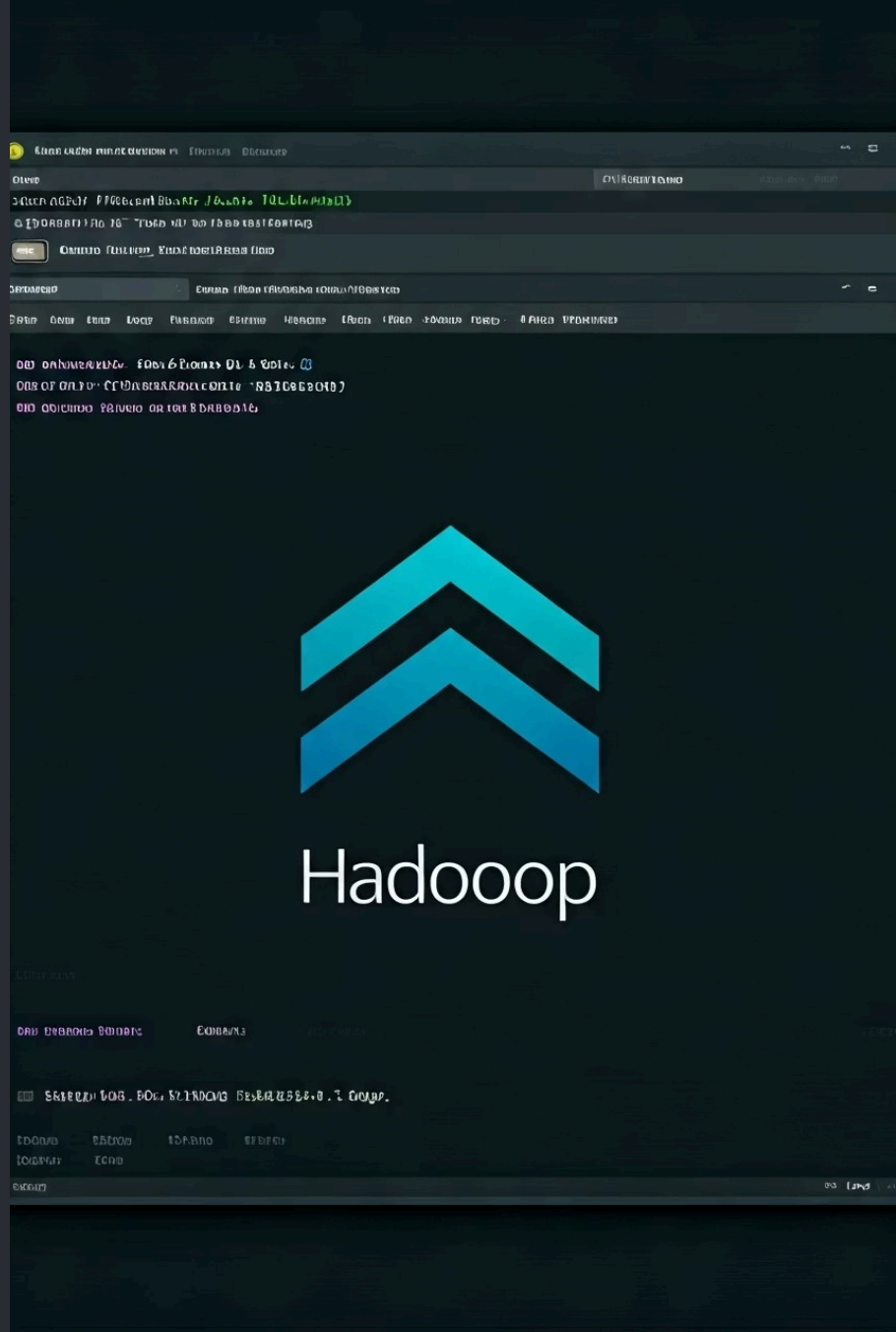
Instalación de Apache Hadoop en Linux (Ubuntu)

Guía paso a paso con explicación detallada de comandos para administradores de sistemas



Introducción a Hadoop y Linux

¿Qué es Apache Hadoop?

