



គ្រប់មីអូតិដពេង

Hortonworks Data Platform (HDP) 3 + Hue + FreeIPA

คำนำ

เอกสารคู่มือติดตั้ง HDP+Hue+FreeIPA ฉบับนี้ เกิดขึ้นจากการตั้งใจที่จะแบ่งปันวิธีการติดตั้งระบบ Hadoop ที่มีหน้าเว็บ UI พร้อมทั้งการพิสูจน์ตัวตนกับ LDAP ตามแบบที่ใช้งานกันทั่วไป แม้ HDP จะไม่ได้ออกรุ่นใหม่แล้ว แต่ของเก่าที่มีอยู่ยังคงใช้งานได้ ซึ่งมันใจว่ายังเป็นประโยชน์สำหรับผู้ที่ต้องการประมวลข้อมูลขนาดใหญ่ อีกเวลาก็ตาม ที่สำคัญต้องขอขอบคุณผู้ที่มีส่วนร่วมในการพัฒนาและสนับสนุนให้เกิดขึ้นมา ไม่ว่าจะเป็นผู้ที่เขียนโค้ด ออกแบบ UI หรือทดสอบตัวระบบ ทุกท่านเป็นส่วนสำคัญที่ทำให้เราสามารถนำเสนอบรรยากาศที่ดีให้กับผู้ใช้งานได้

เอกสารนี้ช่วยกันเขียนโดยทีมงานของคลัสเตอร์คิท มีคุณสมภพ สุวรรณประดิษฐ์ คุณกิตติรักษ์ ม่วงมิงสุข เป็นผู้เขียน และคุณนันท์นภัส ม่วงมิงสุข เป็นผู้ตรวจสอบ หวังเป็นอย่างยิ่งว่าจะเป็นประโยชน์ตามสมควร

พวกราคคลัสเตอร์คิท

สารบัญ

1	บทนำ.....	5
2	การเตรียมระบบก่อนการติดตั้ง.....	5
3	FreeIPA.....	6
3.1	ติดตั้ง FreeIPA server.....	6
3.2	ติดตั้ง FreeIPA Client.....	9
3.3	LDAP binding account.....	9
4	Apache Ambari.....	10
4.1	ติดตั้ง Ambari.....	10
4.2	ติดตั้ง MariaDB.....	10
4.3	ติดตั้ง MySQL JDBC.....	10
4.4	สร้าง Database สำหรับ Ambari และของเพื่อน.....	11
4.5	คอนฟิก ambari-server.....	12
5	ติดตั้ง Hortonwork Data Platform (HDP).....	14
5.1	ติดตั้ง HDP ผ่าน Apache ambari.....	14
6	การทำ High Availability.....	21
6.1	การทำ Namenode HA.....	21
6.2	Enable ResourceManager HA Wizard.....	26
7	Hue.....	29
7.1	ติดตั้ง Hue.....	29
7.2	ปรับแต่ง Hue configuration file (/root/hue/conf/hue.ini).....	30
7.3	ย้ายฐานข้อมูลมาใช้ MariaDB.....	32
7.4	กำหนด proxy user hadoop สำหรับ Hue.....	32
7.5	กำหนด proxy user oozie สำหรับ Hue.....	32
7.6	สร้าง home directory for hue user.....	33
7.7	ติดตั้ง hadoop-httpfs.....	33
8	Zeppelin authentication configuration.....	35
9	Ranger authentication with LDAP.....	37
9.1	Import LDAP CA cert.....	37
9.2	Config Ranger authen LDAP.....	37

10 การเพิ่มบัญชีผู้ใช้และการใช้งาน.....	40
11 การใช้งาน.....	43
11.1 บริการ ssh.....	43
11.2 Hue บริการหน้าเว็บสำหรับ Hadoop Ecosystems.....	43

การติดตั้งระบบ Hortonworks HDP3+Hue+FreeIPA

1 บทนำ

การติดตั้ง Apache Hadoop ด้วย Hortonworks Data Platform (HDP) 3 ตามเอกสารฉบับนี้นั้น จะมีการติดตั้งเว็บสำหรับเข้าใช้งาน Hadoop ที่ชื่อว่า Hue (<https://gethue.com/>) และกำหนดให้มีการยืนยันตัวตน กับ LDAP เซิร์ฟเวอร์ ในที่นี้จะใช้ซอฟต์แวร์ FreeIPA ด้วยองค์ประกอบเหล่านี้จะได้ระบบ Hadoop ที่พร้อมใช้งานได้เป็นอย่างดี มีการพิสูจน์ตัวตนทั้งในส่วนของ SSH, Hue และ Zeppelin

สำหรับเครื่องที่จะติดตั้ง HDP นั้น เราใช้ระบบปฏิบัติการ CentOS-7 ส่วนเครื่องที่ติดตั้ง FreeIPA เป็น Rocky Linux 9

2 การเตรียมระบบก่อนการติดตั้ง

- 1) เตรียม repository ของ HDP ในเอกสารนี้ใช้ repo ของ <http://makeopensourcegreatagain.com/> ควรจะ Mirror มาทำเป็น local repo
- 2) สร้าง root ssh key โดยจากเครื่อง ambari server ต้องสามารถ ssh ไปยังทุกเครื่องในระบบได้โดยไม่ติดต้องระบุรหัสผ่าน
- 3) ปิดการทำงานของ SELinux กำหนดตัวแปร SELINUX ในไฟล์ /etc/selinux/config ให้เป็น permissive หรือ Disable
- 4) กำหนดค่าโดเมนเนมของทุกเครื่องใน DNS เพื่อให้สามารถติดต่อไปยังทุกเครื่องได้โดยใช้ชื่อเครื่อง
- 5) ตั้งชื่อเครื่องทุกเครื่องในคลัสเตอร์ให้เป็น fully qualified domain name ตามที่กำหนดใน DNS เช่น edge.example.com `hostnamectl set-hostname edge.example.com`
- 6) ตัวอย่างชื่อเครื่องในเอกสารนี้

Hostname	roles
ipa.example.com	LDAP, Kerberos and DNS server
edge.example.com	Ambari, Hue, DB
master1.example.com	Namenode, Yarn resourcemanager
master2.example.com	Namenode, Yarn resourcemanager
worker1.example.com	Data node, Hbase Region server
worker2.example.com	Data node, Hbase Region server
worker3.example.com	Data node, Hbase Region server

* การตั้งโดเมนที่มีคนอื่นเป็นเจ้าของ ไม่สามารถเอาไปกับโดเมนจริง ๆ ได้ ในเอกสารนี้เราใช้ Private IP ทำเป็นตัวอย่างใช้งานภายใน

3 FreeIPA

FreeIPA เป็น LDAP และ Kerberos เซิร์ฟเวอร์ของค่าย Redhat ที่แจกจ่ายให้ใช้งานเสรี หากจะใช้งาน LDAP จำเป็นต้องตั้งชื่อเครื่องให้เป็น Fully Qualified Domain Name (FQDN) และหากจะใช้งาน Kerberos จำเป็นจะต้องกำหนดชื่อโดเมนผ่าน DNS

3.1 ติดตั้ง FreeIPA server

ในระบบนี้ใช้งานเพียงส่วนของ LDAP ไม่ได้รวมถึง Kerberos (หากจะใช้งาน Kerberos ทุกเครื่องต้องมี hostname ใน DNS สามารถติดตั้ง ipa-server-dns เพิ่มได้) และติดตั้งบนระบบปฏิบัติการ Rocky9 การติดตั้ง FreeIPA ทำได้โดยการใช้คำสั่งต่อไปนี้

```
yum install -y ipa-server
```

หลังจากนั้นรันคำสั่งต่อไปนี้เพื่อคอนฟิก

```
ipa-server-install
```

คำสั่งข้างต้นจะไปคอนฟิกค่าต่าง ๆ ให้มีค่าที่เราต้องกรอกหลัก ๆ คือ รหัสผ่านของ Directory Manager และ IPA Admin ผลการรันสคริปต์เป็นดังต่อไปนี้

```
[root@freeipa ~]# ipa-server-install
```

```
The log file for this installation can be found in /var/log/ipaserver-install.log
```

```
=====
This program will set up the IPA Server.
```

```
Version 4.10.2
```

```
This includes:
```

- * Configure a stand-alone CA (dogtag) for certificate management
- * Configure the NTP client (chronyd)
- * Create and configure an instance of Directory Server
- * Create and configure a Kerberos Key Distribution Center (KDC)
- * Configure Apache (httpd)
- * Configure SID generation
- * Configure the KDC to enable PKINIT

```
To accept the default shown in brackets, press the Enter key.
```

```
Do you want to configure integrated DNS (BIND)? [no]:
```

```
Enter the fully qualified domain name of the computer  
on which you're setting up server software. Using the form  
<hostname>.<domainname>
```

```
Example: master.example.com
```

Server host name [ipa.example.com]:

The domain name has been determined based on the host name.

Please confirm the domain name [example.com]:

The kerberos protocol requires a Realm name to be defined.
This is typically the domain name converted to uppercase.

Please provide a realm name [EXAMPLE.COM]:

Certain directory server operations require an administrative user.
This user is referred to as the Directory Manager and has full access
to the Directory for system management tasks and will be added to the
instance of directory server created for IPA.
The password must be at least 8 characters long.

Directory Manager password:

Password (confirm):

The IPA server requires an administrative user, named 'admin'.
This user is a regular system account used for IPA server administration.

IPA admin password:

Password (confirm):

Trust is configured but no NetBIOS domain name found, setting it now.
Enter the NetBIOS name for the IPA domain.
Only up to 15 uppercase ASCII letters, digits and dashes are allowed.
Example: EXAMPLE.

NetBIOS domain name [EXAMPLE]:

Do you want to configure chrony with NTP server or pool address? [no]:

The IPA Master Server will be configured with:

Hostname: ipa.example.com

IP address(es): 192.168.2.10

Domain name: example.com

Realm name: EXAMPLE.COM

The CA will be configured with:

Subject DN: CN=Certificate Authority,O=EXAMPLE.COM

Subject base: O=EXAMPLE.COM

Chaining: self-signed

Continue to configure the system with these values? [no]: yes

The following operations may take some minutes to complete.

Please wait until the prompt is returned.

...
...
...

Configuring client side components



This program will set up IPA client.
Version 4.10.2

```
Using existing certificate '/etc/ipa/ca.crt'.
Client hostname: ipa.example.com
Realm: EXAMPLE.COM
DNS Domain: example.com
IPA Server: ipa.example.com
BaseDN: dc=example,dc=com

Configured /etc/sssd/sssd.conf
Systemwide CA database updated.
Adding SSH public key from /etc/ssh/ssh_host_ed25519_key.pub
Adding SSH public key from /etc/ssh/ssh_host_ecdsa_key.pub
Adding SSH public key from /etc/ssh/ssh_host_rsa_key.pub
Could not update DNS SSHFP records.
SSSD enabled
Configured /etc/openldap/ldap.conf
Configured /etc/ssh/ssh_config
Configured /etc/ssh/sshd_config.d/04-ipa.conf
Configuring example.com as NIS domain.
Client configuration complete.
The ipa-client-install command was successful
```

```
Please add records in this file to your DNS system: /tmp/ipa.system.records.h8lkr016.db
=====
Setup complete
```

Next steps:

1. You must make sure these network ports are open:
 - TCP Ports:
 - * 80, 443: HTTP/HTTPS
 - * 389, 636: LDAP/LDAPS
 - * 88, 464: kerberos
 - UDP Ports:
 - * 88, 464: kerberos
 - * 123: ntp
2. You can now obtain a kerberos ticket using the command: 'kinit admin'
This ticket will allow you to use the IPA tools (e.g., ipa user-add)
and the web user interface.

