



ORACLE

Oracle Cloud Security Assessment

Mutant – Oracle Cloud Infrastructure

Tenancy: MutantRogue / NewCoCobranca

January 26, Version [1.0]

Copyright © 2026, Oracle and/or its affiliates

Confidential – Oracle Restricted

Purpose statement

The purpose of the cloud security assessment is to comprehensively evaluate the organization's security posture in its cloud environment, identifying vulnerabilities and gaps that could compromise data integrity, confidentiality, and availability. This assessment aims to enhance information security maturity, strengthen defenses against cyber threats, ensure compliance with regulations, and foster a culture of security awareness. Through periodic assessments, the organization proactively mitigates risks, ensuring continuous security for cloud-configured assets and maintaining stakeholders' trust in data protection capabilities.

Disclaimer

This document in any form, software or printed matter, contains proprietary information that is the exclusive property of Oracle. Your access to and use of this confidential material is subject to the terms and conditions of your Oracle software license and service agreement, which has been executed and with which you agree to comply. This document and information contained herein may not be disclosed, copied, reproduced or distributed to anyone outside Oracle without prior written consent of Oracle. This document is not part of your license agreement, nor can it be incorporated into any contractual agreement with Oracle or its subsidiaries or affiliates.

This document is for informational purposes only and is intended solely to assist you in planning for the implementation and upgrade of the product features described. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions. The development, release, timing, and pricing of any features or functionality described in this document remains at the sole discretion of Oracle. Due to the nature of the product architecture, it may not be possible to safely include all features described in this document without risking significant destabilization of the code.

Purpose	6
Security Overview	7
Customer Isolation	7
Data Encryption	7
Security Controls	7
Visibility	7
Secure Hybrid Cloud	7
High Availability	7
Verifiably Secure Infrastructure	7
Basic Security Considerations	7
Compliance	8
CIS Security Assessment	9
Summary	9
1 - Identity and Access Management	14
1.1 Ensure service level admins are created to manage resources of particular service	14
1.2 Ensure permissions on all resources are given only to the tenancy administrator group	15
1.3 Ensure IAM administrators cannot update tenancy Administrators group	16
1.4 Ensure IAM password policy requires minimum length of 14 or greater	17
1.5 Ensure IAM password policy expires passwords within 365 days	18
1.6 Ensure IAM password policy prevents password reuse	19
1.7 MFA is enabled for all users with a console password	20
1.8 Ensure user API keys rotate within 90 days	21
1.9 Ensure user customer secret keys rotate within 90 days	21
1.10 Ensure user auth tokens rotate within 90 days or less	21
1.11 Ensure user IAM Database Passwords rotate within 90 days	22
1.12 Ensure API keys are not created for tenancy administrator users	22
1.13 Ensure all OCI IAM user accounts have a valid and current email address	23
1.14 Ensure Instance Principal authentication is used for OCI instances, OCI Cloud Databases and OCI Functions to access OCI resources	23
1.15 Ensure storage service-level admins cannot delete resources they manage	24
1.16 Ensure OCI IAM credentials unused for 45 days or more are disabled	25
1.17 Ensure there is only one active API Key for any single OCI IAM user	26
2 - Networking	27
2.1 Ensure no security lists allow ingress from 0.0.0.0/0 to port 22	27
2.2 No security lists allow ingress from 0.0.0.0/0 to port 3389	28
2.3 Ensure no network security groups allow ingress from 0.0.0.0/0 to port 22	30
2.4 Ensure no network security groups allow ingress from 0.0.0.0/0 to port 3389	31

2.5 Ensure the default security list of every VCN restricts all traffic except ICMP within VCN	32
2.6 Ensure Oracle Integration Cloud (OIC) access is restricted to allowed sources	32
2.7 Ensure Oracle Analytics Cloud (OAC) access is restricted to allowed sources or deployed within a Virtual Cloud Network	33
2.8 Ensure Oracle Autonomous Shared Databases (ADB) access is restricted to allowed sources or deployed within a Virtual Cloud Network	34
3 - Compute	35
3.1 Ensure Compute Instance Legacy Metadata service endpoint is disabled	35
3.2 Ensure Secure Boot is enabled on Compute Instance	36
3.3 Ensure In-transit Encryption is enabled on Compute Instance	37
4 - Logging and Monitoring	39
4.1 Ensure default tags are used on resources	39
4.2 Create at least one notification topic and subscription to receive monitoring alerts	39
4.3 Ensure a notification is configured for Identity Provider changes	40
4.4 Ensure a notification is configured for IdP group mapping changes	41
4.5 Ensure a notification is configured for IAM group changes	43
4.6 Ensure a notification is configured for IAM policy changes	44
4.7 Ensure a notification is configured for user changes	46
4.8 Ensure a notification is configured for VCN changes	48
4.9 Ensure a notification is configured for changes to route tables	49
4.10 Ensure a notification is configured for security list changes	51
4.11 Ensure a notification is configured for network security group changes	52
4.12 Ensure a notification is configured for changes to network gateways	54
4.13 Ensure VCN flow logging is enabled for all subnets	57
4.14 Ensure Cloud Guard is enabled in the root compartment of the tenancy	58
4.15 Ensure a notification is configured for Oracle Cloud Guard problems detected	58
4.16 Ensure customer created Customer Managed Key (CMK) is rotated at least annually	60
4.17 Write level Object Storage logging is enabled for all buckets	61
4.18 Ensure a notification is configured for Local OCI User Authentication	63
5 - Storage	63
5.1.1 Ensure no Object Storage buckets are publicly visible	63
5.1.2 Ensure Object Storage Buckets are encrypted with a Customer Managed Key (CMK)	64
5.1.3 Ensure Versioning is Enabled for Object Storage Buckets	65
5.2.1 Ensure Block Volumes are encrypted with Customer Managed Keys (CMK)	66
5.2.2 Ensure Boot Volumes are encrypted with Customer Managed Keys (CMK)	67
5.3.1 Ensure File Storage Systems are encrypted with Customer Managed Keys (CMK)	68
6 Asset Management	69
6.1 Create at least one compartment in your tenancy to store cloud resources	69
6.2 Ensure no resources are created in the root compartment	69
OCI SERVICES	71

Recommendations:	72
Vulnerability Scanning Service	72
Learn more about Vulnerability Scanning Service.	72
Data Safe	73
Cloud Guard	73
OCI Bastion	73
Security Zones	74
Vault	74

Purpose

The purpose of conducting a cloud security assessment is to evaluate an organization's information security posture comprehensively and thoroughly within its cloud environment. This process aims to identify vulnerabilities, security gaps, and inadequate practices that may compromise the integrity, confidentiality, and availability of data and systems hosted in the cloud.

By undertaking this assessment, the organization seeks to enhance its maturity in information security, strengthening its defenses against cyber threats and ensuring compliance with regulations and security standards. Additionally, the process provides valuable insights for the development and implementation of risk mitigation strategies specific to the cloud environment.

The benefits of a cloud security assessment are manifold. Not only does it help protect the organization's digital assets against emerging threats, but it also contributes to building a strong security culture, promoting awareness and accountability among all stakeholders regarding information security.

In the long run, conducting periodic cloud security assessments enables the organization to stay proactive in identifying and addressing potential vulnerabilities, thus ensuring the continuous security of assets configured in the cloud. This results in a significant reduction in the risk of security incidents, as well as strengthening the trust of customers, partners, and stakeholders in the organization's ability to effectively protect their data in the digital age.

Security Overview

Oracle's mission is to build cloud infrastructure and platform services for your business to have effective and manageable security to run your mission-critical workloads and store your data with confidence. Oracle Cloud Infrastructure's security approach is based on seven core pillars. Each pillar has multiple solutions designed to maximize the security and compliance of the platform.

Customer Isolation

Allow customers to deploy their application and data assets in an environment that commits full isolation from other tenants and Oracle's staff.

Data Encryption

Protect customer data at-rest and in-transit in a way that allows customers to meet their security and compliance requirements for cryptographic algorithms and key management.

Security Controls

Offer customers effective and easy-to-use security management solutions that allow them to constrain access to their services and segregate operational responsibilities to reduce risk associated with malicious and accidental user actions.

Visibility

Offer customers comprehensive log data and security analytics that they can use to audit and monitor actions on their resources, allowing them to meet their audit requirements and reduce security and operational risk.

Secure Hybrid Cloud

Enable customers to use their existing security assets, such as user accounts and policies, as well as third-party security solutions when accessing their cloud resources and securing their data and application assets in the cloud.

High Availability

Offer fault-independent data centers that enable high availability scale-out architectures and are resilient against network attacks, ensuring constant uptime in the face of disaster and security attack.

Verifiably Secure Infrastructure

Follow rigorous processes and use effective security controls in all phases of cloud service development and operation. Demonstrate adherence to Oracle's strict security standards through third-party audits, certifications, and attestations. Help customers demonstrate compliance readiness to internal security and compliance teams, their customers, auditors, and regulators.

Also, Oracle employs some of the world's foremost security experts in information, database, application, infrastructure, and network security. By using Oracle Cloud Infrastructure, our customers directly benefit from Oracle's deep expertise and continuous investments in security.

Basic Security Considerations

The following principles are fundamental to using any application securely:

- Keep software up to date. Use the latest product release and any patches that apply to it.
- Limit privileges as much as possible. Give users only the access necessary to perform their work. Review user privileges periodically to determine relevance to current work requirements.



ORACLE

- Monitor system activity. Establish who is expected to access which system components, and how often, and monitor those components.
- Learn about and use the Oracle Cloud Infrastructure security features. For more information, see [Security Services](#).
- Use secure best practices. For more information, see [Security Best Practices](#).
- Keep up to date on security information. Oracle regularly issues security-related patch updates and security alerts. Install all security patches as soon as possible. See the [Critical Patch Updates and Security Alerts website](#).

Compliance

Oracle Cloud Infrastructure is built for enterprises. We operate under practices aligned with the ISO/IEC 27002 Code of Practice for information security controls, from which we have identified a comprehensive set of security controls that apply to our business. Oracle Cloud Infrastructure is still a new product line, and we must operate for a period for these security controls and our operations to undergo external audit. As an enterprise cloud, we plan to pursue a broad suite of industry and government certifications, audits, and regulatory programs. For more information, see [Oracle Cloud Compliance](#).

CIS Security Assessment

CIS Oracle Cloud Infrastructure Foundations Benchmark v3.0.0 provides prescriptive guidance for establishing a secure baseline configuration for the Oracle Cloud Infrastructure environment. The scope of this benchmark is to establish a base level of security for Oracle Cloud Infrastructure services.

The benchmark is, however, not an exhaustive list of all possible security configurations and architecture. You should take the benchmark as a starting point and do the required site-specific tailoring wherever needed and when it is prudent to do so.

Download the complete CIS Oracle Cloud Infrastructure Foundations Benchmark v3.0.0 with audit instructions to help remediate the issues. <https://downloads.cisecurity.org/#/>

Summary

Number of controls per domain and non-compliant controls on tenancy **MutantRogue**:

DOMAINS	TOTAL CONTROLS	FAILED	PASSED
Identity and Access Management	17	9	8
Networking	8	6	2
Compute	3	3	0
Logging and Monitoring	18	14	4
Object Storage	6	6	0
Asset Management	2	1	1

CIS Benchmark Recommendation Summary table

Item	Description	Status
1 - Identity and Access Management		
1.1	Ensure service level admins are created to manage resources of particular service	FAILED
1.2	Ensure permissions on all resources are given only to the tenancy administrator group	FAILED
1.3	Ensure IAM administrators cannot update tenancy Administrators group	FAILED
1.4	Ensure IAM password policy requires minimum length of 14 or greater	OK
1.5	Ensure IAM password policy expires passwords within 365 days	FAILED
1.6	Ensure IAM password policy prevents password reuse	FAILED
1.7	Ensure MFA is enabled for all users with a console password	FAILED
1.8	Ensure user API keys rotate within 90 days or less	OK
1.9	Ensure user customer secret keys rotate within 90 days or less	OK
1.10	Ensure user auth tokens rotate within 90 days or less	OK
1.11	Ensure API keys are not created for tenancy administrator users	OK
1.12	Ensure all OCI IAM user accounts have a valid and current email address	OK
1.13	Ensure Dynamic Groups are used for OCI instances, OCI Cloud Databases and OCI Function to access OCI resources	FAILED
1.14	Ensure storage service-level admins cannot delete resources they manage	OK
1.15	Ensure storage service-level admins cannot delete resources they manage	FAILED

Item	Description	Status
1.16	Ensure OCI IAM credentials unused for 45 days or more are disabled	FAILED
1.17	Ensure there is only one active API Key for any single OCI IAM user	OK
2 - Networking		
2.1	Ensure no security lists allow ingress from 0.0.0.0/0 to port 22	FAILED
2.2	Ensure no security lists allow ingress from 0.0.0.0/0 to port 3389	FAILED
2.3	Ensure no network security groups allow ingress from 0.0.0.0/0 to port 22	FAILED
2.4	Ensure no network security groups allow ingress from 0.0.0.0/0 to port 3389	FAILED
2.5	Ensure the default security list of every VCN restricts all traffic except ICMP	FAILED
2.6	Ensure Oracle Integration Cloud (OIC) access is restricted to allowed sources	OK
2.7	Ensure Oracle Analytics Cloud (OAC) access is restricted to allowed sources or deployed within a Virtual Cloud Network	OK
2.8	Ensure Oracle Autonomous Shared Databases (ADB) access is restricted to allowed sources or deployed within a Virtual Cloud Network	FAILED
3 - Compute		
3.1	Ensure Compute Instance Legacy Metadata service endpoint is disabled	FAILED
3.2	Ensure Secure Boot is enabled on Compute Instance	FAILED
3.3	Ensure In-transit Encryption is enabled on Compute Instance	FAILED
4 - Logging and Monitoring		
4.1	Ensure default tags are used on resources	OK
4.2	Create at least one notification topic and subscription to receive monitoring alerts	OK
4.3	Ensure a notification is configured for Identity Provider changes	FAILED
4.4	Ensure a notification is configured for IdP group mapping changes	FAILED
4.5	Ensure a notification is configured for IAM group changes	FAILED
4.6	Ensure a notification is configured for IAM policy changes	FAILED
4.7	Ensure a notification is configured for user changes	FAILED
4.8	Ensure a notification is configured for VCN changes	FAILED
4.9	Ensure a notification is configured for changes to route tables	FAILED
4.10	Ensure a notification is configured for security list changes	FAILED
4.11	Ensure a notification is configured for network security group changes	FAILED
4.12	Ensure a notification is configured for changes to network gateways	FAILED
4.13	Ensure VCN flow logging is enabled for all subnets	FAILED
4.14	Ensure Cloud Guard is enabled in the root compartment of the tenancy	OK
4.15	Ensure a notification is configured for Oracle Cloud Guard problems detected	FAILED
4.16	Ensure customer created Customer Managed Key (CMK) is rotated at least annually	FAILED
4.17	Ensure write level Object Storage logging is enabled for all buckets	FAILED
4.18	Ensure a notification is configured for Local OCI User Authentication (Automated	OK
5 - Storage		
Object Storage		
5.1.1	Ensure no Object Storage buckets are publicly visible	FAILED
5.1.2	Ensure Object Storage Buckets are encrypted with a Customer Managed Key (CMK)	FAILED

Item	Description	Status
5.1.3	Ensure Versioning is Enabled for Object Storage Buckets	FAILED
Block Volumes		
5.2.1	Ensure Block Volumes are encrypted with Customer Managed Keys (CMK)	FAILED
5.2.2	Ensure boot volumes are encrypted with Customer Managed Key (CMK)	FAILED
File Storage Service		
5.3.1	Ensure File Storage Systems are encrypted with Customer Managed Keys (CMK)	FAILED
6 - Asset Management		
6.1	Create at least one compartment in your tenancy to store cloud resources	OK
6.2	Ensure no resources are created in the root compartment	FAILED

