How “AWS Hubs” influence Multi-Account Strategies

November 2019

Paul Dunlop

dunlop.geek.nz

APN Ambassador

Notices

You are responsible for making your own independent assessment of the information in this document. This document: (a) is for informational purposes only, (b) represents current AWS product offerings and practices, which are subject to change without notice by AWS, (c) does not create any commitments or assurances from Paul or AWS and its affiliates, suppliers or licensors and (d) is the authors own view on how AWS are implementing hub and spoke architectures. AWS products or services are provided “as is” by AWS without warranties, representations, or conditions of any kind, whether express or implied. The responsibilities and liabilities of AWS to its customers are controlled by AWS agreements, and this document is not part of, nor does it modify, any agreement between AWS and its customers.

Contents

1.0 Introduction

If you have been around in AWS land for a while then you will have seen a bit of change over the years when it comes to accounts, multi-account strategies and AWS services popping up to fix the quagmire they've created with the complexity of cross account management.

This paper seeks to provide a history of the past 5 years of evolution and how AWS has employed the Hub and spoke model for their services and how they tie into a multi-account strategy as a result. It is written from the Authors perspective and is not endorsed by their employer or AWS. Some facts in this paper as a result may be subjectively tainted. The Author wishes to apologize for this and where facts are wrong please feel free to reach out to correct them.

If you are placed to start a fresh AWS tenancy, then this document should help you out to get a feel for where things are at in AWS at the end of 2019.

2.0 AWS Account History

Originally AWS meant for a customer to have 1 AWS account that covered the globe.

However, cloud adoption mostly started in an organisation by one or two people inside a team that saw the value in using AWS for their project.

Over time it became apparent to many organisations that as a result of this, there were in fact many AWS accounts they owned without proper oversight.

Thus, the [disorganised] multi-account enterprise was born.

The other side of the coin around this era was that AWS offered a channel for Resellers to partner with them.

With so many accounts having their own billing relationship with AWS, customers and resellers needed a simplified offering. It was simply too much having tens to hundreds of individual invoices to pay and manage.

AWS responded by creating the Consolidated Billing model, announced Feb 9 2010. This allowed one to designate an AWS account as a Consolidated Billing account. AWS resource accounts could then be invited and linked for the purposes of rolling up the billing to the consolidated billing account. A further benefit of having consolidated roll up was the lowering of your overall costs since the usage calcs apply to the rolled-up values not the individual accounts.

As a result of this enhancement, the basic Multi-Account strategy was born.

This also generated an evolution on account best practice advice. AWS advised that the consolidated billing account should only be used for billing consolidation purposes. No resources should be stood up in this account and only enable the correct personnel access to the billing interfaces as a result of the importance of this segregation.

Thus, circa 2015 the AWS multi account strategy was simple. Keep your billing, identity and resources separated.

Translated this meant every company would have three accounts as its minimum baseline. 1x Billing, 1x Identity and 1x resource.

