

Hortonworks Data Platform

Installing and Upgrading HDP on Windows

(Dec 2, 2014)

Hortonworks Data Platform: Installing and Upgrading HDP on Windows

Copyright © 2012-2015 Hortonworks, Inc. Some rights reserved.

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.



Except where otherwise noted, this document is licensed under
Creative Commons Attribution ShareAlike 3.0 License.
<http://creativecommons.org/licenses/by-sa/3.0/legalcode>

Table of Contents

1. Before You Begin	1
1.1. HDP Components	1
1.2. Minimum System Requirements	2
1.2.1. Hardware Recommendations	2
1.2.2. Operating System Requirements	2
1.2.3. Software Requirements	3
1.2.4. (Optional) MS SQL Server for Hive and Oozie Database Instances	3
1.2.5. (Optional) MySQL Client for Ranger	4
1.3. Preparing for Hadoop Installation	4
1.3.1. Gather Hadoop Cluster Information	5
1.3.2. Configure Network Time Server (NTS)	5
1.3.3. Set Interfaces to IPv4 Preferred	5
1.3.4. (Optional) Create Hadoop user	6
1.3.5. Enable Remote PowerShell Script Execution	6
1.3.6. Enable Remote PowerShell Execution for Nodes in a Workgroup	7
1.3.7. Enable Networking Configurations for Active Directory Domains	7
1.3.8. Configure Ports	14
1.3.9. Install Required Software	17
2. Defining Cluster Properties	22
2.1. Download the HDP Installer	22
2.2. Using the HDP Setup Interface	22
2.3. Set HDP Properties	23
2.4. Set High Availability Properties	27
2.5. Set Ranger Properties	28
2.6. Complete the GUI Installation Process	33
2.7. Manually Creating a Cluster Properties File	33
3. Quick Start for a Single Node HDP Installation	42
4. Deploying a Multi-node HDP Cluster	46
4.1. About the HDP MSI Installer and HDP Public Properties	46
4.1.1. HDP MSI Installer command format	46
4.1.2. Standard Installer Options	47
4.1.3. HDP Public Properties	47
4.2. Option 1: Central Push Install Using a Deployment Service	49
4.3. Option 2: Central HDP Install Using the Push Install HDP Script	50
4.4. Option 3: Installing HDP from the Command Line	52
5. Configuring HDP Components and Services	55
5.1. Enable HDP Services	55
5.2. (MS SQL Only): Configure Hive when Metastore DB is in a Named Instance....	56
5.3. Configure MapReduce on HDFS	57
5.4. Configure HBase on HDFS	57
5.5. Configure Hive on HDFS	57
5.6. Configure Tez for Hive	58
5.7. Configure Node Label Support for YARN Applications	59
5.8. Configure Ranger Security	60
5.9. Using Apache Slider	60
5.10. (Optional) Install Microsoft SQL Server JDBC Driver	60
5.11. Start HDP Services	61
6. Validating the Installation	62

7. Upgrading HDP Manually	63
7.1. Before You Begin	63
7.2. Back Up Critical HDFS Metadata	63
7.3. Back Up Your Configuration Files	64
7.4. Stop Running HDP Services	65
7.5. Uninstall HDP on All Nodes	65
7.6. Update the HDP Cluster Properties File	65
7.7. Installing HDP 2.2 and Maintaining Your Prior Data	67
7.8. Prepare the Metastore Databases	68
7.9. Upgrade HDFS Metadata	68
7.10. Upgrade HBase	69
7.11. Upgrade Oozie	69
7.12. Upgrade Tez	70
7.13. Adding Node Label Support for YARN Applications	71
7.14. Configure Ranger Security	72
7.15. Using Apache Slider	72
7.16. Starting HDP Services	73
7.17. Setting up HDP	73
7.18. Validating Your Data	74
7.19. Verify that HDP Services are Working	74
7.20. Finalize Upgrade	74
7.21. Troubleshooting	75
8. Managing HDP on Windows	76
8.1. Starting HDP Services	76
8.2. Enabling NameNode High Availability	76
8.3. Validating HA Configuration	76
8.4. Stopping HDP Services	78
9. Troubleshooting Your Deployment	79
9.1. Cluster Information	79
9.2. File Locations, Logging, and Common HDFS Commands	84
9.2.1. File Locations	84
9.2.2. Enabling Logging	87
9.2.3. Common HDFS Commands	87
10. Uninstalling HDP	90
11. Appendix: Adding a Smoketest User	91

List of Tables

1.1. Hadoop cluster components 5

1. Before You Begin

This section provides you with information you should review before you install the Hortonworks Data Platform (HDP) on Windows:

- [HDP Components](#)
- [Minimum System Requirements](#)
- [Preparing for Hadoop Installation](#)

1.1. 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](#) in the *Getting Started Guide for Windows*.

While it is possible to deploy all of HDP on a single host, a single-node installation is appropriate only for initial evaluation (see the [Cluster Planning Guide for Windows](#)). In general you should use at least three hosts: one master host and two slaves.

1.2. Minimum System Requirements

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

- [Hardware recommendations](#)
- [Operating System Requirements](#)
- [Software Requirements](#)
- (Optional) [MS SQL Server for Hive and Oozie Database Instances](#)
- (Optional) [MySQL Client for Ranger](#)

1.2.1. Hardware Recommendations

Although there is no single hardware requirement for installing HDP, there are some basic guidelines. See sample setups in the [Cluster Planning Guide for Windows](#).



Note

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

1.2.2. Operating System Requirements

The following operating systems are supported:

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

1.2.3. Software Requirements

The following table provides installation information for each software prerequisite.

You can also use Microsoft SQL Server for Hive and Oozie metastores. If you plan to install Ranger, MySQL is required. For more information, see [\(Optional\) MS SQL Server for Hive and Oozie Database Instances](#) and [\(Optional\) MySQL Client for Ranger](#).

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 Required Software .
Java JDK	JDK 1.7.0_51	PATH	Add the directory where the 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 Required Software .
		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 added automatically.	Install with default parameters; see Install Required Software .
Microsoft .NET Framework	4.0	PATH	Default location added automatically.	Install with default parameters; see Install Required Software .

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. (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 is deployed and available in your environment, and that your database administrator creates the following databases and users. You will need the following information when you configure HDP:

- For Hive, create a SQL database instance:
 1. Create a Hive database instance in SQL and record its name, such as `hive_dbname`.
 2. Create Hive user on SQL, add it 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, create a SQL database instance:
 1. Create an Oozie database instance and record its name, such as `oozie_dbname`.
 2. Create Oozie user on SQL, add it 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.
- The following steps are required after installing SQL Server; refer to SQL Server documentation for more information:
 1. Ensure that `TCP/IP` is enabled under **SQL Server Network Configuration**. This might require restarting SQL Server.
 2. Add firewall exceptions for SQL Server and SQL Browser services.
 3. Before using SQL server for Hive or Oozie metastores, set up the Microsoft SQL Server JDBC Driver (see [\(Optional\) Install Microsoft SQL Server JDBC Driver](#) after installing Hive or Oozie.

1.2.5. (Optional) MySQL Client for Ranger

Ranger offers a centralized security framework, including security policy administration: authorization, accounting, and data protection. It is an optional component. Ranger requires MySQL. If you plan to install Ranger, complete the following steps before installing HDP:

1. Install [MySQL Client](#).
2. Add the path to `mysql.exe` to your system `PATH` variable.
3. Set the `JAVA_HOME` variable to the installed JDK version; for example,
`$ENV:JAVA_HOME="c:\Java\jdk1.7.0_67"`
4. Download the [MySQL Connector Jar file](#).
5. Copy the jar file into your installation folder:
 - If you plan to install Ranger using the MSI Setup GUI, copy the jar file into the folder containing the MSI.
 - If you plan to install Ranger using the command-line interface, copy the jar file into the folder containing your cluster properties file.

1.3. Preparing 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 Hadoop Cluster Information](#)
- [Configure the Network Time Server \(NTS\)](#)

- [Set Interfaces to IPv4 Preferred](#)
- [\(Optional\) Create Hadoop User](#)
- [Enable Remote PowerShell Script Execution](#)
- [Configure Ports](#)
- [Install Required Software](#)

1.3.1. Gather Hadoop Cluster Information

Before deploying your HDP installation, collect the Hostname or IPv4 address of each the following cluster components:

Table 1.1. Hadoop cluster components

Required components	<ul style="list-style-type: none"> • NameNode and optional Secondary NameNode • ResourceManager • Hive Server • SlaveNode • WebHCat • Client Host
Optional components	<ul style="list-style-type: none"> • ZooKeeper • HBase Master • Flume • Knox Gateway • Ranger (requires MySQL client) • Microsoft SQL Server configured with a Hive and Oozie database instance, system account names and passwords

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

1.3.2. Configure Network Time Server (NTS)

The clocks of all nodes in your cluster must be able to synchronize with each other. To configure this for Windows Server, use the instructions provided at [How to configure an authoritative time server in Windows Server](#).

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 IPv6 or its specific components in Windows](#)) or set the preference to IPv4.

Ensure that the host's fully-qualified domain name ("FQDN") resolves to an IPv4 address as follows:

1. To verify that IPv4 is set to preferred, enter:

```
ipconfig /all
```

The system should display:

```
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. To flush the DNS cache, enter:

```
ipconfig /flushdns
```

3. To verify that the hostname of the system resolves to the correct IP address, enter:

```
ping -a 10.0.0.2
```

The system should display:

```
Pinging win08r2-nodel.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 services run under the ownership of a Windows user account. The HDP installer establishes a `hadoop` user as follows:

- If the `hadoop` user account does not exist, HDP installer automatically creates a local user with random password.
- If the `hadoop` user account already exists, HDP installer changes the current password to a new random password.

The random password is passed on the command line throughout the install process, and then discarded. The administrator can change the password later, but it must be changed 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 HDP utility scripts require remote execution of PowerShell scripts on all nodes in the Hadoop cluster. For example, the scripts for starting and stopping the entire cluster with a single command (provided with HDP) require remote scripting. Therefore, we strongly recommend that you complete the following steps at every host in your cluster.

1.3.6. Enable Remote PowerShell 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 following instructions do not work if the Administrator account has an empty password.

To enable remote scripting using PowerShell commands:

1. On each host in the cluster, open a PowerShell window with "Run as Administrator" privileges, and enter:

```
Set-ExecutionPolicy "AllSigned"  
Enable-PSRemoting  
Set-item wsman:localhost\client\trustedhosts -value "host1,host2"
```

where `host1,host2` is a list of comma-separated host names in your cluster (for example, "HadoopHost1, HadoopHost2, HadoopHost3").

2. For each host in the cluster, open a PowerShell window with "Run as Administrator" privileges, and enter:

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

where `Host1,Host2` is a list of comma-separated host names in your cluster (for example, "HadoopHost1, HadoopHost2, HadoopHost3").

1.3.7. Enable Networking Configurations for Active Directory Domains

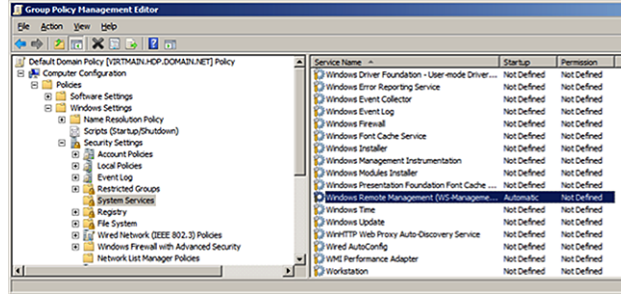
To enable remote scripting and configure domain policies for Windows Remote Management, complete the following instructions on a domain controller machine.



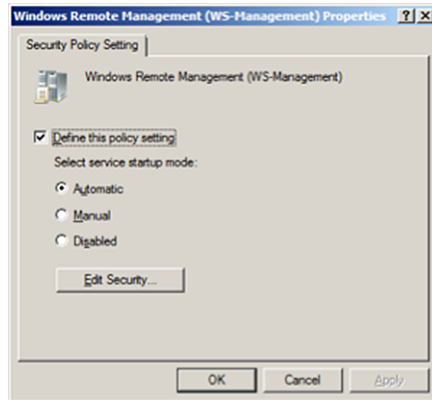
Note

All of the following actions are performed via Group Policy Management > Default Domain Policy > Edit.

1. Set the WinRM service to autostart.
 - a. Go to Computer Configuration > Policies > Windows Settings > Security Settings > Windows Remote Management (WS-Management).



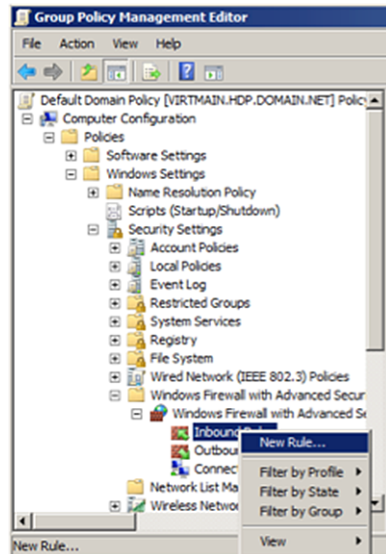
b. Set Startup Mode to Automatic.



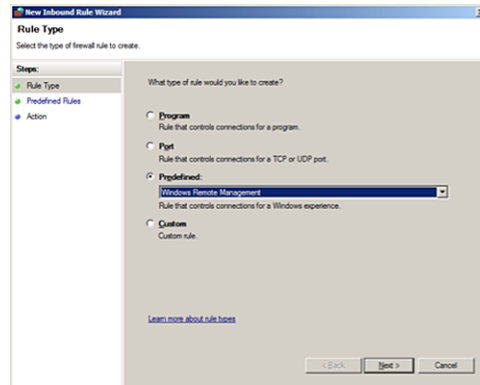
2. Add firewall exceptions to allow the service to communicate.

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

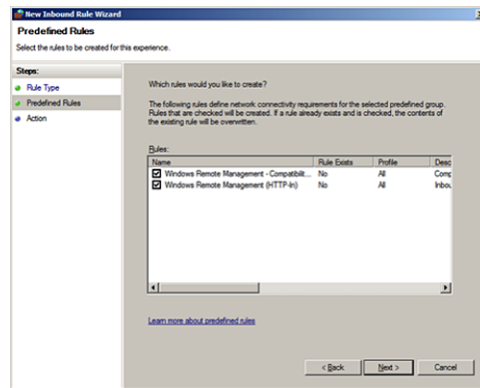
b. To create a new Inbound Rule, right-click Windows Firewall with Advanced Security.



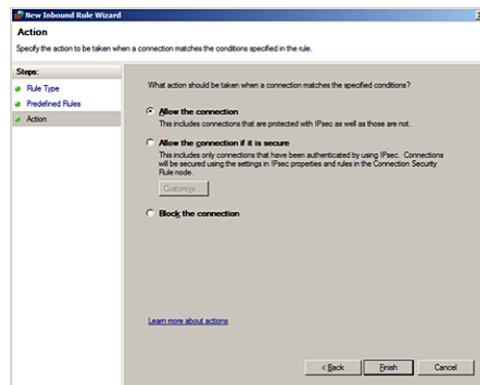
c. Specify the rule type as Predefined, Windows Remote Management.



