**درس : Big Data استاد: خانم دکتر صفری**

**دانشجو : رقیه یزدانی تمرین : نصب Hadoop و استفاده از آن**

# نصب Hadoop و استفاده از آن

برای نصب Hadoop و انجام مثالهایی از MapReduce، نیاز داریم که حداقل دو VM داشته باشیم که یکی از آنها نقش نود master و بقیه، نقش نودهای slave را داشته باشند. در این گزارش، یک نود master و یک نود slave خواهیم داشت. برای ایجاد VM، از نرم افزار VirtualBox استفاده شده است و برای هر VM، سیستم عامل Ubuntu 18.04 نصب خواهد شد. لذا لازم است که فایل .iso مربوط به این نسخه از ubuntu را دانلود کرده باشیم که حجم آن حدود 2GB است.

# نصب Virtual Box و ایجاد VM

با توجه به اینکه سیستم عامل host (لپ‌تاپ)، windows 10 است، نسخه VirtualBox 6.1.4 متناسب با آن با حجم 110 MB دانلود و نصب گردید. پس از نصب ابتدا مسیر پیش فرض محل ذخیره سازی فایلهای VM از مسیر زیر، تغییر داده شد :

VirtualBox 🡪 File 🡪 preference 🡪 default machine folder

مسیر F:\VirtualBox-VMs برای ایجاد VMها و فایلهای مربوطه تعریف شد.

سپس با انتخاب آیکن new، یک VM از نوع Linux با نسخه ubuntu(64bit) ایجاد شد و برای آن 3GB Memory و یک دیسک 20GB در نظر گرفته شد. سپس از مسیر زیر فایل .iso سیستم عامل ubuntu را به آن معرفی می‌کنیم :

Select vm 🡪 Setting 🡪 Storage 🡪 select cd icon 🡪 choose a disk file 🡪 ubuntu.iso

همچنین از مسیر زیر تعیین می‌کنیم که هنگام بالا آمدن VM، از CD بوت شود و سپس VM را start می‌کنیم تا نصب ubuntu آغاز شود :

Select vm 🡪 Setting 🡪 System 🡪 select CD as first boot order

Select vm 🡪 start

یک VM دیگر هم مشابه بالا به عنوان نود slave ایجاد می کنیم.

# تنظیمات شبکه در Virtual Box

پس از نصب ubuntu، بصورت پیش‌فرض کارت شبکه بصورت NAT نصب می‌شود و این امکان را می‌دهد که از داخل VM مثلاً به اینترنت وصل شد و یا IP مربوط به host را ping کرد. VM بصورت پیش فرض آدرس 10.0.2.15 را می‌گیرد. در این حالت، از host نمی توان VM را ping کرد.

نیازی که برای راه‌اندازی Hadoop وجود دارد، اینست که دو تا VM بسازیم و این VM ها همدیگر را ping کنند و host هم آنها را ping کند. برای پیاده سازی این مساله، VM را stop کرده و ابتدا برای خود virtualBox یک کارت شبکه از نوع HostOnly می‌سازیم که البته پیش فرض یکی دارد و می توان از آن هم استفاده کرد.

VirtualBox : File 🡪 Host Network Manger

سپس برای هر VM به بخش setting رفته و قسمت network را انتخاب کرده و غیر از کارت شبکه موجود که NAT است، کارت شبکه دوم را هم از نوع HostOnly اضافه می‌کنیم و سپس VM را start می‌کنیم.

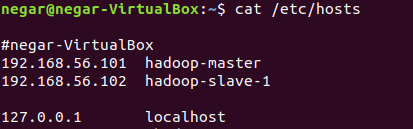
با دستور ip addr show در ترمینال لینوکس، میتوان ip این کارت جدید را دید. حال هم VM می‌تواند host را ping کند و هم host می‌تواند VM را ping کند. بهتر است ip استاتیک تنظیم کرد و فایل /etc/hosts را اصلاح کرد.

