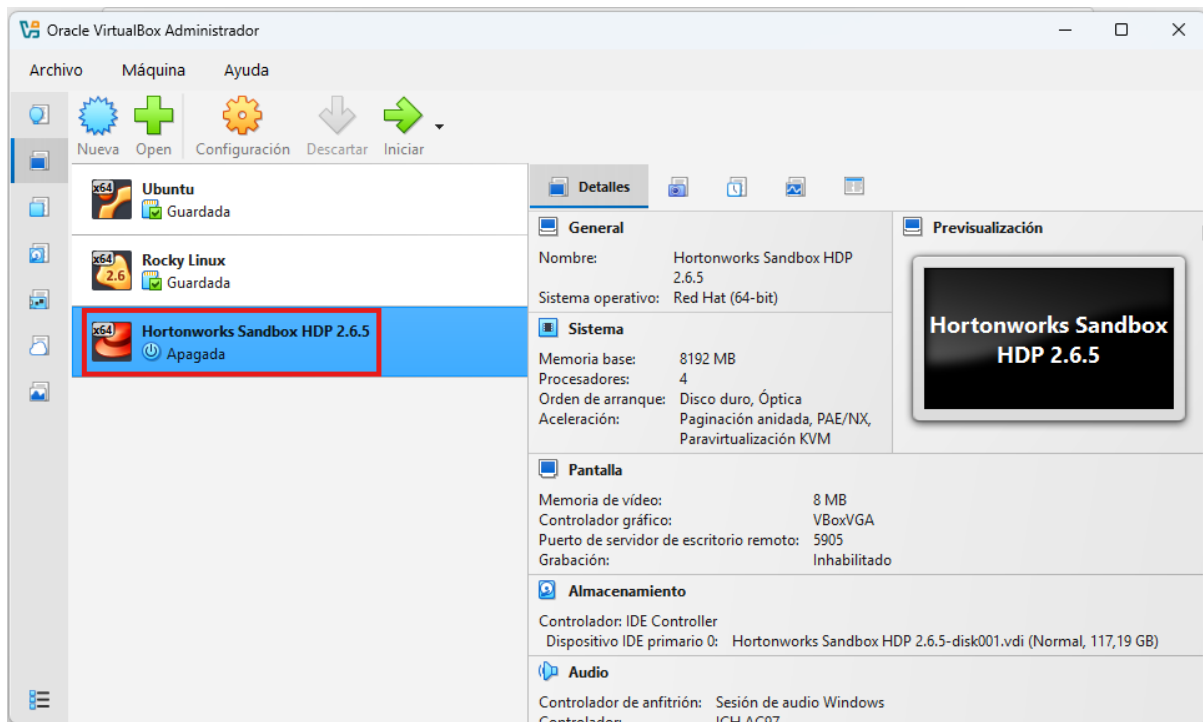
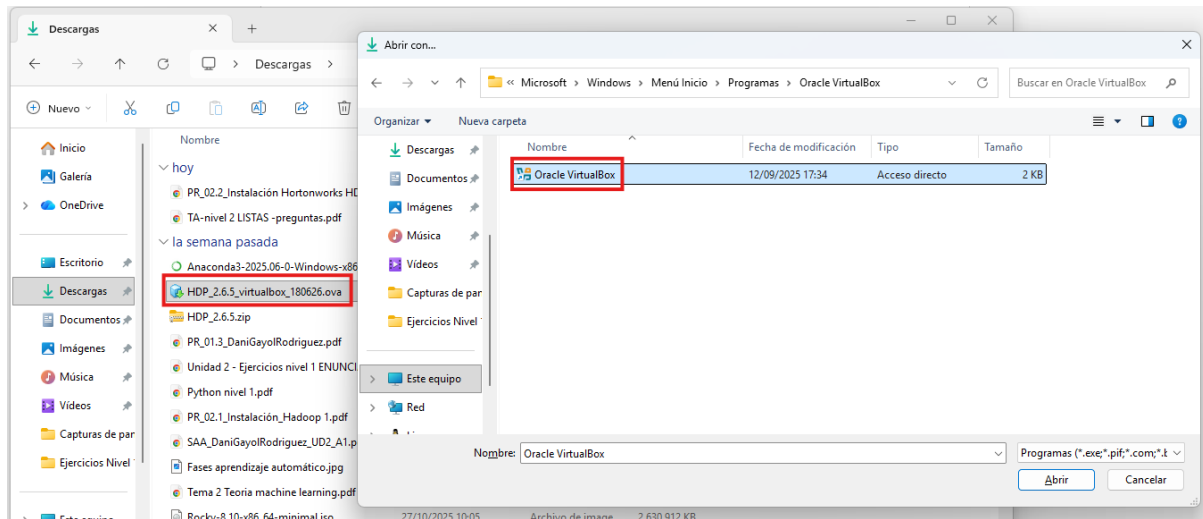
