AWS Landing Zone Upgrade Guide

AWS Upgrade Guide

May 2020

Notice: AWS Landing Zone must be deployed by your AWS Account team or a certified partner to ensure that your account meets the required prerequisites to successfully deploy this solution.

AWS Control Tower is the recommended option for self-service landing zones. For more information please visit AWS Control Tower.

Contents

Overview

This guide will help you upgrade the AWS Landing Zone solution. When the solution is deployed into your AWS account, it deploys two components: AWS Landing Zone Framework and AWS Landing Zone Configuration. When upgrading the solution, the framework component is updated using the AWS CloudFormation update stack option. The configuration component requires you to manually apply the updates and run the AWS CodePipeline.

> **Note:** Every upgrade of the solution is sequential. For example, to upgrade from v1.0 to v3.0, you must first upgrade to v2.1, before you can upgrade to v2.2.

The upgrade options and steps are shown below:

Mandatory

Using the **mandatory** steps will only upgrade the AWS Landing Zone Framework services such as AWS CodePipeline, AWS CodeBuild, AWS Step functions, AWS Lambda functions in the AWS Master account. Note that following these steps will not update your existing Landing Zone configuration and will only require you to apply minor update(s) to the Account Vending Machine template. This option has lowest risk of upgrade.

Highly Recommended

Using the **highly recommended** steps build on top of the mandatory steps, and require you to make updates to your existing Landing Zone configuration files: `manifest.yaml`, AWS CloudFormation templates, and parameters files.

Nice to Have

Using the **nice to have** steps build on top of the mandatory and highly recommended steps, and require you to make updates to your existing Landing Zone Configuration: `manifest.yaml`, templates, and parameters files.

> **Note:** Carefully review each step of the upgrade sections before upgrading the AWS Landing Zone solution.

Answer the following question below to help decide which upgrade option and steps you should follow:

- Since deploying the Landing Zone solution, have you ever changed any file(s) in your Landing Zone Configuration ZIP file (`aws-landing-zone-configuration.zip`)?
- If you answered **No** to the above question, you should follow the Mandatory, Highly Recommended, and Nice to Have steps to upgrade your Landing Zone.
- If you answered **Yes** to the above question, you should at minimum follow the Mandatory steps. Additionally, please review your existing Landing Zone Configuration and the proposed changes in Highly Recommended steps with your AWS Account team to decide if you want to perform those steps to upgrade your Landing Zone.

> **Guidance:** We recommend that the entire upgrade is completed in sandbox/pre-production environment, and roll the upgrade out to production. If you perform a roll-back, you may encounter issues depending when it fails in the upgrade process.

Recommended Maintainance Steps

The AWS Managed AD and Directory Connector for AWS SSO Add-On product deploys an `AWS-Landing-Zone-SharedServicesRDGW` stack set that uses the `AWS::SSM::Parameter` parameter to get the latest AMI ID and uses it in an Auto Scaling group launch configuration. The launch configuration is static and should be periodically updated to use the latest AMI ID. For more information on how to update the AMI ID, see Appendix A.

Upgrade Instructions (v2.3.1 to v2.4.0)

Overview

This version addresses performance and stability in larger deployments. The API calls in the Launch AVM process and optimized to reduce the occurrence of API throttling. The AWS CloudFormation StackSet deployment is updated to improve baseline resource deployment to reduce the occurrence of timeouts.

Release Notes

This release includes following features and bug fixes:

- Enable automatic key rotation for the AWS KMS key: `AwsLandingZoneKMSKey`
- Optimized stack instance deployment workflow - consume 60% less time to deploy same number of stack instances
- Reduced stack set operation fault tolerance to 10 percent
- Optimized LaunchAVM stage to reduce throttling exceptions
- Change IAM Password Policy baseline resource runtime from NodeJS to Python to avoid future NodeJS updates
- Updated all python3.6 runtimes to python 3.8 (and 3.7 for inline lambda functions)
- Added retry mechanism for AWS Organizations APIs
- Updated state machine execution names in LaunchAVM stage to avoid name conflict exception.
- Use Virtual Hosted-Style URLs (path-style URLs will be deprecated in Sept 2020)
- Use regional endpoint for Amazon S3 APIs

Mandatory Upgrade Steps

Performing these upgrade steps only upgrades your Landing Zone Framework and keeps your existing Landing Zone configuration as is.

Use the following procedure to perform the mandatory upgrade:

1. **Backup** your existing Landing Zone Configuration ZIP file (`aws-landing-zone-configuration.zip`) from your Landing Zone Configuration Amazon S3 bucket (i.e. `aws-landing-zone-configuration-<ACCOUNT_ID>-<REGION>`). This version does not require any configuration changes.
2. Navigate to the AWS CloudFormation Console, select the **Landing Zone Initiation Stack**, and select **Upgrade**.

WARNING: Any changes made to the AWS Landing Zone Framework services such as AWS CodePipeline, AWS CodeBuild, AWS Step functions, AWS Lambda functions, and related IAM Roles as part of the initiation template or on the deployed resources, those changes will be overwritten by this step.

3. Use the linked Amazon S3 template URL.
4. Wait for the Update Stack to complete.

Highly Recommended Upgrade Steps

1. **Update** the KMS key policy for **AwsLandingZoneKMSKey** by adding your IAM user/role ARN under the **Allow use of the key** section of the policy.

Note: This is required to grant your own IAM user/role permission to use the KMS key for downloading/uploading the LZ configuration ZIP file.

```
      {
        "Sid": "Allow use of the key",
        "Effect": "Allow",
        "Principal": {
          "AWS": [
            "arn:aws:iam::xxxxxxxxxxxx:role/Admin",
”arn:aws:iam::xxxxxxxxxxxx:role/LandingZoneHandshakeSMLambdaRo
le",
"arn:aws:iam::xxxxxxxxxxxx:role/LandingZoneDeploymentLambdaRol
e",
"arn:aws:iam::xxxxxxxxxxxx:role/StateMachineLambdaRole",
"arn:aws:iam::xxxxxxxxxxxx:role/LandingZoneLambdaRole",
"arn:aws:iam::xxxxxxxxxxxx:role/StateMachineTriggerLambdaRole"
,
"arn:aws:iam::xxxxxxxxxxxx:role/LandingZoneCodePipelineRole"
          ]
        }
```

Update IAM Password Policy Runtime

1. **Create** a temporary directory called LZ_v240 on your local machine in your preferred location. For example: `/temp/LZ_v240`.
2. **Download** and **Unzip** the LZ Configuration ZIP file (`aws-landing-zone-configuration.zip`) from your LZ Configuration S3 bucket (`aws-landing-zone-configuration-<ACCOUNT_ID>-<REGION>`) into the LZ_ v240 directory.
3. **Replace** `./templates/aws_baseline/aws-landing-zone-iam-password-policy.template` with the updated linked template.

   This new template version replaces the Node.js Lambda function with one written in Python. You can view the new source by downloading the Lambda source.
4. **Zip** all the files under the `/temp/LZ_v240` directory to `aws-landing-zone-configuration.zip`.
5. **Upload** the `aws-landing-zone-configuration.zip` to your LZ Configuration Amazon S3 bucket (`aws-landing-zone-configuration-<ACCOUNT_ID>-<REGION>`). This will trigger the LZ CodePipeline.
6. Wait for the LZ CodePipeline to successfully finish.

Upgrade Instructions (v2.3 to v2.3.1)

Overview

This version addresses the deprecation of Node.js 8 AWS Lambda runtime and fixes an issue preventing successful deployment of the AWS Managed AD and Directory Connector for the AWS SSO add-on. The auto-update workflow will **not** update the provisioned products. You have to manually provision or update the existing provisioned add-on products.

Release Notes

This release includes following features and bug fixes:

- Changed the Lambda runtime from Node.js 8 to Nodejs10.x for the IAM Password Policy Lambda.
- Changed the Lambda runtime for the AWS Centralized Logging Solution Add-On to Nodejs12.x.

Bug Fixes

Fixed a bug in the AWS Managed AD and Directory Connector for AWS SSO add-on that prevented successful deployment in the previous v1.2 addon - fails during the Pipeline job execution in the stage Core Resources.

Mandatory Upgrade Steps

Performing these upgrade steps only upgrades your Landing Zone Framework and keeps your existing Landing Zone configuration as is.

Use the following procedure to perform the mandatory upgrade:

5. **Backup** your existing Landing Zone Configuration ZIP file (`aws-landing-zone-configuration.zip`) from your Landing Zone Configuration Amazon S3 bucket (i.e. `aws-landing-zone-configuration-<ACCOUNT_ID>-<REGION>`). This version does not require any configuration changes.
6. Navigate to the AWS CloudFormation Console, select the **Landing Zone Initiation Stack**, and select **Upgrade**.

> **WARNING**: Any changes made to the AWS Landing Zone Framework services such as AWS CodePipeline, AWS CodeBuild, AWS Step functions, AWS Lambda functions, and related IAM Roles as part of the initiation template or on the deployed resources, those changes will be overwritten by this step.

7. Use the linked Amazon S3 template URL.
8. Wait for the Update Stack to complete.

Update IAM Password Policy Runtime

2. **Create** a temporary directory called LZ_v231 on your local machine in your preferred location. For Example: /temp/LZ_v231
3. **Download** and **Unzip** the LZ Configuration ZIP file (`aws-landing-zone-configuration.zip`) from your LZ Configuration S3 bucket (i.e. `aws-landing-zone-configuration-<ACCOUNT_ID>-<REGION>`) into the LZ_ v231 directory.
4. **Replace** `./templates/aws_baseline/aws-landing-zone-iam-password-policy.template` with the updated linked template.

   The sample below depicts the changes to introduced in this release:

```
...  ...  @@ -139,7 +139,7 @@
139  139              !Sub |
140  140                'use strict';
141  141                const AWS = require('aws-sdk');
142       -            const response = require('cfn-response');
     142  +            const response = require('./cfn-response');
143  143              const iam = new AWS.IAM({apiVersion: '2010-05-08'});
144  144              exports.handler = (event, context, cb) => {
145  145                console.log(`Invoke: ${!JSON.stringify(event)}`);
...  ...  @@ -184,7 +184,7 @@
184  184          Handler: 'index.handler'
185  185          MemorySize: 128
186  186          Role: !GetAtt 'LambdaRole.Arn'
187       -        Runtime: 'nodejs8.10'
     187  +        Runtime: 'nodejs10.x'
188  188          Timeout: 60
189  189      LambdaLogGroup:
190  190        Type: 'AWS::Logs::LogGroup'
...  ...
```

5. **Zip** all the files under the `/temp/LZ_v231` directory as an `aws-landing-zone-configuration.zip` file.
6. **Upload** the `aws-landing-zone-configuration.zip` to your LZ Configuration Amazon S3 bucket (`aws-landing-zone-configuration-`*`<ACCOUNT_ID>`*`-`*`<REGION>`*). This will trigger the LZ CodePipeline.
7. Wait for the LZ CodePipeline to successfully finish.

Update AWS Managed AD and Directory Connector for AWS SSO

This step is only required if you are using the **AWS Managed AD and Directory Connector for AWS SSO** addon.

1. In the primary account, open **Service Catalog**.
2. Select the AWS Managed AD and Directory Connector for AWS SSO from the list of Provisioned Products.
3. Select **Update** from the **ACTIONS** drop-down.

> **Note:** v1.2 is the only version available. The previous version can no longer be deployed and so has been removed.

4. Select **v1.2** and choose **NEXT.**
5. Verify the parameters, and choose **next**. You shouldn't have any changes.

6. On the final confirmation page verify the details and choose **UPDATE**.
7. Wait for the LZ CodePipeline to successfully finish.

Update AWS Centralized Logging Solution Add-On

This step is only required if you are using the **AWS Centralized Logging Solution** add-on.

1. In the primary account, open **Service Catalog**.
2. Select the AWS Centralized Logging Solution from the list of Provisioned Products.
3. Select **Update** from the ACTIONS drop-down.

> **Note:** v1.2 is the only version available. The previous version can no longer be deployed and so has been removed.

4. Select **v1.2** and choose NEXT.
5. Verify the parameters, and choose **next**. You shouldn't have any changes.
6. On the final confirmation page verify the details and choose UPDATE.
7. Wait for the LZ CodePipeline to successfully finish.

Upgrade Instructions (v2.2 to v2.3)

Overview

In this version we added the capability to Auto-update Add-On Portfolio(s) and product(s). When setting up an AWS Landing Zone, customers can choose how they would like their Service Catalog Add-On portfolio(s) and/or product(s) to be updated. The scope of the auto-update is limited to the addition of a new portfolio, addition of new products, and addition of new versions for existing products. Note that the auto-update workflow will **not** update the provisioned products. You have to manually provision or update the existing provisioned add-on products. This allows you to receive an email notification when new or new versions of Add-On resources are available in the Service Catalog console without updating the solution template manually.

You have the option to opt-out of the auto-update functionality changing the **AWS Manages Service Catalog Add-On Portfolio** template parameter to `Manual Updates`. If you choose to opt out of this functionality, we recommend subscribing to the RSS Feed to learn about future add-on releases.

Release Notes

This release includes following features and bug fixes:

- Added an AWS Lambda function to publish new Service Catalog Add-On portfolio(s) or product(s).
- Added an Amazon CloudWatch Event to invoke auto-update workflow every day.
- Added parameter to initiation template to allow customers to choose either auto update the Add-On resources or manually update the template.
- Added parameter to specify a notification email to notify customers once auto-update workflow finish.

Bug Fixes

- Fixed error handling of intermittent issue: during new account creation an exception is thrown if STS service has not been enabled due to account initialization. The bug fix will force a retry after 5 minutes.
- Handled Scaling Issue with Service Catalog API (search_provisioned_product). The API response does not return all the provisioned products in the response pages if there are more than 100 provisioned products. Added `sortBy` key in the API to restore this behavior.

