

AWS S3 Glacier

[AWS CLI 사용가이드]

(주)에이클라우드

2019 April 15

기술팀

AWS S3 Glacier

[AWS CLI 사용가이드]

1. AWS IAM 계정 생성 및 설정

1. IAM 계정 생성 및 Groups 권한 설정
2. SECURITY CREDENTIAL 생성

2. AWSCLI 설치

1. PIP 설치
2. AWS CLI 설치
3. AWSCLI 설정

3. CLI 로 GLACIER 제어

1. Upload Archive
2. DOWNLOAD ARCHIVE
3. DELETE ARCHIVE

1. AWS IAM 계정 생성 및 설정

1.1 IAM 계정 생성 및 Groups 권한 설정

2. [https:// aws.amazon.com](https://aws.amazon.com)
3. AWS Management Console 접속



로그인 ⓘ

AWS 계정의 이메일 주소

또는 IAM 사용자로 로그인하기 위해 계정 ID 또는 계정 별칭 을 입력하십시오.

다음

———— AWS를 처음 사용하십니까? ————

AWS 계정 새로 만들기



계정:

acloud01

사용자 이름:

암호:

로그인

[루트 계정 자격 증명으로 로그인](#)

root 로그인

IAM 계정로그인

1.2 IAM 계정 생성 및 Groups 권한 설정

IAM 계정 생성 및 GROUPS 권한 설정 IAM 사용자를 직접 생성하여 Administrators 그룹에 추가하려면

1. <https://console.aws.amazon.com/iam/>에서 IAM 콘솔에 로그인합니다.
2. 탐색 창에서 [Users]와 [Add user]를 차례대로 선택합니다.
3. [User name]으로 “Admin”와 같은 사용자 이름을 입력합니다. 이름은 문자, 숫자, 그리고 다음과 같은 기호로 구성될 수 있습니다. 더하기(+), 등호(=), 쉼표(,),

마침표(.), 앳(@), 밑줄(_), 하이픈(-). 이름은 대소문자를 구분하지 않으며 최대 64 자입니다.

4. AWS Management Console access 옆의 확인란을 선택하고 [Custom password]를 선택한 다음 텍스트 상자에 새 사용자의 암호를 입력합니다. 선택적으로 [Require password reset]을 선택하여 다음에 사용자가 로그인할 때 의무적으로 새 암호를 선택하도록 설정할 수 있습니다.

5. [Next: Permissions]를 선택합니다.

6. [Set permissions for user] 페이지에서 [Add user to group]을 선택합니다.

7. [Create group]을 선택합니다.

8. [Create group] 대화 상자에 새 그룹의 이름을 입력합니다. 이름은 문자, 숫자, 그리고 다음과 같은 기호로 구성될 수 있습니다. 더하기(+), 등호(=), 쉼표(,), 마침표(.), 앳(@), 밑줄(_), 하이픈(-). 이름은 대소문자를 구분하지 않으며 최대 128 자입니다.

9. [Filter]로 [Job function]을 선택합니다.

10. 정책 목록에서 [Admin Access] 옆의 확인란(check box)을 선택합니다. 그런 다음 [Create group]을 선택합니다.

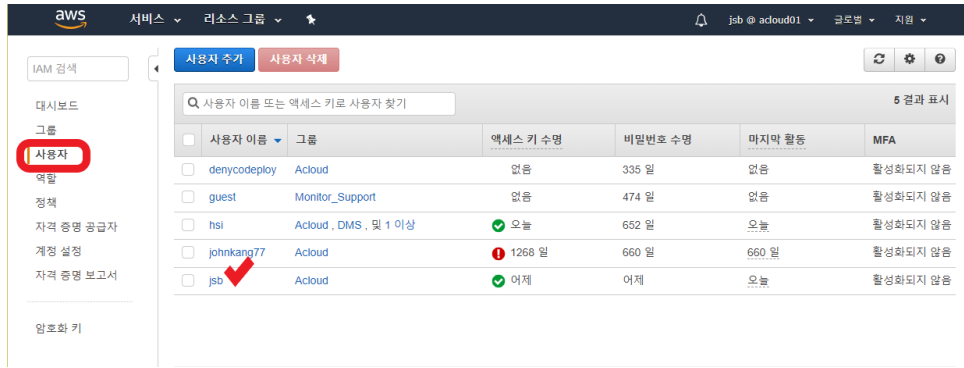
11. 그룹 목록으로 돌아가 새로 만든 그룹 옆의 확인란을 선택합니다. 목록에서 그룹을 확인하기 위해 필요한 경우 [Refresh]를 선택합니다.