OCI Services Summary

STATUS	SERVICE	Description
FAILED	Vulnerability Scanning Service	Some scanning recipes were found in the Tenant, but not all instances were scanned.
FAILED	Data Safe	No target databases are registered in Datasafe
FAILED	Cloud Guard	Cloud Guard service is enabled in Tenant but only 3 compartments are checked
FAILED	Bastion	Only one Bastions instance was found in Tenant and it seems to be used for test proposal only.
FAILED	Security Zones	No Security Zones were created in Tenant
OK	Vault	No Vault instances were found in Tenant

Number of controls per domain and non-compliant controls on tenancy **NewCoCobranca**:

DOMAINS	TOTAL CONTROLS	FAILED	PASSED
Identity and Access Management	17	7	10
Networking	8	3	5
Compute	3	3	0
Logging and Monitoring	18	13	5
Object Storage	6	5	1
Asset Management	2	1	1

CIS Benchmark Recommendation Summary table

Item	Description	Status
1 - Identity and Access Management		
1.1	Ensure service level admins are created to manage resources of particular service	FAILED
1.2	Ensure permissions on all resources are given only to the tenancy administrator group	OK
1.3	Ensure IAM administrators cannot update tenancy Administrators group	FAILED
1.4	Ensure IAM password policy requires minimum length of 14 or greater	OK
1.5	Ensure IAM password policy expires passwords within 365 days	OK
1.6	Ensure IAM password policy prevents password reuse	FAILED
1.7	Ensure MFA is enabled for all users with a console password	FAILED
1.8	Ensure user API keys rotate within 90 days or less	OK
1.9	Ensure user customer secret keys rotate within 90 days or less	OK
1.10	Ensure user auth tokens rotate within 90 days or less	OK
1.11	Ensure API keys are not created for tenancy administrator users	OK
1.12	Ensure all OCI IAM user accounts have a valid and current email address	OK
1.13	Ensure Dynamic Groups are used for OCI instances, OCI Cloud Databases and OCI Function to access OCI resources	FAILED
1.14	Ensure storage service-level admins cannot delete resources they manage	OK
1.15	Ensure storage service-level admins cannot delete resources they manage	FAILED
1.16	Ensure OCI IAM credentials unused for 45 days or more are disabled	FAILED
1.17	Ensure there is only one active API Key for any single OCI IAM user	OK
2 - Networking		
2.1	Ensure no security lists allow ingress from 0.0.0.0/0 to port 22	FAILED
2.2	Ensure no security lists allow ingress from 0.0.0.0/0 to port 3389	FAILED
2.3	Ensure no network security groups allow ingress from 0.0.0.0/0 to port 22	OK
2.4	Ensure no network security groups allow ingress from 0.0.0.0/0 to port 3389	OK
2.5	Ensure the default security list of every VCN restricts all traffic except ICMP	FAILED
2.6	Ensure Oracle Integration Cloud (OIC) access is restricted to allowed sources	OK
2.7	Ensure Oracle Analytics Cloud (OAC) access is restricted to allowed sources or deployed within a Virtual Cloud Network	OK

Item	Description	Status
2.8	Ensure Oracle Autonomous Shared Databases (ADB) access is restricted to allowed sources or deployed within a Virtual Cloud Network	OK
3 – Compute		
3.1	Ensure Compute Instance Legacy Metadata service endpoint is disabled	FAILED
3.2	Ensure Secure Boot is enabled on Compute Instance	FAILED
3.3	Ensure In-transit Encryption is enabled on Compute Instance	FAILED
4 - Logging and Monitoring		
4.1	Ensure default tags are used on resources	OK
4.2	Create at least one notification topic and subscription to receive monitoring alerts	OK
4.3	Ensure a notification is configured for Identity Provider changes	FAILED
4.4	Ensure a notification is configured for IdP group mapping changes	FAILED
4.5	Ensure a notification is configured for IAM group changes	FAILED
4.6	Ensure a notification is configured for IAM policy changes	FAILED
4.7	Ensure a notification is configured for user changes	FAILED
4.8	Ensure a notification is configured for VCN changes	FAILED
4.9	Ensure a notification is configured for changes to route tables	FAILED
4.10	Ensure a notification is configured for security list changes	FAILED
4.11	Ensure a notification is configured for network security group changes	FAILED
4.12	Ensure a notification is configured for changes to network gateways	FAILED
4.13	Ensure VCN flow logging is enabled for all subnets	FAILED
4.14	Ensure Cloud Guard is enabled in the root compartment of the tenancy	OK
4.15	Ensure a notification is configured for Oracle Cloud Guard problems detected	FAILED
4.16	Ensure customer created Customer Managed Key (CMK) is rotated at least annually	OK
4.17	Ensure write level Object Storage logging is enabled for all buckets	FAILED
4.18	Ensure a notification is configured for Local OCI User Authentication (Automated	OK
5 - Storage		
Object Storage		
5.1.1	Ensure no Object Storage buckets are publicly visible	OK
5.1.2	Ensure Object Storage Buckets are encrypted with a Customer Managed Key (CMK)	FAILED
5.1.3	Ensure Versioning is Enabled for Object Storage Buckets	FAILED
Block Volumes		
5.2.1	Ensure Block Volumes are encrypted with Customer Managed Keys (CMK)	FAILED
5.2.2	Ensure boot volumes are encrypted with Customer Managed Key (CMK)	FAILED
File Storage Service		
5.3.1	Ensure File Storage Systems are encrypted with Customer Managed Keys (CMK)	FAILED
6 - Asset Management		
6.1	Create at least one compartment in your tenancy to store cloud resources	OK
6.2	Ensure no resources are created in the root compartment	FAILED

OCI Services Summary

STATUS	SERVICE	Description
FAILED	Vulnerability Scanning Service	No scanning recipes were found in the Tenant
FAILED	Data Safe	No target databases are registered in Datasafe
OK	Cloud Guard	Cloud Guard service is enabled in Tenant
OK	Bastion	Bastions instance was found in Tenant
FAILED	Security Zones	No Security Zones were created in Tenant
FAILED	Vault	No Vault instances were found in Tenant

1 - Identity and Access Management

1.1 Ensure service level admins are created to manage resources of particular service

To apply least-privilege security principle, one can create service-level administrators in corresponding groups and assigning specific users to each service-level administrative group in a tenancy. This limits administrative access in a tenancy.

It means service-level administrators can only manage resources of a specific service.

Example policies for global/tenant level service-administrators

```
Allow group VolumeAdmins to manage volume-family in tenancy
Allow group ComputeAdmins to manage instance-family in tenancy
Allow group NetworkAdmins to manage virtual-network-family in tenancy
```

Organizations have various ways of defining service-administrators. Some may prefer creating service administrators at a tenant level and some per department or per project or even per application environment (dev/test/production etc.). Either approach works so long as the policies are written to limit access given to the service-administrators.

Example policies for compartment level service-administrators

```
Allow group NonProdComputeAdmins to manage instance-family in compartment dev
Allow group ProdComputeAdmins to manage instance-family in compartment production
Allow group A-Admins to manage instance-family in compartment Project-A
Allow group A-Admins to manage volume-family in compartment Project-A
```

Rationale:

Creating service-level administrators helps in tightly controlling access to Oracle Cloud Infrastructure (OCI) services to implement the least-privileged security principle.

Result:

Tenancy MutantRogue: **Non-Compliant (19 of 124 items)**

Tenancy MutantRogue: **Non-Compliant (3 of 54 items)**

We found policies granting "manage all-resources" in tenancy. Substitute these policies, creating service level administrators to implement the least privileged security principle.

Remediation

Refer to the policy syntax document and create new policies if the audit results indicate that the required policies are missing. This can be done via OCI console or OCI CLI/SDK or API.

From Command Line:

```
oci iam policy create [OPTIONS]
```

Creates a new policy in the specified compartment (either the tenancy or another of your compartments). If you're new to policies, see [Getting Started with Policies](#). You must specify a name for the policy, which must be unique across all policies in your tenancy and cannot be changed. You must also specify a description for the policy (although it can be an empty string). It does not have to be unique, and you can change it anytime with `UpdatePolicy`. You must specify one or more policy statements in the `statements` array. For information about writing policies, see [How Policies Work](#) and [Common Policies](#).

1.2 Ensure permissions on all resources are given only to the tenancy administrator group

There is a built-in OCI IAM policy enabling the Administrators group to perform any action within a tenancy. In the OCI IAM console, this policy reads:

```
Allow group Administrators to manage all-resources in tenancy
```

Administrators create more users, groups, and policies to provide appropriate access to other groups. Administrators should not allow any-other-group full access to the tenancy by writing a policy like this:

```
Allow group any-other-group to manage all-resources in tenancy
```

The access should be narrowed down to ensure the least-privileged principle is applied.

Rationale:

Permission to manage all resources in a tenancy should be limited to a small number of users in the 'Administrators' group for break-glass situations and to set up users/groups/policies when a tenancy is created.

No group other than 'Administrators' in a tenancy should need access to all resources in a tenancy, as this violates the enforcement of the least privilege principle.

Result:

Tenancy MutantRogue: **Non-Compliant (1 of 124 items)**

Tenancy NewCoCobranca: **Compliant**

Remediation:

Refer to the policy syntax document and create new policies if the audit results indicate that the required policies are missing. This can be done via OCI console or OCI CLI/SDK or API.

From Command Line

```
oci iam policy create [OPTIONS]
```

Creates a new policy in the specified compartment (either the tenancy or another of your compartments). If you're new to policies, see [Getting Started with Policies](#). You must specify a name for the policy, which must be unique across all policies in your tenancy and cannot be changed. You must also specify a description for the policy (although it can be an empty string). It does not have to be unique, and you can change it anytime with `UpdatePolicy`. You must specify one or more policy statements in the `statements` array. For information about writing policies, see [How Policies Work](#) and [Common Policies](#).

Recommendation:

Evaluate if tenancy-wide administrative access is needed for the identified policy and update it to be more restrictive.

1.3 Ensure IAM administrators cannot update tenancy Administrators group

Tenancy administrators can create more users, groups, and policies to provide other service administrators access to OCI resources.

For example, an IAM administrator will need to have access to manage resources like compartments, users, groups, dynamic-groups, policies, identity-providers, tenancy tagnamespaces, tag-definitions in the tenancy.

The policy that gives IAM-Administrators or any other group full access to 'groups' resources should not allow access to the tenancy 'Administrators' group. The policy statements would look like:

```
Allow group IAMAdmins to inspect users in tenancy
Allow group IAMAdmins to use users in tenancy where target.group.name != 'Administrators'
Allow group IAMAdmins to inspect groups in tenancy
Allow group IAMAdmins to use groups in tenancy where target.group.name != 'Administrators'
```

Note: You must include separate statements for 'inspect' access, because the `target.group.name` variable is not used by the `ListUsers` and `ListGroups` operations

Rationale:

These policy statements ensure that no other group can manage tenancy administrator users or the membership to the 'Administrators' group thereby gain or remove tenancy administrator access.

Result:

Tenancy MutantRogue: **Non-Compliant (4 of 124 items)**

Tenancy NewCoCobranca: **Non-Compliant (2 of 54 items)**

Remediation:

From Console:

1. Login to [OCI Console](#).
2. Select **Identity & Security** from the Services menu.
3. Select **Policies** from Identity & Security menu.

4. Click on an individual policy under the Name heading.
5. Ensure Policy statements look like this:

```
Allow group IAMAdmins to use users in tenancy where target.group.name != 'Administrators'  
Allow group IAMAdmins to use groups in tenancy where target.group.name != 'Administrators'
```

Recommendation:

Evaluate if tenancy-wide administrative access is needed for the identified policy and update it to be more restrictive.

1.4 Ensure IAM password policy requires minimum length of 14 or greater

Password policies are used to enforce password complexity requirements. IAM password policies can be used to ensure passwords are at least a certain length and are composed of certain characters. It is recommended the password policy require a minimum of:

- password length 14 characters
- contain at least 1 non-alphabetic character (Number or “Special Character”).

Rationale:

In keeping with the overall goal of having users create a password that is not overly weak, an eight-character minimum password length is recommended for an MFA account, and 14 characters for a password only account. In addition, maximum password length should be made as long as possible based on system/software capabilities and not restricted by policy.

In general, it is true that longer passwords are better (harder to crack), but it is also true that forced password length requirements can cause user behavior that is predictable and undesirable. For example, requiring users to have a minimum 16-character password may cause them to choose repeating patterns like fourfourfourfour or passwordpassword that meet the requirement but aren’t hard to guess. Additionally, length requirements increase the chances that users will adopt other insecure practices, like writing them down, re-using them or storing them unencrypted in their documents.

Password composition requirements are a poor defense against guessing attacks. Forcing users to choose some combination of upper-case, lower-case, numbers, and special characters has a negative impact. It places an extra burden on users and many will use predictable patterns (for example, a capital letter in the first position, followed by lowercase letters, then one or two numbers, and a “special character” at the end). Attackers know this, so dictionary attacks will often contain these common patterns and use the most common substitutions like, \$ for s, @ for a, 1 for l, 0 for o.

Passwords that are too complex in nature make it harder for users to remember, leading to bad practices. In addition, composition requirements provide no defense against common attack types such as social engineering or insecure storage of passwords.

Result:

Tenancy MutantRogue: **Compliant**

Tenancy NewCoCobranca: **Compliant**

1.5 Ensure IAM password policy expires passwords within 365 days

IAM password policies can require passwords to be rotated or expired after a given number of days. It is recommended that the password policy expire passwords after 365 and are changed immediately based on events.

Rationale:

Excessive password expiration requirements do more harm than good, because these requirements make users select predictable passwords, composed of sequential words and numbers that are closely related to each other.

In these cases, the next password can be predicted based on the previous one (incrementing a number used in the password for example). Also, password expiration requirements offer no containment benefits because attackers will often use credentials as soon as they compromise them. Instead, immediate password changes should be based on key events including, but not limited to:

1. Indication of compromise
2. Change of user roles
3. When a user leaves the organization.

Not only does changing passwords every few weeks or months frustrate the user, it's been suggested that it does more harm than good, because it could lead to bad practices by the user such as adding a character to the end of their existing password.

In addition, we also recommend a yearly password change. This is primarily because for all their good intentions users will share credentials across accounts. Therefore, even if a breach is publicly identified, the user may not see this notification, or forget they have an account on that site. This could leave a shared credential vulnerable indefinitely. Having an organizational policy of a 1-year (annual) password expiration is a reasonable compromise to mitigate this with minimal user burden.

Result:

Tenancy MutantRogue: **Non-Compliant (1 of 2 items)**

Tenancy NewCoCobranca: **Compliant**

Remediation:**From Console:**

1. Go to Identity Domains: <https://cloud.oracle.com/identity/domains/>
2. Select the **Compartment** the Domain to remediate is in
3. Click on the **Domain** to remediate
4. Click on **Settings**
5. Click on **Password policy**

6. Click each Password policy in the domain
7. Click **Edit password rules**
8. Change **Expires after (days)** to 365

Recommendation:

Evaluate password reuse policies are inline with your organizational standard.

1.6 Ensure IAM password policy prevents password reuse

IAM password policies can prevent the reuse of a given password by the same user. It is recommended the password policy prevent the reuse of passwords.

Rationale:

Enforcing password history ensures that passwords are not reused in for a certain period of time by the same user. If a user is not allowed to use last 24 passwords, that window of time is greater. This helps maintain the effectiveness of password security.

