Oracle Cloud Infrastructure- Security Best Practices

Agenda

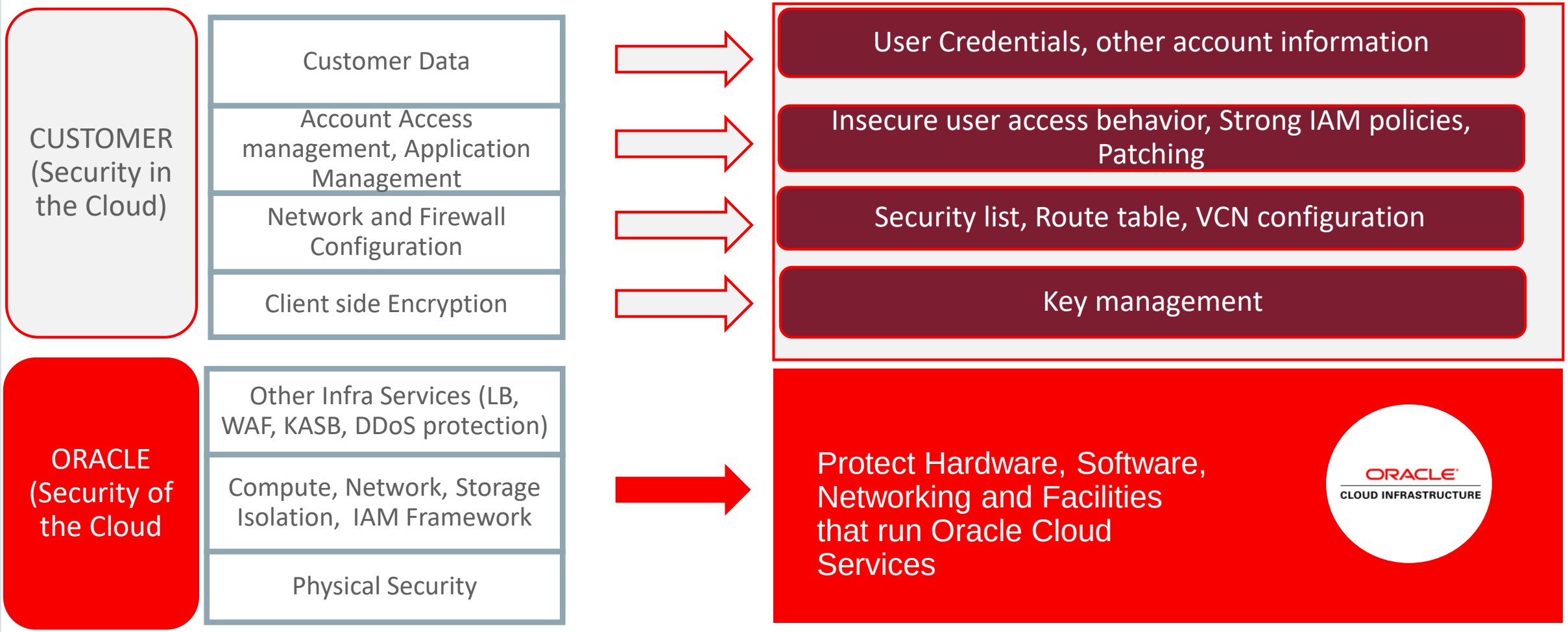
- Shared Security Responsibility Model
- IAM Best Practices
- Securing Compute
- Securing Database
- Securing Block Volume
- Securing File Storage
- Securing Object Storage
- Securing Data Transfer
- Securing Resource Manager
- Network Security Architecture: VCN, Load balancer, and DNS
- Security Testing Policy
- Announcements and Notifications

Shared Security Responsibility Model

Key Points

With Oracle Cloud Infrastructure, enterprise customers get unparalleled control of and transparency into their applications running in the cloud, including:

Shared Responsibility Model in Oracle Cloud Infrastructure



Oracle Cloud Infrastructure Security Capabilities At a Glance

1	**Customer Isolation**	Bare Metal Instance, VM Instance, VCN IAM, Compartments
2	**Data Encryption**	Default Encryption for Storage, Key Management, DB Encryption
3	**Security Controls**	User Authentication and Authorization, Instance Principals, Network Security Control, Web Access Firewall
4	**Visibility**	Audit Logs, CASB Based monitoring and enforcement
5	**Secure Hybrid Cloud**	Identity Federation Third Party Security Solution, IPSEC VPN, Fast Connect
6	**High Availability**	Fault-independent data center, Fault Domain, SLA
7	**Verifiably Secure Infrastructure**	Security Operations, Compliance Certification and Attestation, Customer penetration and Vulnerability testing

Understanding the Environment

When planning your Oracle Cloud Infrastructure deployment, consider the following:

Which resources must be protected?

