|  |  |  |
| --- | --- | --- |
| **S.No** | **Topic** | **Details** |
|  | **Web Reference** |  |
|  |  | <https://www.youtube.com/watch?v=MSA3ByxmA1s&list=PLf0swTFhTI8pOZ4VBSGerKUmF9USWL6vd&index=2> |
|  | Admin docs | <https://github.com/dgadiraju/code/tree/master/hadoop/administration> |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  | **General info** |  |
| 1 | Log path | /var/log/cloudera-scm-agent  /var/log/cloudera-scm-server |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  | **My AWS Setup** |  |
|  | **Ip address** | Can ping from any internet  Private and public IPs  10.0.0.250 52.24.68.5  10.0.0.251 52.32.1.150  10.0.0.252 52.33.131.153  Private IP – for intranet (or) once login to any ip, use this ip to connect internally   * Ping 10.0.0.12 * Ping 10.0.0.13 * Ping 10.0.0.14   Root password: hadoop1  Hduser password:hadoop2 |
|  | **Pre-request/Initial setup for new nodes** |  |
| 1 | **To connect to <Master & Slave Nodes>** | “kumar\_kp3.pem” is the Amazon Web Service – ‘keypair’ name explained in ‘QuickRefAWS’  Make sure to keep keyPair (kumar\_kp3.pem) under >Cygwin/home/kumar/.ssh  >chmod 600 kumar\_kp3.pem  Login using  ssh -i "kumar\_kp3.pem" ec2-user@ec2-52-24-68-5.us-west-2.compute.amazonaws.com  ssh -i "kumar\_kp3.pem" ec2-user@ec2-52-32-1-150.us-west-2.compute.amazonaws.com  ssh -i "kumar\_kp3.pem" ec2-user@ec2-52-33-131-153.us-west-2.compute.amazonaws.com |
| 1.1 | Disk space | # To get the disk space  >df -h  >df -i |
| 1.2 | Root password  <hadoop1> | # Set password for root:  >sudo passwd root -- need to set the new password. Ex: hadoop1 |
| 1.3 | Login to root in both master and slave | >sudo su - root -- NO Need to enter password  > su root -- need to enter password : hadoop1 |
| **2** | **Rename host**  **<Master & Slave Nodes>** | Do NOT Rename the host with some random name, CM and CDH is getting hostname from different location, so mismatch occurred between host and Cloudera Manager. Use the below steps for reference but do not apply step 2 for rename the host |
| 2.1 | Check the current host name | >hostname –l ---- will give you the options under hostname command  >hostname –v --- will give you the hostname. /etc/sysconfig/network  >hostname –f --- will give you the actual hostname |
| 2.2 | Option1: edit network using perl (not working) | >perl –p –i –e “s/localhost.localdomain/${hostname}/g” /etc/sysconfig/network |
| 2.3 | Option2 **:** edit network using vi editor  *<To setup alias for current private ip. Do NOT include ‘all’ private ips but only include current ip alias>* | **># STEP 1 -- Update the current hostname in /etc/sysconfig/network (use public IP id, so that it will be unique)**  >cd /etc/sysconfig/  >vi network  HOSTNAME=<identify the public host and assign>  ec2-52-24-68-5.us-west-2.compute.amazonaws.com  ec2-52-32-1-150.us-west-2.compute.amazonaws.com  ec2-52-33-131-153.us-west-2.compute.amazonaws.com  **Ex:**  # old hostname in network  HOSTNAME=ip-10-0-0-12.us-west-2.compute.internal  # new hostname in network  HOSTNAME= ec2-52-24-68-5.us-west-2.compute.amazonaws.com  (or)  HOSTNAME= ec2-52-32-1-150.us-west-2.compute.amazonaws.com  (or)  HOSTNAME= ec2-52-33-131-153.us-west-2.compute.amazonaws.com  # save the networkfile |
| 2.4 | Option 2.1: edit hosts  *<To setup alias for each private ip. Need to include ‘all’ private ips and alias here>* | **> ## STEP 2 -- add all hostnames in /etc/hosts. This is the alias for each. Use public IP as alias for private ip (mandatory)**  >sudo su - root  > cd /etc/  > vi hosts  # Do not give any random hostname  Private vs public ‘actual’ host (public ip given as alias and hostname)  10.0.0.250 ec2-52-24-68-5.us-west-2.compute.amazonaws.com  (and)  10.0.0.251 ec2-52-32-1-150.us-west-2.compute.amazonaws.com  (and)  10.0.0.252 ec2-52-33-131-153.us-west-2.compute.amazonaws.com |
| 2.5 | Do the same change in all nodes.. (or) copy paste the hosts to all nodes (from master) | Better open individual nodes instead of copy paste /etc/hosts to all nodes  Ex: for copy command.. just for reference  scp /etc/hosts root@hdslave1:/etc/hosts  scp /etc/hosts root@hdslave2:/etc/hosts |
| 2.5 | Reboot | Reboot the server one the hostname is updated. It is mandatory  >sudo reboot  >hostname -v |
| 2.6 | Check the current host name | >hostname –l ---- will give you the options under hostname command  >hostname –v --- will give you the hostname. /etc/sysconfig/network  >hostname –f --- will give you the actual hostname |
