UACM
Universidad Autónoma de la Ciudad de México
Nada humano me es ajeno

COLEGIO DE CIENCIA Y TECNOLOGÍA

LICENCIATURA EN INGENIERÍA EN SISTEMAS ELECTRÓNICOS Y DE TELECOMUNICACIONES

Implementación de la herramienta para Big Data "Apache Hadoop", en clústeres de sistemas digitales en el laboratorio B-404

TESIS

QUE PARA OPTAR POR EL TÍTULO DE

LICENCIADO EN INGENIERÍA EN SISTEMAS ELECTRÓNICOS Y DE TELECOMUNICACIONES

PRESENTA:

DAVID DE LA CRUZ ALEJANDRE

DIRECTOR

M. EN C. JOEL YAZBEK BUENDÍA GÓMEZ

Ciudad de México, mayo de 2021.

UACM
Universidad Autónoma de la Ciudad de México
Nada humano me es ajeno

CA Coordinación Académica

SISTEMA BIBLIOTECARIO DE INFORMACIÓN Y DOCUMENTACIÓN

UNIVERSIDAD AUTÓNOMA DE LA CIUDAD DE MÉXICO
COORDINACIÓN ACADÉMICA

RESTRICCIONES DE USO PARA LAS TESIS DIGITALES

DERECHOS RESERVADOS©

Agradecimientos

A la Universidad Autónoma de la Ciudad de México, por ser la Universidad que sin prejuicios me abrió sus puertas a todo este maravilloso tiempo de aprendizaje, por brindarme la oportunidad de cursar y de ser parte de quien ahora representara con orgullo esta gran casa de estudios, y sobre todo me ha permitido conocer a personas que me han abierto la mente y me han enseñado lo maravillosos que es luchar por lo por todo lo que imaginamos.

A Joel Yazbek Buendía Gómez, estimado profesor y director de este trabajo, le agradezco por todo el apoyo que me ha brindado, por su gran paciencia, sus grandes ideas y sobre todo por su tiempo brindado en cada momento de dudas y tropiezos, y sobre todo por haberme dado la oportunidad de formar parte de este trabajo.

A mis profesores y lectores, José Ignacio Castillo Velázquez, Ricardo Galindo Reyes y Juan Carlos Aguilar Franco, por el tiempo que me han dedicado lo largo de mi formación universitaria y por las observaciones que no solo me ayudaron a mejorar mi trabajo recepcional, sino a mejorar todo mi perfil personal y educativo como punto hacia un mejor crecimiento profesional.

Dedicatoria

A mi madre, Maria Lina Alejandre Barrera donde quiera que estés mamá y desde donde me puedas ver este trabajo es dedicado para ti, por el apoyo que siempre me diste con el empuje que siempre me enseñaste y con la dedicación y las ganas de nunca rendirse fueron lo que me impulsaron a no decaer. Por qué este logro también es tuyo.

Gracias por ser mi pilar no solo en vida sino ahora desde el cielo y nuca terminaré de decirte cuanto te agradezco porque nunca será suficiente agradecerte, podre llevarme hojas y hojas escribiendo todo lo que me has enseñado y que lo tengo tatuado en cuerpo, alma y ser y que no terminaré de escribir.

Nunca terminaré de decirte cuanto te amo y amaré sin embargo creo y sé que este mundo es una parada y que en algún momento podré verte y contarte todo lo que hice. Así que solo me queda hoy entregarte este logro más de todos lo que has visto para que lo guardes, no en una foto porque eso es solo un recuerdo y cuando la tomas no lo vives ni lo sientes, sino que lo guardes en tu corazón mamá y tus ojos donde quieras que estés.

A mis hermanos, por todo su cariño y apoyo incondicional por no dejarme solo, por la inspiración día a día Sara Reyes y Magda Reyes, gracias por siempre ser esa fuente de inspiración a seguir luchando por cumplir los sueños y sabes que si se puede. A ti Froilán Márquez por ser mi fortaleza en tiempos difíciles por tus enseñas y consejos por tu forma de compartir la vida y de valorar lo que Dios nos da. Rocio de la Cruz a ti en especial gracias por confiar en mí por recordarme siempre que mamá está con nosotros y que esta orgullosa de todo lo que hacemos por todas las enseñas que me compartiste desde pequeño y que hoy sigo aprendiendo de ti que nunca hay que renunciar a nuestros sueños y que nunca hay que darnos por vencido siempre hay que intentar hasta que Dios nos preste vida.

David Zavala, Alfredo Garcia, Victor Monsivais, Daniel Mercado, Luis Villafaña e Isaías Arellano a cada uno de ustedes gracias no solo por ser compañeros de camión, de desayunos, comidas y hasta cenas fuera de la universidad. Si no por ser una gran familia que me acompaña día a día en esta travesía, ahora somos parte de la representación de esta casa de estudios y que estoy sinceramente agradecidos por todo su apoyo, confianza y sinceridad que me ha hecho crecer no solo como estudiante sino como persona. Jessica Laija gracias por ser mi acompañante en esta faceta de mi vida por ser esa persona en que también inspira la lucha día a día por no quedarnos

atrás y seguir aprendiendo de todo y de todos. Saben que no hay palabras para compartir las emociones que siento al escribir esto, pero están en este espacio de texto y en mi vida gracias a todos.

Marco luna, Alejandro Bravo y Cristian Romero gracias por no solo ser amigos sino hermanos y que gracias a ustedes conocí lo que realmente es una amistad sincera en las buenas, así como en las malas cuando tienes y cuando no tienes. Gracias por estar en este segmento de mi vida y que son parte de este logro.

A mi tío Constantino gracias por ser como un padre para mí por adoptarme y enseñarme lo bueno que es disfrutar la vida con lo que tengas y que no es malo aprender de todo un poco, porque no sabes cuándo lo aplicaras en tu vida. Gracias por recordarme que no hay que perder el piso en ningún momento.

Para ti Norma de la Cruz y Aline Muñoz aunque no están físicamente están en mis pensamientos y son parte de mi familia les dedico este pequeño espacio en este trabajo y que quiero que sepan que forma parte de mi vida y que en algún momento tendrán la oportunidad de leer esta hoja y saber que también fue gracias a ustedes que pude concluir esta gran etapa de mi vida.

Resumen

En este trabajo se instaló Apache Hadoop sobre 4 distintos clústeres de computadoras disponibles en el laboratorio B-404, con la finalidad de determinar cuáles de ellos son aptos para el almacenamiento y procesamiento masivo de datos. Lo anterior permitió definir cuáles tecnologías fueron aptas para la instalación del Framework Apache Hadoop. A pesar de que estas tecnologías que se ocuparon son totalmente diferentes, tanto en arquitectura como procesamiento y capacidad de almacenamiento, el análisis en estas permitió identificar con mayor claridad las tecnologías aptas para trabajos futuros de almacenamiento y procesamiento.

La adaptación de las tecnologías en nuevos clústeres dentro del laboratorio B-404 para futuros proyectos se definieron con el nombre de cada tecnología de uso como: LF (LuFac) Clúster de equipos de cómputo (3 computadoras), GL(Galileo) Clúster de 3 tarjetas de desarrollo Galileo Intel Gen 2 y RP(Raspberry) Clúster con 3 tarjetas de desarrollo Raspberry Pi 2B; de la misma manera se utilizó una de las tarjetas del clúster XEXELO 0 (clúster de alto desempeño) que se encuentra ya en funcionamiento dentro del laboratorio B-404.Los resultados mostraron que solamente los clúster XEXELO 0 y LUFAC presentan características adecuadas para usarse en procesos masivos de datos.

Índice general

Índice de figuras

Índice de tablas

Introducción

En la actualidad, estamos viviendo en una nueva era de la humanidad donde la tecnología ha presentado cambios radicales en los medios tecnológicos de comunicación, de modo que la sociedad no se da cuenta del alcance que estos tienen hoy en día, cómo realizar llamadas en tiempo real a cualquier parte del mundo, tomar fotografías en alta definición o incluso expresar ideas en pocos caracteres que en segundos provocan un gran impacto en nuestra sociedad, de manera que esta evolución tecnológica ha cambiado la vida actual de la humanidad.

Cabe mencionar, que las actividades que hoy damos por hecho suceden en tan solo un instante, actividades que hace algunos años no existían y que solamente se podían presentar en la imaginación, ahora el avance de la tecnología ha cambiado nuestra calidad de vida, modificando el entorno en el que interactuamos en un solo clic, hoy en día podemos solicitar, buscar y acceder a la información en la red de redes conocida como Internet, esta información es de fácil acceso para los usuarios que cuentan con una conexión a la red, de tal forma que, hace que las búsquedas se generen en intervalos cortos de tiempo.

Así como las televisoras venden y hacen publicidad de productos o servicios a través de comerciales, hoy también todo lo relacionado con Internet cuenta con este tipo de publicidad, en específico, cada usuario conectado tiene una sección definida de datos en la Internet, que corresponde a un historial de búsqueda, que permite definir las necesidades específicas del usuario; esto permite incluir en sus ventanas de búsqueda tipos de productos que en cierto momento consultaron.

Al contrario de los comerciales en la televisión que tenían poco impacto, la publicidad en la red es específica de acuerdo con nuestras consultas, la gran cantidad de datos que como usuarios generamos se almacenan en la nube para ser explotados y analizados.

El software Apache Hadoop es un entorno de trabajo especializado en grandes cantidades de datos, el cual fue diseñado para el almacenamiento y análisis inicial de Big Data. Sus principales características y los grandes beneficios que puede brindar

son: lograr un buen análisis, almacenamiento y gestión de datos y, “como esencia principal” la velocidad del procesamiento.

El desarrollo de este proyecto se enfoca en realizar pruebas con cada tecnología de desarrollo disponible en el laboratorio B-404, instalando Apache Hadoop en un prototipo de clúster que servirá como herramienta básica para realizar: el análisis e instalación de dicho software, además de generar un documento como guía de referencia para aprender a implementar un ambiente Apache Hadoop.

El desarrollo de este proyecto se enfoca en realizar una instalación básica con cada tecnología de desarrollo disponible en el laboratorio B-404, instalando Apache Hadoop en un clúster ya creado denominado XEXELO 0, así mismo se definió la infraestructura para los nuevos clústeres denominados LF, GL y RP determinando si estos tres y XEXELO 0 son funcionales con la instalación básica de Apache Hadoop, por consiguiente servirá de referencia para utilizar solo las tecnologías aptas para trabajos futuros en clústeres o almacenamiento masivo. Este trabajo recepcional también generará un documento como guía de referencia para aprender a instalar el software Apache Hadoop.

Formulación del problema

Se desea determinar las capacidades de las tecnologías para instalar e iniciar el servicio de Apache Hadoop, como un software activo, el cual será puesto a prueba con la instalación y configuración básica en las diferentes tecnologías de desarrollo con las que cuenta el laboratorio B-404, y con ello determinar si son de utilidad estas fuentes de tecnológicas para poder seguir trabajando con el perfil de almacenamiento y procesamiento.

Objetivos generales

Implementar el marco básico de instalación de Apache Hadoop en diferentes clústeres para analizar sus prestaciones en cada sistema, e identificar la viabilidad de las tecnologías de implementación.

Crear la infraestructura de conexión y procesamiento dentro del laboratorio para determinar cuáles son las tecnologías aptas para instalar y arrancar el software Apache Hadoop y utilizar el ya creado XEXELO 0.

Por último, generar un documento que describa la instalación básica del sistema como inicio de proyectos futuros de clústeres y almacenamiento.

Objetivos particulares

1. Delimitar las herramientas de software y requerimientos de hardware para la implementación de Apache Hadoop como sistema básico de arranque.
2. Realizar la instalación e integración básica en las tecnologías para la construcción del ambiente Apache Hadoop.
3. Determinar que tecnologías son aptas para la instalación básica de Apache Hadoop.
4. Diseñar una guía para facilitar el proceso de instalación del ambiente Apache Hadoop.

Justificación

Este trabajo permitirá generar la documentación básica necesaria para los estudiantes que deseen desenvolverse en las nuevas maneras de trabajar en los clústeres de tal forma que puedan iniciar una investigación más a detalle del uso y la forma de trabajar con estos, ofreciendo una recopilación de las pruebas básicas realizadas en las tecnologías de desarrollo del laboratorio B-404, y así mismo brindar un resumen general de la instalación del software utilizado para este trabajo y de este modo delimitar que tecnologías tienen un mejor impacto para futuros trabajos en clústeres.

Alcance

El proyecto se limita a la instalación de Apache Hadoop en tecnologías con arquitecturas de 32 y 64 bits con las que cuenta el laboratorio verificando que sea funcional.

El desarrollo del proyecto incluye

1. Habilitar el panel de control central con el que cuenta Apache Hadoop de modo que se verifique el funcionamiento correcto del sistema en arranque incluyendo un resumen general de este.
2. Documentar la instalación, configuración e integración de cada una de las herramientas dentro del laboratorio, requeridas para habilitar el ambiente Apache Hadoop.

Capítulo 1. Marco teórico

Los seres humanos estamos creando y almacenando información constantemente en grandes cantidades y sin darnos cuenta. Por ello se han desarrollado técnicas y dispositivos de almacenamiento de datos, los cuales requieren ser administrados de manera clara y ordenada para acceder de manera rápida a la información que será utilizada. En este capítulo se describen los conceptos principales que se necesitan en el desarrollo de esta tesis.

1.1 Big Data

"Big Data" es un término que describe volúmenes de datos a gran escala, y relaciona todos los sistemas que se necesitan gestionar, almacenar, procesar y con ello valorar la cantidad de datos a trabajar. [1]

Para las empresas lo importante es la información que obtienen de estos datos. El Big Data provee de herramientas para analizar datos y así obtener beneficios económicos.

Los usuarios de las redes de comunicaciones generan información a través de Facebook, YouTube u otros servicios online disponibles en Internet. De manera paralela se almacenan y archivan la información que los usuarios generan, estos datos son analizados en herramientas que son destinadas en estos tipos de trabajo.

Los usuarios continúan hoy en día generando enormes cantidades de datos en la red, pero ahora no solo los humanos somos los únicos que generamos datos. Con la llegada del internet de las cosas (IoT), hay un mayor número de objetos y dispositivos conectados a Internet que generan datos como: temperatura, humedad o alguna cantidad producto de una medición. El uso de estos dispositivos conectados a Internet determina un aumento de la información.

1.1.1 Características del Big Data

Una de las características más sobresalientes del Big Data es el valor de los grandes datos que fluyen en la red, esto debido a las necesidades de buscar y encontrar lo que en la vida diaria necesitamos. Encargados de comprender y formular las consultas adecuadas para dirigir la búsqueda a lo que realmente se necesita, las empresas apuestan a las tendencias de desarrollo de nuevos proyectos, nuevos productos y tecnologías para obtener información de cada usuario.

Las características que describen el “Big Data” se les denominan como *Las “4 V” del Big Data* y son [1]:

- Volumen: Es la cantidad de datos que se generan y se tienen que almacenar y procesar alrededor de los Petabytes.
- Velocidad: Es la relación entre el tiempo y cantidad de datos que se generan, se procesan o transmiten.
- Variedad: los datos pueden generarse de diferentes fuentes y cada una puede tener formatos complejos para interpretar la información.
- Variabilidad: es otra de las dimensiones dentro del Big data que se refiere a la gran variación de los flujos de datos desde la necesidad de conectar combinar hasta filtrar la información.

1.1.2 La importancia del Big Data

Es la primera vez en la historia que los usuarios, instituciones y empresas han generado tantos y tan variados datos. La cuestión es qué hacer con ellos y cómo. Quien tenga acceso a todos estos datos, sean empresas o usuarios, y sepan cómo interpretarlos correctamente, tendrán una gran ventaja**.** [2]

Todo esto no es algo nuevo, ya que antes los usuarios solo tenían acceso a la información limitada por medio de las revistas, periódicos o comerciales de televisión. Con el tiempo, muchos hogares comenzaron con el acceso a Internet, lo que permitió el incremento de información de manera exponencial, de tal forma que ha posibilitado

que hoy en día la apuesta a las inversiones y negocios sea el enfoque al análisis de datos.

Las empresas especializadas en asumir este importante trabajo deciden dar el paso adelante, y contratar a uno o dos expertos en esta materia, para poner en marcha todo este trabajo en busca de resultados. Podría decirse, que antes se consideraba que subir información a la Internet era inservible o sin importancia, pero ahora, es el tesoro de muchas aplicaciones, productos y nuevas tecnologías. [2]

La importancia del Big Data en estos momentos para instituciones, escuelas y empresas es el tamaño de datos y la variedad de estos lo cual no gira en torno a cuántos datos se pueden tener o almacenar ya sean estructurados o no estructurados, sino qué hacer con ellos. Podemos tomar datos de cualquier fuente de información y analizarlos para hallar respuestas siempre encontrar el medio para ser explotados que hagan posibles los siguientes puntos [3]:

I. Reducción de tiempo para encontrar un producto de interés.

II. Desarrollo de nuevos productos y soluciones óptimas para estos productos de interés.

III. Toma de decisiones acertadas al momento de buscar el producto necesitado. Cuando se combina el Big Data con una analítica poderosa, se pueden realizar tareas relacionadas con negocios.

