

Creating Active Directory Domain Services in Oracle Cloud Infrastructure

Quick Start

ORACLE WHITE PAPER | JANUARY 2019





Disclaimer

The following is intended to outline our general product direction. It is intended for information purposes only and may not be incorporated into any contract. It is not a commitment to deliver any material, code, or functionality, and should not be relied upon in making purchasing decisions. The development, release, and timing of any features or functionality described for Oracle's products remains at the sole discretion of Oracle.

Revision History

The following revisions have been made to this white paper since its initial publication:

Date	Revision
January 9, 2019	Initial publication

You can find the most recent versions of the Oracle Cloud Infrastructure white papers at <https://cloud.oracle.com/iaas/technical-resources>.



Table of Contents

Overview	4
Assumptions	4
Setting Up the Network Environment	5
Create a VCN	6
Create a NAT Gateway	6
Create a Private Security List	6
Create a Private Route Table	6
Create Security List Rules	7
Create Subnets	8
Creating a Bastion Host	8
Creating the Windows Instances	8
Configuring the Forest and Domain Controllers	9
Create the Primary Domain Controller	10
Add a Second Domain Controller	16
Add a New Host	19
Conclusion	22
References	23
Appendix A: ActiveDirectoryInit.ps1	23
Appendix B: ActiveDirectoryInit2.ps1	25
Appendix C: AddComputer.ps1	26
Appendix D: NewComputer.ps1	26



Overview

Active Directory Domain Services are a proven solution for identity management. Oracle Cloud Infrastructure can help you build and extend your current Active Directory forest. This white paper walks you through the process of creating an Active Directory environment in your Oracle Cloud Infrastructure tenancy. Two domain controllers are installed, one active and one read-only, each in a different availability domain for redundancy. A third system is used as a test server to ensure that you can both join to and log in to the domain established in Oracle Cloud Infrastructure.

This document provides the following information:

- How to automate the deployment of your Active Directory servers
- Best practices for building a simple Active Directory environment and joining domains
- Scripts that you can use to help automate your deployment in an Oracle Cloud Infrastructure environment

The following topics are out of scope and therefore *not* covered:

- Active Directory design and topologies
- Large forest, tree, and leaf designs
- Group policies or policy management

Assumptions

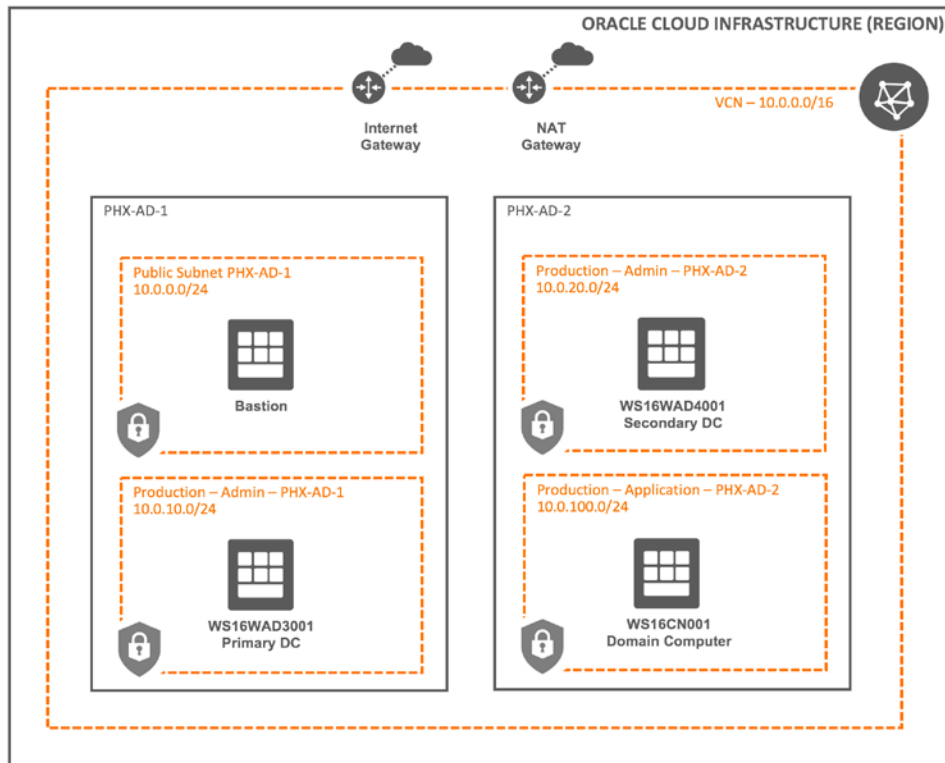
To perform the actions in this paper, you must have a non-root compartment.

Also, you should be familiar with the [fundamentals of the Oracle Cloud Infrastructure](#). If this is the first time that you have used the platform, we recommend walking through the [getting started tutorial](#).

You should also have a basic understanding of [Active Directory concepts](#).

Setting Up the Network Environment

The following diagram depicts the components of the environment that this white paper includes:



Best Practice: The domain controllers should not be accessible externally from the internet. Create a separate subnet for your domain assets like Active Directory domain controllers and a separate subnet for your application servers.

A bastion host is used to access the environment to prevent exposing the Remote Desktop Protocol (RDP) ports of the Active Directory domain controllers to the internet. RDP sessions are tunneled through an SSH connection to a bastion host. Separate subnets (as illustrated in the diagram) are used to host the primary and secondary domain controllers created in the steps that follow. Because subnets are associated with availability domains, each of the domain controllers resides in different availability domains, thereby creating an Active Directory domain structure that is resilient to availability domain issues. In the examples that follow, the virtual cloud network (VNC) IP space of 10.0.0.0/16 is used.

Best Practice: Always be as descriptive as possible when naming Oracle Cloud Infrastructure components. Descriptive names make it easier when you have to revisit an environment later.

Create a VCN

Use the Oracle Cloud Infrastructure Console to [create the virtual cloud network \(VCN\)](#) and related resources, including the internet gateway for the bastion host, public routing tables, and security lists for the public subnet. Two more public subnets are created but aren't used in this environment. More networking resources for the private segments of the environment are created in the following sections.

Create the following VCN and related resources: **vcn01**

Create a NAT Gateway

[Create a NAT gateway](#) to allow the instances that have only private IP addresses to access internet resources.

Create the following NAT gateway: **nat-gateway**

Create a Private Security List

When you create a subnet in the following steps, you must select a security list. [Create an empty security list](#) now and add the rules in a later step.

Create the following security list: **Production - Active Directory**

Create a Private Route Table

[Create a route table](#) to use for the private subnets. Private subnets automatically can route to other private subnets in the VCN. The NAT gateway that you created is used by this route table for all internet destinations, which allows instances that have only private IP addresses to access internet resource.

Create the following route table with a **0.0.0.0/0** route to **nat-gateway**:

Name	Target Type	Destination CIDR Block	Target Selection
Production - Active Directory - NAT	NAT Gateway	0.0.0.0/0	nat-gateway

Create Security List Rules

Active Directory uses several protocols to communicate, including RPC, NetBIOS, SMB, LDAP, Kerberos, WINS, and DNS. All of the protocols are listed here, although your configuration might use only some of them. If a protocol (for example, WINS) is not used in your environment, you can remove it from the list.

