AWS MULTIPLE ACCOUNT SECURITY STRATEGY *"How do I manage multiple AWS accounts for security purposes?"*

Overview

Amazon Web Services (AWS) is designed to enable customers to achieve huge gains in productivity, innovation, and cost reduction when they move to the AWS cloud. AWS offers a variety of services and features that allow for flexible control of cloud computing resources and also of the AWS account(s) managing those resources. On the account level, these options are designed to help provide proper cost allocation, agility, and security, however customers are sometimes unsure of how to best implement an account strategy—especially when working with multiple AWS accounts. This document provides customers with account-level considerations, best practices, and high-level strategic guidance to help structure and manage multiple AWS accounts for security purposes. For information about organizing multiple accounts for billing purposes, see the AWS Multiple Account Billing Strategy Solution Brief.

The following sections assume basic knowledge of AWS accounts, AWS Identity and Access Management (IAM), Amazon Simple Storage Service (Amazon S3), AWS CloudTrail, AWS Organizations, and identity federation.

General Best Practices

Customers leverage AWS services to increase speed and business agility, and so it is common for AWS account structures to change over time. That said, AWS account security is even easier to manage when implemented consistently and uniformly. Therefore, when considering a security strategy for multiple accounts, we recommend that you do not over-engineer your initial account structure in an attempt to create a perfect, immutable set of AWS accounts. Instead, determine the different ways your company will use AWS accounts, and take an iterative approach to structuring and securing them. With adaptability in mind, consider these account security best practices:

- **Clearly define an AWS account-creation process.** Understanding who in your company is creating AWS accounts and what each account will be used for is an essential part of an AWS account security strategy. Many companies create a centralized process for creating AWS accounts; however, centralization is not required as long as the process is well defined and allows a business to maintain an account inventory.
- **Define a company-wide AWS usage policy**. Implement well-defined AWS usage polices and vet them with company stakeholders to align business and security requirements. This empowers customers to leverage AWS without internal confusion about what is or is not within policy guidelines. An AWS usage policy should include a company's minimal security baseline requirements for the different ways they will use AWS, such as which services are approved for use or what security or encryption features must be enabled.
- **Create a security account structure for managing multiple accounts.** Creating a security relationship between accounts makes it even easier for companies to assess the security of AWS-based deployments, centralize security monitoring and management, manage identity and access, and provide audit and compliance monitoring services.
- **Leverage AWS APIs and scripts.** Many customers leverage the AWS APIs and custom developed scripts to automatically and consistently apply baseline configurations across multiple AWS accounts. Consider leveraging compliance-monitoring scripts to provide insight into how well a company's accounts comply with its defined policies and standards for AWS usage.

Implementation Considerations

The following sections offer advice and guidelines to help identify the appropriate security strategy for managing multiple AWS accounts. Keeping an iterative approach in mind, assess your current security policies and leverage an AWS account structure that will facilitate compliance with your IT and security objectives.

When to Create Multiple Accounts

While there is no one-size-fits-all answer for how many AWS accounts a particular customer should have, most companies will want to create more than one AWS account because multiple accounts provide the highest level of resource and security isolation. Answering "yes" to any of the following questions is a good indication that you should consider creating additional AWS accounts:

- *Does the business require administrative isolation between workloads?*
  Administrative isolation by account provides the most straightforward approach for granting independent administrative groups different levels of administrative control over AWS resources based on the workload, development lifecycle, business unit (BU), or data sensitivity.
- *Does the business require limited visibility and discoverability of workloads?*
  Accounts provide a natural boundary for visibility and discoverability. Workloads cannot be accessed or viewed unless an administrator of the account enables access to users managed in another account.
- *Does the business require isolation to minimize blast radius?*
  Blast-radius isolation by account provides a mechanism for limiting the impact of a critical event such as a security breach, if an AWS Region or Availability Zone becomes unavailable, account suspensions, etc. Separate accounts help define boundaries and provide natural blast-radius isolation.
- *Does the business require strong isolation of recovery and/or auditing data?*
  Businesses that are required to control access and visibility to auditing data due to regulatory requirements can isolate their recovery data and/or auditing data in an account separate from where they run their workloads (e.g., writing CloudTrail logs to a different account).

