

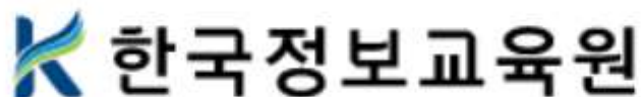
REPORT



- 클라우드 자동화 기반의 애플리케이션 개발 및 관리자 양성 -



과 목 명	클라우드 자동화 기반의 애플리케이션 개발 및 관리자 양성
주 제	Aws 기반 이미지 저장소 웹앱
팀 명	Edelwiess
팀 장	박선재
팀 원	조성연, 오문성, 김동준
담 임	김범택 선생님
제 출 일	2022년 04월 13일



목차

■ 네트워크 구축	----- 1
VPC	
서브넷	
인터넷	
게이트웨이	
라우팅 테이블 생성	
■ EC2 웹 서버 구축	----- 2
인스턴스 생성	
푸티 접속	
아파치 설치	
추가 기능 설정	
삭제 하는 방법	
■ 웹 서버 정적 배포	----- 3
웹 서버 정적 배포	
파일 질라	
■ AWS 로드밸런서 구성하기(ELB)	----- 4
로드밸런서란	
ELB 생성하기	
EC2 보안그룹 변경	
ELB를 통해 EC2 접속 확인하기	
ELB의 대상 인스턴스 확인	
■ AWS 데이터베이스 생성하기(S3)	----- 5
RDS	
Bucket	

[VPC 생성하기 + 퍼블릭 서브넷 2개 프라이빗 서브넷 2)]

VPC 설정

생성할 리소스 정보
VPC 리소스만 생성 또는 VPC, 서브넷 등을 생성

☐ VPC만

☒ VPC, 서브넷 등

이름 태그 자동 생성 정보
이름 태그의 값을 입력합니다. 이 값은 VPC의 모든 리소스에 대한 이름 태그를 자동으로 생성하는 데 사용됩니다.

☒ 자동 생성

Edelwies

IPv4 CIDR 블록 정보
CIDR 표기법을 사용하여 VPC의 시작 IP와 크기를 결정합니다.

10.0.0.0/16 65,536 IPs

IPv6 CIDR 블록 정보

☒ IPv6 CIDR 블록 없음

☐ Amazon 제공 IPv6 CIDR 블록

테넌시 정보

기본값

가용 영역(AZ) 정보
서브넷을 프로비저닝할 AZ 수를 선택합니다.고가용성을 위해서는 최소 2개 이상의 AZ를 사용하는 것이 좋습니다.

1 2 3

▼ AZ 사용자 지정

▼ AZ 사용자 지정

첫 번째 가용 영역

ap-northeast-2a

두 번째 가용 영역

ap-northeast-2c

퍼블릭 서브넷 수 정보
VPC에 추가할 퍼블릭 서브넷 수입니다. 인터넷을 통해 공개적으로 액세스할 수 있어야 하는 웹 애플리케이션에는 퍼블릭 서브넷을 사용합니다.

0 2

▶ 퍼블릭 서브넷 CIDR 블록 사용자 지정

프라이빗 서브넷 수 정보
VPC에 추가할 프라이빗 서브넷 수입니다. 프라이빗 서브넷을 사용하여 퍼블릭 액세스가 필요 없는 백엔드 리소스를 보호합니다.

0 2 4

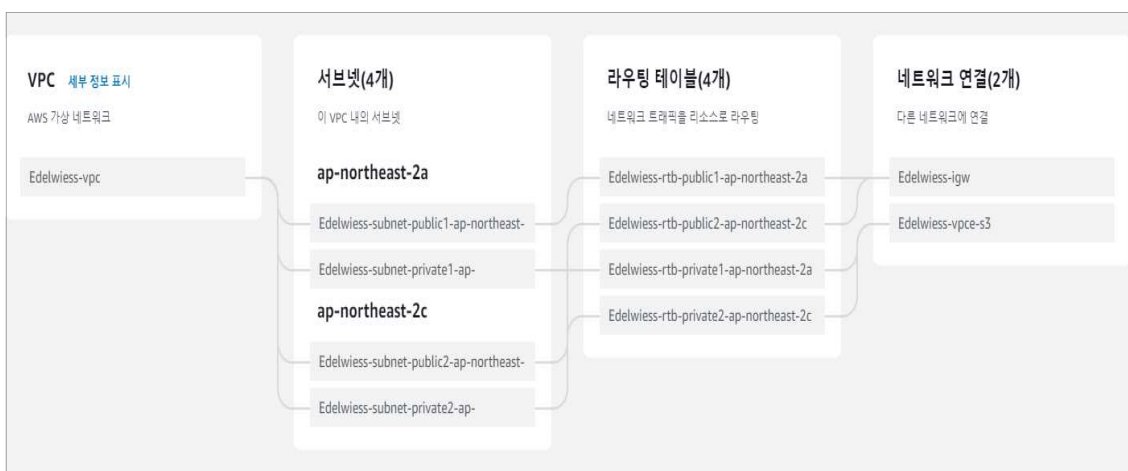
▶ 프라이빗 서브넷 CIDR 블록 사용자 지정

NAT 게이트웨이(\$\$) 정보
NAT 게이트웨이를 생성할 가용 영역(AZ) 수를 선택합니다.

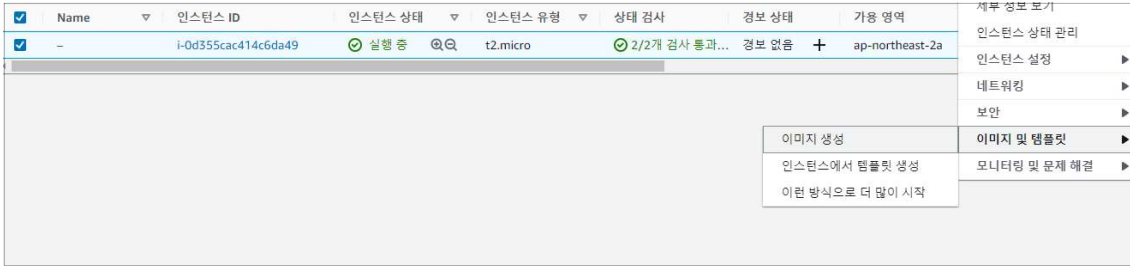
없음 1개의 AZ에서 AZ당 1개

VPC 엔드포인트 정보
엔드포인트는 VPC에서 S3에 직접 액세스하여 NAT 게이트웨이 요금을 줄이고 보안을 강화할 수 있습니다. 기본적으로 모든 액세스 정책이 사용됩니다. 언제든지 이 정책을 사용자 지정할 수 있습니다.

없음 S3 게이트웨이



[웹 서비스를 위한 구성요소 설치 후 인스턴스 이미지 생성]



인스턴스 ID
i-0d355cac414c6da49

이미지 이름
edewiess-image

최대 127자. 생성 후에는 수정할 수 없습니다.

이미지 설명 - 선택 사항
edewiess-image

최대 255자

재부팅 안 함
☐ 활성화

인스턴스 볼륨

볼륨 유형	디바이스	스냅샷	크기	볼륨 유형	IOPS	처리량	종료 시 삭제	암호화됨
EBS	/dev/s...	볼륨에서 새 스냅샷 생성	8	EBS 전용 SSD - gp2	100		<input checked="" type="checkbox"/> 활성화	<input type="checkbox"/> 활성화

볼륨 추가

이미지 생성 프로세스 중에 Amazon EC2는 위의 각 볼륨의 스냅샷을 생성합니다.

태그 - 선택 사항
태그는 사용자가 AWS 리소스에 할당하는 레이블입니다. 각 태그는 키와 값(선택 사항)으로 구성됩니다. 태그를 사용하여 리소스를 검색 및 필터링하거나 AWS 비용을 추적할 수 있습니다.

☒ 이미지와 스냅샷을 함께 태그 지정
이미지와 스냅샷에 동일한 태그를 지정합니다.

☐ 이미지와 스냅샷을 별도로 태그 지정
이미지와 스냅샷에 다른 태그를 지정합니다.

리소스에 연결된 태그가 없습니다.

태그 추가

태그를 50개 더 추가할 수 있습니다.

[생성한 이미지로 2번째 인스턴스를 배포]

Amazon Machine Images(AMI) (1/1) 정보

내 소유 검색

Name	AMI ID	AMI 이름	원본	소유자	표시 여부	상태
-	ami-05f658d5304e7e470	edewiess-image	189971110175/edewiess-image	189971110175	프라이빗	사용 가능

그림 9 Edelwiess vpc, public c zone에 인스턴스를 생성한다

단계 3: 인스턴스 세부 정보 구성

요구 사항에 적합하게 인스턴스를 구성합니다. 동일한 AMI의 여러 인스턴스를 시작하고 스팟 인스턴스를 요청하여 보다 저렴한 요금을 활용하며 인스턴스에 액세스 관리 역할을 할당하는 등 다양한 기능을 사용할 수 있습니다.

인스턴스 개수 1 Auto Scaling 그룹 시작

구매 옵션 ☐ 스팟 인스턴스 요청

네트워크 vpc-0148d1bf3ed0bd2ee | Edelwiess-vpc 새 VPC 생성

서브넷 subnet-093791125898d9411 | Edelwiess-subnet-priv 새 서브넷 생성
subnet-093791125898d9411 | Edelwiess-subnet-private1-ap-northeast-2a | ap-northeast-2a
subnet-029b8391469b9add7 | Edelwiess-subnet-public2-ap-northeast-2c | ap-northeast-2c
subnet-0725b73467e572f0d | Edelwiess-subnet-public1-ap-northeast-2a | ap-northeast-2a
subnet-0472a5a1a4d823546 | Edelwiess-subnet-private2-ap-northeast-2c | ap-northeast-2c

퍼블릭 IP 자동 할당

호스트 이름 유형 서버넷 사용 설정 (IP 이름)

DNS Hostname ☒ Enable IP name (A record) DNS requests
☒ 리소스 기반 IPv4(A 레코드) DNS 요청 활성화
☐ 리소스 기반 IPv6(AAAA 레코드) DNS 요청 활성화

기존에 인스턴스1 과 연결했던 보안그룹을 선택하여 적용한다

단계 6: 보안 그룹 구성

보안 그룹은 인스턴스에 대한 트래픽을 제어하는 방화벽 규칙 세트입니다. 이 페이지에서는 특정 트래픽을 인스턴스에 도달하도록 허용할 규칙을 추가할 수 있습니다. 예를 들면 웹 서버를 설정 HTTPS 트래픽에 대한 무제한 액세스를 허용하는 규칙을 추가합니다. 새 보안 그룹을 생성하거나 아래에 나와 있는 기존 보안 그룹 중에서 선택할 수 있습니다. Amazon EC2 보안 그룹에 대해