| 2.7 | Ping to alias | >ping <hostname>  >ping hdmaster  >ping hdslave1  >ping hdslave2  >ping 52.24.68.5  >ping 52.32.1.150  >ping 52.33.131.153 |
| **3** | **Add groups and users and give access to the user**  **<Master & Slave Nodes>** | #Add a group name called ‘hadoop’ and user ‘hduser’  # Apply this command from root |
| 3.1 | Create group & user. Assign user to wheel  <hduser pwd: hadoop2> | >groupadd hadoop;  >useradd hduser –g hadoop;  >passwd hduser; -- Ex: hadoop2  >usermod –aG wheel hduser; -- *Ubuntu doesn’t has wheel*  Both UNIX and Linux have a ‘default’ group called wheel. If user is member of this group he/she can use su command and sudo to root. |
| 3.2 | To chk the user (hduser) details | >id hduser; -- after run the above command |
| 3.3 | Check sudoers | >cat /etc/sudoers – before run the below perl command  *Note: sudoers has a command by default. Running perl command will ‘uncomment’ the below wheel lines*  # %wheel ALL=(ALL) ALL  # %wheel ALL=(ALL) NOPASSWD:ALL |
| 3.4 | Update sudoers (to include hduser) | #Add hduser to sudoers. so that hduser can apply sudo commands  Instead of open and edit sudoers, we are applying perl command to find and replace without open sudoers. Here s/ represent string, followed by the string to be find, followed by string to be replaced and /g represent globally  #Add hduser to sudoers (visudo) and set up password using passwd command  >perl –p –i –e “s/\# \%wheel/\%wheel/g” /etc/sudoers |
| 3.5 | Check sudoers after update | >cat /etc/sudoers – after run the below perl command |
|  |  |  |
| **4** | **Firewall & Cloudera Services**  **<Master & Slave Nodes>** | **Stop and disable the Firewall: iptables, ip6tables, ipchains, etc** |
| 4.1 | stop and disable Firewall services.  Also…  *<Start cloudera-scm services>* | #stop and disable Firewalls and make sure the chkconfig is off. Start cloudera-scm-server and chkconfig is on for the same  >sudo su root  >**service iptables status;**  **>service iptables stop;**  **>service ip6tables status;**  **>service ip6tables stop;**  **> service cloudera-scm-server status**  **> service cloudera-scm-server start**  **>service cloudera-scm-agent status**  **>service cloudera-scm-agent start**  **>service cloudera-scm-server-db status**  **>service cloudera-scm-server-db start**  **>service mysqld status**  **>service mysqld start**  >service ipchains status;  >service ipchains stop;  **>chkconfig --list iptables**  **>chkconfig iptables off**  **> chkconfig --list ip6tables**  **>chkconfig ip6tables off**  **> chkconfig --list cloudera-scm-server**  **>chkconfig cloudera-scm-server on**  > chkconfig --list cloudera-scm-agent  >chkconfig cloudera-scm-agent on  > chkconfig --list cloudera-scm-server-db  >chkconfig cloudera-scm-server-db on  > chkconfig --list mysqld  >chkconfig mysqld on  > chkconfig --list ipchains  > chkconfig ipchains off |
| **4.2** | **Port**  <To check telnet port is open or not> | How to get port #?  > cat /etc/services | grep netstat  >netstat -lpn  >netstat –lp  > netstat -nltp | grep 7180 |
| **4.3** | **Port**  **<** **how to make linux listen on a port>**  *<Forcing linux to use a particular port but it is temporary>* | *Forcing linux to use a particular port but it is temporary*  >nc –l <port#>  Ex: 3 steps  >netstat -nat | grep 7180  > nc -l 7180  >netstat -nat | grep 7180  <http://unix.stackexchange.com/questions/214471/how-to-create-a-tcp-listener> |
| **4.4** | Disable SELINUX (manual)  Firewall | >vi /etc/selinux/config  SELINUX=disabled  >sestatus -- check after reboot |
| **4.4** | **Disable selinux (automatic)**  **<Master & Slave Nodes>** | **Not only for Hadoop, for most of the software, it should be disabled as a standard process. It is enabled by default** |
| 4.4 | **Disable selinux (automatic)**  **<Master & Slave Nodes>** | >perl -p -i -e "s/^SELINUX=enforcing/SELINUX=disabled/g" /etc/selinux/config |
| **5** | **Enable password authentication** *<<Optional>>*  **<Master & Slave Nodes>** |  |
| 5.1 |  | #Enable password authentication  >cd /etc/ssh  >cat sshd\_config  >perl -p -i -e "s/^PasswordAuthentication no/PasswordAuthentication yes/g" /etc/ssh/sshd\_config  >service sshd restart |
| **7** | **Disable vm.swappiness**  **<Master & Slave Nodes>** | ***<<*** *Virtual Memory Swappniess>>* |