Dentro de esta misma gestión de análisis y exploración de datos nos encontramos con varias trabas dentro de esto como lo son: la captura de datos y las fuentes que lo generan; donde serán almacenados, el costo de recolección y almacenamiento “como encontrar un dato en un amplio mundo de información y de ser posible, en tiempo real” [3] como explotar analizar y encontrar respuestas concretas.

El Big data ha podido ayudar a abordar ciertas actividades empresariales, desarrollando productos y servicios de calidad haciendo partícipe a empresas como: Netflix, Amazon, Google, IBM, HP las cuales usan el Big Data para determinar la demanda de los clientes brindando estos servicios y productos.

No solo servicios de compra, sino prevenciones de accidentes o enfermedades, por ejemplo, el caso descrito en el libro Big Data La revolución de los datos masivos en donde se extrae lo siguiente [4]:

"En 2009 se descubrió un nuevo virus de la gripe. Un nuevo brote de virus conocido con el nombre de H1N1 que se expandió rápidamente.

En Estados Unidos los Centros de Control y Prevención de Enfermedades (CDC) pedían a los médicos que publicaran un llamado a la prevención. Pero publicar la información o hacer un llamado tardaba de dos a tres semanas de retraso, lo cual podía ocasionar que la gente que ya tuviera los síntomas y no tomaran precauciones el retraso de estas semanas era algo de gran impacto y crucial por el brote de la infección

Por otro lado, unas cuantas semanas antes de que el virus H1N1 ocupase los titulares, unos ingenieros de Google, habían publicado un artículo en la revista NATURE que "predecía" la propagación de la gripe y no solo eso, sino que señalaba las regiones específicas de infección y esto era gracias a que Google recibía más de tres millones de consultas diarias, las cuales archivaba grandes datos los cuales podía trabajar y analizar. Google utilizó los términos más buscados dentro de sus bases de datos para poder realizar el análisis certero".

Los retos de manejar los datos dentro de Big Data son: capturar datos y almacenarlos, considerando el costo que conlleva almacenarlos y de ser posible hacerlo en tiempo real, analizarlos y finalmente, una vez encontrada la información requerida invertir en el área correspondiente [5]. Las empresas hoy en día toman decisiones de qué productos vender o qué servicios brindar con la información recopilada.

Pero estas grandes cantidades de información serian obsoletas sino se tuviera lugar donde guardarlos (bases de datos) donde hablamos de capacidad de almacenamiento,

al mismo tiempo el gran procesamiento que se necesitan para manejar toda esta información. Las empresas requieren de resultados evaluados en ventas o requerimientos de igual modo los resultados electorales que se puedan anticipar en alguna elección.

En el siguiente apartado explicaremos la gran importancia de los clústeres y por qué son de gran relevancia dentro del almacenamiento de datos masivo, los clústeres acompañados de este conjunto de información, forma de almacenamiento, modos de procesamiento y la actualización de las tecnologías forman en conjunto parte de la nube.

1.2 Los clústeres

Hoy en día los clústeres desempeñan un papel importante en la solución de problemas de las ciencias, las ingenierías y del comercio moderno.

Un clúster es la unión de varias computadoras mediante el uso de componentes de comunicación, formando de manera lógica una sola computadora, se comporta como un solo sistema de alto desempeño que está interconectado y funciona como una sola unidad de procesamiento de información.

Para que un clúster funcione como tal, no basta solo con conectar entre sí las computadoras, es necesario proveer un sistema de manejo del clúster, el cual es el encargado de interactuar con el usuario y los procesos que se ejecutan en este.

1.2.1 Clasificación de los clústeres.

La principal tarea de un clúster es: mejorar el rendimiento y/o la disponibilidad de los servicios que se prestan, por lo cual los clústeres se llegan a clasificar dependiendo de la tarea desempeñada como [6]:

1. Clúster de Alto Rendimiento

Los clústeres de alto rendimiento necesitan: enormes cantidades de memoria, grandes capacidades de procesamiento y la combinación de ambas.

2. Clúster de Alta Disponibilidad

Su principal función es brindar el 100% de un servicio, con la confianza de que el software detecte fallas y accione ante estas al instante y el hardware sea redundante para contener fallas físicas.

3. Clúster de Alta Eficiencia

Permite realizar el mayor número de tareas en el menor tiempo posible, por este motivo este tipo de clústeres permite realizar cálculos con exactitud sin esperar demasiado.

1.3 Escalabilidad

Un sistema escalable es aquel que puede aumentar su capacidad, agregando equipos de cómputo al sistema, por consecuencia mejora el rendimiento y las tareas asignadas se ejecutan con mayor velocidad.

1.4 Almacenamiento

Almacenar información, requiere de sistemas donde sea posible escribir y leer a una gran velocidad, ya que la cantidad de datos cada vez va aumentando de manera exponencial. Nuevas tecnologías y soluciones son presentadas diariamente.

El incremento de los datos y la disponibilidad de los servicios va en aumento, por tanto, las empresas han trabajado en servicios y almacenamientos masivos algunos ejemplos son Amazon, Microsoft Azure e IBM, que se han enfocado en almacenar y brindar servicios de calidad.

El almacenamiento en la nube es una solución personalizada que permite administrar de manera centralizada cualquier información dentro de la red desde cualquier parte del mundo. Además de eso, el servicio de la Nube también está por detrás de las tecnologías de almacenamiento y procesamiento las cuales puedan brindar.

La distribución de los servicios de la nube no se ven a simple vista del lado del usuario, ya que, sin tener idea de la importancia, de acceder a nuestros servicios, archivos e imágenes en cualquier momento, desde cualquier dispositivo suele no importar (Ver figura 1). El poder acceder a ciertos servicios, es lo que en realidad todas las empresas quieren brindarles a los clientes es buscar “servicios sin interrupciones”.

Figura 1. Almacenamiento en la nube y acceso

1.5 Sistemas Distribuidos.

Los sistemas distribuidos se definen como: "*Sistemas cuyos componente hardware y software, que están en computadoras conectadas en red, se comunican y coordinan sus acciones mediante la interfaz de paso de mensajes, para el logro de un objetivo"* [7]. Lo cual permite repartir las tareas en cada servidor o computadora conectado en la misma red, dividiendo las tareas con la idea de que el trabajo no se cargue en una sola máquina, lo cual permite prestar un servicio sin saturar la red mejorando el rendimiento y procesamiento de las actividades que se requieran ejecutar.

1.5.1 Servicios distribuidos

Desde la perspectiva del modelo distribuido que predomina en la actualidad, este permite repartir las tareas, procesamiento y recursos, sobre todo, en cada máquina presente en el servicio, por ello las interfaces gráficas como servicio son primordiales para atender las actividades en paralelo de cada usuario que solicite un servicio o ejecute un comando en la red, con ayuda de esto se permite repartir tareas y resolver solicitudes que los clientes requieran.

Los servicios que hoy en día provee Internet son dedicados, esto quiere decir que las aplicaciones, páginas web, servicios de correo electrónico entre otros (Ver figura 2) están atendiendo en cualquier instante de tiempo el servicio configurado. Las solicitudes requeridas por los clientes son puntuales por lo cual se espera que al momento de solicitar la petición del servicio pueda ser atendido de forma rápida y clara.

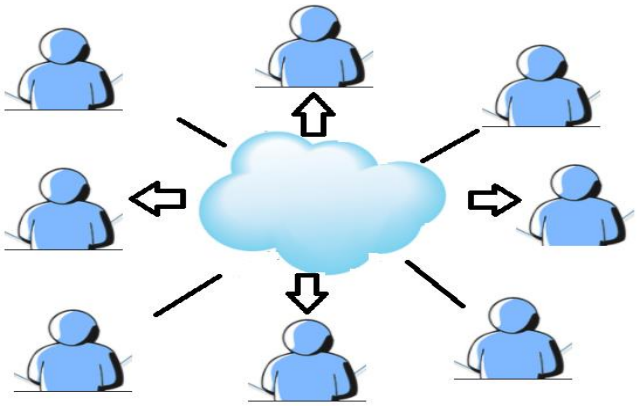


Figura 2. Datos y servicios dedicados.

1.6 Cloud Computing

Definir con precisión el término Cloud seria entrar en una gran discusión, ya que es impreciso definir el término por su infinidad de significados. Pero una de las definiciones que podemos manejar de manera más clara dentro de este trabajo y aceptadas y pertinente que adapta el National Institute for Standars and Tecnology de los Estados Unidos:

"«La computación en la nube es un modelo para permitir el acceso a la red ha pedido, conveniente y ubicuo a un conjunto de recursos informáticos configurables (por ejemplo, redes, servidores, almacenamiento, aplicaciones y servicios) que se puede aprovisionar y liberar rápidamente con un mínimo esfuerzo de gestión o interacción del proveedor de servicios.

Este modelo de nube se compone de cinco características esenciales, tres modelos de servicio y cuatro de implementación

modelos.»" [8]

Es importante destacar que la información recopilada en textos, videos, imágenes y música permite conocer las nuevas tendencias de tal manera en que subimos información ya sea personal o general, permite determinar los servicios que la comunidad necesita como: correo electrónico (Yahoo!), música (Spotify), fotos y videos (Facebook), localización en tiempo real (Google Maps) entre otras. La mayor parte de estos servicios son suministrados por centros de datos instalados por empresas como: Google, Microsoft, IBM, Dell, Oracle, Amazon entre otras. Las cuales se han dedicado a generar y brindar servicios de calidad [2].

Todas estas características como: servicio, calidad, rapidez, atención y seguridad que el Cloud Computing desarrolla, son visibles a ojos de los clientes, de tal forma que solo el cliente se preocupa por un buen servicio de calidad con alta demanda y disponibilidad, cualidades que el Cloud Computing o mejor conocida como “nube” brinda en cada servicio día con día.

Muchos de nosotros solo decimos o utilizamos el término, sin saber realmente el impacto que tiene el término nube en la información que hoy día se maneja, y solo decimos “¡Súbelo a la nube!”. Pero realmente, qué significa o asimismo a que nos referimos con el término nube, en la mayoría de los usuarios este término se usa de manera ambigua.

Es importante destacar que los usuarios o clientes de la nube solo almacenan y solicitan servicios, sin importar las limitantes o costos de software, hardware, actualizaciones de sistemas, así como la contratación de nuevos empleados y consultores. Esto indica que los usuarios y clientes no intervienen en la construcción de estos grandes servicios, de manera que se aplica una tarifa medida por uso o cuota fija de suscripción de los servicios que se brindarán al cliente con la política “Use solo lo que quiera, pague solo lo que usa”, uno de los muchos servicios que se manejan hoy en día con esta política es Netflix, Spotify, Amazon entre otros.

El cómputo en la nube realmente toma el modelo que se ha generalizado. Es una forma nueva y evolucionada de cómputo de utilidad en el cual muchos de los recursos (hardware, software, almacenamiento, comunicaciones, etc.) pueden ser combinados y recombinados sobre la marcha dentro de las capacidades específicas o de los servicios que los clientes requieren. Desde procesos de manera rápida y precisa para proyectos de alto desempeño hasta la capacidad de almacenamiento para respaldos de grado empresarial, el cómputo en la nube puede hoy en día ser entregado virtualmente en cualquier capa de TI.

1.6.1 Servicios de implementación dentro de Cloud Computing

“El Instituto Nacional de Estándares y Tecnología (National Institute of Standards and Technology) determina el modelo de la nube con cinco características esenciales, tres modelos de servicio y cuatro modelos de despliegue” [2] dentro del mismo núcleo

la nube está conformada por dos autores principales dentro del desarrollo que son: vendedores y proveedores motivos por el cual proporcionan y facilitan las tecnologías, creando y ofreciendo los servicios para los clientes o usuarios finales que utilizan estos servicios de manera libre o delimitada dependiendo la demanda del servicio.

La idea de la “nube” se puede presentar de maneras diversas dependiendo las múltiples necesidades y las ventajas que estas puedan traer, como usuarios de esta gran tecnología. Los servicios más usuales que demanda la nube según el instituto NIST para los usuarios finales y empresas son [1]:

- Software como servicio (SaaS).

Las aplicaciones basadas en Cloud, o software como servicio, se ejecutan en sistemas distantes, escritos en diferentes lenguajes de programación e instalados en cualquier parte de una región. Donde reside un software instalado previamente configurado que pertenecen y son administrados por terceros, conectando a los usuarios a través de Internet y, por lo general, a través de un navegador web, permitiendo la accesibilidad desde cualquier dispositivo con conexión haciendo ligera la disponibilidad y uso del software.

- Plataforma como servicio (PaaS).

Este servicio permite que en mayor parte los desarrolladores en esencia de software desarrollen aplicaciones sobre la nube, aprovechando la capacidad máxima de la arquitectura rentada ofreciendo recursos de solicitud y almacenamiento como: Windows Azure, Google App Engine, Amazon entre otras.

- Infraestructura como servicio (IaaS)

Las implementaciones, aplicaciones y servicios que la nube puede ofrecer, permiten definir de manera clara el recurso, servicio y tarea, motivó que permite dividir a la nube en tres modelos definidos como [1]

1.6.2 Despliegue de la nube

Las implementaciones, aplicaciones y servicios que la nube puede ofrecer, permiten definir de manera clara el recurso, servicio y tarea, motivó que permite dividir a la nube en tres modelos definidos como (*Diferencias entre una nube pública, una nube privada y una nube híbrida* | *Microsoft Azure*):

- NUBE PÚBLICA: Son administradas o gestionadas por empresas que prestan servicios y las cuales atienden a grupos plurales de clientes (público en general, grupo industrial) con ayuda de los servidores, sistemas de almacenamiento y otras infraestructuras que se utilizan de forma corporativa, motivo por el cual ofrecen servicios de manera general a través de Internet.

- NUBE PRIVADA: integrada por una organización independiente que usa tecnologías de computación en la nube. Se caracterizan por ser administradas por los mismos propietarios del servicio por dentro o fuera de la red con permisos exclusivos

- NUBE HÍBRIDA: Es una combinación de los tipos anteriores lo cual da origen a que los clientes puedan ser propietarios de unas partes, compartir otras de manera controlada.

En el siguiente capítulo encontraremos la descripción y objetivos de Apache Hadoop como framework, así mismo se definirán los sistemas operativos y tecnologías donde se instaló Apache Hadoop, en consecuencia, se podrá definir las tecnologías que son aptas para el sistema que se utilizará en el laboratorio de redes de computadoras.

Capítulo 2 Descripción de la Infraestructura del Servicio Apache Hadoop

La necesidad de cómputo en las aplicaciones de servicios tecnológicos ha crecido a lo largo del tiempo de manera interrumpida. Para atender estas demandas, la investigación en computación ha seguido principalmente dos caminos: incrementar la capacidad de cómputo de las computadoras, y desarrollar técnicas de cómputo distribuido y paralelo.

En este capítulo se busca que los lectores conozcan Apache Hadoop, un framework diseñado en java, para el almacenamiento y análisis de Big Data, de modo que se mostrarán las ventajas de implementar esta herramienta en diferentes tecnologías de hardware dentro del laboratorio de redes de computadoras. De la misma manera se describirán las características de los sistemas operativos y los grandes beneficios que puede brindar Apache Hadoop como framework implementado.

2.1 Apache Hadoop

Apache Hadoop es un marco de referencia (framework) que permite el almacenamiento y procesamiento de enormes cantidades de datos en un clúster, existen tres formas de ejecutar Apache Hadoop las cuales son [18]:

1) Standalone: Es el modo de operación por defecto en un solo nodo, no se necesita alguna configuración de clúster.

2) Servidor: Es un sistema basado en un cliente servidor, que se ejecuta todo en modo local.

3) Distribuido: Cuenta con una infraestructura de comunicación completa con varios nodos de almacenamiento.