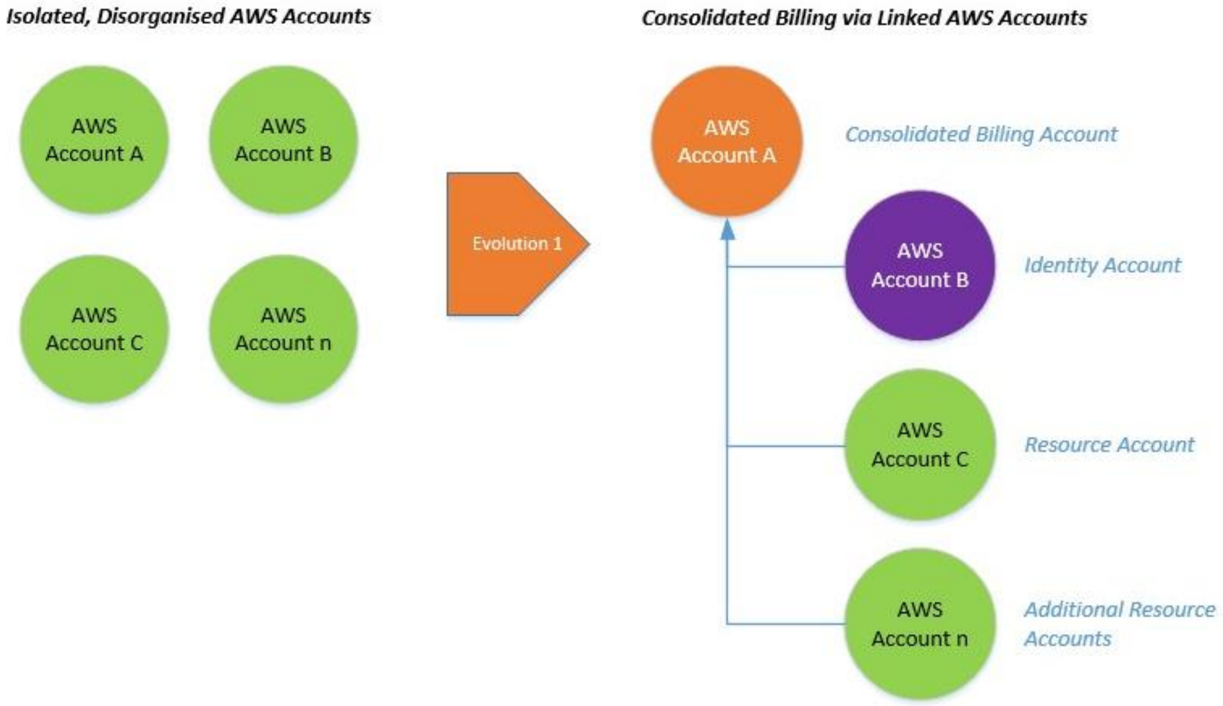


Figure 1: Evolution from Disorganised to Consolidated Billing

This model persisted for a few years until AWS released Organizations on Feb 27 2017 touting:

“With Organizations, you can create groups of accounts and then apply policies to those groups. Organizations enables you to centrally manage policies across multiple accounts, without requiring custom scripts and manual processes.”

When they said “centrally manage policies” at the time they didn’t mean IAM policies, they meant Service Control Policies (SCP). They have since expanded into IAM policies being centrally managed by the Org. The power of a SCP is that it can turn off services in an AWS account which enhances the granular security implementation based on the accounts role. i.e. an AWS account dedicated to networking doesn’t need Redshift data warehouse enabled so you can use an SCP to turn it off etc.

Side note: AWS also released AWS directory services at the same time as it was the founding service that enabled the “grouping” nature of Organisations. Therefore accounts were grouped in the flavor of an Organizational Unit, a place where you would also associate the SCP. Very akin to Active Directory and Group Policies.

On the initial release it was up to customers as to the OU structure and SCP associations but to give an idea the below diagram shows a standard segmentation and policy association. Also, on initial release all consolidated billing accounts became the organisation root account but in order to enable organization features all the child accounts needed to accept the org by logging in as root and doing the acceptance. An interesting issue for the Resellers of the world.

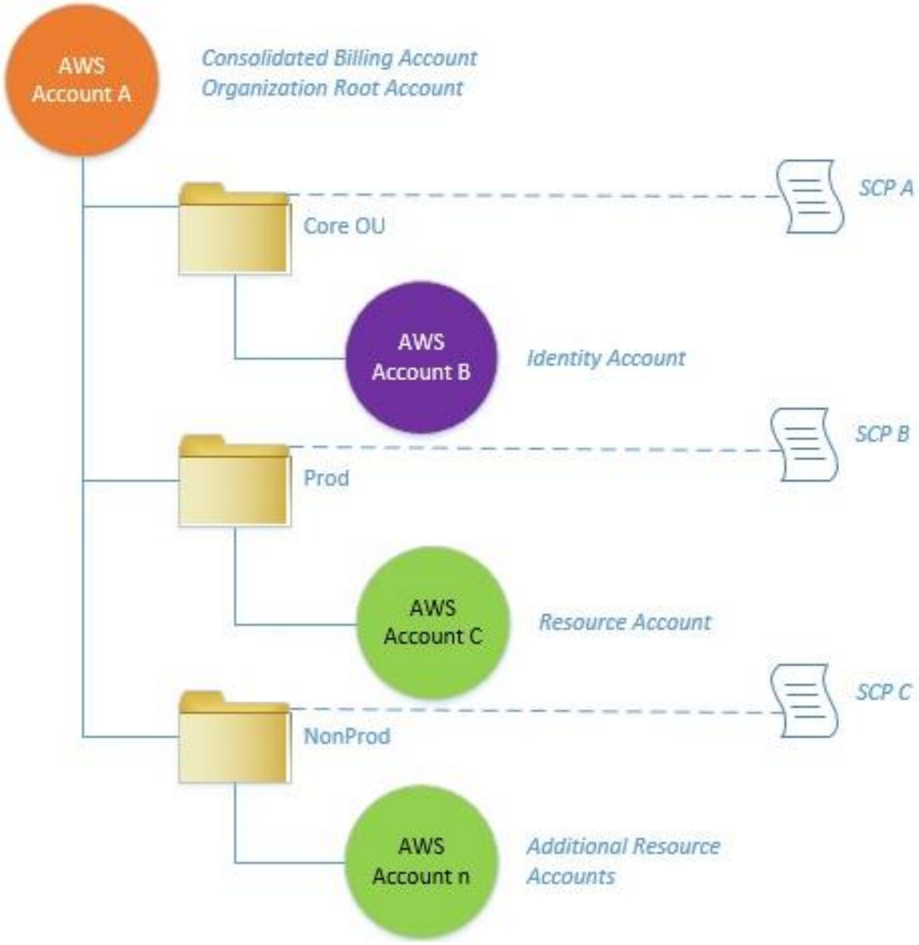


Figure 2: AWS Organizations example

Organizations was seen as becoming the "second" iteration of the centralized hub to control multiple accounts in AWS.

