분석서버 관리 매뉴얼

|  |
| --- |
|  |

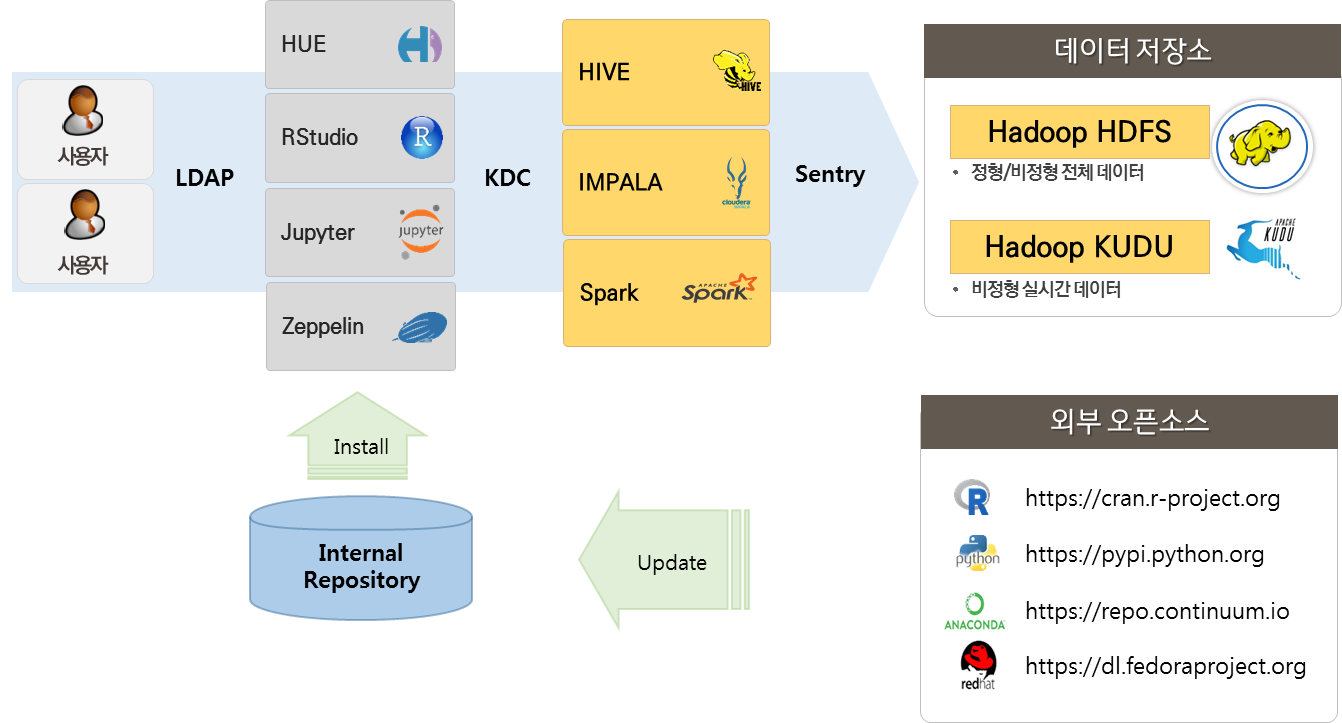
TEC-DP-04

Ver. 1.0

업무명 : 인프라

# 분석환경 개요

## 분석환경 구성도



## 관리자 역할

|  |  |
| --- | --- |
| 관리항목 | 상세 내역 |
| 사용자 관리 | 사용자 계정 관리 (LDAP설정)  Sentry 권한 부여  분석서버 사용자 Home 관리  사용자 Session 관리 |
| 분석 Tool 관리 | 분석 Tool 설정 파일 관리  분석 Tool 시작 및 종료  분석 Package 설치 |
| 내부 Repository 관리 | 외부 오픈소스 Update 확인  방화벽 Open 신청  내부 Repository 구성  내부 Repository Update |

## 분석서버 정보

### 하드웨어 구성

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| 용도 | 서버명 | 제품명 | CPU | MEM | DISK | OS |
| 분석서버 | HDPMLL01 | DELL R740 | 2.1GHz \* 2 | 512G | 600G\*2,2.4T\*6 | Redhat 7.5 |

서버명: 서버명에 대한 별도의 제약조건은 없으며, 일반적으로 회사 내 인프라 부서의 서버명 명명 규칙에 따라 정한다. 서버명 명명 규칙은 업무구분-어플리케이션 용도-OS 종류-넘버링의 순으로 하는 것이 일반적이다.

