ORACLE

OIC Outbound IP Address TCE - Emerging Tech

Pattern Definition Document

November, 2021 | Version 1.0

1 VERSION CONTROL

Version	Author	Date	Comment
1.0	Roberto Cavenaghi	Nov 2nd, 2021	Updates after internal review
0.3	Roberto Cavenaghi	Oct 11th, 2021	Adopted new layout
0.2	Roberto Cavenaghi	Oct 8th, 2021	Added physical details
0.1	Roberto Cavenaghi	Oct 6th, 2021	Initial version

2 REVIEWERS

Name	Organization	Position	Contact
Giovanni Conte	Oracle	W/L Integration Architect	giovanni.conte@oracle.com
Marcel Straka	Oracle	W/L Integration Practice Leader	marcel.straka@oracle.com
Alexander Hodicke	Oracle	W/L Arch Best Practice Leader	alexander.hodicke@oracle.com

3 ABBREVIATIONS LIST

Abbreviation	Meaning
OCI	Oracle Cloud Infrastructure
NGW	NAT Gateway
OIC	Oracle Integration Cloud
API	Applicaton Programming Interface
ATP	Oracle Autonomous Transaction Processing
PaaS	Platform as a service
IaaS	Infrastructure as a service
SLA	Service Level Agreement
FAQs	Frequently Asked Questions

TABLE OF CONTENTS

4 NAME

OIC Outbound IP Address

5 PROBLEM

This pattern address one specific need when using Oracle Integration Cloud OIC for invoking an external web service: define a custom IP address as outbound address for OIC external calls, in order to reach the external service. Some external services need to whitelist the client IP address to allow proper client filtering security. OIC has a unique IP address used at whole Data Center level, and this remains always the same for all the OIC instances inside the Data Center: that's fine because this is the IP address to specify in the whitelist. There are cases in which the requirement of IP filtering is extented, because the client IP is also used as the actual identity of the caller, so the same address for all the integration instances doesn't work anymore. Furthermore the IP address can also be associated with a security token and the pair (ip address, token) is checked by the server on every invocation: if the key doesn't belong to the expected client IP address the service return a security error.

6 CONTEXT

The actual OIC running on OCI Gen2 infrastructure manages outbound call from the Integration Service using an internal NAT service valid for all the integration instances inside the Data Center, regardless of the tenancy they belong to (one ip address for all the Integration instances). This doesn't allow to support the security requirement cases described in the previous chapter.

The requirement is to be able to use a specific IP address as outbound address (not a general one) for an Integration instance or even for a set of integration flows defined in a single Integration instance.

7 SOLUTION

A viable solution to this problem is the usage of a mix of basic OIC and OCI components:

- OIC Connectivity agent;
- OCI Compute;
- OCI Reserved Public IP;
- OCI NAT Gateway.

7.1 OIC Connectivity Agent

The connectivity agent is an OIC component that act as a pass through between the Oracle Integration instance and the target services deployed in a private subnet. The connectivity agent is a java standalone executable, it needs to be installed in a host (OCI or on premise) that has network connectivity with the target service (usually same subnet). Reference to Integration agent documentation is here.

7.2 Compute

The OCI Compute service enables you to provision and manage compute hosts in the cloud. You can launch compute instances with shapes that meet your resource requirements (CPU, memory, network bandwidth, and storage). Computes come in various flavors: VM, bare metal, etc.

7.3 Reserved Public IP

Reserved Public IP is an OCI object that represent a public IP address and has it own lifecycle (as opposed to Ephemeral IP address). It can exists also unrelated to any other network component like NIC, Load Balancer, etc. A Reserved Public IP Address is part of a pool of IP addresses that can be also provided by the user.
Reference to Public IP Address OCI object documentation is here.

7.4 NAT gateway

OCI Network Address Translator Gateway is a network component that allows only outbound connection to Internet, no inbound connectivity is possible. It hides internal IP addresses of the clients exposing an unique IP address to the server.

7.5 Overall solution picture

The overall solution architecture is shown in Figure 1 where all the previously described components are present, the arrows represent the data movement along the architecture.

