Identity & Access Management (IAM)

EXAM CHEAT SHEET

SOLUTIONS ARCHITECT

- Names of users, groups, and roles must be unique within the account. They are not distinguished by case, for example, you cannot create groups named both "ADMINS" and "admins".
- IAM policies are not region specific. Conditions can be used to restrict by region.
- You can create resource-based policies (as opposed to IAM policies) only for Amazon S3 buckets (bucket policies and ACLs), Amazon Glacier vaults (vault access policies), Amazon SNS topics, Amazon SQS queues, and AWS Key Management Service encryption keys. Resource-based policies include a Principal element to specify an IAM identity that can access that resource.
- Not all AWS services support resource-level permissions in IAM policies and not all actions.
- Resource based tagging can be used with IAM policies to restrict access to resources by user, group or role.
- IAM Managed policies for job functions:
    - Administrator
    - Billing
    - Database Administrator
    - Data Scientist
    - Developer Power User
    - Network Administrator
    - System Administrator
    - Security Auditor
    - Support User
    - View-Only User
- By default, a request is denied, but this can be overridden by an allow. In contrast, if a policy explicitly denies a request, that deny can't be overridden.
- When you use the AWS Management Console to delete an IAM user, IAM automatically deletes the following information for you:
    - The user
    - Any group memberships
    - Any password associated with the user
    - Any access keys belonging to the user
    - All inline policies embedded in the user
    - Any associated MFA device
- Methods of federating users:

    - Amazon Cognito (developer authenticated identities, guest access or public identity service provider).
    - Public Identity Service Providers or OpenID Connect (Facebook, Google, Amazon etc).
    - Identity provider software package that supports SAML 2.0 (Security Assertion Markup Language 2.0).
    - Creating a custom identity broker application that authenticates users (e.g. with the enterprise's LDAP or Active Directory service). The application then assumes temporary credentials for the user.
- IAM roles should be used for applications on EC2 not AWS credentials.
- IAM roles for web identity federation should be used for web application users to access resources.
- IAM roles with cross account access grant users in one AWS account access to resources in another account.
- Root user cannot access a bucket if the bucket policy doesn't specify the root user as principal although root user can change the bucket policy.
- You can use a password policy to do these things:
    - Set a minimum password length.
    - Require specific character types, including uppercase letters, lowercase letters, numbers, and non-alphanumeric characters. Be sure to remind your users that passwords are case sensitive.
    - Allow all IAM users to change their own passwords.
        - Note: When you allow your IAM users to change their own passwords, IAM automatically allows them to view the password policy. IAM users need permission to view the account's password policy in order to create a password that complies with the policy.
    - Require IAM users to change their password after a specified period of time (enable password expiration).
    - Prevent IAM users from reusing previous passwords.
    - Force IAM users to contact an account administrator when the user has allowed his or her password to expire.
- 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
    - Use Policy Conditions for Extra Security
    - Monitor Activity in Your AWS Account

- All the above and;
- A user can only change their own password if they have permission to.
- You're not required to specify any Condition elements in an IAM policy.
- Statement and Effect are required elements.
- If you want the URL for your sign-in page to contain your company name (or other friendly identifier) instead of your AWS account ID, you can create an alias for your AWS account ID.
- Your sign-in page URL has the following format, by default.
    - https://Your_AWS_Account_ID.signin.aws.amazon.com/console/
- If you create an AWS account alias for your AWS account ID, your sign-in page URL will look like the following example.
    - https://Your_Alias.signin.aws.amazon.com/console/
- You can create a policy from templates or using the policy generator.
- Costs cannot be tracked by user, group or role.
- Access can be stopped without deleting a user by disabling a user's access key, which means it can't be used for API calls (including console).
- Know and understand the example IAM policies at: http://docs.aws.amazon.com/IAM/latest/UserGuide/access_policies_examples.html
- Know and understand IAM troubleshooting: http://docs.aws.amazon.com/IAM/latest/UserGuide/troubleshoot.html