Apache Hadoop es un framework de código abierto diseñado específicamente para el procesamiento distribuido de grandes volúmenes de datos. Permite almacenar y procesar datasets masivos utilizando clusters de computadoras, facilitando el análisis de Big Data de manera eficiente y escalable.

Su arquitectura distribuida permite procesar petabytes de información utilizando hardware commodity, reduciendo costos significativamente.

¿Por qué utilizar Linux?

Linux, especialmente Ubuntu, se ha consolidado como el sistema operativo estándar en el ecosistema de Big Data por varias razones fundamentales: ofrece mayor estabilidad en entornos de producción, cuenta con herramientas nativas para gestión de servidores, y proporciona un rendimiento óptimo para aplicaciones distribuidas.

La comunidad Linux ofrece soporte extenso y documentación actualizada constantemente.

Objetivo del Tutorial

Instalar, configurar y ejecutar Hadoop paso a paso en Ubuntu 24.04 LTS

Requisitos del Sistema

Especificaciones Mínimas

- **Núcleos de CPU:** 2 cores (recomendado 4+)
- **Memoria RAM:** 8 GB (ideal 16 GB para producción)
- **Espacio en disco:** 30 GB mínimo disponible
- **Sistema operativo:** Ubuntu 24.04 LTS (Long Term Support)

Consideraciones Técnicas

Estas especificaciones son adecuadas para entornos de desarrollo y pruebas. Para ambientes de producción con cargas de trabajo intensivas, se recomienda incrementar significativamente los recursos, especialmente RAM y almacenamiento.

Puede utilizarse VirtualBox, VMware o instalación directa en hardware físico.

Instalación de Java Development Kit

Hadoop requiere Java para ejecutarse. Java 8 u 11 son las versiones más estables y compatibles con Hadoop 3.x. A continuación, instalamos y configuramos el JDK correctamente.

01

Actualizar repositorios del sistema

```
sudo apt-get update
```

Actualiza la lista de paquetes disponibles desde los repositorios configurados en Ubuntu, asegurando que instalemos las versiones más recientes.

02

Instalar JDK por defecto

```
sudo apt-get install default-jdk
```

Instala la versión predeterminada del Java Development Kit disponible en los repositorios de Ubuntu.

03

Instalar OpenJDK 8 (recomendado)

```
sudo apt-get install openjdk-8-jdk
```

Instala específicamente Java 8, versión ampliamente probada y recomendada por la comunidad Hadoop para máxima compatibilidad.

04

Seleccionar versión de Java

```
sudo update-alternatives --config java
```

Permite elegir qué versión de Java utilizar cuando existen múltiples instalaciones en el sistema.

05

Configurar variable JAVA_HOME

```
sudo nano /etc/profile.d/java.sh
```

Crea un archivo de configuración para establecer permanentemente la variable de entorno JAVA_HOME, esencial para que Hadoop localice Java.