When to Create a Security Account Structure

Most AWS customers with multiple AWS accounts would like to organize them in some sort of security hierarchy. If you answer "yes" to any of the following questions, it is a good indication that you should consider establishing one or more security relationships between AWS accounts:

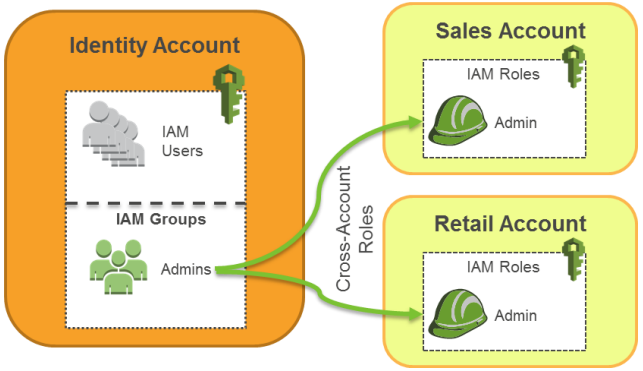
- *Do you want to manage AWS user identities in one account and federate access to other accounts?*
- *Do you want to centrally store, secure, analyze, and report on AWS generated log data from services such as AWS CloudTrail, AWS Config, Amazon S3, Amazon CloudFront, Elastic Load Balancing, or Amazon VPC Flow Logs?*
- *Do you want to empower security and compliance organizations to apply security baselines and monitor security compliance across multiple AWS accounts?*
- *Do you want to centrally manage approved Amazon Elastic Compute Cloud (Amazon EC2), Amazon Machine Images (AMIs), or AWS Service Catalog portfolios and products?*

AWS Security Account Structures

Many companies group AWS accounts into logical structures to better organize and secure their AWS resources. While AWS accounts are not technically hierarchical, you can use organizational units (OUs) with AWS Organizations to create hierarchical and logical account groupings. When creating a new AWS account in an organization, you can use service control policies (SCPs) to filter and restrict the services and actions its users and roles are able to access. All management of the organization is performed centrally in the master account; this includes creating SCPs, creating OUs, and attaching SCPs to OUs (IAM policies are managed on the account level, independently of AWS Organizations). The following security account structures are based on common approaches for creating and securing AWS account groups.

Identity Account Structure

This account structure can be beneficial for customers who want to create and manage all their users in a single account and enable user and group access to resources in other accounts. This model uses IAM cross-account roles to grant access control from one account to another. IAM roles grant temporary access to an AWS account based on the role's IAM policy and trust relationships. IAM cross-account roles establish a trust relationship between AWS accounts, granting predetermined users, groups, or roles in one AWS account permission to perform specific API actions in another AWS account. For example, to establish an identity account structure between IAM users in a parent identity account and other BU accounts, grant cross-account roles to users or groups in the parent account that allow them to manage AWS resources in the necessary BU accounts.

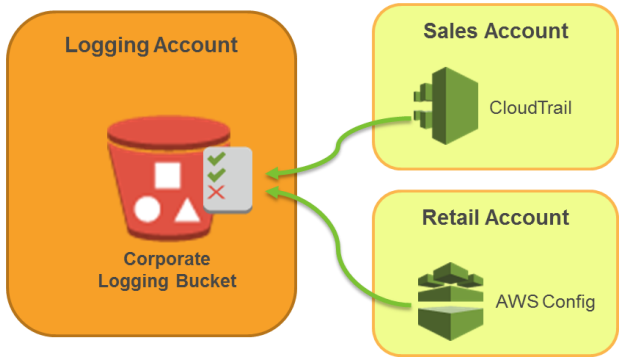


Customers can use IAM to store and manage user identities for an identity account, or they can federate users from a different centralized directory. AWS provides out-of-the-box federation capabilities from IAM (using cross-

account roles), AWS Directory Service,[1] or from existing identity stores using SAML 2.0. Regardless of where user identities are managed, each associated account has IAM roles and permissions that are mapped to users or groups in order to control access. Some commonly used roles in this pattern include granting permissions for read-only access (for compliance or security organizations), IAM administration, or specific AWS service administration. See the AWS website for information about the Cross-Account Manager, which is a prescriptive, AWS-provided solution that uses managed services to automate the configuration of cross-account access in the AWS Management Console. The solution leverages existing Microsoft Active Directory or Simple AD credentials and automatically manages the IAM roles and permissions necessary to give federated users and groups access to multiple AWS accounts.