12. [Next: Review]를 선택하여 새 사용자에게 추가될 그룹 멤버십의 목록을 확인합니다. 계속 진행할 준비가 되었으면 [Create user]를 선택합니다.

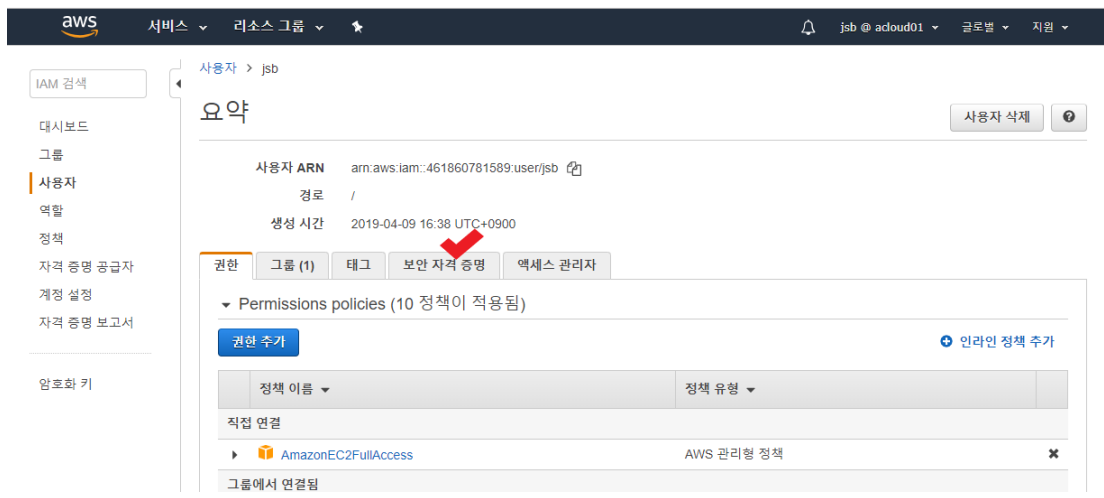
1.3 Security Credentials 생성

외부 사용자/툴에서 AWS 리소스로 보안 접속을 위한 키를 생성하여야 합니다.

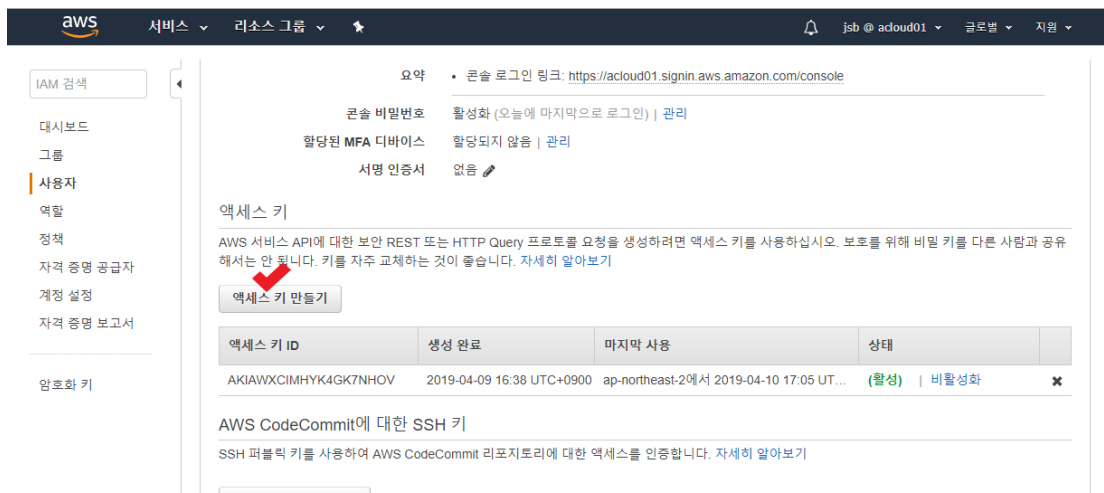
1. 앞서 생성한 IAM 사용자 계정을 선택합니다.



2. 보안 자격 증명 Tab 으로 이동합니다.



3. 액세스 키 만들기를 수행합니다.



4. 아래와 같이 팝업 창의 액세스 키 ID 와 비밀 액세스 키를 따로 적어 두거나, [.csv 파일 다운로드]를 통하여 해당 값을 local 데이터로 저장합니다.



2. AWS CLI 설치

AWS CLI 는 명령줄 셸의 명령을 사용하여 AWS 제품과 상호 작용할 수 있는 오픈 소스 도구입니다. 최소한의 구성으로, 원하는 터미널 프로그램에 있는 명령 프롬프트에서 브라우저 기반 AWS Management 콘솔이 제공하는 것과 동일한 기능을 사용할 수 있습니다.

