



Projetlys



GLOBAL AZURE VIRTUAL



RCS : 488 173 428
SIRET : 488 173 428 00035
TVA intra : FR37488173428
D-U-N-S® : 28 585 9612

Projetlys
39 rue des Docks, 69009 LYON
04.72.57.19.00



Informations Document :

Référence du document	Version	Date de diffusion	Département émetteur
PLYS-44-17	1.0		

Document rédigé par :

Document validé par :

Nicolas BONNET	Architecte technique		Choisissez un élément.
Choisissez un élément.	Choisissez un élément.	Choisissez un élément.	Choisissez un élément.
Choisissez un élément.	Choisissez un élément.	Choisissez un élément.	Choisissez un élément.



Suivi des modifications :

Version	Date	Objet



Liste des destinataires :

Société	Nom, Prénom	Fonction	Présent	Diffusion
			<input type="checkbox"/>	<input type="checkbox"/>
			<input type="checkbox"/>	<input type="checkbox"/>
			<input type="checkbox"/>	<input type="checkbox"/>
			<input type="checkbox"/>	<input type="checkbox"/>
			<input type="checkbox"/>	<input type="checkbox"/>



1. Mise en place de Tags à l'aide de Powershell.....	3
1.1 Connexion à Azure	3
1.2 Création d'un groupe de ressource	4
1.3 Création d'une machine virtuelle	4
1.4 Mise en place des Tags	8
2. Mise en place de Azure Policy	10



1. Mise en place de Tags à l'aide de Powershell

Depuis une console Powershell, saisissez la commande **Install-Module -Name Az -AllowClobber -Scope CurrentUser**

```
Administrator: Windows PowerShell
PS C:\Users\nicolas> Install-Module -Name Az -AllowClobber -Scope CurrentUser

NuGet provider is required to continue
PowerShellGet requires NuGet provider version '2.8.5.201' or newer to interact with NuGet-based repositories. The NuGet
provider must be available in 'C:\Program Files\PackageManagement\ProviderAssemblies' or
'C:\Users\nicolas\AppData\Local\PackageManagement\ProviderAssemblies'. You can also install the NuGet provider by
running 'Install-PackageProvider -Name NuGet -MinimumVersion 2.8.5.201 -Force'. Do you want PowerShellGet to install
and import the NuGet provider now?
[Y] Yes [N] No [S] Suspend [?] Help (default is "Y"): █
```

1.1 Connexion à Azure

Réduisez la politique d'exécution des scripts Powershell à l'aide de la commande **Set-executionpolicy unrestricted**

```
Administrator: Windows PowerShell
PS C:\Users\nicolas> set-executionpolicy unrestricted

Execution Policy Change
The execution policy helps protect you from scripts that you do not trust. Changing the execution policy might expose
you to the security risks described in the about_Execution_Policies help topic at
https://go.microsoft.com/fwlink/?LinkID=135170. Do you want to change the execution policy?
[Y] Yes [A] Yes to All [N] No [L] No to All [S] Suspend [?] Help (default is "N"): y
PS C:\Users\nicolas> █
```

Importer le module **az** à l'aide de la commande **Import-module az** puis procédez à la connexion à la plateforme Azure en utilisant la commande **Connect-azaccount**

```
Administrator: Windows PowerShell
PS C:\Users\nicolas> connect-azaccount
```

Saisissez les identifiants et mot de passe d'un compte administrateur puis procédez à la connexion.



1.2 Création d'un groupe de ressource

Depuis l'invite de commande powershell, saisissez la commande **New-AzResourceGroup -Name GlobalAzure -Location "WestEurope"** pour procéder à la création d'un ressource group.

```
Sélection Administrateur : Windows PowerShell
PS C:\Windows\system32> New-AzResourceGroup -Name GlobalAzure -Location "WestEurope"

ResourceGroupName : GlobalAzure
Location           : westeurope
ProvisioningState  : Succeeded
Tags               :
```

1.3 Création d'une machine virtuelle

Il est nécessaire dans un premier temps de saisir les informations de la machine virtuelle. Exécutez les commandes ci-dessous afin de configurer dans des variables le nom d'utilisateur et son mot de passe.

