[Skip to end of metadata](https://wiki.gsretail.com/pages/viewpage.action?pageId=118179285" \l "page-metadata-end)

* [문광모/클라우드팀](https://wiki.gsretail.com/display/~kwangmo.moon)님이 작성, [2022-08-24](https://wiki.gsretail.com/pages/diffpagesbyversion.action?pageId=118179285&selectedPageVersions=1&selectedPageVersions=2)에 최종 변경

[Go to start of metadata](https://wiki.gsretail.com/pages/viewpage.action?pageId=118179285#page-metadata-start)

* [ETL 서버 정보 및 아키텍처](https://wiki.gsretail.com/pages/viewpage.action?pageId=118179285#id-06.OpenSourceETL%EC%84%9C%EB%B2%84%EA%B5%AC%EC%84%B1-ETL%EC%84%9C%EB%B2%84%EC%A0%95%EB%B3%B4%EB%B0%8F%EC%95%84%ED%82%A4%ED%85%8D%EC%B2%98)
  + [OS 그룹 및 계정 생성](https://wiki.gsretail.com/pages/viewpage.action?pageId=118179285#id-06.OpenSourceETL%EC%84%9C%EB%B2%84%EA%B5%AC%EC%84%B1-OS%EA%B7%B8%EB%A3%B9%EB%B0%8F%EA%B3%84%EC%A0%95%EC%83%9D%EC%84%B1)
* [소프트웨어 설치](https://wiki.gsretail.com/pages/viewpage.action?pageId=118179285#id-06.OpenSourceETL%EC%84%9C%EB%B2%84%EA%B5%AC%EC%84%B1-%EC%86%8C%ED%94%84%ED%8A%B8%EC%9B%A8%EC%96%B4%EC%84%A4%EC%B9%98)
  + [Embulk 설치](https://wiki.gsretail.com/pages/viewpage.action?pageId=118179285#id-06.OpenSourceETL%EC%84%9C%EB%B2%84%EA%B5%AC%EC%84%B1-Embulk%EC%84%A4%EC%B9%98)
  + [python3 설치](https://wiki.gsretail.com/pages/viewpage.action?pageId=118179285#id-06.OpenSourceETL%EC%84%9C%EB%B2%84%EA%B5%AC%EC%84%B1-python3%EC%84%A4%EC%B9%98)
  + [OS 라이브러리 설치](https://wiki.gsretail.com/pages/viewpage.action?pageId=118179285#id-06.OpenSourceETL%EC%84%9C%EB%B2%84%EA%B5%AC%EC%84%B1-OS%EB%9D%BC%EC%9D%B4%EB%B8%8C%EB%9F%AC%EB%A6%AC%EC%84%A4%EC%B9%98)
  + [MySQL 설치](https://wiki.gsretail.com/pages/viewpage.action?pageId=118179285#id-06.OpenSourceETL%EC%84%9C%EB%B2%84%EA%B5%AC%EC%84%B1-MySQL%EC%84%A4%EC%B9%98)
* [소프트웨어 구성](https://wiki.gsretail.com/pages/viewpage.action?pageId=118179285#id-06.OpenSourceETL%EC%84%9C%EB%B2%84%EA%B5%AC%EC%84%B1-%EC%86%8C%ED%94%84%ED%8A%B8%EC%9B%A8%EC%96%B4%EA%B5%AC%EC%84%B1)
  + [MySQL DB 구성](https://wiki.gsretail.com/pages/viewpage.action?pageId=118179285#id-06.OpenSourceETL%EC%84%9C%EB%B2%84%EA%B5%AC%EC%84%B1-MySQLDB%EA%B5%AC%EC%84%B1)
  + [airflow db 구성](https://wiki.gsretail.com/pages/viewpage.action?pageId=118179285#id-06.OpenSourceETL%EC%84%9C%EB%B2%84%EA%B5%AC%EC%84%B1-airflowdb%EA%B5%AC%EC%84%B1)
  + [airflow DB 연결](https://wiki.gsretail.com/pages/viewpage.action?pageId=118179285#id-06.OpenSourceETL%EC%84%9C%EB%B2%84%EA%B5%AC%EC%84%B1-airflowDB%EC%97%B0%EA%B2%B0)
* [소프트웨어 기동 및 중지](https://wiki.gsretail.com/pages/viewpage.action?pageId=118179285#id-06.OpenSourceETL%EC%84%9C%EB%B2%84%EA%B5%AC%EC%84%B1-%EC%86%8C%ED%94%84%ED%8A%B8%EC%9B%A8%EC%96%B4%EA%B8%B0%EB%8F%99%EB%B0%8F%EC%A4%91%EC%A7%80)
  + [MySQL 기동 및 중지](https://wiki.gsretail.com/pages/viewpage.action?pageId=118179285#id-06.OpenSourceETL%EC%84%9C%EB%B2%84%EA%B5%AC%EC%84%B1-MySQL%EA%B8%B0%EB%8F%99%EB%B0%8F%EC%A4%91%EC%A7%80)
  + [Airflow 웹 서버 / 스케줄러 기동 및 중지](https://wiki.gsretail.com/pages/viewpage.action?pageId=118179285#id-06.OpenSourceETL%EC%84%9C%EB%B2%84%EA%B5%AC%EC%84%B1-Airflow%EC%9B%B9%EC%84%9C%EB%B2%84/%EC%8A%A4%EC%BC%80%EC%A4%84%EB%9F%AC%EA%B8%B0%EB%8F%99%EB%B0%8F%EC%A4%91%EC%A7%80)