Esencialmente Apache Hadoop funciona como una herramienta de almacenamiento, Inspirado en el proyecto de Google denominado “File System” Apache Hadoop abre una vertiente dentro del análisis de datos con sus tres arquitecturas fundamentales de creación denominadas:

1. Hadoop MapReduce
2. Hadoop Distributed File System (HDFS)
3. Hadoop Common

2.2 Arquitecturas principales de Apache Hadoop

Apache Hadoop incluye varios módulos motivo por el cual permite ampliar la manera de implementar sus herramientas de forma específica, dando soporte al procesamiento y análisis de grandes datos, esto nos permite dividir de forma clara las arquitecturas fundamentales como:

2.2.1 Hadoop MapReduce

Consiste en dos funciones definidas por el usuario: "*map"* y "*reduce"* dando lugar a dos espacios, por este motivo este conjunto de pares es definido como (clave-valor) por tanto la entrada se define con este espacio de dos condenadas (k, v). La función "*map"* se manda a llamar para cada uno de estos pares definidos, la función "*map"* una vez ejecutada determina clave-valor como (k´, v´) intermedios.
El framework ordena estos valores por medio de (k´) mandado a llamar a la función "*reduce"* para cada grupo de parejas donde "*reduce"* produce un cero o más resultados definidos [18]

En otras palabras, la función *"map"* trabaja con grandes volúmenes de datos, se encarga de dividirlos en varias partes, cada una de ellas con colecciones de registros, luego la función "*map"* se ejecuta por cada colección de datos, para finalmente calcular el conjunto de valores intermedios basados en el procesamiento de cada registro (Ver figura 3).

La Función Reduce se ejecuta por cada elemento del conjunto de valores intermedios obtenidos, lo que hace es reducir el conjunto de los valores que comparten una clave para obtener un conjunto más pequeño (Ver figura 3).

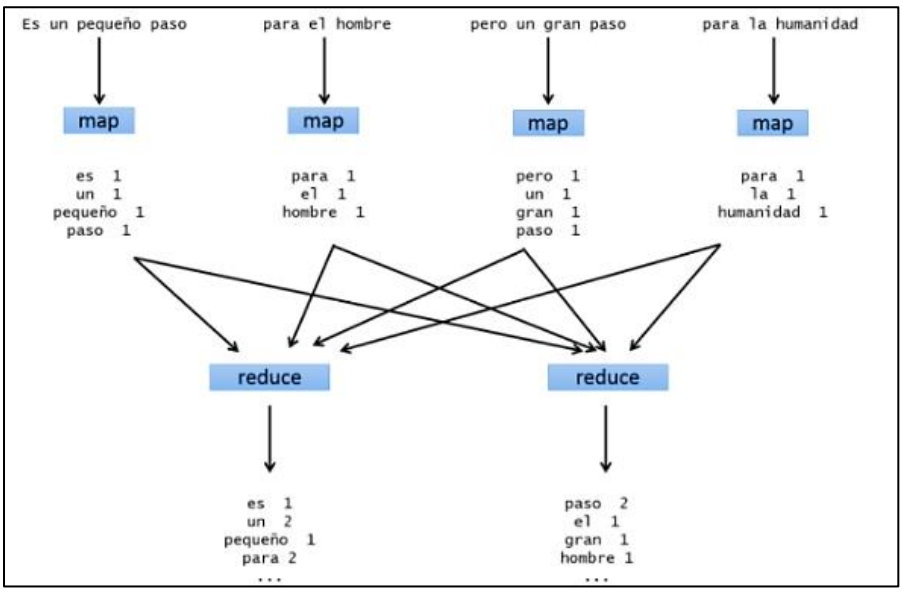


Figura 3. Ejemplo de las funciones Map-Reduce

2.2.2 Sistema de archivos distribuido de Hadoop (Distributed File System HDFS)

Es un sistema de archivos distribuido que proporciona acceso de alto rendimiento diseñado para almacenar grandes cantidades de datos en consecuencia poder leer y escribir esto se debe a que HDFS se basa en la idea del patrón de procesamiento leyendo una vez y escribiendo muchas veces, de modo que implica en su mayoría el conjunto de todos sus nodos MapReduce utiliza en conjunto esta herramienta. (White, 2007, págs. 43-50)[10]

Este tipo de aplicaciones que se ejecutan dentro de HDFS necesitan tener un tiempo de respuesta de corto plazo lo cual genera una reducción de tiempo permitiendo la entrega de servicio sin retrasos.

Este sistema de archivos permite tener un gran impacto dentro de la escalabilidad y disponibilidad debido a la recolección de datos y a la tolerancia a los fallos. Los componentes son los siguientes:

- NameNode: Se trata de la máquina maestra del clúster, se encarga de regular el acceso a los archivos por parte del cliente, controla el sistema de archivos que tiene cada nodo y los bloques de cada uno de estos manteniéndolos en memoria.

- DataNode: Son los nodos conectados a la máquina maestra del clúster, son los esclavos encargados de leer y escribir los requerimientos de los clientes. En cada nodo se encuentran replicados los archivos que están compuestos por bloques.

Cuando un archivo es cargado al clúster HDFS es dividido en bloques, estos bloques son distribuidos a través de los nodos del clúster. De igual forma mediante la distribución se realiza la replicación. El factor de réplica es configurable con respecto al usuario, pero generalmente se establece en tres, permitiendo que HDFS sea tolerante a fallas. La figura 4 describe la forma de carga de un archivo a HDFS (White, 2007, págs. 63-75) [10].

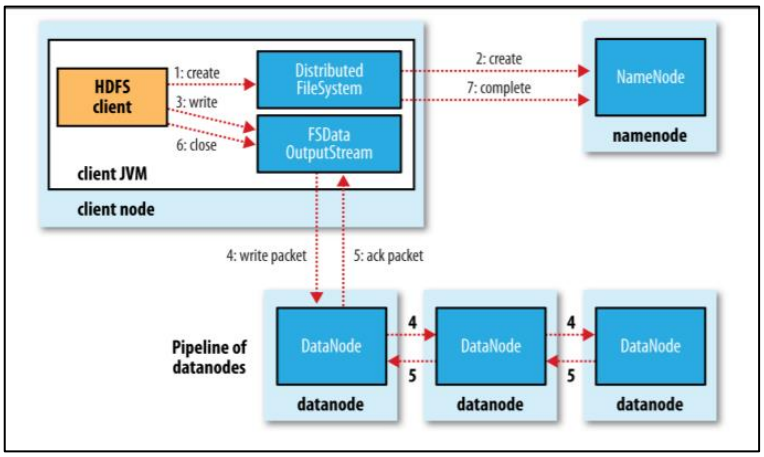


Figura 4. Distribución de archivos a los nodos [10].

Los bloques dentro de HDFS son en esencia más grande en comparación con los de un disco, el motivo de que los bloques de HDFS sean más grande es que se busca reducir el tiempo de búsqueda de información en estos bloques por lo que “asignar tareas en MapReduce normalmente este operar en un bloque a la vez, por lo que si tiene muy pocas tareas (menos que los nodos en el clúster), sus trabajos se ejecutarán más lento de lo que podrían hacerlo de otra manera” (White, 2007, pág. 72) [10] por este motivo usar esta herramienta hace énfasis a tener un mayor número de nodos para trabajar.

- Características de HDFS

● Almacenamiento de datos distribuido, tolerante a fallos.

● Analogía con un sistema de archivos, no una Base de Datos.

● HDFS ha demostrado que es escalable hasta 200 Peta Bytes de almacenamiento en un clúster de 4500 nodos

2.3 Distribuciones de sistemas operativos

En esta sección se presentarán las características de los sistemas operativos (OS) que se utilizaron en el proyecto para la implementación de Apache Hadoop. Respecto al software, los clústeres que se implementaron en cada tecnología se utilizó la misma configuración no importando la distribución del sistema operativo.

2.4 Distribución CEntOS

CEntOS (Community Enterprise Operating System) es un sistema operativo, de código libre y abierto para cualquier persona que desee utilizarlo. Es una distribución que se mantiene activa por paquetes liberados por la distribución Red Hat, de esta manera CentOS Linux se enfoca en ser operacionalmente compatible con RHEL (Red Hat Enterprise Linux).

Por lo cual CEntOS es totalmente una distribución libre y no hay que pagar por ella para usarla, además desde la versión de CEntOS 5 cada versión es mantenida por 10 años, por medio de actualizaciones de seguridad, de la misma manera los intervalos de actualización han quedado más tiempo por la relación de los paquetes fuentes que se liberan.

CEntOS es creado o desarrollado por un pequeño conjunto de desarrolladores que han ido creciendo con el tiempo, a la vez que los desarrolladores centrales están apoyados por un gran número de usuarios que trabajan activamente, entre ellos, los

usuarios de empresa, los administradores de red, administradores de sistemas, gerentes, principales contribuyentes de Linux de todas partes del mundo (*Sobre CentOS*). (acerca de CentOS., 2019) [11]

- Requisitos de instalación bajo la premisa de CEntOS

Para tener la mejor instalación de CentOS es necesario una serie de requisitos mínimos del sistema. En principio tenemos que diferenciar entre dos tipos de entorno de instalación.

Con entorno de escritorio, con eso nos referimos a que ya existe un conjunto principal de programas que ofrece al usuario una sencilla y cómoda interacción con su equipo, los requisitos de hardware para esta opción son (*Sobre CentOS*) (acerca de CentOS., 2019)[11]:

- Memoria RAM: 2GB, como mínimo.
- Espacio en el disco duro: 20 GB, como mínimo, pero es recomendado 40 GB.
- PROCESADOR: Intel x86-compatible (32 bit) y x86-64 (64 bit)

Sin entorno gráfico en versión minimal, que es el segundo modelo que ofrece CentOS, los requisitos de hardware son:

- Memoria RAM: 64 MB, como mínimo.
- Espacio en el disco duro:1024MB, como mínimo, pero es recomendado 2GB.
- Procesador: Intel x86-compatible (32bit) y x86-64 (64 bit)

- Sistema de archivos y directorios de CentOS

En Linux y Unix todo se podría considerar como archivos, las carpetas son archivos, y los dispositivos son considerados archivos. Todos se organizan en una estructura jerárquica, de tipo árbol. El nivel más alto de archivos es “/” o directorio raíz. Así sucesivamente todos los directorios están debajo de la raíz.

Por debajo de este hay un gran importante número de grupos de directorios común a la mayoría de las distribuciones de GNU/Linux.

A continuación, se mostrarán algunos de los directorios que aparecen normalmente bajo raíz (/) (acerca de CentOS., 2019) [11]:

- /bin : Aplicaciones *bin*arias importantes.
- /boot : Archivos de configuración del arranque.
- /dev : Los archivos utilizados como entrada y salida
- /etc : Archivos de configuración, scripts de arranque, *etc*.
- /home : Directorios personales de los usuarios
- /lib : Bibliotecas del sistema.
- /mnt : Sistemas de archivos montados manualmente en el disco duro.
- /opt : Ubicación donde instalar aplicaciones opcionales.
- /proc : Directorio dinámico que mantiene información sobre el estado del sistema.
- /root : Directorio del usuario principal con permisos completos.
- /sys : Archivos del sistema.
- /tmp : *Archivos temporales*.
- /usr : Aplicaciones y archivos de acceso a usuarios
- /var : Archivos de varios tipos, como archivos de registros y bases de datos.

2.5 Distribución Debian

Debian es un proyecto de personas que han creado un sistema operativo libre. El cual es un conjunto de programas y utilidades básicas que hacen que la computadora, realice su trabajo básico y le permite ejecutar otros programas. De la misma manera que CentOS (Debian - Acerca de Debian. (2020), 2019) [12].

La atención que pone Debian a los detalles nos permite producir una distribución de alta calidad, estable y escalable. La instalación puede configurarse fácilmente para cumplir diversas funciones, desde cortafuegos reducidos al mínimo, a estaciones de trabajo científicas o servidores de red de alto rendimiento en este caso Apache Hadoop.

Debian es un sistema operativo (S.O.) de libre distribución, nace como una apuesta por separar en sus versiones el software libre del software no libre, para esto debe respetar 4 libertades:

1. libertad para usarlo.
2. libertad para modificarlo.
3. libertad para copiarlo.
4. libertad para distribuir las modificaciones.

Comparación de las distribuciones CentOS vs Debian

Se presenta en la tabla 1 algunas comparaciones de los dos (SO) en la tabla, también se muestra en la figura 5 como podríamos encontrar los emblemas principales de los sistemas operativos ya mencionados.

CentOS	Debian
Red Hat distribución más usada por la compatibilidad	No tiene tanta demanda por la compatibilidad
Problema con las actualizaciones mayores de X a Y	Actualizaciones mucho más sencillas de repositorios y versiones
Estabilidad	Estabilidad
CentOS es propiedad del corporativo de Red Hat ampliamente utilizado por empresas.	Debian es un proyecto comunitario que no es ligado a ningún corporativo.

Tabla 1. Comparación de los S.O CentOS Y Debian

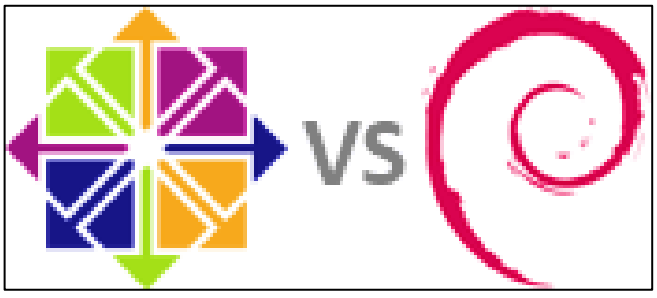


Figura 5. Emblemas de los sistemas operativos utilizados CentOS y Debian.

2.6 Plataforma Java

Es un lenguaje de programación de propósito general con la intención de que los programas se escriban una sola vez y se puedan ejecutar en cualquier dispositivo. Lo cual es posible, ya que JAVA cuenta con una máquina virtual JVM (Java Virtual Machine) (Java, 2016)[13] que brinda portabilidad al lenguaje, ya que es el más usado en la mayoría de las plataformas de la industria.

2.6.1 Descripción de Java

Supongamos que un cliente nos encarga una aplicación y nosotros, en un entorno de desarrollo Windows, diseñamos lo que el cliente nos está pidiendo diseñamos tanto el código fuente y la compilamos para que se genere el archivo ejecutable. Le damos el archivo ejecutable al cliente que lo prueba en su sistema operativo Windows y funciona con éxito. Pero a los pocos días nos llama y menciona que la aplicación no funciona en sus sistemas Linux.

La aplicación no es portable para otros sistemas. El código fuente que hemos diseñado lo hemos compilado (Ver figura 6) para que genere un lenguaje máquina entendible para el sistema operativo Windows, pero ese archivo ejecutable otro sistema operativo no lo entiende.

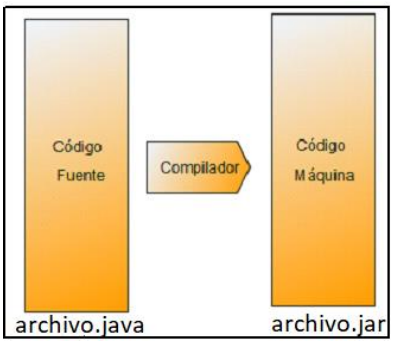


Figura 6. Proceso habitual de compilación.

Java es un lenguaje de programación con el que podemos realizar cualquier tipo de programa. En la actualidad es un lenguaje muy extendido, está desarrollado por la compañía Sun Microsystems, siempre enfocado a cubrir las necesidades tecnológicas más puntuales (Java, 2016)[13].

Una de las principales características por las que Java se ha hecho muy famoso es que es un lenguaje independiente de la plataforma. Eso quiere decir que si hacemos

un programa en Java podrá funcionar en cualquier computadora del mercado. Es una ventaja significativa para los desarrolladores de software, pues antes tenían que hacer un programa para cada sistema operativo. Esto lo consigue porque se ha creado una Máquina de Java para cada sistema que hace de puente entre el sistema operativo y el programa de Java y posibilita que este último se entienda perfectamente.

La independencia de plataforma es una de las razones por las que Java es interesante para Internet, ya que muchas personas deben tener acceso con computadoras distintas.

Java tiene dos conjuntos de herramientas que nos permiten comenzar a trabajar con este entorno como JDK y JRE

2.6.2 Kit de Desarrollo de Java (JDK)

El JDK es el Java Development Kit, que traducido significa herramientas de desarrollo para java, en él encontramos un compilador javac que es el encargado de convertir nuestro código fuente (.java) en Byte-code (.class), esto posteriormente será interpretado y ejecutado con la JVM, también encontramos herramientas que nos permiten generar los javadoc (encargado de generar la documentación de nuestro código), de la misma manera encontramos jvisualvm, de tal manera que nos muestra la información a detalle sobre las aplicaciones que están corriendo actualmente en la JVM.

2.6.3 JRE

El compilador de *JAVA* compila el código fuente a un código máquina intermedio, llamado *Byte-code*, que el JRE interpreta según el sistema operativo, pasándolo al código máquina que este entienda ver figura 7.