# نصب Hadoop

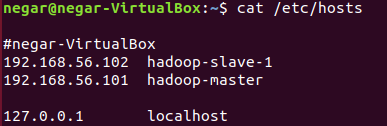
مراحل آماده سازی نودهای master و slave بر اساس داکیومنت تهیه شده توسط آقای مولایی (TA) انجام شد که بصورت کلی شامل مراحل زیر می‌باشد :

* ویرایش فایل /etc/hosts

محتوای فایل برای نود master :



محتوای فایل برای نود slave :

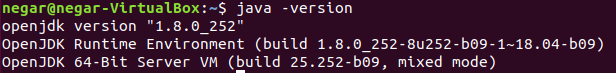


**نکته :** یکی از مشکلاتی که بعد از نصب و اجرای hadoop برای اینجانب پیش آمد، این بود که روی پورت 9000 از 127.0.0.1 در واقع listen می‌کرد نه روی 192.168.56.101 که با جابجا کردن سطرهای مربوطه در این فایل، مشکل حل شد، لذا بهتر است که برای هر نود، نام تعیین شده برای آن بالاتر از سطر localhost قرار گیرد.

* نصب openjdk روی هر دو نود

$ sudo apt-get update

$ sudo apt-get install openjdk-8-jdk



* نصب openssh روی هر دو نود

دستور سوم؛ نسبت به سند نصب آقای مولایی، اضافه شده است.

$ sudo apt-get install openssh-server

$ ssh-keygen -t rsa

$ cat /home/hadoop/.ssh/id\_rsa.pub >> .ssh/authorized\_keys

دستور زیر روی نود master اجرا می‌شود:

$ ssh-copy-id negar@hadoop-slave-1

دستور زیر روی نود slave اجرا می‌شود:

$ ssh-copy-id negar@hadoop-master

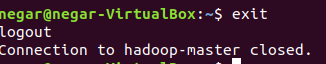
پس از نصب موفقیت‌آمیز ssh، باید بتوان از روی نود master با صدور دستور زیر بدون نیاز به واردکردن Password به نود slave متصل شد :



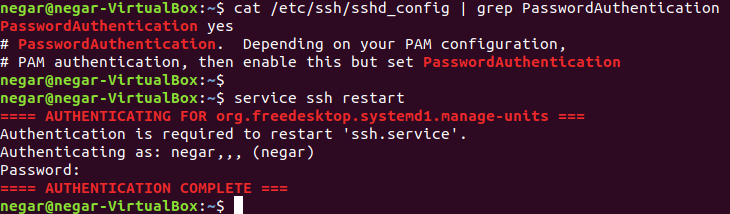


و بالعکس از روی نود slave نیز باید بتوان به همین صورت به نود master متصل شد :





**نکته :** به دلیل اینکه بعداً با استفاده از نرم افزار Bitvise ssh client برای انتقال فایلها از host به این نودها استفاده می‌شود و این نرم افزار از پروتکل ssh استفاده کرده و معمولاً از متد password برای اتصال به نودها استفاده می‌کند، بهتر است که فایل /etc/ssh/sshd\_config را ویرایش کرده و PasswordAuthentication yes را در آن داشته باشیم. پس از آن لازم است که سرویس ssh را restart کنیم.



* دانلود hadoop و انتقال آن به نودهای master و slave

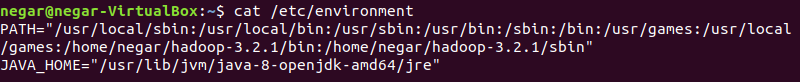
نسخه hadoop 3.2.1 از سایت <https://hadoop.apache.org/releases.html> دانلود شده و از طریق نرم افزار Bitvise ssh client به نودهای master و slave منتقل شده و با دستور زیر extract شد :

$ sudo tar -xzvf hadoop-3.2.1.tar.gz

سپس برای دادن مجوز read,write,exec به all دستور زیر اجرا شد، در این دستور -R اضافه شده، تا دسترسی برای تمام زیرشاخه ها هم داده شود.

