시스템 운영관리 지침

KIGAM 한국지질자원연구원
지질자원데이터센터

B1. 시스템 운영관리 지침

서론

배경 및 목적

이 문서는 GEO, GDR 표준운영절차 (Standard Operating Procedure)를 기술한다.

적용 범위

본 규정은 한국지질자원연구원 지질자원데이터센터에서 관리 중인 GEO, GDR 운영 방침에 적용한다.

관련 용어 및 약어

- GEO : 지오빅데이터 오픈플랫폼 (Geo Big Data Open Platform)
- GDR : 지질자원 연구데이터 리포지터 (Geoscience Data Repository)
- 클라우드 : 인터넷을 통해 IT 리소스를 유연하게 이용하는 기술
- 온프레미스 : 기업이 자체 데이터센터에서 모든 IT 인프라를 운영하는 방식
- IPS (Intrusion Prevention System) : 네트워크에서 악의적인 트래픽 및 공격을 탐지하고 차단하는 시스템 (공격 탐지 및 차단, 패킷 검사, 시그니처 기반 감지)
- WAF (Web Application Firewall) : 웹 애플리케이션에서 발생하는 웹 애플리케이션 취약점을 보호하는 시스템 (웹 애플리케이션 취약점 보호, HTTP 요청 및 응답 검사)
- FW (Firewall) : 네트워크 트래픽을 제어하고 외부에서 내부로의 액세스를 관리하는 시스템 (패킷 필터링, 포트 및 프로토콜 제어, 네트워크 분리 및 보호)

시스템 개요

> 지오빅데이터 오픈플랫폼 (GEO Big Data Open Platform) 시스템 소개

GEO는 클라우드 기반 서비스로, 지질 자원 데이터와 관련 정보를 원격으로 액세스하고 활용할 수 있는 플랫폼이다.
사용자는 인터넷을 통해 지질 관련 데이터를 쉽게 검색하고 활용할 수 있다. 이는 데이터를 클라우드 서버에서 호스팅하고 웹 기반 인터페이스를 통해 접근 가능하게 함으로써 사용자에게 편리한 액세스를 제공한다.

> 지질자원 연구데이터 리포지터 (Geoscience Data Repository) 시스템 소개

GDR은 온프레미스(사내 서버 및 네트워크) 기반 서비스로, 지질 자원 연구 데이터의 안전한 보존과 관리를 위한 플랫폼이다.
연구 기관 내부에서 데이터를 저장하고 관리하며, 연구자들은 내부 네트워크를 통해 데이터에 접근한다.

시스템 환경

시스템 환경 구성

구분		GEO	GDR
인프라		클라우드 기반 호스팅	온프라미스 기반 호스팅
IP		115.68.202.95	203.247.179.88
호스트명		kigamcloud-375	gdr0
H/W	CPU	8Core (Intel(R) Xeon(R) CPU E5-2620 v4 @ 2.10GHz)	12Core(Intel(R) Xeon(R) CPU E5-2670 v3 @ 2.30GHz)
	Memory	64GB	64GB
	HDD	OS : 50GB, DATA : 500GB	OS: 250GB, WEB: 200GB, DB: 200GB, DATA: 354TB
S/W	OS	Ubuntu 20.04 LTS	CentOS Linux release 7.9.2009
	Docker	Docker 20.10.7	Docker version 20.10.5
	WEB	nginx 1.19.10	nginx 1.19.8
	WAS	apache tomcat 8.5	apache tomcat 9.0.63
	DBMS	postgres 10	postgres 10
	Search Engine	apache solr 7	apache solr 7
보안		침입방지(IPS, Intrusion Prevention System) 보안관제(WAF, Web Application Firewall), 초당 200MB 트래픽 처리	침입방지(IPS, Intrusion Prevention System) 보안관제(WAF, Web Application Firewall)

시스템 관리 - GEO

GEO 시스템은 Docker를 기반으로 소프트웨어를 관리하고 운영한다.
Docker는 소프트웨어를 컨테이너로 패키징하여 이식성과 확장성을 제공하며, 시스템 관리를 간편하게 해준다.
본 문서는 Docker를 활용하여 시스템을 관리하는 방법에 대한 안내를 제공한다.

