

STEP BY STEP GUIDE TO ORACLE DATA GUARD

Table of Contents

Overview	2
Usage.....	2
Details of the Build Process.....	3
Target Data Guard Environment	3
Provisioning the infrastructure	4
Configuring Primary	9
Disable Firewalls.....	9
Set Oracle user password.....	10
Create DB Instance.....	10
Update .bashrc.....	11
Configure Listener	13
Configure Name Resolution	18
Check Password File	21
Enable ArchiveLog Mode	21
Check DB Instance Configuration	23
Enable Force Logging	30
Create Standby Redo Logs.....	31
Set Database Parameters	34
Create PFILE for Standby Server (optional).....	40
Create Control File for Standby Server (optional).....	42
Check Directory Structure on Primary Server	42
Configuring Standby.....	44
Disable Firewalls.....	45
Set Oracle user password.....	45
Update .bashrc.....	45
Create Subdirectory Structure	47
Create Password File	47
Configure Listener	48
Configure Name Resolution	51

Create PFILE	53
Start Standby DB as NOMOUNT.....	53
Copy DB from Primary using RMAN and Modify SPFILE Parameters.....	54
Check Standby DB Configuration	59
Start the Apply Redo Process	72
Verify Data Guard Configuration	73
Testing Switchover and Switchback without Data Guard Broker	82
Testing Failover without Data Guard Broker	86
Configuring Data Guard Broker	90
Testing SwitchOver and SwitchBack using Data Guard Broker	101

Overview

This setup is similar to Tim Gorman's setup here [Implement Oracle Data Guard on an Azure Linux virtual machine - Azure Virtual Machines | Microsoft Learn](#)

For Data Guard configuration, I use the example here:

<https://docs.oracle.com/en/database/oracle/oracle-database/19/sbydb/creating-oracle-data-guard-physical-standby.html#GUID-B511FB6E-E3E7-436D-94B5-071C37550170>

Usage

Subscription: admin@m365x533836.onmicrosoft.com

Resource Group: cenk-oracledg-custom

Username: `azureuser`

Password: `OracleLab123`

After logging into the VMs (OracleVM1 and OracleVM2) as "azureuser" via Bastion, do the following:

```
$ sudo su - oracle
```

If the VMs are rebooted, you must ensure that the Oracle listener is started.

```
oracle$ lsnrctl stop  
oracle$ lsnrctl start
```

Next, you ensure that the DBs are operational as well:

- On VM1, start the DB normally using “`SQL> startup`” (already automated)
- On VM2, use “`SQL> STARTUP NOMOUNT`”;

In VM2, you can restart the Redo Log Transport as follows:

```
SQL> ALTER DATABASE RECOVER MANAGED STANDBY DATABASE CANCEL;  
SQL> ALTER DATABASE RECOVER MANAGED STANDBY DATABASE DISCONNECT;
```

On VM1 or VM2, you can view the Data Guard configuration:

```
oracle$ dgmgrl sys/OracleLab123@cdb1  
DGMGRL> SHOW CONFIGURATION;
```

If necessary, you can also switch to root user to shut down the VMs:

```
$ sudo su -
```

Details of the Build Process

Target Data Guard Environment

We would like to build the following environment:

Field Name	PRIMARY	STANDBY	Explanation
hostname	OracleVM1	OracleVM2	VM name on Azure
DB_NAME	dubai	dubai	must be identical
DOMAIN_NAME	localdomain	localdomain	
DB_UNIQUE_NAME	dubai	dubaistby	must be different

ORACLE_SID	dubai	dubaistby	
Static Listener Service Name	dubaiserv	dubaistbserv	Can be named anything as long as it matches tnsnames.ora
Connect identifier in tnsnames.ora	DXB	DXBSTBY	

Provisioning the infrastructure

1 – Start the Azure Cloud Shell. <https://shell.azure.com>

```
LOCATION=eastus
RESOURCE_GROUP="cenk-oracledg-custom"
VM_USERNAME="azureuser"
VM_PASSWORD="OracleLab123"
VNET_NAME="${RESOURCE_GROUP}VNet"
```

```
mod@Azure:~$ LOCATION=eastus
mod@Azure:~$ RESOURCE_GROUP="cenk-oracledg-custom"
mod@Azure:~$ VM_USERNAME="azureuser"
mod@Azure:~$ VM_PASSWORD="OracleLab123"
mod@Azure:~$ VNET_NAME="${RESOURCE_GROUP}VNet"
mod@Azure:~$ 
```

```
az extension add --name bastion
mod@Azure:~$ az extension add --name bastion
Extension 'bastion' 0.2.4 is already installed.
mod@Azure:~$ 
```

```
az group create --name $RESOURCE_GROUP --location $LOCATION
mod@Azure:~$ az group create --name $RESOURCE_GROUP --location $LOCATION
{
  "id": "/subscriptions//resourceGroups/cenk-oracledg-custom",
  "location": "eastus",
  "managedBy": null,
  "name": "cenk-oracledg-custom",
  "properties": {
    "provisioningState": "Succeeded"
  },
  "tags": null,
  "type": "Microsoft.Resources/resourceGroups"
}
mod@Azure:~$ 
```

```
az network vnet create --resource-group $RESOURCE_GROUP --location $LOCATION --name $VNET_NAME --address-prefix "10.0.0.0/16"
```

```
mod@Azure:-$ az network vnet create --resource-group $RESOURCE_GROUP --location $LOCATION --name $VNET_NAME --address-prefix "10.0.0.0/16"
Command group 'az network' is in preview and under development. Reference and support levels: https://aka.ms/CLI_refstatus
{
  "newVNet": {
    "addressSpace": {
      "addressPrefixes": [
        "10.0.0.0/16"
      ]
    },
    "enableDdosProtection": false,
    "etag": "W/"d3617292-993e-45c9-bee9-e4ee633107fd"",
    "id": "/subscriptions//resourceGroups/cenk-oracledg-custom/providers/Microsoft.Network/virtualNetworks/cenk-oracledg-customVNet",
    "location": "eastus",
    "name": "cenk-oracledg-customVNet",
    "provisioningState": "Succeeded",
    "resourceGroup": "cenk-oracledg-custom",
    "resourceGuid": "d9f8bd5b-81f9-419c-9e74-347a7885b5f3",
    "subnets": [],
    "type": "Microsoft.Network/virtualNetworks",
    "virtualNetworkPeerings": []
  }
}
mod@Azure:-$ []
```

```
az network vnet subnet create --resource-group $RESOURCE_GROUP --name AzureBastionSubnet --vnet-name $VNET_NAME --address-prefixes 10.0.0.0/24
```

```
mod@Azure:-$ az network vnet subnet create --resource-group $RESOURCE_GROUP --name AzureBastionSubnet --vnet-name $VNET_NAME --address-prefixes 10.0.0.0/24
Command group 'az network' is in preview and under development. Reference and support levels: https://aka.ms/CLI_refstatus
{
  "addressPrefix": "10.0.0.0/24",
  "delegations": [],
  "etag": "W/"239a4ff4-cf18-49ea-a38d-1ac84772ca36"",
  "id": "/subscriptions//resourceGroups/cenk-oracledg-custom/providers/Microsoft.Network/virtualNetworks/cenk-oracledg-customVNet/subnet
s/AzureBastionSubnet",
  "name": "AzureBastionSubnet",
  "privateEndpointNetworkPolicies": "Disabled",
  "privateLinkServiceNetworkPolicies": "Enabled",
  "provisioningState": "Succeeded",
  "resourceGroup": "cenk-oracledg-custom",
  "type": "Microsoft.Network/virtualNetworks/subnets"
}
mod@Azure:-$ []
```

```
az network vnet subnet create --resource-group $RESOURCE_GROUP --name OracleSubnet --vnet-name $VNET_NAME --address-prefixes 10.0.1.0/24
```

```
mod@Azure:-$ az network vnet subnet create --resource-group $RESOURCE_GROUP --name OracleSubnet --vnet-name $VNET_NAME --address-prefixes 10.0.1.0/24
Command group 'az network' is in preview and under development. Reference and support levels: https://aka.ms/CLI_refstatus
{
  "addressPrefix": "10.0.1.0/24",
  "delegations": [],
  "etag": "W/"8ed454f3-892f-43ac-af40-746013651678"",
  "id": "/subscriptions//resourceGroups/cenk-oracledg-custom/providers/Microsoft.Network/virtualNetworks/cenk-oracledg-customVNet/subnet
s/OracleSubnet",
  "name": "OracleSubnet",
  "privateEndpointNetworkPolicies": "Disabled",
  "privateLinkServiceNetworkPolicies": "Enabled",
  "provisioningState": "Succeeded",
  "resourceGroup": "cenk-oracledg-custom",
  "type": "Microsoft.Network/virtualNetworks/subnets"
}
mod@Azure:-$ []
```

```
az network nsg create --name OracleVM-NSG --resource-group $RESOURCE_GROUP --location $LOCATION
```

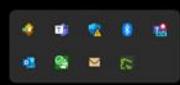
```

mod@Azure:~$ az network nsg create --name OracleVM-NSG --resource-group $RESOURCE_GROUP --location $LOCATION
Command group 'az network' is in preview and under development. Reference and support levels: https://aka.ms/CLI\_refstatus

{
  "NewNSG": {
    "defaultSecurityRules": [
      {
        "access": "Allow",
        "description": "Allow inbound traffic from all VMs in VNET",
        "destinationAddressPrefix": "VirtualNetwork",
        "destinationAddressPrefixes": [],
        "destinationPortRange": "*",
        "destinationPortRanges": [],
        "direction": "Inbound",
        "etag": "W/"fa268732-d45a-46c2-a04d-31b73d7ac4da"",
        "id": "/subscriptions//resourceGroups/cenk-oracledg-custom/providers/Microsoft.Network/networkSecurityGroups/OracleVM-NSG/defaultSecurityRules/AllowVnetInBound",
        "name": "AllowVnetInbound",
        "priority": 65800,
        "protocol": "*",
        "provisioningState": "Succeeded",
        "resourceGroup": "cenk-oracledg-custom",
        "sourceAddressPrefix": "VirtualNetwork",
        "sourceAddressPrefixes": [],
        "sourcePortRange": "*",
        "sourcePortRanges": [],
        "type": "Microsoft.Network/networkSecurityGroups/defaultSecurityRules"
      },
      {
        "access": "Allow",
        "description": "Allow inbound traffic from azure load balancer",
        "destinationAddressPrefix": "*",
        "destinationAddressPrefixes": [],
        "destinationPortRange": "*",
        "destinationPortRanges": [],
        "direction": "Inbound",
        "etag": "W/"fa268732-d45a-46c2-a04d-31b73d7ac4da"",
        "id": "/subscriptions//resourceGroups/cenk-oracledg-custom/providers/Microsoft.Network/networkSecurityGroups/OracleVM-NSG/defaultSecurityRules/AllowAzureLoadBalancerInBound",
        "name": "AllowAzureLoadBalancerInBound",
        "priority": 65801,
        "protocol": "*",
        "provisioningState": "Succeeded",
        "resourceGroup": "cenk-oracledg-custom",
        "sourceAddressPrefix": "AzureLoadBalancer",
        "sourceAddressPrefixes": [],
        "sourcePortRange": "*",
        "sourcePortRanges": [],
        "type": "Microsoft.Network/networkSecurityGroups/defaultSecurityRules"
      },
      {
        "access": "Deny",
        "description": "Deny all inbound traffic",
        "destinationAddressPrefix": "*",
        "destinationAddressPrefixes": [],
        "destinationPortRange": "*",
        "destinationPortRanges": [],
        "direction": "Inbound",
        "etag": "W/"fa268732-d45a-46c2-a04d-31b73d7ac4da"",
        "id": "/subscriptions//resourceGroups/cenk-oracledg-custom/providers/Microsoft.Network/networkSecurityGroups/OracleVM-NSG/defaultSecurityRules/DenyAllInBound",
        "name": "DenyAllInbound",
        "priority": 65500,
        "protocol": "*",
        "provisioningState": "Succeeded",
        "resourceGroup": "cenk-oracledg-custom",
        "sourceAddressPrefix": "*",
        "sourceAddressPrefixes": [],
        "sourcePortRange": "*",
        "sourcePortRanges": [],
        "type": "Microsoft.Network/networkSecurityGroups/defaultSecurityRules"
      }
    ],
    "sourceAddressPrefix": "*",
    "sourceAddressPrefixes": [],
    "sourcePortRange": "*",
    "sourcePortRanges": [],
    "type": "Microsoft.Network/networkSecurityGroups/defaultSecurityRules"
  },
  {
    "access": "Allow",
    "description": "Allow outbound traffic from all VMs to all VMs in VNET",
    "destinationAddressPrefix": "VirtualNetwork",
    "destinationAddressPrefixes": [],
    "destinationPortRange": "*",
    "destinationPortRanges": [],
    "direction": "Outbound",
    "etag": "W/"fa268732-d45a-46c2-a04d-31b73d7ac4da"",
    "id": "/subscriptions//resourceGroups/cenk-oracledg-custom/providers/Microsoft.Network/networkSecurityGroups/OracleVM-NSG/defaultSecurityRules/AllowVnetOutBound",
    "name": "AllowVnetOutBound",
    "priority": 65800,
    "protocol": "*",
    "provisioningState": "Succeeded",
    "resourceGroup": "cenk-oracledg-custom",
    "sourceAddressPrefix": "VirtualNetwork",
    "sourceAddressPrefixes": [],
    "sourcePortRange": "*",
    "sourcePortRanges": [],
    "type": "Microsoft.Network/networkSecurityGroups/defaultSecurityRules"
  },
  {
    "access": "Allow",
    "description": "Allow outbound traffic from all VMs to Internet",
    "destinationAddressPrefix": "Internet",
    "destinationAddressPrefixes": []
  }
}

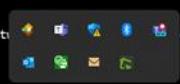
```





```

"destinationAddressPrefixes": [],
"destinationPortRange": "*",
"destinationPortRanges": [],
"direction": "Outbound",
"etag": "W/"fa268732-d45a-46c2-a04d-31b73d7ac4da\"",
"id": "/subscriptions/$AZURE_SHELL_ID/resourceGroups/cenk-oracledg-custom/providers/Microsoft.Network/networkSecurityGroups/OracleVM-NSG/defaultSecurityRules/AllowInternetOutBound",
"name": "AllowInternetOutBound",
"priority": 65801,
"protocol": "*",
"provisioningState": "Succeeded",
"resourceGroup": "cenk-oracledg-custom",
"sourceAddressPrefix": "*",
"sourceAddressPrefixes": [],
"sourcePortRange": "*",
"sourcePortRanges": [],
"type": "Microsoft.Network/networkSecurityGroups/defaultSecurityRules"
},
{
"access": "Deny",
"description": "Deny all outbound traffic",
"destinationAddressPrefix": "*",
"destinationAddressPrefixes": [],
"destinationPortRange": "*",
"destinationPortRanges": [],
"direction": "Outbound",
"etag": "W/"fa268732-d45a-46c2-a04d-31b73d7ac4da\"",
"id": "/subscriptions/$AZURE_SHELL_ID/resourceGroups/cenk-oracledg-custom/providers/Microsoft.Network/networkSecurityGroups/OracleVM-NSG/defaultSecurityRules/DenyAllOutBound",
"name": "DenyAllOutBound",
"priority": 65500,
"protocol": "*",
"provisioningState": "Succeeded",
"resourceGroup": "cenk-oracledg-custom",
"sourceAddressPrefix": "*",
"sourceAddressPrefixes": [],
"sourcePortRange": "*",
"sourcePortRanges": [],
"type": "Microsoft.Network/networkSecurityGroups/defaultSecurityRules"
}
],
"etag": "W/"fa268732-d45a-46c2-a04d-31b73d7ac4da\"",
"id": "/subscriptions/$AZURE_SHELL_ID/resourceGroups/cenk-oracledg-custom/providers/Microsoft.Network/networkSecurityGroups/OracleVM-NSG",
"location": "eastus",
"name": "OracleVM-NSG",
"provisioningState": "Succeeded",
"resourceGroup": "cenk-oracledg-custom",
"resourceGuid": "d8066081-dcc8-4f45-9a07-89426edfeb61",
"securityRules": [],
"type": "Microsoft.Network/networkSecurityGroups"
}
}
mod@Azure:~$ 
```



```

az vm availability-set create --resource-group $RESOURCE_GROUP --name OracleVMAvailabilitySet --platform-fault-domain-count 2 --platform-update-domain-count 2
mod@Azure:~$ az vm availability-set create --resource-group $RESOURCE_GROUP --name OracleVMAvailabilitySet --platform-fault-domain-count 2 --platform-update-domain-count 2
{
  "id": "/subscriptions/$AZURE_SHELL_ID/resourceGroups/cenk-oracledg-custom/providers/Microsoft.Compute/availabilitySets/OracleVMAvailabilitySet",
  "location": "eastus",
  "name": "OracleVMAvailabilitySet",
  "platformFaultDomainCount": 2,
  "platformUpdateDomainCount": 2,
  "proximityPlacementGroup": null,
  "resourceGroup": "cenk-oracledg-custom",
  "sku": {
    "capacity": null,
    "name": "Aligned",
    "tier": null
  },
  "statuses": null,
  "tags": {},
  "type": "Microsoft.Compute/availabilitySets",
  "virtualMachines": []
}
mod@Azure:~$ 
```

```

az vm create --resource-group $RESOURCE_GROUP --name OracleVM1 --availability-set OracleVMAvailabilitySet --image Oracle:oracle-database-19-3:oracle-database-19-0904:latest --size Standard_DS1_v2 --authentication-type password --admin-$AZURE_SHELL_ID
mod@Azure:~$ 
```

```

username $VM_USERNAME --admin-password $VM_PASSWORD --vnet-name $VNET_NAME --
subnet OracleSubnet --nsg OracleVM-NSG --os-disk-size-gb 32
mod@Azure:-$ az vm create --resource-group $RESOURCE_GROUP --name OracleVM1 --availability-set OracleVMAvailabilitySet --image Oracle:oracle-database-19-3:oracle-database-19-0904:latest --size Standard_DS1_v2 --authentication-type password --admin-username $VM_USERNAME --admin-password $VM_PASSWORD --vnet-name $VNET_NAME --subnet OracleSubnet --nsg OracleVM-NSG --os-disk-size-gb 32
Ignite (November) 2023 onwards "az vm/vmss create" command will deploy Gen2-Trusted Launch VM by default. To know more about the default change and Trusted Launch, please visit https://aka.ms/TLaD
It's recommended to create with '--public-ip-sku Standard'. Please be aware that the default Public IP will be changed from Basic to Standard in the next release. Also note that Basic option will be removed in the future.
{
  "fqdns": "",
  "id": "/subscriptions//resourceGroups/cenk-oracledg-custom/providers/Microsoft.Compute/virtualMachines/OracleVM1",
  "location": "eastus",
  "macAddress": "00-22-48-30-4E-E7",
  "powerState": "VM running",
  "privateIpAddress": "10.0.1.4",
  "publicIpAddress": "172.174.25.188",
  "resourceGroup": "cenk-oracledg-custom",
  "zones": ""
}
mod@Azure:-$ []

```

```

az vm create --resource-group $RESOURCE_GROUP --name OracleVM2 --availability-set OracleVMAvailabilitySet --image Oracle:oracle-database-19-3:oracle-database-19-0904:latest --size Standard_DS1_v2 --authentication-type password --admin-username $VM_USERNAME --admin-password $VM_PASSWORD --vnet-name $VNET_NAME --
subnet OracleSubnet --nsg OracleVM-NSG --os-disk-size-gb 32
mod@Azure:-$ az vm create --resource-group $RESOURCE_GROUP --name OracleVM2 --availability-set OracleVMAvailabilitySet --image Oracle:oracle-database-19-3:oracle-database-19-0904:latest --size Standard_DS1_v2 --authentication-type password --admin-username $VM_USERNAME --admin-password $VM_PASSWORD --vnet-name $VNET_NAME --subnet OracleSubnet --nsg OracleVM-NSG --os-disk-size-gb 32
Ignite (November) 2023 onwards "az vm/vmss create" command will deploy Gen2-Trusted Launch VM by default. To know more about the default change and Trusted Launch, please visit https://aka.ms/TLaD
It's recommended to create with '--public-ip-sku Standard'. Please be aware that the default Public IP will be changed from Basic to Standard in the next release. Also note that Basic option will be removed in the future.
{
  "fqdns": "",
  "id": "/subscriptions//resourceGroups/cenk-oracledg-custom/providers/Microsoft.Compute/virtualMachines/OracleVM2",
  "location": "eastus",
  "macAddress": "00-00-3A-8A-62-7C",
  "powerState": "VM running",
  "privateIpAddress": "10.0.1.5",
  "publicIpAddress": "172.190.148.72",
  "resourceGroup": "cenk-oracledg-custom",
  "zones": ""
}
mod@Azure:-$ []

```

```

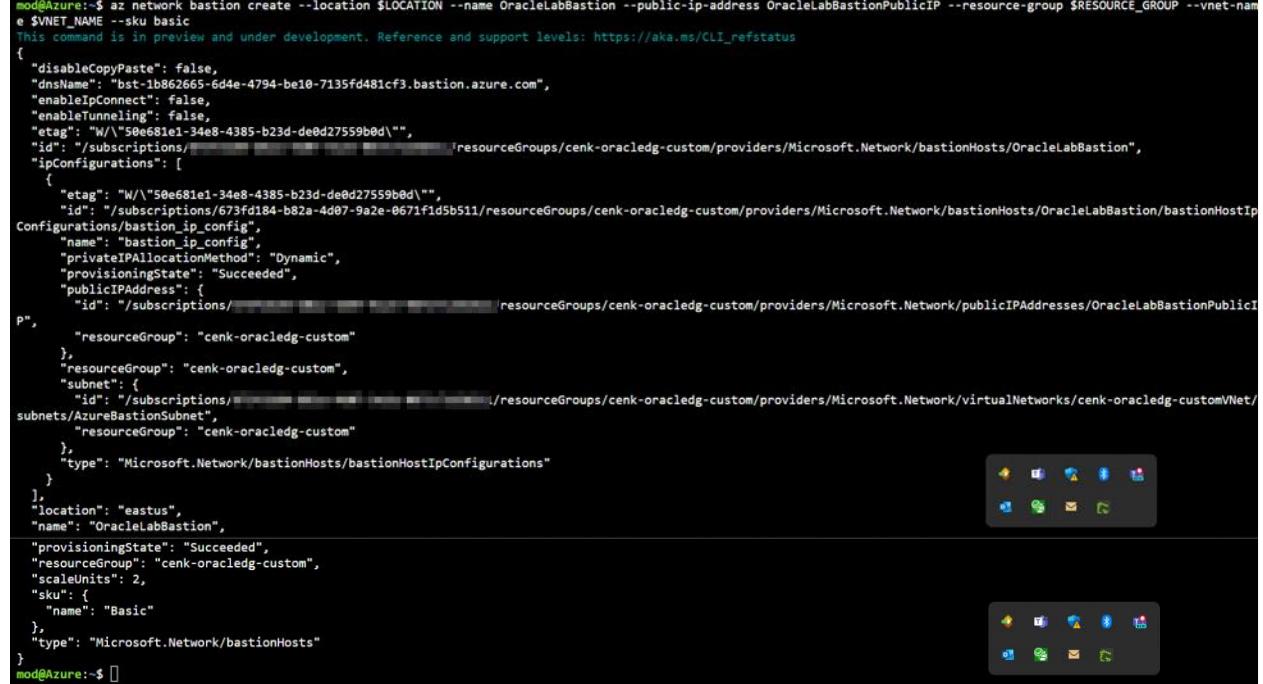
az network public-ip create --resource-group $RESOURCE_GROUP --name OracleLabBastionPublicIP --sku Standard
mod@Azure:-$ az network public-ip create --resource-group $RESOURCE_GROUP --name OracleLabBastionPublicIP --sku Standard
Command group 'az network' is in preview and under development. Reference and support levels: https://aka.ms/CLI_refstatus
[Coming breaking change] In the coming release, the default behavior will be changed as follows when sku is Standard and zone is not provided: For zonal regions, you will get a zone-redundant IP indicated by zones:[1","2","3"]; For non-zonal regions, you will get a non zone-redundant IP indicated by zones:null.
{
  "publicIp": {
    "ddosSettings": {
      "protectionMode": "VirtualNetworkInherited"
    },
    "etag": "W/\\"86cc3946-37e7-4972-9012-964d7e315a0a\\\"",
    "id": "/subscriptions//resourceGroups/cenk-oracledg-custom/providers/Microsoft.Network/publicIPAddresses/OracleLabBastionPublicIP",
    "idleTimeoutInMinutes": 4,
    "ipAddress": "172.190.148.209",
    "ipTags": [],
    "location": "eastus",
    "name": "OracleLabBastionPublicIP",
    "provisioningState": "Succeeded",
    "publicIPAddressVersion": "IPv4",
    "publicIPAllocationMethod": "Static",
    "resourceGroup": "cenk-oracledg-custom",
    "resourceGuid": "618e60c8-a1dd-4447-bc4a-0d9839a7d3ac",
    "sku": {
      "name": "Standard",
      "tier": "Regional"
    },
    "type": "Microsoft.Network/publicIPAddresses"
  }
}
mod@Azure:-$ []