| 7.1 | Command | >cd /etc  >cat sysctl.conf  >echo “vm.swappiness=0” >> /etc/sysctl.conf  Refer quick\_ref\_unix\_linux.doc (Topic Swap Memory) |
| 7.2 | Situation 1 | You want to node to only swap Hadoop daemon data from RAM to disk when absolutely necessary. What should you do?  **Ans:** Set vm.swappiness to 0 in /etc/sysctl.conf |
| 7.3 | Situation 2 | You suspect that your NameNode is incorrectly configured, and is swapping memory to disk. Which Linux commands help you to identify whether swapping is occurring?  Ans: Top, Free and vmstat |
| 7.4 | Reboot the server | >sudo su root  >init 6 – To reboot |
| 8.0 | Login to root or hduser | Exit from root. From Now on, you can login to root (or) hduser to access your cluster.  Ex:  >ssh hduser@NovisyncCDH500 |
| 8.0 | Remote connect using alias  *<make sure user name hduser is available in the remote>* | # login to hdmaster then ssh to hdslave1 . This throws the below error because ssh-keygen process not yet completed.  >ssh hduser@hdslave1  >ssh [hduser@52.24.68.5](mailto:hduser@52.24.68.5)  >ssh [hduser@52.32.1.150](mailto:hduser@52.32.1.150)  >ssh [hduser@52.33.131.153](mailto:hduser@52.33.131.153)  *<The authenticity of host 'hdslave1 (10.0.0.13)' can't be established>* |
| 8.0 | **Reboot the server**  **<Master & Slave Nodes>** | # to reboot  > init 6 |
| 9 | **Password less login between all nodes.**  **Setup ssh**  **<ONLY on Master>** | 1. Need to setup ssh to provide password less login from node to master, so that the process won’t stop in the middle for password. 2. Login to hduser 3. In master, Run ssh-keygen command, this will create a hidden directory called .ssh under current user and generate two files: 4. Id\_rsa 5. Id\_rsa.pub 6. Copy the above two files to all the slave/agent. 7. This is a general practice, not only for Hadoop.   *<below commands are Just for reference, do NOT execute>*  *To check that you can ssh to the localhost without a passphrase. If ssh localhost is not working then try the below steps*  *$ ssh-keygen -t dsa -P '' -f ~/.ssh/id\_dsa*  *$ cat ~/.ssh/id\_dsa.pub >> ~/.ssh/authorized\_keys*  *$ chmod 0600 ~/.ssh/authorized\_keys* |
| 9.1 | >ssh-keygen  *<to be done from hduser, not from root>* | This command will create .ssh folder under current user (hduser). Run the ssh-keygen on the master node where cloudera manager will be installed  >sudo su hduser  >ssh-keygen  *<<No password, just press enter without password>>* |
| 10 | **Copy id\_rsa.pub** | Copy id\_rs.pub to all nodes in Cluster (created on master and copy to client machine)  Ref: quick\_ref\_unix\_linux.doc (id\_rs.pub) |
| 10.1 | This command will create a directory called .ssh under hduser  **<Master & Slave Nodes>** | In the master , change the .ssh permission using chmod.  > chmod 700 ~/.ssh  In Slave, create .ssh and chmod to 700  >ssh hduser@hdslave1 'mkdir -p ~/.ssh;chmod 700 ~/.ssh'  >ssh hduser@hdslave2 'mkdir -p ~/.ssh;chmod 700 ~/.ssh'   * After hostname change   > chmod 700 ~/.ssh  >ssh hduser@52.32.1.150 'mkdir -p ~/.ssh;chmod 700 ~/.ssh'  >ssh hduser@52.33.131.153 'mkdir -p ~/.ssh;chmod 700 ~/.ssh' |
| 10.2 | Copy Id\_rsa.pub to slave nodes from master node  *<to be executed from hduser>*  *<make sure id\_rsa.pub is available under hduser/.ssh/>* | Id\_rsa.pub from master node will be moved to authorized\_keys in slave node  >cat ~/.ssh/id\_rsa.pub | ssh hduser@hdslave1 "cat >> ~/.ssh/authorized\_keys"  >cat ~/.ssh/id\_rsa.pub | ssh hduser@hdslave2 "cat >> ~/.ssh/authorized\_keys"   * After hostname change   >cat ~/.ssh/id\_rsa.pub | ssh hduser@52.32.1.150 "cat >> ~/.ssh/authorized\_keys"  >cat ~/.ssh/id\_rsa.pub | ssh hduser@52.33.131.153 "cat >> ~/.ssh/authorized\_keys" |
| 10.3 | Validate passwordless login between all nodes | # Make sure to come out of root and login as hduser  Login to hduser from master. Connect to all the slaves (hduser) without password |
|  |  |  |
|  |  |  |
| **10** | **Prepare all nodes** | **#check file system on all nodes** |
| 10.1 | *<optional>* | >ssh hduser@52.24.68.5 "df -h"  >ssh hduser@52.32.1.150 "df -h"  >ssh hduser@52.33.131.153 "df -h" |
| **11** | **Download Cloudera binaries** | **#Download cloudera binaries on hduser**  ***<<Details available in QuickRef\_CDH\_CM.doc>>*** |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |
|  |  |  |