> **서버접속**

SSH(Secure Shell) 접속을 통해 GEO 서버에 접속할 수 있으며, 접속 시 다음과 같은 화면이 출력된다.

```
ssh geo@[redacted] -p 2202
```

```
[ks79ks@tunnel ~]$ ssh geo@[redacted] -p 2202
The authenticity of host '[115.68.202.95]:2202 ([115.68.202.95]:2202)' can't be established.
ECDSA key fingerprint is SHA256:Mk5Ql2MowGeOI3+H6EF6cCq94JTc9zKPmlHoTNY7Zxo.
ECDSA key fingerprint is MD5:20:e4:c0:67:4a:f2:aa:2a:b1:78:73:69:49:39:14:80.
Are you sure you want to continue connecting (yes/no)? yes
Warning: Permanently added '[115.68.202.95]:2202' (ECDSA) to the list of known hosts.
geo@115.68.202.95's password:
Welcome to Ubuntu 20.04 LTS (GNU/Linux 5.4.0-135-generic x86_64)

 * Documentation:  https://help.ubuntu.com
 * Management:     https://landscape.canonical.com
 * Support:        https://ubuntu.com/advantage

  System information as of Fri Sep 15 18:08:41 KST 2023

  System load:  4.46               Users logged in:                  1
  Usage of /:   76.0% of 48.27GB   IPv4 address for br-650add947f6f: 172.22.0.1
  Memory usage: 58%                IPv4 address for docker0:         172.17.0.1
  Swap usage:   0%                 IPv4 address for ens3:            10.1.0.23
  Processes:    408

  => There is 1 zombie process.

 * Strictly confined Kubernetes makes edge and IoT secure. Learn how MicroK8s
   just raised the bar for easy, resilient and secure K8s cluster deployment.

   https://ubuntu.com/engage/secure-kubernetes-at-the-edge

163 updates can be installed immediately.
10 of these updates are security updates.
To see these additional updates run: apt list --upgradable


The list of available updates is more than a week old.
To check for new updates run: sudo apt update
New release '22.04.3 LTS' available.
Run 'do-release-upgrade' to upgrade to it.


Last login: Fri Sep 15 18:06:53 2023 from 115.68.220.244
geo@kigamcloud-375:~$
```

* 접속 시 SSH 접속 계정에 대한 비밀번호가 필요하며, 유지보수 업체 또는 클라우드 업체를 통해 확인 필요

> **설치경로**

설치경로 이동 후 목록을 확인하면 다음과 같은 목록이 출력된다.

```
cd /app/geo
ls -al
```

```
geo@kigamcloud-375:/app/geo$ cd /app/geo && ls -al
total 56
drwxrwxr-x 10 geo geo 4096 Sep 15 18:24 .
drwxr-xr-x  3 geo geo 4096 Jun  2  2022 ..
-rw-rw-r--  1 geo geo  757 Sep 19  2021 .env
drwxrwxr-x  9 geo geo 4096 Oct 20  2020 .git
drwxrwxr-x  2 geo geo 4096 Sep  1 16:39 apps
drwxrwxr-x 10 geo geo 4096 Sep 13 19:03 build
drwxrwxr-x  5 geo geo 4096 Sep 13 19:18 conf
lrwxrwxrwx  1 geo geo   13 Feb 17  2021 data -> /mnt/geo/data
-rwxrwxr-x  1 geo geo 8677 Sep 13 19:43 docker-compose.yml
drwxrwxr-x  4 geo geo 4096 Sep 20  2022 solr
drwxrwxr-x  2 geo geo 4096 Oct 20  2020 solr-lib
drwxrwxr-x  4 geo geo 4096 Nov  1  2021 solr2
drwxrwxr-x  3 geo geo 4096 Aug 31 11:27 static
geo@kigamcloud-375:/app/geo$
```

* Docker Compose 도구를 사용하는 경우 /app/geo 경로에서 실행 필요

폴더/파일 안내

폴더/파일	설명
.env	환경설정 파일
apps	GEO 어플리케이션 실행 파일로 구성된 폴더
build	Docker 컨테이너 이미지 빌드를 위한 설정 파일로 구성된 폴더
conf	어플리케이션, WEB, WAS 등 설정 파일로 구성된 폴더
data	원문파일, DB 데이터, 검색 색인 정보 저장을 위한 폴더
docker-compose.yml	Docker 어플리케이션 정의 및 관리를 위한 설정 파일
solr	Apache Solr 검색엔진 소스 폴더
solr-lib	Apache Solr 검색엔진 라이브러리 폴더
static	정적파일 폴더