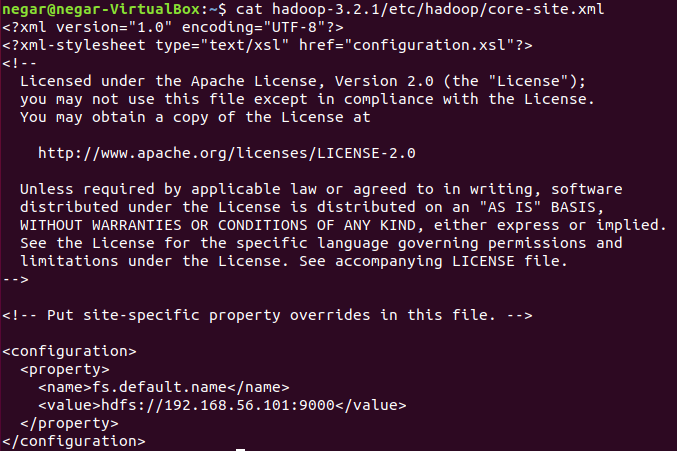
$ sudo chmod -R a+rwx ~/hadoop-3.2.1/

* تنظیم JAVA\_HOME و PATH روی هر دو نود

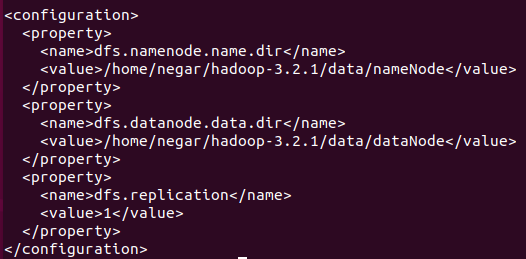


* کانفیگ Hadoop Master و کپی تنظیمات به نود slave

فایل core-site.xml را از مسیر نصب hadoop بصورت زیر ویرایش کرده و بخشهای <configuration> را به آن اضافه می‌کنیم که در واقع پورت 9000 را روی نود master برای listen کردن به نودهای slave تعریف می‌کند :



به همین ترتیب، فایل hadoop-3.2.1/etc/hadoop/hdfs-site.xml را ویرایش کرده و بخش <configuration> را برای معرفی مسیر nameNode و dataNode بصورت زیر به آن اضافه می‌کنیم :



همانطور که از تصویر پیداست لازم است که قبل از آن مسیرهای data/nameNode و data/dataNode در مسیر نصب hadoop ایجاد شود و سپس در فایل hdfs-site.xml به آن ارجاع شود. همچنین با توجه به اینکه یک نود slave داریم، مقدار مشخصه dfs.replication را برابر با 1 مقدار می‌دهیم.

سپس این تنظیمات را با استفاده از دستور زیر به نود slave هم کپی می‌کنیم:

$ scp hadoop-3.2.1/etc/hadoop/\* negar@hadoop-slave-1:/home/negar/hadoop-3.2.1/etc/hadoop/

چنانچه روی نداشتن دسترسی در مقصد، مشکلی وجو داشت، می‌توان ابتدا با دستور زیر owner این فلدر را به کاربری که برای ssh کانفیگ شده، تغییر داد و سپس از scp استفاده کرد.

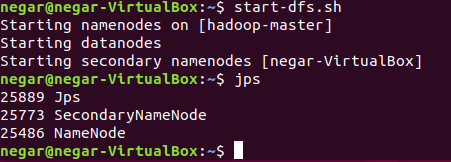
$ sudo chown -R negar:negar Hadoop-3.2.1/

در مرحله بعد hdfs را بصورت زیر فرمت می کنیم :

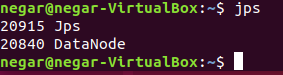
$ source /etc/environment

$ hdfs namenode -format

حال میتوان hdfs را start کرد. خروجی اجرای start-dfs.sh و سپس jps در نود master بصورت زیر است :