Mandatory Upgrade Steps

Performing these upgrade steps only upgrades your Landing Zone Framework and keeps your existing Landing Zone configuration as is.

Use the following procedure to perform the mandatory upgrade:

1. **Backup** your existing Landing Zone Configuration ZIP file (`aws-landing-zone-configuration.zip`) from your Landing Zone Configuration Amazon S3 bucket (`aws-landing-zone-configuration-`*`<ACCOUNT_ID>`*`-`*`<REGION>`*). This version does not require any configuration changes.

9. **Navigate** to the AWS CloudFormation Console, select the Landing Zone Initiation Stack, and select **Upgrade**.

> **WARNING**: Any changes made to the AWS Landing Zone Framework services such as AWS CodePipeline, AWS CodeBuild, AWS Step functions, AWS Lambda functions, and related IAM Roles as part of the initiation template or on the deployed resources, those changes will be overwritten by this step.

10. Use the linked Amazon S3 template.
11. **Navigate** to the bottom of the input parameters and find the **AWS Manages Service Catalog Add-On Portfolio** template parameter to either enable or disable the auto-update workflow. Select **AWS Managed** to enable or **Manually Update** to disable the auto-update workflow.

12. If you selected **AWS Managed**, enter an email address for the template parameter **Add-On Updates Notification Email** to be notified once the auto-update workflow completes.

> **Note**: If you choose manual updates then you can wait to be notified by the RSS feed and then manually update using the latest initiation template linked above.
>
> These changes will not update the existing LZ configuration ZIP file.

13. Wait for the Update Stack to complete.

Upgrade Instructions (v2.1 to v2.2)

Release Notes

Release includes following features and bug fixes:

- RSS Feed Notifications
- Deleted stack instances from the Baseline Resource StackSet if the regions are removed from the manifest file (Deletion Mechanism for Baseline Resources)
- Added retry mechanism in LaunchAVM State Machine to handle exceptions during provisioned product update.
- Added metadata and/or updated template with reduced permissions per CFN-Nag warnings.
- Added retain policy to protect the resources from deletion:
  - VPC Resources (used in AVM and SharedServices VPC)
  - Active Directory Resource (Add-On)
- Updated SCP Policy (preventive guardrail) to protect resources managed by AWS Landing Zone (in sync with AWS Control Tower)
- Added solution prefix to the resources created by the solution (in sync with AWS Control Tower)

Performance Enhancements

- Parallel LaunchAVM State Machine executions - deploy/update batch of accounts per execution. Batch size is configurable by the customer.
- Updated LaunchAVM State Machine - added retry with back-off algorithm to handle API failure
- Added retry mechanism with back-off algorithm for known APIs (describe*, list*, and get*)
- Added support for Boolean and none-type parameters values in Add-On Products

- Automatically remove unnecessary white spaces in a SCP policy to handle SCP (service) limits.

Bug Fixes

- LaunchAVM State - Handle more than 20 accounts in the same the same organization unit
- LaunchAVM State - Handle error: maximum (25,000) entries in the execution history
- Store SCP policies in an Amazon S3 bucket instead of SSM Parameter Store to avoid max value limit for SSM Value
- Added retry logic to get GuardDuty master in the Handshake State Machine
- Handle undefined max password age in the CFN parameters of IAM Password Policy Template.
- CoreAccounts Stage (CodePipeline - first run) fails in an account without Organization

Mandatory Upgrade Steps

Performing these upgrade steps only upgrades your Landing Zone Framework and keeps your existing Landing Zone configuration as is.

Use the following procedure to perform the mandatory upgrade:

1. **Backup** your existing Landing Zone Configuration ZIP file (`aws-landing-zone-configuration.zip`) from your Landing Zone Configuration Amazon S3 bucket (`aws-landing-zone-configuration-<ACCOUNT_ID>-<REGION>`)
2. Navigate to the AWS CloudFormation Console, select the Landing Zone Initiation Stack, and select **Upgrade**.

> **WARNING**: Any changes made to the AWS Landing Zone Framework services such as AWS CodePipeline, AWS CodeBuild, AWS Step functions, AWS Lambda functions, and related IAM Roles as part of the initiation template or on the deployed resources, those changes will be overwritten by this step.

3. Use the linked Amazon S3 template URL.

> **Note**: Do not change any input parameters, it will not update the existing LZ configuration ZIP file.

4. Wait for the Update Stack to complete.

Highly Recommended Upgrade Steps

1. **Update** the KMS key policy for **AwsLandingZoneKMSKey** by adding your IAM user/role ARN under the **Allow use of the key** section of the policy.

Note: This is required to grant your own IAM user/role permission to use the KMS key for downloading/uploading the LZ configuration ZIP file.

```
      {
        "Sid": "Allow use of the key",
        "Effect": "Allow",
        "Principal": {
          "AWS": [
            "arn:aws:iam::xxxxxxxxxxxx:role/Admin",
”arn:aws:iam::xxxxxxxxxxxx:role/LandingZoneHandshakeSMLambdaRo
le",
"arn:aws:iam::xxxxxxxxxxxx:role/LandingZoneDeploymentLambdaRol
e",
"arn:aws:iam::xxxxxxxxxxxx:role/StateMachineLambdaRole",
"arn:aws:iam::xxxxxxxxxxxx:role/LandingZoneLambdaRole",
"arn:aws:iam::xxxxxxxxxxxx:role/StateMachineTriggerLambdaRole"
,
"arn:aws:iam::xxxxxxxxxxxx:role/LandingZoneCodePipelineRole"
          ]
        }
```

14. **Create** a temporary directory called LZ_**v22** on your local machine in your preferred location. For Example: /temp/**LZ_v22**
15. **Download** and **Unzip** the **LZ** Configuration ZIP file (`aws-landing-zone-configuration.zip`) from your LZ Configuration S3 bucket (`aws-landing-zone-configuration-<ACCOUNT_ID>-<REGION>`) into the **LZ_ v22** directory.
16. **Download** and **Unzip** the LZ_ v22 Sample Configuration Zip file and place into **LZ_v22_sample** directory. For Example: /temp/**LZ_v22_sample** directory
17. **Copy** the SCP policies from the sample configuration zip downloaded in the previous step, into the policies directory of your LZ configuration folder.
18. Open `manifest.yaml` and make the following code update:
    a. Update the `organization_policies` with the new file names:
       - `aws-landing-zone-core-mandatory-preventive-guardrails.json`
       - `aws-landing-zone-non-core-mandatory-preventive-guardrails.json`

```
# Landing Zone Service Control Policies
organization_policies:
  - name: protect-cloudtrail-config
    description: To prevent from deleting or disabling CloudTrail and Config
    policy_file: policies/prevent_deleting_cloudtrails_config.json
  - name: aws-landing-zone-core-mandatory-preventive-guardrails
    description: To prevent from deleting or disabling resources in core accounts managed by AWS Landing Zone
    policy_file: policies/aws-landing-zone-core-mandatory-preventive-guardrails.json
    #Apply to accounts in the following OU(s)
    apply_to_accounts_in_ou:
      - core
  - name: aws-landing-zone-non-core-mandatory-preventive-guardrails
    description: To prevent from deleting or disabling resources in non-core accounts managed by AWS Landing Zone
    policy_file: policies/aws-landing-zone-non-core-mandatory-preventive-guardrails.json
    #Apply to accounts in the following OU(s)
    apply_to_accounts_in_ou:
      - applications
```

19. Navigate to the SCP policies console and detach all the OUs from the `protect-cloudtrail-config` policy.
20. Delete the `protect-cloudtrail-config` policy via the console.
21. If you have not customized these files, replace it with the new files from the sample configuration directory. If you have previously customized these files, open following two files and make the following code update:

a. parameters/aws_baseline/aws-landing-zone-primary-vpc.json

b. parameters/core_accounts/aws-landing-zone-shared-services-vpc.json

```
  },
  {
    "ParameterKey": "ManagedResourcePrefix",
    "ParameterValue": "aws-landing-zone"
  }
]
```

22. If you have not customized this file, replace it with the new file from the sample configuration directory. If you have previously customized these files, open `templates/aws_baseline/aws-landing-zone-vpc.template` and update following code:

a. Add the **`ManagedResourcePrefix`** parameter highlighted below in green.

```
Parameters:
  ManagedResourcePrefix:
    Type: 'String'
    Description: 'Prefix for the managed resources'
  AvailabilityZones:
    Description: 'List of Availability Zones to use for the subnets in the VPC.'
    Type: CommaDelimitedList
  CreateAdditionalPrivateSubnets:
    AllowedValues:
```

c. Add the following deletion policy to retain all the network resources to avoid accidental deletion to all the resources in this template:

```
  DeletionPolicy: Retain
  Properties:
```

23. If you have not customized this file, replace it with the new file from the sample configuration directory. If you have previously customized these files, open `templates/core_accounts/aws-landing-zone-logging.template` and update following code (to align with AWS Control Tower Guardrail):

a. Remove the **`LoggingAccountId`** parameter

```
AWSTemplateFormatVersion: 2010-09-09
Description: Create a S3 logging bucket in the logging account.

Parameters:
  LoggingAccountId:
    Type: 'String'
    Description: AWS Account Id of the logging account.
  SSEAlgorithm:
```

b. Add the new parameter **`AWSLogsS3KeyPrefix`** highlighted below in green

```
    Description: 'KMS key ID required if SSE algorithm is aws:kms.'
  AWSLogsS3KeyPrefix:
    Type: 'String'
    Description: 'Organization ID to use as the S3 Key prefix for storing the audit logs'
```

c. Update the **`S3KmsBucket`** resource with correct logging configuration, if **`UseKMS`** condition is set to true.

```
  S3KmsBucket:
    DeletionPolicy: Retain
    Condition: UseKMS
    Type: AWS::S3::Bucket
    Properties:
      BucketName: !Sub aws-landing-zone-logs-${AWS::AccountId}-${AWS::Region}
      VersioningConfiguration:
        Status: Enabled
      LoggingConfiguration:
        DestinationBucketName: !Ref S3LoggingBucket
        DestinationBucketName: !Ref S3KmsLoggingBucket
      BucketEncryption:
```

d. For the resource '**S3KmsBucketPolicy**' update the bucket name to **`S3KmsBucket`** and **`prefix`** in the resource with correct bucket delivery policy

```
          Resource:
                  - Fn::Join:
                      - ""
                      -
                        - "arn:aws:s3:::"
                        - !Ref "S3Bucket"
                        - "/AWSLogs/*/*"
                        - !Ref "S3KmsBucket"
                        - !Sub "/${AWSLogsS3KeyPrefix}/AWSLogs/*/*"

# Create buckets using S3-SSE keys for default encryption
```

e. For the resource '**S3BucketPolicy**' update only the **`prefix`** in the resource with correct bucket delivery policy.

```
Resource:
        - Fn::Join:
            - ""
            -
              - "arn:aws:s3:::"
              - !Ref "S3Bucket"
              - "/AWSLogs/*/*"
              - !Sub "/${AWSLogsS3KeyPrefix}/AWSLogs/*/*"
```

24. If you have not customized this file, replace it with the new file from the sample configuration directory. If you have previously customized these files, open `parameters/core_accounts/aws-landing-zone-logging.json` and update following code (to align with AWS Control Tower Guardrail):

a. Remove the **`LoggingAccountId`** parameter highlighted below in red

```
[
  {
    "ParameterKey": "LoggingAccountId",
    "ParameterValue": "$[alfred_ssm_/org/member/logging/account_id]"
  },
  {
    "ParameterKey": "SSEAlgorithm",
    "ParameterValue": "AES256"
  },
  {
     "ParameterKey": "KMSMasterKeyID",
```

b. Add the **`AWSLogsS3KeyPrefix`** parameter highlighted below in green to the parameter file

```
    },
    {
        "ParameterKey": "AWSLogsS3KeyPrefix",
        "ParameterValue": "$[alfred_ssm_/org/primary/organization_id]"
    }
]
```

25. If you have not customized this file, replace it with the new file from the sample configuration directory. If you have previously customized these files, open `parameters/aws_baseline/aws-landing-zone-enable-cloudtrail.json` and update the `.json` file as shown below (to align with AWS Control Tower Guardrail):

```
        "ParameterValue": "$[alfred_ssm_/org/primary/sns_topic_arn]"
    },
    {
        "ParameterKey": "TrailBucket",
        "ParameterValue": "$[alfred_ssm_/org/member/logging/bucket_name]"
    },
    {
      "ParameterKey": "AWSLogsS3KeyPrefix",
      "ParameterValue": "$[alfred_ssm_/org/primary/organization_id]"
    }
]
```

26. If you have not customized this file, replace it with the new file from the sample configuration directory. If you have previously customized these files, open `templates/aws_baseline/aws-landing-zone-enable-cloudtrail.template` and update the template as shown below (to align with AWS Control Tower Guardrail):

a. Add the new parameter **`AWSLogsS3KeyPrefix`** highlighted below in green

```
  AllowedValues: [1, 3, 5, 7, 14, 30, 60, 90, 120, 150, 180, 365, 400, 545, 731, 1827, 3653]

AWSLogsS3KeyPrefix:
  Type: 'String'
  Description: 'Organization ID to use as the S3 Key prefix for storing the audit logs'
```

b. Update the **CloudTrail name** and add the **`S3 Key Prefix`** for CloudTrail logs highlighted below in green

```
Resources:
  Trail:
    Type: AWS::CloudTrail::Trail
    Properties:
      TrailName: AWS-Landing-Zone-BaselineCloudTrail
      S3BucketName: !Ref TrailBucket
      S3KeyPrefix: !Ref AWSLogsS3KeyPrefix
      SnsTopicName: !If
        - IsPublishToTopic
        - !Ref SNSTopic
        - !Ref AWS::NoValue
      IsLogging: True
```

> **Note:** The following changes must be done in two parts as explained below. Also, this change will update the Amazon S3 prefix where Amazon CloudTrail and AWS Config logs are stored. The AWS Landing Zone admin should either copy the logs to the new location or update the application to point to the new location.

