

Lab5. EC2를 이용해서 Linux Server Instance에 Apache Web Server 올리기

목적

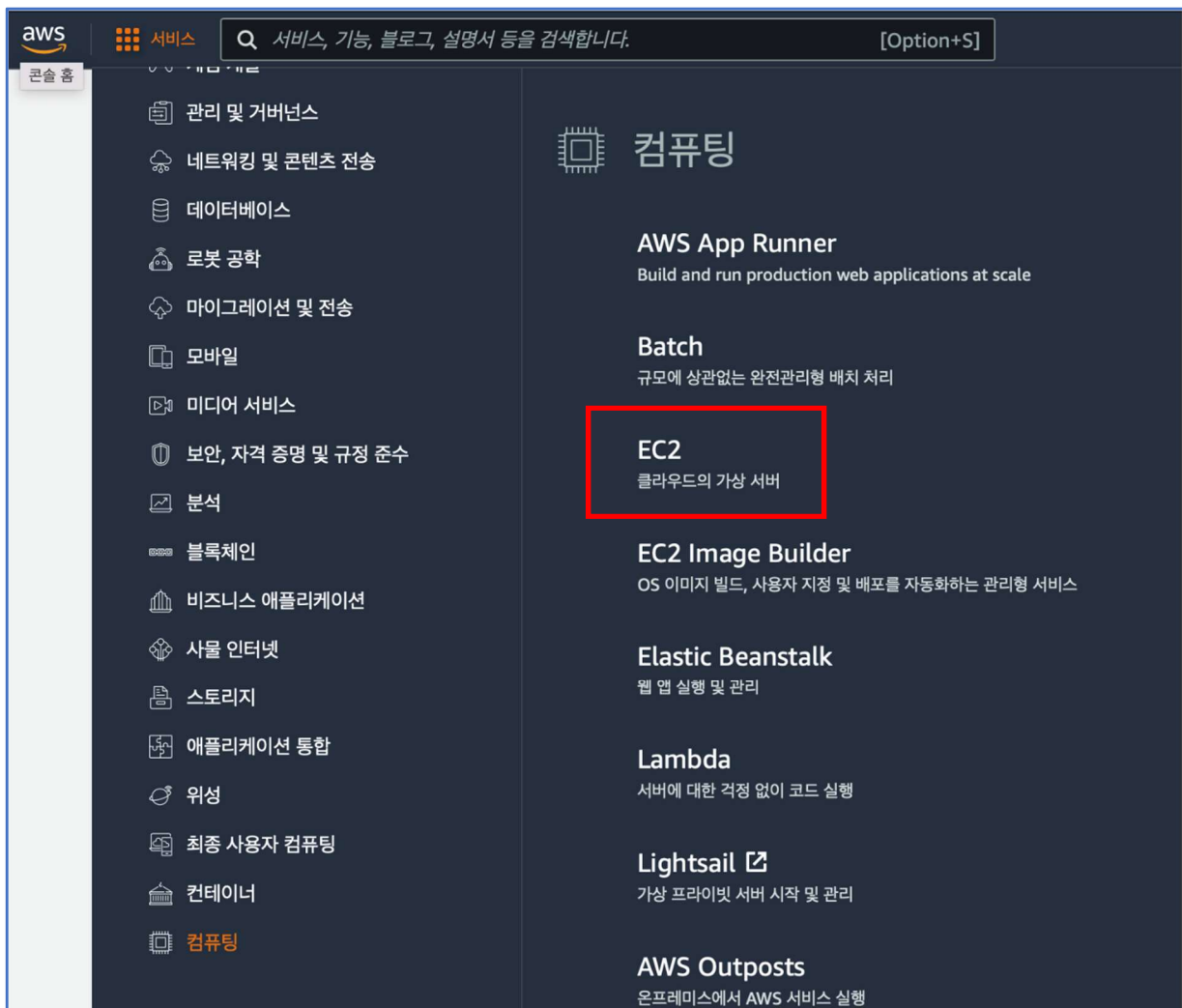
앞의 Lab4에서 Amazon EC2(Elastic Compute Cloud) Windows Server 인스턴스를 통해 Microsoft Web Server를 설치하고 간단한 홈페이지를 작성해서 웹브라우저에서 확인했다. 이제 같은 Amazon EC2환경에서 Lab3에서 생성한 Linux Server 인스턴스를 이용해서 Apache Web Server를 설치하고 간단한 웹페이지를 만들어본다. 이 학습은 AWS Free-Tier를 활용하여 진행한다.

사전 준비물

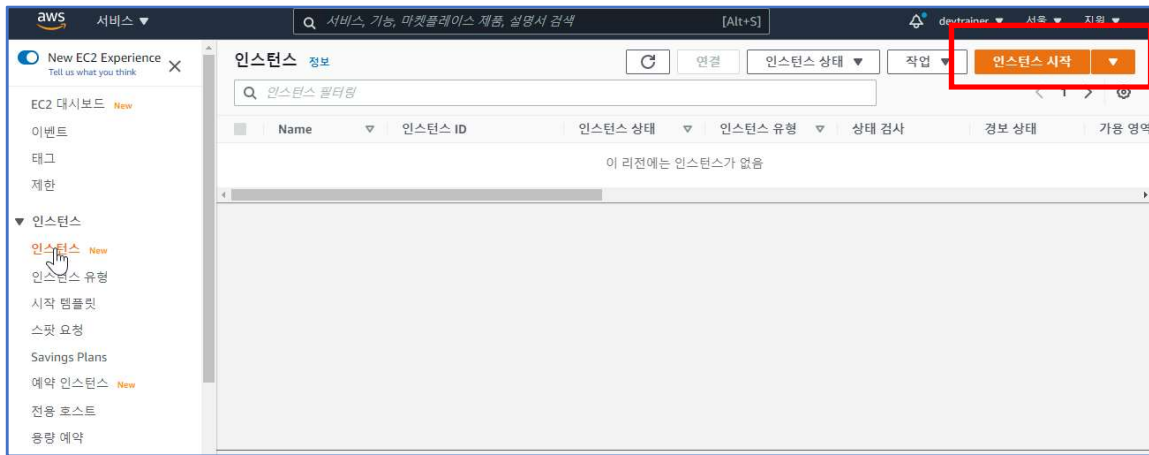
AWS Free-Tier 계정

사용자 데이터로 Ubuntu Server + Apache Web Server 설치하기

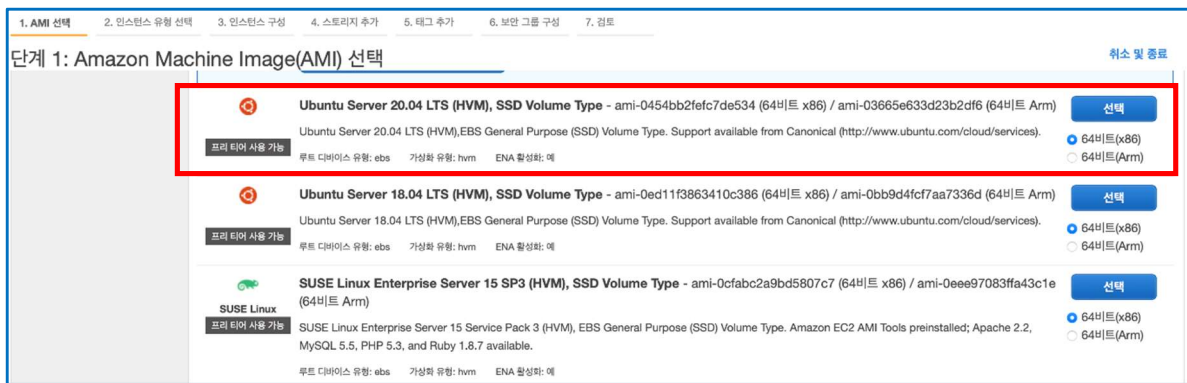
1. AWS 콘솔에 로그인 후, EC2 페이지로 이동한다.