보안 그룹 할당: ☐ 새 보안 그룹 생성
☒ 기존 보안 그룹 선택

보안 그룹 ID	이름	설명
<input type="checkbox"/> sg-075517ef1a4b74e49	default	default VPC security group
<input checked="" type="checkbox"/> sg-08cfd18d3ccba5cd	edelwiess-SG	edelwiess-SG

[배포한 인스턴스 구성]

#ubuntu로 구현 (순서대로 입력)

root패스워드 설정 : `sudo passwd root`

root로 로그인 : `su root`

apt update : `apt-get update` , `apt-get upgrade`

jdk 설치 : `apt-get install openjdk-8-jre-headless` , `apt-get install openjdk-8-jdk`

설치된 java 버전 확인 : `java -version` , `javac -version`

java 홈 위치 확인 : `which java`

톰캣 설치(tomcat9) : `apt-get install tomcat9`

톰캣 위치 확인 : `find / -name tomcat9`

톰캣 실행 : `service tomcat9 start`

포트가 열려있는지 확인하기

`apt install net-tools ->` 확인할 수 있는 툴 설치

`netstat -ntl ->` 명령어로 8080 포트 확인

mysql 설치(가장 최신 버전이 설치 됨)

`apt-get install mysql-server`

`apt-get install mysql-client`

mysql 버전 확인 : `mysql -v`

jdbc드라이버 설치 : `apt-get install libmysql-java`

`/usr/share/java/mysql-connector-java.jar` 위치에 connector가 설치된다

jdk와 tomcat9을 jdbc드라이버로 연결해주자

`ln -s /usr/share/java/mysql-connector-java.jar`

`/usr/share/tomcat7/lib/mysql-connector-java.jar`

톰캣 재시작

`service tomcat9 restart`

[트래픽 부하 분산을 위한 로드 밸런싱 구현]

기본 구성

로드 밸런서 이름

이름은 AWS 계정 내에서 고유해야 하며 로드 밸런서가 생성된 후에는 변경할 수 없습니다.

edelwiess-LB

하이픈을 포함하여 최대 32자의 영숫자 문자가 허용되지만 이름은 하이픈으로 시작하거나 끝낼 수 없습니다.

제도 정보

로드 밸런서를 생성한 후에는 스키마를 변경할 수 없습니다.

인터넷 연결

인터넷 연결 로드 밸런서는 인터넷을 통해 클라이언트의 요청을 대상으로 라우팅합니다. 퍼블릭 서브넷이 필요합니다. [더 알아보기](#)

내부의

내부 로드 밸런서는 개인 IP 주소를 사용하여 클라이언트에서 대상으로 요청을 라우팅합니다.

IP 주소 유형 정보

서브넷에서 사용하는 IP 주소 유형을 선택합니다.

IPv4

내부 로드 밸런서에 권장됩니다.

듀얼스택

IPv4 및 IPv6 주소를 포함합니다.

그림 11 외부에 웹서버를 배포하기 위해 퍼블릭 서브넷이 필요한 인터넷 연결 가능으로 옵션을 선택, 주소 유형은 IPv4로 선택

네트워크 매핑 정보

로드 밸런서는 IP 주소 설정에 따라 선택한 서브넷의 대상으로 트래픽을 라우팅합니다.

VPC 정보

대상에 대한 가상 사설 클라우드(VPC)를 선택합니다. 인터넷 게이트웨이가 있는 VPC만 선택할 수 있습니다. 로드 밸런서를 생성한 후에는 선택한 VPC를 변경할 수 없습니다. 대상에 대한 VPC를 확인하려면 [대상 그룹을 확인하십시오](#).

Edelwiess-vpc

vpc-0148d1bf3ed0bd2ee

IPv4: 10.0.0.0/16

매핑 정보

각 영역에 대해 하나 이상의 가용 영역과 서브넷을 선택합니다. 가용 영역을 두 개 이상 선택하는 것이 좋습니다. 로드 밸런서는 선택한 가용 영역의 대상에만 트래픽을 라우팅합니다. 로드 밸런서 또는 VPC에서 지원하지 않는 영역은 선택할 수 없습니다. 로드 밸런서가 생성되면 서브넷을 추가할 수 있지만 제거할 수는 없습니다.

☒ ap-북동쪽-2a

서브넷

서브넷 0725b73467e572f0d Edelwiess-subnet-public1-ap-northeast-2a ▼

IPv4 설정

AWS에서 할당

☒ ap-북동쪽-2c

서브넷

서브넷 029b8391469b9add7 Edelwiess-subnet-public2-ap-northeast-2c ▼

IPv4 설정

AWS에서 할당

그림 12 로드밸런싱을 사용할 네트워크를 매핑해 준다. 인스턴스1이 속해있는 A zone과 인스턴스2가 속해있는 C zone을 매핑

[로드밸런서의 보안 그룹]

기본 세부 정보

보안 그룹 이름 정보
LB-group

설명 정보
LB-group

VPC 정보
vpc-0148d1bf3ed0bd2ee

인바운드 규칙 정보

보안 그룹 규칙 ID	유형 정보	프로토콜 정보	포트 범위 정보	소스 정보	설명 - 선택 사항 정보
sgr-0dc77e68ce5829067	MySQL/Aurora	TCP	3306	사용자 ...	
sgr-0caf70c0e94a3a538	HTTP	TCP	80	사용자 ...	

규칙 추가

취소 변경 사항 미리 보기 규칙 저장

그림 14 사용자가 접속하는 80포트와 DB로의 값을 전달 받을 3306포트를 열어준다.

리스너 및 라우팅 정보

리스너는 구성한 프로토콜과 포트를 사용하여 연결 요청을 확인하는 프로세스입니다. 리스너가 수신한 트래픽은 상황에 따라 라우팅됩니다. 로드 밸런서를 생성한 후 리스너당 여러 규칙과 여러 인증서를 지정할 수 있습니다.

▼ 경청자 HTTP:80 [제거하다]

규약: HTTP 포트: 80

기본 작업 정보
대상 그룹 선택

리스너 추가

그림 15 로드밸런서로 접근하는 80포트의 연결 요청을 넘겨주는 리스너를 등록한다. 그리고 타겟으로 할 대상 그룹을 생성한다.

대상 유형 선택

- ☒ 인스턴스
 - 특정 VPC 내의 인스턴스에 대한 로드 밸런싱을 지원합니다.
 - Amazon EC2 Auto Scaling 사용 촉진 EC2 용량을 관리하고 확장합니다.
- ☐ IP 주소
 - VPC 및 온프레미스 리소스에 대한 로드 밸런싱을 지원합니다.
 - 동일한 인스턴스에서 여러 IP 주소 및 네트워크 인터페이스로의 라우팅을 용이하게 합니다.
 - 마이크로서비스 기반 아키텍처로 유연성을 제공하여 애플리케이션 간 통신을 단순화합니다.
 - IPv6 대상을 지원하여 종단 간 IPv6 통신 및 IPv4-to-IPv6 NAT을 지원합니다.
- ☐ 람다 함수
 - 단일 Lambda 함수로의 라우팅을 용이하게 합니다.
 - Application Load Balancer에만 액세스할 수 있습니다.
- ☐ 애플리케이션 로드 밸런서
 - Network Load Balancer가 특정 VPC 내에서 TCP 요청을 수락하고 라우팅할 수 있는 유연성을 제공합니다.
 - Application Load Balancer와 함께 고정 IP 주소 및 PrivateLink 사용을 촉진합니다.

대상 그룹 이름
edelwiess-group

하이픈을 포함하여 최대 32자의 영숫자 문자가 허용되지만 이름은 하이픈으로 시작하거나 끝날 수 없습니다.

규약: HTTP 포트: 80

VPC
대상 그룹에 포함할 인스턴스가 있는 VPC를 선택합니다.
Edelwiess-vpc
vpc-0148d1bf3ed0bd2ee
IPv4: 10.0.0.0/16

그림 16 특정 VPC 내의 인스턴스에 대한 로드 밸런싱을 위해 유형을 인스턴스로 선택하고 포트 80을 입력해 준다.

Available instances (2)

Filter resources by property or value

Instance ID	Name	State	Security groups	Zone	Subnet ID
i-0d355cac414c6da49		running	edelwiess-SG	ap-northeast-2a	subnet-0725b73467e572f0d
i-005e90fb1aef42e7e		running	edelwiess-SG	ap-northeast-2c	subnet-029b8391469b9add7

0 selected

Ports for the selected instances
Ports for routing traffic to the selected instances.

8080

1-65535 (separate multiple ports with commas)

Include as pending below

2 selections are now pending below. Include more or register targets when ready.

Review targets

Targets (2)

Remove all pending

All Filter resources by property or value

Remove	Health status	Instance ID	Name	Port	State	Security groups	Zone	Subnet ID
X	Pending	i-0d355cac414c6da49		8080	running	edelwiess-SG	ap-northeast-2a	subnet-0725b73467e572f0d
X	Pending	i-005e90fb1aef42e7e		8080	running	edelwiess-SG	ap-northeast-2c	subnet-029b8391469b9add7

그림 17 로드밸런싱으로 구현할 대상인 인스턴스1, 인스턴스2를 로드밸런스 대상 타겟으로 등록해 준다. 사용자가 80번 포트로 접속하면 8080포트 연결해 주기 위해 타겟은 8080포트로 지정한다.

리스너 및 라우팅 정보

리스너는 구성된 프로토콜과 포트를 사용하여 연결 요청을 확인하는 프로세스입니다. 리스너가 수신한 트래픽은 사양에 따라 라우팅됩니다. 로드 밸런서를 생성한 후 리스너당 여러 규칙과 여러 인증서를 지정할 수 있습니다.

▼ 경청자 HTTP:80

제거하다

규약 포트 기본 작업 정보

HTTP 80

1-65535

를 향해서 edelwiess-group

Target type: Instance, IPv4

HTTP

대상 그룹 생성

리스너 추가

등록한 대상 타겟을 리스너 그룹에 포함시킨다.

대상 그룹 (1/1) 정보

대상 그룹 검색 또는 필터링

이름	ARN	포트	규약	대상 유형	로드
edelwiess-group3	arn:aws:elasticloadbalancing...	80	HTTP	사제	예발

행위 ▲ 대상 그룹 생성

Delete

Register targets

Edit health check settings

Edit attributes

Manage tags

그림 19 향후 사용자가 로그인한 세션을 고정시켜주기 위해서 대상 그룹 옵션을 수정해 준다.

Attributes

Restore defaults

Deregistration delay

The time to wait for in-flight requests to complete while deregistering a target. During this time, the state of the target is draining.

300

seconds

0-3600

Slow start duration

During this period, a newly registered target receives an increasing share of requests, until it reaches its fair share.

