the `fsck` command to fix any file system errors.

   ```
   hdfs fsck / -files -blocks -locations > dfs-old-fsck-1.log
   ```

   The console output is printed to the `dfs-old-fsck-1.log` file.

3. Capture the complete namespace directory tree of the file system:

   ```
   hdfs fs -lsr / > dfs-old-lsr-1.log
   ```

4. Create a list of DataNodes in the cluster:

   ```
   hdfs dfsadmin -report > dfs-old-report-1.log
   ```

5. Capture output from fsck command:

   ```
   hdfs fsck / -block -locations -files > fsck-old-report-1.log
   ```

Note

Verify there are no missing or corrupted files/replicas in the `fsck` command output.

7.2.1. Save the HDFS namespace

To save the HDFS namepsace:

1. Open the command prompt using the HDFS user account and go to the Hadoop home directory:

```
runas /user:hadoop "cmd /K cd %HADOOP_HOME%"
```

2. Place the NameNode in safe mode, to keep HDFS from accepting any new writes:

```
hdfs dfsadmin -safemode enter
```

3. Save the namespace.

```
hdfs dfsadmin -saveNamespace
```

Note

Do NOT leave safe mode from this point onwards. HDFS should not accept any new writes.

4. Finalize the namespace:

```
hdfs namenode -finalize
```

5. On the machine that hosts the NameNode, copy the following checkpoint directories into a backup directory:

```
%HADOOP_HDFS_HOME%\hdfs\nn\edits\current
%HADOOP_HDFS_HOME%\hdfs\nn\edits\image
%HADOOP_HDFS_HOME%\hdfs\nn\edits\previous.checkpoint
```

7.3. Backing Up Your Configuration Files

Copy customized configuration files from the home directories of each of the components to a backup directory. Configuration files are found in the Component Home directory, such as:

```
%HADOOP_HOME%\conf
%FLUME_HOME%\conf
%HBASE_CONF_DIR%\conf
%HCATALOG_HOME%\conf
```

7.4. Stopping Running HDP Services

Stop all running services. First stop the remote services and then the local services, by running the following commands from any node in the cluster:

```
%HADOOP_NODE_INSTALL_ROOT%\stop_remote_hdp_services.cmd
%HADOOP_NODE_INSTALL_ROOT%\stop_local_hdp_services.cmd
```

7.5. Uninstalling HDP on All Nodes

On each cluster node, uninstall the HDP for Windows using one of the following methods:

- **Command line**: Open the command prompt as an administrator and run the following command on each node in the cluster:

```
msiexec /lv hdp_uninstall.log /qb  /x c:\MSI_INSTALLER\hdp-1.3.0.0-GA.
winpkg.msi HDP_DIR=C:\hdp DESTROY_DATA=no
```

Note

Use the MSI installer file that goes with the version of HDP you are uninstalling. For example, to uninstall HDP 1.3 use the `hdp-1.3.0.0.winpkg.msi` file. If you do not have the installer uninstall from the **Program and Features** > **Uninstall Program** window.

- **Control Panel**: Open **Programs and Features**, and then right-click on **Horton Works Data Platform for Windows** and select **Uninstall**.

  This uninstall option keeps existing data in place, maintaining the data directories for HDFS and MapReduce.

7.6. Update the HDP Cluster Properties File

To keep the same metadata and user data when upgrading, use the directory settings and database settings in the cluster properties file of the version you are upgrading from for the base of the new cluster properties file.

Using the existing HDP cluster properties file, make the following changes:

- For **HDP 1.3 to HDP 2.1**:
  - Change `JOBTRACKER_HOST` to `RESOURCEMANAGER_HOST` (leave the definition the same).
  - (Optional) Add a definition for `CLIENT_HOSTS`.
  - (Optional) Add a definition for `KNOX_HOST`.
  - (Optional) Add a definition for `STORM_NIMBUS`.
  - (Optional) Add a definition for `STORM_SUPERVISORS`.
  - (Optional) Add a definition for `FALCON_HOSTS`.
  - Change `HA_NAMENODE_HOST` to `NN_HA_STANDBY_NAMENODE_HOST` if you are upgrading a High Availability cluster
  - Add a definition for `DB_PORT`. (Default port for derby is1527.)

- (Required) Add a definition for `IS_TEZ`.
- (Required) Add a definition for `IS_PHOENIX`.

- For upgrade from **HDP 2.0 to HDP 2.1**:
  - (Optional) Add a definition for `KNOX_HOST`.
  - (Optional) Add a definition for `STORM_NIMBUS`.
  - (Optional) Add a definition for `STORM_SUPERVISORS`.
  - (Optional) Add a definition for `FALCON_HOSTS`.
  - Add a definition for `DB_PORT`. (Default port for derby is1527.)
  - (Required) Add a definition for `IS_TEZ`.
  - (Required) Add a definition for `IS_PHOENIX`.
  - For High Availability clusters, update the HA properties as follows:
    - Change `HA_NAMENODE_HOST` to `NN_HA_STANDBY_NAMENODE_HOST`.
    - Change `HA_JOURNALNODE_HOSTS` to `NN_HA_JOURNALNODE_HOSTS`.
    - Change `HA_CLUSTER_NAME` to `NN_HA_CLUSTER_NAME`.
    - Change `HA_JOURNALNODE_EDITS_DIR` to `NN_HA_JOURNALNODE_EDITS_DIR`

Save the new `cluster.properties` file to use with the installer.

7.7. Installing HDP and Maintaining Your Prior Data

To Install HDP on all your nodes while maintaining your prior data:

1. Download the HDP for Windows MSI installer from here.
2. Copy the installer and the new cluster.properties file to all nodes of the cluster.
3. Run installation from command line on each node in the cluster:

```
msiexec /qb /i "c:\MSI_INSTALL\hdp-2.1.10.0.winpkg.msi" /lv "hdp.
log" HDP_LAYOUT="C:\MSI_INSTALL\cluster.properties" HDP_DIR="C:\hdp"
 HDP_USER_PASSWORD=H0rton!#%works DESTROY_DATA="no" HDP="yes"
```

Tip

New command line properties were added to support the optional HDP components, see HDP MSI Installer Properties. The following example, installs a basic cluster with HBase:

```
msiexec /qn /i  D:\MSI_Install\hdp-2.1.10.0.winpkg.msi /lv D:\
MSI_Install\hdp.log HDP_LAYOUT=D:\MSI_Install\cluster.properties
 HDP_DIR=D:\hdp DESTROY_DATA=yes HDP_USER_PASSWORD=#TestUser123!
 HDP=yes KNOX=no FALCON=no STORM=no HBase=yes STORM=no FLUME=no
```

4. Verify that you have installed HDP on all nodes of your cluster. Do NOT start any services yet.

7.8. Prepare the Metastore Databases

Hive uses a relational database to store metadata. This section assumes that you used SQL Server to store Hive metadata.