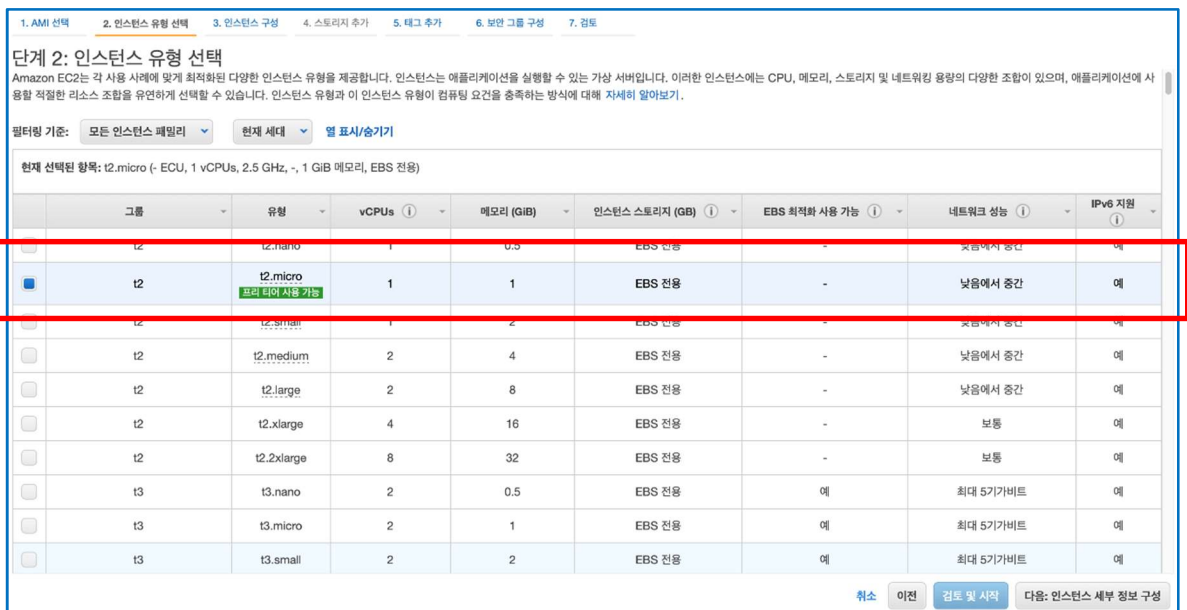
2. 우측 상단의 [인스턴스 시작] 오렌지 색 버튼을 클릭한다.



3. [단계 1: Amazon Machine Image(AMI) 선택] 페이지에서 [Ubuntu Server 20.04 LTS(HVM), SSD Volume Type] 서버를 찾은 후 [64비트(x86)]이 선택되어 있는 것을 확인한 후 [선택] 버튼을 클릭한다.



4. [단계 2: 인스턴스 유형 선택] 페이지에서, [t2.micro]를 선택 후, [다음: 인스턴스 세부 정보 구성] 버튼을 클릭한다.



5. [단계 3:인스턴스 세부 정보 구성] 페이지에서, [네트워크]는 이전 Lab에서 생성한 lab-vpc, [서브넷]은 public subnet, [퍼블릭 IP 자동 할당]은 활성화를 선택한다.

1. AMI 선택 2. 인스턴스 유형 선택 3. 인스턴스 구성 4. 스토리지 추가 5. 태그 추가 6. 보안 그룹 구성 7. 검토

단계 3: 인스턴스 세부 정보 구성

기본 VPC 없음. 다른 VPC 또는 새 기본 VPC 생성을(를) 선택합니다.

요구 사항에 적합하게 인스턴스를 구성합니다. 동일한 AMI의 여러 인스턴스를 시작하고 스카프 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 ☐ 스카프 인스턴스 요청

네트워크 vpc-00a61e8f2edc671ff | lab2-vpc 새 VPC 생성
기본 VPC가 없습니다. 새 기본 VPC 생성.

서브넷 subnet-099a38aacaeb51a0e | lab2-subnet-public1 새 서브넷 생성
4096개 IP 주소 사용 가능

퍼블릭 IP 자동 할당 활성화

호스트 이름 유형 서브넷 사용 설정 (IP 이름)

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

배치 그룹 ☐ 배치 그룹에 인스턴스 추가

용량 예약 열기

취소 이전 검토 및 시작 다음: 스토리지 추가

6. 계속 페이지를 아래로 스크롤하여 [고급 세부 정보] 섹션의 [사용자 데이터]에 다음과 같은 사용자 데이터를 입력한다. 그리고 [다음:스토리지 추가]를 클릭하여 4단계로 이동한다.

```
1 #!/bin/bash
2 sudo apt update
3 sudo apt install -y apache2
4 sudo a2enmod ssl
5 sudo a2ensite default-ssl.conf
6 sudo systemctl reload apache2
7 sudo mv /var/www/html/index.html /var/www/html/index.bak
8 sudo bash -c 'echo "<html><h1>Hello, AWS World!</h1></html>" > /var/www/html/index.html'
```

1. AMI 선택 2. 인스턴스 유형 선택 3. 인스턴스 구성 4. 스토리지 추가 5. 태그 추가 6. 보안 그룹 구성 7. 검토

단계 3: 인스턴스 세부 정보 구성

eth0 새 네트워크 인터페이스 subnet-099a38aac 자동 할당 IP 추가

선택한 서브넷은 IPv6 CIDR이 없으므로 IPv6를 지원하지 않습니다.

디바이스 추가

고급 세부 정보

Enclave ☐ 활성화

메타데이터 액세스 가능 활성화됨

메타데이터 버전 V1 및 V2(토른 선택 사항)

메타데이터 토큰 응답 옵션 제한 1

Allow tags in metadata 비활성화됨

사용자 데이터 ☒ 텍스트로 ☐ 파일로 ☐ 입력이 이미 base64로 인코딩됨

```
sudo a2enmod ssl
sudo a2ensite default-ssl.conf
sudo systemctl reload apache2
sudo mv /var/www/html/index.html /var/www/html/index.bak
sudo bash -c 'echo "<html><h1>Hello, AWS World!</h1></html>" > /var/www/html/index.html'
```

취소 이전 검토 및 시작 다음: 스토리지 추가

7. [단계 4:스토리지 추가] 페이지에서, Linux Server는 스토리지 크기가 8GiB로 맞춰져 있는데, Free-Tier 자격으로 최대 사용할 수 있는 스토리지 크기는 30GB이기 때문에 Linux Server 인스턴스 스토리지 크기를 **30GiB**로 설정한다. [다음:태그 추가] 버튼을 클릭한다.

