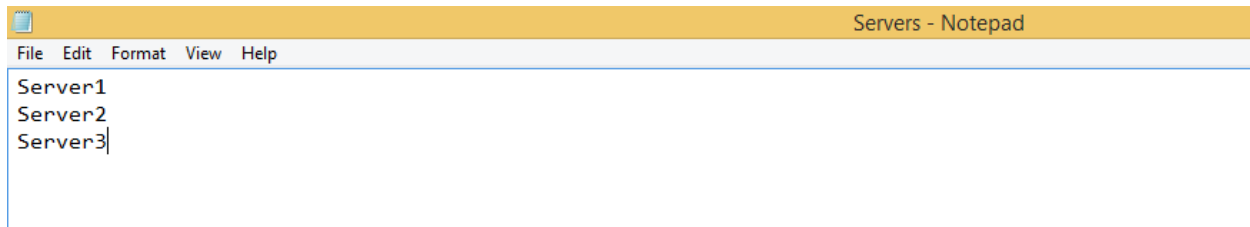


## How to Search Windows Servers 2012 R2 Roles & Feature (Installed / Not Installed)

Step: 1

Create TXT File

Write your Server Machine Names:



```
Servers - Notepad
File Edit Format View Help
Server1
Server2
Server3
```

Copy this File in C Drive with Path like C:\Servers.txt

Step: 2

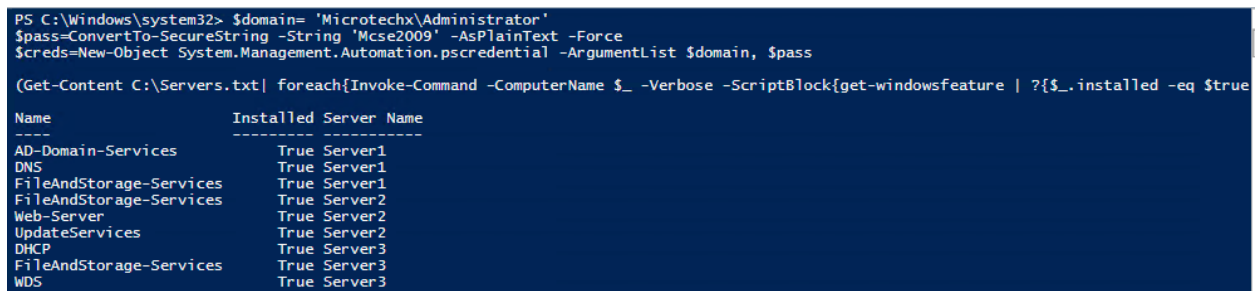
Add your Domain Name & Credentials in Script

Query the Roles Installed On Servers:

```
$domain= 'Microtechx\Administrator'
$pass=ConvertTo-SecureString -String 'Mcse2009' -AsPlainText -Force
$creds=New-Object System.Management.Automation.pscredential -ArgumentList $domain,
$pass

(Get-Content C:\Servers.txt| foreach{Invoke-Command -ComputerName $_ -Verbose -
ScriptBlock{get-windowsfeature | ?{$_ .installed -eq $true -and $_ .featuretype -eq
'Role'}} | select name, installed -ExcludeProperty subfeatures} -Credential $creds}) |
ft -Property Name, Installed, @{name='Server Name';expression={$_.pscomputername}} -
AutoSize
```

Output:



```
PS C:\Windows\system32> $domain= 'Microtechx\Administrator'
$pass=ConvertTo-SecureString -String 'Mcse2009' -AsPlainText -Force
$creds=New-Object System.Management.Automation.pscredential -ArgumentList $domain, $pass

(Get-Content C:\Servers.txt| foreach{Invoke-Command -ComputerName $_ -Verbose -ScriptBlock{get-windowsfeature | ?{$_ .installed -eq $true

Name                Installed Server Name
-----
AD-Domain-Services  True     Server1
DNS                  True     Server1
FileAndStorage-Services True     Server1
FileAndStorage-Services True     Server2
Web-Server           True     Server2
UpdateServices      True     Server2
DHCP                 True     Server3
FileAndStorage-Services True     Server3
WDS                  True     Server3
```

Query the Roles Not Installed On Servers:

You have to put False at place of True Argument:

```
$_installed -eq $true with $_installed -eq $false
```

```
$domain= 'Microtechx\Administrator'  
$pass=ConvertTo-SecureString -String 'Mcse2009' -AsPlainText -Force  
$creds=New-Object System.Management.Automation.pscredential -ArgumentList $domain,  
$pass
```

```
(Get-Content C:\Servers.txt| foreach{Invoke-Command -ComputerName $_ -Verbose -  
ScriptBlock{get-windowsfeature | ?{$_installed -eq $false -and $_featuretype -eq  
'Role'}} | select name, installed -ExcludeProperty subfeatures} -Credential $creds}) |  
ft -Property Name, Installed, @{name='Server Name';expression={$_.pscomputername}} -  
AutoSize
```

```
PS C:\Windows\system32> $domain= 'Microtechx\Administrator'  
$pass=ConvertTo-SecureString -String 'Mcse2009' -AsPlainText -Force  
$creds=New-Object System.Management.Automation.pscredential -ArgumentList $domain, $pass  
  
(Get-Content C:\Servers.txt| foreach{Invoke-Command -ComputerName $_ -Verbose -ScriptBlock{get-windowsfeature | ?{$_installed -eq $false -and $_featuretype -eq 'Role'}} | :  
Name  
-----  
Installed Server Name  
-----  
AD-Certificate False Server1  
ADFS-Federation False Server1  
ADLDS False Server1  
ADRMS False Server1  
Application-Server False Server1  
DHCP False Server1  
Fax False Server1  
Hyper-V False Server1  
NPAS False Server1  
Print-Services False Server1  
RemoteAccess False Server1  
Remote-Desktop-Services False Server1  
VolumeActivation False Server1  
Web-Server False Server1  
WDS False Server1  
ServerEssentialsRole False Server1  
UpdateServices False Server1  
AD-Certificate False Server2  
AD-Domain-Services False Server2  
ADFS-Federation False Server2  
ADLDS False Server2  
ADRMS False Server2  
Application-Server False Server2  
DHCP False Server2  
DNS False Server2  
Fax False Server2  
Hyper-V False Server2  
NPAS False Server2  
Print-Services False Server2
```