Be sure to back up the CA certificates stored in /root/cacert.p12
These files are required to create replicas. The password for these
files is the Directory Manager password
The ipa-server-install command was successful

เป็นอันเสร็จการติดตั้ง FreeIPA Server

3.2 ติดตั้ง FreeIPA Client

FreeIPA Client จะประกอบไปด้วยส่วนของ LDAP Client และ Kerberos Client ซึ่งระบบไปคอนฟิกให้ ต้องติดตั้งที่ทุกเครื่องในระบบ สามารถติดตั้งด้วยคำสั่งต่อไปนี้

```
yum install -y ipa-client
```

แล้วรันคำสั่งต่อไปนี้เพื่อคอนฟิก (mkhomedir กำหนดให้สร้าง Home directory ให้ผู้ใช้เมื่อ login)

```
ipa-client-install --domain=example.com --server=ipa.example.com --mkhomedir
```

ตอบคำถามและระบุค่าดังต่อไปนี้

```
Proceed with fixed values and no DNS discovery? [no]: yes  
Continue to configure the system with these values? [no]: yes
```

```
User authorized to enroll computers: admin  
Password for admin@EXAMPLE.COM:
```

3.3 LDAP binding account

ในการณ์ที่แอปพลิเคชันต้องการ Binding mayoría LDAP server ในระบบ Hadoop นี้ Ranger ต้องการ Binding บางคนอาจใช้บัญชีผู้ใช้ admin ของ LDAP แต่ในทางปฏิบัติเพื่อความปลอดภัย เราชาระสร้าง user สำหรับ Binding ขึ้นมา ที่มีสิทธิ์แบบ Read only ในระบบนี้ทำที่เครื่อง edge ทำได้ตามขั้นตอนต่อไปนี้

1) ติดตั้ง openldap-clients

```
yum install -y openldap-clients
```

2) สร้าง system account for ldap binding ให้สั่ง

```
ldapmodify -x -D 'cn=Directory Manager' -W  
dn: uid=system,cn=sysaccounts,cn=etc,dc=example,dc=com  
changetype: add  
objectclass: account  
objectclass: simpleSecurityObject  
uid: system  
userPassword: password  
passwordExpirationTime: 20380119031407Z  
nsIdleTimeout: 0  
<blank line>  
^D
```

หมายเหตุ ส่องบรรทัดท้าย <blank line> ให้เว้นบรรทัดว่าง แล้วจึงกด Ctrl+D อ้างอิง

<https://www.freeipa.org/page/HowTo/LDAP>

4 Apache Ambari

4.1 ติดตั้ง Ambari

4.1.1 ติดตั้ง ambari repository โดยดาวน์โหลดไฟล์ ambari.repo จาก repository ที่เตรียมไว้ ไปวางไว้ที่ /etc/yum.repos.d/

```
# wget -nv http://makeopensourcegreatagain.com/rpms/mosga.repo -O /etc/yum.repos.d/ambari.repo
```

4.1.2 แก้ไขไฟล์ ambari.repo โดยแก้จาก https เป็น http

4.1.3 ติดตั้ง Apache Ambari

```
# yum install -y ambari-server
```

4.1.4 ถ้าทำ Local repo ก็ให้โหลด JDK และ JCE มาวางใน Local repo ด้วย แล้วแก้ไขลิงค์ดาวน์โหลด JDK ในไฟล์ /etc/ambari-server/conf/ambari.properties จาก hortonworks repo เป็น local repo ของเรา เช่น

```
# sed -i 's/public-repo-1.hortonworks.com/192.168.2.10\!/hortonworks-repos/g'
```

4.2 ติดตั้ง MariaDB

MariaDB ทำหน้าที่เป็นฐานข้อมูลหลังบ้านให้กับ Ambari Server ในการเอกสารนี้เลือกใช้ MariaDB server ที่มากับ CentOS-7 ติดตั้งไว้ที่เครื่อง edge

```
yum install -y mariadb-server
```

กำหนดให้ MariaDB เริ่มการทำงานทุกครั้งที่เปิดเครื่อง และเริ่มการทำงานของ MariaDB ทันที

```
systemctl enable --now mariadb
```

เพื่อความปลอดภัยให้รันคำสั่งต่อไปนี้ เพื่อกำหนดรหัสผ่านให้ root ลบสิ่งที่ไม่จำเป็น และทำให้ระบบมั่นคงขึ้น

```
/usr/bin/mysql_secure_installation
```

4.3 ติดตั้ง MySQL JDBC

ติดตั้งที่เครื่อง edge

```
# wget https://dev.mysql.com/get/Downloads/Connector-J/mysql-connector-java-5.1.46.tar.gz
# tar xvf mysql-connector-java-5.1.46.tar.gz
# mkdir -p /usr/share/java
# cp mysql-connector-java-5.1.46/mysql-connector-java-5.1.46.jar /usr/share/java
# ln -sf /usr/share/java/mysql-connector-java-5.1.46.jar /usr/share/java/mysql-connector-java.jar
```

4.4 สร้าง Database สำหรับ Ambari และผองเพื่อน

อ้างอิง https://docs.hortonworks.com/HDPDocuments/Ambari-2.7.4.0/administering-ambari/content/amb_example_install_mysql-mariadb_for_multiple_components.html

```
# login to mariadb
mysql -u root -p

# create database
create database ambari;
grant all privileges on ambari.* to 'ambari'@'localhost' identified by 'password';
grant all privileges on ambari.* to 'ambari'@'%' identified by 'password';

create database hive DEFAULT CHARACTER SET utf8;
grant all privileges on hive.* to 'hive'@'localhost' identified by 'password';
grant all privileges on hive.* to 'hive'@'%' identified by 'password';

create database ranger DEFAULT CHARACTER SET utf8;
grant all privileges on ranger.* to 'rangeradmin'@'localhost' identified by 'password';
grant all privileges on ranger.* to 'rangeradmin'@'%' identified by 'password';

create database rangerkms DEFAULT CHARACTER SET utf8;
grant all privileges on rangerkms.* to 'rangerkms'@'localhost' identified by 'password';
grant all privileges on rangerkms.* to 'rangerkms'@'%' identified by 'password';

create database oozie DEFAULT CHARACTER SET utf8;
grant all privileges on oozie.* to 'oozie'@'localhost' identified by 'password';
grant all privileges on oozie.* to 'oozie'@'%' identified by 'password';

create database superset DEFAULT CHARACTER SET utf8;
grant all privileges on superset.* to 'superset'@'localhost' identified by 'password';
grant all privileges on superset.* to 'superset'@'%' identified by 'password';

create database druid DEFAULT CHARACTER SET utf8;
grant all privileges on druid.* to 'druid'@'localhost' identified by 'password';
grant all privileges on druid.* to 'druid'@'%' identified by 'password';

create database hue DEFAULT CHARACTER SET utf8 DEFAULT COLLATE utf8_general_ci;
grant all ON hue.* TO 'hue'@'%' IDENTIFIED BY 'password';
grant all on hue.* to 'hue'@'localhost' identified by 'password';
```

รันสคริปต์เพื่อสร้างตารางในฐานข้อมูล ambari

```
mysql -u root -p
use ambari;
source /var/lib/ambari-server/resources/Ambari-DDL-MySQL-CREATE.sql
```

4.5 ค่อนฟิก ambari-server

สั่งคำสั่ง ambari-server setup เพื่อเริ่ม setup ambari โดยระบบจะตรวจสอบ SELinux, ติดตั้ง JDK, config Database ตามลำดับ

```
# ambari-server setup
Using python /usr/bin/python
Setup ambari-server
Checking SELinux...
SELinux status is 'enabled'
SELinux mode is 'permissive'
WARNING: SELinux is set to 'permissive' mode and temporarily disabled.
OK to continue [y/n] (y)?
Customize user account for ambari-server daemon [y/n] (n)? y
Enter user account for ambari-server daemon (root):
Adjusting ambari-server permissions and ownership...
Checking firewall status...
WARNING: iptables is running. Confirm the necessary Ambari ports are accessible. Refer
to the Ambari documentation for more details on ports.
OK to continue [y/n] (y)?
Checking JDK...
[1] Oracle JDK 1.8 + Java Cryptography Extension (JCE) Policy Files 8
[2] Custom JDK
=====
Enter choice (1):
To download the Oracle JDK and the Java Cryptography Extension (JCE) Policy Files you
must accept the license terms found at javase/terms/license/index.html and not
accepting will cancel the Ambari Server setup and you must install the JDK and JCE
files manually.
Do you accept the Oracle Binary Code License Agreement [y/n] (y)?
Downloading JDK from -x64.tar.gz to /var/lib/ambari-server/resources/jdk-8u112-linux-
x64.tar.gz
jdk-8u112-linux-x64.tar.gz... 100% (174.7 MB of 174.7 MB)
Successfully downloaded JDK distribution to /var/lib/ambari-server/resources/jdk-8u112-
linux-x64.tar.gz
Installing JDK to /usr/jdk64/
Successfully installed JDK to /usr/jdk64/
Downloading JCE Policy archive from jce_policy-8.zip to
/var/lib/ambari-server/resources/jce_policy-8.zip
Successfully downloaded JCE Policy archive to
/var/lib/ambari-server/resources/jce_policy-8.zip
Installing JCE policy...
Check JDK version for Ambari Server...
JDK version found: 8
Minimum JDK version is 8 for Ambari. Skipping to setup different JDK for Ambari Server.
Checking GPL software agreement...
GPL License for LZ0: https://www.gnu.org/licenses/old-licenses/gpl-2.0.en.html
Enable Ambari Server to download and install GPL Licensed LZ0 packages [y/n] (n)?
Completing setup...
```

```

Configuring database...
Enter advanced database configuration [y/n] (n)? y
Configuring database...
=====
Choose one of the following options:
[1] - PostgreSQL (Embedded)
[2] - Oracle
[3] - MySQL / MariaDB
[4] - PostgreSQL
[5] - Microsoft SQL Server (Tech Preview)
[6] - SQL Anywhere
[7] - BDB
=====
Enter choice (1): 3
Hostname (localhost):
Port (3306):
Database name (ambari):
Username (ambari):
Enter Database Password (bigdata):

Re-enter password:

Configuring ambari database...
Enter full path to custom jdbc driver: /usr/share/java/mysql-connector-java-5.1.46.jar
Configuring remote database connection properties...
WARNING: Before starting Ambari Server, you must run the following DDL against the database
to create the schema: /var/lib/ambari-server/resources/Ambari-DDL-MySQL-CREATE.sql
Proceed with configuring remote database connection properties [y/n] (y)?
Extracting system views...
ambari-admin-2.7.4.0.118.jar
....
Ambari repo file contains latest json url hdp_urlinfo.json, updating stacks repoinfos with
it...
Adjusting ambari-server permissions and ownership...
Ambari Server 'setup' completed successfully.

```

เริ่มการทำงานของ ambari-server

```

# ambari-server setup --jdbc-db=mysql --jdbc-driver=/usr/share/java/mysql-connector-java-
5.1.46.jar
# chkconfig ambari-server on
# service ambari-server start
Using python /usr/bin/python
Starting ambari-server
Ambari Server running with administrator privileges.
Organizing resource files at /var/lib/ambari-server/resources...
Ambari database consistency check started...
Server PID at: /var/run/ambari-server/ambari-server.pid
Server out at: /var/log/ambari-server/ambari-server.out
Server log at: /var/log/ambari-server/ambari-server.log
Waiting for server start.....
Server started listening on 8080

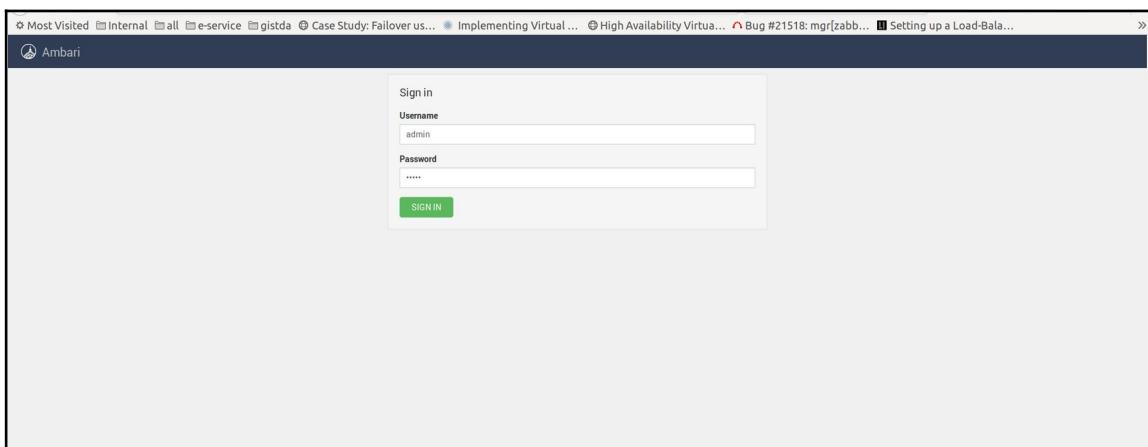
DB configs consistency check: no errors and warnings were found.
Ambari Server 'start' completed successfully.

