OCI MAP Foundations Remediation

Introduction

This ZIP file provides terraform code and a Resource Manager UI that can be used to remediate the following MAP Findings:

Logging Monitoring and Alerting Events and Notifications:

- LMA6 - Notifications enabled - IAM/Network
- LMA7 - Notification sent to correct people
- LMA8 - Connectivity alerting & monitoring enabled
- LMA9 - Alerts sent to correct people

Cloud Security and Posture Management - Cloud Guard:

- SPM1 - CSPM integrated with tenancy
- SPM2 - Cloud Guard enabled and configured
- SPM3 - Responder - Cloud Events enabled
- SPM5 - Critical/High Alerts are sent to correct people

Cloud Governance - Budget:

- CGO2 - Root Level budget has been defined
- CGO3 - Budget is forecast based

Prerequisites

The following permissions are needed to run the stack using the resource manager.

Note: Some policies(Cloudevents-rule, Alarms, ons-family) can be further restricted and refined if required.

Please replace " `<Remediation Group>` " with an appropriate group in your tenancy.

The reference to " `<Stack Compartment>` " should be replaced within an existing compartment used for housing Security or Shared resources. The stack can also be created at the tenancy level but it is not recommended.

Resource Manager Permissions:

```
Allow group <Remediation Group> to manage orm-stacks in compartment <Stack Compartment>
Allow group <Remediation Group> to manage orm-jobs in compartment <Stack Compartment>
```

Remediation Permissions:

```
Allow group <Remediation Group> to inspect all-resources in tenancy
Allow group <Remediation Group> to read all-resources in tenancy
Allow group <Remediation Group> to manage cloud-guard-family in tenancy
Allow group <Remediation Group> to manage cloudevents-rules in tenancy
Allow group <Remediation Group> to manage usage-budgets in tenancy
Allow group <Remediation Group> to manage alarms in tenancy
Allow group <Remediation Group> to manage ons-family in tenancy
Allow group <Remediation Group> to manage policies in tenancy
```

Considerations before running the Terraform script

Stack placement:

We recommend creating the stack in a Security or Shared compartment and not in the Root compartment. The location of the stack doesn't have any effect on the resources created by the stack.

Service Label:

Consider what service label to use and if you have any existing naming convention to follow. A Service Label is a unique label that gets prepended to all resources created by this stack. Max length of 8 alphanumeric characters starting with a letter.

Multi Region deployment:

If you are subscribed to multiple regions, you will need to create a stack per region as the terraform provider works on a regional basis.

Compartment for Network Events and Alarms:

Consider what compartment should be used for Network Event Notifications and Alarms. Typically, an existing networking or shared services compartment is used.

Compartment for Security Events:

Consider what compartment should be used for Security Event Notifications. Typically, an existing security or shared services compartment is used.

Email recipients for Security Events, Network Events and Alarms:

Consider what Distribution Lists or individuals should receive emails related to :

- Security Events
- Network Events

- Connectivity Alarms

Review Cloud Guard configuration:

Review the current Cloud Guard configuration. If a target already exists at the root level, this wil cause an issue when running the Cloud Guard remediation. Evaluate if the current target can be removed and be created by the MAP Remediation.

- **Budget information:**

The budget creation requires a Monthly spending threshold. For tenancies that have existing and stable workload, the Cost Analysis tooling in OCI can show how the current monthly spending looks.

Consider, who should receive a Budget Alert when it is forecasted that the monthly threshold will be exceeded.

- **Enforced tagging strategy:**

The provisioning of resources will fail, If you have setup tags to be required at resource creations, since the stack cannot provide these tags.

How to run the stack

Start by creating a stack in the Home Region and leaving the "Home Region Deployment" box checked. This will create both global(Cloud guard, Budget and IAM Events) and regional resources(Network Events and Alarms) in the home region.

After successfully applying the stack in the home region you need to switch to the next region and create a new stack and deselect "Home Region Deployment". Provide the same service label as used in the home region and provide information for "Network Event Notifications" and "Alarms for FastConnect...".

1. Stack Creation
2. Customization
3. Plan and Apply
4. Review the Log
5. Run stack in additional non-home regions if needed
6. Accept the Email Subscriptions

Expected outcome and known issues

When deploy in your home region, you should expect Terraform to create 21 resources if all remediations have been selected. This can be validated by reviewing the Terraform Log after a successful Apply Job

Cloud Guard Errors when running the Apply Job:

You will see an API - "CreateTarget" error when running an Apply job, if you have selected the Cloud Guard remediation and Cloud Guard has already been partially configured., i.e. has a target defined at the root level. If Cloud Guard is not yet operationalized in your tenancy, evaluate if deleting any existing target is feasible and then rerunning the Apply Job. This will enable and configure Cloud Guard as recommended by Oracle.

Destroying Notification resources is slow:

This delay is due to the way the Notification API works when destroying topics. Typically, a 10 minutes delay is to be expected.

TF requires network connectivity to github.com

The remediation code references modules hosted publicly on Github - Oracle Quickstart. When Terraform initializes it will need network access to download the referenced modules. This may be relevant when using 3rd party tooling.

Detailed Log Levels are set to None:

When troubleshooting any issues when running the remediation TF in Resource Manager, we recommend enabling detailed logging in the advanced section of the Plan or Apply popup window and rerunning the job.

Terraform version errors due to version mismatch:

The TF code requires terraform 1.2.x. When using 3rd party tooling you need to ensure the matching terraform binaries are used.