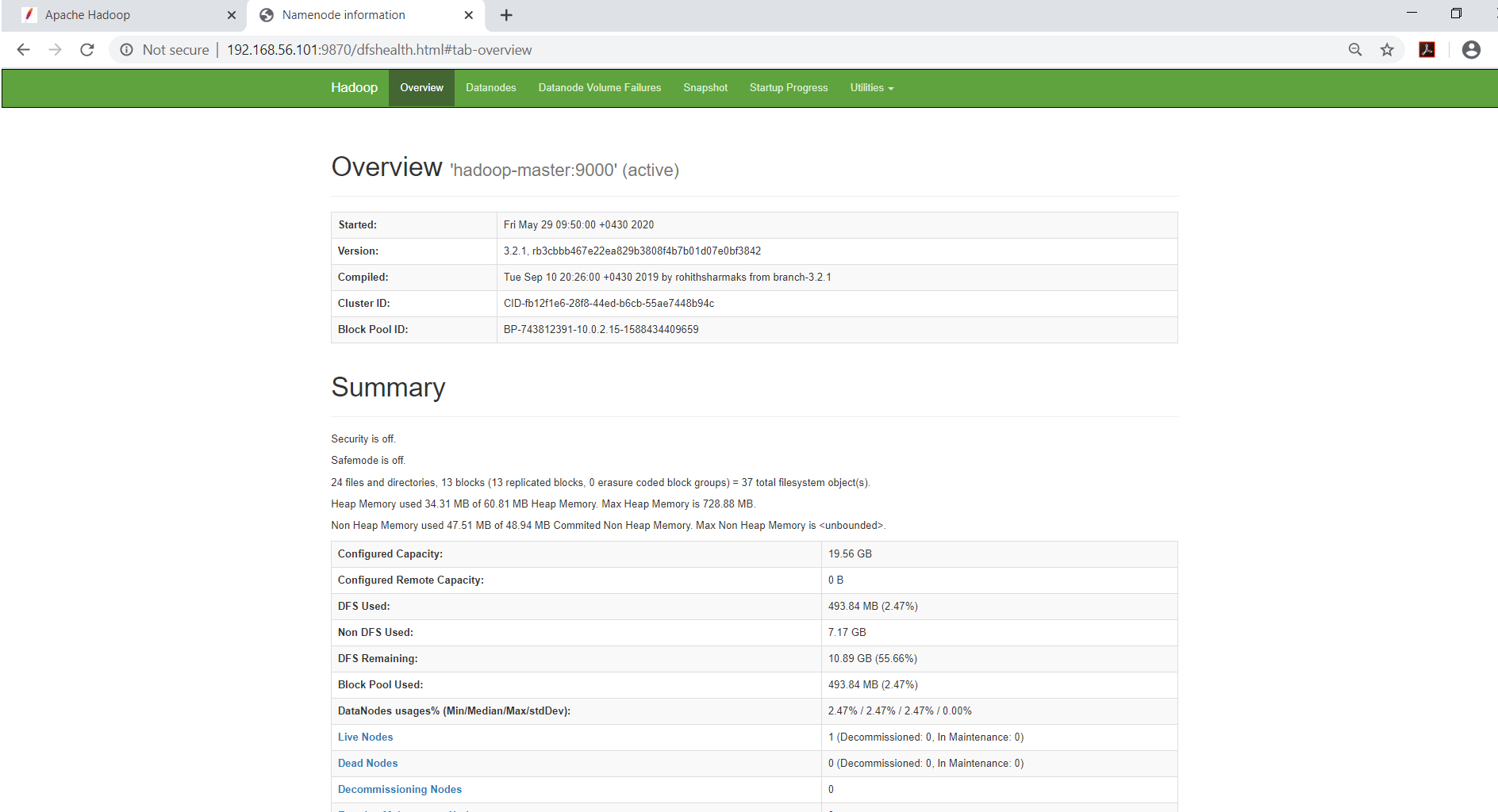


خروجی اجرای jps در نود slave بصورت زیر می‌باشد :



حال می توان با وارد کردن آدرس <http://hadoop-master:9870/> به کنسول وبی hadoop متصل شد. شکل زیر صفحه اول آن را نشان می دهد. (از Host با ip مربوط به master متصل شده ایم)

با توجه به اینکه یک نود slave داریم در بخش، Live Nodes عدد 1 نشان داده شده است.

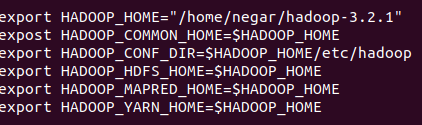


با کلیک بر روی Live Nodes، اطلاعات و ip مربوط به Datanode نمایش داده می‌شود :

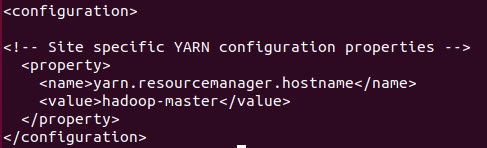


* تنظیمات YARN

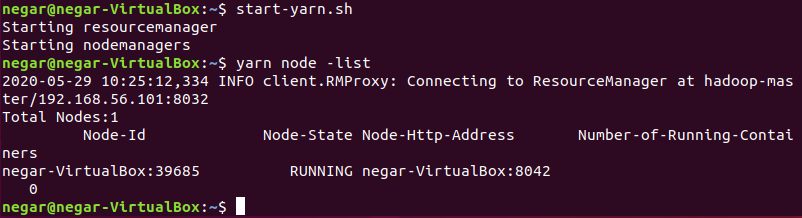
فایل .bashrc را هم در نود master و هم در slave بصورت زیر ویرایش می کنیم :



همچنین فایل hadoop-3.2.1/etc/hadoop/yarn-site.xml را ویرایش کرده و بخش زیر را به آن اضافه می‌کنیم و در واقع نود master را به عنوان resource manager تعیین می‌کنیم :