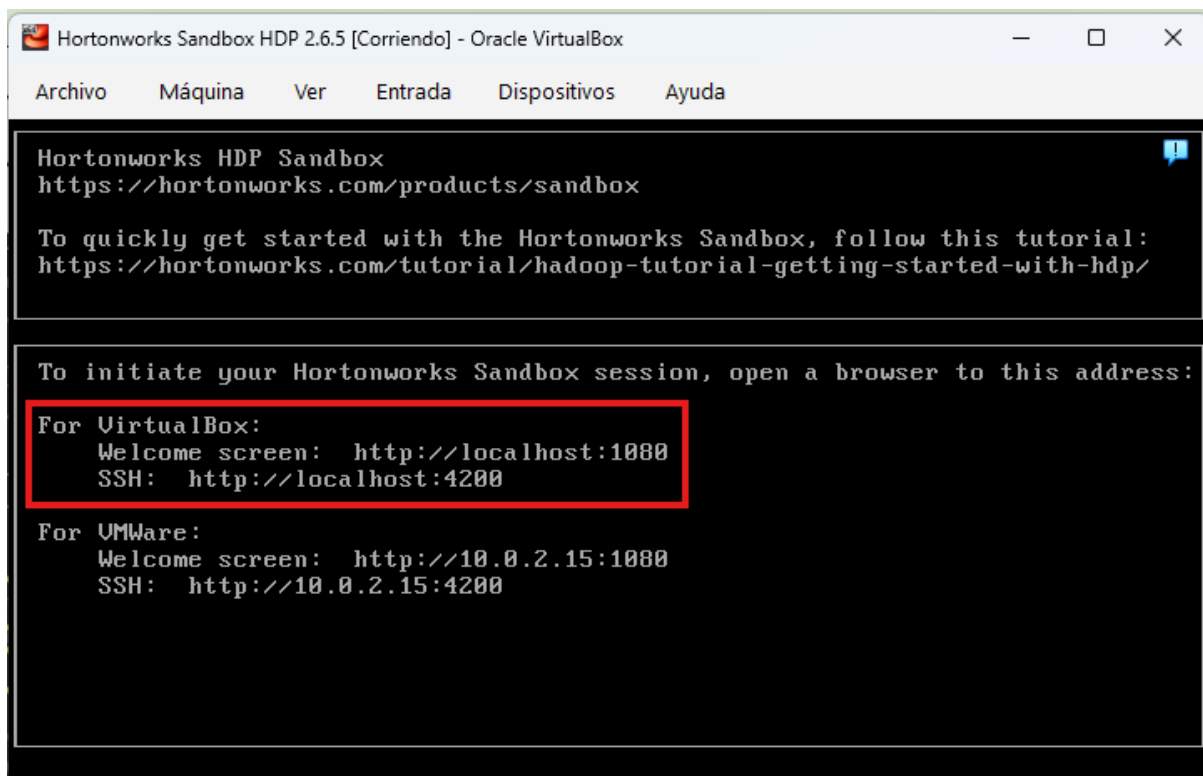