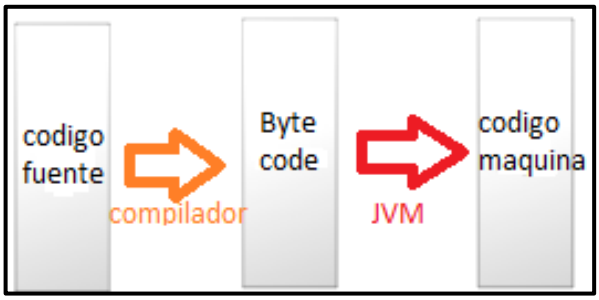


Figura 7. Proceso de compilación en Java.

Para cada dispositivo, por tanto, debe haber una JRE específica, ya sea un teléfono móvil, un PC con sistema operativo *Windows* XP o un horno que usa microondas. De forma que JRE conoce el conjunto de instrucciones de la plataforma destino, y traduce un código escrito en lenguaje *Java* (común para todas) al código nativo que es capaz de entender el hardware de la plataforma.

Una de las preguntas dentro del ambiente de desarrollo con java al iniciar es ¿Podemos instalar el JDK sin el JRE?, la respuesta es no, pero podemos instalar JRE sin el JDK, la respuesta es sí, debido a que el JDK está destinado a usuarios que requieran crear aplicaciones en el lenguaje java.

2.7 Relación de las arquitecturas y sistemas operativos con Hadoop.

La arquitectura de los diferentes hardware permitió determinar el rendimiento, así como la capacidad de poder implementar Big Data dentro del ambiente de Apache Hadoop.

2.7.1 Tarjeta de desarrollo Raspberry Pi

El desarrollo reciente de la placa Raspberry Pi ha brindado nuevas oportunidades para mejorar las herramientas para la educación. El bajo costo significa que podría ser una opción viable para desarrollar soluciones en los sectores de educación en los países en desarrollo [14] por ende en la construcción de este clúster se ocuparon 3 tarjetas de desarrollo disponibles en el laboratorio generando un pequeño clúster. A continuación, se describen los tipos de Raspberry Pi.

Raspberry Pi Modelo B

Es una placa de computadora (SBC) Single Board Computer de bajo coste, motivo que permite nombrarlo como una computadora portátil, aproximadamente del tamaño de una tarjeta de crédito mostrada en la figura 8, desarrollada en el Reino Unido por la fundación Raspberry Pi en la Universidad de Cambrige en 2011, con el objetivo de estimular la enseñanza de la informática en las escuelas [14].

El concepto general de la tarjeta es que soporte varios componentes conectados a la computadora en común, posibilitando el trabajo con su procesador, por este motivo podemos decir que actúa como una computadora. Raspberry es un producto de software libre de código abierto, por ello podemos instalarle un S.O. de código abierto, las distribuciones de Debian tienen compatibilidad con esta tarjeta en especial el S.O. denominado Raspbian.

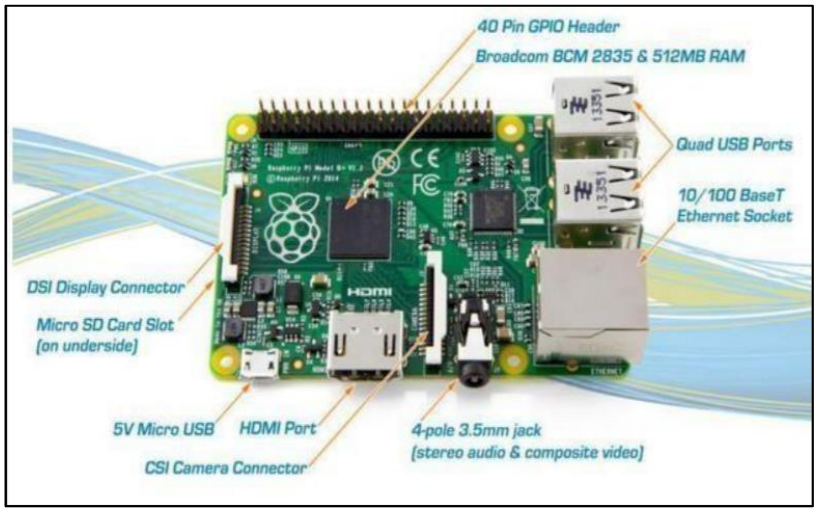


Figura 8. Tarjeta Raspberry Pi B [21].

SoC	Broadcom BCM2835
CPU	ARM 1176JZFS a 700 MHz
GPU	Videocore 4
RAM	256 MB
Video	HDMI y RCA
Resolución	1080p
Audio	HDMI y 3.5 mm
USB	2 x USB 2.0
Redes	Ethernet 10/100
Suministro de energía eléctrica	micro USB

Tabla 2. Especificaciones Tarjeta Raspberry Pi B

En todas las versiones de Raspberry incluye un procesador Broadcom, una memoria RAM, una GPU, puertos USB, HDMI, Ethernet, 40 pines GPIO y un conector para cámara, ninguna de sus ediciones incluye memoria para almacenamiento, siendo esta en su primera versión una tarjeta MicroSD mostrada en la tabla 2 la cual refiere a las especificaciones técnicas.

2.7.2 Tarjeta de desarrollo Intel Galileo

Intel se compromete a proporcionar los mejores procesadores, tableros y herramientas a su comunidad. La primera iniciativa de Intel es la introducción de las placas Intel Galileo Gen 2, que son compatibles con ciertos encabezados de software con Arduino.

La placa Intel galileo gen 2 es la primera en una familia de placas de desarrollo y prototipos certificados por Arduino, basadas en la arquitectura Intel y específicamente diseñadas para fabricantes, estudiantes y educación. La placa Intel galileo Gen 2 figura 9, que ofrece a los usuarios un entorno de desarrollo de hardware y software de código abierto, completa y amplía la línea de productos Arduino para ofrecer funciones informáticas más avanzadas con lo que se familiarizaron con los prototipos Arduino. La placa de desarrollo Intel Gen 2 ha sido diseñada para la compatibilidad con

hardware, software y PIN, con una amplia variedad de placas Arduino de la misma manera se incorpora la programación esquema Arduino [15].

Especificaciones Técnicas [15]:

- El procesador de aplicaciones Intel® Quark™ SoC X1000, una arquitectura de conjunto de instrucciones de procesador Intel® Pentium® de 32 bits, con un solo núcleo y subproceso compatible con ISA, que funciona a velocidades de hasta 400 MHz.
- Compatibilidad con una amplia variedad de 37 interfaces de E/S estándar en la industria, entre ellas la ranura mini-PCI Express de tamaño completo, el puerto Ethernet de 100 Mb, la ranura microSD, el host USB y el puerto cliente USB.
- DDR3 de 256 MB, SRAM de 512 kb integrada, Flash NOR de 8 MB y EEPROM de 8 kb estándar en la placa, más compatibilidad con tarjeta microSD de hasta 32 GB.
- Compatibilidad de hardware y pines con una amplia variedad dispositivos externos.
- Programable a través del entorno de desarrollo integrado (IDE) Arduino que es compatible con los sistemas operativos host Microsoft Windows, Mac OS y Linux.
- Se utiliza un sistema operativo específico dentro de esta tarjeta en el desarrollo de Linux denominado Debian, que permite el desarrollo de las configuraciones pertinentes acordes al proyecto.

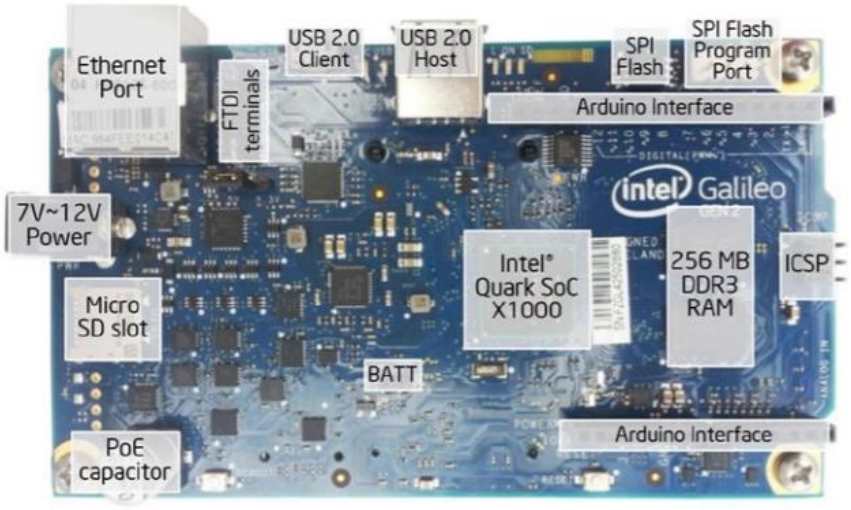


Figura 9. Placa Intel Galileo Gen 2 [15].

Para este clúster se ocuparon 3 tarjetas de desarrollo Intel Galileo Gen 2 que permitieron instalar y medir el uso de software Apache Hadoop. Con la finalidad de poder extender la infraestructura de este clúster se necesitaban los resultados de estas 3 primeras tarjetas, resultados que más adelante se exponen, que permitirían expandir el clúster a más de 10 tarjetas.

2.7.3 Computadoras Lufac

LUFAC es una computadora de escritorio personal, diseñada y construida para ser utiliza dentro del laboratorio B-404 en el periodo de prueba de este clúster se utilizaron 3 computadoras Lufac que permitieron la instalación y el análisis de resultados del Software Apache Hadoop. Este tipo de computadoras fueron armadas para realizar ciertas tareas que serán ejecutadas en un futuro en el laboratorio, Apache Hadoop fue el caso inicial para realizar las primeras pruebas de rendimiento.

Placa H97M-PLUS (Ver figura 10).

Es la placa base fundamental, cuenta con tres principales apartados que la hacen una gran computadora 1) el microprocesador (CPU Intel Core i5-3750K 3.4/3.8 GHz), 2) La memoria de acceso aleatorio (RAM 16GB (8GBx2)) y 3) las ranuras de expansión. Asus que es el principal fabricante de esta tarjeta buscó cambiar la apariencia usando una temática que combinaba el color negro el PCB y otros componentes como dorado para resaltar los disipadores (Asus, 2019)[16].

Figura 10. Tarjeta Madre Asus H97M-PLUS [16].

Especificaciones

En la tabla 3 se muestran las especificaciones y características de la tarjeta madre LUFAC utilizada en las computadoras dentro del laboratorio.

Especificaciones Generales	
Marca	Asus
Modelo	H97M-PLUS
Procesador	
Para Procesadores	Intel
CPU Socket	Socket 1150
- Chipset	
Modelo	Intel H97
- Memoria	
Ranuras de Memoria	4
Memoria Compatible	DDR3 1600/1333 MHz Non-ECC, Un-bufferedMemory
Máxima Memoria Soportada	32 GB
Arquitectura de Memoria	Dual Channel
- Red Integrada	
Chipset de Red	Intel I218V
Máxima Velocidad en Red	Gigabit LAN 10/100/1000 Mb/s
Puertos de Audio	6
- Especificaciones	
Factor de Forma	Micro-ATX

Tabla 3. Especificaciones Tarjeta Madre H97M-PLUS [16].

2.7.4 Clúster Xexelo 0

El clúster Xexelo 0 está compuesto por cuatro servidores que son administrados de manera remota. Los servidores son homogéneos, es decir, presentan las mismas características físicas. Dichos servidores fueron adquiridos gracias a los recursos otorgados por la SECITI. En la tabla 4 las características de los servidores.

Especificaciones

Marca	Supermicro
Modelo	X9DRD-IF/LD
Número de Procesadores	2
Tipo de procesador	Intel Xeon E5-2620a 2.1 GHZ
Hyper-threding	Si, 12 núcleos por procesador
Memoria RAM	32GB
Disco duro	1TB
NIC de procesamiento	10Gbps
NIC de monitoreo	1Gbps

Tabla 4. Especificaciones del clúster XEXELO 0

En una de las tarjetas del clúster Xexelo 0 se instaló el software Apache Hadoop con los requerimientos base que tiene instalado ya el clúster. Ya que Xexelo 0 se ocupa directamente para otras pruebas de investigación. Por consiguiente, se delimitó el uso para la instalación y evaluación del software, esto permitió determinar el rendimiento del software ya instalado.

Capítulo 3. Instalación de Apache Hadoop

En este capítulo se describe como se instaló y configuró el ambiente de trabajo de Apache-Hadoop, explicando cuáles son los requerimientos iniciales, la instalación básica y el funcionamiento del entorno de trabajo usando para verificar el funcionamiento del sistema.

En el desarrollo de este proyecto se realizaron diversas instalaciones de prueba, con diferentes versiones y configuraciones de Apache Hadoop, a continuación, se presenta el proceso que determinamos que es necesario para levantar el sistema en las cuatro tecnologías que se propusieron en el laboratorio. Este procedimiento se divide en cuatro secciones para configurar que son:

1. Actualización de Software.
2. Plataforma de computación JAVA para Apache Hadoop
3. Comunicación entre las entidades del clúster de Apache Hadoop
4. Instalación y configuración de los archivos del ambiente de trabajo de Apache Hadoop en modo clúster.

3.1 Actualización de software

El framework Apache Hadoop se instaló en computadoras y tarjetas de experimentación, utilizando como sistema operativo diversas distribuciones de Linux (CEntOS y Debian). Las actualizaciones de software del sistema operativo suelen ser sencillas en la forma de ejecutar, pero realmente una actualización puede beneficiar o afectar a nuestro sistema.

De manera particular para los equipos que utilizamos en este proyecto, una actualización en el sistema operativo en las tarjetas de desarrollo Raspberry Pi e Intel Galileo, generan un problema al momento de actualizar, se afectan los usuarios creados y sus permisos de ejecución (Ver figura 11).

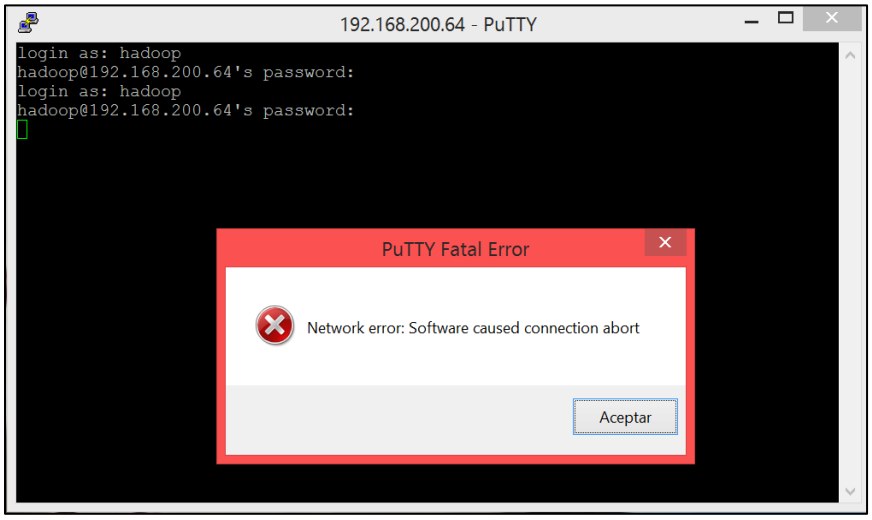


Figura 11. Error al querer ingresar a nuestro usuario después de un upgrade en Galileo y Raspberry.

A continuación, se detalla los aspectos fundamentales al mejorar y actualizar el sistema operativo:

Primero, el parámetro “upgrade”, moderniza cualquier versión y actualiza toda la lista de los índices de paquetes dentro del software ya instalado, es decir, hace una comparación entre la lista de paquetes actuales que ya tenemos en nuestro sistema, contra los repositorios más actualizados en la red. Con base en ello nos indica que paquetes (programas) se pueden mejorar con una nueva versión. En otras palabras, el sistema sabe qué hay de nuevo, que se puede modernizar o actualizar, por lo regular son archivos que se encuentran en el directorio /var/lib/apt/lists. (Capítulo 2. Gestión de paquetes Debian. (2020), 2019) [17]

Segundo, en cuanto al parámetro “update”, mejora las versiones de todos los paquetes que hayan compilado previamente, haciendo el mismo procedimiento que upgrade, busca las nuevas actualizaciones de los paquetes (Ver figura 12).

