The XXX project

----Environmental installation by skymind

文件修订历史记录(Document revision history)			
Version	revision date	author	decription
1.0.0	July 26th,2017	Wang Feng	first draft
1.0.1	July 28th,2017	W F	Perform tasks according to document
1.0.2	August 6th,2017	W F	Improving document

（一）OS install

① The operation system is cenos 7.3

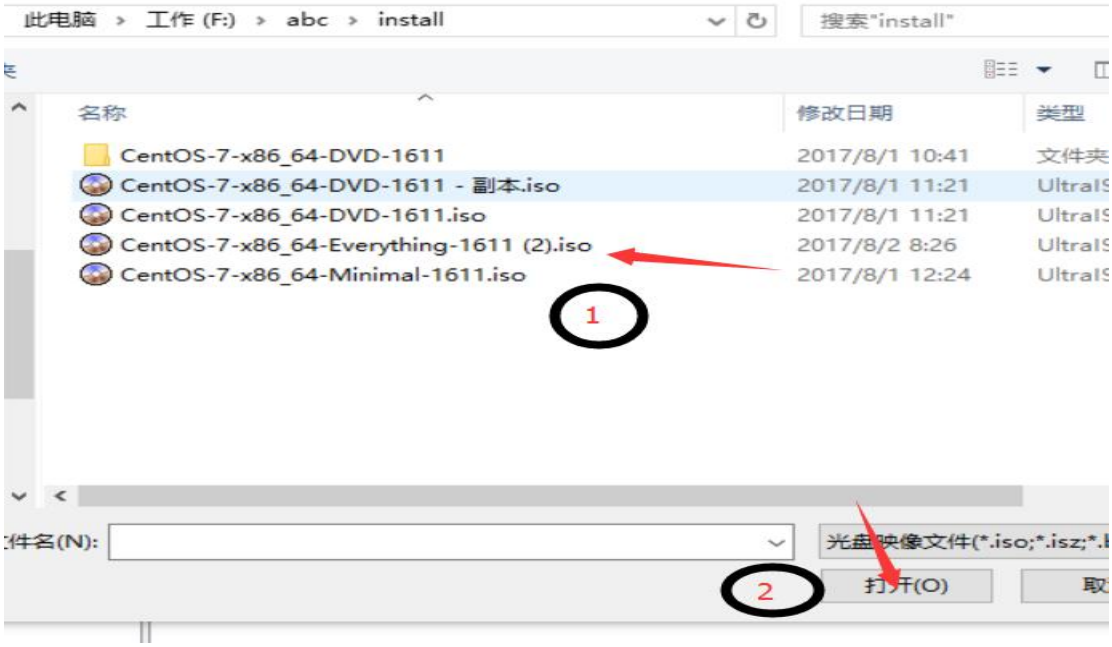
② The os link: https://www.centos.org/download/

③ Download ultraISO,this is link:www.ezbsystems.com

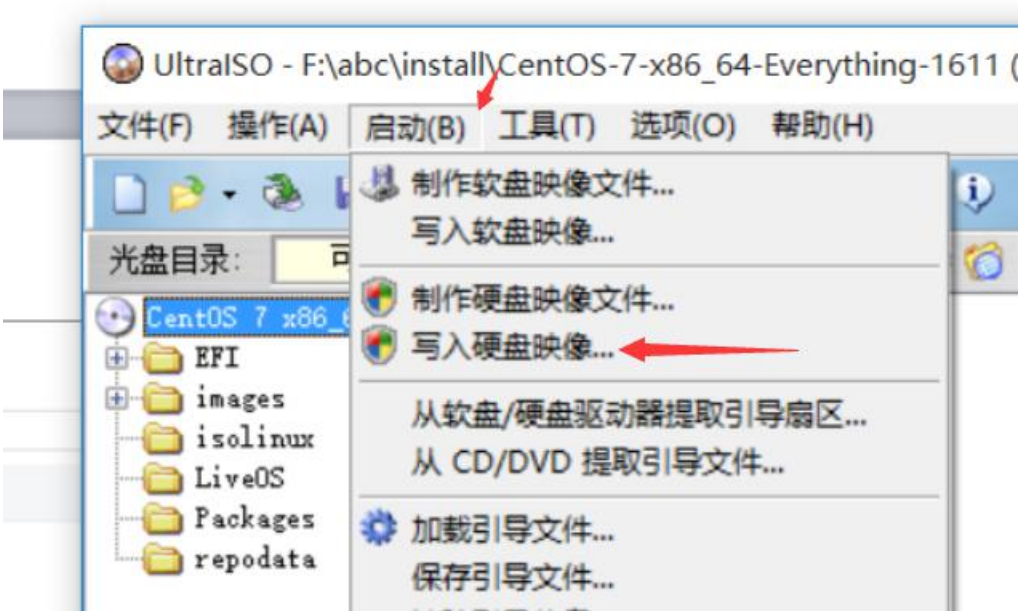
poll code ==========:

④ Making U disk

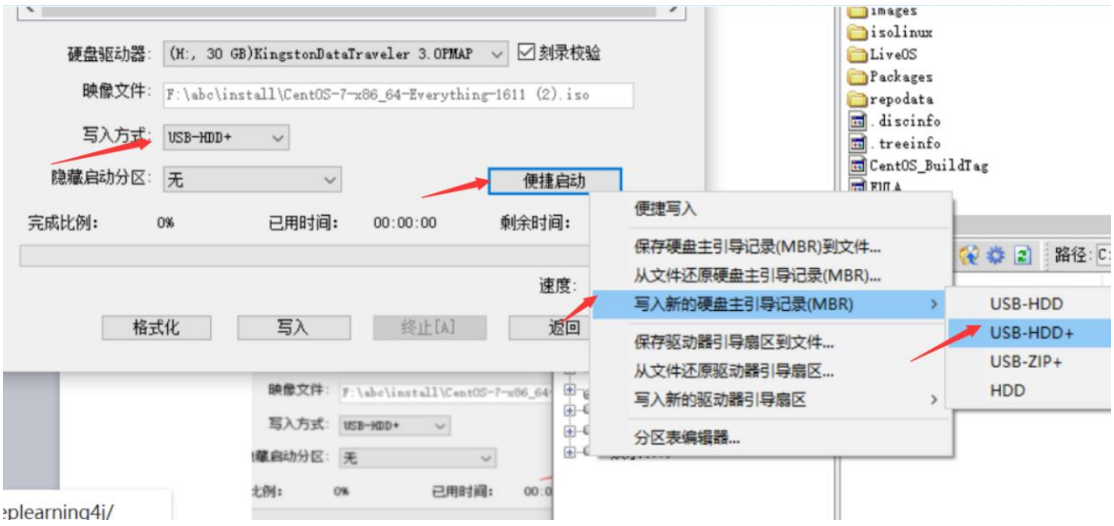
Opening ultraISO software,and then File->open->choice cenos-xxx.iso