- > \$LocalAdminUser = "LocalAdmin"
- > \$LocalAdminSecurePassword = ConvertTo-SecureString "MyVMPass0" -AsPlainText -Force

```
Administrator: Windows PowerShell
PS C:\Users\nicolas> $VMLocalAdminSecurePassword = ConvertTo-SecureString "MyVMPass0" -AsPlainText -Force
PS C:\Users\nicolas>
```

Indiquez la région où sera stockée la machine virtuelle

- > \$LocationName = "WestEurope"

```
Administrator: Windows PowerShell
PS C:\Windows\system32> $LocationName = "WestEurope"
PS C:\Windows\system32>
```

Les informations nécessaires à la création d'une machine virtuelle vont être configuré dans des variables. Ces dernières seront utilisées par la suite. Les informations renseignées sont le groupe de ressource, le nom de l'ordinateur, le nom de la VM, la taille de la VM, le nom du réseau virtuel, le nom de la carte réseau, le nom du sous-réseau, l'adresse du sous-réseau et l'adresse du réseau de la machine virtuel.



- > \$ResourceGroupName = "GlobalAzure"
- > \$ComputerName = "VMGlobalAzure"
- > \$VMName = "VMPowershell"
- > \$VMSize = "Standard_A1_v2"
- > \$NetworkName = "VnetFormAz"
- > \$NICName = "VMNICAz"
- > \$SubnetName = "SubnetLan"
- > \$SubnetAddressPrefix = "10.0.0.0/24"
- > \$VnetAddressPrefix = "10.0.0.0/16"

```

Administrateur : Windows PowerShell

PS C:\Windows\system32> $ResourceGroupName = "GlobalAzure"
PS C:\Windows\system32> $ComputerName = "VMGlobalAzure"
PS C:\Windows\system32> $VMName = "VMPowershell"
PS C:\Windows\system32> $VMSize = "Standard_A1_v2"
PS C:\Windows\system32> $NetworkName = "VnetFormAz"
PS C:\Windows\system32> $NICName = "VMNICAz"
PS C:\Windows\system32> $SubnetName = "SubnetLan"
PS C:\Windows\system32> $SubnetAddressPrefix = "10.0.0.0/24"
PS C:\Windows\system32> $VnetAddressPrefix = "10.0.0.0/16"
PS C:\Windows\system32>
  
```

Le réseau peut maintenant être configuré. Dans un premier temps, nous allons procéder à la création du virtual network et du sous-réseau. Certaines variables créées précédemment vont être utilisées. Dans un premier temps nous procédons à la configuration de la variable subnet qui contient la configuration du sous-réseau.

- > \$Subnet = New-AzVirtualNetworkSubnetConfig -Name \$SubnetName -AddressPrefix \$SubnetAddressPrefix

```

Administrator: Windows PowerShell

PS C:\Users\nicolas> $Subnet = New-AzVirtualNetworkSubnetConfig -Name $SubnetName -AddressPrefix $SubnetAddressPrefix
WARNING: Breaking changes in the cmdlet 'New-AzVirtualNetworkSubnetConfig' :
WARNING: - Update Property Name

WARNING: Cmdlet invocation changes :
  Old Way : -ResourceId
  New Way : -NatGatewayId
WARNING: - Update Property Name

WARNING: Cmdlet invocation changes :
  Old Way : -InputObject
  New Way : -NatGateway
WARNING: NOTE : Go to https://aka.ms/azps-changewarnings for steps to suppress this breaking change warning, and other
information on breaking changes in Azure PowerShell.
PS C:\Users\nicolas>
  
```

Le réseau virtuel de la futur VM peut maintenant être créé. Exécutez la commande suivante pour procédez à sa création.