```

```

az network bastion create --location $LOCATION --name OracleLabBastion --public-ip-address OracleLabBastionPublicIP --resource-group $RESOURCE_GROUP --vnet-name $VNET_NAME --sku basic
mod@Azure:-$ az network bastion create --location $LOCATION --name OracleLabBastion --public-ip-address OracleLabBastionPublicIP --resource-group $RESOURCE_GROUP --vnet-name $VNET_NAME --sku basic
This command is in preview and under development. Reference and support levels: https://aka.ms/CLI_refstatus
{
  "disableCopyPaste": false,
  "dnsName": "bast-1b862665-6d4e-4794-be10-7135fd481cf3.bastion.azure.com",
  "enableIpConnect": false,
  "enableTunneling": false,
  "etag": "W/\"50e681e1-34e8-4385-b23d-de0d27559b0d\"",
  "id": "/subscriptions/.../resourceGroups/cenk-oracledg-custom/providers/Microsoft.Network/bastionHosts/OracleLabBastion",
  "ipConfigurations": [
    {
      "etag": "W/\"50e681e1-34e8-4385-b23d-de0d27559b0d\"",
      "id": "/subscriptions/673fd184-b82a-4d07-9a2e-0671f1d5b511/resourceGroups/cenk-oracledg-custom/providers/Microsoft.Network/bastionHosts/OracleLabBastion/bastionHostIpConfigurations/bastion_ip_config",
      "name": "bastion_ip_config",
      "privateIPAllocationMethod": "Dynamic",
      "provisioningState": "Succeeded",
      "publicIPAddress": {
        "id": "/subscriptions/.../resourceGroups/cenk-oracledg-custom/providers/Microsoft.Network/publicIPAddresses/OracleLabBastionPublicIP"
      },
      "resourceGroup": "cenk-oracledg-custom"
    },
    {
      "etag": "W/\"50e681e1-34e8-4385-b23d-de0d27559b0d\"",
      "id": "/subscriptions/.../resourceGroups/cenk-oracledg-custom/providers/Microsoft.Network/virtualNetworks/cenk-oracledg-customVNet/subnets/AzureBastionSubnet",
      "resourceGroup": "cenk-oracledg-custom"
    }
  ],
  "location": "eastus",
  "name": "OracleLabBastion",
  "provisioningState": "Succeeded",
  "resourceGroup": "cenk-oracledg-custom",
  "scaleUnits": 2,
  "sku": {
    "name": "Basic"
  },
  "type": "Microsoft.Network/bastionHosts"
}
mod@Azure:-$ []

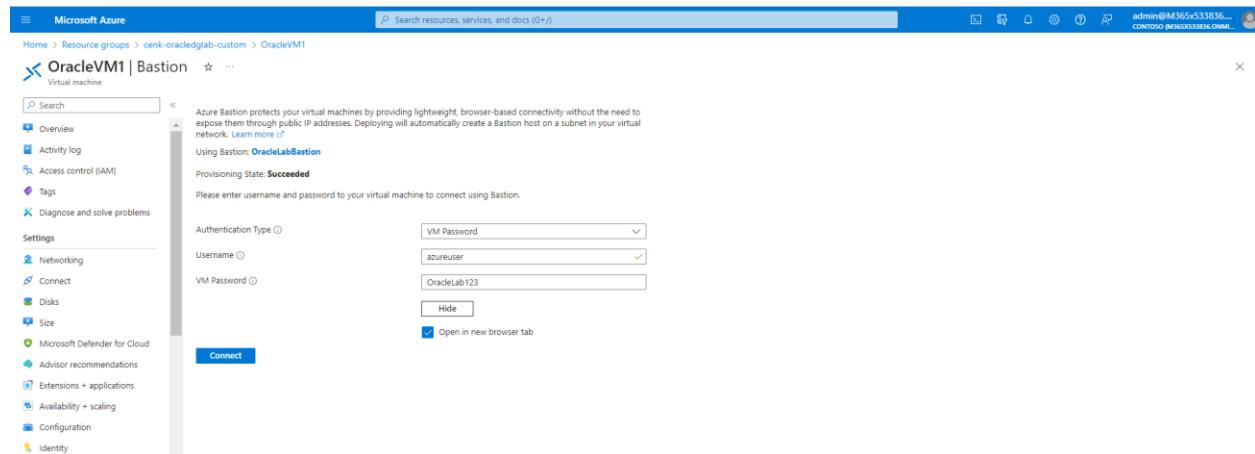
```



Configuring Primary

Disable Firewalls

Login as “azureuser” via Bastion.



```
sudo systemctl stop firewalld  
sudo systemctl disable firewalld
```

```
[azureuser@OracleVM1 ~]$  
[azureuser@OracleVM1 ~]$ ls -l  
total 0  
[azureuser@OracleVM1 ~]$ sudo systemctl stop firewalld  
We trust you have received the usual lecture from the local System  
Administrator. It usually boils down to these three things:  
#1) Respect the privacy of others.  
#2) Think before you type.  
#3) With great power comes great responsibility.  
[sudo] password for azureuser:  
[azureuser@OracleVM1 ~]$ sudo systemctl disable firewalld  
Removed symlink /etc/systemd/system/multi-user.target.wants/firewalld.service.  
Removed symlink /etc/systemd/system/dbus-org.fedoraproject.FirewallD1.service.  
[azureuser@OracleVM1 ~]$ █
```

Set Oracle user password

```
sudo passwd oracle
```

```
[azureuser@OracleVM1 ~]$ sudo passwd oracle  
Changing password for user oracle.  
New password:  
██████████ PASSWORD: The password contains the user name in some form  
Retype new password:  
passwd: all authentication tokens updated successfully.  
[azureuser@OracleVM1 ~]$ █
```

Create DB Instance

We will have a database called "dubai" with "global_dbname = dubai.localdomain", "sid=dubai".

Since global_dname is a combination of "db_name" and "db_domain", we will see that "db_name" is set to "dubai" and "db_domain" is set to "localdomain".

The "dbca" command may take up to 30-35 mins. Remember to refresh the Azure Cloud Shell window.

```
sudo su - oracle
```

```
dbca -silent -createDatabase -datafileDestination /u01/app/oracle/oradata -  
templateName General_Purpose.dbc -gdbName dubai.localdomain -sid dubai -  
responseFile NO_VALUE -characterSet AL32UTF8 -sysPassword OracleLab123 -
```

```
systemPassword OracleLab123 -createAsContainerDatabase true -numberOfPDBs 1 -  
pdbName pdb1 -pdbAdminPassword OracleLab123 -databaseType MULTIPURPOSE -  
automaticMemoryManagement false -storageType FS
```

```
[oracle@OracleVM1 ~]$ dbca -silent -createDatabase -datafileDestination /u01/app/oracle/oradata -templateName General_Purpose.dbc -gdbName dubai.localdomain  
-sid dubai -responseFile NO_VALUE -characterSet AL32UTF8 -sysPassword OracleLab123 -systemPassword OracleLab123 -createAsContainerDatabase true -numberOfPDBs  
1 -pdbName pdb1 -pdbAdminPassword OracleLab123 -databaseType MULTIPURPOSE -automaticMemoryManagement false -storageType FS  
Prepare for db operation  
8% complete  
Copying database files  
31% complete  
Creating and starting Oracle instance  
32% complete  
36% complete  
40% complete  
43% complete  
46% complete  
Completing Database Creation  
51% complete  
53% complete  
54% complete  
Creating Pluggable Databases  
58% complete  
77% complete  
Executing Post Configuration Actions  
100% complete  
Database creation complete. For details check the logfiles at:  
/u01/app/oracle/cfgtoollogs/dbca/dubai.  
Database Information:  
Global Database Name:dubai.localdomain  
System Identifier(SID):dubai  
Look at the log file "/u01/app/oracle/cfgtoollogs/dbca/dubai/dubai.log" for further details.  
[oracle@OracleVM1 ~]$  
[oracle@OracleVM1 ~]$  
[oracle@OracleVM1 ~]$  
[oracle@OracleVM1 ~]$
```

Update .bashrc

```
ORACLE_HOME=/u01/app/oracle/product/19.0.0/dbhome_1; export ORACLE_HOME  
ORACLE_SID=dubai; export ORACLE_SID
```

```
[oracle@OracleVM1 dubai]$ ORACLE_HOME=/u01/app/oracle/product/19.0.0/dbhome_1; export ORACLE_HOME  
[oracle@OracleVM1 dubai]$ ORACLE_SID=dubai; export ORACLE_SID  
[oracle@OracleVM1 dubai]$
```

```
echo "export ORACLE_HOME=/u01/app/oracle/product/19.0.0/dbhome_1" >> ~/.bashrc
```

```
echo "export ORACLE_SID=dubai" >> ~/.bashrc
```

```
[oracle@OracleVM1 dubai]$ echo "export ORACLE_HOME=/u01/app/oracle/product/19.0.0/dbhome_1" >> ~/.bashrc  
[oracle@OracleVM1 dubai]$ echo "export ORACLE_SID=dubai" >> ~/.bashrc  
[oracle@OracleVM1 dubai]$
```

```
more ~/.bashrc
```

```
[oracle@OracleVM1 dubai]$ more ~/.bashrc
# .bashrc

# Source global definitions
if [ -f /etc/bashrc ]; then
    . /etc/bashrc
fi

# Uncomment the following line if you don't like systemctl's auto-paging feature:
# export SYSTEMD_PAGER=

# User specific aliases and functions
export ORACLE_HOME=/u01/app/oracle/product/19.0.0/dbhome_1
export TNS_ADMIN=$ORACLE_HOME/network/admin
export PATH=/usr/local/bin:/bin:/usr/local/sbin:/usr/sbin:/home/oracle/.local/bin:/home/oracle/bin:$ORACLE_HOME/bin
export ORACLE_HOME=/u01/app/oracle/product/19.0.0/dbhome_1
export ORACLE SID=dubai
[oracle@OracleVM1 dubai]$
```

Let us check the SPFILE location:

```
cd /u01/app/oracle/product/19.0.0/dbhome_1/dbs/
ls
[oracle@OracleVM1 ~]$ cd /u01/app/oracle/product/19.0.0/dbhome_1/dbs/
[oracle@OracleVM1 dbs]$ ls
nc_dubai.dat init.ora lkDUBAI orapwdubai spfiledubai.ora
[oracle@OracleVM1 dbs]$
```

(Before enabling archivelog mode, it would be better to create the listener.ora and tnsnames.ora files to ensure proper networking).

At this time, there is no listener.ora or tnsnames.ora file:

```
cd $ORACLE_HOME
find . -name "*.ora"
[oracle@OracleVM1 dbs]$ cd $ORACLE_HOME
[oracle@OracleVM1 dbhome_1]$ find . -name "*.ora"
./inventory/Templates/drdaas/admin/drdaas.ora
./hs/admin/initdg4odbc.ora
./hs/admin/extproc.ora
./srvm/admin/init.ora
./mgw/admin/sample_mgw.ora
./network/admin/samples/listener.ora
./network/admin/samples/sqlnet.ora
./network/admin/samples/tnsnames.ora
./drdaas/admin/drdaas.ora
./rdbms/admin/externaljob.ora
./rdbms/install/filemap.ora
./dbs/init.ora
./dbs/spfiledubai.ora
./env.ora
[oracle@OracleVM1 dbhome_1]$
```

Since the DB instance is already running, we can connect to it:

```
sqlplus / as sysdba
[oracle@OracleVM1 dbhome_1]$ sqlplus / as sysdba

SQL*Plus: Release 19.0.0.0.0 - Production on Tue Nov 14 11:44:06 2023
Version 19.3.0.0.0

Copyright (c) 1982, 2019, Oracle. All rights reserved.

Connected to:
Oracle Database 19c Enterprise Edition Release 19.0.0.0.0 - Production
Version 19.3.0.0.0

SQL> █
```

Configure Listener

Let us configure the listener. At this time, there is no listener.ora file.

```
ls $ORACLE_HOME/network/admin
```

```
[oracle@OracleVM1 ~]$ ls $ORACLE_HOME/network/admin
samples shrept.lst
[oracle@OracleVM1 ~]$ █
```

The database is operational by default. We will start the listener.

```
lsnrctl status
[oracle@OracleVM1 dbhome_1]$ lsnrctl status

LSNRCTL for Linux: Version 19.0.0.0.0 - Production on 15-NOV-2023 21:23:35

Copyright (c) 1991, 2019, Oracle. All rights reserved.

Connecting to (ADDRESS=(PROTOCOL=tcp)(HOST=)(PORT=1521))
TNS-12541: TNS:no listener
TNS-12560: TNS:protocol adapter error
  TNS-00511: No listener
    Linux Error: 111: Connection refused
[oracle@OracleVM1 dbhome_1]$ █
```

```
lsnrctl start
```

```
[oracle@OracleVM1 dbhome_1]$ lsnrctl start
LSNRCTL for Linux: Version 19.0.0.0.0 - Production on 15-NOV-2023 21:24:09
Copyright (c) 1991, 2019, Oracle. All rights reserved.

Starting /u01/app/oracle/product/19.0.0/dbhome_1/bin/tnslsnr: please wait...

TNSLSNR for Linux: Version 19.0.0.0.0 - Production
Log messages written to /u01/app/oracle/diag/tnslsnr/OracleVM1/listener/alert/log.xml
Listening on: (DESCRIPTION=(ADDRESS=(PROTOCOL=tcp)(HOST=OracleVM1.s1m4bs3lylpudmep1kj11hlqod.bx.internal.cloudapp.net)(PORT=1521)))

Connecting to (ADDRESS=(PROTOCOL=tcp)(HOST=)(PORT=1521))
STATUS of the LISTENER
-----
Alias                      LISTENER
Version                    TNSLSNR for Linux: Version 19.0.0.0.0 - Production
Start Date                 15-NOV-2023 21:24:12
Uptime                     0 days 0 hr. 0 min. 2 sec
Trace Level                off
Security                   ON: Local OS Authentication
SNMP                       OFF
Listener Log File          /u01/app/oracle/diag/tnslsnr/OracleVM1/listener/alert/log.xml
Listening Endpoints Summary...
  (DESCRIPTION=(ADDRESS=(PROTOCOL=tcp)(HOST=OracleVM1.s1m4bs3lylpudmep1kj11hlqod.bx.internal.cloudapp.net)(PORT=1521)))
The listener supports no services
The command completed successfully
[oracle@OracleVM1 dbhome_1]$
```

We have to wait for about 2 mins. Let us check the default listener service names:

```
lsnrctl services
[oracle@OracleVM1 dbhome_1]$ lsnrctl services
LSNRCTL for Linux: Version 19.0.0.0.0 - Production on 15-NOV-2023 21:24:54
Copyright (c) 1991, 2019, Oracle. All rights reserved.

Connecting to (ADDRESS=(PROTOCOL=tcp)(HOST=)(PORT=1521))
Services Summary...
Service "0a38712998ea5e54e0630401000a9543.localdomain" has 1 instance(s).
  Instance "dubai", status READY, has 1 handler(s) for this service...
    Handler(s):
      "DEDICATED" established:0 refused:0 state:ready
        LOCAL SERVER
Service "866b637b62fdf7a65e053f706e80a27ca.localdomain" has 1 instance(s).
  Instance "dubai", status READY, has 1 handler(s) for this service...
    Handler(s):
      "DEDICATED" established:0 refused:0 state:ready
        LOCAL SERVER
Service "dubai.localdomain" has 1 instance(s).
  Instance "dubai", status READY, has 1 handler(s) for this service...
    Handler(s):
      "DEDICATED" established:0 refused:0 state:ready
        LOCAL SERVER
Service "dubaIXDB.localdomain" has 1 instance(s).
  Instance "dubai", status READY, has 1 handler(s) for this service...
    Handler(s):
      "D000" established:0 refused:0 current:0 max:1022 state:ready
        DISPATCHER <machine: OracleVM1, pid: 24108>
        (ADDRESS=(PROTOCOL=tcp)(HOST=OracleVM1.s1m4bs3lylpudmep1kj11hlqod.bx.internal.cloudapp.net)(PORT=21859))
Service "pdb1.localdomain" has 1 instance(s).
  Instance "dubai", status READY, has 1 handler(s) for this service...
    Handler(s):
      "DEDICATED" established:0 refused:0 state:ready
        LOCAL SERVER
The command completed successfully
[oracle@OracleVM1 dbhome_1]$
```

We see the following service names that have dynamically registered themselves:

- dubai.localdomain (same as global_dbname)
- dubaiXDB.localdomain (<SID>XDB.localdomain)
- pdb1.localdomain

Here is the “SERVICE_NAMES” parameter:

```
sqlplus / as sysdba
SQL> show parameter SERVICE_NAMES
SQL> exit
[oracle@OracleVM1 dbhome_1]$ sqlplus / as sysdba

SQL*Plus: Release 19.0.0.0.0 - Production on Wed Nov 15 21:26:08 2023
Version 19.3.0.0.0

Copyright (c) 1982, 2019, Oracle. All rights reserved.

Connected to:
Oracle Database 19c Enterprise Edition Release 19.0.0.0.0 - Production
Version 19.3.0.0.0

SQL> show parameter SERVICE_NAMES

NAME                           TYPE        VALUE
-----                         string      dubai.localdomain
SQL> █
»
```

AS OF ORACLE 19C, the "SERVICE_NAMES" DATABASE PARAMETER IS DEPRECATED.

Now, we will create a static listener service by creating a \$ORACLE_HOME/network/admin/listener.ora file.

The “GLOBAL_DBNAME” in listener.ora file MUST match the “SERVICE_NAME” parameter in tnsnames.ora. We will create a static listener service called “dubaiserv”.

```
vi $ORACLE_HOME/network/admin/listener.ora

LISTENER =
  (DESCRIPTION_LIST =
    (DESCRIPTION =
      (ADDRESS = (PROTOCOL = TCP) (HOST = OracleVM1) (PORT = 1521))
      (ADDRESS = (PROTOCOL = IPC) (KEY = EXTPROC1521)))
    )
  )
SID_LIST_LISTENER =
  (SID_LIST =
    (SID_DESC =
      (GLOBAL_DBNAME = dubaiserv)
      (ORACLE_HOME = /u01/app/oracle/product/19.0.0/dbhome_1)
      (SID_NAME = dubai)
    )
  )
```

```
)  
ADR_BASE_LISTENER = /u01/app/oracle  
[oracle@OracleVM1 dbhome_1]$ more $ORACLE_HOME/network/admin/listener.ora  
LISTENER =  
(DESCRIPTION_LIST =  
  (DESCRIPTION =  
    (ADDRESS = (PROTOCOL = TCP)(HOST = OracleVM1)(PORT = 1521))  
    (ADDRESS = (PROTOCOL = IPC)(KEY = EXTPROC1521))  
  )  
)  
SID_LIST_LISTENER =  
(SID_LIST =  
  (SID_DESC =  
    (GLOBAL_DBNAME = dubaiserv)  
    (ORACLE_HOME = /u01/app/oracle/product/19.0.0/dbhome_1)  
    (SID_NAME = dubai)  
  )  
)  
ADR_BASE_LISTENER = /u01/app/oracle  
[oracle@OracleVM1 dbhome_1]$ █
```

Let us restart the listener:

```
lsnrctl stop  
[oracle@OracleVM1 dbhome_1]$ lsnrctl stop  
LSNRCTL for Linux: Version 19.0.0.0.0 - Production on 15-NOV-2023 21:28:32  
Copyright (c) 1991, 2019, Oracle. All rights reserved.  
Connecting to (DESCRIPTION=(ADDRESS=(PROTOCOL=TCP)(HOST=OracleVM1)(PORT=1521)))  
The command completed successfully  
[oracle@OracleVM1 dbhome_1]$ █
```

```
lsnrctl start
```

```
[oracle@OracleVM1 dbhome_1]$ lsnrctl start
LSNRCTL for Linux: Version 19.0.0.0.0 - Production on 15-NOV-2023 21:29:03
Copyright (c) 1991, 2019, Oracle. All rights reserved.

Starting /u01/app/oracle/product/19.0.0/dbhome_1/bin/tnslsnr: please wait...

TNSLSNR for Linux: Version 19.0.0.0.0 - Production
System parameter file is /u01/app/oracle/product/19.0.0/dbhome_1/network/admin/listener.ora
Log messages written to /u01/app/oracle/diag/tnslsnr/OracleVM1/listener/alert/log.xml
Listening on: (DESCRIPTION=(ADDRESS=(PROTOCOL=tcp)(HOST=OracleVM1.s1m4bs3lylpudmep1kj11hlqod.bx.internal.cloudapp.net)(PORT=1521)))
Listening on: (DESCRIPTION=(ADDRESS=(PROTOCOL=ipc)(KEY=EXTPROC1521)))

Connecting to (DESCRIPTION=(ADDRESS=(PROTOCOL=TCP)(HOST=OracleVM1)(PORT=1521)))
STATUS of the LISTENER
-----
Alias           LISTENER
Version        TNSLSNR for Linux: Version 19.0.0.0.0 - Production
Start Date     15-NOV-2023 21:29:03
Uptime         0 days 0 hr. 0 min. 0 sec
Trace Level    off
Security       ON: Local OS Authentication
SNMP           OFF
Listener Parameter File /u01/app/oracle/product/19.0.0/dbhome_1/network/admin/listener.ora
Listener Log File /u01/app/oracle/diag/tnslsnr/OracleVM1/listener/alert/log.xml
Listening Endpoints Summary...
  (DESCRIPTION=(ADDRESS=(PROTOCOL=tcp)(HOST=OracleVM1.s1m4bs3lylpudmep1kj11hlqod.bx.internal.cloudapp.net)(PORT=1521)))
  (DESCRIPTION=(ADDRESS=(PROTOCOL=ipc)(KEY=EXTPROC1521)))
Services Summary...
Service "dubaiserv" has 1 instance(s).
  Instance "dubai", status UNKNOWN, has 1 handler(s) for this service...
The command completed successfully
[oracle@OracleVM1 dbhome_1]$
```

Again, we wait for 2 minutes. Then let us list the listeners:

```
lsnrctl services
Copyright (c) 1991, 2019, Oracle. All rights reserved.

Connecting to (DESCRIPTION=(ADDRESS=(PROTOCOL=TCP)(HOST=OracleVM1)(PORT=1521)))
Services Summary...
Service "0a38712998ea5e54e0630401000a9543.localdomain" has 1 instance(s).
  Instance "dubai", status READY, has 1 handler(s) for this service...
    Handler(s):
      "DEDICATED" established:0 refused:0 state:ready
        LOCAL SERVER
Service "86b637b62fdf7a65e053f706e80a27ca.localdomain" has 1 instance(s).
  Instance "dubai", status READY, has 1 handler(s) for this service...
    Handler(s):
      "DEDICATED" established:0 refused:0 state:ready
        LOCAL SERVER
Service "dubai.localdomain" has 1 instance(s).
  Instance "dubai", status READY, has 1 handler(s) for this service...
    Handler(s):
      "DEDICATED" established:0 refused:0 state:ready
        LOCAL SERVER
Service "dubaiXDB.localdomain" has 1 instance(s).
  Instance "dubai", status READY, has 1 handler(s) for this service...
    Handler(s):
      "D000" established:0 refused:0 current:0 max:1022 state:ready
        DISPATCHER <machine: OracleVM1, pid: 24108>
          (ADDRESS=(PROTOCOL=tcp)(HOST=OracleVM1.s1m4bs3lylpudmep1kj11hlqod.bx.internal.cloudapp.net)(PORT=21859))
Service "dubaiserv" has 1 instance(s.) [REDACTED]
  Instance "dubai", status UNKNOWN, has 1 handler(s) for this service...
    Handler(s):
      "DEDICATED" established:0 refused:0
        LOCAL SERVER
Service "pdb1.localdomain" has 1 instance(s).
  Instance "dubai", status READY, has 1 handler(s) for this service...
    Handler(s):
      "DEDICATED" established:0 refused:0 state:ready
        LOCAL SERVER
The command completed successfully
[oracle@OracleVM1 dbhome_1]$
```

Now, we see that “dubaiserv” service is created. It has “status=UNKNOWN” which is fine.