```

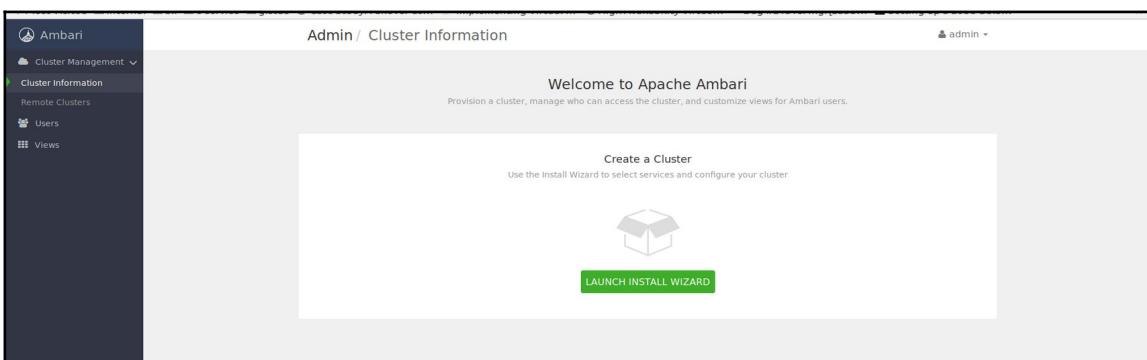
5 ติดตั้ง Hortonwork Data Platform (HDP)

5.1 ติดตั้ง HDP ผ่าน Apache ambari

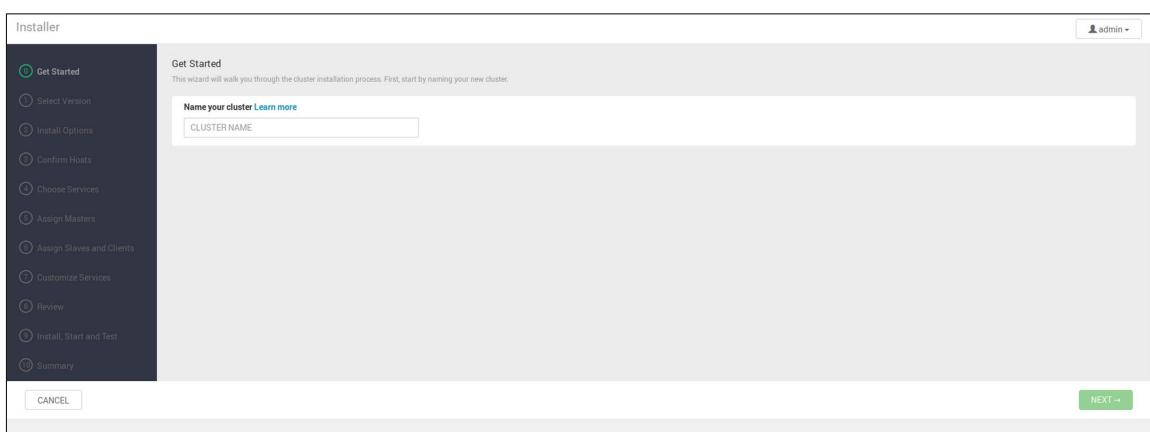
5.1.1 เปิด browser ไปที่ <http://<edge IPAddress>:8080> จะพบหน้า login เข้า Ambari Web UI โดยใช้ user: admin password: admin



5.1.2 คลิกปุ่ม LAUNCH INSTALL WIZARD เพื่อเริ่มการติดตั้ง



5.1.3 0. Get Started: ตั้งชื่อระบบคลัสเตอร์ที่จะสร้าง



5.1.4 1: Select Version

หัวข้อ Repositories ให้เลือก “Use Local Repository” และ Base URL ให้ซึ่ปัจจุบัน local repository ของ redhat7 โดยระบุ URL ตามลิงค์ด้านล่าง ส่วน Distros อื่น ๆ ให้ทำการ Remove ออกจากนั้นคลิก NEXT

HDP-3.1: <http://makeopensourcegreatagain.com/rpms/>

HDP-UTILS: <http://makeopensourcegreatagain.com/rpms/HDP-UTILS-1.1.0.22/>

5.1.5 2 : Install Options

Target Hosts : ใส่ชื่อ FQDN เครื่องที่จะติดตั้ง Hadoop ทั้งหมด ดังนี้

- edge.example.com
- master1.example.com
- master2.example.com
- worker1.example.com
- worker2.example.com
- worker3.example.com

Host Registration Information : ใส่ private key เพื่อให้ Ambari สามารถรีโมทไปติดตั้ง Agent เครื่องในระบบได้โดยไม่ติด password



5.1.6 3 : Confirm Hosts

จะทำการติดตั้ง Agent ไปยังเครื่องต่าง ๆ ในระบบ

Ambari - Cluster Install Wizard - Mozilla Firefox
Openlandscape Cloud | 203.150.243.106/HDP/centos | Ambari - Cluster Install | +
Installer
Confirm Hosts
Registering your hosts.
Please confirm the host list and remove any hosts that you do not want to include in the cluster.

Host	Progress	Status	Action
edge.example.com	<div style="width: 100%; background-color: #2e6b2e; height: 10px;"></div>	Success	
master1.example.com	<div style="width: 100%; background-color: #2e6b2e; height: 10px;"></div>	Success	
master2.example.com	<div style="width: 100%; background-color: #2e6b2e; height: 10px;"></div>	Success	
worker1.example.com	<div style="width: 100%; background-color: #2e6b2e; height: 10px;"></div>	Success	
worker2.example.com	<div style="width: 100%; background-color: #2e6b2e; height: 10px;"></div>	Success	
worker3.example.com	<div style="width: 100%; background-color: #2e6b2e; height: 10px;"></div>	Success	

Show: All (6) | Installing (0) | Registering (0) | Success (6) | Fail (0)
Items per page: 25 | 1 - 6 of 6

5.1.7 4 : Choose services

เลือกบริการต่าง ๆ ที่จะติดตั้งในระบบ Hadoop ในเอกสารนี้จะเลือกติดตั้ง YARN-MapReduce2, Tez, Hive, Hbase, Pig, Sqoop, Oozie, ZooKeeper, Infra Solr, Ambari Metrics, Atlas, Kafka, Ranger, SmartSense, Spark2 และ Zeppelin Notebook จากนั้นคลิก NEXT

Installer
Choose File System
Choose which file systems you want to install on your cluster.
Service Version Description
HDFS 3.1.1 Apache Hadoop Distributed File System

Choose Services
Choose which services you want to install on your cluster.

Service	Version	Description
YARN + MapReduce2	3.1.1	Apache Hadoop MapReduce (YARN)
Tez	0.9.1	Tez is the next generation Hadoop Query Processing framework written on top of YARN.
Hive	3.1.0	Data warehouse system for ad-hoc queries & analysis of large datasets and table & storage management service.
Hbase	2.0.2	Non-relational distributed database and centralized service for configuration management & synchronization.
Pig	0.16.0	Scripting platform for analyzing large datasets.
Sqoop	1.4.7	Tool for transferring bulk data between Apache Hadoop and structured data stores such as relational databases.
Oozie	4.3.1	System for workflow coordination and execution of Apache Hadoop jobs. This also includes the installation of the optional Oozie Web Console which relies on and will install the ExtJS Library.
ZooKeeper	3.4.6	Centralized service which provides highly reliable distributed coordination.
Storm	1.2.1	Apache Hadoop Streaming framework.
Accumulo	1.7.0	Robust, scalable, high performance distributed key/value store.
Infra Solr	0.1.0	Code shared service used by Ambari managed components.
Ambari Metrics	0.1.0	A system for metrics collection that provides storage and retrieval capability for metrics collected from the cluster.
Atlas	1.1.0	Atlas Metadata and Governance platform.
Kafka	2.0.0	A high-throughput distributed messaging system.
Knox	1.0.0	Provides a single point of authentication and access for Apache Hadoop services in a cluster.
Log Search	0.3.0	Log aggregation, analysis, and visualization for Ambari managed services. This service is Technical Preview.
Ranger	1.2.0	Comprehensive security for Hadoop.
Ranger KMS	1.2.0	Key Management Server.
SmartSense	1.5.1.2.7.4.0-118	SmartSense - Hadoop SmartSense Tool (HST) helps quickly gather configuration, metrics, logs from common HDP services that aids to quickly troubleshoot support cases and receive cluster-specific recommendations.
Spark2	2.3.2	Apache Spark 2.3 is a fast and general engine for large-scale data processing.
Zeppelin Notebook	0.8.0	A web-based notebook that enables interactive data analytics. It enables you to make beautiful data-driven, interactive and collaborative documents with SQL, Scala and more.
Druid	0.12.1	A fast column-oriented distributed data store.
Superset	0.23.0	Superset is a data exploration platform designed to be visual, intuitive and interactive. This service is Technical Preview.

BACK CANCEL NEXT

5.1.8 5 : Assign Masters

เลือก Host สำหรับบริการต่าง ๆ จะติดตั้งอยู่ในเครื่องใด

Assign Masters

Assign master components to hosts you want to run them on.

SNameNode:	master1.example.com (3.7 GB, 2 cores)
NameNode:	master2.example.com (3.7 GB, 2 cores)
Timeline Service V1.5:	edge.example.com (15.5 GB, 4 cores)
ResourceManager:	master1.example.com (3.7 GB, 2 cores)
Timeline Service V2.0 Reader:	edge.example.com (15.5 GB, 4 cores)
YARN Registry DNS:	edge.example.com (15.5 GB, 4 cores)
History Server:	edge.example.com (15.5 GB, 4 cores)
Hive Metastore:	edge.example.com (15.5 GB, 4 cores)
HiveServer2:	edge.example.com (15.5 GB, 4 cores)
HBase Master:	master1.example.com (3.7 GB, 2 cores)
Oozie Server:	edge.example.com (15.5 GB, 4 cores)
ZooKeeper Server:	master1.example.com (3.7 GB, 2 cores)
ZooKeeper Server:	edge.example.com (15.5 GB, 4 cores)
ZooKeeper Server:	master2.example.com (3.7 GB, 2 cores)
Grafana:	edge.example.com (15.5 GB, 4 cores)
Metrics Collector:	edge.example.com (15.5 GB, 4 cores)
Activity Explorer:	edge.example.com (15.5 GB, 4 cores)
HST Server:	edge.example.com (15.5 GB, 4 cores)
Activity Analyzer:	edge.example.com (15.5 GB, 4 cores)
Spark2 History Server:	edge.example.com (15.5 GB, 4 cores)

edge-example.com (15.5 GB, 4 cores)

- Timeline Service V1.5
- Timeline Service V2.0 Reader
- YARN Registry DNS
- History Server
- Hive Metastore
- HiveServer2
- Oozie Server
- ZooKeeper Server
- Grafana
- Metrics Collector
- Activity Explorer
- HST Server
- Activity Analyzer
- Spark2 History Server

master1.example.com (3.7 GB, 2 cores)

- SNameNode
- ResourceManager
- HBase Master
- ZooKeeper Server

master2.example.com (3.7 GB, 2 cores)

- NameNode
- ZooKeeper Server

BACK CANCEL NEXT →

5.1.9 6: Assign Slaves and Clients

เลือก Host สำหรับเป็น Datanode และ Client ตามภาพด้านล่าง จากนั้นคลิก NEXT

Installer

Assign Slaves and Clients

Assign slave and client components to hosts you want to run them on.
Hosts that are assigned master components are shown with ●.
"Client" will install HDFS Client, YARN Client, MapReduce2 Client, Tez Client, Hive Client, HBase Client, Pig Client, Sqoop Client, Oozie Client, ZooKeeper Client and Spark2 Client.