As a best practice, all the domain controllers should be in a subnet that either has no external IP addresses or has no access from the internet. As a result, you might want to enable all ports to communicate between your subnets and the Active Directory subnets. However, be aware that this still opens potential paths of attack from those subnets. Therefore, it's a best practice to open only the following ports between the subnets:

Name	Protocol	Port
RDP	TCP	3389
DNS	TCP, UDP	53
LDAP	TCP, UDP	389
LDAP over SSL	TCP	636
Global catalog LDAP	TCP	3268
Global catalog LDAP over SSL	TCP	3269
Kerberos	TCP, UDP	88
RPC endpoint mapper	TCP, UDP	135
NetBIOS name service	TCP, UDP	137
NetBIOS datagram service	UDP	138
NetBIOS session service	TCP	139
SMB over IP (Microsoft-DS)	TCP, UDP	445
WINS resolution	TCP, UDP	1512
WINS replication	TCP, UDP	42

[Create ingress rules](#) on the **Production - Active Directory** security list to allow the required port communication into the new Active Directory subnets (these rules must exist to allow traffic between the two domain controller subnets).

Create Subnets

As mentioned previously, you need at least two private subnets (a third subnet in the third availability domain can be used for extra availability of the Active Directory environment).

[Create the following subnets:](#)

Name	Availability Domain	CIDR Block	Route Table	Security Lists
Production - Admin - PHX-AD-1	PHX-AD-1	10.0.10.0/24	Production - Active Directory - NAT	Production - Active Directory
Production - Admin - PHX-AD-2	PHX-AD-2	10.0.20.0/24	Production - Active Directory - NAT	Production - Active Directory
Production - Application - PHX-AD-2	PHX-AD-2	10.0.100.0/24	Production - Active Directory - NAT	Production - Active Directory

Creating a Bastion Host

A bastion host is used to access the Active Directory environment. This secures RDP sessions by tunneling them through an SSH tunnel. For more information about bastion hosts, see the [Bastion Hosts: Protected Access for Virtual Cloud Networks](#) white paper.

[Create a bastion host](#) with the following details:

Name	Image	Shape	Availability Domain	Subnet
Bastion	Oracle Linux 7.5	VM.Standard2.1	PHX-AD-1	Public Subnet PHX-AD-1

Creating the Windows Instances

The example in this white paper uses three Windows Server 2016 instances. Two are used for the Active Directory domain controllers, and the third is joined to the domain as a new host. Use the following properties when you create the instances in the following section. (The shape used in this paper is a recommendation; scale it up or down as needed).

Name	Image	Shape	Availability Domain	Subnet
WS16WAD3001	Windows Server 2016 Standard VM	VM.Standard2.1	PHX-AD-1	Production - Admin - PHX-AD-1
WS16WAD4001	Windows Server 2016 Standard VM	VM.Standard2.1	PHX-AD-2	Production - Admin - PHX-AD-2
WS16CN001	Windows Server 2016 Standard VM	VM.Standard2.1	PHX-AD-2	Production - Application - PHX-AD-2

For each instance, note the RFC1918 IP addresses:

Instance	RFC1918 IP
DC-1	10.0.10.2
DC-2	10.0.20.2
Test-SRV	10.0.100.2

Configuring the Forest and Domain Controllers

You can create your initial domain controller in several different ways. This paper uses [Microsoft PowerShell](#) integrated with [Cloudbase Init](#) to reduce the amount of manual interaction with the Active Directory setup. The scripts provided in the appendices install the necessary Windows Server features, such as the .NET Framework, Active Directory Domain Services, and the DNS server components. Four PowerShell scripts are used to create this environment:

- Appendix A: ActiveDirectoryInit.ps1: Create the forest and promote the server to an Active Directory domain controller.
- Appendix B: ActiveDirectoryInit2.ps1: Build the second host and promote it to be the replica domain controller.
- Appendix C: AddComputer.ps1: Prepare the domain for a new computer join.
- Appendix D: NewComputer.ps1: Join a Windows Server to the domain at launch time.

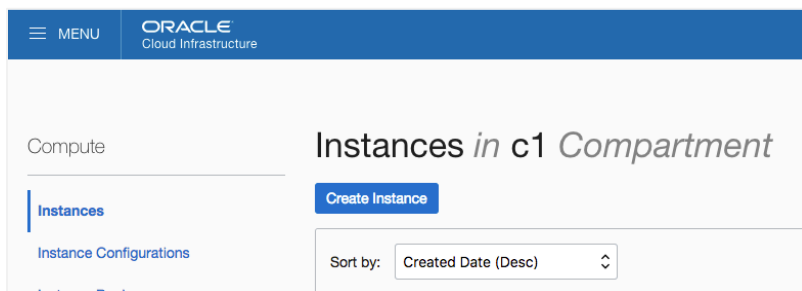
This paper uses the Oracle Cloud Infrastructure Console to demonstrate how to create the compute instances. You need the following information:

- Your domain administrator password. A best practice is to ensure that you change your domain administrator password immediately after you create the domain controllers.

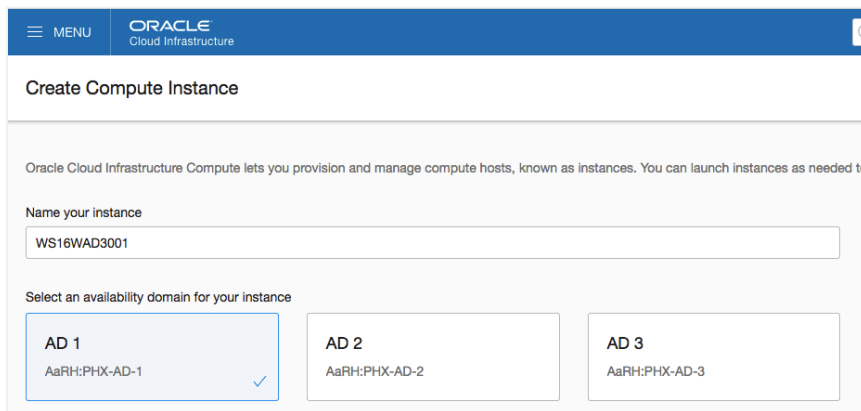
- The name of the domain that you are about to create.
- A one-time password that you will use when joining new computers to the domain.

Create the Primary Domain Controller

1. In the Oracle Cloud Infrastructure Console, go to the Compute section and click **Create Instance**.



2. Provide a name for the instance (WS16WAD3001) and select the availability domain (PHX-AD-1).



3. Choose the operating system (Windows Server 2016 Standard) and image version.

ORACLE Cloud Infrastructure

Create C Browse All Images

Operating System

- ☐ Oracle Linux 6.10
- ☐ Oracle Linux 7.5
- ☐ Windows Server 2008 R2
- ☐ Windows Server 2012 R2 Datacenter
- ☐ Windows Server 2012 R2 Standard
- ☐ Windows Server 2016 Datacenter
- ☒ Windows Server 2016 Standard

1 Selected

Image Version

Gen2-2018.10.13-0

Agreement for Platform Image "Windows Server 2016 Standard" (English | Deutsch)

☒ I have reviewed and accept the [Oracle and Microsoft Windows Terms of Use](#)

Select Image Cancel

4. Select the instance type (virtual machine) and the instance shape (VM.Standard2.1).

Choose instance type

Virtual Machine
A virtual machine is an independent computing environment that runs on top of physical bare metal hardware.

