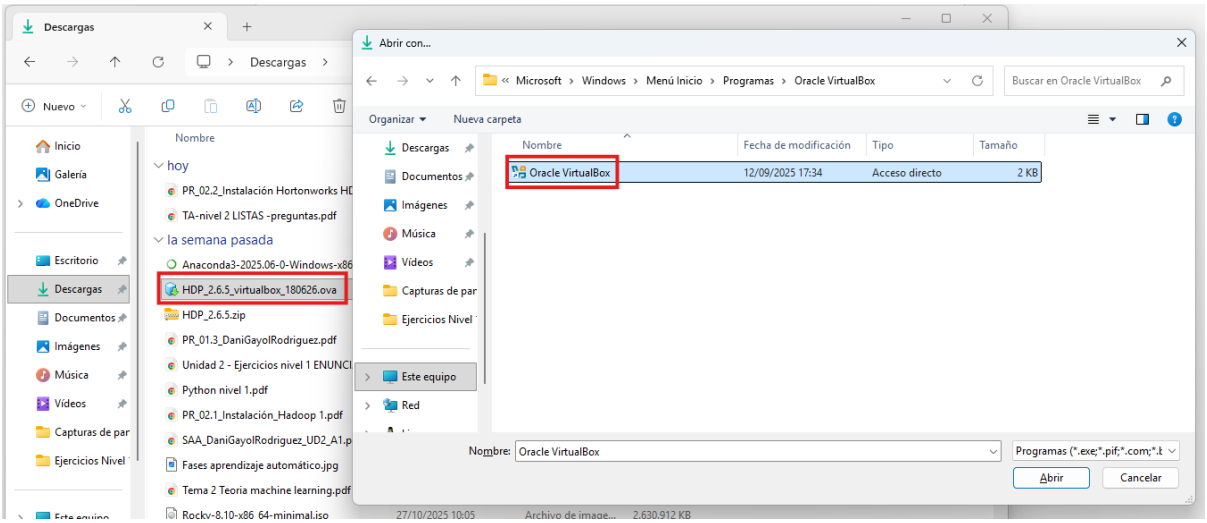
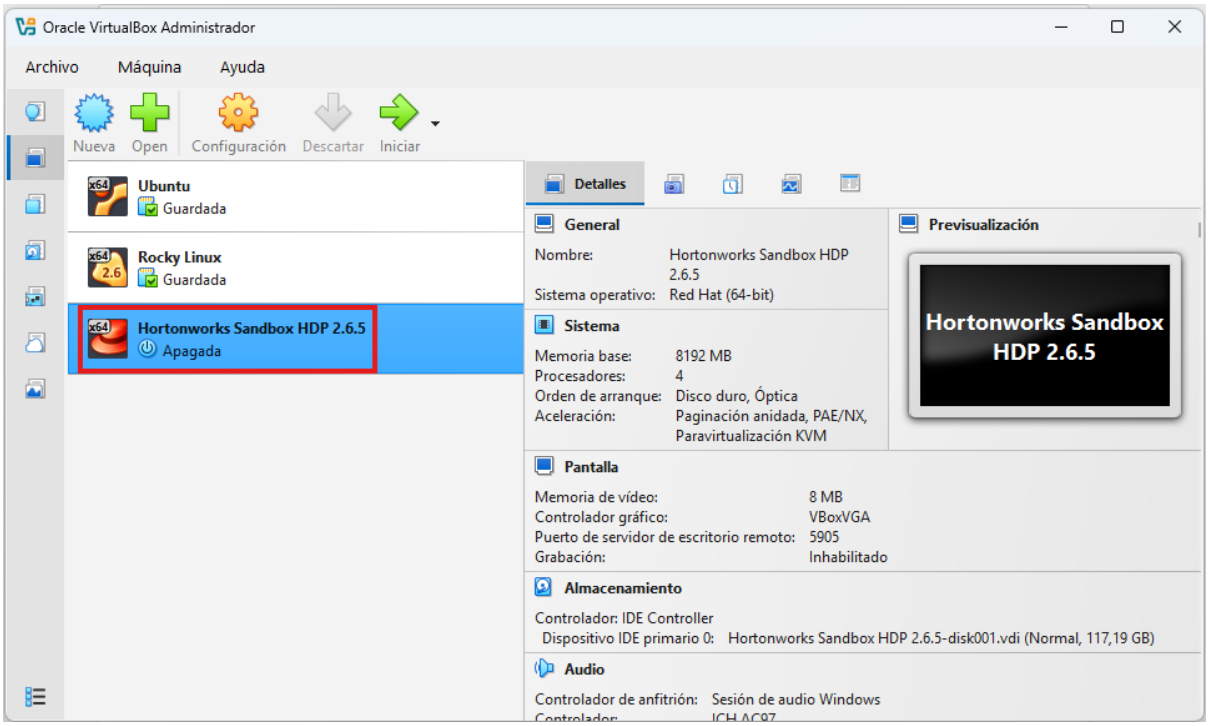
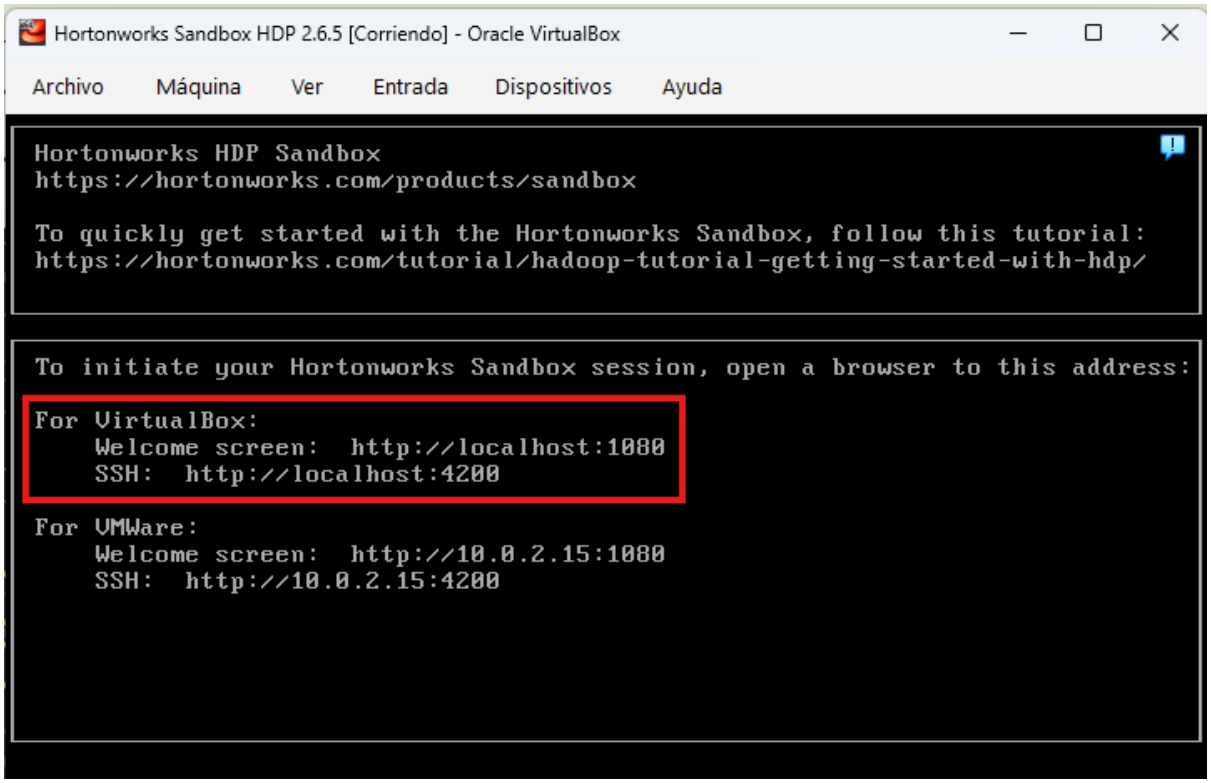
PR_02.2 Dani Gayol Rodríguez