1. AMI 선택 2. 인스턴스 유형 선택 3. 인스턴스 구성 4. 스토리지 추가 5. 태그 추가 6. 보안 그룹 구성 7. 검토

단계 4: 스토리지 추가

인스턴스가 다음 스토리지 디바이스 설정으로 시작됩니다. 추가 EBS 볼륨 및 인스턴스 스토어 볼륨을 인스턴스에 연결하거나 루트 볼륨의 설정을 편집할 수 있습니다. 인스턴스를 시작한 후 추가 EBS 볼륨을 연결할 수도 있지만, 인스턴스 스토어 볼륨은 연결할 수 없습니다. Amazon EC2의 스토리지 옵션에 대해 [자세히 알아보십시오](#).

볼륨 유형	디바이스	스냅샷	크기(GiB)	볼륨 유형	IOPS	처리량(MB/초)	종료 시 삭제	암호화
루트	/dev/sda1	snap-0fdb1de3deef1ceb0	30	범용 SSD(gp2)	100/3000	해당 사항 없음	<input checked="" type="checkbox"/>	암호화5

[새 볼륨 추가](#)

프리 티어 사용 가능 고객은 최대 30GB의 EBS 범용(SSD) 또는 마그네틱 스토리지를 사용할 수 있습니다. 프리 티어 자격 및 사용량 제한에 대해 [자세히 알아보기](#).

▼ Shared file systems

You currently don't have any file systems on this instance. Select "Add file system" button below to add a file system.

[Add file system](#)

[취소](#) [이전](#) [검토 및 시작](#) [다음: 태그 추가](#)

8. [태그 추가] 버튼을 누른다.

1. AMI 선택 2. 인스턴스 유형 선택 3. 인스턴스 구성 4. 스토리지 추가 5. 태그 추가 6. 보안 그룹 구성 7. 검토

단계 5: 태그 추가

태그는 대문자를 구별하는 키-값 페어로 이루어져 있습니다. 예를 들어 키가 Name이고 값이 Webserver인 태그를 정의할 수 있습니다. 태그 속성은 볼륨, 인스턴스 또는 둘 다에 적용될 수 있습니다. 태그는 모든 인스턴스 및 볼륨에 적용됩니다. Amazon EC2 리소스 태그 지정에 대해 [자세히 알아보기](#).

키 (최대 128자)	값 (최대 256자)	인스턴스	볼륨	네트워크 인터페이스
이 리소스에는 현재 태그가 없습니다.				

[태그 추가] 버튼 또는 Name 태그를 추가하려면 클릭합니다. [을\(를\)](#) 선택합니다. IAM 정책에 태그를 생성할 수 있는 권한이 포함되어 있는지 확인합니다.

[태그 추가](#) (최대 50개 태그)

9. [키]에 "Name"을, [값]에 "lab5-ubuntu-ec2"을 입력한 다음, [다음:보안 그룹 구성] 버튼을 클릭한다. 태그는 해당 인스턴스를 표현하는 여러 이름으로 사용될 수 있다. EC2의 이름을 붙인다고 생각하고 넣으면 된다. 여러 인스턴스가 있을 경우 이를 태그별로 구분하면 검색이나 그룹 찾기 편하므로 여기서 본인 서비스의 인스턴스를 나타낼 수 있는 값으로 등록하면 된다.

1. AMI 선택 2. 인스턴스 유형 선택 3. 인스턴스 구성 4. 스토리지 추가 5. 태그 추가 6. 보안 그룹 구성 7. 검토

단계 5: 태그 추가

태그는 대소문자를 구별하는 키-값 페어로 이루어져 있습니다. 예를 들어 키가 Name이고 값이 Webserver인 태그를 정의할 수 있습니다. 태그 복사본은 볼륨, 인스턴스 또는 둘 다에 적용될 수 있습니다. 태그는 모든 인스턴스 및 볼륨에 적용됩니다. Amazon EC2 리소스 태그 지정에 대해 [자세히 알아보기](#).

키 (최대 128자)	값 (최대 256자)	인스턴스	볼륨	네트워크 인터페이스
Name	lab5-ubuntu-ec2	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>	<input checked="" type="checkbox"/>

다른 태그 추가 (최대 50개 태그)

취소 이전 검토 및 시작 **다음: 보안 그룹 구성**

10. [단계 6:보안 그룹 구성] 페이지에서, 기본 보안 그룹 설정 정보 확인한다. 웹서버는 HTTP(80)과 HTTPS(443)의 규칙이 필요하다. 새 규칙을 추가하기 위해 [규칙 추가]를 클릭한다.

1. AMI 선택 2. 인스턴스 유형 선택 3. 인스턴스 구성 4. 스토리지 추가 5. 태그 추가 6. 보안 그룹 구성 7. 검토

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 들어 웹 서버를 설정하여 인터넷 트래픽을 인스턴스에 도달하도록 허용하려는 경우 HTTP 및 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 [자세히 알아보기](#).

보안 그룹 할당: ☒ 새 보안 그룹 생성 ☐ 기존 보안 그룹 선택

보안 그룹 이름: launch-wizard-4

설명: launch-wizard-4 created 2022-03-18T14:46:16.123+09:00

유형	프로토콜	포트 범위	소스	설명
SSH	TCP	22	사용자 지정 0.0.0.0/0	예: SSH for Admin Desktop

규칙 추가

경고

소스가 0.0.0.0/0인 규칙은 모든 IP 주소에서 인스턴스에 액세스하도록 허용합니다. 알려진 IP 주소의 액세스만 허용하도록 보안 그룹을 설정하는 것이 좋습니다.

취소 이전 검토 및 시작



11. 다음과 같이 HTTP와 HTTPS 규칙을 추가하고 [검토 및 시작]을 클릭하여 7단계 검토단계로 이동한다.

- A. [유형] : HTTP
- B. [프로토콜] : TCP
- C. [포트 범위] : 80
- D. [소스] : "위치 무관"
- E. [유형] : HTTPS
- F. [프로토콜] : TCP
- G. [포트 범위] : 443
- H. [소스] : "위치 무관"

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 설정하여 인터넷 트래픽을 인스턴스에 도달하도록 허용하려는 경우 HTTP 및 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 [자세히 알아보기](#).

보안 그룹 할당: ☒ 새 보안 그룹 생성
☐ 기존 보안 그룹 선택

보안 그룹 이름: launch-wizard-4
 설명: launch-wizard-4 created 2022-03-18T14:46:16.123+09:00

유형	프로토콜	포트 범위	소스	설명
SSH	TCP	22	사용자 지정 0.0.0.0/0	예: SSH for Admin Desktop
HTTP	TCP	80	위치 무관 0.0.0.0/0, ::/0	예: SSH for Admin Desktop
HTTPS	TCP	443	위치 무관 0.0.0.0/0, ::/0	예: SSH for Admin Desktop

경고
 소스가 0.0.0.0/0인 규칙은 모든 IP 주소에서 인스턴스에 액세스하도록 허용합니다. 알려진 IP 주소의 액세스만 허용하도록 보안 그룹을 설정하는 것이 좋습니다.