Bare Metal Machine
A bare metal compute instance gives you dedicated physical server access for highest performance and strong isolation.

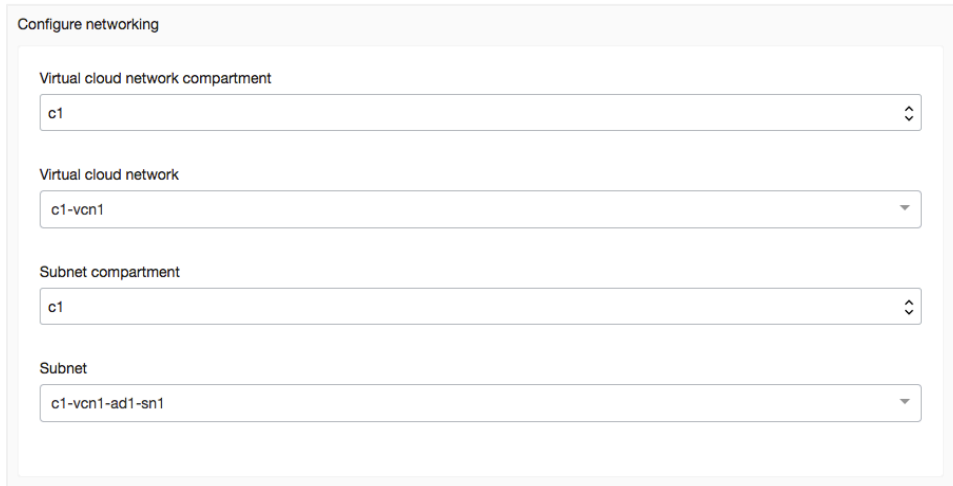
Choose instance shape

VM.Standard2.1
1 Core OCPU, 15 GB Memory

Change Shape

Note: You can choose a larger boot volume size or encrypt the boot volume via the [Key Management service](#). This white paper doesn't address this function.

5. Configure your network connection.



Configure networking

Virtual cloud network compartment
c1

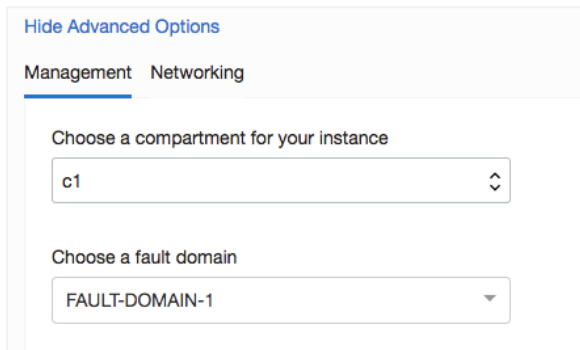
Virtual cloud network
c1-vcn1

Subnet compartment
c1

Subnet
c1-vcn1-ad1-sn1

Best Practice: Ensure that your new domain controllers are in the private subnet.

6. Under **Advanced Options**, select the **Management** view, and then choose the compartment and fault domain.



Hide Advanced Options

Management Networking

Choose a compartment for your instance
c1

Choose a fault domain
FAULT-DOMAIN-1

7. In the **User Data** section, select **Paste cloud-init script** and add the script to create the domain controller. Copy the **ActiveDirectoryInit.ps1** script from Appendix A and paste it in the text box:

You can choose to specify a startup script that will run when your instance boots up or restarts. Startup scripts can be used to install software and updates, and to ensure that services are running within the virtual machine.

☐ Choose cloud-init script file ☒ Paste cloud-init script

```
#ps1_sysnative
#####
# Title: ActiveDirectoryInit.ps1
# Version & Date: v1 31 Oct 2018
# Creator: john.s.parker@oracle.com
# Warning: This script is a representation of how to use Powershell to create an Active Directory Domain controller
```

☒ Enable monitoring

8. Click **Create**.

The script takes approximately 20 minutes to complete the installation of the Windows features and the Active Directory tools.

ORACLE Cloud Infrastructure

Compute » Instances » Instance Details

WS16WAD3001

Create Custom Image Start Stop Reboot Terminate Apply Tag(s) Create Instance Configuration

Instance Information Tags

Instance Information

Availability Domain: AaRH:PHX-AD-1	Image: Windows-Server-2016-Standard-Edition-VM-Gen2-2018.10.13-0
Fault Domain: FAULT-DOMAIN-1	OCID: ...g3ngua Show Copy
Region: phx	Launched: Wed, 31 Oct 2018 23:29:43 GMT
Shape: VM.Standard2.1	Compartment: c1
Virtual Cloud Network: c1-vcn1	Launch Mode: NATIVE
Maintenance Reboot: -	

Primary VNIC Information

Private IP Address: 10.50.0.196	Internal FQDN: ws16wad3001... Show Copy
---------------------------------	---

You can log in and monitor the progress by viewing the `stage1.txt` log at `C:\DomainJoin`. The log should show `Success = True` for .NET Framework, Active Directory Domain Services, Active Directory Administrative Center, and DNS Server Tools.

```
stage1 - Notepad
File Edit Format View Help
PSCompatibleVersions: 1.0, 2.0, 3.0, 4.0, 5.0, 5.1.14393.2515
BuildVersion: 10.0.14393.2515
CLRVersion: 4.0.30319.42000
WSManStackVersion: 3.0
PSRemotingProtocolVersion: 2.3
SerializationVersion: 1.1.0.1
*****
Transcript started, output file is C:\DomainJoin\stage1.txt

Success Restart Needed Exit Code      Feature Result
-----
True      No                Success      { .NET Framework 3.5 (includes .NET 2.0 and...
True      No                Success      { Active Directory Domain Services, Remote ...
True      No                Success      { Active Directory Administrative Center, A...
True      No                Success      { DNS Server Tools }
*****
Windows PowerShell transcript end
End time: 20181106213523
*****
```