The Predefined rule automatically creates two rules:



d. Configure Action as Allow the connection.

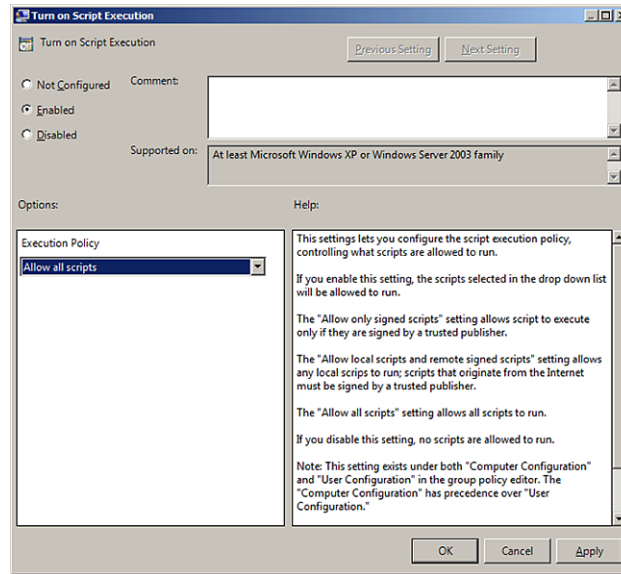


e. Click Finish.

3. Set script execution policy.

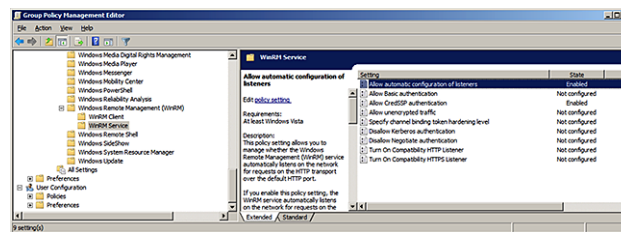
a. Go to Computer Configuration > Policies > Administrative Templates > Windows Components > Windows PowerShell.

- b. At Setting, select Turn on Script Execution.
- c. Set Execution Policy to Allow all scripts.



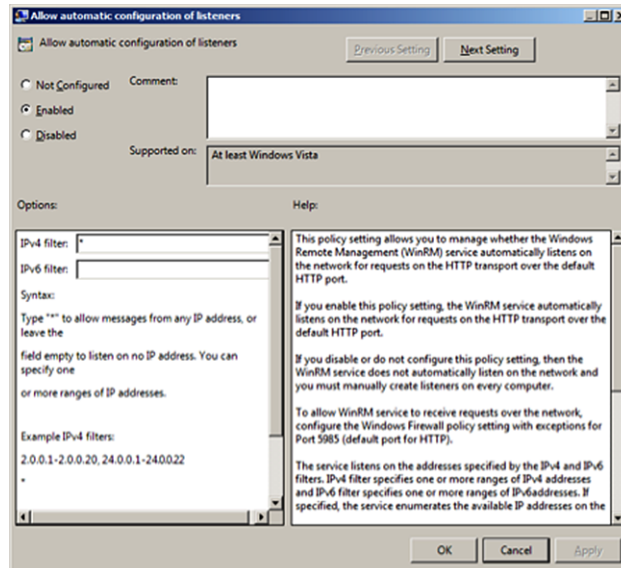
4. Set up the WinRM service.

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

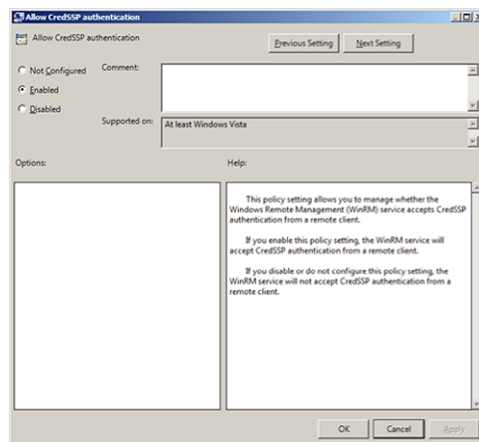


b. Create a WinRM listener.

- i. To allow automatic configuration of listeners, select Enabled, and then set IPv4 filter to * (all addresses) or specify a range:



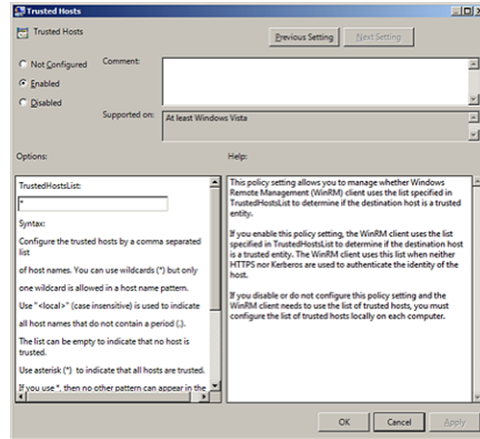
ii. Allow CredSSP authentication and click OK.



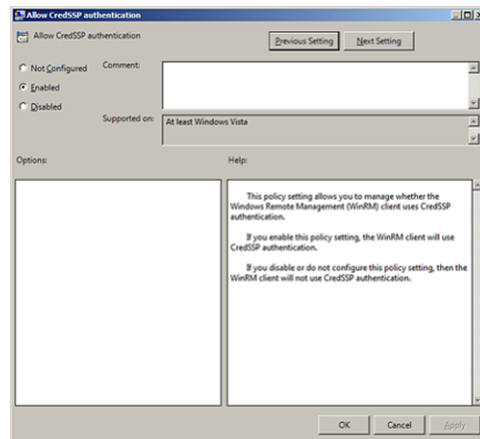
5. Set up the WinRM client.

- a. Go to Computer Configuration > Policies > Administrative Templates > Windows Components > Windows Remote Management (WinRM) > WinRM Client.
- b. Configure the trusted host list (the IP addresses of the computers that can initiate connections to the WinRM service).

Set `TrustedHostsList` to `*` (all addresses) or specify a range.

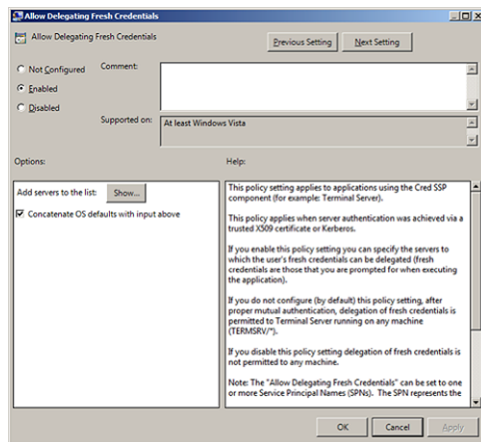


- c. Set Allow CredSSP authentication to Enabled, and click OK.

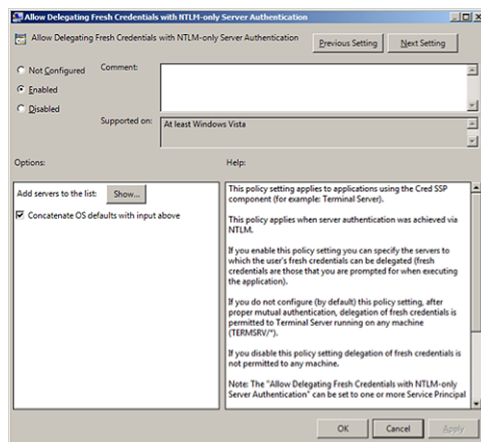


6. Enable credentials delegation.

- Go to Computer Configuration > Policies > Administrative Templates > System > Credentials Delegation.
- To allow delegation of fresh credentials, select Enabled.
- Under Options, select Show. Set WSMAN to * (all addresses) or specify a range. Click Next Setting.



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



7. Enable the creation of WSMAN SPN.
 - a. Go to **Start > Run**. In the dialog box, enter `ADSIEdit.msc`. Click **Enter**.
 - b. Expand the **OU=Domain Controllers** menu item and select **CN=domain controller hostname**.
 - c. Go to **Properties > Security > Advanced > Add**.
 - d. Enter `NETWORK SERVICE`, click **Check Names**, then click **OK**.
 - e. In the **Permission** field, select **Validated write to service principal name**.
 - f. Click **Allow**.
 - g. To save your changes, click **OK**.

8. Restart the WinRM service and update policies.

- a. At the domain controller machine, open a PowerShell window and enter:

```
Restart-Service WinRM
```

- b. At each of the other hosts in domain, enter:

```
gpupdate /force
```

- c. Ensure that SPN-s WSMAN is created for your environment.

At your domain controller machine, enter:

```
setspn -l Domain_Controller_Hostname
```

You should see output similar to the following:

```
Administrator: Windows PowerShell
Windows PowerShell
Copyright (C) 2009 Microsoft Corporation. All rights reserved.

PS C:\Users\Administrator> Restart-Service WinRM
PS C:\Users\Administrator> setspn -l WSMAN/NTLMAUTH
Setspn: Command completed successfully.
PS C:\Users\Administrator> setspn -l WSMAN/NTLM
Setspn: Command completed successfully.
PS C:\Users\Administrator> setspn -l WSMAN/HTTP
Setspn: Command completed successfully.
PS C:\Users\Administrator>
```

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

```
setspn -l Domain_Controller_Hostname
```

You should see output similar to the following:

```
Administrator: Windows PowerShell
PS C:\Users\Administrator> setspn -l WSMAN/NTLMAUTH
Setspn: Command completed successfully.
PS C:\Users\Administrator>
```

1.3.8. Configure Ports

HDP uses multiple ports for communication with clients and between service components. For example, the Hive Metastore port represents the port that the metastore database can use to connect and communicate. To enable HDP communication, open the specific ports that HDP uses.

To open specific ports, you can set the access rules in Windows. For example, the following command opens up port 80 in the active Windows Firewall:

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

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 following tables specify which ports must be opened for specific ecosystem components to communicate with each other. Open the appropriate ports before you install HDP.

1.3.8.1. HDFS Ports

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

Service	Servers	Default Ports Used	Protocol	Description	Needs End-user Access?	Configuration Parameters
NameNode WebUI	MasterNodes (NameNode and any backup NameNodes)	50070	http	WebUI to look at current status	Yes (typically admins, Dev/Support teams)	dfs.http.address
NameNode metadata service	MasterNodes (NameNode and any backup NameNodes)	8020/9000	IPC	File system metadata operations	Yes (all clients who need to interact directly with the HDFS)	Embedded in URI specified by fs.defaultFS
DataNode	All Slave Nodes	50076	http	DataNode WebUI to access the status, logs, etc.	Yes (typically admins, Dev/Support teams)	dfs.datanode.http.address
DataNode	All Slave Nodes	50010		Data transfer		dfs.datanode.address
DataNode	All Slave Nodes	50020	IPC	Metadata operations	No	dfs.datanode.address
Secondary NameNode	Secondary NameNode and any backup Secondary NameNodes	50090	http	Checkpoint for NameNode metadata	No	dfs.secondary.http.address

1.3.8.2. YARN Ports

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

Service	Servers	Default Ports Used	Protocol	>Description	Needs End-user Access?	Configuration Parameters
Resource Manager WebUI	Master Nodes (Resource Manager and any back-up Resource Manager node)	8088	http	WebUI for Resource Manager	Yes	yarn.resource-manager.webapp.address
Resource Manager WebUI	Master Nodes (Resource Manager and any back-up Resource Manager node)	8090	https	WebUI for Resource Manager	Yes	yarn.resource-manager.webapp.https.address
Resource Manager	Master Nodes (Resource Manager and	8032	IPC	For application submissions	Yes (All clients who need to submit	yarn.resource-manager.admin.address

Admin Interface	any back-up Resource Manager node)				the YARN applications including Hive, Hive server, Pig)	
Resource Manager Scheduler	Master Nodes (Resource Manager and any back-up Resource Manager node)	8033		Administrative interface	Yes (Typically admins and support teams)	yarn.resource-manager.scheduler.address
NodeManager Web UI	All Slave Nodes	8031	http	Resource Manager interface	Yes (Typically admins, Dev/Support teams)	yarn.nodemanager.webapp.address

1.3.8.3. Hive Ports

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

Service	Servers	Default Ports Used	Protocol	Description	Needs End-user Access?	Configuration Parameters
HiveServer2	Hiveserver 2 machine (usually a utility machine)	10001	thrift	Service for programmatically (Thrift/JDBC) connecting to Hive	Yes	ENV Variable HIVE_PORT
HiveServer	HiveServer machine (usually a utility machine)	10000	thrift	Service for programmatically (Thrift/JDBC) connecting to Hive	Yes (Clients who need to connect to Hive either programmatically 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

1.3.8.4. WebHCat Ports

The following table lists the default ports used by the WebHCat service.

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

1.3.8.5. HBase Ports

The following table lists the default ports used by the HBase services.

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
HMaster Info WebUI	Master Nodes (HBase master Node and backup HBase Master node if any)	60010	http	The port for the HBaseMaster WebUI. 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
ZooKeeper	All ZooKeeper Nodes	2888		Port used by ZooKeeper peers to talk to each other. For further information, see here .	No	hbase.zookeeper.peerport
ZooKeeper	All ZooKeeper Nodes	3888		Port used by ZooKeeper for leader election. For further information, see here .		hbase.zookeeper.leaderport
		2181		Property from ZooKeeper's config <code>zoo.cfg</code> . The port at which the clients connect.		hbase.zookeeper.property.clientPort

1.3.9. Install Required Software

Install the following software on each node in the cluster:

- Python v2.7 or higher
- Java Development Kit version: 1.7.0_51
- Microsoft Visual C++, 2010 only
- Microsoft .NET Framework v4.0

See the following subsections for more information.

1.3.9.1. 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, the `WORKSPACE` environment variable refers to the full path of the workspace directory where the installer is located; for example:

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

After setting the environment variable from the command prompt using `setx`, restart PowerShell. Alternately, if you are using a script you might want to set `WORKSPACE` as a standard PowerShell 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

To manually install Python in your local environment:

1. Download [Python](#) into the workspace directory.
2. Install Python and update the `PATH` environment variable.

From the PowerShell window, as the Administrator, enter:

```
$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
```



Note

If the downloaded Python MSI name is different from `python-2_7_5_amd64.msi`, substitute the correct MSI file name.

- Microsoft Visual C++ 2010 Redistributable Package (64-bit)
 1. Download [Microsoft Visual C++ 2010 Redistributable Package \(64-bit\)](#) to the workspace directory.
 2. From PowerShell, as Administrator, enter:

```
& "$env:WORKSPACE\vc_redist_x64.exe" /q /norestart /log "$env:WORKSPACE\
C_2010_install.log"
```

- Microsoft .NET framework 4.0
 1. Download [Microsoft .NET Framework 4.0](#) to the workspace directory.
 2. From PowerShell, as Administrator, enter:

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

- JDK version 7
 1. Check which version of Java is currently installed. From a command shell or PowerShell window, enter:

```
java -version
```

If the JDK version is less than v1.6 update 31, uninstall the Java package.