2.1 Pip 설치

Python 용 패키지 관리자인 pip 를 사용하여 대부분의 Linux 배포에서 AWS Command Line Interface 및 해당 종속성을 설치할 수 있습니다

중요

awscli 패키지는 APT 및 yum 과 같은 다른 패키지 관리자용 리포지토리에서 사용할 수 있지만, pip 에서 가져오거나 번들설치관리자를 사용하지 않는 한 최신 버전이 보장되지 않습니다.

1. pip --version 을 실행하여 해당 버전의 Linux 에 Python 과 pip 가 이미 포함되어 있는지 확인합니다.

```
$ pip --version
```

pip 가 없는 경우 설치되어 있는 Python 버전을 확인합니다.

```
$ python --version
```

또는

```
$ python3 --version
```

Python 이 없는경우 Python 설치를 해줍니다.

Python 설치

1. Python 이 이미 설치되어 있는지 확인합니다.

```
$ python -version
```

2. Python 2.7 이상이 설치되어 있지 않은 경우 배포의 패키지 관리자를 사용하여 Python 을 설치합니다. 다음과 같이 명령과 패키지 이름이 다릅니다.

```
$ sudo apt-get install python3
```

(Debian 계열 : APT , Red Hat 계열 : yum , SUSE 계열 : zypper 를 사용합니다.)

3. python 설치가 되었는지 확인합니다.

(Python 설치가 완료되면 *Python Packaging Authority* 에서 제공하는 스크립트를 사용하여 설치합니다.)

2. curl 명령을 사용하여 설치 스크립트를 다운로드합니다.

```
$ curl -O https://bootstrap.pypa.io/get-pip.py
```

3. Python 으로 스크립트를 실행하여 pip 및 기타 필요한 지원 패키지의 최신 버전을 다운로드하여 설치합니다.

```
$ python get-pip.py --user
```

4. pip 가 있는 경로가 PATH 변수의 일부인지 확인하십시오.

```
export PATH=~/.local/bin:$PATH
```

이 명령은 경로(이 예제에서 ~/.local/bin)를 현재 PATH 변수 앞에 추가합니다.

5. 현재 세션에 프로필을 다시 로드하여 해당 변경 사항을 적용합니다.

```
$ source ~/.bash_profile
```

6. 이제 pip 가 올바르게 설치되었는지 확인합니다.

```
$ pip --version
```

2.2.AWS CLI 설치

1. pip 를 사용하여 AWS CLI 를 설치하십시오.

```
$ pip install awscli --upgrade -user
```

2. --user 스위치를 사용하면 pip 가 AWS CLI 를 ~/.local/bin 에 설치합니다

3. 최신 버전으로 업그레이드하려면 설치 명령을 다시 실행합니다.

```
$ pip install awscli --upgrade --user
```

2.3AWS CLI 설정

1. CLI 에 AWS Configure 명령을 입력합니다.

2. ACCESS KEY 와 리전, 파일형식을 잡아줍니다.

```
C:\W>aws configure
AWS Access Key ID [*****BA7P]:
AWS Secret Access Key [*****0ozD]:
Default region name [ap-northeast-2]:
Default output format [None]:
```

3. CLI 환경에서 사용할 Credential 인증이 완료되었습니다.

3. CLI 로 GLACIER 제어

Glacier 은 관리 콘솔을 제공합니다. 이 콘솔에서는 볼트를 생성하거나 삭제할 수 있습니다. 하지만 그 밖에 다른 Glacier 작업에서는 AWS Command Line Interface(CLI)를 사용하거나 코드를 작성해야 합니다.

※Glacier 를 처음 사용할 경우 용어와 개념을 알아야 합니다.

Archive: Glacier 에 데이터가 저장되는 최소단위 객체 하나의 크기는 1byte~40TB 까지 지원하며 아카이브를 업로드, 삭제할 수 있지만 수정하거나 덮어쓸 수 없습니다.

Valut: Glacier 에서 생성할 수 있는 최상위 디렉터리입니다. 리전별로 생성해야 하며 각 리전별로 최대 1000 개 까지 생성 가능합니다.

Vault Inventory: 볼트에 저장된 아카이브의 목록과 각 아카이브의 크기, 생성날짜, 아카이브 설명 정보를 표시합니다. 24 시간에 한번씩 업데이트 됩니다.

2.1 upload archive

아카이브는 단일 작업으로, 혹은 여러 파트로 나누어 업로드할 수 있습니다.