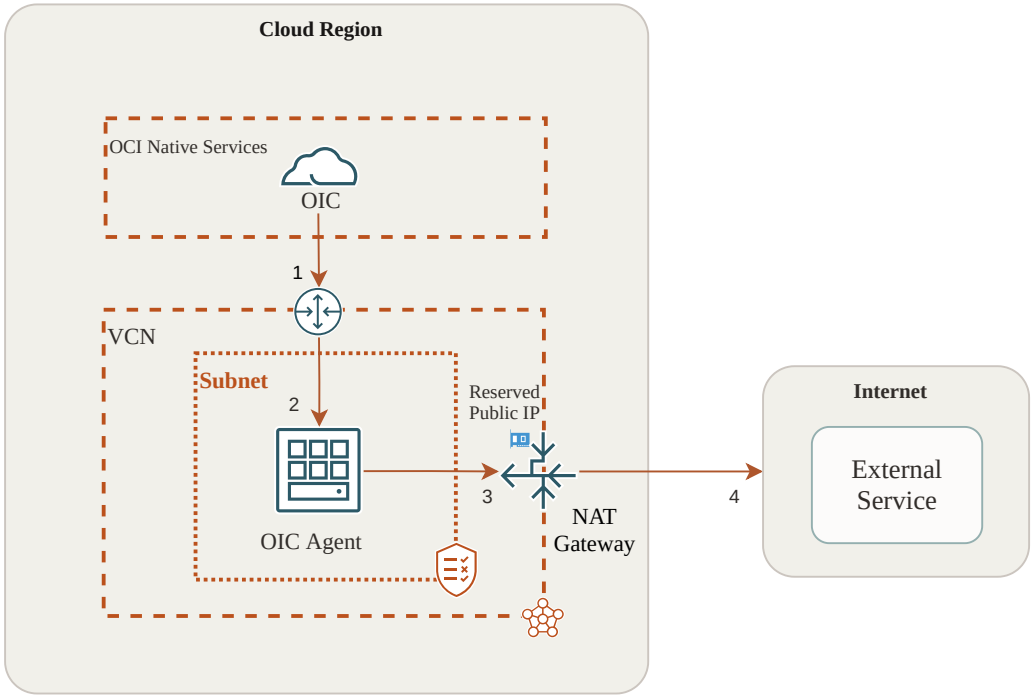

Figure 1: Logical Architecture

The steps pointed out in the diagram are:

1. an OIC integration flow invokes the external service, but since it is configured to use the connectivity agent the request is parked waiting to be picked up by the agent;
2. the request is picked up by the Connectivity Agent using the Service Gateway;
3. the Connectivity Agent makes the request to the external service, using the NAT gateway configured for using the Reserved Public IP;
4. the payload is sent to the external service that can accept the request because the client IP is a trusted one.

8 TECHNICAL DETAILS

Figure 2 shows a physical representation of the previous logical architecture with the connectivity details for each component.

