AWS Governance

14 March 2024 15:53

1. Introduction to AWS Organizations:
    - AWS Organizations allows organizations to create and manage multiple AWS accounts centrally, streamlining management processes and enhancing security, billing, and organizational structure.
2. Account Types:
    - Management Account: Also known as the payer account, it serves as the central administrative and billing account for the entire organization. It hosts and manages the AWS organization and controls various aspects of the organizational structure.
    - Member Account: These are all other AWS accounts belonging to the organization, linked to the management account. Member accounts can be managed collectively under the organization and may include development, testing, and production accounts.
3. Key Features of AWS Organizations:
    - Consolidated Billing: Bills from all member accounts are consolidated and rolled up to the single payer account, simplifying the billing process and enabling cost tracking at a granular level.
    - Aggregate Usage Discounts: By combining usage across all member accounts, organizations can qualify for volume-based discounts, resulting in cost savings.
    - Sharing Reserved Instances and Savings Plans: Purchases of reserved instances and savings plans made in one account can be shared and utilized across other accounts within the organization.
4. Concepts:
    - Multi-Account Design and Strategy: Enables centralized management of multiple accounts, offering improved security, cost management, and resource organization.
    - Organizational Units (OUs): Logical groupings of multiple accounts within the organization, allowing structured account separation and management.
    - Service Control Policies (SCPs): Define permissions and restrictions for resources and actions within accounts, providing centralized control over allowed or denied actions.
    - Centralized Logging Accounts: Recommended for aggregating CloudTrail logs from all member accounts, enhancing security and simplifying auditing.
    - Cross-Account Roles: Best practice for managing resources across the organization, enabling secure access to member accounts.
5. Example of SCP Application:
    - Demonstrated how SCPs are applied to different organizational units and accounts within an AWS organization to enforce specific permissions and restrictions.
6. Exam Tips:
    - Understand the purpose and benefits of AWS Organizations for managing multi-account environments.
    - Know how SCPs, consolidated billing, and resource sharing work within AWS Organizations.
    - Be familiar with concepts such as organizational units, centralized logging, and cross-account roles.
    - Practice reading and understanding SCP JSON policies for the exam.

1. AWS Resource Access Manager (RAM):
    - RAM is a service provided by AWS that simplifies the sharing of resources across different AWS accounts, even if they are outside your organization.
    - It streamlines resource sharing and management while improving resource utilization and access control.
2. Shared Resources:
    - RAM allows sharing various resources, including transit gateways, VPC subnets, license manager resources, Route 53 Resolver rules and endpoints, and dedicated

hosts.
    - These resources help optimize resource utilization and enable efficient management across accounts.
3. Owners and Participants:
    - In the context of sharing VPC resources, owners create and manage the resources intended to be shared.
    - Participants are the recipients of shared resources. They can provision services into shared resources but cannot modify or delete them. Owners maintain control over the configuration and management of shared resources.
4. Exam Tips:
    - RAM simplifies resource sharing and eliminates the need to recreate resources for each account. However, be mindful of the associated costs for shared resources.
    - Understand the appropriate use cases for sharing VPC resources like subnets compared to VPC peering.
    - Recognize the limitations of participant accounts in modifying or deleting shared resources for effective resource sharing.

By understanding these concepts, you'll be well-prepared to utilize AWS RAM effectively for resource sharing across your AWS accounts.