1.이 작업은 아카이브를 볼트에 추가합니다. 이는 동기식 작업이며 업로드에 성공하면 데이터가 영구적으로 유지됩니다.

```
aws glacier upload-archive --account-id <value> --vault-name my-vault -  
body archive.zip
```

```
aws glacier --account-id <value> --vault-name <value>
```

account-id-는 aws configure 에서 credential 인증을 한 현재 사용자를 나타냅니다.

```
{
```

```

    "archiveId": "kKB7ymWJVpPSwhGP6ycSOAekp9ZYe_--zM_mw6k76ZFGIEWQX-
ybtRDvc2VkPSDtfKmQrj0IRQLSGsNuDp-
AJVlu2ccmDSyDUmZwKbwbpAdGATGDiB3hH00bjbGehXTcApVud_wyDw",

    "checksum":
"969fb39823836d81f0cc028195fcdcbbbe76cdde932d4646fa7de5f21e18aa67",

    "location": "/0123456789012/vaults/my-
vault/archives/kKB7ymWJVpPSwhGP6ycSOAekp9ZYe_--zM_mw6k76ZFGIEWQX-
ybtRDvc2VkPSDtfKmQrj0IRQLSGsNuDp-
AJVlu2ccmDSyDUmZwKbwbpAdGATGDiB3hH00bjbGehXTcApVud_wyDw"
}

```

중요

Upload 후 24 시간후 볼트에 인벤토리에 아카이브가 업데이트되며 그 전까지는 확인할 수 없습니다.

2.2 archive download

Amazon S3 Glacier(Glacier)에서 아카이브를 가져오는 것은 비동기식 작업이기 때문에 먼저 작업을 시작하여 마친 후에 출력을 다운로드합니다.

1.아카이브를 검색 할 방법이 없기 때문에 아카이브가 어떤 볼트에 있는지 알아야합니다

```

$ aws glacier list-vaults --account-id -

# the output
{
  "VaultList":[
    {
      "SizeInBytes":9102027675,
      "VaultARN":"arn:aws:glacier:eu-west-
1:123456789:vaults/photos",
      "LastInventoryDate":"2017-01-16T08:06:17.699Z",
      "VaultName":"photos",
      "NumberOfArchives":2,
      "CreationDate":"2017-01-15T10:59:58.303Z"
    }
  ]
}

```

2.볼트 이름이 있으면 인벤토리 다운로드 작업을 시작할 수 있습니다. 아카이브를 식별 할 수 있도록 아카이브가 필요합니다.

```
$ aws glacier initiate-job --account-id - --vault-name photos --job-parameters '{"Type": "inventory-retrieval"}'

# the output
{
  "location":"/123456789/vaults/photos/jobs/loisaHeJvzYrsM9aewR1",
  "jobId":"loisaHeJvzYrsM9aewR1"
}
```

중요

작업 ID 를 기록해 두십시오. 다음 명령에 대해서는 작업 ID 가 필요합니다. 작업의 상태를 확인하려면 `describe-job` 명령을 사용할 수 있습니다. 3-5 시간이 소요됩니다.

3.인벤토리 파일을 다운로드하는 명령은 다음과 같습니다 `get-job-output`. 이 명령은 모든 작업의 출력을 다운로드하는 데 사용됩니다.

```
$ aws glacier get-job-output --account-id - --vault-name photos --job-id loisaHeJvzYrsM9aewR1 inventory.json

# the output
{
  "status":200,
  "acceptRanges":"bytes",
  "contentType":"application/json"
}
```

4.인벤토리 파일이 어떻게 보이는지 아래에서 확인하십시오. 이 파일에는 업로드 중에 제공된 아카이브에 대한 설명이 들어 있습니다. 아카이브 ID 를 적어 둡니다.

```
$ cat inventory.json

# the output
{
  "VaultARN":"arn:aws:glacier:eu-west-1:123456789:vaults/photos",
  "InventoryDate":"2017-01-16T08:06:17Z",
  "ArchiveList":[
    {
      "ArchiveId":"VCRQDrz8kryyhV90q1Gq",
      "ArchiveDescription":"2016 photos",
      "CreationDate":"2017-01-15T18:49:17Z",
      "Size":4534377311,
      "SHA256TreeHash":"0e7ab9f3ad2adb7a40d1..."
    },
    {
```

```

    "ArchiveId": "2CTtNlWpf6PdtDIH62Zy",
    "ArchiveDescription": "2017 photos",
    "CreationDate": "2017-01-15T20:53:55Z",
    "Size": 4567584828,
    "SHA256TreeHash": "1fc71814e54d3986bb3d..."
  }
]
}

