Role-Based Access Control (RBAC)

- Traditional permissions model based on job functions
- **Challenges**:
    - Requires maintaining multiple policies
    - Time-consuming to update when new resources are added
    - Less flexible for dynamic environments

Attribute-Based Access Control (ABAC)

- **Definition**: Authorization strategy defining permissions based on attributes
- **Key Characteristics**:
    - Uses tags (key-value pairs) for access control
    - Applies to both IAM resources and AWS resources
- **Benefits**:
    - More flexible than traditional resource-listing policies
    - Enables granular permissions without constant updates
    - Highly scalable approach
    - Fully auditable

Tagging in AWS

- **Metadata Structure**:
    - Key/value pair resource labels
    - Up to 50 tags per resource
    - Case-sensitive
- **Practical Uses**:
    - Billing identification
    - Resource filtering
    - Access control management

Identity Federation Strategies

Identity Federation Fundamentals

- **Core Concept**: Trust system between authentication and resource access parties
- **Components**:
    - **Identity Provider (IdP)**: Authenticates users
        - Examples:
            - OIDC: Amazon, Facebook, Google
            - SAML: Active Directory, Shibboleth
    - **Service Provider (SP)**: Controls resource access
        - Examples: AWS services, social platforms, online banking

AWS Identity Federation Services

- **Supported Services**:
    1. AWS IAM
    2. AWS IAM Identity Center
    3. AWS Security Token Service (STS)
    4. Amazon Cognito

Amazon Cognito

- **Features**:
    - Authentication for web/mobile applications
    - User management
    - Federated identity support
- **Key Components**:
    - **User Pools**: User directory with authentication
    - **Identity Pools**: Create unique user identities and permissions

Multi-Account Management

AWS Organizations

- **Purpose**: Centralized management of multiple AWS accounts
- **Key Features**:
    - Consolidated billing
    - Hierarchical account grouping
    - Centralized policy control

Service Control Policies (SCPs)

- **Function**: Set maximum permissions across organization
- **Characteristics**:
    - Applied to root, organizational units (OUs), or specific accounts
    - Cannot be overridden by local administrators
    - Do not directly grant permissions

Permissions Boundaries and SCPs Comparison

Permission Boundary	**Organizational SCP**
Applies to individual IAM entities	Applies to entire organization/OU
Defines maximum identity-based policy permissions	Defines maximum account member permissions
Typically scopes resource access	Often used to deny specific services

Advanced Access Control Techniques

Policy Evaluation Logic

- Multiple policy types can impact permissions
- Explicit deny in any policy overrides allow
- Effective permissions = intersection of all applicable policies

Best Practices

- Use IAM groups for consistent access rights
- Implement ABAC for scalable permissions
- Leverage identity federation for centralized authentication
- Utilize AWS Control Tower for governance

Mermaid Diagram: Identity Federation Flow