From <https://chat.openai.com/c/d6fe90a1-a00b-4374-9bfc-54c5891880f3>

1. Need for Cross-Account Access:
    - As organizations manage multiple AWS accounts, setting up efficient and secure cross-account access becomes essential to avoid security vulnerabilities and administrative overhead.
    - Cross-account role access allows temporary access with well-defined permissions and access controls, eliminating the need for creating duplicate IAM users across accounts.
2. Example Scenario:
    - We discussed an example scenario where a third-party auditor needs read-only access to AWS accounts for audit purposes.
    - The steps involved creating a new role with minimum privileges, updating the trust policy to allow the auditor's account to assume the role, providing the ARN of the internal role to the auditor, and testing the access.
3. Benefits of Cross-Account Role Access:
    - Cross-account role access offers a secure and efficient method for accessing accounts within AWS.
    - It eliminates the need for managing long-term access keys and IAM users across accounts, offering rolling temporary credentials for specific tasks.
    - Temporary credentials can be easily revoked or rotated, enhancing security and control.
4. Exam Tips:
    - When encountering scenarios involving temporary credentials, consider roles, especially cross-account IAM roles, as the preferred solution over long-term IAM users.
    - Understand the roles of permissions policy and trust policy in setting up cross-account access.
    - Avoid using permanent credentials like access keys for temporary access scenarios and prioritize the use of roles.
    - Remember that role assumption is temporary, and credentials expire based on the specified duration for the role.

By familiarizing yourself with the process of setting up cross-account role access and understanding its benefits, you'll be better equipped to manage access across AWS accounts efficiently and securely

1. AWS Config Overview:
    - AWS Config is a service provided by AWS for inventory management and control of your infrastructure.
    - It maintains a historical record of configuration changes made to your resources over time, providing visibility into resource evolution and history.
    - Config enables you to define rules to ensure resource compliance with specific configuration requirements.
    - The service integrates with Amazon SNS to send notifications and alerts when resource configurations change or do not comply with defined rules.
    - It operates on a per-region basis, meaning it must be enabled independently in each region where resources need to be recorded.
2. Monitoring State of Architectures with Config:
    - Config helps discover architecture components, define rules for compliance, and monitor resource configurations over time.
    - It allows you to create custom rules tailored to your organization's unique standards.
    - Rules can be evaluated on a schedule or based on configuration change events, providing flexibility in monitoring.
    - Config is a monitoring and assessment tool, not a preventative tool, meaning it records and alerts on changes but does not prevent them.
3. Remediating Config Findings:
    - AWS Config enables automatic remediation of non-compliant configurations using Systems Manager Automation Documents or custom Lambda functions.
    - Automation Documents can be AWS-managed or custom-made to address specific remediation tasks.
    - Automatic remediation attempts can be retried until non-compliant configurations are corrected.
4. Alerts and Events:
    - Config easily integrates with Amazon SNS topics to send notifications and alerts when configuration changes or compliance deviations occur.
    - Events can be sent to Amazon EventBridge for routing to other services, enabling real-time alerting and event-driven automation.
5. Exam Tips:
    - Understand the purpose and capabilities of AWS Config for inventory management and compliance monitoring.
    - Know how to create and manage Config rules, both AWS-managed and custom.
    - Familiarize yourself with automatic remediation options using Automation Documents and Lambda functions.
    - Remember that Config operates on a per-region basis and is not a preventative tool but rather a monitoring and assessment tool.
    - Be aware of cost considerations, as Config is not a free service and charges are based on the number of configuration items recorded and rules enabled.

>

1. AWS Directory Service Overview:
    - AWS Directory Service is a fully managed version of Active Directory, allowing users to run AD inside AWS without the complexity of setting it up on-premises.
    - It provides the familiar AD tools and features, making it easier for organizations to manage their directory services in the cloud.
2. Types of Directory Service:
    - Managed Microsoft AD: This is a fully featured Active Directory suite managed by AWS. It's suitable for organizations looking to build out a complete AD environment

in AWS without managing the infrastructure themselves.
    - AD Connector: AD Connector creates a tunnel between the AWS environment and the on-premises AD. It allows users to authenticate against AD inside AWS while keeping user data on-premises. It's useful for scenarios where organizations want to maintain their AD infrastructure on-premises but need authentication services in AWS.
    - Simple AD: Simple AD provides basic authentication services inside AWS. It's powered by a Linux Samba Active Directory-compatible server and offers simplified AD functionality. It's suitable for organizations that don't need the full suite of AD features.
3. Exam Tips:
    - Understand the differences between Managed Microsoft AD, AD Connector, and Simple AD, including their use cases and capabilities.
    - Know that AWS Directory Service is a fully managed service, and it's preferable to use it over manually setting up AD on EC2 instances.
    - Understand when to use each type of Directory Service based on organizational requirements and preferences, such as migrating everything to AWS or leaving AD on-premises.
    - Remember that the exam favors using managed services over unmanaged ones, so consider using AWS Directory Service instead of setting up and managing AD infrastructure manually.