Unfortunately overtime this hasn't been the case. The actual situation is much more complex than this and it is this that this paper seeks to explore and expand upon to allow Architects to strategies with clarity their next iterative journey of their multi-account strategy.

AWS Organizations also released the concepts of:

- The Organization (the central object), has an entity ID and is an umbrella to consolidate all your AWS accounts you want to centrally manage.
- The Root, the starting point for organizing your accounts under this in structures that map to your business.
- The Master account, a special account acting as the paying account (aka consolidated billing) and the organisation management account.

- Member accounts, aka resource accounts

Whether you adopt an environment centric or app centric model for your resources the Hubs we explore in this paper will be a core feature to any model.

3.0 Programmatic Account Creation

The biggest capability that Organizations enabled upon release was the ability to do an API call to “programmatically” create an AWS account.

Suddenly, the multi-account landscape changed in regard to the ease and speed in which AWS accounts could be created.

In July of 2017, only 5 months after Orgs came out AWS released this blog post https://aws.amazon.com/blogs/security/how-to-use-aws-organizations-to-automate-end-to-end-account-creation/.

This became the first publicly scripted AWS capability to not only create an account using a script, but also hydrating it with a baseline by default. At the time a New Zealand based Partner called API Talent engineered and built a YAML based capability that was akin to CloudFormation to create accounts and hydrate them. It was very exciting times. Partners could now say they could automate account creation to a customer’s security standards! A very favorable position to be in.

By 2019 (the year as of writing this) there are several variations of programmatic account creation that have been developed. They are, but are not limited to, the following list:

- The original bash script solution in the security post above
- The original AWS Account Factory script (the term now applies to the Control Tower feature)
- AWS Landing Zone
- AWS Control Tower
- HashiCorp – AWS Terraform Landing Zone

With a notable mention for AWS Deployment Framework as a capable account hydration tool used by many in conjunction with an account creation process as the ADF is triggered to hydrate an account when it is moved into an Org OU.

This paper does not seek to review each one of these, needless to say each of them gives a way to:

- Automate AWS Account Creation
- Move the Account to an OU that associates the desired Service Control Policies & Organizational controls.
- Hydrate the new account with / to a pre-defined baseline

The problem is the cross over of features and use cases between all these tools. It makes it very difficult to determine which ones to use. Does using Control Tower preclude using Organisations? Can the AWS Deployment Framework work with Control Tower account factory provisioned accounts? etc.

It is this aspect this paper seeks to aid you with by breaking down the Hubs that AWS has birthed, where they have come from, what services each hub provides, the accounts they typically land in, and which tools / approaches to best manage them with.

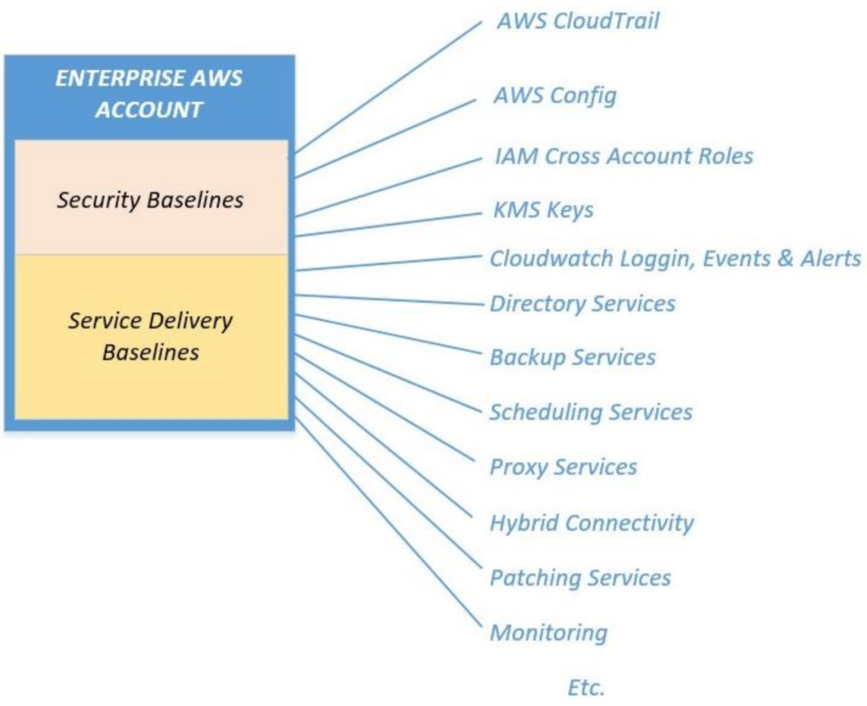
To better enable this outcome for you, a good place to start is the pre-defined baseline each account needs by default, as this is where the Hub conversation starts to get interesting.