2. Go to the [Oracle Java SE Downloads](#) page and download the JDK installer to the workspace directory.
3. From PowerShell, as Administrator, enter:

```
$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[space]Files is not allowed.

4. Verify that Java installed correctly and that the Java application directory is in your PATH environment variable.

From a command shell or PowerShell window, enter:

```
java -version
```

The system should display:

```
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)
```

1.3.9.2. Installing Required Software Manually

This section explains how to install the following software:

- Python
- Microsoft Visual C++ 2010 Redistributable Package (64 bit)
- Microsoft .NET framework 4.0
- Java JDK

Python

1. Download [Python](#) 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 add 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
```

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.

Oracle Java JDK

1. Download the [Oracle JDK](#) and install to a directory that contains no white space in the path, such as c:\Java.
2. Go to Control Panel > System and click Advanced system settings.
3. Click Advanced.
4. Click Environment Variables.
5. Add a new system environment variable, JAVA_HOME. The value for this variable should be the installation path for the Java Development Kit; for example, c:\Java\jdk1.7.0_51.
6. Click OK.
7. To validate the environment variable you just added, enter the following command at a command-line prompt:

```
echo %JAVA_HOME%
```

You should see the path you specified when you created JAVA_HOME:

```
c:\Java\jdk1.7.0_45\
```

8. As Administrator, update the PATH variable:
 - a. Under System Variables, find PATH. Click Edit.

- b. After the last entry in the Path value, enter a semi-colon and add the installation path to the JDK. For example:

```
...;c:\Java\jdk1.7.0_51\bin
```

- c. Click OK.

- d. To validate the change you just made, open a DOS CLI and enter:

```
java -version
```

DOS should return the expected Java version and details; for example, `java version "1.7.0"`.

9. Click OK to close the Environment Variables dialog box.

2. Defining Cluster Properties

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

After downloading the HDP installer, use one of the following methods to modify the cluster properties file:

- **Option 1:** Use the HDP Setup Interface to generate a cluster properties file for GUI to use, or to export a generated `clusterproperties.txt` file for a CLI installation. (Recommended for first-time users and single-node installations.)
- **Option 2:** Manually define cluster properties in a file. (Recommended for users who are familiar with their systems and with HDP requirements.)

2.1. Download 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-0.4.19.2.2.0.0-2060`
 - `gplcompression.dll`
 - `lzo2.dll`

2.2. Using the HDP Setup Interface

You can define cluster properties with the HDP Setup Interface or define them manually (described in the next subsection). If you use the HDP Setup Interface you can either export the configuration and use it to deploy HDP from the command line (or within a script), or you can start deployment from the Setup Interface itself.

To start the Setup Interface, enter the following command at the command prompt:

```
runas /user:administrator msixec /i "hdp-2.2.0.0.winpkg.msi"  
MSIUSEREALADMINDETECTION=1
```

The HDP Setup form displays.

(The following image shows the form with the Main components tab selected.)

2.3. Set HDP Properties

The top part of the form, which includes HDP directory, Log directory, Data directory, Namenode data directory and Datanode data directory, is filled in with default values. Customize these entries as needed, and note whether you are configuring a single- or multi-node installation.

1. Complete the fields at the top of the HDP Setup form:

Configuration Property Name	Description	Example Value	Mandatory/Optional/Conditional
HDP directory	HDP installation directory	d:\hdp	Mandatory
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:\hadoop\logs	Mandatory
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\data	Mandatory
Namenode data directory	Determines where on the local filesystem the HDFS name node should store the name table (fsimage). You can add multiple comma-separated data locations for multiple data directories.	d:\hdpdata\hdfs	Mandatory

Datanode data directory	Determines where on the local filesystem an HDFS data node should store its blocks. You can add multiple comma-separated data locations for multiple data directories.	d:\hdpdata\hdfs	Mandatory
-------------------------	--	-----------------	-----------

2. To choose single- or multi-node deployment, select one of the following:

- **Configure Single Node** – installs all cluster nodes on the current host; the hostname fields are pre-populated with the name of the current computer (see [Quick Start for Single Node Installation](#)).
- **Configure Multi Node** – creates a property file, which you can use for cluster deployment or to manually install a node (or subset of nodes) on the current computer.

3. Specify whether or not you want to delete existing HDP data.

If you want to delete existing HDP data, select `Delete existing HDP data` and supply the `hadoop` user password in the field immediately below. (You can either shield the password while entering it or select `Show` to show it.)

4. **Mandatory:** Enter the 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.

5. Specify component-related values:

Configuration Property Name	Description	Example Value	Mandatory/Optional/Conditional
NameNode Host	The FQDN for the cluster node that will run the NameNode master service.	NAMENODE_MASTER. acme.com	Mandatory
Secondary NameNode Host	The FQDN for the cluster node that will run the Secondary NameNode master service. (Not applicable for HA.)	SECONDARY_NN_MASTER. acme.com	Mandatory/NA
ResourceManager Host	The FQDN for the cluster node that will run the YARN Resource Manager master service.	RESOURCE_MANAGER. acme.com	Mandatory
Hive Server Host	The FQDN for the cluster node that will run the Hive Server master service.	HIVE_SERVER_MASTER. acme.com	Mandatory
Oozie Server Host	The FQDN for the cluster node that will run the Oozie Server master service.	OOZIE_SERVER_MASTER. acme.com	Mandatory
WebHCat Host	The FQDN for the cluster node that will run the WebHCat master service.	WEBHCAT_MASTER. acme.com	Mandatory

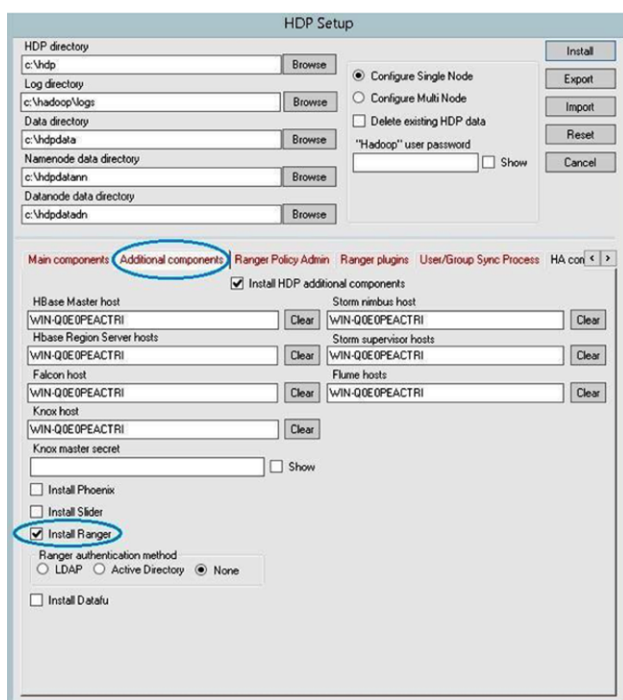
Slave hosts	A comma-separated list of FQDN for those cluster nodes that will run the DataNode and TaskTracker services.	slave1.acme.com, slave2.acme.com, slave3.acme.com	Mandatory
Client hosts	A comma-separated list of FQDN for those cluster nodes that will store JARs and other job-related files.	client.acme.com, client1.acme.com, client2.acme.com	Optional
Zookeeper Hosts	A comma-separated list of FQDN for those cluster nodes that will run the ZooKeeper hosts.	ZOOKEEPER-HOST.acme.com	Mandatory
Enable LZO codec	Use LZO compression for HDP.	Selected	Optional
Use Tez in Hive	Install Tez on the Hive host.	Selected	Optional
Enable GZip compression	Enable gzip file compression.	Selected	Optional
Install Oozie Webconsole	Install web-based console for Oozie.	Selected	Optional

6. Enter database information for Hive and Oozie at the bottom of the form:

Configuration Property Name	Description	Example Value	Mandatory/Optional
Hive DB Username	User account credentials for Hive metastore database instance. Ensure that this user account has appropriate permissions.	hive_user	Mandatory
Hive DB Password	User account credentials for Hive metastore database instance. Ensure that this user account has appropriate permissions.	hive_pass	Mandatory
Oozie DB Name	Database for Oozie metastore. If using SQL Server, ensure that you create the database on the SQL Server instance.	ooziedb	Mandatory
Oozie DB Username	User account credentials for Oozie metastore database instance. Ensure that this user account has appropriate permissions.	oozie_user	Mandatory
Oozie DB Password	User account credentials for Oozie metastore database instance. Ensure that this user account has appropriate permissions.	oozie_pass	Mandatory
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 or derby	Mandatory

Configuration Property Name	Description	Example Value	Mandatory/Optional
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.	sqlserver1.acme.com	Mandatory
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	Optional

7. Next, step through the other tabs in the HDP Setup form, specifying cluster property values. Click the `Additional components` tab.



8. Complete the following fields:

Configuration Property Name	Description	Example Value	Mandatory/Optional/Conditional
HBase Master host	The FQDN for the cluster node that runs the HBase master	HBASE-MASTER.acme.com	Mandatory
Storm nimbus host	The FQDN for the cluster node that runs the Storm Nimbus master service	storm-host.acme.com	Optional
HBase region Server hosts	A comma-separated list of FQDN for cluster nodes that run the HBase Region Server services	slave1.acme.com, slave2.acme.com, slave3.acme.com	Mandatory

Storm supervisor hosts	A comma-separated list of FQDN for those cluster nodes that run the Storm Supervisors.	storm-sup=host.acme.com	Optional
Falcon host	The FQDN for the cluster node that runs Falcon	falcon-host.acme.com	
Flume hosts	A comma-separated list of FQDN for cluster nodes that run Flume	flume-host.acme.com	Optional
Knox host	The FQDN for the cluster node that runs Knox	knox-host.acme.com	Mandatory
Knox Master secret	Password for starting and stopping the gateway	knox-secret	Mandatory
Install Phoenix	Install Phoenix on the HBase server	Selected	Optional
Install Slider	Install Slider platform services for the YARN environment	Selected	Optional
Install Ranger	Installs Ranger security	Selected	Optional
Ranger authentication method	Authentication method used by Ranger	Selected: LDAP, Active Directory, or None	Optional
Install Datafu	Install Datafu user-defined functions for data analysis	Selected	Optional

2.4. Set High Availability Properties

To ensure that a multi-node cluster remains available, configure and enable High Availability. The configuration process for 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 namenode High Availability, select the `HA components` tab. Define the locations and names of hosts in a cluster that are available to act as `JournalNodes` and the Resource Manager. Specify a standby namenode in case the primary namenode fails.

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

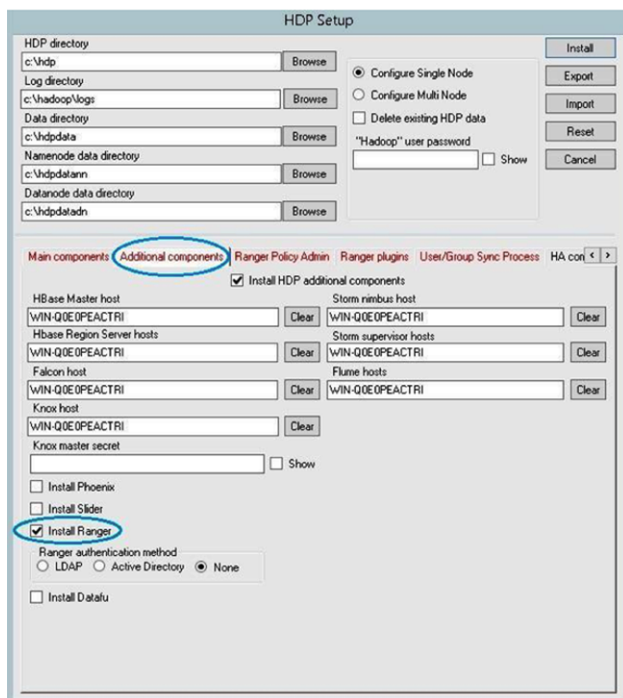
Configuration Property Name	Description	Example Value	Mandatory/Optional
Enable HA	Whether to deploy a highly available NameNode or not.	yes or no	Optional
NN Journal Node Hosts	A comma-separated list of FQDN for cluster nodes that will run the JournalNode processes.	journalnode1.acme.com, journalnode2.acme.com, 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 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	d:\hadoop\journal	Optional

Configuration Property Name	Description	Example Value	Mandatory/Optional
	use a single path for this configuration.		
NN Standby 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 Standby Host	The FQDN of the standby resource manager host.	rm-standby-host.acme.com	Optional
Enable Knox HA	Enable Knox High Availability mode	Selected	Optional
Enable Oozie HA	Enable Oozie High Availability mode	Selected	Optional
Oozie server hosts	A comma-separated list of FQDN for cluster nodes that will run Oozie servers.	ooziehost1.acme.com, ooziehost2.acme.com	Optional
Oozie loadbalancer url	URL for Oozie Load Balancer	http://oozie.lb.com:11000/oozie	Optional

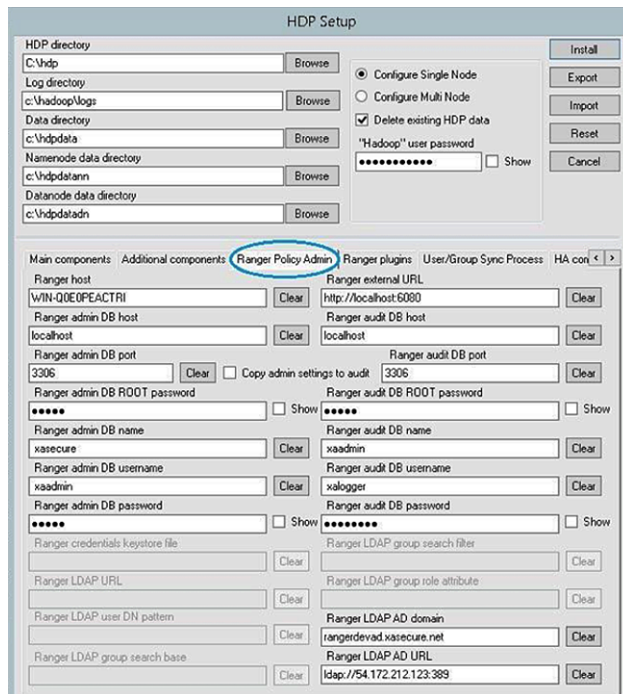
2.5. Set Ranger Properties

(Optional) To configure Ranger using the Setup GUI, complete the following steps.

