

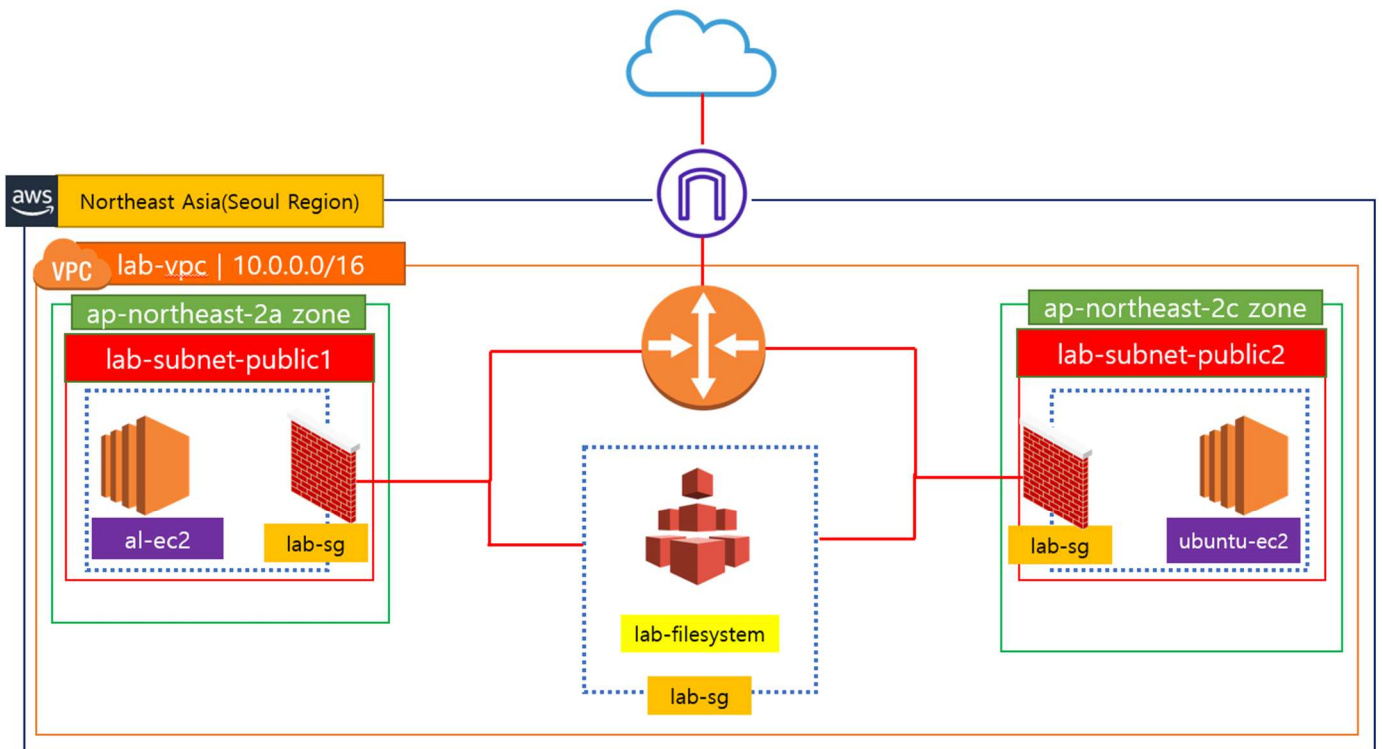
Lab. Create Amazon EFS

목적

이번 실습에서는 Amazon EFS에 대해 다룬다. 파일 시스템을 설정하고 이 파일 시스템에 각각 다른 AZ에 있는 EC2 인스턴스가 NFS를 통해 같은 파일 시스템에 접근하고 파일을 생성하는 일련의 작업을 수행한다.

사전 준비물

AWS Free-Tier 계정



1. 다음과 같이 **lab-vpc**를 생성한다.

- A. [생성할 리소스] : VPC 등
- B. [이름 태그 자동 생성] > [자동 생성] : lab
- C. [IPv4 CIDR] : 10.0.0.0/16
- D. [테넌시] : 기본값
- E. [가용 영역(AZ)] : 2
- F. [첫 번째 가용 영역] : ap-northeast-2a
- G. [두 번째 가용 영역] : ap-northeast-2c
- H. [퍼블릭 서브넷 수] : 2
- I. [프라이빗 서브넷 수] : 2

VPC 설정

생성할 리소스 정보

VPC 리소스 또는 VPC 및 기타 네트워킹 리소스만 생성합니다.

☐ VPC만

☒ VPC 등

이름 태그 자동 생성 정보

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

☒ 자동 생성

lab

IPv4 CIDR 블록 정보

CIDR 표기법을 사용하여 VPC의 시작 IP와 크기를 결정합니다.

10.0.0.0/1665,536 IPs

CIDR 블록 크기는 /16에서 /28 사이여야 합니다.