Old path: aws-landing-zone-logs-<ACCOUNT_ID>-<REGION>/AWSLogs/<ACCOUNT_ID>

New Path: aws-landing-zone-logs-<ACCOUNT_ID>-<REGION>/<ORG_ID>/AWSLogs/<ACCOUNT_ID>

Part 1: AWS Config Service – Delete baseline

1. Update the manifest file, and comment the AWS Config baseline resource and the dependencies in the `manifest.yaml` file.
2. Comment out the **`EnableConfig`** baseline resource.

   This step will remove the AWS Config service from all the member accounts. In Part 2, the AWS Config service will be added back into all the member accounts. Doing this allows you to update the stack without causing the **Update Stack** for the `EnableConfig` template.

```
- name: ConfigRole
  baseline_products:
    - AWS-Landing-Zone-Account-Vending-Machine
  template_file: templates/aws_baseline/aws-landing-zone-enable-config-role.template
  deploy_method: stack_set

  # This template deploys the AWS Config service.
  # It can be deployed in multiple regions.
# - name: EnableConfig
#   baseline_products:
#     - AWS-Landing-Zone-Account-Vending-Machine
#   depends_on:
#    - ConfigRole
#   template_file: templates/aws_baseline/aws-landing-zone-enable-config.template
#   parameter_file: parameters/aws_baseline/aws-landing-zone-enable-config.json
#   deploy_method: stack_set
#   regions:
#     - ap-northeast-1
#     - ap-northeast-2
#     - ap-south-1
#     - ap-southeast-1
#     - ap-southeast-2
#     - ca-central-1
#     - eu-central-1
#     - eu-west-1
#     - eu-west-2
#     - eu-west-3
#     - sa-east-1
#     - us-east-1
#     - us-east-2
#     - us-west-1
#     - us-west-2
```

3. Comment out the dependency on **`EnableConfig`** resource in the following baseline resources:

- EnableConfigRulesGlobal
- EnableConfigRules
- EnableNotifications

```
- name: EnableConfigRulesGlobal
  baseline_products:
    - AWS-Landing-Zone-Account-Vending-Machine
  # depends_on:
  #   - EnableConfig
  template_file: templates/aws_baseline/aws-landing-zone-config-rules-global.template
  parameter_file: parameters/aws_baseline/aws-landing-zone-config-rules-global.json
  deploy_method: stack_set

  # This template deploys the Config Rules that monitor the local resources.
  # It can be deployed in multiple regions
- name: EnableConfigRules
  baseline_products:
    - AWS-Landing-Zone-Account-Vending-Machine
  # depends_on:
  #   - EnableConfig
  template_file: templates/aws_baseline/aws-landing-zone-config-rules.template
  parameter_file: parameters/aws_baseline/aws-landing-zone-config-rules.json
  deploy_method: stack_set
  regions: ⋯

- name: EnableNotifications
  baseline_products:
    - AWS-Landing-Zone-Account-Vending-Machine
  depends_on:
    - EnableCloudTrail
    # - EnableConfig
  template_file: templates/aws_baseline/aws-landing-zone-notifications.template
  parameter_file: parameters/aws_baseline/aws-landing-zone-notifications.json
  deploy_method: stack_set
```

4. If you have not customized this file, replace it with the new file from the sample configuration directory. If you have previously customized these files, open the `templates/aws_baseline/aws-landing-zone-enable-config.template` and update the following code (to align with AWS Control Tower Guardrail):
5. Add the new parameter **`AWSLogsS3KeyPrefix`** highlighted below in green

```
AWSLogsS3KeyPrefix:
  Type: 'String'
  Description: 'Organization ID to use as the S3 Key prefix for storing the audit logs'
```

6. Add a **static name** for the configuration recorder.

```
Resources:

  ConfigRecorder:
    Type: AWS::Config::ConfigurationRecorder
    Properties:
      Name: AWS-Landing-Zone-BaselineConfigRecorder
      RoleARN: !Sub arn:aws:iam::${AWS::AccountId}:role/AWS-Landing-Zone-ConfigRecorderRole
```

7. Add the **`S3 key prefix`** for the delivery channel logs highlighted below in green

```
    ConfigDeliveryChannel:
      Type: AWS::Config::DeliveryChannel
      Properties:
        Name: !If
          - IsGeneratedDeliveryChannelName
          - !Ref AWS::NoValue
          - !Ref DeliveryChannelName
        ConfigSnapshotDeliveryProperties:
          DeliveryFrequency: !FindInMap
            - Settings
            - FrequencyMap
            - !Ref Frequency
        S3BucketName: !Ref BucketName
        S3KeyPrefix: !Ref AWSLogsS3KeyPrefix
        SnsTopicARN: !Join
```

8. If you have not customized this file, replace it with the new file from the sample configuration directory. If you have previously customized these files, open `parameters/aws_baseline/aws-landing-zone-enable-config.json` and update following code (to align with AWS Control Tower Guardrail):
9. Add the **`AWSLogsS3KeyPrefix`** parameter to the parameter file highlighted below in green

```
    },
    {
        "ParameterKey": "AWSLogsS3KeyPrefix",
        "ParameterValue": "$[alfred_ssm_/org/primary/organization_id]"
    }
]
```

10. **Zip** all the files under the **/temp/**LZ_**v22** directory as a `aws-landing-zone-configuration.zip` file.
11. **Upload** the `aws-landing-zone-configuration.zip` to your LZ Configuration Amazon S3 Bucket (i.e. `aws-landing-zone-configuration-`<ACCOUNT_ID>`-`<REGION>). This will kick off the LZ CodePipeline.
12. If you have deployed the AWS Managed AD and Directory Connector for AWS SSO Add-On. You must update the provisioned product to the latest version because the **PrimaryAccountVPC** stackset resource use `templates/aws_baseline/aws-landing-zone-vpc.template`. The new version has the updated parameter file.

```
  },
  {
    "ParameterKey": "ManagedResourcePrefix",
    "ParameterValue": "aws-landing-zone"
  }
]
```

13. Wait for the LZ CodePipeline to successfully finish. Note that you must finish Part 2 below after the CodePipeline finishes successfully.

Part 2: AWS Config Service – Add baseline

1. **Download** and **Unzip** the **LZ** Configuration ZIP file (`aws-landing-zone-configuration.zip`) from your LZ Configuration S3 Bucket (i.e. `aws-landing-zone-configuration-<ACCOUNT_ID>-<REGION>`) into the **LZ_ v22** directory.
2. Update the manifest file, and comment the AWS Config baseline resource and its dependencies in the `manifest.yaml` file.
3. Uncomment the **`EnableConfig`** baseline resource. This will add the AWS Config service to all the member accounts.

```
- name: ConfigRole
  baseline_products:
    - AWS-Landing-Zone-Account-Vending-Machine
  template_file: templates/aws_baseline/aws-landing-zone-enable-config-role.template
  deploy_method: stack_set

  # This template deploys the AWS Config service.
  # It can be deployed in multiple regions.
- name: EnableConfig
  baseline_products:
    - AWS-Landing-Zone-Account-Vending-Machine
  depends_on:
   - ConfigRole
  template_file: templates/aws_baseline/aws-landing-zone-enable-config.template
  parameter_file: parameters/aws_baseline/aws-landing-zone-enable-config.json
  deploy_method: stack_set
  regions:
    - ap-northeast-1
    - ap-northeast-2
    - ap-south-1
    - ap-southeast-1
    - ap-southeast-2
    - ca-central-1
    - eu-central-1
    - eu-west-1
    - eu-west-2
    - eu-west-3
    - sa-east-1
    - us-east-1
    - us-east-2
    - us-west-1
    - us-west-2
```

4. Uncomment the dependency on **`EnableConfig`** resource in the following baseline resources:

- EnableConfigRulesGlobal
- EnableConfigRules
- EnableNotifications

```
- name: EnableConfigRulesGlobal
  baseline_products:
    - AWS-Landing-Zone-Account-Vending-Machine
  depends_on:       # uncomment this line
    - EnableConfig  # uncomment this line
  template_file: templates/aws_baseline/aws-landing-zone-config-rules-global.template
  parameter_file: parameters/aws_baseline/aws-landing-zone-config-rules-global.json
  deploy_method: stack_set

  # This template deploys the Config Rules that monitor the local resources.
  # It can be deployed in multiple regions
- name: EnableConfigRules
  baseline_products:
    - AWS-Landing-Zone-Account-Vending-Machine
  depends_on:       # uncomment this line
    - EnableConfig  # uncomment this line
  template_file: templates/aws_baseline/aws-landing-zone-config-rules.template
  parameter_file: parameters/aws_baseline/aws-landing-zone-config-rules.json
  deploy_method: stack_set
  regions:⋯

- name: EnableNotifications
  baseline_products:
    - AWS-Landing-Zone-Account-Vending-Machine
  depends_on:
    - EnableCloudTrail
    - EnableConfig # uncomment this line
  template_file: templates/aws_baseline/aws-landing-zone-notifications.template
  parameter_file: parameters/aws_baseline/aws-landing-zone-notifications.json
  deploy_method: stack_set
```

5. **Zip** all the files under the **/temp**/LZ_**v22** directory as a `aws-landing-zone-configuration.zip` file.
6. **Upload** the `aws-landing-zone-configuration.zip` to your LZ Configuration Amazon S3 Bucket (i.e. `aws-landing-zone-configuration-<ACCOUNT_ID>-<REGION>`). This will kick off the LZ CodePipeline. Wait for the LZ CodePipeline to successfully finish.

Upgrade Instructions (v2.0.3 to v2.1)

Release Notes

Release includes following features and bug fixes:

- Support for creating nested Organizational Units (OU)
- Apply Service Control Policy (SCP) at OU level instead of Account level

- Update for AWS Managed AD and Directory Connector for AWS SSO Add-On; allow Directory Connector to be deployed in all AWS SSO supported regions
- Update for AWS Centralized Logging Solution Add-On to retain Cognito user pool & Elasticsearch domain even after the solution stack is deleted.
- Add the new input parameter for the LZ Initiation template to enable AWS Security Monitoring in all regions (production) vs current region (Immersion Day/Demo)
- Use regional STS endpoints in place of global endpoint
- Performance improvements
  - Core Resource Stage - If the core resource template or parameter or account or region(s) does not change, it will skip the update stack set.
  - Service Catalog Stage – It will skip creating a new version if no changes were made in AVM. This improves the LaunchAVM Stage. Currently every time the pipeline runs, it creates a new version of AVM. With this optimization, if nothing has changed in AVM, it will not generate the new version of AVM.
- Bug Fixes
  - Unable to deploy AVM with Network in US-West-1 region
  - Landing zone API throttle limit exceed error while describing stack
  - When user moves any of the core account from core to different OU, by updating the manifest, the LaunchAVM moves the Core account back into the core OU
  - Manifest with only the PRIMARY account in 'core' OU fails
  - Intermittent CodeBuild stage failure due to S3 error: Access Denied
  - GuardDuty findings were not sent to the security alert email.

Mandatory Upgrade Steps

Performing these upgrade steps only upgrades your Landing Zone Framework and keeps your existing Landing Zone configuration as is.

Use the following procedure to perform the mandatory upgrade:

1. **Backup** your existing Landing Zone Configuration ZIP file (`aws-landing-zone-configuration.zip`) from your Landing Zone Configuration Amazon S3 Bucket (i.e. `aws-landing-zone-configuration-<ACCOUNT_ID>-<REGION>`)
2. Navigate to the AWS CloudFormation Console, select the **Landing Zone Initiation Stack**, and select **Upgrade**.

> **WARNING**: Any changes made to the AWS Landing Zone Framework services such as AWS CodePipeline, AWS CodeBuild, AWS Step functions, AWS Lambda functions and related IAM Roles as part of the initiation template or on the deployed resources, those changes will be overwritten by this step.

3. Use the linked Amazon S3 template URL.

> **Note**: Do not change any input parameters, it will not update the existing LZ configuration ZIP file.

4. Wait for the Update Stack to complete
5. **Update** the KMS key policy for **AwsLandingZoneKMSKey** by adding your IAM user/role ARN under the **Allow use of the key** section of the policy.

