COGNITION

LEARN IT. KNOW IT. SHARE IT.

Azure Automation fun

w/ ARM Templates, Runbooks and Hybrid Workers

my DevOps Cloud Automation adventures

About Esther

- **20 years of Technical Consulting**
  - Solutions Architect at **cognition IT**
  - Design | Implement | Troubleshooting

- **Community programs**
  - Citrix Technology Professional (CTP)
  - Microsoft Most Valuable Professional (MVP)
  - CUGC Women in Tech Mentorship program leader

- **DevOps, automation**
  - Automation of Windows | Citrix
  - REST APIs | JSON | PowerShell

Your Presenter: **Esther Barthel** **Solutions Architect**

@virtues_IT http://nl.linkedin.com/in/ebarthel http://www.virtues.it

LEARN IT. KNOW IT. SHARE IT.

Agenda

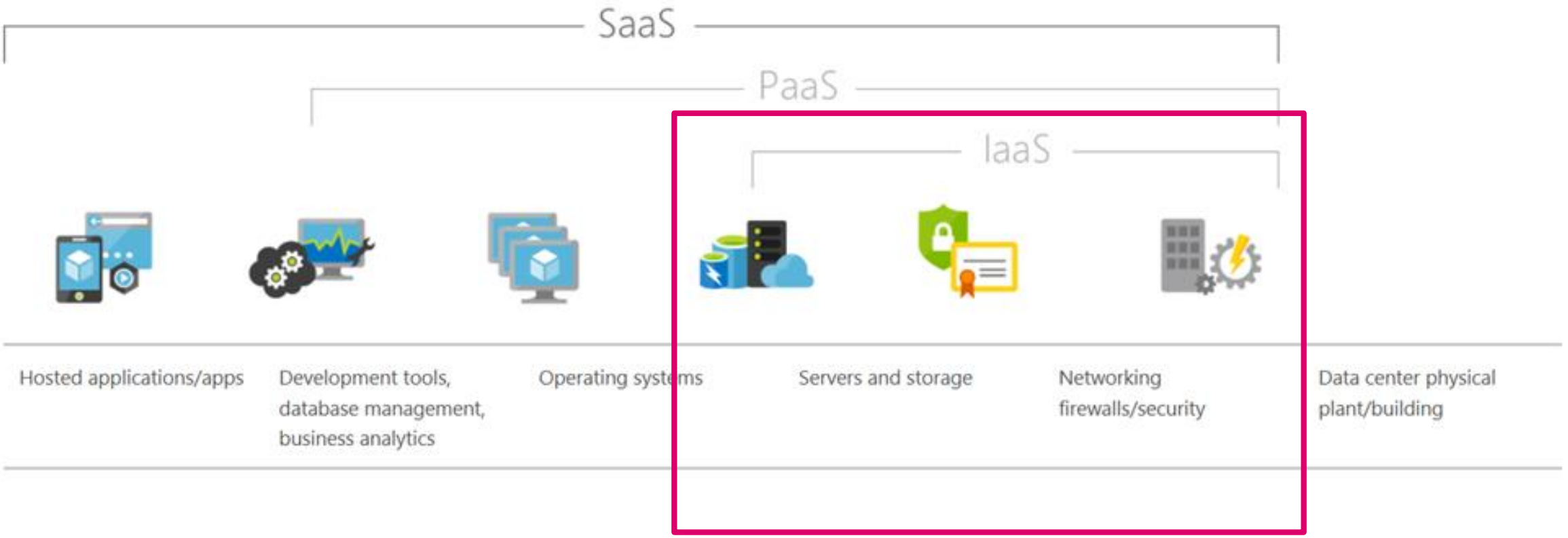
- Azure IaaS
- ARM Templates
  - JSON formatting
  - Az PowerShell
- Azure Automation
  - Runbooks based on PowerShell
  - Hybrid Workers
- Demo

Azure IaaS

SERVICES & OBJECTS

COGNITION

Azure IaaS

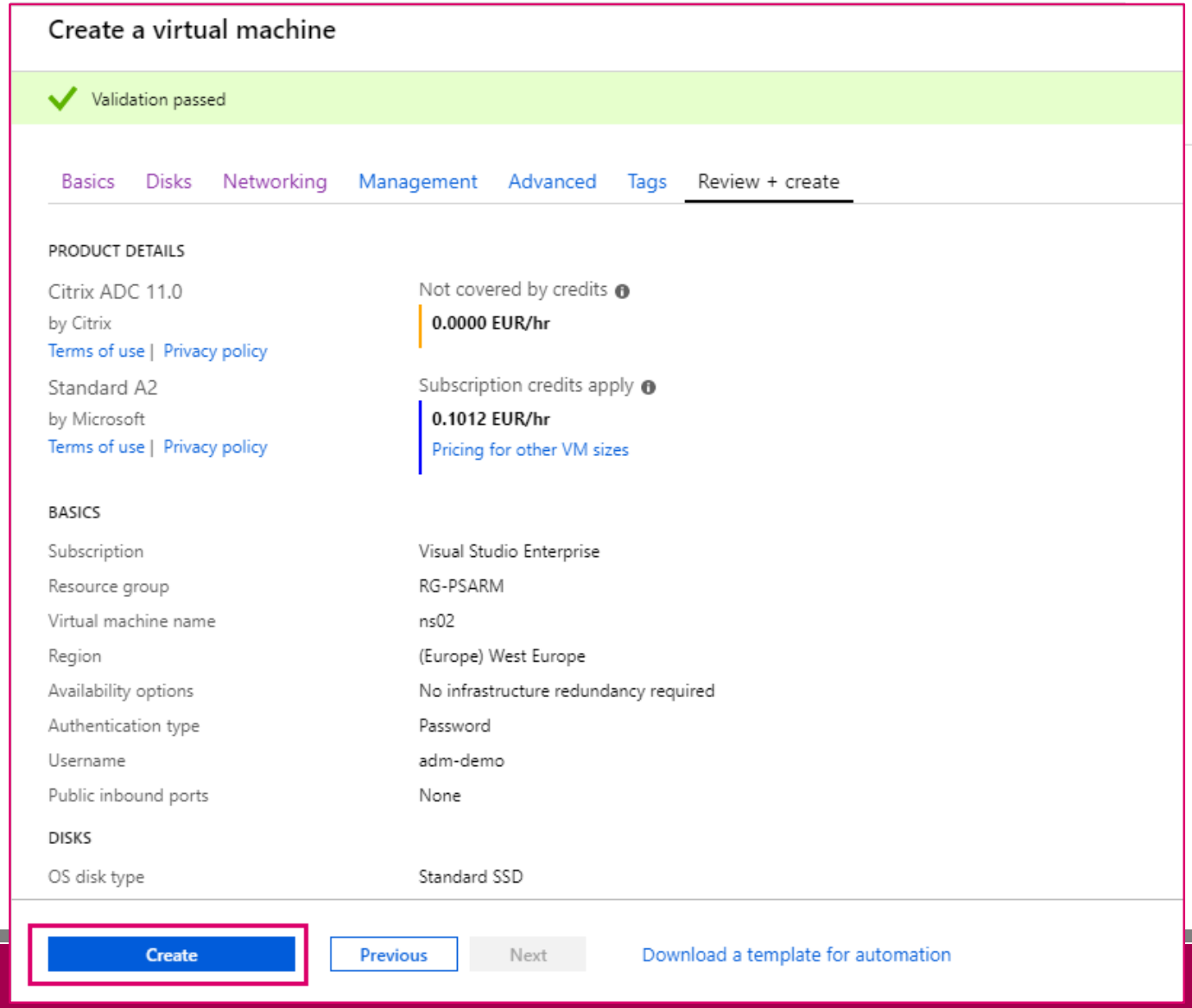


COGNITION

Azure IaaS

Family	Description
A-series	cost-effective, general purpose compute
D(S) -series	cost-effective, general purpose compute (faster CPU and local SSD)
F(s)-series	best value compute capability per core, lower memory footprint
G(S)-series	highest memory footprint
H-series	very high end workloads (eg CFD)
N-series	GPU-enabled. NV for visualisation, NC for compute

Azure IaaS


Create a virtual machine
Validation passed
Basics
Disks
Networking
Management
Advanced
Tags
Review + create
PRODUCT DETAILS
Citrix ADC 11.0
by Citrix
Terms of use | Privacy policy
Standard A2
by Microsoft
Terms of use | Privacy policy
Not covered by credits
0.0000 EUR/hr
Subscription credits apply
0.1012 EUR/hr
Pricing for other VM sizes
BASICS
Subscription
Visual Studio Enterprise
Resource group
RG-PSARM
Virtual machine name
ns02
Region
(Europe) West Europe
Availability options
No infrastructure redundancy required
Authentication type
Password
Username
adm-demo
Public inbound ports
None
DISKS
OS disk type
Standard SSD
Create
Previous
Next
Download a template for automation

ARM Templates

JSON & EXPRESSIONS

COGNITION

ARM Template

… structure of an **Azure Resource Manager template**.

The template consists of **JSON** and expressions that you can use to construct values for your deployment.

sources: https://docs.microsoft.com/en-us/azure/azure-resource-manager/resource-group-authoring-templates
https://docs.microsoft.com/en-us/azure/templates/ (Reference Guide!)