Result:

Tenancy MutantRogue: **Non-Compliant (2 of 2 items)**

Tenancy NewCoCobranca: **Non-Compliant (1 of 1 item)**

Remediation:

1. Login to [OCI Console](#)
2. Select **Identity & Security** from the Services menu
3. Select **Domains** from the Identity & Security menu.
4. Select the **Compartment** which the Domain to remediate is in.
5. Click on the **Domain**.
6. Click on **Settings**.
7. Click on **Password policy**.
8. Click **Edit password rules**
9. Update the Password length (minimum) setting to 14 or greater

Additionally

10. Under The Passwords must meet the following character requirements section, update the number given in Special (minimum) setting to 1 or greater or Under The Passwords must meet the following character requirements section, update the number given in Numeric (minimum) setting to 1 or greater
11. Click Save changes

Recommendation:

Evaluate if password reuse policies are in line with your organizational standard.

1.7 MFA is enabled for all users with a console password

Multi-factor authentication is a method of authentication that requires the use of more than one factor to verify a user's identity. With MFA enabled in the IAM service, when a user signs into Oracle Cloud Infrastructure, they are prompted for their username and password, which is the first factor (something that they know). The user is then prompted to provide a second verification code from a registered MFA device, which is the second factor (something that they have). The two factors work together, requiring an extra layer of security to verify the user's identity and complete the sign-in process. OCI IAM supports two-factor authentication using a password (first factor) and a device that can generate a time-based one-time password (TOTP) (second factor).

Rationale:

Multi factor authentication adds an extra layer of security during the login process and makes it harder for unauthorized users to gain access to OCI resources.

Result:

Tenancy MutantRogue: **Non-Compliant (74 of 104 items)**

Tenancy NewCoCobranca: **Non-Compliant (27 of 48 items)**

Remediation:

Each user must enable MFA for themselves using a device they will have access to every time they sign in. The configuration is available on the user's profile, "Enable Multi-Factor Authentication". Users may download and use OAM (Oracle Mobile Authenticator) or a TOTP capable APP.

An administrator cannot enable MFA for another user but can enforce MFA by identifying the list of non-complaint users, notifying them, or disabling access by resetting password for non-complaint accounts.

From Console:

1. Go to <https://cloud.oracle.com/identity/domains/>
2. Select **Domains** from Identity menu.
3. Select the domain
4. Click **Security**
5. Click Sign-on polices then the "**Default Sign-on Policy**"
6. Under the sign-on rules header, click the **three dots** on the rule with the highest priority.
7. Select **Edit sign-on rule**
8. Make a change to ensure that allow access is selected and prompt for an additional factor is enabled.

From Command Line:

Execute the following:

```
oci iam user ui-password create-or-reset --user-id <OCID of the non-compliant user>
```

Note: Check the “CIS Oracle Cloud Infrastructure Foundations Benchmark” for Cloud guard instructions to audit this setting.

Recommendation:

Evaluate if local users are required. For Break Glass accounts ensure MFA is in place.

1.8 Ensure user API keys rotate within 90 days

API keys are used by administrators, developers, services and scripts for accessing OCI APIs directly or via SDKs/OCI CLI to search, create update or delete OCI resources. The API key is an RSA key pair. The private key is used for signing the API requests and the public key is associated with a local or synchronized user's profile.

Rationale:

It is important to secure and rotate an API key every 90 days or less as it provides the same level of access that a user it is associated with has.

In addition to a security engineering best practice, this is also a compliance requirement. For example, PCI-DSS Section 3.6.4 states, "Verify that key-management procedures include a defined cryptoperiod for each key type in use and define a process for key changes at the end of the defined crypto period(s)."

Result:

Tenancy MutantRogue: **Compliant**

Tenancy NewCoCobranca: **Compliant**

1.9 Ensure user customer secret keys rotate within 90 days

Object Storage provides an API to enable interoperability with Amazon S3. To use this Amazon S3 Compatibility API, you need to generate the signing key required to authenticate with Amazon S3. This special signing key is an Access Key/Secret Key pair. Oracle generates the Customer Secret key to pair with the Access Key.

Rationale: It is important to secure and rotate a customer secret key every 90 days or less as it provides the same level of object storage access that a user is associated with has.

Result:

Tenancy MutantRogue: **Compliant**

Tenancy NewCoCobranca: **Compliant**

1.10 Ensure user auth tokens rotate within 90 days or less

Auth tokens are authentication tokens generated by Oracle. You use auth tokens to authenticate with APIs that do not support the Oracle Cloud Infrastructure signature-based authentication. If the service requires an auth token, the service-specific documentation instructs you to generate one and how to use it.

Rationale:

It is important to secure and rotate an auth token every 90 days or less as it provides the same level of access to APIs that do not support the OCI signature-based authentication as the user associated to it.

Result:

Tenancy MutantRogue: **Compliant**

Tenancy NewCoCobranca: **Compliant**

1.11 Ensure user IAM Database Passwords rotate within 90 days

Users can create and manage their database password in their IAM user profile and use that password to authenticate to databases in their tenancy. An IAM database password is a different password than an OCI Console password. Setting an IAM database password allows an authorized IAM user to sign in to one or more Autonomous Databases in their tenancy.

An IAM database password is a different password than an OCI Console password. Setting an IAM database password allows an authorized IAM user to sign in to one or more Autonomous Databases in their tenancy.

Result:

Tenancy MutantRogue: **Compliant**

Tenancy NewCoCobranca: **Compliant**

1.12 Ensure API keys are not created for tenancy administrator users

Tenancy administrator users have full access to the organization's OCI tenancy. API keys associated with user accounts are used for invoking the OCI APIs via custom programs or clients like CLI/SDKs. The clients are typically used for performing day-to-day operations and should never require full tenancy access. Service-level administrative users with API keys should be used instead.

Rationale:

For performing day-to-day operations tenancy administrator access is not needed. Service-level administrative users with API keys should be used to apply privileged security principles.

Result:

Tenancy MutantRogue: **Compliant**

Tenancy NewCoCobranca: **Compliant**

1.13 Ensure all OCI IAM user accounts have a valid and current email address

All OCI IAM local user accounts have an email address field associated with the account. It is recommended to specify an email address that is valid and current. If you have an email address in your user profile, you can use the Forgot Password link on the sign on page to have a temporary password sent to you.

Rationale:

Having a valid and current email address associated with an OCI IAM local user account allows you to tie the account to identity in your organization. It also allows that user to reset their password if it is forgotten or lost.

Result:

Tenancy MutantRogue: **Non-Compliant (75 of 104 items)**

Tenancy NewCoCobranca: **Non-Compliant (24 of 48 items)**

Remediation:

From OCI Console:

1. Login to [OCI Console](#).
2. Select **Identity & Security** from the Services menu.
3. Select **Domains** from the Identity menu.
4. For each domain listed, click on the name and select **Users**.
5. Click on an individual user under the Username heading.
6. Ensure a valid and current email address is next to Email and Recovery email.

From Command Line

Execute the following for each non-compliant user:

```
oci iam user update --user-id <user-ocid> --email '<email address>'
```

Recommendation:

Add emails to users to allow them to use the 'Forgot Password' feature and uniquely identify the user. For service accounts it could be a mail alias.

1.14 Ensure Instance Principal authentication is used for OCI instances, OCI Cloud Databases and OCI Functions to access OCI resources

OCI instances, OCI database and OCI functions can access other OCI resources either via an OCI API key associated to a user or via Instance Principal. Instance Principal authentication can be achieved by inclusion in a Dynamic Group that has an IAM policy granting it the required access or using an OCI IAM policy that has request.principal added

to the where clause. Access to OCI Resources refers to making API calls to another OCI resource like Object Storage, OCI Vaults, etc.

Instance Principal reduces the risks related to hard-coded credentials. Hard-coded API keys can be shared and require rotation, which can open them up to being compromised. Compromised credentials could allow access to OCI services outside of the expected radius.

Result:

Tenancy MutantRogue: **Compliant**

Tenancy NewCoCobranca: **Compliant**

1.15 Ensure storage service-level admins cannot delete resources they manage

To apply the separation of duties security principle, one can restrict service-level administrators from being able to delete resources they are managing. It means service-level administrators can only manage resources of a specific service but not delete resources for that specific service.

Example policies for global/tenant level for block volume service-administrators:

```
Allow group VolumeUsers to manage volumes in tenancy where request.permission!='VOLUME_DELETE'  
Allow group VolumeUsers to manage volume-backups in tenancy where request.permission!='VOLUME_BACKUP_DELETE'
```

Example policies for global/tenant level for file storage system service-administrators:

```
Allow group FileUsers to manage file-systems in tenancy where request.permission!='FILE_SYSTEM_DELETE'  
Allow group FileUsers to manage mount-targets in tenancy where request.permission!='MOUNT_TARGET_DELETE'  
Allow group FileUsers to manage export-sets in tenancy where request.permission!='EXPORT_SET_DELETE'
```

Example policies for global/tenant level for object storage system service-administrators:

```
Allow group BucketUsers to manage objects in tenancy where request.permission!='OBJECT_DELETE'  
Allow group BucketUsers to manage buckets in tenancy where request.permission!='BUCKET_DELETE'
```

Rationale:

Creating service-level administrators without the ability to delete the resource they are managing helps in tightly controlling access to Oracle Cloud Infrastructure (OCI) services by implementing the separation of duties security principle.

Result:

Tenancy MutantRogue: **Non-Compliant (136 of 124 items)**

Tenancy NewCoCobranca: **Non-Compliant (72 of 54 items)**

Remediation:

Add the appropriate where condition to any policy statement that allows the storage service-level to manage the storage service.

From Console:

1. Login to [OCI Console](#).
2. Select **Identity & Security** from the Services menu.
3. Select **Domains** from the Identity menu.
4. In the compartment dropdown, choose the compartment.
5. Go to **Policies**.
6. Open each policy to view the policy statements.
7. Add the appropriate where condition to any policy statement that allows the storage service-level to manage the storage service.

From Command Line:

1. Execute the following command:

```
for compid in `oci iam compartment list --compartment-id-in-subtree TRUE 2>/dev/null | jq -r '.data[] | .id`
do
  for policy in `oci iam policy list --compartment-id $compid 2>/dev/null | jq -r '.data[] | .id`
  do
    output=`oci iam policy list --compartment-id $compid 2>/dev/null | jq -r '.data[] | .id, .name,
.statements`
    if [ ! -z "$output" ]; then echo $output; fi
  done
done
```

2. Verify the policies to ensure that the policy statements that grant access to storage service-level administrators have a condition that excludes access to delete the service they are the administrator for.

Recommendation:

To apply a separation of duties security principle, it is recommended to restrict service-level administrators from being able to delete resources they are managing.

1.16 Ensure OCI IAM credentials unused for 45 days or more are disabled

OCI IAM Local users can access OCI resources using different credentials, such as passwords or API keys. It is recommended that credentials that have been unused for 45 days or more be deactivated or removed.

Rationale:

Disabling or removing unnecessary OCI IAM local users will reduce the window of opportunity for credentials associated with a compromised or abandoned account to be used.

Result:

Tenancy MutantRogue: **Non-Compliant (102 of 104 items)**

Tenancy NewCoCobranca: **Non-Compliant (48 of 48 items)**

Remediation:

From OCI Console:

1. Login to [OCI Console](#).
2. Select **Identity & Security** from the Services menu.
3. Select **Domains** from the Identity menu.
4. For each domain listed, click on the name and select **Users**.
5. Click on an individual user under the **Username** heading.
6. Click **More action**
7. Select **Deactivate**

From Command Line:

1. Create a input.json:

```
{
  "operations": [
    { "op": "replace", "path": "active", "value": false}
  ],
  "schemas": ["urn:ietf:params:scim:api:messages:2.0:PatchOp"],
  "userId": "<user-ocid>"
}
```

2. Execute the below:

```
oci identity-domains user patch --from-json file:///file.json --endpoint <identity-domain-endpoint>
```

1.17 Ensure there is only one active API Key for any single OCI IAM user

API Keys are long-term credentials for an OCI IAM user. They can be used to make programmatic requests to the OCI APIs directly or via, OCI SDKs or the OCI CLI

Rationale:

Having a single API Key for an OCI IAM reduces attack surface area and makes it easier to manage.

Impact:

Deletion of an OCI API Key will remove programmatic access to OCI APIs

Result:

Tenancy MutantRogue: **Compliant**

Tenancy NewCoCobranca: **Compliant**

2 - Networking

2.1 Ensure no security lists allow ingress from 0.0.0.0/0 to port 22

Security lists provide stateful or stateless filtering of ingress/egress network traffic to OCI resources on a subnet level. It is recommended that no security group allows unrestricted ingress access to port 22.

Rationale:

Removing unfettered connectivity to remote console services, such as Secure Shell (SSH), reduces a server's exposure to risk.

Impact:

For updating an existing environment, care should be taken to ensure that administrators currently relying on an existing ingress from 0.0.0.0/0 have access to ports 22 and/or 3389 through another network security group or security list.

Result:

Tenancy MutantRogue: **Non-Compliant (59 of 199 items)**

Tenancy NewCoCobranca: **Non-Compliant (56 of 139 items)**

Remediation:

From OCI Console:

1. Login to [OCI Console](#).
2. Click in the search bar, top of the screen.
3. Type Advanced Resource Query and hit enter.
4. Click the Advanced Resource Query button in the upper right of the screen.
5. Enter the following query:

```
query SecurityList resources where (IngressSecurityRules.source = '0.0.0.0/0' && IngressSecurityRules.protocol = 6 && IngressSecurityRules.tcpOptions.destinationPortRange.max = 22 && IngressSecurityRules.tcpOptions.destinationPortRange.min = 22)
```

6. For each security list in the returned results, click the security list name.
7. Either edit the ingress rule to be more restrictive, delete the ingress rule or click on the VCN and terminate the security list as appropriate.

From Command Line

1. Execute the following command:

```
oci search resource structured-search --query-text "query SecurityList resources where (IngressSecurityRules.source = '0.0.0.0/0' && IngressSecurityRules.protocol = 6 && IngressSecurityRules.tcpOptions.destinationPortRange.max = 22 && IngressSecurityRules.tcpOptions.destinationPortRange.min = 22)"
```

2. For each of the security lists identified get its details.

```
oci network security-list get --security-list-id <security_list_id>
```

3. Then you can either update the Security List or delete it:

Update the Security List: To update the security list, copy the ingress-security-rules element from the JSON returned by the above get call, edit it appropriately and use it in the following command

```
oci network security-list update --security-list-id <security-list-id> -- ingress-security-rules '<ingress security rules JSON>'
```

Delete the Security List: To delete the security list, copy security list ID and run the following command

```
oci network security-list delete --security-list-id <security_list_id>
```

Note: Check the “CIS Oracle Cloud Infrastructure Foundations Benchmark” for Cloud guard instructions to audit this setting.

Recommendation:

Review the security lists. If they are not used (attached to a subnet) they should be deleted if possible or empty. For attached security lists it is recommended to restrict the CIDR block to only allow access to Port 22 from known networks.

2.2 No security lists allow ingress from 0.0.0.0/0 to port 3389

Security lists provide stateful or stateless filtering of ingress/egress network traffic to OCI resources on a subnet level. It is recommended that no security group allows unrestricted ingress access to port 3389.

Rationale:

Removing unfettered connectivity to remote console services, such as Remote Desktop Protocol (RDP), reduces a server's exposure to risk.

Impact:

For updating an existing environment, care should be taken to ensure that administrators currently relying on an existing ingress from 0.0.0.0/0 have access to ports 22 and/or 3389 through another network security group or security list.