> **Note:** This is required to grant your own IAM user/role permission to use the KMS key for downloading/uploading the LZ configuration ZIP file.

```
    {
      "Sid": "Allow use of the key",
      "Effect": "Allow",
      "Principal": {
        "AWS": [
          "arn:aws:iam::xxxxxxxxxxxx:role/Admin",
”arn:aws:iam::xxxxxxxxxxxx:role/LandingZoneHandshakeSMLam
bdaRole",
"arn:aws:iam::xxxxxxxxxxxx:role/LandingZoneDeploymentLamb
daRole",
"arn:aws:iam::xxxxxxxxxxxx:role/StateMachineLambdaRole",
"arn:aws:iam::xxxxxxxxxxxx:role/LandingZoneLambdaRole",
"arn:aws:iam::xxxxxxxxxxxx:role/StateMachineTriggerLambda
Role",
"arn:aws:iam::xxxxxxxxxxxx:role/LandingZoneCodePipelineRo
le"
        ]
      }
```

6. **Create** a temporary directory called LZ_**v21** on your local machine in your preferred location. For Example: /temp/**LZ_v21**
7. **Download** and **Unzip** the **LZ** Configuration ZIP file (`aws-landing-zone-configuration.zip`) from your LZ Configuration S3 Bucket (i.e. `aws-landing-zone-configuration-<ACCOUNT_ID>-<REGION>`) into the **LZ_ v21** directory.
8. Open templates/aws_baseline/aws-landing-zone-avm.template.j2 and make the following code updates:

a. Remove AttachSCP Custom Resource

```
# SCP Custom Resource - Attach SCP
#

  AttachSCP:
    Type: Custom::ServiceControlPolicy
    DependsOn:
      - Organizations
      - DetachSCP
      - ExpungeVPC
        {%- for resource in manifest.baseline_resources %}
          {%- if manifest.portfolios[portfolio_index].products[product_index].name in resource.baseline_products
%}
            {%- if resource.name != 'PrimaryVPC' %}
      - StackSet{{resource.name}}
            {%- endif %}
          {%- endif %}
        {%- endfor %}
    Properties:
      AccountId: !GetAtt 'Organizations.AccountId'
      PolicyList: !FindInMap [OUToSCPMap, !Ref OrgUnitName, SCP]
      Operation: Attach
      ServiceToken: {{ lambda_arn }}
      key: {{ uuid }}

#
```

b. Remove DetachSCP Custom Resource

```
# SCP Custom Resource - Detach SCP
#

  DetachSCP:
    DependsOn:
      - Organizations
    Type: Custom::ServiceControlPolicy
    Properties:
      AccountId: !GetAtt 'Organizations.AccountId'
      PolicyList: !FindInMap [OUToSCPMap, !Ref OrgUnitName, SCP]
      Operation: Detach
      ServiceToken: {{ lambda_arn }}
      key: {{ uuid }}

#
```

c. Remove all related dependsOn condition for DetachSCP

```
  StackSet{{resource.name}}:
    DependsOn:
      - Organizations
      - DetachSCP
        {%- if resource.depends_on %}
            {%- for depends_on in resource.depends_on %}
      - StackSet{{ depends_on }}
```

```
    Condition: CreateVPC
    DependsOn:
      - Organizations
      - DetachSCP
        {%- if resource.depends_on %}
            {%- for depends_on in resource.depends_on %}
      - {{ depends_on }}
```

```
  GuardDutyMemberof{{ account.name.title() }}Account{{region.title() | replace("-","") }}:
    DependsOn:
      - Organizations
      - DetachSCP
    Type: Custom::HandShakeStateMachine
    Properties:
      ServiceType: GuardDuty
```

```
ExpungeVPC:
  DependsOn:
    - Organizations
    - DetachSCP
  Type: Custom::ExpungeVPC
  Properties:
      AccountList:
```

d. Remove Mappings: OUToSCPMap

```
Mappings:
  OUToSCPMap:
    {%- for ou in manifest.organizational_units %}
        {%- set include_ou = { 'flag': False } %}
        {%- for policy in manifest.organization_policies %}
            {%- if ou.name in policy.apply_to_accounts_in_ou %}
              {%-   if include_ou.update({'flag':True}) %}{%- endif %}
            {%- endif %}
        {%- endfor %}
        {%- if include_ou.flag %}
        {{ ou.name }}:
            SCP:
            {%- for policy in manifest.organization_policies %}
                {%- if ou.name in policy.apply_to_accounts_in_ou %}
                - {{ policy.name }}
                {%- endif %}
            {%- endfor %}
        {%- else %}
        {{ ou.name }}:
            SCP: []
        {%- endif %}
    {%- endfor %}
```

e. Remove key: {{ uuid }} from the "Organizations" Custom resource

```
      OUNameDelimiter: '{{manifest.nested_ou_delimiter}}'
      {%- endif %}
      ServiceToken: {{ lambda_arn }}
      key: {{ uuid }}
```

9. Add OUNameDelimiter key and related Jinja code under Organizations Custom resource

```
      OUName: !Ref OrgUnitName
      AccountName: !Ref AccountName
      AccountEmail: !Ref AccountEmail
      {%- if manifest.nested_ou_delimiter != '' %}
      OUNameDelimiter: '{{manifest.nested_ou_delimiter}}'
      {%- endif %}
      ServiceToken: {{ lambda_arn }}
      key: {{ uuid }}
```

```
{%- if manifest.nested_ou_delimiter != '' %}
OUNameDelimiter: '{{manifest.nested_ou_delimiter}}'
{%- endif %}
```

10. Add AccountList under Properties for VPCCalcuator Custom resource

```
    DependsOn:
      - Organizations
    Properties:
      AccountList:
        - !GetAtt 'Organizations.AccountId'
      VPCCidr: !Ref VPCCidr
      PublicSubnets: !FindInMap
        - VPC
```

```
        AccountList:
          - !GetAtt 'Organizations.AccountId'
```

11. Open policies/prevent_deleting_cloudtrails_config.json and make the following code update:

    a. Remove Allow */* policy statement

```
            "Effect": "Allow",
            "Action": "*",
            "Resource": "*"
```

    b. Add condition for all the statements in the document

```
            "Resource": "*"
            "Resource": "*",
            "Condition": {
                "ArnNotLike": {

"aws:PrincipalARN":"arn:aws:iam::*:role/AWSCloudFormationStackSetExecutionRole"
                }
            }
```

```
            "Resource": "*"
            "Resource": "*",
            "Condition": {
                "ArnNotLike": {

"aws:PrincipalARN":"arn:aws:iam::*:role/AWSCloudFormationStackSetExecutionRole"
                }
            }
```

```
"Condition": {
      "ArnNotLike": {

"aws:PrincipalARN":"arn:aws:iam::*:role/AWSCloudFormation
StackSetExecutionRole"
       }
}
```

12. Open `manifest.yaml` and make the following code update:

    a. Remove `apply_baseline_to_accounts_in_ou` key (with the value) under products sections

```
      # Do you wish to auto-apply this baseline to accounts everytime a new version of AVM product is created
by pipeline?
      apply_baseline_to_accounts_in_ou:
        - core
        - applications
```

b. If you are planning to create the nested OUs, then add `nested_ou_delimiter` key (with the value) in the top section of the `manifest.yaml`

```
version: 2018-06-14
lock_down_stack_sets_role: No
nested_ou_delimiter: '#'
```

Allowed values for `nested_ou_delimiter` are as follows:

```
: (Colon)
. (Dot)
- (Hyphen)
_ (Underscore)
; (Semicolon)
# (Hash)
| (Pipe)
```

13. Open `templates/core_accounts/aws-landing-zone-guardduty-master`.template and add the following lines of code under **Resources:** section:

```
Resources:
……
  SNSNotificationPolicy:
    Type: AWS::SNS::TopicPolicy
    Metadata:
      cfn_nag:
        rules_to_suppress:
          - id: F18
            reason:  "Condition  restricts  permissions  to  current
account."
    Properties:
      Topics:
        - !Ref SNSNotificationTopic
      PolicyDocument:
        Statement:
          - Sid: __default_statement_ID
            Effect: Allow
            Principal:
              AWS: "*"
            Action:
            - SNS:GetTopicAttributes
            - SNS:SetTopicAttributes
            - SNS:AddPermission
            - SNS:RemovePermission
            - SNS:DeleteTopic
            - SNS:Subscribe
            - SNS:ListSubscriptionsByTopic
```

```
              - SNS:Publish
              - SNS:Receive
              Resource: !Ref SNSNotificationTopic
              Condition:
                StringEquals:
                  AWS:SourceOwner: !Sub ${AWS::AccountId}
            - Sid: TrustCWEToPublishEventsToMyTopic
              Effect: Allow
              Principal:
                Service: events.amazonaws.com
              Action: sns:Publish
              Resource: !Ref SNSNotificationTopic
```

14. **Zip** all the files under the **/temp/**LZ_**v21** directory as a `aws-landing-zone-configuration.zip` file.
15. **Upload** the `aws-landing-zone-configuration.zip` to your LZ Configuration Amazon S3 Bucket (i.e. `aws-landing-zone-configuration-<ACCOUNT_ID>-<REGION>`). This will kick off the LZ CodePipeline.
16. Wait for the LZ CodePipeline to successfully finish.

Upgrade Instructions (v2.0.2 to v2.0.3)

Release Notes

Release includes following bug fixes:

- Update the NodeJS version from v6.10 to v8.10 for the AWS CloudFormation template that configures IAM Password policy for newly vended accounts, since NodeJS v6.10 is reaching EOL by end of April, 2019

Mandatory Upgrade Steps

Performing these upgrade steps only upgrades your Landing Zone Framework and keeps your existing Landing Zone configuration as is.

Use the following procedure to perform the mandatory upgrade:

1. **Backup** your existing Landing Zone Configuration ZIP file (`aws-landing-zone-configuration.zip`) from your Landing Zone Configuration Amazon S3 Bucket (i.e. `aws-landing-zone-configuration-<ACCOUNT_ID>-<REGION>`)
2. Navigate to the AWS CloudFormation Console, select the **Landing Zone Initiation Stack**, and select **Upgrade**.
3. Use the linked Amazon S3 template URL.

> **Note**: Do not change any input parameters, it will not update the existing LZ configuration ZIP file.

4. Wait for the Update Stack to complete
5. **Update** the KMS key policy for **AwsLandingZoneKMSKey** by adding your IAM user/role ARN under the **Allow use of the key** section of the policy.

> **Note:** This is required to grant your own IAM user/role permission to use the KMS key for downloading/uploading the LZ configuration ZIP file.

```
    {
      "Sid": "Allow use of the key",
      "Effect": "Allow",
      "Principal": {
        "AWS": [
          "arn:aws:iam::xxxxxxxxxxxx:role/Admin",
”arn:aws:iam::xxxxxxxxxxxx:role/LandingZoneHandshakeSMLam
bdaRole",
"arn:aws:iam::xxxxxxxxxxxx:role/LandingZoneDeploymentLamb
daRole",
"arn:aws:iam::xxxxxxxxxxxx:role/StateMachineLambdaRole",
```

```
  "arn:aws:iam::xxxxxxxxxxxx:role/LandingZoneLambdaRole",
  "arn:aws:iam::xxxxxxxxxxxx:role/StateMachineTriggerLambda
  Role",
  "arn:aws:iam::xxxxxxxxxxxx:role/LandingZoneCodePipelineRo
  le"
          ]
        }
```

6. **Create** a temporary directory called LZ_**v203** on your local machine in your preferred location. For Example: /temp/**LZ_v203**
7. **Download** and **Unzip** the **LZ** Configuration ZIP file (`aws-landing-zone-configuration.zip`) from your LZ Configuration S3 Bucket (i.e. `aws-landing-zone-configuration-<ACCOUNT_ID>-<REGION>`) into the **LZ_ v203** directory.
8. Open the /aws_baseline/aws-landing-zone-iam-password-policy.template and make the following code update:

templates/aws_baseline/aws-landing-zone-iam-password-policy.template

Lines 165 to 175 (Context lines: 5, 20, 100)

```
165  165              }
166  166            };
167  167        Handler: 'index.handler'
168  168        MemorySize: 128
169  169        Role: !GetAtt 'LambdaRole.Arn'
170             Runtime: 'nodejs6.10'
     170        Runtime: 'nodejs8.10'
171  171        Timeout: 60
172  172    LambdaLogGroup:
173  173      Type: 'AWS::Logs::LogGroup'
174  174      Properties:
175  175        LogGroupName: !Sub '/aws/lambda/${IamPasswordPolicyCustomResource}'
```

9. **Zip** all the files under the **/temp/LZ_v203** directory as a `aws-landing-zone-configuration.zip` file.
10. **Upload** the `aws-landing-zone-configuration.zip` to your LZ Configuration Amazon S3 Bucket (i.e. `aws-landing-zone-configuration-<ACCOUNT_ID>-<REGION>`). This will kick off the LZ CodePipeline.
11. Wait for the LZ CodePipeline to successfully finish.

Upgrade Instructions (v2.0.1 to v2.0.2)

Release Notes

Release includes following bug fixes:

- Fixed the bug introduced in v2.0.1 where the LaunchAVM stage of pipeline attempted to remove the existing VPCs provisioned from AVM in the existing vended accounts.

Mandatory Upgrade Steps

Performing these upgrade steps only upgrades your Landing Zone Framework and keeps your existing Landing Zone configuration as is.

Use the following procedure to perform the mandatory upgrade:

1. Navigate to the AWS CloudFormation Console, select the **Landing Zone Initiation Stack**, and select **Upgrade**.
2. Use the linked Amazon S3 template URL.

> **Note**: Do not change any input parameters, it will not update the existing LZ configuration ZIP file.

3. Wait for the Update Stack to complete.

Upgrading from v2.0 to v2.0.1

Release Notes

This release includes the following bug fixes:

- Fixes the issue when the Landing Zone pipeline completes, it may leave one or more accounts without the desired SCP(s) attached to it.
- Fixes for the StackSet State Machine to be able to update the override parameters on the stack instances, by invoking the Update Stack Instance on the existing stacks. Note that this only happens if the instances have the parameter override.
- The LaunchAVM State Machine shows more detailed logging when the last stage of pipeline LaunchAVM fails.
- Adding a new Organizational Unit to the manifest.yaml file, no longer requires the OU to be associated with any SCP.
- The last stage of the CodePipeline LaunchAVM will not fail if it finds the SUSPENDED accounts inside the Landing Zone managed OU. The account will be moved out of the OU to Organizations root.

- Optimizations for the BaselineResource stage of the CodePipeline to execute faster, it will no longer perform the UpdateStackSet workflow, if the template and parameter files have not been updated since the last pipeline run.

Mandatory Upgrade Steps

Performing these upgrade steps only upgrades your Landing Zone framework and keeps your existing Landing Zone configuration as is.

Use the following procedure to perform the mandatory upgrade:

1. **Backup** your existing Landing Zone Configuration ZIP file (`aws-landing-zone-configuration.zip`) from your Landing Zone Configuration Amazon S3 Bucket (i.e. `aws-landing-zone-configuration-<ACCOUNT_ID>-<REGION>`)
2. Navigate to the AWS CloudFormation Console, select the **Landing Zone Initiation Stack**, and select **Upgrade**.
3. Use the linked Amazon S3 template URL.