4.0 Historical AWS Account Pre-Defined Baselines

Each account in the enterprise must be preconfigured to ensure it abides by the enterprises standards and policies. These determine the “pre-defined baselines” or as some vendors all it the service delivery platform. What is required to secure and operate the account to an acceptable standard. This applies to both hub and spoke accounts, thus making many hub accounts spokes of other hubs.

The pre-defined baseline has been generally broken down into the following historical areas of concern:

1. Core Security services needing to be set up, such as:
   a. AWS CloudTrail
   b. AWS Config & Config Rules
   c. AWS GuardDuty
   d. AWS CloudWatch Event rules
   e. Cross Account Roles & “IAM Service Accounts”
2. Core Operational services needing to be set up, such as:
   a. logging
   b. event delivery
   c. networking
   d. monitoring



It is not desirable to have these set up and managed manually and in isolation per account. Rather, it is desirable to have management and aggregation automated and centralised in some form.

The solutions to these centralisation issues have come in many forms historically with customers often having to tackle these complexities themselves. A non-trivial task that makes customers cry out that “AWS Is Hard!”.

Hence AWS have been tackling these pain points by introducing varying Hubs & solutions (as those reference in section 3).

The irony is that each hub that forms part of the Pre-Defined baseline ends up being a spoke of another hub that is part of the Pre-Defined baseline which forms a new Mesh in AWS to be considered. This is covered in section 6.

Hubs come in varying forms such as actual AWS services (AWS Security Hub), Account “Roles” (AKA Shared Services), scripts or solutions (Ops Automator etc.). The following section seeks to outline these in more detail.

5.0 Hubs

Let’s start with the basics.

What is a hub?

> A hub is a centralised control plane that both manages and aggregate its spokes. The history of which has stemmed as a concept from Airports:
>
> *The words "hub" and "spoke" create a pretty vivid image of how the system works. A hub is a central airport that flights are routed through, and spokes are the routes that planes take out of*

the hub airport. Most major airlines have multiple hubs. They claim that hubs allow them to offer more flights for passengers.

What is the benefit of the hub and spoke system?

The most immediate benefit of hub and spoke model is in:

- One place to manage the configuration of the spoke accounts
- One place to aggregate into for the spoke accounts.

What is the detriment of the hub and spoke system?

- The security aspects for connectivity in the model is highly complex, hence automation is key.
- The **account hierarchy** becomes a **mesh** as there is often a 1 to 1 relationship between a hub and an AWS account. The mesh is depicted further in section 6.0.

The following sections outline what Hubs exist in AWS by the end of 2019, and what account they would typically be hosted from.

5.1 Hub List

The following lists the hubs that you should consider within your account architecture:

Hub	Info
Audit Hub	Originally defined by: The AWS Landing Zone as the Logging account but now the Audit as per Control Tower standards Purpose of this hub: • Security teams BCP access • Centralised logging for core security service logs Services provided by this hub are: • Logging bucket for AWS CloudTrail • Logging bucket for AWS Config • BCP Cross Account Roles for Security into all other accounts Other hubs/services that provide similar services are: • Security Hub
Deployment Hub	Originally defined by: The AWS Deployment Framework Purpose of this hub: • Centralised pipeline management • Centralised resource deployment to all other accounts Services provided by this hub are: • Staged, parallel, multi-account, cross-region deployments of applications or resources via the structure defined in AWS Organizations while taking

	advantage of services such as AWS CodePipeline, AWS CodeBuild and AWS CodeCommit to alleviate the heavy lifting and management compared to a traditional CI/CD setup. Other hubs/services that provide similar services are: • Control Tower / Landing Zone hydrate accounts but don’t manage pipelines centrally.
Directory Services Hub	Originally defined by: Organisations / Landing Zone Purpose of this hub: • Centralised Active Directory Services provided by this hub are: • Managed Active Directory Services • DS for SSO Integration Other hubs/services that provide similar services are: • n/a
Event Hub	Originally defined by: The AWS Event Bridge service. Purpose of this hub: • Centralised eventing via cross account method • Serverless event bus • Connect applications together using data from your own applications, Software-as-a-Service (SaaS) applications, and AWS services. Services provided by this hub are: • Delivers a stream of real-time data from event sources, such as Zendesk, Datadog, or Pagerduty, and routes that data to targets like AWS Lambda. • Routing rules determine where to send data to build application architectures that react in real time to all of your data sources. Other hubs/services that provide similar services are: • None really though this expands on the CloudWatch Event Rule concepts.
Firewall Hub	Originally defined by: AWS Firewall Manager capability. Purpose of this hub: • Establish a centralised Firewall Manager administrator account Services provided by this hub are: • Roll out AWS WAF rules all at once for your Application Load Balancers and Amazon CloudFront distributions across all of the accounts in your AWS organization • Manage prolific Security Groups