COGNITION

JavaScript Object Notation (JSON)

Syntax rules

- data is in **name/value pairs**

A name/value pair consists of a field name (in double quotes), followed by a colon, followed by a value:

Example

```
"name":"John"
```

COGNITION

JavaScript Object Notation (JSON)

Syntax rules

- data is separated by **commas**

Example

```
{
"employee":{ "name":"John", "age":30, "city":"New York" }
}
```

COGNITION

JavaScript Object Notation (JSON)

Syntax rules

- curly braces {} hold **objects**

Example

```
{
"employee":{ "name":"John", "age":30, "city":"New York" }
}
```

COGNITION

JavaScript Object Notation (JSON)

Syntax rules

- square brackets [] hold **arrays**

Example

```
[ "Ford", "BMW", "Fiat" ]
```

COGNITION

JavaScript Object Notation (JSON)

Nested Arrays in JSON Objects

Values in an array can also be another array, or even another JSON object:

Example

```
myObj = {
    "name":"John",
    "age":30,
    "cars": [
        { "name":"Ford", "models":[ "Fiesta", "Focus", "Mustang" ] },
        { "name":"BMW", "models":[ "320", "X3", "X5" ] },
        { "name":"Fiat", "models":[ "500", "Panda" ] }
    ]
}
```

COGNITION

ARM Template

Template format

In its simplest structure, a template has the following elements:

JSON Copy

```
{
    "$schema": "http://schema.management.azure.com/schemas/2015-01-01/deploymentTemplate.json#",
    "contentVersion": "",
    "parameters": {  },
    "variables": {  },
    "functions": [  ],
    "resources": [  ],
    "outputs": {  }
}
```

sources: https://docs.microsoft.com/en-us/azure/azure-resource-manager/resource-group-authoring-templates
https://docs.microsoft.com/en-us/azure/templates/ (Reference Guide!)

COGNITION

ARM Template

```
{
    "$schema": "http://schema.management.azure.com/schemas/2015-01-01/deploymentTemplate.json#",
    "contentVersion": "",
    "parameters": {
        "<parameter-name>" : {
            "type" : "<type-of-parameter-value>",
            "defaultValue": "<default-value-of-parameter>",
            "allowedValues": [ "<array-of-allowed-values>" ],
            "minValue": <minimum-value-for-int>,
            "maxValue": <maximum-value-for-int>,
            "minLength": <minimum-length-for-string-or-array>,
            "maxLength": <maximum-length-for-string-or-array-parameters>,
            "metadata": {
                "description": "<description-of-the parameter>"
            }
        }
    },
    "variables": {
        "<variable-name>": "<variable-value>",
        "<variable-object-name>": {
            <variable-complex-type-value>
        },
        "<variable-object-name>": {
            "copy": [
        },
        "copy": [
    },
    "functions": [
      {
        "namespace": "<namespace-for-your-function>",
        "members": {
          "<function-name>": {
            "parameters": [
              {
                "name": "<parameter-name>",
                "type": "<type-of-parameter-value>"
              }
            ],
            "output": {
              "type": "<type-of-output-value>",
              "value": "<function-expression>"
            }
          }
        }
      }
    ],
    "resources": [
      {
          "condition": "<boolean-value-whether-to-deploy>",
          "apiVersion": "<api-version-of-resource>",
          "type": "<resource-provider-namespace/resource-type-name>",
          "name": "<name-of-the-resource>",
          "location": "<location-of-resource>",
          "tags": {
              "<tag-name1>": "<tag-value1>",
              "<tag-name2>": "<tag-value2>"
          },
          "comments": "<your-reference-notes>",
          "copy": {
              "name": "<name-of-copy-loop>",
              "count": "<number-of-iterations>",
              "mode": "<serial-or-parallel>",
              "batchSize": "<number-to-deploy-serially>"
          },
          "dependsOn": [
              "<array-of-related-resource-names>"
          ],
          "properties": {
              "<settings-for-the-resource>",
              "copy": [
                  {
                      "name": ,
                      "count": ,
                      "input": {}
                  }
              ]
          },
          "resources": [
              "<array-of-child-resources>"
          ]
      }
    ],
    "outputs": {
        "<outputName>" : {
            "type" : "<type-of-output-value>",
            "value": "<output-value-expression>"
        }
    }
}
```

COGNITION

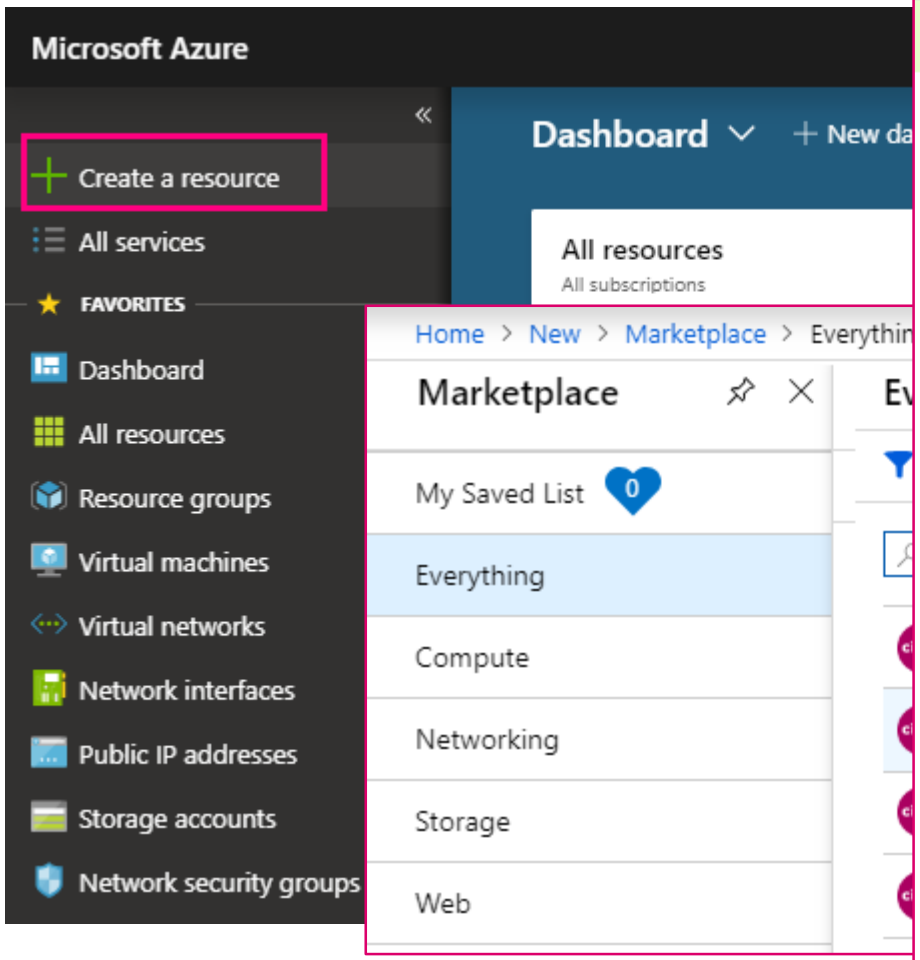
ARM Template

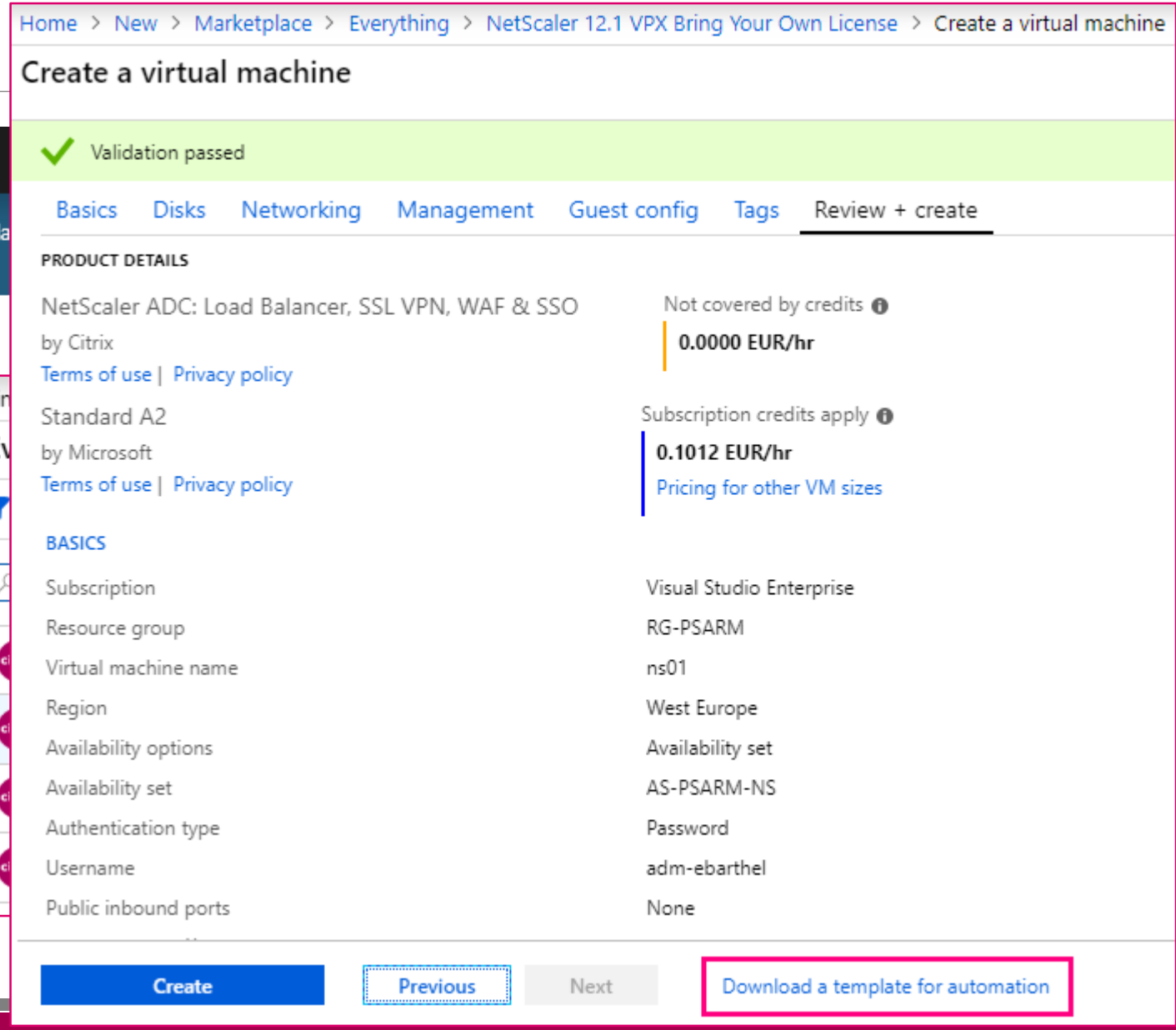
Quickstart: Create and deploy Azure Resource Manager templates by using the Azure portal

