Reading 1.8: Role Based Access in AWS

Throughout these last few lessons, there have been sprinklings of IAM best practices. It's helpful to have a brief summary of some of the most important IAM best practices you need to be familiar with before building out solutions on AWS.

Lock Down the AWS Root User

The root user is an all-powerful and all-knowing identity within your AWS account. If a malicious user were to gain control of root-user credentials, they would be able to access every resource within your account, including personal and billing information. To lock down the root user:

- Don't share the credentials associated with the root user.
- Consider deleting the root user access keys.
- Enable MFA on the root account.

Follow the Principle of Least Privilege

Least privilege is a standard security principle that advises you to grant only the necessary permissions to do a particular job and nothing more. To implement least privilege for access control, start with the minimum set of permissions in an IAM policy and then grant additional permissions as necessary for a user, group, or role.

Use IAM Appropriately

IAM is used to secure access to your AWS account and resources. It simply provides a way to create and manage users, groups, and roles to access resources within a single AWS account. IAM is **not** used for website authentication and authorization, such as providing users of a website with sign-in and sign-up functionality. IAM also does **not** support security controls for protecting operating systems and networks.

Use IAM Roles When Possible

Maintaining roles is easier than maintaining users. When you assume a role, IAM dynamically provides temporary credentials that expire after a defined period of time, between 15 minutes and 36 hours. Users, on the other hand, have long-term credentials in the form of user name and password combinations or a set of access keys. User access keys only expire when you or the admin of your account rotates these keys. User login credentials expire if you have applied a password policy to your account that forces users to rotate their passwords.

Consider Using an Identity Provider

If you decide to make your cat photo application into a business and begin to have more than a handful of people working on it, consider managing employee identity information through an identity provider (IdP). Using an IdP, whether it be an AWS service such as AWS IAM Identity Center (Successor to AWS Single Sign-On) or a third-party identity provider, provides you a single source of truth for all identities in your organization. You no longer have to create separate IAM users in AWS. You can instead use IAM roles to provide permissions to identities that are federated from your IdP. For example, you have an employee, Martha, that has access to multiple AWS accounts. Instead of creating and managing multiple IAM users named Martha in each of those AWS accounts, you can manage Martha in your company's IdP. If Martha moves within the company or leaves the company, Martha can be updated in the IdP, rather than in every AWS account you have.

Consider AWS IAM Identity Center (Successor to AWS Single Sign-On)

If you have an organization that spans many employees and multiple AWS accounts, you may want your employees to sign in with a single credential. AWS IAM Identity Center is an IdP that lets your users sign in to a user portal with a single set of credentials. It then provides them access to all their assigned accounts and applications in one central location. AWS IAM Identity Center is the successor to AWS Single Sign-On. If you hear instructors in this course mentioning AWS SSO, they are referring

to AWS IAM Identity Center. AWS IAM Identity Center is similar to IAM, in that it offers a directory where you can create users, organize them in groups, and set permissions across those groups, and grant access to AWS resources. However, AWS IAM Identity Center has some advantages over IAM. For example, if you're using a third-party IdP, you can sync your users and groups to AWS IAM Identity Center. This removes the burden of having to re-create users that already exist elsewhere, and it enables you to manage those users from your IdP. More importantly, AWS IAM Identity Center separates the duties between your IdP and AWS, ensuring that your cloud access management is not inside or dependent on your IdP.