Other hubs/services that provide similar services are:

- n/a

Network Hub

Originally defined by:

The AWS Transit Gateway & Landing Zone solution (shared services VPC)

Purpose of this hub:

- Centralise network management
- Centralise hybrid connectivity for:
  - VPNs
  - Direct Connects

Services provided by this hub are:

- Transit Gateway
- VPN
- Direct Connect / Direct Connect Gateway
- Subnet Sharing across accounts via RAM

Other hubs/services that provide similar services are:

- AWS Landing Zone. Had/has the concept of a shared service account and VPC but no transit gateway.

Operations Hub

Originally defined by:

Historical account strategy has had an ops account in the past

Purpose of this hub:

- Centralise the Ops teams functionality

Services provided by this hub are:

- Gold Image (AMI) management / sharing
- Systems Manager Patch Aggregation
- Antivirus

Other hubs/services that provide similar services are:

- Shared services hub in the AWS Landing Zone

Organization Hub

Originally defined by:

The AWS Organizations service.

Purpose of this hub:

- AWS Organizations management hub.
- Control Tower hub.

Services provided by this hub are:

- Consolidated billing
- All the AWS Organization features

Scheduling Hub

Originally defined by:

	The EC2 Scheduler in 2016 but now provided by AWS Instance Scheduler or AWS Systems Manager Purpose of this hub: • Centralised Start / Stop of EC2 instances Services provided by this hub are: • Scheduled start / stop of EC2 instances and RDS databases instance Other offerings that provide similar services are: • AWS Ops Automator
Security Hub	Originally defined by: The AWS Security Hub service. Purpose of this hub: • Hosts AWS Security Hub services Services provided by this hub are: • Aggregates, organizes, and prioritizes your security alerts, or findings, from multiple AWS services, such as Amazon GuardDuty, Amazon Inspector, and Amazon Macie, as well as from AWS Partner solutions. • Findings are visually summarized on integrated dashboards with actionable graphs and tables. • Continuously monitor your environment using automated compliance checks based on the AWS best practices and industry standards your organization follows Other hubs/services that provide similar services are: • n/a
SSO Hub	Originally defined by: The AWS Organizations service. Purpose of this hub: • Centralised single sign on to the Org and its accounts Services provided by this hub are: • AWS SSO • Federation to SaaS third party vendors Other hubs/services that provide similar services are: • Azure AD

5.2 Hubs to AWS Account Mapping

The following table outlines the AWS accounts that typically hosts the Hub:

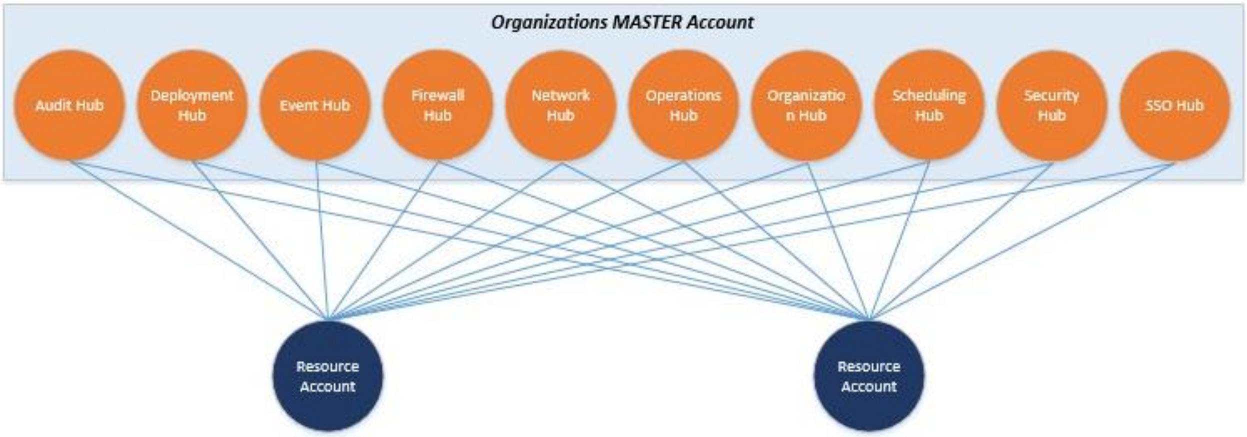
Hub Name	Account Name of Hosting Account