PR_02.2 Dani Gayol Rodríguez

PR_02.2 Dani Gayol Rodríguez.....	1
1.) Instala la máquina descargada en VirtualBox.....	1
3.) Terminal Access	3
4.) Añade los hosts que se especifican para poder acceder a los servicios sin tener que teclear direcciones IP	5
5.) Acceder al panel de administración de Ambari	7
6.) Habilitar la contraseña de Admin	9
7.) Explorar el panel de control de Ambari	11

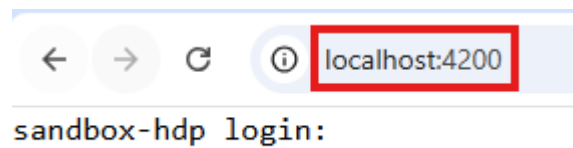
1.) Instala la máquina descargada en VirtualBox



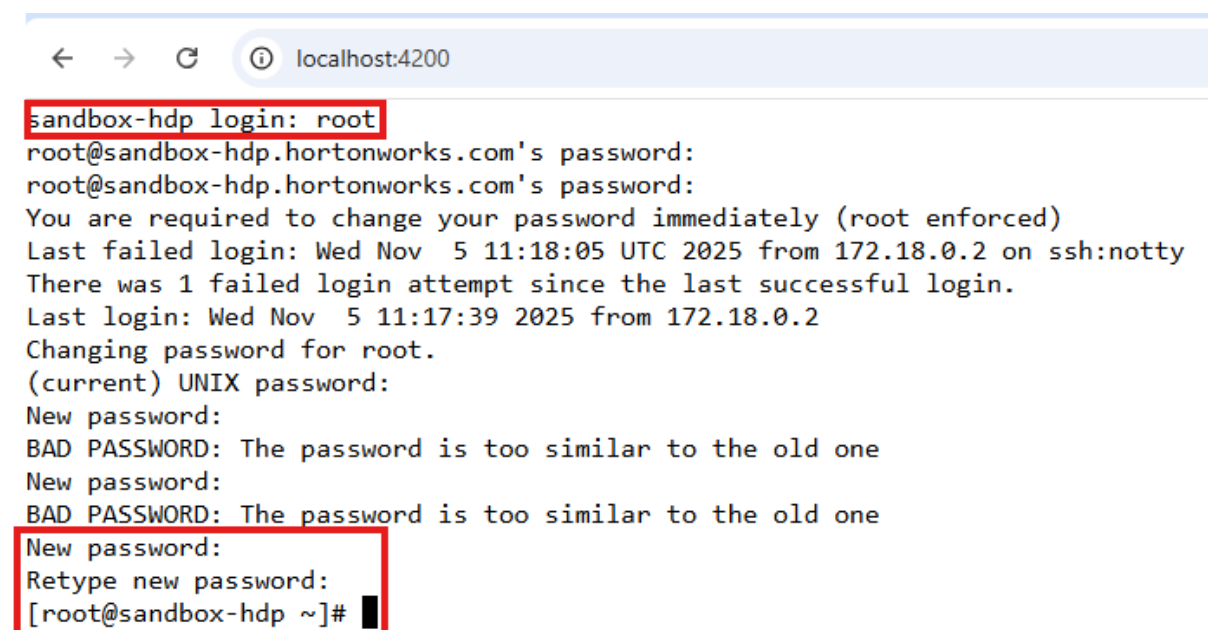


3.) Terminal Access

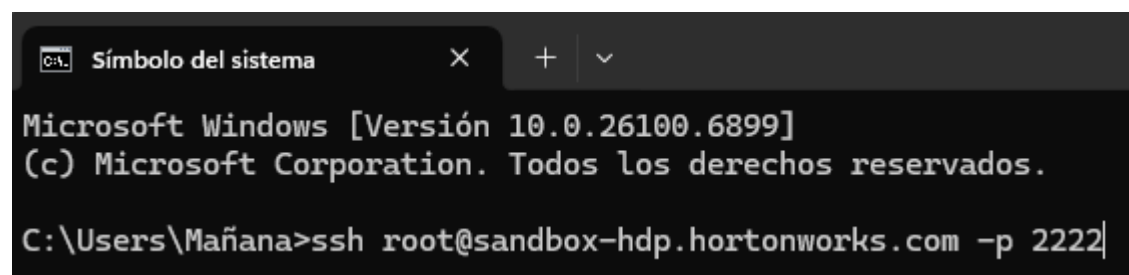
Ahora tenemos que abrir el navegador y escribir <http://localhost:4200>



Te pedirá que inicies sesión, puedes entrar como “root”, para ello el nombre es “root” y la contraseña “hadoop”, luego te va a pedir que la cambies la contraseña. Si no quieres iniciar sesión como root, también puedes iniciar sesión como un usuario normal pero entonces tendrás que usar el comando "sudo"



Ahora para entrar mediante ssh desde la terminal tenemos que abrir la consola de comandos y escribir lo siguiente



Si hacemos esto nos va a salir un error ya que no hicimos el apartado de ponerle un hostname a la IP del sandbox ya que para ello necesitamos los permisos de administrador de windows, pero podemos conectarnos de otra forma usando el “localhost” o la IP de la máquina

```
root@sandbox-hdp:~  
Microsoft Windows [Versión 10.0.26100.6899]  
(c) Microsoft Corporation. Todos los derechos reservados.  
  
C:\Users\Mañana>ssh root@localhost -p 2222  
root@localhost's password:  
Last login: Thu Nov  6 07:58:45 2025 from 172.18.0.3  
[root@sandbox-hdp ~]# |
```