9. After the first reboot, log in to the host with the **domain\administrator** account to execute the last script with the RunOnce script. The first login with the domain administrator account starts the RunOnce script and provides you a reference for when the entire process will be complete.

```
Administrator: Windows PowerShell
For more information about this setting, see Knowledge Base article 942564
(http://go.microsoft.com/fwlink/?LinkId=104751).

Install-ADDSForest

Validating environment and user input
All tests completed successfully
[oooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooooo]
Installing new forest
waiting for DNS installation to finish

infrastructure, you should manually create a delegation to this DNS server in the parent zone to ensure
reliable name resolution from outside the domain "cesa.corp". Otherwise, no action is required.

WARNING: Windows Server 2016 domain controllers have a default for the security setting named "Allow
cryptography algorithms compatible with Windows NT 4.0" that prevents weaker cryptography algorithms
when establishing security channel sessions.

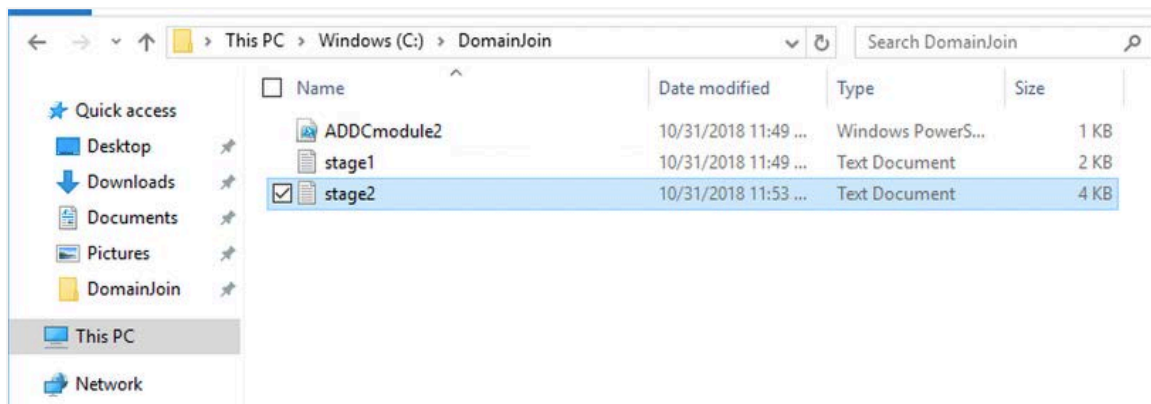
For more information about this setting, see Knowledge Base article 942564
(http://go.microsoft.com/fwlink/?LinkId=104751).

WARNING: This computer has at least one physical network adapter that does not have static IP
address(es) assigned to its IP Properties. If both IPv4 and IPv6 are enabled for a network adapter, both
IPv4 and IPv6 static IP addresses should be assigned to both IPv4 and IPv6 Properties of the physical
network adapter. Such static IP address(es) assignment should be done to all the physical network
adapters for reliable Domain Name System (DNS) operation.

WARNING: A delegation for this DNS server cannot be created because the authoritative parent zone cannot
be found or it does not run Windows DNS server. If you are integrating with an existing DNS
infrastructure, you should manually create a delegation to this DNS server in the parent zone to ensure
reliable name resolution from outside the domain "cesa.corp". Otherwise, no action is required.
```

After the RunOnce script completes, the instance restarts automatically as part of the process.

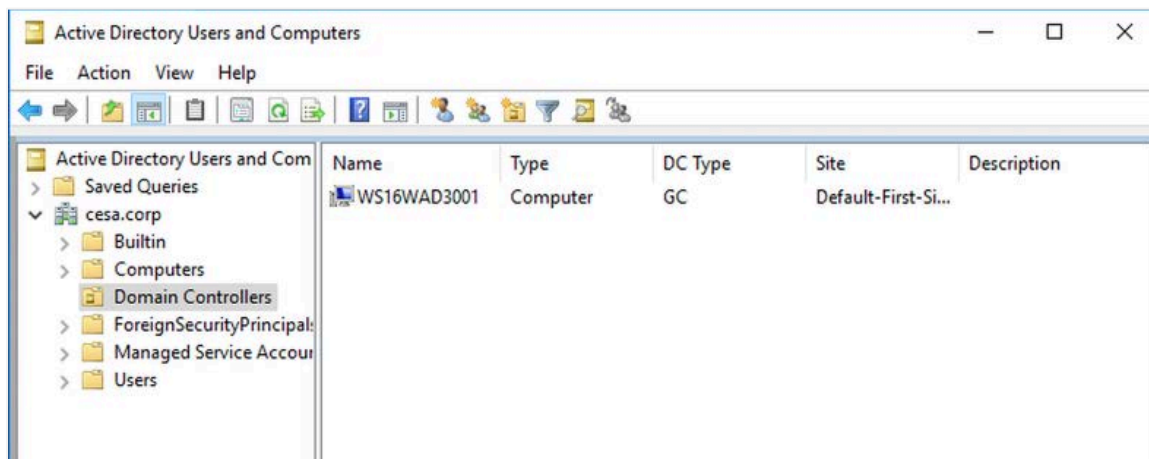
10. Log back in and check the logs to ensure that there are no errors. The logs are `stage1.txt` and `stage2.txt` located in `C:\DomainJoin`.



For success, `stage2.txt` should have Warnings but no Errors.

```
stage2 - Notepad
File Edit Format View Help
WSManStackVersion: 3.0
PSRemotingProtocolVersion: 2.3
SerializationVersion: 1.1.0.1
*****
Transcript started, output file is C:\DomainJoin\stage2.txt
WARNING: Windows Server 2016 domain controllers have a default for the security setting
For more information about this setting, see Knowledge Base article 942564 (http://go.
WARNING: This computer has at least one physical network adapter that does not have st
WARNING: A delegation for this DNS server cannot be created because the authoritative
WARNING: Windows Server 2016 domain controllers have a default for the security setting
For more information about this setting, see Knowledge Base article 942564 (http://go.
WARNING: This computer has at least one physical network adapter that does not have st
WARNING: A delegation for this DNS server cannot be created because the authoritative
*****
Windows PowerShell transcript end
End time: 20181106230924
*****
```

11. Verify that the domain has been successfully created by opening the **Start** menu and selecting **Windows Administrative Tools > Active Directory Users and Computers**.



Now you have the first domain controller in the new Active Directory forest. The new forest is ready for configuration that is not covered in this paper, such as group policies, more domain trusts, and DNS configurations.

Add a Second Domain Controller

Repeat steps 1–6 in the previous section to create a backup domain controller. Make the appropriate changes in the name of the instance and in setting the appropriate [availability domain](#) and the correct fault domain to ensure that you have proper redundancy for your domain. The next series of steps use the script from Appendix B.

Best Practice: To ensure best availability, we recommend that you deploy across multiple availability domains, or fault domain within one availability domain.