07-09-2018 • 10 minuten om te lezen • Medewerkers

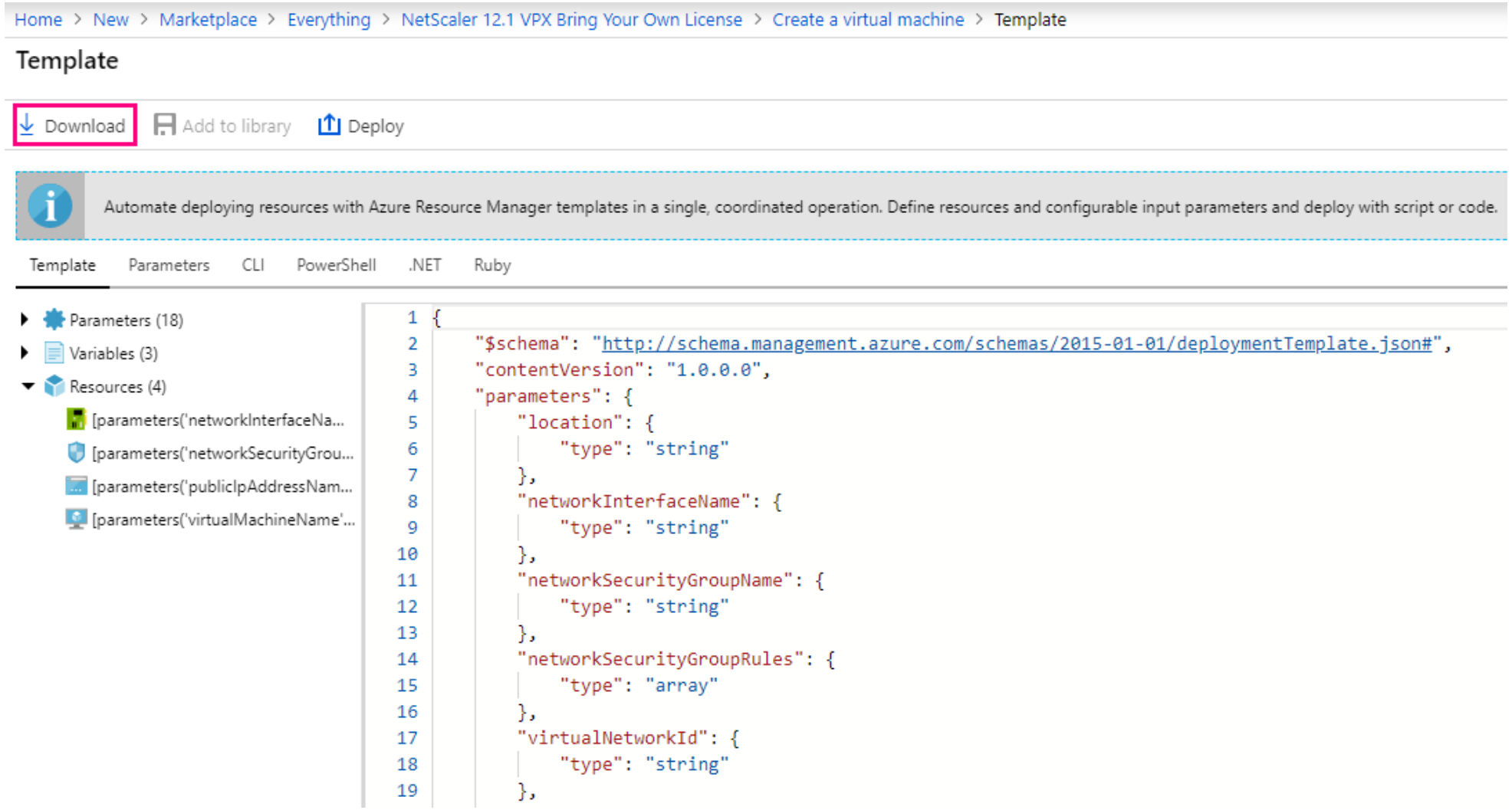
Learn how to create your first Azure Resource Manager template by generating one using the Azure portal, and how to edit and deploy the template from the portal.

ARM Template





ARM Template



COGNITION

ARM Template

```
{
    "$schema": "http://schema.management.
    "contentVersion": "1.0.0.0",
    "parameters": {
        "location": {
            "type": "string",
            "defaultValue": "West Europe"
        },
```

```
    "variables": {
        "vnetId": "[resourceId('RG-PSARM','
        "subnetRef1": "[concat(variables('v
        "subnetRef2": "[concat(variables('v
        "nic1Name": "[concat(parameters('vi
        "nic2Name": "[concat(parameters('vi
        "nsg1Id": "[resourceId(resourceGrou
        "nsg2Id": "[resourceId(resourceGrou
        "PIPName": "[concat(parameters('vir
    },
```

```
    "outputs": {
        "adminUsername": {
            "type": "string",
            "value": "[parameters('adminUsername')]"
        },
        "adminPassword": {
            "type": "string",
            "value": "[parameters('adminPassword')]"
        },
        "virtualMachineName": {
            "type": "string",
            "value": "[parameters('virtualMachineName')]"
        }
    }
```

```
    "resources": [
        {
            "name": "[variables('nic1Name')]",
            "type": "Microsoft.Network/NetworkInterfaces",
            "apiVersion": "2016-09-01",
            "location": "[resourceGroup().location]",
            "dependsOn": [
                "[concat('Microsoft.Network/virtualNetworks/', parameters('virtualNetworkName'))]"
            ],
            "properties": {
                "ipConfigurations": [
                    {
                        "name": "NSIP",
                        "properties": {
                            "subnet": {
                                "id": "[variables('subnetRef1')]"
                            },
                            "privateIPAllocationMethod": "Dynamic",
                            "primary" : true
                        }
                    },
                    {
                        "name": "SNIP-Backend",
                        "properties": {
                            "subnet": {
                                "id": "[variables('subnetRef1')]"
                            },
                            "privateIPAllocationMethod": "Dynamic",
                            "primary" : false
                        }
                    }
                ],
                "networkSecurityGroup": {
                    "id": "[variables('nsg1Id')]"
                }
            },
```

COGNITION

ARM Template

Microsoft Azure

Overview | Solutions | Products | Documentation | Pricing | Training | Marketplace | Partners | Support | Blog | More

Azure / Azure Templates / Network / Network Interfaces

Filter by title

- Overview
- Reference
- MySQL
- Network
  - Application Gateways
  - Application Security Groups
  - Azure Firewalls
  - Connections
  - DNS Zones
  - Express Route Gateways
  - Load Balancers
  - Local Network Gateways
  - Network Interfaces
    - TAP Configurations
  - Network Profiles
  - Network Security Groups

Microsoft.Network/networkInterfaces template reference

09/26/2018 • 22 minutes to read • Contributors

API Version: 2018-08-01

Template format

To create a Microsoft.Network/networkInterfaces resource, add the following JSON to the resources section of your template.

JSON Copy

```
{
  "name": "string",
  "type": "Microsoft.Network/networkInterfaces",
  "apiVersion": "2018-08-01",
  "location": "string",
  "tags": {},
  "properties": {
    "virtualMachine": {
      "id": "string"
    },
    "networkSecurityGroup": {
      "id": "string",
      "location": "string",
      "tags": {},
      "properties": {
        "securityRules": [
```

In this article

- Template format
- Property values
- Quickstart templates

COGNITION

ARM Template

Microsoft Azure

Overview | Solutions | Products | Documentation | Pricing | Training | Marketplace | Partners | Support | Blog | More

Azure / Azure Templates / Network / Network Interfaces

Filter by title

- Overview
- Reference
- MySQL
- Network
  - Application Gateways
  - Application Security Groups
  - Azure Firewalls
  - Connections
  - DNS Zones
  - Express Route Gateways
  - Load Balancers
  - Local Network Gateways
  - Network Interfaces
    - TAP Configurations
  - Network Profiles
  - Network Security Groups

Property values

The following tables describe the values you need to set in the schema.