ETL 서버 정보 및 아키텍처

| **구분** | **10.52.222.171** |  |
| --- | --- | --- |
| 소프트웨어 구성 | * MySQL v8 * Airflow v2.2.5 * Embulk v0.9.24 * Python v3 |  |
| 소프트웨어 설명 | * MySQL : Airflow Repository * Airflow : ETL 배치 스케줄러 * Embulk : ETL | embulk 별도 기동 중지 없음 |
| Application OS User | * MySQL : mysql / gs!my!@34 * Airflow : airflow / airflow * Embulk : embulk / embulk |  |
| Service URL | * Airflow : [http://10.52.222.171:8080](http://10.52.222.171:8080/) | (admin / admin) |

**OS 그룹 및 계정 생성**

|  |
| --- |
| -- group  groupadd embulk  groupadd mysql  groupadd airflow    -- user  useradd airflow -g airflow  useradd embulk -g embulk  useradd mysql -g mysql    -- pwd  airflow / airflow  embulk / embulk  mysql /  gs!my!@34 |

소프트웨어 설치

**Embulk 설치**

|  |
| --- |
| # root 계정 수행 : java 설치  yum install java-1.8.0-openjdk-devel  yum install java-1.8.0-openjdk    embulk 계정 수행 :  # 소스 다운로드  curl --create-dirs -o ~/.embulk/bin/embulk -L "<https://dl.embulk.org/embulk-latest.jar>"    # 권한 설정  chmod +x ~/.embulk/bin/embulk    # 환경설정  echo 'export PATH="$HOME/.embulk/bin:$PATH"' >> ~/.bashrc    # 환경 등록  source ~/.bashrc    # 플러그인 다운로드  embulk gem list  : list 확인  embulk gem install embulk-input-postgresql  embulk gem install embulk-output-postgresql  embulk gem install embulk-input-oracle  embulk gem install embulk-output-oracle  embulk gem install embulk-input-mysql  embulk gem install embulk-output-mysql  embulk gem install embulk-output-bigquery |

**python3 설치**

|  |
| --- |
| # root 수행  # python3 설치  yum install python3    # 버전 확인  python -V , python3 -V  Python 3.6.8    # 버전 등록 및 변경  update-alternatives --install /bin/python python /bin/python2.7 1  update-alternatives --install /bin/python python /bin/python3.6 2  update-alternatives --config python    # pip 등록 및 버전 확인  ln -s /bin/pip3.6 /bin/pip  pip -V    # yum = python3 연결  yum install python3 : error 확인  -- 아래 2개 파일에서 맨 윗줄 수정  vi /usr/bin/yum  vi /usr/libexec/urlgrabber-ext-down  #!/usr/bin/python => #!/usr/bin/python2.7  yum <-- 테스트 확인    pip install -U pip setuptools |