We can test the listener service by using Easy Connect format:

```
sqlplus sys/OracleLab123@OracleVM1:1521/dubaiserv as sysdba
[oracle@OracleVM1 oracle]$ sqlplus sys/OracleLab123@OracleVM1:1521/dubaiserv as sysdba

SQL*Plus: Release 19.0.0.0.0 - Production on Wed Nov 15 22:41:43 2023
Version 19.3.0.0.0

Copyright (c) 1982, 2019, Oracle. All rights reserved.

Connected to:
Oracle Database 19c Enterprise Edition Release 19.0.0.0.0 - Production
Version 19.3.0.0.0

SQL> 
```

We will use the listener instance named “dubaiserv” in the tnsnames.ora and set “SERVICE_NAME=dubaiserv”

Configure Name Resolution

Now, it is time to modify tnsnames.ora file. At this time, there are no tnsnames.ora file:

```
ls $ORACLE_HOME/network/admin
```

```
[oracle@OracleVM1 dbhome_1]$ ls $ORACLE_HOME/network/admin
listener.ora  samples  shrept.lst
[oracle@OracleVM1 dbhome_1]$ 
```

```
vi $ORACLE_HOME/network/admin/tnsnames.ora

DXB =
(DESCRIPTION =
(ADDRESS_LIST =
(ADDRESS = (PROTOCOL = TCP)(HOST = OracleVM1)(PORT = 1521))
)
(CONNECT_DATA =
(SID = dubai)
(SERVICE_NAME=dubaiserv)
)
)

DXBSTB =
(DESCRIPTION =
(ADDRESS_LIST =
```

```

        (ADDRESS = (PROTOCOL = TCP)(HOST = OracleVM2)(PORT = 1521))
    )
  (CONNECT_DATA =
    (SID = dubaistb)
    (SERVICE_NAME=dubaistbserv)
  )
)
)

[oracle@OracleVM1 dbhome_1]$ more $ORACLE_HOME/network/admin/tnsnames.ora
DXB =
(DESCRIPTION =
  (ADDRESS_LIST =
    (ADDRESS = (PROTOCOL = TCP)(HOST = OracleVM1)(PORT = 1521))
  )
  (CONNECT_DATA =
    (SID = dubai)
    (SERVICE_NAME=dubaiserv)
  )
)

DXBSTB =
(DESCRIPTION =
  (ADDRESS_LIST =
    (ADDRESS = (PROTOCOL = TCP)(HOST = OracleVM2)(PORT = 1521))
  )
  (CONNECT_DATA =
    (SID = dubaistb)
  >>    (SERVICE_NAME=dubaistbserv)
  )
)

[oracle@OracleVM1 dbhome_1]$ ■

```

Let us test the name resolution:

```

tnsping DXB
[oracle@OracleVM1 dbhome_1]$ tnsping DXB
TNS Ping Utility for Linux: Version 19.0.0.0.0 - Production on 15-NOV-2023 21:35:22
Copyright (c) 1997, 2019, Oracle. All rights reserved.
Used parameter files:

Used TNSNAMES adapter to resolve the alias
Attempting to contact (DESCRIPTION = (ADDRESS_LIST = (ADDRESS = (PROTOCOL = TCP)(HOST = OracleVM1)(PORT = 1521))) (CONNECT_DATA = (SID = dubai) (SERVICE_NAME
=dubaiserv)))
OK (10 msec)
[oracle@OracleVM1 dbhome_1]$ ■

```

At this time, there is no listener on the standby server. It is not reachable.

```
tnsping DXBSTB
```

```
[oracle@OracleVM1 dbhome_1]$ tnsping DXBSTB
TNS Ping Utility for Linux: Version 19.0.0.0.0 - Production on 15-NOV-2023 21:36:05
Copyright (c) 1997, 2019, Oracle. All rights reserved.

Used parameter files:

Used TNSNAMES adapter to resolve the alias
Attempting to contact (DESCRIPTION = (ADDRESS = (PROTOCOL = TCP)(HOST = OracleVM2)(PORT = 1521)) (CONNECT_DATA = (SID = dubaistb) (SERVICE_NAME=dubaistbserv)))
TNS-12541: TNS:no listener
[oracle@OracleVM1 dbhome_1]$
```

Optionally, check the sqlnet.ora file:

```
more $ORACLE_HOME/network/admin/sqlnet.ora
[oracle@OracleVM1 ~]$ more $ORACLE_HOME/network/admin/sqlnet.ora
/u01/app/oracle/product/19.0.0/dbhome_1/network/admin/sqlnet.ora: No such file or directory
[oracle@OracleVM1 ~]$
```

Even though Azure handles DNS, let us modify the /etc/hosts file:

```
ifconfig
[oracle@OracleVM1 ~]$ ifconfig
eth0: flags=4163<UP,BROADCAST,RUNNING,MULTICAST> mtu 1500
    inet 10.0.1.4 netmask 255.255.255.0 broadcast 10.0.1.255
        inet6 fe80::20d:3aff:fe58:524 prefixlen 64 scopeid 0x20<link>
            ether 00:0d:3a:58:05:24 txqueuelen 1000 (Ethernet)
            RX packets 426744 bytes 561524759 (535.5 MiB)
            RX errors 0 dropped 0 overruns 0 frame 0
            TX packets 58890 bytes 12496971 (11.9 MiB)
            TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

lo: flags=73<UP,LOOPBACK,RUNNING> mtu 65536
    inet 127.0.0.1 netmask 255.0.0.0
        inet6 ::1 prefixlen 128 scopeid 0x10<host>
            loop txqueuelen 1000 (Local Loopback)
            RX packets 976 bytes 61197 (59.7 KiB)
            RX errors 0 dropped 0 overruns 0 frame 0
            TX packets 976 bytes 61197 (59.7 KiB)
            TX errors 0 dropped 0 overruns 0 carrier 0 collisions 0

[oracle@OracleVM1 ~]$
```

```
hostname
[oracle@OracleVM1 ~]$ hostname
OracleVM1
[oracle@OracleVM1 ~]$
```

Before editing /etc/hosts we must go back to “azureuser”

```
exit
```

```
sudo vi /etc/hosts

[oracle@OracleVM1 ~]$ exit
logout
[azureuser@OracleVM1 ~]$ sudo vi /etc/hosts

127.0.0.1 localhost localhost.localdomain localhost4 localhost4.localdomain4
::1 localhost localhost.localdomain localhost6 localhost6.localdomain6
10.0.1.4 OracleVM1 OracleVM1.localdomain
```

```
sudo su - oracle
```

Check Password File

Let us check the presence of the password file:

```
ls $ORACLE_HOME/dbs/orapw*
[oracle@OracleVM1 dbs]$ ls $ORACLE_HOME/dbs/orapw*
/u01/app/oracle/product/19.0.0/dbhome_1/dbs/orapwchicago
[oracle@OracleVM1 dbs]$
```

Enable ArchiveLog Mode

Next, we will enable archivelog mode.

```
sqlplus / as sysdba
SQL> SELECT log_mode FROM v$database;
```

```
[oracle@OracleVM1 ~]$ sqlplus / as sysdba

SQL*Plus: Release 19.0.0.0.0 - Production on Wed Nov 15 21:42:05 2023
Version 19.3.0.0.0

Copyright (c) 1982, 2019, Oracle. All rights reserved.

Connected to:
Oracle Database 19c Enterprise Edition Release 19.0.0.0.0 - Production
Version 19.3.0.0.0

SQL> SELECT log_mode FROM v$database;

LOG_MODE
-----
NOARCHIVELOG

SQL> █
```

```
SQL> SHUTDOWN IMMEDIATE;  
SQL> STARTUP MOUNT;  
SQL> ALTER DATABASE ARCHIVELOG;  
SQL> ALTER DATABASE OPEN;
```

```
SQL> SHUTDOWN IMMEDIATE;  
ORA-01109: database not open
```

```
Database dismounted.  
ORACLE instance shut down.  
SQL> STARTUP MOUNT;  
ORACLE instance started.
```

```
Total System Global Area  998244240  bytes
Fixed Size                9142160  bytes
Variable Size            260046848  bytes
Database Buffers        725614592  bytes
Redo Buffers              3440640  bytes
Database mounted.
SQL> ALTER DATABASE ARCHIVELOG;
```

Database altered.

|SQL> ALTER DATABASE OPEN;

Database altered.

SQL>

Check DB Instance Configuration

Let us check the basic configuration of the primary database:

```
sqlplus / as sysdba
SQL> select host_name from v$instance;
SQL> select host_name from v$instance;
HOST_NAME
-----
OracleVM1
SQL> ■
```

```
SQL> select open_mode from v$database;
SQL> select open_mode from v$database;
OPEN_MODE
-----
READ WRITE
SQL> ■
```

```
SQL> select database_role from v$database;
SQL> select database_role from v$database;
DATABASE_ROLE
-----
PRIMARY
SQL> ■
```

```
SQL> show parameter DB_NAME
SQL> show parameter DB_NAME
NAME                           TYPE      VALUE
-----                         string    dubai
```

```
SQL> show parameter DB_DOMAIN
```

```
SQL> show parameter DB_DOMAIN
NAME                      TYPE        VALUE
-----
db_domain                 string      localdomain
SQL> █
```

```
SQL> show parameter DB_UNIQUE_NAME
SQL> show parameter db_unique_name
NAME                      TYPE        VALUE
-----
db_unique_name             string      dubai
SQL> █
```

```
SQL> select name, value from v$parameter where name='db_unique_name';
```

```
SQL> select name, value from v$parameter where name='db_unique_name';
NAME
-----
VALUE
-----
db_unique_name             dubai
SQL> █
```

```
SQL> show parameter INSTANCE_NAME;
```

```
SQL> show parameter INSTANCE_NAME;
NAME                      TYPE        VALUE
-----
instance_name              string      dubai
SQL> █
```

SID is set to “dubai”:

```
SQL> select instance from v$thread;
```

```
SQL> select instance from v$thread;
INSTANCE
-----
dubai
SQL> █
```

```
SQL > select name, value from v$parameter where name='instance_name';
SQL> select name, value from v$parameter where name='instance_name';

NAME
-----
VALUE
-----
instance_name
dubai

SQL> █
```

The DB_NAME is “DUBAI” (which is <SID>):

```
SQL> select name from v$database;

SQL> select name from v$database;

NAME
-----
DUBAI

SQL> █
```

```
SQL> select name,value from v$parameter where name='db_name';
```

```
SQL> select name,value from v$parameter where name='db_name';

NAME
-----
VALUE
-----
db_name
dubai

SQL> █
```

```
SQL> show parameter DB_BLOCK_SIZE
SQL> show parameter DB_CACHE_SIZE
```

```
SQL> show parameter DB_BLOCK_SIZE
NAME                                     TYPE        VALUE
-----
db_block_size                           integer     8192
SQL> show parameter DB_CACHE_SIZE
NAME                                     TYPE        VALUE
-----
db_cache_size                           big integer 0
```

```
SQL> show parameter CONTROL_FILES
SQL> show parameter CONTROL_FILES
NAME                                     TYPE        VALUE
-----
control_files                           string      /u01/app/oracle/oradata/DUBAI/
                                         control01.ctl, /u01/app/oracle/
                                         /oradata/DUBAI/control02.ctl
SQL> ■
```

```
SQL> show parameter DB_CREATE_FILE_DEST  (location of datafiles)
SQL> show parameter DB_CREATE_ONLINE_LOG_DEST_1  (location of online redo logs)
SQL> show parameter LOG_ARCHIVE_DEST_1  (location of archive redo logs)
SQL> show parameter DB_RECOVERY_FILE_DEST  (Flash Recovery Area)
SQL> show parameter UNDO_TABLESPACE  (location of undo tablespace)
```

```

SQL> show parameter DB_CREATE_FILE_DEST
NAME                      TYPE        VALUE
-----
db_create_file_dest        string
SQL> show parameter DB_CREATE_ONLINE_LOG_DEST_1
NAME                      TYPE        VALUE
-----
db_create_online_log_dest_1    string
SQL> show parameter LOG_ARCHIVE_DEST_1
NAME                      TYPE        VALUE
-----
log_archive_dest_1          string
log_archive_dest_10         string
log_archive_dest_11         string
log_archive_dest_12         string
log_archive_dest_13         string
log_archive_dest_14         string
log_archive_dest_15         string
log_archive_dest_16         string
log_archive_dest_17         string
log_archive_dest_18         string
log_archive_dest_19         string
SQL> show parameter DB_RECOVERY_FILE_DEST
NAME                      TYPE        VALUE
-----
db_recovery_file_dest       string
db_recovery_file_dest_size  big integer 0
SQL> show parameter UNDO_TABLESPACE
NAME                      TYPE        VALUE
-----
undo_tablespace            string      UNDOTBS1
SQL> ■

```

Since “DB_CREATE_FILE_DEST” was blank, let’s check the actual location of the datafiles.

```

SQL> select name from v$datafile;
SQL> select name from v$datafile;
>>
NAME
-----
/u01/app/oracle/oradata/DUBAI/system01.dbf
/u01/app/oracle/oradata/DUBAI/sysaux01.dbf
/u01/app/oracle/oradata/DUBAI/undotbs01.dbf
/u01/app/oracle/oradata/DUBAI/pdbseed/system01.dbf
/u01/app/oracle/oradata/DUBAI/pdbseed/sysaux01.dbf
/u01/app/oracle/oradata/DUBAI/users01.dbf
/u01/app/oracle/oradata/DUBAI/pdbseed/undotbs01.dbf
/u01/app/oracle/oradata/DUBAI/pdb1/system01.dbf
/u01/app/oracle/oradata/DUBAI/pdb1/sysaux01.dbf
/u01/app/oracle/oradata/DUBAI/pdb1/undotbs01.dbf
/u01/app/oracle/oradata/DUBAI/pdb1/users01.dbf

11 rows selected.

SQL> ■

```

The database files are in /u01/app/oracle/oradata/<SID>/ as specified when running DBCA.

Since DB_CREATE_ONLINE_LOG_DEST_1 was blank, let us check the actual location of the online redo logs:

```
SQL> col status format a9
SQL> col member format a55
SQL> select g.group#, g.status, f.member
   from v$log g, v$logfile f
  where g.group# = f.group#;
SQL> col status format a9
SQL> col member format a55
SQL> select g.group#, g.status, f.member
   from v$log g, v$logfile f
  where g.group# = f.group#;
2   3
 GROUP# STATUS      MEMBER
-----
 3 INACTIVE  /u01/app/oracle/oradata/DUBAI/redo03.log
 2 INACTIVE  /u01/app/oracle/oradata/DUBAI/redo02.log
 1 CURRENT   /u01/app/oracle/oradata/DUBAI/redo01.log
SQL> █
```

IT SEEMS THAT BY DEFAULT THE ONLINE REDO LOG FILES ARE IN THE SAME FOLDER AS THE DATA FILES.
THIS IS NOT IDEAL.

Similarly, LOG_ARCHIVE_DEST_1 was also blank. Here is the actual location of the archive redo logs:

```
SQL> select dest_name, status, destination from v$archive_dest where
dest_name='LOG_ARCHIVE_DEST_1';
SQL> select dest_name, status, destination from v$archive_dest where dest_name='LOG_ARCHIVE_DEST_1';
DEST_NAME
-----
STATUS
-----
DESTINATION
-----
LOG_ARCHIVE_DEST_1
VALID
/u01/app/oracle/product/19.0.0/dbhome_1/dbs/arch
SQL> █
```

At this time, the archive log location is \$ORACLE_HOME/dbs/arch. Later we will modify it when enabling flashback.

```
SQL> ARCHIVE LOG LIST
```

```
SQL> exit
SQL> ARCHIVE LOG LIST
Database log mode          Archive Mode
Automatic archival        Enabled
Archive destination        /u01/app/oracle/product/19.0.0/dbhome_1/dbs/arch
Oldest online log sequence 5
Next log sequence to archive 7
Current log sequence       7
SQL> █
```

```
ls -l $ORACLE_HOME/dbs
[oracle@OracleVM1 ~]$ ls -l $ORACLE_HOME/dbs
total 20
-rw-rw----. 1 oracle oinstall 1544 Nov 14 12:06 hc_dubai.dat
-rw-r--r--. 1 oracle oinstall 3079 May 14 2015 init.ora
-rw-r-----. 1 oracle oinstall    24 Nov 14 10:49 lkDUBAI
-rw-r-----. 1 oracle oinstall 2048 Nov 14 10:52 orapwdubai
-rw-r-----. 1 oracle oinstall 3584 Nov 14 12:06 spfiledubai.ora
[oracle@OracleVM1 ~]$ █
```

```
sqlplus as sysdba
SQL> show parameter SGA_TARGET (size of SGA)
SQL> show parameter PGA_AGGREGATE_TARGET (size of PGA)
SQL> show parameter PROCESSES (max no of processes)
SQL> show parameter SESSIONS (max no of sessions)
```

```

SQL> show parameter SGA_TARGET
NAME                      TYPE        VALUE
-----
sga_target                big integer 952M
SQL> show parameter PGA_AGGREGATE_TARGET
NAME                      TYPE        VALUE
-----
pga_aggregate_target      big integer 317M
SQL> show parameter PROCESSES
NAME                      TYPE        VALUE
-----
aq_tm_processes           integer    1
db_writer_processes       integer    1
gcs_server_processes     integer    0
global_txn_processes     integer    1
job_queue_processes       integer   20
log_archive_max_processes integer    4
processes                 integer  300
SQL> show parameter SESSIONS
NAME                      TYPE        VALUE
-----
java_max_sessionspace_size integer    0
java_soft_sessionspace_limit integer    0
license_max_sessions      integer    0
license_sessions_warning  integer    0
sessions                  integer  472
shared_server_sessions    integer
SQL> █

```

```

SQL> show parameter FAL_SERVER
SQL> show parameter FAL_SERVER
NAME                      TYPE        VALUE
-----
fal_server                string
SQL> █

```

FAL_SERVER is not yet defined.

Enable Force Logging

Now, we will enable force logging:

```

SQL> ALTER DATABASE FORCE LOGGING;
SQL> ALTER SYSTEM SWITCH LOGFILE;

```

```
SQL> ALTER DATABASE FORCE LOGGING;  
Database altered.  
  
SQL> ALTER SYSTEM SWITCH LOGFILE;  
System altered.  
  
SQL> █
```

Since we forced “switch”, we can see the new archive redo log:

```
SQL> SELECT NAME FROM V$ARCHIVED_LOG;  
SQL> SELECT NAME FROM V$ARCHIVED_LOG;  
  
NAME  
-----  
/u01/app/oracle/product/19.0.0/dbhome_1/dbs/arch1_7_1152874358.dbf  
  
SQL> █
```

The following is optional:

```
SQL> alter system checkpoint;  
SQL> alter system checkpoint;  
  
System altered.  
  
SQL> █
```

However, later we will see that the new archive redo log files will be created at /u01/app/oracle/<SID>/<SID>/archivelog/ folder!! We will see that the old archive file “arch1_7-1152874358.dbf is also moved there.

Create Standby Redo Logs

Now it is time to create Standby Redo Logs. We should check the actual size of the current Online Redo Logs before creating Standby Redo Logs. The Standby Redo Logs have to be the same size as Online Redo Logs.

```
SQL> SELECT GROUP#, BYTES/1024/1024 MB FROM V$LOG;
```

```
SQL> SELECT GROUP#, BYTES/1024/1024 MB FROM V$LOG;
      GROUP#      MB
----- -----
      1        200
      2        200
      3        200
```

Here is the thread count:

```
SQL> SELECT THREAD#, INSTANCE FROM V$THREAD;
SQL> SELECT THREAD#, INSTANCE FROM V$THREAD;
```

```
      THREAD#
-----
INSTANCE
-----
      1
dubai
```

No of standby redo logs = (no of online redo logs x thread count) + 1

Now we create standby redo logs, setting the same size as the primary database redo logs. We will use the same location as online redo logs:

```
SQL> ALTER DATABASE ADD STANDBY LOGFILE
  ('/u01/app/oracle/oradata/DUBAI/standby_redo01.log') SIZE 200M;
SQL> ALTER DATABASE ADD STANDBY LOGFILE
  ('/u01/app/oracle/oradata/DUBAI/standby_redo02.log') SIZE 200M;
SQL> ALTER DATABASE ADD STANDBY LOGFILE
  ('/u01/app/oracle/oradata/DUBAI/standby_redo03.log') SIZE 200M;
SQL> ALTER DATABASE ADD STANDBY LOGFILE
  ('/u01/app/oracle/oradata/DUBAI/standby_redo04.log') SIZE 200M;
```

```
SQL> ALTER DATABASE ADD STANDBY LOGFILE ('/u01/app/oracle/oradata/DUBAI/standby_redo01.log') SIZE 200M;
Database altered.

SQL> ALTER DATABASE ADD STANDBY LOGFILE ('/u01/app/oracle/oradata/DUBAI/standby_redo02.log') SIZE 200M;
Database altered.

SQL> ALTER DATABASE ADD STANDBY LOGFILE ('/u01/app/oracle/oradata/DUBAI/standby_redo03.log') SIZE 200M;
Database altered.

SQL> ALTER DATABASE ADD STANDBY LOGFILE ('/u01/app/oracle/oradata/DUBAI/standby_redo04.log') SIZE 200M;
Database altered.

SQL> █
```

Note that while we created the Standby Redo Logs on the primary server, they will be copied over to standby server along with the primary database when we do RMAN DUPLICATE.

```
SQL> SELECT GROUP#,THREAD#,SEQUENCE#,ARCHIVED,STATUS FROM V$STANDBY_LOG;
SQL> SELECT GROUP#,THREAD#,SEQUENCE#,ARCHIVED,STATUS FROM V$STANDBY_LOG;
      GROUP#    THREAD#  SEQUENCE# ARC STATUS
-----  -----  -----
        4          0          0 YES UNASSIGNED
        5          0          0 YES UNASSIGNED
        6          0          0 YES UNASSIGNED
        7          0          0 YES UNASSIGNED

SQL> █
```

Let us view both the online redo logs and the standby redo logs.

```
SQL> set lines 100
SQL> col type format a10
SQL> col member format a50
SQL> SELECT TYPE, MEMBER FROM V$LOGFILE;
SQL> exit
```

```

SQL> set lines 100
SQL> col type format a10
SQL> col member format a50
SQL> SELECT TYPE, MEMBER FROM V$LOGFILE;

TYPE      MEMBER
-----
ONLINE    /u01/app/oracle/oradata/DUBAI redo03.log
ONLINE    /u01/app/oracle/oradata/DUBAI redo02.log
ONLINE    /u01/app/oracle/oradata/DUBAI redo01.log
STANDBY   /u01/app/oracle/oradata/DUBAI standby_redo01.log
STANDBY   /u01/app/oracle/oradata/DUBAI standby_redo02.log
STANDBY   /u01/app/oracle/oradata/DUBAI standby_redo03.log
STANDBY   /u01/app/oracle/oradata/DUBAI standby_redo04.log

7 rows selected.

SQL>

```

We see that online redo logs and standby redo logs reside at /u01/app/oradata/DUBAI/. THIS IS NOT IDEAL SINCE THE SAME LOCATION HOLDS THE DATABASE FILES AND CONTROL FILES AS WELL:

```

ls -l /u01/app/oracle/oradata/DUBAI
SQL> exit
Disconnected from Oracle Database 19c Enterprise Edition Release 19.0.0.0.0 - Production
Version 19.3.0.0.0
[oracle@OracleVM1 ~]$ ls -l /u01/app/oracle/oradata/DUBAI
total 3363980
-rw-r----- 1 oracle oinstall 18726912 Nov 15 22:02 control01.ctl
-rw-r----- 1 oracle oinstall 18726912 Nov 15 22:02 control02.ctl
drwxr-x--- 1 oracle oinstall 116 Nov 15 21:18 pdb1
drwxr-x--- 1 oracle oinstall 146 Nov 15 20:53 pdbseed
-rw-r----- 1 oracle oinstall 209715712 Nov 15 21:54 redo01.log
-rw-r----- 1 oracle oinstall 209715712 Nov 15 22:02 redo02.log
-rw-r----- 1 oracle oinstall 209715712 Nov 15 21:44 redo03.log
-rw-r----- 1 oracle oinstall 209715712 Nov 15 21:59 standby_redo01.log
-rw-r----- 1 oracle oinstall 209715712 Nov 15 21:59 standby_redo02.log
-rw-r----- 1 oracle oinstall 209715712 Nov 15 21:59 standby_redo03.log
-rw-r----- 1 oracle oinstall 209715712 Nov 15 21:59 standby_redo04.log
-rw-r----- 1 oracle oinstall 576724992 Nov 15 22:00 sysaux01.dbf
-rw-r----- 1 oracle oinstall 943726592 Nov 15 22:00 system01.dbf
-rw-r----- 1 oracle oinstall 135274496 Nov 15 22:02 temp01.dbf
-rw-r----- 1 oracle oinstall 288366592 Nov 15 22:00 undotbs01.dbf
-rw-r----- 1 oracle oinstall 5251072 Nov 15 21:54 users01.dbf
[oracle@OracleVM1 ~]$ 

```

Set Database Parameters

Next, we need to **set Primary Database Initialization Parameters**.