서비스 확인

설치경로에서 다음 명령어 실행 시 Docker 에서 동작중인 서비스 및 상태를 확인할 수 있다.

```
cd /app/geo
docker-compose ps
```

```
geo@kigamcloud-375:/app/geo$ docker-compose ps
      Name                       Command               State                    Ports
----------------------------------------------------------------------------------------------------------------
bookstack          /init                               Up      443/tcp, 80/tcp
bookstack-db       /init                               Up      0.0.0.0:33306->3306/tcp,:::33306->3306/tcp
geo_adm_1          catalina.sh run                     Up      8080/tcp
geo_dip_1          catalina.sh run                     Up      8080/tcp
geo_front_1        /script.sh                          Up      0.0.0.0:443->443/tcp,:::443->443/tcp,
                                                               0.0.0.0:80->80/tcp,:::80->80/tcp
geo_irods-db_1     docker-entrypoint.sh postgres       Up      5432/tcp
geo_irods_1        /irods-docker-entrypoint.sh         Up      0.0.0.0:1247->1247/tcp,:::1247->1247/tcp, 1248/tcp,
                                                               20000/tcp, 20001/tcp, 20002/tcp, 20003/tcp, 20004/tcp,
                                                               20005/tcp, 20006/tcp, 20007/tcp, 20008/tcp, 20009/tcp
geo_openapi_1      catalina.sh run                     Up      8080/tcp
geo_repo-db_1      docker-entrypoint.sh postg ...      Up      0.0.0.0:5432->5432/tcp,:::5432->5432/tcp
geo_serv_1         catalina.sh run                     Up      8080/tcp
geo_session-db_1   docker-entrypoint.sh redis ...      Up      6379/tcp
geo_solr_1         docker-entrypoint.sh solr- ...      Up      0.0.0.0:8983->8983/tcp,:::8983->8983/tcp
geo@kigamcloud-375:/app/geo$
```

* "Up"는 실행 중을 나타내며, "Exit"는 종료된 상태를 나타냄

컨테이너(서비스) 안내

컨테이너(서비스)	포트(내부)	설명
front	[redacted]	웹서버 (NGINX)
adm	[redacted]	GEO 관리자 시스템
dip	[redacted]	GEO 사용자 시스템
serv	[redacted]	GEO 관리자 모듈 시스템

컨테이너(서비스)	포트(내부)	설명
openapi	[redacted]	GEO 서비스 OpenAPI 시스템
irods	[redacted]	대규모 데이터 관리 및 저장을 위한 데이터 관리 솔루션 (iRODS)
solr	[redacted]	GEO 검색엔진 (Apache Solr)
repo-db	[redacted]	GEO 데이터베이스 (PostgreSQL)
irods-db	[redacted]	iRODS 데이터베이스 (PostgreSQL)
session-db	[redacted]	GEO 세션 DB (Redis)
bookstack	[redacted]	GEO 문서화 서비스 (BookStack)
bookstack-db	[redacted]	BookStack 데이터베이스 (MariaDB)

서비스 시작

docker-compose.yml 파일에 정의한 서비스를 배포하고 백그라운드에서 실행하며, 이미 시작된 서비스는 제외하고 종료된 서비스만 시작한다.

```
cd /app/geo
docker-compose up -d
```

- docker-compose : Docker Compose 도구를 실행하는 명령
- up : Docker Compose에서 서비스를 시작하는 명령
- -d 또는 --detach : 컨테이너를 백그라운드에서 실행

```
geo@kigamcloud-375:/app/geo$ docker-compose up -d
Starting geo_openapi_1    ... done
Starting bookstack-db     ... done
Starting geo_irods-db_1   ... done
Starting geo_repo-db_1    ... done
Starting geo_front_1      ... done
Starting geo_solr_1       ... done
Starting geo_session-db_1 ... done
Starting bookstack        ... done
Starting geo_dip_1        ... done
Starting geo_irods_1      ... done
Starting geo_adm_1        ... done
Starting geo_serv_1       ... done
geo@kigamcloud-375:/app/geo$
```

docker-compose.yml 파일에 정의한 서비스 중 특정 서비스만을 지정해서 실행할 수 있다.

```
cd /app/geo
docker-compose up -d adm
```

서비스 종료

실행중인 서비스의 컨테이너를 중지(stop)한다.

```
cd /app/geo
docker-compose stop
```

```
geo@kigamcloud-375:/app/geo$ docker-compose stop
Stopping bookstack-db     ... done
Stopping bookstack        ... done
Stopping geo_front_1      ... done
Stopping geo_irods-db_1   ... done
Stopping geo_dip_1        ... done
Stopping geo_irods_1      ... done
Stopping geo_adm_1        ... done
Stopping geo_serv_1       ... done
Stopping geo_session-db_1 ... done
Stopping geo_solr_1       ... done
Stopping geo_openapi_1    ... done
Stopping geo_repo-db_1    ... done
geo@kigamcloud-375:/app/geo$
```