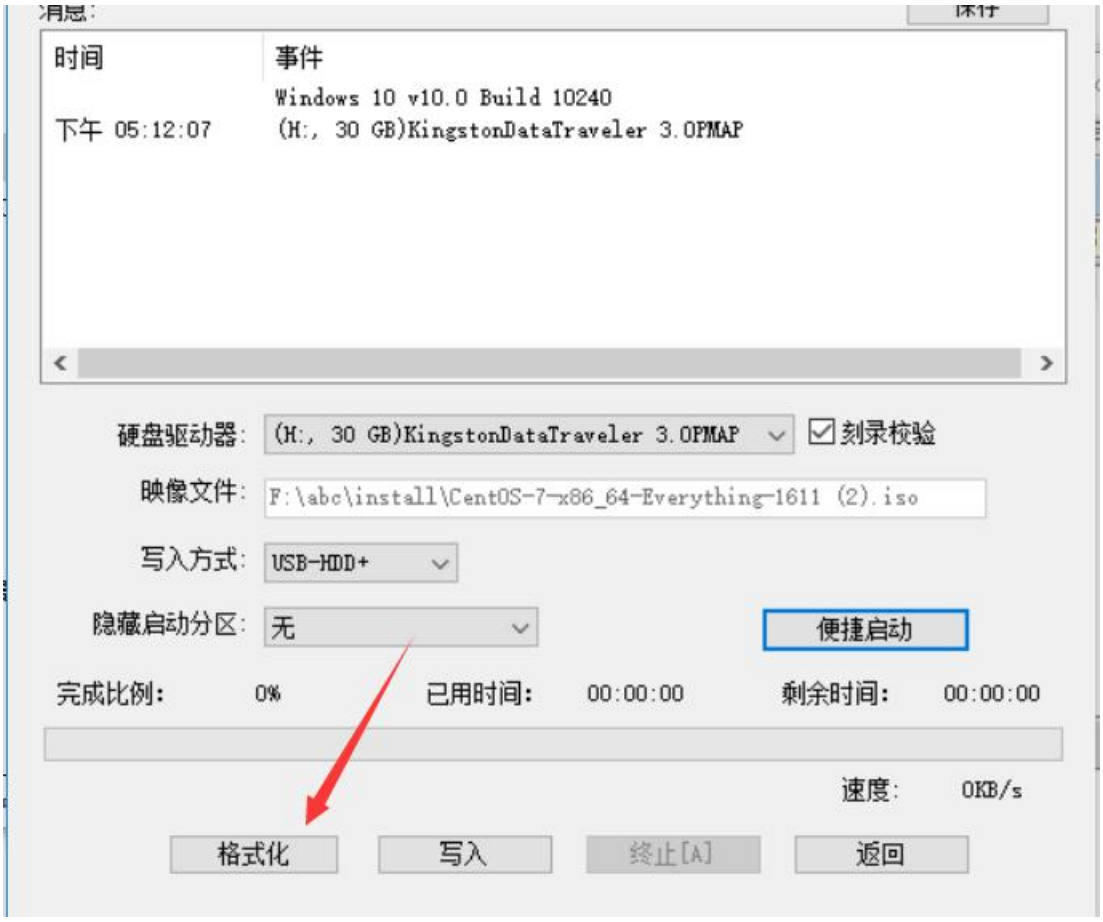
Then click: run(启动)->write disk mirror image (写入硬盘映像)，As I use the Chinese version software, so the translation into English may not match the vocabulary of the english software, please check it



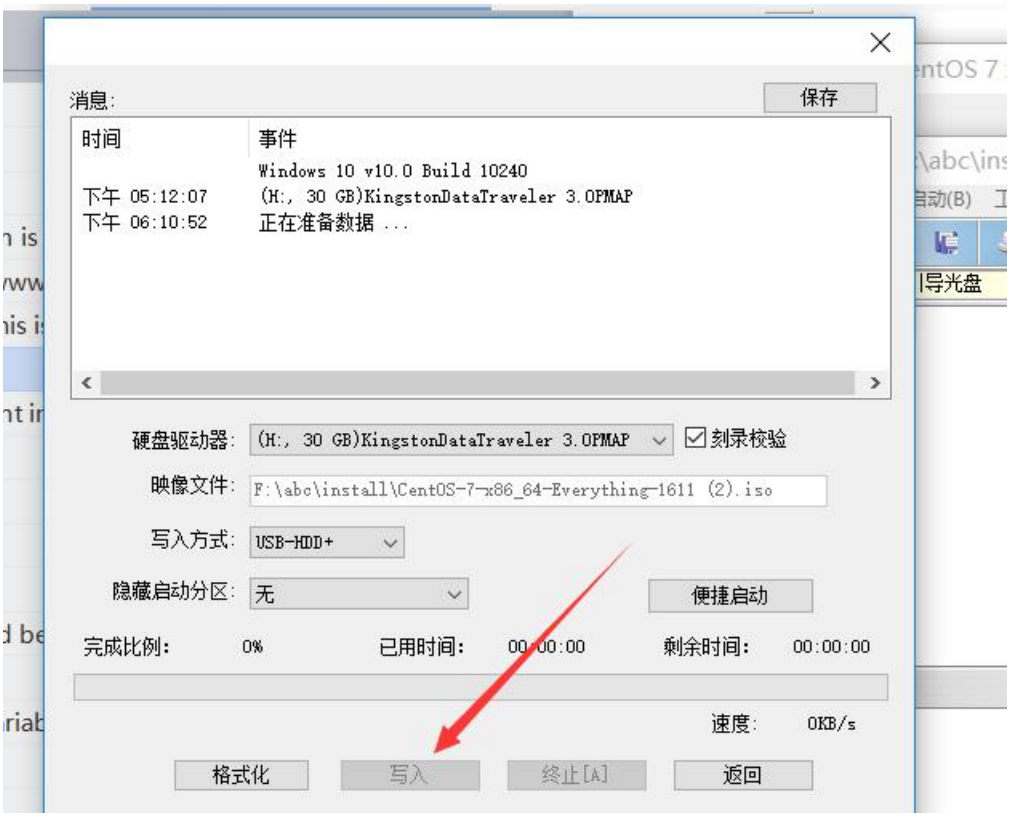
And then click quickly start,choice write a new hard disk master boot record(MBR),click usb-hdd+