1.) Instala la máquina descargada en VirtualBox

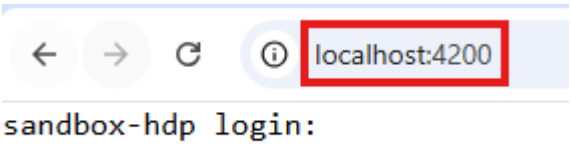




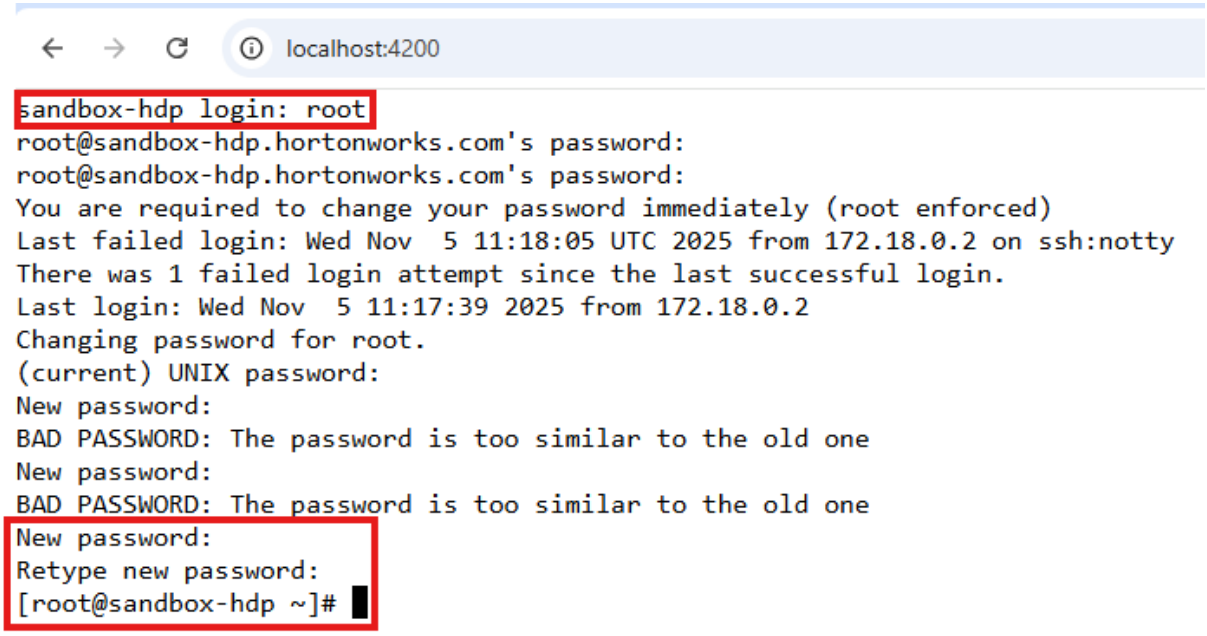

Hortonworks Sandbox HDP 2.6.5 [Corriendo] - Oracle VirtualBox
Archivo
Máquina
Ver
Entrada
Dispositivos
Ayuda
Hortonworks HDP Sandbox
https://hortonworks.com/products/sandbox
To quickly get started with the Hortonworks Sandbox, follow this tutorial:
https://hortonworks.com/tutorial/hadoop-tutorial-getting-started-with-hdp/
To initiate your Hortonworks Sandbox session, open a browser to this address:
For VirtualBox:
Welcome screen: http://localhost:1080
SSH: http://localhost:4200
For VMWare:
Welcome screen: http://10.0.2.15:1080
SSH: http://10.0.2.15:4200

