

## [AWS] 6. 인증과 권한부여 AWS IAM

- **References**

- AWS 공인 솔루션스 아키텍트 스터디 가이드 - 어소시에이트 3/e - 6장
- <https://inpa.tistory.com/entry/AWS-iam-개념-원리user-group-policy-role-IAM-계정-정책-생성?category=947441>

## ※ AWS Identity and Access Management (IAM) 개요

- **AWS IAM**

- 계정 내 ID를 기반으로 접근성을 긴밀히 제어하는 정책 생성
- 다양한 유형의 키와 토큰을 이용한 자신의 신분 증명
- AWS 외부의 연합신분서비스를 이용한 IAM과 연계 가능한 SSO 솔루션 제공
- 리소스의 안전한 관리를 위한 계정 및 롤 환경설정, Best Practice 파악

- IAM 기반의 신분 관리

- AWS 계정 생성 시 Root 유저의 신분이 만들어지며 모든 서비스와 리소스에 접근할 수 있는 권한을 갖고 있어, 최대한 노출 가능성을 피하고 엄격히 보호해야 함 (리눅스와 동일)
- 다른 계정을 일상적인 업무에 사용할 것을 권장하며 Security Status (보안 상태) 섹션에서 루트 유저에 대한 접근가능성 경고를 제공한다.

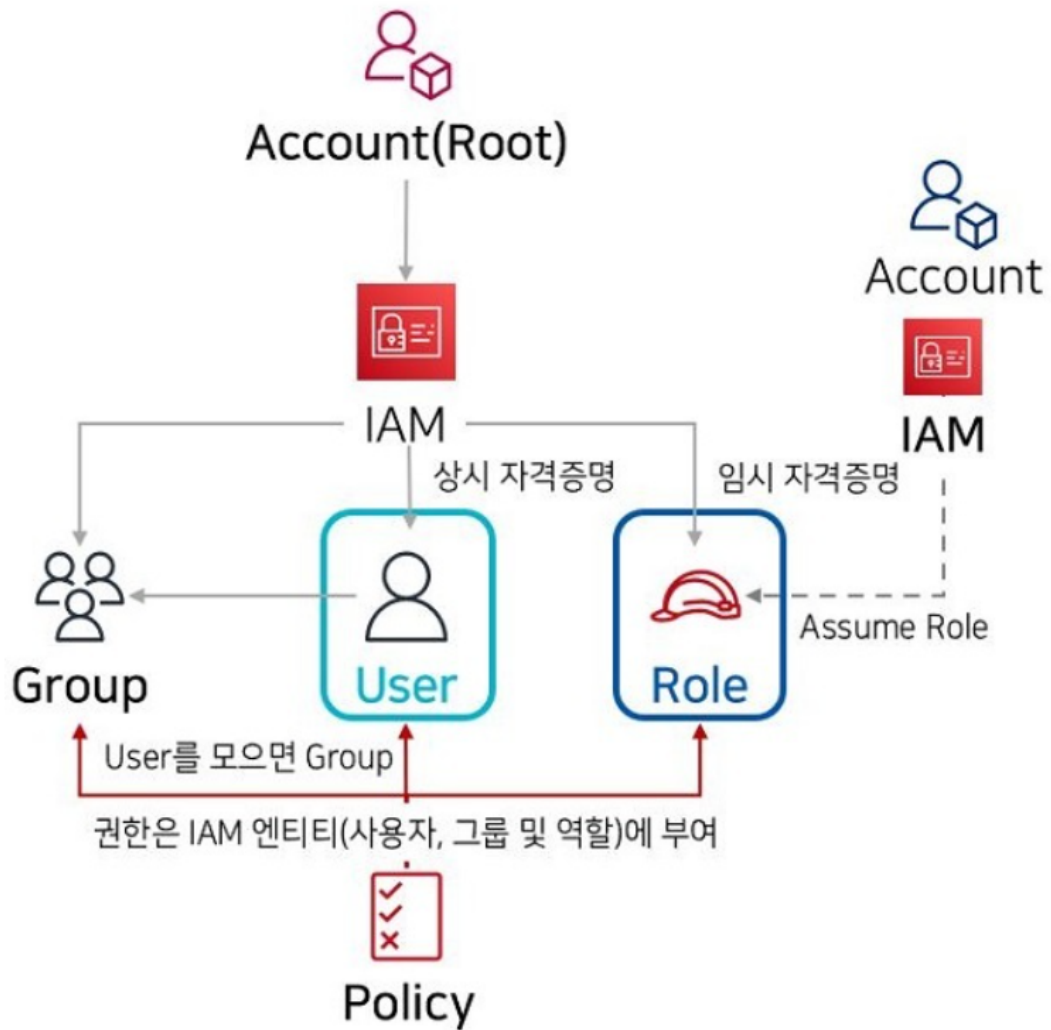
## ※ AWS Identity and Access Management (IAM) 상세