Microsoft.Network/networkInterfaces object

Name	Type	Required	Value
name	string	Yes	
type	enum	Yes	Microsoft.Network/networkInterfaces
apiVersion	enum	Yes	2018-08-01
location	string	No	Resource location.
tags	object	No	Resource tags.
properties	object	Yes	Properties of the network interface. - NetworkInterfacePropertiesFormat object
resources	array	No	tapConfigurations

In this article

- Template format
- Property values
- Quickstart templates

COGNITION

ARM Template

Microsoft Azure

Overview ⌄ Solutions Products ⌄ Documentation Pricing Training Marketplace ⌄ Partners ⌄ Support ⌄ Blog More ⌄

Azure / Azure Templates / Network / Network Interfaces

Filter by title

- Overview
- Reference
- MySQL
- Network
  - Application Gateways
  - Application Security Groups
  - Azure Firewalls
  - Connections
  - DNS Zones
  - Express Route Gateways
  - Load Balancers
  - Local Network Gateways
  - Network Interfaces
    - TAP Configurations
  - Network Profiles
  - Network Security Groups

NetworkInterfaceIPConfigurationPropertiesFormat object

Name	Type	Required	Value
privateIPAddress	string	No	Private IP address of the IP configuration.
privateIPAllocationMethod	enum	No	Defines how a private IP address is assigned. Possible values are: 'Static' and 'Dynamic'. - Static or Dynamic
privateIPAddressVersion	enum	No	Available from Api-Version 2016-03-30 onwards, it represents whether the specific ipconfiguration is IPv4 or IPv6. Default is taken as IPv4. Possible values are: 'IPv4' and 'IPv6'. - IPv4 or IPv6
subnet	object	No	Subnet bound to the IP configuration. - Subnet object
primary	boolean	No	Gets whether this is a primary customer address on the network interface.
publicIPAddress	object	No	Public IP address bound to the IP configuration. - PublicIPAddress object
applicationSecurityGroups	array	No	Application security groups in which the IP configuration is included. - ApplicationSecurityGroup object

In this article

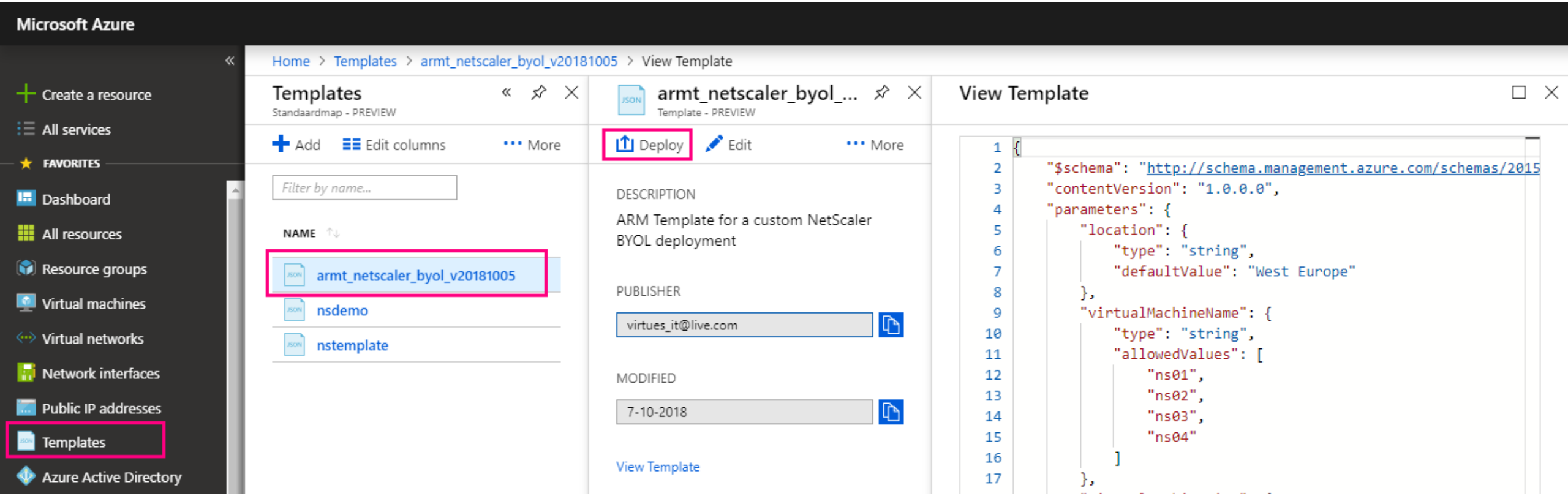
- Template format
- Property values
- Quickstart templates

COGNITION

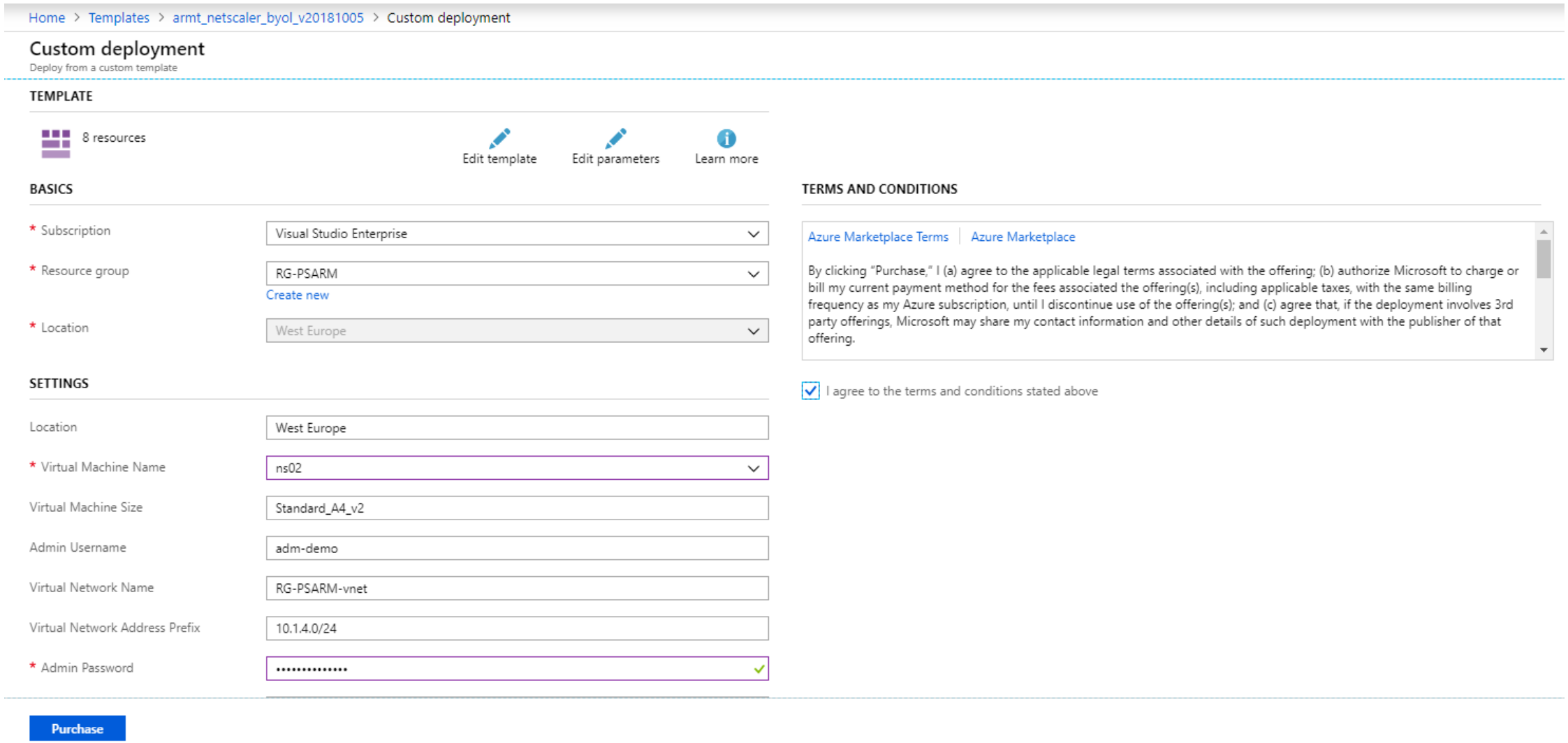
ARM Template

```
{
    "name": "[variables('nic2Name')]",
    "type": "Microsoft.Network/networkInterfaces",
    "apiVersion": "2016-09-01",
    "location": "[resourceGroup().location]",
    "dependsOn": [
        "[concat('Microsoft.Network/publicIpAddresses/', variables('PIPName'))]",
        "[concat('Microsoft.Network/virtualNetworks/', parameters('virtualNetworkName'))]"
    ],
    "properties": {
        "ipConfigurations": [
            {
                "name": "VIP-NSG-Public",
                "properties": {
                    "subnet": {
                        "id": "[variables('subnetRef2')]"
                    },
                    "privateIPAllocationMethod": "Dynamic",
                    "primary" : false,
                    "publicIpAddress": {
                        "id": "[resourceId('Microsoft.Network/publicIpAddresses', variables('PIPName'))]"
                    }
                }
            },
```

ARM Template



ARM Template



COGNITION

ARM Template

Your deployment is underway

Check the status of your deployment, manage resources, or troubleshoot deployment issues. Pin this page to your dashboard to easily find it next time.