3.) Terminal Access

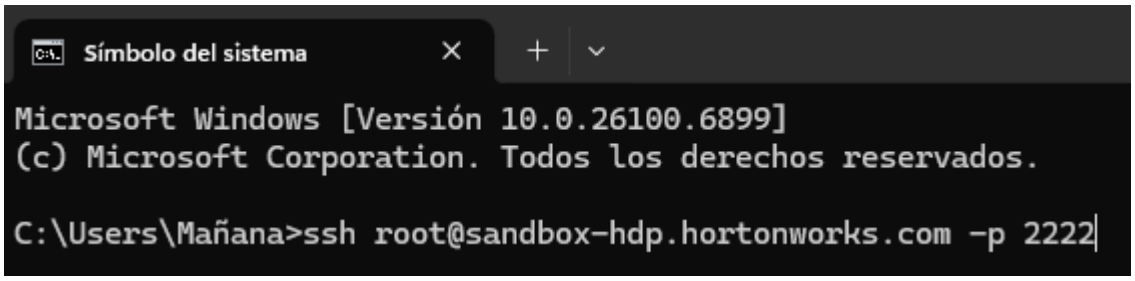
Ahora tenemos que abrir el navegador y escribir http://localhost:4200



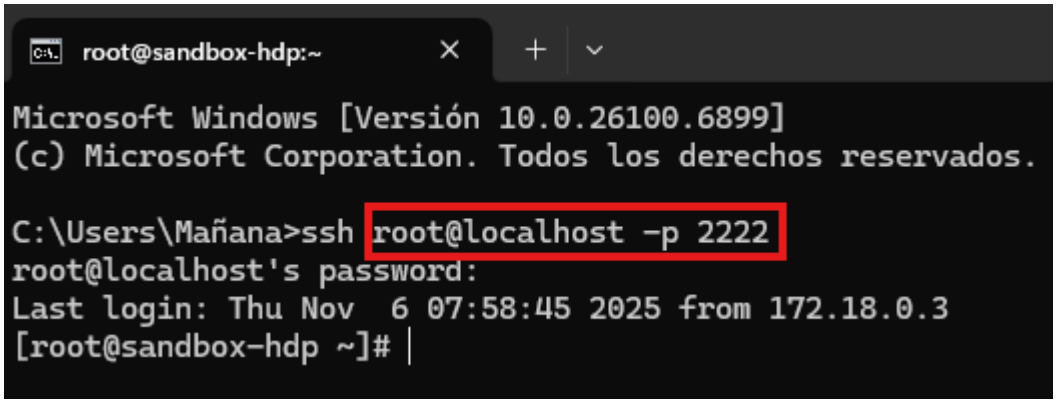
Te pedirá que inicies sesión, puedes entrar como “root”, para ello el nombre es “root” y la contraseña “hadoop”, luego te va a pedir que la cambies la contraseña. Si no quieres iniciar sesión como root, también puedes iniciar sesión como un usuario normal pero entonces tendrás que usar el comando "sudo”



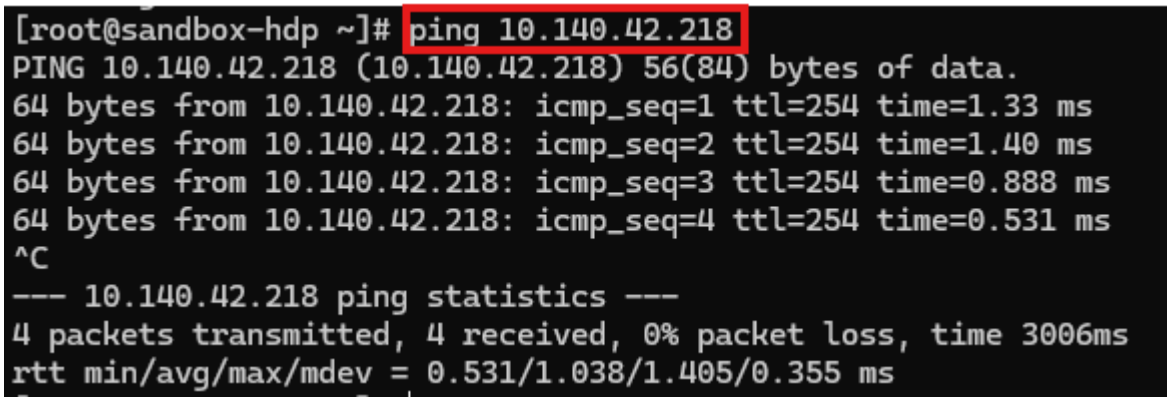
Ahora para entrar mediante ssh desde la terminal tenemos que abrir la consola de comandos y escribir lo siguiente



