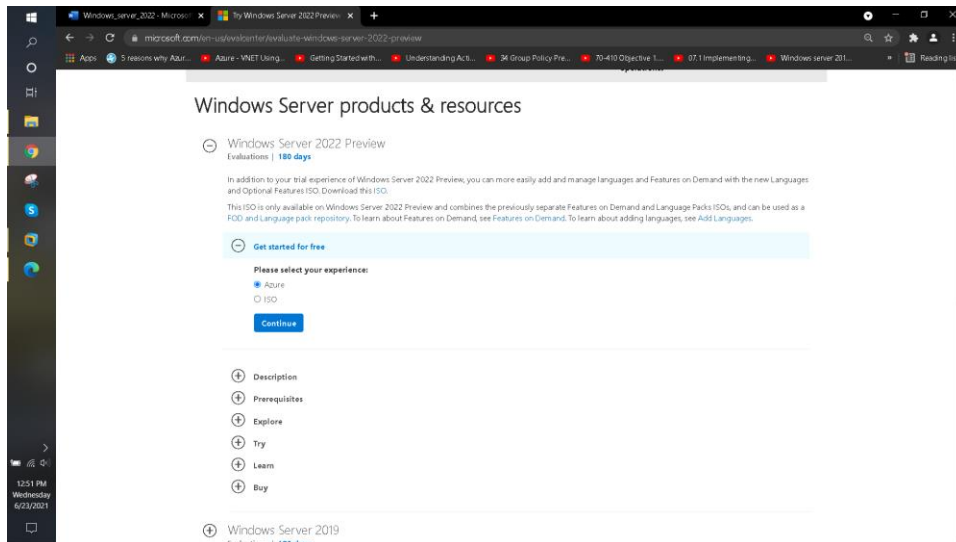
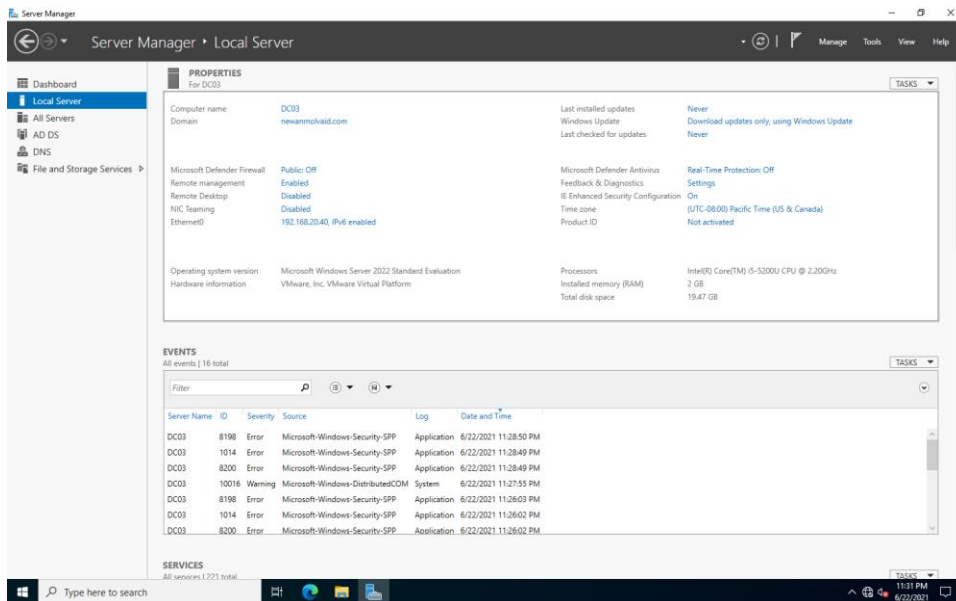


From below screen shot it can be seen that we are on 2022 server which is in preview version till 23/June/2021. However, testing our existing PowerShell commands and promotion of this server as a domain controller can give us fact check on proof of concept (POC). Latest windows server not only help to keep our AD evergreen but also provide new features that add more value and are cloud ready.