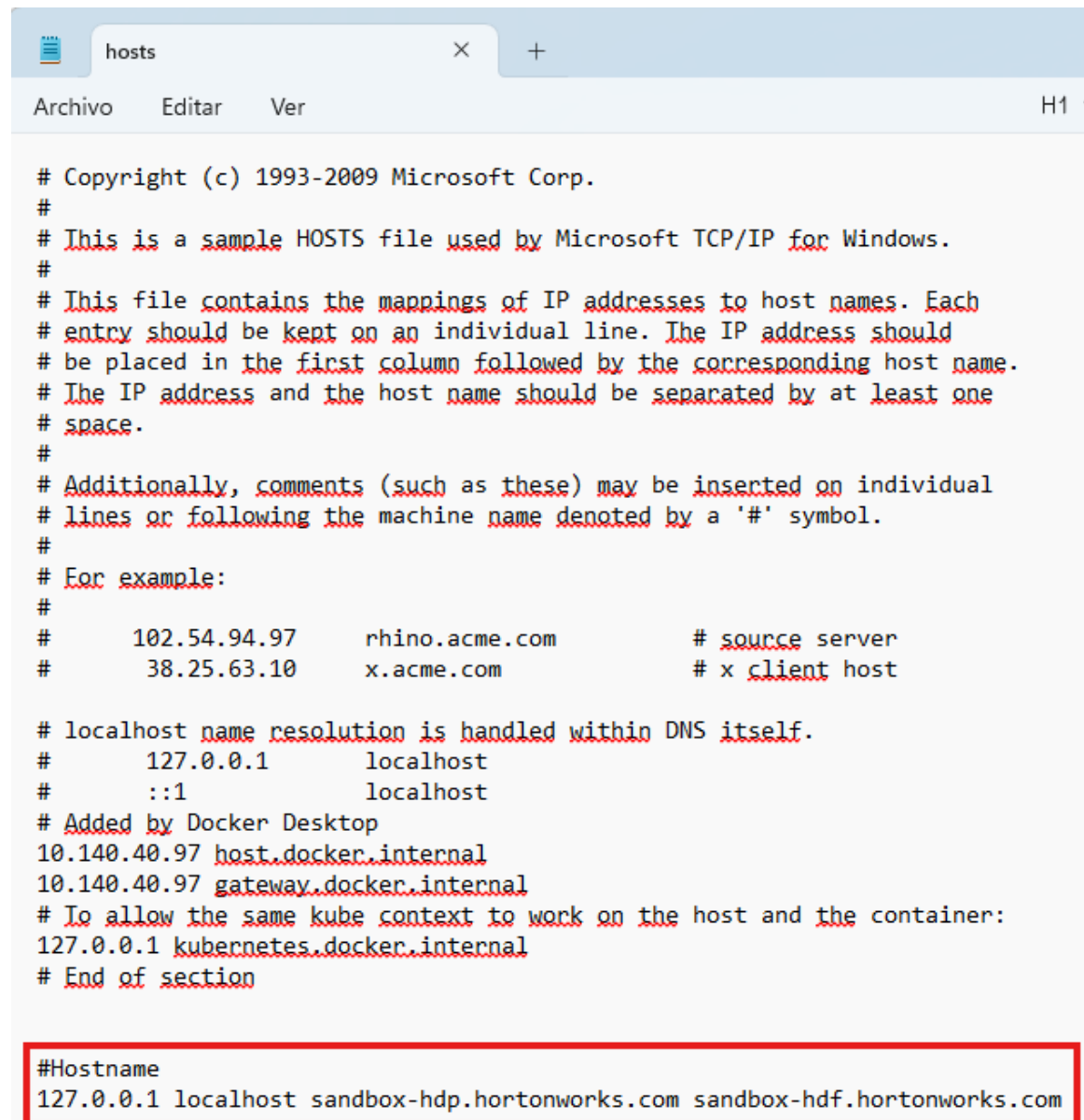
Ahora podemos hacer un ping para probar que todo está funcionando correctamente

```
[root@sandbox-hdp ~]# ping 10.140.42.218  
PING 10.140.42.218 (10.140.42.218) 56(84) bytes of data.  
64 bytes from 10.140.42.218: icmp_seq=1 ttl=254 time=1.33 ms  
64 bytes from 10.140.42.218: icmp_seq=2 ttl=254 time=1.40 ms  
64 bytes from 10.140.42.218: icmp_seq=3 ttl=254 time=0.888 ms  
64 bytes from 10.140.42.218: icmp_seq=4 ttl=254 time=0.531 ms  
^C  
--- 10.140.42.218 ping statistics ---  
4 packets transmitted, 4 received, 0% packet loss, time 3006ms  
rtt min/avg/max/mdev = 0.531/1.038/1.405/0.355 ms
```