실행중인 서비스의 컨테이너를 중지(stop)하고 삭제(remove)한다.
삭제 시 컨테이너와 함께 생성된 리소스도 삭제되나 관리가 필요한 리소스는 모두 별도 경로에 저장/관리하고 있어 삭제(remove)해도 서비스 운영

에 지장이 없다.

```
cd /app/geo
docker-compose down
```

서비스 로그

Docker에서 실행중인 컨테이너 로그를 실시간으로 확인할 수 있다.

```
cd /app/geo
docker-compose logs -f --tail 10 dip
```

- docker-compose : Docker Compose 도구를 실행하는 명령
- logs : Docker 컨테이너 로그를 출력하는 명령
- -f 또는 --follow : 로그를 실시간으로 보기 위한 옵션
- --tail 10 : 최근 로그 출력을 제한하기 위한 옵션
- app : Docker Compose 파일에서 정의한 서비스의 이름, 이름이 지정된 경우 해당 서비스의 로그만 출력, 이름이 지정되지 않은 경우 전체 서비스의 로그 출력

```
geo@kigamcloud-375:/app/geo$ docker-compose logs -f --tail 10 dip
Attaching to geo_dip_1
dip_1          | DIP :09-17 10:44:53[redAnnotationBeanPostProcessor] WARN- Autowired annotation is not supported on sta
tic fields: private static final java.lang.String kr.re.kigam.geo.web.OpenApiController.wmsUrl
dip_1          | DIP :09-17 10:44:59[k.r.k.geo.ServletInitializer  ] INFO- Started ServletInitializer in 27.799 seconds
 (JVM running for 49.257)
dip_1          | 17-Sep-2023 10:44:59.370 INFO [main] org.apache.coyote.AbstractProtocol.start Starting ProtocolHandler
 ["http-nio-8080"]
dip_1          | 17-Sep-2023 10:44:59.418 INFO [main] org.apache.catalina.startup.Catalina.start Server startup in 4774
4 ms
dip_1          | DIP :09-17 10:45:23[k.r.kigam.geo.SecurityConfig$1] INFO- Redirecting to Target Url: /mgeo/map/main.do
?process=geology_50k
dip_1          | DIP :09-17 10:45:23[nhancerBySpringCGLIB$$f6f0efd0] INFO- 세션 타임아웃 30 분
```

웹서버(Nginx, front) 로그는 장기 저장을 위해 별도로 저장되고 있어 해당 경로에서 확인할 수 있다.

```
cd /app/geo/data/nginx-logs
ls
```

```
geo@kigamcloud-375:/app/geo/data/nginx-logs$ ls
access.log               access.log-20230821.gz    error.log               error.log-20230821.gz
access.log-20230722.gz   access.log-20230822.gz    error.log-20230722.gz   error.log-20230822.gz
access.log-20230723.gz   access.log-20230823.gz    error.log-20230723.gz   error.log-20230823.gz
access.log-20230724.gz   access.log-20230824.gz    error.log-20230724.gz   error.log-20230824.gz
access.log-20230725.gz   access.log-20230825.gz    error.log-20230725.gz   error.log-20230825.gz
access.log-20230726.gz   access.log-20230826.gz    error.log-20230726.gz   error.log-20230826.gz
```

- 로그 파일 생성 : 매일 접속 로그는 하루 단위로 새로운 파일로 생성
- 로그 파일 압축 : 매일 생성된 로그 파일은 익일에 압축되어 저장
- 보관 기간 : 압축된 로그 파일은 최소 3년간 보관되며 장기적인 보안 검토, 감사, 규정 준수 등의 목적으로 활용
- 삭제 정책 : 3년 이상 경과한 로그 파일은 삭제, 오래된 로그 데이터를 관리하고 저장 공간 절약