0

seconds

Requires 30 to 900 seconds to enable, or 0 seconds to disable. This attribute cannot be combined with the **Least outstanding requests** algorithm.

Load balancing algorithm

Determines how the load balancer selects targets from this target group when routing requests.

☒ Round robin
 ☐ Least outstanding requests

Cannot be combined with the **Slow start duration** attribute.

☒ Stickiness

The type of stickiness associated with this target group. If enabled, the load balancer binds a client's session to a specific instance within the target group.

Stickiness type

☒ Load balancer generated cookie
 ☐ Application-based cookie

Stickiness duration

1

days

1 second - 7 days

Cancel

Save changes

그림 20 기간 기반 고정 세션을 이용하여서 특정 기간 동안 이용자의 세션이 한 곳으로 고정 되도록 설정한다.

임의로 등록해 두었던, 인스턴스의 보안그룹을 수정한다.

인바운드 규칙 정보

보안 그룹 규칙 ID	유형 정보	프로토콜 정보	포트 범위 정보	소스 정보	설명 - 선택 사항 정보	
sgr-07aa5f9601e0d9058	사용자 지정 TCP	TCP	8080	사용자 ...	Q	삭제
sgr-014d40869223d4ccc	MySQL/Aurora	TCP	3306	사용자 ...	Q 0.0.0.0/0 X	삭제
sgr-0585dde0c7ee0ff56	SSH	TCP	22	사용자 ...	Q sg-0219d0e10a05f5de6 X	삭제
sgr-06ff2a4cc51f78c36	모든 ICMP - IPv4	ICMP	전체	사용자 ...	Q 0.0.0.0/0 X	삭제
sgr-0f1418d53e399ae0b	NFS	TCP	2049	사용자 ...	Q 0.0.0.0/0 X	삭제

규칙 추가

취소

변경 사항 미리 보기

규칙 저장


그림 21 LB-group으로 만들어두었던 보안그룹을 인스턴스의 보안 그룹으로 포함시켜 수정한다. (ICMP는 인스턴스 간의 통신을 확인을 위함, NFS는 인스턴스간의 마운트를 위함)


- 9 -


[DB 생성하기]


엔진 옵션


엔진 유형 정보


☐ Amazon Aurora


☒ MySQL


☐ MariaDB


☐ PostgreSQL


☐ Oracle


☐ Microsoft SQL Server


에디션

☒ MySQL Community

알려진 문제/제한 사항

알려진 문제/제한 사항 [여기](#)를 검토하여 특정 데이터베이스 버전과 발생할 수 있는 호환성 문제를 확인하세요.

버전

MySQL 8.0.28

템플릿

해당 사용 사례를 충족하는 샘플 템플릿을 선택하세요.

☐ 프로덕션
고가용성 및 빠르고 일관된 성능을 위해 기본값을 사용하세요.

☐ 개발/테스트
이 인스턴스는 프로덕션 환경 외부에서 개발 용도로 마련되었습니다.

☒ 프리 티어
RDS 프리 티어를 사용하여 새로운 애플리케이션을 개발하거나, 기존

그림 22 MySQL 최신 버전인 8.0.28 버전을 사용한다.(프리티어사용)

설정

DB 인스턴스 식별자 정보

DB 인스턴스 이름을 입력하세요. 이름은 현재 AWS 리전에서 AWS 계정이 소유하는 모든 DB 인스턴스에 대해 고유해야 합니다.

edelwiess-db

DB 인스턴스 식별자는 대소문자를 구분하지 않지만 'mydbinstance'와 같이 모두 소문자로 저장됩니다. 제약: 1자~60자의 영숫자 또는 하이픈으로 구성되어야 합니다. 첫 번째 문자는 글자이어야 합니다. 하이픈 2개가 연속될 수 없습니다. 끝에 하이픈이 올 수 없습니다.

자격 증명 설정

마스터 사용자 이름 정보

DB 인스턴스의 마스터 사용자에 로그인 ID를 입력하세요.

root

1~16자의 영숫자. 첫 번째 문자는 글자여야 합니다.

☐ 암호 자동 생성
Amazon RDS에서 사용자를 대신하여 암호를 생성하거나 사용자가 직접 암호를 지정할 수 있습니다.

마스터 암호 정보

제약 조건: 8자 이상의 인쇄 가능한 ASCII 문자. 다음은 포함할 수 없습니다. /(슬래시), '(작은따옴표), "(큰따옴표) 및 @(앳 기호).

암호 확인 정보

DB 인스턴스 클래스

DB 인스턴스 클래스 정보

☐ 스탠다드 클래스(m 클래스 포함)

☐ 메모리 최적화 클래스(r 및 x 클래스 포함)

☒ 버스터블 클래스(t 클래스 포함)

db.t3.micro

2 vCPUs 1 GiB RAM 네트워크: 2,085Mbps

그림 23 설정장에서 사용할 DB의 이름과 계정, 비밀번호를 등록한다
(스토리지 자동조정 활성화 체크해제)

- 10 -

연결

Virtual Private Cloud(VPC) 정보
이 DB 인스턴스의 가상 네트워킹 환경을 정의하는 VPC.
Edelwiess-vpc (vpc-0148d1bf3ed0bd2ee) ▼
해당 DB 서브넷 그룹이 있는 VPC만 나열됩니다.

❗ 데이터베이스를 생성한 후에는 VPC를 변경할 수 없습니다.

서브넷 그룹 정보
선택한 VPC에서 DB 인스턴스가 어떤 서브넷과 IP 범위를 사용할 수 있는지를 정의하는 DB 서브넷 그룹.
새 DB 서브넷 그룹 생성 ▼

퍼블릭 액세스 정보
☐ 예
VPC 외부의 Amazon EC2 인스턴스 및 디바이스는 데이터베이스에 연결할 수 있습니다. 데이터베이스에 연결할 수 있는 VPC 내부의 EC2 인스턴스 및 디바이스를 지정하는 하나 이상의 VPC 보안 그룹을 선택하세요.
☒ 아니요
RDS는 데이터베이스에 퍼블릭 IP 주소를 할당하지 않습니다. VPC 내부의 Amazon EC2 인스턴스 및 디바이스만 데이터베이스에 연결할 수 있습니다.

VPC 보안 그룹
데이터베이스에 대한 액세스를 허용할 VPC 보안 그룹을 선택합니다. 보안 그룹 규칙이 적절한 수신 트래픽을 허용하는지 확인합니다.
☒ 기존 항목 선택
기존 VPC 보안 그룹 선택
☐ 새로 생성
새 VPC 보안 그룹 생성

기존 VPC 보안 그룹
VPC 보안 그룹 선택 ▼
db-group X

가용 영역 정보
ap-northeast-2a ▼

그림 24 동일 VPC 내에서 DB를 작동시키기 위해서 Edelwiess-vpc를 선택, VPC 내부의 인스턴스만 접근이 가능하도록 퍼블릭 액세스는 허용하지 않는다. 위에서 만든 DB용 보안 그룹을 포함 시키고 생성한다.

RDS > 데이터베이스 > edelwiess-db

edelwiess-db 수정 작업 ▼

요약

DB 식별자 edelwiess-db	CPU 2.82%	상태 사용 가능	클래스 db.t3.micro
역할 인스턴스	현재 활동 0 연결	엔진 MySQL Community	리전 및 AZ ap-northeast-2a

연결 & 보안 | 모니터링 | 로그 및 이벤트 | 구성 | 유지 관리 및 백업 | 태그

연결 & 보안

엔드포인트 및 포트 엔드포인트 edelwiess-db.cfbrn7vatvmj.ap-northeast-2.rds.amazonaws.com 포트 3306	네트워킹 가용 영역 ap-northeast-2a VPC Edelwiess-vpc (vpc-0148d1bf3ed0bd2ee) 서브넷 그룹 default-vpc-0148d1bf3ed0bd2ee 서브넷 subnet-0472a5a1a4d823546 subnet-0725b73467e572f0d subnet-029b8391469b9add7 subnet-093791125898d9411	보안 VPC 보안 그룹 db-group (sg-03fcd7e281a7682d) 활성 퍼블릭 액세스 가능 아니요 인증 기관 rds-ca-2019 인증 기관 날짜 August 23, 2024, 02:08 (UTC+2:08)
--	---	--

그림 25 생성된 DB의 앤드포인트를 통해서 인스턴스에서 DB로 접근을 해본다.

```
root@instance1:~# mysql -u root -ptest1234 -h edelwiess-db.cfbrn7vatvmj.ap-northeast-2.rds.amazonaws.com
mysql: [Warning] Using a password on the command line interface can be insecure.
Welcome to the MySQL monitor.  Commands end with ; or \g.
Your MySQL connection id is 17
Server version: 8.0.28 Source distribution

Copyright (c) 2000, 2022, Oracle and/or its affiliates.

Oracle is a registered trademark of Oracle Corporation and/or its
affiliates. Other names may be trademarks of their respective
owners.

Type 'help;' or '\h' for help. Type '\c' to clear the current input statement.
mysql>
```

인스턴스에서 생성한 DB로의 접근을 확인한다.

[war파일로 웹 서비스 배포하기]

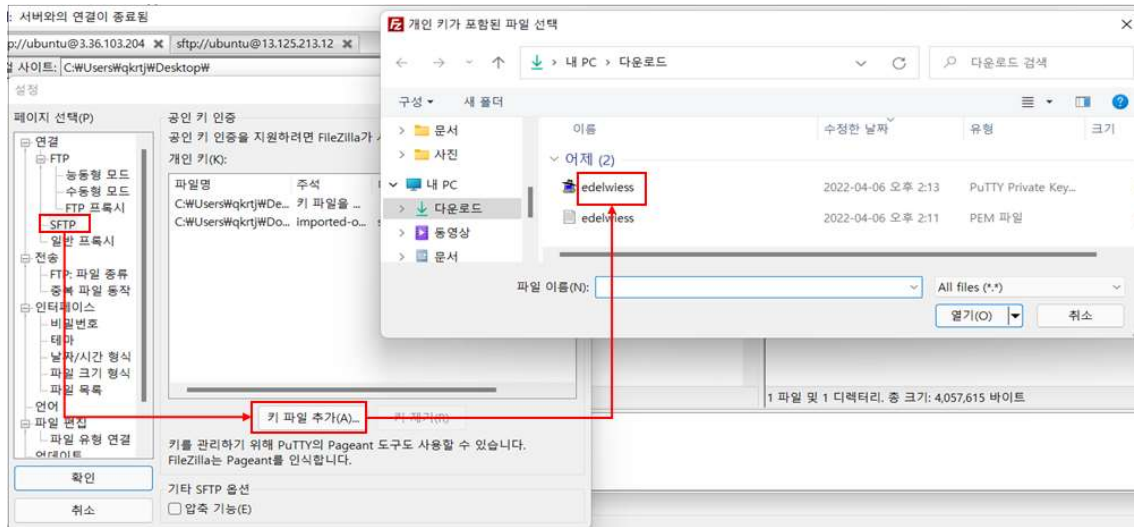


그림 27 FileZilla로 war 파일을 업로드하기 위해 ssh접속을 위한 키를 추가해 준다.