To upgrade an existing MS SQL database for Hive run the `%HIVE_HOME%\scripts\metastore\upgrade\mssql\hive-txn-schema-0.13.0.mssql.sql` on the Microsoft SQL server that contains the Hive database instance.

Note

If you used the Derby database option, you can skip this section.

If you use a new database name and set up new users, then you must add this new information into the `clusterproperties.txt` file used to upgrade to HDP 2.1.

7.9. Upgrading HDFS Metadata

To upgrade the HDFS Metadata, run the following steps on your NameNode:

1. Run the NameNode upgrade:

```
runas /user:hdfs "cmd /K hdfs namenode -upgrade"
```

2. On each DateNode, start the datanode service:

```
sc start datanode
```

3. Leave the command prompt open until the process completes. To see the status of the upgrade open a browser and connect to the NameNode on port 500070 (`http://namenode-host:50070`).

Note

The amount of time it takes to upgrade HDFS depends upon the amount of data and number of nodes in your environment. It may take only take a few minutes, but could also take an hour.

4. In the NameNode Administrative Interface verify that the number of DataNodes matches the number of DataNodes in your environment.

5. Abort the command prompt using ctl+c to end the NameNode upgrade process.

6. Start the NameNode service:

```
sc start namenode
```

7. Open a browser and connect to the NameNode on port 50070 (`http://namenode-host:50070`) and verify that SafeMode is off.

7.10. Upgrading HBase

To upgrade HBase, you must run the following commands as the `hdfs` user on both the HBase Master and the RegionServers hosts:

1. Start the ZooKeeper service:

```
sc start zkServer
```

2. Check for HFiles in V1 format. HBase 0.96.0 discontinues support for HFileV1, but HFileV1 was a common format prior to HBase 0.94. Run the following command to check if there are HFiles in V1 format:

```
%HBASE_HOME%\bin\hbase.cmd upgrade -check
```

3. Upgrade HBase.

```
%HBASE_HOME%\bin\hbase.cmd upgrade -execute
```

   You should see a completed Znode upgrade with no errors.

4. Start all rest HDP services.

```
sc start rest
```

Note

If the Apache Hadoop rest service is disabled, run the following command to enable it:

```
sc config name=rest start= demand
```

7.11. Upgrading Oozie

To upgrade Oozie, run the following commands as the `hdfs` user:

1. Run `ooziedb.cmd` as the HDFS user.

```
runas /user:hdfs "%OOZIE_HOME%\bin\ooziedb.cmd upgrade -run"
```

2. Replace your configuration after upgrading. Copy `oozie\conf` from the backup to the `oozie\conf` directory on each server and client.

3. Replace the content of `/user/hdfs/share` in HDFS. On the Oozie server host:

   a. Back up the `/user/hdfs/share folder` in HDFS and then delete it. If you have any custom files in this folder back them up separately and then add them back after the share folder is updated.

```
mkdir C:\tmp\oozie_tmp
```

```
runas /user:hdfs "cmd /c hdfs dfs -copyToLocal /user/hdfs/share    C:\tmp
\oozie_tmp\oozie_share_backup"
runas /user:hdfs "cmd /c hdfs dfs -rm -r /user/hdfs/share"
```

b. Add the latest share libs.

```
runas /user:hdfs "cmd /c hdfs dfs -copyFromLocal %OOZIE_HOME%\share /
user/hdfs/.
```

4. Start the Oozie service:

```
sc start oozie
```

7.12. Starting HDP Services

After you complete all data upgrades, you can start all HDP services from a master node as follows:

```
%HADOOP_NODE_INSTALL_ROOT%\start_local_hdp_services.cmd
%HADOOP_NODE_INSTALL_ROOT%\start_remote_hdp_services.cmd
```

For more information, see HDP services.

7.13. Setting up HDP

After starting the local and remote services, run the following commands to set up HDP:

1. Run the following command to open a command prompt with the Hadoop user in the Hadoop Home directory:

```
runas /user:hadoop "cmd /K cd %HADOOP_HOME%\bin"
```

2. Make, set ownership and permissions on the following directories:

```
hadoop.cmd dfs -mkdir -p /mapred/history/done /mapred/history/
done_intermediate
hadoop.cmd dfs -chmod -R 1777 /mapred/history/done_intermediate
hadoop.cmd dfs -chmod 770 /mapred/history/done
hadoop.cmd dfs -chown -R hadoop:hadoopUsers /mapred
hadoop.cmd dfs -chmod 755 /mapred /mapred/history
hadoop.cmd dfs -mkdir /tmp
hadoop.cmd dfs -chmod 777 /tmp
hadoop.cmd dfs -mkdir /app-logs
hadoop.cmd dfs -chown hadoop:hadoopUsers /app-logs
hadoop.cmd dfs -chmod 1777 /app-logs
hadoop.cmd dfs -mkdir -p /yarn /yarn/generic-history/
hadoop.cmd dfs -chmod -R 700 /yarn
hadoop.cmd dfs -chown -R hadoop:hadoop /yarn
hadoop.cmd dfs -mkdir -p /apps/hbase/data
hadoop.cmd dfs -chown hadoop:hadoop /apps/hbase/data
hadoop.cmd dfs -chown hadoop:hadoop /apps/hbase/data/..
hadoop.cmd dfs -mkdir -p /user/hbase
hadoop.cmd dfs -chown hadoop:hadoop /user/hbase
hadoop.cmd dfs -mkdir -p /user/hive /hive/warehouse
hadoop.cmd dfs -chown hadoop:hadoop /user/hive
hadoop.cmd dfs -chmod -R 755 /user/hive
hadoop.cmd dfs -chown -R hadoop:users /hive/warehouse
hadoop.cmd dfs -chmod -R 777 /hive/warehouse
```

```
hadoop.cmd dfs -mkdir -p /user/hcat
hadoop.cmd dfs -chown -R hadoop:hadoop /user/hcat
hadoop.cmd dfs -mkdir -p /apps/webhcat
hadoop.cmd dfs -chown -R hadoop:users /apps/webhcat
hadoop.cmd dfs -chmod -R 755 /apps/webhcat
```

Important

When using Tez for Hive, you must also Set up Tez for Hive

7.14. Validating Your Data

Verify that your data is intact by comparing the HDFS data directory tree with the HDP 1.3 or HDP 2.0 tree.