Deployment name: virtues_it_live.com.armt_netscaler_byol_v20181005
Subscription: Visual Studio Enterprise
Resource group: RG-PSARM

DEPLOYMENT DETAILS (Download)

Start time: 7-10-2018 13:59:31
Duration: 55 seconds
Correlation ID: 8d2f1098-fa8c-48c3-abe2-3989bdf2cf5c

RESOURCE	TYPE	STATUS	OPERATION DETAILS
ns02-nic1	Microsoft.Network/NetworkInterfaces	Created	Operation details
ns02-nic2	Microsoft.Network/networkInterfaces	Created	Operation details
ns02-PIP	Microsoft.Network/publicIpAddresses	OK	Operation details
nsg-ns-external	Microsoft.Network/networkSecurityGro...	OK	Operation details
nsg-ns-internal	Microsoft.Network/networkSecurityGro...	OK	Operation details
RG-PSARM-vnet	Microsoft.Network/virtualNetworks	OK	Operation details
AS-PSARM-NS	Microsoft.Compute/AvailabilitySets	OK	Operation details

COGNITION

ARM Template

Your deployment is complete

Check the status of your deployment, manage resources, or troubleshoot deployment issues. Pin this page to your dashboard to easily find it next time.

Deployment name: virtues_it_live.com.armt_netscaler_byol_v20181005
Subscription: Visual Studio Enterprise
Resource group: RG-PSARM

DEPLOYMENT DETAILS (Download)

Start time: 7-10-2018 13:59:31
Duration: 4 minutes 46 seconds
Correlation ID: 8d2f1098-fa8c-48c3-abe2-3989bdf2cf5c

RESOURCE	TYPE	STATUS	OPERATION DETAILS
ns02	Microsoft.Compute/virtualMachines	OK	Operation details
ns02-nic1	Microsoft.Network/NetworkInterfaces	Created	Operation details
ns02-nic2	Microsoft.Network/networkInterfaces	Created	Operation details
ns02-PIP	Microsoft.Network/publicIpAddresses	OK	Operation details
nsg-ns-external	Microsoft.Network/networkSecurityGro...	OK	Operation details
nsg-ns-internal	Microsoft.Network/networkSecurityGro...	OK	Operation details
RG-PSARM-vnet	Microsoft.Network/virtualNetworks	OK	Operation details
AS-PSARM-NS	Microsoft.Compute/AvailabilitySets	OK	Operation details

Az

AZURE POWERSHELL MODULE

Az

COGNITION

Install the Azure PowerShell module

12/13/2018 • 4 minutes to read • Contributors

This article tells you how to install the Azure PowerShell modules using PowerShellGet. These instructions work on Windows, macOS, and Linux platforms. For the Az module, currently no other installation methods are supported.

Requirements

Azure PowerShell works with PowerShell 5.1 or higher on Windows, or PowerShell Core 6.x and later on all platforms. If you aren't sure if you have PowerShell, or are on macOS or Linux, install the latest version of PowerShell Core.

Install the Azure PowerShell module

```
PS C:\Users\hwadmin> Install-Module -Name Az

PS C:\Users\hwadmin> Get-InstalledModule -Name Az

Version    Name                                Repository           Description
-------    ----                                ----------           -----------
2.1.0      Az                                  PSGallery            Microsoft Azure PowerShell - Cmdlets to manage resources in Azure. This module is compatible with WindowsPowerShell and PowerShell Core....


PS C:\Users\hwadmin>
```

PowerShell, run the following command in an elevated session ("Run as Administrator" on Windows, or with superuser privileges on macOS or Linux):

PowerShell | Copy | Try It

```
Install-Module -Name Az -AllowClobber
```

If you don't have access to administrator privileges, you can install for the current user by adding the `-Scope` argument.

PowerShell | Copy | Try It

```
Install-Module -Name Az -AllowClobber -Scope CurrentUser
```

sources: https://docs.microsoft.com/en-us/powershell/azure/install-az-ps?view=azps-2.2.0

Az

COGNITION

Install the Azure PowerShell module

12/13/2018 • 4 minutes to read • Contributors

This article tells you how to install the Azure PowerShell modules using PowerShellGet. These instructions work on Windows, macOS, and Linux platforms. For the Az module, currently no other installation methods are supported.

Requirements

Azure PowerShell works with PowerShell 5.1 or higher on Windows, or PowerShell Core 6.x and later on all platforms. If you aren't sure if you have PowerShell, or are on macOS or Linux, install the latest version of PowerShell Core.

Install the Azure PowerShell module

```
PS C:\Users\hwadmin> Install-Module -Name Az

PS C:\Users\hwadmin> Get-InstalledModule -Name Az

Version    Name
-------    ----
2.1.0      Az


PS C:\Users\hwadmin>
```

prefix the Install-Module cmdlet with **PowerShellGet** to prevent errors

compatible with WindowsPowerShell and PowerShell Core....

PowerShell, run the following command in an elevated session ("Run as Administrator" on Windows, or with superuser privileges on macOS or Linux):

PowerShell | Copy | Try It

```
Install-Module -Name Az -AllowClobber
```

If you don't have access to administrator privileges, you can install for the current user by adding the `-Scope` argument.

PowerShell | Copy | Try It

```
Install-Module -Name Az -AllowClobber -Scope CurrentUser
```

sources: https://docs.microsoft.com/en-us/powershell/azure/install-az-ps?view=azps-2.2.0