Audit Hub	Audit / Logging
Deployment Hub	Deployment
Directory Services Hub	Shared Services
Event Hub	Shared Services
Firewall Hub	Master
Network Hub	Network
Operations Hub	Shared Services
Organization Hub	Master
Scheduling Hub	Shared Services
Security Hub	Security
SSO Hub	Master

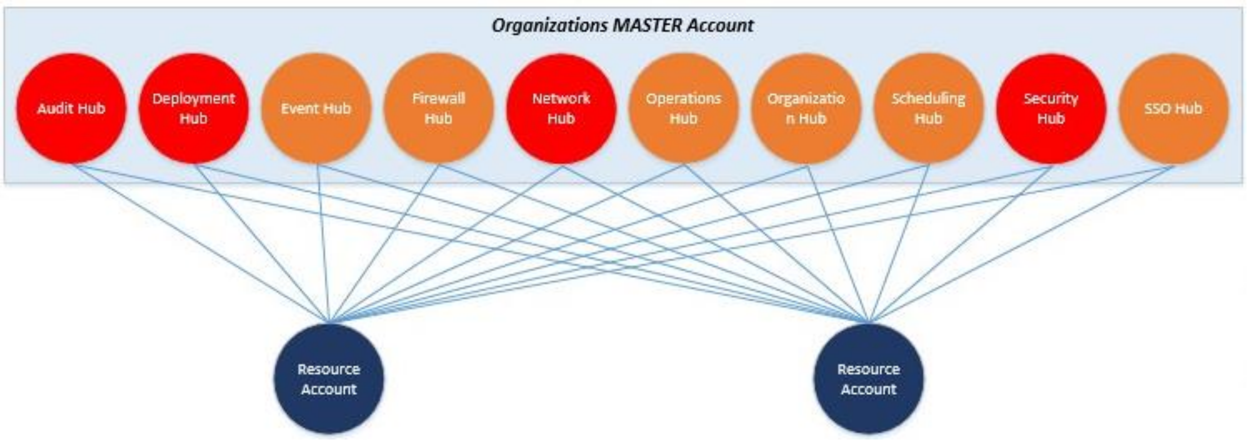
As you can see there is some consolidation of role to account already occurring for the hubs which is a good thing.

6.0 The New Account and Hub Mesh

Its not possible to achieve a single account that acts as all hubs. i.e.

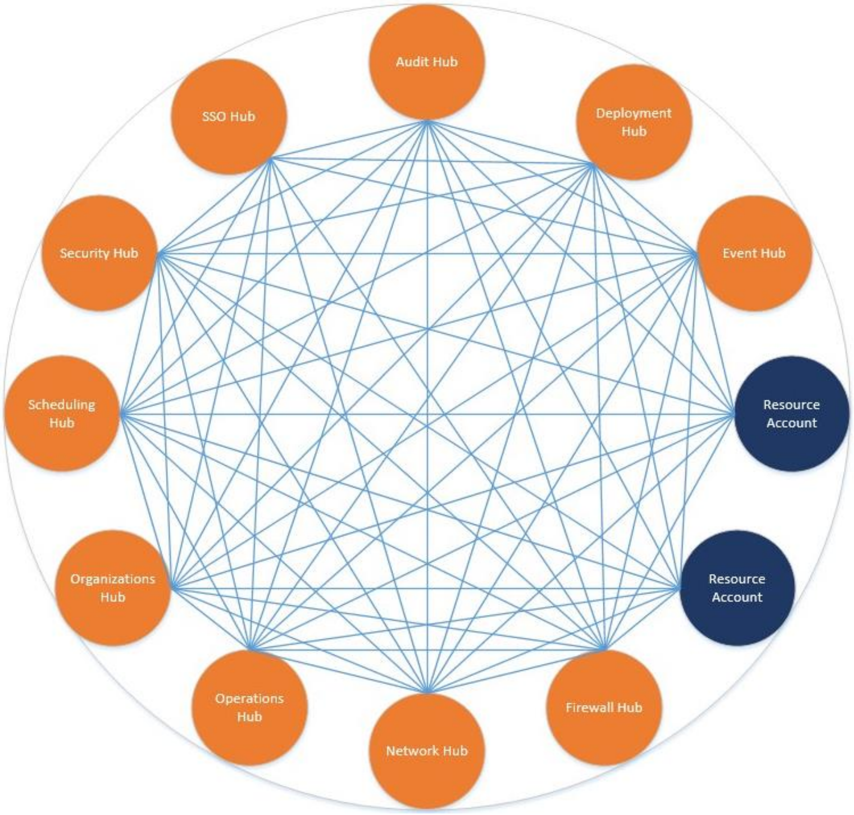


This would be great because cross over account access management is simplified, however it does not provide account level segmentation for audit, network and deployment etc which crucially require it. E.g. the highlighted hubs below get targeted to be hosted in their own accounts.