> **Note**: Do not change any input parameters, it will not update the existing LZ configuration ZIP file.

4. Wait for the Update Stack to complete
5. **Update** the KMS key policy for **AwsLandingZoneKMSKey** by adding your IAM user/role ARN under the **Allow use of the key** section of the policy.

> **Note:** This is required to grant your own IAM user/role permission to use the KMS key for downloading/uploading the LZ configuration ZIP file.

```
    {
      "Sid": "Allow use of the key",
      "Effect": "Allow",
      "Principal": {
        "AWS": [
          "arn:aws:iam::xxxxxxxxxxxx:role/Admin",
"arn:aws:iam::xxxxxxxxxxxx:role/LandingZoneHandshakeSMLambdaRole",
"arn:aws:iam::xxxxxxxxxxxx:role/LandingZoneDeploymentLambdaRole",
"arn:aws:iam::xxxxxxxxxxxx:role/StateMachineLambdaRole",
"arn:aws:iam::xxxxxxxxxxxx:role/LandingZoneLambdaRole",
"arn:aws:iam::xxxxxxxxxxxx:role/StateMachineTriggerLambdaRole",
"arn:aws:iam::xxxxxxxxxxxx:role/LandingZoneCodePipelineRole"
        ]
      }
```

6. **Create** the following temporary directories on your local machine in your preferred location: **LZ_v2** and **LZ_v201**. For Example: /temp/**LZ_v2** and /temp/**LZ_v201**

7. Download and Unzip the **LZ_v201** Configuration Zip file and place into LZ_v201 directory. For Example: /temp/LZ_v2 directory
8. **Download** and **Unzip** the **LZ** Configuration ZIP file (`aws-landing-zone-configuration.zip`) from your LZ Configuration S3 Bucket (i.e. `aws-landing-zone-configuration-<ACCOUNT_ID>-<REGION>`) into the **LZ_v2** directory. For Example: /temp/**LZ_v2** directory

> **Note:** In the steps below, make sure you are editing and/or copying the configuration files into /temp/**LZ_v2** directory

9. If you have not customized the `templates/aws_baseline/aws-landing-zone-avm.template.j2` file, replace it with the linked file.

   If you have customized the file, **compare** the `templates/aws_baseline/aws-landing-zone-avm.template.j2` file between **LZ_v2** and **LZ_v201** in text editor and edit the following lines shown below. Check to make sure any other customizations that you made before are not overwritten with the changes below. Remove the lines highlighted in red and add the lines highlighted in green below:

```
309  320    DetachSCP:
310  321      DependsOn:
311  322        - Organizations
312  323      Type: Custom::ServiceControlPolicy
313  324      Properties:
314  325        AccountId: !GetAtt 'Organizations.AccountId'
315  326        PolicyList: !FindInMap [OUToSCPMap, !Ref OrgUnitName, SCP]
316  327        Operation: Detach
317  328        ServiceToken: {{ lambda_arn }}
     329        key: {{ uuid }}
318  330
```

```
391  403  AttachSCP:
392  404    Type: Custom::ServiceControlPolicy
393  405    DependsOn:
394  406      - Organizations
395  407      - DetachSCP
     408      - ExpungeVPC
396  409        {%- for resource in manifest.baseline_resources %}
397  410          {%- if manifest.portfolios[portfolio_index].products[product_index].name in resource.baseline_products %}
398  411            {%- if resource.name != 'PrimaryVPC' %}
399  412      - StackSet{{resource.name}}
400  413            {%- endif %}
401  414          {%- endif %}
402  415        {%- endfor %}
403  416    Properties:
404  417      AccountId: !GetAtt 'Organizations.AccountId'
405  418      PolicyList: !FindInMap [OUToSCPMap, !Ref OrgUnitName, SCP]
406  419      Operation: Attach
407  420      ServiceToken: {{ lambda_arn }}
     421      key: {{ uuid }}
```

> If you plan to perform the **Strongly Recommended** steps, skip the following steps, and go straight to the Strongly Recommended section.

10. **Zip** all the files under the **/temp/LZ_v201** directory as a `aws-landing-zone-configuration.zip` file.
11. **Upload** the `aws-landing-zone-configuration.zip` to your LZ Configuration Amazon S3 Bucket (i.e. `aws-landing-zone-configuration-<ACCOUNT_ID>-<REGION>`). This will kick off the LZ CodePipeline.
12. Wait for the LZ CodePipeline to successfully finish.

Strongly Recommended Upgrade Steps

> **Note:** You must perform the Mandatory Upgrade, before starting this upgrade.

1. **Compare** the `parameters/aws_baseline/aws-landing-zone-config-rules.json` file between **LZ_v2** and **LZ_v201** in text editor.
2. Modify the parameter EnableS3ServerSideEncryptionRule from false to true.

```
18  18    {
19  19      "ParameterKey": "EnableS3ServerSideEncryptionRule",
20          "ParameterValue": "false"
        20  "ParameterValue": "true"
21  21    },
```

3. **Compare** the `parameters/aws_baseline/aws-landing-zone-primary-vpc.json` file between **LZ_v2** and **LZ_v201** in text editor.
4. **Modify** the ParameterKey: *<AvailabilityZones>*

```
1   1   [
2   2     {
3   3       "ParameterKey": "AvailabilityZones",
4           "ParameterValue": "$[alfred_genaz_2]"
        4   "ParameterValue": "$[alfred_genaz_2]",
        5   "ssm_parameters": [
        6     {
        7       "name": "/org/member/primary_vpc/dummy_az_list",
        8       "value": "$[AZ]"
        9     }
        10  ]
5   11    },
6   12    {
7   13      "ParameterKey": "NumberOfAZs",
8   14      "ParameterValue": "2"
9   15    }
```

5. **Compare** the templates/aws_baseline/aws-landing-zone-iam-password-policy.template file between **LZ_v2** and **LZ_v201** in text editor.
6. Make the following code updates highlighted in green:

```
148 148             };
149 149             if (event.RequestType === 'Delete') {
150 150               iam.deleteAccountPasswordPolicy({}, done);
151 151             } else if (event.RequestType === 'Create' || event.RequestType === 'Update') {
152 152               iam.updateAccountPasswordPolicy({
153               -     AllowUsersToChangePassword: Boolean(event.ResourceProperties.AllowUsersToChangePassword),
154               -     HardExpiry: Boolean(event.ResourceProperties.HardExpiry),
    153           +     AllowUsersToChangePassword: Boolean(event.ResourceProperties.AllowUsersToChangePassword === 'true'),
    154           +     HardExpiry: Boolean(event.ResourceProperties.HardExpiry === 'true'),
155 155                 MaxPasswordAge: event.ResourceProperties.MaxPasswordAge,
156 156                 MinimumPasswordLength: event.ResourceProperties.MinimumPasswordLength,
157 157                 PasswordReusePrevention: event.ResourceProperties.PasswordReusePrevention,
158               -     RequireLowercaseCharacters: Boolean(event.ResourceProperties.RequireLowercaseCharacters),
159               -     RequireNumbers: Boolean(event.ResourceProperties.RequireNumbers),
160               -     RequireSymbols: Boolean(event.ResourceProperties.RequireSymbols),
161               -     RequireUppercaseCharacters: Boolean(event.ResourceProperties.RequireUppercaseCharacters),
    158           +     RequireLowercaseCharacters: Boolean(event.ResourceProperties.RequireLowercaseCharacters === 'true'),
    159           +     RequireNumbers: Boolean(event.ResourceProperties.RequireNumbers === 'true'),
    160           +     RequireSymbols: Boolean(event.ResourceProperties.RequireSymbols === 'true'),
    161           +     RequireUppercaseCharacters: Boolean(event.ResourceProperties.RequireUppercaseCharacters === 'true'),
162 162               }, done);
163 163             } else {
164 164               cb(new Error(`unsupported RequestType: ${!event.RequestType}`));
165 165             }
166 166           };
```

7. **Compare** the `templates/aws_baseline/aws-landing-zone-vpc.template` file between **LZ_v2** and **LZ_v201** in text editor.
8. Make the following code updates highlighted in green:

```
176 176 Conditions:
177 177   PublicSubnetsCondition: !Equals [!Ref 'CreatePublicSubnets', 'true']
178 178   3AZCondition: !Or [!Equals [!Ref 'NumberOfAZs', '3'], !Condition '4AZCondition']
179 179   4AZCondition: !Equals [!Ref 'NumberOfAZs', '4']
180       - 3AZPublicCondition: !Or [!And [!Condition '3AZCondition', !Condition 'PublicSubnetsCondition'], !Condition '4AZCondition']
    180   + 3AZPublicCondition: !And [!Condition '3AZCondition', !Condition 'PublicSubnetsCondition']
181 181   4AZPublicCondition: !And [!Condition '4AZCondition', !Condition 'PublicSubnetsCondition']
182 182   AdditionalPrivateSubnetsCondition: !And [!Equals [!Ref 'CreatePrivateSubnets', 'true'],
183 183     !Equals [!Ref 'CreateAdditionalPrivateSubnets', 'true']]
184 184   AdditionalPrivateSubnets&3AZCondition: !And [!Condition 'AdditionalPrivateSubnetsCondition',
185 185     !Condition '3AZCondition']
```

```
186 186   AdditionalPrivateSubnets&4AZCondition: !And [!Condition 'AdditionalPrivateSubnetsCondition',
187 187     !Condition '4AZCondition']
188 188   NATGatewayCondition: !And [!Condition 'PrivateSubnetsCondition', !Condition 'PublicSubnetsCondition']
189 189   NATGateway&3AZCondition: !And [!Condition 'NATGatewayCondition', !Condition '3AZCondition']
190 190   NATGateway&4AZCondition: !And [!Condition 'NATGatewayCondition', !Condition '4AZCondition']
    191   + AdditionalPrivateSubnets&NATGatewayCondition: !And [!Condition 'AdditionalPrivateSubnetsCondition', !Condition 'NATGatewayCondition']
    192   + AdditionalPrivateSubnets&NATGateway&3AZCondition: !And [!Condition 'AdditionalPrivateSubnets&3AZCondition', !Condition 'NATGateway&3AZCondition']
    193   + AdditionalPrivateSubnets&NATGateway&4AZCondition: !And [!Condition 'AdditionalPrivateSubnets&4AZCondition', !Condition 'NATGateway&4AZCondition']
191 194   NVirginiaRegionCondition: !Equals [!Ref 'AWS::Region', us-east-1]
192 195   PrivateSubnetsCondition: !Equals [!Ref 'CreatePrivateSubnets', 'true']
193 196   PrivateSubnets&3AZCondition: !And [!Condition 'PrivateSubnetsCondition', !Condition '3AZCondition']
194 197   PrivateSubnets&4AZCondition: !And [!Condition 'PrivateSubnetsCondition', !Condition '4AZCondition']
195 198   Public&PrivateSubnetsCondition: !And [!Condition 'PublicSubnetsCondition', !Condition 'PrivateSubnetsCondition']
196 199   Public&PrivateSubnets&3AZCondition: !And [!Condition 'PublicSubnetsCondition', !Condition 'PrivateSubnetsCondition', !Condition '3AZCondition']
197 200   Public&PrivateSubnets&4AZCondition: !And [!Condition 'PublicSubnetsCondition', !Condition 'PrivateSubnetsCondition', !Condition '4AZCondition']
198 201   S3VPCEndpointCondition: !And [!Condition 'PrivateSubnetsCondition', !Not [!Or [
199 202         !Equals [!Ref 'AWS::Region', us-gov-west-1], !Equals [!Ref 'AWS::Region',
200 203           cn-north-1]]]]
201 204   TransitVPCCondition: !Equals [!Ref 'TransitVPC', 'true']
```

```
494  497   PrivateSubnet1BRoute:
495          Condition: AdditionalPrivateSubnetsCondition
       498   Condition: AdditionalPrivateSubnets&NATGatewayCondition
496  499     Type: AWS::EC2::Route
497  500     Properties:
498  501       RouteTableId: !Ref 'PrivateSubnet1BRouteTable'
499  502       DestinationCidrBlock: 0.0.0.0/0
500  503       NatGatewayId: !If [NATGatewayCondition, !Ref 'NATGateway1', !Ref 'AWS::NoValue']
```

```
553  556   PrivateSubnet2BRoute:
554          Condition: AdditionalPrivateSubnetsCondition
       557   Condition: AdditionalPrivateSubnets&NATGatewayCondition
555  558     Type: AWS::EC2::Route
556  559     Properties:
557  560       RouteTableId: !Ref 'PrivateSubnet2BRouteTable'
558  561       DestinationCidrBlock: 0.0.0.0/0
559  562       NatGatewayId: !If [NATGatewayCondition, !Ref 'NATGateway2', !Ref 'AWS::NoValue']
```

```
612  615   PrivateSubnet3BRoute:
613          Condition: AdditionalPrivateSubnets&3AZCondition
       616   Condition: AdditionalPrivateSubnets&NATGateway&3AZCondition
614  617     Type: AWS::EC2::Route
615  618     Properties:
616  619       RouteTableId: !Ref 'PrivateSubnet3BRouteTable'
617  620       DestinationCidrBlock: 0.0.0.0/0
618  621       NatGatewayId: !If [NATGatewayCondition, !Ref 'NATGateway3', !Ref 'AWS::NoValue']
```

```
671  674   PrivateSubnet4BRoute:
672          Condition: AdditionalPrivateSubnets&4AZCondition
       675   Condition: AdditionalPrivateSubnets&NATGateway&4AZCondition
673  676     Type: AWS::EC2::Route
674  677     Properties:
675  678       RouteTableId: !Ref 'PrivateSubnet4BRouteTable'
676  679       DestinationCidrBlock: 0.0.0.0/0
677  680       NatGatewayId: !If [NATGatewayCondition, !Ref 'NATGateway4', !Ref 'AWS::NoValue']
```

9. **Zip** all the files under the **/temp/ LZ_v1/** directory as `aws-landing-zone-configuation.zip`.
10. **Upload** the `aws-landing-zone-configuation.zip` to your LZ Configuration Amazon S3 bucket (i.e. `aws-landing-zone-configuration-<ACCOUNT_ID>-<REGION>`). This will kick off the LZ CodePipeline.
11. Wait for the LZ CodePipeline to successfully finish.