1. Under **Advanced Options**, in the **User Data** section, select **Paste cloud-init script**. Copy the **ActiveDirectorInit2.ps1** script from Appendix B and paste it in the text box. In the script, adjust the `$DnsServer` variable to the private IP address for the domain controller that you just created.

```
$DnsServer = 'Private IP for Current DC'
```

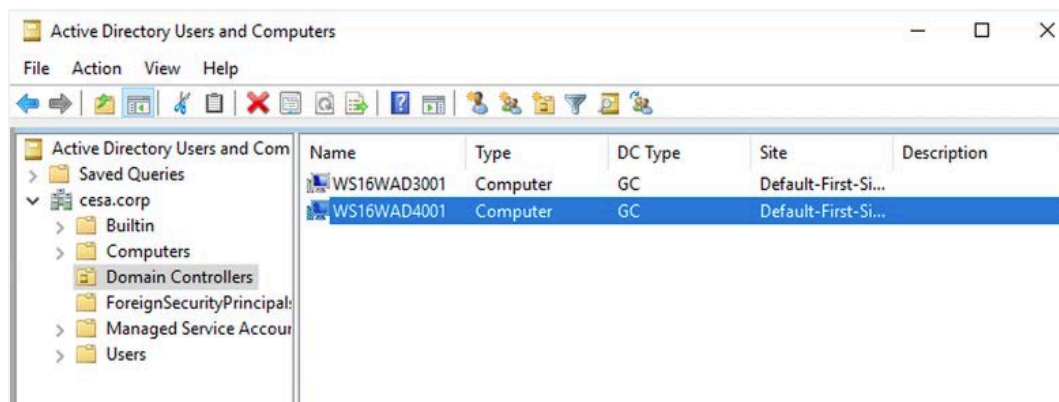

☐ Choose cloud-init script file
 ☒ Paste cloud-init script

```

$EncryptedPass = ConvertTo-SecureString $Password -AsPlainText -Force
$Credential = New-Object -TypeName System.Management.Automation.PSCredential $EncryptedPass
$DnsServer = '192.168.0.1'
#Set the Administrator Password and activate the Windows Firewall
#net user Administrator $EncryptedPass /logon /add
  
```

2. Click **Create**.

You can monitor the progress by watching the **Domain Controllers** section in the current domain controller under **Active Directory Users and Computers**. It takes approximately 20 minutes to install all of the necessary Windows Server features and add the server to the domain.



3. After the final reboot of the host, check the installation by logging in with the domain administrator password. Check the `C:\DomainJoin\stage3.txt` log. Also verify that the Active Directory tools are loaded on the host.

The `stage3.txt` file should not have any errors and should show `Success = True` for the .NET Framework, Active Directory Domain Services, Active Directory Administrative Center, and DNS Server Tools. You should see only Warnings and no Errors, Success for the DCPromo, and that a restart is required. The script will reboot your host after five minutes.

```
Stage3 - Notepad
File Edit Format View Help
*****
Transcript started, output file is C:\DomainJoin\Stage3.txt

*****
Success Restart Needed Exit Code      Feature Result
-----
True      No           Success      { .NET Framework 3.5 (includes .NET 2.0 and...
True      No           Success      { Active Directory Domain Services, Remote ...
True      No           Success      { Active Directory Administrative Center, A...
True      No           Success      { DNS Server Tools}
WARNING: Windows Server 2016 domain controllers have a default for the security setting named "Allow c
For more information about this setting, see Knowledge Base article 942564 (http://go.microsoft.com/fw
WARNING: This computer has at least one physical network adapter that does not have static IP address(
WARNING: A delegation for this DNS server cannot be created because the authoritative parent zone cann
WARNING: Windows Server 2016 domain controllers have a default for the security setting named "Allow c
For more information about this setting, see Knowledge Base article 942564 (http://go.microsoft.com/fw
WARNING: This computer has at least one physical network adapter that does not have static IP address(
WARNING: A delegation for this DNS server cannot be created because the authoritative parent zone cann

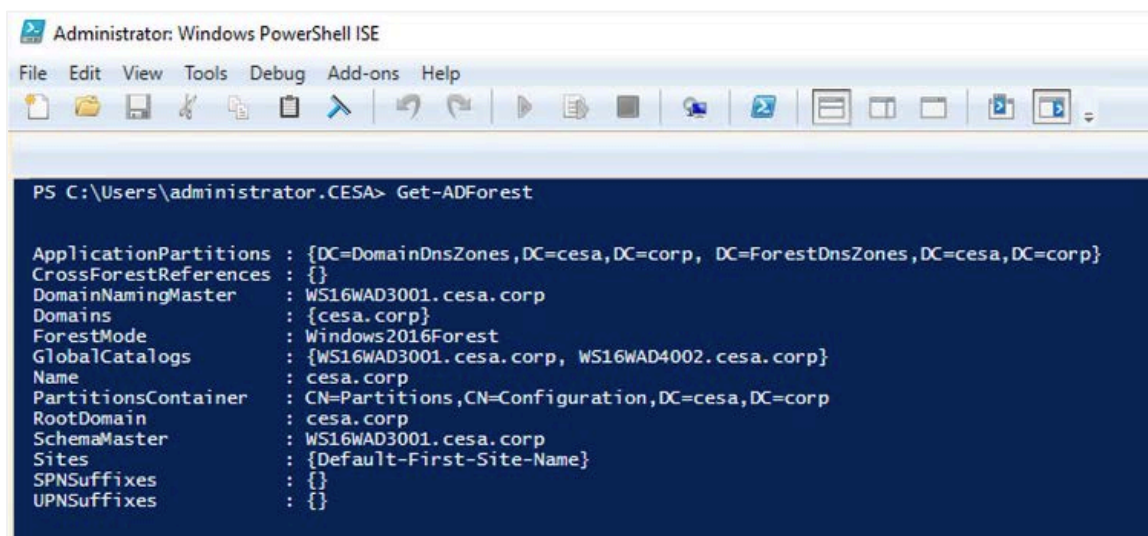
Message      : You must restart this computer to complete the operation.

Context      : DCPromo.General.4
RebootRequired : True
Status       : Success

*****
Windows PowerShell transcript end
End time: 20181106235025
*****
```

4. Run `Get-ADForest` from the PowerShell command prompt to verify communication with the domain.

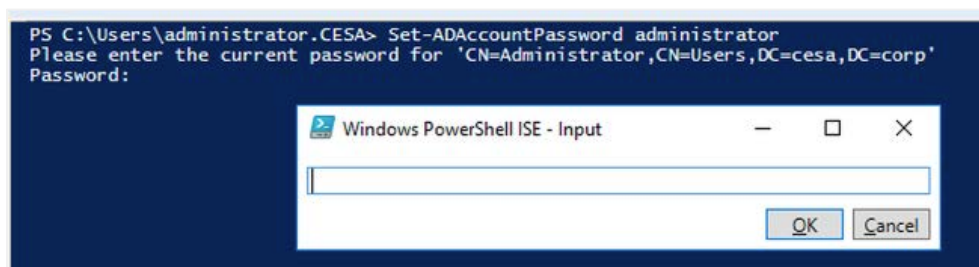
The output should show the correct domains and name for your Active Directory domain and forest.



```
Administrator: Windows PowerShell ISE
File Edit View Tools Debug Add-ons Help
PS C:\Users\administrator.CESA> Get-ADForest

ApplicationPartitions : {DC=DomainDnsZones,DC=cesa,DC=corp, DC=ForestDnsZones,DC=cesa,DC=corp}
CrossForestReferences : {}
DomainNamingMaster    : WS16WAD3001.cesa.corp
Domains               : {cesa.corp}
ForestMode            : Windows2016Forest
GlobalCatalogs       : {WS16WAD3001.cesa.corp, WS16WAD4002.cesa.corp}
Name                  : cesa.corp
PartitionsContainer    : CN=Partitions,CN=Configuration,DC=cesa,DC=corp
RootDomain            : cesa.corp
SchemaMaster          : WS16WAD3001.cesa.corp
Sites                 : {Default-First-Site-Name}
SPNSuffixes           : {}
UPNSuffixes           : {}
```

5. After the domain controllers are installed, change your domain administrator password by using the `Set-ADAccountPassword` command. Ensure that you use a strong password that meets the password standards of your organization.