IPv6 CIDR 블록 정보

☒ IPv6 CIDR 블록 없음

☐ Amazon 제공 IPv6 CIDR 블록

테넌시 정보

기본값

가용 영역(AZ) 수 정보

서브넷을 프로비저닝할 AZ 수를 선택합니다.고가용성을 위해서는 최소 2개 이상의 AZ를 사용하는 것이 좋습니다.

1

2

3

▼ 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

프라이빗 서브넷 수 정보

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

0

2

4

J. [서브넷 CIDR 블록 사용자 지정]

- i. [ap-northeast-2a 퍼블릭 서브넷 CIDR 블록] : 10.0.10.0/24
- ii. [ap-northeast-2c 퍼블릭 서브넷 CIDR 블록] : 10.0.20.0/24
- iii. [ap-northeast-2a 프라이빗 서브넷 CIDR 블록] : 10.0.30.0/24
- iv. [ap-northeast-2c 프라이빗 서브넷 CIDR 블록] : 10.0.40.0/24

K. [NAT 게이트웨이] : AZ당 1개

L. [VPC 엔드포인트] : 없음

M. [DNS 옵션] : 모두 체크

▼ 서버넷 CIDR 블록 사용자 지정

ap-northeast-2a 퍼블릭 서버넷 CIDR 블록

10.0.10.0/24256 IPs

ap-northeast-2c 퍼블릭 서버넷 CIDR 블록

10.0.20.0/24256 IPs

ap-northeast-2a 프라이빗 서버넷 CIDR 블록

10.0.30.0/24256 IPs

ap-northeast-2c 프라이빗 서버넷 CIDR 블록

10.0.40.0/24256 IPs

NAT 게이트웨이(\$) 정보

NAT 게이트웨이를 생성할 가용 영역(AZ) 수를 선택합니다. 각 NAT 게이트웨이마다 요금이 부과됩니다.

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
----	----------

DNS 옵션 정보

☒ DNS 호스트 이름 활성화

☒ DNS 확인 활성화

미리 보기



2. 이제 **EFS**를 사용할 **EC2** 인스턴스를 생성한다. 인스턴스는 2개 생성하는데, 각각의 설정정보는 다음과 같다.

A. lab-amazon-ec2

- ① **[AMI]** : Amazon Linux 2 AMI(HVM)- Kernel 5.10, SSD Volume Type, 64비트(x86)
- ② **[인스턴스 유형]** : t2.micro
- ③ **[키 페어] > [새 키 페어 생성]** : lab-amazon-ec2-key.pem
- ④ **[네트워크] > [편집]** : lab-vpc, lab-subnet-public1-ap-northeast-2a, 퍼블릭 IP 자동 할당 활성화
- ⑤ **[보안 그룹] > [새 보안 그룹 생성]** : lab-sg, SSH, 22, 위치 무관, 0.0.0.0/0
- ⑥ **[스토리지]** : 30 GiB, gp2

▼ 요약

인스턴스 개수 정보

1

소프트웨어 이미지(AMI)

Amazon Linux 2 Kernel 5.10 AMI...[더 보기](#)
ami-03a633fd8200146fb

가상 서버 유형(인스턴스 유형)

t2.micro

방화벽(보안 그룹)

새 보안 그룹

스토리지(볼륨)

1개의 볼륨 - 30GiB

취소

인스턴스 시작

B. lab-ubuntu-ec2

- ① **[AMI]** : Ubuntu Server 20.04 LTS(HVM), SSD Volume Type, 64비트(x86)
- ② **[인스턴스 유형]** : t2.micro
- ③ **[키 페어]** > **[새 키 페어 생성]** : lab-ubuntu-ec2-key.pem
- ④ **[네트워크]** > **[편집]** : lab-vpc, lab-subnet-public2-ap-northeast-2c, 퍼블릭 IP 자동 할당 활성화
- ⑤ **[보안 그룹]** > **[기존 보안 그룹 선택]** : lab-sg, SSH, 22, 위치 무관, 0.0.0.0/0
- ⑥ **[스토리지]** : 30GiB, gp2

▼ 요약

인스턴스 개수 정보

1

소프트웨어 이미지(AMI)

Canonical, Ubuntu, 22.04 LTS, ...[더 보기](#)
ami-086cae3329a3f7d75

가상 서버 유형(인스턴스 유형)

t2.micro

방화벽(보안 그룹)

lab-sg

스토리지(볼륨)

