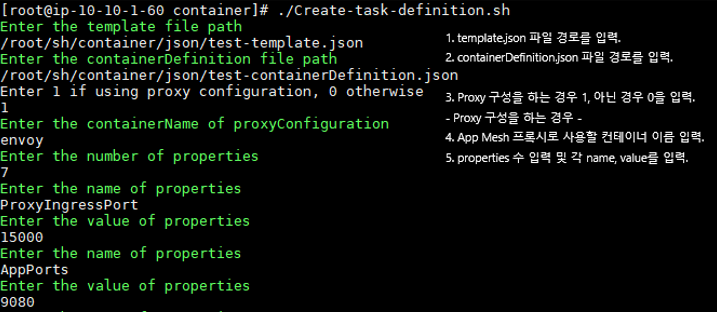
**- ECS 작업 정의 생성 -**

**0. ECS 작업 정의 생성관련 유의 사항**

* ECS-task-definition-template.sh, ECS-task-definition-container.sh, ECS-Create-task-definition.sh 총 3개의 파일로 이루어져 있습니다.
  + ECS-task-definition-container.sh는 작업 정의에 들어가는 파라미터 중 **컨테이너 정의** json 파일을 생성합니다.
  + ECS-task-definition-template.sh는 작업 정의에 들어가는 파라미터 중 컨테이너 정의를 **제외한 파라미터**들을 기입하여 json 파일을 생성합니다.
* 각 쉘 스크립트에선 jq 명령어를 사용하고 있으니 아래 커맨드를 통해 해당 쉘 스크립트를 구동하는 인스턴스 내에 jq를 설치해야 합니다.
  + $ sudo curl -L https://github.com/stedolan/jq/releases/download/jq-1.6/jq-linux64 -o /usr/local/bin/jq  
    $ sudo chmod a+x /usr/local/bin/jq
* aws configure 명령어를 통해 access key, secret key를 등록해야 합니다.
* 쉘 스크립트가 실행되는 경로 이하의 ‘**json**’ 디렉토리에 입력한 파라미터 값에 맞게 json 파일이 생성되도록 코드가 작성되어 있기 때문에, 해당 쉘 스크립트가 존재하는 위치에서 $ mkdir json로 디렉토리를 생성해야 합니다.

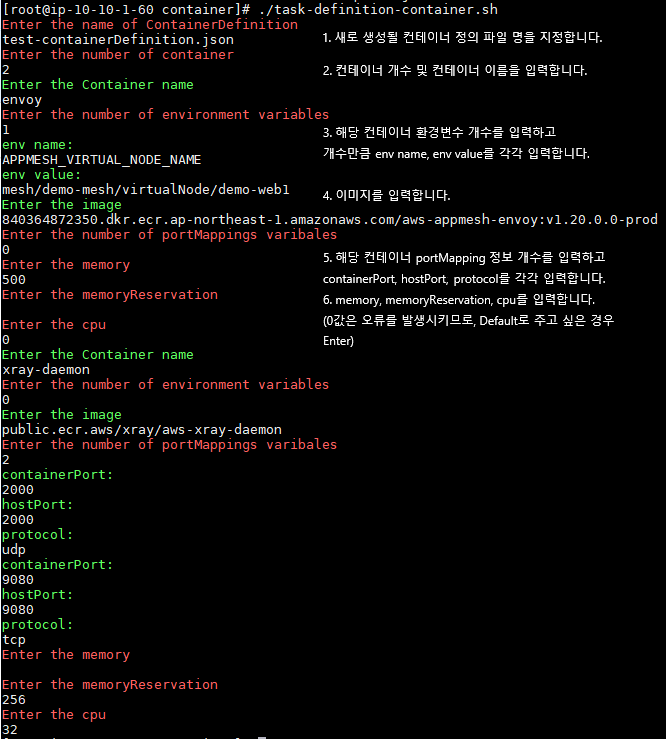
**1. ECS 작업 정의 생성**

* 명령어: ./ECS-Create-task-definition.sh  
  주어진 ECS-task-definition-container.sh과 ECS-task-definition-template.sh로 생성된 json 파일이 필요합니다. (경로를 입력)
* 스크립트 실행 과정