Host	all none	all none	all none	all none	all none	all none	all none	all none	all none
edge.example.com●	<input type="checkbox"/> DataNode	<input checked="" type="checkbox"/> NFSGateway	<input type="checkbox"/> NodeManager	<input type="checkbox"/> RegionServer	<input checked="" type="checkbox"/> Phoenix Query Server	<input checked="" type="checkbox"/> Livy for Spark2 Server	<input checked="" type="checkbox"/> Spark2 Thrift Server	<input checked="" type="checkbox"/> Client	
master1.example.com●	<input type="checkbox"/> DataNode	<input type="checkbox"/> NFSGateway	<input type="checkbox"/> NodeManager	<input type="checkbox"/> RegionServer	<input type="checkbox"/> Phoenix Query Server	<input type="checkbox"/> Livy for Spark2 Server	<input type="checkbox"/> Spark2 Thrift Server	<input type="checkbox"/> Client	
master2.example.com●	<input type="checkbox"/> DataNode	<input type="checkbox"/> NFSGateway	<input type="checkbox"/> NodeManager	<input type="checkbox"/> RegionServer	<input type="checkbox"/> Phoenix Query Server	<input type="checkbox"/> Livy for Spark2 Server	<input type="checkbox"/> Spark2 Thrift Server	<input type="checkbox"/> Client	
worker1.example.com	<input checked="" type="checkbox"/> DataNode	<input type="checkbox"/> NFSGateway	<input checked="" type="checkbox"/> NodeManager	<input checked="" type="checkbox"/> RegionServer	<input type="checkbox"/> Phoenix Query Server	<input type="checkbox"/> Livy for Spark2 Server	<input type="checkbox"/> Spark2 Thrift Server	<input type="checkbox"/> Client	
worker2.example.com	<input checked="" type="checkbox"/> DataNode	<input type="checkbox"/> NFSGateway	<input checked="" type="checkbox"/> NodeManager	<input checked="" type="checkbox"/> RegionServer	<input type="checkbox"/> Phoenix Query Server	<input type="checkbox"/> Livy for Spark2 Server	<input type="checkbox"/> Spark2 Thrift Server	<input type="checkbox"/> Client	
worker3.example.com	<input checked="" type="checkbox"/> DataNode	<input type="checkbox"/> NFSGateway	<input checked="" type="checkbox"/> NodeManager	<input checked="" type="checkbox"/> RegionServer	<input type="checkbox"/> Phoenix Query Server	<input type="checkbox"/> Livy for Spark2 Server	<input type="checkbox"/> Spark2 Thrift Server	<input type="checkbox"/> Client	

Items per page: 25 ▾ 1 - 6 of 6 ▶

BACK CANCEL NEXT →

5.1.10 7: Customize Services → CREDENTIALS

ระบุ Username , Password ตามที่ท่านสร้างฐานข้อมูลไว้ จากนั้นคลิก NEXT

Installer

CREDENTIALS — DATABASES — DIRECTORIES — ACCOUNTS — ALL CONFIGURATIONS

Please provide credentials for these services

	Username*	Password*	Confirm Password*
Grafana Admin	admin
Hive Database	hive
Oozie Database	oozie
Activity Explorer's Admin	N/A

BACK CANCEL NEXT →

5.1.11 7: Customize Services → DATABASES เชื่อมต่อ service ต่าง ๆ กับฐานข้อมูล MariaDB
หัวข้อ Hive Database เลือก Existing MySQL / MariaDB คลิก TEST CONNECTION จากนั้น คลิก
NEXT

Ambari - Cluster Install Wizard - Mozilla Firefox

File Edit View History Bookmarks Tools Help

Openlandscape Cloud 203.150.243.106/HDP/centos Ambari - Cluster Install V

CREDENTIALS — DATABASES — DIRECTORIES — ACCOUNTS — ALL CONFIGURATIONS

Please choose and configure the appropriate databases for these services

HIVE OOZIE RANGER

Hive Database: Existing MySQL / MariaDB

Hive Database Type: mysql

JDBC Driver Class: com.mysql.jdbc.Driver

Database Name: hive

Database Password:
TEST CONNECTION Connection OK

<https://dev.mysql.com/downloads/connector/j/>

5.1.12 7: Customize Services → DIRECTORIES กำหนดเป็น Default คลิก NEXT

Installer

CREDENTIALS — DATABASES — DIRECTORIES — ACCOUNTS — ALL CONFIGURATIONS

HDFS YARN MAPREDUCE2 TEZ HIVE HBASE OOZIE ZOOKEEPER AMBARI METRICS SPARK2

Default (6) Filter...

DATA DIRS

DataNode directories
/hadoop/hdfs/data

NameNode directories
/hadoop/hdfs/namenode

SecondaryNameNode Checkpoint directories
/hadoop/hdfs/namesecondary

NFSGateway dump directory
/tmp/hdfs-nfs

NameNode Backup directory
/tmp/upgrades

5.1.13 7: Customize Services → ACCOUNTS กำหนดเป็น Default คลิก NEXT

Installer

CREDENTIALS — DATABASES — DIRECTORIES — ACCOUNTS — ALL CONFIGURATIONS

Please review these settings for Service Accounts

Use Ambari to Manage Service Accounts and Groups
 Use Ambari to Manage Group Memberships
 Use Ambari to Manage Service Accounts UIDs

Users/Groups	Usernames
Smoke User	ambari-qa
Hadoop Group	hadoop
Ambari Metrics User	ams
HBase User	hbase
HDFS User	hdfs
Proxy User Group	users
Hive User	hive
Mapreduce User	mapred
Oozie User	oozie
Livy2 Group	livy
Livy2 User	livy
Spark2 Group	spark
Spark2 User	spark

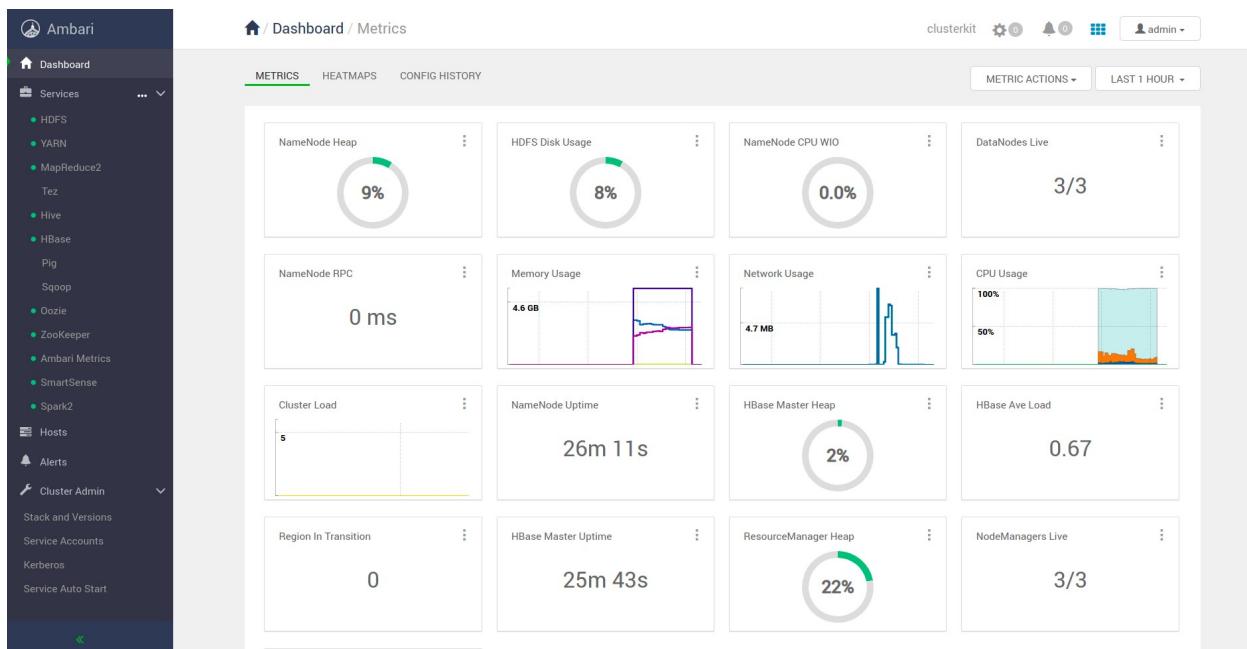
5.1.14 7: Customize Services → ALL CONFIGURATIONS กำหนดเป็น Default คลิก NEXT

The screenshot shows the Ambari Installer interface during the 'Customize Services' step. The 'All Configurations' tab is selected under the HDFS section. The 'SETTINGS' tab is active. On the left, a sidebar shows the progress of the setup steps. On the right, there are two sections: 'NameNode' and 'DataNode', each containing several configuration parameters with sliders and dropdown menus.

5.1.15 8: Review สรุปส่วนต่าง ๆ ที่จะติดตั้งในระบบ เมื่อคลิก next Ambari จะเริ่มติดตั้งไปยังเครื่องในคลัสเตอร์ คลิก DEPLOY จากนั้นรอจนติดตั้งสำเร็จ กดปุ่ม Complete

The screenshot shows the Ambari Installer interface during the 'Review' step. It displays a summary of the configuration settings, including Admin Name, Cluster Name, Total Hosts, Repositories, and Services. At the bottom, there are three buttons: GENERATE BLUEPRINT, PRINT, and DEPLOY.