Step: 3

Find Feature Installed On Servers:

Replace Role with Feature -eq 'Role'-eq 'Feature'

```
$domain= 'Microtechx\Administrator'  
$pass=ConvertTo-SecureString -String 'Mcse2009' -AsPlainText -Force  
$creds=New-Object System.Management.Automation.pscredential -ArgumentList $domain,  
$pass
```

```
(Get-Content C:\Servers.txt| foreach{Invoke-Command -ComputerName $_ -Verbose -  
ScriptBlock{get-windowsfeature | ?{$_installed -eq $true -and $_featuretype -eq  
'Feature'}} | select name, installed -ExcludeProperty subfeatures} -Credential $creds})  
| ft -Property Name, Installed, @{name='Server Name';expression={$_.pscomputername}} -  
AutoSize
```

```

PS C:\Windows\system32> $domain= 'Microtechx\Administrator'
$pass=ConvertTo-SecureString -String 'Mcse2009' -AsPlainText -Force
$creds=New-Object System.Management.Automation.pscredential -ArgumentList $domain, $pass

(Get-Content C:\Servers.txt| foreach{Invoke-Command -ComputerName $_ -Verbose -ScriptBlock{get-windowsfeature | ?{$_installed -eq $true -and $_.featuretype -eq 'Feature'}}

Name                               Installed Server Name
----                               -
NET-Framework-Features             True Server1
NET-Framework-Core                 True Server1
NET-Framework-45-Features          True Server1
NET-Framework-45-Core              True Server1
NET-WCF-Services45                 True Server1
NET-WCF-TCP-PortSharing45          True Server1
GPMC                                True Server1
RSAT                                True Server1
RSAT-Role-Tools                     True Server1
RSAT-AD-Tools                       True Server1
RSAT-AD-PowerShell                 True Server1
RSAT-ADD5                           True Server1
RSAT-AD-AdminCenter                True Server1
RSAT-ADD5-Tools                     True Server1
RSAT-DNS-Server                     True Server1
FS-SMB1                             True Server1
User-Interfaces-Infra              True Server1
Server-Gui-Mgmt-Infra              True Server1
Server-Gui-Shell                    True Server1
PowerShellRoot                     True Server1
PowerShell                           True Server1
PowerShell-V2                       True Server1
PowerShell-ISE                      True Server1
Wow64-Support                       True Server1
NET-Framework-Features             True Server2
NET-Framework-Core                 True Server2
NET-HTTP-Activation                True Server2
NET-Non-HTTP-Activ                 True Server2
NET-Framework-45-Features          True Server2
NET-Framework-45-Core              True Server2
NET-Framework-45-ASPNET            True Server2
NET-WCF-Services45                 True Server2

```

Find Features Not Installed on Servers:

`$_installed -eq $true` with `$_installed -eq $false`

```

$domain= 'Microtechx\Administrator'
$pass=ConvertTo-SecureString -String 'Mcse2009' -AsPlainText -Force
$creds=New-Object System.Management.Automation.pscredential -ArgumentList $domain,
$pass

```

```

(Get-Content C:\Servers.txt| foreach{Invoke-Command -ComputerName $_ -Verbose -
ScriptBlock{get-windowsfeature | ?{$_installed -eq $false -and $_.featuretype -eq
'Feature'}} | select name, installed -ExcludeProperty subfeatures} -Credential $creds})
| ft -Property Name, Installed, @{name='Server Name';expression={$_.pscomputername}} -
AutoSize

```

```

PS C:\Windows\system32> $domain= 'Microtechx\Administrator'
$pass=ConvertTo-SecureString -String 'Mcse2009' -AsPlainText -Force
$creds=New-Object System.Management.Automation.pscredential -ArgumentList $domain, $pass

(Get-Content C:\Servers.txt| foreach{Invoke-Command -ComputerName $_ -Verbose -ScriptBlock{get-windowsfeature | ?{$_installed -eq $false -and $_.featuretype -eq 'Feature'}}

Name                               Installed Server Name
----                               -
NET-HTTP-Activation                False Server1
NET-Non-HTTP-Activ                 False Server1
NET-Framework-45-ASPNET            False Server1
NET-WCF-HTTP-Activation45          False Server1
NET-WCF-MSMQ-Activation45         False Server1
NET-WCF-Pipe-Activation45         False Server1
NET-WCF-TCP-Activation45          False Server1
BITS                                False Server1
BITS-IIS-Ext                       False Server1
BITS-Compact-Server               False Server1
BitLocker                           False Server1
BitLocker-NetworkUnlock            False Server1
BranchCache                         False Server1
NFS-Client                          False Server1
Data-Center-Bridging               False Server1
Direct-Play                         False Server1
EnhancedStorage                     False Server1
Failover-Clustering                False Server1
Web-WHC                             False Server1
InkAndHandwritingServices           False Server1
Internet-Print-Client              False Server1
IPAM                                False Server1
ISNS                                False Server1
LPR-Port-Monitor                   False Server1
ManagementOdata                    False Server1

```