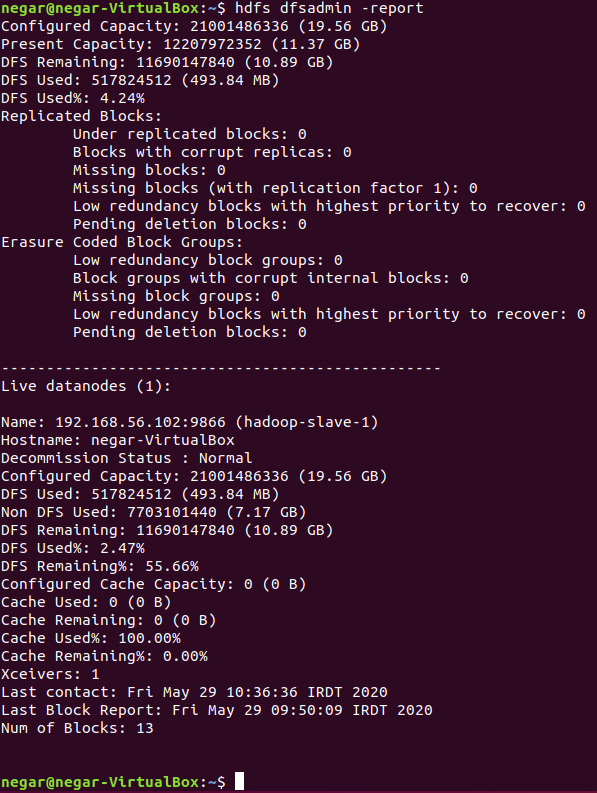


حال با اجرای دستورات زیر در نود master ابتدا yarn را start کرده و سپس لیست نودهای آن که در اینجا یکی است (یک نود slave داریم) را مشاهده می کنیم :



با دستور زیر نیز میتوان تنظیمات کلی hadoop را مشاهده کرد که تصویر خروجی آن در ادامه آمده است :

$ hdfs dfsadmin -report



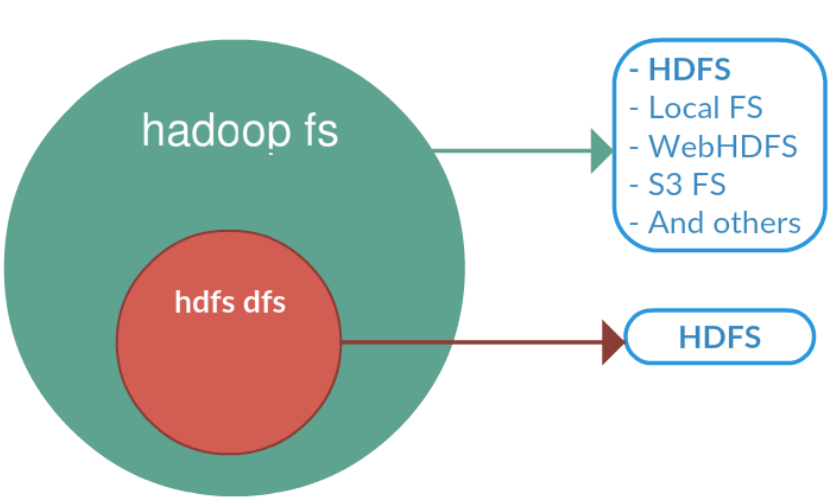
# کپی و انتقال فایل به Hadoop

برای دیدن محتویات مسیر ریشه hdfs می توان از یکی از دستورات زیر استفاده کرد :

$ hadoop fs -ls /

$ hdfs dfs -ls /

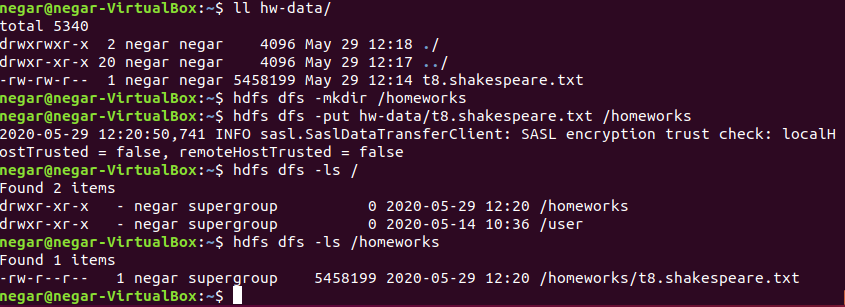
**نکته :** دستور hadoop fs نسبت به hdfs dfs عمومی‌تر است و اجازه کار با انواع فایل‌سیستم ها را می‌دهد اما hdfs dfs تنها برای فایل‌سیستم HDFS قابل استفاده است. در اینجا که ما با HDFS کار می‌کنیم، این دو دستور مترادف هستند و در ادامه گزارش از hdfs dfs استفاده شده است. شکل زیر، این مساله را نشان می‌دهد.