서비스 성능

Docker에서 실행 중인 컨테이너의 실시간 성능 및 자원 사용 데이터를 확인할 수 있다.

```
docker stats
```

```
CONTAINER ID   NAME                CPU %     MEM USAGE / LIMIT     MEM %     NET I/O             BLOCK I/O           PIDS
351a28167bb6   bookstack-db        0.02%     85.44MiB / 31.36GiB   0.27%     1.02MB / 3.27MB     21.2MB / 1.83MB     20
1c1cfa803e2c   bookstack           0.02%     83.41MiB / 31.36GiB   0.26%     4.06MB / 2.34MB     61.7MB / 656kB      34
40a7548bc1ae   geo_front_1         3.48%     17.06MiB / 31.36GiB   0.05%     949MB / 755MB       14.4MB / 8.19kB     7
7859ab5fb368   geo_irods-db_1      0.00%     26.65MiB / 31.36GiB   0.08%     1.17MB / 3.34MB     16.2MB / 106kB      7
ba1fc76cfe77   geo_dip_1           1.67%     2.798GiB / 31.36GiB   8.92%     223MB / 420MB       175MB / 0B          64
19923a64209d   geo_irods_1         0.16%     59.82MiB / 31.36GiB   0.19%     5.5MB / 17.2MB      60.6MB / 131kB      19
45b1fef9df31   geo_adm_1           2.22%     2.901GiB / 31.36GiB   9.25%     189MB / 215MB       122MB / 0B          47
90526ef8c32d   geo_serv_1          0.81%     2.999GiB / 31.36GiB   9.56%     33.8MB / 12.2MB     219MB / 0B          42
f4ffd8a8d0f6   geo_session-db_1    0.60%     32.95MiB / 31.36GiB   0.10%     94.1MB / 120MB      5.58MB / 163MB      4
73077764c9c9   geo_solr_1          0.50%     582.2MiB / 31.36GiB   1.81%     3.16MB / 32.9MB     93.3MB / 36.9kB     44
f202b36f011d   geo_openapi_1       0.11%     2.459GiB / 31.36GiB   7.84%     47.7kB / 0B         58.4MB / 0B         38
264fbe8dabca   geo_repo-db_1       409.81%   2.489GiB / 31.36GiB   7.94%     42MB / 138MB        4.52GB / 10MB       139
```

- CPU % : 컨테이너의 CPU 사용량 및 사용률 표시
- MEM % : 컨테이너의 메모리 사용량 및 사용률 표시
- NET I/O : 컨테이너의 네트워크 입력 및 출력 데이터 양을 표시
- BLOCK I/O : 컨테이너 블록 디바이스로의 I/O 활동을 표시

시스템 관리 - GDR

GDR 시스템은 Docker를 기반으로 소프트웨어를 관리하고 운영한다.
Docker는 소프트웨어를 컨테이너로 패키징하여 이식성과 확장성을 제공하며, 시스템 관리를 간편하게 해준다.
본 문서는 Docker를 활용하여 시스템을 관리하는 방법에 대한 안내를 제공한다.

서버접속

SSH(Secure Shell) 접속을 통해 GDR 서버에 접속할 수 있으며, 접속 시 다음과 같은 화면이 출력된다.

```
ssh gdr@[redacted] -p 2202
```

```
Last login: Tue Sep 19 18:42:55 2023 from psms.kigam.re.kr
[gdr@gdr0 ~]$
```

- 접속 시 SSH 접속 계정에 대한 비밀번호가 필요하며, 유지보수 업체를 통해 확인 필요

설치경로

설치경로 이동 후 목록을 확인하면 다음과 같은 목록이 출력된다.

```
cd /web/gdr_idr
ls -al
```

```
[gdr@gdr0 gdr_idr]$ cd /web/gdr_idr/
[gdr@gdr0 gdr_idr]$ ls -al
total 24
drwxr-xr-x 5 gdr  gdr    74 Sep 20 10:08 .
drwxr-xr-x 7 root root  114 May 13  2021 ..
drwxr-xr-x 2 gdr  gdr  4096 Dec 15  2022 apps
-rw-r--r-- 1 gdr  gdr  9528 Nov 13  2022 docker-compose.yml
-rw-r--r-- 1 gdr  gdr  1736 Jul  4  2022 .env
drwxrwxr-x 9 gdr  gdr    86 Dec  3  2021 logs
drwxrwxr-x 2 gdr  gdr  4096 Apr 10 14:07 ssl
[gdr@gdr0 gdr_idr]$
```

* Docker Compose 도구를 사용하는 경우 /web/gdr_idr 경로에서 실행 필요

폴더/파일 안내

폴더/파일	설명
.env	환경설정 파일
apps	GDR 어플리케이션 실행 파일로 구성된 폴더
docker-compose.yml	Docker 어플리케이션 정의 및 관리를 위한 설정 파일
ssl	GDR 어플리케이션 SSL 인증서 설정 파일로 구성 된 폴더
logs	어플리케이션 로그 저장을 위한 폴더

데이터 폴더 안내

폴더/파일	설명
/data/gdr_idr/build	Docker 컨테이너 이미지 빌드를 위한 설정 파일로 구성된 폴더
/data/gdr_idr/conf	어플리케이션, WEB, WAS 등 설정 파일로 구성된 폴더

폴더/파일	설명
/data/gdr_idr/data	원문파일, DB 데이터, 검색 색인 정보 저장을 위한 폴더
/db/postgresql	DB 데이터 정보 저장을 위한 폴더
/data/gdr_idr/solr	Apache Solr 검색엔진 소스 폴더
/data/gdr_idr/static	정적파일 폴더