Then formatting


消息:
保存
时间
事件
Windows 10 v10.0 Build 10240
下午 05:12:07
(H:, 30 GB)KingstonDataTraveler 3.0PMAP
硬盘驱动器:
(H:, 30 GB)KingstonDataTraveler 3.0PMAP
刻录校验
映像文件:
F:\abc\install\CentOS-7-x86_64-Everything-1611 (2).iso
写入方式:
USB-HDD+
隐藏启动分区:
无
便捷启动
完成比例:
0%
已用时间:
00:00:00
剩余时间:
00:00:00
速度:
0KB/s
格式化
写入
终止[A]
返回

And then write :



Then ,Inserting the U disk to the machine that should be set USB U disk starting in the blos

（二）Base environment install

yum groupinstall 'Development Tools' or yum install "Development Tools"
yum install glibc-static libstdc++-static
yum install kernel-devel kernel-headers #gcc make
#yum install python-devel

(三) Cmake install

① Website
The office website is https://cmake.org/
wget https://cmake.org/files/v3.9/cmake-3.9.0-Linux-x86_64.tar.gz
② Whether the cmake had been installed
rpm -qa|grep cmake
cmake --version(if the version is less than 3.2,installed again)
If the cmake is exist, delete it :yum remove cmake

```
[root@d8 ~]# cmake -version
cmake version 2.8.12.2
[root@d8 ~]# rpm -qa|grep cmake
cmake-2.8.12.2-2.el7.x86_64
[root@d8 ~]# yum remove cmake
```

③ Install cmake

Download Source Distribution or Binary Distribution,here is binary distribution,so

tar -zxvf cmake-3.9.0-Linux-x86_64.tar.gz

mv cmake-3.9.0-Linux-x86_64 /usr

④ Setting eviroment variable

Vi /etc/profile

```
#cmake path
export CMAKE_HOME=/usr/cmake-3.9.0-Linux-x86_64
export PATH=$PATH:$CMAKE_HOME/bin
```

⑤ Test

```
[root@d10 usr]# cmake -version
cmake version 3.9.0

CMake suite maintained and supported by Kitware (kitware.com/cmake).
```

（四）Gcc install

Tip:the cuda 8 must compile by gcc that the version is less than 5

I had installed that is 4.9 version again(these is same intalling way)

① website

The office website is http://gcc.gnu.org/

wget ftp://nl.mirror.babylon.network/gcc/releases/gcc-7.1.0/gcc-7.1.0.tar.gz

Other links:https://gcc.gnu.org/mirrors.html

② Whether the gcc had been installed

rpm -qa|grep gcc

If the gcc vesion is less than 4.9,and then install again,but old gcc may not be deleted

③ Install gcc

tar -jxvf gcc-7.1.0.tar.gz

cd gcc-7.1.0

./contrib/download_prerequisites

mkdir build

cd build

../configure --enable-checking=release --enable-languages=c,c++ --disable-multilib

make

make install

Shutdown -r now

④ Test

```
[root@d10 ~]# gcc -v
Using built-in specs.
COLLECT_GCC=gcc
COLLECT_LTO_WRAPPER=/usr/local/libexec/gcc/x86_64-pc-linux-gnu/7.1.0/lto-wrapper
Target: x86_64-pc-linux-gnu
Configured with: ../configure --enable-checking=release --enable-languages=c,c++ --disable-multilib
Thread model: posix
gcc version 7.1.0 (GCC)
```

⑤ deleted Residual gcc 4.8,as follow

[root@d10 libnd4j]# rpm -qa|grep gcc

gcc-gfortran-4.8.5-11.el7.x86_64

gcc-4.8.5-11.el7.x86_64

rpm -e xxx(gcc*),

⑥ Maybe do some check,as follow(but I didn't do)

⑦ Check the dynamic library

strings /usr/lib64/libstdc++.so.6 | grep GLIBC

If the GCC dynamic library is an old version of the show. These problems, because when upgrading GCC, DLL dynamic library generation does not replace the old version of GCC.

Handling problems:

Execute the following command, generates a search GCC compiled the latest library:

⑧ Find / -name "libstdc++.so*"

⑨ The latest libstdc++.so.6.0.21 library is copied to the /usr/lib64 directory:

/home/yl/gcc-build-4.9.2/x86_64-unknown-linux-gnu/libstdc++-v3/src/.libs/libstdc++.so.6.0.20 /usr/lib64/

⑩ After copying, to modify the default dynamic library system: soft connection reconstruction default library.switch to the /usr/lib64 directory:

cd /usr/lib64/

⑪ Delete the original soft connection:

rm -rf libstdc++.so.6

⑫ The default library soft connection to the latest library:

ln -s libstdc++.so.6.0.20 libstdc++.so.6

⑬ The default upgrade is complete. The dynamic re run the following command to check the dynamic library:

Strings /usr/lib64/libstdc++.so.6 | grep GLIBC

⑭ To complete the installation.