06

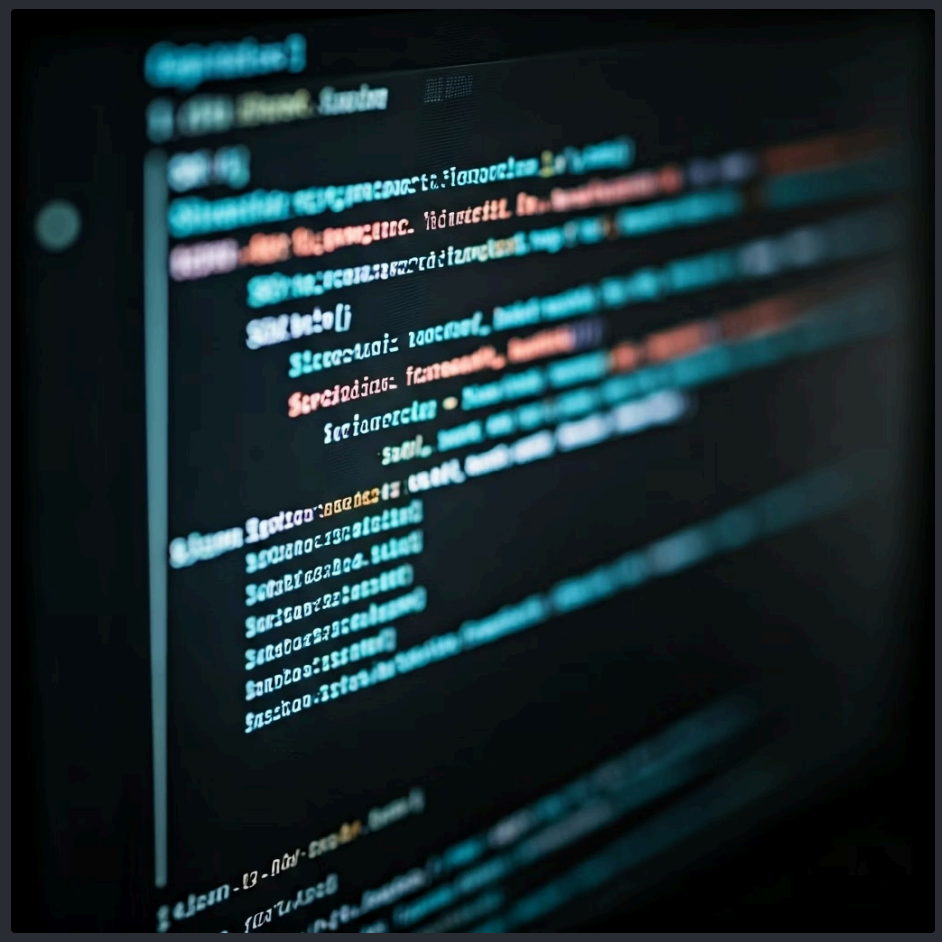
Verificar configuración

```
env | grep JAVA_HOME
```

Después de reiniciar con `reboot`, este comando confirma que la variable JAVA_HOME se configuró correctamente y está disponible globalmente.

Creación y Configuración del Usuario Hadoop

Por seguridad y organización, es fundamental crear un usuario dedicado exclusivamente para administrar y ejecutar los servicios de Hadoop. Este usuario tendrá permisos específicos y facilitará la gestión del cluster.



1 Crear usuario hadoop

```
sudo adduser hadoop
```

Crea un nuevo usuario del sistema llamado "hadoop" con su directorio home correspondiente.

2 Cambiar al usuario hadoop

```
su hadoop
```

Cambia la sesión actual al usuario hadoop recién creado para ejecutar los siguientes comandos con sus privilegios.

3 Generar claves SSH

```
ssh-keygen
```

Genera un par de claves SSH (pública y privada) necesarias para la autenticación sin contraseña entre nodos del cluster.

4 Copiar clave a localhost

```
ssh-copy-id 127.0.0.1
```

Copia la clave pública al mismo host para permitir conexiones SSH locales sin solicitar contraseña, requisito fundamental para Hadoop.

Importante: La autenticación SSH sin contraseña es crítica para que los procesos de Hadoop puedan comunicarse entre sí automáticamente. No omitas este paso.

Descarga e Instalación de Hadoop

Descargamos la versión estable de Hadoop desde el sitio oficial y la instalamos en la ubicación estándar del sistema. Hadoop 3.1.3 es una versión probada y estable, ideal para entornos de producción y aprendizaje.

Descargar Hadoop

Visita **hadoop.apache.org/releases** y descarga la versión deseada (ejemplo: hadoop-3.1.3.tar.gz). Utiliza wget o curl desde la terminal para descargar directamente.

Descomprimir archivo

```
sudo tar -zxvf hadoop-3.1.3.tar.gz
```

Extrae el contenido del archivo comprimido. Los parámetros significan: z (gzip), x (extraer), v (verbose), f (archivo).