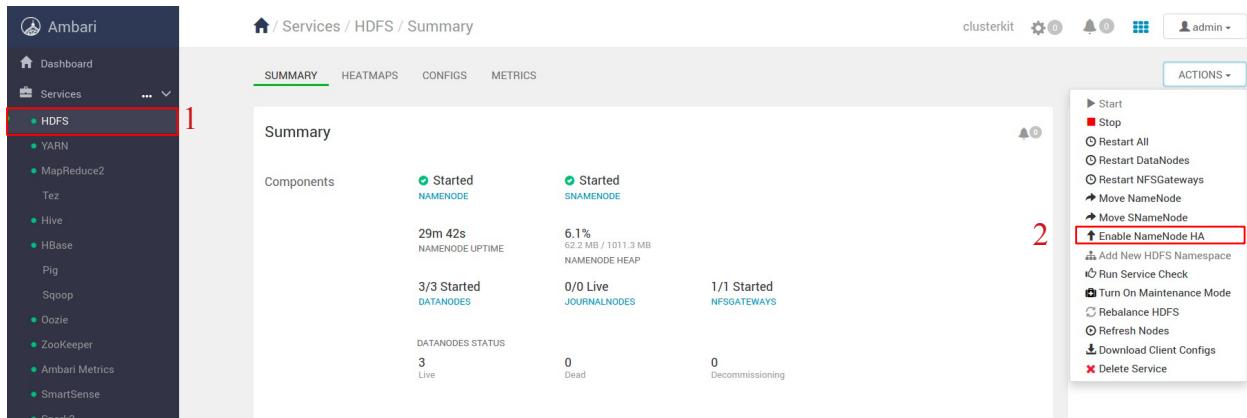
การติดตั้ง Hortonworks Data Platform (HDP) สำเร็จ



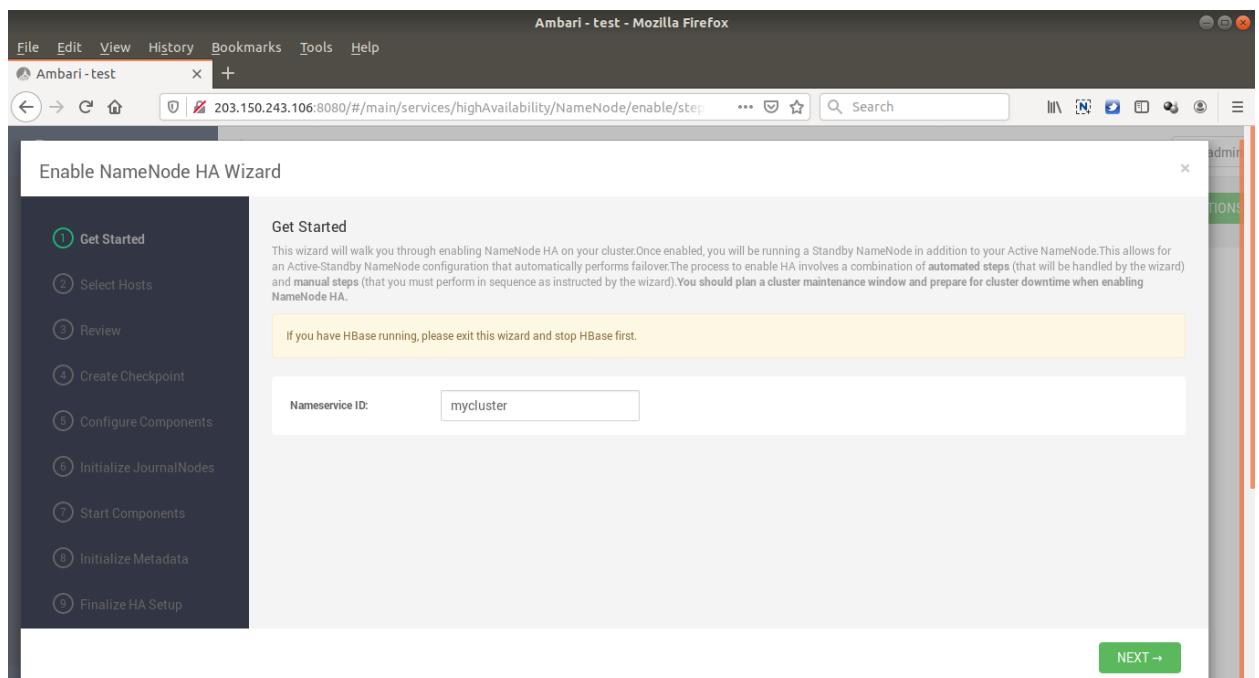
6 การทำ High Availability

6.1 การทำ Namenode HA

6.1.1 หน้า Ambari ไปที่ HDFS คลิก Actions จากนั้นเลือก Enable NameNode HA



6.1.2 จะปรากฏหน้า Wizard ทำการระบุ Nameservice ID ที่จะเป็น NameNode FQDN หลังจาก เช็ค HA เรียบร้อยแล้ว จากนั้นคลิก NEXT

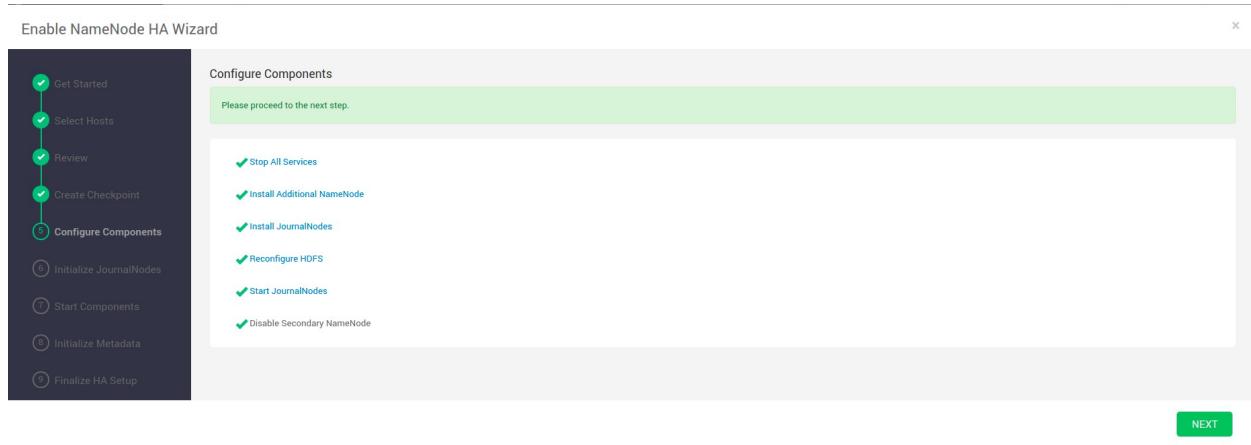


6.1.3 Select Hosts: เลือกเครื่องที่เป็น Namenode และ JournalNode

6.1.4 Review: สรุปการปรับแต่งที่จะทำ NameNode HA จากนั้นคลิก NEXT

6.1.5 Create Checkpoint : จังหวะนี้ให้รีโมทไปที่เครื่อง master1 เพื่อเปลี่ยนเป็น safe mode ใช้คำสั่ง `sudo su hdfs -l -c 'hdfs dfsadmin -safemode enter'` และสร้าง checkpoint ใช้คำสั่ง `sudo su hdfs -l -c 'hdfs dfsadmin -saveNamespace'` จากนั้นคลิก Next

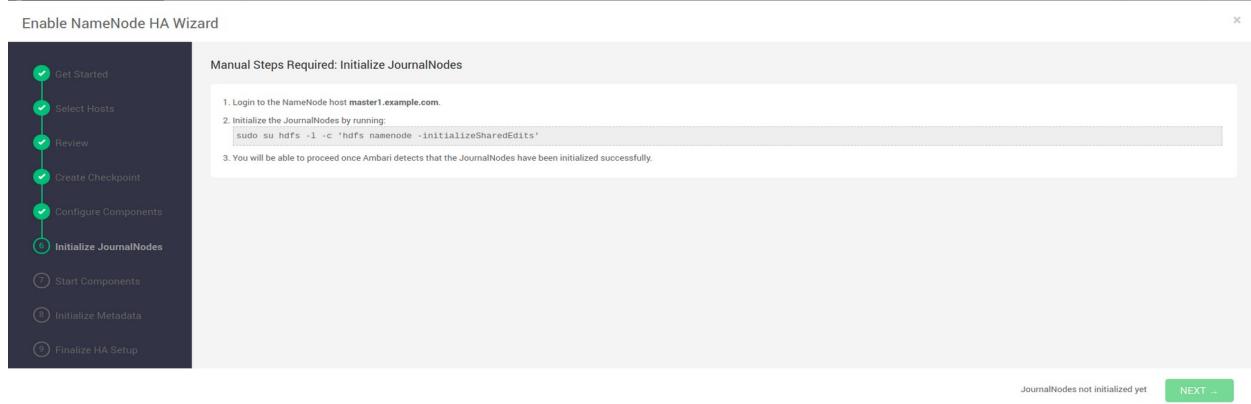
6.1.6 Configure Components : รอดำเนินการคอนฟิก เมื่อสำเร็จคลิก NEXT



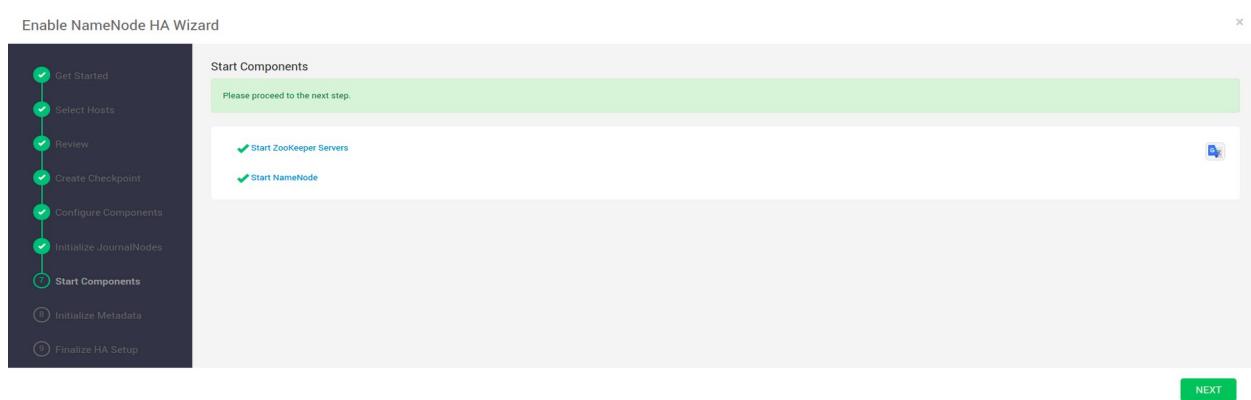
6.1.7 Manual Steps Required : Initialize JournalNodes