제품명: 하드웨어 제품에 대한 별도의 제약 조건은 없다. 보통 장애 발생 시 신속히 서비스를 받을 수 있도록 회사내에서 가장 많이 사용하는 벤더의 제품을 선호한다.

CPU: CPU는 빠를수록 많을수록 좋다. 보통 분석서버를 사용하는 사용자의 수준에 따라 정하는 것이 일반적이다. 모델링 분석을 하는 사용자는 8Core, 일반적인 통계분석 사용자는 4Core, 간단한 자료 수집 사용자는 1Core 수준으로 계산한다. 필요에 따라 고성능의 분석을 위해서는 GPU를 사용한다.

MEM: 메모리도 사용자의 수와 자료의 크기에 따라 정한다. 보통 512GB ~ 1TB 로 구성한다.

DISK: 분석서버에는 많은 용량의 DISK를 필요로 한다. 우선 OS DISK와 분석용 DISK를 분리하여 구성하는 것을 권고한다. OS용 DISK는 300GB 이상 2개로 구성을 권고한다. 분석 수행시 보통 사용자 Home에 임시파일이 생성되는 경우가 많으므로 사용자 Home영역을 1TB이상 할당하는 것이 관리에 용이하다. 가능하다면 해당 영역을 스토리지로 할당하면 성능 향상에 도움이 된다. 내부 Repository를 구성하는 경우에는 5TB이상을 할당하는 것이 좋다. Python에서 제공하는 무료 Package의 용량이 계속 증가하고 있기 때문이다.

OS: OS는 Linux계열을 사용한다. 시스템 관리에 별다른 제약 조건-보안 및 모니터링 소프트웨어와의 호환성-이 없다면 최신 버전의 OS를 설치하는 것을 권고한다.

### Network 및 IP 주소

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 용도 | 서버명 | Network Interface | IP 주소 | Gateway 주소 |
| 분석서버 | HDPMLL01 | 10Gbps FC | 172.16.x.x/24 | 172.16.x.x |

Network Interface: 네트워크 카드의 속도는 서버의 기본 인터페이스에서 1Gbps를 지원한다. 하지만 하둡이나 DB등의 다른 서버에 분석 데이터가 존재하고 대량의 데이터를 자주 송수신해야 할 경우에는 10Gbps 네트워크 카드를 사용하는 것을 권고하다. 또한 네트워크 장비의 장애 발생 시에도 Down 없이 사용하기 위해서는 Bonding 구성이 필요하다.

IP주소: 분석 사용자 및 관리자가 접근 가능한 IP를 할당받아 사용한다. IP주소는 보통 회사내의 인프라 부서에서 관리한다.

Gateway 주소: 서버 IP와 통신하기 위한 Gateway IP이다. IP의 Subnet범위 내의 주소를 사용해야 한다.

### 파일 시스템 구성

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| 서버명 | DISK | VG | LV | Size | Type | Mount | 용도 |
| HDPML01 | 300G\*2  (RAID-1) | vgroot | lvroot | 100G | xfs | / | OS영역 |
| lvtmp | 100G | xfs | /tmp |
| lvcrash | 16G | xfs | /var/crash |
| swap | 64G | swap |  |
| 2.4T\*6  (RIAD-5) | vgdata | lvhome | 1TB | xfs | /home | 사용자Home |
| lvdata | 500G | xfs | /apps | 분석 Tool |
| lvrepo | 6TB | xfs | /repo | Repository |

서버의 DISK는 OS용도와 어플리케이션 용도에 따라 분리하여 구성하는 것을 권고한다.

OS영역은 보통 RAID-1(Mirror)으로 구성한다. 운영되는 소프트웨어에서 /tmp영역을 사용하는 경우가 많을 경우 별도의 Filesystem으로 분리하며, 서버의 Fault발생시 crash dump가 파일이 생성되는 /var/crash 영역도 별도로 분리한다. swap은 Disk 여유에 따라 적당한 사이즈로 할당한다.