Mover a ubicación estándar

```
sudo mv hadoop-3.1.3 /usr/local/hadoop
```

Mueve el directorio extraído a /usr/local/hadoop, la ubicación convencional para software instalado manualmente en Linux.

Asignar permisos

```
sudo chown hadoop:hadoop /usr/local/hadoop -R
```

Cambia recursivamente el propietario de todos los archivos de Hadoop al usuario hadoop, permitiéndole administrar la instalación sin permisos root.

Configuración de Variables de Entorno

Las variables de entorno permiten que el sistema y las aplicaciones localicen automáticamente los ejecutables de Hadoop y Java. Esta configuración es esencial para ejecutar comandos de Hadoop desde cualquier ubicación en la terminal.

1

Crear archivo de configuración

```
sudo nano /etc/profile.d/hadoop.sh
```

Abre el editor nano para crear un archivo de script que se ejecutará automáticamente al iniciar sesión, estableciendo las variables necesarias.

2

Agregar rutas de Hadoop y Java

```
export HADOOP_HOME=/usr/local/hadoop
export
PATH=$PATH:$HADOOP_HOME/bin:$HADOOP_HOME/sbin
export JAVA_HOME=/usr/lib/jvm/java-8-openjdk-amd64
```

Define HADOOP_HOME apuntando a la instalación, añade los directorios bin y sbin al PATH para acceder a comandos, y establece JAVA_HOME.

3

Verificar variables cargadas

```
env | grep -i -E "hadoop|yarn"
```

Tras reiniciar la sesión o ejecutar `source /etc/profile`, este comando filtra y muestra todas las variables de entorno relacionadas con Hadoop y YARN.

4

Comprobar instalación

```
hadoop version
```

Ejecuta el comando hadoop para verificar que el sistema reconoce el ejecutable y muestra la versión instalada correctamente.

Configuración de Archivos XML Principales

Hadoop utiliza archivos XML para su configuración. Cada archivo controla aspectos específicos del comportamiento del sistema distribuido. La correcta configuración de estos archivos es fundamental para el funcionamiento del cluster.

core-site.xml

Función: Define el sistema de archivos predeterminado y parámetros fundamentales del core de Hadoop.

Configuración clave: Especifica la URI del NameNode (hdfs://localhost:9000) que indica dónde reside el sistema de archivos distribuido.

hdfs-site.xml

Función: Configura parámetros específicos de HDFS, incluyendo replicación de datos y ubicaciones de almacenamiento.

Configuración clave: Define el factor de replicación (típicamente 3) y las rutas donde NameNode y DataNode almacenarán metadata y datos.

mapred-site.xml

Función: Especifica el framework de ejecución para trabajos MapReduce.