**OS 라이브러리 설치**

|  |
| --- |
| cat mysql\_yum\_install.sh    #!/bin/sh  #################################################  # Script : mysql\_yum\_install.sh  # Date : 2022. 04. 14  # Decription : Mysql YUM INSTALL  # History : Moon Kwang Mo , first - 2022.04.14  #################################################  ###  ### 0. mysql YUM INSTALL & SETTING  ###  pwd=`pwd`  cd ${pwd}    echo "#zlib" > yum\_t.lst  echo "#curl" >> yum\_t.lst  echo "#gcc" >> yum\_t.lst  echo "#cpp" >> yum\_t.lst  echo "#gcc-c++" >> yum\_t.lst  echo "#openssl" >> yum\_t.lst  echo "#openssl-devel" >> yum\_t.lst  echo "#ncurses-devel" >> yum\_t.lst  echo "#bzip2-devel" >> yum\_t.lst  echo "#bison" >> yum\_t.lst  echo "#perl" >> yum\_t.lst  echo "#make" >> yum\_t.lst  echo "#cmake" >> yum\_t.lst    cat ${pwd}/yum\_t.lst |sed -e s/#//g > yum.lst  rm -rf ${pwd}/yum\_t.lst    ####  #### 1. yum install start  ####  MYSQL\_DIR=`pwd`  cd ${MYSQL\_DIR}  DATE=`date +%Y%m%d`  YUM\_LOG=${MYSQL\_DIR}/${DATE}\_yum\_install.log  rm -rf ${YUM\_LOG}  #### CONFILG    rk=0  rk=`cat ${pwd}/yum.lst |wc -l`    i=1  while  (( ${i} < ${rk} ))  do    for list in `cat ${pwd}/yum.lst`  do    check=0  check=`yum list installed ${list} |grep -i ${list} |wc -l`    if [ ${check} -eq 0 ]  then     echo " [${i}/${rk}] : ${list} no package :  check : ${check} " |tee -a ${YUM\_LOG}     echo " yum install ${list} start & Exit , install shell restart " |tee -a ${YUM\_LOG}            yum -y install ${list}     echo " yum ${list} install end !!!!  restart shell script !!!!!!!!!!!!!!" |tee -a ${YUM\_LOG}    echo " " |tee -a ${YUM\_LOG}  echo "##### START SHELL : `date +%Y-%m-%d:%H:%M:%S` #####################################" |tee -a ${YUM\_LOG}  #          exit 0    fi     echo " [${i}/${rk}] : ${list} install Yes !!!!  : check : ${check} " |tee -a ${YUM\_LOG}     echo " "    i=`expr $i \+ 1`    done  done |

**MySQL 설치**

|  |
| --- |
| cat mysql\_install.sh ( ROOT 수행 )    #!/bin/sh  #################################################  # Script : mysql\_install.sh  # Date : 2022. 04. 14  # Decription : Mysql INSTALL  # History : Moon Kwang Mo , first - 2022.04.14  #################################################  ## dir create (as root)  mkdir -p /applications/INST  chown -R mysql:mysql /applications/INST  chmod -R 755 /applications/INST    cd /applications/INST    wget <http://dev.mysql.com/get/Downloads/MySQL-8.0/mysql-8.0.18-linux-glibc2.12-x86_64.tar.xz>  chown -R mysql.mysql /applications/INST/mysql-8.0.18-linux-glibc2.12-x86\_64.tar.xz    tar xvf /applications/INST/mysql-8.0.18-linux-glibc2.12-x86\_64.tar.xz  mv /applications/INST/mysql-8.0.18-linux-glibc2.12-x86\_64/ /applications/  chown -R mysql.mysql /applications/mysql-8.0.18-linux-glibc2.12-x86\_64/    cd /applications  ln -s /applications/mysql-8.0.18-linux-glibc2.12-x86\_64 mysql  chown -R mysql.mysql /applications/mysql    #### 수동 편집  ## 2. my.cnf CONFIG  vi /applications/mysql/my.cnf    ## 3. DB CREATE & SETTING  ln -s /applications/mysql/my.cnf /etc/my.cnf  mkdir -p /applications/mysql/tmp  mkdir -p /applications/mysql/log    ## 4. su - mysql  # !  mysql init exe  /applications/mysql/bin/mysqld --initialize --user=mysql  /applications/mysql/bin/mysqld\_safe --defaults-file=/etc/my.cnf --user=mysql &    A temporary password is generated for root@localhost: iuhajbhoe6/H    /applications/mysql/bin/mysql -uroot -p --socket=/applications/mysql/tmp/mysql.sock  alter user user() identified by 'gs!my!@34' ;    ## login-path 잡기  /applications/mysql/bin/mysql\_config\_editor set --login-path=myroot --user=root --password --host=localhost  /applications/mysql/bin/mysql --login-path=myroot --socket=/applications/mysql/tmp/mysql.sock  /applications/mysql/bin/mysqladmin --login-path=myroot --socket=/applications/mysql/tmp/mysql.sock --port=3306 shutdown |