Below screen shot help us to see that windows server 2022 can be successfully be promoted as a DC.

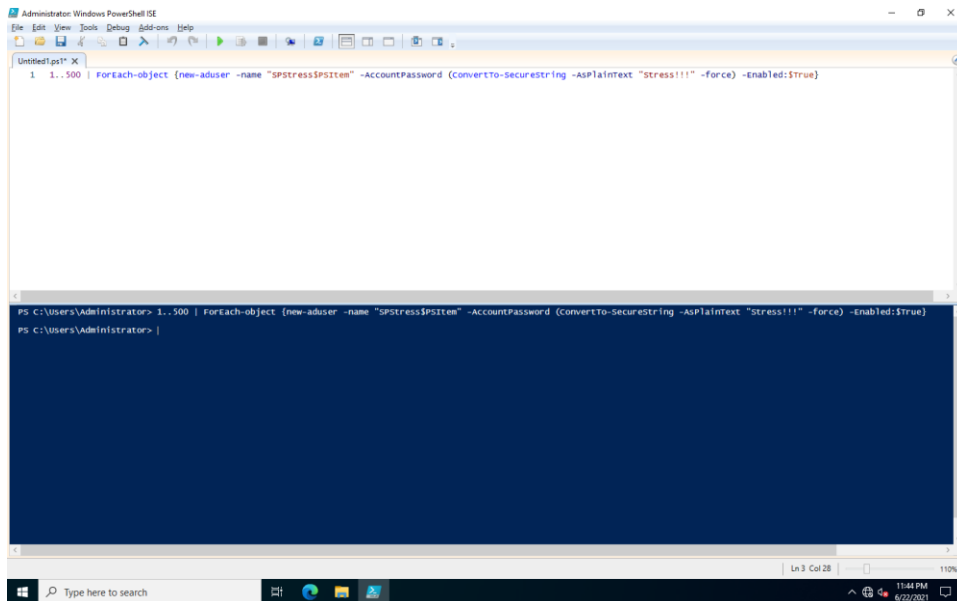
Now we will test how user creation and existing PowerShell command will work on it.



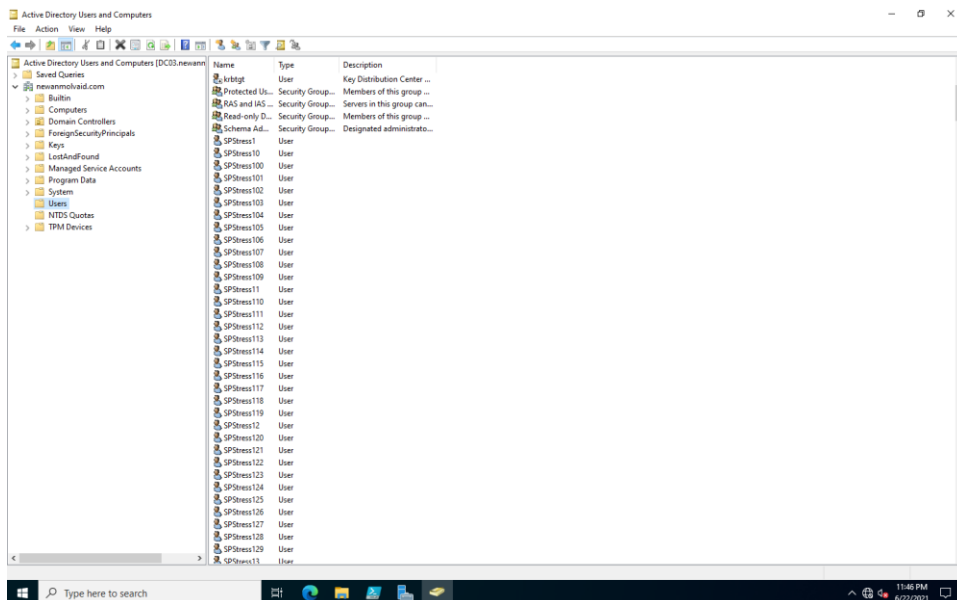
Creating 500 test users with their account enabled.