- Protect customer data
- Protect internal data
- Protect system components from

Who are you protecting data from?

- One subscribers' data from other subscribers
- How much access to give a system administrator

What will happen if protections on a strategic resource fail?

- Inconvenience or great damage to you or your customers.
- Security ramifications of each resource

IAM Best Practices

IAM Best Practices

Users /Credentials	• Restrict users in a group only to compartments they need access to • Do not use the tenancy administrator account for day-to-day operations. • Specify secure and auditable "break-glass" procedures for using administrator accounts for emergencies • Disable tenancy administration access immediately when an employee leaves the organization. • Create security policies which prevent administrator account lock-out (in case tenancy administrator leaves the company). • Do not share IAM user accounts across multiple users • Deactivate rather then delete user when appropriate • Do not hard code sensitive IAM credentials in software or documents accessible to a wide audience
Tags	• Use Defined tags both to manage and Audit resources
Groups	• Membership of tenancy administrator group only on a as-needed basis • Create groups and IAM policy carefully considering the role of each user in the group
Passwords	• Tenancy administrators should use high-complexity passwords, along with MFA, and periodically rotate their passwords. • Create a strong console password for each IAM user • Rotate IAM passwords and API keys regularly, every 90 days or less

IAM Best Practices

Security Policy and Federation	• Start with least privilege access to IAM groups for accessing resources
	  ➢ Inspect
	  ➢ Read,
	  ➢ Use
	  ➢ manage
	• Use Conditions in IAM policies for finer granularity of resource access (when needed)
	• Use instance principals only after carefully evaluating security hardness of the compute instance
	• Use Federated login even for non-Administrator users when possible
	  ➢ Especially important for enterprises using custom policies for user authentication (e.g MFA)
	• Use federating login (IDCS) and Administrator user
	  ➢ Create a federation administrators group that maps to the federated IdP administrator group.
	  ➢ Same IAM policy can be used to govern both groups
	• Utilize MFA

Securing Compute

Securing Compute

Permissions and Access	• DELETE permissions only to tenancy and compartment admins.
	• Prefer Instance Principals over credentials rotation
	  ➢ Easier control via dynamic group configuration
	  ➢ Private key(for API calls) is rotated regularly
	• Restrict access to instance metadata(OCID, Display name, Dynamic group data)
	  ➢ Use iptables or other methods to restrict access to root user only
	• Harden ssh access
	  ➢ Use public-key logins only
	  ➢ Disable root logins
	  ➢ Disable password logins
	  ➢ Change SSH port to non-standard port

Securing Compute

Security List and Firewall	• Use VCN security lists to allow instance access from authorized IP addresses. • Use applications like Fail2ban ➢ Blacklists IP addresses involved in brute-force login attempts • Use additional host-based firewalls (such as iptables and firewalld) rules; ➢ Restrict network access to instances(ports, protocols, and packet types) ➢ Rules can be configured, saved, and initialized on every instance boot.
Harden Installed OS	• Install and update only the software packages needed for operation • Minimize number of active services (print, sendmail, sshd etc). ➢ If possible run services on separate instance • Lock down network services (limit number of connection for each service by specifying limits in the configuration file /etc/xinetd.conf) • Check what services are running on the system using port scanning utilities

Securing Compute

Harden Installed OS	• Configure TCP wrappers and set up firewalls with Netfilter and Iptables • Use built-in kernel subsystem called Netfilter for packet filtering firewall (Three functions, Packet filtering, NAT, IP masquerading) • Disable ssh if not needed. If it is required then tighten its configuration by editing parameters in /etc/ssh/sshd_config
Mounts, File Permissions, and Ownerships	• Use separate disk partitions for operating system and user data • Mount the /usr file system as read-only (when update is needed simply remount /usr as read/write) • Limit user access on non-root local file systems (set the noexec, nosuid, and nodev mount options) • Use POSIX Access Control Lists (ACLs) for richer access control