COGNITION

Az

```
Current Installed Modules:

Version    Name                                Repository           Description
-------    ----                                ----------           -----------
2.1.0      Az                                  PSGallery            Microsoft Azure PowerShell - Cmdlets to manage resources in Azure. This module is compatible with WindowsPowerShell and PowerShell Core....
1.5.2      Az.Accounts                         PSGallery            Microsoft Azure PowerShell - Accounts credential management cmdlets for Azure Resource Manager in Windows PowerShell and PowerShell Core....
1.0.1      Az.Aks                              PSGallery            Microsoft Azure PowerShell - Azure managed Kubernetes cmdlets for Windows PowerShell and PowerShell Core....
1.1.0      Az.AnalysisServices                 PSGallery            Microsoft Azure PowerShell - Analysis Services cmdlets for Windows PowerShell and PowerShell Core....
1.1.0      Az.ApiManagement                    PSGallery            Microsoft Azure PowerShell - Api Management service cmdlets for Azure Resource Manager in Windows PowerShell and PowerShell Core....
1.0.0      Az.ApplicationInsights              PSGallery            Microsoft Azure PowerShell - Application Insights management cmdlets for Azure Resource Manager in Windows PowerShell and PowerShell Core.  Creates and ma...
1.2.2      Az.Automation                       PSGallery            Microsoft Azure PowerShell - Automation service cmdlets for Azure Resource Manager in Windows PowerShell and PowerShell Core....
1.1.0      Az.Batch                            PSGallery            Microsoft Azure PowerShell - Batch service cmdlets for Azure Resource Manager in Windows PowerShell and PowerShell Core....
1.0.0      Az.Billing                          PSGallery            Microsoft Azure PowerShell - Billing service cmdlets for Azure Resource Manager in Windows PowerShell and PowerShell Core....
1.2.0      Az.Cdn                              PSGallery            Microsoft Azure PowerShell - CDN service cmdlets for Azure Resource Manager in Windows PowerShell and PowerShell Core....
1.1.1      Az.CognitiveServices                PSGallery            Microsoft Azure PowerShell - Cognitive Services management cmdlets for Azure Resource Manager in Windows PowerShell and PowerShell Core. Creates and manag...
2.1.0      Az.Compute                          PSGallery            Microsoft Azure PowerShell - Compute service cmdlets for Azure Resource Manager in Windows PowerShell and PowerShell Core.  Manages virtual machines, host...
1.0.1      Az.ContainerInstance                PSGallery            Microsoft Azure PowerShell - Container Instance cmdlets for Azure Resource Manager in Windows PowerShell and PowerShell Core....
1.0.1      Az.ContainerRegistry                PSGallery            Microsoft Azure PowerShell - Container Registry service cmdlets for Azure Resource Manager in Windows PowerShell and PowerShell Core....
1.1.1      Az.DataFactory                      PSGallery            Microsoft Azure PowerShell - Data Factory service cmdlets for Azure Resource Manager in Windows PowerShell and PowerShell Core....
1.0.0      Az.DataLakeAnalytics                PSGallery            Microsoft Azure PowerShell - Data Lake Analytics in Windows PowerShell and PowerShell Core....
1.2.1      Az.DataLakeStore                    PSGallery            Microsoft Azure PowerShell - Azure Data Lake Store cmdlets in Windows PowerShell and PowerShell Core....
1.0.0      Az.DeploymentManager                PSGallery            PowerShell .Net Core Microsoft Azure PowerShell - Deployment Manager cmdlets for Azure Resource Manager....
1.0.0      Az.DevTestLabs                      PSGallery            Microsoft Azure PowerShell - DevTest Labs service cmdlets for Azure Resource Manager in Windows PowerShell and PowerShell Core....
1.1.0      Az.Dns                              PSGallery            Microsoft Azure PowerShell - DNS service cmdlets for Azure Resource Manager in Windows PowerShell and PowerShell Core....
1.1.1      Az.EventGrid                        PSGallery            Microsoft Azure PowerShell - Event Grid service cmdlets for Azure Resource Manager in Windows PowerShell and PowerShell Core....
1.1.0      Az.EventHub                         PSGallery            Microsoft Azure PowerShell - Event Hubs service cmdlets for Azure Resource Manager in Windows PowerShell and PowerShell Core....
1.0.0      Az.FrontDoor                        PSGallery            Microsoft Azure PowerShell - Front Door service cmdlets for Azure Resource Manager in Windows PowerShell and PowerShell Core....
2.0.0      Az.HDInsight                        PSGallery            Microsoft Azure PowerShell - HDInsight service cmdlets for Azure Resource Manager in Windows PowerShell and PowerShell Core....
1.1.0      Az.IotHub                           PSGallery            Microsoft Azure PowerShell - IoT Hub service cmdlets for Azure Resource Manager in Windows PowerShell and PowerShell Core....
1.2.0      Az.KeyVault                         PSGallery            Microsoft Azure PowerShell - Key Vault service cmdlets for Azure Resource Manager in Windows PowerShell and PowerShell Core....
1.2.1      Az.LogicApp                         PSGallery            Microsoft Azure PowerShell - Logic Apps cmdlets for Azure Resource Manager in Windows PowerShell and PowerShell Core....
1.1.0      Az.MachineLearning                  PSGallery            Microsoft Azure PowerShell - Machine Learning Web Services cmdlets for Azure Resource Manager in Windows PowerShell and PowerShell Core....
1.0.0      Az.MarketplaceOrdering              PSGallery            Microsoft Azure PowerShell - Marketplace Ordering agreements service cmdlets for Azure Resource Manager in Windows PowerShell and PowerShell Core....
1.1.0      Az.Media                            PSGallery            Microsoft Azure PowerShell - Media service cmdlets for Azure Resource Manager in Windows PowerShell and PowerShell Core....
1.2.1      Az.Monitor                          PSGallery            Microsoft Azure PowerShell - Monitor service cmdlets for Azure Resource Manager in Windows PowerShell and PowerShell Core....
1.8.1      Az.Network                          PSGallery            Microsoft Azure PowerShell - Networking service cmdlets for Azure Resource Manager in Windows PowerShell and PowerShell Core....
1.1.0      Az.NotificationHubs                 PSGallery            Microsoft Azure PowerShell - Notification Hubs cmdlets for Azure Resource Manager in Windows PowerShell and PowerShell Core....
1.2.0      Az.OperationalInsights              PSGallery            Microsoft Azure PowerShell - Operational Insights service cmdlets for Azure Resource Manager in Windows PowerShell and PowerShell Core.
1.1.0      Az.PolicyInsights                   PSGallery            Microsoft Azure PowerShell - Azure Policy Insights cmdlets for Windows PowerShell and PowerShell Core. Allows querying policy evaluation events and compli...
1.1.0      Az.PowerBIEmbedded                  PSGallery            Microsoft Azure PowerShell - Power BI Embedded service management cmdlets for Azure Resource Manager in Windows PowerShell and PowerShell Core.  Creates a...
1.4.0      Az.RecoveryServices                 PSGallery            Microsoft Azure PowerShell - Recovery Services cmdlets for Azure Resource Manager in Windows PowerShell and PowerShell Core....
1.1.0      Az.RedisCache                       PSGallery            Microsoft Azure PowerShell - Redis Cache service cmdlets for Azure Resource Manager in Windows PowerShell and PowerShell Core....
1.0.1      Az.Relay                            PSGallery            Microsoft Azure PowerShell - Relay service cmdlets for Azure Resource Manager in Windows PowerShell and PowerShell Core....
1.4.0      Az.Resources                        PSGallery            Microsoft Azure PowerShell - Azure Resource Manager and Active Directory cmdlets in Windows PowerShell and PowerShell Core.  Manages subscriptions, tenant...
1.1.0      Az.ServiceBus                       PSGallery            Microsoft Azure PowerShell - Service Bus service cmdlets for Azure Resource Manager in Windows PowerShell and PowerShell Core....
1.0.1      Az.ServiceFabric                    PSGallery            Microsoft Azure PowerShell - Service Fabric cmdlets for Azure Resource Manager in Windows PowerShell and PowerShell Core....
1.0.2      Az.SignalR                          PSGallery            Microsoft Azure PowerShell - Azure SignalR service commands for Windows PowerShell and PowerShell Core....
1.10.0     Az.Sql                              PSGallery            Microsoft Azure PowerShell - SQL service cmdlets for Azure Resource Manager in Windows PowerShell and PowerShell Core....
1.3.0      Az.Storage                          PSGallery            Microsoft Azure PowerShell - Storage service data plane and management cmdlets for Azure Resource Manager in Windows PowerShell and PowerShell Core.  Crea...
1.0.0      Az.StreamAnalytics                  PSGallery            Microsoft Azure PowerShell - Stream Analytics service cmdlets for Azure Resource Manager in Windows PowerShell and PowerShell Core....
1.0.1      Az.TrafficManager                   PSGallery            Microsoft Azure PowerShell - Traffic Manager service cmdlets for Azure Resource Manager in Windows PowerShell and PowerShell Core....
1.2.1      Az.Websites                         PSGallery            Microsoft Azure PowerShell - App Service (Web Apps) service cmdlets for Azure Resource Manager in Windows PowerShell and PowerShell Core....
```

COGNITION

Az

Az PowerShell 2.2.0

Search API

- Azure PowerShell
- Introducing the new Az module
- Install
- Tutorials
- Sample Scripts
- Release notes
- Reference
  - Accounts
  - Resources
    - Resources
    - Export-AzResourceGroup
    - Get-AzDenyAssignment
    - Get-AzDeployment
    - Get-AzDeploymentOperation
    - Get-AzLocation
    - Get-AzManagedApplication

Resources

Export-AzResourceGroup	Captures a resource group as a template and saves it to a file.
Get-AzDenyAssignment	Lists Azure RBAC deny assignments at the specified scope.
Get-AzDeployment	Get deployment
Get-AzDeploymentOperation	Get deployment operation
Get-AzLocation	Gets all locations and the supported resource providers for each location.
Get-AzManagedApplication	Gets managed applications
Get-AzManagedApplicationDefinition	Gets managed application definitions
Get-AzManagementGroup	Gets Management Group(s)
Get-AzProviderFeature	Gets information about Azure provider features.
Get-AzProviderOperation	Gets the operations for an Azure resource provider that are securable using Azure RBAC.

sources: https://docs.microsoft.com/en-us/powershell/module/az.resources/?view=azps-2.2.0#resources