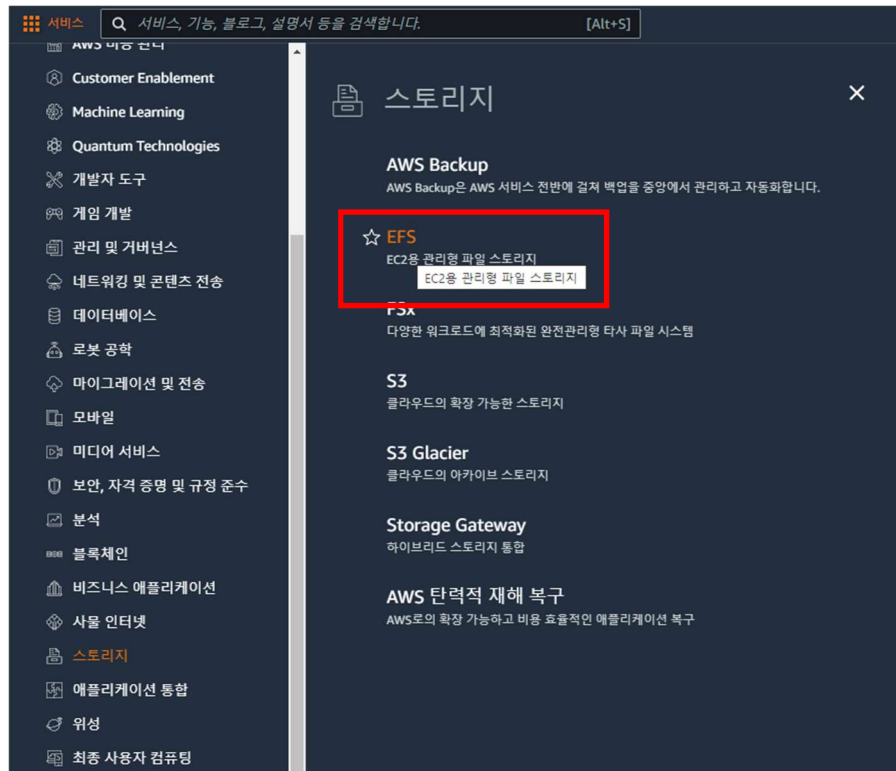
1개의 볼륨 - 30GiB

취소

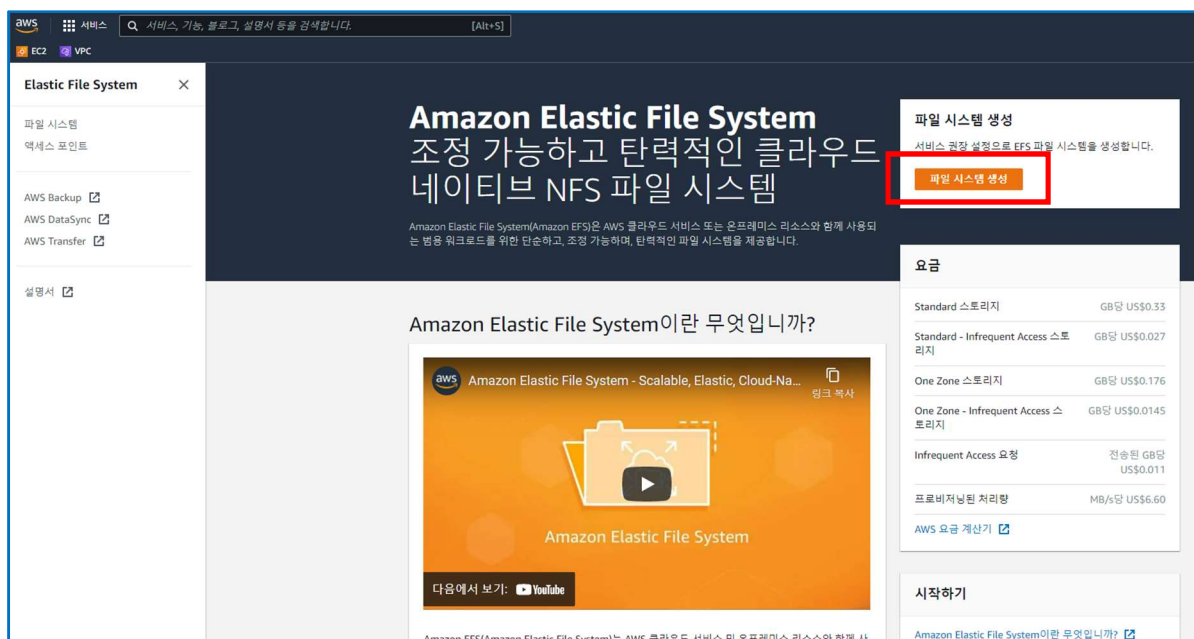
인스턴스 시작

인스턴스 (2) 정보								🔄	연결
🔍 Instance를 속성 또는 (case-sensitive) 태그로 찾기									
<input type="checkbox"/>	Name ↗	인스턴스 ID	인스턴스 상태	인스턴스 유형	상태 검사	경보 상태	가용 영역		
<input type="checkbox"/>	lab-amazon-ec2	i-07f5dd455557b74f6	🟢 실행 중 🔍	t2.micro	🟢 2/2개 검사 통과	경보 없음 +	ap-northeast-2a		
<input type="checkbox"/>	lab-ubuntu-ec2	i-01c91e033aa9d5b73	🟢 실행 중 🔍	t2.micro	🟢 2/2개 검사 통과	경보 없음 +	ap-northeast-2c		

3. [서비스] > [스토리지] > [EFS]를 선택한다.