Securing Compute

Managing Users and Authentication	• Check the system for unused and unlocked user accounts • Set passwords on any accounts that aren't protected • No non-root user accounts have the user ID of 0
Additional Kernel Security Mechanisms	• Address Space Layout Randomization (ASLR). • Data Execution Prevention (DEP). • Position Independent Executables (PIE).
Data Encryption	• Downloaded software packages are signed. • Install RPMs using SSL protocol • Use full-disk encryption such as dm-crypt Use eCryptfs utilities (performs encryption at the file system–level, can be applied to protect individual files or directories)
Random Number generator	• Use built in hardware random number generator support for security applications requiring random numbers.

Securing Database

Securing Database

Database Access Control	• Strong password for users authentication to the database • Use Oracle provided PL/SQL script to verify database password complexity ($ORACLE_HOME/rdbms/admin/UTLPWDMG.SQL) • Use VCN security lists and Private subnet (when appropriate) • Use service gateway or NAT gateway for OS patching and backup
Data Durability	• Restrict Database delete permissions using IAM policy • Use RMAN to do additional periodic backups of databases ➢ Store encrypted backup copies in local storage (block volumes, for example) or Object Storage.

Securing Database

Database Access Control

- Strong password for users authentication to the database
- Use Oracle provided PL/SQL script to verify database password complexity ($ORACLE_HOME/rdbms/admin/UTLPWDMG.SQL)
- Use VCN security lists and Private subnet (when appropriate)to enforce network access control to database instances
- Use service gateway or NAT gateway to perform OS patching and backup

Data Durability

- Database delete permissions to a minimum number of IAM users and groups.
- Use RMAN to do additional periodic backups of databases
  - Store encrypted backup copies in local storage (block volumes, for example) or Object Storage.

Securing Database

Database Encryption and Key Management

- All databases created in Oracle Cloud Infrastructure are encrypted using (TDE)
  - Unencrypted database from on-premise to Oracle Cloud Infrastructure using RMAN, will not be encrypted automatically. Encrypt the database after migration
- User-created tablespaces are encrypted by default in Oracle Cloud Infrastructure Database
- When using Oracle wallet to store TDE keys for new database instance, use a strong password (if not using ‘auto open’)
- Periodically rotate the TDE master key (90 days or less)
- Use Oracle Key Vault (OKV) for managing Oracle TDE master keys. OKV(key management appliance) can store, rotate, and audit accesses to TDE master keys

Securing Database

Database Patching

- Keep patches up-to-date
  - Patchsets and Patch Set Updates (PSUs) are released on a quarterly basis
  - Contain security fixes and additional high-impact/low-risk critical bug fixes

Database Security Configuration Checking

- Use Oracle Database Security Assessment Tool (DBSAT) periodically
  - Provides automated security configuration checks
  - Performs security checks for user privilege analysis, database authorization controls, auditing polices, database listener configuration, OS file permissions, and sensitive data stored

Database Security Auditing

- Install Oracle Audit Vault and Database Firewall (AVDF) to monitor database audit logs and create alerts

Securing Database

Database Backup

- Use Managed backups (Console or the API)
  - Oracle manages object store user and rotates credentials (every 3 days).
  - Oracle Encrypts all managed backups in the object store.
  - uses TDE Encryption encrypting the backups.
- If not using managed backups, change object store passwords at regular intervals.

Securing Block Volume

Securing Block Volume

- Block Volume and boot volume are grouped under ‘volume-family’;
  - Assign least privilege access for IAM users and groups ‘volume-family’
- To minimize loss of data due to inadvertent;
  - Give VOLUME_DELETE, VOLUME_ATTACHMENT_DELETE and VOLUME_BACKUP_DELETE permissions to a minimum possible set of IAM users and groups
  - DELETE permissions should be given only to tenancy and compartment administrators.
- Create periodic backups of volumes
- Use dm-crypt, veracrypt, and Bit-Locker for additional encryption(volumes and their backups are encrypted at rest using AES-256)

Securing File Storage

Securing File Storage

- All file-system data is encrypted at rest using AES-128 by default
- NFSv3 mount target is identified by a DNS name and is mapped to an IP address;
  - Use VCN security lists (of the mount target subnet) to configure network access to the mount target from only authorized IP addresses.
- Authorize users to mount file systems using IAM security policies (console only)
- For data durability, take periodic snapshots of the file system
  - To minimize accidental deletion of data, constrain the set of users having privileges to delete mount targets, file-systems, and snapshots.
- Access to mounted NFS file systems from a remote host is determined by POSIX user and group permissions
  - Use ‘all_squash’ option (Convert incoming requests, from ALL users including root), to map all users to nfsnobody
- Use NFS ACLs to enforce access control to the mounted file system.