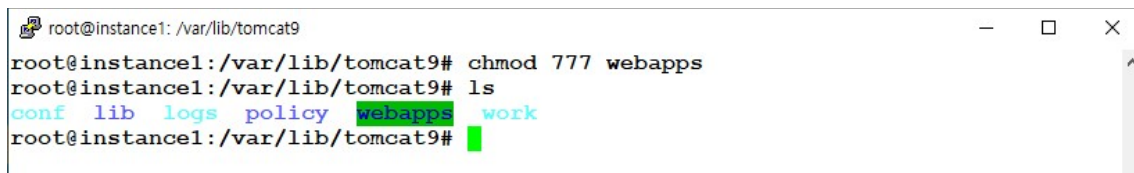


그림 28 파일을 업로드 할 위치의 폴더인 webapps의 퍼미션을 조정해준다

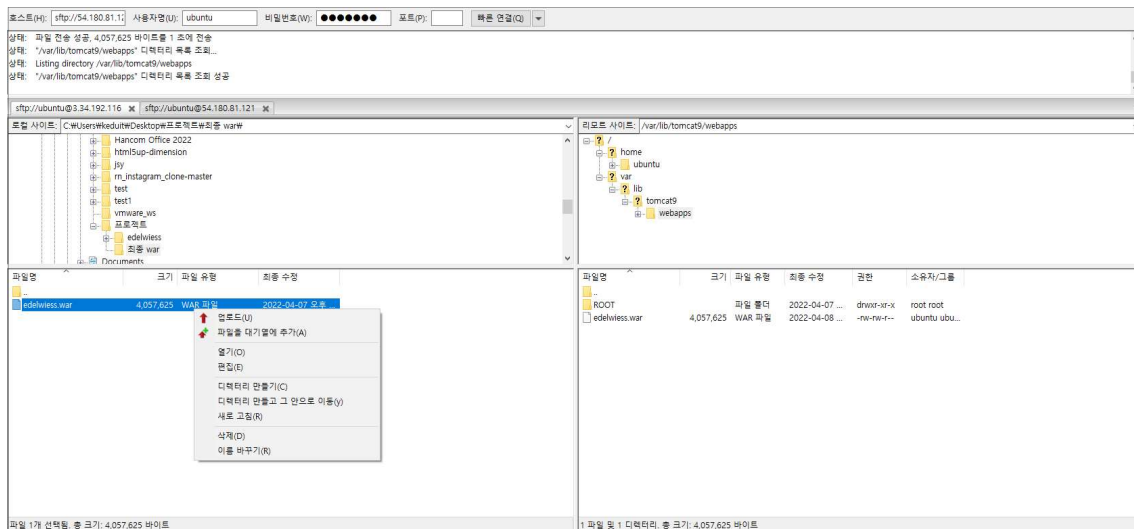


그림 29 퍼미션을 조정해준 폴더에 war파일을 업로드 한다.

(배포를 수행할 경로 /var/lib/tomcat9/webapps)

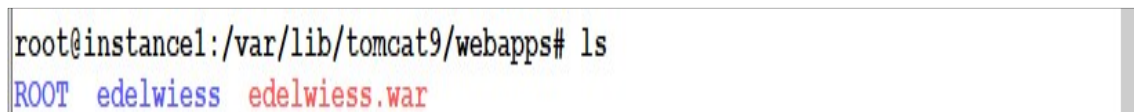


그림 30 업로드한 위치를 확인해보면 같은 이름의 폴더가 생성 된 것을 볼 수 있다.

[DB 생성 & table 생성]

```
mysql> use edelwiess;
Database changed
mysql> create table member (
  -> name varchar(10),
  -> email varchar(50) primary key,
  -> pwd varchar(12),
  -> phone varchar(13),
  -> message varchar(100));
Query OK, 0 rows affected (0.04 sec)

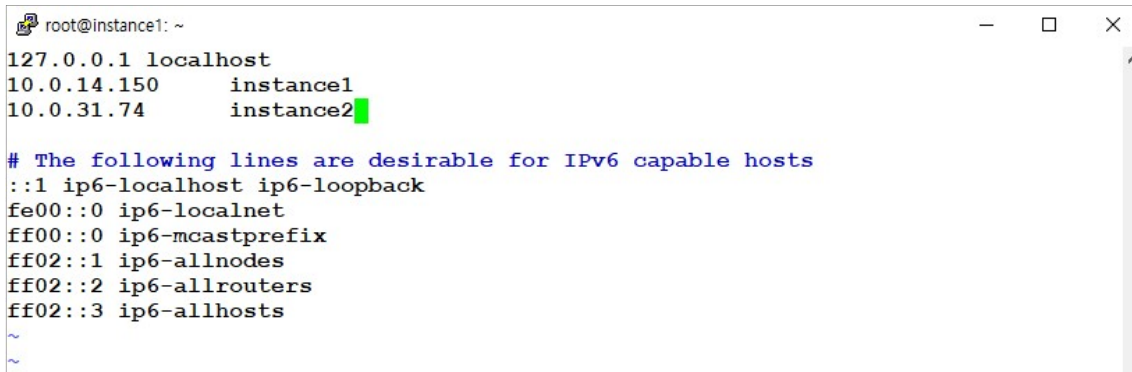
mysql> desc member;
+-----+-----+-----+-----+-----+-----+
| Field | Type          | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| name  | varchar(10)   | YES  |     | NULL    |       |
| email | varchar(50)   | NO   | PRI | NULL    |       |
| pwd   | varchar(12)   | YES  |     | NULL    |       |
| phone | varchar(13)   | YES  |     | NULL    |       |
| message | varchar(100) | YES  |     | NULL    |       |
+-----+-----+-----+-----+-----+-----+
5 rows in set (0.01 sec)
```

```
mysql> create table file (
  -> fileName varchar(200),
  -> fileRealName varchar(200));
Query OK, 0 rows affected (0.05 sec)

mysql> desc file
  -> ;
+-----+-----+-----+-----+-----+-----+
| Field          | Type          | Null | Key | Default | Extra |
+-----+-----+-----+-----+-----+-----+
| fileName       | varchar(200)  | YES  |     | NULL    |       |
| fileRealName   | varchar(200)  | YES  |     | NULL    |       |
+-----+-----+-----+-----+-----+-----+
2 rows in set (0.00 sec)
```

[세션이 달라도 공유된 파일이 보이도록 인스턴스들 간의 마운트적용]

그림 33 vi /etc/hosts



```
root@instance1: ~
127.0.0.1 localhost
10.0.14.150 instance1
10.0.31.74 instance2

# The following lines are desirable for IPv6 capable hosts
::1 ip6-localhost ip6-loopback
fe00::0 ip6-localnet
ff00::0 ip6-mcastprefix
ff02::1 ip6-allnodes
ff02::2 ip6-allrouters
ff02::3 ip6-allhosts
~
```

#instance1

그림 34 ip를 사용할 때 쉽게 사용하기 위해서 vi /etc/hosts 파일에 ip와 hostname을 맵핑

```
root@instance1:~# apt install nfs-kernel-server
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  keyutils libnfsidmap2 libtirpc1 nfs-common rpcbind
Suggested packages:
  watchdog
```


그림 35 NFS를 사용하기 위한 서버 설치

```
root@instance1:~# apt -y install nfs-common
Reading package lists... Done
Building dependency tree
Reading state information... Done
nfs-common is already the newest version (1:1.3.4-2.1ubuntu5.5).
nfs-common set to manually installed.
0 upgraded, 0 newly installed, 0 to remove and 4 not upgraded.
```

NFS 설치

```
root@instance1:/var/lib/tomcat9/webapps/edelwiess/images# chmod 777 upload
root@instance1:/var/lib/tomcat9/webapps/edelwiess/images# ls
bg.jpg cloud.jpg overlay.png pic01.jpg pic02.jpg pic03.jpg upload
```

그림 36 공유할 폴더의 권한을 모두 허용으로 수정

```
/var/lib/tomcat9/webapps/edelwiess/images/upload 10.0.31.74(rw,sync,no_ro
ot_squash)
~
```

vi /etc/exports에서 공유폴더 upload를 사용할 ip와 권한을 지정해 준다

#instance2

instance1과 마찬가지로 NFS를 사용하기 위해 설치를 먼저 진행한다.

apt install nfs-kernel-server

apt install nfs-common

```
root@instance2:~# mount -t nfs instance1:/var/lib/tomcat9/webapps/edelwiess/images/
upload /var/lib/tomcat9/webapps/edelwiess/images/upload
```

ip주소(맵핑한 hostname)와 공유할 디렉토리를 지정하여 마운트한다.

```
10.0.14.150:/var/lib/tomcat9/webapps/edelwiess/images/upload /var/lib/tomcat9/we
bapps/edelwiess/images/upload
nfs defaults
0 0
~
~
```

vi /etc/fstab에서 재부팅시 자동으로 마운트가 될 수 있도록 마운트 정보를 입력하여 준다.

```
root@instance2:/var/lib/tomcat9/webapps/edelwiess/images/upload# mount | grep /uplo
ad
instance1:/var/lib/tomcat9/webapps/edelwiess/images/upload on /var/lib/tomcat9/weba
pps/edelwiess/images/upload type nfs4 (rw,relatime,vers=4.2,rsize=131072,wsiz=1310
72,namlen=255,hard,proto=tcp,timeo=600,retrans=2,sec=sys,clientaddr=10.0.31.74,loca
l lock=none,addr=10.0.14.150)
```

마운트가 잘 되었는지 확인

[배포 확인]

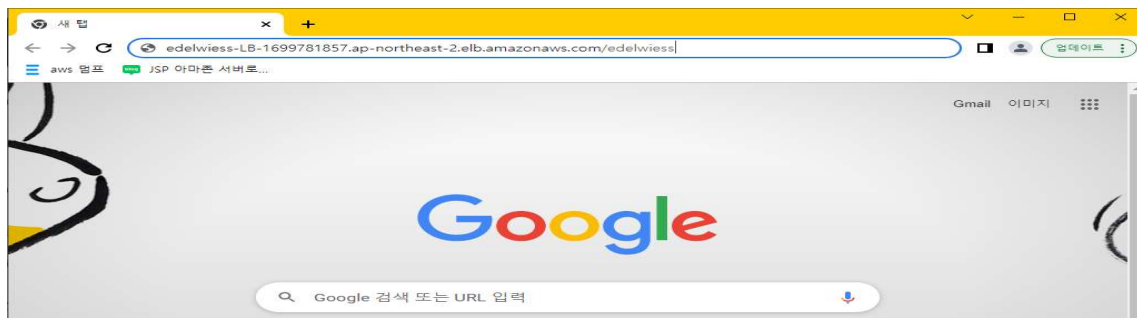


그림 41 로드밸런서의[DNS name]/[배포한 폴더명]을 이용하여 배포한 웹 사이트에 접속해 본다.