The danger here is that by segregating an account per hub the model becomes very complex very fast and will cause deployment and security headaches.

The diagram below seeks to show case this complexity around how Hubs are forming a new Mesh in AWS. The mesh complexity used to belong to the VPC Peering world only. It has now entered the Account Strategy.



Whilst Transit Gateway might save the day for simplifying VPC Peering mesh complexities, we still don't have an effective, consolidated, easy to implement / view on how to manage the AWS Account Mesh because AWS Organisations master account hasn't, and is unlikely to ever, achieve this outcome. We only have programmatic account creation and hydration tools and methods.

It is useful to review the varying hubs in AWS and consolidate them into a single account where they share similar roles. For example, the Network hub AWS account could be re-used as the Firewall hub's account. However, if you consider Firewalling a security topic rather than network then the Security hub AWS account could be where you consolidate the two. It should be up to each organisation to consider a: what each hubs solution is and b: how to consolidate them into the account architecture. Ideally, the game plan should be reducing this mesh instead of increasing it, taking a page from the VPC peering mesh and transit gateway response as a historical lesson learnt.

7.0 Recommendations

The following sections outline the recommended baseline for an enterprise wishing to set up a new landing zone, tenancy enterprise ref arch, post 2019. With the rate of innovation inside AWS this is unlikely to remain static however so check back with this whitepaper for new iterations on this scope.

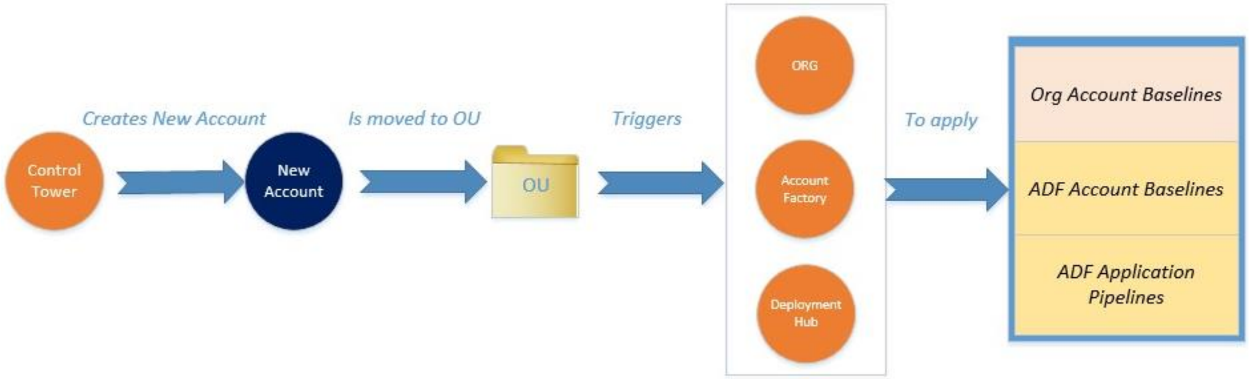
7.1 AWS Services and Solutions to use

It is recommended that a hybrid tool set be employed made up of the following:

- **AWS Organizations** for:
  - Account Billing consolidation
  - Service Control Policies
  - Assigning the Firewall hub to an account
- **AWS Control Tower** to:
  - Automate AWS account creation via the Account Factory
  - Setup guardrails
- **AWS Deployment Framework** in a Deployment account to:
  - Centrally control release management for Application pipelines in spoke accounts
  - Hydrate account baselines & integration into hubs using CI/CD where Control Tower cannot

7.2 Account Creation & Hydration Method

A generalized view of the account creation and hydration process is as follows:



Control tower creates a new account with the account factory which, as part of the process moves the account to an OU. The OU pre-determines the SCPs applied.

The Organisation automatically provisions the CloudTrail and Config log centralisation & aggregation solution within the new account.

The act of putting the account under an OU causes the AWS deployment framework to notice a new account that needs to have the baseline applied for that OU. So a dev account could get a different baseline compared to prod.

Once the account is hydrated to its correct baseline the account becomes available for staff to deploy resources via CloudFormation templates using the AWS Deployment Framework from the deployment hub.