[취소](#) [이전](#) [검토 및 시작](#)

12. [단계 7:인스턴스 시작 검토] 페이지에서, 지금까지 구성한 정보를 확인 한 다음, 수정 및 변경사항이 없다면 [시작하기] 버튼을 클릭한다.

단계 7: 인스턴스 시작 검토

인스턴스 시작 세부 정보를 검토하십시오. 이전으로 돌아가서 각 섹션에 대한 변경 내용을 편집할 수 있습니다. 키 페어를 인스턴스에 할당하고 시작 프로세스를 완료하려면 [시작]을 클릭합니다.

AMI 세부 정보

AMI: Ubuntu Server 20.04 LTS (HVM), SSD Volume Type - ami-0454bb2f9c7de534
 설명: Ubuntu Server 20.04 LTS (HVM), EBS General Purpose (SSD) Volume Type. Support available from Canonical (<http://www.ubuntu.com/cloud/services>).
 루트 디바이스 유형: ebs, 가상화 유형: hvm

인스턴스 유형

인스턴스 유형	ECU	vCPUs	메모리 (GiB)	인스턴스 스토리지 (GiB)	EBS 최적화 사용 가능	네트워크 성능
t2.micro	-	1	1	EBS 전용	-	Low to Moderate

보안 그룹

[취소](#) [이전](#) [시작하기](#)

13. [기존 키 페어 선택] 드롭다운 버튼을 클릭하면 보이는 3개의 항목 중에 “새 키 페어 생성”을 선택하고, [키 페어 이름]에 “lab5-ubuntu-ec2-key”를 입력 후 [키 페어 다운로드] 버튼을 클릭하여 “lab5-ubuntu-ec2-key.pem” 파일을 로컬 컴퓨터에 보관한다. 그리고 [인스턴스 시작] 버튼을 클릭하여 지금까지 설정한 인스턴스 환경정보를 이용하여 새 인스턴스를 생성한다.

기존 키 페어 선택 또는 새 키 페어 생성

키 페어는 AWS에 저장하는 퍼블릭 키와 사용자가 저장하는 프라이빗 키 파일로 구성됩니다. 이 둘을 모두 사용하여 SSH를 통해 인스턴스에 안전하게 접속할 수 있습니다. Windows AMI의 경우 인스턴스에 로그인하는 데 사용되는 암호를 얻으려면 프라이빗 키 파일이 필요합니다. Linux AMI의 경우, 프라이빗 키 파일을 사용하면 인스턴스에 안전하게 SSH로 연결할 수 있습니다. Amazon EC2는 ED25519 및 RSA 키 페어 유형을 지원합니다.

참고: 선택한 키 페어가 이 인스턴스에 대해 승인된 키 세트에 추가됩니다. 퍼블릭 AMI에서 기존 키 페어 제거에 대해 자세히 알아보십시오.

새 키 페어 생성

키 페어 유형

☒ RSA ☐ ED25519

키 페어 이름

lab5-ubuntu-ec2-key

키 페어 다운로드

계속하려면 먼저 프라이빗 키 파일(*.pem 파일)을 다운로드해야 합니다. 액세스할 수 있는 안전한 위치에 저장합니다. 파일은 생성되고 나면 다시 다운로드할 수 없습니다.

취소

인스턴스 시작

14. 인스턴스는 보통 5 ~ 10분 정도 시간이 걸린다. 페이지 우측 하단의 [인스턴스 보기]를 클릭한다.

시작 상태

✓

지금 인스턴스를 시작 중입니다.

다음 인스턴스 시작이 개시됨: I-054d4fff80598f301 시작 로그 보기

ⓘ

예상 요금 알림 받기

결제 알림 생성 AWS 결제 예상 요금이 사용자가 정의한 금액을 초과하는 경우(예를 들면 프리 티어를 초과하는 경우) 이메일 알림을 받습니다.

인스턴스에 연결하는 방법

인스턴스를 시작 중이며, 사용할 준비가 되어 실행 중 상태가 될 때까지 몇 분이 걸릴 수도 있습니다. 새 인스턴스에서는 사용 시간이 즉시 시작되어 인스턴스를 중지 또는 종료할 때까지 계속 누적입니다.

인스턴스 보기를 클릭하여 인스턴스의 상태를 모니터링합니다. 인스턴스가 실행 중 상태가 되고 나면 [인스턴스] 화면에서 인스턴스에 연결할 수 있습니다. 인스턴스에 연결하는 방법 알아보기.

다음은 시작에 도움이 되는 유용한 리소스입니다.

- Linux 인스턴스에 연결하는 방법
- AWS 프리 티어에 대해 알아보기

- Amazon EC2: 사용 설명서
- Amazon EC2: 토론 포럼

인스턴스가 시작되는 동안 다음을 수행할 수도 있습니다.

- 상태 검사 결과 생성 해당 인스턴스가 상태 검사를 통과하지 못하는 경우 알림을 받습니다. (추가 요금이 적용될 수 있음)
- 추가 EBS 볼륨 생성 및 연결 (추가 요금이 적용될 수 있음)
- 보안 그룹 관리