- `$Vnet = New-AzVirtualNetwork -Name $NetworkName -ResourceGroupName $ResourceGroupName -Location $LocationName -AddressPrefix $VnetAddressPrefix -Subnet $Subnet`

Administrator: Windows PowerShell

```
PS C:\Users\nicolas> $Vnet = New-AzVirtualNetwork -Name $NetworkName -ResourceGroupName $ResourceGroupName -Location $LocationName -AddressPrefix $VnetAddressPrefix -Subnet $Subnet
PS C:\Users\nicolas>
```

Nous allons procéder à la création de la carte réseau de la futur VM. Exécutez la commande suivante pour procédez à sa création.

- `$NIC = New-AzNetworkInterface -Name $NICName -ResourceGroupName $ResourceGroupName -Location $LocationName -SubnetId $Vnet.Subnets[0].Id`

Administrator: Windows PowerShell

```
PS C:\Users\nicolas> $NIC = New-AzNetworkInterface -Name $NICName -ResourceGroupName $ResourceGroupName -Location $LocationName -SubnetId $Vnet.Subnets[0].Id
PS C:\Users\nicolas>
```

Les identifiants du compte local de la machine virtuelle vont être ajouté à une variable \$credential. Cette dernière sera utilisée par la suite.

- `$Credential = New-Object System.Management.Automation.PSCredential ($LocalAdminUser,$LocalAdminSecurePassword);`

Administrateur : Windows PowerShell

```
PS C:\Windows\system32> $Credential = New-Object System.Management.Automation.PSCredential ($LocalAdminUser,$LocalAdminSecurePassword);
PS C:\Windows\system32>
```

La machine virtuelle peut désormais être créé. Pour cela utiliser les commandes suivantes. Dans un premier temps nous allons effectuer la création d'une variable contenant la configuration de la machine virtuelle. Les différentes commandes permettent de configurer les différents éléments de la VM

- `$VirtualMachine = New-AzVMConfig -VMName $VMName -VMSize $VMSize`



Administrator: Windows PowerShell

```
PS C:\Users\nicolas> $VirtualMachine = New-AzVMConfig -VMName $VMName -VMSize $VMSize
PS C:\Users\nicolas>
```

- \$VirtualMachine = Set-AzVMOperatingSystem -VM \$VirtualMachine -Windows -ComputerName \$ComputerName -Credential \$Credential

Administrator: Windows PowerShell

```
PS C:\Users\nicolas> $VirtualMachine = Set-AzVMOperatingSystem -VM $VirtualMachine -Windows -ComputerName $ComputerName -Credential $Credential
PS C:\Users\nicolas>
```

- \$VirtualMachine = Add-AzVMNetworkInterface -VM \$VirtualMachine -Id \$NIC.Id

Administrator: Windows PowerShell

```
PS C:\Users\nicolas> $VirtualMachine = Add-AzVMNetworkInterface -VM $VirtualMachine -Id $NIC.Id
```

- \$VirtualMachine = Set-AzVMSourceImage -VM \$VirtualMachine -PublisherName 'MicrosoftWindowsServer' -Offer 'WindowsServer' -Skus '2016-Datacenter' -Version latest

Administrator: Windows PowerShell

```
PS C:\Users\nicolas> $VirtualMachine = Set-AzVMSourceImage -VM $VirtualMachine -PublisherName 'MicrosoftWindowsServer' -Offer 'WindowsServer' -Skus '2016-Datacenter' -Version latest
PS C:\Users\nicolas>
```

La création de la machine virtuelle peut désormais débuter. Exécutez la commande ci-dessous pour démarrer la création de la VM.

- New-AzVM -ResourceGroupName \$ResourceGroupName -Location \$LocationName -VM \$VirtualMachine -Verbose

```
PS C:\Users\nicolas> New-AzVM -ResourceGroupName $ResourceGroupName -Location $LocationName -VM $VirtualMachine -Verbose
VERBOSE: Performing the operation "New" on target "VMPowershell".
```




La VM est correctement créé.