1. Run the following command to open a command prompt with the HDFS user in the Hadoop Home directory:

   ```
   runas /user:hadoop "cmd /K cd %HADOOP_HOME%\bin"
   ```

2. Run an `lsr` report on your upgraded system. Execute the following command from the Hadoop command line:

   ```
   hadoop fs -lsr / > dfs-new-lsr-1.log
   ```

3. Compare the directory listing to the older HDP directories. All old directories, files and timestamps should match. There will be some new entries in the HDP directory listing:

   - `/apps/hbase` is only in HDP and is used by HBase (new when upgrading from 1.3 to 2.1)
   - `/mapred/system/jobtracker` will have a new timestamp

4. Run a `fsck` report on your upgraded system. Execute the following command from the **Hadoop Command Line**:

   ```
   hdfs fsck / -blocks -locations -files fsck-new-report-1.log
   ```

5. Compare this fsck report to the prior to upgrade report to check the validity of your current HDFS data.

7.15. Verifying that HDP Services are Working

Run the provided smoke tests as the hdfs user or Create a Smoke Test User in HDFS and run as the Smoke Test user to verify that the HDP services work as expected:

```
runas /user:smoketestuser "cmd /K %HADOOP_NODE_INSTALL_ROOT%\Run-SmokeTests.
cmd"
```

Note

You can also verify HDP 2.0 services by running the following Desktop Shortcuts as the hadoop user or smoketest user **Hadoop Name Node status**, **HBase Master status**, and **Hadoop YARN status**.

7.16. Finalize Upgrade

When you are satisfied that HDP is successfully functioning with the data maintained from the HDP 1.3 or 2.0 cluster, finalize the upgrade.

Note

After you have finalized the upgrade, you cannot revert.

1. Run the following command to open a command prompt with the HDFS user in the HDFS Home directory:

   ```
   runas /user:hdfs "cmd /K cd %HDFS_HOME%\bin"
   ```

   Where %HDFS_HOME% is the HDFS home directory, for example `c:\hdp`.

2. To finalize the upgrade, run the following command:

   ```
   hdfs dfsadmin -finalizeUpgrade
   ```

7.17. Troubleshooting

The following command runs the smoke test:

```
%HADOOP_NODE_INSTALL_ROOT%\Run-SmokeTests.cmd
```

Note

Always run the smoke test as the smoke test user.

7.17.1. Troubleshooting HBase Services not Starting

If the HBase RegionServer and Master is not starting, it could be because a parenthesis in the Path variable caused a problem during setup of the services.

To fix this, run the following commands from an Administrator command prompt:

```
%HBASE_HOME%\bin\hbase.cmd --service master start > %HBASE_HOME%\bin\master.
xml
%HBASE_HOME%\bin\hbase.cmd --service regionserver start > %HBASE_HOME%\bin\
regionserver.xml
%HBASE_HOME%\bin\hbase.cmd --service rest > %HBASE_HOME%\bin\rest.xml
%HBASE_HOME%\bin\hbase.cmd --service thrift > %HBASE_HOME%\bin\thrift.xml
```

Then restart the HBase services.

7.17.2. Troubleshooting Flume Services not Starting

If Flume is not starting, it could be because `flumeservice.xml` is missing.

To fix this, Navigate to `%FLUME_HOME%\bin` and locate the `flumeagent.xml` file. If the file does not exist, locate `flumeservice.xml` file and rename it to `flumeagent.xml`.

After the file is renamed, go to **Windows Services** and restart the **Flume agent service**.

8. Managing HDP on Windows

This section describes how to manage HDP on Windows services.

8.1. Starting the HDP Services

The HDP on Windows installer sets up Windows services for each HDP component across the nodes in a cluster. Use the instructions given below to start HDP services from any host machine in your cluster.

Complete the following instructions as the administrative user:

1. Start the HDP cluster, by running the following command from any host in your cluster.

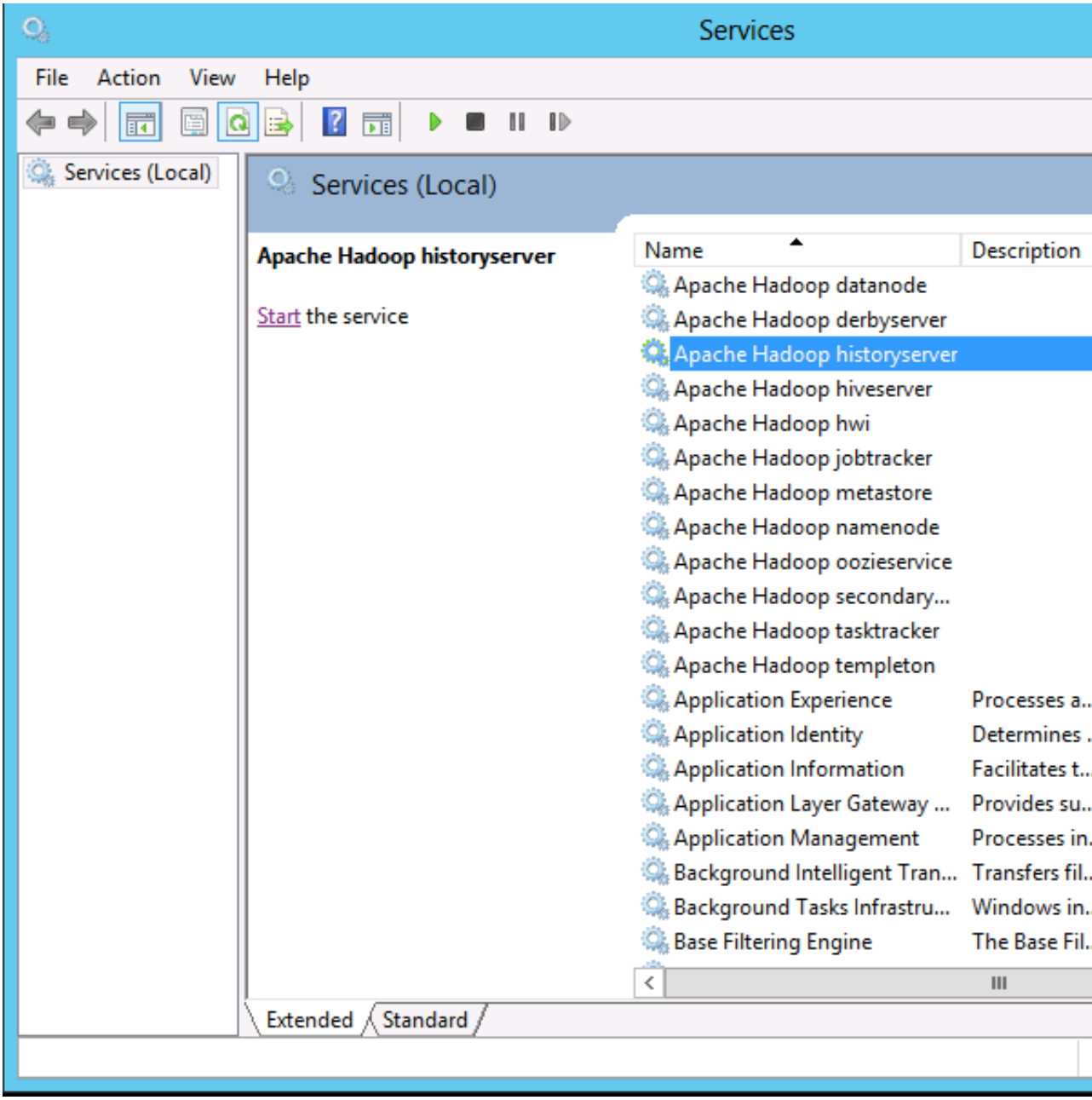
```
%HADOOP_NODE_INSTALL_ROOT%\start_remote_hdp_services.cmd
```

Important

If you want to Enable NameNode High Availability, do so while starting HDP services. Do NOT wait until all services have started.