Securing Object Storage

Securing Object Storage

Public Buckets Security Controls

- Assign least privilege access for IAM users and groups to resource types in object-family
  - ➢ Use Pre-Authenticated request when possible for users with IAM credentials (Rather then making bucket public)
- BUCKET_UPDATE permission should be restricted to a minimal set of IAM groups.(To minimize possibility of existing buckets being made public inadvertently or maliciously)

Pre-Authenticated Request (PAR)

- Restrict PAR_MANAGE (ability to create URLs which grant time-bound read or write access to objects) IAM permission to appropriate users
- Note down the PAR URL created. By design, it is not possible to retrieve a forgotten PAR URL. If you forget a PAR URL, you must create a new PAR.

Securing Data Transfer

Securing Data Transfer

Oracle offers offline data transfer solutions that let you migrate large amounts of data to buckets in a tenancy in Oracle Cloud Infrastructure. Data transfer solutions include:

Data Transfer Disk

- Data transfer administrator uses the Data Transfer Utility to create disks
- Data Transfer Utility uses dm-crypt and LUKS utilities(standard Linux) to encrypt block devices (AES-256 bit)
  - OCI also generates a passphrase to protect encryption keys, Copy this passphrase to a durable, secure location for future reference
  - All network communication between the Data Transfer Utility and Oracle Cloud Infrastructure is encrypted in-transit using Transport Layer Security (TLS)
  - For additional security encrypt your data with your own encryption keys before copying to disks
  - Generate a manifest file using the Data Transfer Utility. The manifest contains an index of all of the copied files and generated data integrity hashes and temporary IAM data transfer upload user

Securing Data Transfer

- Create a unique IAM data transfer upload user(used in manifest file) for each transfer job and then delete that user after your data is uploaded to Oracle Cloud Infrastructure

- After processing a transfer package, Oracle returns all transfer disks attached to the transfer package
- Data on the disk is made unrecoverable before shipping the transfer disks back
- In addition to Data transfer utility the OCI Console can also be used

Securing Data Transfer

Data Transfer Appliance

Data Transfer Appliance is offline data transfer solutions that lets you migrate petabyte-scale datasets to Oracle Cloud Infrastructure. Send your data as files on one or more secure, high-capacity, Oracle-supplied storage appliances to an Oracle transfer site which is then uploaded to Object Storage Bucket

- Ensure Appliances tamper-evident security tie on the transit case is intact
- Match the number on the physical security ties to the numbers logged by Oracle in the transfer appliance details

All other Best practice recommendation listed for ‘Data Transfer Disk’ section apply to Data Transfer Appliance as well

Securing Resource Manager

Securing Resource Manager

- Allows to automate installing and provisioning of resources by committing the provisioning instructions to configuration files ("infrastructure-as-code" model)
- Provisioning instructions are executed as "jobs“
- Resources that are provisioned when you run the jobs are organized into "stacks.“
- Executing jobs and provisioning stacks is gated using role-based access control (RBAC)
  - Create specific group/groups authorized to perform operations on stacks and jobs
  - Use IAM policy to allow access to ‘Jobs’ or ‘Stacks’ resource (see example below)
  - Use IAM policy to restrict type of operation (read, write etc) on resources (Jobs, Stacks)

Allow group <group_name> to read orm-stacks in compartment (Get a stack Terraform configuration)

Allow group <group_name> to inspect orm-jobs in compartment (List jobs)

- Create IAM policy that prevents certain actions.

Securing Resource Manager

- Create IAM policy that prevents certain actions.

Allow group <group_name> to manage orm-jobs in compartment where any {target.job.operation = 'PLAN', target.job.operation = 'APPLY'} (prevents from running Destroy jobs on a stack.