서비스 확인

설치경로에서 다음 명령어 실행 시 Docker 에서 동작중인 서비스 및 상태를 확인할 수 있다.

```
[gdr@gdr0 postgresql]$ cd /web/gdr_idr/
[gdr@gdr0 gdr_idr]$ docker-compose ps
        Name                          Command               State                 Ports
-----------------------------------------------------------------------------------------------
gdr_idr_adm_1          catalina.sh run                  Up      8080/tcp
gdr_idr_auth_1         catalina.sh run                  Up      8080/tcp
gdr_idr_dip_1          catalina.sh run                  Up      8080/tcp
gdr_idr_front_1        /docker-entrypoint.sh ngin ...   Up      0.0.0.0:443->443/tcp,
                                                                0.0.0.0:80->80/tcp
gdr_idr_kigam-adm_1    catalina.sh run                  Up      8080/tcp
gdr_idr_repo-db_1      docker-entrypoint.sh postgres    Up      0.0.0.0:5432->5432/tcp
gdr_idr_serv_1         catalina.sh run                  Up      8080/tcp
gdr_idr_session-db_1   docker-entrypoint.sh redis ...   Up      6379/tcp
gdr_idr_sip_1          catalina.sh run                  Up      8080/tcp
gdr_idr_solr_1         docker-entrypoint.sh solr- ...   Up      0.0.0.0:8983->8983/tcp
[gdr@gdr0 gdr_idr]$
```
*

"Up"는 실행 중을 나타내며, "Exit"는 종료된 상태를 나타냄

컨테이너(서비스) 안내

컨테이너(서비스)	포트(내부)	설명
front	[redacted]	웹서버 (NGINX)
adm	[redacted]	GDR 관리자 시스템
kigam-adm	[redacted]	GDR 전용 관리자 시스템
auth	[redacted]	GDR 사용자 인증 시스템
sip	[redacted]	GDR 연구데이터 제출 시스템
dip	[redacted]	GDR 사용자 시스템
serv	[redacted]	GDR 관리자 모듈 시스템
solr	[redacted]	GDR 검색엔진 (Apache Solr)
repo-db	[redacted]	GDR 데이터베이스 (PostgreSQL)
session-db	[redacted]	GDR 세션 DB (Redis)

서비스 시작

docker-compose.yml 파일에 정의한 서비스를 배포하고 백그라운드에서 실행하며, 이미 시작된 서비스는 제외하고 종료된 서비스만 시작한다.

```
cd /web/gdr_idr
docker-compose up -d
```

- docker-compose : Docker Compose 도구를 실행하는 명령
- up : Docker Compose에서 서비스를 시작하는 명령
- -d 또는 --detach : 컨테이너를 백그라운드에서 실행

```
[gdr@gdr0 gdr_idr]$ docker-compose up -d
Starting gdr_idr_front_1      ... done
Starting gdr_idr_solr_1       ... done
Starting gdr_idr_session-db_1 ... done
Starting gdr_idr_repo-db_1    ... done
Starting gdr_idr_kigam-adm_1  ... done
Starting gdr_idr_adm_1        ... done
Starting gdr_idr_sip_1        ... done
Starting gdr_idr_dip_1        ... done
Starting gdr_idr_auth_1       ... done
Starting gdr_idr_serv_1       ... done
[gdr@gdr0 gdr_idr]$
```

docker-compose.yml 파일에 정의한 서비스 중 특정 서비스만을 지정해서 실행할 수 있다.

```
cd /web/gdr_idr
docker-compose up -d adm
```

서비스 종료

실행중인 서비스의 컨테이너를 중지(stop)한다.

```
cd /web/gdr_idr
docker-compose stop
```

```
[gdr@gdr0 gdr_idr]$ docker-compose stop
Stopping gdr_idr_sip_1          ... done
Stopping gdr_idr_kigam-adm_1    ... done
Stopping gdr_idr_dip_1          ... done
Stopping gdr_idr_serv_1         ... done
Stopping gdr_idr_adm_1          ... done
Stopping gdr_idr_auth_1         ... done
Stopping gdr_idr_solr_1         ... done
Stopping gdr_idr_front_1        ... done
Stopping gdr_idr_repo-db_1      ... done
Stopping gdr_idr_session-db_1 ... done
[gdr@gdr0 gdr_idr]$
```

실행중인 서비스의 컨테이너를 중지(stop)하고 삭제(remove)한다.
삭제 시 컨테이너와 함께 생성된 리소스도 삭제되나 관리가 필요한 리소스는 모두 별도 경로에 저장/관리하고 있어 삭제(remove)해도 서비스 운영에 지장이 없다.

```
cd /web/gdr_idr
docker-compose down
```