4.) Añade los hosts que se especifican para poder acceder a los servicios sin tener que teclear direcciones IP

Para modificarlo necesitamos abrir el bloc de notas como administrador y luego buscar el archivo “hosts” en la siguiente ruta para añadir una línea de comando

“C:\Windows\System32\drivers\etc\hosts”



```
# Copyright (c) 1993-2009 Microsoft Corp.
#
# This is a sample HOSTS file used by Microsoft TCP/IP for Windows.
#
# This file contains the mappings of IP addresses to host names. Each
# entry should be kept on an individual line. The IP address should
# be placed in the first column followed by the corresponding host name.
# The IP address and the host name should be separated by at least one
# space.
#
# Additionally, comments (such as these) may be inserted on individual
# lines or following the machine name denoted by a '#' symbol.
#
# For example:
#
#       102.54.94.97       rhino.acme.com          # source server
#       38.25.63.10       x.acme.com              # x client host


# localhost name resolution is handled within DNS itself.
#       127.0.0.1         localhost
#       ::1               localhost
# Added by Docker Desktop
10.140.40.97 host.docker.internal
10.140.40.97 gateway.docker.internal
# To allow the same kube context to work on the host and the container:
127.0.0.1 kubernetes.docker.internal
# End of section

#Hostname
127.0.0.1 localhost sandbox-hdp.hortonworks.com sandbox-hdf.hortonworks.com
```

Ahora después de hacer esto, si nos deja entrar desde la consola de comandos utilizando lo siguiente

root@sandbox-hdp:~

Mañana@A26P52 MTNGW64 ~ (master)

\$ ssh root@sandbox-hdp.hortonworks.com -p 2222

root@sandbox-hdp.hortonworks.com's password:

Last login: Thu Nov 6 09:24:57 2025 from 172.18.0.3

[root@sandbox-hdp ~]# pwd

/root

[root@sandbox-hdp ~]# whoami

root

[root@sandbox-hdp ~]# ping 10.140.42.218

PING 10.140.42.218 (10.140.42.218) 56(84) bytes of data.

64 bytes from 10.140.42.218: icmp_seq=1 ttl=254 time=0.533 ms

64 bytes from 10.140.42.218: icmp_seq=2 ttl=254 time=1.33 ms

64 bytes from 10.140.42.218: icmp_seq=3 ttl=254 time=0.635 ms

64 bytes from 10.140.42.218: icmp_seq=4 ttl=254 time=0.976 ms

^C

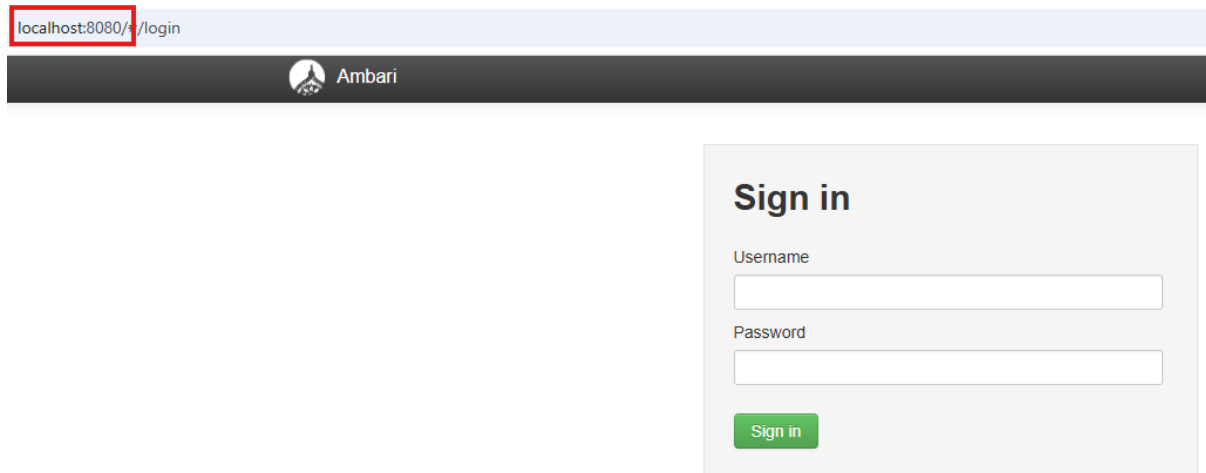
--- 10.140.42.218 ping statistics ---

4 packets transmitted, 4 received, 0% packet loss, time 3014ms

rtt min/avg/max/mdev = 0.533/0.870/1.338/0.317 ms

5.) Acceder al panel de administración de Ambari

El dashboard de Ambari está en el puerto “8080”