그림 42 접속한 메인화면

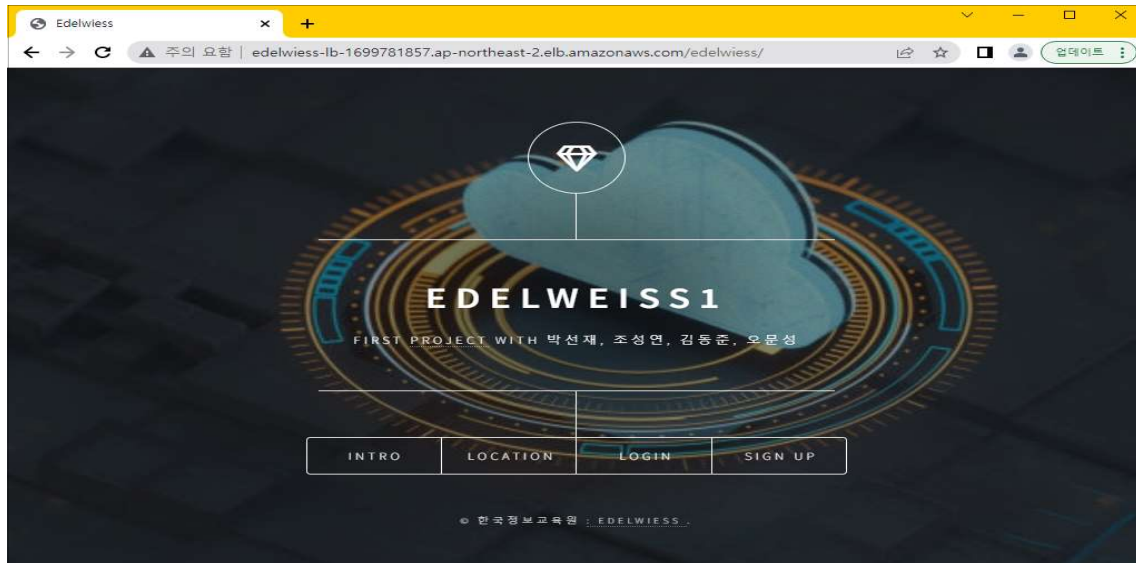


그림 43 회원가입 페이지

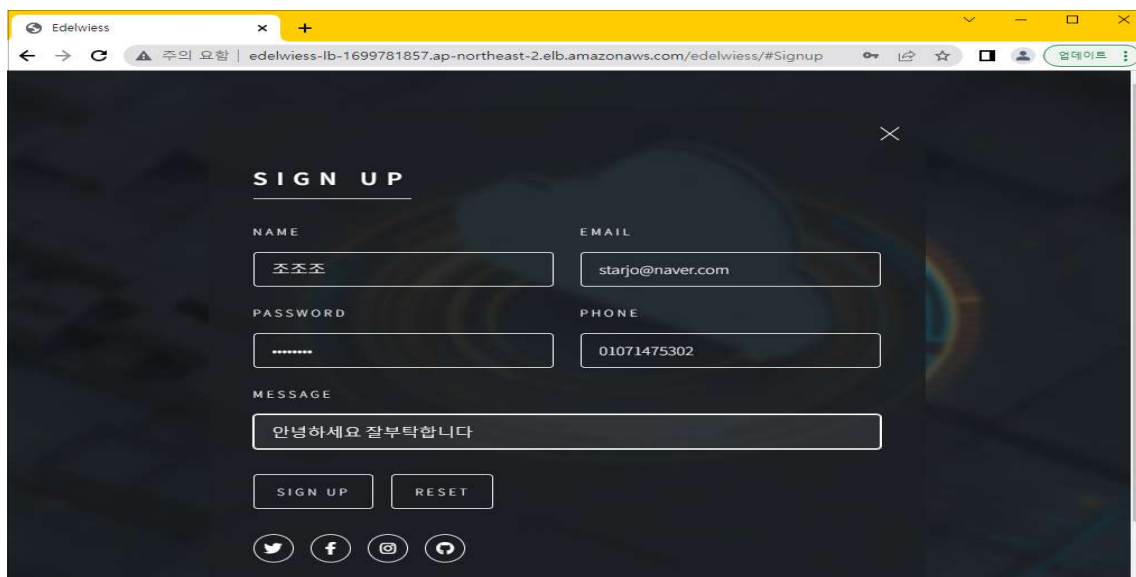


그림 44 회원가입 완료 메시지 창

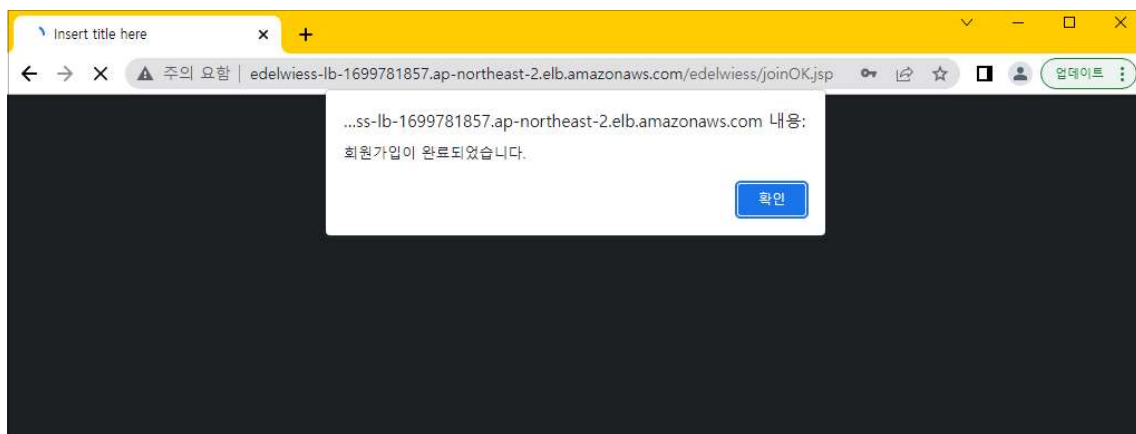


그림 45 DB에 접속하여 회원가입한 회원의 정보 조회

```
mysql> mysql> select * from member;
```

name	email	pwd	phone	message
조조조	starjo@naver.com	test1234	01071475302	안녕하세요 잘 부탁드립니다
조성연	starjo5@naver.com	test1234	01071475302	안녕하세요

2 rows in set (0.00 sec)

그림 46 회원가입한 이메일과 패스워드로 로그인 하기

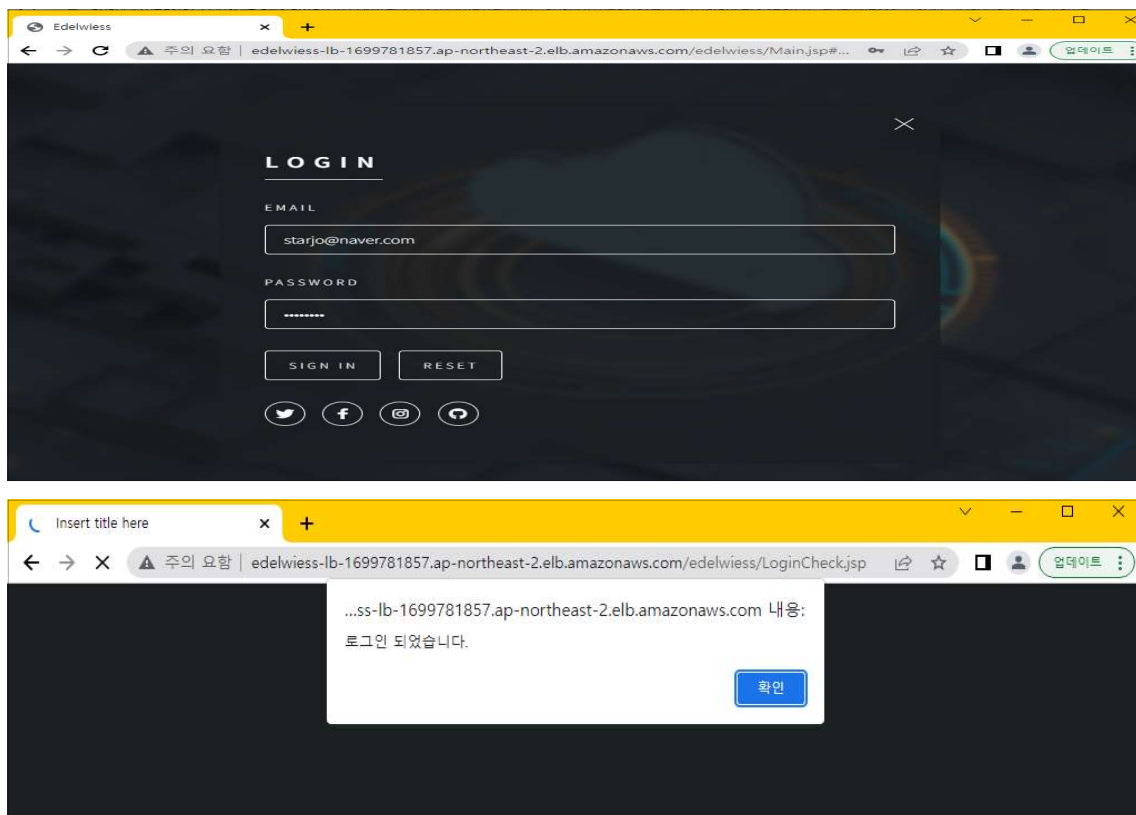
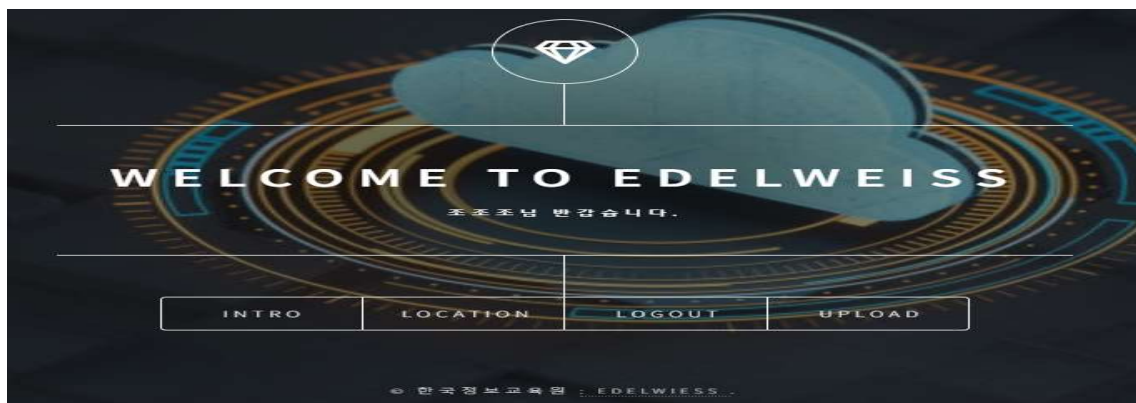