1. Enable Ranger from the Additional components tab, and specify its authentication method.



2. Click the Ranger Policy Admin tab in the middle of the HDP Setup Form.

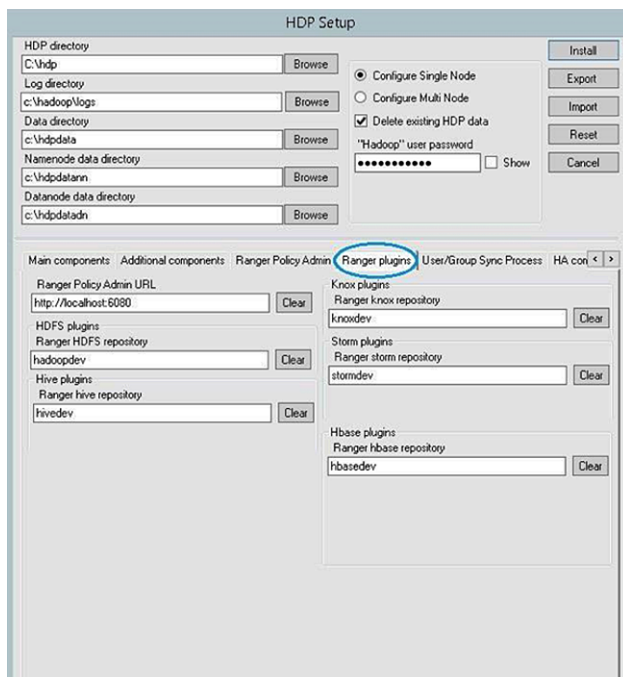


3. Enter host information, credentials for database saving policies, Admin user credentials, and Audit user credentials.

Configuration Property Name	Description	Example Value	Mandatory/Optional/Conditional
Ranger host	Hostname of the host where Ranger-Admin and Ranger-UserSync services will be installed	WIN-Q0E0PEACTR1	Mandatory
Ranger external URL	URL used for Ranger	localhost:8080	Optional
Ranger admin DB host	MySQL server instance for use by the Ranger Admin database host. (MySQL should be up and running at installation time.)	localhost	Mandatory
Ranger admin DB port	Port number for Ranger-Admin database server	3306	Mandatory
Ranger admin DB name	Ranger-Admin policy database name	ranger (default)	Mandatory
Ranger admin DB username	Ranger-Admin policy database username	rangeradmin (default)	Mandatory
Ranger admin DB password	Password for the Ranger admin DB user	RangerAdminPassW0Rd	Mandatory
Copy admin settings to audit	Use admin settings for audit database	Selected	
Ranger audit DB host	Host for Ranger Audit database. (MySQL should be up and running at installation time). This can be the same as the Ranger host, or you can specify a different server.	localhost	Mandatory

Ranger audit DB name	Ranger audit database name. This can be a different database in the same database server mentioned above.	ranger_audit (default)	Mandatory
Ranger audit DB port	Port number where Ranger-Admin runs audit service	3306	Mandatory
Ranger audit DB ROOT password	Database password for the Ranger audit DB username (required for audit database creation)	RangerAuditPassW0Rd	Mandatory
Ranger audit DB username	Database user that performs all audit logging operations from Ranger plugins	rangerlogger (default)	Mandatory
Ranger audit DB password	Database password for the Ranger audit DB username	RangerAuditPassW0Rd	Mandatory
Ranger LDAP AD domain	Active Directory Domain Name used for AD login	rangerad.net	Mandatory if authentication method is AD
Ranger LDAP AD URL	Active Directory LDAP URL for authentication of users	ldap://ad.rangerad.net:389	Mandatory if authentication method is AD

4. Click the Ranger Plugins tab in the middle of the HDP Setup Form.

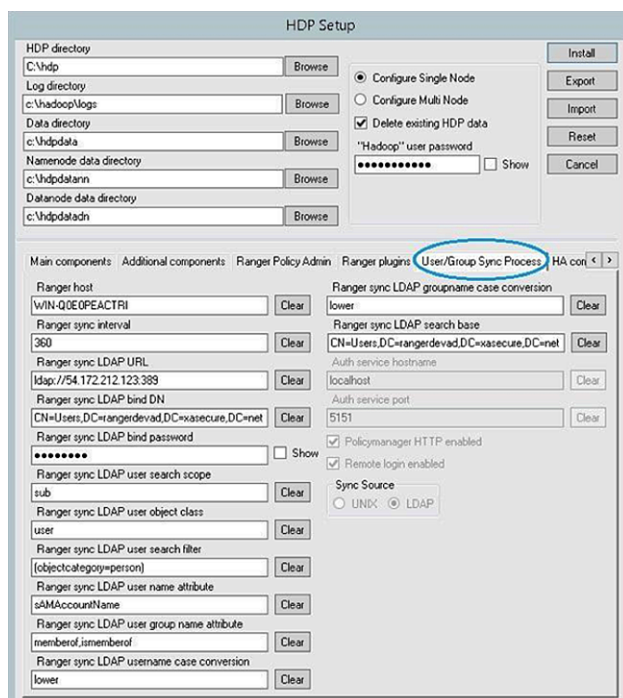


5. Complete the following fields. These allow communication between Ranger-Admin and each plugin.

Configuration Property Name	Description	Example Value	Mandatory/Optional/Conditional
Ranger Policy Admin URL	URL used within policy admin tool when a link to	localhost:6080	Optional

	its own page is generated in the policy admin tool website		
Knox agents: Ranger Knox repository	The repository name used in Policy Admin Tool for defining policies for Knox	knoxdev	Mandatory if using Ranger on Knox
HDFS agents: Ranger HDFS repository	The repository name used in Policy Admin Tool for defining policies for HDFS	hadoopdev	Mandatory if using Ranger on HDFS
Storm agents: Ranger storm repository	The repository name used in Policy Admin Tool for defining policies for Storm	stormdev	Mandatory if using Ranger on Storm
Hive agents: Ranger hive repository	The repository name used in Policy Admin Tool for defining policies for Hive	hivedev	Mandatory if using Ranger on Hive
HBase agents: Ranger hbase repository	The repository name used in Policy Admin Tool for defining policies for HBase	hbasedev	Mandatory if using Ranger on HBase

6. Click the **User/Group Sync Process** tab in the middle of the HDP Setup Form.



7. Complete the following fields.

- a. Add the Ranger-Admin host URL to Ranger User/Group Sync; this enables communication between Ranger-Admin and the User-Sync service.
- b. Set appropriate values for the other parameters based on sync source:
 - If users will be synchronized from an LDAP server, supply LDAP server credentials and all properties associated with synchronizing users and groups from the LDAP server.

- If users will be synchronized with an Active Directory, supply Active Directory credentials and all properties associated with synchronizing users and groups via Active Directory.

Configuration Property Name	Description	Example Value	Mandatory/Optional/Conditional
Ranger host	Hostname of the host where Ranger-Admin and Ranger-UserSync services will be installed	WIN-Q0EOPEACTR1	Mandatory
Ranger sync interval	Specifies the interval (in minutes) between synchronization cycles. Note: the second sync cycle will NOT start until the first sync cycle is complete.	5	Mandatory
Ranger sync LDAP search base	Search base for users	ou=users,dc=hadoop,dc=apache	Mandatory
Ranger sync LDAP URL	LDAP URL for synchronizing users	ldap://ldap.example.com:389	Mandatory
Ranger sync LDAP bind DN	LDAP bind DN used to connect to LDAP and query for users and group. This must be a user with admin privileges to search the directory for users/groups.	cn=admin,ou=users,dc=hadoop,dc=apache,dc=org	Mandatory
Ranger sync LDAP bind password	Password for the LDAP bind DN	LdapAdminPassW0rd	Mandatory
Ranger sync LDAP user search scope	Scope for user search	base, one, and sub are supported values	Mandatory
Ranger sync LDAP user object class	Object class to identify user entries	person (default)	Mandatory
Ranger sync LDAP user search filter	Additional filter constraining the users selected for syncing	[objectcategory=person]	Optional
Ranger sync LDAP user name attribute	Attribute from user entry that will be treated as username	cn (default)	Mandatory
Ranger sync LDAP user group name attribute	Attribute from user entry whose values will be treated as group values to be pushed into the Policy Manager database.	One or more attribute names separated by commas, such as: memberof,ismemberof	Mandatory
Ranger sync LDAP username case conversion	Convert all usernames to lowercase or uppercase	none: no conversion; keep as-is in SYNC_SOURCE. lower: (default) convert to lowercase when saving usernames to the Ranger database. upper: convert to uppercase when saving usernames to the Ranger db.	Mandatory
Ranger sync LDAP group name case conversion	Convert all groupnames to lowercase or uppercase	(same as username case conversion)	Mandatory

8. After specifying Ranger-UserSync properties, make sure that the following properties are defined on other tabs:

- On the Additional Components tab, set the Ranger authentication method to LDAP, Active Directory, or None, based on your synchronization source.
- On the Ranger Policy Admin tab, make sure that you have specified Authentication Properties.

2.6. Complete the GUI Installation Process

To continue with the GUI installation process, select `Install`.

To clear all fields and start over again, select `Reset`.

To export your HDP Setup configuration as a `clusterproperties.txt` file and switch to the CLI installation process, select `Export`. Export stops the GUI installation process and produces a `clusterproperties.txt` file based on your GUI fields. Before exporting, verify that all information is accurate.

2.7. Manually Creating a Cluster Properties File

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

1. Create a file for the cluster properties, or use the sample `clusterproperties.txt` file extracted from the HDP Installation zip file. You'll pass the name of the cluster properties file to the `msiexec` call when you install HDP. The following examples use the filename `clusterproperties.txt`.
2. Add the properties to the `clusterproperties.txt` file as described in the table below. As you add properties, keep in mind the following:
 - All properties in the cluster properties 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) to specify 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 cluster properties file, if the hosts are multi-homed or have multiple NIC cards, make sure that each name or IP address 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 Deploying HDP

Configuration Property Name	Description	Example Value	Mandatory/Optional
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:\hadoop\logs	Mandatory
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\data	Mandatory
HDFS_NAMENODE_DATA_DIR	Determines where on the local filesystem the HDFS name node should store the name table (fsimage). You can add multiple comma-separated data locations for multiple data directories.	d:\hadoop\data\hdfs\nn,c:\hdpdata,d:\hdpdatann	Mandatory
HDFS_DATANODE_DATA_DIR	Determines where on the local filesystem an HDFS data node should store its blocks. You can add multiple comma-separated data locations for multiple data directories.	d:\hadoop\data\hdfs\dn,c:\hdpdata,d:\hdpdatadn	Mandatory
NAMENODE_HOST	The FQDN for the cluster node that will run the NameNode master service.	NAMENODE-MASTER.acme.com	Mandatory
SECONDARY_NAMENODE_HOST	The FQDN for the cluster node that will run the Secondary NameNode master service.	SECONDARY-NN-MASTER.acme.com	Mandatory when no HA
RESOURCEMANAGER_HOST	The FQDN for the cluster node that will run the YARN Resource Manager master service.	RESOURCE-MANAGER.acme.com	Mandatory
HIVE_SERVER_HOST	The FQDN for the cluster node that will run the Hive Server master service.	HIVE-SERVER-MASTER.acme.com	Mandatory
OOZIE_SERVER_HOST	The FQDN for the cluster node that will run the Oozie Server master service.	OOZIE-SERVER-MASTER.acme.com	Mandatory
WEBHCAT_HOST	The FQDN for the cluster node that will run the WebHCat master service.	WEBHCAT-MASTER.acme.com	Mandatory
FLUME_HOSTS	A comma-separated list of FQDN for those cluster nodes that will run the Flume service.	FLUME-SERVICE1.acme.com, FLUME-SERVICE2.acme.com, FLUME-SERVICE3.acme.com	Mandatory
HBASE_MASTER	The FQDN for the cluster node that will run the HBase master.	HBASE-MASTER.acme.com	Mandatory
HBASE_REGIONSERVERS	A comma-separated list of FQDN for those cluster nodes that will run the HBase Region Server services.	slave1.acme.com, slave2.acme.com, slave3.acme.com	Mandatory

SLAVE_HOSTS	A comma-separated list of FQDN for those cluster nodes that will run the DataNode and TaskTracker services.	slave1.acme.com, slave2.acme.com, slave3.acme.com	Mandatory
ZOOKEEPER_HOSTS	A comma-separated list of FQDN for those cluster nodes that will run the ZooKeeper hosts.	ZOOKEEPER-HOST.acme.com	Optional
FALCON_HOST	A comma-separated list of FQDN for those cluster nodes that will run the Falcon hosts.	falcon.acme.com, falcon1.acme.com, falcon2.acme.com	Optional
KNOX_HOST	The FQDN of the Knox Gateway host.	KNOX-HOST.acme.com	Optional
STORM_SUPERVISORS		STORM-HOST.acme.com	
STORM_NIMBUS		STORM-HOST.acme.com	
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 or derby	Mandatory
DB_PORT	Port address, required only if you are using SQL Server for Hive and Oozie metastores.	1433 (default)	Optional
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.	sqlserver1.acme.com	Mandatory
HIVE_DB_NAME	Database for Hive metastore. If using SQL Server, ensure that you create the database on the SQL Server instance.	hivedb	Mandatory
HIVE_DB_USERNAME	User account credentials for Hive metastore database instance. Ensure that this user account has appropriate permissions.	hive_user	Mandatory
HIVE_DB_PASSWORD	User account credentials for Hive metastore database instance. Ensure that this user account has appropriate permissions.	hive_pass	Mandatory
OOZIE_DB_NAME	Database for Oozie metastore. If using SQL Server, ensure that you create the database on the SQL Server instance.	ooziedb	Mandatory
OOZIE_DB_USERNAME	User account credentials for Oozie metastore	oozie_user	Mandatory

	database instance. Ensure that this user account has appropriate permissions.		
OOZIE_DB_PASSWORD	User account credentials for Oozie metastore database instance. Ensure that this user account has appropriate permissions.	oozie_pass	Mandatory
DEFAULT_FS	Default file system.	HDFS	
RESOURCEMANAGER_HOST	Host used for Resource Manager		
IS_TEZ	Installs the Tez component on Hive host.	YES or NO	Optional
ENABLE_LZO	Enables the LZO codec for compression in HBase cells.	YES or NO	Optional
IS_PHOENIX	Installs Phoenix on the HBase hosts.	YES or NO	Optional
IS_HDFS_HA	Specify whether to enable High Availability for HDFS	YES or NO	Mandatory

Configuration Values: 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 your cluster properties file, and set their values as follows:



Note

To enable High Availability, you must also run several HA-specific commands when you start cluster services.

Configuration Property Name	Description	Example Value	Mandatory/Optional
HA	Whether to deploy a highly available NameNode or not.	yes or no	Optional
NN_HA_JOURNALNODE_HOSTS	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_EDIT_PATH	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	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_RESOURCE_MANAGER_HOST	Mandatory: Hostname of the standby resource manager host.	rm-standby-host.acme.com	Optional
-------------------------------------	---	--------------------------	----------

Configuration Values: Ranger

Note: "Mandatory" means that the property must be specified if Ranger is enabled.