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

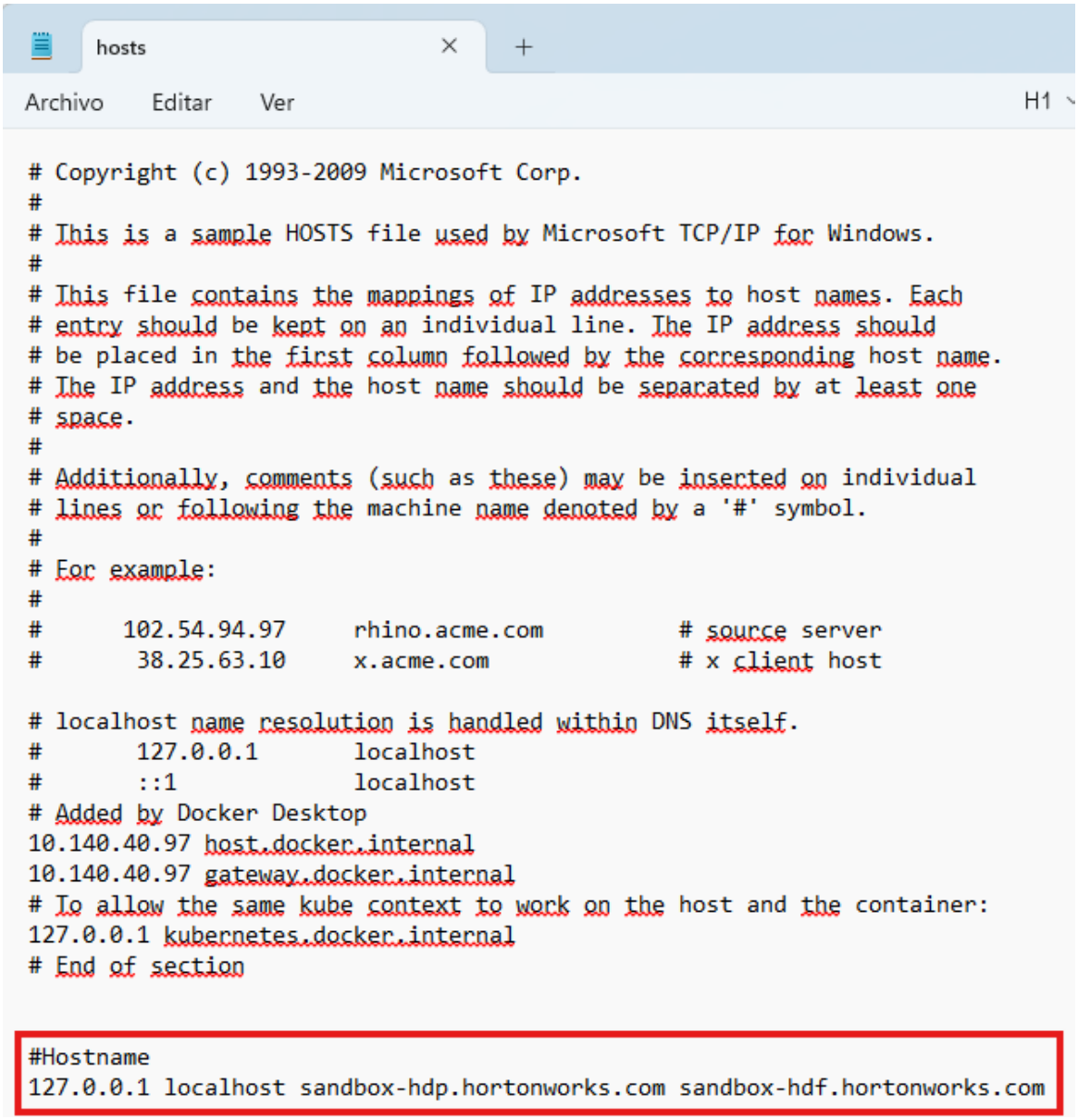


Ahora podemos hacer un ping para probar que todo está funcionando correctamente

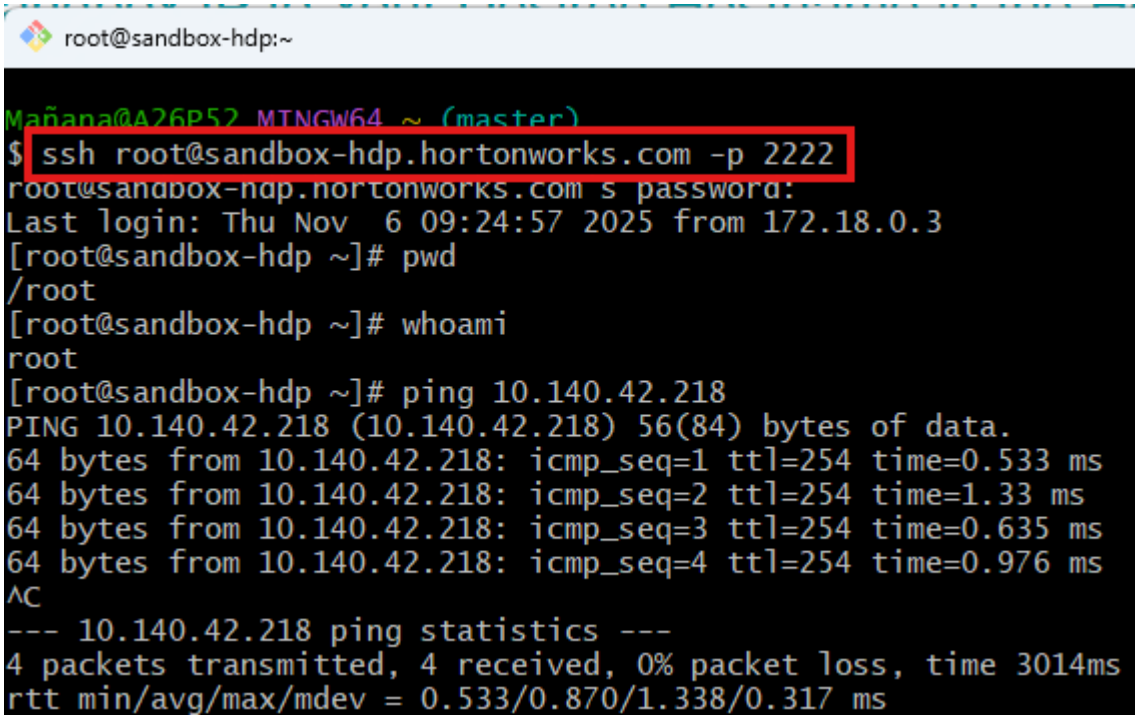


4.) Añade los hosts que se especifican para poder acceder a los servicios sin tener que teclear direcciones IP