소프트웨어 구성

**MySQL DB 구성**

|  |
| --- |
| -- etc error 처리 --  # mysql : 33060 OFF 처리    [mysqld]  mysqlx=OFF    #아래 변수 에러시 아래 값으로 다시 입력  secure\_file\_priv = ""    # Table Upper/Lower 변수 에러시 DB초기화 옵션 주고 진행  lower\_case\_table\_names=1  /applications/mysql/bin/mysqld --initialize --user=mysql --lower\_case\_table\_names=1    MySQL 8.0부터는 default\_authentication\_plugin 이 mysql\_native\_password 에서 caching\_sha2\_password 로 변경되었다.  \* native\_password 사용하기 위해서 default 설정 변경  [mysqld]  default\_authentication\_plugin=mysql\_native\_password |

**airflow db 구성**

|  |
| --- |
| CREATE DATABASE airflow CHARACTER SET utf8mb4 COLLATE utf8mb4\_unicode\_ci;    CREATE USER 'airflow' IDENTIFIED BY 'gs!air!@34';  CREATE USER 'airflow'@localhost IDENTIFIED BY 'gs!air!@34';  CREATE USER 'airflow'@'%' IDENTIFIED BY 'gs!air!@34';  create user gsdba@'%' IDENTIFIED BY 'gs!my!@34';    GRANT ALL PRIVILEGES ON airflow.\* TO airflow ;  GRANT ALL PRIVILEGES ON airflow.\* TO 'airflow'@localhost ;  GRANT ALL PRIVILEGES ON airflow.\* TO 'airflow'@'%' ;  GRANT ALL PRIVILEGES ON airflow.\* TO 'gsdba'@'%' ;    -- native\_password로 생성 안했을경우 변경  ALTER USER 'airflow'@'localhost' IDENTIFIED WITH mysql\_native\_password BY 'gs!air!@34';  ALTER USER 'airflow'@'localhost' IDENTIFIED WITH mysql\_native\_password BY 'gs!air!@34';  ALTER USER 'airflow'@'%' IDENTIFIED WITH mysql\_native\_password BY 'gs!air!@34'; |

**airflow DB 연결**

|  |
| --- |
| 1. vi airflow.cfg    sql\_alchemy\_conn = mysql://airflow:gs!air!@34@10.52.124.77/airflow\_dba?charset=utf8  load\_examples = False  executor = LocalExecutor    2. airflow db init    > db init 설정중 에러 나는 경우 에러에 맞게 조치    # mysql 모듈 에러일 경우 설치  pip3 install apache-airflow  pip3 install 'apache-airflow[mysql]'    3. airflow 사용 계정 생성 (아래 그대로 복사)  airflow users create \   --username admin \   --firstname FIRST\_NAME \   --lastname LAST\_NAME \   --role Admin \   --password admin \   --email kwangmo.moon@gmail.com |

소프트웨어 기동 및 중지

**MySQL 기동 및 중지**

|  |
| --- |
| 기동 : /applications/mysql/bin/mysqld\_safe --defaults-file=/etc/my.cnf --user=mysql &  중지 : mysqladmin -uroot -p'gs!my!@34' shutdown --socket=/applications/mysql/tmp/mysql.sock |

**Airflow 웹 서버 / 스케줄러 기동 및 중지**

|  |
| --- |
| 중지 : pkill -9 -ef airflow  기동 : nohup airflow webserver -p 8080 &         nohup airflow scheduler &  확인 : ps -ef | grep airflow         ps -ef | grep "airflow scheduler"    ## restart Shell  cat air\_restart.sh    echo " "  echo "####################################################"    echo " "  echo "####################################################"  pwd=`pwd`  cd ${pwd}    echo " airflow kill "  pkill -9 -ef airflow  sleep 3  echo " "  echo "####################################################"    echo " airflow ps check "  ps -ef | grep airflow    echo " airflow scheduler check "  ps -ef | grep "airflow scheduler"  sleep 1    echo " "  echo "####################################################"    echo " airflow seb start "  nohup airflow webserver -p 8080 &  sleep 2    echo " airflow scheduler start "  nohup airflow scheduler  > /dev/null 2>&1 &    echo " "  echo "####################################################"    echo " airflow ps check "  ps -ef | grep airflow    echo " airflow scheduler check "  ps -ef | grep "airflow scheduler" |