You now have a primary Active Directory domain controller and a secondary domain controller to facilitate a complete Active Directory forest in your Oracle Cloud Infrastructure tenancy. Ensure that you don't skip step 5, which can create a security threat to your Active Directory domain. You should also add any of your group policies and users that you require in your environment.

Add a New Host

Now you can add new hosts to your domain. There are many ways to join new computers to the new Active Directory domain. This paper uses a Microsoft PowerShell example for using a predefined computer credential to add the new host to the domain. The Microsoft website has more examples that you can use to add hosts to your domain.

Best Practice: We recommend that you use the [Microsoft PowerShell](#) example of using a predefined computer credential to add new hosts to your domain.

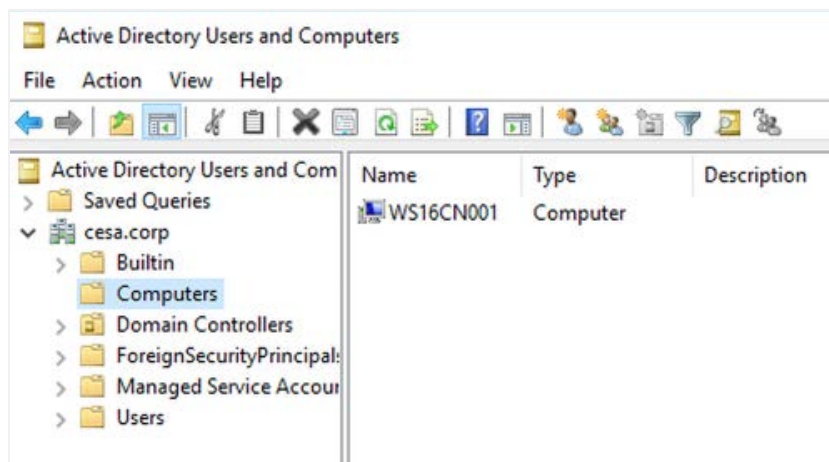
1. Log in to your primary domain controller with a domain administrator account.
2. Open a PowerShell window and run the **AddComputer.ps1** script from Appendix C.

The script executes the `New-ADComputer` command to add the new computer record in the domain controller.

```
PS C:\Users\administrator.CESA> $NewComputerName = "WS16CN001"
New-ADComputer -Name $NewComputerName -AccountPassword (ConvertTo-SecureString -String 'TempJoinPAS$' -AsPlainText -Force)

PS C:\Users\administrator.CESA>
```

3. Verify that the computer was added by checking **Computers** section of **Active Directory Users and Computers**.



After you have added the computer, you are ready to create the instance in your Oracle Cloud Infrastructure tenancy.

4. Sign in to the Oracle Cloud Infrastructure Console and create the Windows Server 2016 instance by following steps 1–6 in the Create the Primary Domain Controller section of this document. Use the appropriate name and place it in the correct availability and fault domains and pick a subnet and shape that are correct for your needs.

- Click **Advanced Options** and select **Paste cloud-init script** in the **User Data** section. Copy the **NewComputer.ps1** script from Appendix D and paste it in the text box.

User data

You can choose to specify a startup script that will run when your instance boots up or restarts. Startup scripts can be used to install software and updates, and to ensure that services are running within the virtual machine.

☐ Choose cloud-init script file ☒ Paste cloud-init script

```
#ps1_sysnative
#####
# Title: newcomputer.ps1
# Version & Date: v1 31 Oct 2018
# Creator: john.s.parker@oracle.com
# Warning: This script is a representation of how to use Powershell to create an Active Directory Domain controller
```

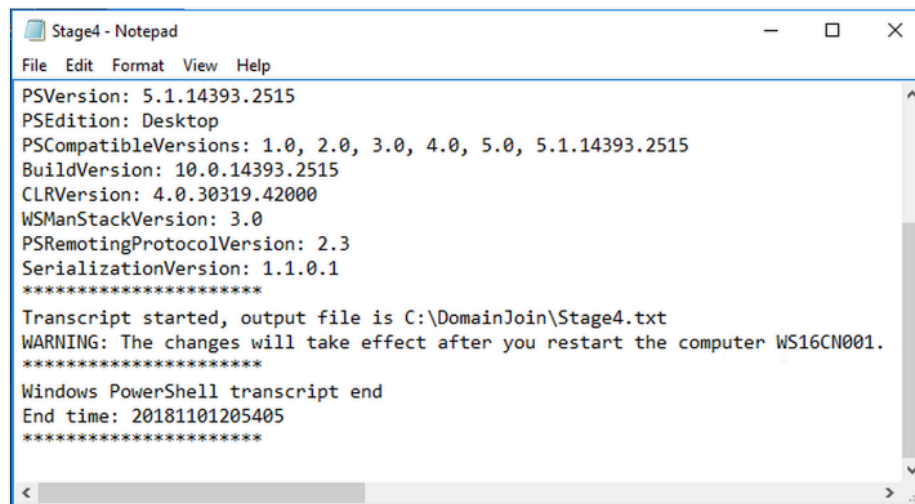
- In the script, update the `$DnsServer` variable with the correct IP address for your domain controller.

```
$DnsServer = '192.168.0.1'
$DomainToJoin = 'cesa.corp'
#####
# Sets the DNS to the DC.
#####
Set-DnsClientServerAddress -InterfaceAlias Ethernet -ServerAddresses $DnsServer
```

- Click **Create**.

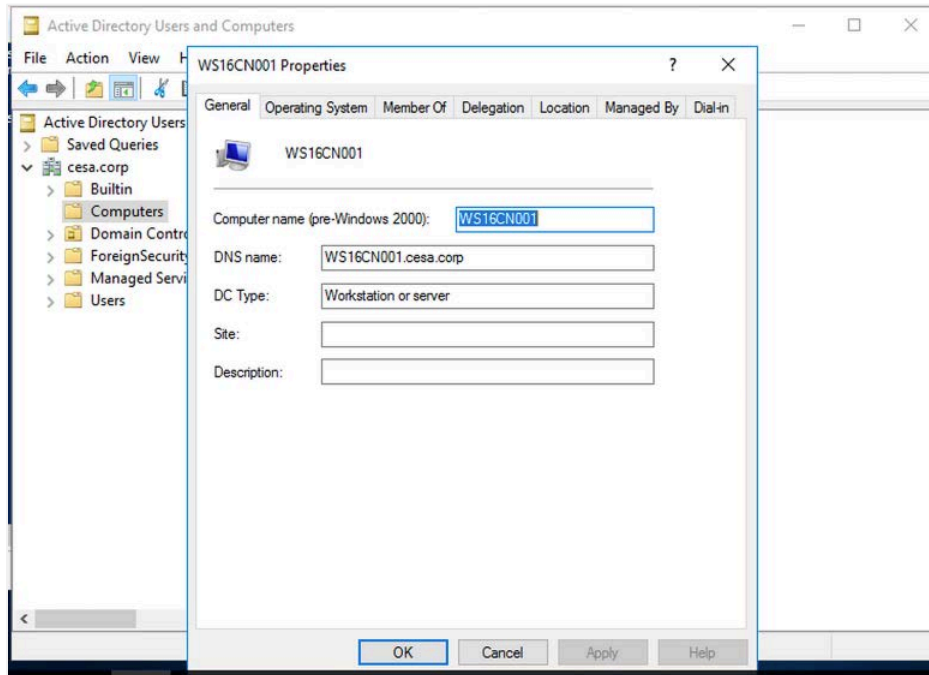
After the new computer has joined the domain, it automatically reboots. The script contains a 5-minute sleep to ensure that domain replication has occurred.