จังหวะนี้ให้รีโมทไปที่เครื่อง master1 เริ่มการทำงาน JournalNodes โดยใช้คำสั่ง

`sudo su hdfs -l -c 'hdfs namenode -initializeSharedEdits'` จากนั้นคลิก Next

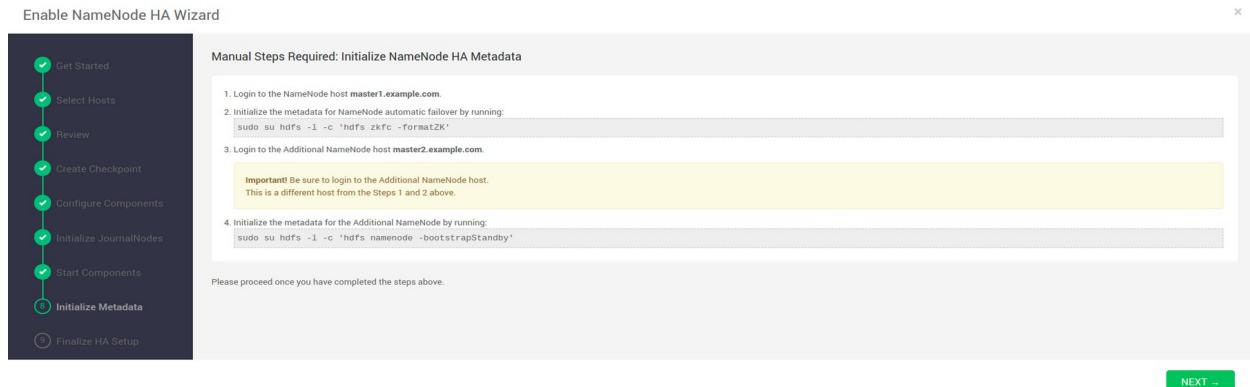


6.1.8 Start Components : เริ่มการทำงาน ZooKeeper Servers และ NameNode

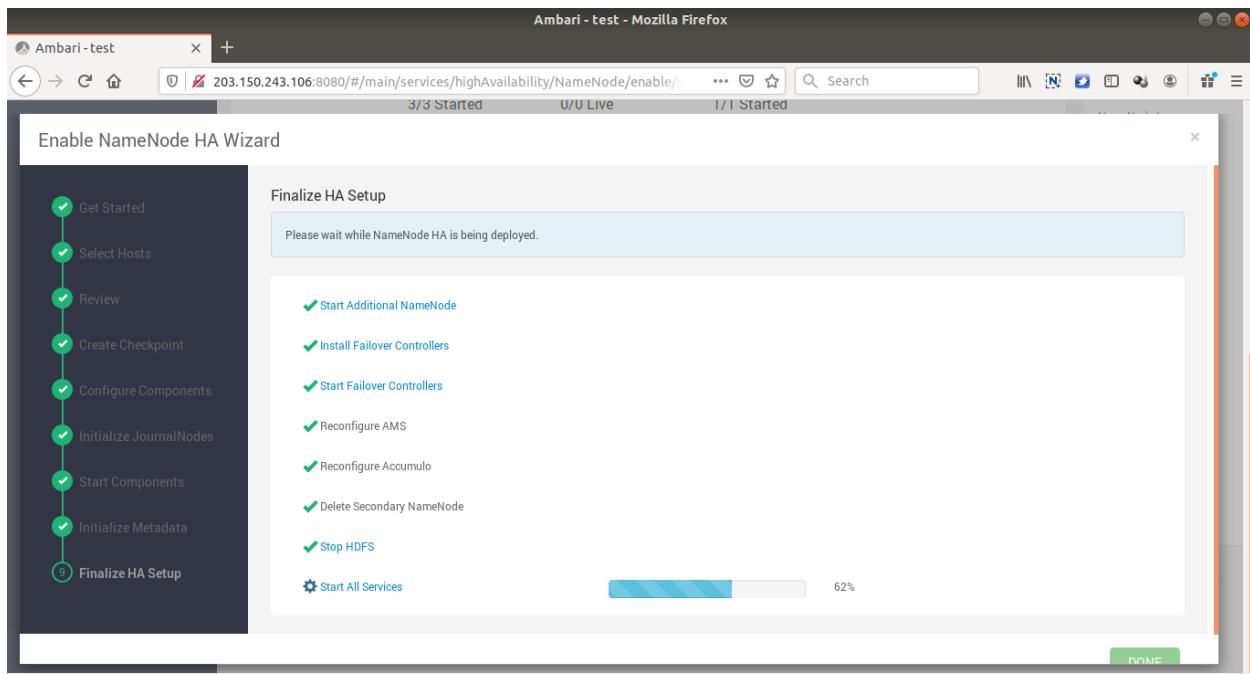


6.1.9 Manual Steps Required: Initialize NameNode HA Metadata

รีโมทไปที่เครื่อง master1 พิมพ์คำสั่ง `sudo su hdfs -l -c 'hdfs zkfc -formatZK'` และรีโมทไปที่เครื่อง master2 พิมพ์คำสั่ง `sudo su hdfs -l -c 'hdfs namenode -bootstrapStandby'` จากนั้นคลิก Next



6.1.10 Ambari จะปรับแต่งระบบให้ HDFS ทำงานแบบ HA ให้รองรับระบบปรับแต่งเรียบร้อย กดปุ่ม DONE เพื่อเสร็จสิ้นการทำงาน



6.2 Enable ResourceManager HA Wizard

6.2.1 ที่หน้า Ambari ไปที่ Services > YARN คลิก Actions จากนั้นเลือก Enable ResourceManager HA

The screenshot shows the Ambari interface for the YARN service. On the left, there's a sidebar with various services listed. The 'YARN' service is selected and highlighted with a red box, labeled with a red number '1'. In the main content area, there's a 'Summary' tab showing component status: Timeline Service V1.5 (Started), ResourceManager (Started), and YARN Client (1 Installed). Below this, it shows NodeManager status: 3 Active, 0 Lost, 0 Unhealthy, and 0 Rebooted. To the right, an 'Actions' dropdown menu is open, also highlighted with a red box and labeled with a red number '2'. This menu contains several options related to ResourceManager HA, with 'Enable ResourceManager HA' being the one selected.

6.2.2 Get Started – คลิก Next เริ่ม Wizard ResourceManager HA Wizard

The screenshot shows the 'Enable ResourceManager HA Wizard' step 1: Get Started. On the left, a navigation bar lists steps 1 through 4: Get Started (selected), Select Host, Review, and Configure Components. The main content area has a title 'Get Started' and a description explaining the wizard's purpose: enabling ResourceManager HA on the cluster. It states that once enabled, a Standby ResourceManager will run in addition to the Active ResourceManager, allowing for an Active-Standby configuration. Below this, there's a note about planning a cluster maintenance window and preparing for downtime. At the bottom right is a green 'NEXT →' button. The bottom of the screen shows the Ambari sidebar with various services like Sqoop, Oozie, ZooKeeper, Accumulo, Ambari Metrics, SmartSense, and Unite.

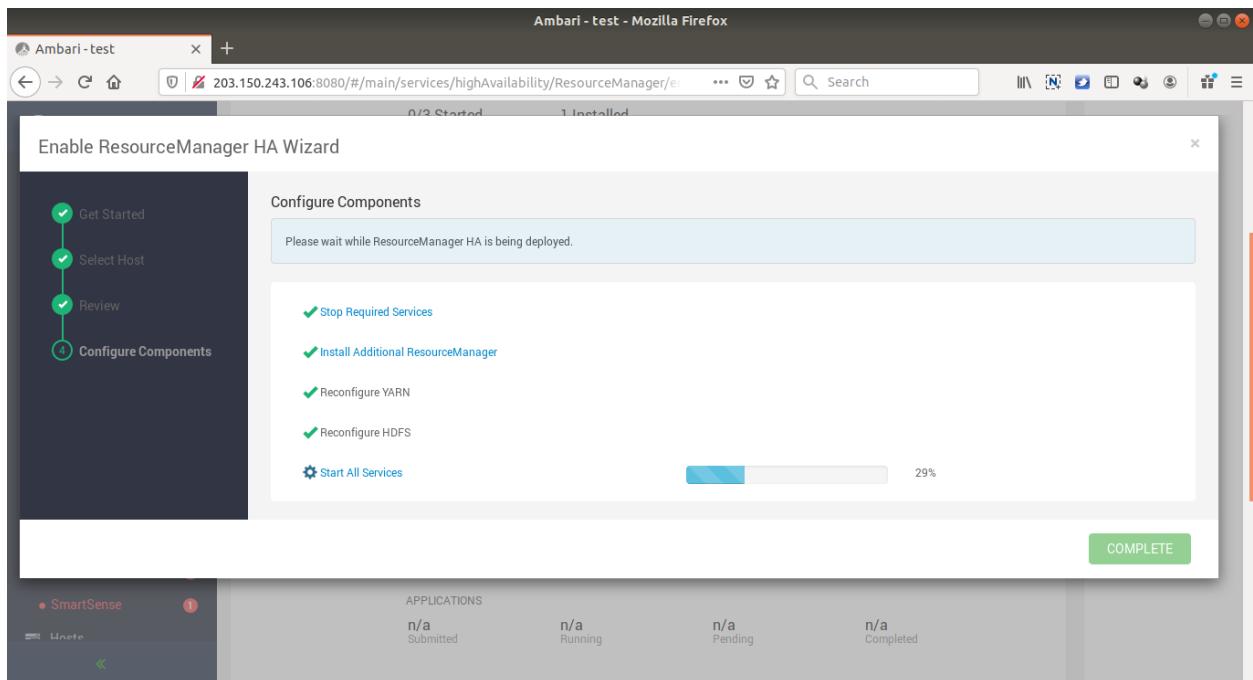
6.2.3 Select host – เลือก host ที่จะทำ ResourceManager HA

The screenshot shows the 'Select Host' step of the 'Enable ResourceManager HA Wizard'. On the left, a navigation sidebar lists steps: Get Started (done), Select Host (done), Review (in progress), and Configure Components (not yet started). The main panel displays two hosts: 'edge.example.com (15.5 GB, 4 cores)' and 'master2.example.com (3.7 GB, 2 cores)'. To the right, a detailed service status table is shown for 'edge.example.com' (15.5 GB, 4 cores), listing services like ResourceManager, Timeline Service V2.0 Reader, YARN Registry DNS, Hive Metastore, etc. Below it is another table for 'master1.example.com' (3.7 GB, 2 cores), showing services like NameNode, Timeline Service V1.5, History Server, and ZooKeeper Server.

6.2.4 Review : หน้าสรุปการค่อนพิกัดก่อนการติดตั้ง

The screenshot shows the 'Review' step of the 'Enable ResourceManager HA Wizard'. The sidebar shows steps: Get Started (done), Select Host (done), Review (done), and Configure Components (not yet started). The main panel shows host selections: 'Current ResourceManager: edge.example.com' and 'Additional ResourceManager: master2.example.com + TO BE INSTALLED'. Below this, a 'Review Configuration Changes' section lists configuration changes for YARN, including 'yarn.resourcemanager.ha.enabled: true', 'yarn.resourcemanager.ha.rm-ids: rm1,rm2', 'yarn.resourcemanager.hostname: edge.example.com', and 'yarn.resourcemanager.resource': 'edge.example.com:8025'.

6.2.5 เริ่มการคอนฟิก ResourceManager HA



รอจนคอนฟิกเรียบร้อย กดปุ่ม COMPLETE เพื่อจบการติดตั้ง

7 Hue

Hue เป็นบริการที่ช่วยให้ผู้ใช้งานสามารถใช้ Hadoop Ecosystems ผ่านหน้าเว็บได้ ไม่ได้มีมา กับ Hortonworks แต่เราสามารถติดตั้งและคอนฟิกให้ใช้งานร่วมกันได้ สามารถดาวน์โหลดและดูรายละเอียดเพิ่มเติมได้ที่ <https://gethue.com/> และแก้คอนฟิกไฟล์ที่ไฟล์โดยตรง ดังรายละเอียดต่อไปนี้

7.1 ติดตั้ง Hue

ในที่นี้จะใช้การติดตั้งผ่าน docker จึงจำเป็นต้องติดตั้ง docker บนลินุกซ์ก่อน

```
wget https://download.docker.com/linux/centos/docker-ce.repo \
      -O /etc/yum.repos.d/docker-ce.repo
yum install -y docker-ce docker-ce-cli containerd.io
systemctl enable --now docker
```

จากนั้นจึงเริ่มการทำงานของ docker hue

```
sudo docker run -d -p 8888:8888 --name hue gethue/hue
```

Hue จะเริ่มการทำงานที่พอร์ต 8888 แต่จะยังคงใช้งานไม่ได้ ต้องคอนฟิกให้รู้จักกับบริการต่าง ๆ ของ Hadoop ก่อนว่าจะทำงานอยู่ที่เครื่องไหน

ในระบบนี้จะใช้วิธีคัดลอกคอนฟิกเรซันไฟล์ออกจากมายังลินุกซ์เพื่อความสะดวกในการแก้ไข แล้วจึงเริ่มการทำงานของ Hue docker ใหม่อีกรอบ

```
mkdir hue
cd hue
sudo docker cp hue:/usr/share/hue/desktop/conf .
```

เริ่มการทำงาน Hue ใหม่ โดยกำหนดให้ใช้คอนฟิกไฟล์ที่กำหนด

```
docker run -d -p 8888:8888 --name hue --network=host \
      -v /root/hue/conf/hue.ini:/usr/share/hue/desktop/conf/hue.ini gethue/hue
```

7.2 ปรับแต่ง Hue configuration file (/root/hue/conf/hue.ini)

1) กำหนดพอร์ตและ time zone

```
http_port=8888  
time_zone=Asia/Bangkok
```

2) กำหนดให้ Hue ใช้ MySQL เป็น DBMS (จริง ๆ เราใช้ MariaDB)

```
[[database]]  
engine=mysql  
host=192.168.2.10  
port=3306  
user=hue  
password=xxxxxx  
name=hue
```

3) กำหนดค่า HDFS

```
[hadoop]
```

```
[[[default]]]  
# Enter the filesystem uri  
fs_defaultfs=hdfs://mycluster #ใส่ชื่อ nameservice ที่กำหนดตอนทำ HDFS NamenodeHA  
  
# ระบบเราใช้ httpfs (วิธีการติดตั้งอยู่ในหัวข้อถัด ๆ ไป)  
webhdfs_url=http://master1.example.com:14000/webhdfs/v1/
```