The parameters exist both in memory and in SPFILE. Since memory copy of certain parameters cannot be modified, we need to use “SCOPE=SPFILE”:

```
sqlplus / as sysdba

SQL> ALTER SYSTEM SET FAL_SERVER='DXBSTB';

SQL> ALTER SYSTEM SET LOG_ARCHIVE_CONFIG='DG_CONFIG=(dubai,dubaistb)';

SQL> ALTER SYSTEM SET LOG_ARCHIVE_DEST_1='LOCATION=USE_DB_RECOVERY_FILE_DEST
VALID_FOR=(ALL_LOGFILES,ALL_ROLES) DB_UNIQUE_NAME=dubai' scope=both;

SQL> ALTER SYSTEM SET LOG_ARCHIVE_DEST_2='SERVICE=DXBSTB ASYNC
VALID_FOR=(ONLINE_LOGFILES,PRIMARY_ROLE) DB_UNIQUE_NAME=dubaistb' scope=both;

SQL> ALTER SYSTEM SET REMOTE_LOGIN_PASSWORDFILE='EXCLUSIVE' SCOPE=SPFILE;

SQL> ALTER SYSTEM SET DB_FILE_NAME_CONVERT='/dubaistb/','/dubai/',
'/DUBAISTB/','/DUBAI/' SCOPE=SPFILE;

SQL> ALTER SYSTEM SET LOG_FILE_NAME_CONVERT='/dubaistb/','/dubai/',
'/DUBAISTB/','/DUBAI/' SCOPE=SPFILE;

SQL> ALTER SYSTEM SET STANDBY_FILE_MANAGEMENT='AUTO';

SQL> ALTER SYSTEM SET LOG_ARCHIVE_FORMAT='%t_%s_%r.arc' SCOPE=SPFILE;

[oracle@OracleVM1 ~]$ sqlplus / as sysdba
SQL*Plus: Release 19.0.0.0 - Production on Wed Nov 15 22:03:25 2023
Version 19.3.0.0.0
Copyright (c) 1982, 2019, Oracle. All rights reserved.

Connected to:
Oracle Database 19c Enterprise Edition Release 19.0.0.0.0 - Production
Version 19.3.0.0.0

SQL> ALTER SYSTEM SET FAL_SERVER='DXBSTB';
System altered.

SQL> ALTER SYSTEM SET LOG_ARCHIVE_CONFIG='DG_CONFIG=(dubai,dubaistb)';
System altered.

SQL> ALTER SYSTEM SET LOG_ARCHIVE_DEST_1='LOCATION=USE_DB_RECOVERY_FILE_DEST VALID_FOR=(ALL_LOGFILES,ALL_ROLES) DB_UNIQUE_NAME=dubai' scope=both;
System altered.

SQL> ALTER SYSTEM SET LOG_ARCHIVE_DEST_2='SERVICE=DXBSTB ASYNC VALID_FOR=(ONLINE_LOGFILES,PRIMARY_ROLE) DB_UNIQUE_NAME=dubaistb' scope=both;
System altered.
```

```
SQL> ALTER SYSTEM SET REMOTE_LOGIN_PASSWORDFILE='EXCLUSIVE' SCOPE=SPFILE;
System altered.
SQL> ALTER SYSTEM SET DB_FILE_NAME_CONVERT='/dubaistb','/dubai/' SCOPE=SPFILE;
System altered.
SQL> ALTER SYSTEM SET LOG_FILE_NAME_CONVERT='/dubaistb','/dubai/' SCOPE=SPFILE;
System altered.
SQL> ALTER SYSTEM SET STANDBY_FILE_MANAGEMENT='AUTO';
System altered.
SQL> ALTER SYSTEM SET LOG_ARCHIVE_FORMAT='%t_%s_%r.arc' SCOPE=SPFILE;
System altered.
SQL> ■
```



We can enable flashback database as follows:

```
SQL> exit

mkdir -p /u01/app/oracle/fast_recovery_area/
[oracle@OracleVM1 ~]$ mkdir -p /u01/app/oracle/fast_recovery_area/
[oracle@OracleVM1 ~]$ ■

sqlplus / as sysdba
SQL> ALTER SYSTEM SET db_flashback_retention_target=2880 scope=BOTH;
SQL> ALTER SYSTEM SET db_recovery_file_dest_size=50G scope=both sid='*';
SQL> ALTER SYSTEM SET
db_recovery_file_dest='/u01/app/oracle/fast_recovery_area' scope=both
sid='*';
SQL> ALTER DATABASE FLASHBACK ON;
[oracle@OracleVM1 ~]$ sqlplus / as sysdba

SQL*Plus: Release 19.0.0.0.0 - Production on Wed Nov 15 22:06:27 2023
Version 19.3.0.0.0

Copyright (c) 1982, 2019, Oracle. All rights reserved.

Connected to:
Oracle Database 19c Enterprise Edition Release 19.0.0.0.0 - Production
Version 19.3.0.0.0

SQL> ALTER SYSTEM SET db_flashback_retention_target=2880 scope=BOTH;
System altered.

SQL> ALTER SYSTEM SET db_recovery_file_dest_size=50G scope=both sid='*';
■ystem altered.

SQL> ALTER SYSTEM SET db_recovery_file_dest='/u01/app/oracle/fast_recovery_area' scope=both sid='*';
System altered.

SQL> ALTER DATABASE FLASHBACK ON;

Database altered.

SQL> ■
```

The following is optional:

```
SQL> alter system set CONTROL_FILE_RECORD_KEEP_TIME=30 scope=BOTH;  
SQL> alter system set CONTROL_FILE_RECORD_KEEP_TIME=30 scope=BOTH;  
System altered.  
SQL> █
```

Let us see which in-memory server parameters have “dubai” in it:

```
SQL> SELECT NAME, VALUE FROM V$PARAMETER WHERE upper(VALUE) LIKE  
upper('%/dubai/%');  
SQL> SELECT NAME, VALUE FROM V$PARAMETER WHERE upper(VALUE) LIKE upper('%/dubai/%');  
  
NAME  
-----  
VALUE  
--  
control_files  
/u01/app/oracle/oradata/DUBAI/control01.ctl, /u01/app/oracle/oradata/DUBAI/contr  
ol02.ctl  
  
core_dump_dest  
/u01/app/oracle/diag/rdbms/dubai/dubai/cdump  
  
audit_file_dest  
/u01/app/oracle/admin/dubai/adump  
  
NAME  
-----  
VALUE  
--  
  
SQL> █
```

Let us check SPFILE contents for the same string:

```
SQL> SELECT NAME, VALUE FROM V$SPPARAMETER WHERE LOWER(VALUE) LIKE '%dubai%';
```

```
SQL> SELECT NAME, VALUE FROM V$SPPARAMETER WHERE LOWER(VALUE) LIKE '%dubai%';
NAME
-----
VALUE
-----
control_files
/u01/app/oracle/oradata/DUBAI/control01.ctl

control_files
/u01/app/oracle/oradata/DUBAI/control02.ctl

db_file_name_convert
/dubaistb/

NAME
-----
VALUE
-----
db_file_name_convert
/dubai/

log_file_name_convert
/dubaistb/

log_file_name_convert
/dubai/

NAME
-----
VALUE
-----
log_archive_dest_1
```

```

LOCATION=USE_DB_RECOVERY_FILE_DEST VALID_FOR=(ALL_LOGFILES,ALL_ROLES) DB_UNIQUE_
NAME=dubai

log_archive_dest_2
SERVICE=DXB$TB ASYNC VALID_FOR=(ONLINE_LOGFILES,PRIMARY_ROLE) DB_UNIQUE_NAME=dub
aistb

log_archive_config

NAME
-----
VALUE
-----
DG_CONFIG=(dubai,dubaistb)

» spatchers
(PROTOCOL=TCP) (SERVICE=dubaiXDB)

audit_file_dest
/u01/app/oracle/admin/dubai/adump

db_name

NAME
-----
VALUE
-----
dubai

12 rows selected.

SQL> ■

```

```

SQL> SELECT NAME, VALUE FROM V$SPPARAMETER WHERE LOWER(VALUE) LIKE '%dxb%';

SQL> SELECT NAME, VALUE FROM V$SPPARAMETER WHERE LOWER(VALUE) LIKE '%dxb%';

NAME
-----
VALUE
-----
log_archive_dest_2
SERVICE=DXB$TB ASYNC VALID_FOR=(ONLINE_LOGFILES,PRIMARY_ROLE) DB_UNIQUE_NAME=dub
aistb

fal_server
DXB$TB

SQL> ■

```

As we can see, most of the parameters are in SPFILE but not in memory.

Since a lot of database parameters have been modified, you may want to shut down and restart the primary database so that the changes that were written only to SCOPE=SPFILE also take effect in memory:

```
SQL> shutdown immediate;
SQL> startup;

SQL> shutdown immediate;
Database closed.
Database dismounted.
ORACLE instance shut down.
SQL> startup;
ORACLE instance started.

Total System Global Area  998244240 bytes
Fixed Size                  9142160 bytes
Variable Size                260046848 bytes
Database Buffers            725614592 bytes
Redo Buffers                 3440640 bytes
Database mounted.
Database opened.
SQL> █
```

Create PFILE for Standby Server (optional)

Now, we will create a PFILE to use when starting the standby server. We have two options.

Option (1) – create a simple PFILE on the standby server. We will see this example later.

Option (2) – create a PFILE from the existing SPFILE on primary server:

```
SQL> CREATE PFILE='/tmp/initdubaistb.ora' FROM SPFILE;
SQL> exit
SQL> CREATE PFILE='/tmp/initdubaistb.ora' FROM SPFILE;

File created.

SQL> █
```

```
more /tmp/initdubaistb.ora
```

```
[oracle@OracleVM1 DUBAI]$ more /tmp/initdubaistb.ora
dubai._data_transfer_cache_size=0
dubai._db_cache_size=658505728
dubai._inmemory_ext_roarea=0
dubai._inmemory_ext_rwarea=0
dubai._java_pool_size=0
dubai._large_pool_size=4194304
dubai._oracle_base='/u01/app/oracle'#ORACLE_BASE set from environment
dubai._pga_aggregate_target=335544320
dubai._sga_target=998244352
dubai._shared_io_pool_size=46137344
dubai._shared_pool_size=276824064
dubai._streams_pool_size=0
dubai._unified_pga_pool_size=0
*.audit_file_dest='/u01/app/oracle/admin/dubai/adump'
*.audit_trail='db'
*.compatible='19.0.0'
»control_file_record_keep_time=30
*.control_files='/u01/app/oracle/oradata/DUBAI/control01.ctl','/u01/app/oracle/oradata/DUBAI/control02.ctl'
*.db_block_size=8192
*.db_domain='localdomain'
*.db_file_name_convert='/dubaistb','/dubai','/DUBAISTB','/DUBAI'
*.db_flashback_retention_target=2880
*.db_name='dubai'
*.db_recovery_file_dest_size=53687091200
*.db_recovery_file_dest='/u01/app/oracle/fast_recovery_area'
*.diagnostic_dest='/u01/app/oracle'
*.dispatchers='(PROTOCOL=TCP) (SERVICE=dubaiXDB)'
*.enable_pluggable_database=true
*.fal_server='DXBSTB'
*.log_archive_config='DG_CONFIG=(dubai,dubaistb)'
*.log_archive_dest_1='LOCATION=USE_DB RECOVERY_FILE_DEST VALID_FOR=(ALL_LOGFILES,ALL_ROLES) DB_UNIQUE_NAME=dubai'
*.log_archive_dest_2='SERVICE=DXBSTB ASYNC VALID_FOR=(ONLINE_LOGFILES,PRIMARY_ROLE) DB_UNIQUE_NAME=dubaistb'
*.log_archive_format='%t_%s_%r.arc'
*.log_file_name_convert='/dubaistb','/dubai','/DUBAISTB','/DUBAI'

*.nls_language='AMERICAN'
*.nls_territory='AMERICA'
*.open_cursors=300
*.pga_aggregate_target=317m
*.processes=300
*.remote_login_passwordfile='EXCLUSIVE'
*.sga_target=951m
*.standby_file_management='AUTO'
*.undo_tablespace='UNDOTBS1'
[oracle@OracleVM1 DUBAI]$
```

Now we could edit the file and make the following changes (we could also overwrite these and others later using the RMAN command line on the standby server):

```
# change it to contain the following parameters:
*.db_unique_name='dxbstb'
*.fal_server='DXB'
*.log_archive_dest_2='SERVICE=DXB ASYNC
VALID_FOR=(ONLINE_LOGFILES,PRIMARY_ROLE) DB_UNIQUE_NAME=dubai'
*.log_archive_config='DG_CONFIG=(dubai,dubaistb)'
```

Then the file should be securely copied over to the Standby server.

However, in this example we will go with the other option of creating a simple PFILE on the standby server.

Create Control File for Standby Server (optional)

This step is optional since a control file is created on the standby without needing to copy the following one.

Here is how you can create a control file:

```
sqlplus / as sysdba
SQL> ALTER DATABASE CREATE STANDBY CONTROLFILE AS '/tmp/dubaistb.ctl';
SQL> exit
[oracle@OracleVM1 ~]$ sqlplus / as sysdba

SQL*Plus: Release 19.0.0.0.0 - Production on Wed Nov 15 22:18:28 2023
Version 19.3.0.0.0

Copyright (c) 1982, 2019, Oracle. All rights reserved.

Connected to:
Oracle Database 19c Enterprise Edition Release 19.0.0.0.0 - Production
Version 19.3.0.0.0

SQL> ALTER DATABASE CREATE STANDBY CONTROLFILE AS '/tmp/dubaistb.ctl';

Database altered.

SQL>
```

This file should be securely copied over to the Standby server. However, if we decide to create PFILE from scratch on standby server, then control file is generated.

Check Directory Structure on Primary Server

Let us check the directory structure on the primary:

```
cd /u01/app/oracle/
ls -l
```

```
[oracle@OracleVM1 ~]$ cd /u01/app/oracle/
[oracle@OracleVM1 oracle]$ ls -l
total 0
drwxr-x---. 1 oracle oinstall 10 Nov 15 20:41 admin
drwxr-x---. 1 oracle oinstall 10 Nov 15 20:44 audit
drwxr-x---. 1 oracle oinstall 24 Nov 15 20:44 cfgtoollogs
drwxr-xr-x. 1 oracle oinstall 0 Mar 17 2020 checkpoints
drwxrwxr-x. 1 oracle oinstall 212 Mar 17 2020 diag
drwxr-xr-x. 1 oracle oinstall 10 Nov 15 22:07 fast_recovery_area
drwxr-x---. 1 oracle oinstall 10 Nov 15 20:41 oradata
drwxr-xr-x. 1 oracle oinstall 12 Mar 17 2020 product
[oracle@OracleVM1 oracle]$ █
```

```
cd oradara/DUBAI
ls -l
[oracle@OracleVM1 oracle]$ cd oradata/DUBAI
[oracle@OracleVM1 DUBAI]$ ls -l
total 3363980
-rw-r-----. 1 oracle oinstall 18726912 Nov 15 22:21 control01.ctl
-rw-r-----. 1 oracle oinstall 18726912 Nov 15 22:21 control02.ctl
drwxr-x---. 1 oracle oinstall 116 Nov 15 21:18 pdb1
drwxr-x---. 1 oracle oinstall 146 Nov 15 20:53 pdbseed
-rw-r-----. 1 oracle oinstall 209715712 Nov 15 22:21 redo01.log
-rw-r-----. 1 oracle oinstall 209715712 Nov 15 22:12 redo02.log
-rw-r-----. 1 oracle oinstall 209715712 Nov 15 22:12 redo03.log
-rw-r-----. 1 oracle oinstall 209715712 Nov 15 21:59 standby_redo01.log
-rw-r-----. 1 oracle oinstall 209715712 Nov 15 21:59 standby_redo02.log
-rw-r-----. 1 oracle oinstall 209715712 Nov 15 21:59 standby_redo03.log
-rw-r-----. 1 oracle oinstall 209715712 Nov 15 21:59 standby_redo04.log
-rw-r-----. 1 oracle oinstall 576724992 Nov 15 22:17 sysaux01.dbf
-rw-r-----. 1 oracle oinstall 943726592 Nov 15 22:19 system01.dbf
-rw-r-----. 1 oracle oinstall 135274496 Nov 15 22:02 temp01.dbf
-rw-r-----. 1 oracle oinstall 288366592 Nov 15 22:17 undotbs01.dbf
-rw-r-----. 1 oracle oinstall 5251072 Nov 15 22:12 users01.dbf
[oracle@OracleVM1 DUBAI]$ █
```

Here is the recovery area:

```
sqlplus / as sysdba
SQL> show parameter recovery
```

```
[oracle@OracleVM1 DUBAI]$ sqlplus / as sysdba
SQL*Plus: Release 19.0.0.0.0 - Production on Wed Nov 15 22:22:00 2023
Version 19.3.0.0.0
Copyright (c) 1982, 2019, Oracle. All rights reserved.

Connected to:
Oracle Database 19c Enterprise Edition Release 19.0.0.0.0 - Production
Version 19.3.0.0.0

SQL> show parameter recovery

NAME                                     TYPE        VALUE
-----                                     -----
db_recovery_file_dest                  string      /u01/app/oracle/fast_recovery_
                                            area
db_recovery_file_dest_size            big integer 50G
recovery_parallelism                 integer     0
remote_recovery_file_dest            string      SQL> 
```

```
SQL> exit
cd /u01/app/oracle/fast_recovery_area
ls -l
SQL> exit
Disconnected from Oracle Database 19c Enterprise Edition Release 19.0.0.0.0 - Production
Version 19.3.0.0.0
[oracle@OracleVM1 DUBAI]$ cd /u01/app/oracle/fast_recovery_area/
[oracle@OracleVM1 fast_recovery_area]$ ls -l
total 0
drwxr-x---. 1 oracle oinstall 58 Nov 15 22:22 DUBAI
[oracle@OracleVM1 fast_recovery_area]$ 
```

```
cd DUBAI
ls -l
[oracle@OracleVM1 fast_recovery_area]$ cd DUBAI
[oracle@OracleVM1 DUBAI]$ ls -l
total 0
drwxr-x---. 1 oracle oinstall 20 Nov 15 22:09 archivelog
drwxr-x---. 1 oracle oinstall 20 Nov 15 22:22 autobackup
drwxr-x---. 1 oracle oinstall 76 Nov 15 22:07 flashback
[oracle@OracleVM1 DUBAI]$ 
```

Configuring Standby

Initially, there was no DB instance on standby. It will be created via the RMAN DUPLICATE process later.

Disable Firewalls

Azure Bastion protects your virtual machines by providing lightweight, browser-based connectivity without the need to expose them through public IP addresses. Deploying will automatically create a Bastion host on a subnet in your virtual network. [Learn more](#)

Using Bastion: **OracleLabBastion**

Provisioning State: **Succeeded**

Please enter username and password to your virtual machine to connect using Bastion.

Authentication Type: VM Password

Username: azureuser

VM Password: OracleLab123

Hide

Open in new browser tab

Connect

```
sudo systemctl stop firewalld
sudo systemctl disable firewalld
```

```
[azureuser@OracleVM2 ~]$ sudo systemctl stop firewalld
[azureuser@OracleVM2 ~]$ sudo systemctl disable firewalld
Removed symlink /etc/systemd/system/multi-user.target.wants/firewalld.service.
Removed symlink /etc/systemd/system/dbus-org.fedoraproject.FirewallD1.service.
[azureuser@OracleVM2 ~]$
```

Set Oracle user password

```
sudo passwd oracle

[azureuser@OracleVM2 ~]$ sudo passwd oracle
Changing password for user oracle.
New password:
BAD PASSWORD: The password contains the user name in some form
Retype new password:
passwd: all authentication tokens updated successfully.
[azureuser@OracleVM2 ~]$
```

Update .bashrc

```
sudo su - oracle
[azureuser@OracleVM2 oracle]$ cd
[azureuser@OracleVM2 ~]$
[azureuser@OracleVM2 ~]$
[azureuser@OracleVM2 ~]$
[azureuser@OracleVM2 ~]$ sudo su - oracle

We trust you have received the usual lecture from the local System
Administrator. It usually boils down to these three things:

#1) Respect the privacy of others.
#2) Think before you type.
#3) With great power comes great responsibility.

[sudo] password for azureuser:
[oracle@OracleVM2 ~]$
```

```
ORACLE_HOME=/u01/app/oracle/product/19.0.0/dbhome_1; export ORACLE_HOME
ORACLE_SID=dubaistb; export ORACLE_SID

[oracle@OracleVM2 dbhome_1]$ ORACLE_HOME=/u01/app/oracle/product/19.0.0/dbhome_1; export ORACLE_HOME
[oracle@OracleVM2 dbhome_1]$ ORACLE_SID=dubaistb; export ORACLE_SID
[oracle@OracleVM2 dbhome_1]$
```



```
echo "export ORACLE_HOME=/u01/app/oracle/product/19.0.0/dbhome_1" >> ~/.bashrc
echo "export ORACLE_SID=dubaistb" >> ~/.bashrc

[oracle@OracleVM2 dbhome_1]$ echo "export ORACLE_HOME=/u01/app/oracle/product/19.0.0/dbhome_1" >> ~/.bashrc
[oracle@OracleVM2 dbhome_1]$ echo "export ORACLE_SID=dubaistb" >> ~/.bashrc
[oracle@OracleVM2 dbhome_1]$
```

Let us check the *.ora files:

```
cd $ORACLE_HOME
find . -name "*.ora"
```

```
[oracle@OracleVM2 ~]$ cd $ORACLE_HOME
[oracle@OracleVM2 dbhome_1]$ find . -name "*.ora"
./inventory/Templates/drdaas/admin/drdaas.ora
./hs/admin/initdg4odbc.ora
./hs/admin/extproc.ora
./srvm/admin/init.ora
./mgw/admin/sample_mgw.ora
./network/admin/samples/listener.ora
./network/admin/samples/sqlnet.ora
./network/admin/samples/tnsnames.ora
»./drdaas/admin/drdaas.ora
./rdbms/admin/externaljob.ora
./rdbms/install/filemap.ora
./dbs/init.ora
./env.ora
[oracle@OracleVM2 dbhome_1]$
```

Create Subdirectory Structure

Next, we create the required folder structure:

```
sudo su - oracle

mkdir -p /u01/app/oracle/oradata/DUBAISTB
mkdir -p /u01/app/oracle/oradata/DUBAISTB/pdbseed
mkdir -p /u01/app/oracle/oradata/DUBAISTB/pdb1
mkdir -p /u01/app/oracle/fast_recovery_area/
mkdir -p /u01/app/oracle/admin/dubaistb/adump

mkdir -p /u01/app/oracle/diag/rdbms/dubaistb/dubaistb/cdump
```

```
[oracle@OracleVM2 ~]$ mkdir -p /u01/app/oracle/oradata/DUBAISTB
[oracle@OracleVM2 ~]$ mkdir -p /u01/app/oracle/oradata/DUBAISTB/pdbseed
[oracle@OracleVM2 ~]$ mkdir -p /u01/app/oracle/oradata/DUBAISTB/pdb1
[oracle@OracleVM2 ~]$ mkdir -p /u01/app/oracle/fast_recovery_area/
[oracle@OracleVM2 ~]$ mkdir -p /u01/app/oracle/admin/dubaistb/adump
[oracle@OracleVM2 ~]$ mkdir -p /u01/app/oracle/diag/rdbms/dubaistb/dubaistb/cdump
[oracle@OracleVM2 ~]$
```

Create Password File

We need to create a password file on the standby server.

```
orapwd file=/u01/app/oracle/product/19.0.0/dbhome_1/dbs/orapwdubais
password=OracleLab123 entries=10 format=12
```

```
[oracle@OracleVM2 dbhome_1]$ orapwd file=/u01/app/oracle/product/19.0.0/dbhome_1/dbs/orapwdubaistb password=OracleLab123 entries=10 format=12  
[oracle@OracleVM2 dbhome_1]$
```

```
ls -l /u01/app/oracle/product/19.0.0/dbhome_1/dbs/  
[oracle@OracleVM2 ~]$ ls -l /u01/app/oracle/product/19.0.0/dbhome_1/dbs/  
total 4  
-rw-r--r-- 1 oracle oinstall 3079 May 14 2015 init.ora  
-rw-r----- 1 oracle oinstall 2048 Nov 15 22:30 orapwdubaistb  
[oracle@OracleVM2 ~]$
```

This sets the password for SYS and SYSTEM.

Modify /etc/hosts (as azureuser)

```
exit  
sudo vi /etc/hosts  
[oracle@OracleVM2 ~]$ exit  
logout  
[azureuser@OracleVM2 ~]$ sudo vi /etc/hosts  
127.0.0.1 localhost localhost.localdomain localhost4 localhost4.localdomain4  
::1 localhost localhost.localdomain localhost6 localhost6.localdomain6  
10.0.1.5 OracleVM2 OracleVM2.localdomain  
~
```

```
sudo su - oracle
```

Configure Listener

Let us configure the listener for the standby.

```
vi $ORACLE_HOME/network/admin/listener.ora  
  
LISTENER =  
(DESCRIPTION_LIST =  
(DESCRIPTION =  
(ADDRESS = (PROTOCOL = TCP) (HOST = OracleVM2) (PORT = 1521))  
(ADDRESS = (PROTOCOL = IPC) (KEY = EXTPROC1521))  
)  
)  
SID_LIST_LISTENER =  
(SID_LIST =  
(SID_DESC =  
(GLOBAL_DBNAME = dubaistbserv)  
(ORACLE_HOME = /u01/app/oracle/product/19.0.0/dbhome_1)  
(SID_NAME = dubaistb)
```

```

        )
)
ADR_BASE_LISTENER = /u01/app/oracle
[oracle@OracleVM2 ~]$ more $ORACLE_HOME/network/admin/listener.ora
LISTENER =
(DESCRIPTION_LIST =
  (DESCRIPTION =
    (ADDRESS = (PROTOCOL = TCP)(HOST = OracleVM2)(PORT = 1521))
    (ADDRESS = (PROTOCOL = IPC)(KEY = EXTPROC1521))
  )
)
SID_LIST_LISTENER =
(SID_LIST =
  (SID_DESC =
    (GLOBAL_DBNAME = dubaistbserv)
    (ORACLE_HOME = /u01/app/oracle/product/19.0.0/dbhome_1)
    (SID_NAME = dubaistb)
  )
)
ADR_BASE_LISTENER = /u01/app/oracle
[oracle@OracleVM2 ~]$ █

```

Let us restart the listener:

```

lsnrctl stop
[oracle@OracleVM2 ~]$ lsnrctl stop

LSNRCTL for Linux: Version 19.0.0.0.0 - Production on 15-NOV-2023 22:35:31

Copyright (c) 1991, 2019, Oracle. All rights reserved.

Connecting to (DESCRIPTION=(ADDRESS=(PROTOCOL=TCP)(HOST=OracleVM2)(PORT=1521)))
TNS-12541: TNS:no listener
TNS-12560: TNS:protocol adapter error
TNS-00511: No listener
  Linux Error: 111: Connection refused
Connecting to (DESCRIPTION=(ADDRESS=(PROTOCOL=IPC)(KEY=EXTPROC1521)))
TNS-12541: TNS:no listener
TNS-12560: TNS:protocol adapter error
TNS-00511: No listener
  Linux Error: 2: No such file or directory
[oracle@OracleVM2 ~]$ █

```

```
lsnrctl start
```

```
[oracle@OracleVM2 ~]$ lsnrctl start

LSNRCTL for Linux: Version 19.0.0.0.0 - Production on 15-NOV-2023 22:35:58

Copyright (c) 1991, 2019, Oracle. All rights reserved.

Starting /u01/app/oracle/product/19.0.0/dbhome_1/bin/tnslsnr: please wait...

TNSLSNR for Linux: Version 19.0.0.0.0 - Production
System parameter file is /u01/app/oracle/product/19.0.0/dbhome_1/network/admin/listener.ora
Log messages written to /u01/app/oracle/diag/tnslsnr/OracleVM2/listener/alert/log.xml
Listening on: (DESCRIPTION=(ADDRESS=(PROTOCOL=tcp)(HOST=OracleVM2)(PORT=1521)))
Listening on: (DESCRIPTION=(ADDRESS=(PROTOCOL=ipc)(KEY=EXTPROC1521)))

Connecting to (DESCRIPTION=(ADDRESS=(PROTOCOL=TCP)(HOST=OracleVM2)(PORT=1521)))
STATUS of the LISTENER
-----
Alias                      LISTENER
Version        TNSLSNR for Linux: Version 19.0.0.0.0 - Production
Start Date     15-NOV-2023 22:36:00
Uptime         0 days 0 hr. 0 min. 1 sec
Trace Level    off
Security       ON: Local OS Authentication
SNMP           OFF
Listener Parameter File   /u01/app/oracle/product/19.0.0/dbhome_1/network/admin/listener.ora
Listener Log File        /u01/app/oracle/diag/tnslsnr/OracleVM2/listener/alert/log.xml
Listening Endpoints Summary...
  (DESCRIPTION=(ADDRESS=(PROTOCOL=tcp)(HOST=OracleVM2)(PORT=1521)))
  (DESCRIPTION=(ADDRESS=(PROTOCOL=ipc)(KEY=EXTPROC1521)))
Services Summary...
Service "dubaistbserv" has 1 instance(s).
  Instance "dubaistb", status UNKNOWN, has 1 handler(s) for this service...
The command completed successfully
[oracle@OracleVM2 ~]$
```

Wait for 2 mins and list the listener services.

```
lsnrctl services
[oracle@OracleVM2 ~]$ lsnrctl services

LSNRCTL for Linux: Version 19.0.0.0.0 - Production on 15-NOV-2023 22:37:02

Copyright (c) 1991, 2019, Oracle. All rights reserved.

Connecting to (DESCRIPTION=(ADDRESS=(PROTOCOL=TCP)(HOST=OracleVM2)(PORT=1521)))
Services Summary...
Service "dubaistbserv" has 1 instance(s).
  Instance "dubaistb", status UNKNOWN, has 1 handler(s) for this service...
    Handler(s):
      "DEDICATED" established:0 refused:0
      LOCAL SERVER
The command completed successfully
[oracle@OracleVM2 ~]$
```

We see a static listener service called “dubaistbserv”.