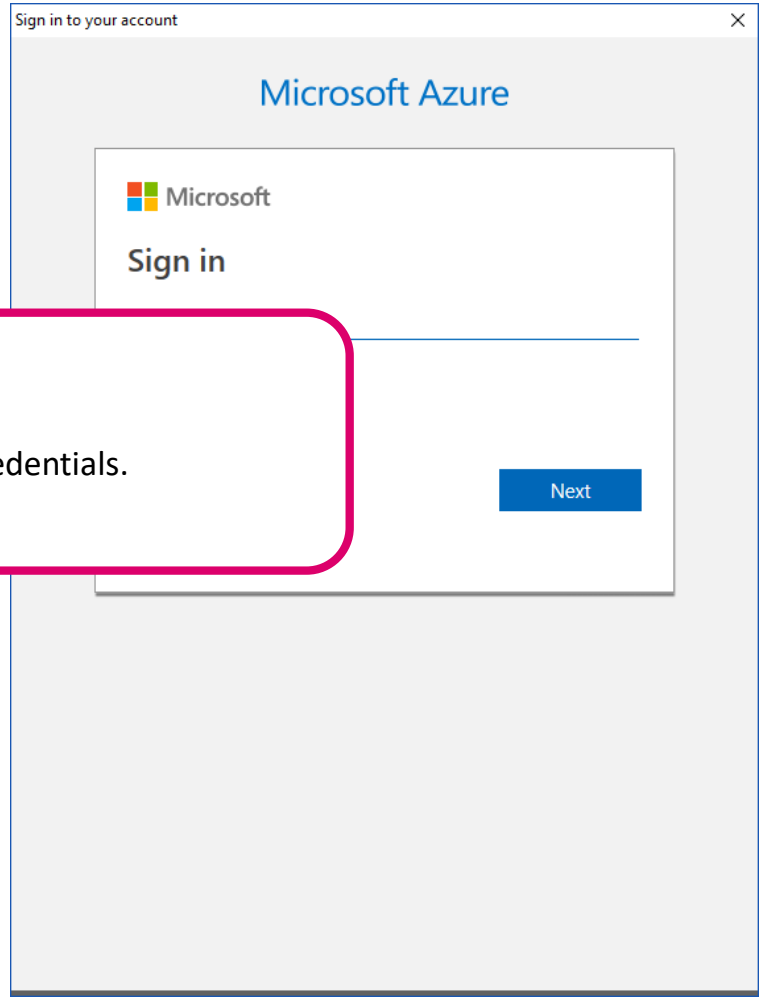
COGNITION

Az – PowerShell

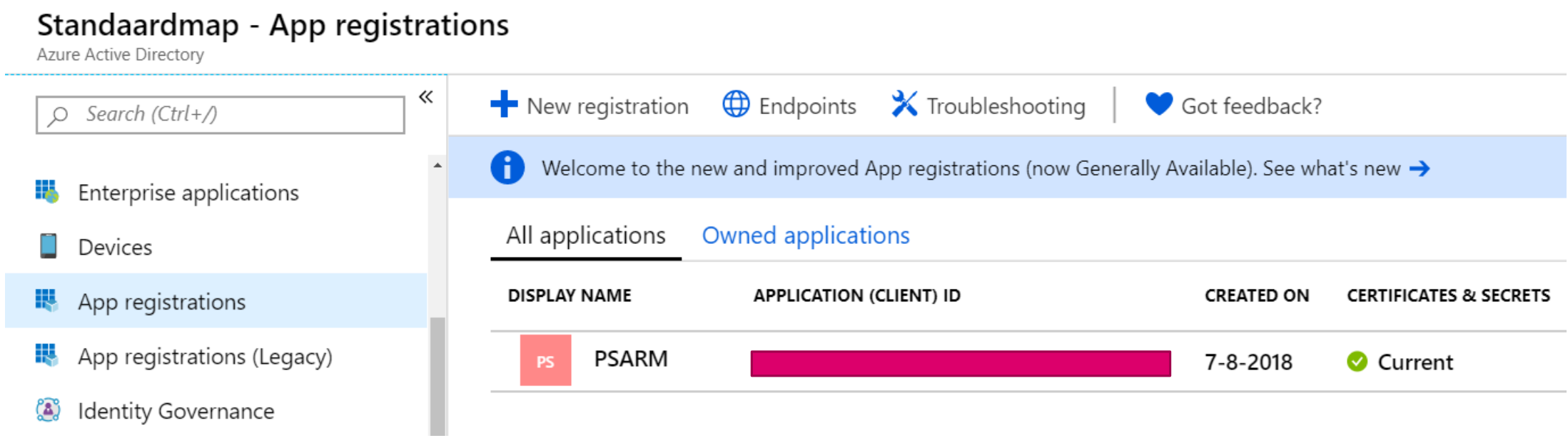
```
# ---------------------
# | Interactive Login |
# ---------------------

# Login using interactive login to Azure (popup)
Login-AzAccount
```

Plan to manage your app or service with **Azure PowerShell?**

Run your script with an **AAD Service Principal**, rather than your own credentials.

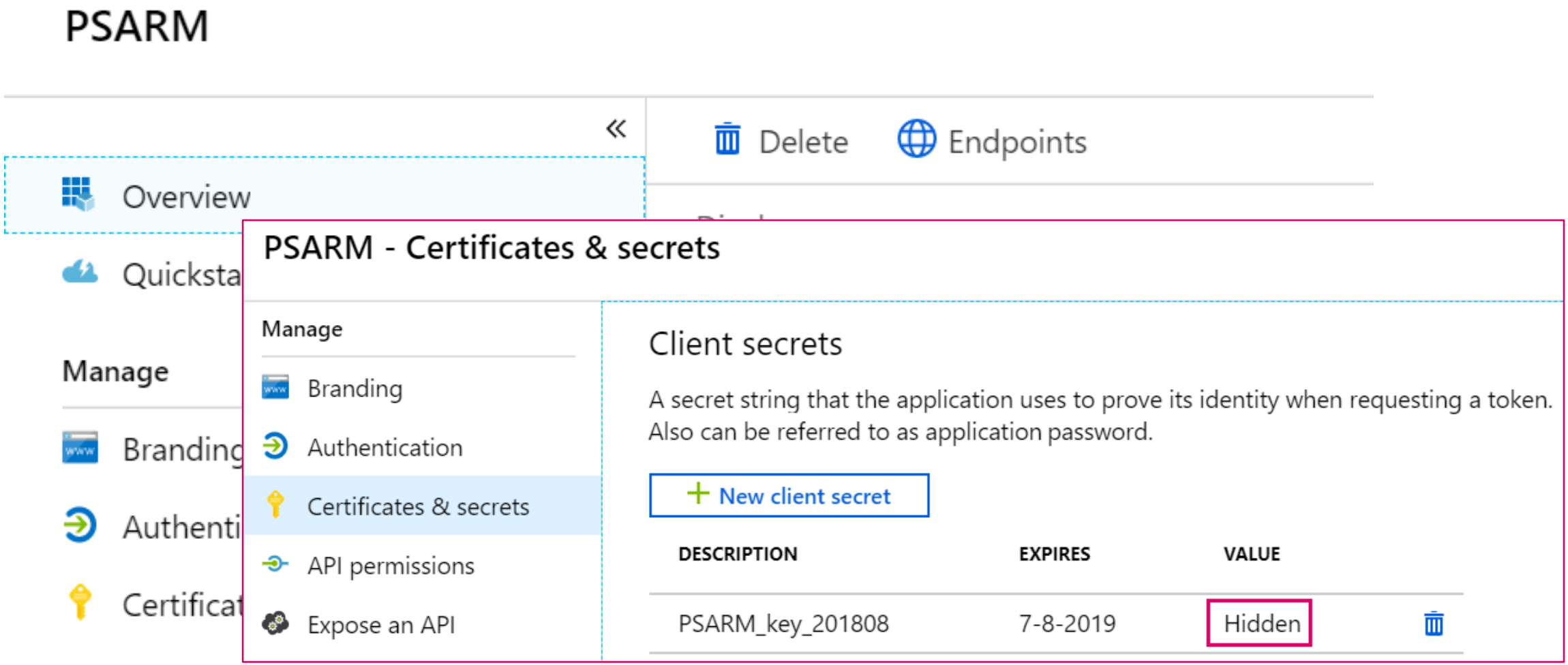


Az – Service Principal



step 1: https://docs.microsoft.com/en-us/azure/azure-resource-manager/resource-group-create-service-principal-portal

Az – AppID, AppKey & TenantID



step 2: https://docs.microsoft.com/en-us/azure/azure-resource-manager/resource-group-create-service-principal-portal

COGNITION

Az – Connect-AzAccount

```
Write-Output ""
Write-Output "## ------------------------------------------------------------"
Write-Output "## | Section 1 - Logging onto Azure using the Az PowerShell Module |"
Write-Output "## ------------------------------------------------------------"
Write-Output ""

#region Logon information
    # Read App secrets from CSV file
        $AppSecrets = Import-Csv -Path "C:\Sources\Azure_App_Secrets.csv" -Delimiter ","
        $ApplicationID = $AppSecrets.AppID
        $ApplicationKey = $AppSecrets.AppKey
        $TenantID = $AppSecrets.TenantID
    # Create Azure Credentials
        $SPpasswd = ConvertTo-SecureString $ApplicationKey -AsPlainText -Force
        $SPCreds = New-Object System.Management.Automation.PSCredential($ApplicationID, $SPpasswd)
#endregion

#region Step 0: Sign in to Azure with Service Principal
    Write-Output "Step 0: Create a session to Azure ... "
    $Session = Connect-AzAccount -Credential $SPCreds -TenantId $TenantID -ServicePrincipal `
                                 -SkipContextPopulation
#endregion
```

COGNITION

Az – Connect-AzAccount

```
Write-Output ""
Write-Output "## ------------------------------------------------------------"
Write-Output "## | Section 1 - Logging onto Azure using the Az PowerShell Module |"
Write-Output "## ------------------------------------------------------------"
Write-Output ""

#regior
    # R
        $TenantID = $AppSecrets.TenantID
    # Create Azure Credentials
        $SPpasswd = ConvertTo-SecureString $ApplicationKey -AsPlainText -Force
        $SPCreds = New-Object System.Management.Automation.PSCredential($ApplicationID, $SPpasswd)
#endregion

#region Step 0: Sign in to Azure with Service Principal
    Write-Output "Step 0: Create a session to Azure ... "
    $Session = Connect-AzAccount -Credential $SPCreds -TenantId $TenantID -ServicePrincipal `
                                 -SkipContextPopulation
#endregion
```

```
Step 0: Create a session to Azure ...
 => Session Created Successful!
```