인스턴스 보기

15. Linux Server 인스턴스가 정상적으로 생성된 후, 해당 인스턴스 요약 페이지로 이동한다. [퍼블릭 IPv4 DNS]의 값을 복사한다.

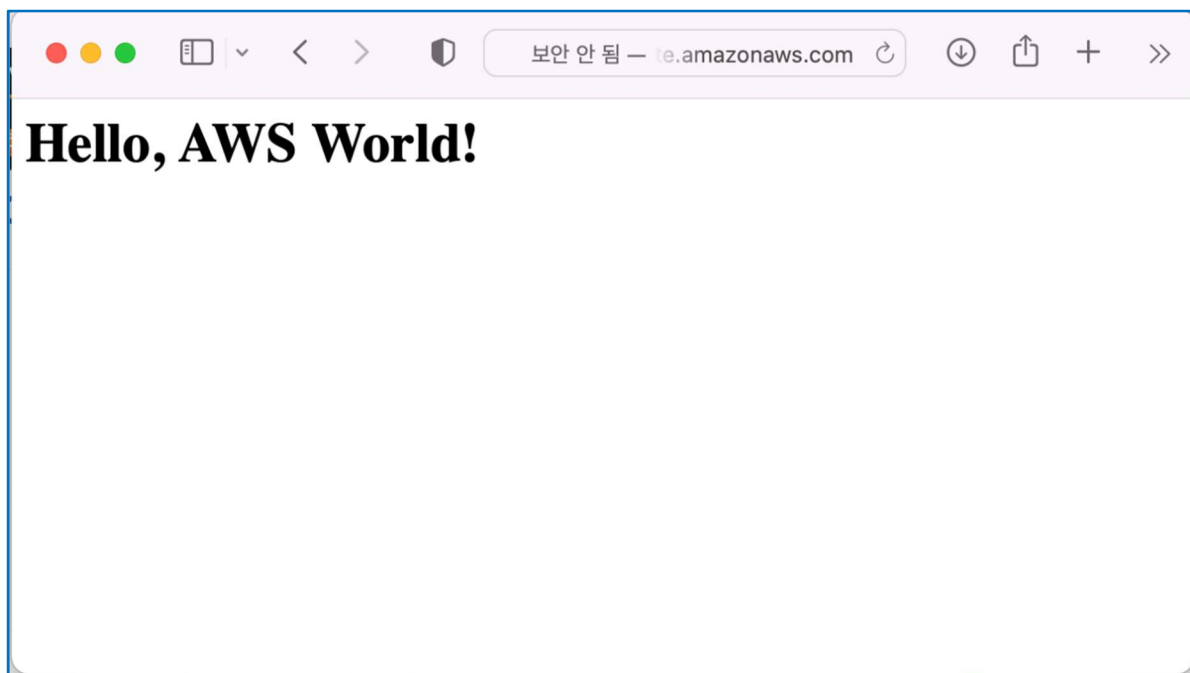
EC2 > 인스턴스 > i-054d4fff80598f301

i-054d4fff80598f301 (lab5-ubuntu-ec2)에 대한 인스턴스 요약 정보

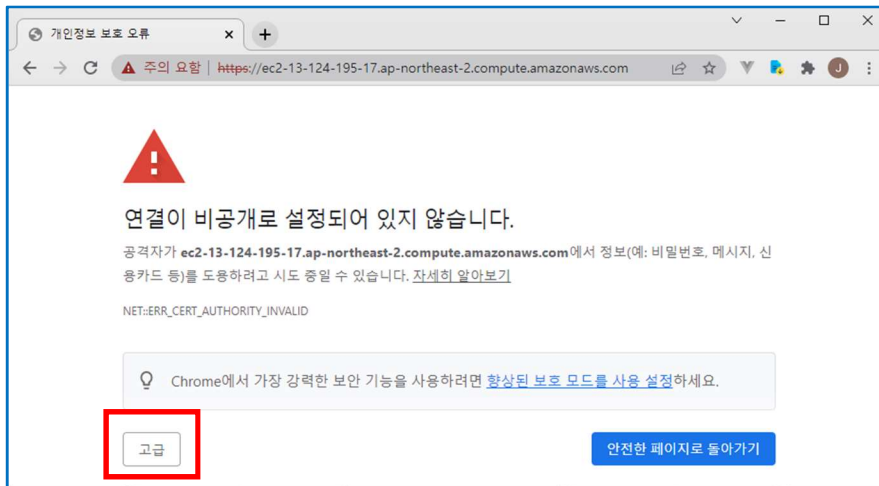
less than a minute 전에 업데이트됨

인스턴스 ID i-054d4fff80598f301 (lab5-ubuntu-ec2)	퍼블릭 IPv4 주소 13.124.195.17 개방 주소법	프라이빗 IPv4 주소 10.0.8.240
IPv6 주소 -	인스턴스 상태 실행 중	퍼블릭 IPv4 DNS ec2-13-124-195-17.ap-northeast-2.compute.amazonaws.com 개방 주소법
호스트 이름 유형 IP 이름: ip-10-0-8-240.ap-northeast-2.compute.internal	프라이빗 IP DNS 이름(IPv4만 해당) ip-10-0-8-240.ap-northeast-2.compute.internal	프라이빗 리소스 DNS 이름 응답 IPv4(A)
인스턴스 유형 t2.micro	탄력적 IP 주소 -	VPC ID vpc-00a61e8f2edc671ff (lab2-vpc)
AWS Compute Optimizer 찾기 권장 사항을 위해 AWS Compute Optimizer에 옵트인합 니다. 자세히 알아보기	IAM 역할 -	서브넷 ID subnet-099a38aacaeb51a0e (lab2-subnet-public1-ap-northeast-2a)

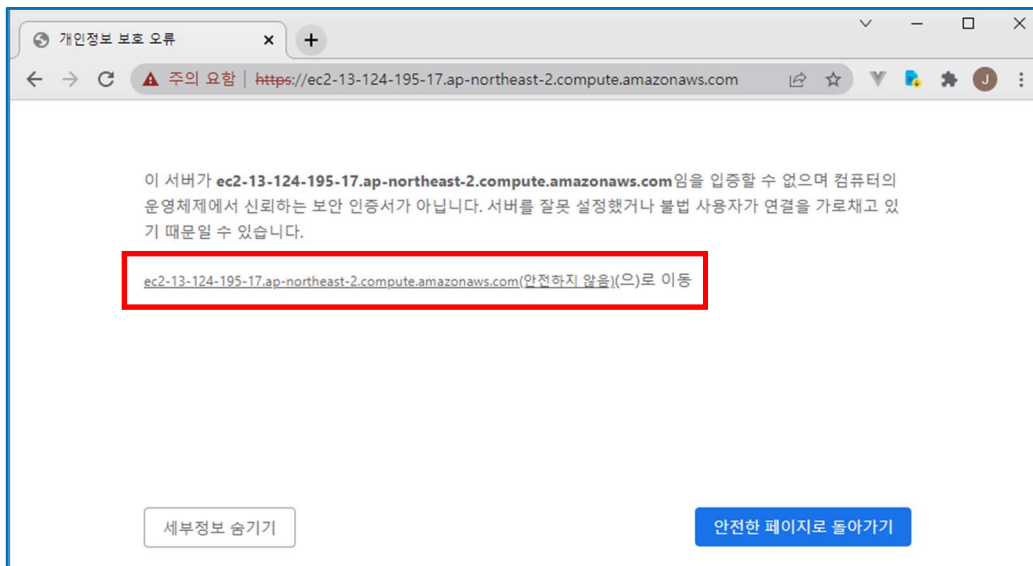
16. 복사한 퍼블릭 IPv4 DNS의 값을 주소표시줄에 붙여넣으면 Linux Server 인스턴스 설치시 입력했던 사용자 데이터대로 Apache Web Server 설치와 사용자 정의 웹페이지의 결과를 볼 수 있다.



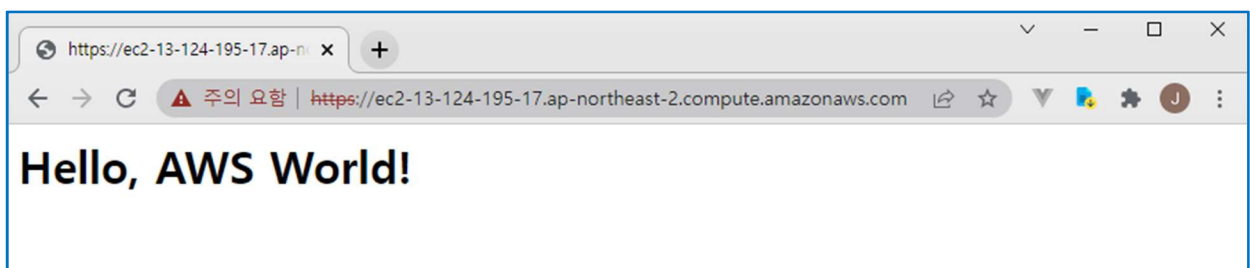
17. 이번에는 HTTP가 아닌 HTTPS 프로토콜로 이동해보자. [고급]을 클릭해보자.



