Lab7. Create Private Subnet, NAT Gateway and Network Connection Test

목적

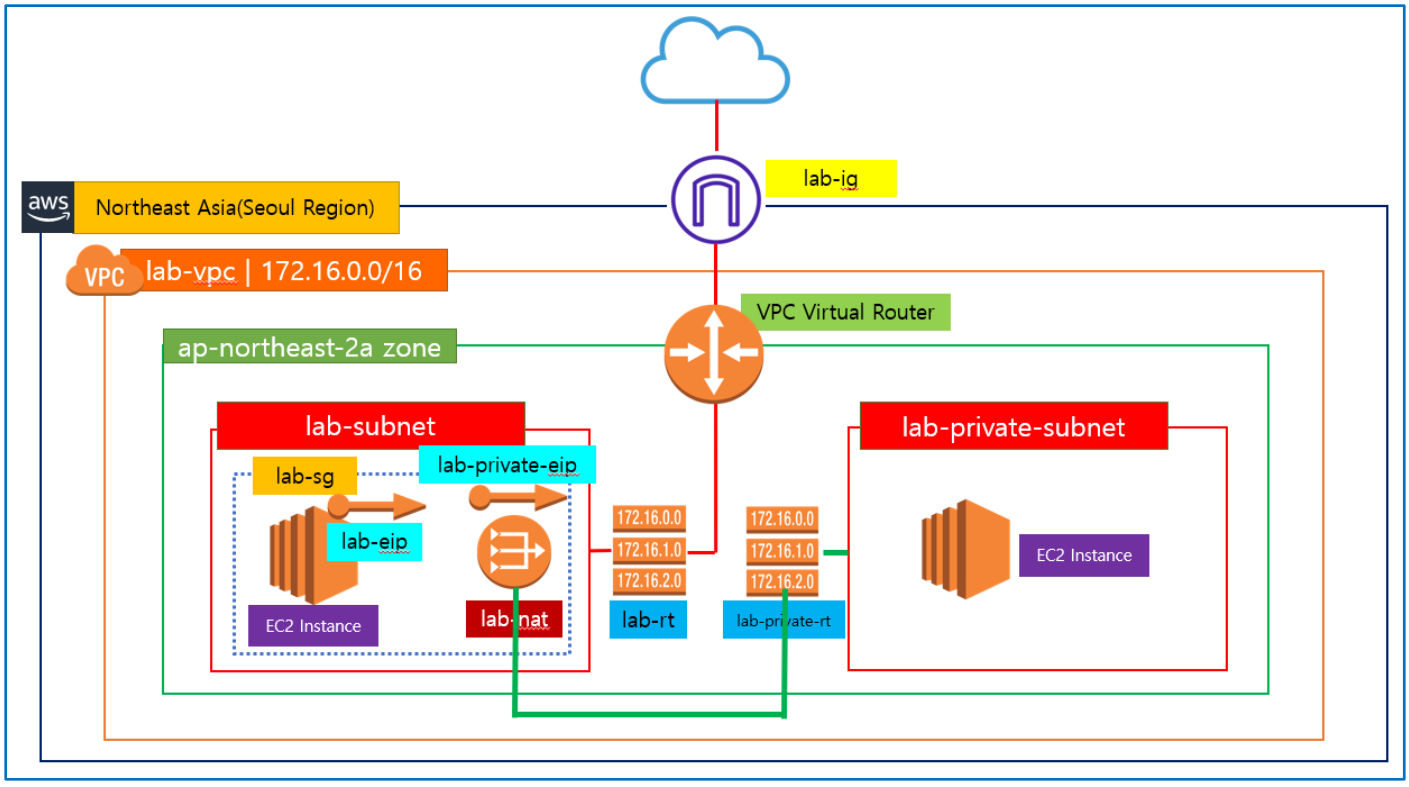
이번 실습에서는 Private Subnet에서 NAT Gateway를 통해 외부 인터넷을 사용가능 하도록 설정한다. 먼저 NAT Gateway에서 사용할 EIP를 생성하고, Private Subnet을 생성하며, NAT Gateway를 생성한다. 그 후, Private Subnet을 위한 Routing Table을 생성한다. 생성한 Routing Table을 Private Subnet에 연결하는데, 이 Routing Table은 NAT Gateway와 연결되어야 한다. 이렇게 해서 Private Subnet의 EC2 인스턴스가 외부 인터넷을 사용하기 위해 Public Subnet에 위치하고 있는 NAT Gateway를 사용하는 실습을 진행한다.

사전 준비물

AWS Free-Tier 계정

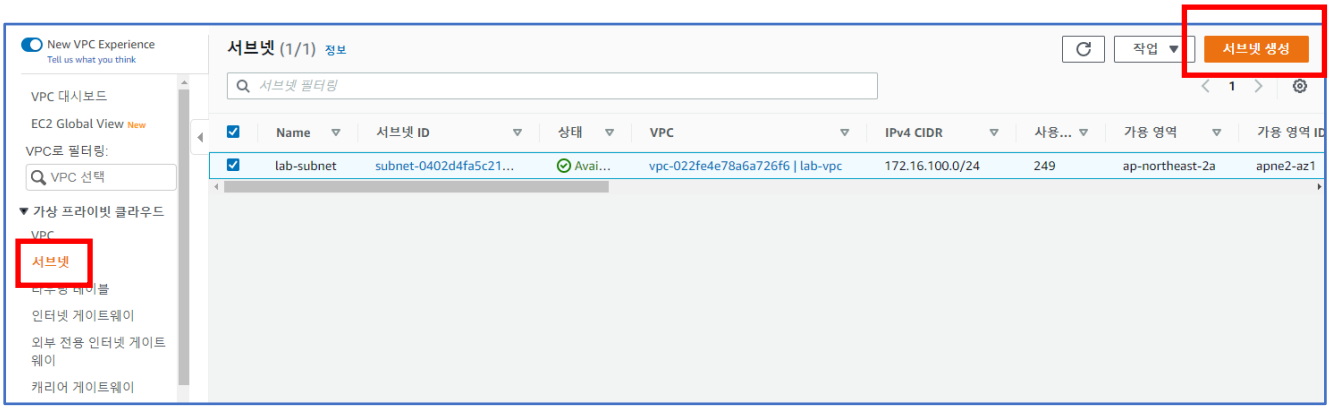
lab-vpc, lab-subnet, lab-eni, lab-ig

lab-rt, lab-nacl, lab-sg

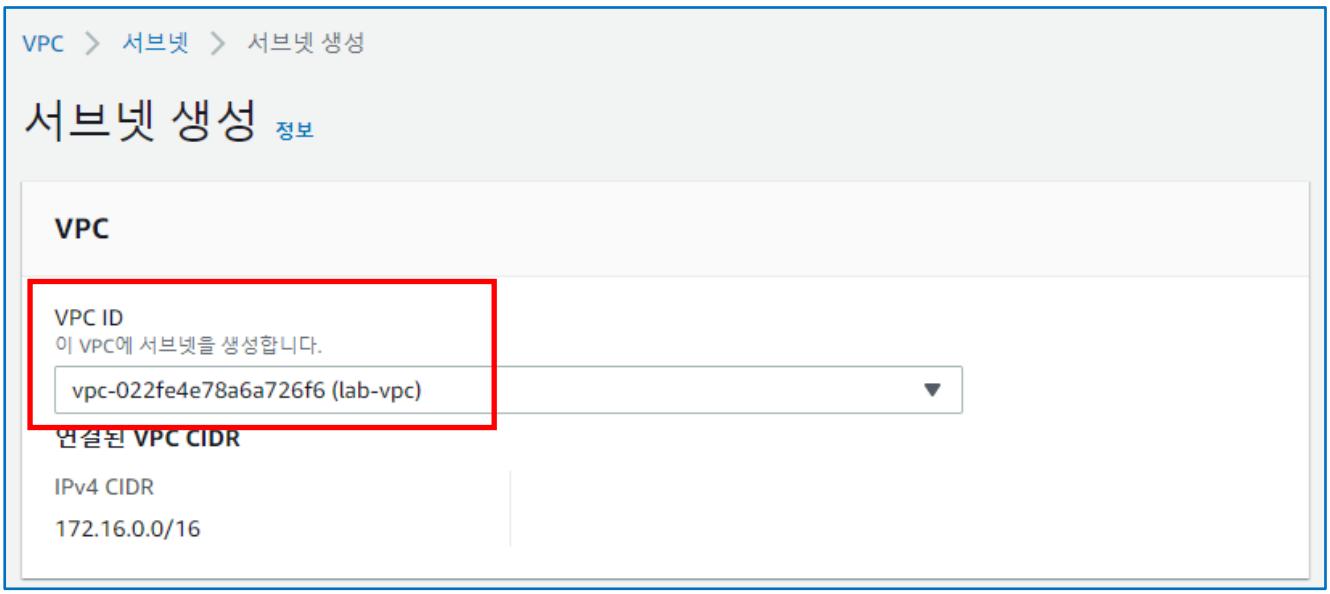


Allocate EIP & Public IP

1. **[서비스]** > **[VPC]** > **[가상 프라이빗 클라우드]** > **[서브넷]**을 클릭하여 **[서브넷]** 페이지로 이동한다. 현재 앞의 Lab에서 생성한 **lab-subnet**이 보인다. 새로 **Private Subnet**을 생성하기 위해 우측 상단의 **[서브넷 생성]** 버튼을 클릭한다.