Configuration Property Name	Description	Example Value	Mandatory/Optional/Conditional
RANGER_HOST	Hostname of the host where Ranger-Admin and Ranger-UserSync services will be installed	WIN-Q0E0PEACTR	Mandatory
RANGER_ADMIN_DB_HOST	MySQL server instance for use by the Ranger Admin database host. (MySQL should be up and running at installation time.)	localhost	Mandatory
RANGER_ADMIN_DB_PORT	Port number for Ranger-Admin database server	3306	Mandatory
RANGER_ADMIN_DB_ROOT_PASSWORD	Database root password (required for policy/audit database creation)	adm2	Mandatory
RANGER_ADMIN_DB_DBNAME	Ranger-Admin policy database name	ranger (default)	Mandatory
RANGER_ADMIN_DB_USERNAME	Ranger-Admin policy database username	rangeradmin (default)	Mandatory
RANGER_ADMIN_DB_PASSWORD	Password for the RANGER_ADMIN_DB_USERNAME user	RangerAdminPassW0Rd	Mandatory
RANGER_AUDIT_DB_HOST	Host for Ranger Audit database. (MySQL should be up and running at installation time). This can be the same as RANGER_ADMIN_DB_HOST or you can specify a different server.	localhost	Mandatory
RANGER_AUDIT_DB_PORT	Port number where Ranger-Admin runs audit service	3306	Mandatory
RANGER_AUDIT_DB_ROOT_PASSWORD	Database password for the RANGER_AUDIT_DB_USERNAME (required for audit database creation)	RangerAuditPassW0Rd	Mandatory
RANGER_EXTERNAL_URL	URL used for Ranger	localhost:8080	Optional
RANGER_AUDIT_DB_DBNAME	Ranger audit database name. This can be a different database in the same database server mentioned above.	ranger_audit (default)	Mandatory
RANGER_AUDIT_DB_USERNAME	Database user that performs all audit logging operations from Ranger plugins	rangerlogger (default)	Mandatory
RANGER_AUDIT_DB_PASSWORD	Database password for the RANGER_AUDIT_DB_USERNAME user	RangerAuditPassW0Rd	Mandatory
RANGER_AUTHENTICATION_METHOD	Authentication Method used to login into the Policy Admin Tool.	None: allows only users created within Policy Admin Tool (default) LDAP: allows	Mandatory

		users to be authenticated using Corporate LDAP. AD: allows users to be authenticated using a Active Directory.	
RANGER_LDAP_URL	URL for the LDAP service	ldap://71.127.43.33:386	Mandatory if authentication method is LDAP
RANGER_LDAP_USERDNPATTERN	LDAP DN pattern used to locate the login user (uniquely)	uid={0},ou=users,dc=ranger2,dc=net	Mandatory if authentication method is LDAP
RANGER_LDAP_GROUPSEARCHBASE	Defines the part of the LDAP directory tree under which group searches should be performed	ou=groups,dc=ranger2,dc=net	Mandatory if authentication method is LDAP
RANGER_LDAP_GROUPSEARCHFILTER	LDAP search filter used to retrieve groups for the login user	(member=uid={0},ou=users,dc=ranger2,dc=net)	Mandatory if authentication method is LDAP
RANGER_LDAP_GROUPROLEATTRIBUTE	Contains the name of the authority defined by the group entry, used to retrieve the group names from the group search filters	cn	Mandatory if authentication method is LDAP
RANGER_LDAP_AD_DOMAIN	Active Directory Domain Name used for AD login	rangerad.net	Mandatory if authentication method is Active Directory
RANGER_LDAP_AD_URL	Active Directory LDAP URL for authentication of user	ldap://ad.rangerad.net:389	Mandatory if authentication method is Active Directory
RANGER_POLICY_ADMIN_URL	URL used within policy admin tool when a link to its own page is generated in the policy admin tool website	localhost:6080	Optional
RANGER_HDFS_REPO	The repository name used in Policy Admin Tool for defining policies for HDFS	hadoopdev	Mandatory if using Ranger on HDFS
RANGER_HIVE_REPO	The repository name used in Policy Admin Tool for defining policies for Hive	hivedev	Mandatory if using Ranger on Hive
RANGER_HBASE_REPO	The repository name used in Policy Admin Tool for defining policies for HBase	hbasedev	Mandatory if using Ranger on HBase
RANGER_KNOX_REPO	The repository name used in Policy Admin Tool for defining policies for Knox	knoxdev	Mandatory if using Ranger on Knox
RANGER_STORM_REPO	The repository name used in Policy Admin Tool for defining policies for Storm	stormdev	Mandatory if using Ranger on Storm
RANGER_SYNC_INTERVAL	Specifies the interval (in minutes) between synchronization cycles. Note: the second sync cycle will NOT start until the first sync cycle is complete.	5	Mandatory
RANGER_SYNC_LDAP_URL	LDAP URL for synchronizing users	ldap://ldap.example.com:389	Mandatory
RANGER_SYNC_LDAP_BIND_DN	LDAP bind DN used to connect to LDAP and query for users and group. This must be a user with admin	cn=admin,ou=users,dc=hadoop,dc=apache,dc=org	Mandatory

	privileges to search the directory for users/groups.		
RANGER_SYNC_LDAP_BIND_PASSWORD	Password for the LDAP bind DN	LdapAdminPassW0rd	Mandatory
RANGER_SYNC_LDAP_USER_SEARCH_SCOPE	Scope for user search	base, one and sub are supported values	Mandatory
RANGER_SYNC_LDAP_USER_OBJECT_CLASS	Object class to identify user entries	person (default)	Mandatory
RANGER_SYNC_LDAP_USER_NAME_ATTRIBUTE	Attribute from user entry that will be treated as user name	cn (default)	Mandatory
RANGER_SYNC_LDAP_USER_GROUP_NAME_ATTRIBUTE	Attribute from user entry whose values will be treated as group values to be pushed into the Policy Manager database.	One or more attribute names separated by commas, such as: memberof,ismemberof	Mandatory
RANGER_SYNC_LDAP_USERNAME_CASE_CONVERSION	Convert all usernames to lowercase or uppercase	none: no conversion; keep as-is in SYNC_SOURCE. lower: (default) convert to lowercase when saving usernames to the Ranger database. upper: convert to uppercase when saving usernames to the Ranger db.	Mandatory
RANGER_SYNC_LDAP_GROUPNAME_CASE_CONVERSION	Convert all groupnames to lowercase or uppercase	(same as username case conversion property)	Mandatory
RANGER_SYNC_LDAP_USER_SEARCH_BASE	Search base for users	ou=users,dc=hadoop,dc=apache,dc=org	Mandatory
AUTHSERVICEHOSTNAME	Server Name (or IP address) where Ranger-Usersync module is running (along with Unix Authentication Service)	localhost (default)	Mandatory
AUTHSERVICEPORT	Port Number where Ranger-Usersync module is running the Unix Authentication Service	5151 (default)	Mandatory
POLICYMGR_HTTP_ENABLED	Flag to enable/disable HTTP protocol for downloading policies by Ranger plugin modules	true (default)	Mandatory
REMOTELOGINENABLED	Flag to enable/disable remote Login via Unix Authentication Mode	true (default)	Mandatory
SYNCSOURCE	Specifies where the user/group information is extracted to be put into ranger database.	LDAP	

Sample Cluster Properties File

The following snapshot illustrates a sample cluster properties file:

```
A Typical Hadoop Cluster.
#Log directory
HDP_LOG_DIR=d:\hadoop\logs
```

```
#Data directory
HDP_DATA_DIR=d:\hadoop\data
HDFS_NAMENODE_DATA_DIR=d:\hadoop\data\hdfs\nn,c:\hdpdata,d:\hdpdatann
HDFS_DATANODE_DATA_DIR=d:\hadoop\data\hdfs\dn,c:\hdpdata,d:\hdpdatadn

#Hosts
NAMENODE_HOST=onprem-ranger1
SECONDARY_NAMENODE_HOST=onprem-ranger1
HIVE_SERVER_HOST=onprem-ranger1
OOZIE_SERVER_HOST=onprem-ranger1
WEBHCAT_HOST=onprem-ranger1
FLUME_HOSTS=onprem-ranger1
HBASE_MASTER=onprem-ranger1
HBASE_REGIONSERVERS=onprem-ranger2
SLAVE_HOSTS=onprem-ranger2
ZOOKEEPER_HOSTS=onprem-ranger1
KNOX_HOST=onprem-ranger2
STORM_SUPERVISORS=onprem-ranger2
STORM_NIMBUS=onprem-ranger1

#Database host
DB_FLAVOR=mssql
DB_PORT=9433
DB_HOSTNAME=singlehcatms7.cloudapp.net

#Hive properties
HIVE_DB_NAME=onpremranger1hive
HIVE_DB_USERNAME=hive
HIVE_DB_PASSWORD=hive

#Oozie properties
OOZIE_DB_NAME=onpremranger1oozie
OOZIE_DB_USERNAME=oozie
OOZIE_DB_PASSWORD=oozie

#ASV/HDFS properties
DEFAULT_FS=HDFS
RESOURCEMANAGER_HOST=onprem-ranger1
IS_TEZ=yes
ENABLE_LZO=yes
RANGER_HOST=onprem-ranger1
RANGER_ADMIN_DB_HOST=localhost
RANGER_ADMIN_DB_PORT=3306
RANGER_ADMIN_DB_ROOT_PASSWORD=hcatest
RANGER_ADMIN_DB_DBNAME= xasecure
RANGER_ADMIN_DB_USERNAME= xaadmin
RANGER_ADMIN_DB_PASSWORD=admin
RANGER_AUDIT_DB_HOST=localhost
RANGER_AUDIT_DB_PORT=3306
RANGER_AUDIT_DB_ROOT_PASSWORD=hcatest
RANGER_EXTERNAL_URL=http://localhost:6080
RANGER_AUDIT_DB_DBNAME= xasecure
RANGER_AUDIT_DB_USERNAME= xalogger
RANGER_AUDIT_DB_PASSWORD=xalogger
RANGER_AUTHENTICATION_METHOD=LDAP
RANGER_LDAP_URL=ldap://71.127.43.33:389
RANGER_LDAP_USERDN_PATTERN=uid={0},ou=users,dc=xasecure,dc=net
RANGER_LDAP_GROUP_SEARCH_BASE=ou=groups,dc=xasecure,dc=net
RANGER_LDAP_GROUP_SEARCH_FILTER=(member=uid={0},ou=users,dc=xasecure,dc=net)
RANGER_LDAP_GROUP_PROLEAT_ATTRIBUTE=cn
```

```
RANGER_POLICY_ADMIN_URL=http://localhost:6080
RANGER_HDFS_REPO=hadoopdev
RANGER_HIVE_REPO=hivedev
RANGER_HBASE_REPO=hbasedev
RANGER_KNOX_REPO=knoxdev
RANGER_STORM_REPO=stormdev
RANGER_SYNC_INTERVAL=360
RANGER_SYNC_LDAP_URL=ldap://10.0.0.4:389
RANGER_SYNC_LDAP_BIND_DN=cn=Administrator,cn=users,dc=hwqe,dc=net
RANGER_SYNC_LDAP_BIND_PASSWORD=Horton!#%works
RANGER_SYNC_LDAP_USER_SEARCH_SCOPE=sub
RANGER_SYNC_LDAP_USER_OBJECT_CLASS=person
RANGER_SYNC_LDAP_USER_NAME_ATTRIBUTE=cn
RANGER_SYNC_LDAP_USER_GROUP_NAME_ATTRIBUTE=memberof,ismemberof
RANGER_SYNC_LDAP_USERNAME_CASE_CONVERSION=lower
RANGER_SYNC_LDAP_GROUPNAME_CASE_CONVERSION=lower
AUTHSERVICEHOSTNAME=localhost
AUTHSERVICEPORT=5151
RANGER_SYNC_LDAP_USER_SEARCH_BASE=cn=users,dc=hwqe,dc=net
POLICYMGR_HTTP_ENABLED=true
REMOTELOGINENABLED=true
SYNCSOURCE=LDAP
```

3. Quick Start for a Single Node HDP Installation

To deploy HDP on a single-node Windows Server machine:

1. **At the host, complete all the prerequisites**, as described in [Before You Begin](#).



Note

Do not install Java in a location that has spaces in the pathname. Before installation, set the `JAVA_HOME` environment variable.

2. **Prepare the single-node machine.**

- a. Configure the firewall. HDP uses multiple ports for communication with clients and between service components. If your corporate policies require maintaining a per-server firewall, you must enable the ports listed in *Configuring Ports*. 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 opens port 80 in the active Windows Firewall:

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

The following command opens 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
```

- b. If your network security policies allow you to open all the ports, use the instructions to [disable the Windows firewall](#).

3. **Install and start HDP.**

- a. Download the [HDP for Windows MSI file](#).

- b. At the command prompt, enter:

```
runas /user:administrator "cmd /C msixec /lv c:\hdplog.txt /i
PATH_to_MSI_file MSIUSERREALADMINDETECTION=1"
```

where `PATH_to_MSI_file` matches the location of the downloaded MSI file. For example:

```
runas /user:administrator "cmd /C msixec /lv c:\hdplog.txt /i c:\
MSI_INSTALL\hdp-2.2.0.0.winpkg.msi MSIUSERREALADMINDETECTION=1"
```

The HDP Setup window appears, pre-populated with the host name of the server and a set of default installation parameters. (The following image shows the **Main components** tab.)

c. Specify the following parameters in the HDP Setup window:

Mandatory parameters

- **Hadoop User Password:** enter the 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:** To use the default embedded database for the single-node HDP installation, choose Derby. (The MSSQL option requires a Microsoft SQL Server database deployed in your environment and available for use by the metastore.)

Optional parameters

- **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 checkbox removes all 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.



Important

Before selecting "Delete Existing HDP Data", make sure you wish to delete **all** existing data from prior HDP installs.

- d. **Install HDP Additional Components:** Select this checkbox to install ZooKeeper, Flume, Storm, Knox, or HBase as HDP services deployed to the single node server.



Note

When deploying HDP with LZO compression enabled, add the following three files (in the HDP for Windows Installation zip) to the directory that contains the HDP for Windows Installer and the cluster properties file:

- `hadoop-lzo-0.4.19.2.2.0.0-2060`
- `gplcompression.dll`
- `lzo2.dll`

4. Click `install` to install HDP.

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

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

When installation is complete, a confirmation window displays.



Warning

If you are reinstalling HDP and wish to delete existing data, but you did not select the **Delete existing HDP data** checkbox, you must format the HDFS file system. *Caution: Do not format the file system if you are upgrading an existing cluster and wish to preserve your data. Formatting will delete your existing HDP data.*

To format the HDFS file system and delete all existing HDP data, open the Hadoop Command Line shortcut on the Windows desktop and run the following command:

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

5. 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.) Enter:

```
%HADOOP_NODE%\start_local_hdp_services
```

Validating the Install

To verify that HDP services work as expected, run the provided smoke tests. These tests validate installed functionality by executing a set of tests for each HDP component.

1. Create a `smoketest` user directory in HDFS, if one does not already exist:

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

2. Run the provided smoke tests as the `hadoop` user:

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

(Alternately, create a `smoketest` user in HDFS as described in [Appendix: Adding a Smoketest User](#), then run the tests as the `smoketest` user.)

4. Deploying a Multi-node HDP Cluster

This section describes the HDP MSI Installer, and explains three different options for deploying a multinode Hadoop cluster from the command line or from a script. When installing HDP from the command line, the Hadoop setup script parses the cluster properties file and determines which services to install for each host.

This section contains the following information:

- [About the HDP MSI Installer and HDP Public Properties](#)
- [Option 1: Central Push Install Using Deployment Service](#)
- [Option 2: Central HDP Install Using a Push Install HDP Script](#)
- [Option 3: Installing HDP from the Command Line](#)

4.1. About the HDP MSI Installer and HDP Public Properties

This section describes the HDP MSI installer command line options and explains which HDP public properties to use when installing a multi-node Hadoop Cluster. The installation process runs in the background.

4.1.1. HDP MSI Installer command format

The HDP MSI Installer command includes the `msiexec` command, a set of standard installer options, and HDP public properties. For example:

```
msiexec /qn /lv log_file /i msi_file MSIUSERREALADMINDETECTION=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
RANGER=YES_or_NO
```

where:

`msiexec /qn /lv log_file /i msi_file MSIUSERREALADMINDETECTION=1` is the set of standard installer options recommended by Hortonworks.

Everything following `/i msi_file MSIUSERREALADMINDETECTION=1` is a public property.

4.1.2. Standard Installer Options

- `/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, the installation log file is created in the directory where `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 about standard `msiexec` options, enter `msiexec /?` in a command prompt.

4.1.3. HDP Public Properties

You can set the following properties when you run `msiexec`:

Property	Mandatory?	Value	Associated Value(s) in Cluster Properties file	Description
DESTROY_DATA	Y	Yes or No	none	Specify <code>No</code> to keep existing data from previous installation. <code>No</code> does not format the NameNode. Specify <code>Yes</code> to remove all existing or previous HDP data and format the NameNode, creating an installation with a clean data slate.
HDP_USER_PASSWORD	Y	<i>password</i>	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_LAYOUT	Y	<i>clusterproperties_full_path</i>	none	Absolute path to the Cluster Properties file. Note that relative paths are not supported and the path may not contain spaces. For example, <code>c:\MSI_Install\clusterproperties.txt</code> .
HDP_DIR	N	<code>install_dir</code>	none	Absolute path to the Hadoop root directory where HDP components are installed.
HDP	N	Yes or No		Setting this to <code>Yes</code> instructs the MSI to

Property	Mandatory?	Value	Associated Value(s) in Cluster Properties file	Description
				install optional HDP components such as Flume, HBase, Knox, Falcon and Storm. When enabled, you must specify the components on the command line; 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	N	Yes or No	FLUME_HOSTS	Includes the installation of Flume components on the hosts matching the name defined in the cluster properties file.
HBASE	N	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	N	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	N	<i>secret</i>	none	Specified only when KNOX="YES". The master secret to protect Knox security components, such as SSL certificates.
FALCON	N	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	N	Yes or No	STORM_NUMBER and STORM_SUPERVISORS	Includes the installation of the Storm components on the host matching the name defined in the cluster properties file.
RANGER	N	Yes or No	RANGER_HOST	Includes the installation of the

Property	Mandatory?	Value	Associated Value(s) in Cluster Properties file	Description
				Ranger Admin and User Sync components on the host matching the name defined in the cluster properties file.

For optional HDP Components such as Knox and Falcon, include `HDP=yes` and specify "yes" or "no" for the components you would like to install or not, respectively. For example: `FLUME=no HBASE=yes KNOX=no FALCON=no STORM=no`.

This command needs to run in the command-line interface of each node in your cluster. 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. For examples of `msiexec` commands, see [Installing HDP from the Command Line](#).

4.2. Option 1: 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 checklist:

1. Identify and configure the hosts for the Hadoop cluster nodes.
2. On the host nodes, complete all the prerequisites described in [Before You Begin](#). Make sure you set an environment variable for `JAVA_HOME`. (Remember, Java cannot be installed in a location where the pathname includes spaces.)

Be especially careful to identify:

- 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
3. Download the [HDP Windows Installation package](#). This package includes a sample cluster properties file called `clusterproperties.txt`.
 4. Create (or edit) the cluster properties file using your host information. See [Define Hadoop 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 accessed 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. Use your standard procedures to push both the HDP Installer MSI and the custom cluster properties file to each node in the cluster.
6. Continue to use your standard procedures to execute the installation remotely, using the parameters documented in [About the HDP MSI Installer and HDP Public Properties](#). For examples of `msiexec` commands, see [Installing HDP from the Command Line](#).



Note

The HDP Installer unpacks the MSI contents to `SystemDrive\HadoopInstallFiles`. A detailed installation log is located at `SystemDrive\HadoopInstallFiles\HadoopSetupTools\hdp-2.2.0.0.winpkg.install`. Do not remove this folder; it is required for uninstalling HDP.

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

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

4.3. Option 2: Central HDP Install Using the Push Install HDP Script

Hortonworks provides a PowerShell script called `push_install_hdp.ps1`, which is included in the resources directory of the installer zip. The script 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 cluster. The script does not require shared storage, it copies the installation files to the target using the Windows Administrative Share.

Before running the script, 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 described in [Before You Begin](#). Make sure you set an environment variable for `JAVA_HOME`. (Remember, Java cannot be installed in a location where the pathname includes spaces.)

Be especially careful to identify:

- 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. Additionally, on each host:
 - a. Enable the Administrative Share:

```
netsh firewall set service type remoteadmin enabled
```
 - b. 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](#). The package includes a sample cluster properties file called `clusterproperties.txt`.
 4. Define your cluster properties and save them in a file; see [Manually Creating a Cluster Properties File](#).



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 accessed 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. Copy the HDP MSI Installer, your custom cluster properties file, and `push_install_hdp.ps1` 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. For information about parameters, see "About the HDP MSI Installer and HDP Public Properties". For examples of `msiexec` commands, see "Installing HDP from the Command Line".
7. On the master install node, open a command prompt with `run as Administrator`, and enter:

```
cd source_pathpowershell
-File push_install_hdp.ps1
source_path
destination_path
clusterproperties_file
files_list
skip m
siexec_command
-parallel
```


where:

- `source_path` is the absolute path to the installation files. This directory must contain the HDP MSI and the cluster properties file, as well as any other files the installer will push to the cluster nodes; for example, `D:\MSI_Install`.
- `destination_path` is the absolute path to an existing directory on the target cluster nodes. All nodes must have this directory. The installer copies `files_list` from `source_path` to `destination_path`. Specify destination path as a local path on the target host; for example, `D:\MSI_Install`.
- `clusterproperties_file` is the name of your custom cluster properties file; for example, `clusterproperties.txt`. (Do NOT include the path to the file.)
- `files_list` is a comma-delimited list of filenames that the installer copies from `source_path` to all cluster hosts.

The list must contain both the cluster property and HDP Installer file names; for example, `hdp-2.2.0.0.winpkg.msi,cluster.properties`. The list cannot 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 (from the Windows Installation zip) into the directory that contains the HDP for Windows Installer, and the cluster.properties file, and include them in the file list:

- `hadoop-lzo-0.4.19.2.2.0.0-2060`
 - `gplcompression.dll`
 - `lzo2.dll`
- `msiexec_command` is the complete installation command that the script executes on the target nodes.

The installer script returns error messages or successful results to the Install Master host. These messages are displayed when the script finishes. Examine these results to ensure that all nodes were successfully installed.

On each node, the HDP Installer unpacks the MSI contents to `SystemDrive\HadoopInstallFiles\HadoopSetupTools\hdp-2.2.0.0.winpkg.install`. This folder is required to uninstall HDP; do not remove it.

4.4. Option 3: Installing HDP from the Command Line

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

1. Complete all the prerequisites described in [Before You Begin](#). Be especially careful to identify:
 - 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
 - Set an environment variable for JAVA_HOME. Java cannot be installed in a location where the path name includes spaces.
2. Download the [HDP Windows Installation package](#). This package includes a sample cluster properties file, called `clusterproperties.txt`.
3. Create a cluster properties file using your host information, See [Define Hadoop Cluster Properties](#).



Note

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 accessed 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.

4. Place the MSI and custom cluster properties file in a local subdirectory on the host. Only the Hadoop Services that match the system's host name in the cluster properties file will be installed.
5. Use the same cluster properties file on every node in the cluster.
6. **(Optional)** When installing HDP with HDFS compression enabled, put the following three files (from the HDP for Windows Installation zip) into the directory that contains the HDP for Windows Installer and the cluster properties file:
 - `hadoop-lzo-0.4.19.2.2.0.0-2060`
 - `gplcompression.dll`
 - `lzo2.dll`
7. The following two examples show `msiexec` commands with HDP parameters.



Warning

These examples assume that you would like to destroy any existing HDP data. If you have existing data that you wish to keep, set `DESTROY_DATA` to `no`.

Open a command prompt with the `run as Administrator` option, and enter the following.

```
msiexec /qn /i c:\MSI_Download\hdp-2.2.0.0\hdp-2.2.0.0\hdp-2.2.0.0.winpkg.msi
/lv c:\MSI_Download\hdp.log
HDP_LAYOUT=c:\MSI_Download\Cluster.properties
HDP_DIR=c:\hdp
HDP=yes
DESTROY_DATA=yes
USEROOT=yes
HDP_USER_PASSWORD=TestUser123
KNOX=no
FALCON=no
STORM=no
RANGER="YesorNo"
KNOX_MASTER_SECRET=password
```

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

```
msiexec /qn /i D:\MSI_Install\hdp-2.2.0.0.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=password
HDP=yes
KNOX=no
FALCON=no
HBase=yes
STORM=no
FLUME=no
RANGER=No
KNOX_MASTER_SECRET=password
```

For a description of command line options, see [Standard Installer Options](#).

8. The HDP Installer unpacks the MSI contents to `SystemDrive\HadoopInstallFiles`. A detailed installation log is located at `SystemDrive\HadoopInstallFiles\HadoopSetupTools\hdp-2.2.0.0.winpkg.install`. This folder is required to uninstall HDP; do not remove it.



Warning

If you are reinstalling HDP and wish to delete existing data but you did not specify `DESTROY_DATA=YES`, you will need to format the HDFS file system. *Do not format the file system if you are upgrading an existing cluster and wish to preserve your data. Formatting will delete your existing HDP data.*

To format the HDFS file system, open the Hadoop Command Line shortcut on the Windows desktop, and then enter:

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

5. Configuring HDP Components and Services

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

Use one of the methods in [Defining Hadoop Cluster Properties](#) to modify the cluster properties file. When you are finishing modifying cluster properties, start HDP services.

This chapter contains the following sections:

- [Enable HDP Services](#)
- [\(MS SQL Only\): Configure Hive when Metastore DB is in a Named Instance](#)
- [Configure MapReduce on HDFS](#)
- [Configure HBase on HDFS](#)
- [Configure Hive on HDFS](#)
- [Configure Tez for Hive](#)
- [Configure Node Label Support for YARN Applications](#)
- [Configure Ranger Security](#)
- [Using Apache Slider](#)
- [\(Optional\) Install Microsoft SQL Server JDBC Driver](#)
- [Start HDP Services](#)

5.1. Enable HDP Services

By default the following HDP services are disabled:

- Apache Falcon
- Apache Flume agent
- Apache Knox REST API
- Apache Hadoop thrift or Apache Hadoop thrift2

To enable these services to start and stop using the Start Local or Remote HDP script, first enable them **in the following order**. Note that the `sc config` command requires a space between the `start` option and its value.

1. Enable Thrift or Thrift2 on a cluster node.



Note

Thrift and Thrift2 use the same port, so they cannot run at the same time.

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

OR

```
sc config thrift2 start= demand
```

2. Enable Apache 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. (MS SQL Only): Configure Hive when Metastore DB is in a Named Instance

If your site uses MS SQL for the Hive metadata store and the Hive database is not in the default instance (that is, in a named instance), you must configure the connection string after the installation completes:

1. On the Hive host, open the `hive-site.xml` file 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:

- `sql-host` is the SQLServer host name
- `instance-name` is the name of the instance that the Hive database is in
- `hive_db` is the name of the Hive database

3. Save the changes to `hive-site.xml`.
4. Finish configuring Hive as described later in this chapter, before restarting the Apache Hadoop Hive service.

5.3. Configure MapReduce on HDFS

To use MapReduce, create the MapReduce history folder, `tmp` directory, application logs, and a YARN folder in HDFS. Then set folder permissions:

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

5.4. Configure HBase on HDFS

To use HBase, create the HBase application data folder, and then set folder permissions:

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

5.5. Configure Hive on HDFS

To use Hive, create the Hive warehouse directory, the Hive and WebHcat user directories, and the WebHcat application folder in HDFS. Then set directory permissions so all users can access them:

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:

```
%HADOOP_HOME%\bin\hdfs dfs -mkdir -p /user/hive /hive/warehouse
```

3. Make a user and application directory for WebHcat:

```
%HADOOP_HOME%\bin\hdfs dfs -mkdir -p /user/hcat
%HADOOP_HOME%\bin\hdfs dfs -mkdir -p /apps/webhcat
```

4. Change the directory owner and permissions:

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

5.6. Configure Tez for Hive

If your cluster properties file specifies `IS_TEZ=yes` (use Tez for Hive), perform the following steps after deployment, using 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\hdfs dfs -mkdir /apps/tez
```

3. Allow all users read and write access:

```
%HADOOP_HOME%\bin\hdfs dfs -chmod -R 755 /apps/tez
```

4. Change the owner of the file to `hadoop`:

```
%HADOOP_HOME%\bin\hdfs dfs -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\hdfs dfs -put %TEZ_HOME%\* /apps/tez
```

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

```
%HADOOP_HOME%\bin\hdfs dfs -rm -r -skipTrash /apps/tez/conf
```

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

Property	Default Value	Description
<code>hive.auto.convert.join.noconditionaltask</code>	<code>true</code>	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 <code>hive.auto.convert.join.noconditionaltask.size</code> property.
<code>hive.auto.convert.join.noconditionaltask.size</code>	<code>10000000 (10 MB)</code>	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 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.

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

- To verify that the installation process succeeded, run smoke tests for Tez and Hive.

5.7. Configure Node Label Support for YARN Applications

Node labels can be used to restrict YARN applications so that the applications run only on cluster nodes that have a specified node label. Node Labels are supported on Windows. To enable node label support, make the following changes. See the Linux Node Labels documentation for more information.

If you do not plan to use node labels, none of these changes are needed.

- Open the command prompt using the `hadoop` account:

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

- Create a top-level YARN application directory in HDFS:

```
%HADOOP_HOME%\bin\hdfs dfs -mkdir -p /system/yarn/node-labels
```

- Make sure permissions are set for write access from the `hadoop` account (`rwX` for all the directories in the path).