- Terraform state (.tfstate) can contain sensitive data, including resource IDs and in some cases sensitive user data like passwords
  - Create a security policy that limits access to reading jobs

Allow group <group_name> to read orm-jobs in compartment

- Resource Manager workflow includes uploading Terraform configuration to the service
  - Do not include sensitive information in your terraform configuration files (.zip file, ORM API can be used to access this data)

Network Security Architecture: VCN, Load balancer, and DNS

Network Security Architecture: VCN, Load balancer, and DNS

Network Segmentation: VCN Subnets

- Periodically monitor Audit logs to review changes to VCN security lists, route table rules, and VCN gateways
- Formulate a tiered subnet strategy for the VCN, to control network access
  - **DMZ subnet** for load balancers
  - **Public subnet** for externally accessible hosts such as intrusion detection (IDS) instances, and web application servers
  - **Private subnet** for internal hosts such as databases
- Use security list rules to control the type of connectivity to hosts in a public subnet
  - In addition configure host-based firewalls such as iptables, firewalld for network access control
- Add service gateway to your VCN to enable DB systems in the private subnet to directly back up to Object Storage without the traffic traversing the internet
  - You must set up the subnet's routing and security lists to enable traffic

Network Security Architecture: VCN, Load balancer, and DNS

Network Segmentation: VCN Security Lists

- Use VCN's security lists to restrict network access to instances in a subnet.
  - Use stateless rules for high-performance applications (if network traffic matches both stateful and stateless security lists, the stateless rule takes precedence)
  - Allow SSH or RDP access only from authorized CIDR blocks rather than(0.0.0.0/0) OR only enable ssh/rdp temporarily using VCN API UpdateSecurityList
  - Configure VCN security lists to allow ICMP pings (or performing instance health checks)
  - Use bastion hosts as a way to control external access (for example, SSH) to VCN hosts in private subnet
  - Use IAM policies to allow only network administrators to make security list changes

Network Security Architecture: VCN, Load balancer, and DNS

Secure Connectivity: VCN Gateways and FastConnect Peering

- Use IAM policy to allow only network administrators to create or modify VCN gateways.
- Restrict internet access to only needed instances
- When available, use two tunnels for IPSEC VPN to achieve high availability and enhance security
- When using Fast Connect use both private peering and public peering (when applicable)

Network Security Architecture: VCN, Load balancer, and DNS

Virtual Security Appliances in a VCN

- When not using NAT gateway
  - Create a security host and route all subnet traffic to it (using route table rules that use a local VCN private IP address as a target)
  - For high availability, assign security host a secondary private IP address (which you can move to a VNIC on a standby host in case of primary host failure)
  - Full network packet capture or network flow logs can be captured on the security host instances using tcpdump, and the logs can be uploaded periodically to an Object Storage bucket
- When using bring-your-own-hypervisor (BYOH) model
  - Run Virtual security appliances on a VM, use secondary VNIC of the VM for direct connectivity to other instances and services in the VCN

Network Security Architecture: VCN, Load balancer, and DNS

Load Balancers

- Enables end-to-end TLS connections between a client's applications and a customer's VCN
  - Use HTTP load balancer when terminating TLS on Load Balancer
  - Use TCP load balancer when terminating TLS on a back-end server
  - Upload your own TLS certificates when needed
- Use Public or Private load balancer as needed to restrict access
- For public load balancers, use a regional public subnet
  - Provides highly available configuration across two different availability domains
  - Configure load balancer firewall rules by setting up the VCN security lists for that subnet
  - Configure VCN security lists for the backend servers to limit traffic only from the public load balancers

Network Security Architecture: VCN, Load balancer, and DNS

DNS Zones and Records

- Incorrect updates or unauthorized deletions of DNS zones and records could result in outage of services
  - ➢ Limit IAM users who can modify DNS zones and records.