2. **[서브넷 생성]** 페이지에서 **[VPC]**는 **lab-vpc**로 설정한다.



3. **[서브넷 설정]** 페이지에서 다음과 같이 각각의 값을 설정한 후, **[서브넷 생성]** 버튼을 클릭한다. 이번 **프라이빗 서브넷**은 앞 **Lab2**에서 생성한 **lab-subnet**과 달리 다른 **가용영역**에 설치하기로 한다.

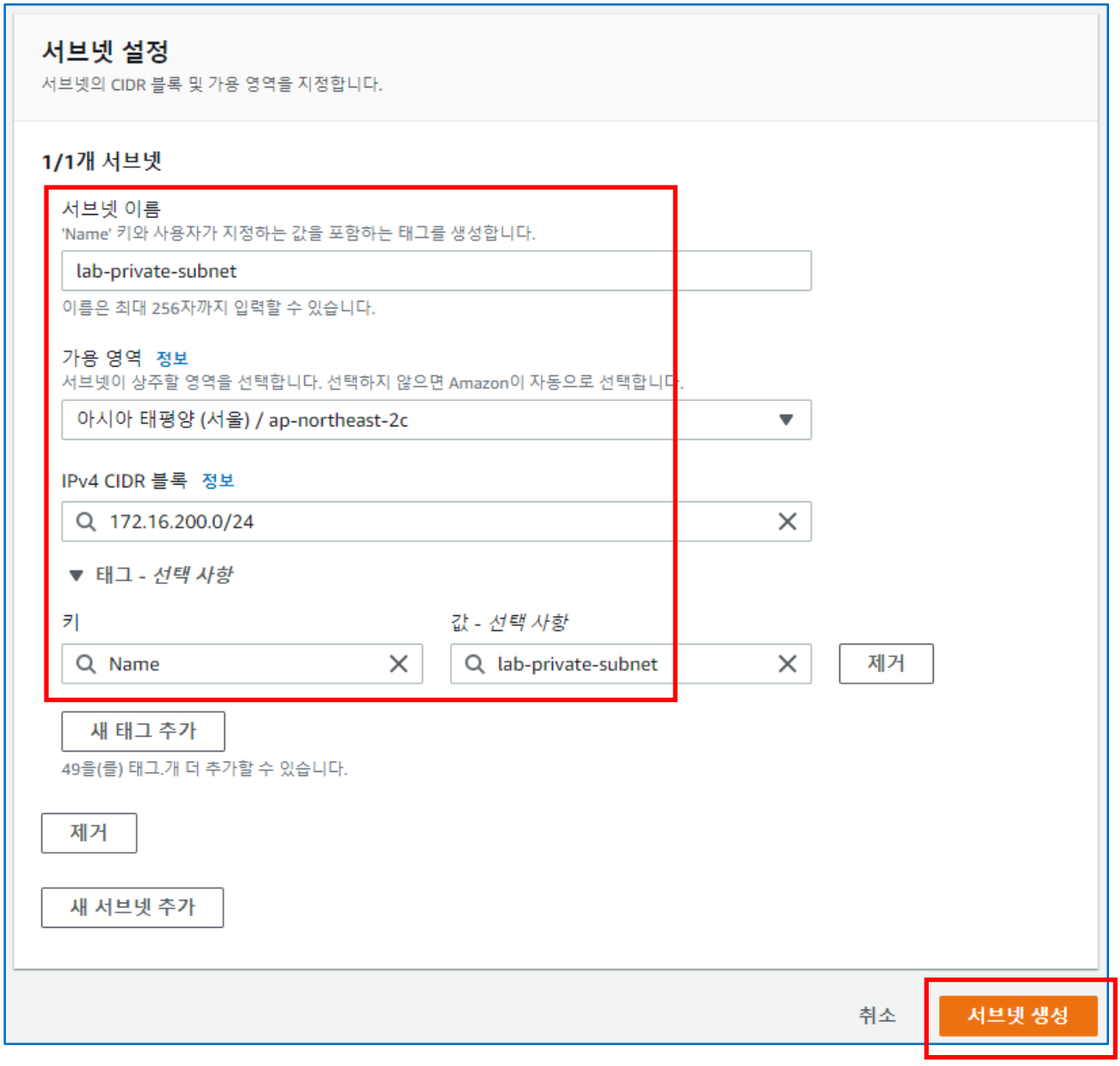
   A. **[서브넷 이름] : lab-private-subnet**

   B. **[가용 영역]** : **ap-northeast-2c**

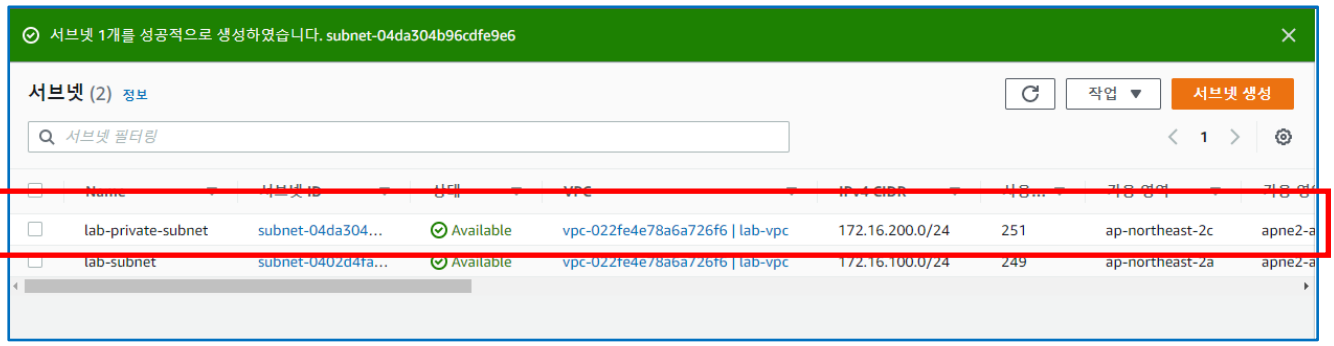
   C. **[IPv4 CIDR]** : **172.16.200.0/24**

   D. **[키] : Name**

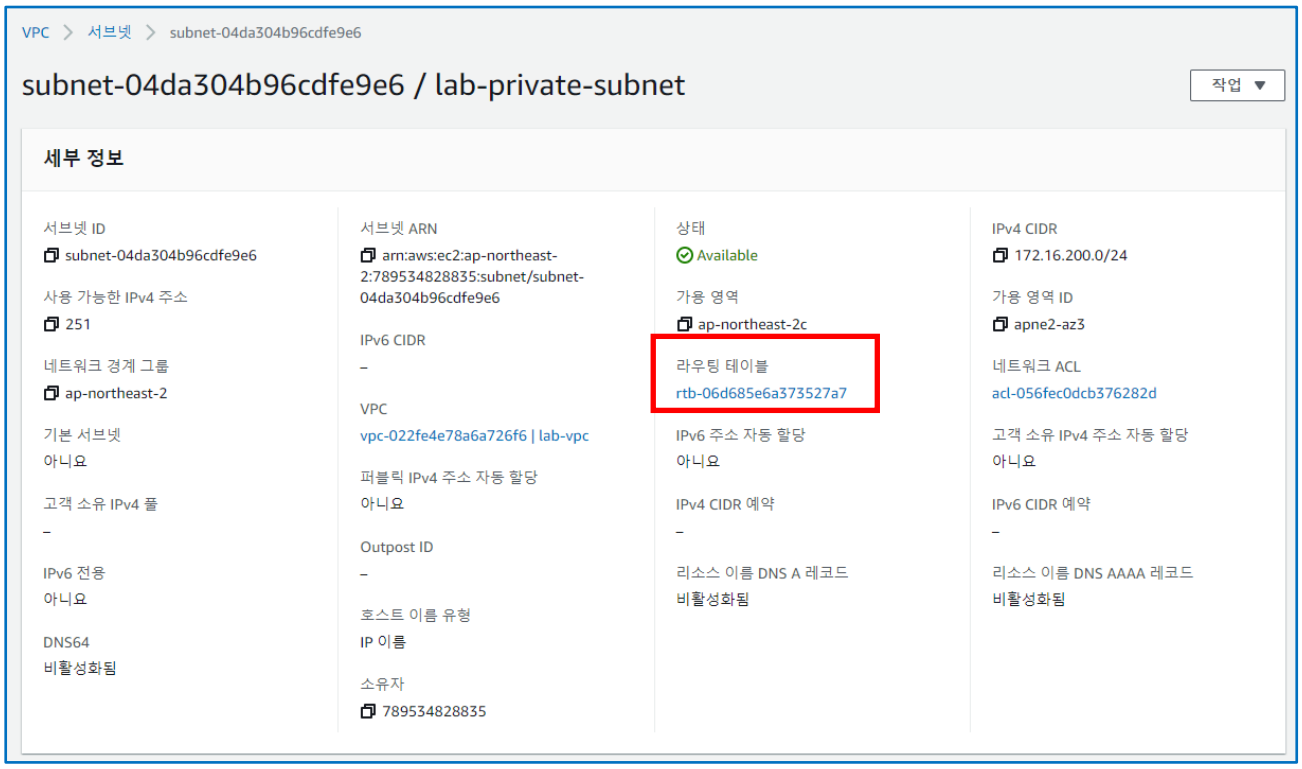
   E. **[값] : lab-private-subnet**



4. **프라이빗 서브넷**이 잘 생성 되었다.