2. Open the **Services** administration pane, **Control Panel** > **Administrative Tools** > **Services**.

   Against all the services that are installed successfully, you should see the following message as highlighted in the screenshot below:



8.2. Enabling NameNode High Availability

Optionally, to enable NameNode High Availability in a multi-node cluster, run the following commands on the primary and standby hosts while services are starting.

Important

Log in to every host and run these commands as an administrator user.

1. On the primary host, run

```
hdfs.cmd namenode -format -force
```

2. On each standby host, run

```
hdfs.cmd namenode -bootstrapStandby -force
```

```
hdfs.cmd zkfc -formatZK -force
```

8.3. Validating HA Configuration

1. Verify the state of each NameNode, using one the following methods:

   - Open the web page for each NameNode in a browser, using the configured URL.

     The HA state of the NameNode should appear in the configured address label. For example: NameNode 'example.com.8020' (standby) .

     Note

     The NameNode state may be "standby" or "active". After bootstrapping, the HA NameNode state is initially "standby".

   - Query the state of a NameNode, using JMX(tag.HAState)
   - Query the service state, using the following command:

     ```
     hdfs haadmin -getServiceState
     ```

2. Verify automatic failover.

   a. Locate the Active NameNode.

      Use the NameNode web UI to check the status for each NameNode host machine.

   b. Cause a failure on the Active NameNode host machine.

      i. Turn off automatic restart of the service.

         A. In **Windows Services** pane, locate the **Apache Hadoop NameNode** service, right-click, and choose **Properties**.

         B. On the **Recovery** tab, select **Take No Action for First, Second, and Subsequent Failures**, then choose **Apply**.

      ii. Simulate a JVM crash.

          For example, you can use the following command to simulate a JVM crash:

          ```
          'taskkill.exe /t /f /im namenode.exe'
          ```

          Alternatively, power-cycle the machine, or unplug its network interface to simulate outage.

          The Standby NameNode state should become Active within several seconds.

Note

The time required to detect a failure and trigger a failover depends on the configuration of `ha.zookeeper.session-timeout.ms` property. The default value is 5 seconds.

iii. Verify that the Standby NameNode state is Active.

A. If a standby NameNode does not activate, verify that HA settings are configured correctly.

B. Check log files for `zkfc` daemons and NameNode daemons to diagnose issues.

8.4. Stopping the HDP Services

The HDP on Windows installer sets up Windows services for each HDP component across the nodes in a cluster. Use the instructions given below to stop HDP services from any host machine in your cluster.

Complete the following instructions as the administrative user:

1. Stop the HDP cluster, by running the following command from any host machine in your cluster.

```
%HADOOP_NODE_INSTALL_ROOT%\stop_remote_hdp_services.cmd
```

9. Troubleshoot Deployment

Use the following instructions on troubleshooting installation issues encountered while deploying HDP on Windows platform:

- Collect Troubleshooting Information
- File locations, Ports, and Common HDFS Commands

9.1. Collect Troubleshooting Information

Use the following commands to collect specific information from a Windows based cluster. This data helps to isolate specific deployment issue.

1. **Collect OS information:** This data helps to determine if HDP is deployed on a supported operating system (OS).

   Execute the following commands on Powershell as an Administrator user:

   ```
   (Get-WmiObject -class Win32_OperatingSystem).Caption
   ```

   This command should provide you information about the OS for your host machine. For example,

   ```
   Microsoft Windows Server 2012 Standard
   ```

   Execute the following command to determine OS Version for your host machine:

   ```
   [System.Environment]::OSVersion.Version
   ```

2. **Determine installed software:** This data can be used to troubleshoot either performance issues or unexpected behavior for a specific node in your cluster. For example, unexpected behavior can be the situation where a MapReduce job runs for longer duration than expected.

   To see the list of installed software on a particular host machine, go to **Control Panel** -> **All Control Panel Items** -> **Programs and Features**.

3. **Detect running processes:** This data can be used to troubleshoot either performance issues or unexpected behavior for a specific node in your cluster.

   You can either press **CTRL** + **SHIFT** + **DEL** on the affected host machine or you can execute the following command on Powershell as an Administrator user:

   ```
   tasklist
   ```

4. **Detect Java running processes:** Use this command to verify the Hadoop processes running on a specific machine.

   As *$HADOOP_USER*, execute the following command on the affected host machine:

   ```
   su $HADOOP_USER
   jps
   ```

   You should see the following output:

```
988 Jps
2816 -- process information unavailable
2648 -- process information unavailable
1768 -- process information unavailable
```

Note that no actual name is given to any process. Ensure that you map the process IDs (pid) from the output of this command to the `.wrapper` file within the `C:\hdp\hadoop-1.1.0-SNAPSHOT\bin` directory.

Note

Ensure that you provide complete path to the Java executable, if Java `bin` directory's location is not set within your PATH.

5. **Detect Java heap allocation and usage:** Use the following command to list Java heap information for a specific Java process. This data can be used to verify the heap settings and thus analyze if a particular Java process is reaching the threshold.

   Execute the following command on the affected host machine:

   ```
   jmap -heap $pid_of_Hadoop_process
   ```

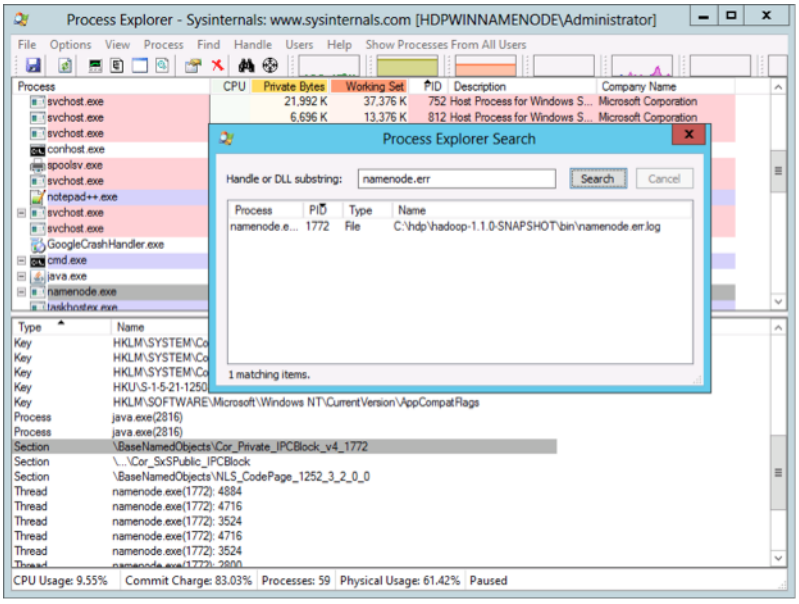
   For example, you should see output similar to the following:

   ```
   C:\hdp\hadoop-1.1.0-SNAPSHOT>jmap -heap 2816
   Attaching to process ID 2816, please wait...
   Debugger attached successfully.
   Server compiler detected.
   JVM version is 20.6-b01

   using thread-local object allocation.
   Mark Sweep Compact GC

   Heap Configuration:
      MinHeapFreeRatio = 40
      MaxHeapFreeRatio = 70
      MaxHeapSize      = 4294967296 (4096.0MB)
      NewSize          = 1310720 (1.25MB)
      MaxNewSize       = 17592186044415 MB
      OldSize          = 5439488 (5.1875MB)
      NewRatio         = 2
      SurvivorRatio    = 8
      PermSize         = 21757952 (20.75MB)
      MaxPermSize      = 85983232 (82.0MB)

   Heap Usage:
   New Generation (Eden + 1 Survivor Space):
      capacity = 10158080 (9.6875MB)
      used     = 4490248 (4.282234191894531MB)
      free     = 5667832 (5.405265808105469MB)
      44.203707787298384% used
   Eden Space:
      capacity = 9043968 (8.625MB)
      used     = 4486304 (4.278472900390625MB)
      free     = 4557664 (4.346527099609375MB)
      49.60548290307971% used
   From Space:
      capacity = 1114112 (1.0625MB)
   ```

```
   used     = 3944 (0.00376129150390625MB)
   free     = 1110168 (1.0587387084960938MB)
   0.35400390625% used
To Space:
   capacity = 1114112 (1.0625MB)
   used     = 0 (0.0MB)
   free     = 1114112 (1.0625MB)
   0.0% used
tenured generation:
   capacity = 55971840 (53.37890625MB)
   used     = 36822760 (35.116920471191406MB)
   free     = 19149080 (18.261985778808594MB)
   65.7880105424442% used
Perm Generation:
   capacity = 21757952 (20.75MB)
   used     = 20909696 (19.9410400390625MB)
   free     = 848256 (0.8089599609375MB)
   96.10139777861446% used
```

6. **Show open files:** Use Process Explorer to determine which processes are locked on a specific file. See Windows Sysinternals - Process Explorer for information on using Process explorer.

   For example, you can use Process Explorer to troubleshoot the file lock issues that prevent a particular process from starting as shown in the screenshot below:



7. **Verify well-formed XML:**

   Ensure that the Hadoop configuration files (for example, `hdfs-site.xml`, etc.) are well formed.

   You can either use **Notepad++** or any third-party tools like **Oxygen**, **XML Spy**, etc. to validate the configuration files. Use the following instructions:

   a. Open the XML file to be validated in Notepad++ and select **XML Tools** -> **Check XML Syntax** .

b. Resolve validation errors, if any.

8. **Detect AutoStart Programs:** This information helps to isolate errors for a specific host machine.

   For example, a potential port conflict between auto-started process and HDP processes, might prevent launch for one of the HDP components.

   Ideally, the cluster administrator must have the information on auto-start programs handy. Use the following command to launch the GUI interface on the affected host machine:

   ```
   C:\Windows\System32\msconfig.exe
   ```

   Click **Startup** tab. Ensure that no startup items are enabled on the affected host machine.

9. **Collect list of all mounts on the machine:** This information determines the drives that are actually mounted or available on the host machine for use. To troubleshoot disks capacity issues, use this command to determine if the system is violating any storage limitations.

   Execute the following command on Powershell:

   ```
   Get-Volume
   ```

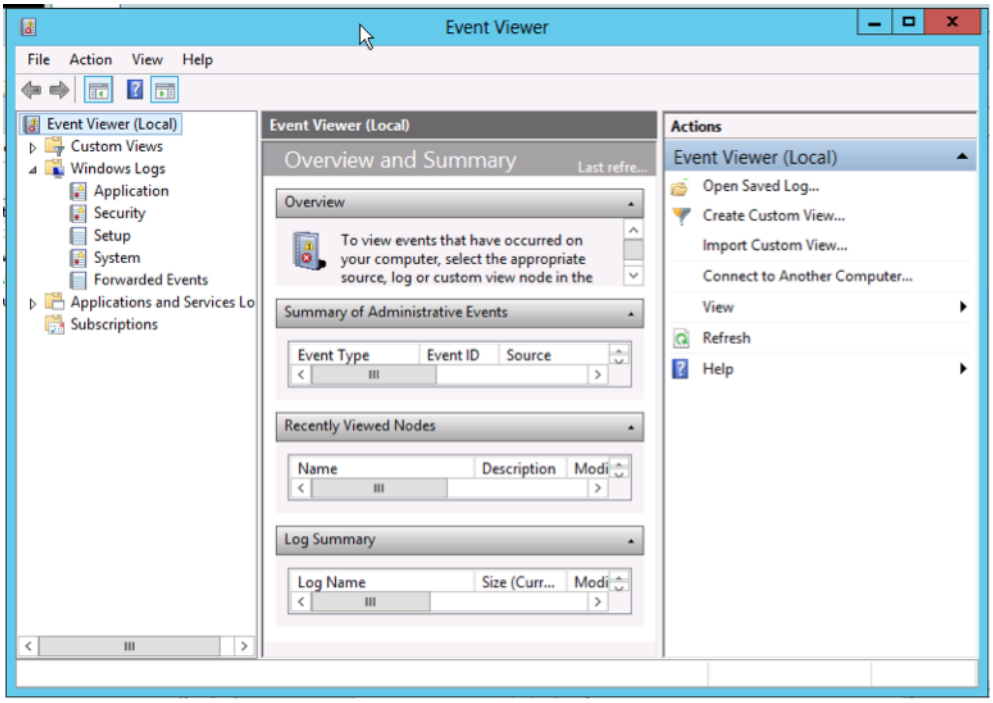
   You should see output similar to the following:

   ```
   DriveLetter       FileSystemLabel  FileSystem       DriveType
    HealthStatus        SizeRemaining              Size
   -----------       ---------------  ----------       ---------
    ------------        -------------               ----
                     System Reserved  NTFS             Fixed            Healthy
                     108.7 MB           350 MB
   C                                  NTFS             Fixed            Healthy
                     10.74 GB         19.97 GB
   D                 HRM_SSS_X64FR... UDF              CD-ROM           Healthy
                          0 B          3.44 GB
   ```

10. **Operating system messages** Use Event Viewer to detect messages with a system or an application.

    Event Viewer can determine if a machine was rebooted or shut down at a particular time. Use the logs to isolate issues for HDP services that were non-operational for a specific time.

    Go to **Control Panel** -> **All Control Panel Items** -> **Administrative Tools** and click the **Event Viewer** icon.