IAM Policy Examples

- Allow Users to Only View Security Lists

Allow group NetworkUsers to inspect security-lists in tenancy

- Prevent Users from Creating External Connection to the Internet

Allow group NetworkUsers to manage internet-gateways in tenancy
where request.permission!='INTERNET_GATEWAY_CREATE‘

- Prevent Users from Updating DNS Records and Zones

Allow group NetworkUsers to manage dns-records in tenancy where all {request.
permission!='DNS_RECORD_DELETE',request.permission!='DNS_RECORD_UPDATE'}
Allow group NetworkUsers to manage dns-zones in tenancy where all
{request.permission!='DNS_ZONE_DELETE', request.permission!='DNS_ZONE_UPDATE'}

Security Testing Policy

Security Testing Policy

Security Testing Policy describes when and how you may conduct certain types of security testing of Oracle Cloud Services, including vulnerability and penetration tests, as well as tests involving data scraping tools

Penetration and Vulnerability Testing

- Oracle regularly performs penetration and vulnerability testing and security assessments against the Oracle cloud infrastructure, platforms, and applications
- Oracle does not assess or test any components that customer manage through or introduce into – including introduction through your development in or creation in - the Oracle Cloud Services (the "**Customer Components**").

Permitted Cloud Penetration and Vulnerability Testing (IaaS)

- Using Customer's own monitoring and testing tools, Customer may conduct penetration and vulnerability tests of your acquired single-tenant Oracle Infrastructure as a Service (IaaS) offerings

Security Testing Policy

Permitted Cloud Penetration and Vulnerability Testing (IaaS)

- Customer may not assess any other aspects or components of these Oracle Cloud Services including the facilities, hardware, software, and networks owned or managed by Oracle or its agents and licensors.
- Customers must notify Oracle prior to conducting any such penetration and vulnerability tests

Permitted Cloud Penetration and Vulnerability Testing (PaaS)

- Using Customer's own monitoring and testing tools, you may conduct penetration and vulnerability tests of your acquired single-tenant PaaS offerings
- Customer may not assess any other aspects or components of these Oracle Cloud Services including the facilities, hardware, software, and networks owned or managed by Oracle or its agents and licensors.
- Customers must notify Oracle prior to conducting any such penetration and vulnerability tests

Security Testing Policy

Permitted Cloud Penetration and Vulnerability Testing (SaaS)

- Penetration and vulnerability testing is not permitted for Oracle Software as a Service (SaaS) offerings

Rules of Engagement

- Testing must not target any other subscription or any other Oracle Cloud customer resources, or any shared infrastructure components
- Must not conduct any tests that will exceed the bandwidth quota or any other subscribed resource for your subscription
- Strictly prohibited from utilizing any tools or services in a manner that perform Denial-of-Service (DoS) attacks or simulations of such, or any “load testing” against any Oracle Cloud asset including yours.
- Any port scanning must be performed in a non-aggressive mode
- Customer responsible for independently validating that the tools or services employed during penetration and vulnerability testing do not perform DoS attacks, or simulations of such, prior to assessment of your instances.

Security Testing Policy

Rules of Engagement

- Social Engineering of Oracle employees and physical penetration and vulnerability testing of Oracle facilities is prohibited
- Must not attempt to access another customer's environment or data, or to break out of any container (for example, virtual machine)
- Any potential security issue related to Oracle Cloud, must be reported to Oracle within 24 hours by conveying the relevant information to My Oracle Support
- Must create a service request within 24 hours and must not disclose this information publicly or to any third party
- In the event you inadvertently access another customer's data, you must immediately terminate all testing and report it to Oracle within one hour by conveying the relevant information to My Oracle Support
- You are responsible for any damages to Oracle Cloud or other Oracle Cloud customers that are caused by your testing activities by failing to abide by these rules of engagement

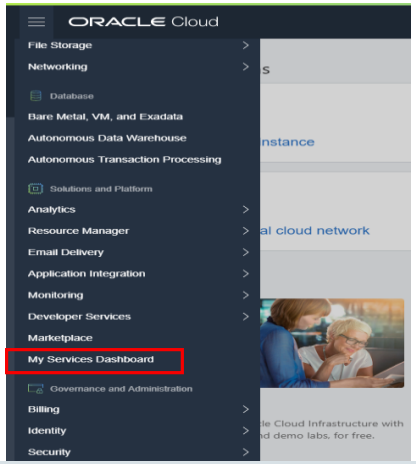
Security Testing Policy

Data Scraping Tools

- Any use of data scraping tools or technologies with Oracle Cloud Services to collect data available through any Oracle user interface or via web service calls requires the express written permission of Oracle.
- Oracle reserves the right to require that your proposed data scraping tools are validated and tested by Oracle prior to use in production, and are subsequently re-validated and tested annually.