The screenshot shows the Ambari login interface. At the top, a light blue header bar contains the text "localhost:8080/" followed by a red-bordered box containing the text "/login". Below this is a dark grey navigation bar with the Ambari logo and the word "Ambari". The main content area is white and features a "Sign in" section on the right. This section has a title "Sign in", two input fields labeled "Username" and "Password", and a green "Sign in" button.

Nosotros vamos a usar dos credenciales, la de “maria_dev” y la de “admin”



This screenshot shows the "Sign in" form with the username "maria_dev" entered in the "Username" field and a masked password "*****" in the "Password" field. Both the input fields and the "Sign in" button are highlighted with red borders.

Ambari

Sandbox

0 ops

0 alerts

DashboardServicesHostsAlertsAdmin

maria_dev

About

Sign out

HDFS

YARN

MapReduce2

Tez

Hive

HBase

Pig

Sqoop

Oozie

ZooKeeper

Falcon

Storm

Flume

Ambari Infra

Atlas

Kafka

Knox

Ranger

Spark2

Zeppelin

Notebook

Druid

Slider

Superset

Actions

MetricsHeatmapsConfig History

Metric ActionsLast 1 hour

HDFS Disk Usage

28%

DataNodes Live

1/1

HDFS Links

NameNode

Secondary NameNode

1 DataNodes

More

Memory Usage

No Data Available

Network Usage

No Data Available

CPU Usage

No Data Available

Cluster Load

No Data Available

NameNode Heap

20%

NameNode RPC

2.20 ms

NameNode CPU WIO

n/a

NameNode Uptime

48.1 min

HBase Master Heap

n/a

HBase Links

No Active Master

1 RegionServers

n/a

More

HBase Ave Load

n/a

HBase Master Uptime

n/a

ResourceManager Heap

19%

ResourceManager Uptime

46.3 min

YARN Memory

0%

NodeManagers Live

1/1

YARN Links

ResourceManager

1 NodeManagers

More

6.) Habilitar la contraseña de Admin

Para empezar, tenemos que abrir la shell de comandos e iniciar sesión como root

```
localhost:4200
sandbox-hdp login: root
root@sandbox-hdp.hortonworks.com's password:
Last login: Thu Nov  6 08:35:56 2025 from 172.18.0.2
[root@sandbox-hdp ~]#
```

Luego tenemos que poner el siguiente comando “ambari-admin-password-reset”

```
[root@sandbox-hdp ~]# ambari-admin-password-reset
Please set the password for admin:
Please retype the password for admin:

The admin password has been set.
Restarting ambari-server to make the password change effective...

Using python /usr/bin/python
Restarting ambari-server
Waiting for server stop...
Ambari Server stopped
Ambari Server running with administrator privileges.
Organizing resource files at /var/lib/ambari-server/resources...
Ambari database consistency check started...
Server PID at: /var/run/ambari-server/ambari-server.pid
Server out at: /var/log/ambari-server/ambari-server.out
Server log at: /var/log/ambari-server/ambari-server.log
Waiting for server start.....
Server started listening on 8080

DB configs consistency_check: no errors and warnings were found.
```

Y ahora vamos a comprobar que podemos iniciar sesión como “admin” en Ambari

Sign in

Username

admin

Password

.....