- **AWS IAM 이란**

- Identify and Access Management 의 약자로 AWS 사용자 및 그룹을 만들고 관리하며, AWS 리소스에 대한 액세스 허용 여부 설정, 전반적인 사용자를 관리하는 서비스이다.
- IAM을 통해 AWS를 사용자에게 AWS에서 제공하는 서비스들, 서비스에 생성된 자원 등에 대한 세분화된 권한을 지정할 수 있다.
  - 예를 들어 언제 어디서 누가 무엇을 어떻게 할 수 있는지를 상세히 설정 가능 (ex. 몇시부터 몇시까지만 특정자료 열람)
- 기능 요약
  - 사용자 생성/관리/계정의 보안 : 사용자의 패스워드 정책 관리 (일정 시간마다 PW 변경)
  - AWS 계정에 대한 공유 액세스 : AWS 계정의 리소스 관리 및 권한을 다른사람에게 부여 가능
  - 세분화 된 권한 : 리소스에 따라 여러사람에게 다양한 권한 부여
  - EC2 애플리케이션 권한 자격 부여 (IAM 역할 사용) : 사용자 뿐만 아닌, EC2 내 실행되는 어플리케이션에 IAM 기능을 이용하여 자격증명 제공 가능
  - 멀티 팩터 인증 (MFA) : 보안 강화를 위한 암호, 액세스 키 뿐만 아닌 디바이스 코드를 제공하는 인증을 추가 가능
  - 자격 증명 연동 : 기업네트워크, 인터넷 자격 증명 공급자와 같이 다른 곳에서 이미 암호가 있는 사용자에게 AWS 계정에 대한 임시 액세스 권한 부여 가능
  - 계정에 별명, 별칭 부여 가능 : 로그인 주소 생성 가능
  - 글로벌 서비스 : AWS 계정은 전세계에서 Unique 하다

- **IAM 구성 요소**

- 크게 사용자(Users), 그룹(Groups), 역할(Roles), 정책(Policies) 으로 구성



## • IAM 사용자 (Users)

- 실제 AWS 서비스 기능과 자원을 이용하는 사람 혹은 어플리케이션
- 사용자가 AWS 서비스를 액세스 하는 것을 조정 할 뿐만 아니라, 어플리케이션이 서비스에 액세스 하는 것을 관리
- 루트(root) 계정은 보호하는것이 권장되며 최선의 절차는 봉인 하는 것
  - 새 계정을 생성하고 어드민에 적합한 권한을 주고 사용 (보통은 AdministratorAccess 정책 적용)
  - 루트 계정과 연계된 모든 액세스키 삭제
  - 길고 복잡한 패스워드를 작성해 안전한 패스워드 볼트에 저장
  - 루트 계정에 MFA(다중인증) 기능 활성화

- 일상적인 어드민 작업에서 루트계정 사용 자제
- 콘솔 사용자 이름 아래 펼침목록 메뉴인 My Security Credentials 페이지에서 보안 설정 내역 변경이 가능
  - Password 업데이트
  - MFA 활성화 및 관리
  - AWS CLI 또는 프로그래밍 SDK를 통한 AWS 리소스를 관리하기 위한 Access Key 생성 및 삭제
  - Amazon CloudFront 배포를 위한 서명 URL 인증용 키페어 생성
  - AWS 서비스에 대한 SAOP 요청을 위한 X.509 인증서 생성

#### • IAM Access Key

- AWS CLI 또는 프로그래밍 SDK를 통한 AWS 리소스를 관리하기 위한 Access Key가 필요
- 관리 방법
  - 미사용 키 비활성화
  - 키 로테이션 (정기적 삭제)
- 키 로테이션 순서
  - 유저별 새 액세스키 생성
  - 애플리케이션의 키 정보를 새 키에 맞춰 업데이트
  - 오래된 키는 비활성화 또는 삭제
  - 키 업데이트 후에도 애플리케이션에 문제가 없는지 몇일간 확인, CLI 명령으로 구형키를 사용하는지 여부 확인 가능
    - `aws iam get-access-key-last-used --access-key-id {key}`
  - 구형키 삭제






#### • IAM 그룹 (Groups)

- 다수의 사용자를 모아놓은 개념

- 여러명의 사용자에게 공통 권한을 주기 위해 사용
- 하나의 사용자는 최대 10개 그룹에 속할 수 있다.