18. 페이지 중간의 (안전하지 않음)의 링크를 클릭한다.



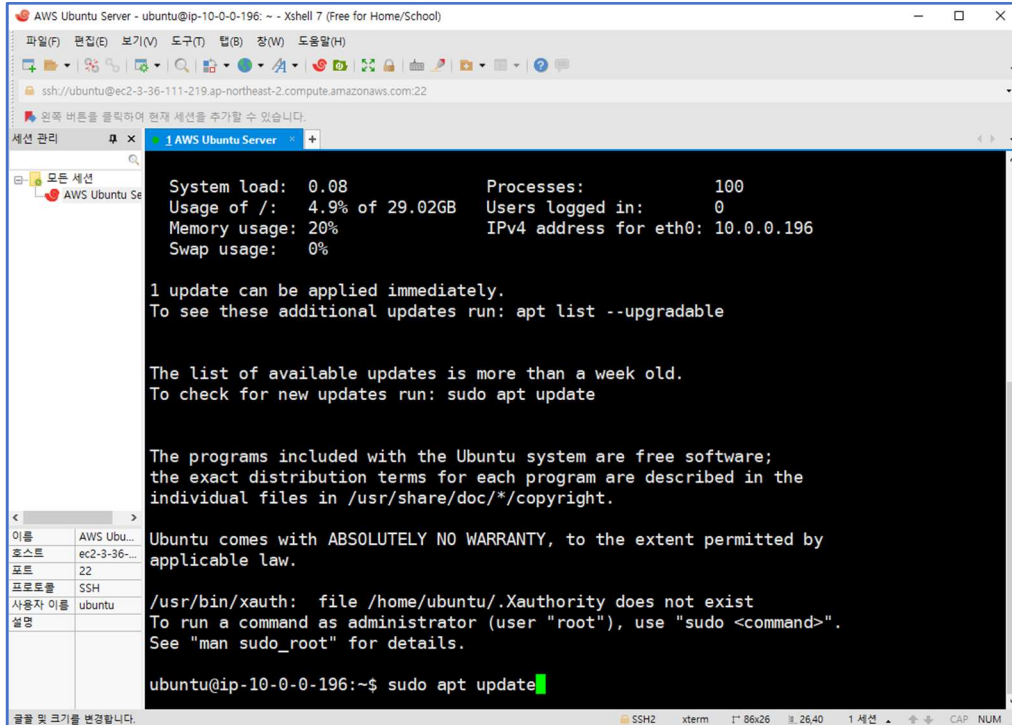
19. 이제 HTTPS로 접근할 수 있다.



웹 서버 설치 및 테스트하기

1. 이 실습에서는 일단 Ubuntu Server EC2를 설치하고 추가로 Apache Web Server를 수동으로 설치하고 **Apache Welcome** 페이지가 정상적으로 뜨는지 확인하기로 한다. 먼저, 다음의 명령을 사용해서 최신 apt list를 업데이트 한다.

\$ sudo apt update



```
AWS Ubuntu Server - ubuntu@ip-10-0-0-196: ~ - Xshell 7 (Free for Home/School)
ssh://ubuntu@ec2-3-36-111-219.ap-northeast-2.compute.amazonaws.com:22

System load:  0.08      Processes:    100
Usage of /:   4.9% of 29.02GB   Users logged in:  0
Memory usage: 20%      IPv4 address for eth0: 10.0.0.196
Swap usage:   0%

1 update can be applied immediately.
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

The programs included with the Ubuntu system are free software;
the exact distribution terms for each program are described in the
individual files in /usr/share/doc/*/copyright.

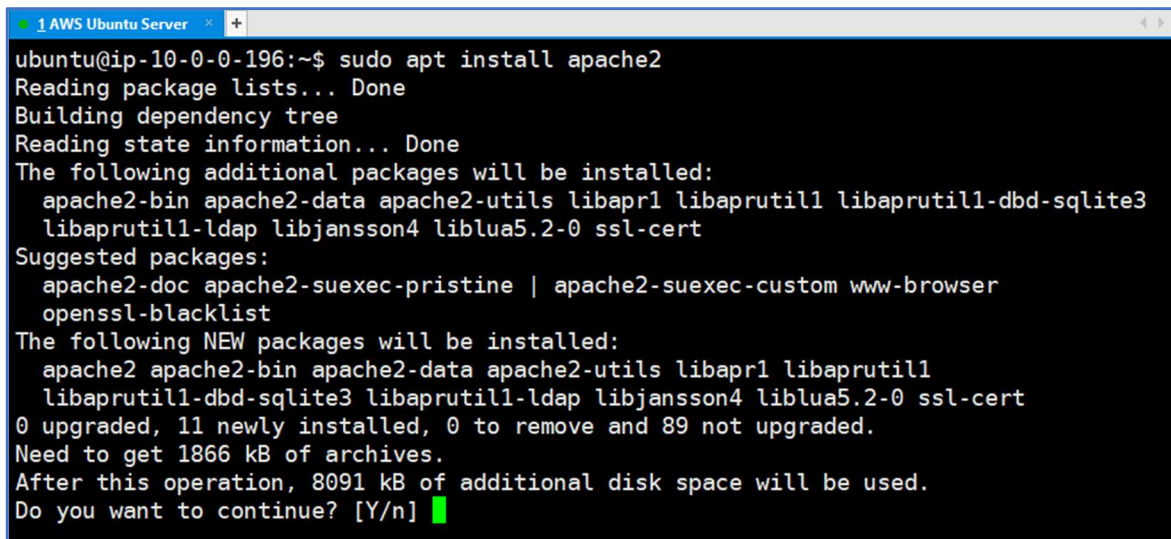
Ubuntu comes with ABSOLUTELY NO WARRANTY, to the extent permitted by
applicable law.

/usr/bin/xauth: file /home/ubuntu/.Xauthority does not exist
To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

ubuntu@ip-10-0-0-196:~$ sudo apt update
```

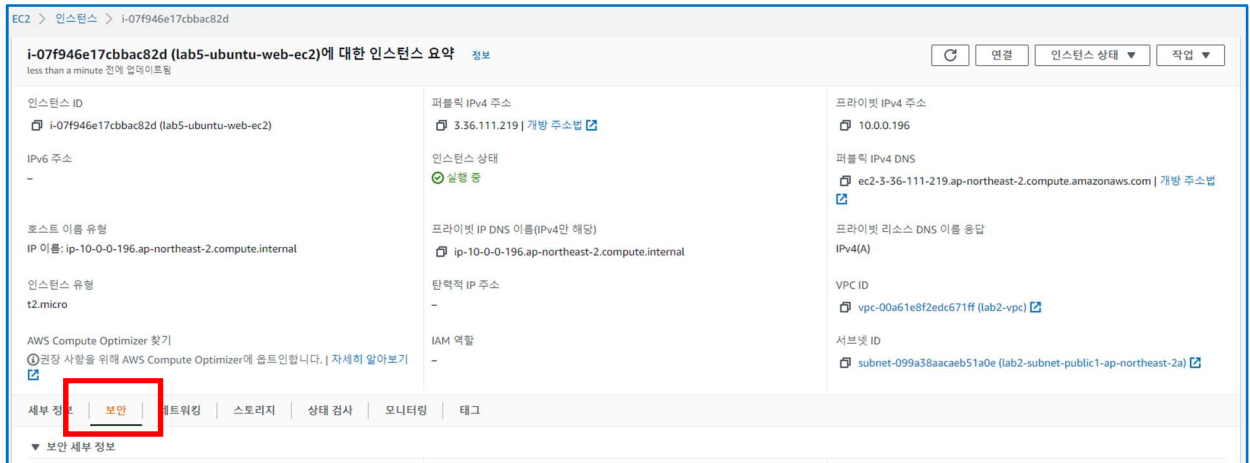
2. 명령 프롬프트에서 다음의 명령어를 사용해서 **Apache Web Server**를 설치한다. 계속 설치를 진행할 것인가 묻는 곳에서 'y'를 넣고 Enter key를 누르거나 기본값으로 'Y'에 맞춰져 있기 때문에 그냥 Enter key를 넣어도 된다.