Upgrade Instructions v1.0.2 to v2.0

Release Notes

- Introduces the following AWS Landing Zone Add-on products (Available as Service Catalog products)
  - Centralized Logging Solution
  - AWS Managed AD and Directory Connector for AWS SSO
- Adds generic Handshake State Machine to perform invite/accept workflow steps for Amazon VPC peering, and Amazon GuardDuty and can be extended for other APIs e.g. Macie, Config Aggregator
- Offers support for remotely sourced templates and parameters in manifest.yaml
- Offers a never expire password for AD connector user.
- Updates the Service Catalog State Machine to apply the template constraint rules on Service Catalog Products such as Account Vending Machine
- Changes the default options for the RDGW instance type to t2.micro
- Bug Fixes
  - AWS Config and AWS Config Rules can be deployed in multiple regions
  - Account Vending Machine Input validation checks if the user selects the Public only VPC pattern, then the Peering option must be false
  - If more than one VPC with the same CIDR attempts to peer with Shared Services VPC, after the second attempt AVM fails and rolls back, and deletes the routes added by the first VPC
  - Fixes the issues with updating the IAM password policy
  - Adds new Account Vending Machine parameter(s) and does not break the last stage LaunchAVM of the pipeline.

Mandatory Upgrade Steps

Performing these upgrade steps only upgrades your Landing Zone framework and keeps your existing Landing Zone configuration as is.

Use the following procedure to perform the mandatory upgrade:

1. **Backup** your existing Landing Zone Configuration ZIP file (`aws-landing-zone-configuration.zip`) from your Landing Zone Configuration Amazon S3 Bucket (i.e. `aws-landing-zone-configuration-<ACCOUNT_ID>-<REGION>`)

2. Navigate to the AWS CloudFormation Console, select the **Landing Zone Initiation Stack**, and select **Upgrade**.
3. Use the linked Amazon S3 template URL.

> **Note**: Do not change any input parameters, it will not update the existing LZ configuration ZIP file.

4. Wait for the Update Stack to complete
5. **Update** the KMS key policy for **AwsLandingZoneKMSKey** by adding your IAM user/role ARN under the **Allow use of the key** section of the policy.

> **Note:** This is required to grant your own IAM user/role permission to use the KMS key for downloading/uploading the LZ configuration ZIP file.

```
    {
      "Sid": "Allow use of the key",
      "Effect": "Allow",
      "Principal": {
        "AWS": [
          "arn:aws:iam::xxxxxxxxxxxx:role/Admin",
"arn:aws:iam::xxxxxxxxxxxx:role/LandingZoneHandshakeSMLambdaRole",
"arn:aws:iam::xxxxxxxxxxxx:role/LandingZoneDeploymentLambdaRole",
"arn:aws:iam::xxxxxxxxxxxx:role/StateMachineLambdaRole",
"arn:aws:iam::xxxxxxxxxxxx:role/LandingZoneLambdaRole",
"arn:aws:iam::xxxxxxxxxxxx:role/StateMachineTriggerLambdaRole",
"arn:aws:iam::xxxxxxxxxxxx:role/LandingZoneCodePipelineRole"
        ]
      }
```

6. **Create** the following temporary directories on your local machine in your preferred location: **LZ_v1** and **LZ_v2**. For Example: /temp/**LZ_v1** and /temp/**LZ_v2**
7. Download and Unzip the **LZ_v2** Configuration Zip file and place into LZ_v2 directory. For Example: /temp/LZ_v2 directory
8. **Download** and **Unzip** the **LZ_v1** Configuration ZIP file (`aws-landing-zone-configuration.zip`) from your LZ Configuration S3 Bucket (i.e. `aws-landing-zone-configuration-<ACCOUNT_ID>-<REGION>`) into the **LZ_v1** directory. For Example: /temp/**LZ_v1** directory

> In the steps below, make sure you are editing and/or copying the configuration files into /temp/**LZ_v1** directory

9. If you have not customized the `templates/aws_baseline/aws-landing-zone-avm.template.j2` file, replace it with the linked file.

If you have customized the file, **compare** the `templates/aws_baseline/aws-landing-zone-avm.template.j2` file between **LZ_v1** and **LZ_v2** in text editor and edit the following lines shown below. Check to make sure any other customizations that you made before are not overwritten with the changes below. Remove the lines highlighted in red and add the lines highlighted in green below:

```
455  455
456  456  #
457  457  # VPC Custom Resource - Get SSM Parameter Values
458  458  #
459  459
460       VPCParameters:
     460  SSMGetParameters:
461  461    Type: Custom::SSMParameters
462  462    Properties:
463            SSMParameterKeys: # TODO, insert with jinja
     463       SSMParameterKeys:
464  464        - /org/member/sharedservices/vpc_private_route_ids
465  465        - /org/member/sharedservices/vpc_id
466  466        - /org/member/sharedservices/account_id
467  467        - /org/member/sharedservices/vpc_region
     468        - /org/member/security/account_id
     469        - /org/member/logging/account_id
468  470      ServiceToken: {{ lambda_arn }}
```

```
487  523  #
488  524  # VPC Custom Resource - Peering
489  525  #
490  526
491       VPCPeering:
     527  VPCPeeringCR:
492  528    Condition: PeerVPC
493       Type: Custom::VPCPeering
     529  Type: Custom::HandShakeStateMachine
494  530    DependsOn:
495  531      - StackSetPrimaryVPC
496  532    Properties:
497       PeeringConnectionKeyPrefix: /org/member/sharedservices
498       PeeringAccountID: !GetAtt 'VPCParameters./org/member/sharedservices/account_id'
499       PeeringVPCID: !GetAtt 'VPCParameters./org/member/sharedservices/vpc_id'
500       PeeringRegion: !GetAtt 'VPCParameters./org/member/sharedservices/vpc_region'
501       AccountName: !Ref AccountName
502       AccountID : !GetAtt 'Organizations.AccountId'
503       VPCID : !GetAtt 'StackSetPrimaryVPC.output_vpcid'
504       Region: !Ref VPCRegion
     533  ServiceType: VPCPeering
     534  HubAccountId: !GetAtt 'SSMGetParameters./org/member/sharedservices/account_id'
     535  HubRegion: !GetAtt 'SSMGetParameters./org/member/sharedservices/vpc_region'
     536  HubVPCId: !GetAtt 'SSMGetParameters./org/member/sharedservices/vpc_id'
     537  SpokeAccountId: !GetAtt 'Organizations.AccountId'
     538  SpokeVPCId: !GetAtt 'StackSetPrimaryVPC.output_vpcid'
     539  SpokeRegion: !Ref VPCRegion
505  540       ServiceToken: {{ lambda_arn }}
```

```
511  546  NewVPCPeerRouting:
512  547    Condition: PeerVPC
513  548    Type: Custom::VPCPeering
514  549    DependsOn:
515  550      - StackSetPrimaryVPC
516       - VPCPeering
     551  - VPCPeeringCR
517  552    Properties:
518  553      AccountID : !GetAtt 'Organizations.AccountId'
519  554      Region: !Ref VPCRegion
520  555      RouteTableIDs: !GetAtt 'StackSetPrimaryVPC.output_privatesubnetroutetables'
521       PeerConnectionID : !GetAtt 'VPCPeering.PeerConnectionID'
522       VPCCIDR: !GetAtt 'VPCPeering.RequesterVPCCIDR'
     556  PeerConnectionID : !GetAtt 'VPCPeeringCR.ConnectionId'
     557  VPCCIDR: !GetAtt 'VPCPeeringCR.HubVPCCIDR'
523  558      ServiceToken: {{ lambda_arn }}
```

```
525  560   SharedVPCPeerRouting:
526  561     Condition: PeerVPC
527  562     Type: Custom::VPCPeering
528  563     DependsOn:
529  564       - StackSetPrimaryVPC
530            - VPCPeering
     565       - VPCPeeringCR
531  566     Properties:
532            AccountID: !GetAtt 'VPCPeering.RequesterAccountID'
533            Region: !GetAtt 'VPCParameters./org/member/sharedservices/vpc_region'
534            RouteTableIDs: !GetAtt 'VPCParameters./org/member/sharedservices/vpc_private_route_ids'
535            PeerConnectionID : !GetAtt 'VPCPeering.PeerConnectionID'
536            VPCCIDR: !GetAtt 'VPCPeering.AccepterVPCCIDR'
     567       AccountID: !GetAtt 'SSMGetParameters./org/member/sharedservices/account_id'
     568       Region: !GetAtt 'SSMGetParameters./org/member/sharedservices/vpc_region'
     569       RouteTableIDs: !GetAtt 'SSMGetParameters./org/member/sharedservices/vpc_private_route_ids'
     570       PeerConnectionID : !GetAtt 'VPCPeeringCR.ConnectionId'
     571       VPCCIDR: !GetAtt 'VPCPeeringCR.SpokeVPCCIDR'
537  572       ServiceToken: {{ lambda_arn }}
```

> If you plan to perform the **Strongly Recommended** steps, skip the following steps, and go straight to the Strongly Recommended section.

10. **Zip** all the files under the **/temp/LZ_v1** directory as a `aws-landing-zone-configuration.zip` file.
11. **Upload** the `aws-landing-zone-configuration.zip` to your LZ Configuration Amazon S3 Bucket (i.e. `aws-landing-zone-configuration-<ACCOUNT_ID>-<REGION>`). This will kick off the LZ CodePipeline.
12. Wait for the LZ CodePipeline to successfully finish.

Highly Recommended Upgrade Steps

These upgrade steps update your existing Landing Zone configuration to enable Amazon GuardDuty, AWS Config, & AWS Config rules in ALL AWS Regions and apply a bugfix to the IamPasswordPolicy.

> Note: You must perform the Mandatory Upgrade, before starting this upgrade.

1. **Compare** the `templates/aws_baseline/aws-landing-zone-avm.template.j2` file between **LZ_v1** and **LZ_v2** in text editor.
2. **Copy** and **Paste** the following highlighted lines below at the end of the file before the **Outputs** section from **LZ_v2** to **LZ_v1**.

```
{%- for ou in manifest.organizational_units %}
{%- for account in ou.core_accounts %}
{%- for resource in account.core_resources %}
{%- if 'guardduty' in resource.name.lower() %}
  {% if resource.regions %}
    {% set region_list =  resource.regions %}
  {% else %}
    {% set region_list =  [manifest.region] %}
  {%- endif %}
{%- for region in region_list %}

#
# GuardDuty Custom Resource - {{ region }} (depends on release/v2.0)
#

  GuardDutyMemberof{{ account.name.title() }}Account{{region.title() | replace("-","") }}:
    DependsOn:
      - Organizations
      - DetachSCP
    Type: Custom::HandShakeStateMachine
    Properties:
      ServiceType: GuardDuty
      HubAccountId: !GetAtt 'SSMGetParameters./org/member/{{ account.name }}/account_id'
      HubRegion: {{ region }}
      SpokeAccountId: !GetAtt 'Organizations.AccountId'
      SpokeRegion: {{ region }}
      SpokeEmailId: !Ref AccountEmail
      ServiceToken: {{ lambda_arn }}

{%- endfor %}
{%- endif %}
{%- endfor %}
{%- endfor %}
{%- endfor %}
```

3. **Compare** the `manifest.yaml` file between **LZ_v1** and **LZ_v2** in text editor.
4. **Copy** and **Paste** the following highlighted lines below, under the **core_resources:SharedTopic** section from **LZ_v2** to **LZ_v1**.

```
        - name: SharedTopic
          template_file: templates/core_accounts/aws-landing-zone-notification.template
          parameter_file: parameters/core_accounts/aws-landing-zone-notification.json
          deploy_method: stack_set
          regions:
            - ap-northeast-1
            - ap-northeast-2
            - ap-south-1
            - ap-southeast-1
            - ap-southeast-2
            - ca-central-1
            - eu-central-1
            - eu-west-1
            - eu-west-2
            - eu-west-3
            - sa-east-1
            - us-east-1
            - us-east-2
            - us-west-1
            - us-west-2
```

5. **Compare** the `manifest.yaml` file between **LZ_v1** and **LZ_v2** in text editor.
6. **Copy** and **Paste** the following highlighted lines below, under the **core_resources:security** section from **LZ_v2** to **LZ_v1**.

```
55          - name: GuardDutyMaster
56            template_file: templates/core_accounts/aws-landing-zone-guardduty-master.template
57            parameter_file: parameters/core_accounts/aws-landing-zone-guardduty-master.json
58            deploy_method: stack_set
59            regions:
60              - ap-south-1
61              - eu-west-3
62              - eu-west-2
63              - eu-west-1
64              - ap-northeast-2
65              - ap-northeast-1
66              - sa-east-1
67              - ca-central-1
68              - ap-southeast-1
69              - ap-southeast-2
70              - eu-central-1
71              - us-east-1
72              - us-east-2
73              - us-west-1
74              - us-west-2
```