Logging Account Structure

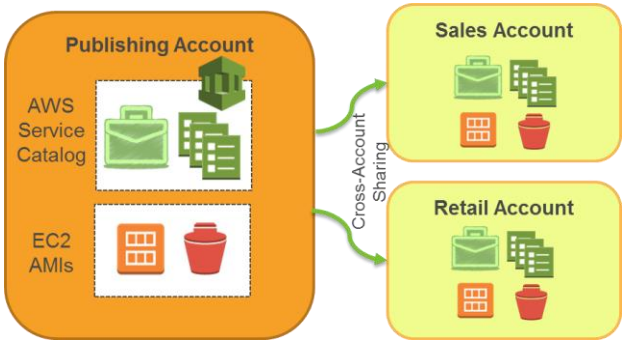
This account structure can be beneficial for customers who want to centrally store, secure, and process AWS log and configuration data. This model configures accounts to send logs and configuration information to a parent logging account, and configures specific service access policies in the parent account to accept the data. For example, to establish an account structure between a parent logging account and other BU accounts, configure AWS CloudTrail or AWS Config in the associated accounts to store API and configuration log data in an S3 bucket that the parent account owns. In an organization, you can use SCPs to restrict member accounts from modifying AWS CloudTrail or AWS Config configuration settings. This design allows the parent account to centrally store, protect (for example, using the S3 MFA Delete or Amazon Glacier Vault Lock features), and analyze sensitive log files independently from each associated account. Other examples include configuring EC2 instances in one account to ship logs to a central account, or configuring CloudWatch Logs subscriptions and AWS Lambda to forward log data from one account to another.



This approach simplifies security monitoring and compliance checking across multiple accounts because it reduces the need to implement a distributed log storage, protection, and analysis solution for each individual account.

Publishing Account Structure

This account structure can be beneficial for customers who want to centrally manage preapproved server images and AWS CloudFormation templates across a company. This model leverages cross-account resource sharing to share AMIs and AWS Service Catalog portfolios created in a parent account with one or more associated accounts. For example, AWS Service Catalog allows IT administrators to create portfolios of products to share with AWS Service Catalog users. These portfolios can, in turn, be shared with other AWS accounts to provide a central repository for service catalog portfolios and products. Likewise, companies can share EC2 AMIs and Amazon Elastic Block Store (Amazon EBS) volumes in order to centrally manage the creation and distribution of approved server images to multiple accounts.



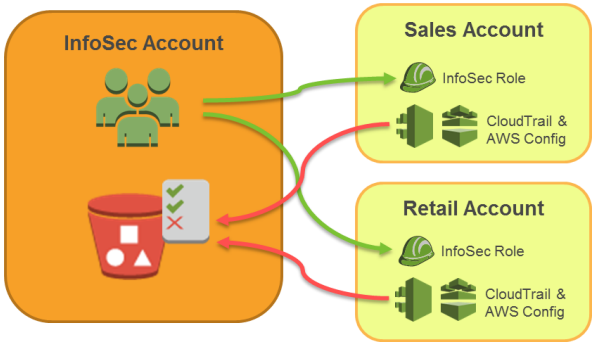
Hybrid AWS Security Account Structures

As evidenced in the previous sections, there are many different ways to establish security relationships between accounts. AWS customers with multiple AWS accounts often leverage consolidated billing, and it can be tempting to use the master billing account for security activities such as centralized IAM administration or storing log files for member accounts. However, most customers eventually choose to create separate security account hierarchies. When determining the appropriate account structure for your security needs, AWS recommends taking an iterative approach because AWS customers rarely require every available option immediately. Some companies might find that a single hierarchical account structure is slow to form, or that separate diagrams are necessary to adequately describe different account relationships. In fact, when separating different account-related concerns (such as billing, security log processing, compliance monitoring, or shared IT services) across multiple accounts, multiple hierarchies are often more desirable than a single hierarchy. Whatever the structure, ensure that account relationships are well documented and well understood. The following examples demonstrate how the various structures listed previously can be combined for Information Security or central IT purposes, and are only a few of the many ways a company can structure and compartmentalize security-related functions.