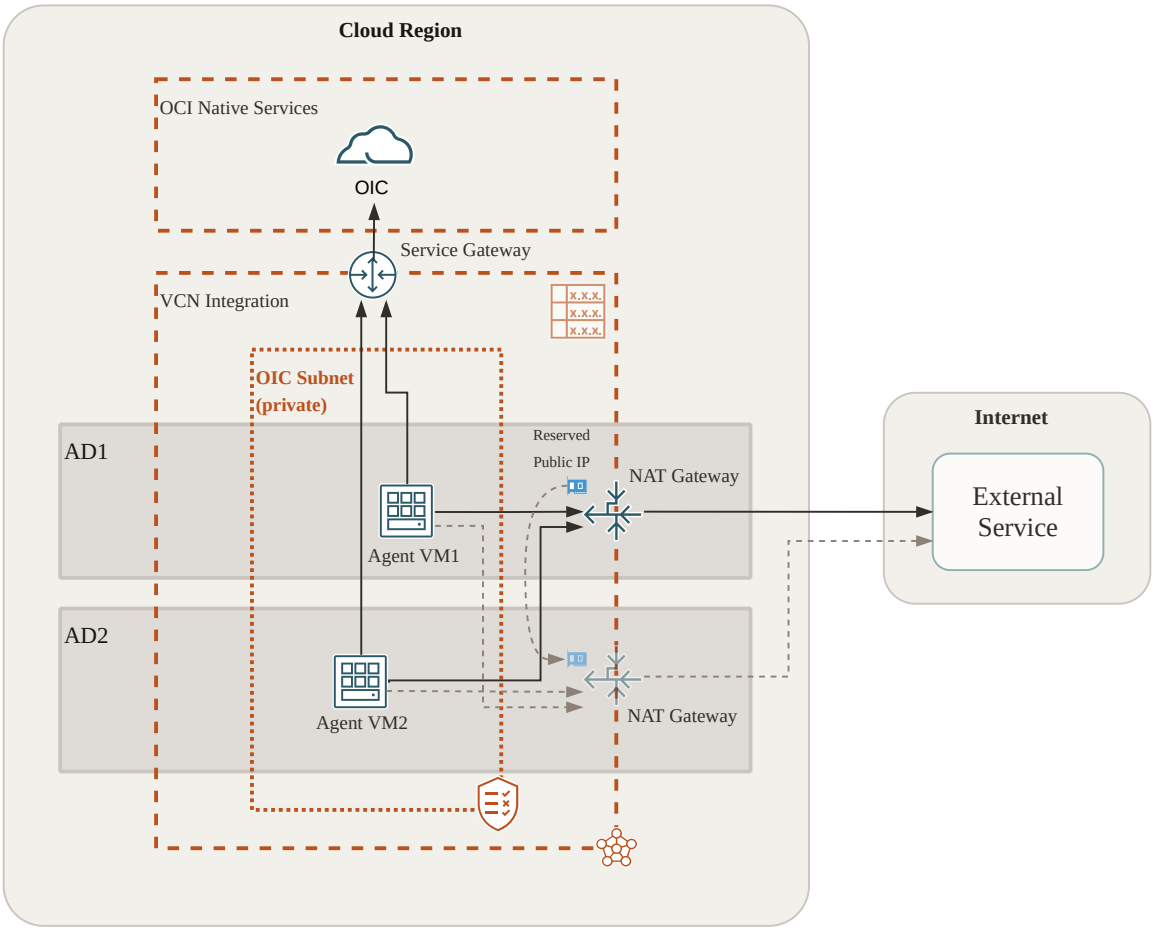

Figure 2: Physical Architecture

The physical architecture focuses on the high availability of the architecture using the OCI Availability Domains where the two VMs are deployed (*AD1* and *AD2*). Such configuration highlights also the high availability of the network device, so the NAT Gateway, using again the Availability Domains: in case of any network outage on one AD a second NAT Gateway on the other AD guarantees the traffic flow.

This architecture achieve a transparent floating IP functionality, which is not available if attaching the IP Address directly to a VNIC (e.g. to a compute).

A floating IP functionality between two computes can also be achieved using Linux software clustering solution like Corosync and Pacemaker. This software solution guarantees a complete lifecycle of the service: every change of the service status triggers a script invocation for manage that specific status (e.g. service down status triggers the start script on the second node). Even if effective this solution is rather complex to configure and needs also small implementations over the available set of predefined scripts. An example for such a configuration, but for a private IP address, can be found here.

8.1 Flows Repartition Example

The solution is flexible because can be applied not only to a whole OIC instance but also on a subset of integration flows, or even to define a set of partitions each with its own outbound IP address. This is achieved through the usage of a specific Connectivity Agent (configured in a specific OIC connection used by the flow) and a specific NAT Gateway:

The following table is an example of grouping the integration flows deployed in an OIC instance, per Departments with a specific and unique IP address each.

OIC Flow Name	OIC Connection Name	Department	Agent	IP address
HireEmployee	HRExternalService	HR	HRAgent	193.125.10.1
TrainEmployee	HRExternalService	HR	HRAgent	193.125.10.1
CreateOpportunity	SalesExternalService	Sales	SalesAgent	193.125.15.1
CreateQuotation	SalesExternalService	Sales	SalesAgent	193.125.15.1
SignContract	SalesExternalService	Sales	SalesAgent	193.125.15.1
BulkInvoice	InvoicingExternalService	Invoicing	InvoicingAgent	193.125.18.1

In Figure 3 it can be seen the solution scaled for fulfill the three-department architecture of the example.