（五）JDK install

The version of the java is 1.8 and 64bit

Website:

http://download.oracle.com/otn-pub/java/jdk/8u144-b01/090f390dda5b47b9b721c7dfaa008135/jdk-8u144-linux-x64.tar.gz

（六）Maven install

The version of the maven is 3.5.0

Website:

http://mirrors.tuna.tsinghua.edu.cn/apache/maven/maven-3/3.5.0/binaries/apache-maven-3.5.0-bin.tar.gz

tar -zxvf apache-maven-3.5.0-bin.tar.gz

mv

（七）Git install

① website

The office website is https://git-scm.com/download/

wget https://Github.com/Git/Git/archive/v2.3.0.tar.gz

Other links:https://www.kernel.org/pub/software/scm/git/

② Whether the git had been installed

rpm -qa|grep git

If the git is exist, delete it ,yum remove git

③ Install git

tar zxvf git-2.11.0.tar.gz

cd git-2.11.0

make prefix=/usr/local/git all

make prefix=/usr/local/git install

whereis git

④ Setting eviroment variable

Vi /etc/profile

#add export PATH=$PATH:/usr/local/git/bin

⑤ Test

git --version

```
[root@d10 git]#
[root@d10 git]# git --version
git version 2.9.4
[root@d10 git]#
```

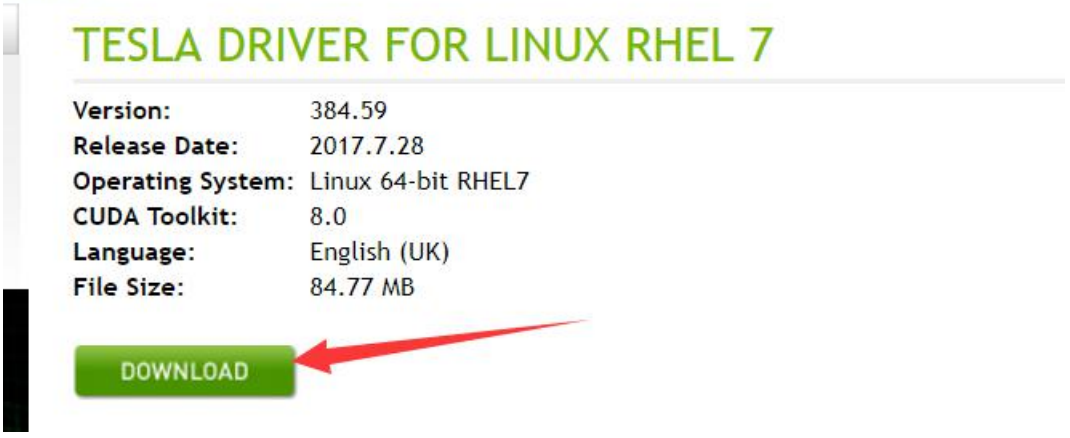
（八）NVIDIA Driver install

① Check graphics card type

lspci | grep VGA OR lspci | grep 3D

② Download driver

UX RHEL 7



③ Authorizing driver software

chmod +x NVIDIA-Linux-x86_64-340.32.run

④ systemctl stop gdm or init3

⑤ lsmod | grep nouveau

⑥ vim /etc/modprobe.d/blacklist.conf /*Nouveau joined to black list*/

blacklist nouveau

⑦ And then

/*bak initramfs*/

mv /boot/initramfs-$(uname -r).img /boot/initramfs-$(uname -r).img.bak

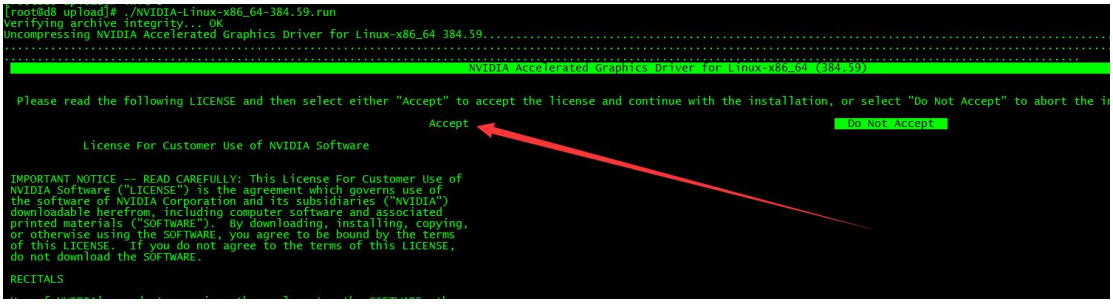
/*rebuild initramfs*/

dracut -v /boot/initramfs-$(uname -r).img $(uname -r)

systemctl isolate multi-user.target /*set (开机启动)到 3 层多用户模式*/

reboot or shutdown -r now

⑧ Running driver software



（九）CUDA install

Website :

https://developer.nvidia.com/compute/cuda/8.0/Prod2/local_installers/cuda_8.0.61_375.26_linux-run

Install progress:refering official manual operation

Test:

./NVIDIA_CUDA-8.0_Samples/bin/x86_64/linux/release/deviceQuery

① website

The office website is http://www.nvidia.com

wget https://developer.nvidia.com/compute/cuda/8.0/Prod2/local_installers/cuda_8.0.61_375.26_linux-run

② Install cuda

Graphics card is Matrox, does not support CUDA, stop installation

③ Set enviroment

Vi /etc/profile

```
#cuda
export CUDA_HOME=/usr/local/cuda-8.0
export PATH=$PATH:$CUDA_HOME/bin
export LD_LIBRARY_PATH=$CUDA_HOME/lib64:$CUDA_HOME/extras/CUPTI/lib64:$LD_LIBRARY_PATH
```