11. **Hardware/system information:** Use this information to isolate hardware issues on the affected host machine.

    Go to **Control Panel** -> **All Control Panel Items** -> **Administrative Tools** and click the **System Information** icon.

12. **Network information:** Use the following commands to troubleshoot network issues.

    - **ipconfig:** This command provides the IP address, validates if the network interfaces are available, and also validates if an IP address is bound to the interfaces. To troubleshoot communication issues between the host machines in your cluster, execute the following command on the affected host machine:

      ```
      ipconfig
      ```

      You should see output similar to the following:

      ```
      Windows IP Configuration

      Ethernet adapter Ethernet 2:

         Connection-specific DNS Suffix  . :
         Link-local IPv6 Address . . . . . : fe80::d153:501e:5df0:f0b9%14
         IPv4 Address. . . . . . . . . . . : 192.168.56.103
         Subnet Mask . . . . . . . . . . . : 255.255.255.0
         Default Gateway . . . . . . . . . : 192.168.56.100

      Ethernet adapter Ethernet:

         Connection-specific DNS Suffix  . : test.tesst.com
         IPv4 Address. . . . . . . . . . . : 10.0.2.15
         Subnet Mask . . . . . . . . . . . : 255.255.255.0
      ```

```
   Default Gateway . . . . . . . . . : 10.0.2.2
```

- **netstat -ano:** This command provides a list of used ports within the system. Use this command to troubleshoot launch issues with HDP master processes. Execute the following command on the host machine to resolve potential port conflict:

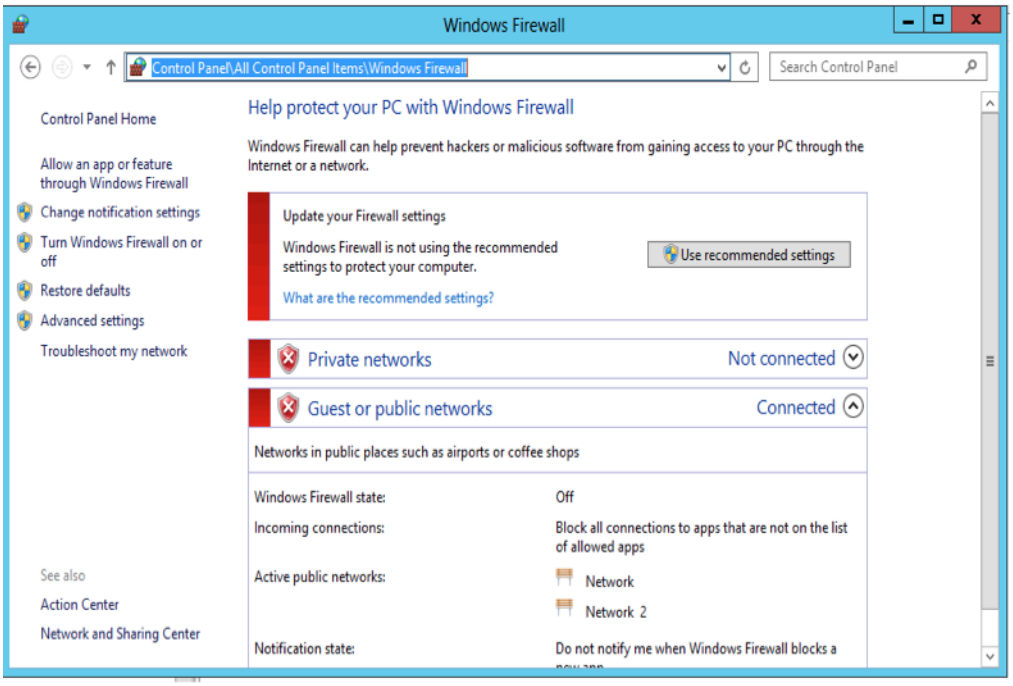
```
netstat -ano
```

You should see output similar to the following:

```
TCP    0.0.0.0:49154          0.0.0.0:0              LISTENING       752
 TCP    [::]:49154             [::]:0                 LISTENING       752
 UDP    0.0.0.0:500            *:*                                    752
 UDP    0.0.0.0:3544           *:*                                    752
 UDP    0.0.0.0:4500           *:*                                    752
 UDP    10.0.2.15:50461        *:*                                    752
 UDP    [::]:500               *:*                                    752
 UDP    [::]:4500              *:*                                    752
```

- **Verify if firewall is enabled on the host machine:** Go to **Control Panel** -> **All Control Panel Items** -> **Windows Firewall** .

You should see the following GUI interface:



9.2. File locations, Ports, and Common HDFS Commands

This section provides a list of files and their locations, port information, and HDFS commands that help to isolate and troubleshoot issues:

- File Locations

- Ports
- Common HDFS Commands

9.2.1. File Locations

- **Configuration files:** These files are used to configure a hadoop cluster.

  1. **core-site.xml:**

     All Hadoop services and clients use this file to locate the NameNode. Therefore, this file must be copied to each node that is either running a Hadoop service or is a client. The Secondary NameNode uses this file to determine location for storing `fsimage` and edits log `<name>fs.checkpoint.dir</name>` locally and location of the NameNode `<name>fs.default.name</name>`. Use the `core-site.xml` file to isolate communication issues with the NameNode host machine.

  2. **hdfs-site.xml:**

     HDFS services use this file. Some important properties of this file are as listed below:

     - HTTP addresses for the two services
     - Replication for DataNodes `<name>dfs.replication</name>`
     - DataNode block storage location `<name>dfs.data.dir</name>`
     - NameNode metadata storage `<name>dfs.name.dir</name>`

     Use `hdfs-site.xml` file to isolate NameNode startup issues. Typically, NameNode startup issues are caused when NameNode fails to load the `fsimage` and edits log to merge. Ensure that the values for all the above properties in `hdfs-site.xml` file are valid locations.

  3. **datanode.xml:**

     DataNode services use the `datanode.xml` file to specify the maximum and minimum heap size for the DataNode service. To troubleshoot issues with DataNode, change the value for `-Xmx` to change the maximum heap size for DataNode service and restart the affected DataNode host machine.

  4. **namenode.xml:**

     NameNode services use the `namenode.xml` file to specify the maximum and minimum heap size for the NameNode service. To troubleshoot issues with NameNode, change the value for `-Xmx` to change the maximum heap size for NameNode service and restart the affected NameNode host machine.

  5. **secondarynamenode.xml:**

     Secondary NameNode services use the `secondarynamenode.xml` file to specify the maximum and minimum heap size for the Secondary NameNode service. To troubleshoot issues with Secondary NameNode, change the value for `-Xmx` to change the maximum heap size for Secondary NameNode service and restart the affected Secondary NameNode host machine.

6. **hadoop-policy.xml:**

   Use the `hadoop-policy.xml` file to configure service-level authorization/ACLs within Hadoop. NameNode accesses this file. Use this file to troubleshoot permission related issues for NameNode.