## • IAM 정책 (Policies)

- 사용자, 그룹, 역할이 무엇을 할 수 있는지에 대한 Permission 설정 모음
- JSON으로 작성
- 만들어진 정책 문서는 IAM 사용자, 그룹, 역할에 연결
- Dashboard의 정책 생성(Create Policy) 에서 정책을 생성하거나, JSON 포맷 텍스트로 직접 정책을 작성할 수 있다.
- 하나의 신분에는 최대 10개의 정책(6,144 미만의 정책 문서) 를 연계할 수 있다.
- 만약 한 신분에 두개의 정책이 서로 상반된 내용을 담고 있으면, 'Deny' 기준을 따르도록 한다. ( 예를들어 S3에 대한 버킷 생성 정책 Allow, Deny의 두개의 정책이 한 신분에 동시에 적용 되었을 시 Deny 적용)

		Policy name ▼	Type	Used as
<input type="radio"/>	▶	 AmazonCognitoDeveloperAu...	AWS managed	None
<input type="radio"/>	▶	 AmazonPersonalizeFullAccess	AWS managed	None
<input type="radio"/>	▶	 AmazonVPCCrossAccountN...	AWS managed	None
<input type="radio"/>	▶	 AWSBackupOperatorPolicy	AWS managed	None
<input type="radio"/>	▶	 AWSCodeBuildDeveloperAc...	AWS managed	None
<input type="radio"/>	▶	NoPermissionsAccessPolicy...	Customer managed	Permissions policy (1)
<input type="radio"/>	▶	SessionManagerPermissions	Customer managed	None

## • IAM Policy 타입 개요

- 자격 증명 기반 (Identity-based policies)
  - AWS 관리형 정책 - AWS 에서 미리 제공하는 정책

- AWS 고객 관리형 정책 - 고객이 직접 만들어 사용하는 정책
    - AWS 인라인 정책 - 단일 사용자, 그룹, 역할(Role) 에 직접 추가 하는 방식
  - 리소스 정책 기반 (Resource-based policies)
  - 권한 경계 기반 (Permissions boundaries)
  - 조직 SCP 기반 (Organizations SCPs)
  - 액세스 제어 리스트 (Access control lists - ACLs)
  - 세션 정책 (Session policies)
- 
- **자격 증명 기반 정책 (Identity-based Policy) → 사용자, 그룹에 정책 적용**
    - AWS 관리형 정책 (AWS Managed Policy)
      - AWS에서 생성 및 관리하는 독립적인 정책
      - 예시로 IAM 정책 탭에서 나오는 미리보기 정책
      - 정책 스스로 정책 이름이 포함된 Amazon 리소스 이름을 갖고있다. (arn)
      - 미리 만들어진 글로벌한 정책
    - AWS 고객관리형 정책 (AWS Customer Managed Policy)
      - 사용자가 정책을 새로 생성해서 커스텀 한 것
      - 만들어진 정책은 Account 내에서만 사용 가능하다.
    - AWS 인라인 정책 (AWS Inline Policy)
      - 1 to 1 정책으로 명시적으로 할당되는 정책
      - 하나의 사용자에게 하나의 정책을 적용하는 개념
      - AWS IAM에서는 Inline Policy 보다 Managed Policy 를 권장하지만, 명시적으로 특정 사용자에게 특정 권한을 주고 싶을 경우에 Inline Policy가 유용하다.
- 
- **리소스 기반 정책 (Resource-Based Policy) → 리소스에 적용**
    - AWS 서비스 리소스에 적용하는 정책

- IAM Policy JSON 문서 구조

## IAM Policy 의 구조

```

{
  "Statement": [{
    "Effect": "Allow or Deny",
    "Principal": "principal",
    "Action": "action",
    "Resource": "arn",
    "Condition": {
      "condition": {
        "key": "value"
      }
    }
  }]
}