그림 48 로그인 완료 화면



[파일 업로드 & 스토리지 동작 확인]

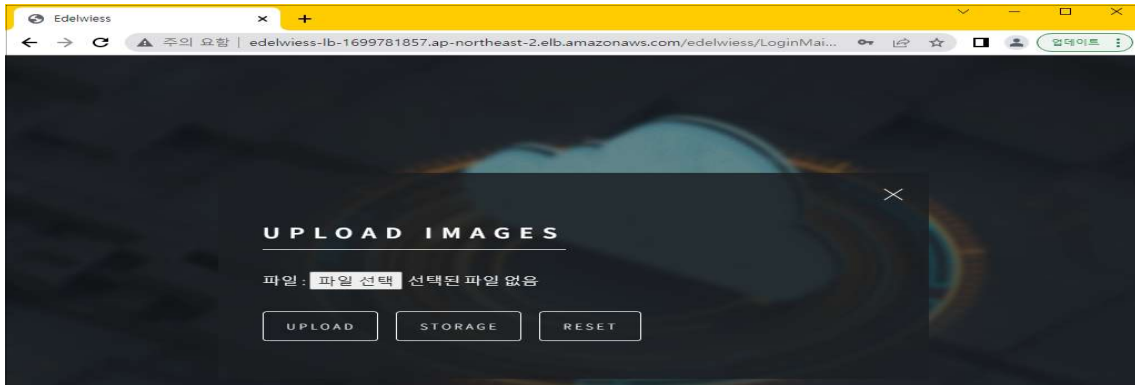


그림 50 파일을 선택하고 업로드를 진행

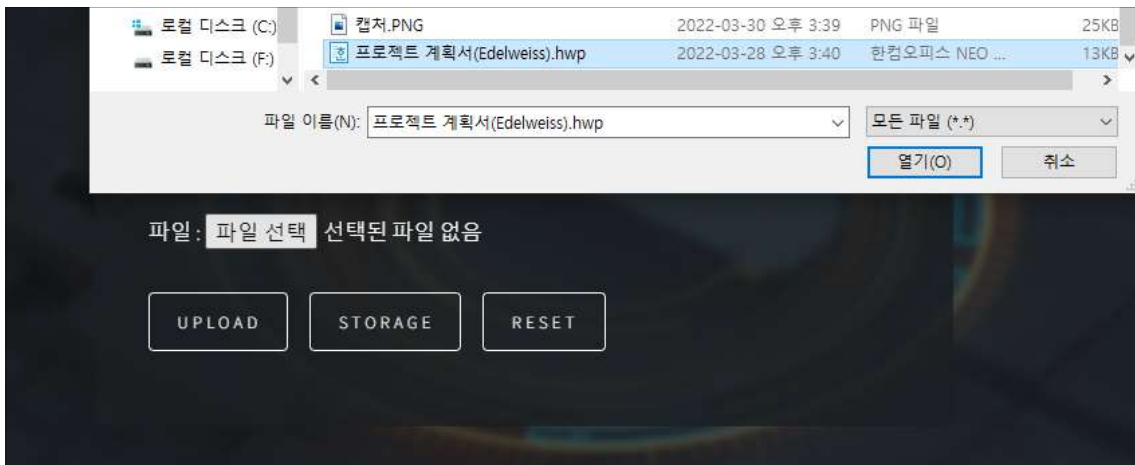


그림 51 업로드한 파일의 실체가 담기는 폴더 확인

```
root@instancel:/var/lib/tomcat9/webapps/edelwiess/images/upload# ls
a.txt  b.txt  캡처.PNG  '프로젝트 계획서 (Edelweiss).hwp'
```

그림 52 STORAGE를 통해 업로드한 파일 확인

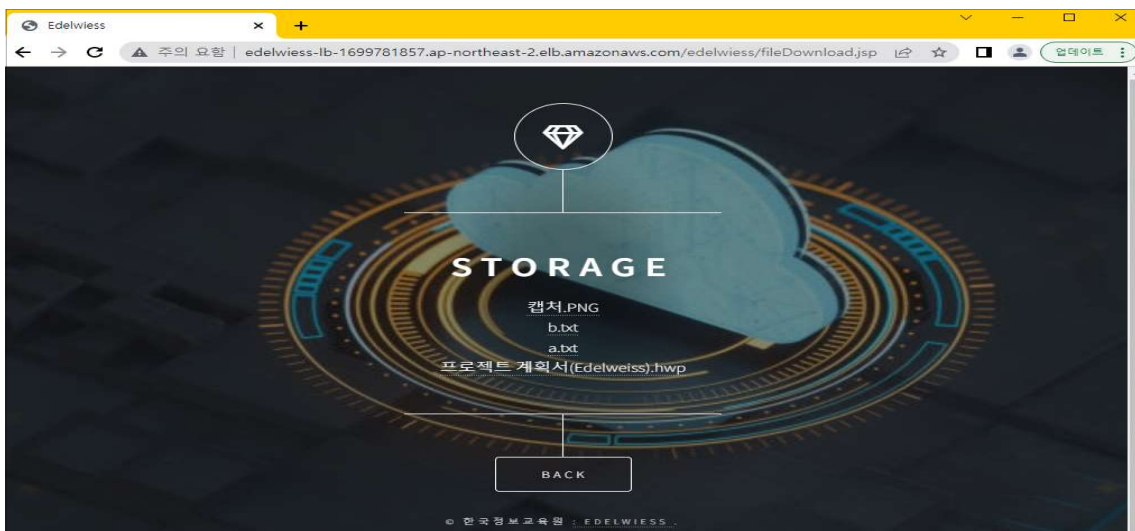


그림 53 클릭으로 다운로드 가능 확인



[AWS - (S3) - Bucket]

그림 54 버킷 만들기 - 이름 설정 - 리전(서울)



그림 55 모든 퍼블릭 액세스 차단 해제



그림 56 버킷 만들기 클릭

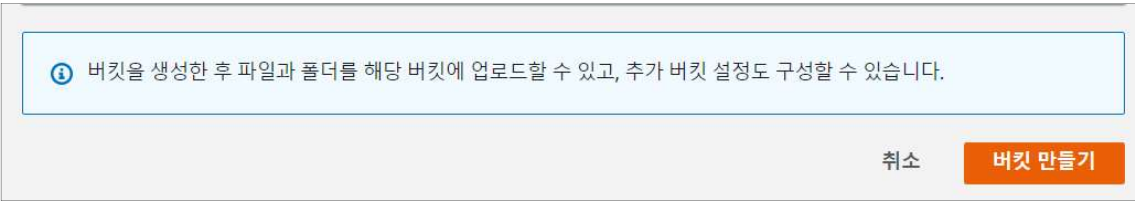


그림 57 이름 클릭



그림 58 권한 클릭 -> 버킷 정책 편집

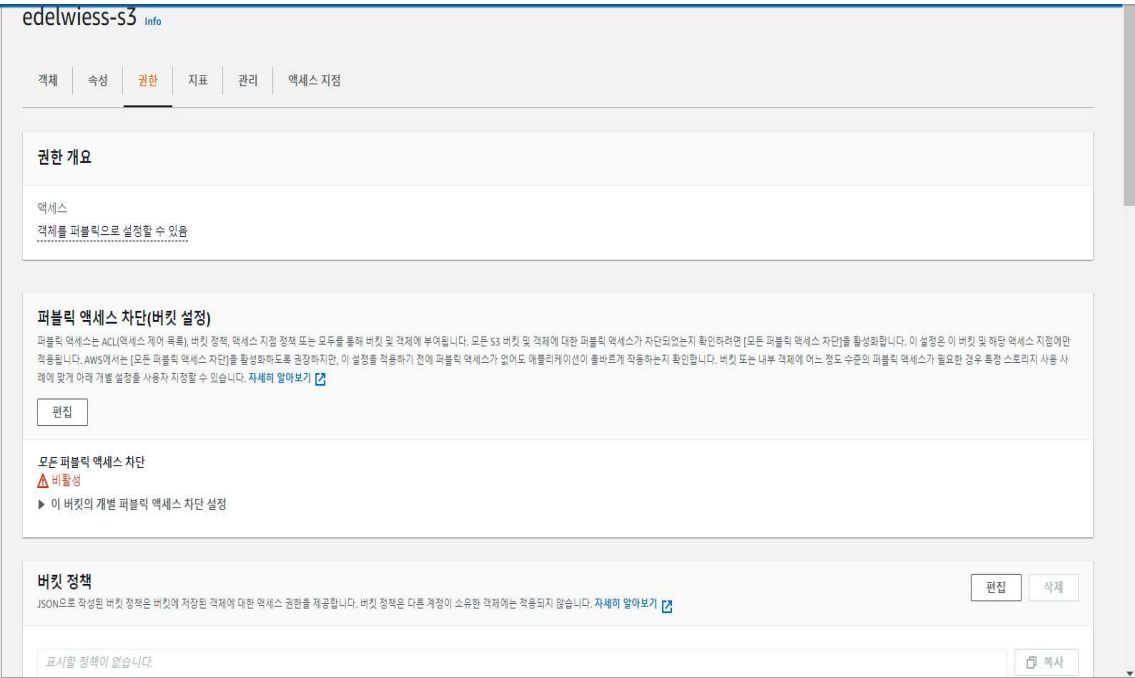


그림 59 정책 생성기 클릭. (새 페이지 창이 뜬다.)