Command used:

1..500 | ForEach-object {new-aduser -name "SPStress\$PSItem" -AccountPassword (ConvertTo-SecureString -AsPlainText "Stress!!!" -force) -Enabled:\$True}



By default, these users are created in container name users. To see them navigate to users container in ADUC as shown below.



Or you can use below PowerShell command.

Get-AdUser -Filter {name -like "S\*"}>

```

1 Get-Aduser -Filter {name -like "s*"}

DistinguishedName : CN=SPStress499,CN=Users,DC=newanmolvoid,DC=com
Enabled            : True
GivenName          : 
Name              : SPStress499
ObjectClass        : user
ObjectGUID         : 3c719f0a-c9cf-4e7f-b44c-4f230a1ce11f
SamAccountName     : SPStress499
SID               : S-1-5-21-661321814-3623857380-3040710237-1601
Surname           : 
UserPrincipalName  : 

DistinguishedName : CN=SPStress500,CN=Users,DC=newanmolvoid,DC=com
Enabled            : True
GivenName          : 
Name              : SPStress500
ObjectClass        : user
ObjectGUID         : 14966081-079e-4c82-b594-d95eb8451b51
SamAccountName     : SPStress500
SID               : S-1-5-21-661321814-3623857380-3040710237-1602
Surname           : 
UserPrincipalName  : 

PS C:\Users\Administrator>

```

Below command is used to get computer details from AD domain.

```

Get-ADComputer -Filter * | select -Property @{n='computername';e={$PSItem.Name}},
@{n='status';e={$PSItem.Enabled}}, @{n='FQDN';e={$PSItem.DNSHostName}},
@{n='samACCOUNTNAME';e={$PSItem.SamAccountName}},
@{n='LOCATION';e={$PSItem.DistinguishedName}}, @{n='ObjectClass';e={$PSItem.ObjectClass}} |
Format-Table -AutoSize

```

```

1 Get-ADComputer -Filter * | select -Property @{n='computername';e={$PSItem.Name}}, @{n='status';e={$PSItem.Enabled}}, @{n='FQDN';e={$PSItem.DNSHostName}}, @{n='samACCOUNTNAME';e={$PSItem.SamAccountName}}, @{n='LOCATION';e={$PSItem.DistinguishedName}}, @{n='ObjectClass';e={$PSItem.ObjectClass}} |
2 Format-Table -AutoSize

PS C:\> Get-ADComputer -Filter * | select -Property @{n='computername';e={$PSItem.Name}}, @{n='status';e={$PSItem.Enabled}}, @{n='FQDN';e={$PSItem.DNSHostName}}, @{n='samACCOUNTNAME';e={$PSItem.SamAccountName}}, @{n='LOCATION';e={$PSItem.DistinguishedName}}, @{n='ObjectClass';e={$PSItem.ObjectClass}} |
Format-Table -AutoSize

computername status FQDN          samACCOUNTNAME LOCATION                                ObjectClass
-----
DC03          True  DC03.newanmolvoid.com DC03$          CN=DC03,OU=Domain Controllers,DC=newanmolvoid,DC=com computer

PS C:\>

```

Getting forest details in a tabular format.

```
Get-ADForest | select -Property @{n='DomainNamingMaster'; e = {$PSItem.DomainNamingMaster}},@{n='Forestmode'; e =
{$PSItem.ForestMode }}, @ {n='Sites'; e = {$PSItem.Sites}}, @ {n='Rootdomain'; e = {$PSItem.RootDomain}},
@ {n='GlobalCatalog'; e = {$PSItem.GlobalCatalogs}},@ {n='SchemaMaster'; e = {$PSItem.SchemaMaster}}, @ {n='Application
Partition'; e = {$PSItem.ApplicationPartitions}},@ {n='partitionContainer'; e = {$PSItem.PartitionsContainer}} | Format-Table -
AutoSize
```

```
PS C:\> Get-ADForest | select -Property @{n='DomainNamingMaster'; e = {$PSItem.DomainNamingMaster}},@{n='Forestmode'; e = {$PSItem.ForestMode }}, @ {n='Sites'; e = {$PSItem.Sites}}, @ {n='Rootdomain'; e = {$PSItem.RootDomain}}, @ {n='GlobalCatalog'; e = {$PSItem.GlobalCatalogs}},@ {n='SchemaMaster'; e = {$PSItem.SchemaMaster}}, @ {n='Application Partition'; e = {$PSItem.ApplicationPartitions}},@ {n='partitionContainer'; e = {$PSItem.PartitionsContainer}} | Format-Table -AutoSize
```