[1] AWS Directory Service offers a managed service supporting Microsoft Active Directory, Simple AD, and AD Connector for federating access from existing AD environments.

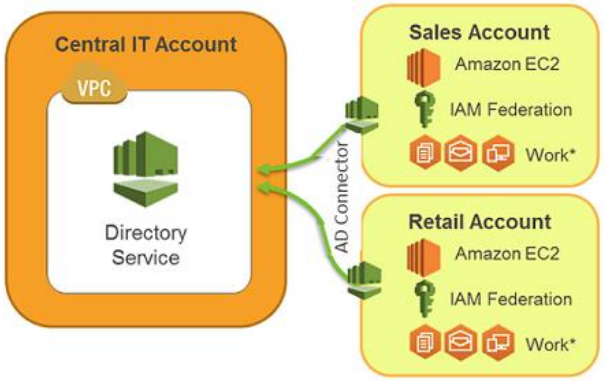
Information Security Account

Dedicated Information Security accounts assume the responsibility for collecting and analyzing security-related data, running compliance scripts, configuring security services (such as CloudTrail and AWS Config), or manage IAM access across other AWS accounts. These accounts are typically owned by an Information Security department that monitors and, in some cases, enforces security compliance through cross-account role access within the multi-account structure. For example, an Information Security account can host an S3 bucket for AWS account logs and can be granted permission to configure AWS logging services in associated BU accounts to send log data to this bucket.

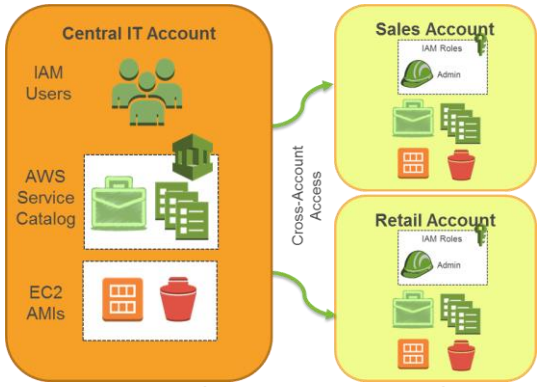


Central IT Account

The responsibilities of central IT accounts vary across companies. They are used to host identity repositories (either a single AWS IAM repository, AWS Directory Service, or a self-managed user repository), federate user access, centrally manage shared AMIs, Amazon EBS snapshots, or AWS Service Catalog portfolios, and provide centralized DNS, logging, configuration management, or software development services. Typically, a central IT department owns these accounts and manages account association and network configuration through cross-account roles. For example, a central IT account can host a user directory to manage user access to EC2 instances, assume IAM roles, and access Amazon WorkDocs, Amazon WorkMail, or Amazon WorkSpaces. Alternatively, a central IT account can contain IAM users and groups who leverage cross-account roles to access associated accounts, as well as centrally managed AMIs and AWS Service Catalog portfolios that are published to other accounts through cross-account permissions.



Central IT Account: Example 1



Central IT Account: Example 2

Resources

AWS Multiple Account Billing Strategy	https://aws.amazon.com/answers/account-management/aws-multi-account-billing-strategy/
AWS Tagging Strategies	https://aws.amazon.com/answers/account-management/aws-tagging-strategies/
AWS Organizations	https://aws.amazon.com/organizations/
Cross-Account Manager	http://docs.aws.amazon.com/solutions/latest/cross-account-manager/welcome.html
AWS Identity and Access Management (IAM)	https://aws.amazon.com/iam/
AWS Directory Service	https://aws.amazon.com/directoryservice/
AWS CloudTrail	https://aws.amazon.com/cloudtrail/
AWS Config	https://aws.amazon.com/config/