   By grasping the concepts and use cases of AWS Directory Service, you'll be better equipped to design and implement directory services solutions in AWS environments.

1. Why Budget?:
    - Cost optimization is an essential part of the AWS Well-Architected Framework.
    - Budgeting using services like Cost Explorer and AWS Budgets allows organizations to gain control over their cloud spending, preventing overspending and unexpected bills.
2. What is Cost Explorer?:
    - Cost Explorer is an easy-to-use tool that provides visualization and analysis of cloud costs in AWS accounts.
    - It allows users to generate custom reports based on various factors, including resource tags, service categories, and account IDs.
    - Users can break down costs bi-monthly, hourly, or more, providing flexibility in analyzing spending patterns.
    - The service includes a built-in forecasting algorithm that estimates spending up to 12 months in advance based on current and potential future usage.
3. Features of Cost Explorer:
    - Time: Specify the time period for cost reports.
    - Filter: Filter costs based on tags, service categories, account IDs, etc.
    - Service: Break down costs based on individual services or service categories.
4. Exam Tips:
    - Watch for answers that involve using Cost Explorer whenever the question pertains to budgeting and controlling spend.
    - Utilize tags effectively for tracking spending, as they are crucial for generating accurate reports and budgets.
    - Understand that Cost Explorer can forecast spending up to 12 months using an internal AWS algorithm.
    - Know that you can view costs on an hourly, monthly, or longer basis, depending on your needs.
    - If you're part of an organization with consolidated billing, the payer account can break down costs per linked account, allowing you to view specific accounts within the organization.

By understanding how to use Cost Explorer effectively and applying best practices for budgeting and cost control, organizations can optimize their cloud spending and ensure efficient resource allocation.

1. What is Cost and Usage Reports (CUR)?:
    - CUR provides the most comprehensive set of cost and usage data available for AWS spending.
    - Users can publish billing reports to an Amazon S3 bucket for centralized collection, making it convenient for integration with third-party spending applications.
    - Reports break down costs by time span (hour, day, month), service, resource, or tags, similar to Cost Explorer.
    - CUR updates reports in S3 buckets at least once a day in CSV format.
2. Use Cases for CUR:
    - Generate reports for organizations, organizational units, or individual accounts, allowing for granular cost analysis.
    - Track savings plans utilization and monitor on-demand capacity reservations.
    - Break down data transfer charges to optimize usage and implement cost-effective solutions like VPC endpoints.
    - Dive deeper into cost allocation tags for resource spending analysis.
3. Enabling CUR:
    - Navigate to the billing dashboard and select Cost and Usage Reports.
    - Specify the report name, content, and delivery options.
    - Configure an S3 bucket for storage and optionally define a path prefix for multiple reports.
    - Choose the time granularity (hourly, daily, monthly) and compression type (GZIP, ZIP, Parquet).
    - Select integrations with Athena, Redshift, or QuickSight for further analysis and visualization.
4. Exam Tips:
    - Understand that CUR offers the most detailed view of AWS spending.
    - Know that CUR integrates with S3 for centralized storage and allows for flexible report generation.
    - Recognize the flexibility of using CUR at different levels within an organization.
    - Remember the integrations with Athena, Redshift, and QuickSight for analysis and visualization.
    - Be aware that CUR updates reports at least once a day automatically.
    - Any mention of detailed cost breakdowns, daily usage reports, or tracking savings plans may involve CUR in the answer.