Result:

Tenancy MutantRogue: **Non-Compliant (20 of 199 items)**

Tenancy NewCoCobranca: **Non-Compliant (39 of 139 items)**

Remediation:

For each security list in the returned results, click the security list name. Either edit the ingress rule to be more restrictive, delete the ingress rule or click on the VCN and terminate the security list as appropriate.

From Console:

1. Login to [OCI Console](#).
2. Click in the search bar, top of the screen.
3. Type **Advanced Resource Query** and hit enter.
4. Click the **Advanced Resource Query** button in the upper right of the screen.
5. Enter the following query:

```
query SecurityList resources where (IngressSecurityRules.source = '0.0.0.0/0' && IngressSecurityRules.protocol = 6 && IngressSecurityRules.tcpOptions.destinationPortRange.max = 3389 && IngressSecurityRules.tcpOptions.destinationPortRange.min = 3389)
```

6. For each **security list** in the returned results, click the security list name.
7. Either edit the **ingress rule** to be more restrictive, **delete** the **ingress rule** or click on the **VCN** and terminate the security list as appropriate.

From Command Line:

1. Execute the following command:

```
oci search resource structured-search --query-text "query SecurityList resources where (IngressSecurityRules.source = '0.0.0.0/0' && IngressSecurityRules.protocol = 6 && IngressSecurityRules.tcpOptions.destinationPortRange.max = 3389 && IngressSecurityRules.tcpOptions.destinationPortRange.min = 3389)"
```

2. For each of the security lists identified get the its details.

```
oci network security-list get --security-list-id <security_list_id>
```

3. Then you can either update the Security List or delete it.

Update the Security List - To update the security list, copy the ingress-security-rules element from the JSON returned by the above get call, edit it appropriately and use it in the following command:

```
oci network security-list update --security-list-id <security-list-id> -- ingress-security-rules '<ingress security rules JSON>'
```

4. **Delete the Security List** - To delete the security list, copy security list ID and run the following command

```
oci network security-list delete --security-list-id <security_list_id>
```

Recommendation:

Review the security lists. If they are not used (attached to a subnet) they should be deleted if possible or empty. For attached security lists it is recommended to restrict the CIDR block to only allow access to Port 3389 from known networks.

2.3 Ensure no network security groups allow ingress from 0.0.0.0/0 to port 22

Network security groups provide stateful filtering of ingress/egress network traffic to OCI resources. It is recommended that no security group allows unrestricted ingress access to port 22.

Rationale:

Removing unfettered connectivity to remote console services, such as Secure Shell (SSH), reduces a server's exposure to risk.

Impact:

For updating an existing environment, care should be taken to ensure that administrators currently relying on an existing ingress from 0.0.0.0/0 have access to ports 22 and/or 3389 through another network security group or security list.

Result:

Tenancy MutantRogue: **Non-Compliant (1 of 13 items)**

Tenancy NewCoCobranca: **Compliant**

Remediation:

From Command Line

For each NSG listed above execute the following command:

```
oci network nsg rules remove -nsg-id <NSG_ID>
```

or update the security rules running the following command:

```
oci network nsg rules update -nsg-id <NSGID from audit output> --security-rules='[<updated security-rules JSON>]'
```

eg:

```
oci network nsg rules update -nsgid <NSG_ID> --securityrules='[{ "description": null, "destination": null, "destination-type": null, "direction": "INGRESS", "icmp-options": null, "id": "709001", "is-stateless": null, "protocol": "6", "source": "140.238.154.0/24", "source-type": "CIDR_BLOCK", "tcp-options": { "destination-port-range": { "max": 22, "min": 22 }, "source-port-range": null }, "udp-options": null }]'
```

Note: Check the "CIS Oracle Cloud Infrastructure Foundations Benchmark" for Cloud guard instructions to audit this setting.

Recommendation:

Review the network security groups. If they are not used(attached to a subnet) they should be deleted if possible or empty. For attached security lists it is recommended to restrict the CIDR block to only allow access to Port 3389 from known networks.

2.4 Ensure no network security groups allow ingress from 0.0.0.0/0 to port 3389

Network security groups provide stateful filtering of ingress/egress network traffic to OCI resources. It is recommended that no security group allows unrestricted ingress access to port 3389.

Rationale:

Removing unfettered connectivity to remote console services, such as Remote Desktop Protocol (RDP), reduces a server's exposure to risk.

Impact:

For updating an existing environment, care should be taken to ensure that administrators currently relying on an existing ingress from 0.0.0.0/0 have access to ports 22 and/or 3389 through another network security group or security list.

Result:

Tenancy MutantRogue: **Non-Compliant (1 of 13 items)**

Tenancy NewCoCobranca: **Compliant**

Remediation:

From Command Line

For each NSG listed above execute the following command:

```
oci network nsg rules remove -nsg-id <NSG_ID>
```

or update the security rules running the following command:

```
oci network nsg rules update -nsg-id <NSGID from audit output> --security-rules='[<updated security-rules JSON>]'
```

eg:

```
oci network nsg rules update -nsgid <NSG_ID> --securityrules='[{ "description": null, "destination": null, "destination-type": null, "direction": "INGRESS", "icmp-options": null, "id": "709001", "is-stateless": null, "protocol": "6", "source": "140.238.154.0/24", "source-type": "CIDR_BLOCK", "tcp-options": { "destination-port-range": { "max": 3389, "min": 3389 }, "source-port-range": null }, "udp-options": null }]'
```

Note: Check the "CIS Oracle Cloud Infrastructure Foundations Benchmark" for Cloud guard instructions to audit this setting.

Recommendation:

Review the network security groups. If they are not used(attached to a subnet) they should be deleted if possible or empty. For attached network security groups it is recommended to restrict the CIDR block to only allow access to Port 3389 from known networks.

2.5 Ensure the default security list of every VCN restricts all traffic except ICMP within VCN

A default security list is created when a Virtual Cloud Network (VCN) is created and attached to the public subnets in the VCN. Security lists provide stateful or stateless filtering of ingress and egress network traffic to OCI resources in the VCN. It is recommended that the default security list does not allow unrestricted ingress and egress access to resources in the VCN.

Rationale:

Removing unfettered connectivity to OCI resource, reduces a server's exposure to unauthorized access or data exfiltration.

Impact:

For updating an existing environment, care should be taken to ensure that administrators currently relying on an existing ingress from 0.0.0.0/0 have access to port 22 through another network security group and servers have egress to specified ports and protocols through another network security group.

Result:

Tenancy MutantRogue: **Non-Compliant (38 of 38 items)**

Tenancy NewCoCobranca: **Non-Compliant (17 of 17 items)**

Remediation:

From OCI Console:

1. Login to [OCI Console](#).
2. Click on **Networking -> Virtual Cloud Networks** from the services menu.
3. For each VCN listed Click on **Security Lists**.
4. Click on **Default Security List for <VCN Name>**.
5. Select the Ingress Rule with 'Source 0.0.0.0/0'
6. Either Edit the Security rule to restrict the source and/or port range or delete the rule.
7. Identify the Egress Rule with 'Destination 0.0.0.0/0, All Protocols'
8. Either Edit the Security rule to restrict the source and/or port range or delete the rule.

Note: Check the "CIS Oracle Cloud Infrastructure Foundations Benchmark" for Cloud guard instructions to audit this setting.

2.6 Ensure Oracle Integration Cloud (OIC) access is restricted to allowed sources

Oracle Integration (OIC) is a complete, secure, but lightweight integration solution that enables you to connect your applications in the cloud. It simplifies connectivity between your applications and connects both your applications that live in the cloud and your applications that still live on premises. Oracle Integration provides secure, enterprise-grade connectivity regardless of the applications you are connecting or where they reside. OIC instances are created

within an Oracle managed secure private network with each having a public endpoint. The capability to configure ingress filtering of network traffic to protect your OIC instances from unauthorized network access is included. It is recommended that network access to your OIC instances be restricted to your approved corporate IP Addresses or Virtual Cloud Networks (VCN)s.

Rationale:

Restricting connectivity to OIC Instances reduces an OIC instance’s exposure to risk.

Impact:

When updating ingress filters for an existing environment, care should be taken to ensure that IP addresses and VCNs currently used by administrators, users, and services to access your OIC instances are included in the updated filters.

Result:

Tenancy MutantRogue: **Compliant**

Tenancy NewCoCobranca: **Compliant**

2.7 Ensure Oracle Analytics Cloud (OAC) access is restricted to allowed sources or deployed within a Virtual Cloud Network

Oracle Analytics Cloud (OAC) is a scalable and secure public cloud service that provides a full set of capabilities to explore and perform collaborative analytics for you, your workgroup, and your enterprise. OAC instances provide ingress filtering of network traffic or can be deployed with in an existing Virtual Cloud Network VCN. It is recommended that all new OAC instances be deployed within a VCN and that the Access Control Rules are restricted to your corporate IP Addresses or VCNs for existing OAC instances.

Rationale:

Restricting connectivity to Oracle Analytics Cloud instances reduces an OAC instance’s exposure to risk.

Impact:

When updating ingress filters for an existing environment, care should be taken to ensure that IP addresses and VCNs currently used by administrators, users, and services to access your OAC instances are included in the updated filters. Also, these changes will temporarily bring the OAC instance offline.

Result:

Tenancy MutantRogue: **Compliant**

Tenancy NewCoCobranca: **Compliant**

2.8 Ensure Oracle Autonomous Shared Databases (ADB) access is restricted to allowed sources or deployed within a Virtual Cloud Network

Oracle Autonomous Database Shared (ADB-S) automates database tuning, security, backups, updates, and other routine management tasks traditionally performed by DBAs. ADB-S provide ingress filtering of network traffic or can be deployed within an existing Virtual Cloud Network (VCN). It is recommended that all new ADB-S databases be deployed within a VCN and that the Access Control Rules are restricted to your corporate IP Addresses or VCNs for existing ADB-S databases.

Rationale:

Restricting connectivity to ADB-S Databases reduces an ADB-S database's exposure to risk.

Impact:

When updating ingress filters for an existing environment, care should be taken to ensure that IP addresses and VCNs currently used by administrators, users, and services to access your ADB-S instances are included in the updated filters.

Result:

Tenancy MutantRogue: **Non-Compliant (1 of 1 item)**

Tenancy NewCoCobranca: **Compliant**

Remediation

From Console

1. For each ADB-S database in the returned results, click the ADB-S database name.
2. Click **Edit** next to **Access Control Rules**.
3. Click **+Another Rule** and add rules as required.
4. Click **Save Changes**.

From CLI

1. Follow the audit procedure.
2. Get the json input format by executing the following command:

```
oci db autonomous-database update --generate-full-command-json-input
```

3. For each of the ADB-S Database identified get its details.
4. Update the whitelistIps, copy the WhiteListIPs element from the JSON returned by the above get call, edit it appropriately and use it in the following command:

```
oci db autonomous-database update --autonomous-database-id <ABD-S OCID> --from-json '<network endpoints JSON>'
```

3 - Compute

3.1 Ensure Compute Instance Legacy Metadata service endpoint is disabled

Compute Instances that utilize Legacy MetaData service endpoints (IMDSv1) are susceptible to potential SSRF attacks. To bolster security measures, it is strongly advised to reconfigure Compute Instances to adopt Instance Metadata Service v2, aligning with the industry's best security practices.

Rationale:

Enabling Instance Metadata Service v2 enhances security and grants precise control over metadata access. Transitioning from IMDSv1 reduces the risk of SSRF attacks, bolstering system protection.

IMDv1 poses security risks due to its inferior security measures and limited auditing capabilities. Transitioning to IMDv2 ensures a more secure environment with robust security features and improved monitoring capabilities.

Impact:

If you disable IMDSv1 on an instance that does not support IMDSv2, you might not be able to connect to the instance when you launch it. IMDSv2 is supported on the following platform images:

- Oracle Autonomous Linux 8.x images
- Oracle Autonomous Linux 7.x images released in June 2020 or later
- Oracle Linux 8.x, Oracle Linux 7.x, and Oracle Linux 6.x images released in July 2020 or later

Other platform images, most custom images, and most Marketplace images do not support IMDSv2. Custom Linux images might support IMDSv2 if cloud-init is updated to version 20.3 or later and Oracle Cloud Agent is updated to version 0.0.19 or later. Custom Windows images might support IMDSv2 if Oracle Cloud Agent is updated to version 1.0.0.0 or later; cloudbase-init does not support IMDSv2.

Result:

Tenancy MutantRogue: **Non-Compliant (383 of 393 item)**

Tenancy NewCoCobranca: **Non-Compliant (362 of 363 items)**

Remediation:

From OCI Console:

1. Login to the [OCI Console](#)
2. Click on the search box at the top of the console and search for compute instance name.
3. Click on the instance name, In the **Instance Details** section, next to Instance Metadata Service, click **Edit**.
4. For the **Instance metadata service**, select the **Version 2 only** option.
5. Click **Save Changes**.

Note: Disabling IMDSv1 on an incompatible instance may result in connectivity issues upon launch.

To re-enable IMDSv1, follow these steps:

1. On the Instance Details page in the Console, click **Edit** next to Instance Metadata Service.

2. Choose the **Version 1 and version 2** option, and save your changes.

From Command Line

Run Below Command:

```
oci compute instance update --instance-id [instance-ocid] --instance-options '{"areLegacyImsEndpointsDisabled": "true"}'
```

This will set Instance Metadata Service to use Version 2 Only.

3.2 Ensure Secure Boot is enabled on Compute Instance

Shielded Instances with Secure Boot enabled prevents unauthorized boot loaders and operating systems from booting. This prevent rootkits, bootkits, and unauthorized software from running before the operating system loads. Secure Boot verifies the digital signature of the system's boot software to check its authenticity.

The digital signature ensures the operating system has not been tampered with and is from a trusted source. When the system boots and attempts to execute the software, it will first check the digital signature to ensure validity. If the digital signature is not valid, the system will not allow the software to run. Secure Boot is a feature of UEFI (Unified Extensible Firmware Interface) that only allows approved operating systems to boot up.

Rationale:

A Threat Actor with access to the operating system may seek to alter boot components to persist malware or rootkits during system initialization. Secure Boot helps ensure that the system only runs authentic software by verifying the digital signature of all boot components.

Impact:

An existing instance cannot be changed to a Shielded instance with Secure boot enabled. Shielded Secure Boot not available on all instance shapes and Operating systems. Additionally, the following limitations exist:

Thus, to enable you have to terminate the instance and create a new one. Also, Shielded instances do not support live migration. During an infrastructure maintenance event, Oracle Cloud Infrastructure live migrates supported VM instances from the physical VM host that needs maintenance to a healthy VM host with minimal disruption to running instances. If you enable Secure Boot on an instance, the instance cannot be migrated, because the hardware TPM is not migratable. This may result in an outage because the TPM can't be migrate from a unhealthy host to healthy host.

Result:

Tenancy MutantRogue: **Non-Compliant (384 of 393 item)**

Tenancy NewCoCobranca: **Non-Compliant (363 of 363 items)**

Remediation:

Note: Secure Boot facility is available on selected VM images and Shapes in OCI. User have to configure Secured Boot at time of instance creation only.

From OCI Console:

1. Navigate to <https://cloud.oracle.com/compute/instances>

2. Select the instance from the Audit Procedure
3. Click **Terminate**.
4. Determine whether or not to permanently delete instance's attached boot volume.
5. Click **Terminate instance**.
6. Click on **Create Instance**.
7. Select Image and Shape which supports Shielded Instance configuration. Icon for Shield in front of Image/Shape row indicates support of Shielded Instance.
8. Click on **edit** of Security Blade.
9. Turn On Shielded Instance, then Turn on the Secure Boot Toggle.
10. Fill in the rest of the details as per requirements.
11. Click **Create**.

