

Identity & Access Management (IAM)

Tejas Parikh (t.parikh@northeastern.edu)

CSYE 6225
Northeastern University

Identity and Access Management (IAM)

According to Gartner

Identity and Access Management (IAM) is the security discipline that enables the right individuals to access the right resources at the right times for the right reasons.

Identity and Access Management (IAM)

- AWS Identity and Access Management (IAM) enables you to securely control access to AWS services and resources for your users.
- Using IAM, you can create and manage AWS users and groups, and use permissions to allow and deny their access to AWS resources.

IAM Features

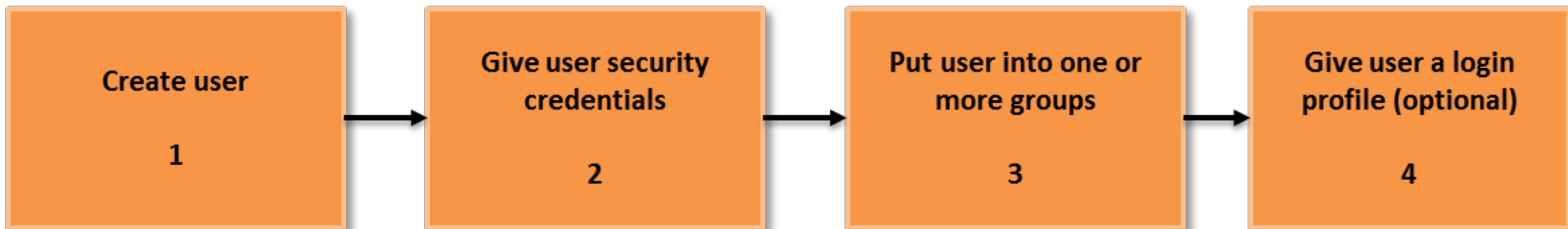
- **Shared access to your AWS account** - You can grant other people permission to administer and use resources in your AWS account without having to share your password or access key.
- **Granular permissions** - You can grant different permissions to different people for different resources.
- **Secure access to AWS resources for applications that run on Amazon EC2** - You can use IAM features to securely give applications that run on EC2 instances the credentials that they need in order to access other AWS resources.

Benefits of IAM

- Enhanced Security
- Granular control
- Temporary Credentials
- Flexible security credential management
- Leverage external identity systems
- Seamlessly integrated into AWS services

Enhanced Security

- IAM enables security best practices by allowing you to grant unique security credentials to users and groups to specify which AWS service APIs and resources they can access.
- IAM is secure by default; users have no access to AWS resources until permissions are explicitly granted.



Granular control

- IAM provides the granularity to control a user's access to specific AWS services and resources using permissions. For example, terminating EC2 instances or reading the contents of an Amazon S3 bucket.

Temporary Credentials

- In addition to defining access permissions directly to users and groups, IAM lets you create roles.
- Roles allow you to define a set of permissions and then let authenticated users or EC2 instances assume them, getting temporary access to the resources you define.

Flexible security credential management

- IAM allows you to authenticate users in several ways, depending on how they want to use AWS services. You can assign a range of security credentials including passwords, key pairs, and X.509 certificates
- You can also enforce [multi-factor authentication \(MFA\)](#) on users who access the AWS Management Console or use APIs

Leverage external identity systems

- You can use IAM to grant your employees and applications access to the AWS Management Console and to AWS service APIs, using your existing identity systems.
- AWS supports federation from corporate systems like Microsoft Active Directory as well as external Web Identity Providers like Google and Facebook.

Seamlessly integrated into AWS services

- IAM is integrated into most AWS services.
- This provides the ability to define access controls from one place in the AWS Management Console that will take effect throughout your AWS environment.

IAM Best Practices

- Lock Away Your AWS Account Root User Access Keys
- Create Individual IAM Users
- Use AWS Defined Policies to Assign Permissions Whenever Possible
- Use Groups to Assign Permissions to IAM Users
- Grant Least Privilege
- Use Access Levels to Review IAM Permissions
- Configure a Strong Password Policy for Your Users
- Enable MFA for Privileged Users
- Use Roles for Applications That Run on Amazon EC2 Instances
- Delegate by Using Roles Instead of by Sharing Credentials
- Rotate Credentials Regularly
- Remove Unnecessary Credentials

Additional Resources

See Lecture Page