Sign in

Ambari Sandbox 0 ops 0 alerts Dashboard Services Hosts Alerts Admin **admin**

- About
- Manage Ambari
- Settings
- Sign out

Metrics Heatmaps Config History

Metric Actions Last 1 hour

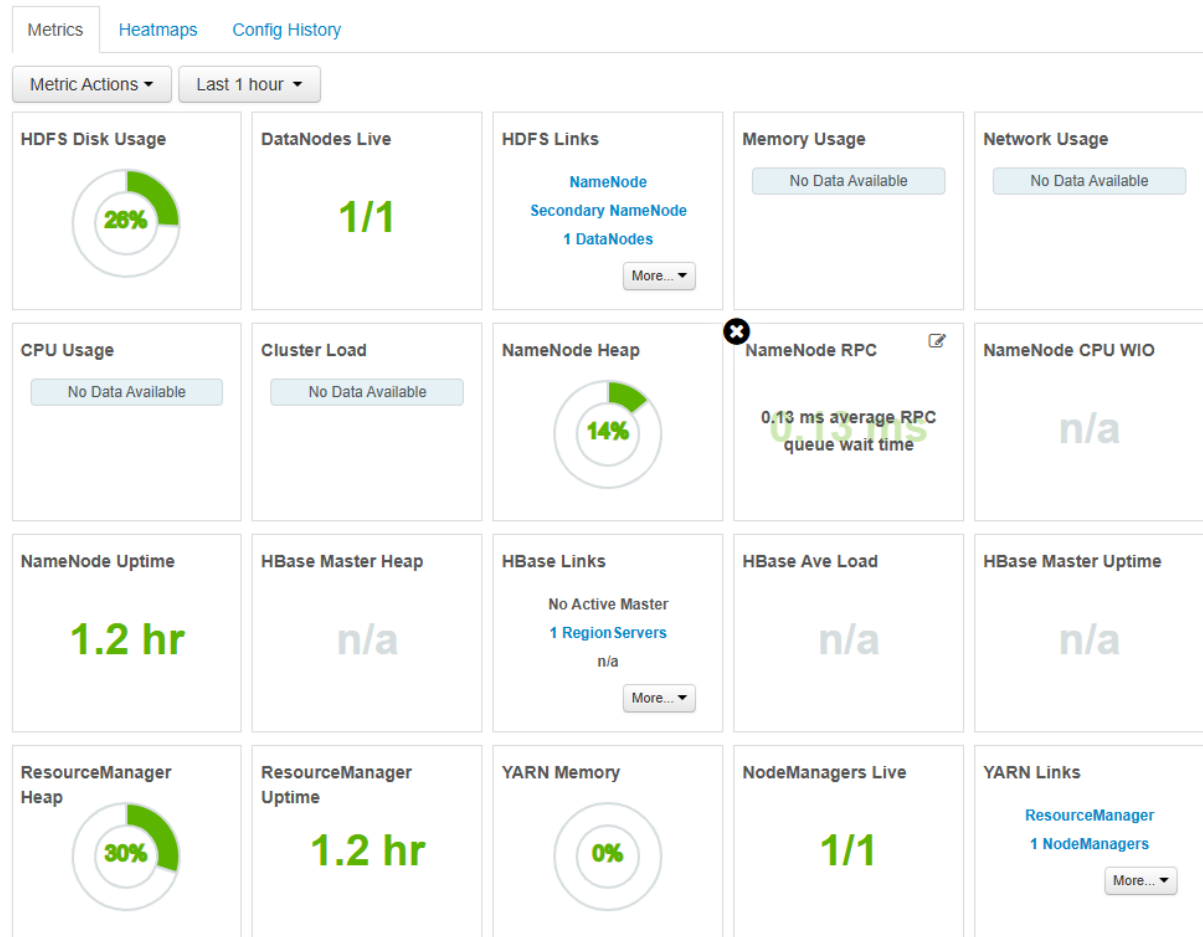
HDFS Disk Usage 26%	DataNodes Live 1/1	HDFS Links NameNode Secondary NameNode 1 DataNodes More...	Memory Usage No Data Available	Network Usage No Data Available
CPU Usage No Data Available	Cluster Load No Data Available	NameNode Heap 38%	NameNode RPC 0.90 ms	NameNode CPU WIO n/a
NameNode Uptime 1.0 hr	HBase Master Heap n/a	HBase Links No Active Master 1 RegionServers n/a More...	HBase Ave Load n/a	HBase Master Uptime n/a
ResourceManager Heap 12%	ResourceManager Uptime 59.0 min	YARN Memory 0%	NodeManagers Live 1/1	YARN Links ResourceManager 1 NodeManagers More...

Actions

7.) Explorar el panel de control de Ambari



Dashboard: muestra el estado general del clúster, uso de la CPU, memoria y disco, estado de los servicios, etc.