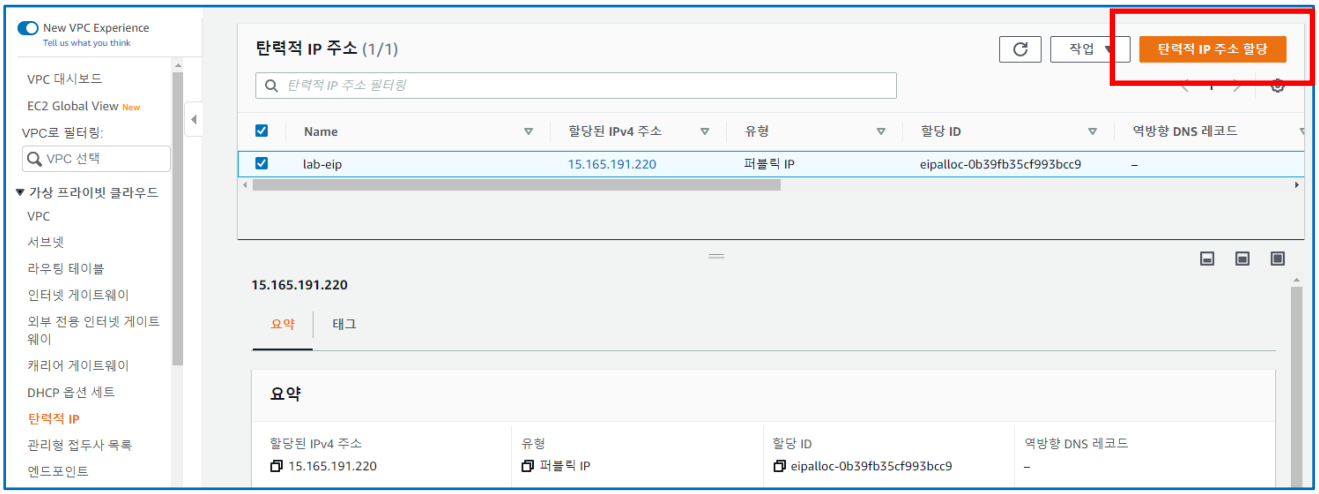


5. 방금 생성한 **프라이빗 서브넷**의 상세 페이지로 가보면 이 **서브넷**은 아직 **기본 라우팅 테이블**을 사용하고 있음을 알 수 있다.

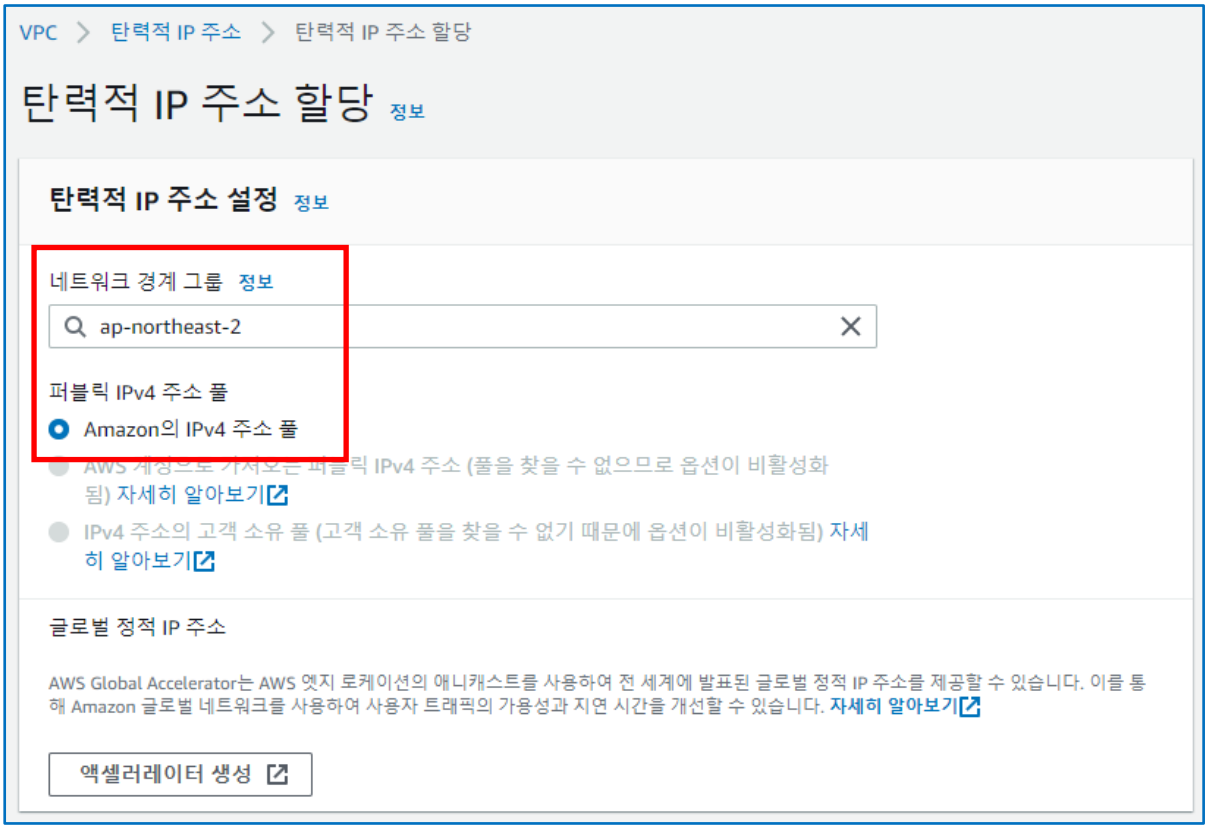


NAT Gateway 생성하기

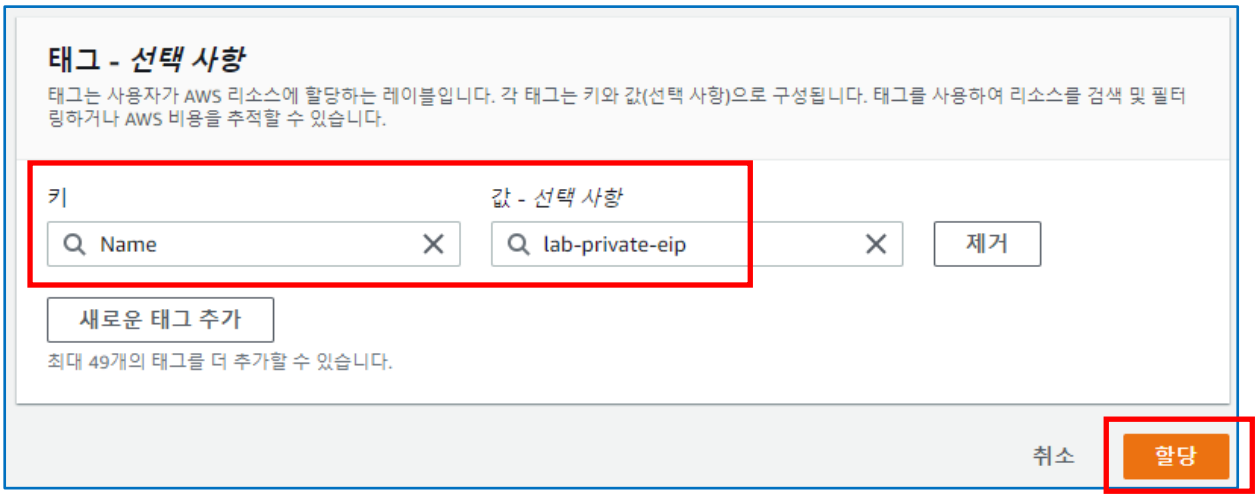
1. **NAT 게이트웨이**를 생성하기 전에 **탄력적 IP**를 생성해야 한다. **[서비스]** > **[VPC]** > **[가상 프라이빗 클라우드]** > **[탄력적 IP]**를 클릭하여 해당 페이지로 이동한다. 앞 Lab에서 생성한 **lab-eip**가 확인된다. 새로 생성하기 위해 우측 상단의 **[탄력적 IP 주소 할당]**을 클릭한다.



