**# Procedure for the automatic function of creating cluster(code expalnation)**

##########################################################

**#!/usr/bin/python**

**print "Content-Type: text/html"**

**print**

**import os**

**import commands**

**import operator**

**import cgitb**

**import cgi**

**cgitb.enable()**

**c=commands.getstatusoutput("nmap -sP 192.168.56.0-255 -n | grep 'Nmap scan' |awk '{print $5}'")**

**print c**

**y=open('b.txt',mode='w')**

**y.write(c[1])**

**y.close()**

**fh=open('b.txt')**

###################################################################

*1.* "#!/usr/bin/python" shebang is added in order to make the file executable. /usr/bin/python is the path. 'os ' module of python is also imported to use the 'system' function of this module which enables to run linux commands using python scripts .

**2.**'cgi ' and 'cgitb 'modules are also imported to enable python-cgi feature ie website supporting features of python.''cgitb.enable()'' enables the module to be used.

**3. )** the 'getstatusoutput' function of 'commands ' module is used to collect the data in string format .

-- 'nmap -sp 192.168.56.0-255 -n' this linux commands goes in to the network and collects the ip ranging from 192.168.56.0 to 192.168.56.255 .

--'grep ' command is used to collect the desired output from a list of various output.In this case we want to extract output lines starting from 'Nmap scan' from the output generated by the command 'nmap -sp 192.168.56.0-255 -n'.

--'awk ' commad is used to cut a string .'print $5' is the parameter passed to the 'awk ' command specifying the criteria for ranging of the characters to be cut out.Here we need the 5th field , thus $5 is passed .

--'"y=open('b.txt',mode='wr')'" this python command creates a file (if not already created) named 'b.txt' with reading and writing permissions given to it .

--"y,write and y.close" functions are used to write data in 'b.txt' and then close it after writing the data.

--"fh=open('b.txt')" this copies the content of b.txt in fh.

-------------------------------------------------------------------------------------------------------------------------------

4. Further code explanation:-

###################################################################

**l=[]**

**for x in fh:**

**if (x=='192.168.56.1\n' or (x=='192.168.56.100\n') or (x=='192.168.56.101')):**

**pass**

**else:**

**p=x.strip()**

**l.append(p)**

**md=dict()**

**for x in l:**

**if (x=='192.168.56.1\n' or x=='192.168.56.100\n' or x=='192.168.56.101'):**

**pass**

**else:**

**g=commands.getstatusoutput("sudo sshpass -p redhat ssh -l root "+x+" free -m |awk 'NR==2{print $2}'")**

**print g**

**d=dict()**

**d[x]=int (g[1])**

**md.update(d)**

**md\_list=sorted(md.items(),key=operator.itemgetter(1))**

###################################################################

**i)**  a empty dictionary is created named 'l' using 'l=[]'

ii) 'for x in fh' for loop now excess the data in variable 'fh' one by one . Ip's 192.168.56.1 and 192.168.56.100 ,192.168.56.102,192.168.56.101 are ip of the main machine and of thevnetwork which need to be excluded and thus they are passed via 'if' and the remaining ip's are appended in the list 'l' after stripping the strings.

**ii)** an empty dictionary is created using the keyword 'dict' and names as ''md'.

iii)now the data in the list 'l' is accessed . using 'ssh ' commands as used earlier 'free -m ' is executed on the nodes . "free -m " collects the RAM data on the machine . Thus all the free available RAM is collected by this command for all the nodes . 'awk ' selects the second fiels of the output of the 'free -m' and then this integer data is saved in the dictionary which is then further added in the md .

**iii)** 'md\_list ' sorts the items if the list 'md' using the key to sort it .the key contains the RAM(free and available ) in descending order.

-------------------------------------------------------------------------------------------------------------------------------

5. Further code explanation:-

###################################################################

**ind=0**

**cnt=1**

**for i in md\_list:**

**if cnt==1:**

**print """<table border="1">**

**<th>IP OF SYSTEM</th>**

**<th>FREE-RAM </th>"""**

**print """<tr>**

**<td>%s </td>"""%md\_list[ind][0]**

**print """<td>%s</td>"""%md\_list[ind][1]**

**print """</tr>"""**

**cnt=cnt+1**

**ind=ind+1**

**print """<form action='http://192.168.56.101/cgi-bin/check.py'>**

**Namenode-ip<input type="text" name='nn' /><br />**

**Job-Tracker<input type="text" name='jt' /><br />**

**"""**

**for i in range(0,(len(md\_list)-2),1):**

**print """Datanode<input type="text" name='dn'/><br/>"""**

**print """<input type="submit" /></form>"""**

###################################################################

-- in the above code the first loop prints the ip's and provide a option of making it a namenode or a datanode to every ip. User can select only one ip as namenode and multiple ips for data'node. this data is sent to the other program.