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”

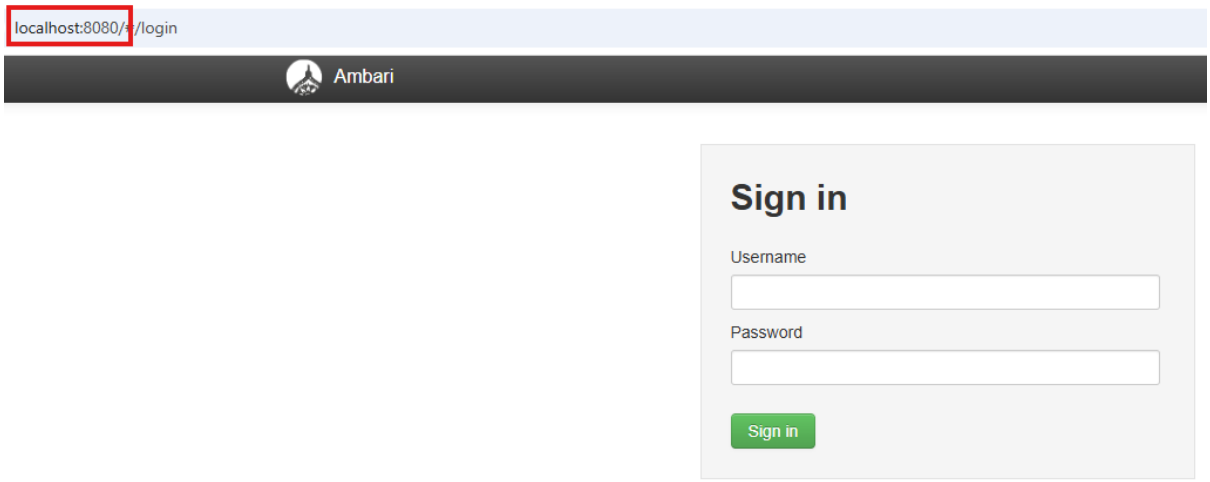


Ahora después de hacer esto, si nos deja entrar desde la consola de comandos utilizando lo siguiente

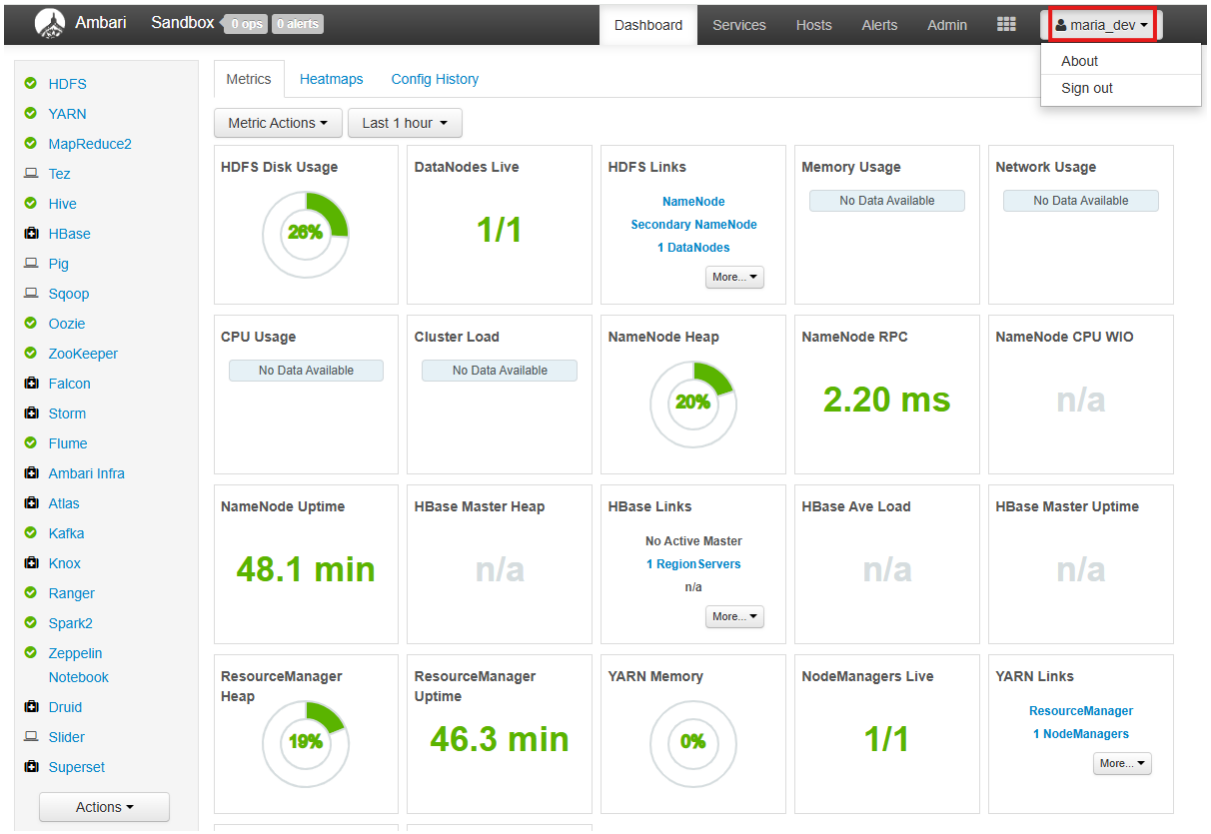

root@sandbox-hdp:~
Mañana@A26P52 MINGW64 ~ (master)
$ ssh root@sandbox-hdp.hortonworks.com -p 2222
root@sandbox-hdp.hortonworks.com's password:
Last login: Thu Nov 6 09:24:57 2025 from 172.18.0.3
[root@sandbox-hdp ~]# pwd
/root
[root@sandbox-hdp ~]# whoami
root
[root@sandbox-hdp ~]# ping 10.140.42.218
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”