4. Amazon EFS 페이지이다. 우측 상단쪽 [파일 시스템 생성] 버튼을 클릭한다.



5. [파일 시스템 생성]페이지이다. 다음 각각의 값을 설정하고 보다 자세한 설정을 위해 [사용자 지정]을 클릭한다.

A. [이름] : lab-filesystem

B. [Virtual Private Cloud(VPC)] : lab-vpc

파일 시스템 생성

권장 설정으로 EFS 파일 시스템을 생성합니다. [자세히 알아보기](#)

이름 - 선택 사항

파일 시스템 이름을 지정합니다.

lab-filessystem

이름에는 문자, 숫자 및 + - = _ : / 기호로 최대 256자까지 포함될 수 있습니다.

Virtual Private Cloud(VPC)

EC2 인스턴스를 파일 시스템에 연결하고자 하는 VPC를 선택합니다.

vpc-0220ac038ac6d4990

lab-vpc

취소

사용자 지정

생성

6. [1단계:파일 시스템 설정] 페이지이다. 다음의 각 값을 설정하고 [다음]을 클릭한다.

- A. [이름] : lab-filessystem
- B. [스토리지 클래스] : Standard
- C. [자동 백업] : 자동 백업 활성화
- D. [수명 주기 관리] > [IA로 전환] : 마지막 액세스 이후 30일 경과
- E. [암호화] : 유휴 시 데이터 암호화 활성화

파일 시스템 설정

일반

이름 - 선택 사항

파일 시스템 이름을 지정합니다.

lab-filessystem

스토리지 클래스 | [자세히 알아보기](#)

Standard

여러 AZ에 걸쳐 데이터를 중복 저장

One Zone

단일 AZ 내에 데이터를 중복 저장

자동 백업

권장 설정을 사용하여 AWS Backup으로 파일 시스템 데이터를 자동으로 백업합니다. 추가 요금이 적용됩니다. [자세히 알아보기](#)

자동 백업 활성화

수명 주기 관리

파일을 Standard - Infrequent Access(IA) 스토리지 클래스로 이동하여 액세스 패턴의 변화에 따라 자동으로 비용을 절감합니다. [자세히 알아보기](#)

IA로 전환

파일을 Standard에서 Standard-Infrequent Access로 전환합니다.

마지막 액세스 이후 30일 경과

IA 외부로 전환

파일을 Standard-Infrequent Access에서 Standard로 전환합니다.

없음

암호화

파일 시스템의 유휴 데이터의 암호화를 활성화하려면 선택합니다. 기본적으로 AWS KMS 서비스 키(aws/elasticfilesystem)를 사용합니다. [자세히 알아보기](#)

유휴 시 데이터 암호화 활성화

▶ 암호화 설정 사용자 지정

F. [처리량 모드] : 버스트

G. [추가 설정] > [성능 모드] : 범용

H. [태그] : Name / lab-filesystem

성능 설정

처리량 모드

파일 시스템의 처리량 제한에 대한 방법을 선택합니다. [자세히 알아보기](#)

☐ 개선됨

다양한 성능 요구 사항이 있는 워크로드에 더 많은 유연성과 더 높은 처리량 수준을 제공합니다.

☒ 버스트

기본적인 성능 요구 사항이 있는 워크로드를 위해 스토리지 양에 따라 확장되는 처리량을 제공합니다.

▼ 추가 설정

성능 모드

필요한 IOPS를 기반으로 파일 시스템의 성능 모드를 설정합니다. [자세히 알아보기](#)

☒ 범용 (권장)

고성능 및 지연 시간에 민감한 애플리케이션을 비롯하여 다양한 워크로드에 적합

☐ 최대 I/O

긴 지연 시간을 허용할 수 있는 고도로 병렬화된 워크로드를 위해 설계

▼ 태그 선택 사항

태그를 추가하여 키-값 페어를 리소스에 연결합니다. [자세히 알아보기](#)

태그 키

태그 값 - 선택 사항

태그 추가

태그를 49개 더 추가할 수 있음

7. [2단계:네트워크 액세스] 페이지이다. 다음의 각 값을 설정하고 [다음]을 클릭한다.