7.3 Hub to Account Model to use

The following table outlines the hubs required and the account model to support them:

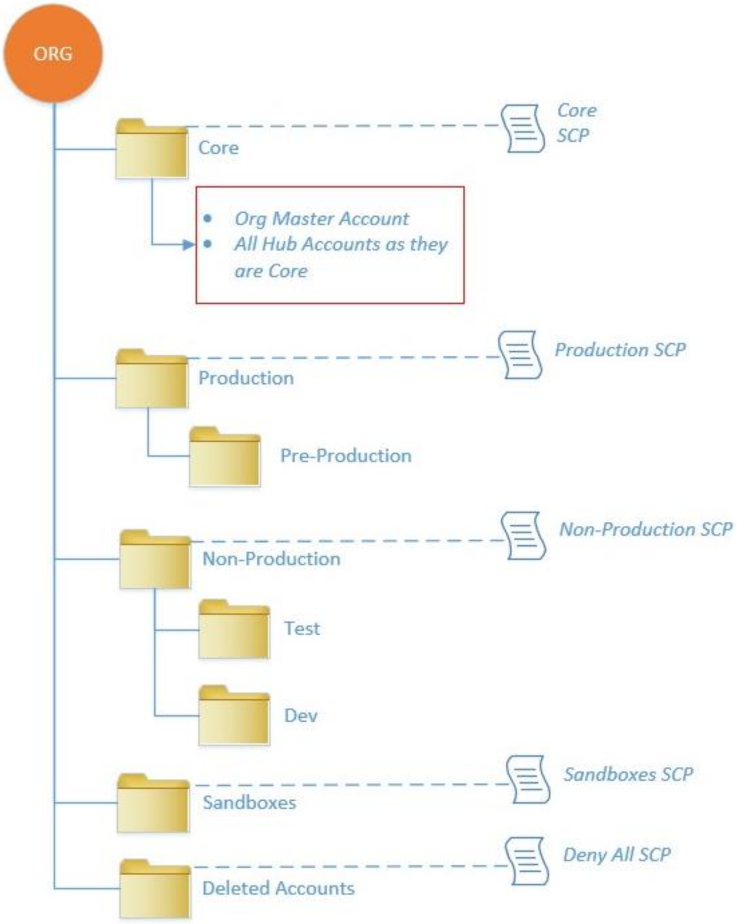
Hub	Account Name	Account created by
Audit Hub	Audit	Control Tower Account Factory
Deployment Hub	Deployment	Control Tower Account Factory
Directory Services Hub	Operations	Control Tower Account Factory
Event Hub	Master	Control Tower Account Factory
Firewall Hub	Network	Control Tower Account Factory
Network Hub	Network	Control Tower Account Factory
Operations Hub	Operations	Control Tower Account Factory
Organization Hub	Master	Manual process
Resource Accounts	[naming convention]	Control Tower Account Factory
Scheduling Hub	Operations	Control Tower Account Factory
Security Hub	Security	Control Tower Account Factory
SSO Hub	Master	Control Tower Account Factory

This still gives an enterprise the flexibility to have their Resources account strategy employed in the varying flavors including sub folders in the OU model as outlined below.

7.4 AWS Organizations OU Model

Taking a view on the data classification being a priority has lent itself to environments being a key place to manage security controls for said data.

The following diagram outlines the AWS Organizational Unit strategy for an enterprise to use considering an environment view.



Notes:

- Pre-Production is for any accounts that are being cloned from production to mirror it for testing. Usually these include production data so holding them to the same set of values as the production account is best practice.
- Sandboxes should be a place where AWS practitioners get to explore and expand on their skill sets. Being overly wide but not completely lax on your security controls here is a good idea.
- Deleted Accounts should be treated like deleted Active Directory users as each account is given a unique email that is set in stone in AWS and cannot be reused. If you ever want to re-use it (the root email address) this is a good way to quickly re-enable a historical account. So don't delete them, rather decommission them with something like aws-nuke and the put them into cold storage under a Deleted OU that has a Deny All SCP policy associated to it.

8.0 Conclusion

AWS provides many hubs and in many forms, which invariably makes each hub become a spoke of another hub as a result.

Several key take ways are:

1. Enterprise architects should work out with Security which Hubs can co-exist in a single AWS account as part of the Account Strategy to minimize the complexity
2. Programmatic AWS Account creation & hydration should be core to the approach moving forward to ensure security baselines are baked into the account
3. Cloud Operations should have a programmatic approach to managing multiple AWS accounts with good:
    a. Cross account role practices
    b. CI/CD considerations in mind

9.0 Contributors

Contributors to this document include:

Paul Dunlop, AWS Cloud Architect & APN Ambassador, dunlop.geek.nz; AWS Partner Network

10.0 Further Reading

For additional information, see:

- https://docs.aws.amazon.com/organizations/latest/userguide/orgs_getting-started_concepts.html
- https://docs.aws.amazon.com/whitepapers/latest/cost-optimization-laying-the-foundation/aws-account-structure.html
- https://aws.amazon.com/answers/account-management/aws-multi-account-billing-strategy/
- https://d0.awsstatic.com/aws-answers/AWS_Multi_Account_Security_Strategy.pdf

11.0 Further Viewing

For additional information, see:

- https://www.youtube.com/watch?time_continue=2&v=pfetMIlo_2s&feature=emb_title
- https://www.youtube.com/watch?v=fxo67UeeN1A

12.0 Document Revisions

Date	**Description**
November 2019	First publication