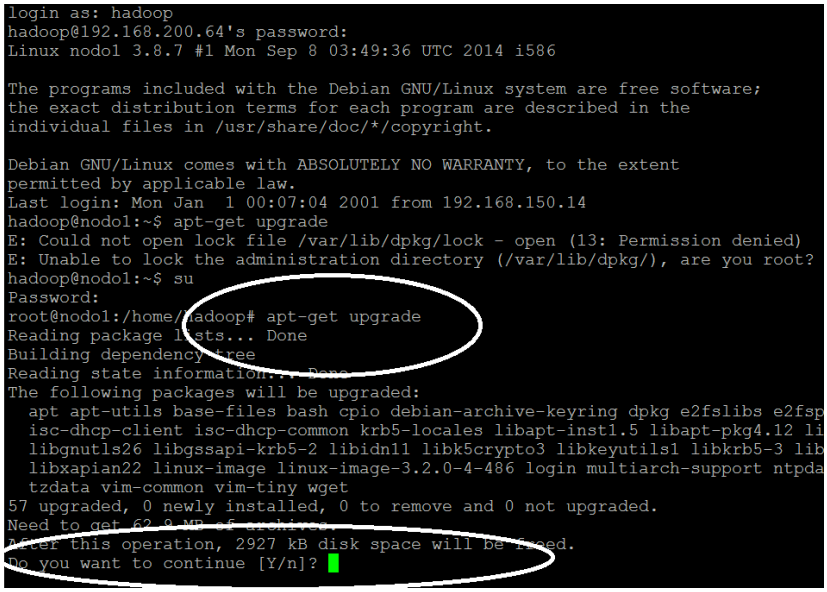


Figura 12. Representación de un update en cualquiera tecnología buscando versiones más recientes de las bibliotecas.

3.2 Plataforma de computación JAVA para Apache Hadoop

El framework Apache Hadoop está escrito en el lenguaje de programación Java, por lo tanto, puede ejecutarse en cualquier plataforma con una JVM previamente instalada (White, 2007) [10]. De manera general, es necesario verificar la instalación correcta de java para que tenga compatibilidad con Apache Hadoop y sus respectivas versiones. En el recuadro verde de la figura 13 muestran los archivos binarios que podemos descargar directamente de los servidores de Oracle, para instalar en arquitecturas de procesamiento de 32 y 64 bits para sistemas Linux

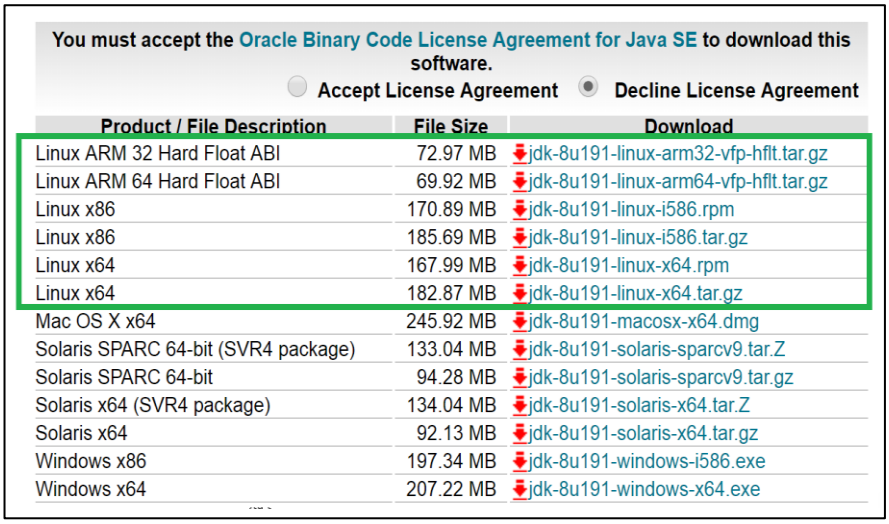


Figura 13. Versiones actuales 2018 de Java.

En primer lugar, hay que tener en claro la compatibilidad de las dos herramientas, en este caso Apache Hadoop y Java, motivo que permite un emparejamiento correcto para determinar el mejor arranque en la instalación. En la página oficial de Apache Hadoop se encuentran las versiones recientes, indicadas en la figura 14 con la flecha azul y de la misma forma con una flecha rosa se marca el binario de descarga. Ciertas especificaciones de IBM, así como, de la página oficial de Apache Hadoop mencionan que todos los JAR de Apache Hadoop están compilados en una versión de ejecución de Java 8 y por esto se sugiere que se actualice a esta versión.

En un caso particular, en los sistemas operativos de las tarjetas de desarrollo Raspberry Pi e Intel Galileo, Java está previamente instalado, por ello, es necesario tomar en cuenta que versión de Apache Hadoop será instalada en estos sistemas para una óptima compatibilidad.

Es indispensable que, en cada sistema de cómputo este instalado el JRE, en el sistema operativo puede estar instalada una versión previa de java y en caso contrario tendríamos que instalar el JRE desde la página oficial, la cual cuenta con una guía de instalación del software necesario para nuestros equipos.

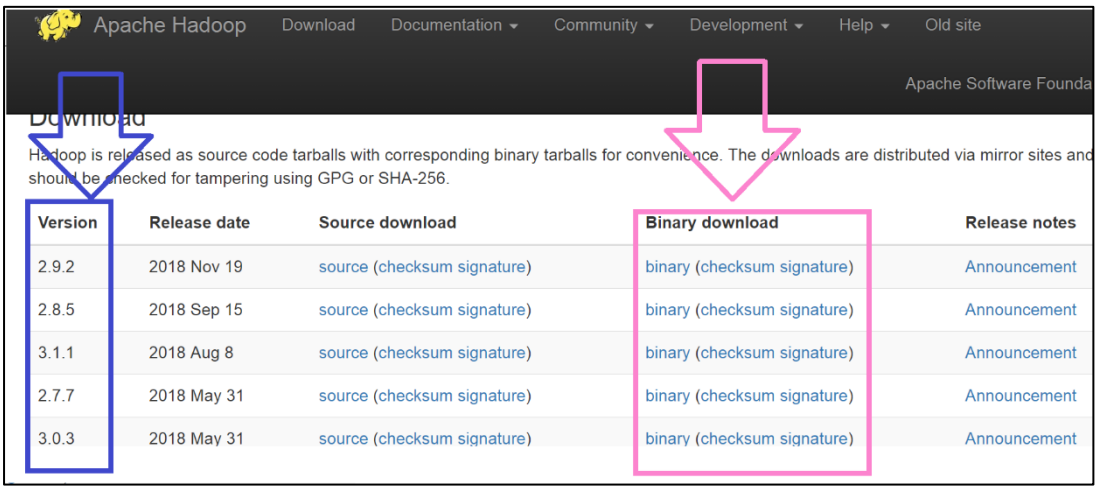


Figura 14. Versiones actuales 2018 de Hadoop

Para configurar el entorno general de arranque de Apache Hadoop se modifican las variables de entorno.

La variable $JAVA_HOME contiene la ruta en donde se instala el software de Java, y la podemos editar con ayuda del manual ubicado en el anexo 1. Dicha variable de entorno, que es necesaria para el arranque, se modifica para determinar la ruta inicial de Apache Hadoop con Java.

Cuando no se cuenta con una correcta compatibilidad de Apache Hadoop con el software Java, los mensajes al momento de arrancar el sistema serán relacionados con la instalación de Java, en general podemos determinar, que sí, estas dos herramientas no están en sincronía, Apache Hadoop no funcionará. Sin embargo, si se iniciará el sistema sin tomar en cuenta la compatibilidad, más adelante se marcarían los errores con java al compilar Apache Hadoop como se marcan en el círculo rojo (Ver Figura 15).

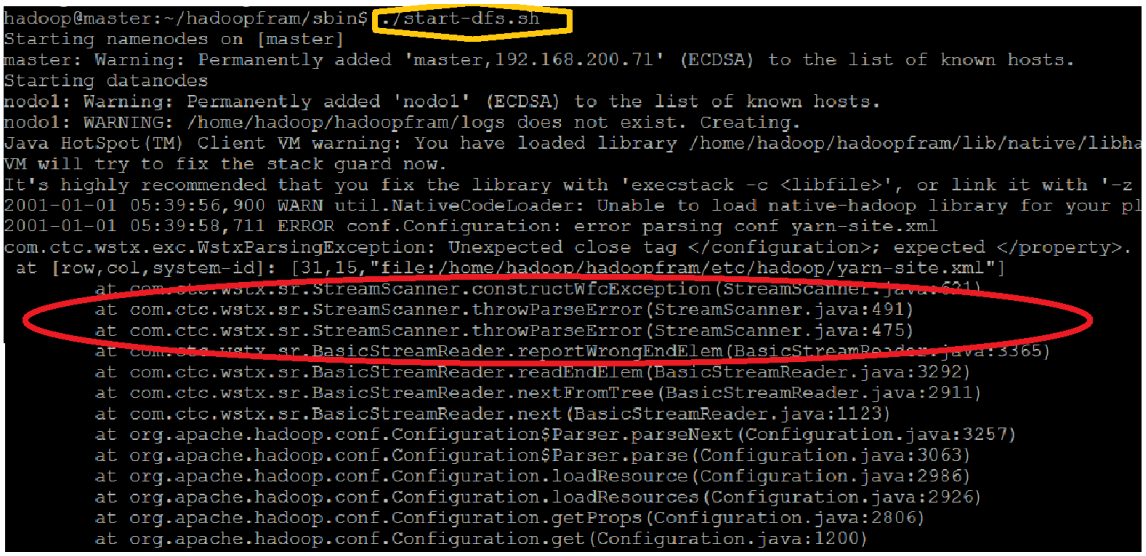


Figura 15. Muestra algunos errores al levantar Apache Hadoop en Java.

3.3 Comunicación entre las entidades del clúster de Apache Hadoop

El entorno de trabajo de Apache Hadoop puede ser instalado en un clúster de cómputo y es necesario que la comunicación entre las entidades sea efectiva y con seguridad, para esto se utilizan los servicios del programa ssh.

El protocolo SSH usa criptografía de llave pública para autenticar a los hosts y a los usuarios. Las llaves de autentificación, llamada llaves de autenticación para SSH, se crean con el programa SSH-KEYGEN, dichas llaves se utilizan para automatizar inicios de sesión y para autentificar host.

La arquitectura de comunicación se denomina maestro-esclavo y la comunicación en este sistema es de suma importancia para determinar un clúster comunicado, la configuración del sistema SSH, define un archivo llave o también conocido como llave autorizada la cual se muestra en el recuadro amarillo de la figura 16. Es decir que esta llave creada en el maestro permite que todas las máquinas esclavo del clúster estén comunicadas haciendo una conexión directa y rápida. Así pues, esta llave se copia a cada uno de los esclavos, permitiendo que el maestro pueda iniciar una comunicación utilizando el protocolo ssh para cada esclavo del clúster.

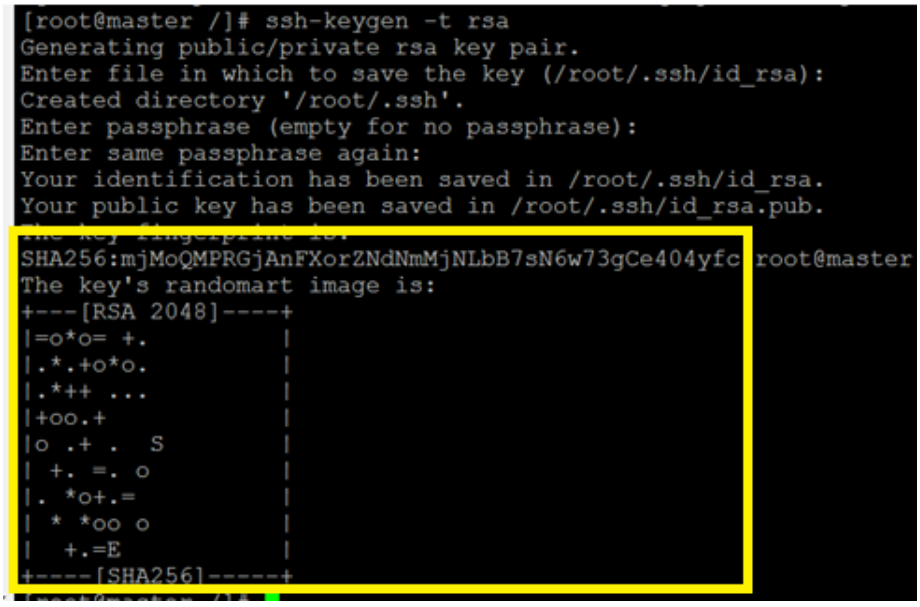


Figura 16. Creación de la llave en el maestro.

Aunque la seguridad o servicios de seguridad no es parte de este proyecto, es una problemática cuando hablamos de llaves. Entre las tareas y problemas que se agregan aparecen, justamente, los de seguridad, de interconexión física y remota con el clúster.

1. Instalación y configuración de los archivos del ambiente de trabajo de Apache Hadoop en modo clúster

El proyecto Apache Hadoop como se mencionó en el anterior capítulo, de manera básica se compone de tres módulos:

1. Hadoop Common. Utilidades comunes para soporte de otros módulos de Hadoop.
2. Hadoop Distributed File System HDFS. Sistema de archivos distribuido que provee acceso de alto rendimiento a los datos de aplicación.
3. Hadoop MapReduce. Sistema basado en YARN, para el procesamiento paralelo para grandes grupos de datos.

Apache Hadoop tiene su propio sitio web de donde se descarga directamente el paquete de instalación, este paquete contiene un grupo de programas que se deben instalar y configurar para que se ejecuten con el sistema.

- Arquitectura del sistema de archivos de Hadoop HDFS

El sistema de archivos de Hadoop está preparado para trabajar con archivos de gran tamaño, almacenados en diferentes nodos a través de una red de datos administrada de manera distribuida y utiliza la filosofía de trabajo por bloque, el tamaño de los bloques que se transportan en el clúster puede ser configurado y va de entre 64MB y hasta 256MB, siendo el valor por defecto del bloque de 128MB (White, 2007) [10].

Los nodos que integran el clúster ejecutan agentes dependiendo el trabajo que realicen, por ejemplo, el nodo maestro se denomina NameNode, es el nodo principal y es el encargado de la gestión de los directorios y de los metadatos, administra los bloques que se envían a los nodos de datos verificando que todos los trabajos concluyan correctamente.

Los nodos llamados DataNode, son los nodos de almacenamiento, y están destinados para el proceso de lectura y escritura de los bloques y de informar su estado al nodo maestro NameNode.

Existe un agente de ejecución llamado SecundaryNameNode que puede ser un nodo, que puede usarse como soporte del NameNodo, ya que almacena una imagen del sistema de archivo y una réplica de los metadatos.

Existen más agentes en el sistema como el TaskTracker que se encarga de ejecutar una tarea de MapReduce en cada nodo y también el agente JobTracker que gestiona las tareas de MapReduce.

Antes de iniciar la configuración de Apache Hadoop necesitaremos configurar 4 puntos para el correcto funcionamiento del framework los cuales se indican a continuación:

1. El entorno para la ejecución de los servicios de Apache Hadoop.
2. Los parámetros de configuración de los servicios de Apache Hadoop.
3. Los mensajes de información.
4. El archivo de esclavos con el listado de los nodos de trabajo del clúster.

Realizado los anteriores puntos, para la instalación de Apache Hadoop iniciamos con la descarga del paquete de archivos binarios y la expansión de los archivos en la carpeta indicada por defecto, dándole los permisos indicados para el uso del framework.

Para configurar el clúster Apache Hadoop plantearemos una serie de pasos extraídos de los manuales de instalación, las configuraciones son por defecto, haciendo solo los cambios necesarios en los archivos de configuración framework.

Para configurar el sistema se debe hacer uso de archivos que le indican cuales variables se modifican en particular para cada sistema. En la administración básica de

los agentes de Apache Hadoop se deben de usar los scripts yarn-env.sh, hadoop-env.sh y mapred-env.sh y con la modificación del contenido de los archivos se configura el entorno de los procesos de los servicios de Apache Hadoop. Todos los archivos se encuentran en la ruta: ~/hadoop/opt/hadoop/etc/hadoop. A continuación, se indica cuáles son los cambios básicos que se realizó en cada uno de los archivos.

3.4.1 Archivo Hadoop-env.sh

En el archivo se debe indicar la ruta para usar java, en nuestro sistema la configuración que determinamos es básica, indicada por la flecha azul (Ver figura17). Por esta razón, la configuración inicial de arranque con el JDK en Apache Hadoop se determina con el directorio donde se encuentran todas las herramientas de Java, en el anexo 1 viene de manera clara que es lo que se modifica dentro de este script.