1.4 Mise en place des Tags

Un tags va être ajouté au groupe de ressource précédemment créé. Depuis l'invite de commande Powershell, saisissez la commande ci-dessous

```
> Set-AzResourceGroup -Name GlobalAzure -Tag @{ "Dept"="GlobalAzure"; "Environment"="Test" }
```

```
Sélection Administrateur : Windows PowerShell
PS C:\Windows\system32> Set-AzResourceGroup -Name GlobalAzure -Tag @{ "Dept"="GlobalAzure"; "Environment"="Test" }
```

```
ResourceGroupName : GlobalAzure
Location           : westeurope
ProvisioningState   : Succeeded
Tags               :
                    Name      Value
                    -
                    Environment Test
                    Dept       GlobalAzure
```

Le tag est bien rajouté au groupe de ressource

GlobalAzure
Groupe de ressources

Rechercher (Ctrl+/) << + Ajouter ≡ Modifier les colonnes 🗑 Delete resource group ↻ Actualiser → Déplacer ⬇ Exporter au format CSV

Abonnement (modifier) : Visual Studio Premium with MSDN Déploiements : Aucun déploiement

ID d'abonnement : 1839a405-62b2-4115-8d50-711703d55b78

Étiquettes (modifier) : Environment : Test Dept : GlobalAzure

Vue d'ensemble | Journal d'activité | Contrôle d'accès (IAM) | Étiquettes

A l'aide de Powershell, nous allons maintenant assigner une étiquette à la machine virtuelle. Nous allons récupérer dans un premier temps la machine virtuelle VMPowershell présentes dans le groupe de ressource

```
> $resource = Get-AzResource -ResourceName VMPowershell -ResourceGroupName GlobalAzure
```



```
Administrateur : Windows PowerShell
PS C:\Windows\system32> $resource = Get-AzResource -ResourceName VMPowershell -ResourceGroupName GlobalAzure
PS C:\Windows\system32>
```

La variable contient bien la machine virtuelle et celle-ci ne possède pas de tag.

```
Administrateur : Windows PowerShell
PS C:\Windows\system32> $resource

Name                : VMPowershell
ResourceGroupName   : GlobalAzure
ResourceType        : Microsoft.Compute/virtualMachines
Location            : westeurope
ResourceId           : /subscriptions/1839a405-62b2-4115-8d50-711703d55b78/resourceGroups/GlobalAzure/providers/Microsoft.
                    Compute/virtualMachines/VMPowershell
Tags                :

PS C:\Windows\system32>
```

Assigner un tag à la machine virtuelle en exécutant la commande suivante

➤ Set-AzResource -Tag @{ "Dept"="GlobalAzure"; "Type"="VM" } -ResourceId \$resource.ResourceId -Force

```
Administrateur : Windows PowerShell
PS C:\Windows\system32> Set-AzResource -Tag @{ "Dept"="GlobalAzure"; "Type"="VM" } -ResourceId $resource.ResourceId -Force
```