어플리케이션 영역은 대량의 크기를 요구하므로 RAID-5 로 구성한다. 분석 사용자의 Home 공간을 별도로 분리 구성하며, 분석 Tool이 설치되는 영역은 분석Tool의 요구사항에 맞추어 적정한 크기의 파일시스템을 할당한다. 분석Tool Update를 위한 Repository 공간은 6TB이상 할당한다.(https://p.datadoghq.com/sb/7dc8b3250-85dcf667bd 에서 Pypi Mirror Size 조회 ) Repository 공간을 마련하기 어렵다면 분석Tool Package 설치 시 방화벽을 잠시 Open하여 설치하고, 다시 방화벽을 차단하면 된다.

### 설치 소프트웨어

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 서버명 | 소프트웨어 | Version | 운영계정 | 용도 |
| HDPMLL01 | httpd | 2.4.6 | root | Repository 공유 |
| R | 3.5.1 | root | R 분석 |
| RStudio | 1.1.463 | rstudio-server | R 개발환경 |
| Anaconda | 5.3.1 | conda | Python 관리 |
| Jupyterhub | 0.9.4 | conda | Python 개발환경 |
| Zeppelin | - | zeppelin | Spark 개발환경 |
| CM Agent | 6.2.0 | root | Hadoop 접속 |
| Hive ODBC | 2.6.1 | root | Hive ODBC |
| Impala ODBC | 2.6.2 | root | Impala ODBC |

httpd는 OS에서 기본 제공하는 Version으로 설치하며, 내부 Repository Directory를 공유하기 위해 사용한다.

R은 R분석을 위한 기본 소프트웨어이다. rpm으로 설치하며, Library이므로 별도의 기동되는 서비스는 없다.

RStudio는 R 개발을 위해 분석가들에게 제공되는 개발 환경이다. rpm으로 설치하며, rstudio-server 라는 계정이 생성된다. RStuio는 서비스 프로세스가 실행되며, 서버 재시작시 자동으로 실행되며, 관리자에 의해 재시작도 가능하다.

Anaconda는 Python 을 관리하기 위한 통합 Package이며, 실행 파일을 통해 설치한다. 설치 시 별도의 계정과 Directory위치를 지정할 수 있다. Library이므로 별도의 기동되는 서비스는 없다.

Jupyterhub는 Python개발을 위해 분석가들에게 제공되는 개발환경이다. anconda로 설정된 환경에 설치되며, 시작 명령을 통해 관리자가 서비스를 실행해야 한다.

Zeppelin은 Spark 개발을 위해 분석가들에게 제공되는 개발환경이며, 실행파일을 통해 설치한다. 별도의 계정과 Directory 위치를 지정할 수 있다. 시작 명령을 통해 관리자가 서비스를 실행해야 한다.

CMAgent는 Hadoop에 접속하기 위해 Cloudera에서 제공하는 Agent파일이다. rpm을 통해 설치하며, 별도의 서비스 프로세스가 기동된다. 해당 프로세스는 서버 재시작시 자동으로 실행된다.

Hive ODBC는 Hadoop에 Hive Query를 수행하기 위해 제공되는 Ribrary이며, Cloudera 공식 사이트를 통해 다운로드하여 rpm파일로 설치한다.

Impala ODBC는 Hadoop에 Impala Query를 수행하기 위해 제공되는 Ribrary이며, Cloudera 공식 사이트를 통해 다운로드하여 rpm파일로 설치한다.

### 서비스 포트 정보

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| 서버명 | 소프트웨어 | Port | Protocol | 용도 |
| HDPMLL01 | httpd | 80/443 | http/https | Repository 공유 |
| RStudio | 8787 | http | R 개발환경 |
| Jupyterhub | 8000 | http | Python 개발환경 |
| Zeppelin | 8043 | http | Spark 개발환경 |

httpd 서비스는 기본 80번 포트를 사용하며, 별도의 port로 변경 가능하다. 회사 보안 규정에 따라 SSL을 구성해야 한다면 기본 443번 포트를 할당한다.

RStudio서비스는 8787번 포트를 사용한다. http가 기본이며, https를 적용하기 위해서는 상용Version을 별도로 구매해야 한다.

Jupyterhub는 8000번 포트를 사용하며, 이는 서비스 기동 할 때 지정할 수 있다.

Zeppelin은 8443번 포트를 사용하며, 별도의 port로 변경 가능하다.