    By leveraging CUR effectively, organizations can gain insights into their AWS spending patterns, identify areas for optimization, and ensure efficient resource allocation

1. AWS Compute Optimizer:
    - Analyzes configuration and utilization metrics of resources to provide optimization recommendations.
    - Supports resources like Amazon EC2, EC2 Auto Scaling Groups, Amazon EBS, and AWS Lambda functions.
    - Can be used in standalone accounts, member accounts within an organization, or management accounts within AWS Organizations.
    - Offers graphical history data and projected utilization metrics for informed decision-

making.
    - Enhanced recommendations, including infrastructure metrics, are available for more accurate insights.
2. Savings Plans:
    - Offer flexible pricing models to provide savings on AWS usage, with potential savings up to 72% on compute.
    - Compute Savings Plans provide lower prices on EC2 instances, AWS Lambda, and Fargate usage.
    - EC2 Instance Savings Plans offer higher savings but are more specific, applying only to specific EC2 instance families and regions.
    - SageMaker Savings Plans apply to SageMaker instances, offering savings regardless of instance family and size.
    - Savings Plans require commitments of 1 or 3 years, with payment options including all upfront, partial upfront, or no upfront payment.
    - Recommendations for Savings Plans are automatically calculated in the AWS billing console, making purchasing decisions easier.
3. Exam Tips:
    - Compute Optimizer provides recommendations based on utilization and configuration metrics, supporting various AWS resources.
    - Savings Plans offer flexible pricing options for compute usage, including Compute Savings Plans, EC2 Instance Savings Plans, and SageMaker Savings Plans.
    - Savings Plans require commitments of 1 or 3 years with different payment options, and recommendations can be viewed and purchased directly in the AWS billing console.
    - Understand the differences between the types of Savings Plans and their respective savings potentials.
    - Consider how Savings Plans are applied within your AWS account structure, including consolidated billing scenarios.

   By leveraging AWS Compute Optimizer and Savings Plans effectively, organizations can optimize their compute spend, improve resource utilization, and achieve significant cost savings.

1. AWS Trusted Advisor Overview:
    - Trusted Advisor is a fully managed auditing tool that checks AWS accounts against industry and customer-established best practices.
    - It provides recommendations to help save money, improve availability, and enhance security.
    - The service operates at an account level and requires no additional configuration beyond opting in.
    - Different support levels offer varying degrees of access to Trusted Advisor checks, with enterprise support granting full access and integration with Amazon EventBridge.
2. Trusted Advisor Categories:
    - Cost Optimization: Recommendations for saving money, such as optimizing instance sizes and managing unused resources.
    - Performance: Checks for improving the speed, efficiency, and responsiveness of resources.
    - Security: Recommendations to enhance account security by adjusting settings and configurations.
    - Fault Tolerance: Suggestions for increasing resiliency and high availability of architectures.
    - Service Limits: Checks the usage of accounts against service quotas for different AWS resources.
3. Exam Tips:
    - Understand the purpose and capabilities of Trusted Advisor, including the

categories it covers.
- Know the differences between support levels and the access to Trusted Advisor checks they provide.
- Familiarize yourself with the types of recommendations offered in each category.
- Remember that Trusted Advisor operates at an account level and requires no additional setup.
- Practice navigating the Trusted Advisor console to understand how to view and interpret recommendations.

By leveraging AWS Trusted Advisor effectively, organizations can ensure that their AWS accounts adhere to best practices, leading to improved efficiency, cost savings, and security posture.