Default Value:

Secure Boot is not Enabled

3.3 Ensure In-transit Encryption is enabled on Compute Instance

The Block Volume service provides the option to enable in-transit encryption for paravirtualized volume attachments on virtual machine (VM) instances.

Rationale:

All the data moving between the instance and the block volume is transferred over an internal and highly secure network. If you have specific compliance requirements related to the encryption of the data while it is moving between the instance and the block volume, you should enable the in-transit encryption option.

Impact:

In-transit encryption for boot and block volumes is only available for virtual machine (VM) instances launched from platform images, along with bare metal instances that use the following shapes: BM.Standard.E3.128, BM.Standard.E4.128, BM.DenseIO.E4.128. It is not supported on other bare metal instances.

Result:

Tenancy MutantRogue: **Non-Compliant (383 of 393 item)**

Tenancy NewCoCobranca: **Non-Compliant (363 of 363 items)**

Remediation:

Note: Secure Boot facility is available on selected VM images and Shapes in OCI. User have to configure Secured Boot at time of instance creation only.

From OCI Console:

1. Navigate to <https://cloud.oracle.com/compute/instances>

2. Select the instance from the Audit Procedure
3. Click **Terminate**.
4. Determine whether or not to permanently delete instance's attached boot volume.
5. Click **Terminate instance**.
6. Click on **Create Instance**.
7. Fill in the details as per requirements.
8. In the **Boot volume** section ensure **Use in-transit encryption** is checked.
9. Fill in the rest of the details as per requirements.
10. Click **Create**.

Default Value:

Enabled

4 - Logging and Monitoring

4.1 Ensure default tags are used on resources

Using default tags is a way to ensure all resources that support tags are tagged during creation. Tags can be based on static values or based on computed values. It is recommended to setup default tags early on to ensure all created resources will get tagged. Tags are scoped to Compartments and are inherited by Child Compartments. The recommendation is to create default tags like "CreatedBy" at the Root Compartment level to ensure all resources get tagged. When using Tags it is important to ensure that Tag Namespaces are protected by IAM Policies otherwise this will allow users to change tags or tag values. Depending on the age of the OCI Tenancy there may already be Tag defaults setup at the Root Level and no need for further action to implement this action.

Rationale:

In the case of an incident having default tags like "CreatedBy" applied will provide info on who created the resource without having to search the Audit logs.

Impact:

There is no performance impact when enabling the above described features.

Result:

Tenancy MutantRogue: **Compliant**

Tenancy NewCoCobranca: **Compliant**

4.2 Create at least one notification topic and subscription to receive monitoring alerts

Notifications provide a multi-channel messaging service that allow users and applications to be notified of events of interest occurring within OCI. Messages can be sent via eMail, HTTPs, PagerDuty, Slack or the OCI Function service. Some channels, such as eMail require confirmation of the subscription before it becomes active.

Rationale:

Creating one or more notification topics allow administrators to be notified of relevant changes made to OCI infrastructure.

Impact:

There is no performance impact when enabling the above described features but depending on the amount of notifications sent per month there may be a cost associated.

Result:

Tenancy MutantRogue: **Compliant**

Tenancy NewCoCobranca: **Compliant**

4.3 Ensure a notification is configured for Identity Provider changes

It is recommended to setup an Event Rule and Notification that gets triggered when Identity Providers are created, updated or deleted. Event Rules are compartment scoped and will detect events in child compartments. It is recommended to create the Event rule at the root compartment level.

Rationale:

OCI Identity Providers allow management of User ID / passwords in external systems and use of those credentials to access OCI resources. Identity Providers allow users to single sign-on to OCI console and have other OCI credentials like API Keys. Monitoring and alerting on changes to Identity Providers will help in identifying changes to the security posture.

Impact:

There is no performance impact when enabling the above-described features but depending on the amount of notifications sent per month there may be a cost associated.

Result:

Tenancy MutantRogue: **Non-Compliant**

Tenancy NewCoCobranca: **Non-Compliant**

There was no notification topic configured for Identity Provider changes on Tenancy MutantRogue nor NewCoCObranca.

Remediation:

From OCI Console:

1. Go to the **Events Service** page: <https://cloud.oracle.com/events/rules>
2. Select the **Compartment** that should host the rule
3. Click **Create Rule**
4. Provide a **Display Name** and **Description**
5. Create a Rule Condition by selecting **Identity** in the Service Name Drop-down and selecting **Identity Provider – Create, Identity Provider - Delete and Identity Provider – Update**
6. In the **Actions** section select **Notifications** as Action Type
7. Select the **Compartment** that hosts the Topic to be used.
8. Select the **Topic** to be used
9. Optionally add Tags to the Rule
10. Click **Create Rule**

From Command Line

1. Find the **topic-id** of the topic the Event Rule should use for sending notifications by using the topic **name** and **Compartment OCID**

```
oci ons topic list --compartment-id <compartment-ocid> --all --query "data [?name=='<topic-name>']".{"name:name,topic_id:\"topic-id\""} --output table
```

2. Create a JSON file called “event_rule.json” to be used when creating the Event Rule. Replace topic id, display name, description and compartment OCID

```
{
  "actions":
    {
      "actions": [
        {
          "actionType": "ONS",
          "isEnabled": true,
          "topicId": "<topic-id>"
        }
      ]
    },
  "condition":
    "{\\\"eventType\\\":[\\\"com.oraclecloud.identitycontrolplane.createidentityprovider\\\",\\\"com.oraclecloud.identitycontrolplane.deleteidentityprovider\\\",\\\"com.oraclecloud.identitycontrolplane.updateidentityprovider\\\"],\\\"data\\\":{}}",
  "displayName": "<display-name>",
  "description": "<description>",
  "isEnabled": true,
  "compartmentId": "<compartment-ocid>"
}
```

3. Create the actual event rule using the file created above:

```
oci events rule create --from-json file://event_rule.json
```

4. Note in the JSON returned that it lists the parameters specified in the JSON file provided and that there is an OCID provided for the Event Rule

Additional Information:

- The same Notification topic can be reused by many Event Rules.
- The generated notification will include an eventID that can be used when querying the Audit Logs in case further investigation is required.

4.4 Ensure a notification is configured for IdP group mapping changes

It is recommended to setup an Event Rule and Notification that gets triggered when Identity Provider Group Mappings are created, updated or deleted. Event Rules are compartment scoped and will detect events in child compartments. It is recommended to create the Event rule at the root compartment level.

Rationale:

IAM Policies govern access to all resources within an OCI Tenancy. IAM Policies use OCI Groups for assigning the privileges. Identity Provider Groups could be mapped to OCI Groups to assign privileges to federated users in OCI. Monitoring and alerting on changes to Identity Provider Group mappings will help in identifying changes to the security posture.

Impact:

There is no performance impact when enabling the above described features but depending on the amount of notifications sent per month there may be a cost associated.

Result:

Tenancy MutantRogue: **Non-Compliant**

Tenancy NewCoCobranca: **Non-Compliant**

There was no notification topic configured on Tenancy MutantRogue nor NewCoCObranca.

Remediation:

From OCI Console:

1. Go to the **Events Service** page: <https://cloud.oracle.com/events/rules>
2. Select the **Compartment** that should host the rule
3. Click **Create Rule**
4. Provide a **Display Name** and **Description**
5. Create a Rule Condition by selecting **Identity** in the Service Name Drop-down and selecting **Idp Group Mapping – Create, Idp Group Mapping – Delete** and **Idp Group Mapping – Update**
6. In the **Actions** section select **Notifications** as Action Type
7. Select the **Compartment** that hosts the Topic to be used.
8. Select the **Topic** to be used
9. Optionally add Tags to the Rule
10. Click **Create Rule**.

From Command Line

1. Find the **topic-id** of the Event Rule which should be used for sending Notifications by using the topic **name** and **Compartment OCID**

```
oci ons topic list --compartment-id <compartment-ocid> --all --query "data [?name=='<topic-name>']".{"name:name,topic_id:\"topic-id\""} --output table
```

2. Create a JSON file to be used when creating the Event Rule. Replace topic id, display name, description, and compartment OCID

```
{
  "actions":
  {
    "actions": [
      {
        "actionType": "ONS",
        "isEnabled": true,
        "topicId": "<topic-id>"
      }
    ]
  },
  "condition":
  "{ \"eventType\": [ \"com.oraclecloud.identitycontrolplane.createidpgroupmapping\", \"com.oraclecloud.identitycontrolplane.deleteidpgroupmapping\", \"com.oraclecloud.identitycontrolplane.updateidpgroupmapping\" ], \"data\": { } },
  \"displayName\": \"<display-name>\",
  \"description\": \"<description>\",
  \"isEnabled\": true,
  \"compartmentId\": \"<compartment-ocid>\"
}
```

3. Create a JSON file to be used when creating the Event Rule. Replace topic id, display name, description, and compartment OCID

```
oci events rule create --from-json file://event_rule.json
```

4. Note in the JSON returned that it lists the parameters specified in the JSON file provided and that there is an OCID provided for the Event Rule

Additional Information:

- The same Notification topic can be reused by many Event Rules.

The generated notification will include an eventID that can be used when querying the Audit Logs in case further investigation is required.

4.5 Ensure a notification is configured for IAM group changes

It is recommended to setup an Event Rule and Notification that gets triggered when IAM Groups are created, updated or deleted. Event Rules are compartment scoped and will detect events in child compartments, it is recommended to create the Event rule at the root compartment level.

Rationale:

IAM Groups control access to all resources within an OCI Tenancy. Monitoring and alerting on changes to IAM Groups will help in identifying changes to satisfy least privilege principle.

Impact:

There is no performance impact when enabling the above described features but depending on the amount of notifications sent per month there may be a cost associated.

Result:

Tenancy MutantRogue: **Non-Compliant**

Tenancy NewCoCobranca: **Non-Compliant**

There was no notification topic configured on Tenancy MutantRogue nor NewCoCObranca.

Remediation:

From OCI Console:

1. Go to the **Events Service** page: <https://cloud.oracle.com/events/rules>
2. Select the **Compartment** that should host the rule
3. Click **Create Rule**
4. Provide a **Display Name** and **Description**
5. Create a Rule Condition by selecting **Identity** in the Service Name Drop-down and selecting **Group – Create**, **Group – Delete** and **Group – Update**

6. In the **Actions** section select **Notifications** as Action Type
7. Select the **Compartment** that hosts the Topic to be used.
8. Select the **Topic** to be used
9. Optionally add Tags to the Rule
10. Click **Create Rule**

From Command Line:

1. Find the **topic-id** of the Event Rule which should be used for sending Notifications by using the topic **name** and **Compartment OCID**

```
oci ons topic list --compartment-id <compartment-ocid> --all --query "data
[?name=='<topic-name>'].{\"name:name,topic_id:\"topic-id\"}" --output table
```

2. Create a JSON file to be used when creating the Event Rule. Replace topic id, display name, description and compartment OCID

```
{
  "actions":
  {
    "actions": [
      {
        "actionType": "ONS",
        "isEnabled": true,
        "topicId": "<topic-id>"
      }
    ]
  },
  "condition":
  "{\\"eventType\":[\\\"com.oraclecloud.identitycontrolplane.creategroup\\\",\\\"com.oraclecloud.identitycontrolplane.delet
egroup\\\",\\\"com.oraclecloud.identitycontrolplane.updategroup\\\"],\\\"data\":{}}",
  "displayName": "<display-name>",
  "description": "<description>",
  "isEnabled": true,
  "compartmentId": "<compartment-ocid>"
}
```

3. Create the actual event rule

```
oci events rule create --from-json file://event_rule.json
```

4. Note in the JSON returned that it lists the parameters specified in the JSON file provided and that there is an OCID provided for the Event Rule

Additional Information:

- The same Notification topic can be reused by many Event Rules.
- The generated notification will include an eventID that can be used when querying the Audit Logs in case further investigation is required.

4.6 Ensure a notification is configured for IAM policy changes

It is recommended to setup an Event Rule and Notification that gets triggered when IAM Policies are created, updated or deleted. Event Rules are compartment scoped and will detect events in child compartments, it is recommended to create the Event rule at the root compartment level.

Rationale:

IAM Policies govern access to all resources within an OCI Tenancy. Monitoring and alerting on changes to IAM policies will help in identifying changes to the security posture.

Impact:

There is no performance impact when enabling the above described features but depending on the amount of notifications sent per month there may be a cost associated.

Result:

Tenancy MutantRogue: **Non-Compliant**

Tenancy NewCoCobranca: **Non-Compliant**

There was no notification topic configured on Tenancy MutantRogue nor NewCoCObranca.

Remediation:

From OCI Console:

1. Go to the **Events Service** page: <https://cloud.oracle.com/events/rules>
2. Select the **Compartment** that should host the rule
3. Click **Create Rule**
4. Provide a **Display Name** and **Description**
5. Create a Rule Condition by selecting **Identity** in the Service Name Drop-down and selecting **Policy – Change Compartment, Policy – Create, Policy – Delete** and **Policy – Update**
6. In the **Actions** section select **Notifications** as Action Type
7. Select the **Compartment** that hosts the Topic to be used.
8. Select the **Topic** to be used
9. Optionally add Tags to the Rule
10. Click **Create Rule**

From Command Line:

1. Find the **topic-id** of the Event Rule which should be used for sending Notifications by using the topic **name** and **Compartment OCID**

```
oci ons topic list --compartment-id <compartment-ocid> --all --query "data\n[?name=='<topic-name>']".{"name:name,topic_id:\\\"topic-id\\\""} --output table
```

2. Create a JSON file to be used when creating the Event Rule. Replace topic id, display name, description and compartment OCID

```
{\n  \"actions\":\n  {\n    \"actions\": [\n      {\n        \"actionType\": \"ONS\",\n        \"isEnabled\": true,\n        \"topicId\": \"<topic-id>\"\n      }\n    ]\n  }\n}
```

```

    ]]
  },
  "condition":
    "{\n\"eventType\":[\n\"com.oraclecloud.identitycontrolplane.createpolicy\", \n\"com.oraclecloud.identitycontrolplane.dele
tepolicy\", \n\"com.oraclecloud.identitycontrolplane.updatepolicy\"], \n\"data\":{}}",
    "displayName": "<display-name>",
    "description": "<description>",
    "isEnabled": true,
    "compartmentId": "<compartment-ocid>"
  }
}

```

3. Create the actual event rule

```
oci events rule create --from-json file://event_rule.json
```

- Note in the JSON returned that it lists the parameters specified in the JSON file provided and that there is an OCID provided for the Event Rule

Additional Information:

- The same Notification topic can be reused by many Event Rules.
- The generated notification will include an eventID that can be used when querying the Audit Logs in case further investigation is required.

4.7 Ensure a notification is configured for user changes

It is recommended to setup an Event Rule and Notification that gets triggered when IAM Users are created, updated, deleted, capabilities updated, or state updated. Event Rules are compartment scoped and will detect events in child compartments, it is recommended to create the Event rule at the root compartment level.

Rationale:

Users use or manage Oracle Cloud Infrastructure resources. Monitoring and alerting on changes to Users will help in identifying changes to the security posture.

Impact:

There is no performance impact when enabling the above described features but depending on the amount of notifications sent per month there may be a cost associated.

Result:

Tenancy MutantRogue: **Non-Compliant**

Tenancy NewCoCobranca: **Non-Compliant**

There was no notification topic configured on Tenancy MutantRogue nor NewCoCObranca.