```

5. 다운로드 요청을 시작하기 전에 명령 `archive_retrieval_request.json` 에서 매개 변수로 사용할 json 파일을 준비해야 합니다

```

$ vi archive_retrieval_request.json
{
  "Type": "archive-retrieval",
  "ArchiveId": "2CTtNlWpf6PdtDIH62Zy",
  "Description": "Download 2017 photos archive"
}

```

6. 아카이브 다운로드 요청을 시작하기 위한 명령은 다음과 같습니다.

```

$ aws glacier initiate-job --account-id - --vault-name photos --job-parameters file://archive_retrieval_request.json

# the output
{
  "location": "/123456789/vaults/photos/jobs/YBcVmT0ypwMBGjupI6oD",
  "jobId": "YBcVmT0ypwMBGjupI6oD"
}

```

7. 작업의 상태를 확인하려면 `describe-job` 명령을 사용할 수 있습니다. 3-5 시간이 소요됩니다. `describe-job` 출력은 다음과 같습니다.

```

$ aws glacier describe-job --account-id - --vault-name photos --job-id YBcVmT0ypwMBGjupI6oD

# the output
{
  "CompletionDate": "2018-12-21T23:42:24.157Z",
  "VaultARN": "arn:aws:glacier:eu-west-1:123456789:vaults/photos",
  "RetrievalByteRange": "0-4567584827",
  "SHA256TreeHash": "1fc71814e54d3986bb3d...",
  "Action": "ArchiveRetrieval",

```

```
"JobDescription": "Download 2017 photos archive",
"ArchiveId": "2CTtNlWpf6PdtDIH62Zy",
"StatusMessage": "Succeeded",
"StatusCode": "Succeeded",
"Completed": true,
"JobId": "YBcVmT0ypwMBGjupI6oD",
"Tier": "Standard",
"ArchiveSHA256TreeHash": "1fc71814e54d3986bb3d...",
"CreationDate": "2018-12-21T19:56:55.189Z",
"ArchiveSizeInBytes": 4567584828
}
```

8. 아카이브를 다운로드 할 준비가 되었습니다. 아카이브가 크다면 분할 업로드가 더 좋습니다

```
$ aws glacier get-job-output --account-id - --vault-name photos --
range bytes=0-1073741823 --job-id YBcVmT0ypwMBGjupI6oD photos

# the output
{
  "status": 206,
  "acceptRanges": "bytes",
  "contentType": "application/octet-stream",
  "checksum": "0b69951bbe9e23faf941...",
  "contentRange": "bytes 0-1073741823/4567584828",
  "archiveDescription": "Download 2017 photos archive"
}
```

2.3 Delete archive

아카이브는 볼트에서 한 번에 하나씩 삭제할 수 있습니다. 아카이브를 삭제하려면 삭제 요청 시 아카이브 ID 를 입력해야 합니다. 아카이브 ID 는 해당 아카이브가 저장된 볼트에서 볼트 인벤토리를 다운로드하여 확인할 수 있습니다.

```
$aws glacier delete-archive --account-id - --vault-name photos --archive-id
2CTtNlWpf6PdtDIH62Zy
```

AWS 클라우드 솔루션

Amazon Web Services(AWS)는 믿을 수 있는 클라우드 기반 솔루션을 제공하여 비즈니스 요구 사항을 충족할 수 있게 지원합니다. AWS 클라우드에서 솔루션을 실행하면 애플리케이션을 더욱 빠르게 시작 및 실행할 수 있으며 Pfizer, Intuit 및 미 해군 등의 조직이 사용하는 것과 동일한 수준의 보안이 제공됩니다. 또한, AWS 는 전 세계적으로 리소스를 제공하기 때문에 고객의 지역에 상관없이 솔루션을 배포할 수 있습니다. AWS 클라우드에서는 광범위한 서비스 세트, 파트너, 지원 옵션을 손쉽게 사용할 수 있기 때문에 사용자는 솔루션의 성공에만 집중할 수 있습니다.

AWS는 “클라우드 기반 솔루션을 위한 믿을 수 있는 선택”입니다.

자세한 정보는 <https://aws.amazon.com/ko/solutions/>을 참조하십시오.



Amazon Web Service, Inc.

Last Updated: December 23, 2011.

P.O. Box 81226
Seattle, WA 98108-1226
USA

<http://aws.amazon.com>



Acloud Co., Ltd.

06159

9F, Hyundai Tower, 423,
Teheran-ro, Gangnam-gu,
Seoul, Korea

<http://www.a-cloud.co.kr>