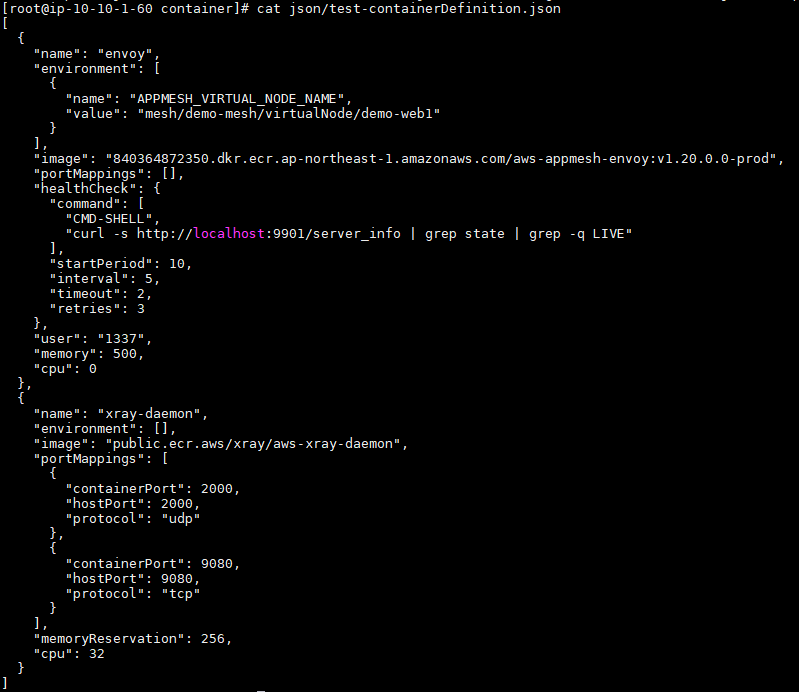


**2-1. ECS 작업 정의 – 컨테이너 정의 생성**

* 스탠다드 컨테이너 정의 파라미터 사용  
  (<https://docs.aws.amazon.com/ko_kr/AmazonECS/latest/developerguide/task_definition_parameters.html#standard_container_definition_params>)
* 명령어: ./ECS-task-definition-container.sh
* 파라미터
  + **name, image,** portMappings, environment, cpu, memory, memoryReservation
* 스크립트 실행 과정

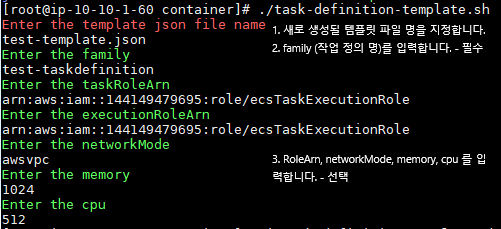


* 스크립트 결과 json파일

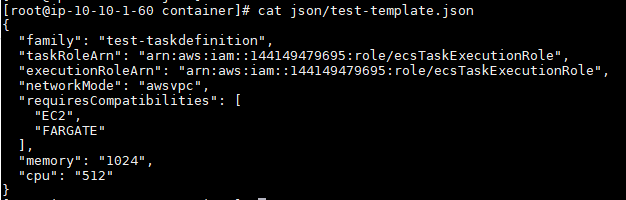


**2-2. ECS 작업 정의 - 템플릿 생성**

* 명령어: ./ECS-task-definition-template.sh
* 파라미터
  + **family(작업 정의 이름),** taskRoleArn, executionRoleArn, networkMode, memory, cpu
* 스크립트 실행 과정



* 스크립트 결과 json파일



* 작업 정의의 유효성을 검사하는 파라미터인 requiresCompatibilities는 EC2, FARGATE, EXTERNAL을 사용할 수 있는데, 임의로 EC2, FARGATE를 지정하였음.

