

COS80001 – Cloud Engineering  
Assignment 2: Deploying a Photo Album Web Application on Oracle Cloud Infrastructure (OCI)

Student Name: Arun Ragavendhar Arunachalam Palaniyappan  
Student ID: 104837257  
Tutorial Time: Friday – 04:30 PM to 06:30 PM  
Tenancy Name: [INSERT YOUR TENANCY NAME]

# I. INTRODUCTION

This report outlines the successful deployment of a dynamic Photo Album web application on Oracle Cloud Infrastructure (OCI). Building on concepts learned from previous AWS-based deployments, this assignment focuses on replicating the same architecture and functionality using OCI services. The implementation includes creating a secure Virtual Cloud Network (VCN), configuring subnets, deploying a MySQL database, launching compute instances, and enabling public access to photos stored in OCI Object Storage. Key tools and services used include Network Security Groups (NSGs), Security Lists, Reserved Public IP, phpMyAdmin, and Oracle Linux virtual machines.

# II. INFRASTRUCTURE SETUP

A comprehensive infrastructure was provisioned on OCI, including all components required to support the web application securely and efficiently.

A. VCN and Subnet Configuration  
A Virtual Cloud Network named "ARagavendharVCN" was created in the us-ashburn-1 region. The VCN included four subnets across two availability domains:

* Public Subnet 1 (10.0.1.0/24)
* Public Subnet 2 (10.0.2.0/24)
* Private Subnet 1 (10.0.3.0/24) for MySQL DB
* Private Subnet 2 (10.0.4.0/24) for Test Instance

Routing tables were configured such that only public subnets had routes to the Internet Gateway. Private subnets used isolated private route tables.

Figure 1: VCN and Subnet Configuration Diagram [PLACEHOLDER]  
Figure 2: Public and Private Route Table Configurations [PLACEHOLDER]

B. Network Security Groups  
Three NSGs were created as per the architecture:

1. Web-tierNSG: Allowed HTTP (80) and SSH (22) from anywhere, ICMP from Private Subnet 2, and TCP from Private Subnet 1.
2. Test-InstanceNSG: Allowed all ingress traffic, with ICMP egress to Public Subnet 2.

Default security lists were disabled to ensure NSG rules were enforced correctly.

Figure 3: NSG Rule Settings - Web-tierNSG [PLACEHOLDER]  
Figure 4: NSG Rule Settings - Test-InstanceNSG [PLACEHOLDER]

C. Virtual Machine Instances

1. Web Server / Bastion Host: Deployed in Public Subnet 2 using Oracle-Linux-8 (VM.Standard.E2.1). Apache, PHP, and phpMyAdmin were installed using a bash script. A Reserved Public IP was attached to ensure persistence.
2. Test Instance: Deployed in Private Subnet 2 with no public IP. SSH access was enabled via the bastion host.

Figure 5: Web Server Instance Configuration [PLACEHOLDER]  
Figure 6: Reserved Public IP Attachment [PLACEHOLDER]  
Figure 7: Private Test Instance [PLACEHOLDER]

D. MySQL Database Setup  
A MySQL 8.0.41 instance was deployed in Private Subnet 1 using the MySQL2 shape with 50GB storage. phpMyAdmin was used to create a database named photosDB, containing a photos table with the following schema:

* photo\_title (VARCHAR)
* description (VARCHAR)
* creation\_date (DATE)
* keywords (VARCHAR)
* object\_reference (VARCHAR)

Figure 8: MySQL Instance Details [PLACEHOLDER]  
Figure 9: phpMyAdmin - Table Schema [PLACEHOLDER]

E. Security Lists  
Custom security lists were configured:

* PublicSubnet2SL: Allowed SSH (22) from anywhere, ICMP from Private Subnet 2, HTTP (80) from all, and TCP to MySQL port 3306.
* PrivateSubnet1SL: Allowed MySQL traffic from Public Subnet 2 only.