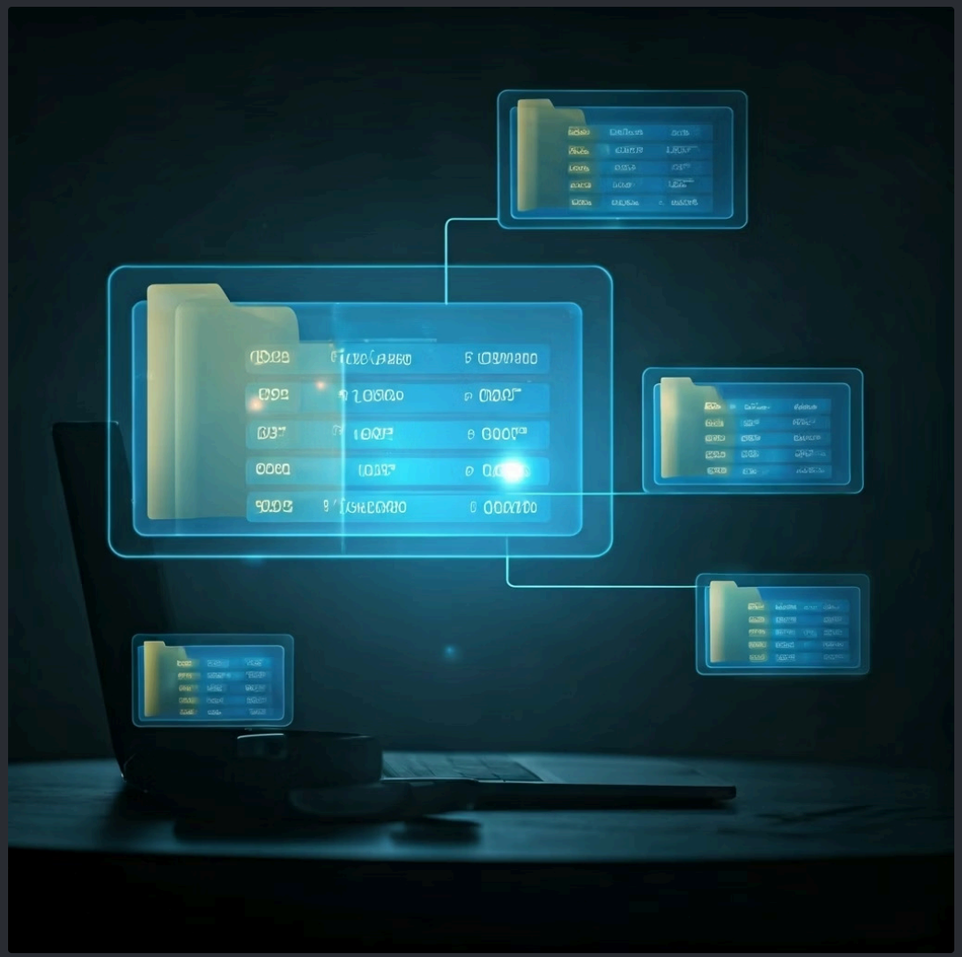
Configuración clave: Indica que se utilizará YARN (mapreduce.framework.name=yarn) como gestor de recursos para ejecutar aplicaciones MapReduce.

yarn-site.xml

Función: Configura YARN (Yet Another Resource Negotiator), el sistema de gestión de recursos del cluster.

Configuración clave: Define servicios auxiliares como el shuffle de MapReduce y especifica direcciones del ResourceManager y NodeManager.

Creación de Directorios de Almacenamiento



Antes de inicializar HDFS, debemos crear los directorios donde se almacenarán la metadata del NameNode y los bloques de datos del DataNode. Estos directorios son críticos para la operación de HDFS.

1 Crear directorio NameNode

```
sudo mkdir -p /hadoop/hdfs/namenode
```

El NameNode almacena aquí la metadata del sistema de archivos: nombres de archivos, estructura de directorios, permisos y ubicación de bloques.

2 Crear directorio DataNode

```
sudo mkdir -p /hadoop/hdfs/datanode
```

El DataNode guarda aquí los bloques reales de datos. En un cluster con múltiples nodos, cada DataNode tendría su propia copia de este directorio.

3 Asignar permisos al usuario hadoop

```
sudo chown -R hadoop:hadoop /hadoop
```

Otorga recursivamente al usuario hadoop propiedad completa sobre toda la estructura de directorios, permitiéndole leer y escribir sin restricciones.

Formateo del NameNode

> ## Paso Crítico: Inicialización de HDFS
>
> El formateo del NameNode es un paso que se ejecuta **únicamente una vez** antes del primer arranque de HDFS. Este proceso crea la estructura inicial de metadata y prepara el sistema de archivos distribuido.

Comando de Formateo

```
./hdfs namenode -format
```

Este comando debe ejecutarse desde el directorio `$HADOOP_HOME/bin` o utilizando la ruta completa. El proceso realiza varias acciones fundamentales:

- Crea el namespace inicial de HDFS
- Genera un identificador único para el cluster (clusterID)
- Inicializa los directorios de metadata del NameNode
- Establece la versión del layout del sistema de archivos
- Prepara la estructura para recibir bloques de datos

⚠️ Advertencia Importante

Nunca reformatees un NameNode que ya está en producción sin hacer backup completo de datos. El reformateo elimina toda la metadata existente, haciendo los datos inaccesibles.

