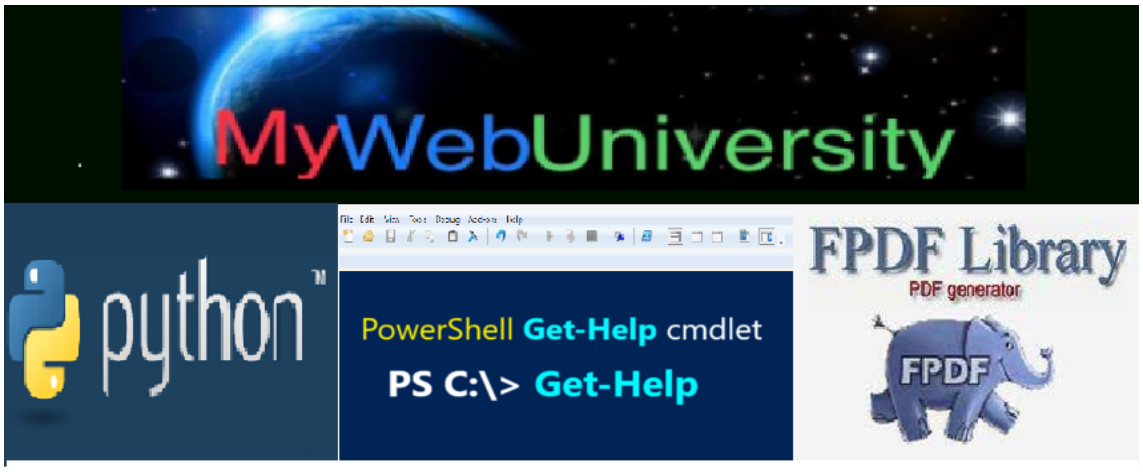

Full credit is given to all the above companies including the Operating System that this PDF file was generated!

Windows PowerShell Get-Help on Cmdlet 'Set-AzDiskSecurityProfile'

PS:\>Get-HELP Set-AzDiskSecurityProfile -Full

NAME

Set-AzDiskSecurityProfile

SYNOPSIS

Set SecurityProfile on managed disk

SYNTAX

Set-AzDiskSecurityProfile [-Disk] <Microsoft.Azure.Commands.Compute.Automation.Models.PSDisk> [-DefaultProfile <Microsoft.Azure.Commands.Common.Authentication.Abstractions.Core.IAzureContextContainer>] [-SecureVMDiskEncryptionSet <System.String>] -SecurityType <System.String> [-Confirm] [-WhatIf] [<CommonParameters>]

DESCRIPTION

Set the SecurityProfile on managed disks.

PARAMETERS

-DefaultProfile <Microsoft.Azure.Commands.Common.Authentication.Abstractions.Core.IAzureContextContainer>

The credentials, account, tenant, and subscription used for communication with Azure.

Required? false

Position? named

Default value None

Accept pipeline input? False

Accept wildcard characters? false

-Disk <Microsoft.Azure.Commands.Compute.Automation.Models.PSDisk>

Disk Security Profile

Required? true

Position? 0

Default value None

Accept pipeline input? True (ByPropertyName, ByValue)

Accept wildcard characters? false

-SecureVMDiskEncryptionSet <System.String>

ResourceId of the disk encryption set to use for enabling encryption at rest.

Required? false

Position? named

Default value None

Accept pipeline input? True (ByPropertyName)

Accept wildcard characters? false

-SecurityType <System.String>

Security Type of Disk

Required? true

Position? named

Default value None

Accept pipeline input? True (ByPropertyName)

Accept wildcard characters? false

-Confirm <System.Management.Automation.SwitchParameter>

Prompts you for confirmation before running the cmdlet.

Required? false

Position? named

Default value False

Accept pipeline input? False

Accept wildcard characters? false

-WhatIf <System.Management.Automation.SwitchParameter>

Shows what would happen if the cmdlet runs. The cmdlet is not run.

Required? false

Position? named

Default value False

Accept pipeline input? False

Accept wildcard characters? false

<CommonParameters>

This cmdlet supports the common parameters: Verbose, Debug,

ErrorAction, ErrorVariable, WarningAction, WarningVariable,

OutBuffer, PipelineVariable, and OutVariable. For more information, see

about_CommonParameters (https:/go.microsoft.com/fwlink/?LinkID=113216).

INPUTS

Microsoft.Azure.Commands.Compute.Automation.Models.PSDisk

System.String

OUTPUTS

Microsoft.Azure.Commands.Compute.Automation.Models.PSDisk

NOTES

---------- Example 1 ----------

$diskconfig = New-AzDiskConfig -DiskSizeGB 10 -AccountType PremiumLRS -OsType Windows -CreateOption FromImage;

$image = '/subscriptions/0000000-0000-0000-0000-000000000000/resourceGroups/ResourceGroup01/providers/Microsoft.Compute/images/TestImage123';

$diskconfig = Set-AzDiskImageReference -Disk $diskconfig -Id $image -Lun 0;

$diskconfig = Set-AzDiskSecurityProfile -Disk $diskconfig -SecurityType "TrustedLaunch";

$disk = New-AzDisk -ResourceGroupName 'ResourceGroup01' -DiskName 'Disk01' -Disk $diskconfig;

$disk.Properties.SecurityProfile.SecurityType == "TrustedLaunch";

Customers can set the SecurityType of managed Disks.