7. **log4j.properties:**

   Use the `log4j.properties` file to modify the log purging intervals of the HDFS logs. This file defines logging for all the Hadoop services. It includes, information related to appenders used for logging and layout. See log4j documentation for more details.

- **Log Files:** The following are sets of log files for each of the HDFS services. They are typically stored in `C:\hadoop\logs\hadoop and C:\hdp\hadoop-1.1.0-SNAPSHOT\bin` by default.

  - **HDFS .out files:** The log files with the `.out` extension for HDFS services are located in `C:\hdp\hadoop-1.1.0-SNAPSHOT\bin` and have the following naming convention:

    - datanode.out.log
    - namenode.out.log
    - secondarynamenode.out.log

    These files are created and written to when HDFS services are bootstrapped. Use these files to isolate launch issues with DataNode, NameNode, or Secondary NameNode services.

  - **HDFS .wrapper files:** The log files with the `.wrapper` extension are located in `C:\hdp\hadoop-1.1.0-SNAPSHOT\bin` and have the following file names:

    - datanode.wrapper.log
    - namenode.wrapper.log
    - secondarynamenode.wrapper.log

    These files contain startup command string to start the service and they also provide the output of the process ID on service startup.

  - **HDFS .log and .err files:**

    The following files are located in `C:\hdp\hadoop-1.1.0-SNAPSHOT\bin`:

    - datanode.err.log
    - namenode.err.log
    - secondarynamenode.err.log

    following files are located in `C:\hadoop\logs\hadoop`:

- hadoop-datanode-*$Hostname*.log
- hadoop-namenode-*$Hostname*.log
- hadoop-secondarynamenode-*$Hostname*.log

These files contain log messages for the running Java service. If there are any errors encountered while the service is already running, the stack trace of the error is logged in the above files.

$Hostname is the host where the service is running. For example, on a node where the hostname is `namemode.example.com`, the file would be saved as `hadoop-namenode-namemodehost.example.com.log`.

Note

By default, these log files are rotated daily. Use `C:\hdp\hadoop-1.1.0-SNAPSHOT\conf\log4j.properties` file to change log rotation duration.

- **HDFS .<date> files:**

  The log files with the `.<date>` extension for HDFS services have the following format:

  - hadoop-namenode-*$Hostname*.log.<date>
  - hadoop-datanode-*$Hostname*.log.<date>
  - hadoop-secondarynamenode-*$Hostname*.log.<date>

  When a `.log` file is rotated, it is appended with the current date.
  An example of the file name would be: `hadoop-datanode-hdp121.localdomain.com.log.2013-02-08`.

  Use these files to compare the past state of your cluster with the current state in order to troubleshoot potential patterns of occurrence.

9.2.2. Enabling Logging

To enable logging change the settings in the `hadoop-env.cmd`. After modifying the `hadoop-env.cmd`, recreate the NameNode service XML and then restart the NameNode.

Note

To enable audit logging change the hdfs.audit.logger value to INFO,RFAAUDIT and then overwrite the NameNode service XML and restart the NameNode.

1. Open the Hadoop Environment script, `%HADOOP_HOME%\etc\hadoop\hadoop-env.cmd`.
2. Prepend the following text in the HADOOP_NAMENODE_OPTS definition, for example to enable Garbage Collection logging:

```
-Xloggc:%HADOOP_LOG_DIR%/gc-namenode.log -verbose:gc -XX:+PrintGCDetails -
XX:+PrintGCTimeStamps -XX:+PrintGCDateStamps
```

For example:

```
set HADOOP_NAMENODE_OPTS=-Xloggc:%HADOOP_LOG_DIR%/gc-namenode.log -
verbose:gc -XX:+PrintGCDetails -XX:+PrintGCTimeStamps -XX:+PrintGCDateStamps
 -Dhadoop.security.logger=%HADOOP_SECURITY_LOGGER% -Dhdfs.audit.logger=
%HDFS_AUDIT_LOGGER% %HADOOP_NAMENODE_OPTS%
```

3. Run the following command to recreate the NameNode service XML:

```
%HADOOP_HOME%\bin\hdfs.cmd --service namenode > %HADOOP_HOME%\bin\namenode.
xml
```

4. Verify that the NameNode Service XML was updated.

5. Restart the NameNode service.

9.2.3. Common HDFS Commands

This section provides common HDFS commands to troubleshoot HDP deployment on Windows platform. An exhaustive list of the commands is available at here.

- **Get Hadoop version:**

  Execute the following command on your cluster host machine:

  ```
  hadoop version
  ```

- **Check block information:** This command provides a directory listing and displays which node contains the block. Use this command to determine if a block is under replicated.

  Execute the following command on your HDFS cluster host machine:

  ```
  hadoop fsck / -blocks -locations -files
  ```

  You should see output similar to the following:

  ```
  FSCK started by hdfs from /10.0.3.15 for path / at Tue Feb 12 04:06:18 PST
  2013
    / <dir>
    /apps <dir>
    /apps/hbase <dir>
    /apps/hbase/data <dir>
    /apps/hbase/data/-ROOT- <dir>
    /apps/hbase/data/-ROOT-/.tableinfo.0000000001 727 bytes, 1 block(s):
     Under replicated blk_-3081593132029220269_1008.
    Target Replicas is 3 but found 1 replica(s). 0.
  blk_-3081593132029220269_1008
     len=727 repl=1 [10.0.3.15:50010]
     /apps/hbase/data/-ROOT-/.tmp <dir>
     /apps/hbase/data/-ROOT-/70236052 <dir>
     /apps/hbase/data/-ROOT-/70236052/.oldlogs <dir>
     /apps/hbase/data/-ROOT-/70236052/.oldlogs/hlog.1360352391409 421 bytes,
  1 block(s):  Under
         replicated blk_709473237440669041_1006.
       Target Replicas is 3 but found 1
  ```

```
      replica(s). 0. blk_709473237440669041_1006 len=421 repl=1 [10.0.3.
15:50010] ...
```

- **HDFS report:** Use this command to receive HDFS status.

  Execute the following command as an HDFS user:

  ```
  hadoop dfsadmin -report
  ```

  You should see output similar to the following:

  ```
  -bash-4.1$ hadoop dfsadmin -report
  Safe mode is ON
  Configured Capacity: 11543003135 (10.75 GB)
  Present Capacity: 4097507328 (3.82 GB)
  DFS Remaining: 3914780672 (3.65 GB)
  DFS Used: 182726656 (174.26 MB)
  DFS Used%: 4.46%
  Under replicated blocks: 289
  Blocks with corrupt replicas: 0
  Missing blocks: 0

  -------------------------------------------------
  Datanodes available: 1 (1 total, 0 dead)

  Name: 10.0.3.15:50010
  Decommission Status : Normal
  Configured Capacity: 11543003135 (10.75 GB)
  DFS Used: 182726656 (174.26 MB)
  Non DFS Used: 7445495807 (6.93 GB)
  DFS Remaining: 3914780672(3.65 GB)
  DFS Used%: 1.58%
  DFS Remaining%: 33.91%
  Last contact: Sat Feb 09 13:34:54 PST 2013
  ```

- **Safemode:** Safemode is a state where no changes can be made to the blocks. HDFS cluster is in safemode state during start up because the cluster needs to validate all the blocks and their locations. Once validated, the safemode is then disabled.