1. AWS Control Tower Overview:
   - Control Tower automates account creation and helps implement security controls using AWS Organizations, IAM, and AWS Config.
   - It offers a quick and standardized way to create and manage secure, compliant multi-account environments based on best practices.
   - Control Tower operates as an extension of AWS Organizations, providing governance features for multi-account environments.
2. Features and Terms:
   - Landing Zone: A well-architected, multi-account environment based on compliance and security best practices.
   - Guardrails: High-level rules that provide continuous governance for AWS environments, including preventative and detective guardrails.
   - Account Factory: A configurable account template for standardizing pre-approved configurations for new AWS accounts.
   - CloudFormation StackSet: Automated deployments of templates to deploy repeated resources for governance.
3. Guardrails:
   - Preventative Guardrails: Enforce governance by disallowing violating actions using service control policies.
   - Detective Guardrails: Detect and alert on non-compliant resources using AWS Config rules.
   - Guardrail statuses include enforced, not enabled, clear, or in violation.
4. Example Diagram:
   - Illustrated a diagram showing the structure of Control Tower, including management, log archive, and audit accounts, as well as member accounts within organizational units.
   - Demonstrated how Control Tower deploys preventative and detective guardrails, sets up CloudTrail logging, and aggregates logs and notifications.
5. Exam Tips:
   - Understand the purpose and capabilities of AWS Control Tower, including its role in automating multi-account governance.
   - Remember key terms such as landing zone, guardrails, account factory, and shared accounts.
   - Differentiate between preventative and detective guardrails and know how they operate.
   - Be familiar with scenarios involving automated account setup, guardrails enforcement, and governance of user account provisioning.

   By leveraging AWS Control Tower, organizations can efficiently manage and govern their AWS environments while ensuring compliance with security and regulatory requirements.

1. AWS License Manager Overview:
    - AWS License Manager simplifies license management for various vendors such as Microsoft, SAP, and Oracle.
    - It centralizes license management across AWS accounts and on-premises environments.
    - The service allows setting usage limits, providing control and visibility into license usage to reduce overages and penalties.
2. Exam Tips:
    - Keep in mind that License Manager is focused solely on managing software licenses, not on deploying services or infrastructure.
    - Look for scenarios involving AWS-hosted license management, hybrid environment license management, or preventing license abuse.

   By leveraging AWS License Manager, organizations can effectively track and manage their software licenses, ensuring compliance and avoiding unnecessary costs
   >

1. AWS Health Overview:
    - AWS Health provides visibility into resource performance and availability, helping users understand how health events affect their services, resources, and accounts.
    - It offers near instant delivery of notifications and alerts, enabling quick troubleshooting and preventive actions.
    - Automation can be implemented using Amazon EventBridge based on incoming events, allowing for rapid response to health events.
2. AWS Health Concepts:
    - AWS Health events include account-specific and public events, which report on issues affecting specific accounts or public AWS services.
    - The Health Dashboard displays account and public events, along with service health information.
    - Events have attributes such as event type code, category, status, and affected entities, providing detailed information about the event.
3. Health Event Examples:
    - Examples of health events include EC2 system reboot maintenance scheduled, EC2 operational issues, and billing suspension notices, among many others.
4. Exam Tips:
    - Understand that AWS Health provides visibility and alerts for AWS resource health and can be leveraged for automation via Amazon EventBridge.
    - Be familiar with the types of health events, including account-specific and public events, and know how to interpret their attributes.
    - Look out for scenarios involving checking alerts for service health and automating responses to health events, such as instance reboots for maintenance.

   By utilizing AWS Health, organizations can proactively monitor the health of their AWS resources and accounts, enabling faster response to issues and better overall system reliability.

1. AWS Service Catalog:
    - Allows organizations to create and manage catalogs of preapproved IT services for deployment in AWS.

- Offers multipurpose catalogs that can include various resources such as virtual machine images, servers, software, databases, and preconfigured components for applications.
- Provides centralized management of IT services using AWS Organizations to maintain consistency and governance across the organization.
- Enables end users to easily deploy preapproved services through self-service capabilities, utilizing CloudFormation templates for provisioning.