서비스 로그

Docker에서 실행중인 컨테이너 로그를 실시간으로 확인할 수 있다.

```
cd /web/gdr_idr
docker-compose logs -f --tail 10 dip
```

- docker-compose : Docker Compose 도구를 실행하는 명령
- logs : Docker 컨테이너 로그를 출력하는 명령
- -f 또는 --follow : 로그를 실시간으로 보기 위한 옵션
- --tail 10 : 최근 로그 출력을 제한하기 위한 옵션
- app : Docker Compose 파일에서 정의한 서비스의 이름, 이름이 지정된 경우 해당 서비스의 로그만 출력, 이름이 지정되지 않은 경우 전체 서비스의 로그 출력

```
[gdr@gdr0 gdr_idr]$ docker-compose logs -f --tail 100 dip
Attaching to gdr_idr_dip_1
dip_1          | DIP :09-20 09:03:57[kr.re.kisti.idr.client        ] INFO- GET /kigam-adm/api/itemAdd/b7d7b944-
 [17ms]
dip_1          | DIP :09-20 09:03:57[kr.re.kisti.idr.client        ] INFO- GET /kigam-adm/api/itemAdd/9301483b-
 [165ms]
dip_1          | DIP :09-20 09:03:57[kr.re.kisti.idr.client        ] INFO- GET /kigam-adm/api/itemAdd/c5af24d0-
 [17ms]
dip_1          | DIP :09-20 09:03:57[kr.re.kisti.idr.client        ] INFO- GET /kigam-adm/api/itemAdd/94b417fc-
 [17ms]
dip_1          | DIP :09-20 09:03:57[kr.re.kisti.idr.client        ] INFO- GET /kigam-adm/api/itemAdd/027fflac-
 [43ms]
dip_1          | DIP :09-20 09:03:57[kr.re.kisti.idr.client        ] INFO- GET /kigam-adm/api/itemAdd/d6399a77-
 [20ms]
dip_1          | DIP :09-20 09:03:57[kr.re.kisti.idr.client        ] INFO- GET /kigam-adm/api/itemAdd/100d8255-
 [50ms]
dip_1          | DIP :09-20 09:03:57[kr.re.kisti.idr.client        ] INFO- GET /kigam-adm/api/itemAdd/a2bdla78-
 [22ms]
```

웹서버(Nginx, front) 및 각 시스템(adm, kigam-adm, auth, dip, sip, serv) 로그는 장기 저장을 위해 별도로 저장되고 있어 해당 경로에서 확인할 수 있다.

```
cd /web/gdr_idr/logs
ls
cd front
```

```
ls
```

```
[gdr@gdr0 front]$ cd /web/gdr_idr/logs/
[gdr@gdr0 logs]$ ls
adm  auth  dip  front  kigam-adm  serv  sip
[gdr@gdr0 logs]$ cd front/
[gdr@gdr0 front]$ ls
access.log  error.log
```

서비스 성능

Docker에서 실행 중인 컨테이너의 실시간 성능 및 자원 사용 데이터를 확인할 수 있다.

```
docker stats
```

```
CONTAINER ID   NAME                     CPU %     MEM USAGE / LIMIT     MEM %     NET I/O           BLOCK I/O         PIDS
42e775adb9c1   gdr_idr_sip_1            0.14%     1.078GiB / 62.8GiB    1.72%     835MB / 868MB     65.1GB / 53.2kB   61
7de0ba91d2c7   gdr_idr_kigam-adm_1      0.12%     1.357GiB / 62.8GiB    2.16%     20.1GB / 15.6GB   84GB / 860kB      62
ca1de994c968   gdr_idr_dip_1            88.66%    1.074GiB / 62.8GiB    1.71%     12.6GB / 1.94GB   71.3GB / 131kB    61
e4ecd1ee217d   gdr_idr_serv_1           106.70%   1.632GiB / 62.8GiB    2.60%     514MB / 292MB     48.7GB / 4.18MB   68
01f0ce47fc2e   gdr_idr_adm_1            0.13%     796.9MiB / 62.8GiB    1.24%     5.35GB / 4.54GB   75.5GB / 451kB    63
8ee3e6557663   gdr_idr_auth_1           0.13%     1.175GiB / 62.8GiB    1.87%     445MB / 232MB     40.5GB / 516kB    58
27992c8c9c85   gdr_idr_solr_1           0.14%     262.4MiB / 62.8GiB    0.41%     23.3MB / 22.4MB   4.4GB / 538MB     54
68872aab0f23   gdr_idr_front_1          0.00%     2.539MiB / 62.8GiB    0.00%     1.22GB / 752MB    33.5MB / 0B       2
86c0e4cbefb8   gdr_idr_repo-db_1        0.00%     479.2MiB / 62.8GiB    0.75%     4.8GB / 21.4GB    23.6GB / 995MB    87
7cc13b7850d2   gdr_idr_session-db_1     0.11%     2.902MiB / 62.8GiB    0.00%     2.95GB / 3.81GB   43.6MB / 1.95GB   4
```