Le tag a été correctement positionné

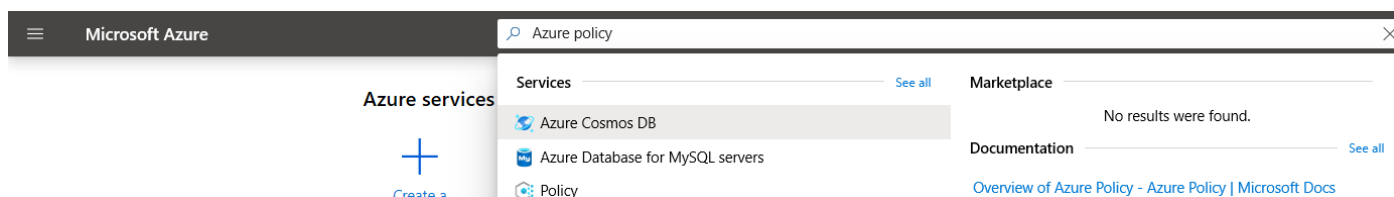
```
Administrateur : Windows PowerShell
PS C:\Windows\system32> $resource

Name                : VMPowershell
ResourceGroupName   : GlobalAzure
ResourceType        : Microsoft.Compute/virtualMachines
Location            : westeurope
ResourceId           : /subscriptions/1839a405-62b2-4115-8d50-711703d55b78/resourceGroups/GlobalAzure/providers/Microsoft.
                    Compute/virtualMachines/VMPowershell
Tags                :
                    Name Value
                    ----
                    Dept GlobalAzure
                    Type VM
```

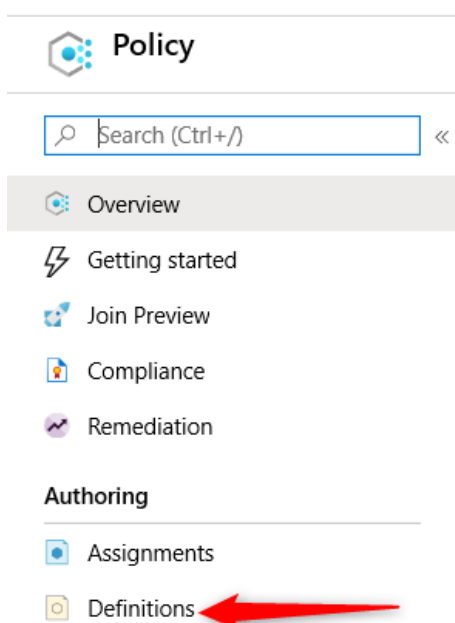


2. Mise en place de Azure Policy

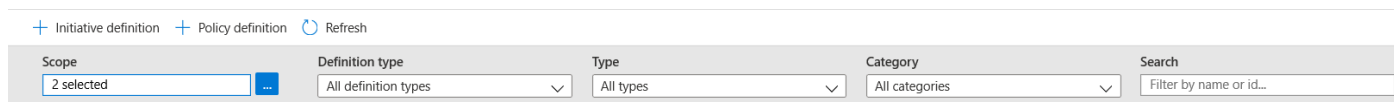
L'ensemble des ressources doivent posséder un même jeu de Tags. Depuis le portail Azure, saisissez **Azure policy** dans la barre recherche puis cliquez sur **Policy**



Cliquez sur **Définitions** dans le menu afin de créer une nouvelle Policy.



Cliquez sur **+ Policy Definition** afin d'ajouter une nouvelle définition de stratégie.



Cliquez sur le bouton pour procéder à la sélection de l'abonnement Azure.



Policy definition

New Policy definition

BASICS

Definition location *



Sélectionnez l'abonnement puis cliquez sur **Select**.

Definition location

Subscription

Microsoft Azure Sponsorship

Saisissez le nom souhaité puis sélectionnez la catégorie Tags.

BASICS

Definition location *

Microsoft Azure Sponsorship



Name * ⓘ

Policy for Tags



Description

Category ⓘ

☐ Create new
☒ Use existing

Tags



Saisissez la policy suivante puis cliquez sur **Save**.