2. Benefits of AWS Service Catalog:
   - Standardization: Helps standardize deployments by restricting launching products to preapproved solutions.
   - Self-service capabilities: Empowers end users to browse and deploy approved services on their own, reducing operational overhead.
   - Fine-grained access control: Allows adding constraints and resource tags for access control using AWS IAM.
   - Versioning: Supports updating catalogs to newer versions and automatically propagating changes to end users.
3. AWS Proton:
   - Automates Infrastructure as Code provisioning and deployment processes for serverless and container-based applications.
   - Defines standardized infrastructure using templates and manages application stacks containing all components.
   - Automatically provisions resources, configures CI/CD pipelines, and deploys code for applications.
   - Supports AWS CloudFormation and Terraform providers for Infrastructure as Code.
4. Exam Tips and Takeaways:
   - AWS Service Catalog provides preapproved services as CloudFormation templates, enabling end users to deploy approved services independently.
   - AWS Proton offers full Infrastructure as Code provisioning and deployment for serverless and container-based architectures, empowering developers with self-service capabilities.
   - Remember the benefits of each service and their respective roles in standardizing deployments and managing infrastructure in AWS environments.

   Understanding and leveraging AWS Service Catalog and AWS Proton can help organizations streamline deployments, maintain governance, and empower end users with self-service capabilities.

1. Well-Architected Framework Review:
   - The Well-Architected Framework consists of six pillars: Operational Excellence, Reliability, Security, Performance Efficiency, Cost Optimization, and Sustainability.
   - It's crucial to understand each pillar well as it forms the basis for evaluating cloud architectures against best practices.
2. AWS Well-Architected Tool:
   - The AWS Well-Architected Tool is a service that provides a consistent process for measuring cloud architectures against AWS best practices.
   - It assists in documenting workloads and architecture choices, providing guides for improving reliability, security, efficiency, and cost-effectiveness.
   - The tool evaluates workloads against years of AWS best practices collected from solutions architects and various businesses using cloud architectures.
   - It caters to specific audiences such as technical teams, CTOs, architecture teams, and operations teams, aiming to provide comprehensive insights into architecture assessments.
3. Exam Tips and Takeaways:
   - Remember that the AWS Well-Architected Tool is used to measure current

workloads against established AWS best practices.
- Understand that the best practices embedded in the tool are accumulated from years of experience from AWS solutions architects and successful cloud-based businesses.
- Recognize that the tool aids in documenting architectural decisions, helping users understand why certain decisions were made and their implications.

Mastering the Well-Architected Framework and utilizing the AWS Well-Architected Tool can significantly enhance the effectiveness and efficiency of cloud architectures.

1. Centralization and Standardization:
   - Ask if processes can be centralized and how to standardize them using AWS services like AWS RAM, Control Tower, and AWS Service Catalog.
2. Enforcement of Standards:
   - Utilize service control policies (SCPs), AWS Config rules, guardrails in Control Tower, and CloudTrail for auditing to enforce standards effectively.
3. Authentication and User Management:
   - Utilize AWS Single Sign-On (SSO) or Cognito for internal and external user management respectively. Consider Active Directory integration using AWS Directory Service.
4. Cross-Account Roles:
   - Leverage cross-account roles for access management as it's superior to creating IAM users for external access.
5. Cost Management:
   - Track costs using tags, Cost Explorer, and AWS Budgets. Automate proactive alerts and responses to optimize spending. Use AWS Compute Optimizer to right-size compute instances.
6. Trusted Advisor:
   - Leverage Trusted Advisor for auditing and reporting purposes, but remember it's not a problem-solving tool. Automate responses using EventBridge and Lambda.
7. Accounts and Licenses:
   - Utilize AWS Control Tower for governance and compliance in multi-account environments. Leverage AWS License Manager for simplifying license management across AWS and on-premises environments.
8. Infrastructure and Deployments:
   - Use AWS Service Catalog for provisioning pre-approved products and services via catalog portfolios written in CloudFormation. AWS Proton automates the provisioning of entire application stacks for container-based or serverless architectures.
9. Well-Architected Tool:
   - Document architectural decisions and measure them against established industry AWS best practices using the Well-Architected Tool.
10. AWS Health:
    - AWS Health provides notifications of both public and account-specific events within AWS. Automate event responses using EventBridge and AWS Lambda.

Remember, automation is key to efficient governance and operations in AWS environments. Understanding and applying these exam tips will help you excel in governance-related questions on the exam.

From <https://chat.openai.com/c/d6fe90a1-a00b-4374-9bfc-54c5891880f3>