Remediation:

From OCI Console:

- Using the search box to navigate to **events**
- Navigate to the **rules** page
- Select the **compartment** that should host the rule

4. Click **Create Rule**
5. Provide a **Display Name** and **Description**
6. Create a Rule Condition by selecting **Identity** in the Service Name Drop-down and selecting:
 - a. **User – Create,**
 - b. **User – Delete,**
 - c. **User – Update,**
 - d. **User Capabilities – Update,**
 - e. **User State – Update**
7. In the **Actions** section select **Notifications** as Action Type
8. Select the **Compartment** that hosts the Topic to be used.
9. Select the **Topic** to be used
10. Optionally add Tags to the Rule
11. Click **Create Rule**

From Command Line:

1. Find the **topic-id** of the Event Rule which should be used for sending Notifications by using the topic **name** and **Compartment OCID**

```
oci ons topic list --compartment-id <compartment-ocid> --all --query "data[?name=='<topic-name>']".{"name:name,topic_id:\"topic-id\""} --output table
```

2. Create a JSON file to be used when creating the Event Rule. Replace topic id, display name, description and compartment OCID

```
{
  "actions":
  {
    "actions": [
      {
        "actionType": "ONS",
        "isEnabled": true,
        "topicId": "<topic-id>"
      }
    ]
  },
  "condition":
  "{\\"eventType\":[\"com.oraclecloud.identitycontrolplane.createuser\", \"com.oraclecloud.identitycontrolplane.deleteuser\", \"com.oraclecloud.identitycontrolplane.updateuser\", \"com.oraclecloud.identitycontrolplane.updateusercapabilities\", \"com.oraclecloud.identitycontrolplane.updateuserstate\"], \"data\": {}}"
,
  "displayName": "<display-name>",
  "description": "<description>",
  "isEnabled": true,
  "compartmentId": "<compartment-ocid>"
}
```

3. Create the actual event rule

```
oci events rule create --from-json file://event_rule.json
```

4. Note in the JSON returned that it lists the parameters specified in the JSON file provided and that there is an OCID provided for the Event Rule

Additional Information:

- The same Notification topic can be reused by many Event Rules.
- The generated notification will include an eventID that can be used when querying the Audit Logs in case further investigation is required.

4.8 Ensure a notification is configured for VCN changes

It is recommended to setup an Event Rule and Notification that gets triggered when Virtual Cloud Networks are created, updated or deleted. Event Rules are compartment scoped and will detect events in child compartments, it is recommended to create the Event rule at the root compartment level.

Rationale:

Virtual Cloud Networks (VCNs) closely resembles a traditional network. Monitoring and alerting on changes to VCNs will help in identifying changes to the security posture.

Impact:

There is no performance impact when enabling the above described features but depending on the amount of notifications sent per month there may be a cost associated.

Result:

Tenancy MutantRogue: **Non-Compliant**

Tenancy NewCoCobranca: **Non-Compliant**

There was no notification topic configured on Tenancy MutantRogue nor NewCoCObranca.

Remediation:

From OCI Console:

1. Go to the **Events Service** page: <https://cloud.oracle.com/events/rules>
2. Select the **compartment** that should host the rule
3. Click **Create Rule**
4. Provide a **Display Name** and **Description**
5. Create a Rule Condition by selecting **Networking** in the Service Name Drop-down and selecting **VCN – Create, VCN - Delete and VCN – Update**
6. In the **Actions** section select **Notifications** as Action Type
7. Select the **Compartment** that hosts the Topic to be used.
8. Select the **Topic** to be used
9. Optionally add Tags to the Rule
10. Click **Create Rule**

From Command Line:

1. Find the **topic-id** of the Event Rule which should be used for sending Notifications by using the topic **name** and **Compartment OCID**

```
oci ons topic list --compartment-id <compartment-ocid> --all --query "data[?name=='<topic-name>']".{"name:name,topic_id:\"topic-id\""} --output table
```

2. Create a JSON file to be used when creating the Event Rule. Replace topic id, display name, description and compartment OCID

```
{
  "actions":
  {
    "actions": [
      {
        "actionType": "ONS",
        "isEnabled": true,
        "topicId": "<topic-id>"
      }
    ]
  },
  "condition":
  [{"eventType":["com.oraclecloud.virtualnetwork.createvcn","\com.oraclecloud.virtualnetwork.deletevcn","\com.o
raclecloud.virtualnetwork.updatevcn"],"data\:{}}",
  "displayName": "<display-name>",
  "description": "<description>",
  "isEnabled": true,
  "compartmentId": "<compartment-ocid>"
}
```

3. Create the actual event rule

```
oci events rule create --from-json file://event_rule.json
```

4. Note in the JSON returned that it lists the parameters specified in the JSON file provided and that there is an OCID provided for the Event Rule

Additional Information:

- The same Notification topic can be reused by many Event Rules.
- The generated notification will include an eventID that can be used when querying the Audit Logs in case further investigation is required.

4.9 Ensure a notification is configured for changes to route tables

It is recommended to setup an Event Rule and Notification that gets triggered when route tables are created, updated or deleted. Event Rules are compartment scoped and will detect events in child compartments, it is recommended to create the Event rule at the root compartment level.

Rationale:

Route tables control traffic flowing to or from Virtual Cloud Networks and Subnets. Monitoring and alerting on changes to route tables will help in identifying changes these traffic flows.

Impact:

There is no performance impact when enabling the above described features but depending on the amount of notifications sent per month there may be a cost associated.

Result:

Tenancy MutantRogue: **Non-Compliant**

Tenancy NewCoCobranca: **Non-Compliant**

There was no notification topic configured on Tenancy MutantRogue nor NewCoCObranca.

Remediation:

From OCI Console:

1. Go to the **Events Service** page: <https://cloud.oracle.com/events/rules>
2. Select the **compartment** that should host the rule
3. Click **Create Rule**
4. Provide a **Display Name** and **Description**
5. Create a Rule Condition by selecting **Networking** in the Service Name Drop-down and selecting **Route Table – Change Compartment, Route Table – Create, Route Table - Delete and Route Table – Update**
6. In the **Actions** section select **Notifications** as Action Type
7. Select the **Compartment** that hosts the Topic to be used.
8. Select the **Topic** to be used
9. Optionally add Tags to the Rule
10. Click **Create Rule**

From Command Line:

1. Find the **topic-id** of the Event Rule which should be used for sending Notifications by using the topic **name** and **Compartment OCID**

```
oci ons topic list --compartment-id <compartment-ocid> --all --query "data[?name=='<topic-name>']".{"name:name,topic_id:\"topic-id\""} --output table
```

2. Create a JSON file to be used when creating the Event Rule. Replace topic id, display name, description and compartment OCID

```
{
  "actions":
  {
    "actions": [
      {
        "actionType": "ONS",
        "isEnabled": true,
        "topicId": "<topic-id>"
      }
    ]
  },
  "condition":
  "{ \"eventType\": [ \"com.oraclecloud.virtualnetwork.changeroutetablecompartment\", \"com.oraclecloud.virtualnetwork.createroutetable\", \"com.oraclecloud.virtualnetwork.deleteroutetable\", \"com.oraclecloud.virtualnetwork.updateroutetable\" ], \"data\": { } }",
  "displayName": "<display-name>",
  "description": "<description>",
  "isEnabled": true,
  "compartmentId": "<compartment-ocid>"
}
```

3. Create the actual event rule

```
oci events rule create --from-json file://event_rule.json
```

4. Note in the JSON returned that it lists the parameters specified in the JSON file provided and that there is an OCID provided for the Event Rule

Additional Information:

- The same Notification topic can be reused by many Event Rules.
- The generated notification will include an eventID that can be used when querying the Audit Logs in case further investigation is required.

4.10 Ensure a notification is configured for security list changes

It is recommended to setup an Event Rule and Notification that gets triggered when security lists are created, updated or deleted. Event Rules are compartment scoped and will detect events in child compartments, it is recommended to create the Event rule at the root compartment level.

Rationale:

Security Lists control traffic flowing into and out of Subnets within a Virtual Cloud Network. Monitoring and alerting on changes to Security Lists will help in identifying changes to these security controls.

Impact:

There is no performance impact when enabling the above described features but depending on the amount of notifications sent per month there may be a cost associated.

Result:

Tenancy MutantRogue: **Non-Compliant**

Tenancy NewCoCobranca: **Non-Compliant**

There was no notification topic configured on Tenancy MutantRogue nor NewCoCObranca.

Remediation:

From OCI Console:

1. Go to the **Events Service** page: <https://cloud.oracle.com/events/rules>
2. Select the **compartment** that should host the rule
3. Click **Create Rule**
4. Provide a **Display Name** and **Description**
5. Create a Rule Condition by selecting **Networking** in the Service Name Drop-down and selecting **Security List – Change Compartment, Security List – Create, Security List - Delete and Security List – Update**
6. In the **Actions** section select **Notifications** as Action Type
7. Select the **Compartment** that hosts the Topic to be used.

8. Select the **Topic** to be used
9. Optionally add Tags to the Rule
10. Click **Create Rule**.

From Command Line:

1. Find the **topic-id** of the Event Rule which should be used for sending Notifications by using the topic **name** and **Compartment OCID**

```
oci ons topic list --compartment-id <compartment-ocid> --all --query "data[?name=='<topic-name>']".{"name:name,topic_id:\"topic-id\""} --output table
```

2. Create a JSON file to be used when creating the Event Rule. Replace topic id, display name, description and compartment OCID

```
{
  "actions":
  {
    "actions": [
      {
        "actionType": "ONS",
        "isEnabled": true,
        "topicId": "<topic-id>"
      }
    ]
  },
  "condition":
  "{ \"eventType\": [\"com.oraclecloud.virtualnetwork.changesecuritylistcompartment\", \"com.oraclecloud.virtualnetwork.createsecuritylist\", \"com.oraclecloud.virtualnetwork.deletesecuritylist\", \"com.oraclecloud.virtualnetwork.updatesecuritylist\"], \"data\": {}}",
  "displayName": "<display-name>",
  "description": "<description>",
  "isEnabled": true,
  "compartmentId": "<compartment-ocid>"
}
```

3. Create the actual event rule

```
oci events rule create --from-json file://event_rule.json
```

4. Note in the JSON returned that it lists the parameters specified in the JSON file provided and that there is an OCID provided for the Event Rule

Additional Information:

- The same Notification topic can be reused by many Event Rules.
- The generated notification will include an eventID that can be used when querying the Audit Logs in case further investigation is required.

4.11 Ensure a notification is configured for network security group changes

It is recommended to setup an Event Rule and Notification that gets triggered when network security groups are created, updated or deleted. Event Rules are compartment scoped and will detect events in child compartments, it is recommended to create the Event rule at the root compartment level.

Rationale:

Network Security Groups control traffic flowing between Virtual Network Cards attached to Compute instances. Monitoring and alerting on changes to Network Security Groups will help in identifying changes these security controls.

Impact:

There is no performance impact when enabling the above described features but depending on the amount of notifications sent per month there may be a cost associated.

Result:

Tenancy MutantRogue: **Non-Compliant**

Tenancy NewCoCobranca: **Non-Compliant**

There was no notification topic configured on Tenancy MutantRogue nor NewCoCObranca.

Remediation:**From OCI Console:**

1. Go to the **Events Service** page: <https://cloud.oracle.com/events/rules>
2. Select the **compartment** that should host the rule
3. Click **Create Rule**
4. Provide a **Display Name** and **Description**
5. Create a Rule Condition by selecting **Networking** in the Service Name Drop-down and selecting **Network Security Group – Change Compartment, Network Security Group – Create, Network Security Group - Delete** and **Network Security Group – Update**
6. In the **Actions** section select **Notifications** as Action Type
7. Select the **Compartment** that hosts the Topic to be used.
8. Select the **Topic** to be used
9. Optionally add Tags to the Rule
10. Click **Create Rule**

From Command Line:

1. Find the **topic-id** of the Event Rule which should be used for sending Notifications by using the topic **name** and **Compartment OCID**

```
oci ons topic list --compartment-id <compartment-ocid> --all --query "data[?name=='<topic-name>']".{"name:name,topic_id:\"topic-id\""} --output table
```

2. Create a JSON file to be used when creating the Event Rule. Replace topic id, display name, description and compartment OCID

```
{
  "actions": {
    "actions": [
      {
        "actionType": "ONS",
```

```

        "isEnabled": true,
        "topicId": "<topic-id>"
    }
  ]
},
"condition":
"{\"eventType\": [\"com.oraclecloud.virtualnetwork.changenetworksecuritygroupcompartment\", \"com.oraclecloud.virtualnetwork.createnetworksecuritygroup\", \"com.oraclecloud.virtualnetwork.deletenetworksecuritygroup\", \"com.oraclecloud.virtualnetwork.updatenetworksecuritygroup\"], \"data\": {}}",
"displayName": "<display-name>",
"description": "<description>",
"isEnabled": true,
"compartmentId": "<compartment-ocid>"
}

```

3. Create the actual event rule

```
oci events rule create --from-json file://event_rule.json
```

- Note in the JSON returned that it lists the parameters specified in the JSON file provided and that there is an OCID provided for the Event Rule

Additional Information:

- The same Notification topic can be reused by many Event Rules.
- The generated notification will include an eventID that can be used when querying the Audit Logs in case further investigation is required.

4.12 Ensure a notification is configured for changes to network gateways

It is recommended to setup an Event Rule and Notification that gets triggered when Network Gateways are created, updated, deleted, attached, detached, or moved. This recommendation includes Internet Gateways, Dynamic Routing Gateways, Service Gateways, Local Peering Gateways, and NAT Gateways. Event Rules are compartment scoped and will detect events in child compartments, it is recommended to create the Event rule at the root compartment level.

Rationale:

Network Gateways act as routers between VCNs and the Internet, Oracle Services Networks, other VCNS, and on-premise networks. Monitoring and alerting on changes to Network Gateways will help in identifying changes to the security posture.

Impact:

There is no performance impact when enabling the above described features but depending on the amount of notifications sent per month there may be a cost associated.

Result:

Tenancy MutantRogue: **Non-Compliant**

Tenancy NewCoCobranca: **Non-Compliant**

There was no notification topic configured on Tenancy MutantRogue nor NewCoCObranca.