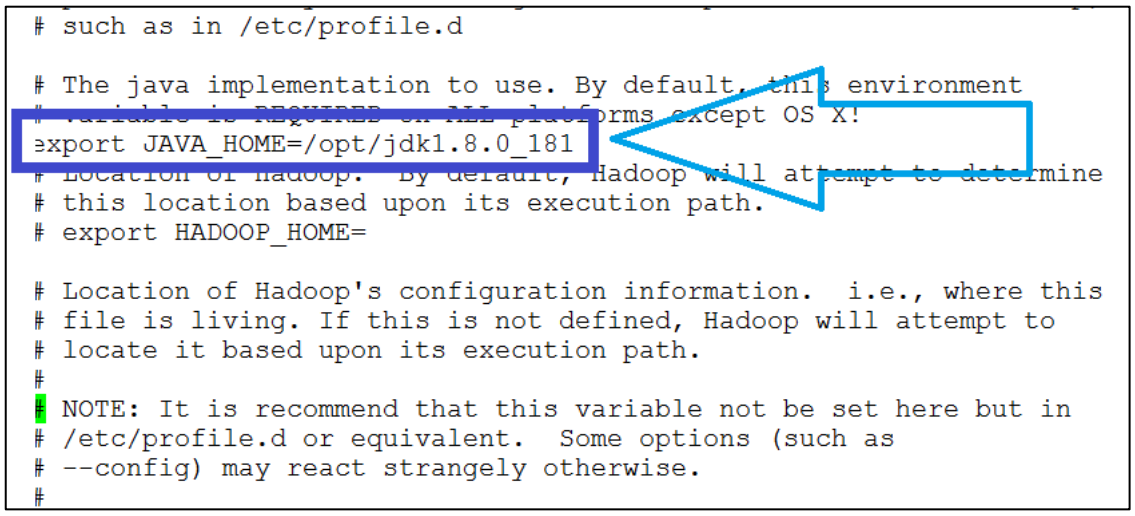


Figura 17. Configuración básica del Hadoop-env.sh.

3.4.2 Archivo Mapred-env.sh

Las variables de entorno de Map Reduce también son configurables utilizando un script y se configuran dependiendo las necesidades de la máquina virtual de Java. Además de las características básicas, en nuestro clúster, también se modifica el archivo mapred-site.xml, y únicamente se le añade las siguientes líneas:

```
<configuration>
        <property>
                <name>mapreduce.framework.name</name>
                <value>yarn</value>
                <description>Executionframework</description>
        </property>
</configuration>
```

Tabla 5. Líneas básicas de configuración.

3.4.3 Core-site.xml

En el archivo core-site.xml ajustamos la configuración del core de Hadoop como son ajustes de Entrada/Salida que son comunes a HDFS Y MapReduce. Al fichero *core-site.xml* se le agregan las siguientes líneas de la tabla 6:

CORE
<configuration>
<property>
<name>fs.defaultFS</name>
<value>hdfs://master:9000</value>
<description>NameNodeURI</description>
</property>
</configuration>

Tabla 6. Configuración de la biblioteca core.

3.4.4 HDFS-SITE.XML

En este archivo se ajustan las propiedades para los servicios *HDFS*: *NameNode*, secondary*NameNode*, y *DataNodes*. Al fichero *hdfs-site.xml* se le agregan las siguientes líneas de la tabla 7:

HDFS
<configuration>
<property>
<name>dfs.replication</name>
<value>3</value>
<description>Default block replication</description>
</property>
<property>
<name>dfs.namenode.name.dir</name>
<value>file:/home/hadoop/hdfs/namenode</value>
</property>
</configuration>

Tabla 7. Configuración de la biblioteca core.

3.4.5 Archivo Yarn-Site.XML

Este archivo contiene información relacionada con los servicios y propiedades de *Yarn*: Resource Manager, Node Manager and History Server.

En las implementaciones que se realizaron en este trabajo, al fichero *yarn-site.xml,* no se le añadieron líneas por la configuración básica de instalación.

En el siguiente capítulo se presenta el resultado de la instalación de Apache Hadoop, una vez configuradas las variables de entorno básicas, en las cuatro tecnologías propuestas, así como en sus respectivos sistemas operativos.

Capítulo 4. Análisis de resultados

En el presente capítulo se mostrarán los resultados obtenidos de las pruebas de instalación realizadas en los diferentes clústeres denominados Xexelo-0, Xexelo-LF, Xexelo-RP y Xexelo-GL en el que se ejecutaron las mismas configuraciones de arranque, pero que en algunas de estas tecnologías no funcionaron.

4.1 Resultados de instalación de las cuatro arquitecturas

Con la configuración del software lista y mostrada en el anexo 1, se iniciaron los demonios de Apache Hadoop y el servidor Apache (httpd) (Ver figura 18 y 19). Estos servicios activos muestran que se realizó una correcta instalación y configuración del sistema, de la misma manera se muestra la hora y fecha del día que se levantaron los servicios indicados con la flecha morada. El sistema registra el nombre del nodo maestro y su dirección IP la cual es igual para todos los clústeres que es asignada por el usuario al momento de configurarse el clúster.

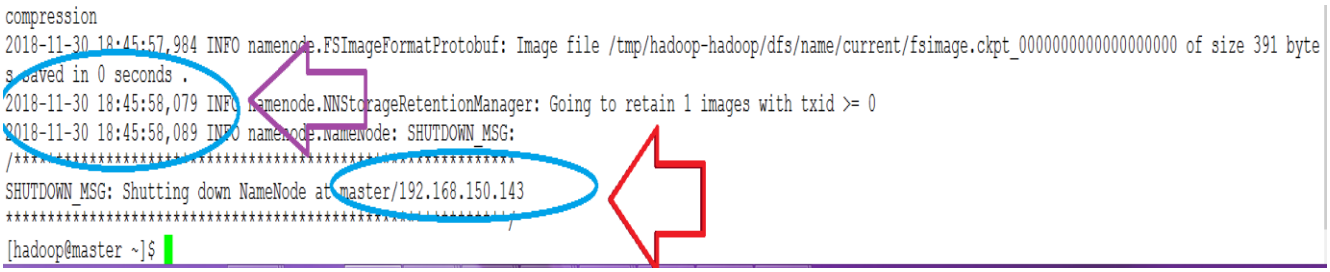


Figura 18. Arranque de sistema de Apache Hadoop en arquitectura de 64 bits.

```
[hadoop@master ~]$ systemctl status httpd
● httpd.service - The Apache HTTP Server
   Loaded: loaded (/usr/lib/systemd/system/httpd.service; disabled; vendor preset: disabled)
   Active: active (running) since vie 2018-11-30 19:00:06 CST; 12s ago
     Docs: man:httpd(8)
           man:apachectl(8)
 Main PID: 1316 (httpd)
   Status: "Total requests: 0; Current requests/sec: 0; Current traffic:   0 B/sec"
   CGroup: /system.slice/httpd.service
           ├─1316 /usr/sbin/httpd -DFOREGROUND
           ├─1317 /usr/sbin/httpd -DFOREGROUND
           ├─1318 /usr/sbin/httpd -DFOREGROUND
           ├─1319 /usr/sbin/httpd -DFOREGROUND
           ├─1320 /usr/sbin/httpd -DFOREGROUND
           └─1321 /usr/sbin/httpd -DFOREGROUND
[hadoop@master ~]$
```

Figura 19. Activo el servicio de httpd.

Una vez que los servicios están activos podemos ingresar a nuestro navegador (Ver figura 20), usando la dirección IP asignada al maestro, podemos ingresar a la página inicial de Apache Hadoop, visualizando que el marco de trabajo está activo y funcionando en el hardware, esto se realiza con cada tecnología.

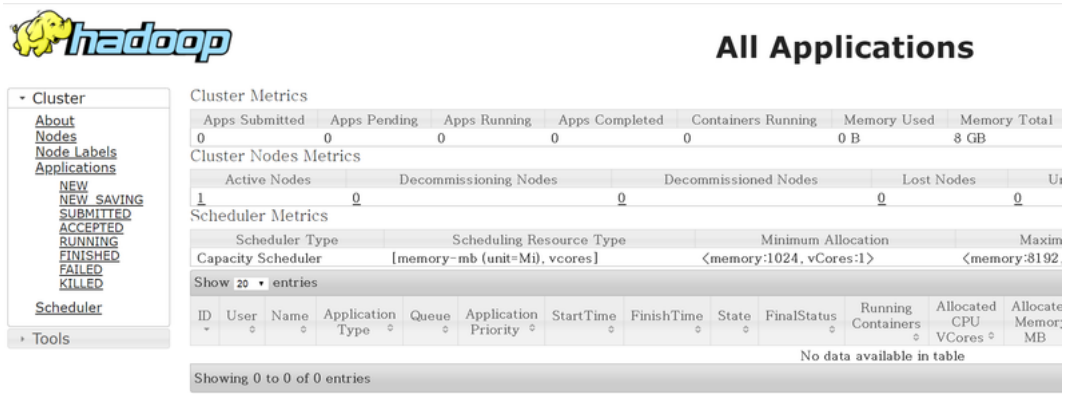


Figura 20. Página principal de aplicaciones y servicios.

En la página principal de Apache Hadoop se muestra la versión del sistema previamente instalado, así como, las pestañas de los DataNodes marcados con la flecha negra figura 18, donde se determina el nombre de cada nodo y su dirección IP. En la misma página se muestra el volumen de cada uno de los nodos indicados con la flecha naranja. Con volúmenes nos referimos a la capacidad de memoria y la cantidad de nodos activos en el clúster (Ver figura 21).

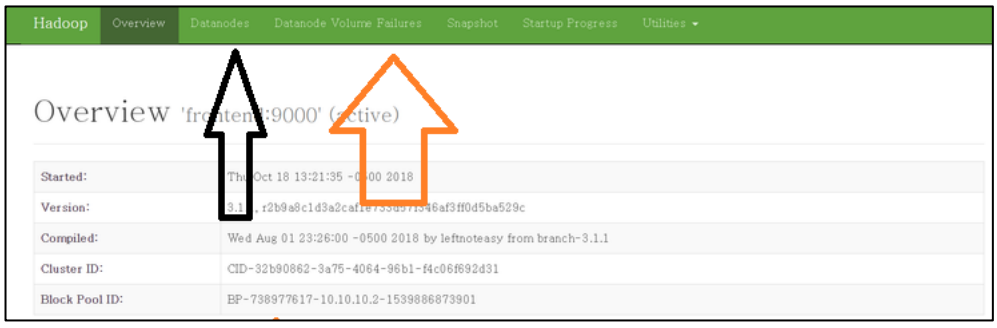


Figura 21. Página principal de todas las aplicaciones de Apache Hadoop.

4.1.1 Raspberry Pi

Para la tarjeta Raspberry se levantó el sistema Hadoop, esta tecnología solo cuenta con un procesador que se indica en la flecha amarilla (Ver figura 22), así que, la mayor parte del sistema está trabajando con este solo núcleo. La memoria RAM de la tarjeta Raspberry PI, flecha roja figura 22, se encarga de trabajar y atender solicitudes junto con el procesador de manera controlada.

Un apoyo fundamental, en caso de que el procesador llegue al 100% de su capacidad, es la memoria Swap flecha rosa figura 22, la cual permite dividir el trabajo del CPU descargando datos de la memoria RAM para que no se sobre cargue de trabajo. No es una mala arquitectura para un clúster, pero la demanda de trabajo solo solicitando el servidor web de Hadoop satura a la tarjeta.

Hadoop se ejecuta dentro de su compilador que es java, en todo momento Hadoop requiere de java desde el momento de la ejecución, el requerimiento se ve reflejado de derecha a izquierda del recuadro morado al naranja como lo marca la flecha (Ver Figura 23) esto permite observar de manera clara la ejecución Hadoop. Del mismo modo que se ejecuta Hadoop se abren las pestañas de algún buscador para visualizarlo en el web browser.

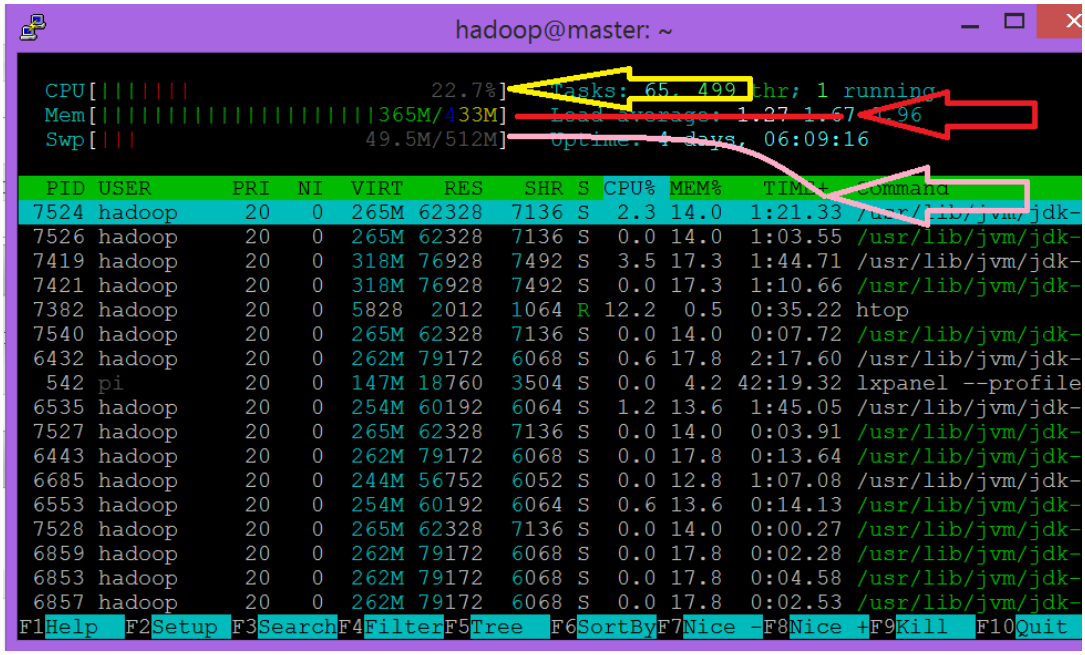


Figura 22. Rendimiento de la memoria RAM y Swap en Raspberry.

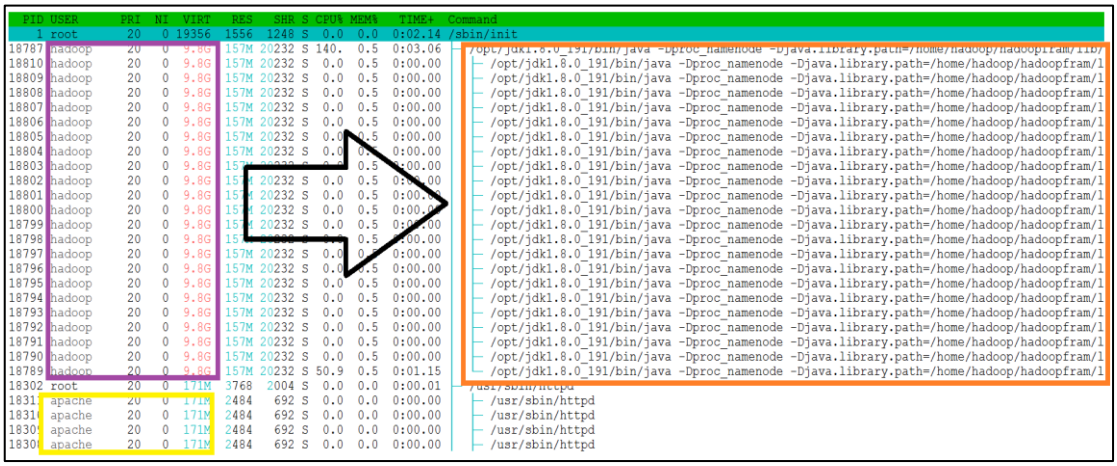


Figura 23. Solicitud de servicio de Apache Hadoop en Raspberry.

Hadoop cuenta dentro de sus especificaciones con un resumen general del clúster el cual genera una consistencia clara de la definición de cada uno de sus nodos, teniendo a la mano los nodos activos o muertos con los que cuente, así como la cantidad de memoria total del clúster para almacenamiento, igualmente cuenta con una suma total de memoria para la instalación de cada nodo. Toda esta información se muestra en el recuadro verde (Ver Figura 24).