4) กำหนดค่าให้ Yarn

```
resourcemanager_api_url=http://master1.example.com:8088  
proxy_api_url=http://master1.example.com:8088  
history_server_api_url=http://master2.example.com:19888  
logical_name=yarn-cluster #คำนี้ดูจากไฟล์คอนฟิกที่ตัวแปร  
yarn.resourcemanager.cluster-id  
  
[[[ha]]]  
# Resource Manager logical name (required for HA)  
logical_name=yarn-cluster  
  
# Un-comment to enable  
submit_to=True  
  
# URL of the ResourceManager API  
resourcemanager_api_url=http://master2.example.com:8088  
proxy_api_url=http://master2.example.com:8088  
history_server_api_url=http://master2.example.com:19888
```

5) กำหนดค่าให้ Hive

```
hive_server_host=edge.example.com  
use_sasl=true
```

6) กำหนดค่าให้ Oozie

```
oozie_url=http://edge.example.com:11000/oozie
```

7) กำหนดค่าให้ Solr

```
[search]  
# URL of the Solr Server  
solr_url=http://edge.example.com:8886/solr/
```

8) กำหนดค่าให้ Zookeeper

```
host_ports=edge.example.com:2181,master1.example.com:2181,master2.example.com:
```

2181

9) กำหนดค่าการ Authentication กับ LDAP

```
backend=desktop.auth.backend.LdapBackend  
base_dn="dc=example,dc=com"  
ldap_url=ldaps://ipa.example.com  
ldap_username_pattern="uid=<username>,cn=users,cn=accounts,dc=example,dc=com"  
create_users_on_login = true  
sync_groups_on_login=true  
search_bind_authentication=true  
user_filter="objectclass=person"  
user_name_attr=uid  
group_filter="objectclass=posixgroup"  
group_name_attr=cn  
group_member_attr=groups
```

10) ปรับแต่งเมนูบนหน้าเว็บ Hue ให้ uncomment ในส่วนของ interpreters ดังต่อไปนี้

```
[[interpreters]]
```

- hive
- pig
- spark2
- mapreduce
- sqoop1
- distcp

7.3 ย้ายฐานข้อมูลมาใช้ MariaDB

เนื่องจากเราเริ่มการทำงานของ Hue ไปแล้ว ฐานข้อมูลจะถูกสร้างบนซอฟต์แวร์ SQLite แต่เราสามารถในคอนฟิกให้มายัง MariaDB จึงต้อง sync โครงสร้างและข้อมูลต่าง ๆ มา ทำได้ดังคำสั่งต่อไปนี้

```
docker exec -it hue /bin/bash  
build/env/bin/hue syncdb  
build/env/bin/hue migrate  
exit
```

เริ่มการทำงานของ Hue ใหม่ เพื่อให้ใช้คอนฟิกที่เราปรับแต่งไป

```
docker restart hue
```

7.4 กำหนด proxy user hadoop สำหรับ Hue

ไปที่ Ambari เมนู HDFS --> Configs --> Advanced และเลื่อนลงมาหา “Custom core-site” จากนั้นคลิก “Add Property...” และเพิ่มรายการ hadoop.proxyuser.hue.hosts และ hadoop.proxyuser.hue.groups โดยมีค่าเป็น * ลงไป จนกว่าจะกด Save

hadoop.proxyuser.hue.hosts	*
hadoop.proxyuser.hue.groups	*
Add Property ...	

Restart services ที่เกี่ยวข้องตามที่ Ambari กำหนด

7.5 กำหนด proxy user oozie สำหรับ Hue

ที่เมนู OOZIE → Configs → Advanced → Custom oozie-site เพิ่มรายการต่อไปนี้

Custom oozie-site

oozie.processing.timezone	GMT+0700
oozie.service.ProxyUserService. proxyuser.hue.groups	*
oozie.service.ProxyUserService. proxyuser.hue.hosts	*

Restart services ที่เกี่ยวข้องตามที่ Ambari กำหนด

7.6 สร้าง home directory for hue user

```
su - hdfs -c "hdfs dfs -mkdir /user/hue"
su - hdfs -c "hdfs dfs -chown hue:hdfs /user/hue"
su - hdfs -c "hdfs dfs -chmod 755 /user/hue"
```

7.7 ติดตั้ง hadoop-httpfs

HTTPFS เป็นบริการ API สำหรับบริการไฟล์ Hue จะ browse ไฟล์ใน HDFS ได้ก็ต้องมีบริการนี้ หรือ บริการ WebHDFS แต่เนื่องจากในระบบนี้เราทำ Active/Standby Namenode จึงเลือกใช้ HTTPFS ที่มีคุณสมบัตรของรับการ Namenode High Availability แต่ Ambari ไม่ได้ติดตั้งมาให้จึงต้องติดตั้งเอง ในระบบนี้จะทำที่เครื่องที่ทำหน้าที่ namenode คือเครื่อง edge, master1 มีขั้นตอนดังต่อไปนี้

```
yum install -y hadoop-httpfs
```

สร้าง httpfs-signature.secret

```
echo "1q2w3e4r" > /usr/hdp/3.1.4.0-315/hadoop/conf/httpfs-signature.secret
```

สร้าง systemd service สำหรับ httpfs ด้วยการสร้างไฟล์ /usr/lib/systemd/system/hadoop-httpfs.service มีเนื้อความดังต่อไปนี้

```
[Unit]
Description = Start Hadoop HttpFS service
After = network.target

[Service]
Type=forking
Restart=always
User=hdfs
ExecStart = /bin/hdfs --daemon start httpfs
[Install]
WantedBy = multi-user.target
```

สร้างไดเรกทอรี temp และกำหนดสิทธิ์

```
mkdir /usr/hdp/3.1.4.0-315/hadoop-hdfs/temp
chmod 1777 /usr/hdp/3.1.4.0-315/hadoop-hdfs/temp
```

reload systemctl เพิ่มให้รู้จัก service ใหม่ และสั่งให้บริการ hadoop-httpfs เริ่มการทำงานทุกครั้งที่เปิดเครื่องพร้อมทั้งเริ่มทำงานทันที

```
systemctl daemon-reload
systemctl enable --now hadoop-httpfs.service
```

กำหนด proxy user oozie สำหรับ Hue ให้สร้างไฟล์

/usr/hdp/current/hadoop-client/conf/httpfs-site.xml มีเนื้อความดังต่อไปนี้

```
<?xml version="1.0" encoding="UTF-8"?>
<?xml-stylesheet type="text/xsl" href="configuration.xsl"?>
<configuration>
  <property>
    <name>httpfs.proxyuser.hue.hosts</name>
    <value>*</value>
  </property>
  <property>
    <name>httpfs.proxyuser.hue.groups</name>
    <value>*</value>
  </property>
</configuration>
```

เริ่มการทำงานของบริการ HTTPFS ใหม่อีกครั้ง

```
systemctl restart hadoop-httpfs
```

จากนั้นจึงเข้าใช้งานเข้าใช้งาน Hue ที่หน้าเว็บ <http://edge.example.com:8888>

8 Zeppelin authentication configuration

การปรับแต่งให้ Zeppelin ยืนยันตัวตนผ่าน LDAP จะต้องเข้าไปคอนฟิกใน Ambari โดยให้ไปที่ Zeppelin Notebook → CONFIGS แล้วเลือกไปที่ “Advanced zeppelin-shiro-ini” แก้ไขในกรอบที่เขียนว่า “shiro_ini_content”

1. remark all in [users] section

```
[users]
# List of users with their password allowed to access Zeppelin.
# To use a different strategy (LDAP / Database / ...) check the shiro doc at ...
# admin = $shiro1$SHA-256$500000$p6Be9+t2hdUXJQj2D0b1fg==, admin
# user1 = $shiro1$SHA-256$500000$G2yymy/qmuZnGY6or4v2KfA==, role1, role2
# user2 = $shiro1$SHA-256$500000$aHBgiuwSgAcP3Xt5mEzeFw==, role3
# user3 = $shiro1$SHA-256$500000$nf0GzH10GbYVoxa7D0l0Sw==, role2
```

2. config ldap in [main] section

```
### A sample for configuring LDAP Directory Realm
ldapRealm = org.apache.zeppelin.realm.LdapGroupRealm
## search base for ldap groups (only relevant for LdapGroupRealm):
ldapRealm.contextFactory.environment[ldap.searchBase] = dc=example,dc=com
ldapRealm.contextFactory.url = ldaps://ipa.example.com
ldapRealm.userDnTemplate = uid={0},cn=users,cn=accounts,dc=example,dc=com
ldapRealm.contextFactory.authenticationMechanism = SIMPLE
```

3. Import LDAP CA Certificate to JDK

รันคำสั่งต่อไปนี้ที่เครื่องที่ Zeppelin ทำงานอยู่ เพื่อนำเข้า CA cert ของ IPA ไปยัง JDK KeyStore ด้วยคำสั่ง keytool ดังต่อไปนี้

```
/usr/jdk64/jdk1.8.0_112/bin/keytool -import -trustcacerts \
-keystore /usr/jdk64/jdk1.8.0_112/jre/lib/security/cacerts \
-alias IPACert \
-file /etc/ipa/ca.crt
```

ระบบจะถามรหัสผ่าน ให้ระบุว่า changeit จากนั้นให้ restart Zeppelin

Tip! ในระบบจริงควรจะเปลี่ยนรหัสผ่านของ KeyStore ตามที่เขียนนำว่าให้เปลี่ยนนะ (changeit) สามารถทำได้ด้วยคำสั่งต่อไปนี้

```
/usr/jdk64/jdk1.8.0_112/bin/keytool -storepasswd \
-keystore /usr/jdk64/jdk1.8.0_112/jre/lib/security/cacerts
```

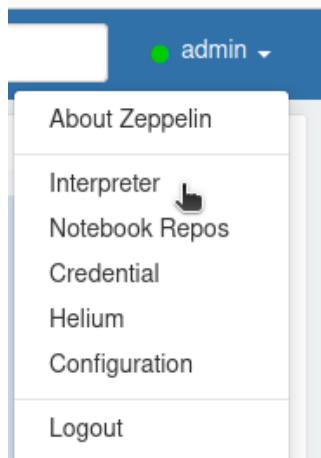
ระบบจะถามรหัสผ่านเก่า แล้วให้ตั้งรหัสผ่านใหม่สองครั้ง เป็นอันเสร็จพิธี

4. ทำ Zeppelin User Impersonate โดยไปเพิ่มพารามิเตอร์เหล่านี้ โดยไปที่ Ambari เมนู HDFS --> Configs --> Advanced และเลือกโนดไปหา “Custom core-site” จากนั้นคลิก “Add Property...”

hadoop.proxyuser.zeppelin.groups	*
hadoop.proxyuser.zeppelin.users	*
hadoop.proxyuser.zeppelin.hosts	*

Add Property ...