- Log in to the new host and check the `C:\DomainJoin\Stage4.txt` log for errors.



```
Stage4 - Notepad
File Edit Format View Help
PSVersion: 5.1.14393.2515
PSEdition: Desktop
PSCompatibleVersions: 1.0, 2.0, 3.0, 4.0, 5.0, 5.1.14393.2515
BuildVersion: 10.0.14393.2515
CLRVersion: 4.0.30319.42000
WSManStackVersion: 3.0
PSRemotingProtocolVersion: 2.3
SerializationVersion: 1.1.0.1
*****
Transcript started, output file is C:\DomainJoin\Stage4.txt
WARNING: The changes will take effect after you restart the computer WS16CN001.
*****
Windows PowerShell transcript end
End time: 20181101205405
*****
```

You can also check the computer properties on the domain controller in the **Active Directory Users and Computers** administrative application.



Now you have a fully functioning Active Directory domain where you can add new computers and expand your domain to fit the needs of your organization.

Conclusion

This white paper walks you through the core steps of building an Active Directory domain, using redundant domain controllers that are in separate Oracle Cloud Infrastructure availability or fault domains and logical subnets to ensure that you are building fault tolerance into your infrastructure. You can build more application servers to add to the domain. These are the building blocks of your Active Directory domain. It's up to you to build your group policies and ensure that your domain meets the standards of your organization.

Oracle Cloud Infrastructure enables you to deploy the building blocks of your Active Directory domain and support any expansion to your Active Directory forests that your organization requires to meet the demanding needs of today's computing environments.

References

- [Oracle Cloud Infrastructure documentation](#)
- [Oracle Cloud Infrastructure regions and availability domains](#)
- [Creating an Oracle Cloud Infrastructure virtual cloud network](#)
- [Oracle Cloud Infrastructure bastion hosts](#)
- [Adding Active Directory Users Guide](#)
- [Active Directory Domain Services](#)
- [Active Directory Domain Services Features](#)
- [Active Directory PowerShell Commands](#)
- [RunOnce Registry Key](#)
- [Microsoft PowerShell Documentation](#)

Appendix A: ActiveDirectoryInit.ps1

```
#ps1_sysnative
#####
# Title: ActiveDirectoryInit.ps1
# Version & Date: v1 31 Oct 2018
# Creator: john.s.parker@oracle.com
# Warning: This script is a representation of how to use PowerShell to create an Active Directory Domain controller
# and build the first DC in a new Active Directory Forest. This script creates and uses the domain administrator
# account
# there are potential for mistakes and destructive actions. USE AT YOUR OWN RISK!!
# This is the first script in the Active Directory Series that will establish the first
# Active Directory Domain Controller. This script will unlock the local administrator account
# this account will become the Domain Administrator.
#
# This script will install the required Windows features that are required for Active
# Directory. This script will install the prerequisites for Active Directory, then create a
# one-time executed script on the login after the reboot. This script will reboot the host
# a total of 2 times to add the windows features, create the forest, and promote the domain controller.
#
# Variables for this script
# $password - this is the password necessary to unlock the administrator account
# - and is used in both runs of the AD build.
# $FullDomainName - the full name for the AD Domain example: CESA.corp
# $ShortDomainName - the short name for the AD Domain example: CESA
# $encrypted - you must encrypt the password so that you can use it as you set up your domain controller
# $addsmodule02 - this is the text block that will be used to create the RunOnceScript that will finish the installation
# - of the domain controller.
# $RunOnceKey - this is the key that will create the command to complete the installation of the domain controller.
Try {
#
# Start the logging in the C:\DoimainJoin directory
#
Start-Transcript -Path "C:\DomainJoin\stage1.txt"
# Global Variables
```



```

$password="P@ssw0rd123!!"
# Set the Administrator Password and activate the Domain Admin Account
#
net user Administrator $password /logonpasswordchg:no /active:yes
# Install the Windows features necessary for Active Directory
# Features
# - .NET Core
# - Active Directory Domain Services
# - Remote Active Directory Services
# - DNS Services
#
Install-WindowsFeature NET-Framework-Core
Install-WindowsFeature AD-Domain-Services
Install-WindowsFeature RSAT-ADDS
Install-WindowsFeature RSAT-DNS-Server
# Create text block for the new script that will be ran once on reboot
#
$addsmodule02 = @"
#ps1_sysnative
Try {
Start-Transcript -Path C:\DomainJoin\stage2.txt
`$password = "P@ssw0rd123!!"
`$FullDomainName = "cesa.corp"
`$ShortDomainName = "CESA"
`$encrypted = ConvertTo-SecureString `$password -AsPlainText -Force
Import-Module ADDSDeployment
Install-ADDSTranscript -Path C:\DomainJoin\stage2.txt
-CreateDnsDelegation:`$false ``
-DatabasePath "C:\Windows\NTDS" ``
-DomainMode "WinThreshold" ``
-DomainName `$FullDomainName ``
-DomainNetbiosName `$ShortDomainName ``
-ForestMode "WinThreshold" ``
-InstallDns:`$true ``
-LogPath "C:\Windows\NTDS" ``
-NoRebootOnCompletion:`$false ``
-SysvolPath "C:\Windows\SYSVOL" ``
-SafeModeAdministratorPassword `$encrypted ``
-Force:`$true
} Catch {
Write-Host $_
} Finally {
Stop-Transcript
}
"@
Add-Content -Path "C:\DomainJoin\ADDCmodule2.ps1" -Value $addsmodule02
# Adding the run once job
#
$RunOnceKey = "HKLM:\Software\Microsoft\Windows\CurrentVersion\RunOnce"
set-itemproperty $RunOnceKey "NextRun" ('C:\Windows\System32\WindowsPowerShell\v1.0\Powershell.exe -
executionPolicy Unrestricted -File ' + "C:\DomainJoin\ADDCmodule2.ps1")
# End the logging
#
} Catch {
Write-Host $_
} Finally {
Stop-Transcript
}
# Last step is to reboot the local host