Even though there is NO DATABASE, we can test the listener service by connecting to idle instance using Easy Connect format:

```
sqlplus sys/OracleLab123@OracleVM2:1521/dubaistbserv as sysdba
[oracle@OracleVM2 ~]$ sqlplus sys/OracleLab123@OracleVM2:1521/dubaistbserv as sysdba
SQL*Plus: Release 19.0.0.0.0 - Production on Wed Nov 15 22:39:14 2023
Version 19.3.0.0.0

Copyright (c) 1982, 2019, Oracle. All rights reserved.

Connected to an idle instance.

SQL> 
```

We will use “SERVICE_NAME=dubaistbserv” in the tnsnames.ora file.

Configure Name Resolution

We need to do the following for name resolution. The tnsnames.ora is identical to the primary:

```
vi $ORACLE_HOME/network/admin/tnsnames.ora

DXB =
  (DESCRIPTION =
    (ADDRESS_LIST =
      (ADDRESS = (PROTOCOL = TCP) (HOST = OracleVM1) (PORT = 1521))
    )
    (CONNECT_DATA =
      (SID = dubai)
      (SERVICE_NAME=dubaiserv)
    )
  )

DXBSTB =
  (DESCRIPTION =
    (ADDRESS_LIST =
      (ADDRESS = (PROTOCOL = TCP) (HOST = OracleVM2) (PORT = 1521))
    )
    (CONNECT_DATA =
      (SID = dubaistb)
      (SERVICE_NAME=dubaistbserv)
    )
  )
```

```
[oracle@OracleVM2 ~]$ more $ORACLE_HOME/network/admin/tnsnames.ora
DXB =
(DESCRIPTION =
  (ADDRESS_LIST =
    (ADDRESS = (PROTOCOL = TCP)(HOST = OracleVM1)(PORT = 1521))
  )
  (CONNECT_DATA =
    (SID = dubai)
    (SERVICE_NAME=dubaiserv)
  )
)

DXBSTB =
(DESCRIPTION =
  (ADDRESS_LIST =
    (ADDRESS = (PROTOCOL = TCP)(HOST = OracleVM2)(PORT = 1521))
  )
  (CONNECT_DATA =
    (SID = dubaistb)
  >>    (SERVICE_NAME=dubaistbserv)
  )
)

[oracle@OracleVM2 ~]$
```

Let us check the connectivity via connect descriptor in tnsnames.ora:

```
tnsping DXBSTB
[oracle@OracleVM2 ~]$ tnsping DXBSTB
TNS Ping Utility for Linux: Version 19.0.0.0.0 - Production on 15-NOV-2023 22:45:05
Copyright (c) 1997, 2019, Oracle. All rights reserved.
Used parameter files:

Used TNSNAMES adapter to resolve the alias
Attempting to contact (DESCRIPTION = (ADDRESS_LIST = (ADDRESS = (PROTOCOL = TCP)(HOST = OracleVM2)(PORT = 1521))) (CONNECT_DATA = (SID = dubaistb) (SERVICE_NAME=dubaistbserv)))
OK (0 msec)
[oracle@OracleVM2 ~]$
```

```
tnsping DXB
[oracle@OracleVM2 ~]$ tnsping DXB
TNS Ping Utility for Linux: Version 19.0.0.0.0 - Production on 15-NOV-2023 22:45:35
Copyright (c) 1997, 2019, Oracle. All rights reserved.
Used parameter files:

Used TNSNAMES adapter to resolve the alias
Attempting to contact (DESCRIPTION = (ADDRESS_LIST = (ADDRESS = (PROTOCOL = TCP)(HOST = OracleVM1)(PORT = 1521))) (CONNECT_DATA = (SID = dubai) (SERVICE_NAME=dubaiserv)))
OK (10 msec)
[oracle@OracleVM2 ~]$
```

From standby, let us also check SQLPLUS* connectivity as well:

```
sqlplus sys/OracleLab123@DXB$ as sysdba
[oracle@OracleVM2 ~]$ sqlplus sys/OracleLab123@DXB$ as sysdba
SQL*Plus: Release 19.0.0.0.0 - Production on Wed Nov 15 22:46:10 2023
Version 19.3.0.0.0

Copyright (c) 1982, 2019, Oracle. All rights reserved.

Connected to an idle instance.

SQL> █
```

```
sqlplus sys/OracleLab123@DXB as sysdba
[oracle@OracleVM2 ~]$ sqlplus sys/OracleLab123@DXB as sysdba
SQL*Plus: Release 19.0.0.0.0 - Production on Wed Nov 15 22:47:14 2023
Version 19.3.0.0.0

Copyright (c) 1982, 2019, Oracle. All rights reserved.

Connected to:
Oracle Database 19c Enterprise Edition Release 19.0.0.0.0 - Production
Version 19.3.0.0.0

SQL> █
```

Create PFILE

Let us create a very simple PFILE for the standby server. NOTE THAT THE “DB_NAME” IS IDENTICAL TO PRIMARY.

```
echo "*.*.db_name='dubai'" > /tmp/initdubaistb.ora
echo "*.*.db_unique_name='dubaistb'" >> /tmp/initdubaistb.ora
[oracle@OracleVM2 ~]$ echo "*.*.db_name='dubai'" > /tmp/initdubaistb.ora
[oracle@OracleVM2 ~]$ echo "*.*.db_unique_name='dubaistb'" >> /tmp/initdubaistb.ora
[oracle@OracleVM2 ~]$ █
```

Start Standby DB as NOMOUNT

Now, we can start the standby DB in NOMOUNT mode and run RMAN.

```
sqlplus / as sysdba
[oracle@OracleVM2 ~]$ sqlplus / as sysdba

SQL*Plus: Release 19.0.0.0.0 - Production on Wed Nov 15 22:48:59 2023
Version 19.3.0.0.0

Copyright (c) 1982, 2019, Oracle. All rights reserved.

Connected to an idle instance.

SQL> ■
```

```
SQL> STARTUP NOMOUNT PFILE='/tmp/initdubaistb.ora';
SQL> show parameter db_name
SQL> show parameter db_unique_name
SQL> exit
[oracle@OracleVM2 ~]$ sqlplus / as sysdba

SQL*Plus: Release 19.0.0.0.0 - Production on Wed Nov 15 22:48:59 2023
Version 19.3.0.0.0

Copyright (c) 1982, 2019, Oracle. All rights reserved.

Connected to an idle instance.

SQL> STARTUP NOMOUNT PFILE='/tmp/initdubaistb.ora';
ORACLE instance started.

Total System Global Area  243268216 bytes
Fixed Size                  8895096 bytes
Variable Size                180355072 bytes
Database Buffers            50331648 bytes
Redo Buffers                 3686400 bytes
SQL> show parameter db_name

NAME                           TYPE        VALUE
-----                         -----
db_name                        string      dubai
SQL> show parameter db_unique_name

NAME                           TYPE        VALUE
-----                         -----
db_unique_name                 string      dubaistb
SQL> ■
```

Copy DB from Primary using RMAN and Modify SPFILE Parameters

While the standby DB is in NOMOUNT mode, we will run RMAN and copy the DB instance from primary to standby. We will also update many of the SPFILE parameters.

(If the following RMAN script fails, you must restart the standby DB in NOMOUNT mode).

```
rman
RMAN> CONNECT TARGET sys/OracleLab123@DXB;
RMAN> CONNECT AUXILIARY sys/OracleLab123@DXBSTB;
RMAN> run {
allocate channel prmy1 type disk;
allocate channel prmy2 type disk;
allocate channel prmy3 type disk;
allocate channel prmy4 type disk;
allocate auxiliary channel stby1 type disk;

DUPLICATE TARGET DATABASE FOR STANDBY FROM ACTIVE DATABASE
SPFILE
  set DB_UNIQUE_NAME='dubaistb'
  set FAL_SERVER='DXB'
  set LOG_ARCHIVE_CONFIG='DG_CONFIG=(dubai,dubaistb)'
  set LOG_ARCHIVE_DEST_1='LOCATION=USE_DB_RECOVERY_FILE_DEST
VALID_FOR=(ALL_LOGFILES,ALL_ROLES) DB_UNIQUE_NAME=dubaistb'
  set LOG_ARCHIVE_DEST_2='SERVICE=DXB ASYNC'
VALID_FOR=(ONLINE_LOGFILES,PRIMARY_ROLE) DB_UNIQUE_NAME=dubai'
  set DB_FILE_NAME_CONVERT='/dubai/','/dubaistb/','/DUBAI/','/DUAISTB/'
  set LOG_FILE_NAME_CONVERT='/dubai/','/dubaistb/','/DUBAI/','/DUAISTB/'
  set STANDBY_FILE_MANAGEMENT='AUTO'
  set LOG_ARCHIVE_FORMAT='%t %s %r.arc'
  set CONTROL_FILES='/u01/app/oracle/oradata/DUAISTB/control01.ctl'
  set AUDIT_FILE_DEST='/u01/app/oracle/admin/dubaistb/adump'
  set DB_RECOVERY_FILE_DEST='/u01/app/oracle/fast_recovery_area'
  NOFILENAMECHECK;
}

[oracle@OracleVM2 DUAISTB]$ rman
Recovery Manager: Release 19.0.0.0.0 - Production on Wed Nov 15 23:16:40 2023
Version 19.3.0.0.0

Copyright (c) 1982, 2019, Oracle and/or its affiliates. All rights reserved.

RMAN> CONNECT TARGET sys/OracleLab123@DXB;
connected to target database: DUBAI (DBID=36867046)

RMAN> CONNECT AUXILIARY sys/OracleLab123@DXBSTB;
connected to auxiliary database: DUBAI (not mounted)

RMAN> run {
allocate channel prmy1 type disk;
allocate channel prmy2 type disk;
allocate channel prmy3 type disk;
allocate channel prmy4 type disk;
allocate auxiliary channel stby1 type disk;

DUPLICATE TARGET DATABASE FOR STANDBY FROM ACTIVE DATABASE
SPFILE
  set DB_UNIQUE_NAME='dubaistb'
  set FAL_SERVER='DXB'
  set LOG_ARCHIVE_CONFIG='DG_CONFIG=(dubai,dubaistb)'
  set LOG_ARCHIVE_DEST_1='LOCATION=USE_DB_RECOVERY_FILE_DEST
VALID_FOR=(ALL_LOGFILES,ALL_ROLES) DB_UNIQUE_NAME=dubaistb'
  set LOG_ARCHIVE_DEST_2='SERVICE=DXB ASYNC VALID_FOR=(ONLINE_LOGFILES,PRIMARY_ROLE) DB_UNIQUE_NAME=dubai'
  set DB_FILE_NAME_CONVERT='/dubai/','/dubaistb/','/DUBAI/','/DUAISTB/'
  set LOG_FILE_NAME_CONVERT='/dubai/','/dubaistb/','/DUBAI/','/DUAISTB/'
  set STANDBY_FILE_MANAGEMENT='AUTO'
  set LOG_ARCHIVE_FORMAT='%t %s %r.arc'
  set CONTROL_FILES='/u01/app/oracle/oradata/DUAISTB/control01.ctl'
  set AUDIT_FILE_DEST='/u01/app/oracle/admin/dubaistb/adump'
  set DB_RECOVERY_FILE_DEST='/u01/app/oracle/fast_recovery_area'
```



```

NOFILENAMECHECK;
}
2> 3> 4> 5> 6> 7> 8> 9> 10> 11> 12> 13> 14> 15> 16> 17> 18> 19> 20> 21> 22> 23>
using target database control file instead of recovery catalog
allocated channel: prmy1
channel prmy1: SID=465 device type=DISK
allocated channel: prmy2
channel prmy2: SID=471 device type=DISK
allocated channel: prmy3
channel prmy3: SID=470 device type=DISK
allocated channel: prmy4
channel prmy4: SID=469 device type=DISK
allocated channel: stby1
channel stby1: SID=38 device type=DISK
Starting Duplicate Db at 15-NOV-23
contents of Memory Script:
{
  backup as copy reuse
  passwordfile auxiliary format '/u01/app/oracle/product/19.0.0/dbhome_1/dbs/orapwdubaitb'  targetfile
  '/u01/app/oracle/product/19.0.0/dbhome_1/dbs/spfiledubaitb.ora' auxiliary format
  '/u01/app/oracle/product/19.0.0/dbhome_1/dbs/spfiledubaitb.ora' ;
  sql clone "alter system set spfile= ''/u01/app/oracle/product/19.0.0/dbhome_1/dbs/spfiledubaitb.ora'''";
}
executing Memory Script
Starting backup at 15-NOV-23
Finished backup at 15-NOV-23
sql statement: alter system set spfile= ''/u01/app/oracle/product/19.0.0/dbhome_1/dbs/spfiledubaitb.ora''

```



```

contents of Memory Script:
{
  sql clone "alter system set db_unique_name =
  ''dubaistb'' comment=
  '''' scope=spfile";
  sql clone "alter system set FAL_SERVER =
  ''DXB'' comment=
  '''' scope=spfile";
  sql clone "alter system set LOG_ARCHIVE_CONFIG =
  ''DG_CONFIG=(dubai,dubaistb)'' comment=
  '''' scope=spfile";
  sql clone "alter system set LOG_ARCHIVE_DEST_1 =
  ''LOCATION=USE DB RECOVERY_FILE_DEST VALID_FOR=(ALL_LOGFILES,ALL_ROLES) DB_UNIQUE_NAME=dubaistb'' comment=
  '''' scope=spfile";
  sql clone "alter system set LOG_ARCHIVE_DEST_2 =
  ''SERVICE=DXB_ASYNC VALID_FOR=(ONLINE_LOGFILES,PRIMARY_ROLE) DB_UNIQUE_NAME=dubai'' comment=
  '''' scope=spfile";
  sql clone "alter system set db_file_name_convert =
  ''/dubai/'', ''/dubaistb/'', ''/DUBAI/'', ''/DUBAISTB/'' comment=
  '''' scope=spfile";
  sql clone "alter system set LOG_FILE_NAME_CONVERT =
  ''/dubai/'', ''/dubaistb/'', ''/DUBAI/'', ''/DUBAISTB/'' comment=
  '''' scope=spfile";
  sql clone "alter system set STANDBY_FILE_MANAGEMENT =
  ''AUTO'' comment=
  '''' scope=spfile";
  sql clone "alter system set LOG_ARCHIVE_FORMAT =
  ''%t %s %r.arc'' comment=
  '''' scope=spfile";
  sql clone "alter system set CONTROL_FILES =
  ''/u01/app/oracle/oradata/DUBAISTB/control01.ctl'' comment=
  '''' scope=spfile";
  sql clone "alter system set AUDIT_FILE_DEST =

```

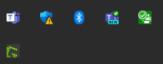


```

''/u01/app/oracle/admin/dubaistb/adump'' comment=
  ''' scope=spfile;
  sql clone "alter system set db_recovery_file_dest =
''/u01/app/oracle/fast_recovery_area'' comment=
  ''' scope=spfile";
  shutdown clone immediate;
  startup clone nomount;
}
executing Memory Script

sql statement: alter system set db_unique_name = ''dubaistb'' comment= '''' scope=spfile
sql statement: alter system set FAL_SERVER = ''DXB'' comment= '''' scope=spfile
sql statement: alter system set LOG_ARCHIVE_CONFIG = ''DG_CONFIG=(dubai,dubaistb)'' comment= '''' scope=spfile
sql statement: alter system set LOG_ARCHIVE_DEST_1 = ''LOCATION=USE_DB_RECOVERY_FILE_DEST VALID_FOR=(ALL_LOGFILES,ALL_ROLES) DB_UNIQUE_NAME=dubaistb'' comment= '''' scope=spfile
sql statement: alter system set LOG_ARCHIVE_DEST_2 = ''SERVICE=DXB ASYNC VALID_FOR=(ONLINE_LOGFILES,PRIMARY_ROLE) DB_UNIQUE_NAME=dubai'' comment= '''' scope=spfile
sql statement: alter system set db_file_name_convert = ''/dubai'', ''/dubaistb'', ''/DUBAI'', ''/DUBAISTB'' comment= '''' scope=spfile
sql statement: alter system set LOG_FILE_NAME_CONVERT = ''/dubai'', ''/dubaistb'', ''/DUBAI'', ''/DUBAISTB'' comment= '''' scope=spfile
sql statement: alter system set STANDBY_FILE_MANAGEMENT = ''AUTO'' comment= '''' scope=spfile
sql statement: alter system set LOG_ARCHIVE_FORMAT = ''%t_%s_%r.arc'' comment= '''' scope=spfile
sql statement: alter system set CONTROL_FILES = ''/u01/app/oracle/oradata/DUBAISTB/control01.ctl'' comment= '''' scope=spfile
sql statement: alter system set AUDIT_FILE_DEST = ''/u01/app/oracle/admin/dubaistb/adump'' comment= '''' scope=spfile
sql statement: alter system set db_recovery_file_dest = ''/u01/app/oracle/fast_recovery_area'' comment= '''' scope=spfile

```



```

Oracle instance shut down
connected to auxiliary database (not started)
Oracle instance started

Total System Global Area     998244240 bytes

Fixed Size                  9142160 bytes
Variable Size                260046848 bytes
Database Buffers              725614592 bytes
Redo Buffers                  3440640 bytes
allocated channel: stby1
channel stby1: SID=424 device type=DISK

contents of Memory Script:
{
  backup as copy current controlfile for standby auxiliary format '/u01/app/oracle/oradata/DUBAISTB/control01.ctl';
}
executing Memory Script

Starting backup at 15-NOV-23
channel prmy1: starting datafile copy
copying standby control file
output file name=/u01/app/oracle/product/19.0.0/dbhome_1/dbs/snapcf_dubai.f tag=TAG20231115T231744
channel prmy1: datafile copy complete, elapsed time: 00:00:03
Finished backup at 15-NOV-23

contents of Memory Script:
{
  sql clone 'alter database mount standby database';
}
executing Memory Script

```



```

sql statement: alter database mount standby database
contents of Memory Script:
{
  set newname for tempfile 1 to
  "/u01/app/oracle/oradata/DUBAISTB/temp01.dbf";
  set newname for tempfile 2 to
  "/u01/app/oracle/oradata/DUBAISTB/pdbseed/temp012023-11-15_20-53-36-209-PM.dbf";
  set newname for tempfile 3 to
  "/u01/app/oracle/oradata/DUBAISTB/pdb1/temp01.dbf";
  switch clone tempfile all;
  set newname for datafile 1 to
  "/u01/app/oracle/oradata/DUBAISTB/system01.dbf";
  set newname for datafile 3 to
  "/u01/app/oracle/oradata/DUBAISTB/sysaux01.dbf";
  set newname for datafile 4 to
  "/u01/app/oracle/oradata/DUBAISTB/undotbs01.dbf";
  set newname for datafile 5 to
  "/u01/app/oracle/oradata/DUBAISTB/pdbseed/system01.dbf";
  set newname for datafile 6 to
  "/u01/app/oracle/oradata/DUBAISTB/pdbseed/sysaux01.dbf";
  set newname for datafile 7 to
  "/u01/app/oracle/oradata/DUBAISTB/users01.dbf";
  set newname for datafile 8 to
  "/u01/app/oracle/oradata/DUBAISTB/pdbseed/undotbs01.dbf";
  set newname for datafile 9 to
  "/u01/app/oracle/oradata/DUBAISTB/pdb1/system01.dbf";
  set newname for datafile 10 to
  "/u01/app/oracle/oradata/DUBAISTB/pdb1/sysaux01.dbf";
  set newname for datafile 11 to
  "/u01/app/oracle/oradata/DUBAISTB/pdb1/undotbs01.dbf";
  set newname for datafile 12 to
  "/u01/app/oracle/oradata/DUBAISTB/pdb1/users01.dbf";

```



```
backup as copy reuse
  datafile 1 auxiliary format
  "/u01/app/oracle/oradata/DUBAISTB/system01.dbf"    datafile
  3 auxiliary format
  "/u01/app/oracle/oradata/DUBAISTB/sysaux01.dbf"    datafile
  4 auxiliary format
  "/u01/app/oracle/oradata/DUBAISTB/undotbs01.dbf"   datafile
  5 auxiliary format
  "/u01/app/oracle/oradata/DUBAISTB/pdbseed/system01.dbf"  datafile
  6 auxiliary format
  "/u01/app/oracle/oradata/DUBAISTB/pdbseed/sysaux01.dbf"  datafile
  7 auxiliary format
  "/u01/app/oracle/oradata/DUBAISTB/users01.dbf"    datafile
  8 auxiliary format
  "/u01/app/oracle/oradata/DUBAISTB/pdbseed/undotbs01.dbf"  datafile
  9 auxiliary format
  "/u01/app/oracle/oradata/DUBAISTB/pdb1/system01.dbf"   datafile
  10 auxiliary format
  "/u01/app/oracle/oradata/DUBAISTB/pdb1/sysaux01.dbf"   datafile
  11 auxiliary format
  "/u01/app/oracle/oradata/DUBAISTB/pdb1/undotbs01.dbf"   datafile
  12 auxiliary format
  "/u01/app/oracle/oradata/DUBAISTB/pdb1/users01.dbf"   ;
  sql 'alter system archive log current';
}
executing Memory Script
executing command: SET NEWNAME
executing command: SET NEWNAME
executing command: SET NEWNAME
renamed tempfile 1 to /u01/app/oracle/oradata/DUBAISTB/temp01.dbf in control file
renamed tempfile 2 to /u01/app/oracle/oradata/DUBAISTB/pdbseed/temp012023-11-15_20-53-36-209-PM.dbf in control file
renamed tempfile 3 to /u01/app/oracle/oradata/DUBAISTB/pdb1/temp01.dbf in control file
```



```
executing command: SET NEWNAME
Starting backup at 15-NOV-23
channel prmy1: starting datafile copy
input datafile file number=00001 name=/u01/app/oracle/oradata/DUBAI/system01.dbf
channel prmy2: starting datafile copy
input datafile file number=00003 name=/u01/app/oracle/oradata/DUBAI/sysaux01.dbf
channel prmy3: starting datafile copy
input datafile file number=00006 name=/u01/app/oracle/oradata/DUBAI/pdbseed/sysaux01.dbf
channel prmy4: starting datafile copy
input datafile file number=00010 name=/u01/app/oracle/oradata/DUBAI/pdb1/sysaux01.dbf
output file name=/u01/app/oracle/oradata/DUBAISTB/pdbseed/sysaux01.dbf tag=TAG20231115T231753
channel prmy3: datafile copy complete, elapsed time: 00:00:26
channel prmy3: starting datafile copy
input datafile file number=00004 name=/u01/app/oracle/oradata/DUBAI/undotbs01.dbf
```



```

output file name=/u01/app/oracle/oradata/DUBAISTB/pdb1/sysaux01.dbf tag=TAG20231115T231753
channel prmy4: datafile copy complete, elapsed time: 00:00:26
channel prmy4: starting datafile copy
input datafile file number=00005 name=/u01/app/oracle/oradata/DUBAI/pdbseed/system01.dbf
output file name=/u01/app/oracle/oradata/DUBAISTB/system01.dbf tag=TAG20231115T231753
channel prmy1: datafile copy complete, elapsed time: 00:00:27
channel prmy1: starting datafile copy
input datafile file number=00009 name=/u01/app/oracle/oradata/DUBAI/pdb1/system01.dbf
output file name=/u01/app/oracle/oradata/DUBAISTB/sysaux01.dbf tag=TAG20231115T231753
channel prmy2: datafile copy complete, elapsed time: 00:00:31
channel prmy2: starting datafile copy
input datafile file number=00008 name=/u01/app/oracle/oradata/DUBAI/pdbseed/undotbs01.dbf
output file name=/u01/app/oracle/oradata/DUBAISTB/pdb1/system01.dbf tag=TAG20231115T231753
channel prmy1: datafile copy complete, elapsed time: 00:00:18
channel prmy1: starting datafile copy
input datafile file number=00011 name=/u01/app/oracle/oradata/DUBAI/pdb1/undotbs01.dbf
output file name=/u01/app/oracle/oradata/DUBAISTB/pdbseed/undotbs01.dbf tag=TAG20231115T231753
channel prmy2: datafile copy complete, elapsed time: 00:00:15
channel prmy2: starting datafile copy
input datafile file number=00007 name=/u01/app/oracle/oradata/DUBAI/users01.dbf
output file name=/u01/app/oracle/oradata/DUBAISTB/undotbs01.dbf tag=TAG20231115T231753
channel prmy3: datafile copy complete, elapsed time: 00:00:20
channel prmy3: starting datafile copy
input datafile file number=00012 name=/u01/app/oracle/oradata/DUBAI/pdb1/users01.dbf
output file name=/u01/app/oracle/oradata/DUBAISTB/pdbseed/system01.dbf tag=TAG20231115T231753
channel prmy4: datafile copy complete, elapsed time: 00:00:21
output file name=/u01/app/oracle/oradata/DUBAISTB/users01.dbf tag=TAG20231115T231753
channel prmy2: datafile copy complete, elapsed time: 00:00:02
output file name=/u01/app/oracle/oradata/DUBAISTB/pdb1/users01.dbf tag=TAG20231115T231753
channel prmy3: datafile copy complete, elapsed time: 00:00:01
output file name=/u01/app/oracle/oradata/DUBAISTB/pdb1/undotbs01.dbf tag=TAG20231115T231753
channel prmy1: datafile copy complete, elapsed time: 00:00:04
Finished backup at 15-NOV-23

sql statement: alter system archive log current