DomainNamingMaster	Forestmode	Sites	Rootdomain	GlobalCatalog	SchemaMaster	Application Partition
DC03.newarmolva.com	windows2016forest	Default-First-Site-Name	newarmolva.com	DC03.newarmolva.com	DC03.newarmolva.com	[DC=ForestDnsZones,DC=newarmolva.com,DC=com,DC=doma...

Below command is used to get Dc details in domain.

```
Get-ADDomainController | select -Property @{n='DC with path'; e={$PSItem.ComputerObjectDN}}, @ {n='DefaultPartition'; e={$PSItem.DefaultPartition}}, @ {n='Domain'; e={$PSItem.Domain}}, @ {n='Enabled'; e={$PSItem.Enabled}}, @ {n='Forest'; e={$PSItem.Forest}}, @ {n='HostName'; e={$PSItem.HostName}}, @ {n='InvocationId'; e={$PSItem.InvocationId}}, @ {n='IPv4Address'; e={$PSItem.IPv4Address}}, @ {n='IPv6Address'; e={$PSItem.IPv6Address}}, @ {n='Global Catalog'; e={$PSItem.IsGlobalCatalog}}, @ {n='Read Only DC'; e={$PSItem.IsReadOnly}}, @ {n='LdapPort'; e={$PSItem.LdapPort}}, @ {n='Name'; e={$PSItem.Name}}, @ {n='NTDSSettingsObjectDN'; e={$PSItem.NTDSSettingsObjectDN}}, @ {n='Operating System'; e={$PSItem.OperatingSystem}}, @ {n='OperationMasterRoles'; e={$PSItem.OperationMasterRoles}}, @ {n='Partitions'; e={$PSItem.Partitions}} | Format-Table -AutoSize
```

```

PS C:\> Get-ADGroup -filter * | select -Property @({n='DistinguishedName'; e={$PSItem.DistinguishedName}}, @({n='GroupCategory'; e={$PSItem.GroupCategory}}, @({n='GroupScope'; e={$PSItem.GroupScope}}, @({n='Name'; e={$PSItem.Name}}, @({n='ObjectClass'; e={$PSItem.ObjectClass}}, @({n='SamAccountName'; e={$PSItem.SamAccountName}}, @({n='SID'; e={$PSItem.SID}} | Format-Table -AutoSize

```

Below PowerShell command is used to get group details in domain:

```

Get-ADGroup -filter * | select -Property @({n='DistinguishedName'; e={$PSItem.DistinguishedName}}, @({n='GroupCategory'; e={$PSItem.GroupCategory}}, @({n='GroupScope'; e={$PSItem.GroupScope}}, @({n='Name'; e={$PSItem.Name}}, @({n='ObjectClass'; e={$PSItem.ObjectClass}}, @({n='SamAccountName'; e={$PSItem.SamAccountName}}, @({n='SID'; e={$PSItem.SID}} | Format-Table -AutoSize

```

```

PS C:\> Get-ADDomain -filter * | select -Property @({n='DistinguishedName'; e={$PSItem.DistinguishedName}}, @({n='GroupCategory'; e={$PSItem.GroupCategory}}, @({n='GroupScope'; e={$PSItem.GroupScope}}, @({n='Name'; e={$PSItem.Name}}, @({n='ObjectClass'; e={$PSItem.ObjectClass}}, @({n='SamAccountName'; e={$PSItem.SamAccountName}}, @({n='SID'; e={$PSItem.SID}} | Format-Table -AutoSize