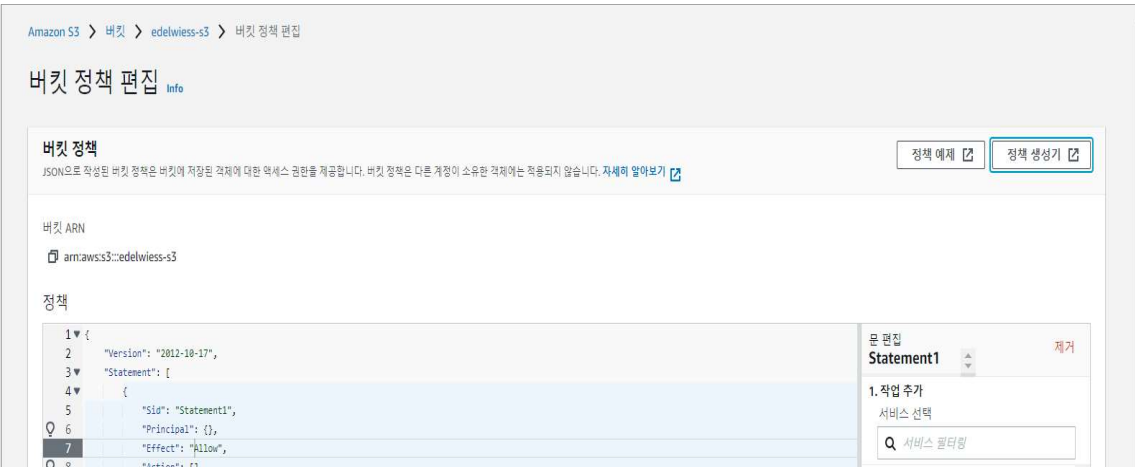


그림 60 S3 Bucket Policy 체크 -> * 입력 -> GetObject 클릭

amazon web services

AWS Policy Generator

The AWS Policy Generator is a tool that enables you to create policies that control access to Amazon Web Services (AWS) products and resources. For more information about creating policies, see [key concepts in Using AWS Identity and Access Management](#). Here are [sample policies](#).

Step 1: Select Policy Type

A Policy is a container for permissions. The different types of policies you can create are an [IAM Policy](#), an [S3 Bucket Policy](#), an [SNS Topic Policy](#), a [VPC Endpoint Policy](#), and an [SQS Queue Policy](#).

Select Type of Policy S3 Bucket Policy

Step 2: Add Statement(s)

A statement is the formal description of a single permission. See [a description of elements](#) that you can use in statements.

Effect ☒ Allow ☐ Deny

Principal

Use a comma to separate multiple values.

AWS Service Amazon S3 ☐ All Services ('*')

Use multiple statements to add permissions for more than one service.

Actions 1 Action(s) Selected ☐ All Actions ('*')

☐ GetMultiRegionAccessPointPolicyStatus
☒ GetObject
☐ GetObjectAcl
☐ GetObjectAttributes
☐ GetObjectLegalHold

Amazon Resource Name (ARN)

그림 61 ARN을 복사 후 붙여넣기



그림 62 앞 페이지에서 ARN 주소를 복사해서 ARN 에 복사 붙여넣기

Step 2: Add Statement(s)

A statement is the formal description of a single permission. See [a description of elements](#) that you can use in statements.

Effect ☒ Allow ☐ Deny

Principal

Use a comma to separate multiple values.

AWS Service Amazon S3 ☐ All Services ('*')

Use multiple statements to add permissions for more than one service.

Actions 1 Action(s) Selected ☐ All Actions ('*')

Amazon Resource Name (ARN) arn:aws:s3:::edelwiess-s3

ARN should follow the following format: `arn:aws:s3:::{BucketName}/{KeyName}`.
Use a comma to separate multiple values.

[Add Conditions \(Optional\)](#)

Add Statement

그림 63 Add Statement 클릭 하면 아래와 같이 뜬다.

Add Statement

You added the following statements. Click the button below to Generate a policy.

Principal(s)	Effect	Action	Resource	Conditions
• *	Allow	• s3:GetObject	arn:aws:s3:::edelwiess-s3	None

Step 3: Generate Policy

A policy is a document (written in the [Access Policy Language](#)) that acts as a container for one or more statements.

Generate Policy **Start Over**

그림 64 Generate Policy 클릭 하면 아래와 같은 화면 뜬다. 이후 JSON 파일 복사해 놓는다.

Policy JSON Document

Click below to edit. To save the policy, copy the text below to a text editor. Changes made below will **not be reflected** in the policy generator tool.

```
{
  "Id": "Policy1649663417182",
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "Stmnt1649663407014",
      "Action": [
        "s3:GetObject"
      ],
      "Effect": "Allow",
      "Resource": "arn:aws:s3:::edelwiess-s3",
      "Principal": "*"
    }
  ]
}
```

This AWS Policy Generator is provided for informational purposes only, you are still responsible for your use of Amazon Web Services technologies and ensuring that you comply with applicable laws, rules, and regulations. For more information, see the [AWS IAM Policy Generator](#) documentation.

Close

그림 65 복사한 JSON 텍스트를 기존 정책에 붙여 넣고 /*을 추가한다.

```
1 {
2   "Id": "Policy1649663417182",
3   "Version": "2012-10-17",
4   "Statement": [
5     {
6       "Sid": "Stmnt1649663407014",
7       "Action": [
8         "s3:GetObject"
9       ],
10      "Effect": "Allow",
11      "Resource": "arn:aws:s3:::edelwiess-s3/*",
12      "Principal": "*"
13    }
14  ]
15 }
```

그림 66 그 후 변경 사항 저장한다.

외부 액세스 미리 보기

취소 **변경 사항 저장**

그림 67 이후 속성으로 가서 제일 밑에 정적 웹 사이트 호스팅 편집으로 간다.

객체 **속성** 권한 지표 관리 액세스 지점

그림 68 편집 클릭



그림 69 활성화 체크 -> index.html, error.html 입력 후 저장.

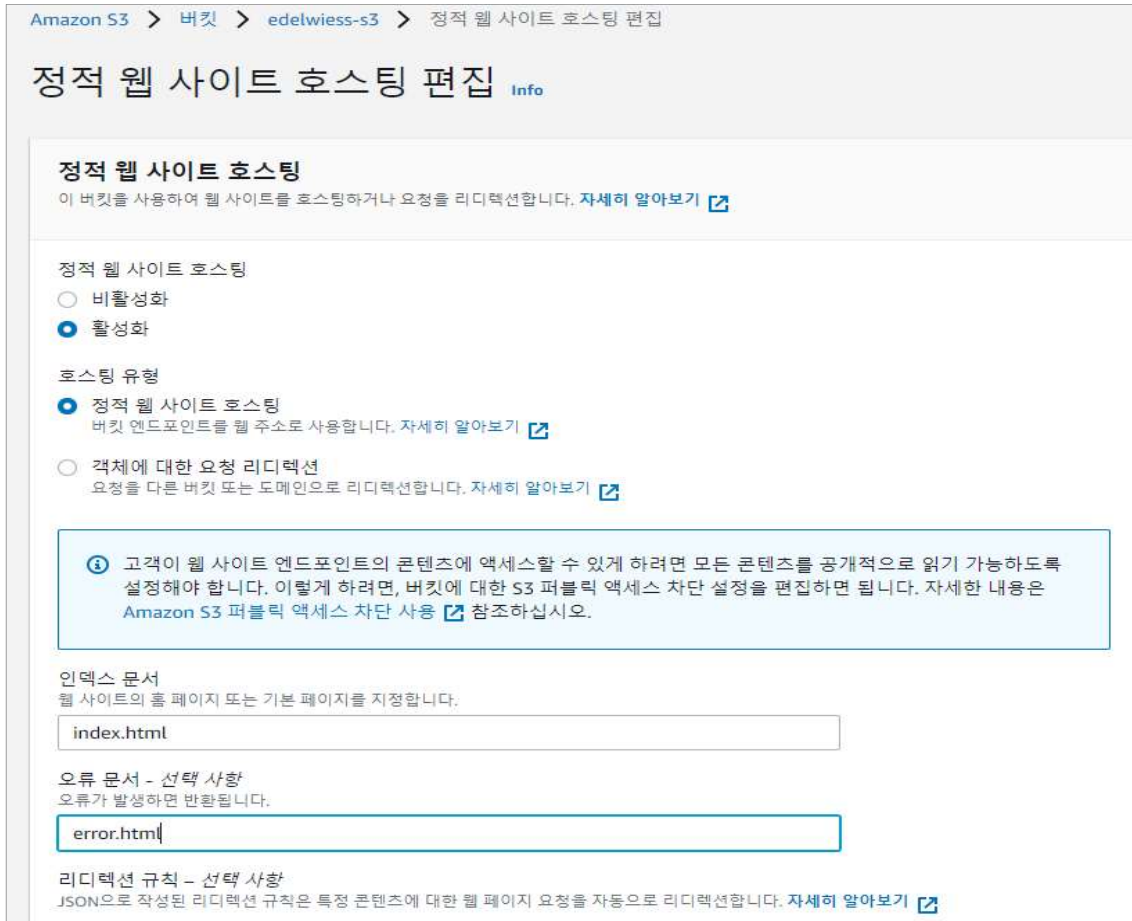


그림 70 저장 클릭.

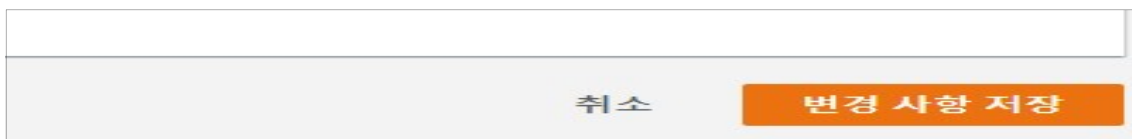


그림 71 IAM 검색 후 이동.