# 내부 Repository 관리

## Repository 정보

|  |  |  |
| --- | --- | --- |
| 구분 | 외부 Repository | 내부 Repository |
| R | https://cran.r-project.org | http://172.16.x.x/lib/cran |
| conda | https://repo.continuum.io | http://172.16.x.x/lib/conda |
| python | https://pypi.python.org | http://172.16.x.x/lib/pypi |
| epel | https://dl.fedoraproject.org | http://172.16.x.x/lib/epel |
| redhat | - | http://172.16.x.x/lib/dvd |

외부Repository에 대응하여 내부에 Repository를 구성한다. 외부 Repository SITE를 내부 Repository로 복제하여 구성하며, 향후 Update가 필요하다면 방화벽 Open 후 별도의 절차를 통해 진행한다.

## Redhat Repository 구성

설치 DVD를 내부 DISK로 복제하여 Redhat Repository를 구성할 수 있다. 외부 인터넷이 차단된 환경에서 서버를 운영할 경우 필요한 추가 소프트웨어를 설치하기 위해서 구성한다. 기본적인 절차는 다음과 같다.

1. DVD를 내부 DISK로 복사
2. Repository 소프트웨어 설치
3. Repository 정보 파일 생성
4. Repository 정보 생성

### DVD를 내부 DISK로 복사

lsblk 명령으로 CD-ROM 장치의 주소를 확인한다. TYPE이 rom으로 되어 있는 장치의 NAME을 확인한다.

|  |
| --- |
| # lsblk |
| NAME MAJ:MIN RM SIZE RO TYPE MOUNTPOINT  sda 8:0 0 32G 0 disk  ├─sda1 8:1 0 1G 0 part /boot  └─sda2 8:2 0 31G 0 part  ├─centos-root 253:0 0 29G 0 lvm /  └─centos-swap 253:1 0 2G 0 lvm [SWAP]  sr0 11:0 1 4.2G 0 rom |

cdrom을 mount할 Directory를 생성하고 DVD를 mount한다.

|  |
| --- |
| # mkdir /cdrom  # mount /dev/sr0 /cdrom |
| mount: /dev/sr0 is write-protected, mounting read-only |

DVD내의 파일들을 내부 Directory로 복사한다.

|  |
| --- |
| # mkdir –p /repo/dvd  # cp –r /cdrom/\* /repo/dvd |

### Repository 소프트웨어 설치

Repository 관련 소프트웨어는 DVD 내에 rpm으로 제공된다. /cdrom/Packages Directory로 이동하여 설치한다. bash에서는 파일명을 입력할 때 Tab Key를 지원하므로 파일명 앞부분을 입력한 후 Tab키를 눌러서 명령어를 입력할 수 있다.

|  |
| --- |
| # cd /cdrom/Packages/  # rpm -ivh deltarpm-3.6-3.el7.x86\_64.rpm  # rpm -ivh python-deltarpm-3.6-3.el7.x86\_64.rpm  # rpm -ivh libxml2-python-2.9.1-6.el7\_2.3.x86\_64.rpm  # rpm -ivh createrepo-0.9.9-28.el7.noarch.rpm |

설치가 완료 된 후 DVD를 마운트 해제 한 후 DVD 제거한다.

|  |
| --- |
| # cd /  # umount /cdrom |

### Repository 정보 파일 생성

Redhat에서는 yum명령으로 소프트웨어를 설치할 수 있다. yum명령은 종속성이 걸려있는 모든 소프트웨어를 검사하여 한번에 설치할 수 있다. yum 구성파일은 /etc/yum.repos.d 에 있으며, 확장자가 .repo인 로 되어 있어야 한다.

복사된 Directory에 대한 정보를 참조할 수 있도록 /etc/yum.repos.d에 신규 파일을 작성한다.

|  |
| --- |
| # vi /etc/yum.repos.d/dvd.repo |
| [dvd]  name=dvd  baseurl=file:///repo/dvd  enabled=1  gpgcheck=0 |

### Repository 정보 생성

createrepo 명령을 통해 dvd.repo를 참조할 수 있도록 한다.

|  |
| --- |
| # createrepo --datebase /repo/dvd/Packages |
| # yum update |

## Repository 환경 구성

R 및 Python 용 Package를 설치하기 위해 Repository 환경구성을 구성해야 한다. Repository 는 httpd를 통해 파일을 공유하므로 Redhat

## EPEL Repository 구성

Extra Packages for Enterprise Linux (EPEL) 를 말한다. 분석 Tool을 설치하기 위해서 필요한 Package들이 있으므로 서버 내부에 구성하는 것을 권고한다.