จากนั้น Restart service ที่เกี่ยวข้องตามที่ Ambari กำหนด



5. ปรับแต่ง Zeppelin Interpreter โดยไปที่หน้าเว็บ Zeppelin และล็อกอินด้วยบัญชีผู้ใช้ admin จากนั้นไปที่เมนู admin มุมบนขวา แล้วเลือกที่ Interpreter

6. ไปที่ spark2 และเลือก Per User, Isolated และ User Impersonate จากนั้นจึง save

spark2 %spark2, %spark2.sql, %spark2.dep, %spark2.pyspark, %spark2.ipyspark, %spark2.r ●

Option

The interpreter will be instantiated in process
 User Impersonate
 Connect to existing process
 Set permission

เป็นอันเสร็จพิธี

9 Ranger authentication with LDAP

9.1 Import LDAP CA cert

รันคำสั่งต่อไปนี้ที่เครื่องที่ Ranger ทำงานอยู่

```
/usr/jdk64/jdk1.8.0_112/bin/keytool -import -trustcacerts \
-keystore /usr/jdk64/jdk1.8.0_112/jre/lib/security/cacerts \
-alias IPAcert \
-file /etc/ipa/ca.crt
```

ระบบจะถามรหัสผ่าน ให้ระบุว่า changeit

จากนั้นให้เปิด Ambari → Ranger → CONFIGS และค้นว่า **ranger.usersync.truststore.file** ให้เปลี่ยนค่าเป็น **/usr/jdk64/jdk1.8.0_112/jre/lib/security/cacerts** และบันทึก จากนั้น restart service ตามที่ Ambari กำหนด

9.2 Config Ranger authen LDAP

ที่ Ambari ให้ไปที่ Ranger → CONFIGS → RANGER USER INFO

Ranger User Info

Enable User Sync

Yes

Sync Source: LDAP/AD

Sync Source

LDAP/AD

COMMON CONFIGS USER CONFIGS GROUP CONFIGS

LDAP/AD URL

ldap://ipa.example.com

Bind User

uid=system,cn=sysaccounts,cn=etc,dc=example,dc=com

ที่แท้ป COMMON CONFIGS

- Enable User Sync = Yes
- LDAP/AD URL:

ldap://ipa.example.com (สามารถใช้ ldaps ได้)

- Bind User: uid=system,cn=sysaccounts,cn=etc,dc=example,dc=com
- Bind User Password:
- Incremental sync: Yes

COMMON CONFIGS **USER CONFIGS** GROUP CONFIGS

Username Attribute
uid

User Object Class
posixaccount

User Search Base
cn=accounts,dc=example,dc=com

COMMON CONFIGS USER CONFIGS **GROUP CONFIGS**

Enable Group Sync

Group Member Attribute
groups

Group Name Attribute
cn

Group Object Class
posixgroup

Group Search Base
cn=groups,cn=accounts,dc=example,dc=com

ที่แท็บ USER CONFIGS

- Username Attribute: **uid**
- User Object Class: **posixaccount**
- User Search Base:
cn=accounts,dc=example,dc=com
- User Group Name Attribute: **uid**
- Group User Map Sync = **No**
- Enable User Search = **No**

ที่แท็บ GROUP CONFIGS

- Enable Group Sync = **Yes**
- Group Member Attribute: **groups**
- Group Name Attribute: **cn**
- Group Object Class: **posixgroup**
- Group Search Base:
cn=groups,cn=accounts,dc=example,dc=com
- Enable Group Search First = **No**
- Sync Nested Groups = **No**

หลังจากคอนฟิกค่าข้างต้นแล้วให้ Save และ

restart service ตามที่ Ambari แจ้ง

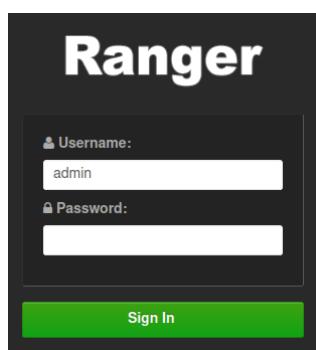
จากนั้นให้เข้าไปที่ Ranger Admin UI (สามารถคลิกได้จากเมนู Quick Links ทางซ้ายมือ)

[/ Services / Ranger / Summary](#)

BDUNINET 0 4 admin ·

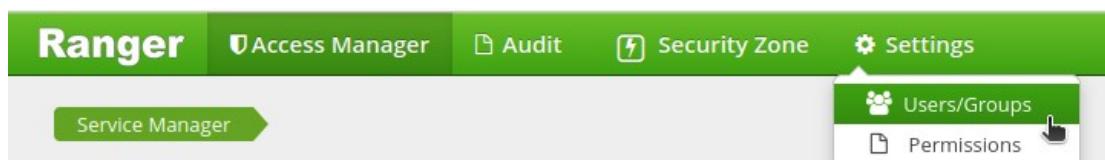
SUMMARY	CONFIGS	ACTIONS									
<h3>Summary</h3> <table border="1"> <thead> <tr> <th>Components</th> <th>Started RANGER ADMIN</th> <th>Started RANGER USERSYNC</th> </tr> </thead> <tbody> <tr> <td>1/1 Started RANGER TAGSYNC</td> <td>Enabled RANGER HDFS PLUGIN</td> <td>Enabled RANGER YARN PLUGIN</td> </tr> <tr> <td>Enabled RANGER HBASE PLUGIN</td> <td>Enabled RANGER ATLAS PLUGIN</td> <td>Disabled RANGER KAFKA PLUGIN</td> </tr> </tbody> </table>			Components	Started RANGER ADMIN	Started RANGER USERSYNC	1/1 Started RANGER TAGSYNC	Enabled RANGER HDFS PLUGIN	Enabled RANGER YARN PLUGIN	Enabled RANGER HBASE PLUGIN	Enabled RANGER ATLAS PLUGIN	Disabled RANGER KAFKA PLUGIN
Components	Started RANGER ADMIN	Started RANGER USERSYNC									
1/1 Started RANGER TAGSYNC	Enabled RANGER HDFS PLUGIN	Enabled RANGER YARN PLUGIN									
Enabled RANGER HBASE PLUGIN	Enabled RANGER ATLAS PLUGIN	Disabled RANGER KAFKA PLUGIN									
<h3>Quick Links</h3> <ul style="list-style-type: none"> Ranger Admin UI 											





จากนั้นให้ล็อกอิน

ไปที่เมนู Setting → Users/Groups



ดูที่แท็บ Users และ Groups ว่าเห็น Users และ Groups ที่มาเพิ่มไว้ใน FreeIPA หรือไม่ ถ้าเข็ตได้ถูกต้องจะเห็น Users และ Groups บน FreeIPA แบบนี้ก็จะสามารถกำหนดสิทธิ์การใช้งานให้กับ User และ Group เหล่านี้ได้

Select	User Name	Email Address	Role	User Source
<input type="checkbox"/>	sqoop		User	External
<input type="checkbox"/>	yarn		User	External
<input type="checkbox"/>	mapred		User	External
<input type="checkbox"/>	amb_ranger_admin		Admin	Internal
<input type="checkbox"/>	demouser		User	External
<input type="checkbox"/>	demobd		User	External

10 การเพิ่มบัญชีผู้ใช้และการใช้งาน

ระบบ Big Data Hadoop ชุดนี้ ได้เชื่อมโยงการตรวจสอบตัวตนผ่านโปรโตคอล LDAP กับ FreeIPA ดังนั้นการจัดการบัญชีผู้ใช้งาน ให้ทำผ่าน FreeIPA สามารถเข้าใช้งานได้ที่ <https://ipa.example.com/>

1. การเพิ่มกลุ่มผู้ใช้งาน

การแบ่งกลุ่มผู้ใช้งานจะทำให้สะดวกในการกำหนดสิทธิ์ให้กับกลุ่มผู้ใช้ ให้เข้าไปที่เมนู Groups และกดปุ่ม +Add เพื่อเพิ่มกลุ่มผู้ใช้

Group name	GID	Description
admins	1932000000	Account administrators group
editors	1932000002	Limited admins who can edit other users
ipausers		Default group for all users
sysadmin	1932000054	
trust admins		Trusts administrators group

2. กำหนดค่า Group name และกด Add

User Groups

Group name	GID
admins	1932000000
clusterkit	1932000056
editors	1932000002
ipausers	
sysadmin	1932000054
trust admins	

3. เพิ่มกลุ่มสำเร็จ จะพบข้อความพร้อม GID

4. เพิ่มผู้ใช้งาน ใหม่ที่เมนู Users และกดปุ่ม +Add

The screenshot shows the 'ROCKY IDENTITY MANAGEMENT' interface. On the left, there's a sidebar with 'User categories' (Active users, Stage users, Preserved users). The main area shows a table of 'Active users' with one entry: 'admin' (User login: admin, First name: Moungrungsuk, Last name: Kittirak, Status: Enabled, UID: 1626600000). Below the table, it says 'Showing 1 to 1 of 1 entries.' At the top right, there are buttons for Refresh, Delete, +Add, Disable, Enable, and Actions.

Add User

User login: kittirak
 First name: Kittirak
 Last name: Moungrungsuk
 Class:
 No private group:
 GID: dropdown menu showing 'admins', 'clusterkit' (selected), 'editors', and 'sysadmin'
 New Password: (empty)
 Verify Password: (empty)
 * Required field

Buttons at the bottom: Add, Add and Add Another, Add and Edit, Cancel

6. เมื่อเพิ่มข้อมูลเสร็จจะกลับมาหน้าจอแสดงรายชื่อผู้ใช้ ให้คลิกเข้าไปที่ชื่อผู้ใช้ที่เพิ่งเพิ่มเข้าไปใหม่ เพื่อปรับแต่ง

<input type="checkbox"/>	kittirak	Kittirak	Moungrungsuk	<input checked="" type="checkbox"/> Enabled	1932000001	kittirak@example.com
--------------------------	----------	----------	--------------	---	------------	----------------------

Login shell: /bin/bash

Buttons at the bottom: Refresh, Revert, Save (highlighted), Actions

5. เพิ่มข้อมูลผู้ใช้งาน ระบุข้อมูลต่อไปนี้ เสร็จแล้วกด Add

- 5.1 User login
- 5.2 First name
- 5.3 Last name
- 5.4 GID เลือกกลุ่มที่จะใช้ผู้ใช้งานอยู่
- 5.5 New Password
- 5.6 Verify Password

7. ที่ต้อง Login shell ให้เปลี่ยนจาก /bin/sh เป็น /bin/bash ดังรูป

8. จากนั้นให้กดปุ่ม Save เป็นอันเสร็จกระบวนการเพิ่มบัญชีผู้ใช้งาน

11 การใช้งาน

ผู้ใช้งานสามารถเข้าใช้บริการผ่านช่องทางดังต่อไปนี้

- ssh edge.example.com
- Hue <http://edge.example.com:8888>
- Zeppelin <http://edge.example.com:9995>

11.1 บริการ ssh

การสร้างบัญชีผู้ใช้งาน FreeIPA นั้น ระบบจะบังคับให้ผู้ใช้งานต้องกำหนดรหัสผ่านใหม่ เมื่อล็อกอินครั้งแรกกับบริการ ssh ดังตัวอย่างต่อไปนี้

```
sh kittirak@edge.example.com
(kittirak@edge.example.com) Password:
(kittirak@edge.example.com) Password expired. Change your password now.
Current Password:
(kittirak@edge.example.com) New password:
(kittirak@edge.example.com) Retype new password:
Creating home directory for kittirak.
Last login: Mon Dec 18 12:25:19 +07 2023
[kittirak@edge ~]$
```

หลังจากล็อกอินเข้าสู่ระบบ ระบบจะแจ้งว่ารหัสผ่านหมดอายุ แล้วจะให้เราระบุรหัสผ่านเดิมอีกครั้ง จากนั้นจึงให้กรอกรหัสผ่านใหม่สองครั้ง รหัสผ่านนี้ถูกบันทึกที่ FreeIPA ก็จะมีผลกับทุกระบบทั้งหน้าเว็บ Hue และ Zeppelin

** การพิมพ์รหัสผ่านบน ssh นั้น เคอร์เซอร์จะไม่ขยับ ให้พิมพ์ไปเลย**

11.2 Hue บริการหน้าเว็บสำหรับ Hadoop Ecosystems

เมื่อล็อกอินบนหน้าเว็บ Hue ครั้งแรก ระบบจะสร้างโฮมไดเรอทอรี (Home directory) บน HDFS (/user/USERNAME) ให้โดยอัตโนมัติ หากผู้ใช้งาน Hadoop บน command line ก็จำเป็นต้องมาล็อกอินบนหน้าเว็บ Hue นี้ก่อนเพื่อให้มีโฮมไดเรกทอรีของตน