Si necesitas reformatear, primero detén todos los servicios de Hadoop y respalda los datos críticos.

Inicio de Servicios de Hadoop

Una vez completada toda la configuración, podemos iniciar los servicios de Hadoop. El sistema levantará múltiples procesos Java que trabajarán coordinadamente para proporcionar las funcionalidades de HDFS y YARN.

NameNode

Gestiona la metadata del sistema de archivos y coordina las operaciones de lectura/escritura

DataNode

Almacena los bloques reales de datos y responde a solicitudes de lectura/escritura

NodeManager

Ejecuta y monitorea containers de aplicaciones en cada nodo

ResourceManager

Administra recursos del cluster y asigna recursos a aplicaciones

---

Iniciar todos los servicios

```
./start-all.sh
```

Script ubicado en `$HADOOP_HOME/sbin` que inicia automáticamente todos los daemons necesarios: NameNode, DataNode, ResourceManager y NodeManager. Alternativamente, puedes usar `start-dfs.sh` y `start-yarn.sh` por separado.

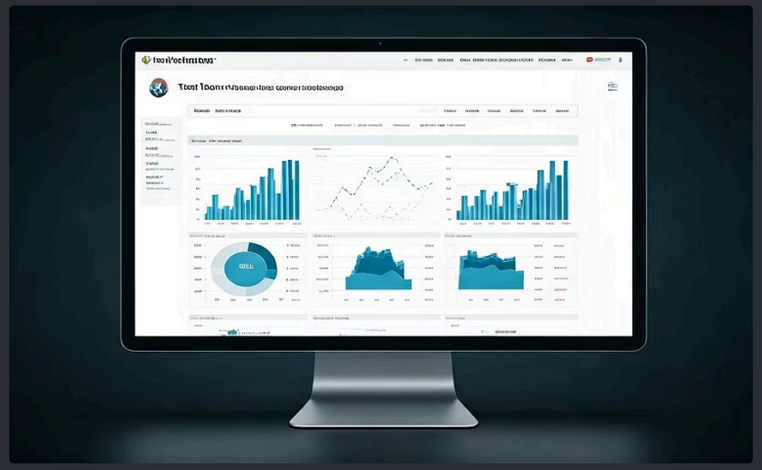
Verificar procesos activos

```
jps
```

Java Process Status muestra todos los procesos Java en ejecución. Deberías ver: NameNode, DataNode, SecondaryNameNode, ResourceManager y NodeManager. Si falta alguno, revisa los logs en `$HADOOP_HOME/logs`.

Interfaces Web de Administración

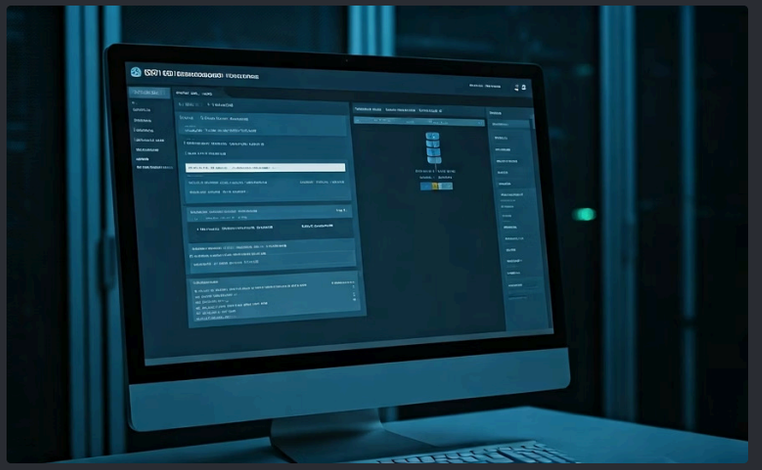
Hadoop proporciona interfaces web intuitivas para monitorear y administrar el cluster. Estas consolas son herramientas esenciales para operadores y administradores, ofreciendo visibilidad completa del estado del sistema en tiempo real.