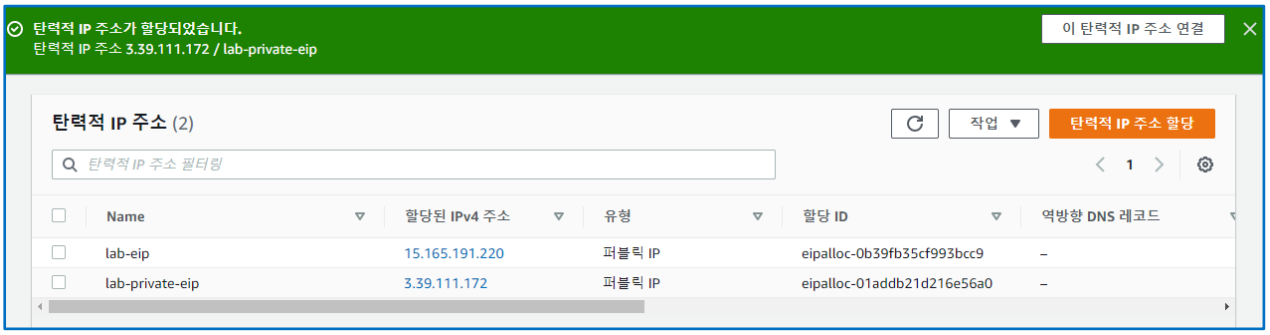
2. **[탄력적 IP 주소 할당]** 페이지에서 기본 값 그대로 사용하기로 한다. 페이지를 스크롤다운한다.



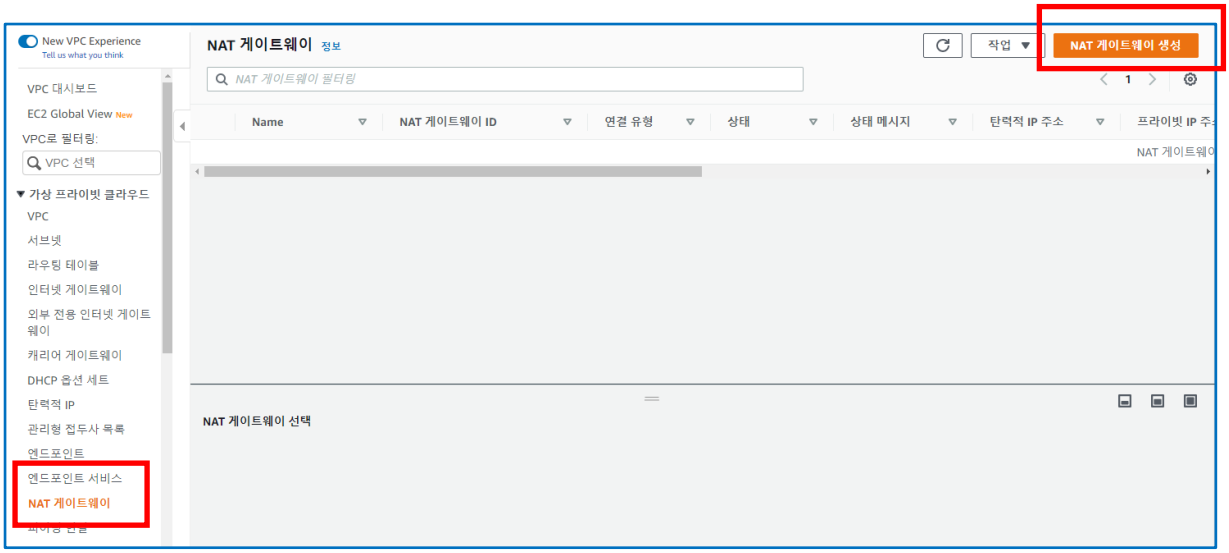
3. **[태그]** 섹션에서 **[새로운 태그 추가]**를 클릭하여, **[키]**를 **Name**, **[값]**을 **lab-private-eip**로 입력한 후, **[할당]** 버튼을 클릭한다. **탄력적 IP**가 **private**이 되지 않지만 이름을 구별하기 위해 **private** 단어를 넣었다.



4. 새로 **탄력적 IP** 주소가 할당되었다.



5. **프라이빗 서브넷**에서 외부 인터넷 구간 통신을 하려면 **NAT 게이트웨이**를 생성하고 **VPC**와 연결해야 한다. **[서비스]** > **[VPC]** > **[가상 프라이빗 클라우드]** > **[NAT 게이트웨이]** 를 클릭하여 **NAT 게이트웨이** 페이지로 이동한다. 우측 상단의 **[NAT 게이트웨이 생성]**을 클릭한다.



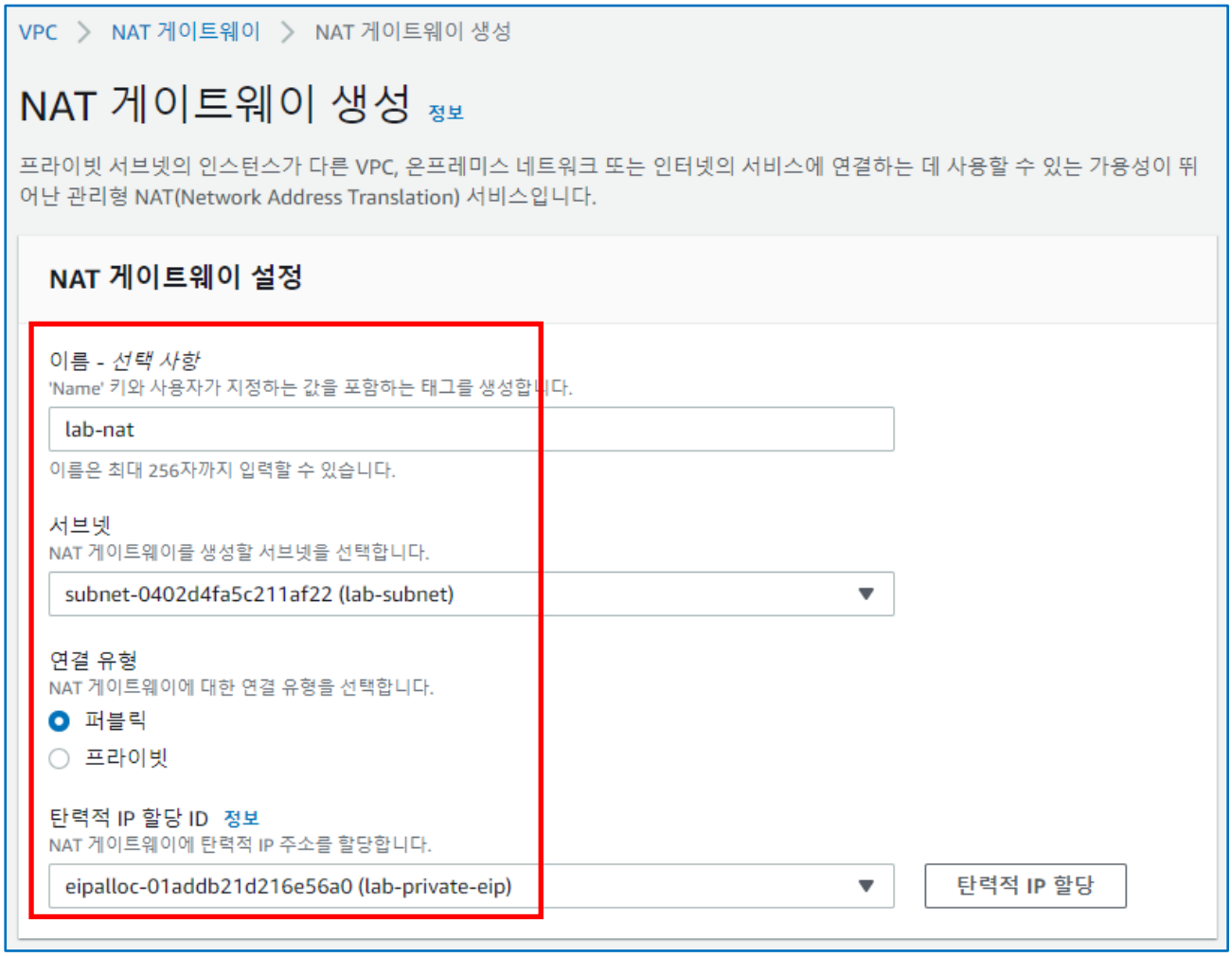
6. **[NAT 게이트웨이 생성]**페이지에서 다음의 각각의 값을 설정한다. 여기서 중요한 것은 지금 생성하는 **NAT 게이트웨이**의 위치는 **퍼블릭 서브넷**에 위치해야 한다는 것이다. 이렇게 하면 **프라이빗 서브넷**에서 **퍼블릭 서브넷**에 속해있는 **NAT 게이트웨이**를 통해 인터넷으로 나갈 수 있기 때문이다.