Example 2: Create a Disk with a Disk Encrption Set with the encryption type of ConfidentialVM_DiskEncryptedWithCustomerKey

$Location = "northeurope";

$KeyVaultName = "val" + $rgname;

$KeyName = "key" + $rgname;

$DesName= "des" + $rgname;

```
$KeySize = 3072;

$SecurePassword = "Password" | ConvertTo-SecureString -AsPlainText -Force;
$User = "Username";
$Cred = New-Object System.Management.Automation.PSCredential ($User, $SecurePassword);

New-AzKeyVault -Name $KeyVaultName -Location $Location -ResourceGroupName $ResourceGroupName -Sku Premium -EnablePurgeProtection -EnabledForDiskEncryption;

# Add Key vault Key
Add-AzKeyVaultKey -VaultName $KeyVaultName -Name $KeyName -Size $KeySize -KeyOps wrapKey,unwrapKey -KeyType RSA -Destination HSM -Exportable -UseDefaultCVMPolicy;

# Capture Keyvault and key details
$KeyVaultId = (Get-AzKeyVault -VaultName $KeyVaultName -ResourceGroupName $ResourceGroupName).ResourceId;
$KeyUrl = (Get-AzKeyVaultKey -VaultName $KeyVaultName -KeyName $KeyName).Key.Kid;

# Create new DES Config and DES
$diskEncryptionType = "ConfidentialVmEncryptedWithCustomerKey";
$desConfig = New-AzDiskEncryptionSetConfig -Location $Location -SourceVaultId $keyvaultId -KeyUrl $keyUrl -IdentityType SystemAssigned -EncryptionType
$diskEncryptionType;
New-AzDiskEncryptionSet -ResourceGroupName $ResourceGroupName -Name $DesName -DiskEncryptionSet $desConfig;
$diskencset = Get-AzDiskEncryptionSet -ResourceGroupName $ResourceGroupName -Name $desName;

# Assign DES Access Policy to key vault
$desIdentity = (Get-AzDiskEncryptionSet -Name $DesName -ResourceGroupName $ResourceGroupName).Identity.PrincipalId;
Set-AzKeyVaultAccessPolicy -VaultName $KeyVaultName -ResourceGroupName $ResourceGroupName -ObjectId $desIdentity -PermissionsToKeys wrapKey,unwrapKey,get
-BypassObjectIdValidation;
```

```
$diskSecurityType = "ConfidentialVM_DiskEncryptedWithCustomerKey";
$diskName = "diskname";
 $diskconfig = New-AzDiskConfig -AccountType Premium_LRS -OsType Windows -CreateOption FromImage -Location $Location;
$diskconfig = Set-AzDiskImageReference -Disk $diskconfig -Id "/Subscriptions/e37510d7-33b6-4676-886f-ee75bcc01871/Providers/Microsoft.Compute/Locations/northeurope/Publishers/MicrosoftWindowsServer/ArtifactTypes/VMImage/Offers/windows-cvm/Skus/2019-datacenter-cvm/Versions/latest";
$diskconfig = Set-AzDiskSecurityProfile -Disk $diskconfig -SecurityType $diskSecurityType -SecureVMDiskEncryptionSet $diskencset.id;
New-AzDisk -ResourceGroupName $ResourceGroupName -DiskName $diskName -Disk $diskconfig;
$disk = Get-AzDisk -ResourceGroupName $ResourceGroupName -DiskName $diskName;
# Verify the SecurityType value.
# $disk.Properties.SecurityProfile.SecurityType returns "ConfidentialVM";
```

Example 3: Set the SecurityType to Standard to avoid TrustedLaunch defaulting.

```
$rgname = <Resource Group Name>;
$loc = <Azure Region>;
New-AzResourceGroup -Name $rgname -Location $loc -Force;
$securityTypeStnd = "Standard";

# Standard SecurityType
  $diskconfig = New-AzDiskConfig -Location $loc -DiskSizeGB 1 -AccountType "Premium_LRS" -OsType "Windows" -CreateOption "Empty" -HyperVGeneration "V1";
$diskname = "diskstnd" + $rgname;
$diskconfig = Set-AzDiskSecurityProfile -Disk $diskconfig -SecurityType $securityTypeStnd;
$diskPr = New-AzDisk -ResourceGroupName $rgname -DiskName $diskname -Disk $diskconfig;
$disk = Get-AzDisk -ResourceGroupName $rgname -DiskName $diskname;
```

```
# Verify $disk.SecurityProfile is null;
```

RELATED LINKS

Online Version: https://learn.microsoft.com/powershell/module/az.compute/set-azdisksecurityprofile