- Change the owner of the file to `hadoop`:

```
%HADOOP_HOME%\bin\hdfs dfs -chown -R hadoop:users /system/yarn
%HADOOP_HOME%\bin\hdfs dfs -chmod -R 700 /system/yarn
```

- Add the following property values to `yarn-site.xml`:

Property	Value
----------	-------

yarn.node-labels.manager-class	org.apache.hadoop.yarn.server.resourcemanager.nodelabels.RMNodeLabelsManager
yarn.node-labels.fs-store.root-dir	/system/yarn/node-labels
yarn.node-labels.fs-store.retry-policy-spec	2000,500

6. To verify that the installation process succeeded, run smoke tests as described in "Validate the Installation".

5.8. Configure Ranger Security

Apache Ranger delivers a comprehensive approach to security for a Hadoop cluster. It provides central security policy administration across the core enterprise security requirements of authorization, accounting, and data protection.

The Ranger Policy Manager and Ranger UserSync are installed in only one host (specified in the HDP Setup Ranger Host parameter); the Ranger plug-ins for corresponding components are installed wherever those components are installed.

Make sure that the MySQL database used by Ranger is set up to connect from any host in the cluster. For multiple hosts, set up the Ranger MySQL database to connect from any host in the cluster using the root password. To do this, enter:

```
grant all privileges on *.* to 'root'@'%' identified by
'RootPasswordHere' ;flush privileges;
```



Note

In HDP v2.2, MySQL is the only database supported for use with Ranger.

5.9. Using Apache Slider

Apache Slider lets you deploy distributed applications across a Hadoop cluster. On the Windows platform, Slider application packages are included in the Windows Installer MSI for HBase and Storm. (Accumulo is supported on Linux, but is not currently supported on Windows). See the Linux document [Running Applications on YARN Using Slider](#).

5.10. (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 is extracted at `c:\Users\Administrator\Downloads\Microsoft SQL Server JDBC Driver 3.0.`)

3. Copy and paste:

```
c:\Users\Administrator\Downloads\Microsoft SQL Server JDBC Driver 3.0\sqljdbc_3.0\enu\sqljdbc4.jar
```

To HIVE_HOME/lib

Where HIVE_HOME is set to D:\hadoop\hive-0.9.0).

5.11. Start HDP Services

Following are steps for starting local and remote services. For more information, see [Managing HDP on Windows](#).

1. Start local services on the Master Nodes:

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

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

2. At any Master Node, start all slave node services:

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

3. At the Knox Gateway:

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

4. Smoke test your installation as described in [Validate the Installation](#).

6. Validating the Installation

After the HDP Cluster installation is completed, run the provided smoke tests to validate the installation. These tests validate installed functionality by executing a set of tests for each HDP component.

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

```
%HADOOP_HOME%\bin\hdfs dfs -mkdir -p /user/smoketestuser
%HADOOP_HOME%\bin\hdfs dfs -chown -R smoketestuser
```

2. Open a command prompt and run the smoke tests as the **hadoop** user:

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

You can also create a smoketest user in HDFS as described in [Appendix:Adding a Smoketest User](#) to this document, then run the tests as the smoketest user.

(Optional) If you installed Ranger, verify that the installation was successful using any or all of the following checks:

1. Check whether the Database `RANGER_ADMIN_DB_NAME` is present in the MySQL server running on `RANGER_ADMIN_DB_HOST`
2. Check whether the Database `RANGER_AUDIT_DB_NAME` is present in the MySQL server running on `RANGER_AUDIT_DB_HOST`
3. Check whether the "ranger-admin" service is installed in `services.msc`
4. Check whether the `ranger-usersync` service is installed in `services.msc`
5. If you plan to use the Ranger Administration Console with the UserSync feature, check whether both services start.
6. Go to the Ranger Administration Console host URL and make sure you can log in using the default user credentials.



Important

If you see installation failures for any HDP component, we recommend that you reinstall HDP.

7. Upgrading HDP Manually

This chapter describes how to upgrade an HDP Windows cluster from HDP 1.3, 2.0, or 2.1 to HDP 2.2. This is an “in-place” upgrade, where your data and metadata do not need to be moved during the upgrade process. However, services must be stopped and reinstalled.

The upgrade instructions in this section apply only to HDP clusters that are not configured for high availability.

All of the instructions in this section are executed at the MS-DOS command prompt.

7.1. Before You Begin

To prepare for upgrade, gather the following information:

- **Cluster Properties:** Save a copy of your existing cluster properties file (for example, `c:\hdp\cluster.properties`) as a starting point for configuring new HDP components.
- **Files required for uninstalling HDP:** Confirm that the uninstallation packages are available on each node in the cluster. The uninstallation package is in `c:\HadoopInstallFiles`. Without this package, the Uninstaller cannot remove HDP from the node.
- **HDP data directory:** Identify where user data and metadata are 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 new service components (such as Falcon, Storm, Knox, and Ranger), and ensure that the hosts meet all the prerequisites for the new components.
- **Hadoop user password:** You must run some of the upgrade and configuration steps with the `hadoop` user, so you must know the password. If you do not know the password for the `hadoop` user, then reset the password to a known password and continue. To change the password, enter:

```
net user hadoop NewPassword
```

7.2. Back Up Critical HDFS Metadata

Back up the following critical data before attempting an upgrade.

1. On the node that hosts the NameNode, open the Hadoop Command Line shortcut (or open a command window in the Hadoop directory). As the `hadoop` user, go to the HDFS home directory:

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

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:

- If you are upgrading from HDP 1.3:

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

- If you are upgrading from a more recent release:

```
hdfs dfs -ls -R / > 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 the `fsck` command:

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

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

6. Save the HDFS namespace:

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

```
hdfs dfsadmin -safemode enter
```

- b. Save the namespace.

```
hdfs dfsadmin -saveNamespace
```



Warning

From this point on, HDFS should not accept any new writes. Stay in safe mode!

- c. Finalize the namespace:

```
hdfs namenode -finalize
```

- d. 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. Back Up Your Configuration Files

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

- %HDFS_HOME%\conf
- %FLUME_HOME%\conf
- %HBASE_CONF_DIR%\conf
- %HCATALOG_HOME%\conf

7.4. Stop Running HDP Services

Stop all running services. From any node in the cluster, enter the following commands to stop the remote services and then stop the local services:

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

7.5. Uninstall HDP on All Nodes

At each cluster node, uninstall HDP for Windows using either of the following methods:

- **From the Command line:** If you have the MSI installer that goes with the version of HDP running at your site (for example, to install HDP 1.3, you need `hdp-1.3.0.0.winpkg.msi`), you can uninstall HDP from the command prompt. At each node in the cluster, open the command prompt as an administrator and enter:

```
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
```

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

The 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 your current version to form the basis of your new cluster properties file.

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

- **Upgrading from HDP 1.3 to HDP 2.2:**
 1. Change `JOBTRACKER_HOST` to `RESOURCEMANAGER_HOST` (leave the definition the same).
 2. **(Optional:)** Add a definition for `CLIENT_HOSTS`.
 3. **(Optional:)** Add a definition for `KNOX_HOST`.
 4. **(Optional:)** Add a definition for `STORM_NIMBUS`.

5. **(Optional:)** Add a definition for `STORM_SUPERVISORS`.
 6. **(Optional:)** Add a definition for `FALCON_HOSTS`.
 7. *If you are upgrading a High Availability cluster, change `HA_NAMENODE_HOST` to `NN_HA_STANDBY_NAMENODE_HOST`.*
 8. Add a definition for `DB_PORT`. (Default port for derby is 1527.)
 9. (Required) Add a definition for `IS_TEZ`.
 10. (Required) Add a definition for `IS_PHOENIX`.
 11. (Required) Add a definition for `HDFS_NAMENODE_DATA_DIR`. (Default is `c:\hdpdata\hdfs`.)
 12. (Required) Add a definition for `HDFS_DATANODE_DATA_DIR`. (Default is `c:\hdpdata\hdfs`.)
 13. (Required) Add a definition for `IS_HDFS_HA`. (Default is "no".)
 14. (Required) If you are using `msiexec` to install HDP 2.2, there are two new mandatory parameters: `RANGER` (default = "no") and `KNOX_MASTER_SECRET` (user-supplied password). If you plan to use Ranger, see "Manually Creating a Cluster Properties File" for a list of Ranger properties.
- **Upgrading from HDP 2.0 to HDP 2.2:**
 1. **(Optional:)** Add a definition for `KNOX_HOST`.
 2. **(Optional:)** Add a definition for `STORM_NIMBUS`.
 3. **(Optional:)** Add a definition for `STORM_SUPERVISORS`.
 4. **(Optional:)** Add a definition for `FALCON_HOSTS`.
 5. Add a definition for `DB_PORT`. (Default port for derby is 1527.)
 6. (Required) Add a definition for `IS_TEZ`.
 7. (Required) Add a definition for `IS_PHOENIX`.
 8. For High Availability clusters, update the HA properties as follows:
 - a. Change `HA_NAMENODE_HOST` to `NN_HA_STANDBY_NAMENODE_HOST`.
 - b. Change `HA_JOURNALNODE_HOSTS` to `NN_HA_JOURNALNODE_HOSTS`.
 - c. Change `HA_CLUSTER_NAME` to `NN_HA_CLUSTER_NAME`.
 - d. Change `HA_JOURNALNODE_EDITS_DIR` to `NN_HA_JOURNALNODE_EDITS_DIR`.
 9. (Required) Add a definition for `HDFS_NAMENODE_DATA_DIR`. (Default is `c:\hdpdata\hdfs`.)
-

10.(Required) Add a definition for `HDFS_DATANODE_DATA_DIR`. (Default is `c:\hdpdata\hdfs`.)

11.(Required) Add a definition for `IS_HDFS_HA`. (Default is "No".)

12.(Required) If you are using `msiexec` to install HDP 2.2, there are two new mandatory parameters: `RANGER` (default = "no") and `KNOX_MASTER_SECRET` (user-supplied password). If you plan to use Ranger, see "Manually Creating a Cluster Properties File" for a list of Ranger properties.

- **Upgrading from HDP 2.1 to HDP 2.2**

1. (Required) Add a definition for `HDFS_NAMENODE_DATA_DIR`. (Default is `c:\hdpdata\hdfs`.)

2. (Required) Add a definition for `HDFS_DATANODE_DATA_DIR`. (Default is `c:\hdpdata\hdfs`.) Note that NameNode and DataNode service startup might fail if the new `datanode` and `namenode` directories do not have **hadoop** user permissions. This could happen if you restore HDFS metadata from a backup. To avoid this issue, add the original `datanode/namenode` path to the cluster properties file before upgrading.

3. (Required) Add a definition for `IS_HDFS_HA`. (Default is "No".)

4. (Required) If you are using `msiexec` to install HDP 2.2, there are two new mandatory parameters: `RANGER` (default = "no") and `KNOX_MASTER_SECRET` "Manually Creating a Cluster Properties File" for a list of Ranger properties.

Save the new cluster properties file for use with the installer.

7.7. Installing HDP 2.2 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](#).
2. Copy the installer and the new cluster properties file to all nodes of the cluster.
3. Run the following command on each node in the cluster:

```
msiexec /qb /i "c:\MSI_INSTALL\hdp-2.2.0.0.winpkg.msi"  
/lv "hdp.log"  
HDP_LAYOUT="c:\MSI_INSTALL\clusterproperties.txt"  
HDP_DIR="c:\hdp"  
HDP_USER_PASSWORD=password  
DESTROY_DATA="no"  
HDP="yes"  
RANGER="YesorNo"  
KNOX_MASTER_SECRET=password
```




Note

Be aware that new command line properties have been added to support optional HDP components. For more information, see [Update the HDP Cluster Properties File](#) and [About the MSI Installer and HDP Public Properties](#).

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. Normally Hive metadata is stored in SQL Server.

To upgrade an existing MS SQL database for Hive, enter the following command in the Microsoft SQL Server process that contains the Hive database instance:

```
%HIVE_HOME%\scripts\metastore\upgrade\mssql\hive-txn-schema-0.13.0.mssql.sql
```



Note

If you use a new database name and define new users, you must add this new information into the cluster properties file that you use to upgrade to HDP 2.2.

7.9. Upgrade HDFS Metadata

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

1. Run the NameNode upgrade as the `hadoop` user. Note: in High Availability cluster, JournalNodes must be started before running `hdfs namenode -upgrade`.

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

2. At each DataNode, 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 window and connect to the NameNode on port 50070 (<http://namenode-host:50070>).

The amount of time it takes to upgrade HDFS depends upon the amount of data and number of nodes in your environment.

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 control-c to end the NameNode upgrade process.
6. Start the NameNode service:

```
sc start namenode
```

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

7.10. Upgrade HBase

To upgrade HBase, run the following commands as the `hadoop` user at 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. Before you upgrade, install the HBase 0.96 binaries on a separate host using the `hbase-site.xml` configuration file from the running HBase 0.94 binaries. Then check for V1-format files as follows:

- a. Download the 0.94.24+ HBase tarball in a machine.
- b. Point its configuration file to a 0.94 cluster.
- c. Check for HFiles in V1 format:

```
./bin/hbase org.apache.hadoop.hbase.util.HFileV1Detector -p hbase-root-data-path
```

- d. In the results, if "Count of HFileV1" returns any files, start the HBase shell and use major compaction for regions that have HFileV1 format.

3. Upgrade HBase.

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

If the command completes successfully, you should see a completed Znode upgrade with no errors.

4. Start all REST HDP services.

```
sc start rest
```

5. If the Apache Hadoop REST service is disabled, run the following command to enable it:

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

7.11. Upgrade Oozie

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

1. Run `ooziedb.cmd`:

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

2. Replace your configuration: At each server and client, copy `oozie\conf` from the backup to the `oozie\conf` directory.
3. Replace the contents of `/user/hdfs/share` in HDFS. At 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. Add them back after the share folder is updated.

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

- b. Add the latest share libraries.

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

- c. Start the Oozie service:

```
sc start oozie
```

7.12. Upgrade Tez

If your cluster properties file specifies `IS_TEZ=yes` (use Tez for Hive), perform the following steps using 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\hdfs dfs -mkdir /apps/tez
```

3. Allow all users read and write access:

```
%HADOOP_HOME%\bin\hdfs dfs -chmod -R 755 /apps/tez
```

4. Change the owner of the file to `hadoop`:

```
%HADOOP_HOME%\bin\hdfs dfs -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\hdfs dfs -put %TEZ_HOME%\* /apps/tez
```

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

```
%HADOOP_HOME%\bin\hdfs dfs -rm -r -skipTrash /apps/tez/conf
```

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

Property	Default Value	Description
<code>hive.auto.convert.join.noconditionaltask</code>	<code>true</code>	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 <code>hive.auto.convert.join.noconditionaltask.size</code> property.
<code>hive.auto.convert.join.noconditionaltask.size</code>	<code>10000000 (10 MB)</code>	Specifies the size used to calculate whether Hive converts a JOIN statement into a MAPJOIN statement. The configuration property is ignored unless <code>hive.auto.convert.join.noconditionaltask</code> is enabled.
<code>hive.optimize.reducededuplication.min.reducer</code>	<code>1</code>	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 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.
<code>hive.tez.container.size</code>	<code>-1</code>	By default, Tez uses the java options from map tasks. Use this property to override that value. Assigned value must match value specified for <code>mapreduce.map.child.java.opts</code> .
<code>hive.tez.java.opts</code>	<code>n/a</code>	Set to the same value as <code>mapreduce.map.java.opts</code> .