   A. **[이름]** : **lab-nat**

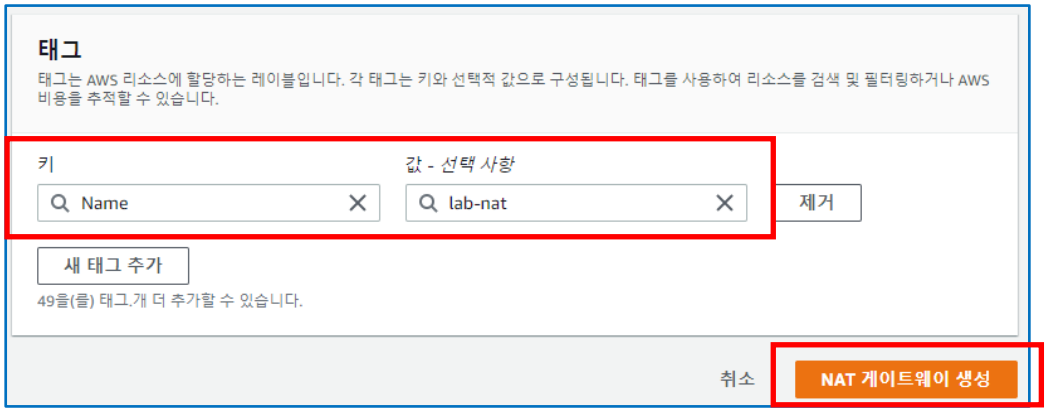
   B. **[서브넷]** : **lab-subnet**

   C. **[연결 유형]** : **퍼블릭**

   D. **[탄력적 IP 할당 ID]** : **lab-private-eip**



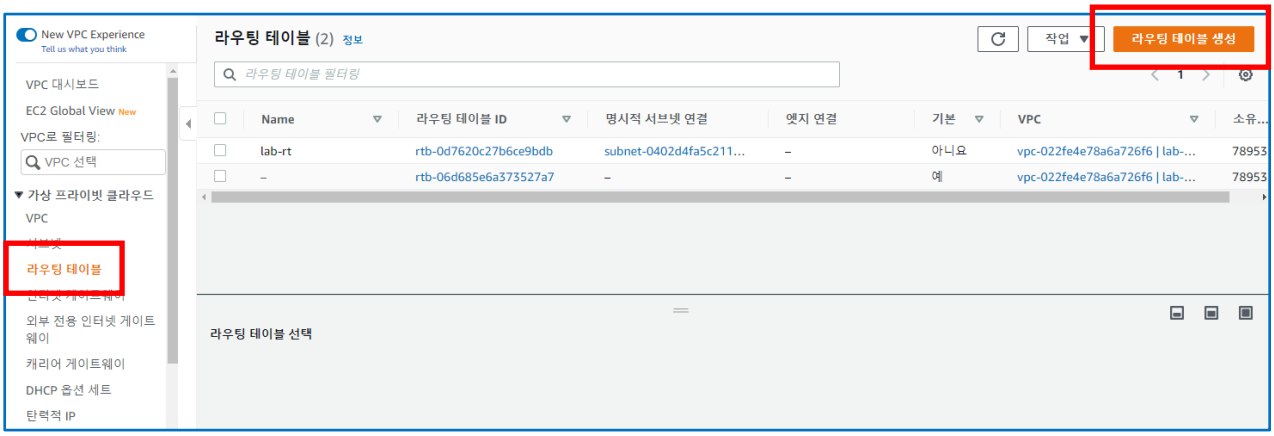
7. **[태그]** 섹션에서 자동으로 설정된 값을 확인하고 **[NAT 게이트웨이 생성]** 버튼을 클릭한다.



8. 성공적으로 **NAT 게이트웨이**가 생성되었다.

Private Routing Table 생성 및 Subnet 연결

1. 기존에 생성한 **lab-vpc**에 **프라이빗 라우팅 테이블**을 생성하기 위해 **[서비스]** > **[VPC]** > **[가상 프라이빗 클라우드]** > **[라우팅 테이블]**을 클릭하여 **라우팅 테이블** 페이지로 이동한다. 우측 상단에 있는 **[라우팅 테이블 생성]** 버튼을 클릭한다.

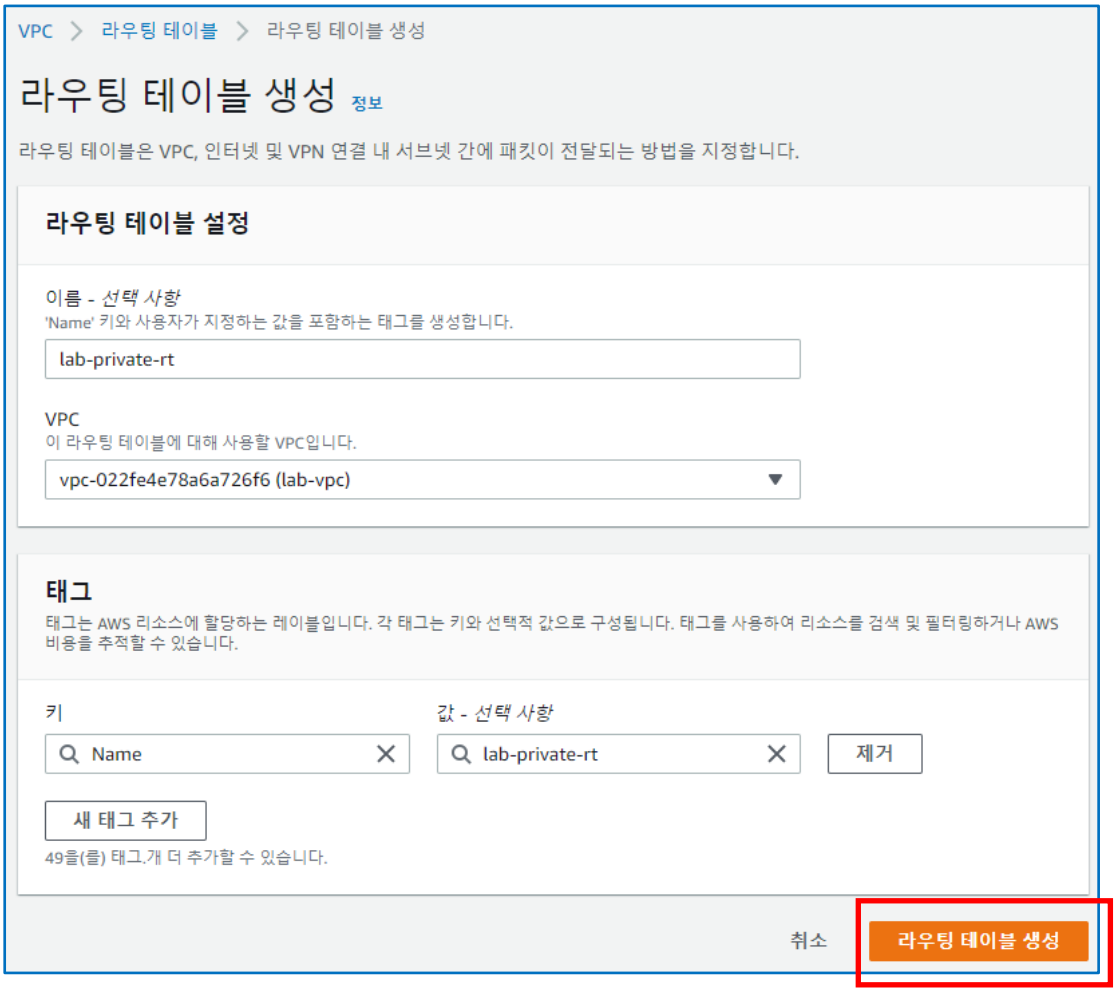


2. **[라우팅 테이블 생성]** 페이지에서 다음과 같이 각각의 값을 설정한 후 **[라우팅 테이블 생성]** 버튼을 클릭한다.