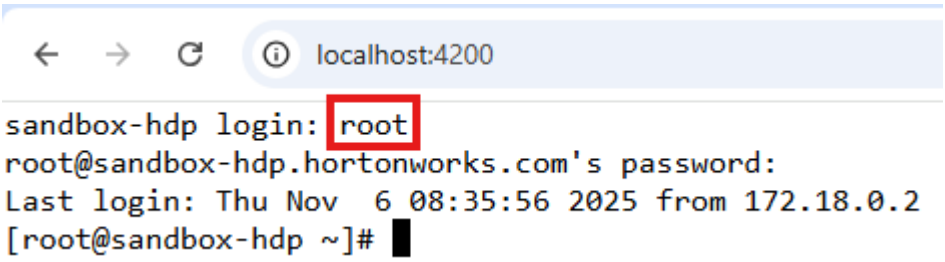


Nosotros vamos a usar dos credenciales, la de “maria_dev” y la de “admin”


Ambari
Sandbox
0 ops
0 alerts
Dashboard
Services
Hosts
Alerts
Admin
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
Metrics
Heatmaps
Config History
Metric Actions
Last 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 Region Servers
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

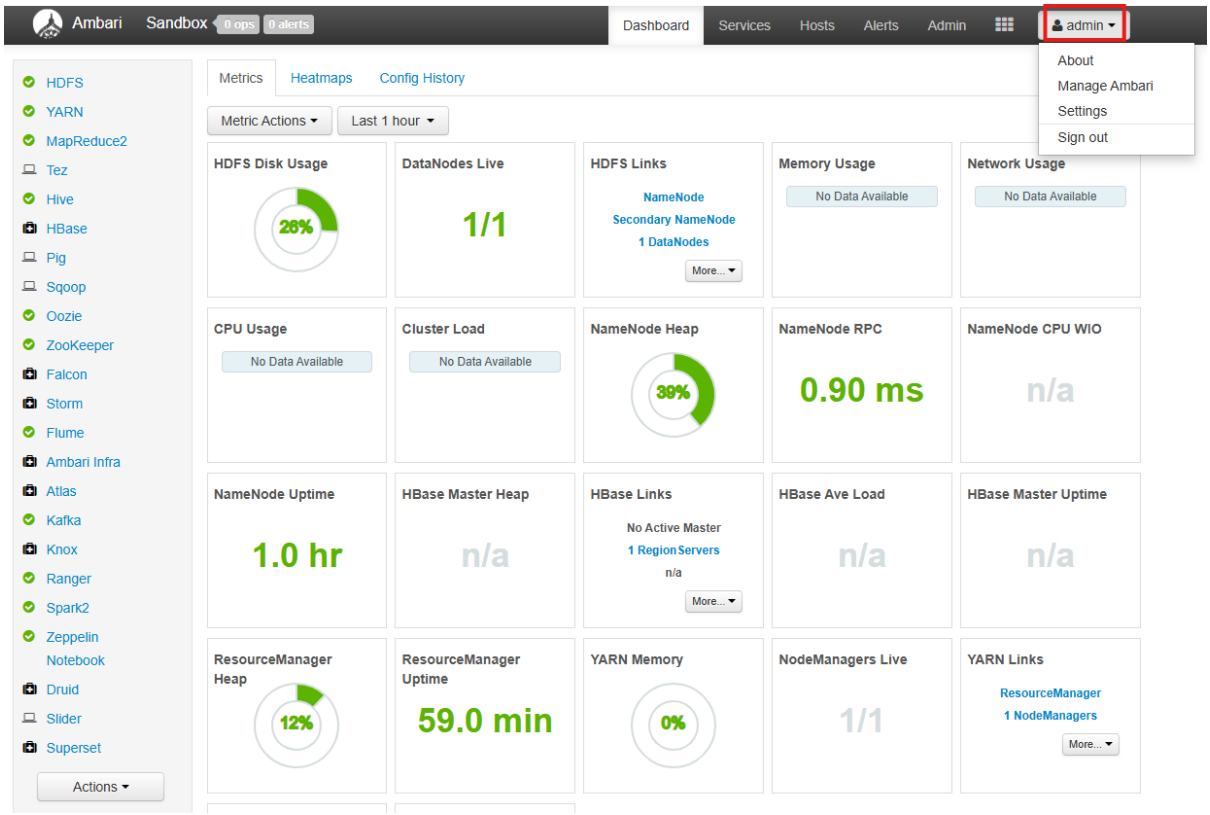


Luego tenemos que poner el siguente comando “ambari-admin-password-reset"

```
[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.
```