```



```

contents of Memory Script:
{
  switch clone datafile all;
}
executing Memory Script

datafile 1 switched to datafile copy
input datafile copy RECID=5 STAMP=1153005525 file name=/u01/app/oracle/oradata/DUBAISTB/system01.dbf
datafile 3 switched to datafile copy
input datafile copy RECID=6 STAMP=1153005525 file name=/u01/app/oracle/oradata/DUBAISTB/sysaux01.dbf
datafile 4 switched to datafile copy
input datafile copy RECID=7 STAMP=1153005525 file name=/u01/app/oracle/oradata/DUBAISTB/undotbs01.dbf
datafile 5 switched to datafile copy
input datafile copy RECID=8 STAMP=1153005525 file name=/u01/app/oracle/oradata/DUBAISTB/pdbseed/system01.dbf
datafile 6 switched to datafile copy
input datafile copy RECID=9 STAMP=1153005525 file name=/u01/app/oracle/oradata/DUBAISTB/pdbseed/sysaux01.dbf
datafile 7 switched to datafile copy
input datafile copy RECID=10 STAMP=1153005525 file name=/u01/app/oracle/oradata/DUBAISTB/users01.dbf
datafile 8 switched to datafile copy
input datafile copy RECID=11 STAMP=1153005525 file name=/u01/app/oracle/oradata/DUBAISTB/pdbseed/undotbs01.dbf
datafile 9 switched to datafile copy
input datafile copy RECID=12 STAMP=1153005525 file name=/u01/app/oracle/oradata/DUBAISTB/pdb1/system01.dbf
datafile 10 switched to datafile copy
input datafile copy RECID=13 STAMP=1153005525 file name=/u01/app/oracle/oradata/DUBAISTB/pdb1/sysaux01.dbf
datafile 11 switched to datafile copy
input datafile copy RECID=14 STAMP=1153005525 file name=/u01/app/oracle/oradata/DUBAISTB/pdb1/undotbs01.dbf
datafile 12 switched to datafile copy
input datafile copy RECID=15 STAMP=1153005525 file name=/u01/app/oracle/oradata/DUBAISTB/pdb1/users01.dbf
Finished Duplicate Db at 15-NOV-23
released channel: prmy1
released channel: prmy2
released channel: prmy3
released channel: prmy4

```



```

released channel: stby1
RMAN> 

```

```

RMAN > exit;
RMAN> 

```

```

Recovery Manager complete.
[oracle@OracleVM2 DUBAISTB]$ 

```

Check Standby DB Configuration

RMAN leaves the standby database in MOUNT mode.

```
sqlplus / as sysdba
SQL> select open_mode from v$database;

[oracle@OracleVM2 DUBAISTB]$ sqlplus / as sysdba

SQL*Plus: Release 19.0.0.0.0 - Production on Wed Nov 15 23:27:45 2023
Version 19.3.0.0.0

Copyright (c) 1982, 2019, Oracle. All rights reserved.

Connected to:
Oracle Database 19c Enterprise Edition Release 19.0.0.0.0 - Production
Version 19.3.0.0.0

SQL> select open_mode from v$database;

OPEN_MODE
-----
MOUNTED
```

If you like you can restart the Standby DB without any PFILE, in MOUNT mode:

```
sqlplus / as sysdba
SQL> shutdown immediate;
SQL> startup mount;

SQL> shutdown immediate;
ORA-01109: database not open

Database dismounted.
ORACLE instance shut down.
SQL> startup mount;
ORACLE instance started.

Total System Global Area  998244240 bytes
Fixed Size                  9142160 bytes
Variable Size                260046848 bytes
Database Buffers            725614592 bytes
Redo Buffers                 3440640 bytes
Database mounted.
SQL>
```

```
SQL> select host_name from v$instance;
```

```
SQL> select host_name from v$instance;
```

```
HOST_NAME
```

```
-----
```

```
OracleVM2
```

```
SQL> █
```

Check the role of the standby DB:

```
SQL> select database_role from v$database;
SQL> select database_role from v$database;
```

```
DATABASE_ROLE
```

```
-----
```

```
PHYSICAL STANDBY
```

```
SQL> █
```

```
SQL> show parameter DB_NAME
SQL> show parameter DB_DOMAIN
SQL> show parameter DB_UNIQUE_NAME
SQL> show parameter INSTANCE_NAME
```

```
SQL> show parameter DB_NAME
```

NAME	TYPE	VALUE
db_name	string	dubai

```
SQL> show parameter DB_DOMAIN
```

NAME	TYPE	VALUE
db_domain	string	localdomain

```
SQL> show parameter DB_UNIQUE_NAME
```

NAME	TYPE	VALUE
db_unique_name	string	dubaistb

```
SQL> show parameter INSTANCE_NAME
```

NAME	TYPE	VALUE
instance_name	string	dubaistb

```
SQL> █
```

```
SQL> select name, value from v$parameter where name='db_unique_name';
```

```
SQL> select name, value from v$parameter where name='db_unique_name';

NAME
-----
VALUE
-----
db_unique_name
dubaistb

SQL> █
```

```
SQL> show parameter INSTANCE_NAME;
SQL> show parameter INSTANCE_NAME;

NAME                      TYPE        VALUE
-----                    string      dubaistb
instance_name
SQL> █
```

```
SQL> select instance from v$thread;

SQL> select instance from v$thread;

INSTANCE
-----
dubaistb

SQL> █
```

```
SQL > select name, value from v$parameter where name='instance_name';

SQL> select name, value from v$parameter where name='instance_name';

NAME
-----
VALUE
-----
instance_name
dubaistb

SQL> █
```

The DB_NAME is identical to primary (as it should be).

```
SQL> select name from v$database;
```

```
SQL> select name from v$database;
```

```
NAME
```

```
-----
```

```
DUBAI
```

```
SQL> █
```

```
SQL> select name,value from v$parameter where name='db_name';
```

```
SQL> select name,value from v$parameter where name='db_name';
```

```
NAME
```

```
-----
```

```
VALUE
```

```
-----
```

```
db_name
```

```
dubai
```

```
SQL> █
```

```
SQL> show parameter DB_BLOCK_SIZE
```

```
SQL> show parameter DB_CACHE_SIZE
```

```
SQL> show parameter DB_BLOCK_SIZE
```

NAME	TYPE	VALUE
db_block_size	integer	8192

```
SQL> show parameter DB_CACHE_SIZE
```

NAME	TYPE	VALUE
db_cache_size	big integer	0

```
SQL> show parameter CONTROL_FILES
```

```
SQL> show parameter CONTROL_FILES
```

NAME	TYPE	VALUE
control_files	string	/u01/app/oracle/oradata/DUBAIS TB/control01.ctl

```
SQL> █
```

```
SQL> SELECT log_mode FROM v$database;
```

```
SQL> SELECT log_mode FROM v$database;
```

```
LOG_MODE
```

```
-----
```

```
ARCHIVELOG
```

```
SQL> ■
```

```
SQL> show parameter DB_CREATE_FILE_DEST (location of datafiles)
SQL> show parameter DB_CREATE_ONLINE_LOG_DEST_1 (location of online redo
logs)
SQL> show parameter LOG_ARCHIVE_DEST_1 (location of archive redo logs)
```

```
SQL> show parameter DB_CREATE_FILE_DEST
```

NAME	TYPE	VALUE
db_create_file_dest	string	

```
SQL> show parameter DB_CREATE_ONLINE_LOG_DEST_1
```

NAME	TYPE	VALUE
db_create_online_log_dest_1	string	

```
SQL> show parameter LOG_ARCHIVE_DEST_1
```

NAME	TYPE	VALUE
log_archive_dest_1	string	LOCATION=USE_DB_RECOVERY_FILE_DEST VALID_FOR=(ALL_LOGFILES, ALL_ROLES) DB_UNIQUE_NAME=dubai_stb

```
log_archive_dest_10
```

```
log_archive_dest_11
```

```
log_archive_dest_12
```

```
log_archive_dest_13
```

```
log_archive_dest_14
```

```
log_archive_dest_15
```

```
log_archive_dest_16
```

NAME	TYPE	VALUE
log_archive_dest_17	string	

```
log_archive_dest_18
```

```
log_archive_dest_19
```

```
SQL> ■
```

```
SQL> show parameter DB_RECOVERY_FILE_DEST (Flash Recovery Area)
```

```
SQL> show parameter UNDO_TABLESPACE (location of undo tablespace)
```

```

SQL> show parameter DB_RECOVERY_FILE_DEST
NAME                                     TYPE        VALUE
-----
db_recovery_file_dest                  string      /u01/app/oracle/fast_recovery_
                                         area
db_recovery_file_dest_size            big integer 50G
SQL> show parameter UNDO_TABLESPACE
NAME                                     TYPE        VALUE
-----
undo_tablespace                      string      UNDOTBS1
SQL> ■

```

```

SQL> select name from v$datafile;
SQL> select name from v$datafile;
SQL>
NAME
-----
/u01/app/oracle/oradata/DUBAISTB/system01.dbf
/u01/app/oracle/oradata/DUBAISTB/sysaux01.dbf
/u01/app/oracle/oradata/DUBAISTB/undotbs01.dbf
/u01/app/oracle/oradata/DUBAISTB/pdbseed/system01.dbf
/u01/app/oracle/oradata/DUBAISTB/pdbseed/sysaux01.dbf
/u01/app/oracle/oradata/DUBAISTB/users01.dbf
/u01/app/oracle/oradata/DUBAISTB/pdbseed/undotbs01.dbf
/u01/app/oracle/oradata/DUBAISTB/pdb1/system01.dbf
/u01/app/oracle/oradata/DUBAISTB/pdb1/sysaux01.dbf
/u01/app/oracle/oradata/DUBAISTB/pdb1/undotbs01.dbf
/u01/app/oracle/oradata/DUBAISTB/pdb1/users01.dbf

11 rows selected.

SQL> ■

```

Let us check the actual location of the online redo logs:

```

SQL> col status format a9
SQL> col member format a55
SQL> select g.group#, g.status, f.member
   from v$log g, v$logfile f
  where g.group# = f.group#;

```

```

SQL> col status format a9
SQL> col member format a55
SQL> select g.group#, g.status, f.member
   from v$log g, v$logfile f
  where g.group# = f.group#;
2      3
GROUP# STATUS      MEMBER
-----
1  UNUSED      /u01/app/oracle/oradata/DUBAISTB redo01.log
3  UNUSED      /u01/app/oracle/oradata/DUBAISTB redo03.log
2  CURRENT     /u01/app/oracle/oradata/DUBAISTB redo02.log

```

SQL>

SQL> █

**IT SEEMS THAT BY DEFAULT THE ONLINE REDO LOG FILES ARE IN THE SAME FOLDER AS THE DATA FILES.
THIS IS NOT IDEAL.**

Here is the actual location of the archive redo logs:

```

SQL> select dest_name, status, destination from v$archive_dest where
dest_name='LOG_ARCHIVE_DEST_1';
SQL> select dest_name, status, destination from v$archive_dest where dest_name='LOG_ARCHIVE_DEST_1';

DEST_NAME
-----
STATUS
-----
DESTINATION
-----
LOG_ARCHIVE_DEST_1
VALID
USE_DB_RECOVERY_FILE_DEST

SQL> █

```

```

SQL> select name, value from v$parameter where name='db_recovery_file_dest';
SQL> select name, value from v$parameter where name='db_recovery_file_dest';

NAME
-----
VALUE
-----
db_recovery_file_dest
/u01/app/oracle/fast_recovery_area

```

SQL> █

```
SQL> ARCHIVE LOG LIST
SQL> exit
SQL> ARCHIVE LOG LIST
Database log mode          Archive Mode
Automatic archival        Enabled
Archive destination        USE_DB_RECOVERY_FILE_DEST
Oldest online log sequence 0
Next log sequence to archive 0
Current log sequence       0
SQL> █
```

```
ls -l /u01/app/oracle/fast_recovery_area
[oracle@OracleVM2 DUBAISTB]$ ls -l /u01/app/oracle/fast_recovery_area
total 0
drwxr-x---. 1 oracle oinstall 38 Nov 15 23:18 DUBAISTB
[oracle@OracleVM2 DUBAISTB]$ █
```

```
cd /u01/app/oracle/fast_recovery_area/DUBAISTB
ls -l
[oracle@OracleVM2 ~]$ cd /u01/app/oracle/fast_recovery_area/DUBAISTB
[oracle@OracleVM2 DUBAISTB]$ ls -l
total 0
drwxr-x---. 1 oracle oinstall 20 Nov 15 22:55 archivelog
drwxr-x---. 1 oracle oinstall  0 Nov 15 23:18 onlinelog
[oracle@OracleVM2 DUBAISTB]$ █
```

```
ls -l $ORACLE_HOME/dbs
[oracle@OracleVM2 ~]$ ls -l $ORACLE_HOME/dbs
total 24
-rw-rw----. 1 oracle oinstall 1544 Nov 14 14:40 hc_dubaistb.dat
-rw-r--r--. 1 oracle oinstall 3079 May 14 2015 init.ora
-rw-r-----. 1 oracle oinstall   24 Nov 14 14:31 lkDUBAISTB
-rw-r-----. 1 oracle oinstall 2048 Nov 14 14:30 orapwdubaistb
-rw-r-----. 1 oracle oinstall 5632 Nov 14 14:40 spfiledubaistb.ora
[oracle@OracleVM2 ~]$ █
```

```
sqlplus / as sysdba

SQL> show parameter SGA_TARGET  (size of SGA)
SQL> show parameter PGA_AGGREGATE_TARGET (size of PGA)
SQL> show parameter PROCESSES (max no of processes)
SQL> show parameter SESSIONS (max no of sessions)
```

```

SQL> show parameter SGA_TARGET
NAME                           TYPE        VALUE
-----
sga_target                      big integer 952M
SQL> show parameter PGA_AGGREGATE_TARGET
NAME                           TYPE        VALUE
-----
pga_aggregate_target           big integer 317M
SQL> show parameter PROCESSES
NAME                           TYPE        VALUE
-----
aq_tm_processes                integer    1
db_writer_processes            integer    1
gcs_server_processes           integer    0
global_txn_processes           integer    1
job_queue_processes             integer   20
log_archive_max_processes      integer    4
processes                      integer  300
SQL> show parameter SESSIONS
NAME                           TYPE        VALUE
-----
java_max_sessionspace_size     integer    0
java_soft_sessionspace_limit  integer    0
license_max_sessions           integer    0
license_sessions_warning       integer    0
sessions                       integer 472
shared_server_sessions          integer
SQL> █

```

Let us check the DB role and whether it is open or not:

```

SQL> SELECT OPEN_MODE, DATABASE_ROLE FROM V$DATABASE;
SQL> SELECT OPEN_MODE, DATABASE_ROLE FROM V$DATABASE;
OPEN_MODE          DATABASE_ROLE
-----
MOUNTED           PHYSICAL STANDBY
SQL> █

```

```

SQL> show parameter FAL_SERVER
SQL> show parameter FAL_SERVER
NAME                           TYPE        VALUE
-----
fal_server                     string      DXB
SQL> █

```

.....

Check the location of the data files:

```
SQL> SELECT NAME FROM V$DATAFILE;

SQL> SELECT NAME FROM V$DATAFILE;
|||
NAME
-----
/u01/app/oracle/oradata/DUBAISTB/system01.dbf
/u01/app/oracle/oradata/DUBAISTB/sysaux01.dbf
/u01/app/oracle/oradata/DUBAISTB/undotbs01.dbf
/u01/app/oracle/oradata/DUBAISTB/pdbseed/system01.dbf
/u01/app/oracle/oradata/DUBAISTB/pdbseed/sysaux01.dbf
/u01/app/oracle/oradata/DUBAISTB/users01.dbf
/u01/app/oracle/oradata/DUBAISTB/pdbseed/undotbs01.dbf
/u01/app/oracle/oradata/DUBAISTB/pdb1/system01.dbf
/u01/app/oracle/oradata/DUBAISTB/pdb1/sysaux01.dbf
/u01/app/oracle/oradata/DUBAISTB/pdb1/undotbs01.dbf
/u01/app/oracle/oradata/DUBAISTB/pdb1/users01.dbf

11 rows selected.

SQL> |||
```

Check online redo logs:

```
SQL> set lines 100
SQL> col status format a9
SQL> col member format a55
SQL> select g.group#, g.status, f.member
   from v$log g, v$logfile f
  where g.group# = f.group#;

SQL> set lines 100
SQL> col status format a9
SQL> col member format a55
SQL> select g.group#, g.status, f.member
   from v$log g, v$logfile f
  where g.group# = f.group#;
  2      3
 GROUP# STATUS      MEMBER
 -----
 1  UNUSED    /u01/app/oracle/oradata/DUBAISTB/redo01.log
 3  UNUSED    /u01/app/oracle/oradata/DUBAISTB/redo03.log
 2  CURRENT   /u01/app/oracle/oradata/DUBAISTB/redo02.log

SQL> |||
```

```
SQL> COLUMN MEMBER FORMAT A50
SQL> SELECT GROUP#, MEMBER FROM V$LOGFILE WHERE TYPE='ONLINE' ORDER BY 1,2;
```

```
SQL> COLUMN MEMBER FORMAT A50
SQL> SELECT GROUP#, MEMBER FROM V$LOGFILE WHERE TYPE='ONLINE' ORDER BY 1,2;

GROUP# MEMBER
-----
1 /u01/app/oracle/oradata/DUBAISTB/redo01.log
2 /u01/app/oracle/oradata/DUBAISTB/redo02.log
3 /u01/app/oracle/oradata/DUBAISTB/redo03.log
```

```
SQL> █
```

Here is the location of archive redo logs:

```
SQL> show parameter DB_RECOVERY_FILE_DEST
```

```
SQL> show parameter DB_RECOVERY_FILE_DEST

NAME                                     TYPE        VALUE
-----                                     -----
db_recovery_file_dest                  string      /u01/app/oracle/fast_recovery_
                                         area
db_recovery_file_dest_size            big integer 50G
```

```
cd /u01/app/oracle/fast_recovery_area
ls -l
```

```
[oracle@OracleVM2 ~]$ cd /u01/app/oracle/fast_recovery_area/
[oracle@OracleVM2 fast_recovery_area]$ ls -l
total 0
drwxr-x---. 1 oracle oinstall 38 Nov 14 14:32 DUBAISTB
[oracle@OracleVM2 fast_recovery_area]$ █
```

```
cd DUBAISTB
ls -l
```

```
[oracle@OracleVM2 fast_recovery_area]$ cd DUBAISTB/
[oracle@OracleVM2 DUBAISTB]$ ls -l
total 0
drwxr-x---. 1 oracle oinstall 20 Nov 14 14:31 archivelog
drwxr-x---. 1 oracle oinstall 0 Nov 14 14:32 onlinelog
[oracle@OracleVM2 DUBAISTB]$ █
```

Check Standby Redo Logs:

```
SQL> set lines 200
SQL> COLUMN MEMBER FORMAT A60
SQL> SELECT GROUP#, MEMBER FROM V$LOGFILE WHERE TYPE='STANDBY' ORDER BY 1,2;
SQL> set lines 200
SQL> COLUMN MEMBER FORMAT A60
SQL> SELECT GROUP#, MEMBER FROM V$LOGFILE WHERE TYPE='STANDBY' ORDER BY 1,2;

GROUP# MEMBER
-----
4 /u01/app/oracle/oradata/DUBAISTB/standby_redo01.log
5 /u01/app/oracle/oradata/DUBAISTB/standby_redo02.log
6 /u01/app/oracle/oradata/DUBAISTB/standby_redo03.log
7 /u01/app/oracle/oradata/DUBAISTB/standby_redo04.log

SQL>
```

```
SQL> SELECT GROUP#, THREAD#, SEQUENCE#, ARCHIVED, STATUS FROM V$STANDBY_LOG;
SQL> SELECT GROUP#, THREAD#, SEQUENCE#, ARCHIVED, STATUS FROM V$STANDBY_LOG;

GROUP#    THREAD#    SEQUENCE# ARC STATUS
-----
4          1           12 YES ACTIVE
5          0            0 YES UNASSIGNE
                           D
6          0            0 YES UNASSIGNE
                           D
7          0            0 YES UNASSIGNE
                           D
```

```
SQL> SELECT TYPE, count(*) FROM V$LOGFILE GROUP BY TYPE;
SQL> SELECT TYPE, count(*) FROM V$LOGFILE GROUP BY TYPE;

TYPE      COUNT(*)
-----
ONLINE        3
STANDBY       4

SQL>
```

Let us check some of the important database parameters.

```
SQL> set linesize 500 pages 100
SQL> col name format a30
SQL> col value format a100

SQL> select name, value from v$parameter
where name in ('db_name', 'db_unique_name', 'log_archive_config',
'log_archive_dest_1', 'log_archive_dest_2',
'log_archive_dest_state_1', 'log_archive_dest_state_2',
'remote_login_password_file', 'log_archive_format',
'log_archive_max_processes', 'fal_server',
'log_file_name_convert', 'db_file_name_convert', 'db_recovery_file_dest',
'standby_file_management')
order by 1;
SQL> set linesize 500 pages 100
SQL> col name format a30
SQL> col value format a100
SQL> select name, value from v$parameter
where name in ('db_name', 'db_unique_name', 'log_archive_config',
'log_archive_dest_1', 'log_archive_dest_2',
'log_archive_state_1', 'log_archive_state_2',
'remote_login_password_file', 'log_archive_format',
'log_archive_max_processes', 'fal_server',
'log_file_name_convert', 'standby_file_management')
order by 1;
  2   3   4   5   6   7   8
NAME          VALUE
-----
db_name        dubai
db_unique_name dubaistb
fal_server     DXB
log_archive_config DG_CONFIG=(dubai,dubaistb)
log_archive_dest_1 LOCATION=USE DB RECOVERY FILE_DEST VALID_FOR=(ALL_LOGFILES,ALL_ROLES) DB_UNIQUE_NAME=dubaistb
log_archive_dest_2 SERVICE=DXB ASYNC VALID_FOR=(ONLINE_LOGFILES,PRIMARY_ROLE) DB_UNIQUE_NAME=dubai
log_archive_format %t_%s_%r.arc
log_archive_max_processes 4
log_file_name_convert /dubai/, /dubaistb/, /DUBAI/, /DUBAISTB/
standby_file_management AUTO
10 rows selected.

SQL> ■
```

Start the Apply Redo Process

Now, we can start the Redo Apply process:

```
SQL> ALTER DATABASE RECOVER MANAGED STANDBY DATABASE DISCONNECT FROM SESSION;
SQL> ALTER DATABASE RECOVER MANAGED STANDBY DATABASE DISCONNECT FROM SESSION;

Database altered.

SQL> ■
```

Verify Data Guard Configuration

The quickest way to check the health of the Data Guard process is to issue the following query on primary and standby servers:

```
SQL> SELECT PROTECTION_MODE, PROTECTION_LEVEL, DATABASE_ROLE ROLE,  
SWITCHOVER_STATUS FROM V$DATABASE;
```

Here is the result from the primary:

```
SQL> SELECT PROTECTION_MODE, PROTECTION_LEVEL, DATABASE_ROLE ROLE,  
SWITCHOVER_STATUS FROM V$DATABASE;  
SQL> SELECT PROTECTION_MODE, PROTECTION_LEVEL, DATABASE_ROLE ROLE, SWITCHOVER_STATUS FROM V$DATABASE;  
PROTECTION_MODE      PROTECTION_LEVEL      ROLE          SWITCHOVER_STATUS  
-----  
MAXIMUM PERFORMANCE  MAXIMUM PERFORMANCE  PRIMARY        TO STANDBY  
SQL> ■
```

Here is the result from the standby:

```
SQL> SELECT PROTECTION_MODE, PROTECTION_LEVEL, DATABASE_ROLE ROLE,  
SWITCHOVER_STATUS FROM V$DATABASE;  
SQL> SELECT PROTECTION_MODE, PROTECTION_LEVEL, DATABASE_ROLE ROLE, SWITCHOVER_STATUS FROM V$DATABASE;  
PROTECTION_MODE      PROTECTION_LEVEL      ROLE          SWITCHOVER_STATUS  
-----  
MAXIMUM PERFORMANCE  MAXIMUM PERFORMANCE  PHYSICAL STANDBY NOT ALLOWED  
SQL> ■
```

On the standby server, you can verify that standby redo logs are being transmitted from the primary database and applied to the standby database:

```
SQL> SELECT ROLE, THREAD#, SEQUENCE#, ACTION FROM V$DATAGUARD_PROCESS;
```

```

SQL> SELECT ROLE, THREAD#, SEQUENCE#, ACTION FROM V$DATAGUARD_PROCESS;
ROLE          THREAD#  SEQUENCE# ACTION
-----  -----
RFS ping           1        12 IDLE
RFS async          1        12 IDLE
managed recovery   1        12 APPLYING_LOG
RFS archive         0        0 IDLE
archive local       0        0 IDLE
redo transport timer 0        0 IDLE
gap manager         0        0 IDLE
archive redo        0        0 IDLE
archive redo        0        0 IDLE
archive redo        0        0 IDLE
redo transport monitor 0        0 IDLE
log writer          0        0 IDLE
12 rows selected.

SQL> ■

```

NOW GO BACK TO PRIMARY - The following query checks the sequence of current standby redo log files.

```

SQL> ALTER SESSION SET nls_date_format='DD-MON-YYYY HH24:MI:SS';
SQL> SELECT SEQUENCE#, APPLIED, FIRST_TIME, NEXT_TIME, NAME
FROM V$ARCHIVED_LOG ORDER BY SEQUENCE#;

```

```

SQL> SELECT SEQUENCE#, APPLIED, FIRST_TIME, NEXT_TIME, NAME
  FROM V$ARCHIVED_LOG ORDER BY SEQUENCE#;
  2
  SEQUENCE# APPLIED      FIRST_TIME          NEXT_TIME
  -----
NAME

  7 NO        15-NOV-2023 21:30:53 15-NOV-2023 21:54:03
/u01/app/oracle/product/19.0.0/dbhome_1/dbs/arch1_7_1152996266.dbf

  8 NO        15-NOV-2023 21:54:03 15-NOV-2023 22:04:03
/u01/app/oracle/fast_recovery_area/DUBAI/archivelog/2023_11_15/o1_mf_1_8_lobjdyr
d_.arc

  9 NO        15-NOV-2023 22:04:03 15-NOV-2023 22:12:04
/u01/app/oracle/fast_recovery_area/DUBAI/archivelog/2023_11_15/o1_mf_1_9_lobjkok

  SEQUENCE# APPLIED      FIRST_TIME          NEXT_TIME
  -----
NAME

  b_.arc

  10 NO       15-NOV-2023 22:12:04 15-NOV-2023 23:08:15
/u01/app/oracle/fast_recovery_area/DUBAI/archivelog/2023_11_15/o1_mf_1_10_lobmvo
js_.arc

  11 NO       15-NOV-2023 23:08:15 15-NOV-2023 23:18:44
/u01/app/oracle/fast_recovery_area/DUBAI/archivelog/2023_11_15/o1_mf_1_11_lobngo
09_.arc

  SEQUENCE# APPLIED      FIRST_TIME          NEXT_TIME
  -----
NAME

```

```

11 YES      15-NOV-2023 23:08:15 15-NOV-2023 23:18:44
DXBSTB

12 NO      15-NOV-2023 23:18:44 15-NOV-2023 23:53:07
/u01/app/oracle/fast_recovery_area/DUBAI/archivelog/2023_11_15/o1_mf_1_12_lobph3
jm_.arc

12 YES      15-NOV-2023 23:18:44 15-NOV-2023 23:53:07

SEQUENCE# APPLIED    FIRST_TIME          NEXT_TIME
-----
NAME
-----
DXBSTB

8 rows selected.

SQL> █
```

As seen above, the archive redo log files are now in `/u01/app/oracle/fast_recovery_area/<SID>/archivelod/` folder. The older *.dbf file is also listed (sequence no 7).