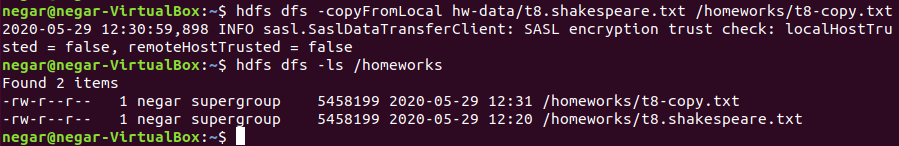
فایل زیر را دانلود کرده و از طریق نرم‌افزار Bitvise به نود master در مسیر hw-data منتقل می‌کنیم:

<http://ocw.mit.edu/ans7870/6/6.006/s08/lecturenotes/files/t8.shakespeare.txt>

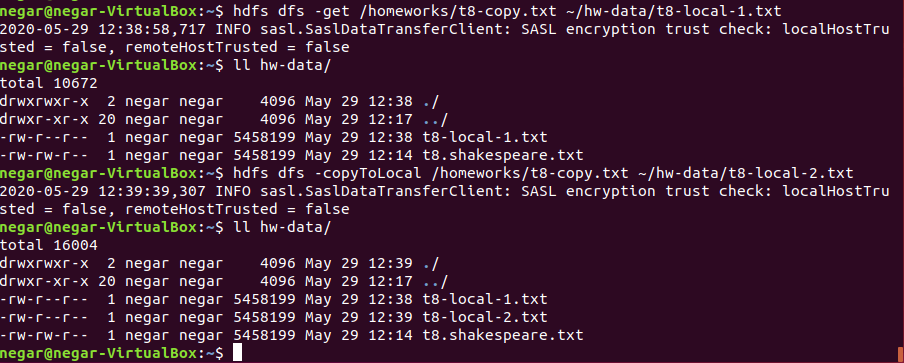
در ادامه روی hdfs یک دایرکتوری به نام homeworks ساخته و فایل بالا را از مسیر hw-data به آن منتقل می‌کنیم :



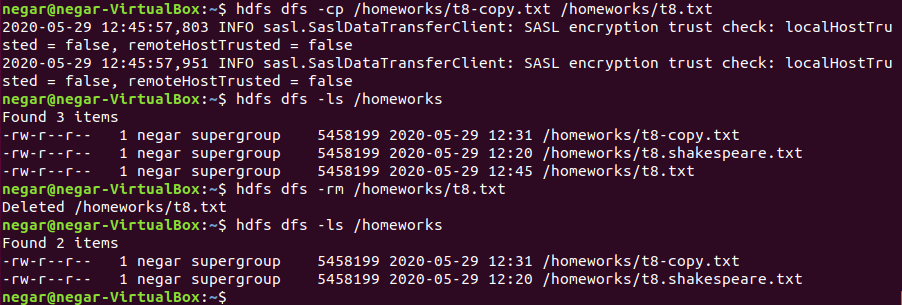
می‌توان بجای -put از -copyFromLocal استفاده کرد. هر دو برای کپی فایل از فایل‌سیستم local به hdfs استفاده می‌شوند. در شکل زیر کپی دیگری از همین فایل با -copyFromLocal در hdfs ایجاد شده است.



بطور برعکس اگر بخواهیم یک فایل را از hdfs به local منتقل کنیم از -get یا -copyToLocal استفاده می‌کنیم. در شکل زیر به هر دو روش، فایل کپی شده است:



برای ایجاد یک کپی از فایل در خود hdfs از -cp و برای حذف یک فایل از hdfs از -rm استفاده می‌شود. در شکل زیر ابتدا یک کپی با نام t8.txt ایجاد شده و سپس حذف شده است :



این موارد را از طریق اینترفیس web هم می‌توان مشاهده کرد که در شکل زیر نشان داده شده است که DFS Browser چگونه محتوای hdfs را نشان می‌دهد. با کلیک روی هر فایل، می‌توان دید که بلاکهای هر فایل و replicat های آن روی چه datanode هایی ذخیره شده‌اند.