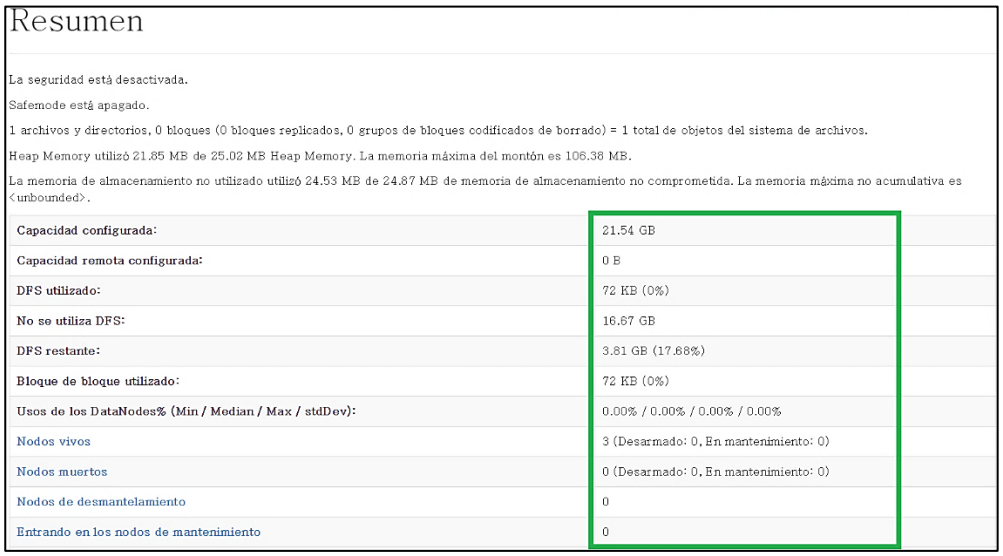


Figura 24. Resumen general del clúster instalado.

4.1.2 Galileo

La tarjeta Intel Galileo Gen 2 cuenta con una memoria RAM de 256 MB y no tiene una memoria swap que ayude a la memoria RAM, por lo cual Intel Galileo con el hecho de levantar los demonios de Hadoop y httpd queda al 100 por ciento de sus capacidades sin tener otro espacio para alguna solicitud más como se muestra en la flecha negra (Ver figura 25-26).

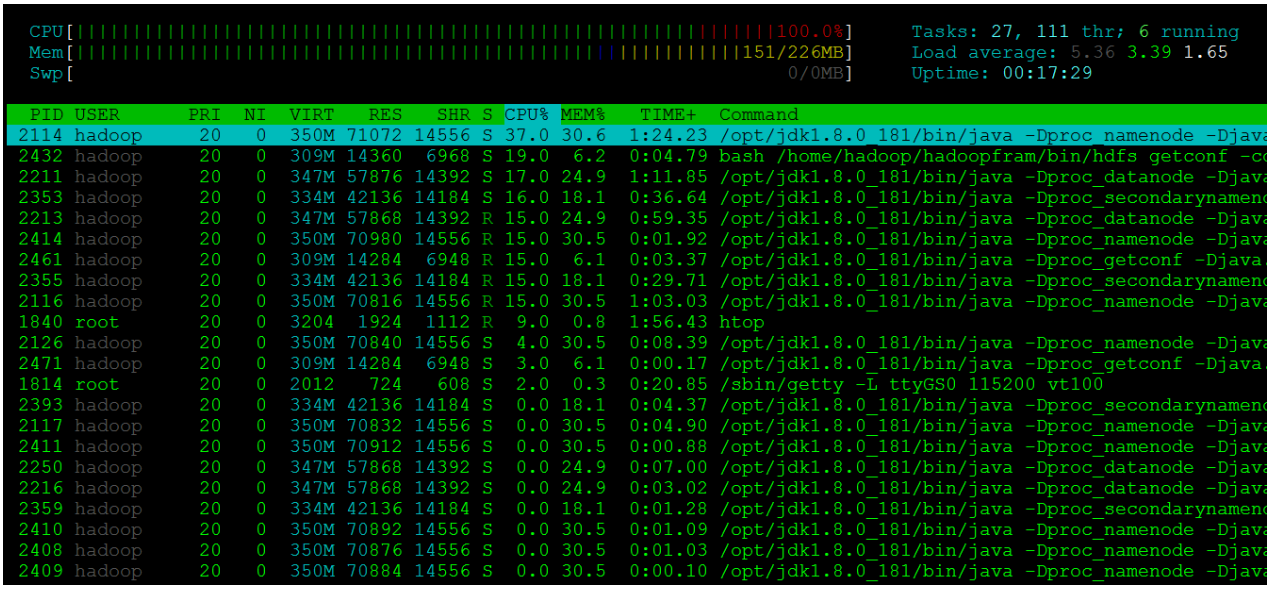


Figura 25. Rendimiento de la tarjeta Galileo Gen 2.

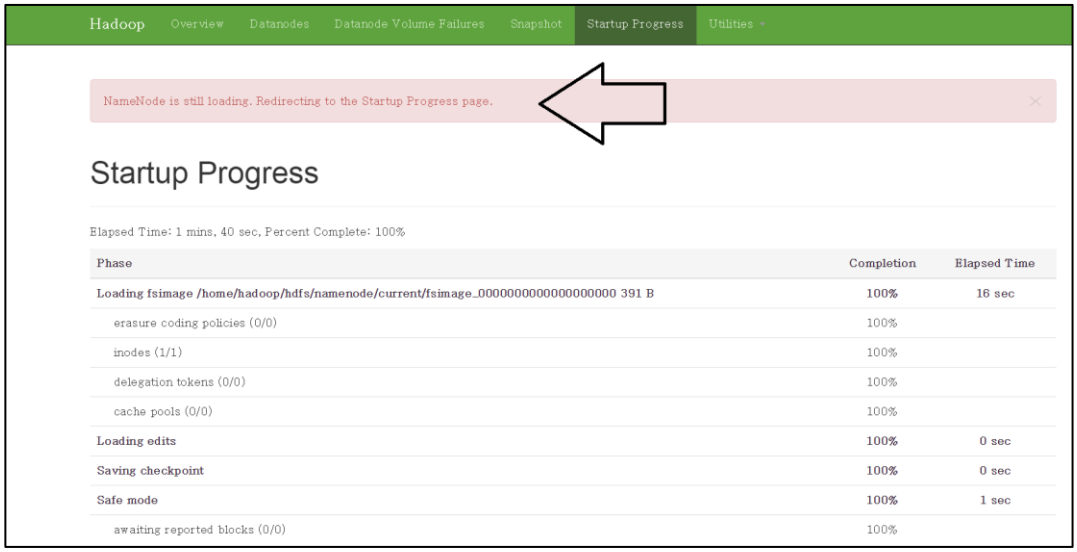


Figura 26. Solo levanta el servicio sin responder a otra solicitud.

4.1.3 Xexelo 0 y Lufac (LF)

Para estos dos clústeres la instalación y configuración del software Apache Hadoop fue exitosa, en los aspectos de actualización y compatibilidad del Apache Hadoop y Java no se tuvo ningún problema, así como, los cambios de variables que el software Apache Hadoop necesita para una configuración básica se realizaron de manera rápida y sencilla.

Al iniciar el clúster se ve reflejado los cambios de forma exitosa, es decir la solicitud del servidor web Apache Hadoop arranca, ya que este servidor en estos 2 clústeres no genera una mayor demanda de procesamiento, por lo tanto, estos dos clústeres un conformado por computadoras (LUFAC) y el otro que es de uso específico XEXELO 0 son esenciales para el trabajo con Apache Hadoop.

Dentro del rendimiento de una activación básica se ve reflejado el uso del clúster y la iteración de sus procesadores. En la figura 27 para un nodo en el clúster XEXELO 0 los 24 procesadores que este tiene se muestran activos, solo levantando el servicio por medio de la página web. En los recuadros vinos se puede percibir el porcentaje de trabajo solo en el proceso de levantar el sistema.

En la figura 28 se ve la activación del servicio en el nodo| de LF y nombre del nodo que se le asigna. En la figura 29 vemos el tamaño del nodo y su nombre un resumen general. Pará estas dos tecnologías la consulta de su servicio al servidor de la página web de Apache Hadoop no demanda servicio de memoria lo que da pie a que estas dos tecnologías son aptas para seguir trabajando con el software Apache Hadoop, lo que tenemos que recordar es que la tecnología LF son computadoras de escritorio dentro del laboratorio.

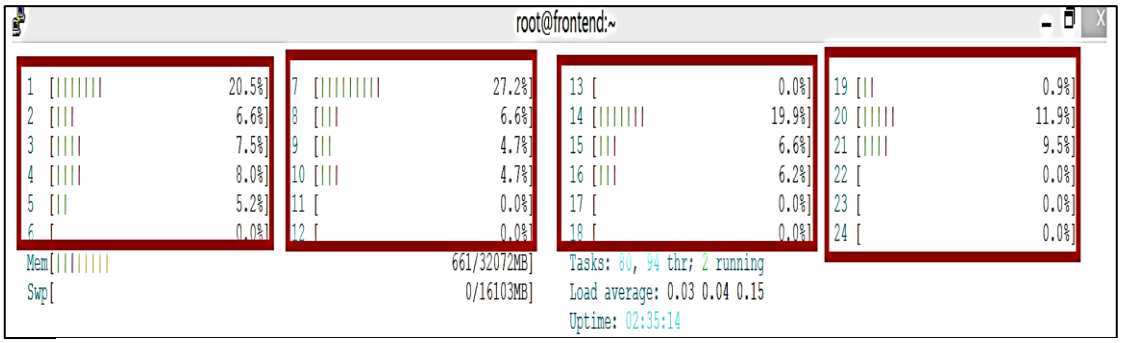


Figura 27. División de tareas en los 24 procesadores en un nodo del clúster Xexelo 0.

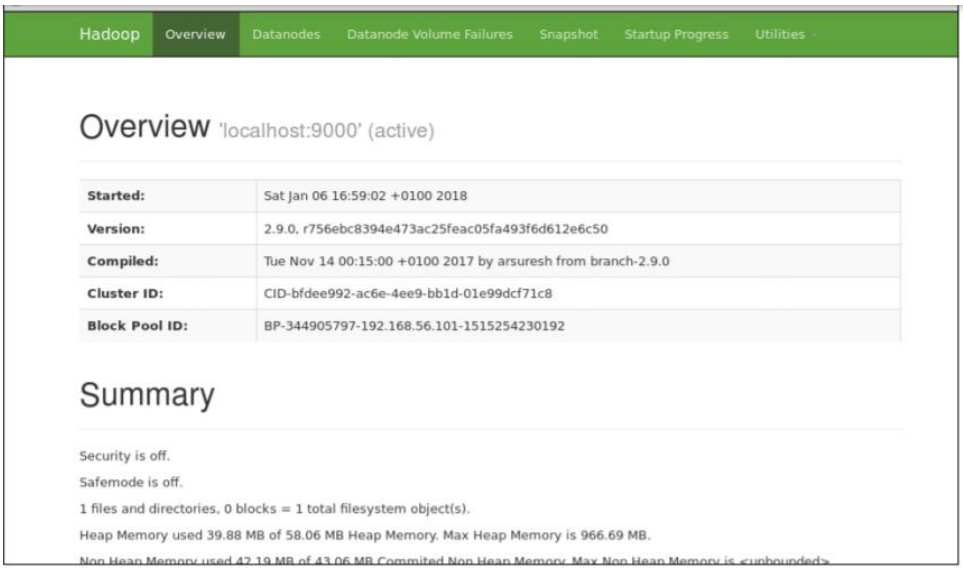


Figura 28. División de tareas en los 24 procesadores en un nodo del clúster Xexelo 0.

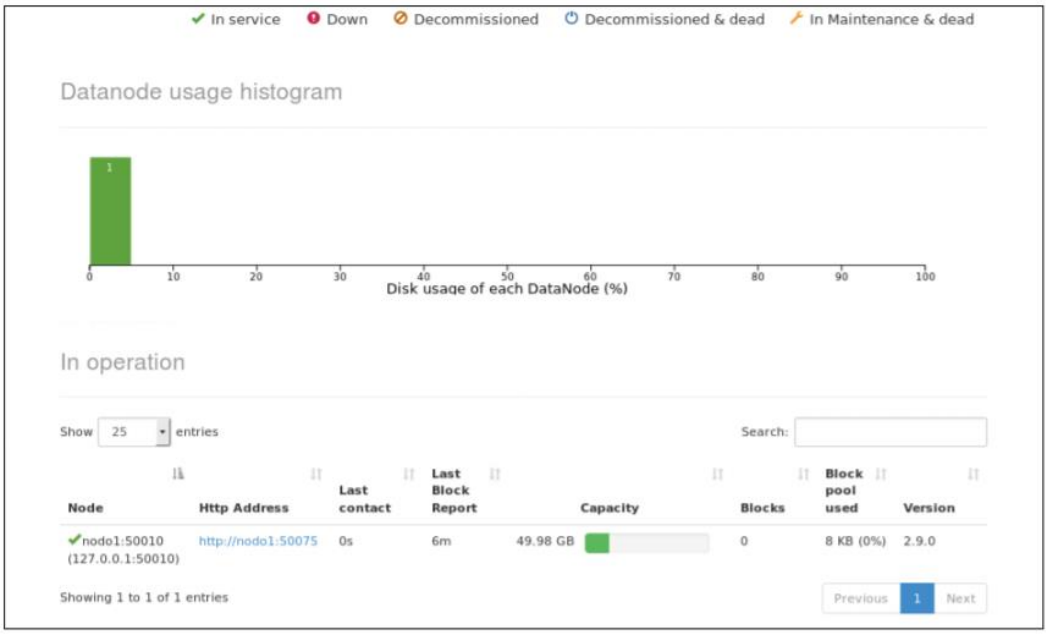


Figura 29. Capacidad del nodo LF.

Es importante tener en cuenta que el clúster XEXELO 0 y LF son máquinas con alta demanda en recursos físicos por lo que da pie a seguir trabajando con estos clústeres. La información del nodo es esencial para especificar las características del clúster y el monitoreo que se realice cuando esté en una actividad de alta demanda.

Conclusión.

En este proyecto se implementaron varios clústeres en diferentes tecnologías de desarrollo, motivo que permitió definir cuáles eran las adecuadas para montar el servicio de Apache Hadoop. Las tecnologías del laboratorio se encuentran en buen estado, y se pudieron hacer las pruebas pertinentes en las tecnologías ya mencionadas.

Las nuevas infraestructuras de clústeres las cuales denominamos como Xexelo-*LF* (computadoras Lufac), Xexelo-GL (Tarjetas Galileo), Xexelo-*RP* (tarjetas Raspberry) y se ocupó el anterior ya creado *Xexelo-0,* que ahora forman parte del proyecto del Laboratorio de sistemas distribuidos y redes de alto desempeño dentro del laboratorio de redes de computadoras, ubicado en el laboratorio B-404 del plantel San Lorenzo Tezonco de la UACM.

Respecto al Hardware, se implementaron arquitecturas de 32 bits para los clústeres Xexelo-GL y Xexelo-RP, la instalación se logró hacer en estos sistemas con ciertas especificaciones las cuales se detallan dentro del manual que se creó.

Para el Sistema Operativo CEntOS instalado en los clústeres Xexelo-LF y Xexelo 0, así como en Xexelo-GL y Xexelo-RP la distribución instalada es Debian. Estos nos permitieron un funcionamiento estable, asimismo la compatibilidad con Apache Hadoop es adecuada para CEntOS y Debian según la tecnología que se ocupó. Es importante destacar que dependiendo la tecnología del clúster es el sistema operativo que le corresponde.

Dado lo anterior, se obtienen las siguientes conclusiones:

- Las imágenes obtenidas de los procesadores durante el arranque en las diferentes tecnologías, denotaba que al momento de iniciar el sistema Apache Hadoop toda la cantidad de memoria RAM se utilizaba, por lo que retrasaba los requerimientos de otros servicios, de ahí que en los clústeres RP (Raspberry) y GL (Galileo) no funciono.
- Dependiendo de la tecnología que se ocupe, se podrá explotar Apache Hadoop a conveniencia de la investigación o resultados que queramos determinar.

- Con las tecnologías ya utilizadas, podemos destacar que los clústeres de Xexelo-LF y Xexelo-0 funcionan adecuadamente para implementar Apache Hadoop son tecnologías aptas para seguir trabajando después de la instalación.

Con estas conclusiones se da por finalizado este proyecto, en el que se diseñó una guía de instalación y se determinó que tecnologías son aptas para instalar un clúster Apache Hadoop, motivo que permite definir cuáles son las tecnologías que se podrán ocupar para trabajos de almacenamiento o procesamiento a futuro.