Servicios: muestra la lista de todos los servicios instalados, cada servicio se puede expandir para ver su estado, logs, métricas y opciones de gestión

SummaryHeatmapsConfigsQuick Links ▾Service Actions ▾

SummaryNo alerts

[NameNode](#) ✔ Started No alerts

[SNameNode](#) ✔ Started No alerts

[DataNodes](#) 1/1 Started

DataNodes Status1 live / 0 dead / 0 decommissioning

[JournalNodes](#) 0/0 JournalNodes Live

[NFSGateways](#) 0/0 Started

NameNode Uptime1.24 hours

NameNode Heap40.6 MB / 240.0 MB (16.9% used)

Disk Usage (DFS Used)2.1 GB / 106.0 GB (1.95%)

Disk Usage (Non DFS Used)25.6 GB / 106.0 GB (24.13%)

Disk Remaining78.3 GB / 106.0 GB (73.92%)

Blocks (total)1082

Block Errors0 corrupt replica / 0 missing / 0 under replicated

Total Files + Directories1314

Upgrade StatusNo pending upgrade

Safe Mode StatusNot in safe mode

MetricsLast 1 hour ▾

NameNode GC count

No Data Available

NameNode GC time

No Data Available

NN Connection Load

No Data Available

NameNode Heap

No Data Available

NameNode Host Load

No Data Available

NameNode RPC

No Data Available

Failed disk volumes

n/a

Blocks With Corrupted Replicas

0

Under Replicated Blocks

0

HDFS Space Utilization

n/a

Hosts: muestra la lista de máquinas que forman el clúster y también muestra los recursos de cada host

Actions ▾

Filter by host and component attributes or search by keyword ...

<input type="checkbox"/>	Name ↕	IP Address ↕	Rack ↕	Cores ↕	RAM ↕	Disk Usage ↕	Load Avg ↕	Versions	Components
<input type="checkbox"/>	✔ sandbox-hdp.hortonworks....	172.18.0.2	/default-rack	4 (4)	7.79GB	<div></div>		HDP-2.6.5.0	56 Components

Show: 10 ▾1 - 1 of 1 ⏪ ⏩

Alerts: muestra las alertas activas y permite gestionar o desactivar alertas.

Actions

Groups: All (85)

Alert Definition Name	Status	Service	Last Status Changed	State
Any	All	All	Any	All
Falcon Server Web UI	CRIT	Falcon	7 years ago	Enabled
Falcon Server Process	CRIT	Falcon	7 years ago	Enabled
Metadata Server Web UI	CRIT	Atlas	7 years ago	Enabled
HBase Master Process	CRIT	HBase	7 years ago	Enabled
HBase RegionServer Process	CRIT	HBase	7 years ago	Enabled
Knox Gateway Process	CRIT	Knox	7 years ago	Enabled
Infra Solr Web UI	CRIT	Ambari Infra	7 years ago	Enabled
Storm Web UI	CRIT	Storm	7 years ago	Enabled
Supervisor Process	CRIT	Storm	7 years ago	Enabled
Nimbus Process	CRIT	Storm	7 years ago	Enabled

85 of 85 definitions showing - [clear filters](#)

Show: 10

1 - 10 of 85

Admin: muestra la configuración del clúster y opciones para añadir servicios o hosts.

Stack

Versions

Upgrade History

Service	Version	Status	Description
HDFS	2.7.3	Installed	Apache Hadoop Distributed File System
YARN	2.7.3	Installed	Apache Hadoop NextGen MapReduce (YARN)
MapReduce2	2.7.3	Installed	Apache Hadoop NextGen MapReduce (YARN)
Tez	0.7.0	Installed	Tez is the next generation Hadoop Query Processing framework written on top of YARN.
Hive	1.2.1000	Installed	Data warehouse system for ad-hoc queries & analysis of large datasets and table & storage management service
HBase	1.1.2	Installed	A Non-relational distributed database, plus Phoenix, a high performance SQL layer for low latency applications.
Pig	0.16.0	Installed	Scripting platform for analyzing large datasets
Sqoop	1.4.6	Installed	Tool for transferring bulk data between Apache Hadoop and structured data stores such as relational databases
Oozie	4.2.0	Installed	System for workflow coordination and execution of Apache Hadoop jobs. This also includes the installation of the optional Oozie Web Console which relies on and will install the ExtJS Library.
ZooKeeper	3.4.6	Installed	Centralized service which provides highly reliable distributed coordination
Falcon	0.10.0	Installed	Data management and processing platform
Storm	1.1.0	Installed	Apache Hadoop Stream processing framework
Flume	1.5.2	Installed	A distributed service for collecting, aggregating, and moving large amounts of streaming data into HDFS