Check the last SEQUENCE#. For this example, it was 12.

Force a switch logfile on the primary:

```

SQL> ALTER SYSTEM SWITCH LOGFILE;

SQL> ALTER SYSTEM SWITCH LOGFILE;
System altered.

SQL> █
```

Now, if we issue the same query, the last entry is SEQUENCE = 13:

```

SQL> ALTER SESSION SET nls_date_format='DD-MON-YYYY HH24:MI:SS';
SQL> SELECT SEQUENCE#, APPLIED, FIRST_TIME, NEXT_TIME, NAME
FROM V$ARCHIVED_LOG ORDER BY SEQUENCE#;
```

```

SQL> ALTER SESSION SET nls_date_format='DD-MON-YYYY HH24:MI:SS';
Session altered.

SQL> SELECT SEQUENCE#, APPLIED, FIRST_TIME, NEXT_TIME, NAME
  FROM V$ARCHIVED_LOG ORDER BY SEQUENCE#;
  2
  SEQUENCE# APPLIED      FIRST_TIME          NEXT_TIME
  ----- -----
NAME
-----
    7 NO      15-NOV-2023 21:30:53 15-NOV-2023 21:54:03
/u01/app/oracle/product/19.0.0/dbhome_1/dbs/arch1_7_1152996266.dbf

    8 NO      15-NOV-2023 21:54:03 15-NOV-2023 22:04:03
/u01/app/oracle/fast_recovery_area/DUBAI/archivelog/2023_11_15/o1_mf_1_8_lobjdyr
d_.arc

»    9 NO      15-NOV-2023 22:04:03 15-NOV-2023 22:12:04
/u01/app/oracle/fast_recovery_area/DUBAI/archivelog/2023_11_15/o1_mf_1_9_lobjkok

  SEQUENCE# APPLIED      FIRST_TIME          NEXT_TIME
  ----- -----
NAME
-----
b_.arc

    10 NO     15-NOV-2023 22:12:04 15-NOV-2023 23:08:15
/u01/app/oracle/fast_recovery_area/DUBAI/archivelog/2023_11_15/o1_mf_1_10_lobmv0
js_.arc

    11 NO     15-NOV-2023 23:08:15 15-NOV-2023 23:18:44
/u01/app/oracle/fast_recovery_area/DUBAI/archivelog/2023_11_15/o1_mf_1_11_lobngo
09_.arc

```

```

SEQUENCE# APPLIED    FIRST_TIME          NEXT_TIME
-----
NAME

           11 YES      15-NOV-2023 23:08:15 15-NOV-2023 23:18:44
DXBSTB

           12 NO       15-NOV-2023 23:18:44 15-NOV-2023 23:53:07
/u01/app/oracle/fast_recovery_area/DUBAI/archivelog/2023_11_15/o1_mf_1_12_lobph3
jm_.arc

           12 YES      15-NOV-2023 23:18:44 15-NOV-2023 23:53:07
»
SEQUENCE# APPLIED    FIRST_TIME          NEXT_TIME
-----
NAME

DXBSTB

           13 NO       15-NOV-2023 23:53:07 16-NOV-2023 08:19:44
DXBSTB

           13 NO       15-NOV-2023 23:53:07 16-NOV-2023 08:19:44
/u01/app/oracle/fast_recovery_area/DUBAI/archivelog/2023_11_16/o1_mf_1_13_locn50
om_.arc

10 rows selected.

SQL> ■

```

NOW GO BACK TO STANDBY SERVER – CHECK IF SEQUENCE 13 IS APPLIED.

```

SQL> ALTER SESSION SET nls_date_format='DD-MON-YYYY HH24:MI:SS';
SQL> SELECT SEQUENCE#, APPLIED, FIRST_TIME, NEXT_TIME, NAME
FROM V$ARCHIVED_LOG ORDER BY SEQUENCE#;

```

```
SQL> ALTER SESSION SET nls_date_format='DD-MON-YYYY HH24:MI:SS';

Session altered.

SQL> SELECT SEQUENCE#, APPLIED, FIRST_TIME, NEXT_TIME, NAME
  FROM V$ARCHIVED_LOG ORDER BY SEQUENCE#;
  2
  SEQUENCE# APPLIED      FIRST_TIME          NEXT_TIME
  ----- -----
NAME
-----
  11 YES      15-NOV-2023 23:08:15 15-NOV-2023 23:18:44
/u01/app/oracle/fast_recovery_area/DUBAISTB/archivelog/2023_11_15/o1_mf_1_11_lob
npl3m_.arc

  12 YES      15-NOV-2023 23:18:44 15-NOV-2023 23:53:07
/u01/app/oracle/fast_recovery_area/DUBAISTB/archivelog/2023_11_15/o1_mf_1_12_lob
ph3rk_.arc

  13 IN-MEMORY 15-NOV-2023 23:53:07 16-NOV-2023 08:19:44

SEQUENCE# APPLIED      FIRST_TIME          NEXT_TIME
  ----- -----
NAME
-----
/u01/app/oracle/fast_recovery_area/DUBAISTB/archivelog/2023_11_16/o1_mf_1_13_lob
n50wn_.arc

SQL> █
```

You can also issue the following query to see the next sequence no:

```
SQL> SELECT ROLE, THREAD#, SEQUENCE#, ACTION FROM V$DATAGUARD_PROCESS;
```

```

SQL> SELECT ROLE, THREAD#, SEQUENCE#, ACTION FROM V$DATAGUARD_PROCESS;

ROLE                THREAD#  SEQUENCE# ACTION
-----              -----
RFS ping                  1        14 IDLE
RFS async                 1        14 IDLE
managed recovery           1        14 APPLYING_LOG
RFS archive                0        0 IDLE
archive local               0        0 IDLE
redo transport timer        0        0 IDLE
gap manager                 0        0 IDLE
archive redo                 0        0 IDLE
archive redo                 0        0 IDLE
archive redo                 0        0 IDLE
redo transport monitor       0        0 IDLE

ROLE                THREAD#  SEQUENCE# ACTION
-----              -----
log writer                   0        0 IDLE

12 rows selected.

SQL> 
```

Now it is time to enable Flashback on the standby server as well. To do this, stop the Redo Apply Recovery on the Standby server:

```

SQL> alter database recover managed standby database cancel;
SQL> alter database recover managed standby database cancel;

Database altered.

SQL> 
```

Enable Flashback on standby server:

```

SQL> alter database flashback on;

SQL> alter database recover managed standby database cancel;

Database altered.

SQL> alter database flashback on;

Database altered.

SQL> 
```

Now, start the Redo Apply Recovery on the Standby server:

```
SQL> ALTER DATABASE RECOVER MANAGED STANDBY DATABASE DISCONNECT FROM SESSION;
SQL> ALTER DATABASE RECOVER MANAGED STANDBY DATABASE DISCONNECT FROM SESSION;
Database altered.

SQL> █
```

Check the running Data Guard processes:

```
SQL> SELECT PROCESS, STATUS, THREAD#, SEQUENCE#, BLOCK#, BLOCKS FROM V$MANAGED_STANDBY;
SQL> SELECT PROCESS, STATUS, THREAD#, SEQUENCE#, BLOCK#, BLOCKS FROM V$MANAGED_STANDBY;

PROCESS      STATUS     THREAD#  SEQUENCE#    BLOCK#    BLOCKS
-----      -----
ARCH        CONNECTED      0          0          0          0
DGRD        ALLOCATED      0          0          0          0
DGRD        ALLOCATED      0          0          0          0
ARCH        CONNECTED      0          0          0          0
ARCH        CLOSING        1         13       118784        783
ARCH        CONNECTED      0          0          0          0
RFS         IDLE           1          0          0          0
RFS         IDLE           1         14        334          1
RFS         IDLE           0          0          0          0
MRP0        APPLYING_LOG   1         14        334      409600

10 rows selected.

SQL> █
```

It is interesting that a control file is created on the standby server:

```
SQL> exit
ls -l /u01/app/oracle/oradata/DUBAISTB/
```

```
SQL> exit
Disconnected from Oracle Database 19c Enterprise Edition Release 19.0.0.0.0 - Production
Version 19.3.0.0.0
[oracle@OracleVM2 ~]$ ls -l /u01/app/oracle/oradata/DUBAISTB/
total 3223468
-rw-r-----. 1 oracle oinstall 18726912 Nov 16 00:00 control01.ctl
drwxr-xr-x. 1 oracle oinstall 96 Nov 15 23:18 pdb1
drwxr-xr-x. 1 oracle oinstall 74 Nov 15 23:18 pdbseed
-rw-r-----. 1 oracle oinstall 209715712 Nov 15 23:18 redo01.log
-rw-r-----. 1 oracle oinstall 209715712 Nov 15 23:18 redo02.log
-rw-r-----. 1 oracle oinstall 209715712 Nov 15 23:18 redo03.log
-rw-r-----. 1 oracle oinstall 209715712 Nov 16 00:00 standby_redo01.log
-rw-r-----. 1 oracle oinstall 209715712 Nov 15 23:18 standby_redo02.log
-rw-r-----. 1 oracle oinstall 209715712 Nov 15 23:18 standby_redo03.log
-rw-r-----. 1 oracle oinstall 209715712 Nov 15 23:19 standby_redo04.log
-rw-r-----. 1 oracle oinstall 576724992 Nov 15 23:57 sysaux01.dbf
-rw-r-----. 1 oracle oinstall 943726592 Nov 15 23:57 system01.dbf
-rw-r-----. 1 oracle oinstall 288366592 Nov 15 23:57 undotbs01.dbf
-rw-r-----. 1 oracle oinstall 5251072 Nov 15 23:57 users01.dbf
[oracle@OracleVM2 ~]$
```

Also, the Flashback recovery area is now populated:

```
cd /u01/app/oracle/fast_recovery_area/
ls -l

[oracle@OracleVM2 ~]$ cd /u01/app/oracle/fast_recovery_area/
[oracle@OracleVM2 fast_recovery_area]$ ls -l
total 0
drwxr-x---. 1 oracle oinstall 56 Nov 15 23:56 DUBAISTB
[oracle@OracleVM2 fast_recovery_area]$
```

The log file is

```
/u01/app/oracle/diag/rdbms/<DB_UNIQUE_NAME>/<SID>/trace/alert_<SID>.log
tail -20
/u01/app/oracle/diag/rdbms/dubaistb/dubaistb/trace/alert_dubaistb.log
```

Testing Switchover and Switchback without Data Guard Broker

Switchover is a controlled process to switch roles from primary to standby.

For primary and standby, check the current status:

```
sqlplus / as sysdba
SQL> select status,instance_name,database_role, protection_mode from
v$database,v$instance;
```

```
SQL> select status,instance_name,database_role, protection_mode from v$database,v$instance;
STATUS      INSTANCE_NAME      DATABASE_ROLE      PROTECTION_MODE
-----      -----
OPEN        dubai            PRIMARY           MAXIMUM PERFORMANCE
```

```
SQL> select status,instance_name,database_role, protection_mode from v$database,v$instance;
STATUS      INSTANCE_NAME      DATABASE_ROLE      PROTECTION_MODE
-----      -----
MOUNTED    dubaistb          PHYSICAL STANDBY  MAXIMUM PERFORMANCE
SQL> █
```

1. First, on the primary server, confirm that synchronization is ready to perform a switchover:

```
SQL> select dest_name,status,error from v$archive_dest where
dest_name='LOG_ARCHIVE_DEST_2';
SQL> select dest_name,status,error from v$archive_dest where dest_name='LOG_ARCHIVE_DEST_2';
DEST_NAME
-----
STATUS      ERROR
-----
LOG_ARCHIVE_DEST_2
VALID

SQL> █
```

```
SQL> select name,value from v$parameter where name='log_archive_dest_2';
SQL> select name,value from v$parameter where name='log_archive_dest_2';
NAME
-----
VALUE
-----
log_archive_dest_2
SERVICE=DXB$TB ASYNC VALID_FOR=(ONLINE_LOGFILES,PRIMARY_ROLE) DB_UNIQUE_NAME=dubaistb

SQL> █
```

Note the “db_unique_name” of the destination is used below:

```
SQL> alter database switchover to dubaistb verify;
```

```
SQL> alter database switchover to dubaistb verify;  
Database altered.  
SQL> █
```

Check if there are any gaps:

```
SQL> select status, gap_status  
from v$archive_dest_status  
where dest_id = 2;  
SQL> select status, gap_status  
from v$archive_dest_status  
where dest_id = 2; 2 3  
  
STATUS      GAP_STATUS  
-----  
VALID        NO GAP  
  
SQL> █
```

2 – Using the primary server, switchover to standby database:

```
SQL> alter database switchover to dubaistb;  
SQL> alter database switchover to dubaistb;  
Database altered.  
SQL> █
```

3 – Now, go to old standby (VM2), and open the database:

```
SQL> alter database open;  
SQL> alter database open;  
Database altered.  
SQL> █
```

4- Go back to old primary (VM1) and start the DB in MOUNT mode and initiate the Redo Apply:

```
SQL> startup mount;  
SQL> alter database recover managed standby database disconnect;
```

```
SQL> startup mount;
ORACLE instance started.

Total System Global Area  998244240 bytes
Fixed Size                  9142160 bytes
Variable Size                281018368 bytes
Database Buffers            704643072 bytes
Redo Buffers                 3440640 bytes
Database mounted.
SQL> alter database recover managed standby database disconnect;

Database altered.

SQL> █
```

5 – Check the current roles:

```
SQL> select status,instance_name,database_role, protection_mode from
v$database,v$instance;
```

On VM1, you will see the following:

```
SQL> select status,instance_name,database_role, protection_mode from v$database,v$instance;

STATUS      INSTANCE_NAME      DATABASE_ROLE      PROTECTION_MODE
-----      -----          -----          -----
MOUNTED      dubai           PHYSICAL STANDBY  MAXIMUM PERFORMANCE

SQL> █
```

On VM2, you will see the following:

```
SQL> select status,instance_name,database_role, protection_mode from
v$database,v$instance;
SQL> select status,instance_name,database_role, protection_mode from v$database,v$instance;

STATUS      INSTANCE_NAME      DATABASE_ROLE      PROTECTION_MODE
-----      -----          -----          -----
OPEN        dubaistb         PRIMARY          MAXIMUM PERFORMANCE

SQL> █
```

To switch back the roles, you follow the same procedure.

Testing Failover without Data Guard Broker

To simulate a failover, we will shut down the primary server:

```
oracle$ exit
azureuser$ sudo su -
root# shutdown now
[oracle@OracleVM1 oracle]$ exit
logout
[azureuser@OracleVM1 ~]$ sudo su -
[sudo] password for azureuser:
Last login: Wed Nov 15 15:50:18 UTC 2023
[root@OracleVM1 ~]# shutdown now
```

On the standby server do the following:

```
SQL> SELECT PROTECTION_MODE, PROTECTION_LEVEL, DATABASE_ROLE ROLE,
SWITCHOVER_STATUS FROM V$DATABASE;
SQL> SELECT PROTECTION_MODE, PROTECTION_LEVEL, DATABASE_ROLE ROLE, SWITCHOVER_STATUS FROM V$DATABASE;
PROTECTION_MODE      PROTECTION_LEVEL      ROLE          SWITCHOVER_STATUS
-----              -----              -----
MAXIMUM PERFORMANCE  MAXIMUM PERFORMANCE  PHYSICAL STANDBY NOT ALLOWED
SQL> ■
```

```
SQL> ALTER DATABASE RECOVER MANAGED STANDBY DATABASE FINISH;
SQL> ALTER DATABASE ACTIVATE STANDBY DATABASE;
```

```
SQL> ALTER DATABASE RECOVER MANAGED STANDBY DATABASE FINISH;
```

```
Database altered.
```

```
SQL> ALTER DATABASE ACTIVATE STANDBY DATABASE;
```

```
Database altered.
```

```
SQL> ■
```

```
SQL> ALTER DATABASE OPEN;
```

```
SQL> SELECT PROTECTION_MODE, PROTECTION_LEVEL, DATABASE_ROLE ROLE, OPEN_MODE,
SWITCHOVER_STATUS FROM V$DATABASE;
```

```

SQL> ALTER DATABASE OPEN;
Database altered.

SQL> SELECT PROTECTION_MODE, PROTECTION_LEVEL, DATABASE_ROLE ROLE, OPEN_MODE, SWITCHOVER_STATUS FROM V$DATABASE;
PROTECTION_MODE      PROTECTION_LEVEL      ROLE          OPEN_MODE
-----              -----              -----
SWITCHOVER_STATUS
-----              -----              -----
MAXIMUM PERFORMANCE  MAXIMUM PERFORMANCE  PRIMARY        READ WRITE
NOT ALLOWED

SQL> ■

```

Once the old primary is back online, we can start it in MOUNT mode and initiate Redo Apply process.

Do the following on the old primary:

```

azureuser$ sudo su - oracle

lsnrctl start
[oracle@OracleVM1 ~]$ lsnrctl start

LSNRCTL for Linux: Version 19.0.0.0.0 - Production on 16-NOV-2023 11:04:32
Copyright (c) 1991, 2019, Oracle. All rights reserved.

Starting /u01/app/oracle/product/19.0.0/dbhome_1/bin/tnslsnr: please wait...

TNSLSNR for Linux: Version 19.0.0.0.0 - Production
System parameter file is /u01/app/oracle/product/19.0.0/dbhome_1/network/admin/listener.ora
Log messages written to /u01/app/oracle/diag/tnslsnr/OracleVM1/listener/alert/log.xml
Listening on: (DESCRIPTION=(ADDRESS=(PROTOCOL=tcp)(HOST=OracleVM1)(PORT=1521)))
Listening on: (DESCRIPTION=(ADDRESS=(PROTOCOL=ipc)(KEY=EXTPROC1521)))

Connecting to (DESCRIPTION=(ADDRESS=(PROTOCOL=TCP)(HOST=OracleVM1)(PORT=1521)))
STATUS of the LISTENER
-----
Alias           LISTENER
Version         TNSLSNR for Linux: Version 19.0.0.0.0 - Production
Start Date     16-NOV-2023 11:04:34
Uptime          0 days 0 hr. 0 min. 0 sec
Trace Level    off
Security        ON: Local OS Authentication
SNMP            OFF
Listener Parameter File  /u01/app/oracle/product/19.0.0/dbhome_1/network/admin/listener.ora
Listener Log File   /u01/app/oracle/diag/tnslsnr/OracleVM1/listener/alert/log.xml
Listening Endpoints Summary...
  (DESCRIPTION=(ADDRESS=(PROTOCOL=tcp)(HOST=OracleVM1)(PORT=1521)))
  (DESCRIPTION=(ADDRESS=(PROTOCOL=ipc)(KEY=EXTPROC1521)))
Services Summary...
Service "dubaiserv" has 1 instance(s).
  Instance "dubai", status UNKNOWN, has 1 handler(s) for this service...
The command completed successfully
[oracle@OracleVM1 ~]$ ■

```

```
sqlplus / as sysdba
```

```
ast login: Thu Nov 16 07:33:53 2023 from vm000001.internal.cloudapp.net
azureuser@OracleVM1 ~]$ sudo su - oracle
[sudo] password for azureuser:
ast login: Thu Nov 16 07:34:28 UTC 2023 on pts/0
oracle@OracleVM1 ~]$ sqlplus / as sysdba

SQL*Plus: Release 19.0.0.0.0 - Production on Thu Nov 16 11:02:43 2023
Version 19.3.0.0.0

Copyright (c) 1982, 2019, Oracle. All rights reserved.

Connected to an idle instance.

SQL> █
```

```
SQL> startup mount;
SQL> SELECT PROTECTION_MODE, PROTECTION_LEVEL, DATABASE_ROLE ROLE,
SWITCHOVER_STATUS FROM V$DATABASE;
SQL> SELECT PROTECTION_MODE, PROTECTION_LEVEL, DATABASE_ROLE ROLE, SWITCHOVER_STATUS FROM V$DATABASE;
PROTECTION_MODE      PROTECTION_LEVEL      ROLE          SWITCHOVER_STATUS
-----  -----  -----  -----
MAXIMUM PERFORMANCE    UNPROTECTED        PRIMARY       NOT ALLOWED

SQL> █
```

We need the SCN at which the old standby database became the new primary database:

```
SQL> SELECT TO_CHAR(STANDBY_BECAME_PRIMARY_SCN) FROM V$DATABASE;
SQL> SELECT TO_CHAR(STANDBY_BECAME_PRIMARY_SCN) FROM V$DATABASE;
TO_CHAR(STANDBY_BECAME_PRIMARY_SCN)
-----
2477447

SQL> █
```

```
SQL> FLASHBACK DATABASE TO SCN 2477447;
```

```
SQL> FLASHBACK DATABASE TO SCN 2477447;
```

```
Flashback complete.
```

```
SQL> █
```

```
SQL> ALTER DATABASE CONVERT TO PHYSICAL STANDBY;
```

```
SQL> ALTER DATABASE CONVERT TO PHYSICAL STANDBY;  
Database altered.  
SQL> █
```

```
SQL> SHUTDOWN IMMEDIATE;  
SQL> STARTUP MOUNT;  
SQL> SHUTDOWN IMMEDIATE;  
ORA-01109: database not open  
  
Database dismounted.  
ORACLE instance shut down.  
SQL> STARTUP MOUNT;  
ORACLE instance started.  
  
Total System Global Area  998244240 bytes  
Fixed Size                  9142160 bytes  
Variable Size              281018368 bytes  
Database Buffers           704643072 bytes  
Redo Buffers                3440640 bytes  
Database mounted.  
SQL> █
```

NOW GO BACK TO THE CURRENT PRIMARY SERVER.

Perform a log switch to ensure the standby database begins receiving redo data from the new primary database, and verify it was sent successfully.

```
SQL> ALTER SYSTEM SWITCH LOGFILE;
```

```
SQL> ALTER SYSTEM SWITCH LOGFILE;  
System altered.  
SQL> █
```

GO BACK TO THE RESTORED SERVER (OLD PRIMARY, NEW PHYSICAL STANDBY), and do the following:

```
SQL> ALTER DATABASE RECOVER MANAGED STANDBY DATABASE DISCONNECT FROM SESSION;
```

```
SQL> ALTER DATABASE RECOVER MANAGED STANDBY DATABASE DISCONNECT FROM SESSION;  
Database altered.  
SQL> █
```

```

SQL> set lines 220
SQL> SELECT PROTECTION_MODE, PROTECTION_LEVEL, DATABASE_ROLE ROLE, OPEN_MODE,
SWITCHOVER STATUS FROM V$DATABASE;
SQL> set lines 220
SQL> SELECT PROTECTION_MODE, PROTECTION_LEVEL, DATABASE_ROLE ROLE, OPEN_MODE, SWITCHOVER_STATUS FROM V$DATABASE;

PROTECTION_MODE      PROTECTION_LEVEL      ROLE          OPEN_MODE           SWITCHOVER_STATUS
-----              -----              -----          -----           -----
MAXIMUM PERFORMANCE  MAXIMUM PERFORMANCE  PHYSICAL STANDBY MOUNTED           TO PRIMARY

SQL> ■

```

Now, if required, you can do a “SwitchOver”.