A. Manual de instalación

Instalación de java y hadoop

- Paso 1

Verificar la seguridad de Linux avanzada y desactivar

Para desactivar la seguridad de Linux es necesario entrar al siguiente directorio

#cd /etc/selinux

En este directorio existen varios archivos, el que se modificara es “config” mediante las siguientes líneas de comando, esto se ejecuta tanto en el nodo master como en los esclavos

#nano config

Después de ejecutar el comando anterior se desprenderá una ventana se cambiará una línea que dice active a disabled

(disabled)

El firewall se desactiva en todos los nodos y el maestro

#systemctl stop firewalld

Se instala Java en el master y nodos esclavos del clúster como root

- JAVA

#yum install java
• Jdk
#cd /opt/
#wget --no-cookies --no-check-certificate --header "Cookie: gpw_e24=http%3A%2F%2Fwww.oracle.com%2F; oraclelicense=acceptsecurebackup-cookie" "http://download.oracle.com/otn-pub/java/jdk/8u131- b11/d54c1d3a095b4ff2b6607d096fa80163/jdk-8u131-linux-i586.tar.gz"

Para instalar java sólo se ejecutará el comando de instalar según la tecnología(Nota: el comando mostrado a continuación es de CentOS depende las distribución para lanzar el comando)

- Paso 2

Comprobar la versión de Java instalada Los binarios Java están disponibles en la variable de entorno PATH. Puede usarlos desde cualquier lugar de su sistema. Comprobemos la versión instalada de Java runtime environment (JRE) en su sistema ejecutando el siguiente comando

#java -version

- Paso 3

Configurar las variables del entorno Java. La mayoría de las aplicaciones basadas en Java usan variables de entorno para funcionar. Establezca las variables de entorno Java usando los siguientes comandos.

Las siguientes líneas se anteponen en el root en la siguiente ruta

#cd /root

#nano .bashrc

Copiamos las siguientes líneas

En la línea de jdkX.X.X_XX podemos sustituirla por el nombre del directorio de java

#export JAVA_HOME=/opt/jdk1.8.0_181
#export JRE_HOME=/opt/jdk1.8.0_181/jre
#export PATH=$PATH:/opt/jdk1.8.0_181/bin:/opt/jdk1.8.0_181/jre/bin

Nueva

export JAVA_HOME=/usr/lib/jvm/java-7-openjdk-i386
export JRE_HOME=/usr/lib/jvm/java-7-openjdk-i386/jre
export PATH=$PATH::/usr/lib/jvm/java-7-openjdk-i386/bin:/usr/lib/jvm/java-7-openjdki386/jre/bin

Creamos usuario en master y slaves

Creamos el usuario Hadoop en todas las maquinas

#useradd /home/hadoop
#passwd hadoop
#reboot

Accederemos al usuario hadoop en el master

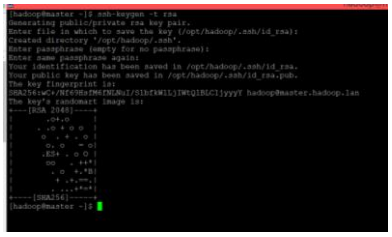
#su hadoop

Se crea el directorio (ssh) solo en el maestro

- Paso 4

ESTO SÓLO SE EJECUTA EN EL MASTER

ssh-keygen -t rsa



```
$ ssh-keygen -t rsa -P '' -f ~/.ssh/id_rsa

  $ cat ~/.ssh/id_rsa.pub >> ~/.ssh/authorized_keys

  $ chmod 0600 ~/.ssh/authorized_keys
```

Copia de llave a los nodos

- Paso 5

#ssh-copy-id localhost

#ssh-copy-id hadoop@192.168.150.X

Para verificar que la llave se completó verificamos en el esclavo

#ssh localhost

#ssh hadoop@192.168.150.X

La instalación de Hadoop se realiza en el directorio del usuario Hadoop en la siguiente ruta:

#home/hadoop

Descargamos hadoop

#wget http://www-eu.apache.org/dist/hadoop/common/hadoop-3.0.3/hadoop-3.0.3.tar.gz

Descomprimir

#tar xzfv hadoop-3.0.3.tar.gz

SE PUEDE O NO REALIZAR

Renombramos

#mv hadoop-2.X.X hadoopfram esto depende del administrador

Permisos de lectura

chown -R hadoop:hadoop /home/hadoop/hadoopfram/

Configuración de Hadoop

Editamos el siguiente archivo

Puede establecer las variables de entorno Hadoop anexar los siguientes comandos para ~/.bashrc

Nota:

~/.bashrc es el lugar para poner cosas que sólo se aplica a bash en sí, como alias y definiciones de función, shell opciones, y del sistema de configuración. (También se puede poner clave de los enlaces que hay, pero para bash que normalmente van en ~/.inputrc.) ~/.bash_profile puede ser utilizado en lugar de ~/.profile, pero es leído por bash, no por cualquier otro shell. (Esto es más preocupante si desea que los archivos de inicialización para trabajar en múltiples máquinas y su shell de inicio de sesión no es bash en todos ellos.) Este es un lugar lógico para incluir ~/.bashrc si es el shell interactivo. Recomiendo el siguiente contenido en ~/.bash_profile:

- Paso 6

Configurar las variables del entorno Java La mayoría de las aplicaciones basadas en Java usan variables de entorno para funcionar. Establezca las variables de entorno Java usando los siguientes comandos

Las siguientes líneas se anteponen en el root en la siguiente ruta

#cd /root

#nano .bashrc

Copiamos las siguientes líneas

En la línea de jdk1.8.0_171 podemos sustituirla por el nombre del directorio de java

#export JAVA_HOME=/opt/jdk1.8.0_131
#export JRE_HOME=/opt/jdk1.8.0_131 /jre
#export PATH=$PATH:/opt/jdk/bin:/opt/jdk1.8.0_131 /jre/bin

En la ruta #home/hadoop/

#nano .bashrc

Y agregamos las siguientes líneas

```
export JAVA_HOME=/opt/jdk1.8.0_131 /bin/java
export PATH=$PATH:$JAVA_HOME/bin

export CLASSPATH=.:$JAVA_HOME/jre/lib:$JAVA_HOME/lib:$JAVA_HOME/lib/tools.jar

export HADOOP_HOME=/home/hadoop/hadoopfram
export HADOOP_INSTALL=$HADOOP_HOME
export HADOOP_MAPRED_HOME=$HADOOP_HOME
export HADOOP_COMMON_HOME=$HADOOP_HOME
export HADOOP_HDFS_HOME=$HADOOP_HOME
export YARN_HOME=$HADOOP_HOME

export HADOOP_OPTS="-Djava.library.path=$HADOOP_HOME/lib/native"
export HADOOP_OPTS=”-Djava.library.path=$HADOOP_HOME/lib/native”
export HADOOP_COMMON_LIB_NATIVE_DIR=$HADOOP_HOME/lib/native
export PATH=$PATH:$HADOOP_HOME/sbin:$HADOOP_HOME/bin
```

Reiniciamos

#source .bashrc

#echo $HADOOP_HOME

#echo $JAVA_HOME

Por último ejecutamos el siguiente comando para verificar que se levantó el daemon de Hadoop

#hadoop version

Ahora editamos el archivo $HADOOP_HOME que se encuentra en la siguiente ruta

cd /home/hadoop/hadoopframe/etc/hadoop

Editamos el siguiente archivo

#nano hadoop-env.sh

Y se ponemos fija la variable del JAVA_HOME.

Dentro del archivo anterior buscamos la línea que comience con export JAVA_HOME y la reemplazamos por:

export JAVA_HOME=/opt/jdk/

Reiniciamos

source .bashrc

para ejecutar este reinicio necesitamos estar en la siguiente ruta

#/home/hadoop

y verificamos nuevamente que Hadoop se ejecute de manera correcta aplicando el siguiente comando

#hadoop versión

Una vez ejecutado el comando anterior nos arroja las siguientes líneas

Hadoop 2.9.1

Subversion https://github.com/apache/hadoop.git -r e30710aea4e6e55e69372929106cf119af06fd0e

Compiled by root on 2018-04-16T09:33Z

Compiled with protoc 2.5.0

From source with checksum 7d6d2b655115c6cc336d662cc2b919bd

This command was run using /home/hadoop/hadoop-2.9.1/share/hadoop/common/hadoop-common-2.9.1.jar

- Paso 7

Configuración de los archivos de Hadoop

En la siguiente ruta están los archivos que se editarán

#cd /home/hadoop/hadoopfram

Se crean los directorios en el master

#mkdir /volumen/datanode #mkdir /volumen/namenode

Enseguida estos directorios se copian en los nodos con las siguientes líneas de comandos desde el master.

#scp direcion_ip_De_Los_Nodos mkdir volumen
scp direcion_ip_De_Los_Nodos mkdir volumen/datanode
scp direcion_ip_De_Los_Nodos mkdir volumen/namnode

- Paso 8

Se editará el archivo que se encuentra en la ruta:

/home/hadoop/hadoopfram/etc/hadoop

core-site.xml

Este archivo se configura para el directorio hdfs por defecto para el localhost con las siguientes líneas.

<property>

<nombre> fs.defaultFS </name>

<valor>hdfs://master.hadoop.lan: 9000</ value> (dirección del master)

</property>

- Paso 9

Una vez modificado el archivo en el master se copiará desde éste a los nodos para verificar que el archivo tenga la misma replica y no tenga ningún error de edición eso se ejecutará con el siguiente comando desde el master

#scp core-site.xml dirección_ip_del_nodo:/home/hadoop/hadoopfram/etc/hadoop

El segundo archivo que se edita se encuentra en la misma ruta:

/home/hadoop/hadoopfram/etc/hadoop

hdfs-site.xml

El *hdfs-site.xml* contiene información sobre cómo Hadoop almacenará la información en el clúster.

```
  <name>dfs.replication</name>

<value>1</value>

  </property>

  <property>

    <name>dfs.name.dir</name>

    <value>file://home/hadoop/volumen/namenode</value>

  </property>

  <property>

    <name>dfs.data.dir</name>

    <value>file://home/hadoop/volumen/datanode</value>

  </property>
```

#scp hdfs-site.xml dirección_ip_del_nodo:/home/hadoop/hadoopfram/etc/hadoop

Se crean los directorios volumen datanode y namennode

```
#mkdir /home/hadoop/volume

mkdir -p / home / hadoop / volumen / namenode
mkdir -p / home / hadoop / volumen / datanode
chown -R hadoop: hadoop / home / hadoop / volumen
```

mapred-site.xml

Se utiliza para especificar quien realiza el MapReduce y el lugar dónde se lleva a cabo. En cuanto a lo primero, lo configuramos para que sea hecho por el YARN (*Yet Another Resource Negotiator*), esto es el componente del framework encargado de esta tarea. Como tenemos un único nodo en nuestro clúster, solo habrá una job *map* y otro *reduce*.

Antes de modificar el archivo, debemos renombrarlo. El nombre por defecto es *mapred-site.xml.template* y queremos que pase a ser *mapred-site.xml*. La modificación es:

```
<property>
    <name>mapreduce.framework.name</name>
    <value>yarn</value>
  </property>
```

Y crear los directorios ‘/home/hadoop/workspace/mapred/system‘ y ‘/home/hadoop/workspace/mapred/local’.

yarn-site.xml

Este archivo se utiliza para configurar YARN en Hadoop. Lo que vamos a hacer es habilitar la fase de *Suffle* para que se pueda hacer entre las fases Map y Reduce.

```
<property>
    <name>yarn.nodemanager.aux-services</name>
    <value>mapreduce_shuffle</value>
</property>
```

- Paso 10

Últimos pasos

Ahora que tenemos lista la configuración debemos formatear el HDSF con:

hadoop namenode –format

Y arrancar el clúster. Para esto último nos vamos a '/home/hadoop/sbin/' y ejecutamos:

./start-dfs.sh

./start-yarn.sh

Referencias:

[1] Gandomi, A. y Haider, M. (2015). Más allá del bombo publicitario: conceptos, métodos y análisis de Big Data. *Revista internacional de gestión de la información*, *35* (2), 137-144.

[2] Aguilar, L. J. (2016). *Big Data, Análisis de grandes volúmenes de datos en organizaciones*. Alfaomega Grupo Editor.

[3] Jorge Serrano Cobos, *Big data y analítica web. Estudiar las corrientes y pescar en un océano de datos*, [en línea] (2014).[Consulta: 23 sep.2018] Disponible: https://recyt.fecyt.es/index.php/EPI/article/view/epi.2014.nov.01/16929

[4] Viktor Mayer-Schönberger y Kenneth Cukier, *Big Data. La revolución de los datos masivos*. [en línea]; Madrid, Houghton Mifflin Harcourt 2013 [Consulta: 26 septiembre 2019] Disponible: http://catedradatos.com.ar/media/3.-Big-data.-La-revolucion-de-los-datos-masivos-Noema-Spanish-Edition-Viktor-Mayer-Schonberger-Kenneth-Cukier.pdf

[5] Camargo-Vega, J. J., Camargo-Ortega, J. F., & Joyanes-Aguilar, L. (2015). Conociendo big data. *Facultad de Ingeniería*, *24*(38), 63-77.

[6] Sinisterra, M. M., Henao, T. M. D., & López, E. G. R. *Clúster de balanceo de carga y alta disponibilidad para servicios web y mail*. [en línea]; Informador Técnico (Colombia) Edición 76, Diciembre 2012 [Consulta: 16 Agosto 2019] Disponible: http://revistas.sena.edu.co/index.php/inf_tec/article/view/34/39

[7] Dr. Víctor J. Sosa S*. Sistemas Distribuidos [en línea]. [Consulta: 21- Apr-2019]. Disponible: https://www.tamps.cinvestav.mx/~vjsosa/clases/sd/01_Sist_Distr_intro.pdf*

[8] Mell, P. and Grance, T., 2021. *The NIST Definition of Cloud Computing*. [online] NIST. Disponible: https://www.nist.gov/publications/nist-definition-cloud-computing

[9] Hernández Dominguez, A., & Hernández Yeja, A. (2015). *Acerca de la aplicación de MapReduce+ Hadoop en el tratamiento de Big Data*. *Revista Cubana de Ciencias Informáticas*, [en línea]. [Consulta 01 junio 2018]. Disponible:

http://scielo.sld.cu/scielo.php?script=sci_arttext&pid=S2227-18992015000300004&lng=es&nrm=iso&tlng=es

[10] White, T. (2012). *Hadoop: The definitive guide*. " O'Reilly Media, Inc.". [en línea] Consultado 02 de septiembre de 2018 en http://grut-computing.com/HadoopBook.pdf

[11] acerca de CentOS. (2020). [en línea]. [Consultado el 7 de octubre de 2019] Disponible: https://www.centos.org/about/

[12] Debian - Acerca de Debian. (2020). [en línea]. [Consultado el 7 de octubre de 2019] Disponible: https://www.debian.org/intro/about

[13] Dónde obtener información técnica de Java. (2020). Consultado el 15 de noviembre de 2016. Disponible: https://www.java.com/es/download/faq/techinfo.xml

[14] *Buy a Raspberry Pi – Raspberry Pi* [en línea]. Consultado el 9 de septiembre de 2020. Disponible https://www.raspberrypi.org/products/

[15] Ramon, M. C. (2014). Intel galileo and intel galileo gen 2. In *Intel® Galileo and Intel® Galileo Gen 2* (pp. 1-33). Apress, Berkeley, CA.

[16] H97MPLUS|TarjetasMadre|ASUSMéxico. [en línea] Consultado el 19 de noviembre 2019. Disponible https://www.asus.com/mx/Motherboards/H97MPLUS/

[17] Reveret, J. (2005). apt-build: optimice los paquetes Debian para su sistema. *Mundo Linux: Sólo programadores Linux*, (78), 4.

[18] Hernández Dominguez, A., & Hernández Yeja, A. (2015). *Acerca de la aplicación de MapReduce+ Hadoop en el tratamiento de Big Data*. *Revista Cubana de Ciencias Informáticas*, [en línea]. [Consulta 01 junio 2018]. Disponible: http://scielo.sld.cu/pdf/rcci/v9n3/rcci04315.pdf.

[19] K. S. A. Y. Boris Lublinsky, Hadoop - Soluciones Big Data, ANAYA MULTIMEDIA, 2013. [en línea]. [Consulta 2 Diciembre 2018]. Disponible: http://234075.jugendpflege-stockelsdorf.de/leer/234075/Hadoop%253A%2Bsoluciones%2Bbig%2Bdata

[20] H97M-PLUS | Tarjetas Madre | ASUS México. (2020). Consultado el 15 de noviembre de 2019 en https://www.asus.com/mx/Motherboards/H97MPLUS/

[21] Gordon Sánchez, B. R., & Tacurí Romero, E. G. (2016). *Diseño e implementación de un módulo didáctico de domótica por medio de láminas con la tarjeta raspberry PI y el programa QT en la universidad politécnica salesiana sede Guayaquil* (Bachelor's thesis).

[22] ¿Qué es big data? | *Oracle México*, [en línea]; (2019). [Consulta: 13 Octubre 2019] Disponible: https://www.oracle.com/mx/big-data/guide/what-is-big-data.html