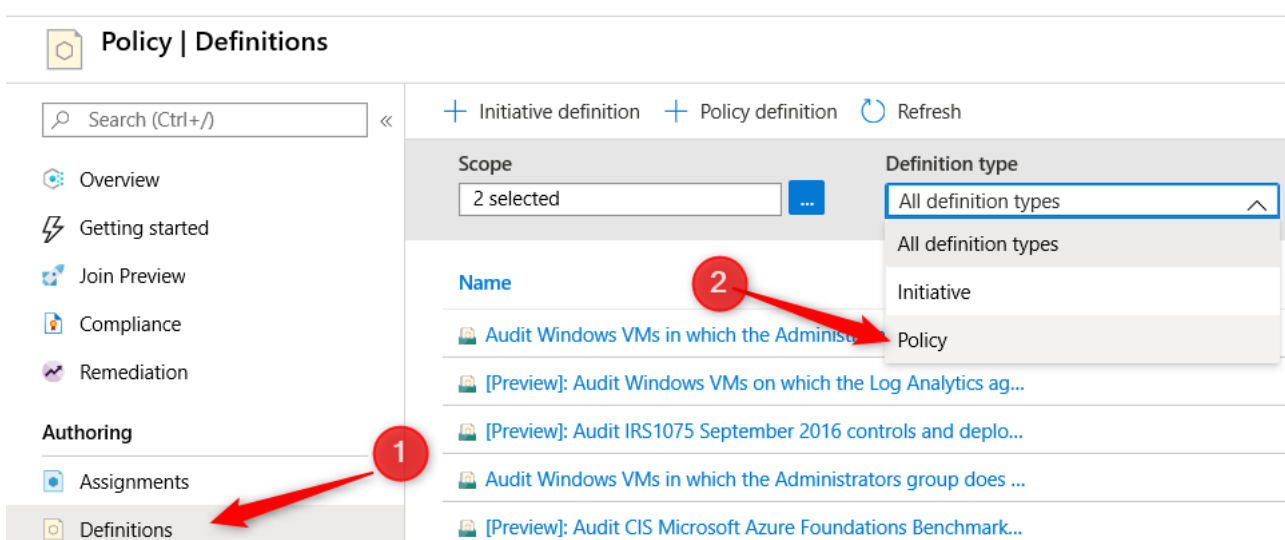
{

```

"policyRule": {
  "if": {
    "anyOf": [{
      "field": "tags['Prod']",
      "notEquals": "[resourcegroup().tags['Prod']]"
    },
    {
      "field": "tags['Prod']",
      "exists": false
    }
  ]
},
"then": {
  "effect": "modify",
  "details": {
    "roleDefinitionIds": [
      "/providers/microsoft.authorization/roleDefinitions/b24988ac-6180-42a0-ab88-20f7382dd24c"
    ],
    "operations": [{
      "operation": "addOrReplace",
      "field": "tags['Prod']",
      "value": "[resourcegroup().tags['Azure']]"
    }]
  }
}
}
}

```

Dans **Policy**, cliquez sur **Definitions** puis dans la liste déroulante **Definitions type** sélectionnez **Policy**.



The screenshot displays the Azure Policy 'Definitions' page. On the left, the 'Definitions' link is highlighted in the sidebar (marked with a red circle 1). The main content area shows a table of policy definitions. A dropdown menu for 'Definition type' is open, and the 'Policy' option is selected (marked with a red circle 2).



Cliquez sur **Policy for tags**

+ Initiative definition + Policy definition Refresh

Scope	Definition type	Type	Category	Search
2 selected	Policy	All types	All categories	Filter by name or id...

Name	Definition location	Policies	Type	Definition type	Category
test	Visual Studio Premium with MSDN		Custom	Policy	Tags
Policy for Tags	Microsoft Azure Sponsorship		Custom	Policy	Tags

Cliquez sur **Assign**.

Assign	Edit definition	Duplicate definition	Delete definition
--------	-----------------	----------------------	-------------------

Une fenêtre apparaît, cliquez sur Next



Policy for Tags

Assign policy

Basics

Parameters Remediation Review + create

Scope

Scope [Learn more about setting the scope *](#)

Microsoft Azure Sponsorship



Exclusions

Optionally select resources to exempt from the policy assignment



Basics

Policy definition

Policy for Tags

Assignment name * ⓘ

Policy for Tags

Description

Policy enforcement ⓘ

Review + create

Cancel

Previous

Next

Cochez la case **Create a remediation Task** puis sélectionnez la région souhaitée.