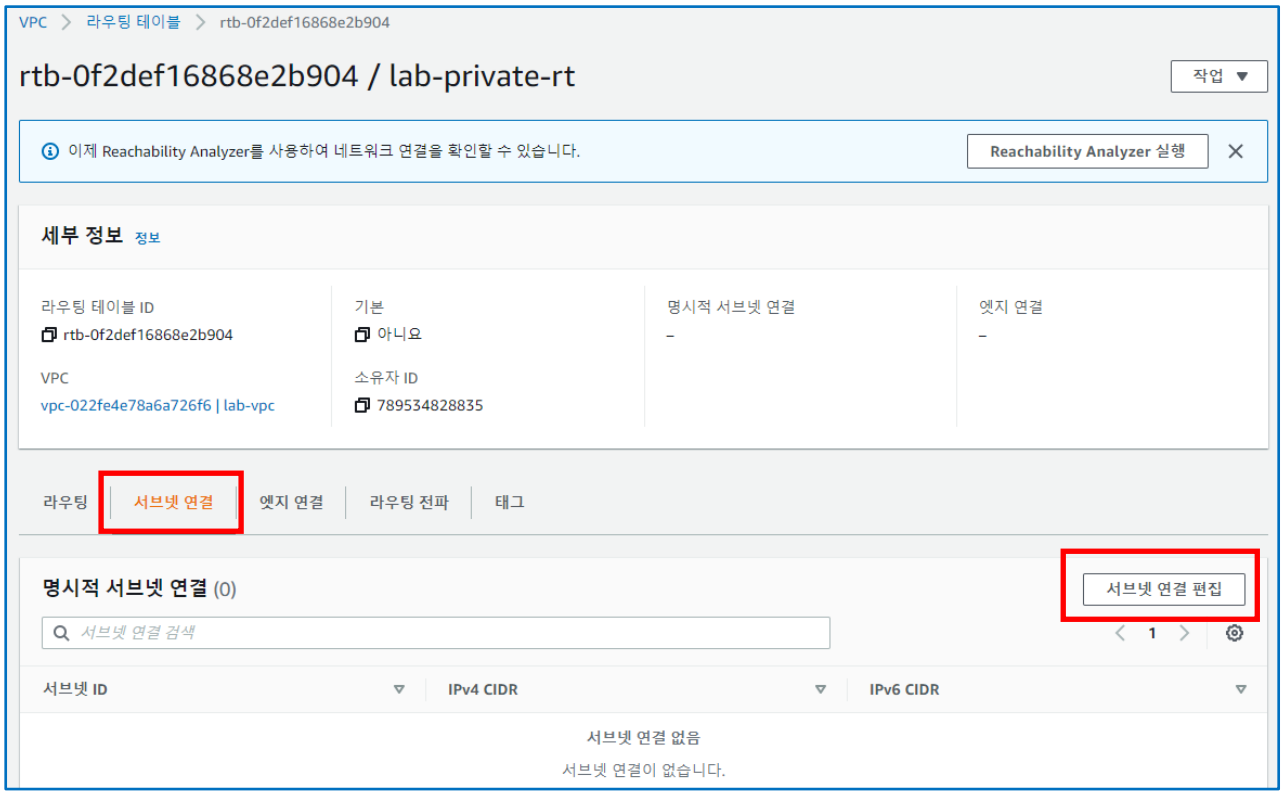
   A. **[이름]** : **lab-private-rt**

   B. **[VPC]** : **lab-vpc**

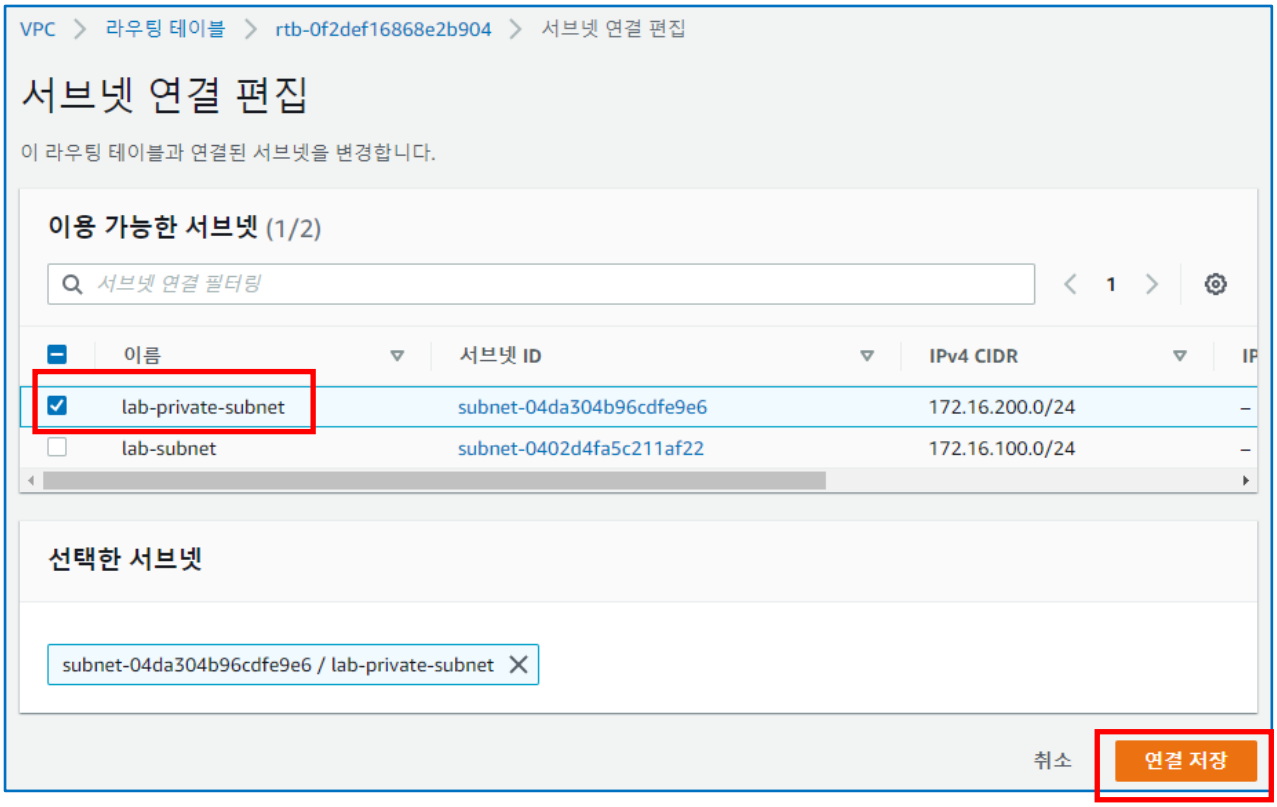
   C. **[태그]** : **Name/lab-private-rt**



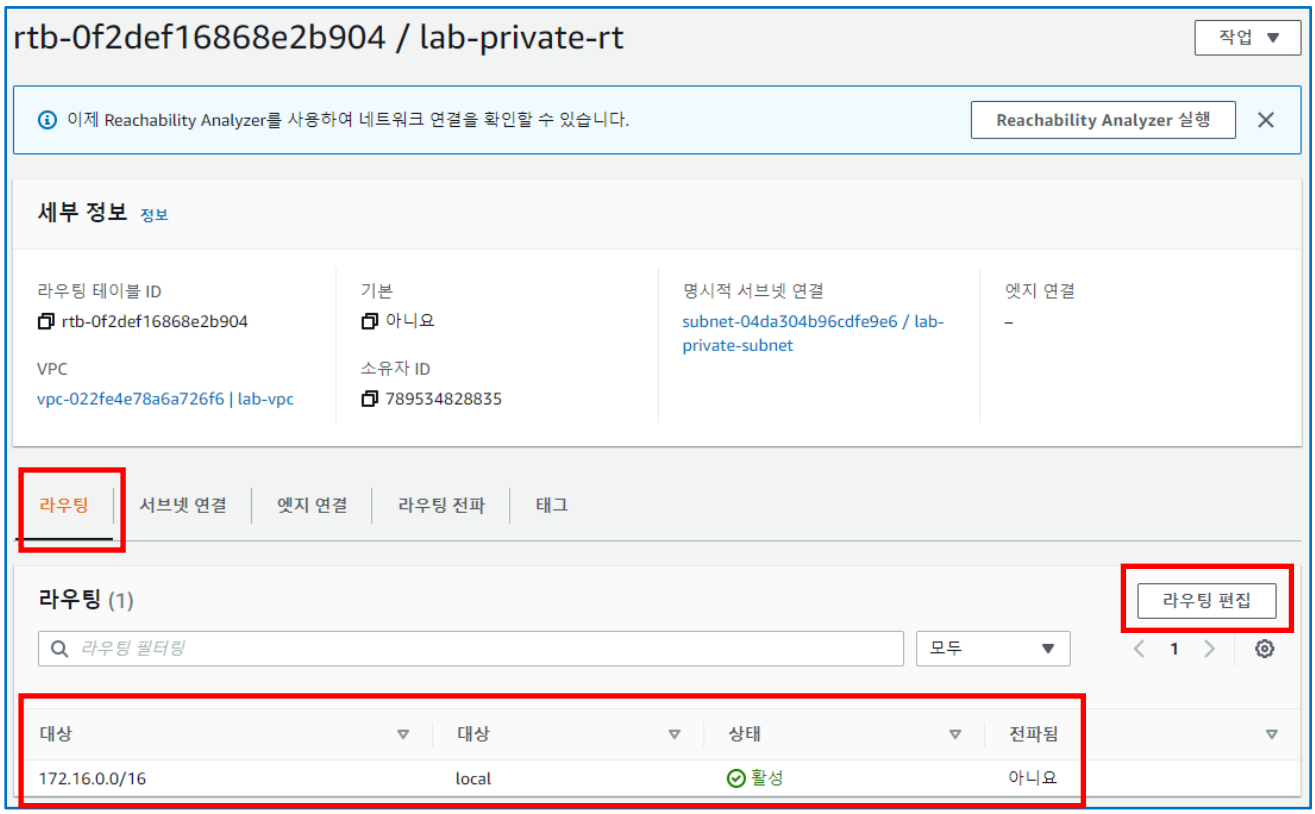
3. **라우팅 테이블** 생성 후, 연결된 **서브넷**이 없기 때문에 **서브넷**에 연결하기 위해 **[서브넷 연결]** 탭의 **[서브넷 연결 편집]**을 클릭한다.



4. **[서브넷 연결 편집]** 페이지에서 **이용 가능한 서브넷** 목록 중 방금 생성한 **lab-private-subnet**을 체크하고 **[연결 저장]**을 클릭한다.