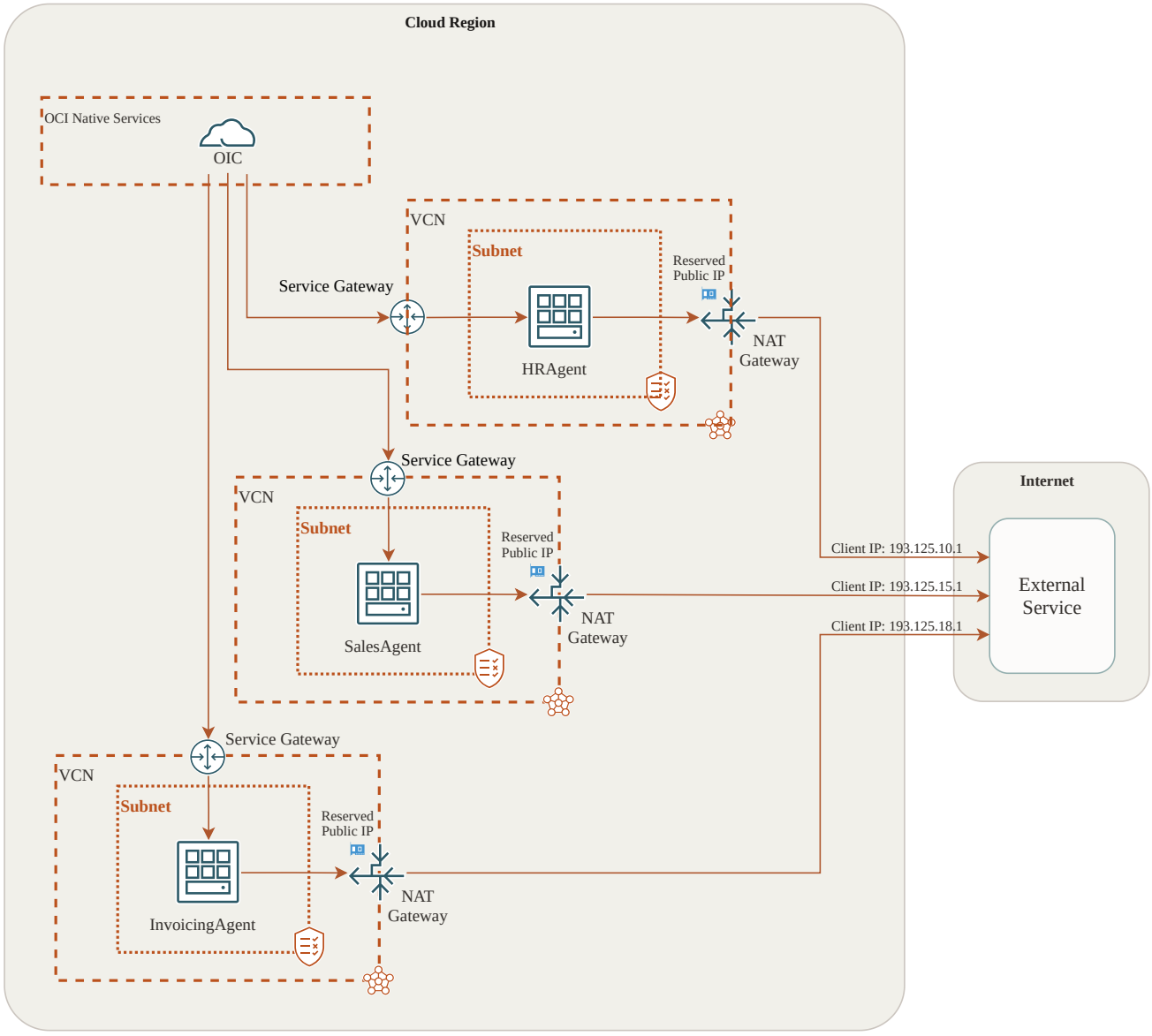

Figure 3: MultiDepartment Architecture