\$ sudo apt install apache2

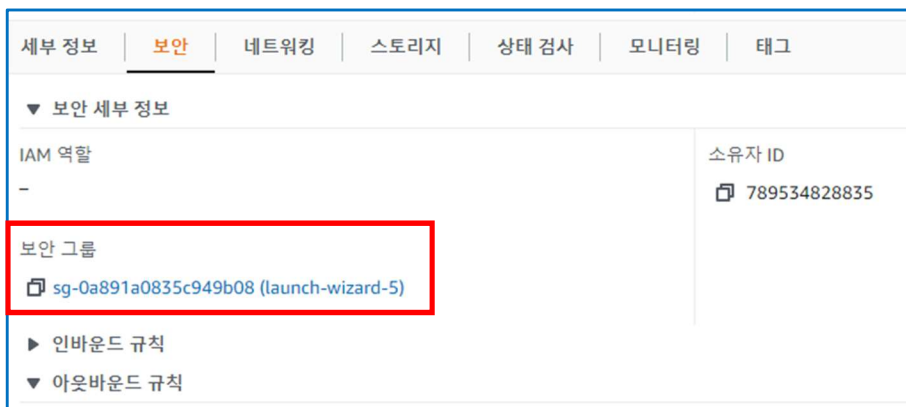


```
1 AWS Ubuntu Server
ubuntu@ip-10-0-0-196:~$ sudo apt install apache2
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following additional packages will be installed:
  apache2-bin apache2-data apache2-utils libapr1 libaprutil1 libaprutil1-dbd-sqlite3
  libaprutil1-ldap libjansson4 liblua5.2-0 ssl-cert
Suggested packages:
  apache2-doc apache2-suexec-pristine | apache2-suexec-custom www-browser
  openssl-blacklist
The following NEW packages will be installed:
  apache2 apache2-bin apache2-data apache2-utils libapr1 libaprutil1
  libaprutil1-dbd-sqlite3 libaprutil1-ldap libjansson4 liblua5.2-0 ssl-cert
0 upgraded, 11 newly installed, 0 to remove and 89 not upgraded.
Need to get 1866 kB of archives.
After this operation, 8091 kB of additional disk space will be used.
Do you want to continue? [Y/n]
```

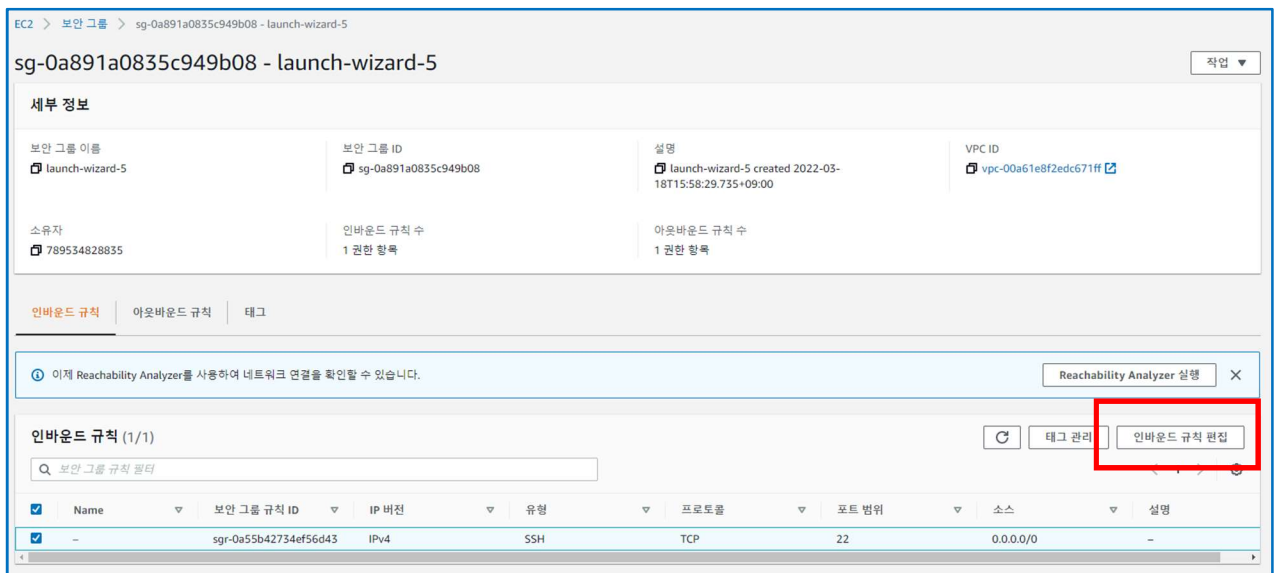

7. Linux Server의 Apache Web Server는 현재 잘 실행되고 있는 것을 알 수 있다. 하지만, 웹브라우저로 확인하려면 **보안그룹**을 수정해야 한다. 다시 AWS EC2 대시보드에 가서 해당 Linux Server 인스턴스를 클릭하여 **[보안]** 탭을 클릭한다.



8. 보안 그룹의 링크를 클릭한다.



9. 보안 그룹 요약창으로 들어왔다. 현재 **[인바운드 규칙]**은 오로지 **SSH**만 사용 가능하다. 추가적인 인바운드 규칙을 생성하기 위해 **[인바운드 규칙 편집]** 버튼을 클릭한다.



10. **[인바운드 규칙 편집]** 페이지에서 **[규칙 추가]** 버튼을 클릭하고 **[유형]**은 “HTTP”, **[소스]**는 “Anywhere-IPv4”을 선택하고 우측 하단에 있는 **[규칙 저장]** 오렌지 버튼을 클릭한다.

EC2 > 보안 그룹 > sg-0a891a0835c949b08 - launch-wizard-5 > 인바운드 규칙 편집

인바운드 규칙 편집 정보

인바운드 규칙은 인스턴스에 도달하도록 허용된 수신 트래픽을 제어합니다.