7. **Compare** the `manifest.yaml` file between **LZ_v1** and **LZ_v2** in text editor.
8. **Copy** and **Paste** the following highlighted lines below, under the **baseline_resources** section from **LZ_v2** to **LZ_v1**.

```
168 # Landing Zone Service Baseline Resources
169 baseline_resources:
170   - name: EnableCloudTrail
171 ...
172     template_file: templates/aws_baseline/aws-landing-zone-enable-cloudtrail.template
173     parameter_file: parameters/aws_baseline/aws-landing-zone-enable-cloudtrail.json
174     deploy_method: stack_set
175
176     # This template deploys the ConfigRecorder IAM role required for enabling AWS Config service
177     # It needs to be deployed in Home region ONLY
178   - name: ConfigRole
179     baseline_products:
180       - AWS-Landing-Zone-Account-Vending-Machine
181     template_file: templates/aws_baseline/aws-landing-zone-enable-config-role.template
182     deploy_method: stack_set
```

9. **Compare** the `manifest.yaml` file between **LZ_v1** and **LZ_v2** in text editor.
10. **Copy** and **Paste** the following highlighted lines below, under the **baseline_resources:EnableConfig** section from **LZ_v2** to **LZ_v1**. Doing so, adds a dependency on ConfigRole and all regions for AWS Config.

```
- name: EnableConfig
  baseline_products:
    - AWS-Landing-Zone-Account-Vending-Machine
  depends_on:
   - ConfigRole
  template_file: templates/aws_baseline/aws-landing-zone-enable-config.template
  parameter_file: parameters/aws_baseline/aws-landing-zone-enable-config.json
  deploy_method: stack_set
  regions:
    - ap-northeast-1
    - ap-northeast-2
    - ap-south-1
    - ap-southeast-1
    - ap-southeast-2
    - ca-central-1
    - eu-central-1
    - eu-west-1
    - eu-west-2
    - eu-west-3
    - sa-east-1
    - us-east-1
    - us-east-2
    - us-west-1
    - us-west-2
```

11. **Compare** the `manifest.yaml` file between **LZ_v1** and **LZ_v2** in text editor.
12. **Copy** and **Paste** the following highlighted lines below, under the **baseline_resources** section from **LZ_v2** to **LZ_v1**.

```
  # This template deploys the Config Rules that monitor the Global resources i.e. IAM
  # It needs to be deployed in Home region ONLY
- name: EnableConfigRulesGlobal
  baseline_products:
    - AWS-Landing-Zone-Account-Vending-Machine
  depends_on:
    - EnableConfig
  template_file: templates/aws_baseline/aws-landing-zone-config-rules-global.template
  parameter_file: parameters/aws_baseline/aws-landing-zone-config-rules-global.json
  deploy_method: stack_set
```

13. **Compare** the `manifest.yaml` file between **LZ_v1** and **LZ_v2** in text editor.
14. **Copy** and **Paste** the following highlighted lines below, under the **baseline_resources:EnableConfigRules** section from **LZ_v2** to **LZ_v1**. Doing so, adds all regions for AWS Config Rules.

```
  - name: EnableConfigRules
    baseline_products:
      - AWS-Landing-Zone-Account-Vending-Machine
...
    template_file: templates/aws_baseline/aws-landing-zone-config-rules.template
    parameter_file: parameters/aws_baseline/aws-landing-zone-config-rules.json
    deploy_method: stack_set
    regions:
      - ap-northeast-1
      - ap-northeast-2
      - ap-south-1
      - ap-southeast-1
      - ap-southeast-2
      - ca-central-1
      - eu-central-1
      - eu-west-1
      - eu-west-2
      - eu-west-3
      - sa-east-1
      - us-east-1
      - us-east-2
      - us-west-1
      - us-west-2
```

15. **Compare** the `manifest.yaml` file between **LZ_v1** and **LZ_v2** in text editor.
16. **Copy** and **Paste** the following highlighted lines below, under the **baseline_resources:EnableNotifications** section from **LZ_v2** to **LZ_v1**. Doing so, adds a dependency on EnableConfig.

```
      - AWS-Landing-Zone-Account-Vending-Machine
    depends_on:
      - EnableCloudTrail
      - EnableConfig
    template_file: templates/aws_baseline/aws-landing-zone-notifications.template
    parameter_file: parameters/aws_baseline/aws-landing-zone-notifications.json
    deploy_method: stack_set
```

17. If you haven't modified the configuration files, Copy the files from **/temp/ LZ_v2** to **/temp/ LZ_v1**.
18. If you have modified the configuration files, manually update the files in **/temp/ LZ_v1** by comparing the corresponding file in **/temp/ LZ_v2**.

- templates/aws_baseline/aws-landing-zone-enable-config.template
- templates/aws_baseline/aws-landing-zone-config-rules.template
- templates/aws_baseline/aws-landing-zone-notifications.template
- parameters/aws_baseline/aws-landing-zone-enable-config.json
- parameters/aws_baseline/aws-landing-zone-config-rules.json

- parameters/aws_baseline/aws-landing-zone-notifications.json

19. **Add** the following configuration files from **/temp/ LZ_v2** to **/temp/ LZ_v1**:

- templates/core_accounts/aws-landing-zone-guardduty-master.template
- templates/aws_baseline/aws-landing-zone-enable-config-role.template
- templates/aws_baseline/aws-landing-zone-config-rules-global.template
- parameters/core_accounts/aws-landing-zone-guardduty-master.json
- parameters/aws_baseline/aws-landing-zone-config-rules-global.json

20. In the **/temp/ LZ_v1/`manifest.yaml`** files, verify the order of the Baseline resources are as follows:

- EnableCloudTrail
- ConfigRole
- EnableConfig [depends_on: ConfigRole]
- EnableConfigRulesGlobal [depends_on: EnableConfig]
- EnableConfigRules [depends_on: EnableConfig]
- EnableNotifications [depends_on: EnableCloudTrail, EnableConfig]
- SecurityRoles
- IamPasswordPolicy
- PrimaryVPC

21. **Remove** the validation schema and scripts from **/temp/ LZ_v1/** by deleting the directory and files inside the **validation/** directory.

22. **Perform** the following bug fix for the IAM Password Policy template: `templates/aws_baseline/aws-landing-zone-iam-password-policy.template`

    a. If you haven't modified the configuration files, copy the files from **/temp/ LZ_v2/...** to **/temp/ LZ_v1/...**

    b. If you have modified the configuration file, **compare** the `templates/aws_baseline/aws-landing-zone-iam-password-policy.template` file between **LZ_v1** and **LZ_v2** in text editor and edit the following lines shown below

```
    if (err) {
      console.log(`Error: ${!JSON.stringify(err)}`);
      response.send(event, context, response.FAILED, {});
      response.send(event, context, response.FAILED, {}, 'CustomResourcePhysicalID');
    } else {
      response.send(event, context, response.SUCCESS, {});
      response.send(event, context, response.SUCCESS, {}, 'CustomResourcePhysicalID');
    }
  };
  if (event.RequestType === 'Delete') {
    iam.deleteAccountPasswordPolicy({}, done);
  } else if (event.RequestType === 'Create' || event.RequestType === 'Update') {
    iam.updateAccountPasswordPolicy({
      AllowUsersToChangePassword: ${AllowUsersToChangePassword},
      HardExpiry: ${HardExpiry},
      MaxPasswordAge: ${MaxPasswordAge},
      MinimumPasswordLength: ${MinimumPasswordLength},
      PasswordReusePrevention: ${PasswordReusePrevention},
      RequireLowercaseCharacters: ${RequireLowercaseCharacters},
      RequireNumbers: ${RequireNumbers},
      RequireSymbols: ${RequireSymbols},
      RequireUppercaseCharacters: ${RequireUppercaseCharacters},
      AllowUsersToChangePassword: Boolean(event.ResourceProperties.AllowUsersToChangePassword),
      HardExpiry: Boolean(event.ResourceProperties.HardExpiry),
      MaxPasswordAge: event.ResourceProperties.MaxPasswordAge,
      MinimumPasswordLength: event.ResourceProperties.MinimumPasswordLength,
      PasswordReusePrevention: event.ResourceProperties.PasswordReusePrevention,
      RequireLowercaseCharacters: Boolean(event.ResourceProperties.RequireLowercaseCharacters),
      RequireNumbers: Boolean(event.ResourceProperties.RequireNumbers),
      RequireSymbols: Boolean(event.ResourceProperties.RequireSymbols),
      RequireUppercaseCharacters: Boolean(event.ResourceProperties.RequireUppercaseCharacters),
```

```
DependsOn: LambdaLogGroup
Version: '1.0'
Properties:
  HardExpiry: !Ref HardExpiry
  AllowUsersToChangePassword: !Ref AllowUsersToChangePassword
  MaxPasswordAge: !Ref MaxPasswordAge
  MinimumPasswordLength: !Ref MinimumPasswordLength
  PasswordReusePrevention: !Ref PasswordReusePrevention
  RequireLowercaseCharacters: !Ref RequireLowercaseCharacters
  RequireNumbers: !Ref RequireNumbers
  RequireSymbols: !Ref RequireSymbols
  RequireUppercaseCharacters: !Ref RequireUppercaseCharacters
  ServiceToken: !GetAtt 'IamPasswordPolicyCustomResource.Arn'
```

23. **Zip** all the files under the **/temp/ LZ_v1/** directory as `aws-landing-zone-configuation.zip`.
24. **Upload** the `aws-landing-zone-configuation.zip` to your LZ Configuration Amazon S3 bucket (i.e. `aws-landing-zone-configuration-<ACCOUNT_ID>-<REGION>`). This will kick off the LZ CodePipeline.
25. Wait for the LZ CodePipeline to successfully finish.

26. In order to force the stack updates for AWS-Landing-Zone-Baseline-IamPasswordPolicy stack, you must change one of the input parameters in `/temp/LZ_v1/parameters/aws_baseline/aws-landing-zone-iam-password-policy.json` (i.e. MinimumPasswordLength = 13).

> If you plan to perform the **Nice to Have** steps, skip the following steps, and go straight to the Nice to Have Upgrade section.

27. **Zip** all the files under the **/temp/LZ_v1** directory as a `aws-landing-zone-configuration.zip` file.
28. **Upload** the `aws-landing-zone-configuration.zip` to your LZ Configuration Amazon S3 Bucket (i.e. `aws-landing-zone-configuration-<ACCOUNT_ID>-<REGION>`). This will kick off the LZ CodePipeline.
29. Wait for the LZ CodePipeline to successfully finish.

Nice to Have Upgrade Steps

These upgrade steps update your existing Landing Zone configuration to updates the Network Type in Account Vending Machine (AVM), Optional Core products cleanup and updates for VPC, RDGW & AD Connector templates

> **Note:** You must perform the Mandatory Upgrade and Strongly Recommended Upgrade before starting this upgrade.

1. **Perform** the updates for Network Types in AVM template:

> **Important**: This change will require to one-time manual update to the Service Catalog provisioned AVM products that were provisioned **with a VPC**. Since it changes the **AllowedValues** for the Network Type, you will need to update the provisioned product with the corresponding new Network Type.

Below is the table mapping the old values to new values for Network Type.

Old Network Type AllowedValues	New Network Type AllowedValues
No-Primary-VPC	No-Primary-VPC
1-Tier-2-AZ-Public-VPC	Public-Only-2-AZ
1-Tier-3-AZ-Public-VPC	Public-Only-3-AZ
1-Tier-4-AZ-Public-VPC	Public-Only-4-AZ
1-Tier-2-AZ-Private-VPC	Private-Only-2-AZ
1-Tier-3-AZ-Private-VPC	Private-Only-3-AZ
1-Tier-4-AZ-Private-VPC	Private-Only-4-AZ

Old Network Type AllowedValues	New Network Type AllowedValues
2-Tier-2-AZ-Public-Private-VPC	Public-and-Private-Subnets-2-AZ
2-Tier-3-AZ-Public-Private-VPC	Public-and-Private-Subnets-3-AZ
2-Tier-4-AZ-Public-Private-VPC	Public-and-Private-Subnets-4-AZ
3-Tier-2-AZ-Public-Private-Private-VPC	Public-and-2-Private-Subnets-2-AZ
3-Tier-3-AZ-Public-Private-Private-VPC	Public-and-2-Private-Subnets-3-AZ
3-Tier-4-AZ-Public-Private-Private-VPC	Public-and-2-Private-Subnets-4-AZ