④ Test

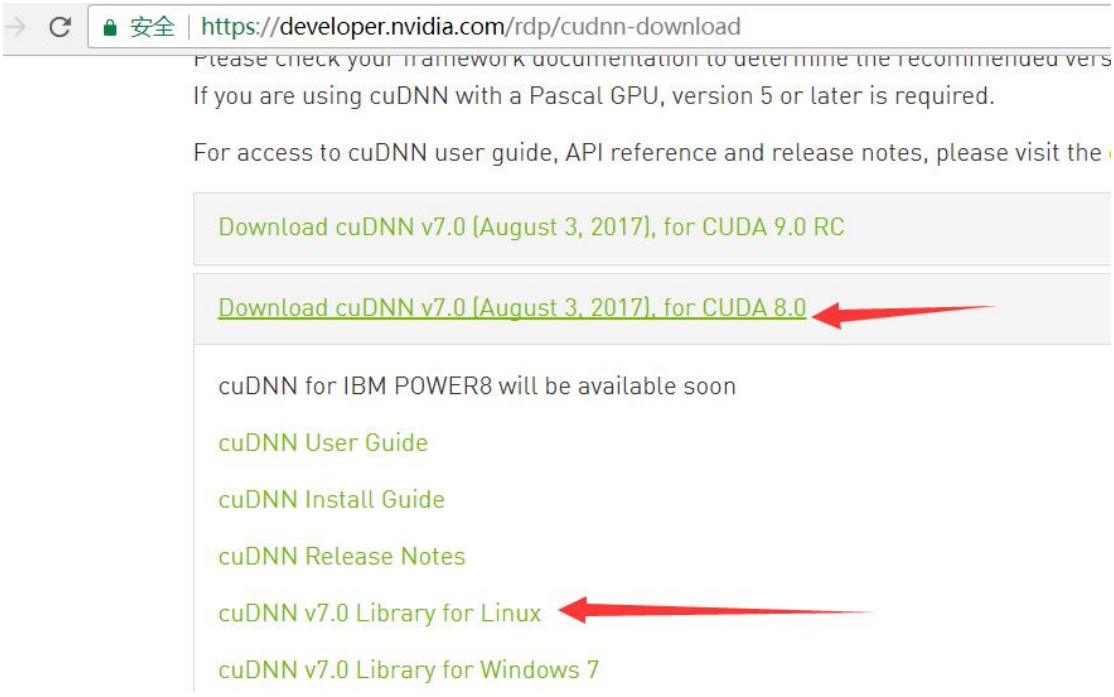
```
[root@d10 dl4j-general-benchmark]# nvcc -V
nvcc: NVIDIA (R) Cuda compiler driver
Copyright (c) 2005-2016 NVIDIA Corporation
Built on Tue_Jan_10_13:22:03_CST_2017
Cuda compilation tools, release 8.0, V8.0.61
[root@d10 dl4j-general-benchmark]# nvidia-smi
Wed Aug  9 18:08:04 2017
+-----------------------------------------------------------------------------+
| NVIDIA-SMI 384.59                 Driver Version: 384.59                    |
|-------------------------------+----------------------+----------------------+
| GPU  Name        Persistence-M| Bus-Id        Disp.A | Volatile Uncorr. ECC |
| Fan  Temp  Perf  Pwr:Usage/Cap|         Memory-Usage | GPU-Util  Compute M. |
|===============================+======================+======================|
|   0  Tesla K40m          Off  | 00000000:03:00.0 Off |                    0 |
| N/A   31C    P0    64W / 235W |      0MiB / 11439MiB |     83%      Default |
+-------------------------------+----------------------+----------------------+

+-----------------------------------------------------------------------------+
| Processes:                                                       GPU Memory |
|  GPU       PID   Type   Process name                             Usage      |
|=============================================================================|
|  No running processes found                                                 |
+-----------------------------------------------------------------------------+
```

（十）CUDNN install

TIP:in fact: we should download cudnnv6.0

① Download cudnn



② Install

tar -xvzf cudnn-8.0-linux-x64-v6.0.tgz

$cp -P cuda/include/cudnn.h /usr/local/cuda-8.0/include

$ cp -P cuda/lib64/libcudnn /usr/local/cuda-8.0/lib64*

*$ chmod a+r /usr/local/cuda-8.0/include/cudnn.h /usr/local/cuda-8.0/lib64/libcudnn**

```
[root@d10 upload]# tar -zxvf cudnn-8.0-linux-x64-v6.0.tgz
cuda/include/cudnn.h
cuda/lib64/libcudnn.so
cuda/lib64/libcudnn.so.6
cuda/lib64/libcudnn.so.6.0.21
cuda/lib64/libcudnn_static.a
[root@d10 upload]# cp -P cuda/include/cudnn.h /usr/local/cuda-8.0/include
[root@d10 upload]# cp -P cuda/lib64/libcudnn* /usr/local/cuda-8.0/lib64
[root@d10 upload]# chmod a+r /usr/local/cuda-8.0/include/cudnn.h /usr/local/cuda-8.0/lib64/libcudnn*
[root@d10 upload]#
```

（十一）LIBND4J install

```
git clone https://github.com/deeplearning4j/libnd4j.gitcd libnd4j

./buildnativeoperations.sh

# and/or when using GPU

# ./buildnativeoperations.sh -c cuda -cc INSERT_YOUR_DEVICE_ARCH_HERE

# i.e. if you have GTX 1070 device, use -cc 61

export LIBND4J_HOME=`pwd`

cd ..
```

（十二）ND4J install

git clone https://github.com/deeplearning4j/nd4j

```
mvn clean install -DskipTests -Dmaven.javadoc.skip=true -pl
'!:nd4j-cuda-7.5,!:nd4j-cuda-7.5-platform,!:nd4j-tests'## More recent 0.6.1 version of the above command

mvn clean install -DskipTests -Dmaven.javadoc.skip=true -pl
'!:nd4j-cuda-8.0,!:nd4j-cuda-8.0-platform,!:nd4j-tests'

# or when using GPU# mvn clean install -DskipTests -Dmaven.javadoc.skip=true -pl '!:nd4j-tests'
```

（十三）Datavec install

```
# build and install datavec

git clone https://github.com/deeplearning4j/datavec.gitcd datavecif [ "$SCALAV" == "" ]; then

  bash buildmultiplescalaversions.sh clean install -DskipTests -Dmaven.javadoc.skip=trueelse

  mvn clean install -DskipTests -Dmaven.javadoc.skip=true -Dscala.binary.version=$SCALAV
-Dscala.version=$SCALAfi
```

（十四）Dl4j install

```
# 构建并安装 deeplearning4j
git clone https://github.com/deeplearning4j/deeplearning4j.git
cd deeplearning4j
mvn clean install -DskipTests -Dmaven.javadoc.skip=true
# 或者为多个 Scala 版本进行交叉编译
# ./buildmultiplescalaversions.sh clean install -DskipTests -Dmaven.javadoc.skip=true
## 如果跳过了 CUDA，可能需要将
## -pl '!:deeplearning4j-cuda-8.0'
## 添加至 mvn clean install 命令，避免构建脚本寻找 cuda 库
```