A. Virtual Private Cloud(VPC) : lab-vpc

B. [탑재 대상]

① [가용 영역] : ap-northeast-2a, [서브넷 ID] : lab-subnet-public1-ap-northeast-2a, [보안 그룹] : lab-sg

② [가용 영역] : ap-northeast-2c, [서브넷 ID] : lab-subnet-public2-ap-northeast-2c, [보안 그룹] : lab-sg

네트워크 액세스

네트워크

Virtual Private Cloud(VPC) 자세히 알아보기

EC2 인스턴스를 파일 시스템에 연결하고자 하는 VPC를 선택합니다.

vpc-0220ac038ac6d4990
lab-vpc

탑재 대상

탑재 대상은 Amazon EFS 파일 시스템을 탑재할 수 있는 NFSv4 엔드포인트를 제공합니다. 가용 영역마다 탑재 대상을 한 개씩 생성하는 것이 좋습니다. 자세히 알아보기

가용 영역

ap-northeast-2a

서브넷 ID

subnet-051b50603a207b483

IP 주소

자동

보안 그룹

보안 그룹 선택

제거

sg-0077c4b4d18797565
lab-sg

ap-northeast-2c

subnet-0fe04e164c474e3db

자동

보안 그룹 선택

제거

sg-0077c4b4d18797565
lab-sg

8. [3단계:파일 시스템 정책] 페이지이다. 기본값 그대로 놓고 [다음]을 클릭한다.

파일 시스템 정책 - 선택 사항

정책 옵션

이러한 공통 정책 옵션 중 하나 이상을 선택하거나 편집기를 사용하여 사용자 지정 정책을 하나 더 생성합니다. 자세히 알아보기

- ☐ 기본적으로 루트 액세스 금지*
- ☐ 기본적으로 읽기 전용 액세스 적용*
- ☐ 익명 액세스 금지
- ☐ 모든 클라이언트에 대해 전송 중 암호화 적용

* 자격 증명 기반 정책은 이러한 기본 권한을 대체할 수 있습니다.

▶ 추가 권한 부여

정책 편집기 (JSON)

지우기

수동으로 변경하면 편집기를 비울 때까지 왼쪽의 정책 옵션을 사용할 수 없습니다.

취소

이전

다음

9. [4단계:검토 및 생성] 페이지이다. 실습을 위해 기본값 그대로 놓고 페이지를 스크롤 다운하여 [생성] 버튼을 클릭한다.

10. 잠시 기다리면 다음 그림과 같이 EFS가 성공적으로 생성된 것을 확인할 수 있다.

성공! 파일 시스템(fs-0e8c98903277b82a7)이 사용 가능합니다.

Amazon EFS > 파일 시스템

파일 시스템 (1)

Q 속성 값으로 필터링

	이름	파일 시스템 ID	암호화됨	총 크기	Standard/On e Zone의 크기	Standard-IA/One Zone-IA의 크기	프로비저닝된 처리량(MiB/s)	파일 시스템 상태	생성 시간	가용 영역
	lab-filesystem	fs-0e8c98903277b82a7	암호화됨	6.00 KiB	6.00 KiB	0바이트	-	사용 가능	Sun, 29 Oct 2023 12:43:38 GMT	Standard

11. 이제 EFS에 접근할 EC2 인스턴스 페이지로 이동한다. 먼저 **lab-amazon-ec2**를 연결한다.

```
[ec2-user@ip-10-0-10-116 ~]$
```

12. 두번째 인스턴스 lab-ubuntu-ec2도 연결한다.

```
Memory usage: 21%           IPv4 address for eth0: 10.0.20.105
Swap usage: 0%

Expanded Security Maintenance for Applications is not enabled.

0 updates can be applied immediately.

Enable ESM Apps to receive additional future security updates.
See https://ubuntu.com/esm or run: sudo pro status

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.

To run a command as administrator (user "root"), use "sudo <command>".
See "man sudo_root" for details.

ubuntu@ip-10-0-20-105:~$
```

13. **EFS** 시스템을 사용하기 위해 **보안그룹의 인바운드 규칙**을 추가해야 한다. 아래와 같이 **[규칙 추가]**를 클릭하여 설정하고 **[규칙 저장]**을 클릭하여 새 **인바운드 규칙**을 추가한다.

- A. [유형] : NFS
B. [프로토콜] : TCP
C. [포트 범위] : 2049

D. [소스] : Anywhere-IPv4

인바운드 규칙 편집 정보

인바운드 규칙은 인스턴스에 도달하도록 허용된 수신 트래픽을 제어합니다.