Remediation:

From OCI Console:

1. Go to the **Events Service** page: <https://cloud.oracle.com/events/rules>
2. Select the **compartment** that should host the rule
3. Click **Create Rule**
4. Provide a **Display Name** and **Description**
5. Create a Rule Condition by selecting Networking in the Service Name Drop-down and selecting

```
DRG - Create
DRG - Delete
DRG - Update
DRG Attachment - Create
DRG Attachment - Delete
DRG Attachment - Update
Internet Gateway - Create
Internet Gateway - Delete
Internet Gateway - Update
Internet Gateway - Change Compartment
Local Peering Gateway - Create
Local Peering Gateway - Delete End
Local Peering Gateway - Update
Local Peering Gateway - Change Compartment
NAT Gateway - Create
NAT Gateway - Delete
NAT Gateway - Update
NAT Gateway - Change Compartment
Service Gateway - Create
Service Gateway - Delete End
Service Gateway - Update
Service Gateway - Attach Service
Service Gateway - Detach Service
Service Gateway - Change Compartment
```

6. In the **Actions** section select **Notifications** as Action Type
7. Select the **Compartment** that hosts the Topic to be used.
8. Select the **Topic** to be used
9. Optionally add Tags to the Rule
10. Click **Create Rule**

From Command Line:

1. Find the **topic-id** of the Event Rule which should be used for sending Notifications by using the topic **name** and **Compartment OCID**

```
oci ons topic list --compartment-id <compartment-ocid> --all --query "data[?name=='<topic-name>']".{"name:name,topic_id:\"topic-id\""} --output table
```

2. Create a JSON file to be used when creating the Event Rule. Replace topic id, display name, description and compartment OCID

```
{
  "actions": {
    "actions": [
      {
        "actionType": "ONS",
        "isEnabled": true,
        "topicId": "<topic-id>"
      }
    ]
  }
}
```

```

    ]
  },
  "condition":
    "{\n\"eventType\":\n\"com.oraclecloud.virtualnetwork.createdrg\", \"com.oraclecloud.virtualnetwork.deletedrg\", \"com.oraclecloud.virtualnetwork.updatedrg\", \"com.oraclecloud.virtualnetwork.createdrgattachment\", \"com.oraclecloud.virtualnetwork.deletedrgattachment\", \"com.oraclecloud.virtualnetwork.updatedrgattachment\", \"com.oraclecloud.virtualnetwork.changeinternetgatewaycompartment\", \"com.oraclecloud.virtualnetwork.createinternetgateway\", \"com.oraclecloud.virtualnetwork.deleteinternetgateway\", \"com.oraclecloud.virtualnetwork.updateinternetgateway\", \"com.oraclecloud.virtualnetwork.changelocalpeeringgatewaycompartment\", \"com.oraclecloud.virtualnetwork.createlocalpeeringgateway\", \"com.oraclecloud.virtualnetwork.deletelocalpeeringgateway.end\", \"com.oraclecloud.virtualnetwork.updatelocalpeeringgateway\", \"com.oraclecloud.natgateway.changenatgatewaycompartment\", \"com.oraclecloud.natgateway.createnatgateway\", \"com.oraclecloud.natgateway.deletenatgateway\", \"com.oraclecloud.natgateway.updatenatgateway\", \"com.oraclecloud.servicegateway.attachserviceid\", \"com.oraclecloud.servicegateway.changeservicegatewaycompartment\", \"com.oraclecloud.servicegateway.createservicegateway\", \"com.oraclecloud.servicegateway.deleteservicegateway.end\", \"com.oraclecloud.servicegateway.detachserviceid\", \"com.oraclecloud.servicegateway.updateservicegateway\"],\n\"data\":\n{}\"}",
    "displayName": "<display-name>",
    "description": "<description>",
    "isEnabled": true,
    "compartmentId": "<compartment-ocid>"
  }
}

```

3. Create the actual event rule

```
oci events rule create --from-json file://event_rule.json
```

- Note in the JSON returned that it lists the parameters specified in the JSON file provided and that there is an OCID provided for the Event Rule

Additional Information:

- The same Notification topic can be reused by many Event Rules.
- The generated notification will include an eventID that can be used when querying the Audit Logs in case further investigation is required.

4.13 Ensure VCN flow logging is enabled for all subnets

VCN flow logs record details about traffic that has been accepted or rejected based on the security list rule.

Rationale:

Enabling VCN flow logs enables you to monitor traffic flowing within your virtual network and can be used to detect anomalous traffic.

Impact:

Enabling VCN flow logs will not affect the performance of your virtual network but it will generate additional use of object storage that should be controlled via object lifecycle management.

By default, VCN flow logs are stored for 30 days in object storage. Users can specify a longer retention period.

Result:

Tenancy MutantRogue: **Non-Compliant (155 of 161 items)**

Tenancy NewCoCobranca: **Non-Compliant (90 of 102 items)**

Remediation:

From OCI Console:

First, if a Capture filter has not already been created, create a Capture Filter by the following steps:

1. Go to the Network Command Center page
(<https://cloud.oracle.com/networking/network-command-center>)
2. Click 'Capture filters'
3. Click 'Create Capture filter'
4. Type a name for the Capture filter in the Name box.
5. Select 'Flow log capture filter'
6. For **Sample rating** select **100%**
7. Scroll to **Rules**
8. For **Traffic disposition** select **All**
9. For **Include/Exclude** select **Include**
10. Level **Source IPv4 CIDR or IPv6 prefix** and **Destination IPv4 CIDR or IPv6 prefix** empty
11. For **IP protocol** select **Include**
12. Click **Create Capture filter**

Second, enable VCN flow logging for your VCN or subnet(s) by the following steps:

1. Go to the Logs page (<https://cloud.oracle.com/logging/logs>)
2. Click the **Enable Service Log** button in the middle of the screen.
3. Select the relevant resource compartment.
4. Select **Virtual Cloud Networks - Flow logs** from the Service drop down menu.

5. Select the relevant resource level from the resource drop down menu either **VCN** or **subnet**.
6. Select the relevant resource from the resource drop down menu.
7. Select the from the Log Category drop down menu that either **Flow Logs - subnet records** or **Flow Logs - vcn records**.
8. Select the Capture filter from above
9. Type a name for your flow logs in the Log Name text box.
10. Select the Compartment for the Log Location
11. Select the Log Group for the Log Location or Click **Create New Group** to create a new log group
12. Click the Enable Log button in the lower left-hand corner.

4.14 Ensure Cloud Guard is enabled in the root compartment of the tenancy

Cloud Guard detects misconfigured resources and insecure activity within a tenancy and provides security administrators with the visibility to resolve these issues. Upon detection, Cloud Guard can suggest, assist, or take corrective actions to mitigate these issues. Cloud Guard should be enabled in the root compartment of your tenancy with the default configuration, activity detectors and responders.

Rationale:

Cloud Guard provides an automated means to monitor a tenancy for resources that are configured in an insecure manner as well as risky network activity from these resources.

Impact:

There is no performance impact when enabling the above described features, but additional IAM policies will be required.

Result:

Tenancy MutantRogue: **Compliant**

Tenancy NewCoCobranca: **Compliant**

Cloud Guard is enabled in the root compartment of the Tenancies.

4.15 Ensure a notification is configured for Oracle Cloud Guard problems detected

Cloud Guard detects misconfigured resources and insecure activity within a tenancy and provides security administrators with the visibility to resolve these issues. Upon detection, Cloud Guard generates a Problem. It is recommended to setup an Event Rule and Notification that gets triggered when Oracle Cloud Guard Problems are created, dismissed or remediated. Event Rules are compartment scoped and will detect events in child compartments. It is recommended to create the Event rule at the root compartment level.

Rationale:

Cloud Guard provides an automated means to monitor a tenancy for resources that are configured in an insecure manner as well as risky network activity from these resources. Monitoring and alerting on Problems detected by Cloud Guard will help in identifying changes to the security posture.

Impact:

There is no performance impact when enabling the above described features but depending on the amount of notifications sent per month there may be a cost associated.

Result:

Tenancy MutantRogue: **Non-Compliant**

Tenancy NewCoCobranca: **Non-Compliant**

There was no notification topic configured for Oracle Cloud Guard problems detected.

Remediation:

From OCI Console:

1. Go to the Events Service page: <https://cloud.oracle.com/events/rules>
2. Select the **Compartment** that hosts the rules
3. Click **Create Rule**
4. Provide a **Display Name** and **Description**
5. Create a Rule Condition by selecting Cloud Guard in the Service Name Dropdown and selecting: **Detected – Problem, Remediated – Problem, and Dismissed - Problem**
6. In the **Actions** section select **Notifications** as Action Type
7. Select the **Compartment** that hosts the Topic to be used.
8. Select the **Topic** to be used
9. Optionally add Tags to the Rule
10. Click **Create Rule**

From Command Line:

1. Find the **topic-id** of the topic the Event Rule should use for sending Notifications by using the topic **name** and **Compartment OCID**

```
oci ons topic list --compartment-id=<compartment OCID> --all --query "data[?name=='<topic_name>'].{\"name:name,topic_id:\\\"topic-id\\\"}" --output table
```

2. Create a JSON file to be used when creating the Event Rule. Replace topic id, display name, description and compartment OCID

```
{
  "actions":
  {
    "actions": [
      {
        "actionType": "ONS",
```

```

        "isEnabled": true,
        "topicId": "<topic id>"
    }
  ],
  "condition":
  "{\\"eventType\":[\\"com.oraclecloud.cloudguard.problemdetected\\",\\"com.oraclecloud.cloudguard.problemdismissed\\",\\"com.oraclecloud.cloudguard.problemed remediated\\"],\\"data\":"}}",
  "displayName": "<display name>",
  "description": "<description>",
  "isEnabled": true,
  "compartmentId": "compartment OCID"
}ccc

```

3. Create the actual event rule

```
oci events rule create --from-json file://event_rule.json
```

- Note in the JSON returned that it lists the parameters specified in the JSON file provided and that there is an OCID provided for the Event Rule

Additional Information:

- Your tenancy might have a different Cloud Reporting region than your home region.
- The same Notification topic can be reused by many Event Rules.
- The generated notification will include an eventID that can be used when querying the Audit Logs in case further investigation is required.

4.16 Ensure customer created Customer Managed Key (CMK) is rotated at least annually

Oracle Cloud Infrastructure Vault securely stores master encryption keys that protect your encrypted data. You can use the Vault service to rotate keys to generate new cryptographic material. Periodically rotating keys limits the amount of data encrypted by one key version.

Rationale:

Rotating keys annually limits the data encrypted under one key version. Key rotation there by reducing the risk in case a key is ever compromised.

Result:

Tenancy MutantRogue: **Non-Compliant**

Tenancy NewCoCobranca: **Compliant**

Remediation

From Console

- Login to [OCI Console](#).
- Select **Identity & Security** from the Services menu.
- Select **Vault**.

4. Click on the individual Vault under the Name heading.
5. Click on the menu next to the time created.
6. Click **Rotate Key**.

From CLI

1. Execute the following:

```
oci kms management key rotate --key-id <key-ocid> --endpoint <managementendpoint-url>
```

4.17 Write level Object Storage logging is enabled for all buckets

Object Storage write logs will log all write requests made to objects in a bucket.

Rationale:

Enabling an Object Storage write log, the **requestAction** property would contain values of **PUT**, **POST**, or **DELETE**. This will provide you more visibility into changes to objects in your buckets.

Impact:

There is no performance impact when enabling the above described features, but will generate additional use of object storage that should be controlled via object lifecycle management.

By default, Object Storage logs are stored for 30 days in object storage. Users can specify a longer retention period.

Result:

Tenancy MutantRogue: **Non-Compliant (166 of 166 items)**

Tenancy NewCoCobranca: **Non-Compliant (158 of 158 items)**

Remediation:

From OCI Console:

First, if a log group for holding these logs has not already been created, create a log group by the following steps

1. Go to the Log Groups page <https://cloud.oracle.com/logging/log-groups>
2. Click the Create Log Groups button in the middle of the screen.
3. Select the relevant compartment to place these logs.
4. Type a name for the log group in the Name box.
5. Add an optional description in the Description box.
6. Click the Create button in the lower left-hand corner.

Second, enable Object Storage write log logging for your bucket(s) by the following steps:

1. Go to the Logs page <https://cloud.oracle.com/logging/logs>
2. Click the Enable Service Log button in the middle of the screen.

3. Select the relevant resource compartment.
4. Select Object Storage from the Service drop down menu.
5. Select the relevant bucket from the resource drop down menu.
6. Select 'Write Access Events' from the Log Category drop down menu.
7. Type a name for your Object Storage write log in the Log Name drop down menu.
8. Click the **Enable Log** button in the lower left-hand corner.

From Command Line:

First, if a log group for holding these logs has not already been created, create a log group by the following steps:

1. Create a log group

```
oci logging log-group create --compartment-id <compartment-id> \
--display-name "<DisplayName>" \
--description "<Description>"
```

The output of the command gives you a work request id. You can query the work request to see the status of the job by issuing the following command:

```
oci logging work-request get --work-request-id <work-request-id>
```

Look for status filed to be **SUCCEEDED**.

Second, enable Object Storage write log logging for your bucket(s) by the following steps:

1. Get the Log group ID needed for creating the Log

```
oci logging log-group list --compartment-id <compartment-id> \
--query 'data[?contains("display-name", `"`<DisplayName>`)].id|join(`\n`, @)' \
--raw-output
```

2. Create a JSON file called config.json with the following content

```
{
  "compartmentid": "<compartment-id>",
  "source": {
    "resource": "<bucket-name>",
    "service": "ObjectStorage",
    "source-type": "OCISERVICE",
    "category": "write"
  }
}
```

The compartment-id is the Compartment OCID of where the bucket is exists. The resource value is the bucket name.

3. Create the Service Log

```
oci logging log create --log-group-id <log-group-id> \
--display-name "<DisplayName>" \
--log-type SERVICE --is-enabled TRUE \
--configuration file://config.json
```

The output of the command gives you a work request id. You can query the work request to see that status of the job by issuing the following command:

```
oci logging work-request get --work-request-id <work-request-id>
```

Look for the status filed to be **SUCCEEDED**.

4.18 Ensure a notification is configured for Local OCI User Authentication

It is recommended that an Event Rule and Notification be set up when a user in the via OCI local authentication. Event Rules are compartment-scoped and will detect events in child compartments. This Event rule is required to be created at the root compartment level.

Rationale:

Users should rarely use OCI local authenticated and be authenticated via organizational standard Identity providers, not local credentials. Access in this matter would represent a break glass activity and should be monitored to see if changes made impact the security posture.

Impact:

There is no performance impact when enabling the above-described features but depending on the amount of notifications sent per month there may be a cost associated.

Result:

Tenancy MutantRogue: **Compliant**

Tenancy NewCoCobranca: **Compliant**

5 - Storage

5.1.1 Ensure no Object Storage buckets are publicly visible

A bucket is a logical container for storing objects. It is associated with a single compartment that has policies that determine what action a user can perform on a bucket and on all the objects in the bucket. By Default a newly created bucket is private. It is recommended that no bucket be publicly accessible.

Rationale:

Removing unfettered reading of objects in a bucket reduces an organization's exposure to data loss.

Impact:

For updating an existing bucket, care should be taken to ensure objects in the bucket can be accessed through either IAM policies or pre-authenticated requests.

Result:

Tenancy MutantRogue: **Non-Compliant**

Tenancy NewCoCobranca: **Compliant**

Remediation

From Console

1. Follow the audit procedure above.

2. For each **bucket** in the returned results, click the Bucket Display Name.
3. Click **Edit Visibility**.
4. Select **Private**.
5. Click **Save Changes**.

From CLI

1. Follow the audit procedure.
2. For each of the buckets identified, execute the following command:

```
oci os bucket update --bucket-name <bucket-name> --public-access-type NoPublicAccess
```

5.1.2 Ensure Object Storage Buckets are encrypted with a Customer Managed Key (CMK)

Oracle Object Storage buckets support encryption with a Customer Managed Key (CMK). By default, Object Storage buckets are encrypted with an Oracle managed key.

Rationale:

Encryption of Object Storage buckets with a Customer Managed Key (CMK) provides an additional level of security on your data by allowing you to manage your own encryption key lifecycle management for the bucket.

Impact:

Encrypting with a Customer Managed Keys requires a Vault and a Customer Master Key. In addition, you must authorize Object Storage service to use keys on your behalf. Required Policy:

```
Allow service objectstorage-<region_name>, to use keys in compartment <compartment-id> where target.key.id = '<key_OCID>'
```

Result:

Tenancy MutantRogue: **Non-Compliant (166 of 166 items)**

Tenancy NewCoCobranca: **Non-Compliant (158 of 158 items)**

Remediation:

From OCI Console:

1. Go to <https://cloud.oracle.com/object-storage/buckets>
2. Click on an individual bucket under the Name heading.
3. Click **Assign** next to **Encryption Key: Oracle managed key**.
4. Select a **Vault**

5. Select a **Master Encryption Key**
6. Click **Assign**.

From Command Line:

1. Execute the following command:

```
oci os bucket update --bucket-name <bucket_name> --kms-key-id <master-encryption-key-id>
```

Default Value:

Oracle Managed Key for Encryption

5.1.3 Ensure Versioning is Enabled for Object Storage Buckets

A bucket is a logical container for storing objects. Object versioning is enabled at the bucket level and is disabled by default upon creation. Versioning directs Object Storage to automatically create an object version each time a new object is uploaded, an existing object is overwritten, or when an object is deleted. You can enable object versioning at bucket creation time or later.