（十五）All install by script code(9、10、11、12）

Website:

https://github.com/deeplearning4j/deeplearning4j/blob/master/build-dl4j-stack.sh

./build-dl4j-stack.sh //for cpu

./build-dl4j-stack.sh -c cuda //for gpu

Other documents:refering https://deeplearning4j.org/buildinglocally

（十六）Scala install

① Download spark

https://downloads.lightbend.com/scala/2.12.2/scala-2.12.2.tgz

② Install spark

Tar -zxvf scala-2.12.2.tgz

Mv scala-2.12.2 /usr

Vi /etc/profile

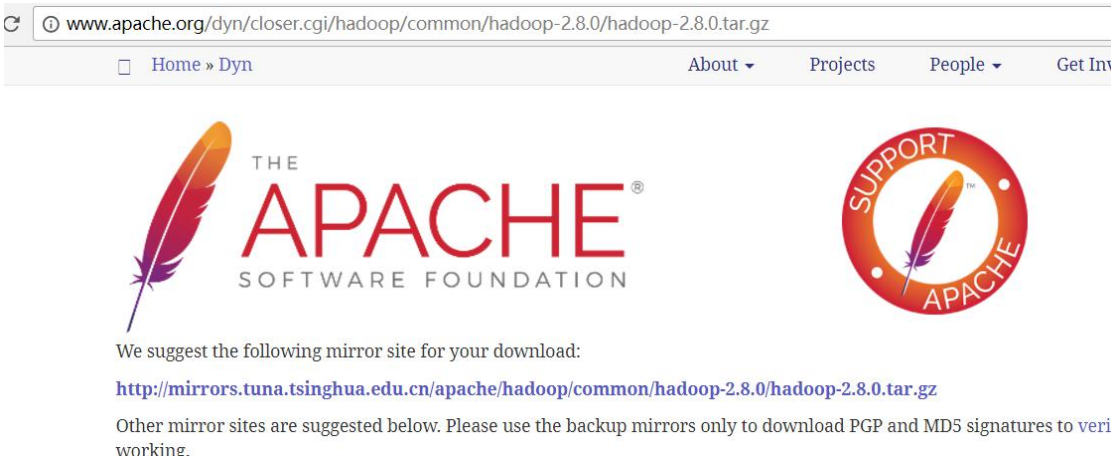
```
#scala
export SCALA_HOME=/usr/scala-2.12.2
export PATH=$PATH:$SCALA_HOME/bin
"/etc/profile" 108L, 2542C written
[root@d8 upload]# source /etc/profile
```

③ Test

```
[root@d8 upload]# scala -version
Scala code runner version 2.12.2 -- Copyright 2002-2017, LAMP/EPFL and Lightbend, Inc.
```

（十七）Hdfs install

① Download hadoop



Or

② Ready eviroment for hadoop

```
ssh-keygen -t rsa -P ''
```

```
[root@d8 .ssh]# ssh-keygen  -t   rsa   -P  ''
Generating public/private rsa key pair.
Enter file in which to save the key (/root/.ssh/id_rsa):
/root/.ssh/id_rsa already exists.
Overwrite (y/n)? y
Your identification has been saved in /root/.ssh/id_rsa.
Your public key has been saved in /root/.ssh/id_rsa.pub.
The key fingerprint is:
45:20:03:1a:fd:0b:70:7e:46:98:a4:fc:84:74:2d:67 root@d8.cecyun.com
The key's randomart image is:
+--[ RSA 2048]----+
|   oo+=o ...     |
|  o.** Eo .      |
|   =+.*    .     |
|    oo +   .     |
|      .+ .S      |
|         .       |
|                 |
|                 |
|                 |
+-----------------+
```

```
[root@d8 .ssh]# scp id_rsa.pub root@d10.cecyun.com:/home
Warning: Permanently added 'd10.cecyun.com' (ECDSA) to the list of known hosts.
Warning: the ECDSA host key for 'd10.cecyun.com' differs from the key for the IP address '172.16.192.20'
Offending key for IP in /root/.ssh/known_hosts:1
root@d10.cecyun.com's password:
id_rsa.pub
```

```
[root@d8 home]# cat id_rsa.pub >> ~/.ssh/authorized_keys
```

```
[root@d8 .ssh]# scp authorized_keys root@d10.cecyun.com:~/.ssh
```

```
[root@d10 home]# ssh d10.cecyun.com
Last login: Mon Aug  7 14:35:00 2017 from 172.16.0.
[root@d8 ~]# exit
logout
Connection to d10.cecyun.com closed.
[root@d10 home]#
```

③ Setting configure file,as follow

Core-site.xml,hdfs-site.xml,yarn-site.xml.mapred-site.xml， hadoop-env.sh

④ Runing commad:hadoop namenode -format

```
[root@d10 bin]# ./hadoop namenode -format
DEPRECATED: Use of this script to execute hdfs command is deprecated.
Instead use the hdfs command for it.

17/08/07 16:08:26 INFO namenode.NameNode: STARTUP_MSG:
/************************************************************
STARTUP_MSG: Starting NameNode
STARTUP_MSG:   user = root
STARTUP_MSG:   host = [redacted]
STARTUP_MSG:   args = [-format]
STARTUP_MSG:   version = 2.8.0
STARTUP_MSG:   classpath = /usr/local/hadoop-2.8.0/etc/hadoop:/usr/local/hadoop-2.8.0/sh
/lib/apacheds-i18n-2.0.0-M15.jar:/usr/local/hadoop-2.8.0/share/hadoop/common/lib/apacheds
sn1-api-1.0.0-M20.jar:/usr/local/hadoop-2.8.0/share/hadoop/common/lib/api-util-1.0.0-M20
.8.0/share/hadoop/common/lib/avro-1.7.4.jar:/usr/local/hadoop-2.8.0/share/hadoop/common/
ons-beanutils-core-1.8.0.jar:/usr/local/hadoop-2.8.0/share/hadoop/common/lib/commons-cli
local/hadoop-2.8.0/share/hadoop/common/lib/commons-collections-3.2.2.jar:/usr/local/hadoo
share/hadoop/common/lib/commons-configuration-1.6.jar:/usr/local/hadoop-2.8.0/share/hadoo
ib/commons-io-2.4.jar:/usr/local/hadoop-2.8.0/share/hadoop/common/lib/commons-lang-2.6.j
cal/hadoop-2.8.0/share/hadoop/common/lib/commons-math3-3.1.1.jar:/usr/local/hadoop-2.8.0/
mmon/lib/curator-client-2.7.1.jar:/usr/local/hadoop-2.8.0/share/hadoop/common/lib/curato
s-2.7.1.jar:/usr/local/hadoop-2.8.0/share/hadoop/common/lib/gson-2.2.4.jar:/usr/local/ha
```