Submitting a Cloud Security Testing Notification

- Submit Cloud security Testing notification through ‘My services Dashboard’



Announcements and Notifications

Announcements and Notifications

Oracle Cloud Infrastructure displays announcements in the Console to communicate timely, important information about service status. These announcements are also accessible via API.

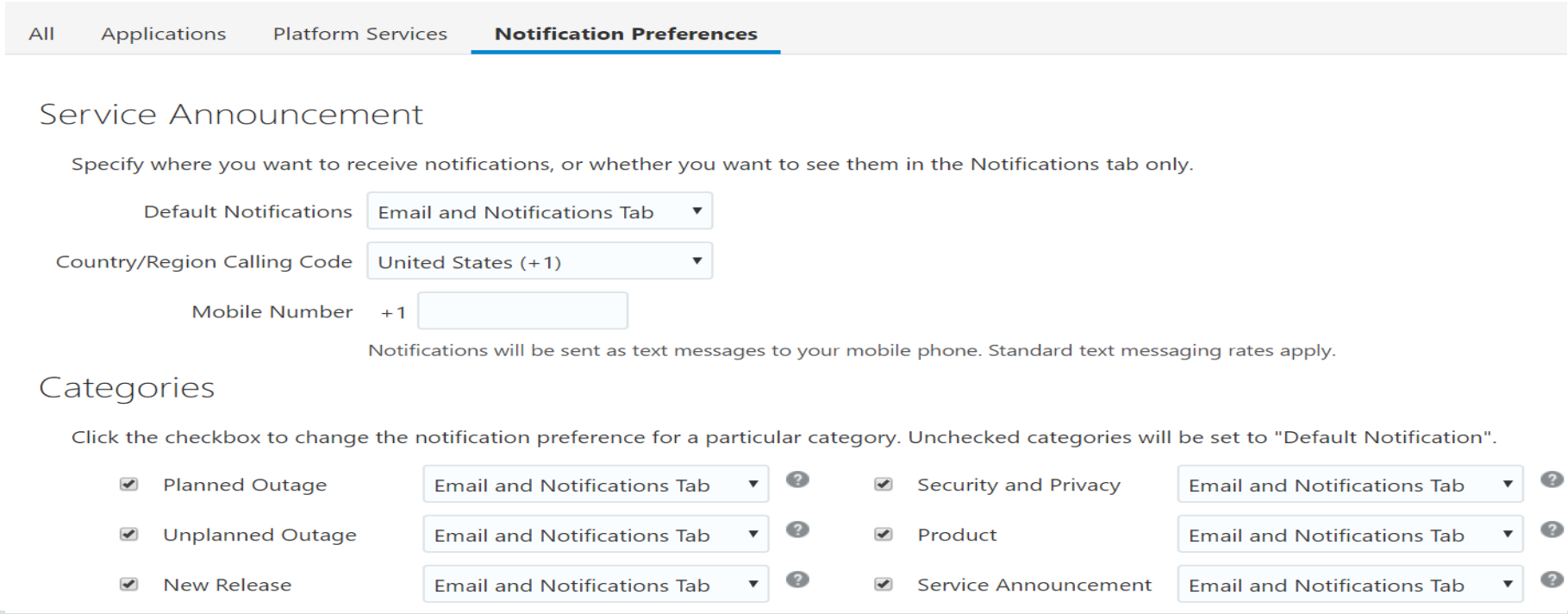
- There are different categories of announcements, such as;
  - **Required action.** You must take specific action within your environment.
  - **Emergency change.** There is a time period during which an unplanned, but urgent, change associated with your environment will take place.
  - **Recommended action.** You have specific action to take within your environment, but the action is not required.
  - **Planned change.** There is a time period during which a planned change associated with your environment will take place.
- For announcements that require action and affect Oracle Cloud Infrastructure Compute instances, you will get 30 days of advance notice.
- Oracle recommends to review the announcements and take appropriate action

Announcements and Notifications

Email and SMS notifications can be set via myservices dash board. These notification include amongst other;

- Security and Privacy
- Planned Outage
- Service Announcement

- Oracle recommends to configure these notifications and take appropriate action



Summary

All of the Oracle Cloud Infrastructure security capabilities have been designed with one goal in mind: allowing Customer to run their mission-critical workloads in the cloud with complete control and confidence.

Oracle continues to invest in all areas and to offer unmatched security and assurance to enterprise customers.