기본적인 절차는 다음과 같다.

1. 최신 EPEL 정보 파일 설치
2. 내부 복제Tool 설치
3. EPEL Package를 내부 DISK로 복사
4. EPEL Repository 정보 파일 생성
5. EPEL Repository 정보 생성

### 최신 EPEL 정보 파일 설치

최신 EPEL 정보를 가지고 있는 epel-release-latest-7.noarch.rpm 파일을 직접 지정하여 yum명령으로 설치한다. 설치를 위해서는 dl.fedoraproject.org 사이트의 443번 포트가 방화벽에 Open되어 있어야 한다. 서버에서 직접 구성할 수 없을 경우에는 인터넷이 사용이 가능한 PC에서 Package 파일들을 직접 다운로드하여 구성할 수도 있다.

|  |
| --- |
| # yum -y install \  https://dl.fedoraproject.org/pub/epel/epel-release-latest-7.noarch.rpm |

### 내부 복제 Tool 설치

yum 명령으로 내부 복제에 필요한 Tool을 설치한다. 해당 Tool의 설치파일은 기본 Repository에 있다.

|  |
| --- |
| # yum -y install yum-utils |

### EPEL Package를 내부 DISK로 복사

reposync 명령을 이용하여 외부의 EPEL Repository 파일들을 내부로 복제할 수 있다.

아래 명령을 사용하면 /repo Directory 아래에 repoid와 동일한 epel이란 Directory가 생성되고, 그 아래에 필요한 Package파일들이 다운로드 된다.

|  |
| --- |
| # reposync --repoid=epel --download\_path=/repo |

### EPEL Repository 정보파일 생성

EPEL Package들이 모두 다운로드가 완료되면, 더 이상 외부의 epel Repository를 참조하지 않도록 Repository 정보 파일을 다른 이름으로 변경한다.

|  |
| --- |
| # cd /etc/yum.reposd.d  # mv epel.repo epel.repo.org |

이후 내부 EPEL Repository를 참조하도록 신규 파일을 작성한다.

|  |
| --- |
| # vi /etc/yum.repos.d/epel.repo |
| [epel]  name=epel  baseurl=file:///repo/epel/Packages  enabled=1  gpgcheck=0 |

### EPEL Repository 정보 생성

createrepo 명령을 통해 epel.repo를 참조할 수 있도록 한다.

|  |
| --- |
| # createrepo --datebase /repo/epel/Packages |
| # yum update |

## R Repository 구성

R에서 사용하는 Package들의 내부 Repository 구성하여 사용자들이 필요한 Package들을 설치할 수 있도록 한다. R 및 RStudio를 업데이트는 별도의 절차를 통해 진행하며, R Repository 내에는 해당 파일들을 지원하지 않는다.

기본적인 구성 절차는 다음과 같다.

1. 내부 복제 Tool 설치
2. R Package를 내부 DISK로 복제
3. Repository 공유 환경 구성

### 내부 복제 Tool 설치

yum 명령으로 내부 복제에 필요한 Tool을 설치한다. 해당 Tool의 설치파일은 기본 Repository에 있다.

|  |
| --- |
| # yum -y install wget |

### R Package를 내부 DISK로 복사

R package는 https://cran.r-project.org에서 제공하며, wget 명령을 이용하여 파일들을 내부로 복제할 수 있다. 다운로드에는 많은 시간이 소요되므로 nohup 명령을 통해 백그라운드로 진행하는 것을 권고한다.

|  |
| --- |
| # nohup wget -P /repo --no-check-certificate -R “index.html” -r https://cran.r-project.org/src/contrib/ > cran.log & |
| # cd /repo  # mv cran.r-project.org cran |

### Repository 공유 환경 구성

R package는 https://cran.r-project.org에서 제공하며, wget 명령을 이용하여 파일들을 내부로 복제할 수 있다. 다운로드에는 많은 시간이 소요되므로 nohup 명령을 통해 백그라운드로 진행하는 것을 권고한다.

|  |
| --- |
| # nohup wget -P /repo --no-check-certificate -R “index.html” -r https://cran.r-project.org/src/contrib/ > cran.log & |
| # cd /repo  # mv cran.r-project.org cran |