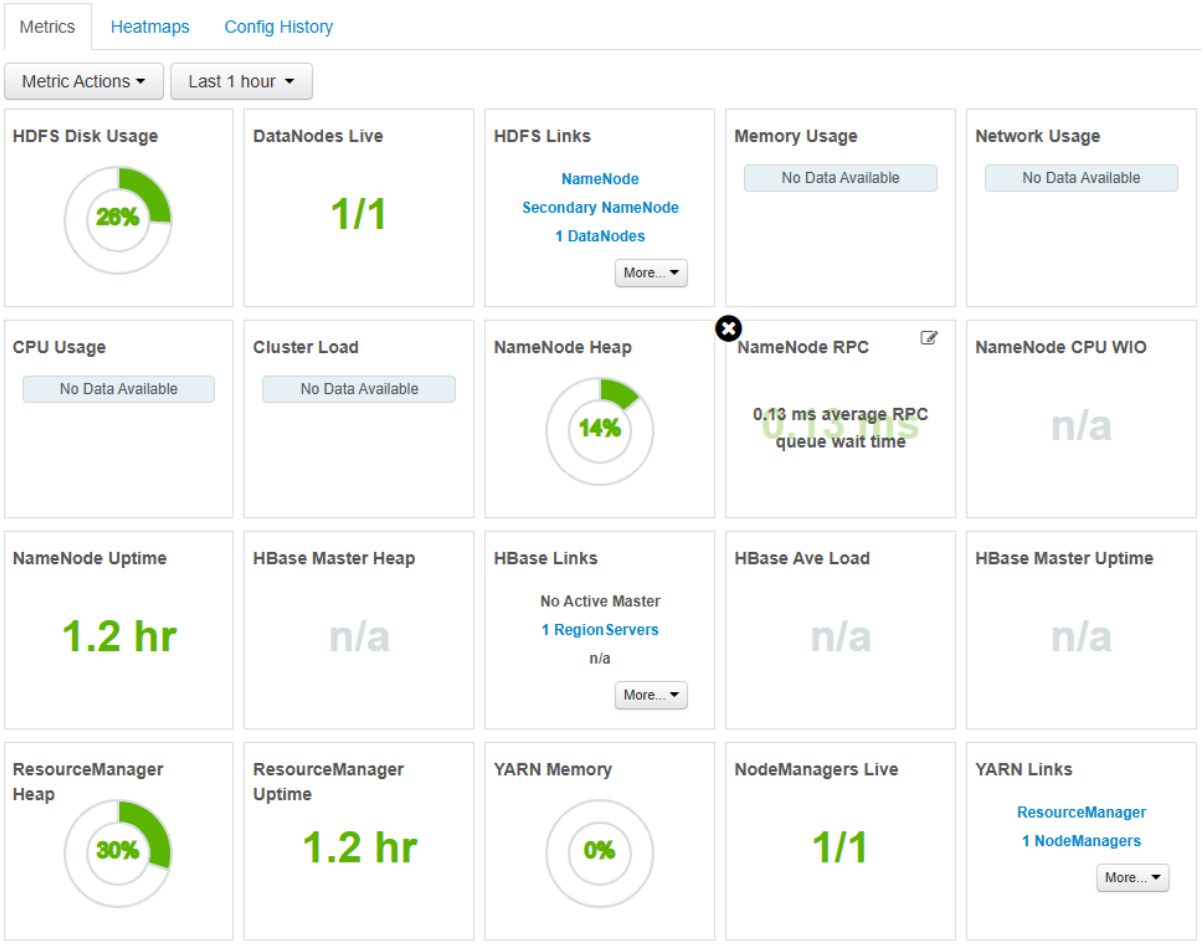
Y ahora vamos a comprobar que podemos iniciar sesión como “admin” en Ambari