**3. ECS 작업 정의 생성 부록**

* ECS 작업 정의에 파라미터의 경우 아래 공식 문서 내용을 참고하시길 바랍니다. (<https://docs.aws.amazon.com/ko_kr/AmazonECS/latest/developerguide/task_definition_parameters.html>)

텍스트이(가) 표시된 사진

자동 생성된 설명

**- ECS 서비스 생성 -**

**0. ECS 서비스 생성관련 유의 사항**

* ECS-loadbalancer.sh, ECS-awsvpcConfiguration.sh, ECS-discovery.sh, ECS-Create-service.sh 총 4개의 파일로 이루어져 있습니다.
* 각 쉘 스크립트에선 jq 명령어를 사용하고 있으니 아래 커맨드를 통해 해당 쉘 스크립트를 구동하는 인스턴스 내에 jq를 설치해야 합니다.
  + $ sudo curl -L https://github.com/stedolan/jq/releases/download/jq-1.6/jq-linux64 -o /usr/local/bin/jq  
    $ sudo chmod a+x /usr/local/bin/jq
* aws configure 명령어를 통해 access key, secret key를 등록해야 합니다.
* 쉘 스크립트가 실행되는 경로 이하의 ‘**json**’ 디렉토리에 입력한 파라미터 값에 맞게 json 파일이 생성되도록 코드가 작성되어 있기 때문에, 해당 쉘 스크립트가 존재하는 위치에서 $mkdir json로 디렉토리를 생성해야 합니다.

**1. ECS** **서비스 생성**

* 명령어: ./ECS-Create-service.sh  
  필수적으로 주어진 ECS-awsvpcConfiguration.sh, ECS-discovery.sh로 생성된 json 파일이 필요합니다. 추가적으로 로드밸런서를 등록하는 경우 ECS-loadbalancer.sh로 생성된 json 파일이 필요합니다. (아래 스크립트 실행 과정 참고)
* 스크립트 실행 과정

텍스트이(가) 표시된 사진

자동 생성된 설명

* 위의 과정을 통해 생성된 서비스

텍스트이(가) 표시된 사진

자동 생성된 설명

* 위의 과정을 통해 생성된 작업

텍스트이(가) 표시된 사진

자동 생성된 설명

**2-1. ECS-awsvpcConfiguration.json 생성**

* awsvpc 네트워크 모드를 사용하는 작업 정의에 필요한 파일.
* 명령어: ./ECS-awsvpcConfiguration.sh
* 파라미터
  + subnets, securityGroups, assignPublicIp (default: DISABLED)
* 스크립트 실행 과정