⑤ Set eviroment :vi /etc/profile

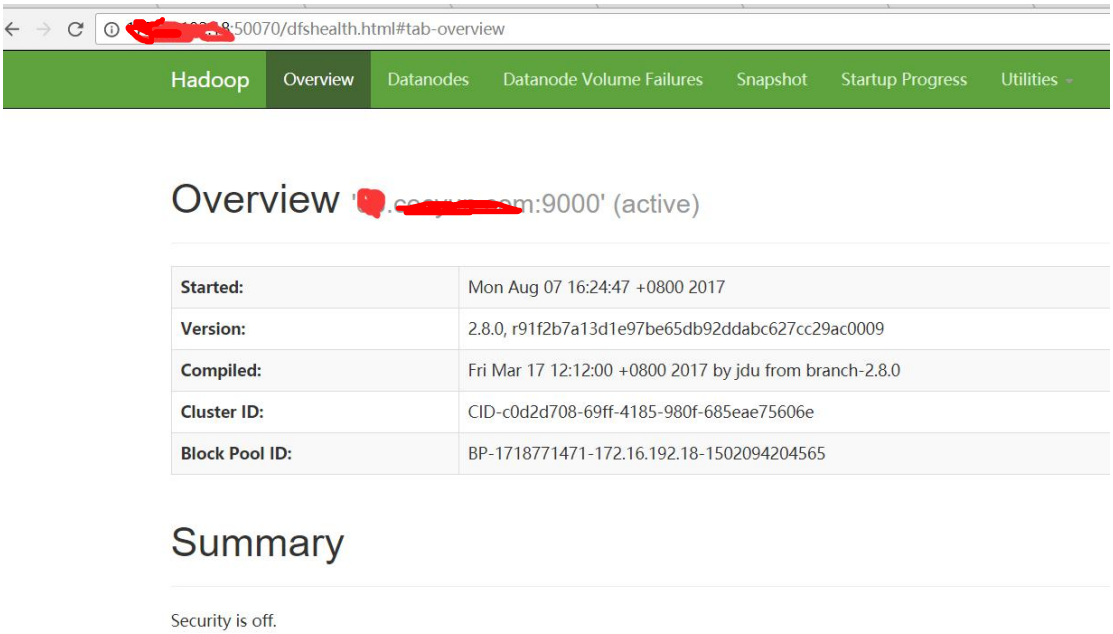
```
#hdfs
export HADOOP_HOME=/usr/local/hadoop-2.8.0
export PATH=$PATH:$HADOOP_HOME/bin:$HADOOP_HOME/sbin
export HADOOP_COMMON_LIB_NATIVE_DIR=$HADOOP_HOME/lib/native
export HADOOP_OPTS="-Djava.library.path=$HADOOP_HOME/lib"
```

⑥ Start hadoop

```
[root@d8 sbin]# jps
16820 Jps
```

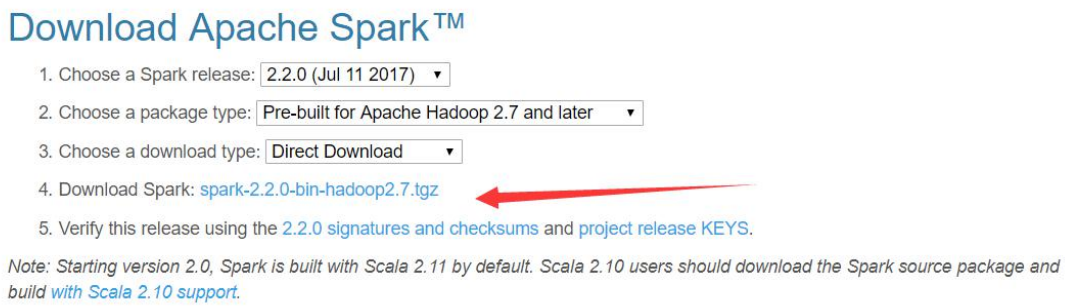
./sbin/start-all.sh OR .sbin/start-dfs.sh .sbin/start-yarn.sh

⑦ browsing web



（十八）Spark install

① Download spark



④ Install spark

```
[root@d10 upload]# tar -zxvf spark-2.2.0-bin-hadoop2.7.tgz
```

```
#spark path
export SPARK_HOME=/usr/spark-2.2.0-bin-hadoop2.7
export PATH=$PATH:$SPARK_HOME/bin
"/etc/profile" 98L, 2314C written
```

Modify spark-config.sh(adding words that's pointed by red arrow)