인바운드 규칙 정보

보안 그룹 규칙 ID	유형 정보	프로토콜 정보	포트 범위 정보	소스 정보	설명 - 선택 사항 정보
sg-0fda632f8134f233c	SSH	TCP	22	사용자 지정	
				0.0.0.0/0	
-	NFS	TCP	2049	Anywhere...	
				0.0.0.0/0	

규칙 추가

14. 위에서 생성한 **EFS** 페이지로 이동한다. 우측 상단의 **[연결]**을 클릭한다.

lab-filesystem (fs-0e8c98903277b82a7) 삭제 **연결**

일반 편집

성능 모드 범용	자동 백업 ✔ 활성화됨
처리량 모드 버스트	암호화됨 d79c25d2-7774-4736-9657-b3a16fa15cad (aws/elasticfilesystem)
수명 주기 관리 IA로 전환: 마지막 액세스 이후 30일 경과 IA 외부로 전환: 없음	파일 시스템 상태 ✔ 사용 가능
가용 영역 Standard	DNS 이름 fs-0e8c98903277b82a7.efs.ap-northeast-2.amazonaws.com

15. **[연결]** 페이지이다. **[자세히 알아보기]** 링크를 클릭해보자.

연결

Linux 인스턴스에 Amazon EFS 파일 시스템을 탑재합니다. **자세히 알아보기**

☒ DNS를 통한 탑재 ☐ IP를 통해 탑재

EFS 탑재 셸퍼 사용:

```
sudo mount -t efs -o tls fs-0e8c98903277b82a7:/ efs
```

NFS 클라이언트 사용:

```
sudo mount -t nfs4 -o nfsvers=4.1,rsize=1048576,wsz=1048576,hard,timeo=600,retr=2,noresvport fs-0e8c98903277b82a7.efs.ap-northeast-2.amazonaws.com:/ efs
```

자세한 내용은 사용 설명서를 참조하십시오. **자세히 알아보기**

닫기

16. [Mounting EFS file systems] 페이지(<https://docs.aws.amazon.com/efs/latest/ug/mounting-fs.html>)로 이동한다. EFS 탑재 헬퍼를 설치하기 위해 [Amazon EFS 클라이언트 수동 설치(Manually installing the Amazon EFS client)] 링크를 클릭한다.

Mounting EFS file systems

PDF | RSS

In the following sections you can learn how to mount your Amazon EFS file system using the Amazon EFS mount helper. In addition, learn how to automatically remount your file system after any system restarts using the file `fstab` file. Using the EFS mount helper, you have the following options for mounting your Amazon EFS file system:

- Mounting on supported EC2 instances
- Mounting with IAM authorization
- Mounting with Amazon EFS access points
- Mounting with an on-premise Linux client
- Auto-mounting EFS file systems when an EC2 instance reboots
- Mounting a file system when creating a new EC2 instance

Note

Amazon EFS does not support mounting from Amazon EC2 Windows instances.

The EFS mount helper is part of the `amazon-efs-utils` package. The `amazon-efs-utils` package is an open-source collection of Amazon EFS tools. For more information, see [Manually installing the Amazon EFS client](#).

Before the Amazon EFS mount helper was available, we recommended mounting your Amazon EFS file systems using the standard Linux NFS client. For more information, see [Mounting file systems without the EFS mount helper](#).

17. [Amazon EFS 클라이언트 수동 설치(Manually installing the Amazon EFS client)] 페이지(<https://docs.aws.amazon.com/efs/latest/ug/installing-amazon-efs-utils.html>)로 이동했다. 만일 EC2 인스턴스의 OS가 Amazon Linux이면 [Amazon Linux 및 Amazon Linux 2에 Amazon EFS 클라이언트 설치] 섹션에서 다음의 명령을 복사한다.

Installing the Amazon EFS client on Amazon Linux and Amazon Linux 2

The `amazon-efs-utils` package for installing on the Amazon Linux and Amazon Linux 2 is available in the following locations:

- The Amazon Linux and Amazon Linux 2 Amazon machine image (AMI) package repositories.
- The AWS [efs-utils](#) GitHub repository.

The following procedure describes how to install `amazon-efs-utils` from the Amazon Linux and Amazon Linux 2 AMI package repositories.

You can also install or update `amazon-efs-utils` from the AWS [efs-utils](#) GitHub repository. For instructions describing how to install and update the Amazon EFS client using GitHub, see [To build and install amazon-efs-utils as an RPM package for Amazon Linux, Amazon Linux 2](#).

To install the Amazon EFS client on other Linux distributions, see [Installing the Amazon EFS client on other Linux distributions](#).

Note

If you're using AWS Direct Connect, you can find installation instructions in [Walkthrough: Create and mount a file system on-premises with AWS Direct Connect and VPN](#).