  The options for safemode command are:

  ```
  hadoop dfsadmin -safemode [enter | leave | get]
  ```

  To enter the safemode, execute the following command on your NameNode host machine:

  ```
  hadoop dfsadmin -safemode enter
  ```

10. Uninstalling HDP

Choose one of the following options to uninstall HDP:

- Option I - Use Windows GUI
- Option II - Use command line utility

10.1. Option I - Use Windows GUI

1. Open the **Programs and Features** Control Panel Pane.
2. Select the program listed: **Hortonworks Data Platform for Windows**.
3. With that program selected, click on the **Uninstall** option.

10.2. Option II - Use Command Line Utility

1. On each cluster host, execute the following command from the command shell:

   ```
   msiexec /x "$MSI_PATH" /lv "$PATH_to_Installer_Log_File"
   ```

   where

   - *$MSI_PATH* is the full path to MSI.
   - *$PATH_to_Installer_Log_File* is full path to Installer log file.

2. Optionally, you can also specify if you want delete the data in target data directories.

   To do this, use the DESTROY_DATA command line option as shown below:

   ```
   msiexec /x "$MSI_PATH" /lv "$PATH_to_Installer_Log_File" DESTROY_DATA="yes"
   ```

Note

During uninstall if DESTROY_DATA is not specified or set to no, data directories are preserved as well as the hadoop user that owns them.

11. Appendix: Adding A User

11.1. Adding a Smoke Test User

Creating a smoke test user lets you run HDP smoke tests without having to run them as the hadoop user. To create a smoke test user:

1. Open a command prompt as the hadoop user:

   ```
   runas /user:hadoop cmd
   ```

2. Change permissions on the MapReduce directory to include other users:

   ```
   %HADOOP_HOME%\bin\hadoop fs -chmod -R 757 /mapred
   ```

3. Create a HDFS directory for the smoketest user:

   ```
   %HADOOP_HOME%\bin\hadoop dfs -mkdir -p /user/smoketestuser
   ```

4. Change ownership to the smoketest user.

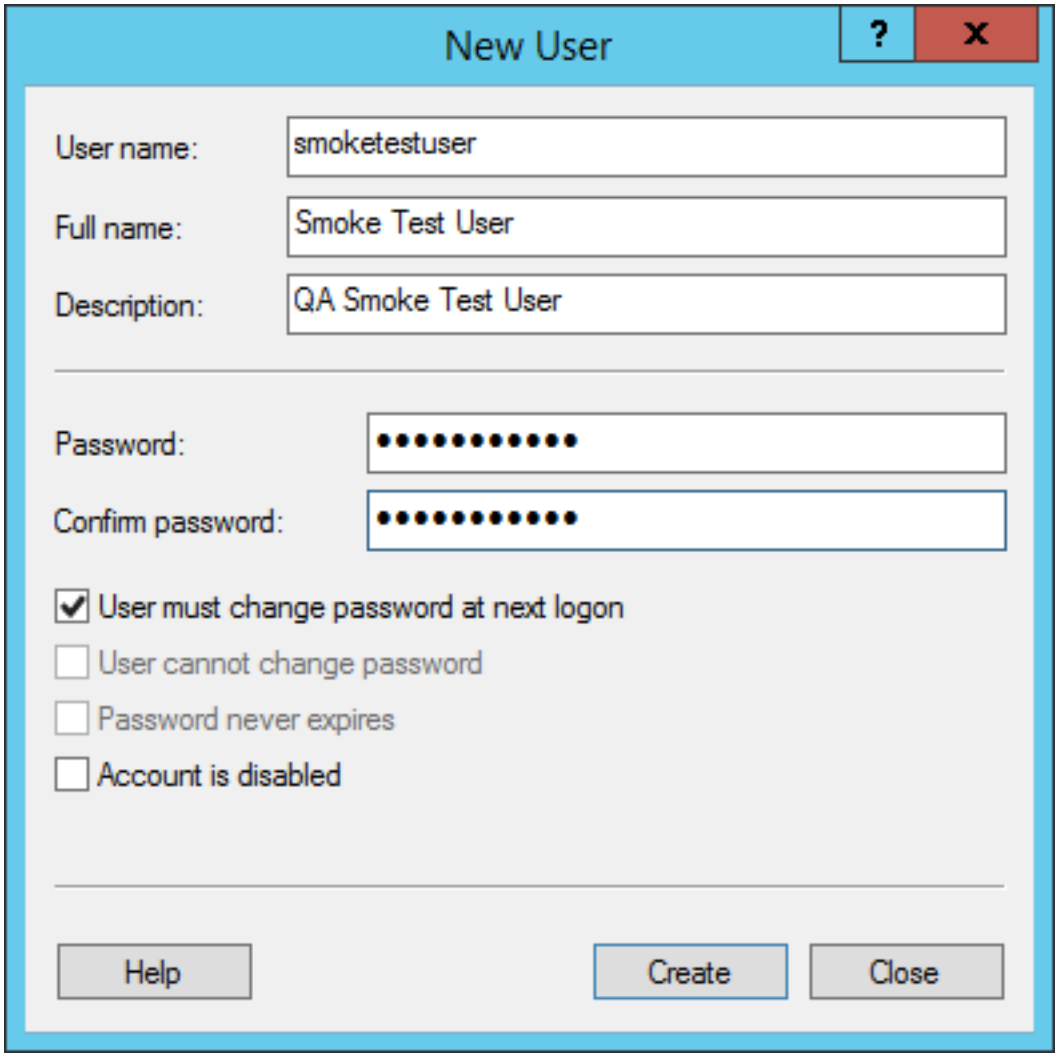
   ```
   %HADOOP_HOME%\bin\hadoop dfs -chown -R smoketestuser /user/smoketestuser
   ```

5. Create a smoketest user account in Windows.

   a. Navigate to Computer Management.

   b. Select **Local Users and Groups** > **File** > **Action** > **New User** on Windows Server 2008 or **Local Users and Groups** > **Action** > **New User** on Windows Server 2012.

      The New User dialog displays:



c. Create the username and password for your smoke test user. Determine password requirements and select **Create**.

6. Validate the smoketest user by running smoketests as the smoketest user.

   a. Switch to a command prompt as the smoktest user. For example:

   ```
   runas /user:smoketestuser cmd
   ```

   b. In the smoketest user, run the smoke tests:

   ```
   %HADOOP_HOME%\Run-Smoketests.cmd
   ```