Figure 10: PublicSubnet2SL Rules [PLACEHOLDER]  
Figure 11: PrivateSubnet1SL Rules [PLACEHOLDER]

# III. PHOTO ALBUM APPLICATION DEPLOYMENT

A. OCI Object Storage Configuration  
An object storage bucket named aragavendhar-photo-bucket was created. The bucket was made publicly accessible using a bucket-wide visibility setting. Several images were uploaded manually and tested using Pre-Authenticated Request (PAR) links.

Figure 12: Bucket Configuration [PLACEHOLDER]  
Figure 13: Uploaded Photos in Bucket [PLACEHOLDER]  
Figure 14: Public Access via Pre-Authenticated URL [PLACEHOLDER]

B. Metadata Entry  
The metadata for each photo was manually entered into the photos table via phpMyAdmin. Each entry contained the correct PAR link as the object reference.

Figure 15: phpMyAdmin - Metadata Records [PLACEHOLDER]

C. Web Application Setup  
The provided photoalbum\_OCI\_v3.0 code was deployed in /var/www/html/cos80001/photoalbum/. The constants.php file was modified with correct database credentials and column mappings. The album.php page displayed the images and metadata successfully.

Figure 16: album.php Page with Functional Output [PLACEHOLDER]  
Figure 17: constants.php Code Snippet [PLACEHOLDER]

# IV. TESTING AND VALIDATION

Connectivity between the public and private subnets was validated using the ping command. The web server (bastion) was used to SSH into the private test instance and verify ICMP reachability. This test confirmed that NSG and Security List rules were correctly enforced.

Figure 18: Terminal - Ping from Web Server to Test Instance [PLACEHOLDER]  
Figure 19: SSH into Test Instance via Bastion [PLACEHOLDER]

# V. CHALLENGES AND LEARNINGS

One major challenge was understanding the layered access control model of OCI—particularly the combined use of NSGs and Security Lists. Initially, NSG rules were not enforced due to interference from the default security list. This was resolved by explicitly removing all rules from the default security list.

Another challenge involved attaching a Reserved Public IP. The initial web server used an ephemeral IP, which changed upon reboot. After studying the OCI documentation, the correct steps to unassign the default public IP and reattach the reserved one were implemented successfully.

The process also reinforced the importance of enabling public access at the bucket level rather than configuring individual object permissions. Understanding Pre-Authenticated Requests (PARs) was crucial for secure and public access to stored objects.

# VI. CONCLUSION

This assignment demonstrated a successful end-to-end deployment of a web-based Photo Album system on Oracle Cloud Infrastructure. All core components—VCN, subnets, compute instances, object storage, database, and web server—were set up following best practices and security principles. The web application functioned correctly with database-backed metadata and public access to image files. This exercise highlighted practical differences between AWS and OCI, especially in access control, IP management, and network configuration, enhancing the student’s understanding of multi-cloud architectures.

# VII. REFERENCES

[1] Oracle Cloud Infrastructure Documentation. (2024). Virtual Cloud Network (VCN). <https://docs.oracle.com/en-us/iaas/Content/Network/Tasks/managingVCNs.htm>  
[2] Oracle Cloud Infrastructure. (2024). Managing MySQL Databases on OCI. <https://docs.oracle.com/en-us/iaas/mysql-database/doc/index.html>  
[3] Oracle Cloud Infrastructure. (2024). Object Storage Overview. <https://docs.oracle.com/en-us/iaas/Content/Object/Concepts/objectstorageoverview.htm>  
[4] Oracle Cloud Infrastructure. (2024). Reserved Public IP Addresses. <https://docs.oracle.com/en-us/iaas/Content/Network/Tasks/managingpublicIPs.htm>  
[5] Swinburne Canvas. (2024). COS80001 Assignment 2 Guidelines & Labs.

**NOTE:** Replace all [PLACEHOLDER] labels with your respective screenshots and data before submission. Ensure all screenshots are readable, have visible usernames, and include clear figure captions as numbered above.