Rationale:

Versioning object storage buckets provides for additional integrity of your data. Management of data integrity is critical to protecting and accessing protected data.

Some customers want to identify object storage buckets without versioning in order to apply their own data lifecycle protection and management policy.

Result:

Tenancy MutantRogue: **Non-Compliant (165 of 166 items)**

Tenancy NewCoCobranca: **Non-Compliant (158 of 158 items)**

Remediation:

From OCI Console:

1. For each bucket in the returned results, click the Bucket Display Name
2. Click **Edit** next to **Object Versioning: Disabled**
3. Click **Enable Versioning**.

From Command Line:

1. Follow the audit procedure
2. For each of the buckets identified get the bucket name:

```
oci os bucket update --bucket-name <bucket_name> --versioning Enabled
```

5.2.1 Ensure Block Volumes are encrypted with Customer Managed Keys (CMK)

Oracle Cloud Infrastructure Block Volume service lets you dynamically provision and manage block storage volumes. By default, the Oracle service manages the keys that encrypt block volumes. Block Volumes can also be encrypted using a customer managed key.

Terminated Block Volumes cannot be recovered and any data on a terminated volume is permanently lost. However, Block Volumes can exist in a terminated state within the OCI Portal and CLI for some time after deleting. As such, any Block Volumes in this state should not be considered when assessing this policy..

Rationale:

Encryption of block volumes provides an additional level of security for your data. Management of encryption keys is critical to protecting and accessing protected data. Customers should identify block volumes encrypted with Oracle service managed keys in order to determine if they want to manage the keys for certain volumes and then apply their own key lifecycle management to the selected block volumes.

Impact:

Encrypting with a Customer Managed Key requires a Vault and a Customer Master Key. In addition, you must authorize the Block Volume service to use the keys you create. Required IAM Policy:

```
Allow service blockstorage to use keys in compartment <compartment-id> where
target.key.id = '<key_OCID>'
```

Result:

Tenancy MutantRogue: **Non-Compliant (286 of 286 items)**

Tenancy NewCoCobranca: **Non-Compliant (334 of 334 items)**

Remediation:

From OCI Console:

1. For each block volume identified, click the link under Display name.
2. If the value for **Encryption Key** is **Oracle-managed key**, click **Assign** next to **Oracle-managed key**.
3. Select a **Vault Compartment** and **Vault**.
4. Select a **Master Encryption Key Compartment** and **Master Encryption key**.
5. Click **Assign**.

From Command Line:

1. For each **volume** identified get its OCID. Execute the following command:

```
oci bv volume-kms-key update -volume-id <volume_OCID> --kms-key-id <kms_key_OCID>
```

5.2.2 Ensure Boot Volumes are encrypted with Customer Managed Keys (CMK)

When you launch a virtual machine (VM) or bare metal instance based on a platform image or custom image, a new boot volume for the instance is created in the same compartment. That boot volume is associated with that instance until you terminate the instance. By default, the Oracle service manages the keys that encrypt this boot volume. Boot Volumes can also be encrypted using a customer managed key.

Rationale:

Encryption of boot volumes provides an additional level of security for your data. Management of encryption keys is critical to protecting and accessing protected data. Customers should identify boot volumes encrypted with Oracle service managed keys in order to determine if they want to manage the keys for certain boot volumes and then apply their own key lifecycle management to the selected boot volumes.

Impact:

Encrypting with a Customer Managed Keys requires a Vault and a Customer Master Key. In addition, you must authorize the Boot Volume service to use the keys you create. Required IAM Policy:

Result:

Tenancy MutantRogue: **Non-Compliant (384 of 384 items)**

Tenancy NewCoCobranca: **Non-Compliant (363 of 363 items)**

Remediation:

From OCI Console:

1. For each Boot Volume in the returned results, click the Boot Volume name
2. Click **Assign** next to **Encryption Key**
3. Select the **Vault Compartment** and **Vault**
4. Select the **Master Encryption Key Compartment** and **Master Encryption key**
5. Click **Assign**.

From Command Line:

1. For each boot volume identified get its OCID. Execute the following command:

```
oci bv boot-volume-kms-key update --boot-volume-id <Boot_Volume_OCID> --kmskey-id <KMS_Key_OCID>
```

5.3.1 Ensure File Storage Systems are encrypted with Customer Managed Keys (CMK)

Oracle Cloud Infrastructure File Storage service (FSS) provides a durable, scalable, secure, enterprise-grade network file system. By default, the Oracle service manages the keys that encrypt FSS file systems. FSS file systems can also be encrypted using a customer managed key.

Rationale:

Encryption of FSS systems provides an additional level of security for your data. Management of encryption keys is critical to protecting and accessing protected data. Customers should identify FSS file systems that are encrypted with Oracle service managed keys in order to determine if they want to manage the keys for certain FSS file systems and then apply their own key lifecycle management to the selected FSS file systems.

Impact:

Encrypting with a Customer Managed Keys requires a Vault and a Customer Master Key. In addition, you must authorize the File Storage service to use the keys you create. Required IAM Policy:

```
Allow service Fss0c1Prod to use keys in compartment <compartment-id> where target.key.id = '<key_OCID>'
```

Result:

Tenancy MutantRogue: **Non-Compliant (22 of 22 items)**

Tenancy NewCoCobranca: **Non-Compliant (13 of 13 items)**

Remediation:

From OCI Console:

1. For each File Storage System in the returned results, click the File System Storage
2. Click **Edit** next to **Encryption Key**
3. Select **Encrypt using customer-managed keys**
4. Select the **Vault Compartment** and **Vault**
5. Select the **Master Encryption Key Compartment** and **Master Encryption key**
6. Click **Save Changes**.

From Command Line:

1. For each File Storage System identified get its OCID. Execute the following command:

```
oci bv volume-kms-key update -volume-id <volume_OCID> --kms-key-id <kms_key_OCID>
```

6 Asset Management

6.1 Create at least one compartment in your tenancy to store cloud resources

When you sign up for Oracle Cloud Infrastructure, Oracle creates your tenancy, which is the root compartment that holds all your cloud resources. You then create additional compartments within the tenancy (root compartment) and corresponding policies to control access to the resources in each compartment.

Compartments allow you to organize and control access to your cloud resources. A compartment is a collection of related resources (such as instances, databases, virtual cloud networks, block volumes) that can be accessed only by certain groups that have been given permission by an administrator.

Rationale:

Compartments are a logical group that adds an extra layer of isolation, organization and authorization making it harder for unauthorized users to gain access to OCI resources.

Impact:

Once the compartment is created an OCI IAM policy must be created to allow a group to resources in the compartment otherwise only group with tenancy access will have access.

Result:

Tenancy MutantRogue: **Compliant**

Tenancy NewCoCobranca: **Compliant**

6.2 Ensure no resources are created in the root compartment

When you create a cloud resource such as an instance, block volume, or cloud network, you must specify to which compartment you want the resource to belong. Placing resources in the root compartment makes it difficult to organize and isolate those resources.

Rationale:

Placing resources into a compartment will allow you to organize and have more granular access controls to your cloud resources.

Impact:

Placing a resource in a compartment will impact how you write policies to manage access and organize that resource.

Result:

Tenancy MutantRogue: **Non-Compliant (4 of 19985 items)**

Tenancy NewCoCobranca: **Non-Compliant (3 of 15843 items)**

Remediation:

From OCI Console:

1. For each item in the returned results, click the item name.
2. Then select **Move Resource** or **More Actions** then **Move Resource**.
3. Select a compartment that is not the root compartment in **CHOOSE NEW COMPARTMENT**.
4. Click **Move Resource**.

From CLI:

1. Follow the audit procedure above.
2. For each bucket item execute the command below:

```
oci os bucket update --bucket-name <bucket-name> --compartment-id <not root compartment-id>
```

3. For other resources use the change-compartment command for the resource type:

```
oci <service-command> <resource-command> change-compartment --<item-id> <item-id> --compartment-id <not root compartment-id>  
i. Example for an Autonomous Database:  
oci db autonomous-database change-compartment --autonomous-database-id <autonomous-database-id> --compartment-id <not root compartment-id>
```

OCI SERVICES

In addition to verifying the CIS recommendations, we also checked the status of OCI's native security services.

The following table shows what we found in Tenancy MutantRogue:

STATUS	SERVICE	Description
FAILED	Vulnerability Scanning Service	Scanning recipes were found in the Tenant, but it's only checking a few instances
FAILED	Data Safe	No target databases are registered in Datasafe
FAILED	Cloud Guard	Cloud Guard service was enabled in Tenant but only checking 3 compartments
FAILED	Bastion	A Bastion instance was found in Tenant but it seems to it was used only as a test.
FAILED	Security Zones	Security Zones were created in Tenant
OK	Vault	Vault instances were found in Tenant

And the following table shows what we found in Tenancy NewCoCobranca:

STATUS	SERVICE	Description
FAILED	Vulnerability Scanning Service	No scanning recipes were found in the Tenand
FAILED	Data Safe	No target databases are registered in Datasafe
OK	Cloud Guard	Cloud Guard service was enabled in Tenant
OK	Bastion	Bastions instances were found in Tenant
FAILED	Security Zones	No Security Zones were created in Tenant
FAILED	Vault	No Vault instances were found in Tenant

Recommendations:

We recommend enabling the services below as part of a strong security posture.

Vulnerability Scanning Service

Oracle Cloud Infrastructure Vulnerability Scanning Service (OCI VSS) is simple, prescriptive, and tightly integrated with the OCI platform. VSS is available to all OCI customers that have paid accounts at no additional cost. The scanning platform includes default plugins and engines for instance and container scanning. The service scans installed packages and artifacts looking for the existence of known vulnerabilities. The service scans for open ports on an instance and it reports on publicly and privately available open ports. These findings will highlight what ports an attacker might use or what application could be shut down by the customer to reduce their attack surface. The scanning agent also reviews the configuration of each instance against specific OS CIS benchmarks enabling customers to see immediately what Operating System (OS) security hardening opportunities exist on their instances.

VSS manages the deployment, configuration, and upgrade of these engines and agents across the customer's OCI tenancy fleet. VSS reports all the findings as problems through Oracle Cloud Guard, with rules and machine learning to prioritize vulnerabilities. Cloud Guard alerting can help customers reduce the time from detection to remediation.

Recommendation:

Create a recipe for scanning instances/containers and targets to use those recipes. Collect all results in Cloud Guard.

[Learn more about Vulnerability Scanning Service.](#)

Data Safe

Oracle Data Safe is an integrated and comprehensive cloud service that ensures data security for cloud databases and helps secure your databases via security and user risk assessments, user activity auditing, sensitive data discovery, and data masking. With this well integrated and easy-to-use solution, cloud database customers of all sizes and in all verticals can meet their database security requirements very easily. As data and applications move to the cloud, the responsibility for securing an organization's assets becomes progressively more complex. While cloud service providers are responsible for securing their global infrastructure and protecting client databases from access by their own personnel, each cloud customer must implement its own measures to secure its users and data.

Recommendation:

Register and start securing your databases via security and user risk assessments, user activity auditing, sensitive data discovery, and data masking.

[Learn more about Data Safe.](#)

Cloud Guard

Misconfigured resources and insecure activity in the cloud represent one of the most difficult problems for security professionals. Misconfigured cloud tenancies around cloud resources can present themselves in many ways; from publicly accessible object storage buckets, unencrypted data storage, sensitive ports open to the internet. The behavior of users and administrators within the cloud are also a concern. Insecure activity in a cloud infrastructure offering is difficult to detect as it oftentimes spans beyond simple detection rules and can be generated from authenticated users. Insecure activity can oftentimes be attributed to different parts of the cyber kill chain such as: intrusion, reconnaissance, exploitation, privilege escalation, exfiltration, etc.

Oracle Cloud Infrastructure Cloud Guard is a cloud security service that detects misconfigured resources and insecure activities. Cloud Guard acts as a log and events aggregator that directly integrates with all major Oracle Cloud Infrastructure services (Compute, Networking, Storage, etc.) providing actionable results. Cloud Guard offers the flexibility to take action on security issues manually or automatically with conditional operators.

Recommendation:

Based on its monitoring (aligned with CIS baselines), Cloud Guard provides in its console recommendations, some already highlighted in this document, they are:

[Learn more about Cloud Guard.](#)

OCI Bastion

Provide restricted and time-limited secure access to resources that don't have public endpoints and require strict resource access controls. Examples include compute instances, bare metal and virtual machines, MySQL, ATP, OKE, and any other resource that allows Secure Shell Protocol (SSH) access. With Oracle Cloud Infrastructure (OCI) Bastion service, customers can enable access to private hosts without deploying and maintaining a jump host. In addition, customers gain improved security posture with identity-based permissions and a centralized, audited, and time-bound SSH session. OCI Bastion removes the need for a public IP for bastion access, eliminating the hassle and potential attack surface from remote access.

Note: We recommend using Bastion to remediate findings from CIS Control 2.1 in public subnets, if the permission is related to remote access use cases without restricted sources, VPN or Fast Connect.

Bastion is a free service.

[Learn more about Bastion.](#)

Security Zones

Security Zones let you be confident that your resources in Oracle Cloud Infrastructure, including Compute, Networking, Object Storage, Block Volume and Database resources, comply with your security policies.

A security zone is associated with one or more compartments and a security zone recipe. When you create and update resources in a security zone, Oracle Cloud Infrastructure validates these operations against the list of policies that are defined in the security zone recipe. If any security zone policy is violated, then the operation is denied. By default, a compartment and any subcompartments are in the same security zone, but you can also create a different security zone for a subcompartment.

[Learn more about Security Zones.](#)

Vault

Vaults are logical entities where Vault service creates and durably stores vault keys and secrets. The type of vault you have determines features and functionality such as degrees of storage isolation, access to management and encryption, scalability, and the ability to back up. The type of vault you have also affects pricing. You cannot change a vault's type after you create the vault.

The Vault service offers different vault types to accommodate your organization's needs and budget. All vault types ensure the security and integrity of the encryption keys and secrets that vaults store. A virtual private vault is an isolated partition on a hardware security module (HSM). Vaults otherwise share partitions on the HSM with other vaults.

Virtual private vaults include 1000 key versions by default. If you don't require the greater degree of isolation or the ability to back up the vault, you don't need a virtual private vault. Without a virtual private vault, you can manage costs by paying for key versions individually, as you need them. (Key versions count toward your key limit and costs. A vault key always contains at least one active key version. Similarly, a secret always has at least one secret version. However, limits on secrets apply to the tenancy, rather than a vault.)

[Learn more about Vault.](#)

Connect with us

Call **+1.800.ORACLE1** or visit **oracle.com**. Outside North America, find your local office at: **oracle.com/contact**.

 blogs.oracle.com

 facebook.com/oracle

 twitter.com/oracle

Copyright © 2026, Oracle and/or its affiliates. This document is provided for information purposes only, and the contents hereof are subject to change without notice. This document is not warranted to be error-free, nor subject to any other warranties or conditions, whether expressed orally or implied in law, including implied warranties and conditions of merchantability or fitness for a particular purpose. We specifically disclaim any liability with respect to this document, and no contractual obligations are formed either directly or indirectly by this document. This document may not be reproduced or transmitted in any form or by any means, electronic or mechanical, for any purpose, without our prior written permission.

Oracle, Java, MySQL, and NetSuite are registered trademarks of Oracle and/or its affiliates. Other names may be trademarks of their respective owners.