텍스트이(가) 표시된 사진

자동 생성된 설명

* 스크립트 결과 json파일

텍스트이(가) 표시된 사진

자동 생성된 설명

**2-2. ECS-discovery.json 생성**

* 명령어: ./ECS-discovery.sh
* 파라미터
  + registryArn
* 스크립트 실행 과정

텍스트이(가) 표시된 사진

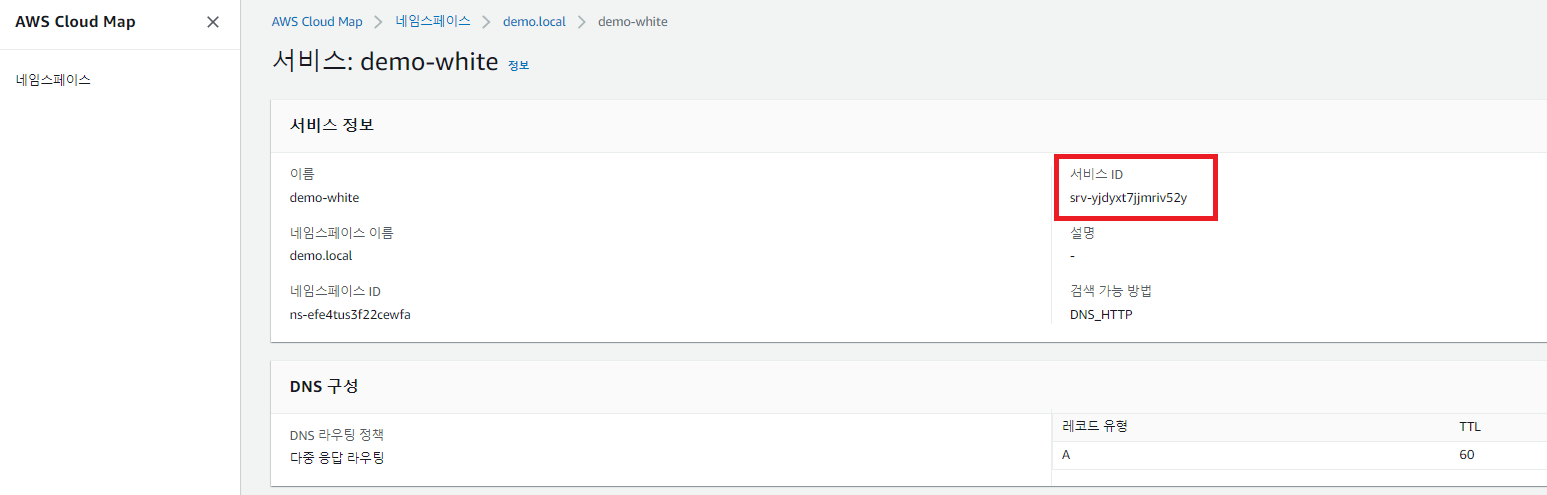
자동 생성된 설명

* 스크립트 결과 json파일

텍스트이(가) 표시된 사진

자동 생성된 설명

* Service ID of Cloud Map 확인 방법



**2-3. ECS-loadbalancer.json 생성**

* 명령어: ./ECS-loadbalancer.sh
* 파라미터
  + targetGroupArn(NLB, ALB), containerName, containerPort
    - ALB, NLB를 사용하는 경우 targetGroupArn 파라미터를 사용하고,  
      CLB를 사용하는 경우 loadBalancerName 파라미터를 사용합니다.
* 스크립트 실행 과정

텍스트이(가) 표시된 사진

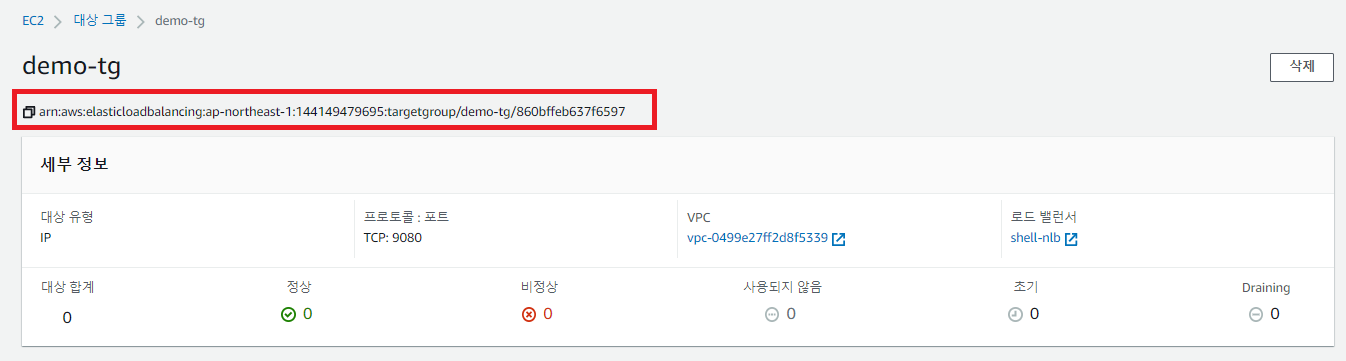
자동 생성된 설명

* 스크립트 결과 json파일

텍스트이(가) 표시된 사진

자동 생성된 설명

* targetGroupArn 확인 방법



**3. ECS 서비스 생성 부록**

텍스트이(가) 표시된 사진

자동 생성된 설명

* ECS 서비스 파라미터의 경우 아래 공식 문서 내용을 참고하시길 바랍니다. (<https://docs.aws.amazon.com/cli/latest/reference/ecs/create-service.html>)