```

**Effect** – 명시된 정책에 대한 허용 혹은 차단

**Principal** – 접근을 허용 혹은 차단하고자 하는 대상  
*"Principal": "AWS": "arn:aws:iam::123456789012:user/username"*

**Action** – 허용 혹은 차단하고자하는 접근 타입  
*"Action": "s3:GetObject"*

**Resource** – 요청의 목적지가 되는 서비스  
*"Resource": "arn:aws:sqs:us-west-2:123456789012:queue1"*

**Condition** – 명시된 조건 유효하다고 판단될 수 있는 조건  
*"StringEqualsIfExists": {"aws:RequestTag/project": ["Pickles"]}*

© 2019 Amazon Web Services, Inc. or its affiliates. All rights reserved.

- SID (string) : Statement ID로 statement 를 구분하기 위해서 사용 (Optional)
- Effect : 액세스를 허용하는지 거부하는지 설정. (Allow / Deny)
- Principal / NotPrincipal : 액세스를 허용하거나 거부할 대상(사용자, 역할 등)을 지정
  - 리소스 기반 정책에서만 사용 (뒤에서 자세히 소개)
- Action / NotAction (string | array) : 서비스의 API Calls를 지정
  - 하나 혹은 여러 개의 Action을 지정할 수 있다.
  - 각 서비스별로 고유의 서비스 접두사(예: DynamoDB는 dynamodb, S3는 s3)가 있다.
  - Action은 (서비스 접두사):(작업) 형식으로 작성 → dynamodb:DeleteItem, s3:GetObject
- Resource, NotResource : Action이 영향을 미치는 리소스 리스트를 지정
  - 작업이 적용되는 리소스 목록을 지정
  - 리소스 기반 정책을 생성하는 경우 선택 사항이 요소를 포함하지 않으면 작업이 적용되는 리소스는 정책이 연결된 리소스

- 리소스를 특정할 수 없는 일부 서비스에서는, Resource를 비워두는 대신 \* 를 입력
- Condition (object) : 조건을 넣어 줄 수 있다. 즉, 조건을 충족되는 경우에만 해당 정책을 적용시킬 수 있는 개념

Conditions	상세 기능
NumericEquals	일치

## • IAM 역할 (Role)

- 사용자의 권한 관리를 용이하게 하기 위해 그룹을 사용하지만, 권한이 다양해질 경우 그룹이 복잡해지고 관리가 어려워져 Role이라는 개념이 등장
- AWS 리소스가 무엇을 할 수 있는지 정의하는 자격 증명서
- 역할을 그룹이나 사용자에게 주는 방식
- 다른 사용자가 역할을 부여 받아 사용 가능
- AWS 사용자 뿐만 아닌 리소스에 대해 폭 넓게 접근 권한 설정 가능
- 일시적 권한을 위임
- Policy의 설정 없이 임시 자격증명서를 통해 임시 세션 토큰을 발급 받게 되고, 역할에 연결된 권한을 사용
- Role은 접근이 필요한 리소스의 신뢰 개체(Trusted entity) 를 정의 하는 방식으로 생성 가능
- 신뢰 개체 정의 후 사용자는 정책 문서 생성 및 부착 또는 사전 정의된 IAM 정책을 할당하는 방식으로 퍼미션 부여
- 신뢰 개체에 새 룰이 포함돼 있는 경우 AWS 는 AWS Security Token Service (STS) 를 이용하여 만료기한이 정해진 보안 토큰을 발행

## • IAM Role Trusted Entity (신뢰 개체)

- AWS 서비스
- AWS 계정



- Amazon, Amazon Cognito, Facebook, Google 로그인으로 접근권한을 증명한 웹 식별 객체
- SAML 2.0 연합 자격인증 객체

## • IAM 자격증명 보고서

- IAM 현재 계정 세팅 및 접속정보를 총합한 보고서
- 계정의 모든 사용자와 암호, 액세스키, MFA 장치 등의 증명상태정보가 들어 있다.
- AWS 콘솔, CLI, API에서 생성요청 및 다운로드 가능
- 4시간 주기로 갱신된다.