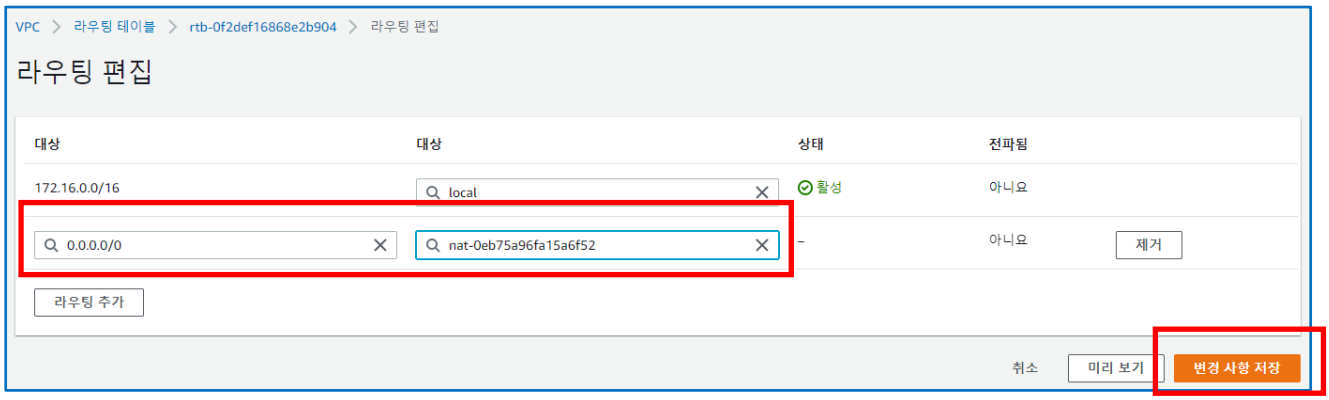
5. 이렇게 해서 **프라이빗 서브넷**은 **기본 라우팅 테이블**이 아닌 **프라이빗 라우팅 테이블**과 연결이 되었다. 하지만 **라우팅** 정보를 보면 아직 외부 인터넷 구간 통신을 위한 **라우팅** 경로가 없는 것을 확인할 수 있다. **[라우팅 편집]** 버튼을 클릭한다.



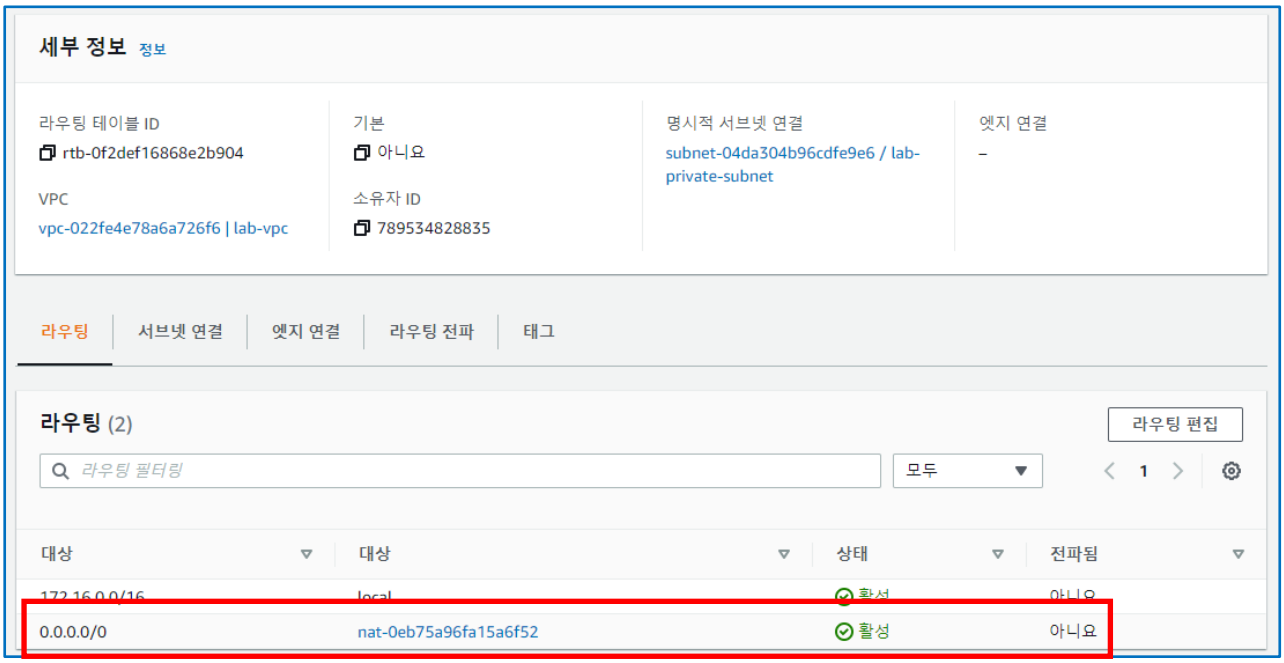
6. **[라우팅 편집]** 페이지에서 **[라우팅 추가]**를 클릭하여 다음과 같이 값을 설정한 후, **[변경 사항 저장]**을 클릭한다.

   A. [대상] : 0.0.0.0/0

   B. **[대상] : NAT 게이트웨이 > lab-nat**



7. **프라이빗 라우팅 테이블**에 **NAT 게이트웨이**를 통해 인터넷과 통신할 수 있는 **라우팅** 경로를 추가하였다.



EC2 인스턴스 생성하여 인터넷 통신 검증하기

1. 다음과 같이 **al-webserver-ec2**를 생성했다. 주의할 점은 **퍼블릭 IP 자동 할당**은 기본값 **비활성화** 그대로 사용한다.