Sign in
Username
admin
Password
•••••
Sign in

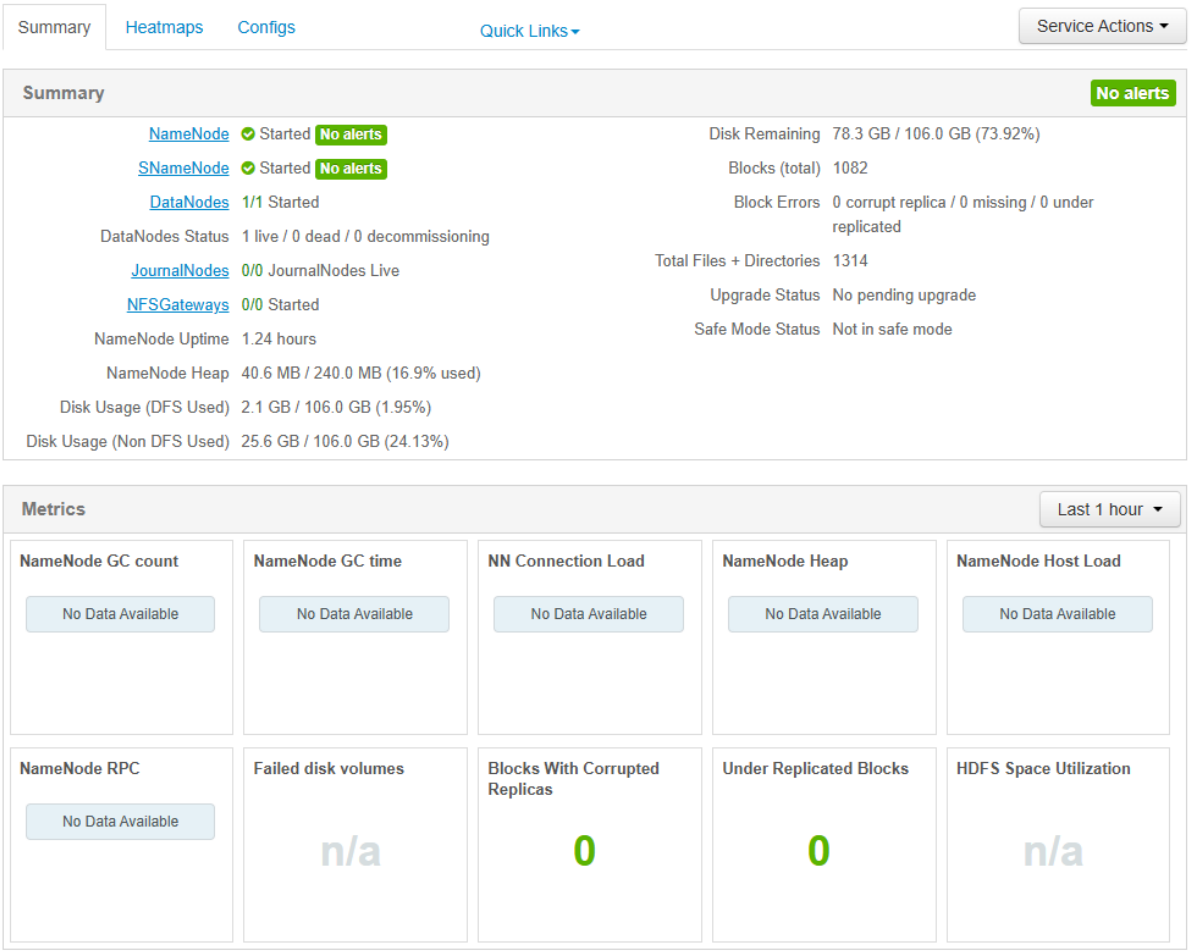

Ambari
Sandbox
0 ops
0 alerts
Dashboard
Services
Hosts
Alerts
Admin
admin
About
Manage Ambari
Settings
Sign out
HDFS
YARN
MapReduce2
Tez
Hive
HBase
Pig
Sqoop
Oozie
ZooKeeper
Falcon
Storm
Flume
Ambari Infra
Atlas
Kafka
Knox
Ranger
Spark2
Zeppelin Notebook
Druid
Slider
Superset
Actions
Metrics
Heatmaps
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...

7.) Explorar el panel de control de Ambari

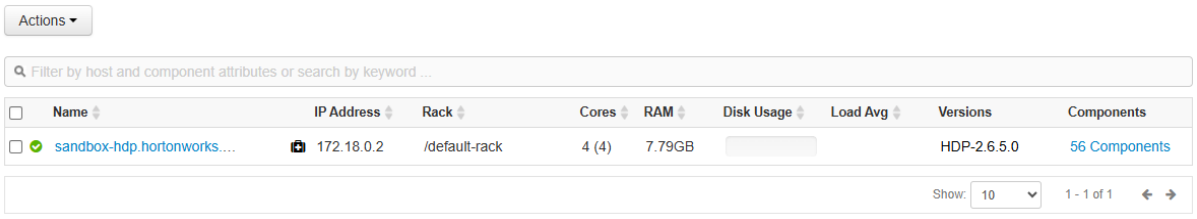
Dashboard: muestra el estado general del clúster, uso de la CPU, memoria y disco, estado de los servicios, etc.



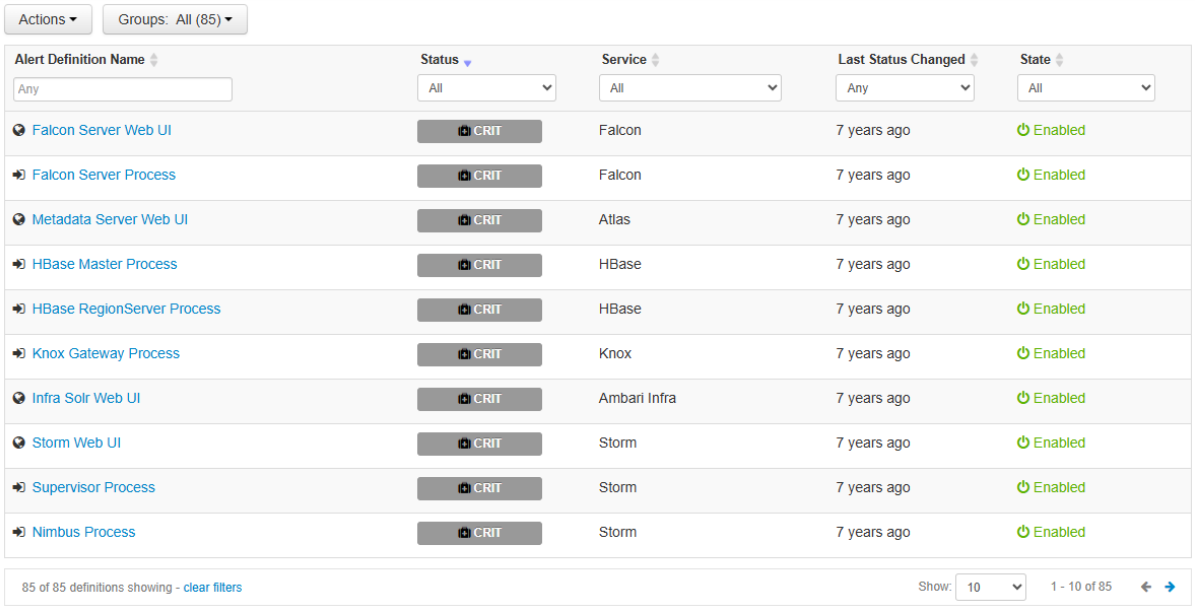
Servicios: muestra la lista de todos los servicios instalados, cada servicio se puede expandir para ver su estado, logs, métricas y opciones de gestión



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



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



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