```




```
#
Restart-Computer -ComputerName "localhost" -Force
```

Appendix B: ActiveDirectoryInit2.ps1

```
#ps1_sysnative
#####
# Title: ActiveDirectoryInit2.ps1
# Version & Date: v1 31 Oct 2018
# Creator: john.s.parker@oracle.com
# Warning: This script is a representation of how to use PowerShell to create an Active Directory Domain controller
#          and build the first DC in a new Active Directory Forest. This script creates and uses the domain administrator
#          account
#          there are potential for mistakes and destructive actions. USE AT YOUR OWN RISK!!
# This is the second script in the Active Directory Series that will establish the second
# Active Directory Domain Controller. This script will unlock the local administrator account.
#
# This script will install the required Windows features that are required for Active
# Directory. This script will install the prerequisites for Active Directory. This script will reboot the host after it has added the
# Windows features installed the Active Directory Services and promoted the domain controller.
#
# Variables for this script
# $password - this is the password necessary to unlock the administrator account
#           - and is used in both runs of the AD build.
# $DomainName - this is the full name of the domain that you will be adding the DC
# $DomainUser - this account must have the Domain Admin role
# $EncryptedPass - the encrypted password
# $Credential - the encrypted domain
# $DnsServer - this is the private IP address of the Primary Domain Controller
Try {
Start-Transcript -Path "C:\DomainJoin\Stage3.txt" -Force
$Password="P@ssw0rd123!!"
$DomainName="cesa.corp"
$DomainUser="cesa\administrator"
$EncryptedPass = ConvertTo-SecureString $Password -AsPlainText -Force
$Credential = New-Object -TypeName System.Management.Automation.PSCredential -ArgumentList $DomainUser,
$EncryptedPass
$DnsServer = '192.168.0.1'
#Set the Administrator Password and activate the Domain Admin Account
net user Administrator $Password /logonpasswordchg:no /active:yes
#####
# Create the Second Domain Controller
#
#####
Install-WindowsFeature NET-Framework-Core
Install-WindowsFeature AD-Domain-Services
Install-WindowsFeature RSAT-ADDS
Install-WindowsFeature RSAT-DNS-Server
Set-DnsClientServerAddress -InterfaceAlias Ethernet -ServerAddresses $DnsServer
Install-ADDSDomainController -InstallDns -Credential $Credential -DomainName $DomainName -
SafeModeAdministratorPassword $EncryptedPass -Force -NoRebootOnCompletion
```


```
} Catch {
Write-Host $_
} Finally {
Stop-Transcript
}
start-sleep -s 300
Restart-Computer -ComputerName "localhost" -Force
```

Appendix C: AddComputer.ps1

```
#####
# Title: AddComputer.ps1
# Version & Date: v1 31 Oct 2018
# Creator: lawrence.gabriel@oracle.com & john.s.parker@oracle.com
# Warning: This script is a representation of how to use PowerShell to create an Active Directory Domain controller
#          and build the first DC in a new Active Directory Forest. This script creates and uses the domain administrator
#          account
#          there are potential for mistakes and destructive actions. USE AT YOUR OWN RISK!!
# This is the third script in the Active Directory Series that will join a computer to your new Active Directory Domain. This
# script
# will create the computer account to the Active Directory Domain. You will need to use an account that has the Add
# Computer domain role.
# Source:
# From https://docs.microsoft.com/en-us/powershell/module/microsoft.powershell.management/add-computer?view=powershell-5.1#examples
# Variables for this script
# $NewComputerName - this is the name of the new computer that you want to add to your domain
#
## Run as Administrator on a domain computer.
$NewComputerName = "WS16CN001"
New-ADComputer -Name $NewComputerName -AccountPassword (ConvertTo-SecureString 'TempJoinPA$$' -
AsPlainText -Force)
```

Appendix D: NewComputer.ps1

```
#ps1_sysnative
#####
# Title: newcomputer.ps1
# Version & Date: v1 31 Oct 2018
# Creator: lawrence.gabriel@oracle.com & john.s.parker@oracle.com
# Warning: This script is a representation of how to use PowerShell to create an Active Directory Domain controller
#          and build the first DC in a new Active Directory Forest. This script creates and uses the domain administrator
#          account
#          there are potential for mistakes and destructive actions. USE AT YOUR OWN RISK!!
# This is the forth script in the Active Directory Series that will join a computer to your new Active Directory Domain. This
# script
# will join the newly created host to an Active Directory Domain.
#
# Variables for this script
# $DnsServer - this is the private IP address of the Primary Domain Controller
# $DnsServer2 - this is the private IP address of the Secondary Domain Controller
# $DomainToJoin - this is the full name of the domain you want to join.
# $JoinCred - this will be the encrypted credential
#
```



```
Try {
Start-Transcript -Path "C:\DomainJoin\Stage4.txt" -Force
$DnsServer = '192.168.0.1'
$DnsServer2 = '192.168.0.2'
$DomainToJoin = 'cesa.corp'
#####
# Sets the DNS to the DC.
#####
Set-DnsClientServerAddress -InterfaceAlias Ethernet -ServerAddresses ($DnsServer, $DnsServer2)
#####
# Build the one time use password
#####
$JoinCred = New-Object pscredential -ArgumentList ([pscustomobject]@{
    UserName = $null
    Password = (ConvertTo-SecureString -String 'TempJoinPA$$' -AsPlainText -Force)[0]
})
Add-Computer -Domain $DomainToJoin -Options UnsecuredJoin,PasswordPass -Credential $JoinCred
} Catch {
Write-Host $_
} Finally {
Stop-Transcript
}
#####
#
# This wait is to ensure that the Add-Computer command finishes before the restart.
#
#####
start-sleep -s 300
Restart-Computer -ComputerName "localhost" -Force
```



Oracle Corporation, World Headquarters
500 Oracle Parkway
Redwood Shores, CA 94065, USA

Worldwide Inquiries
Phone: +1.650.506.7000
Fax: +1.650.506.7200

CONNECT WITH US



Integrated Cloud Applications & Platform Services

Copyright © 2019, Oracle and/or its affiliates. All rights reserved. This document is provided for information purposes only, and the contents hereof are subject to change without notice. This document is not warranted to be error-free, nor subject to any other warranties or conditions, whether expressed orally or implied in law, including implied warranties and conditions of merchantability or fitness for a particular purpose. We specifically disclaim any liability with respect to this document, and no contractual obligations are formed either directly or indirectly by this document. This document may not be reproduced or transmitted in any form or by any means, electronic or mechanical, for any purpose, without our prior written permission.

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

Intel and Intel Xeon are trademarks or registered trademarks of Intel Corporation. All SPARC trademarks are used under license and are trademarks or registered trademarks of SPARC International, Inc. AMD, Opteron, the AMD logo, and the AMD Opteron logo are trademarks or registered trademarks of Advanced Micro Devices. UNIX is a registered trademark of The Open Group. 0119

Creating Active Directory Domain Services in Oracle Cloud Infrastructure
January 2019

Authors: John S Parker (john.s.parker@oracle.com); Lawrence Gabriel (lawrence.gabriel@oracle.com); based on work by Barry Shilmover



Oracle is committed to developing practices and products that help protect the environment.