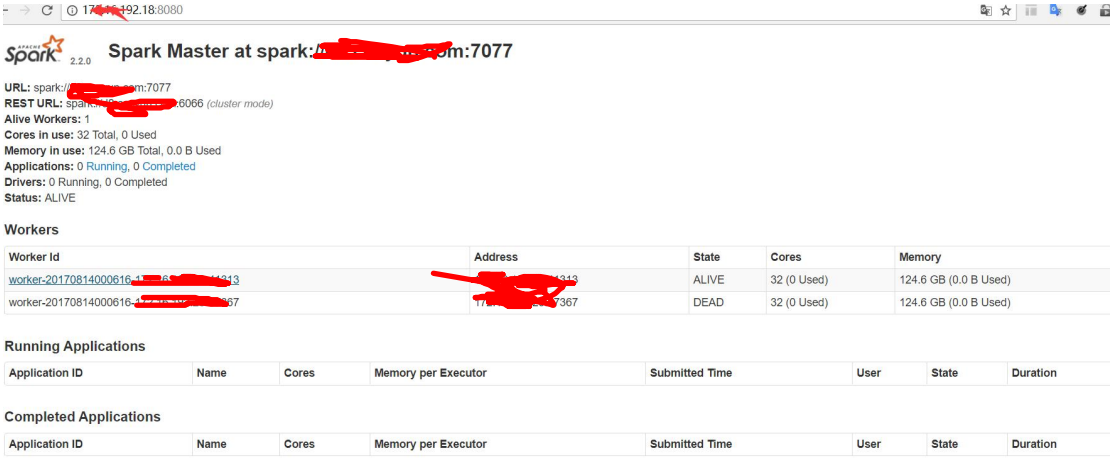
```
# Add the PySpark classes to the PYTHONPATH:
if [ -z "${PYSPARK_PYTHONPATH_SET}" ]; then
  export PYTHONPATH="${SPARK_HOME}/python:${PYTHONPATH}"
  export PYTHONPATH="${SPARK_HOME}/python/lib/py4j-0.10.4
  export PYSPARK_PYTHONPATH_SET=1
fi
export JAVA_HOME=/usr/jdk1.8.0_144
~
~
~
~
```

⑤ Test

```
[root@d8 upload]# ./spark-shell
-bash: ./spark-shell: No such file or directory
[root@d8 upload]# cd /usr/spark-2.2.0-bin-hadoop2.7/bin/
[root@d8 bin]# ./spark-shell
Using Spark's default log4j profile: org/apache/spark/log4j-defaults.properties
Setting default log level to "WARN".
To adjust logging level use sc.setLogLevel(newLevel). For SparkR, use setLogLevel(newLevel).
17/08/07 13:28:49 WARN NativeCodeLoader: Unable to load native-hadoop library for your platform... using builtin-ja
17/08/07 13:28:55 WARN ObjectStore: Version information not found in metastore. hive.metastore.schema.verification
17/08/07 13:28:55 WARN ObjectStore: Failed to get database default, returning NoSuchObjectException
17/08/07 13:28:56 WARN ObjectStore: Failed to get database global_temp, returning NoSuchObjectException
Spark context Web UI available at http://172.16.192.18:4040
Spark context available as 'sc' (master = local[*], app id = local-1502083730737).
Spark session available as 'spark'.
Welcome to
      ____              __
     / __/__  ___ _____/ /__
    _\ \/ _ \/ _ `/ __/  '_/
   /___/ .__/\_,_/_/ /_/\_\   version 2.2.0
      /_/

Using Scala version 2.11.8 (Java HotSpot(TM) 64-Bit Server VM, Java 1.8.0_144)
Type in expressions to have them evaluated.
Type :help for more information.

scala>
```



（十九）Test All

```
[INFO] Finished at: 2017-08-07T12:03:03+08:00
[INFO] Final Memory: 31M/1544M
[INFO] ------------------------------------------------------------------------
[root@d8 dl4j-examples]# mvn exec:java -Dexec:mainClass="org.deeplearning
[INFO] Scanning for projects...
[INFO]
[INFO] ------------------------------------------------------------------------
[INFO] Building DeepLearning4j Examples 0.9.0
[INFO] ------------------------------------------------------------------------
[INFO]
[INFO] --- exec-maven-plugin:1.4.0:java (default-cli) @ dl4j-examples ---
o.d.e.c.AnimalsClassification - Load data....
o.d.e.c.AnimalsClassification - Build model....
o.n.l.f.Nd4jBackend - Loaded [JCublasBackend] backend
o.n.n.NativeOpsHolder - Number of threads used for NativeOps: 32
o.n.l.a.o.e.DefaultOpExecutioner - Backend used: [CUDA]; OS: [Linux]
o.n.l.a.o.e.DefaultOpExecutioner - Cores: [32]; Memory: [26.7GB];
o.n.l.a.o.e.DefaultOpExecutioner - Blas vendor: [CUBLAS]
o.n.l.j.o.e.CudaExecutioner - Device name: [Tesla K40m]; CC: [3.5]; Total
o.d.n.m.MultiLayerNetwork - Starting MultiLayerNetwork with WorkspaceMode
o.d.e.c.AnimalsClassification - Train model....
o.n.n.Nd4jBlas - Number of threads used for BLAS: 0
o.d.d.i.MultipleEpochsIterator - Epoch 1, number of batches completed 3
o.d.o.l.ScoreIterationListener - Score at iteration 0 is 1.40197558831570
o.d.o.l.ScoreIterationListener - Score at iteration 1 is 1.56358739585684
o.d.o.l.ScoreIterationListener - Score at iteration 2 is 1.539839288619 81
o.d.d.i.MultipleEpochsIterator - Epoch 2, number of batches completed 3
o.d.o.l.ScoreIterationListener - Score at iteration 3 is 1.509722489157 01
o.d.o.l.ScoreIterationListener - Score at iteration 4 is 1.494261719315 29
```

```
[INFO] Building DeepLearning4j Examples 0.9.0
[INFO] ------------------------------------------------------------------------
[INFO]
[INFO] --- exec-maven-plugin:1.4.0:java (default-cli) @ dl4j-examples
o.d.e.c.AnimalsClassification - Load data....
o.d.e.c.AnimalsClassification - Build model....
o.n.l.f.Nd4jBackend - Loaded [CpuBackend] backend
o.n.n.NativeOpsHolder - Number of threads used for NativeOps: 8
o.n.n.Nd4jBlas - Number of threads used for BLAS: 8
o.n.l.a.o.e.DefaultOpExecutioner - Backend used: [CPU]; OS: [Linux]
o.n.l.a.o.e.DefaultOpExecutioner - Cores: [32]; Memory: [26.7GB];
o.n.l.a.o.e.DefaultOpExecutioner - Blas vendor: [OPENBLAS]
o.d.n.m.MultiLayerNetwork - Starting MultiLayerNetwork with WorkspaceM
o.d.e.c.AnimalsClassification - Train model....
o.d.d.i.MultipleEpochsIterator - Epoch 1, number of batches completed
```