인바운드 규칙 정보

보안 그룹 규칙 ID	유형 <small>정보</small>	프로토콜 <small>정보</small>	포트 범위 <small>정보</small>	사용자 지정	설명 - 선택 사항 <small>정보</small>	
sgf-0a55b42734ef56d43	SSH ▼	TCP	22	<div>사용자 지정</div> <div>Anywhere-IPv4</div> <div>Anywhere-IPv6</div> <div>0.0.0.0/0 ✕</div>		삭제
-	HTTP ▼	TCP	80	<div>내 IP</div> <div>사용자 지정 ... ▲</div>		삭제

규칙 추가

취소

변경 사항 미리 보기

규칙 저장

11. 다시 보안그룹창으로 돌아왔다. 아래쪽 **[인바운드 규칙]**을 보면 방금 추가한 **HTTP 유형**을 확인할 수 있다.

인바운드 규칙

아웃바운드 규칙

태그

이제 Reachability Analyzer를 사용하여 네트워크 연결을 확인할 수 있습니다.

Reachability Analyzer 실행

인바운드 규칙 (2)

🔄

태그 관리

인바운드 규칙 편집

🔍 보안 그룹 규칙 필터

<

1

>

⚙️

<input type="checkbox"/>	Name ▾	보안 그룹 규칙 ID ▾	IP 버전 ▾	유형 ▾	프로토콜 ▾	포트 범위 ▾	소스 ▾
<input type="checkbox"/>	-	sgr-097f7aa9c9fa9d033	IPv4	HTTP	TCP	80	0.0.0.0/0
<input type="checkbox"/>	-	sgr-0a55b42734ef56d43	IPv4	SSH	TCP	22	0.0.0.0/0

12. 인스턴스 요약 창에서 현재 Linux Server 인스턴스의 **[퍼블릭 IPv4]**를 확인할 수 있는데, 이 주소를 복사한다.

EC2 > 인스턴스 > i-07f946e17cbbac82d

i-07f946e17cbbac82d (lab5-ubuntu-web-ec2)에 대한 인스턴스 요약 정보

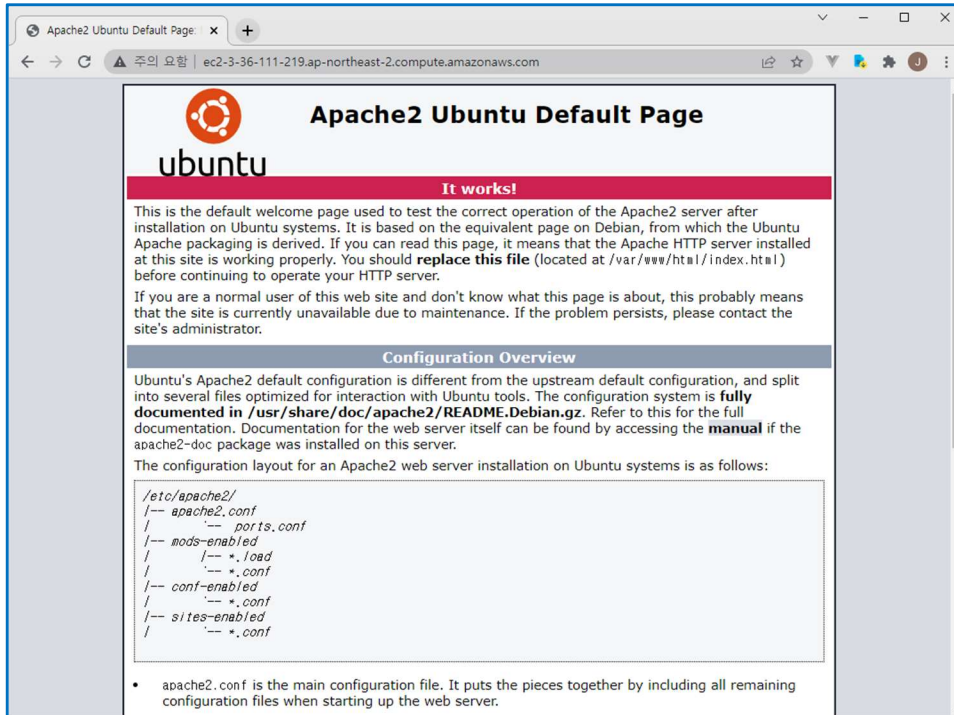
less than a minute 전에 업데이트됨

🔄
연결
인스턴스 상태 ▼
작업 ▼

인스턴스 ID i-07f946e17cbbac82d (lab5-ubuntu-web-ec2) IPv6 주소 - 호스트 이름 유형 IP 이름: ip-10-0-0-196.ap-northeast-2.compute.internal	퍼블릭 IPv4 주소 3.36.111.219 개방 주소법 인스턴스 상태 🟢 실행 중 프라이빗 IP DNS 이름(IPv4만 해당) ip-10-0-0-196.ap-northeast-2.compute.internal 탄력적 IP 주소 - IAM 역할 -	프라이빗 IPv4 주소 10.0.0.196 <div> 🟢 퍼블릭 IPv4 DNS 복사됨 </div> ec2-36-111-219-ap-northeast-2.compute.amazonaws.com 개방 주소법 프라이빗 리소스 DNS 이름 응답 IPv4(A) VPC ID vpc-00a61e8f2edc671ff (lab2-vpc) 서브넷 ID subnet-099a38aaceb51a0e (lab2-subnet-public1-ap-northeast-2a)
--	--	---

AWS Compute Optimizer 찾기
 ① 권장 사항을 위해 AWS Compute Optimizer에 옵트인합니다. | [자세히 알아보기](#)

13. 방금 복사한 퍼블릭 IPv4 주소를 웹 브라우저를 열고 주소창에 복사한 주소를 붙여 넣는다. 그러면 아래 그림과 같이 **Apache Web Server**의 **Welcome** 화면을 확인하게 될 것이다.



14. 만일 위의 그림을 편집해서 여러분이 원하는 홈페이지를 만들려면 다음과 같은 작업을 수행한다. 다시 **SSH Tool**로 Linux Server 인스턴스를 연결한 다음, 명령 프롬프트에서 다음의 명령을 수행한다.

```
$ sudo nano /var/www/html/index.html
```

15. Nano 라는 프로그램을 사용하여 편집한다. 다음과 같이 간단한 HTML 코드를 입력한다.

```
GNU nano 4.8 /var/www/html/index.html Modified
<html>
  <head>
    <title>Hello, There!!!</title>
  </head>
  <body>
    <div style='color:blue;font-weight:900;font-size:3em' align='center'>
      Welcome My Home Page
    </div>
  </body>
</html>
```

^G Get Help ^O Write Out ^W Where Is ^K Cut Text ^J Justify ^C Cur Pos
^X Exit ^R Read File ^_ Replace ^U Paste Text ^T To Spell ^_ Go To Line

16. **index.html**을 저장하기 위해 **Ctrl + o**를 누른다. 그러면 **"File Name to Write : /var/www/html/index.html"** 이렇게 메시지가 나오면 Enter Key를 눌러서 저장한다.
17. Nano 프로그램을 빠져나오기 위해 키보드의 **Ctrl + x**를 누른다.
18. 웹 브라우저를 열고 Linux Server의 퍼블릭 **IPv4** 주소를 다시 입력한다. 또는 이미 웹 브라우저가 열려 있는 상태이면 Refresh 버튼을 눌러서 페이지를 다시 로딩한다.