```

Below command is used for getting details of the domain:

```

Get-ADDomain | select -Property @({n='child domains'; e={$PSItem.ChildDomains}}, @({n='DNSRoot'; e={$PSItem.DNSRoot}}, @({n='ParentDomain'; e={$PSItem.ParentDomain}}, @({n='PDCEmulator'; e={$PSItem.PDCEmulator}}, @({n='ReplicaDirectoryServers'; e={$PSItem.ReplicaDirectoryServers}}, @({n='RIDMaster'; e={$PSItem.RIDMaster}}, @({n='DomainMode'; e={$PSItem.DomainMode}}, @({n='Forest'; e={$PSItem.Forest}}, @({n='InfrastructureMaster

```

```
'e={$PSItem.InfrastructureMaster}}, @{n='Name';e={$PSItem.Name}}, @{n='NetBIOSName';e={$PSItem.NetBIOSName}} |
Format-Table -AutoSize
```

```
PS C:\> Get-ADDomain | select -Property @('child domains';e={$PSItem.ChildDomains}), @('DNGRoot';e={$PSItem.DNGRoot}), @('ParentDomain';e={$PSItem.ParentDomain}), @('PDCemulator';e={$PSItem.PDCemulator}), @('ReplicaDirectoryServers';e={$PSItem.ReplicaDirectoryServers}), @('RIDMaster';e={$PSItem.RIDMaster}), @('DomainMode';e={$PSItem.DomainMode}), @('Forest';e={$PSItem.Forest}), @('InfrastructureMaster';e={$PSItem.InfrastructureMaster}), @('Name';e={$PSItem.Name}), @('NetBIOSName';e={$PSItem.NetBIOSName})

child domains DNGRoot ParentDomain PDCemulator ReplicaDirectoryServers RIDMaster DomainMode Forest InfrastructureMaster Name
-----
newamovaid.com DC01.newamovaid.com DC01.newamovaid.com DC01.newamovaid.com DC01.newamovaid.com windows2016Domain newamovaid.com DC01.newamovaid.com newamovaid
```

Below command is used to get OU structure details in domain:

```
Get-ADOrganizationalUnit -Filter * | select -Property @{n='City';e={$PSItem.City}}, @{n='Country';e={$PSItem.Country}},
@{n='DistinguishedName';e={$PSItem.DistinguishedName}},
@{n='LinkedGroupPolicyObjects';e={$PSItem.LinkedGroupPolicyObjects}}, @{n='ManagedBy';e={$PSItem.ManagedBy}},
@{n='Name';e={$PSItem.Name}}, @{n='ObjectClass';e={$PSItem.ObjectClass}}, @{n='State';e={$PSItem.State}} | Format-Table -
AutoSize
```

```
PS C:\> Get-ADOrganizationalUnit -Filter * | select -Property @('City';e={$PSItem.City}), @('Country';e={$PSItem.Country}), @('DistinguishedName';e={$PSItem.DistinguishedName}), @('LinkedGroupPolicyObjects';e={$PSItem.LinkedGroupPolicyObjects}), @('ManagedBy';e={$PSItem.ManagedBy}), @('Name';e={$PSItem.Name}), @('ObjectClass';e={$PSItem.ObjectClass}), @('State';e={$PSItem.State}) | Format-Table -AutoSize

City Country DistinguishedName ManagedBy Name ObjectClass State
-----
ou=Domain Controllers,DC=newamovaid,DC=com CN=[6AC178AC-65AF-11D2-945F-00C04FB884F9], CN=Policies, CN=System, DC=newamovaid, DC=com Domain Controllers organizationalunit
```

Below command is used for extracting each object from AD:

```
PS C:\> Get-ADObject -Filter * | select -Property @('DistinguishedName','e[($item.DistinguishedName)], @('Name','e[($item.Name)], @('ObjectClass','e[($item.ObjectClass)] | Format-Table -AutoSize
```