YARN ResourceManager

URL: http://localhost:8088

Panel de control principal de YARN que muestra aplicaciones en ejecución, completadas y fallidas. Proporciona métricas de utilización de recursos (CPU, memoria), estado de los NodeManagers, colas de aplicaciones y logs detallados. Permite monitorear el rendimiento del cluster y diagnosticar problemas.



HDFS NameNode

URL: http://localhost:9870

Interfaz de administración de HDFS que presenta información sobre el sistema de archivos distribuido: capacidad total y utilizada, número de archivos y bloques, estado de salud de DataNodes, operaciones de lectura/escritura, y explorador del sistema de archivos. Esencial para gestión de almacenamiento.

Consejo: Marca estas URLs en tu navegador para acceso rápido durante el desarrollo y troubleshooting. Si instalaste en una máquina virtual, reemplaza "localhost" con la IP de la VM.

Resolución de Problemas Comunes

❌ Descarga corrupta o incompleta

Síntoma: Errores al descomprimir o archivos faltantes tras la extracción.

Solución: Verifica el checksum del archivo descargado comparándolo con el proporcionado en el sitio oficial. Descarga nuevamente desde un mirror diferente si es necesario. Utiliza `md5sum` o `sha256sum` para validar integridad.

🔒 Permisos incorrectamente asignados

Síntoma: Errores de "Permission denied" al ejecutar comandos o iniciar servicios.

Solución: Revisa sistemáticamente que el usuario hadoop sea propietario de `/usr/local/hadoop` y `/hadoop`. Ejecuta `sudo chown -R hadoop:hadoop` en ambos directorios. Verifica permisos de ejecución en scripts con `chmod +x`.

☕ Versión de Java incompatible

Síntoma: Hadoop no inicia o muestra errores relacionados con clases Java no encontradas.

Solución: Confirma que JAVA_HOME apunta a Java 8 u 11. Ejecuta `echo $JAVA_HOME` y `java -version` para verificar. Edita `hadoop-env.sh` y establece explícitamente JAVA_HOME si es necesario. Hadoop 3.x no es compatible con Java 7 o versiones anteriores.

🔑 Problemas con autenticación SSH

Síntoma: Solicita contraseña al iniciar servicios o no puede conectarse a localhost.

Solución: Verifica que `~/.ssh/authorized_keys` contenga la clave pública. Prueba `ssh localhost` manualmente para diagnosticar. Asegúrate de que el servicio SSH esté activo con `sudo systemctl status ssh`. Los permisos de .ssh deben ser 700 y de authorized_keys 600.

Conclusiones y Próximos Pasos

✅ Lo que hemos logrado

- Instalación completa de Hadoop 3.1.3 en Ubuntu 24.04 LTS
- Configuración correcta de HDFS con NameNode y DataNode funcionales
- Integración de YARN para gestión de recursos del cluster
- Establecimiento de entorno seguro con usuario dedicado y SSH sin contraseña
- Acceso a interfaces web de monitoreo y administración
- Sistema completamente operativo y listo para procesamiento Big Data

Ejecutar MapReduce

Desarrolla y ejecuta tus primeros trabajos MapReduce para procesar grandes volúmenes de datos distribuidos

Configurar Cluster Multi-Nodo

Expande tu instalación añadiendo múltiples DataNodes y NodeManagers para verdadera distribución

Integrar Ecosistema Hadoop

Instala herramientas complementarias como Hive, Pig, HBase, Spark para ampliar capacidades analíticas

> Tu entorno Hadoop está completamente configurado y operativo. Ahora cuentas con una plataforma robusta para explorar el procesamiento distribuido de datos a gran escala. El siguiente desafío es optimizar configuraciones según tus necesidades específicas y comenzar a procesar datasets reales.