Identity and Access Management service - Global

- All users, roles, policies etc are global

Principals

- principal is an identitly allowed to interact with AWS resources.
- can be permanent or temporary
- 3 types: root, IAM user, roles & temporary security tokens

Users

```
- Root user (Owner of AWS account)
- IAM users
- Default new users (non root) have non-explcit deny (No access to anything)
- Represent Users or Applications
```

Roles

```
- collection of permissions (policies)
- can be assigned to EC2 instance
- used to grant cross account access
```

Authentication

```
- 3 methods:
    - User/Password
    - Access Key - Key Id and Secret Key
    - Access Key/Session Token - assumed role
```

Policies

```
- Policies have versions
    - you can specify which is current
    - you can delete versions
- Used for Authorization
- Policy contains:
    - Effect (allow/deny)
    - Service - aws service (e.g s3)
    - Resource (ARN) (e.g. specific bucket)
    - Action - action within service (e.g. write)
    - Condition - optional contidion (e.g. only if ip matches or time interval)
- Policy may exists only for User (user policy, inline policy) or as Managed
policy
- Policy may be associated directly to User or through Group
- Policy may be attached to a Role
```

Groups

- Used to Attach the same policies to multiple users

Conflicting Policies

```
- explicit deny always wins
- explicit allow override implicit deny
```

Cross Account Access

```
- Account A wants to have access to resources in account B
- Create role in account B with necessary permissions
- Add trust policy for the role with Principal AWS account A
- Create a role in account A which allows AssumeRole from account B
- You can enforce MFA for cross account access
```

Identity Federation

```
- two types of Identiti Providers:
    - OpenId Connect for Google, Facebook etc,
    - SAML compliant such as Active Directory Federation Services (ADFS)
- issues temporary security tokens with a role (can be specified by internal
directory by username or some metadata)
-
```

STS

- Service which creates tokens
- Methods Available:

    - AssumeRoleWithSAML – for AD
    - AssumeRoleWithWebIdentity – for OpenID
    - GetSessionToken to generate Token with longer timeout (up to 36h). Must use long term AWS security credentials

- By default session tokens have a timeout of 1 hour. Configurable timeout 15 minutes to 1h

- You have to pass policy when you call service to get a token, you can scope down permission from the role