- CPU % : 컨테이너의 CPU 사용량 및 사용률 표시
- MEM % : 컨테이너의 메모리 사용량 및 사용률 표시
- NET I/O : 컨테이너의 네트워크 입력 및 출력 데이터 양을 표시
- BLOCK I/O : 컨테이너 블록 디바이스로의 I/O 활동을 표시

시스템 백업 및 복구 - GEO

GEO 시스템은 클라우드 기반 서비스로 클라우드 업체에서 제공하는 백업 서비스를 이용합니다.

백업대상

서버	경로	비고
GEO (115.68.202.95)	/app/[redacted]	GEO 어플리케이션 설정 및 소스
	/mng/[redacted]	GEO DB, 데이터파일, 로그
KOMM (115.68.202.161)	/home/[redacted]	KOMM WAS 설정 및 로그
	/home/[redacted]	KOMM GEO 데이터
	/home/[redacted]	KOMM 데이터파일
KGRIS (115.68.202.161)	/home/[redacted]	KGRIS 소스
	/home/[redacted]	KGRIS GEO 데이터
	/home/[redacted]	KGRIS WAS 설정 및 로그
	[redacted]	KGRIS DB (Mysql)
	[redacted]	KGRIS DB (Postgresql)

백업

구분	백업 정책
OS	백업 대상 서버 1대 단위로 백업 수행 백업 스케줄, 대상, 보관, 주기등 백업스크립트 작성 지원 주간단위 OS 영역 백업
Block Storage	백업 대상을 기준으로 1일 1회 증분 백업 일주일 보관 월 1회 Full Backup

복구

이벤트 발생 시 별도의 클라우드 저장소로 복구하며, 이를 사용하여 복구 작업 수행

시스템 백업 및 복구 - GDR

GDR 시스템은 온프레미스(사내 서버 및 네트워크) 기반 서비스로 백업 솔루션을 사용하여 네트워크 백업을 수행합니다.

백업대상

서버	경로	비고
GDR ([redacted])	/web	GDR 어플리케이션 설정 및 소스, 로그
	/data	GDR 데이터파일
	/db	GDR DB

백업

구분	백업 정책
OS, /web, /db	백업대상서버 1 대 단위로 백업수행 백업주기 : 일간 백업스케줄 : 일1 회 증분백업, 1 주1 회 전체백업 보관주기 : 5 개월 백업방식 : 백업솔루션사용한 네트웍백업
/data	백업대상서버 1 대 단위로 백업수행 백업주기 : 일간 백업스케줄 : 일1 회 증분백업 , 2 주1 회 전체백업 보관주기 : 5 개월 백업방식 : 백업솔루션사용한 네트웍백업

복구

이벤트 발생 시 저장소로 복구하며, 이를 사용하여 복구 작업 수행

시스템 문의처

한국지질자원연구원 지질자원데이터센터

- 담당업무 : GEO, GDR 관리 총괄
- 담당자 : 한종규 책임연구원(센터장), 042-868-3297, jghan@kigam.re.kr
- 담당자 : 한수연 선임연구원, 042-868-3182, syhan@kigam.re.kr

한국지질자원연구원 지식정보실

- 담당업무 : 지질자원연구원 PMS, 결제시스템 등 내부시스템
- 담당자 : 강신명 실장, 042-868-3490, kangsa@kigam.re.kr

한국지질자원연구원 안전보안실

- 담당업무 : 개인정보보호, 개인정보위탁사업 업체 현장점검
- 담당자 : 전상준 책임기술원, 042-868-3737, sjjeon@kigam.re.kr

유지보수업체 - 아르고넷

- 담당업무 : GEO, GDR 유지보수
- 담당자 : 이규선 책임, [redacted]
- 담당자 : 채혜인 선임, [redacted]

유지보수업체 - 지토피아시스템

- 담당업무 : 지질자원 주제도 및 GEO 하위시스템
- 담당자 : 송지명 차장, [redacted]

KoreaV 공공클라우드

- 담당업무 : 클라우드 인프라 관리
- 담당자 : 서상현, [redacted]
  * 담당자 변경이 잦음