## • 접근권한 관리 도구

- AWS는 IAM 이외에도 사용자 및 리소스 관리에 다양한 도구를 제공

사용자 접근권한 관리 도구	설명
Amazon Cognito	모바일 앱 및 웹 앱 개발자를 위한 회원가입 및 로그인 기능 제공 - <b>Cognito user pool</b> : 애플리케이션에 회원가입 및 로그인 기능을 추가할 수 있다. - <b>Cognito Identity pool</b> : 애플리케이션 사용자에게 다른 서비스에 대한 임시 접근권한을 부여 가능하다.
AWS Managed Microsoft AD	AWS Directory Service 를 통한 AD 접근, 액티브 디렉토리 접근 서비스 AD Connector를 이용하면 AWS 서비스와 온프레미스 Microsoft Active Directory를 바로 연결 가능
AWS Single Sign-On	AWS Directory Service로 관리되는 Microsoft AD의 신분 확인및 권한부여작업과 AWS Organizations에 포함된 다수의 AWS 계정에서도 사용 가능하다. Salesforce, Box, Office365 등 다양한 애플리케이션과 SAML2.0 을 지원하는 커스텀 앱 에서도 사용 가능
키 암호화 및 보안 자격증명 도구	설명
AWS Key Management Service	AWS 서비스를 이용하기 위한 암호화 키 생성 및 관리 서비스 키 생성, 순회, 삭제 기능 제공 AWS CloudTrail과 KMS를 통합해 기업의 감사업무 및 준법감시 업무에 사용 가능 하다.
AWS Secrets Manager	IAM Role은 Credential(인증자격) 정보를 안전하게 전달할 수 없다, dldp 패스워드와 서드파티 API 키 등 애플리케이션에서 필요로 하는 시크릿 리소스를

	전문적으로 다룰 수 있는 도구가 AWS Secrets Manager
AWS CloudHSM	CloudHSM Hardware Security Module 의 약자로 웹 서버의 암호화 키 생성, 정렬, 관리 부담을 덜어주는 것이 주 목적이며 웹서버 인프라의 암호화 작업을 위해 전용 가상 연산 기기 클러스터링 한다 애플리케이션 데몬으로 CloudHSM 클라이언트를 실행하여 HSM 클러스터를 활성화 가능하며, HSM 과 완벽하게 암호화된 방식으로만 소통하도록 설정 가능 AWS KMS와 유사하지만 다음과 같은 특징 및 장점 1. 키는 높은 수준의 보안성이 검증된 전용 서드 파티 HSM에 저장 2. FIPS 140-2 규정에 부합 3. PKCS#11, Java JCE, Microsoft CNG 인터페이스를 통해 애플리케이션과 통합 기능 제공 4. VPC 내 고성능 암호화 생성 가속 기능(대량의 암호 생성시 사용)

## ※ 추가 스터디 자료

### • Microsoft Active Directory (AD 란)

- <https://co-no.tistory.com/30> (참고)
- 중앙에서 Admin이 사용자 인증 및 권한 부여 처리가 가능 하도록 하여, 기업 내의 자원 및 권한 관리에 용이
- LDAP(Lightweight Directory Access Protocol) : TCP/IP 위에서 디렉토리 서비스를 조회하고 수정하는 응용 프로토콜.
- 디렉토리 서비스(Directory Service) : 분산된 네트워크 환경에서 네트워크의 사용자와 네트워크 자원에 대한 정보를 중앙의 저장소에 통합하고 조직 및 관리하는 응용 소프트웨어. 사용자와 공유된 자원 사이에서 추상 계층으로 동작.
- 기본적으로 AD는 사용자가 마이크로소프트 IT 환경에서 업무를 수행하는 데 도움을 주는 데이터베이스이자 서비스 집합,
  - **데이터베이스(또는 디렉토리)** 는 환경에 대한 중요한 정보를 담고 있으며, 사용자와 컴퓨터 목록, 누가 무엇을 할 수 있는지에 대한 정보 등이 포함
  - **서비스** 는 IT 환경에서 일어나는 대부분의 활동을 제어, 특히 서비스는 일반적으로 사용자가 입력하는 사용자 ID와 비밀번호를 확인하는 방법으로, 사용자가

주장하는 본인이 맞는지 검증하고(인증), 각기 허용된 데이터에만 액세스할 수 있도록 한다.

- AD 관련 용어정리

- 도메인 : AD에서 기본이 되는 관리 대상 단위이자, AD 가 설치된 윈도우 서버가 하나의 도메인
- 도메인 컨트롤러 : 도메인을 관리하는 서버 컴퓨터