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

8. To verify that the installation process succeeded, run smoke tests for Tez and Hive.

7.13. Adding Node Label Support for YARN Applications

Node labels can be used to restrict YARN applications so that they run only on cluster nodes that have a specified node label. Node Labels are supported on Windows. To enable node label support, make the following changes. See the Linux Node Labels documentation for more information. If you do not plan to use node labels, none of these changes are needed.

1. Open the command prompt using the `hadoop` account:

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

2. Create a top-level YARN application directory in HDFS:

```
%HADOOP_HOME%\bin\hdfs dfs -mkdir /system/yarn/node-labels
```

3. Make sure permissions are set for write access from the `hadoop` account (rwx for all the directories in the path).

4. Change the owner of the file to `hadoop`:

```
%HADOOP_HOME%\bin\hdfs dfs -chown -R hadoop:users /apps/tez
```

5. Add the following property values to `yarn-site.xml`:

Property	Value
<code>yarn.node-labels.manager-class</code>	<code>org.apache.hadoop.yarn.server.resourcemanager.nodelabels.RMNodeLabelsM</code>
<code>yarn.node-labels.fs-store.root-dir</code>	<code>/system/yarn/node-labels</code>
<code>yarn.node-labels.fs-store.retry-policy-spec</code>	<code>2000,500</code>

6. To verify that the installation process succeeded, run smoke tests for Tez and Hive as described in [Validating the Installation](#).

7.14. Configure Ranger Security

Apache Ranger delivers a comprehensive approach to security for a Hadoop cluster. It provides central security policy administration across the core enterprise security requirements of authorization, accounting, and data protection. For more information about requirements and settings, see [Minimum System Requirements](#) and [Define Cluster Properties](#).

The Ranger Policy Manager and Ranger UserSync are installed in only one host (specified in the HDP Setup Ranger Host parameter); the Ranger plug-ins for corresponding components are installed wherever those components are installed.

Make sure that the MySQL database used by Ranger is set up to connect from any host in the cluster. For multiple hosts, set up the Ranger MySQL database to connect from any host in the cluster using the root password. To do this, enter:

```
grant all privileges on *.* to 'root'@'%' identified by
'RootPasswordHere' ;flush privileges;
```



Note

In HDP v2.2, MySQL is the only database supported for use with Ranger.

7.15. Using Apache Slider

Apache Slider lets you deploy distributed applications across a Hadoop cluster. On the Windows platform, Slider application packages are included in the Windows Installer msi for HBase and Storm. (Accumulo is available on Linux, but is not currently supported on Windows). Until more information is available, please see the Linux document [Running Applications on YARN Using Slider](#).

7.16. Starting HDP Services

The following steps describe how to start local and remote services. For more information, see [Managing HDP on Windows](#).

1. Start local services on the Master Nodes:

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

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

2. At any Master Node, start all slave node services:

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

3. At the Knox Gateway:

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

4. Smoke test your installation as described in [Validating the Installation](#).

7.17. Setting up HDP

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

1. Open a command prompt using the `hadoop` user in the Hadoop home directory:

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

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

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

```
hdfs dfs -mkdir -p /apps/webhcat
hdfs dfs -chown -R hadoop:users/apps/webhcat
hdfs dfs -chmod -R 755 /apps/webhcat
```

3. When using Tez for Hive, you must also [Upgrade Tez](#).

7.18. Validating Your Data

Verify that your data is intact by comparing the HDFS data directory tree with your previous directory tree.

1. Run the following command as the `hadoop` user, from the Hadoop home directory:

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

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

```
hdfs fs -ls R / > fs-new-lsr-1.log
```

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

- `/apps/hbase` is used by HBase (new when upgrading from 1.3 to 2.2)
- `/mapred/system/jobtracker` should have a new timestamp

4. Run an `fsck` report on your upgraded system. From the Hadoop command line:

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

5. To check the validity of your current HDFS data, compare this `fsck` report to the report generated before the upgrade.

7.19. Verify that HDP Services are Working

To verify that the HDP services work as expected, run the provided smoke tests as the `hadoop` user. (Alternately, create a smoke test user in HDFS and run the test as that user. See [Appendix:Adding a Smoketest User](#).)

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
- Hadoop YARN status

7.20. Finalize Upgrade

When you are satisfied that HDP is successfully functioning with the data from your HDP 1.3, 2.0, or 2.1 cluster, finalize the upgrade.



Warning

Once you have finalized the upgrade, you cannot revert.

1. Open a command prompt with the `hadoop` user in the HDFS Home directory:

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

2. Finalize the upgrade:

```
hdfs dfsadmin -finalizeUpgrade
```

7.21. Troubleshooting

To validate the installation, run the smoke test suite as described in "Validating the Installation".

If the HBase RegionServer and Master do not start, check to see if 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
```

Restart the HBase services.

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.

To help diagnose installation issues, see log file information in [File Locations, Logging, and Common HDFS Commands](#).

8. Managing HDP on Windows

This section describes how to manage HDP on Windows.

8.1. Starting HDP Services

The HDP Windows installer sets up Windows services for each HDP component across the nodes in a cluster. Use the following instructions 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.



Important

To Enable NameNode High Availability, do so while starting HDP services. Do not wait until all services have started.

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

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

You should see a list of installed services and their status.

8.2. Enabling NameNode High Availability

If you are enabling NameNode High Availability in a multi-node cluster, you can run the following commands on the primary and standby hosts while services are starting. Log in to every host and run these commands as administrator.

1. On the primary host, run

```
hdfs namenode -format -force
```

2. On each standby host, run

```
hdfs namenode -bootstrapStandby -force hdfs zkfc -formatZK -force
```

8.3. Validating HA Configuration

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

- a. 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`.

- b. Query the state of a NameNode using `JMX(tag.HAState)`
- c. 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 the 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 an 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 the HA settings are configured correctly.
 - b. To diagnose issues, check log files for `zkfc` daemons and NameNode daemons.

8.4. Stopping HDP Services

The HDP-on-Windows installer sets up Windows services for each HDP component across the nodes in a cluster. To stop HDP services, run the following command from any host machine in your cluster, while logged on as the administrative user:

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

9. Troubleshooting Your Deployment

Use the following information to troubleshoot issues encountered while deploying HDP on the Windows platform:

9.1. Cluster Information

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

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

To list the operating system, run the following command in PowerShell as an Administrator user:

```
(Get-WmiObject -class Win32_OperatingSystem).Caption Microsoft Windows Server 2012 Standard
```

To list the OS Version for your host machine, enter:

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

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

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 specify the complete path to the Java executable, if the 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 whether a specific Java process is reaching the threshold.

Execute the following command on the affected host machine:

```
map -heap pid_of_Hadoop_process
```

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:
```

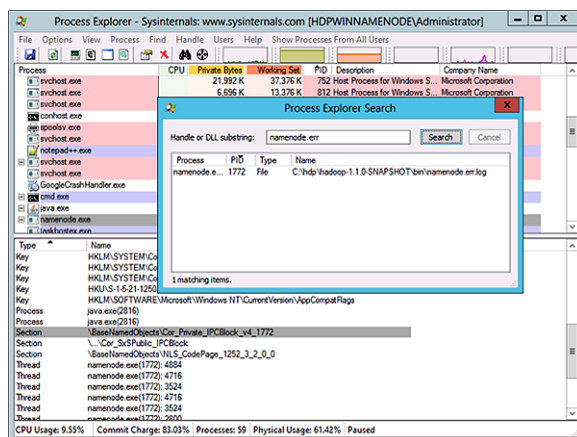
```

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. For information on how to use Process Explorer, see [Windows Sysinternals - 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 following screenshot:



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 third-party tools like Oxygen, XML Spy, etc., to validate the configuration files. Here are instructions for Notepad++:

- Open the XML file to be validated in Notepad++ and select `XML Tools -> Check XML Syntax`.
- 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**. Ensure that no startup items are enabled on the affected host machine.

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

```
Get-Volume
```

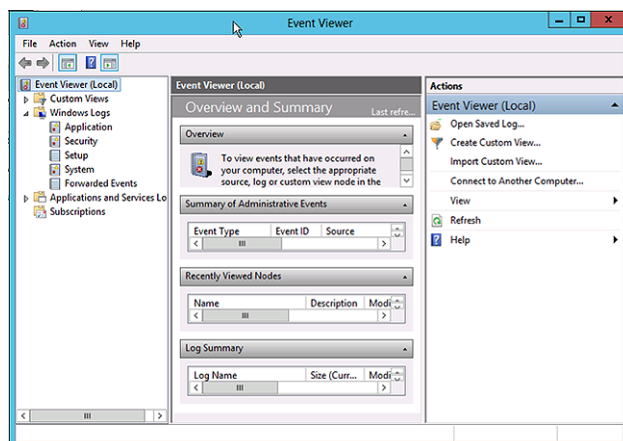
You should see output similar to the following:

Drive Letter	FileSystem Label	FileSystem	DriveType	HealthStatus	Size	Remaining Size
System C	Reserved	NTFS	Fixed	Healthy	108.7 MB	350 MB
D	HRM_SSS...	UDF	CD-ROM	Healthy	10.74 GB	19.97 GB
					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, checks that the network interfaces are available, and validates whether an IP address is bound to the interfaces. To troubleshoot communication issues among 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 generates a list of ports used within the system. To troubleshoot launch issues and resolve potential port conflicts with HDP master processes, run the following command on the host machine:

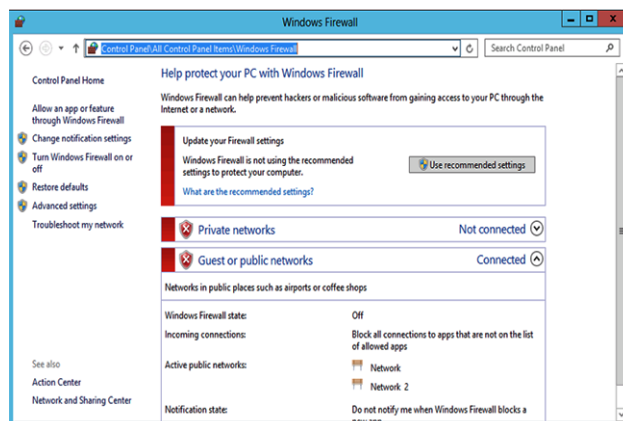
```
netstat -ano
```

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 a 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, Logging, and Common HDFS Commands

This section provides a list of files and their locations, instructions for enabling logging, and a list of HDFS commands that help isolate and troubleshoot issues.

9.2.1. File Locations

- **Configuration files:** These files are used to configure a hadoop cluster.
 - `core-site.xml`: All Hadoop services and clients use this file to locate the NameNode, so this file must be copied to each node that is either running a Hadoop service or is a client node. The Secondary NameNode uses this file to determine the location for storing `fsimage` and edits log `namefs.checkpoint.dir/name` locally, and the location of the NameNode `namefs.namedefault.name/name`.

Use the `core-site.xml` file to isolate communication issues with the NameNode host machine.

- `hdfs-site.xml`: HDFS services use this file, which contains a number of important properties. These include:
 - HTTP addresses for the two services
 - Replication for DataNodes `namedfs.replication/name`
 - DataNode block storage location `namedfs.data.dir/name`
 - NameNode metadata storage `namedfs.name.dir/name`

Use the `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 the location properties in `hdfs-site.xml` are valid locations.

- `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`, which changes the maximum heap size for the DataNode service. Restart the affected DataNode host machine.

- `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`, which changes the maximum heap size for NameNode service. Restart the affected NameNode host machine.

- `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`, which changes the maximum heap size for Secondary NameNode service. Restart the affected Secondary NameNode host machine.

- `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.

- `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. For more details, see the [log4j documentation](#).

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

- **HDFS .out files:** Log files with the `.out` extension are located in `c:\hdp\hadoop-1.1.0-SNAPSHOT\bin`. They have the following naming conventions:

- `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 the startup command string to start the service, and list process ID output 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`

The 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 `host3`, the file would be saved as `hadoop-namenode-host3.log`.



Note

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

- HDFS `<.date>` files:

Log files with the `<.date>` extension 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, the current date is appended to the filename; for example: `hadoop-datanode-hdp121.localdomain.com.log.2013-02-08`.

Use these files to compare the past state of your cluster with the current state, to identify potential patterns.

9.2.2. Enabling Logging

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

To enable audit logging, change the `hdfs.audit.logger` value to `INFO,RFAAUDIT`. 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 --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 HDFS commands is available [here](#).

1. **Get the Hadoop version:** Run the following command on your cluster host machine:

```
hadoop version
```

2. **Check block information:** This command provides a directory listing and displays which node contains the block. Run this command on your HDFS cluster host machine to determine if a block is under-replicated.

```
hdfs 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] ...

```

3. **HDFS report:** Use this command to receive HDFS status. Execute the following command as the hadoop user:

```
hdfs 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

```

4. **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, safemode is then disabled.

The options for safemode command are: `hdfs dfsadmin -safemode [enter | leave | get]`

To enter safemode, execute the following command on your NameNode host machine:
`hdfs dfsadmin -safemode enter`

10. Uninstalling HDP

Choose one of the following options to uninstall HDP.

Use the 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.

Use the Command Line Utility:

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

```
msiexec /x MSI_PATH /lv PATH_to_Installer_Log_File DESTROY_DATA=no
```

where

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

Note that this `msiexec` command retains data from your HDP installation. To delete existing HDP data, set `DESTROY_DATA=yes`.

11. Appendix: Adding a Smoketest User

Creating a smoketest user enables you to run HDP smoke tests without having to run them as the `hadoop` user.

To create a smoketest 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\hdfs fs -chmod -R 757 /mapred
```

3. Create an HDFS directory for the smoketest user:

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

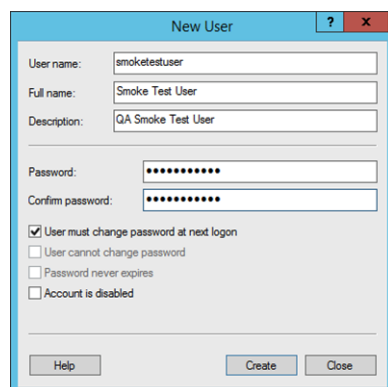
4. Change ownership to the smoketest user.

```
%HADOOP_HOME%\bin\hdfs 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 `smoketest` user. Determine password requirements and select `Create`.

6. Validate the `smoketest` user by running the smoke tests as the `smoketest` user.

- a. Switch to a command prompt as the `smoketest` user. For example:

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

- b. As the `smoketest` user, run the smoke tests:


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