Configuring Data Guard Broker

In Oracle 19c, the Data Guard Broker takes care of the “LOG_ARCHIVE_DEST_2” parameter.

If you have “LOG_ARCHIVE_DEST_2” parameter defined, you will get the following error when configuring Data Guard Broker:

Error: ORA-16698: member has a LOG_ARCHIVE_DEST_n parameter with SERVICE attribute set

- (1) Ensure that the primary DB is up and running.

```

SQL> set lines 200
SQL> SELECT DB_UNIQUE_NAME, OPEN_MODE, PROTECTION_MODE, PROTECTION_LEVEL,
DATABASE_ROLE ROLE, SWITCHOVER_STATUS FROM V$DATABASE;
SQL> set lines 200
SQL> SELECT DB_UNIQUE_NAME, OPEN_MODE, PROTECTION_MODE, PROTECTION_LEVEL, DATABASE_ROLE ROLE, SWITCHOVER_STATUS FROM V$DATABASE;

DB_UNIQUE_NAME          OPEN_MODE          PROTECTION_MODE      PROTECTION_LEVEL      ROLE          SWITCHOVER_STATUS
-----                  -----          -----          -----          -----          -----
dubai                   READ WRITE        MAXIMUM PERFORMANCE  MAXIMUM PERFORMANCE  PRIMARY       TO STANDBY

SQL> ■

```

Also, ensure that the standby DB is up and running.

```

SQL> set lines 200
SQL> SELECT DB_UNIQUE_NAME, OPEN_MODE, PROTECTION_MODE, PROTECTION_LEVEL,
DATABASE_ROLE ROLE, SWITCHOVER_STATUS FROM V$DATABASE;
SQL> set lines 200
SQL> SELECT DB_UNIQUE_NAME, OPEN_MODE, PROTECTION_MODE, PROTECTION_LEVEL, DATABASE_ROLE ROLE, SWITCHOVER_STATUS FROM V$DATABASE;

DB_UNIQUE_NAME          OPEN_MODE          PROTECTION_MODE      PROTECTION_LEVEL      ROLE          SWITCHOVER_STATUS
-----                  -----          -----          -----          -----          -----
dubaistb                MOUNTED          MAXIMUM PERFORMANCE  MAXIMUM PERFORMANCE  PHYSICAL STANDBY NOT ALLOWED

SQL> ■

```

- (2) You need to clear the value for LOG_ARCHIVE_DEST_2 both on the primary and on the standby database.

```
SQL> select name, value from v$parameter where name='log_archive_dest_2';
SQL> alter system set LOG_ARCHIVE_DEST_2='' SCOPE=BOTH sid='*';
SQL> select name, value from v$parameter where name='log_archive_dest_2';

NAME
-----
VALUE
-----
log_archive_dest_2
SERVICE=DXB$TB ASYNC VALID_FOR=(ONLINE_LOGFILES,PRIMARY_ROLE) DB_UNIQUE_NAME=dub
aistb

SQL> alter system set LOG_ARCHIVE_DEST_2='' SCOPE=BOTH sid='*';
System altered.

SQL> [REDACTED]
```

```
SQL> select name, value from v$parameter where name='log_archive_dest_2';

NAME
-----
VALUE
-----
log_archive_dest_2
SERVICE=DXB ASYNC VALID_FOR=(ONLINE_LOGFILES,PRIMARY_ROLE) DB_UNIQUE_NAME=dubai

SQL> alter system set LOG_ARCHIVE_DEST_2='' SCOPE=BOTH sid='*';
System altered.

SQL> [REDACTED]
```

- (3) Set the parameter dg_broker_config_file1 and dg_broker_config_file2 on all instances of primary and standby.

The default location of the broker configuration file is \$ORACLE_HOME/dbs or \$ORACLE_HOME/database.

You must create the directory for Data Guard Broker Configuration files. In this example, we will use the default location of \$ORACLE_HOME/dbs

The configuration file format: drn<db_unique_name>.dat

Do the following on primary:

```
SQL > ALTER SYSTEM SET dg_broker_config_file1 =
  '/u01/app/oracle/product/19.0.0/dbhome_1/dbs/dr1dubai.dat' scope=both
  sid='*';
SQL> ALTER SYSTEM SET dg_broker_config_file2 =
  '/u01/app/oracle/product/19.0.0/dbhome_1/dbs/dr2dubai.dat' scope=both
  sid='*';
SQL> ALTER SYSTEM SET dg_broker_config_file1 = '/u01/app/oracle/product/19.0.0/dbhome_1/dbs/dr1dubai.dat' scope=both sid='*';
System altered.
SQL> ALTER SYSTEM SET dg_broker_config_file2 = '/u01/app/oracle/product/19.0.0/dbhome_1/dbs/dr2dubai.dat' scope=both sid='*';
System altered.
```

Do the following on standby:

```
SQL > ALTER SYSTEM SET dg_broker_config_file1 =
  '/u01/app/oracle/product/19.0.0/dbhome_1/dbs/dr1dubaistb.dat' scope=both
  sid='*';
SQL> ALTER SYSTEM SET dg_broker_config_file2 =
  '/u01/app/oracle/product/19.0.0/dbhome_1/dbs/dr2dubaistb.dat' scope=both
  sid='*';
SQL> ALTER SYSTEM SET dg_broker_config_file1 = '/u01/app/oracle/product/19.0.0/dbhome_1/dbs/dr1dubaistb.dat' scope=both sid='*';
System altered.
SQL> ALTER SYSTEM SET dg_broker_config_file2 = '/u01/app/oracle/product/19.0.0/dbhome_1/dbs/dr2dubaistb.dat' scope=both sid='*';
System altered.
```

If you have ASM, set as follows:

```
SQL> ALTER SYSTEM SET dg_broker_config_file1 = '<+disk group/file_name>.dat'
  scope=both sid='*';
SQL> ALTER SYSTEM SET dg_broker_config_file2 = '<+disk group/file_name>.dat'
  scope=both sid='*';
```

Now, we need to modify the listener.ora files on both primary and standby servers. Then restart the listeners.

The new listener should be named <DB_UNIQUE_NAME_DGMGRL.domain>

Here is the new listener file on primary:

```
more $ORACLE_HOME/network/admin/listener.ora
LISTENER =
```

```

(DESCRIPTION_LIST =
  (DESCRIPTION =
    (ADDRESS = (PROTOCOL = TCP) (HOST = OracleVM1) (PORT = 1521))
    (ADDRESS = (PROTOCOL = IPC) (KEY = EXTPROC1521))
  )
)
SID_LIST_LISTENER =
  (SID_LIST =
    (SID_DESC =
      (GLOBAL_DBNAME = dubaiserv)
      (ORACLE_HOME = /u01/app/oracle/product/19.0.0/dbhome_1)
      (SID_NAME = dubai)
    )
    (SID_DESC =
      (GLOBAL_DBNAME = dubai_DGMGRL.localdomain)
      (ORACLE_HOME = /u01/app/oracle/product/19.0.0/dbhome_1)
      (SID_NAME = dubai)
    )
  )
ADR_BASE_LISTENER = /u01/app/oracle

```

Now, let's restart the listener on the primary server:

```

lsnrctl stop
lsnrctl start

```

```

[oracle@OracleVM1 admin]$ lsnrctl stop

LSNRCTL for Linux: Version 19.0.0.0.0 - Production on 23-NOV-2023 21:51:36

Copyright (c) 1991, 2019, Oracle. All rights reserved.

Connecting to (DESCRIPTION=(ADDRESS=(PROTOCOL=TCP)(HOST=OracleVM1)(PORT=1521)))
The command completed successfully
[oracle@OracleVM1 admin]$

```

```
[oracle@OracleVM1 admin]$ lsnrctl start
LSNRCTL for Linux: Version 19.0.0.0.0 - Production on 23-NOV-2023 22:53:24
Copyright (c) 1991, 2019, Oracle. All rights reserved.

Starting /u01/app/oracle/product/19.0.0/dbhome_1/bin/tnslsnr: please wait...

TNSLSNR for Linux: Version 19.0.0.0.0 - Production
System parameter file is /u01/app/oracle/product/19.0.0/dbhome_1/network/admin/listener.ora
Log messages written to /u01/app/oracle/diag/tnslsnr/OracleVM1/listener/alert/log.xml
Listening on: (DESCRIPTION=(ADDRESS=(PROTOCOL=tcp)(HOST=OracleVM1)(PORT=1521)))
Listening on: (DESCRIPTION=(ADDRESS=(PROTOCOL=ipc)(KEY=EXTPROC1521)))

Connecting to (DESCRIPTION=(ADDRESS=(PROTOCOL=TCP)(HOST=OracleVM1)(PORT=1521)))
STATUS of the LISTENER
-----
Alias                      LISTENER
Version        TNSLSNR for Linux: Version 19.0.0.0.0 - Production
Start Date     23-NOV-2023 22:53:24
Uptime         0 days 0 hr. 0 min. 0 sec
Trace Level   off
Security       ON: Local OS Authentication
SNMP           OFF
Listener Parameter File /u01/app/oracle/product/19.0.0/dbhome_1/network/admin/listener.ora
Listener Log File  /u01/app/oracle/diag/tnslsnr/OracleVM1/listener/alert/log.xml
Listening Endpoints Summary...
  (DESCRIPTION=(ADDRESS=(PROTOCOL=tcp)(HOST=OracleVM1)(PORT=1521)))
  (DESCRIPTION=(ADDRESS=(PROTOCOL=ipc)(KEY=EXTPROC1521)))
Services Summary...
Service "dubai_DGMGRL.localdomain" has 1 instance(s).
  Instance "dubai", status UNKNOWN, has 1 handler(s) for this service...
Service "dubaiserv" has 1 instance(s).
  Instance "dubai", status UNKNOWN, has 1 handler(s) for this service...
The command completed successfully
[oracle@OracleVM1 admin]$
```

Here is the listener on the standby server:

```
more $ORACLE_HOME/network/admin/listener.ora
```

```
LISTENER =
(DESCRIPTION_LIST =
(DESCRIPTION =
(ADDRESS = (PROTOCOL = TCP) (HOST = OracleVM2) (PORT = 1521))
(ADDRESS = (PROTOCOL = IPC) (KEY = EXTPROC1521))
)
)
SID_LIST_LISTENER =
(SID_LIST =
(SID_DESC =
(GLOBAL_DBNAME = dubaistbserv)
(ORACLE_HOME = /u01/app/oracle/product/19.0.0/dbhome_1)
(SID_NAME = dubaistb)
)
(SID_DESC =
```

```

        (GLOBAL_DBNAME = dubaistb_DGMGRl.localdomain)
        (ORACLE_HOME = /u01/app/oracle/product/19.0.0/dbhome_1)
        (SID_NAME = dubaistb)
    )
)
ADR_BASE_LISTENER = /u01/app/oracle

```

Now, let's restart the listener on the standby server:

```
lsnrctl stop
lsnrctl start
```

```
[oracle@OracleVM2 admin]$ lsnrctl stop

LSNRCTL for Linux: Version 19.0.0.0.0 - Production on 23-NOV-2023 21:45:43

Copyright (c) 1991, 2019, Oracle. All rights reserved.

Connecting to (DESCRIPTION=(ADDRESS=(PROTOCOL=TCP)(HOST=OracleVM2)(PORT=1521)))
The command completed successfully
```

```
[oracle@OracleVM2 admin]$ lsnrctl start

LSNRCTL for Linux: Version 19.0.0.0.0 - Production on 23-NOV-2023 22:50:48

Copyright (c) 1991, 2019, Oracle. All rights reserved.

Starting /u01/app/oracle/product/19.0.0/dbhome_1/bin/tnslsnr: please wait...

TNSLSNR for Linux: Version 19.0.0.0.0 - Production
System parameter file is /u01/app/oracle/product/19.0.0/dbhome_1/network/admin/listener.ora
Log messages written to /u01/app/oracle/diag/tnslsnr/OracleVM2/listener/alert/log.xml
Listening on: (DESCRIPTION=(ADDRESS=(PROTOCOL=tcp)(HOST=OracleVM2)(PORT=1521)))
Listening on: (DESCRIPTION=(ADDRESS=(PROTOCOL=ipc)(KEY=EXTPROC1521)))

Connecting to (DESCRIPTION=(ADDRESS=(PROTOCOL=TCP)(HOST=OracleVM2)(PORT=1521)))
STATUS of the LISTENER
-----
Alias                      LISTENER
Version                    TNSLSNR for Linux: Version 19.0.0.0.0 - Production
Start Date                 23-NOV-2023 22:50:48
Uptime                     0 days 0 hr. 0 min. 0 sec
Trace Level                off
Security                   ON: Local OS Authentication
SNMP                       OFF
Listener Parameter File   /u01/app/oracle/product/19.0.0/dbhome_1/network/admin/listener.ora
Listener Log File          /u01/app/oracle/diag/tnslsnr/OracleVM2/listener/alert/log.xml
Listening Endpoints Summary...
  (DESCRIPTION=(ADDRESS=(PROTOCOL=tcp)(HOST=OracleVM2)(PORT=1521)))
  (DESCRIPTION=(ADDRESS=(PROTOCOL=ipc)(KEY=EXTPROC1521)))
Services Summary...
Service "dubaistb_DGMGRl.localdomain" has 1 instance(s).
  Instance "dubaistb", status UNKNOWN, has 1 handler(s) for this service...
Service "dubaistbserv" has 1 instance(s).
  Instance "dubaistb", status UNKNOWN, has 1 handler(s) for this service...
The command completed successfully
[oracle@OracleVM2 admin]$
```

We do not need to change tnsnames.ora since Data Guard Broker knows how to construct the connection string properly.

Do the following both on the Primary and Standby servers:

```
sqlplus / as sysdba
SQL> ALTER SYSTEM SET dg_broker_start=true sid='*';
SQL> EXIT;
```

```
[oracle@OracleVM2 admin]$ sqlplus / as sysdba
SQL*Plus: Release 19.0.0.0.0 - Production on Thu Nov 23 22:27:44 2023
Version 19.3.0.0.0

Copyright (c) 1982, 2019, Oracle. All rights reserved.

Connected to:
Oracle Database 19c Enterprise Edition Release 19.0.0.0.0 - Production
Version 19.3.0.0.0

SQL> ALTER SYSTEM SET dg_broker_start=true sid='*';
System altered.

SQL> exit;
Disconnected from Oracle Database 19c Enterprise Edition Release 19.0.0.0.0 - Production
Version 19.3.0.0.0
[oracle@OracleVM2 admin]$
```

On standby server, using specifically manual authentication (and not OS authentication), start Data Guard Manager on the Primary DB and create a Data Guard Manager configuration:

```
dgmgrl sys/OracleLab123@dxb
DGMGRl> CREATE CONFIGURATION my_dg_config AS PRIMARY DATABASE IS dubai
CONNECT IDENTIFIER IS dxb;
DGMGRl> ADD DATABASE dubaistb AS CONNECT IDENTIFIER IS dxbstb MAINTAINED AS
PHYSICAL;
DGMGRl> ENABLE CONFIGURATION;
```

```
[oracle@OracleVM1 admin]$ dgmgrl sys/OracleLab123@dxb
DGMGRL for Linux: Release 19.0.0.0.0 - Production on Thu Nov 23 22:28:56 2023
Version 19.3.0.0.0

Copyright (c) 1982, 2019, Oracle and/or its affiliates. All rights reserved.

Welcome to DGMGRL, type "help" for information.
Connected to "dubai"
Connected as SYSDBA.
DGMGRL> CREATE CONFIGURATION my_dg_config AS PRIMARY DATABASE IS dubai CONNECT IDENTIFIER IS dxb;
Configuration "my_dg_config" created with primary database "dubai"
DGMGRL> ADD DATABASE dubaistb AS CONNECT IDENTIFIER IS dxbstb MAINTAINED AS PHYSICAL;
Database "dubaistb" added
DGMGRL> ENABLE CONFIGURATION;
Enabled.
DGMGRL> █
```

```
DGMGRL> SHOW CONFIGURATION;
```

```
DGMGRL> show configuration;

Configuration - my_dg_config

» Protection Mode: MaxPerformance
Members:
    dubai      - Primary database
    dubaistb - Physical standby database

Fast-Start Failover: Disabled

Configuration Status:
SUCCESS  (status updated 32 seconds ago)

DGMGRL> █
```

You can also get all the properties of the Data Guard Broker:

```
DGMGRL>SHOW DATABASE VERBOSE dubai;
```

```
DGMGRL> SHOW DATABASE VERBOSE dubai;

Database - dubai

Role:           PRIMARY
Intended State: TRANSPORT-ON
Instance(s):
  dubai

Properties:
  DGConnectIdentifier      = 'dxb'
  ObserverConnectIdentifier = ''
  FastStartFailoverTarget   = ''
  PreferredObserverHosts    = ''
  LogShipping                = 'ON'
  RedoRoutes                 = ''
  » LogXptMode               = 'ASYNC'
  DelayMins                  = '0'
  Binding                     = 'optional'
  MaxFailure                  = '0'
  ReopenSecs                   = '300'
  NetTimeout                   = '30'
  RedoCompression             = 'DISABLE'
  PreferredApplyInstance       = ''
  ApplyInstanceTimeout         = '0'
  ApplyLagThreshold           = '30'
  TransportLagThreshold        = '30'
  TransportDisconnectedThreshold = '30'
  ApplyParallel                 = 'AUTO'
  ApplyInstances                = '0'
  StandbyFileManagement        = ''
  ArchiveLagTarget              = '0'
  LogArchiveMaxProcesses        = '0'
  LogArchiveMinSucceedDest      = '0'
```

```
DGMGRL>SHOW DATABASE VERBOSE dubaistb;
```

```
DGMGRL> show database verbose dubaistb;

Database - dubaistb

  Role:          PHYSICAL STANDBY
  Intended State: APPLY-ON
  Transport Lag:   0 seconds (computed 1 second ago)
  Apply Lag:     0 seconds (computed 1 second ago)
  Average Apply Rate: 2.00 KByte/s
  Active Apply Rate: 684.00 KByte/s
  Maximum Apply Rate: 1.05 MByte/s
  Real Time Query: OFF
  Instance(s):
    dubaistb

  Properties:
  »» DGConnectIdentifier      = 'dxbstb'
  »» ObserverConnectIdentifier = ''
  »» FastStartFailoverTarget   = ''
  »» PreferredObserverHosts   = ''
  »» LogShipping               = 'ON'
  »» RedoRoutes                = ''
  »» LogXptMode                = 'ASYNC'
  »» DelayMins                 = '0'
  »» Binding                   = 'optional'
  »» MaxFailure                = '0'
  »» ReopenSecs                 = '300'
  »» NetTimeout                 = '30'
  »» RedoCompression            = 'DISABLE'
  »» PreferredApplyInstance     = ''
  »» ApplyInstanceTimeout       = '0'
  »» ApplyLagThreshold          = '30'
  »» TransportLagThreshold      = '30'
  »» TransportDisconnectedThreshold = '30'
  »» ApplyParallel              = 'AUTO'
```

You can check the DMON process which runs on the primary DB:

```
DGMGRL> exit

oracle$ ps -ef | grep -i ora_dmon
[oracle@OracleVM1 admin]$ ps -ef | grep -i ora_dmon
oracle  7899  1 0 22:26 ?        00:00:00 ora_dmon_dubai
oracle  9479 6399 0 22:40 pts/0    00:00:00 grep --color=auto -i ora_dmon
[oracle@OracleVM1 admin]$
```

You will see that the LOG_ARCHIVE_DEST_2 parameter is automatically populated by the Broker.

```
oracle$ sqlplus / as sysdba
SQL> show parameter LOG_ARCHIVE_DEST_2
```

```
[oracle@OracleVM1 admin]$ sqlplus / as sysdba

SQL*Plus: Release 19.0.0.0.0 - Production on Thu Nov 23 22:42:37 2023
Version 19.3.0.0.0

Copyright (c) 1982, 2019, Oracle. All rights reserved.

Connected to:
Oracle Database 19c Enterprise Edition Release 19.0.0.0.0 - Production
Version 19.3.0.0.0

SQL> show parameter LOG_ARCHIVE_DEST_2

NAME                                     TYPE        VALUE
-----                                     -----
log_archive_dest_2                      string      service="dxbstb", ASYNC NOAFFI
                                            RM delay=0 optional compressio
                                            n=disable max_failure=0 reopen
                                            =300 db_unique_name="dubaistb"
                                            net_timeout=30, valid_for=(on
                                            line_logfile,all_roles)
»
log_archive_dest_20                     string
log_archive_dest_21                     string
log_archive_dest_22                     string
log_archive_dest_23                     string
log_archive_dest_24                     string

NAME                                     TYPE        VALUE
-----                                     -----
log_archive_dest_25                     string
log_archive_dest_26                     string
log_archive_dest_27                     string
log_archive_dest_28                     string
log_archive_dest_29                     string

SQL> █
```

here is how you can disable Data Guard Broker:

```
SQL> ALTER SYSTEM SET DG_BROKER_START=FALSE;
DGMGRL> DISABLE CONFIGURATION;
DGMGRL> REMOVE CONFIGURATION;
```

Testing SwitchOver and SwitchBack using Data Guard Broker

When using DGMGRL, you should always use the DB_UNIQUE_NAME.

Start DGMGRL on the primary:

```
dgmgrl sys/OracleLab123@dxb
DGMGRL> switchover to dubaistb;
DGMGRL> switchover to dubaistb;
Performing switchover NOW, please wait...
Operation requires a connection to database "dubaistb"
Connecting ...
Connected to "dubaistb"
Connected as SYSDBA.
New primary database "dubaistb" is opening...
Operation requires start up of instance "dubai" on database "dubai"
Starting instance "dubai"...
Connected to an idle instance.
ORACLE instance started.
Connected to "dubai"
Database mounted.
Connected to "dubai"
Switchover succeeded, new primary is "dubaistb"
DGMGRL> █
```

Now, login to the NEW primary (OracleVM2) and check its status:

```
sqlplus / as sysdba
SQL> set lines 200
SQL> SELECT DB_UNIQUE_NAME, OPEN_MODE, PROTECTION_MODE, PROTECTION_LEVEL,
DATABASE_ROLE, SWITCHOVER_STATUS FROM V$DATABASE;
```

```
[oracle@OracleVM2 admin]$ sqlplus / as sysdba
SQL*Plus: Release 19.0.0.0.0 - Production on Thu Nov 23 23:05:09 2023
Version 19.3.0.0.0
Copyright (c) 1982, 2019, Oracle. All rights reserved.

Connected to:
Oracle Database 19c Enterprise Edition Release 19.0.0.0.0 - Production
Version 19.3.0.0.0

SQL> set lines 200
SQL> SELECT DB_UNIQUE_NAME, OPEN_MODE, PROTECTION_MODE, PROTECTION_LEVEL, DATABASE_ROLE ROLE, SWITCHOVER_STATUS FROM V$DATABASE;
DB_UNIQUE_NAME          OPEN_MODE       PROTECTION_MODE   PROTECTION_LEVEL    ROLE           SWITCHOVER_STATUS
-----              -----          -----          -----          -----          -----
dubaistb                READ WRITE      MAXIMUM PERFORMANCE  MAXIMUM PERFORMANCE PRIMARY        TO STANDBY
SQL> ■
```

Finally, check the status of the NEW standby (OracleVM1):

```
sqlplus / as sysdba
SQL> set lines 200
SQL> SELECT DB_UNIQUE_NAME, OPEN_MODE, PROTECTION_MODE, PROTECTION_LEVEL,
DATABASE_ROLE ROLE, SWITCHOVER_STATUS FROM V$DATABASE;
[oracle@OracleVM1 admin]$ sqlplus / as sysdba
SQL*Plus: Release 19.0.0.0.0 - Production on Thu Nov 23 23:03:23 2023
Version 19.3.0.0.0
Copyright (c) 1982, 2019, Oracle. All rights reserved.
SQL>
Connected to:
Oracle Database 19c Enterprise Edition Release 19.0.0.0.0 - Production
Version 19.3.0.0.0

SQL> set lines 200
SQL> SELECT DB_UNIQUE_NAME, OPEN_MODE, PROTECTION_MODE, PROTECTION_LEVEL, DATABASE_ROLE ROLE, SWITCHOVER_STATUS FROM V$DATABASE;
DB_UNIQUE_NAME          OPEN_MODE       PROTECTION_MODE   PROTECTION_LEVEL    ROLE           SWITCHOVER_STATUS
-----              -----          -----          -----          -----          -----
dubai                  MOUNTED        MAXIMUM PERFORMANCE  MAXIMUM PERFORMANCE PHYSICAL STANDBY NOT ALLOWED
SQL>
```

When needed, you can switch back to "dubai" instance from either one of the servers:

```
dgmgrl sys/OracleLab123@dxb
DGMGRL> switchover to dubai;
```

```
[oracle@OracleVM2 admin]$ dgmgrl sys/OracleLab123@dxb
DGMGRL for Linux: Release 19.0.0.0.0 - Production on Thu Nov 23 23:06:55 2023
Version 19.3.0.0.0
>
Copyright (c) 1982, 2019, Oracle and/or its affiliates. All rights reserved.

Welcome to DGMGRL, type "help" for information.
Connected to "dubai"
Connected as SYSDBA.
DGMGRL> switchover to dubai;
Performing switchover NOW, please wait...
New primary database "dubai" is opening...
Operation requires start up of instance "dubaistb" on database "dubaistb"
Starting instance "dubaistb"...
Connected to an idle instance.
ORACLE instance started.
Connected to "dubaistb"
Database mounted.
Connected to "dubaistb"
Switchover succeeded, new primary is "dubai"
DGMGRL> █
```