2. **Compare** the templates `/aws_baseline/aws-landing-zone-avm.template.j2` file between **LZ_v1** and **LZ_v2** in text editor.
3. **Remove** the lines highlighted below in red, and **add** the lines highlighted below in green, to update the AllowedValues and Mappings for the VPC.

```
Description: VPC options
AllowedValues:
  - No-Primary-VPC
  - 1-Tier-2-AZ-Public-VPC
  - 1-Tier-3-AZ-Public-VPC
  - 1-Tier-4-AZ-Public-VPC
  - 1-Tier-2-AZ-Private-VPC
  - 1-Tier-3-AZ-Private-VPC
  - 1-Tier-4-AZ-Private-VPC
  - 2-Tier-2-AZ-Public-Private-VPC
  - 2-Tier-3-AZ-Public-Private-VPC
  - 2-Tier-4-AZ-Public-Private-VPC
  - 3-Tier-2-AZ-Public-Private-Private-VPC
  - 3-Tier-3-AZ-Public-Private-Private-VPC
  - 3-Tier-4-AZ-Public-Private-Private-VPC
  - Public-Only-2-AZ
  - Public-Only-3-AZ
  - Public-Only-4-AZ
  - Private-Only-2-AZ
  - Private-Only-3-AZ
  - Private-Only-4-AZ
  - Public-and-Private-Subnets-2-AZ
  - Public-and-Private-Subnets-3-AZ
  - Public-and-Private-Subnets-4-AZ
  - Public-and-2-Private-Subnets-2-AZ
  - Public-and-2-Private-Subnets-3-AZ
  - Public-and-2-Private-Subnets-4-AZ
```

```
VPC:
  1-Tier-2-AZ-Public-VPC:
  Public-Only-2-AZ:
    AvailabilityZones: 2
    PublicSubnets:
        - PublicSubnet1CIDR
...
    CreateAdditionalPrivateSubnets: 'false'
    CreatePrivateSubnets: 'false'
    CreatePublicSubnets: 'true'
  1-Tier-3-AZ-Public-VPC:
  Public-Only-3-AZ:
    AvailabilityZones: 3
    PublicSubnets:
        - PublicSubnet1CIDR
...
    CreateAdditionalPrivateSubnets: 'false'
    CreatePrivateSubnets: 'false'
    CreatePublicSubnets: 'true'
  1-Tier-4-AZ-Public-VPC:
  Public-Only-4-AZ:
    AvailabilityZones: 4
    PublicSubnets:
        - PublicSubnet1CIDR
...
    CreateAdditionalPrivateSubnets: 'false'
    CreatePrivateSubnets: 'false'
    CreatePublicSubnets: 'true'
  1-Tier-2-AZ-Private-VPC:
  Private-Only-2-AZ:
    AvailabilityZones: 2
    PublicSubnets: []
```

```
  1-Tier-3-AZ-Private-VPC:
  Private-Only-3-AZ:
    AvailabilityZones: 3
    PublicSubnets: []
    PrivateSubnets:
...
    CreateAdditionalPrivateSubnets: 'false'
    CreatePrivateSubnets: 'true'
    CreatePublicSubnets: 'false'
  1-Tier-4-AZ-Private-VPC:
  Private-Only-4-AZ:
    AvailabilityZones: 4
    PublicSubnets: []
    PrivateSubnets:
...
    CreateAdditionalPrivateSubnets: 'false'
    CreatePrivateSubnets: 'true'
    CreatePublicSubnets: 'false'
  2-Tier-2-AZ-Public-Private-VPC:
  Public-and-Private-Subnets-2-AZ:
    AvailabilityZones: 2
    PublicSubnets:
        - PublicSubnet1CIDR
...
    CreateAdditionalPrivateSubnets: 'false'
    CreatePrivateSubnets: 'true'
    CreatePublicSubnets: 'true'
  2-Tier-3-AZ-Public-Private-VPC:
  Public-and-Private-Subnets-3-AZ:
    AvailabilityZones: 3
    PublicSubnets:
```

```
    CreatePublicSubnets: 'true'
  2-Tier-4-AZ-Public-Private-VPC:
  Public-and-Private-Subnets-4-AZ:
    AvailabilityZones: 4
    PublicSubnets:
      - PublicSubnet1CIDR
...
    CreateAdditionalPrivateSubnets: 'false'
    CreatePrivateSubnets: 'true'
    CreatePublicSubnets: 'true'
  3-Tier-2-AZ-Public-Private-Private-VPC:
  Public-and-2-Private-Subnets-2-AZ:
    AvailabilityZones: 2
    PublicSubnets:
      - PublicSubnet1CIDR
...
    CreateAdditionalPrivateSubnets: 'true'
    CreatePrivateSubnets: 'true'
    CreatePublicSubnets: 'true'
  3-Tier-3-AZ-Public-Private-Private-VPC:
  Public-and-2-Private-Subnets-3-AZ:
    AvailabilityZones: 3
    PublicSubnets:
      - PublicSubnet1CIDR
...
    CreateAdditionalPrivateSubnets: 'true'
    CreatePrivateSubnets: 'true'
    CreatePublicSubnets: 'true'
  3-Tier-4-AZ-Public-Private-Private-VPC:
  Public-and-2-Private-Subnets-4-AZ:
    AvailabilityZones: 4
```

4. Under the **/temp/LZ_v1/** directory, **create** a new folder named `template_constraints`.
5. **Copy** the `template_constraints/aws-landing-zone-avm-rules.json` file from **/temp/LZ_v2/** to **/temp/LZ_v1/** directory.
6. **Compare** the `manifest.yaml` file between **LZ_v1** and **LZ_v2** in text editor.
7. **Edit** the following highlighted lines below, under the **portfolios:section**, look for the name **AWSLandingZone - Baseline**, and add the **rules_file**.

```
153   description: Baseline Products for AWS Landing Zone
154   owner: AWS Solutions
155 ...
156       # This is the skeleton template for the AVM
157       skeleton_file: templates/aws_baseline/aws-landing-zone-avm.template.j2
158       parameter_file: parameters/aws_baseline/aws-landing-zone-avm.json
159       rules_file: template_constraints/aws-landing-zone-avm-rules.json
160       # Hide/Disable the old version of the product in Service Catalog
161       hide_old_versions: true
162       # Is this is a baseline product? e.g. AVM ?
```

8. Perform the Optional Core Products cleanup:

> If you have **NOT** deployed the LZ v1.0 **Optional Core Products**, perform the following clean up steps, otherwise skip this step.

   a. Update the LZ_v1/ `manifest.yaml` file by removing AWS Landing Zone-Core from baseline_resources in the following sections:

- Portfolios Section

```
212 212 # Landing Zone Service Catalog portolfios/products (Optional/Baseline)
213 213 portfolios:
214       - name: AWS Landing Zone - Core
215         description: Optional Core Products
216         owner: AWS Solutions
217         principal_role: ${alfred_ssm_/org/primary/service_catalog/principal/role_arn}
218         # These products will prompt the user to select target Account Email and Region
219         products:
220           - name: AWS Centralized Logging Solution
221             description: Install the centralized log aggregation and monitoring solution
222             template_file: templates/optional_products/aws-landing-zone-centralized-logging-primary.template
223             skeleton_file: templates/optional_products/aws-landing-zone-centralized-logging-primary-skeleton.template.j2
224             ssm_parameters:
225               - name: /org/member/centrallogging/es_domain
226                 value: ${output_DomainEndpoint}
227               - name: /org/member/centrallogging/master_role
228                 value: ${output_MasterRole}
229               - name: /org/member/centrallogging/kibana_url
230                 value: ${output_KibanaLoginURL}
231             # Hide/Disable the old version of the product in Service Catalog
232             hide_old_versions: true
233             # Is this is a baseline product? e.g. AVM ?
234             product_type: optional
235             launch_constraint_role: ${alfred_ssm_/org/primary/service_catalog/constraint/role_arn}
236 214   - name: AWS Landing Zone - Baseline
237 215     description: Baseline Products for AWS Landing Zone
238 216     owner: AWS Solutions
239 217     principal_role: ${alfred_ssm_/org/primary/service_catalog/principal/role_arn}
240 218     products:
```

- CentralizedLoggingSpoke section

```
366 344     parameter_override: true
367
368       # Uncomment these lines to install the Spoke template in every member account for the Centralized Logging Solution
369       # - name: CentralizedLoggingSpoke
370       #   baseline_products:
371       #     - AWS-Landing-Zone-Account-Vending-Machine
372       #   template_file: templates/aws_baseline/aws-landing-zone-centralized-logging-spoke.template
373       #   parameter_file: parameters/aws_baseline/aws-landing-zone-centralized-logging-spoke.json
374       #   deploy_method: stack_set
```

b. **Delete** the following files from the **/temp/LZ_v1/** directory:

- parameters/aws_baseline/aws-landing-zone-centralized-logging-spoke.json
- templates/aws_baseline/aws-landing-zone-centralized-logging-spoke.template
- templates/optional_products/aws-landing-zone-centralized-logging-primary-skeleton.template.j2
- templates/optional_products/aws-landing-zone-centralized-logging-primary.template
- templates/optional_products

c. Manually delete **AWS Landing Zone - Core** from the Service Catalog Portfolio, and delete the Product: **AWS Centralized Logging Solution** from the Service Catalog Console.

> Note: Do not remove the similar product AWS Centralized Logging Solution-Landing Zone Add-On created by v2.0.

9. Perform the VPC template updates:

a. **Copy** the `templates/aws_baseline/aws-landing-zone-vpc.template` from **/temp/LZ_v2/** to **/temp/LZ_v1/** directory.

b. **Edit** the parameters/core_accounts/aws-landing-zone-primary-account-vpc.json file to remove the **NATInstanceType** parameter.

```
28  28    {
29          "ParameterKey": "NATInstanceType",
30          "ParameterValue": "t2.small"
31      },
32      {
33  29      "ParameterKey": "PrivateSubnet1ACIDR",
```

c. **Edit** the parameters/core_accounts/aws-landing-zone-shared-services-vpc.json file to remove the **NATInstanceType** parameter.

```
24  24    {
25  25        "ParameterKey": "CreatePublicSubnets",
26  26        "ParameterValue": "true"
27  27    },
28  28    {
29          "ParameterKey": "NATInstanceType",
30          "ParameterValue": "t2.small"
31      },
32      {
33  29      "ParameterKey": "PrivateSubnet1ACIDR",
```

10. **Perform** the RDGW and AD Connector template updates:

> **Note:** If your current deployment of Landing Zone was deployed with Managed AD & RDGW (Shared-services) and AD Connector (Primary) and you configured AWS SSO, this change will require you to disconnect AWS SSO from Managed AD. Additionally, at the end of deployment you will need to reconfigure AWS SSO again. This update will retain the users/groups configured in Managed AD, but will update the RDGW Launch Configuration and AD Connector deployments.

a. **Verify** if the SSM parameter: **/org/directory_service/connector_password** has the up-to-date password for the **connector user**, used by AD Connector in primary account to connect to the Managed AD in shared-services account. If not, update the password for AD Connector in primary account to match the SSM parameter**:/org/directory_service/connector_password**

b. Update the /temp/LZ_v1/`manifest.yaml` file to remove the highlighted lines below from core_resources: SharedServicesRDGW and core_resources: PrimaryADConnector.

```
              value: $[output_DomainMemberSGID]
          - name: SharedServicesRDGW
            template_file: templates/core_accounts/aws-landing-zone-rdgw.template
            parameter_file: parameters/core_accounts/aws-landing-zone-rdgw.json
            deploy_method: stack_set
            regions:
              - us-east-1
            ssm_parameters:
              - name: /org/member/sharedservices/rdgw_ip1
                value: $[output_EIP1]
      # Organization's Master account
      - name: primary  # NOTE: DO NOT MODIFY THIS ACCOUNT NAME AND IT SHOULD BE THE LAST CORE ACCOUNT IN THE LIST
        ssm_parameters:
...
              value: $[output_PrivateSubnet2AID]
            - name: /org/primary/vpc_private_route_ids
              value: $[output_PrivateSubnetRouteTables]
          - name: PrimaryADConnector
            template_file: templates/core_accounts/aws-landing-zone-aws-ad-connector.template
            parameter_file: parameters/core_accounts/aws-landing-zone-aws-ad-connector.json
            deploy_method: stack_set
            regions:
              - us-east-1
  - name: applications
    include_in_baseline_products:
```

c. **Remove** the following configuration files from **temp/LZ_v1/…**:

- templates/core_accounts/aws-landing-zone-rdgw.template
- parameters/core_accounts/aws-landing-zone-rdgw.json
- templates/core_accounts/aws-landing-zone-aws-ad-connector.template
- parameters/core_accounts/aws-landing-zone-aws-ad-connector.json

11. **Zip** all the files under the **temp/LZ_v1/**directory as a `aws-landing-zone-configuration.zip`
12. **Upload** the `aws-landing-zone-configuration.zip` to your LZ Configuration Amazon S3 Bucket (i.e. aws-landing-zone-configuration-<ACCOUNT_ID>-<REGION>). This will kick off the LZ CodePipeline.
13. Wait for the LZ CodePipeline to successfully finish.
14. **Note:** If you performed the steps for RDGW & AD Connector template updates, then deploy the **AWS Managed AD and Directory Connector for AWS SSO - Landing Zone Add-On product** from AWS Service Catalog. Please refer to the AWS Landing Zone User guide Add-On Productions section for detailed instructions.

Appendix A: Update the AMI ID for Auto Scaling Group Launch Configuration

This change is applicable to the `AWS-Landing-Zone-SharedServicesRDGW` stack set that deploys Auto Scaling group Launch Configuration. The easiest option is to update the Stack Set parameter that would update the launch configuration during stack update.

1. Navigate to the AWS CloudFormation console.
2. Choose **StackSets** and select the **AWS-Landing-Zone-SharedServicesRDGW** stack set.
3. On the **Actions** menu, choose **Edit StackSet details**
4. On the **Choose a template** page, choose whether you want to update the current template, specify an S3 URL to another template, or upload a new template to AWS CloudFormation.

In the following procedure, we selected the current template.

5. Choose **Use current template**, and then select **Next**.
6. On the **Specify StackSet details** page, modify parameter values and specify deployment targets.
7. Change the value of the **RDGWInstanceType** parameter from current instance size to a different size temporarily. For example, change it from **t2.micro** to **t2.large**. Then, choose **Next**. This will also update the launch configuration with the latest AMI ID.
8. On the **Configure StackSet options** page, no changes are needed, but you can update, delete, or add new tags here if desired.
9. On the **Set deployment options** page, keep the default value of **1** and **By number** for **Maximum concurrent accounts**. Keep the default **Failure tolerance** of **0**, and keep the **By number** default option. Choose **Next**.
10. On the **Review** page, review your choices and your stack set properties. To make changes, choose **Edit** in the upper-right corner of an area where you want to change properties.

    Before you can update the stack set, you must select the check box in the **Capabilities** section to acknowledge that some of the resources that you are updating with the stack set might require new IAM resources and permissions. For more information about required permissions, see Acknowledging IAM Resources in *AWS CloudFormation Templates guide*.
11. When you are are ready to create your stack set, select **Submit**.

AWS CloudFormation will automatically apply your updates to your stack set. You can view the progress and status of update operations on the **Operations** tab, and view the updated **RDGWInstanceType** parameter in the **Parameter** tab.

Once the stack set operation has SUCCEEDED. The parameter value can be rolled back to the original value to avoid extra cost in case the instance size was increased. To rollback the changes follow the previous steps.