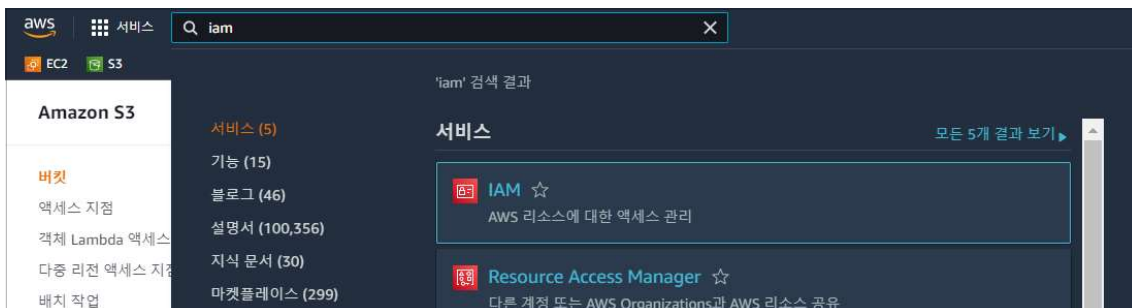


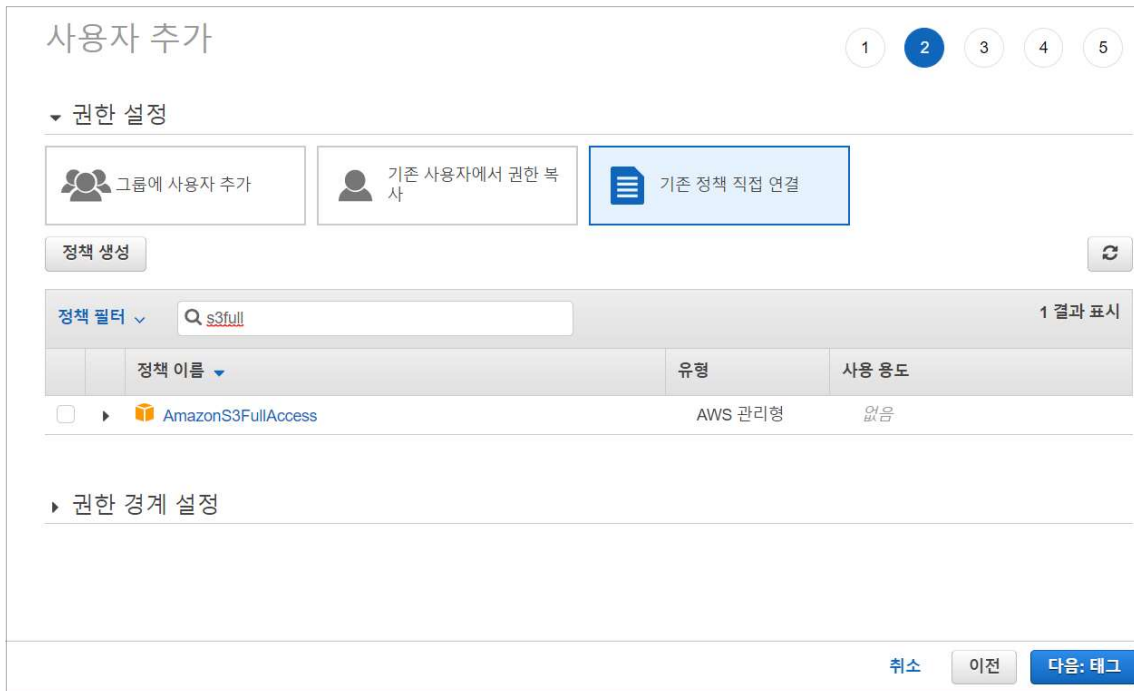
그림 72 사용자 클릭 -> 사용자 추가 클릭 .



그림 73 사용자 이름 설정 후 -> 액세스 키 -프로그래밍 방식 액세스 ->다음:권한 클릭



그림 74 기존 정책 직접 연결 클릭 -> 정책 필터에 S3full 검색 후 AmazonS3Full Access 체크



태그는 넘어간다.

권한 요약

다음 정책이 위에 표시된 사용자에게 연결됩니다.

유형	이름
관리형 정책	AmazonS3FullAccess

태그

태그가 추가되지 않았습니다.

취소

이전

사용자 만들기

✔ 성공

아래에 표시된 사용자를 생성했습니다. 사용자 보안 자격 증명을 보고 다운로드할 수 있습니다. AWS Management Console 로그인을 위한 사용자 지침을 이메일로 보낼 수도 있습니다. 지금이 이 자격 증명을 다운로드할 수 있는 마지막 기회입니다. 하지만 언제든지 새 자격 증명을 생성할 수 있습니다.

AWS Management Console 액세스 권한이 있는 사용자가 <https://135973764126.signin.aws.amazon.com/console>에 로그인할 수 있습니다.

↓
.csv 다운로드

	사용자	액세스 키 ID	비밀 액세스 키
▶	✔ edelwiess-user	AKIAR7KFK7QPCH2EWIHG	***** 표시

Access key ID	Secret access key
AKIAR7KFK7QPFSXW5UPM	qCgrA1dWPHArMdhjWyCzzAz32ANtNnd44x1F7iNC

```
root@server1:~# apt -y install awscli
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
```

```
root@server1:~# aws configure
AWS Access Key ID [None]: AKIAR7KFK7QPFSXW5UPM
AWS Secret Access Key [None]: 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
Default region name [None]: ap-northeast-2
Default output format [None]:
```

- 24 -

그림 81 aws s3 ls

```
root@server1:~# aws s3 ls
2022-04-11 07:53:32 edelwiess-s3
```

그림 82 폴더만들기 로 폴더를 하나 생성한다.



그림 83 폴더 이름 입력 후 폴더만들기 클릭.

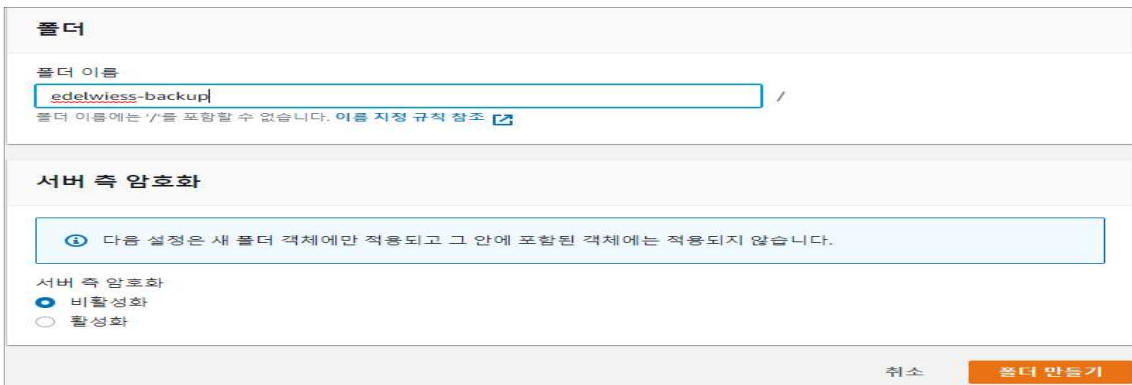


그림 84 폴더가 생성된 것을 확인 할 수 있다.



그림 85 aws s3 ls 버킷이름 입력 해보면 생성된 파일 목록이 나온다.

```
root@server1:~# aws s3 ls edelwiess-s3
PRE edelwiess-backup/
root@server1:~#
```

그림 86 crontab -e -> 1번 엔터

```
root@server1:~# crontab -e
no crontab for root - using an empty one

Select an editor. To change later, run 'select-editor'.
 1. /bin/nano        <---- easiest
 2. /usr/bin/vim.basic
 3. /usr/bin/vim.tiny
 4. /bin/ed

Choose 1-4 [1]: 1
crontab: installing new crontab
root@server1:~# crontab -l
* * * * * /root/backup.sh
```

그림 87 * * * * * /root/backup.sh 입력 후 Ctrl + x 저장 후 Y 입력 후 엔터.

```

root@server1: ~
GNU nano 2.9.3 /tmp/crontab.gNIP15/crontab Modified
* * * * * /root/backup.sh
# Edit this file to introduce tasks to be run by cron.
#
# Each task to run has to be defined through a single line
# indicating with different fields when the task will be run
# and what command to run for the task
#
# To define the time you can provide concrete values for
# minute (m), hour (h), day of month (dom), month (mon),
# and day of week (dow) or use '*' in these fields (for 'any').#
# Notice that tasks will be started based on the cron's system
# daemon's notion of time and timezones.
#
# Output of the crontab jobs (including errors) is sent through
# email to the user the crontab file belongs to (unless redirected).
#
# For example, you can run a backup of all your user accounts
# at 5 a.m every week with:
# Save modified buffer? (Answering "No" will DISCARD changes.)
Y Yes
N No
AC Cancel

```

그림 88 이후 crontab -l 해보면 크론탭이 예약 된게 보인다.

```

root@server1:~# crontab -l
* * * * * /root/backup.sh

```

그림 89 vi 편집기로 sh 파일 생성.

```

root@server1:~# vi backup.sh

#!/bin/bash
frame=$(date '+%y%m%d%H%M')
tar -zcvf /var/lib/tomcat9/webapps/edelwiess/images/upload/${frame}.tar.gz
/var/lib/tomcat9/webapps/edelwiess/images/upload
aws s3 cp /var/lib/tomcat9/webapps/edelwiess/images/upload/*.tar.gz
s3://edelwiess-s3/edelwiess-backup/
rm -rf /var/lib/tomcat9/webapps/edelwiess/images/upload/*.tar.gz

#!/bin/bash
frame=$(date '+%y%m%d%H%M')
tar -zcvf /var/lib/tomcat9/webapps/edelwiess/images/upload/${frame}.tar.gz /var/lib/tomcat9/webapps/edelwiess/images/upload
aws s3 cp /var/lib/tomcat9/webapps/edelwiess/images/upload/*.tar.gz s3://edelwiess-s3/edelwiess-backup/
rm -rf /var/lib/tomcat9/webapps/edelwiess/images/upload/*.tar.gz

```

입력 후 저장.

그림 91 백업 파일이 생성되지 않는다면 권한 설정을 해주지 않은 것이다. 실행 권한을 주자.

```

root@server1:~# chmod +x backup.sh
root@server1:~# ls
apache-tomcat-9.0.62  apache-tomcat-9.0.62.tar.gz  backup.sh  snap
root@server1:~# ls /var/lib/tomcat9/webapps/edelwiess/images/upload
2204110859.tar.gz  AWS-Edelwiess-최종결과보고서.hwp

```

그림 92 압축 파일이 1분마다 생성되는 것을 확인 할 수 있다.

```

2204110859.tar.gz  2204110900.tar.gz  2204110901.tar.gz  2204110902.tar.gz  AWS-Edelwiess-최종결과보고서.hwp
root@server1:~# ls /var/lib/tomcat9/webapps/edelwiess/images/upload
2204110859.tar.gz  2204110900.tar.gz  2204110901.tar.gz  2204110902.tar.gz  AWS-Edelwiess-최종결과보고서.hwp
root@server1:~#

```

그림 93 이후 aws 버킷 폴더에서 확인해보면 자동 생성된 백업파일이 업로드 됨.

객체 (4)

객체는 Amazon S3에 저장되어 있는 기본 엔티티입니다. Amazon S3 인벤토리 [링크](#)를 사용하여 버킷에 있는 모든 객체의 목록을 얻을 수 있습니다. 다른 사용자가 객체에 액세스할 수 있게 하려면 명시적으로 권한을 부여해야 합니다. 자세히 알아보기 [링크](#)

<input type="checkbox"/>	이름	유형	마지막 수정	크기	스토리지 클래스
<input type="checkbox"/>	2204110859.tar.gz	gz	2022. 4. 11. pm 5:59:03 PM KST	5.5MB	Standard
<input type="checkbox"/>	2204110904.tar.gz	gz	2022. 4. 11. pm 6:04:03 PM KST	5.5MB	Standard
<input type="checkbox"/>	2204110905.tar.gz	gz	2022. 4. 11. pm 6:05:03 PM KST	5.5MB	Standard
<input type="checkbox"/>	2204110906.tar.gz	gz	2022. 4. 11. pm 6:06:03 PM KST	5.5MB	Standard

그림 94 crontab -r을 하면 중지 된다.

```

root@server1:~# crontab -r
root@server1:~# crontab -l
no crontab for root

```