Policy for Tags

Assign policy

Basics Parameters **Remediation** Review + create

By default, this assignment will only take effect on newly created resources. Existing resources can be updated via a remediation task after the policy is assigned. For deployIfNotExists policies, the remediation task will deploy the specified template. For modify policies, the remediation task will edit tags on the existing resources.

☒ Create a remediation task ⓘ

Policy to remediate

Policy for Tags

Managed Identity

Policies with the deployIfNotExists and modify effect types need the ability to deploy resources and edit tags on existing resources respectively. To do this, a managed identity will be created for you.

[Learn more about Managed Identity.](#)

☒ Create a Managed Identity

Managed identity location *

(Europe) West Europe

Permissions

This identity will also be given the following permissions:

Contributor

Review + create

Cancel

Previous

Next

Cliquez sur **Create**

Policy for Tags

Assign policy

Basics Parameters **Remediation** **Review + create**

Basics

Scope	Microsoft Azure Sponsorship
Exclusions	--
Policy definition	Policy for Tags
Assignment name	Policy for Tags
Description	--
Policy enforcement	Enabled
Assigned by	Nicolas BONNET

Parameters

i No parameter changes detected.

Remediation

Create managed identity	Yes
Managed identity location	westeurope
Create a remediation task	Yes

Create

Cancel

Previous

Next



Dans **Policy**, cliquez sur **Overview**. La Policy mais n'est pas encore démarré. Le démarrage prend plusieurs minutes pour débiter.

Home > Policy

Policy

Search (Ctrl+/)

Overview

Getting started

Join Preview

Compliance

Remediation

Authoring

Assignments

Definitions

Related Services

Blueprints (preview)

Scope

2 selected

Overall resource compliance 98%
168 out of 172

Non-compliant initiatives 1
out of 2

Non-compliant policies 21
out of 102

Non-compliant resources 4
out of 172

LEARN MORE
Learn about Policy or Onboarding tutorial

Name	Scope	Compliance state	Resource compliance	Non-Compliant Resources	Non-compliant policies
ASC Default (subscription: 1839a405-62...	Visual Studio Premium with MS...	Non-compliant	98% (168 out of 172)	4	21
Policy for Tags	Microsoft Azure Sponsorship	Not started	100% (0 out of 0)	0	0
Audit Windows VMs in which the Admi...	Microsoft Azure Sponsorship/In...	Compliant	100% (0 out of 0)	0	0

View all

Après l'application de la policy, l'étiquette est appliquée.

Overview

Getting started

Join Preview

Compliance

Remediation

Authoring

Assignments

Definitions

Related Services

Scope

Overall resource compliance 98%
168 out of 172

Non-compliant initiatives 1
out of 2

Non-compliant policies 21
out of 102

Non-compliant resources 4
out of 172

LEARN MORE
Learn about Policy or Onboarding tutorial

Name	Scope	Compliance state	Resource compliance	Non-Compliant Resources	Non-compliant policies
ASC Default (subscription: 1839a405-62...	Visual Studio Premium with MS...	Non-compliant	98% (168 out of 172)	4	21
Policy for Tags	Microsoft Azure Sponsorship	Compliant	100% (0 out of 0)	0	0
Audit Windows VMs in which the Admi...	Microsoft Azure Sponsorship/In...	Compliant	100% (0 out of 0)	0	0

View all

AppNicolas
App Service

Search (Ctrl+/)

Overview

Activity log

Access control (IAM)

Tags

Diagnose and solve problems

Security

Deployment

Browse Stop Swap Restart Delete Get

The new version of Chrome is changing how it handles cross-site and ifra

Resource group (change) : [Linkedin](#)

Status : Running

Location : West Europe

Subscription (change) : [Microsoft Azure Sponsorship](#)

Subscription ID : e665bd79-1208-466c-b372-efdf40b1fd5

Tags (change) : Prod :