   A. **[OS]** : **Amazon Linux 2 AMI (HVM) – Kernel 5.10, SSD Volume Type, 64비트**

   B. [인스턴스 유형] : **t2.micro**

   C. **[네트워크 설정]** : **lab-vpc**, **lab-private-subnet**

   D. **[사용자 데이터]** : **userdata.txt** 참조

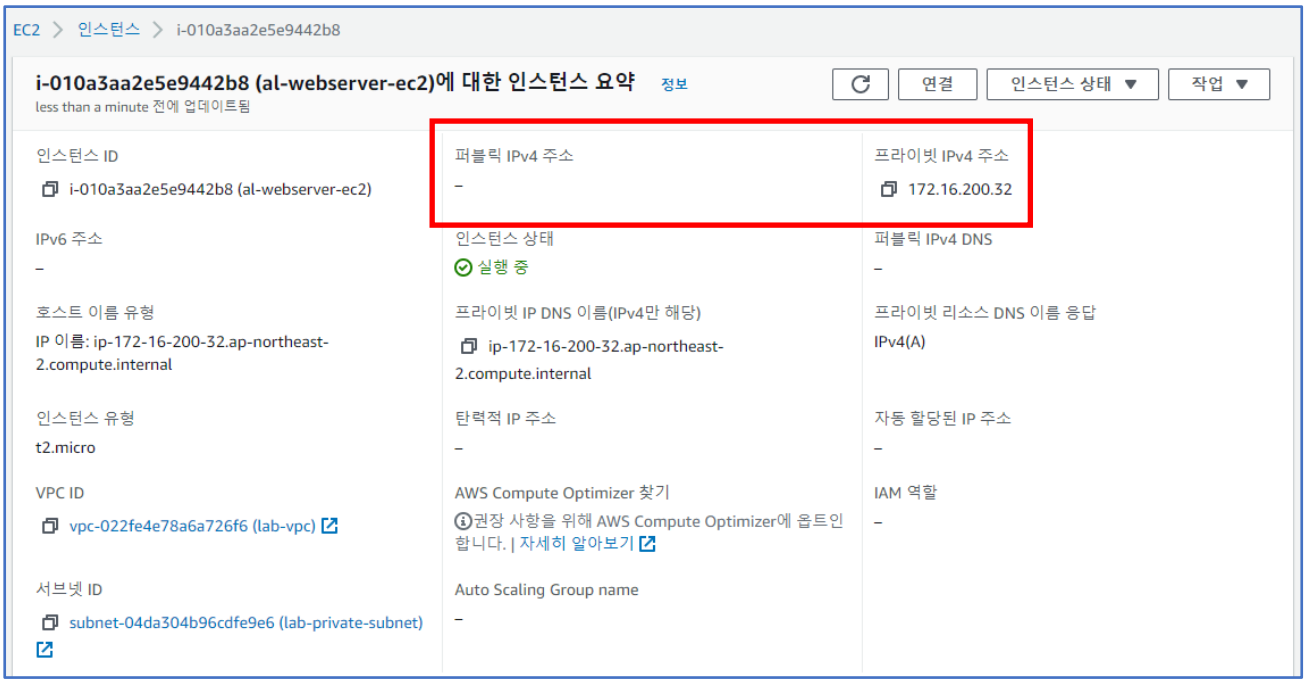
   E. **[EBS]** : **범용 SSD(gp2) 30GiB**

   F. **[태그]** : **Name/al-webserver-ec2**

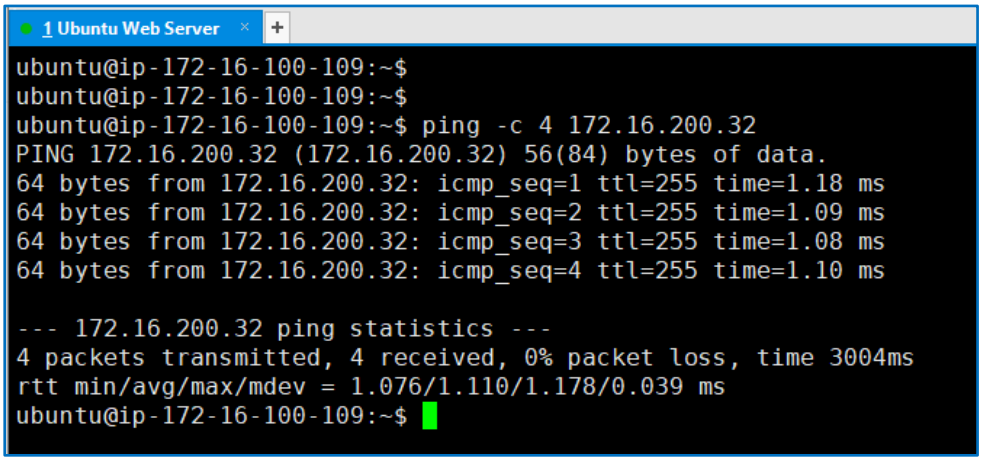
   G. **[보안 그룹]** : **기존 보안 그룹 선택/lab-sg**

   H. **[키 페어]** : **키 페이 생성하지 않음.**

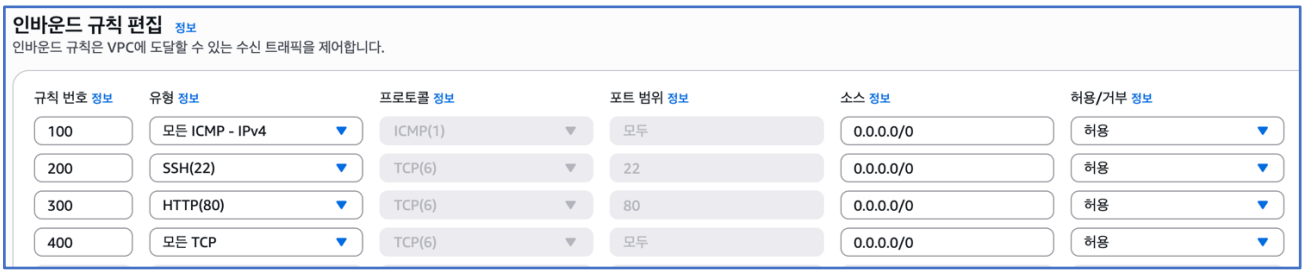
2. 아래 그림에서 보면 방금 생성한 EC2 **인스턴스**의 **퍼블릭 IPv4 주소**가 없다는 것이다. **프라이빗 IPv4 주소**만 확인할 수 있다. 그렇기 때문에 방금 생성한 **인스턴스**에 **PING 테스트**나 **SSH**를 통한 연결을 불가능하다.



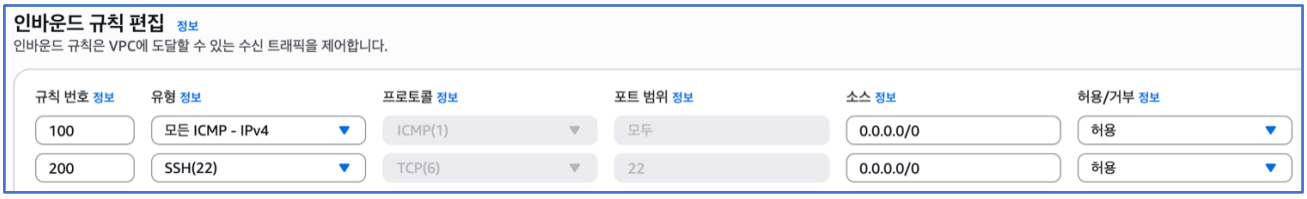
3. 그래서 **퍼블릭 서브넷**에 속해있는 **Lab6**에서 생성했던 인스턴스를 **SSH**로 연결해서 로컬 통신을 통해 방금 생성한 **프라이빗 EC2**에 연결해보자. 로컬 통신은 문제 없이 **PING 테스트** 성공이라는 것을 확인할 수 있다.



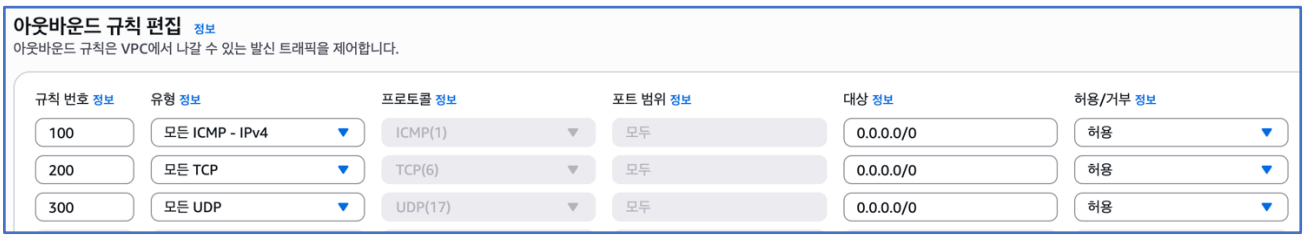
4. 퍼블릭 서브넷에서 프라이빗 서브넷으로 SSH 연결하려면 추가작업을 해야 한다. 먼저 Lab5에서 생성했던 lab-nacl의 인바운드 규칙을 다음과 같이 설정한다.



5. 새로 lab-private-nacl을 생성하여 lab-private-subnet과 연결한 다음, 다음과 같이 인바운드 규칙을 생성한다.



6. 아웃바운드 규칙은 일반적인 규칙대로 생성한다.

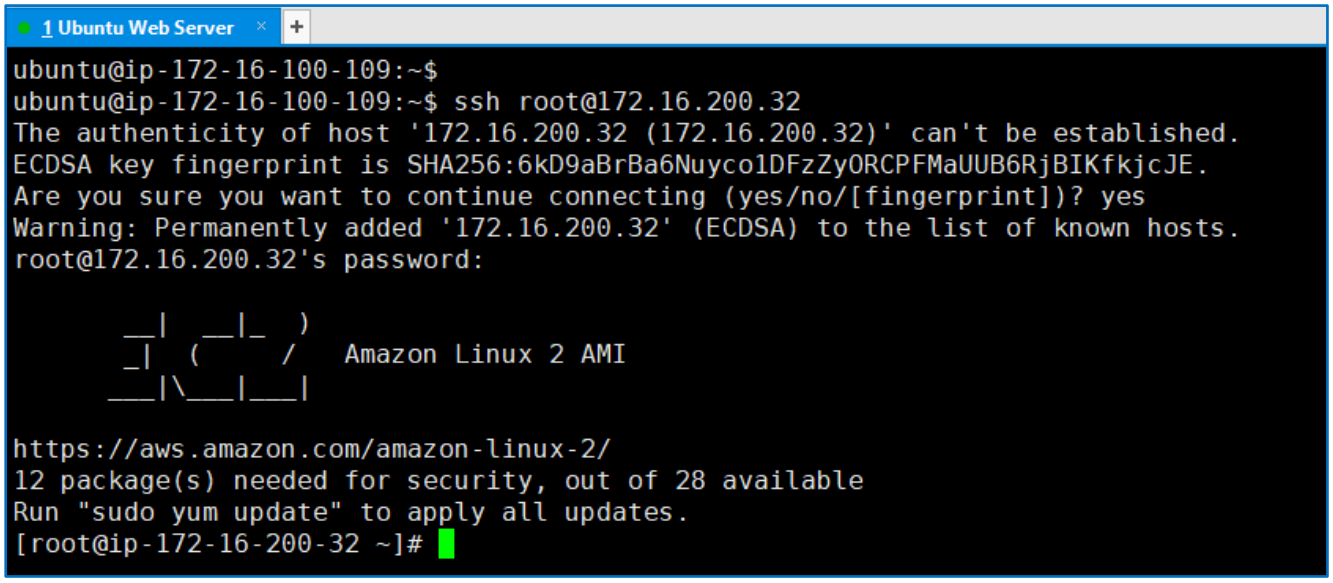


7. **SSH** 연결을 해보자. 인스턴스 생성시 지정했던 사용자 데이터에서 패스워드를 **qwer1234**로 설정했었다.

   A. **$ ssh root@172.16.200.32(프라이빗 IP)**

   B. **Are you sure you want to continue connecting (yes/no/[fingerprint])? yes**

   C. **password : qwer1234**



8. 로컬 통신을 통해 새로 생성한 인스턴스에 잘 연결되었다. 인터넷 연결이 되는지 확인해 보자. 아래의 그림과 같이 구글에 대한 PING 테스트가 성공적이다. 즉, **프라이빗 서브넷**에 있는 **al-webserver-ec2** 인스턴스는 외부에서는 접근할 수 없지만, **NAT 게이트웨이**를 통해 인터넷 통신이 가능함을 확인할 수 있다.