To install the `amazon-efs-utils` package on Amazon Linux 2 and Amazon Linux

1. Make sure that you've created an Amazon Linux or Amazon Linux 2 EC2 instance. For information on how to do this, see [Step 1: Launch an Instance](#) in the *Amazon EC2 User Guide for Linux Instances*.
2. Access the terminal for your instance through Secure Shell (SSH), and log in with the appropriate user name. For more information on how to do this, see [Connecting to your Linux instance using SSH](#) in the *Amazon EC2 User Guide for Linux Instances*.
3. Run the following command to install the `amazon-efs-utils` package.

```
sudo yum install -y amazon-efs-utils
```



```
$ sudo yum install -y amazon-efs-utils
```

18. SSH 클라이언트 툴에서 **al-ec2** 창에서 위에서 복사한 명령어를 붙여넣기 해서 실행한다.

```
Downloading packages:
(1/2): amazon-efs-utils-1.35.0-1.amzn2.noarch.rpm | 57 kB 00:00:00
(2/2): stunnel5-5.58-1.amzn2.0.1.x86_64.rpm | 165 kB 00:00:00
-----
Total | 1.6 MB/s | 221 kB 00:00:00
Running transaction check
Running transaction test
Transaction test succeeded
Running transaction
  Installing : stunnel5-5.58-1.amzn2.0.1.x86_64 1/2
  Installing : amazon-efs-utils-1.35.0-1.amzn2.noarch 2/2
  Verifying : amazon-efs-utils-1.35.0-1.amzn2.noarch 1/2
  Verifying : stunnel5-5.58-1.amzn2.0.1.x86_64 2/2

Installed:
  amazon-efs-utils.noarch 0:1.35.0-1.amzn2

Dependency Installed:
  stunnel5.x86_64 0:5.58-1.amzn2.0.1

Complete!
[ec2-user@ip-10-10-10-116 ~]$
```

19. 다시 **[Amazon EFS 클라이언트 수동 설치(Manually installing the Amazon EFS client)]** 페이지 (<https://docs.aws.amazon.com/efs/latest/ug/installing-amazon-efs-utils.html>)로 가서, 만일 EC2 인스턴스의 OS 가 Linux의 다른 배포판이라면, 특별히 이번 실습에서는 **Ubuntu** 배포판을 **lab-ubuntu-ec2**에 설치했기 때문에 페이지를 스크롤 다운하여 **[다른 Linux 배포판에서 Amazon EFS 클라이언트 설치(Installing the Amazon EFS client on other Linux distributions)]** 섹션으로 이동하여 목록 중에서 **[빌드 및 설치 amazon-efs-utils 우분투와 데비안을 위한 데비안 패지지로(To build and install amazon-efs-utils as a Debian package for Ubuntu and Debian)]**로 이동한다.

Installing the Amazon EFS client on other Linux distributions

If you don't want to get the `amazon-efs-utils` package from the Amazon Linux or Amazon Linux 2 AMI package repositories, it is also available on GitHub.

After you clone the package, you can build and install `amazon-efs-utils` using one of the following methods, depending on the package type supported by your Linux distribution:

- **RPM** – This package type is supported by Amazon Linux, Amazon Linux 2 Red Hat Linux, CentOS, and similar.
- **DEB** – This package type is supported by Ubuntu, Debian, and similar.

▶ To build and install `amazon-efs-utils` as an RPM package for Amazon Linux, Amazon Linux 2, and Linux distributions other than OpenSUSE or SLES

▶ To build and install `amazon-efs-utils` as an RPM package for OpenSUSE and SLES

▶ To build and install `amazon-efs-utils` as a Debian package for Ubuntu and Debian

20. 해당 페이지의 순서대로 차례로 실행한다.

```
$ sudo apt-get update
```

```
$ sudo apt-get -y install git binutils
```

```
$ git clone https://github.com/aws/efs-utils
```

```
$ cd efs-utils
```

```
$ ./build-deb.sh
```

```
$ sudo apt-get -y install ./build/amazon-efs-utils*.deb
```

```
proc-fs-nfsd.mount is a disabled or a static unit, not starting it.
rpc-gssd.service is a disabled or a static unit, not starting it.
rpc-statd-notify.service is a disabled or a static unit, not starting it.
rpc-statd.service is a disabled or a static unit, not starting it.
rpc-svcgssd.service is a disabled or a static unit, not starting it.
rpc_pipefs.target is a disabled or a static unit, not starting it.
var-lib-nfs-rpc_pipefs.mount is a disabled or a static unit, not starting it.
Setting up amazon-efs-utils (1.35.0) ...
Processing triggers for man-db (2.10.2-1) ...
Processing triggers for libc-bin (2.35-0ubuntu3.3) ...
Scanning processes...
Scanning linux images...

Running kernel seems to be up-to-date.

No services need to be restarted.

No containers need to be restarted.

No user sessions are running outdated binaries.

No VM guests are running outdated hypervisor (qemu) binaries on this host.
N: Download is performed unsandboxed as root as file '/home/ubuntu/efs-utils/build/amazon-efs-utils-1.35.0-1_all.deb' couldn't be accessed by user '_apt'. - pkgAcquire::Run (13: Permission denied)
ubuntu@ip-10-0-20-105:~/efs-utils$
```

21. lab-amazon-ec2 명령창에서 다음의 명령으로 **efs** 디렉토리를 생성한다.

```
$ mkdir efs
```

```
[ec2-user@ip-10-0-10-116 ~]$ mkdir efs
[ec2-user@ip-10-0-10-116 ~]$ ls
efs
[ec2-user@ip-10-0-10-116 ~]$
```

22. 위 15번의 **[연결]** 페이지에서 **[EFS 탑재 헬퍼 사용]**의 부분을 복사한 다음, lab-amazon-ec2 명령창에서 붙여넣기 하여 실행한다.

연결

Linux 인스턴스에 Amazon EFS 파일 시스템을 탑재합니다. 자세히 알아보기 [\[?\]](#)

☒ DNS를 통한 탑재 ☐ IP를 통해 탑재

EFS 탑재 셸퍼 사용:

```
sudo mount -t efs -o tls fs-0e8c98903277b82a7:/ efs
```

NFS 클라이언트 사용:

```
sudo mount -t nfs4 -o nfsvers=4.1,rsize=1048576,wsiz=1048576,hard,timeo=600,retrans=2,noresvport fs-0e8c98903277b82a7.efs.ap-northeast-2.amazonaws.com:/ efs
```

자세한 내용은 사용 설명서를 참조하십시오. 자세히 알아보기 [\[?\]](#)

닫기

\$ sudo mount -t efs -o tls fs-xxxxxxxxxx:/ efs

```
[ec2-user@ip-10-0-10-116 ~]$ sudo mount -t efs -o tls fs-0e8c98903277b82a7:/ efs
[ec2-user@ip-10-0-10-116 ~]$
```

23. 다음의 명령으로 확인할 수 있다.

\$ sudo df -h

```
[ec2-user@ip-10-0-10-116 ~]$ sudo df -h
Filesystem      Size  Used Avail Use% Mounted on
devtmpfs         468M   0  468M   0% /dev
tmpfs            477M   0  477M   0% /dev/shm
tmpfs            477M 524K  476M   1% /run
tmpfs            477M   0  477M   0% /sys/fs/cgroup
/dev/xvda1       30G  1.7G   29G   6% /
tmpfs            96M   0   96M   0% /run/user/1000
tmpfs            96M   0   96M   0% /run/user/0
127.0.0.1:/      8.0E   0  8.0E   0% /home/ec2-user/efs
[ec2-user@ip-10-0-10-116 ~]$
```

24. 또 다른 EC2 인스턴스인 lab-ubuntu-ec2에서도 동일한 작업을 수행한다.

\$ cd ~

\$ mkdir efs

\$ ls

\$ sudo mount -t efs -o tls fs-0836bc1abb7a14a8a:/ efs

```
ubuntu@ip-10-0-20-105:~$ ls
efs  efs-utils
ubuntu@ip-10-0-20-105:~$ sudo mount -t efs -o tls fs-0e8c98903277b82a7:/ efs
ubuntu@ip-10-0-20-105:~$
```

25. 역시 **lab-ubuntu-ec2**에서도 마운트 정보를 확인하자.

```
$ sudo df -h
```

```
ubuntu@ip-10-0-20-105:~$ sudo df -h
Filesystem      Size  Used Avail Use% Mounted on
/dev/root        29G   1.8G   28G   7% /
tmpfs            475M    0   475M   0% /dev/shm
tmpfs            190M  924K   190M   1% /run
tmpfs            5.0M    0    5.0M   0% /run/lock
/dev/xvda15      105M   6.1M   99M    6% /boot/efi
tmpfs            95M   4.0K   95M    1% /run/user/1000
127.0.0.1:/      8.0E    0   8.0E   0% /home/ubuntu/efs
ubuntu@ip-10-0-20-105:~$
```

26. EFS 테스트를 위해 **lab-ubuntu-ec2** 머신에서 다음과 같이 간단한 파일 한 개를 생성하고 간단한 메시지를 입력했다.

```
$ cd efs
```

```
$ sudo nano efstest.txt
```

```
Hello, Amazon EFS^^
```

```
ubuntu@ip-10-0-20-105:~$ cd efs
ubuntu@ip-10-0-20-105:~/efs$ sudo nano efstest.txt
ubuntu@ip-10-0-20-105:~/efs$ cat efstest.txt
Hello, Amazon EFS^^
ubuntu@ip-10-0-20-105:~/efs$
```

27. **lab-amazon-ec2**에서 확인해 보자.

```
$ cd efs
```

```
$ ls
```

```
$ cat efstest.txt
```

```
[ec2-user@ip-10-0-10-116 ~]$ cd efs
[ec2-user@ip-10-0-10-116 efs]$ ls
efstest.txt
[ec2-user@ip-10-0-10-116 efs]$ cat efstest.txt
Hello, Amazon EFS^^
[ec2-user@ip-10-0-10-116 efs]$
```

28. 결국 2개의 EC2 인스턴스가 NFS를 통해 같은 EFS를 사용한다는 것을 확인할 수 있다.