COGNITION

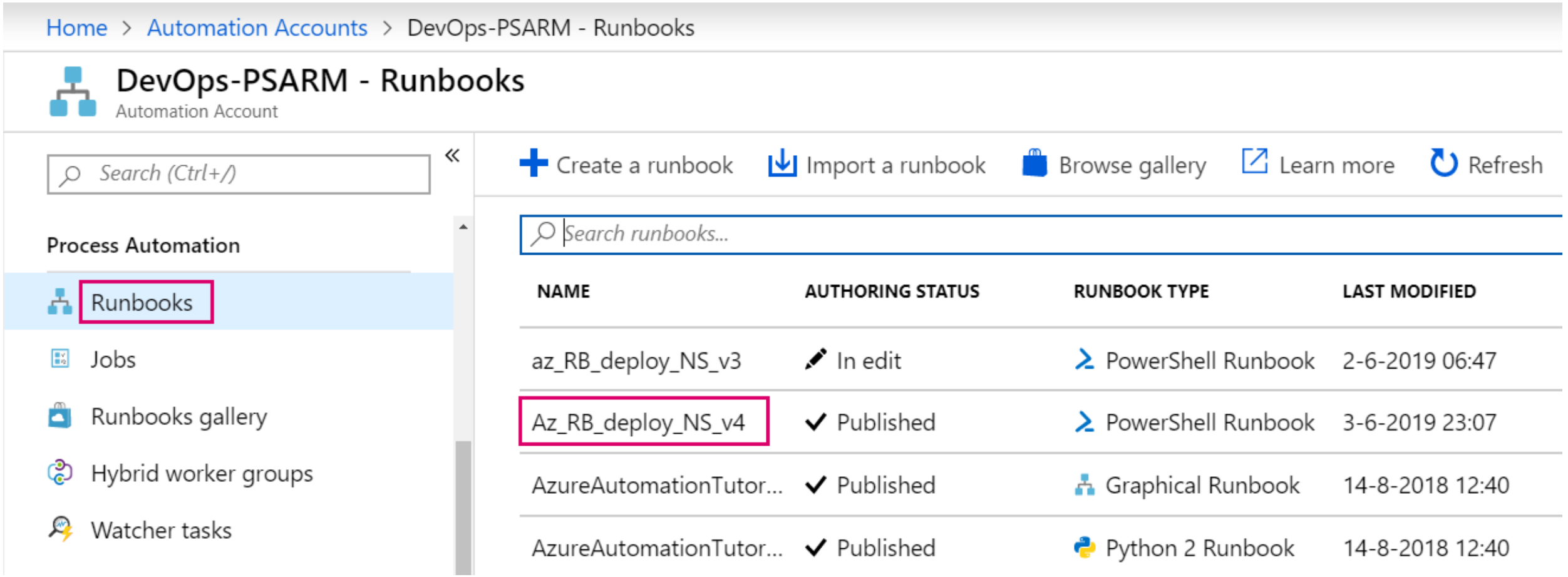
Az – New-AzResourceGroupDeployment

```
Write-Output ""
Write-Output "## ------------------------------------------------------------------------------"
Write-Output "## | Section 2 - Deploy NetScaler VPX BYOL, based on custom ARM Template (w/ 2 NICs) |"
Write-Output "## ------------------------------------------------------------------------------"
Write-Output ""

## Create Hashtable object
Write-Output "Step 2: Create a Hashtable Object that contains all the ARM Template variables and values."
$objTemplateParameter = @{}

## Add the parameter values to it
$objTemplateParameter.Add('location', 'westeurope')
$objTemplateParameter.Add('virtualMachineName', $NetScalerName)
$objTemplateParameter.Add('virtualMachineSize', 'Standard_A4_v2')
$objTemplateParameter.Add('adminUsername', $NSUsername)
$objTemplateParameter.Add('adminPassword', $SecurePassword)
$objTemplateParameter.Add('virtualNetworkName', 'RG-PSARM-vnet')
$objTemplateParameter.Add('virtualNetworkAddressPrefix', '10.1.4.0/24')
$objTemplateParameter.Add('availabilitySetName', 'AS-PSARM-NS')
$objTemplateParameter.Add('nic1SubnetName', 'sn-internal')
$objTemplateParameter.Add('nic1SubnetAddressPrefix', '10.1.4.0/26')
$objTemplateParameter.Add('nic2SubnetName', 'sn-external')
$objTemplateParameter.Add('nic2SubnetAddressPrefix', '10.1.4.64/26')
$objTemplateParameter.Add('networkSecurityGroup1Name', 'nsg-ns-internal')
$objTemplateParameter.Add('networkSecurityGroup2Name', 'nsg-ns-external')

## Create NetScaler, using ARM Template and TemplateParameterObject for (input) parameters
New-AzResourceGroupDeployment -ResourceGroupName $ResourceGroupName -TemplateFile $strTemplateFile `
                              -TemplateParameterObject $objTemplateParameter
```

Azure Automation

RUNBOOKS & HYBRID WORKERS

Azure Automation



COGNITION

Azure Automation

Home > Automation Accounts > DevOps-PSARM - Runbooks > Az_RB_deploy_NS_v4

Az_RB_deploy_NS_v4
Runbook

Search (Ctrl+/)

Start | View | Edit | Link to schedule | Delet

- Overview
- Activity log

Resources

- Jobs
- Schedules

Runbook settings

- Properties
- Description
- Logging and tracing

Resource group
RG-PSARM

Subscription ID

Account
DevOps-PSARM

Status
Published

Location
West Europe

Runbook type
PowerShell Runbook

Recent Jobs

STATUS	CREATED	LAST UPDATI
Completed	4-6-2019 00:43:30	4-6-2019 0
Stopped	4-6-2019 00:36:00	4-6-2019 0
Completed	3-6-2019 23:08:07	3-6-2019 2

Start Runbook
Az_RB_deploy_NS_v4

Parameters

NETSCALERNAME
Default will be used
Optional, String, Default: "ns01"

NSPASSWORD
Default will be used
Optional, String, Default: "NetScalerDemo!"

NSUSERNAME
Default will be used
Optional, String, Default: "adm-demo"

Run Settings

Run on
Azure | Hybrid Worker

Choose Hybrid Worker group
HWG-PSARM

OK

COGNITION

Hybrid Worker

```
# Getting Started with Az Module
Install-Module Az

#region Logon information
    $AppSecrets = Import-Csv -Path "C:\Sources\Azure_App_Secrets.csv" -Delimiter ","
    $SubscriptionID = $AppSecrets.SubscriptionID
#endregion

#region Install Hybrid Worker
    # Script variables
    $AutomationAccountName = "DevOps-PSARM"
    $AAResourceGroupName = "RG-PSARM"
    $LogAnalyticsWorkspaceName = "OMS-WS-PSARM"
    $OMSResourceGroupName = "RG-PSARM"
    $HybridGroupName = "HWG-PSARM"

    # Install Hybrid Worker locally and add to specified Hybrid Worker Group in Azure
    C:\Scripts\New-OnPremiseHybridWorker.ps1 -AutomationAccountName $AutomationAccountName `
    -AAResourceGroupName $AAResourceGroupName -OMSResourceGroupName $OMSResourceGroupName `
    -HybridGroupName $HybridGroupName -SubscriptionId $SubscriptionID `
    -WorkspaceName $LogAnalyticsWorkspaceName
#endregion
```

sources: https://docs.microsoft.com/en-us/azure/automation/automation-hybrid-runbook-worker
https://docs.microsoft.com/en-us/azure/automation/automation-windows-hrw-install

COGNITION

Azure Automation in Action

COGNITION

Azure Automation in action

Section 1 – Log on to Azure using the Az PowerShell Module

```
Connect-AzAccount -Credential $SPCreds -TenantId $TenantID -ServicePrincipal
```

Section 2 – Deploy my VM, based on a custom ARM Template

```
New-AzResourceGroupDeployment -ResourceGroupName $ResourceGroupName `
-TemplateFile $strTemplateFile -TemplateParameterObject $objTemplateParameter
```

Section 3 – Retrieve VM configuration from Azure (reserved IP addresses, etc)

```
Get-AzVirtualNetwork -Name $vNetName -ResourceGroupName $ResourceGroupName

Get-AzNetworkInterfaceIpConfig -name "VIP-NSG-Public").PrivateIpAddress
```

IP configurations

* Subnet: sn-external (10.1.4.64/26)

NAME	IP VER...	TYPE	PRIVATE IP ADDRESS	PUBLIC IP ADDRESS	
VIP-NSG-Public	IPv4	Secondary	10.1.4.68 (Dynamic)	13.94.151.44 (ns01-PIP)	...
VIP-LB-SF	IPv4	Secondary	10.1.4.69 (Dynamic)	-	...
VIP-NSG-Internal	IPv4	Primary	10.1.4.70 (Dynamic)	-	...
SNIP-Public	IPv4	Secondary	10.1.4.71 (Dynamic)	-	...

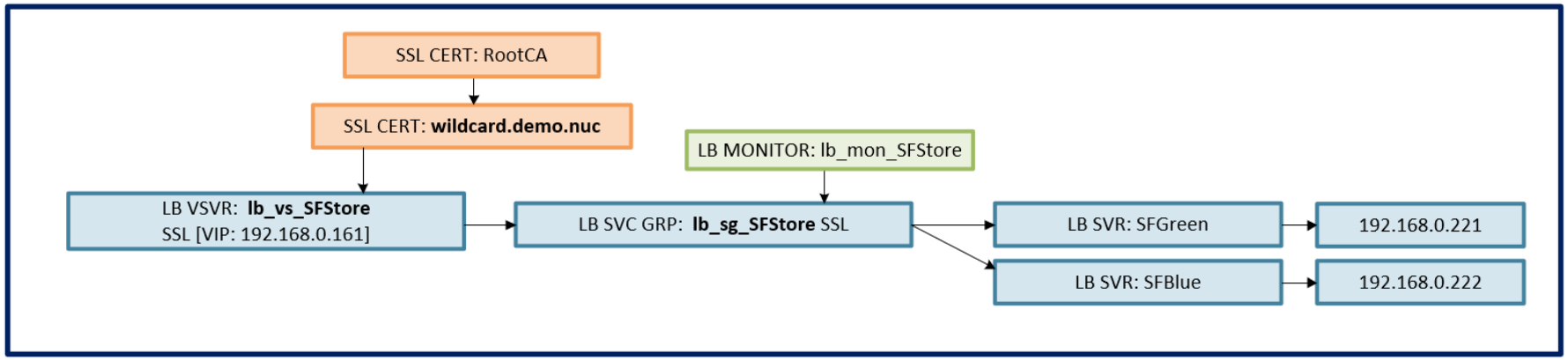
COGNITION

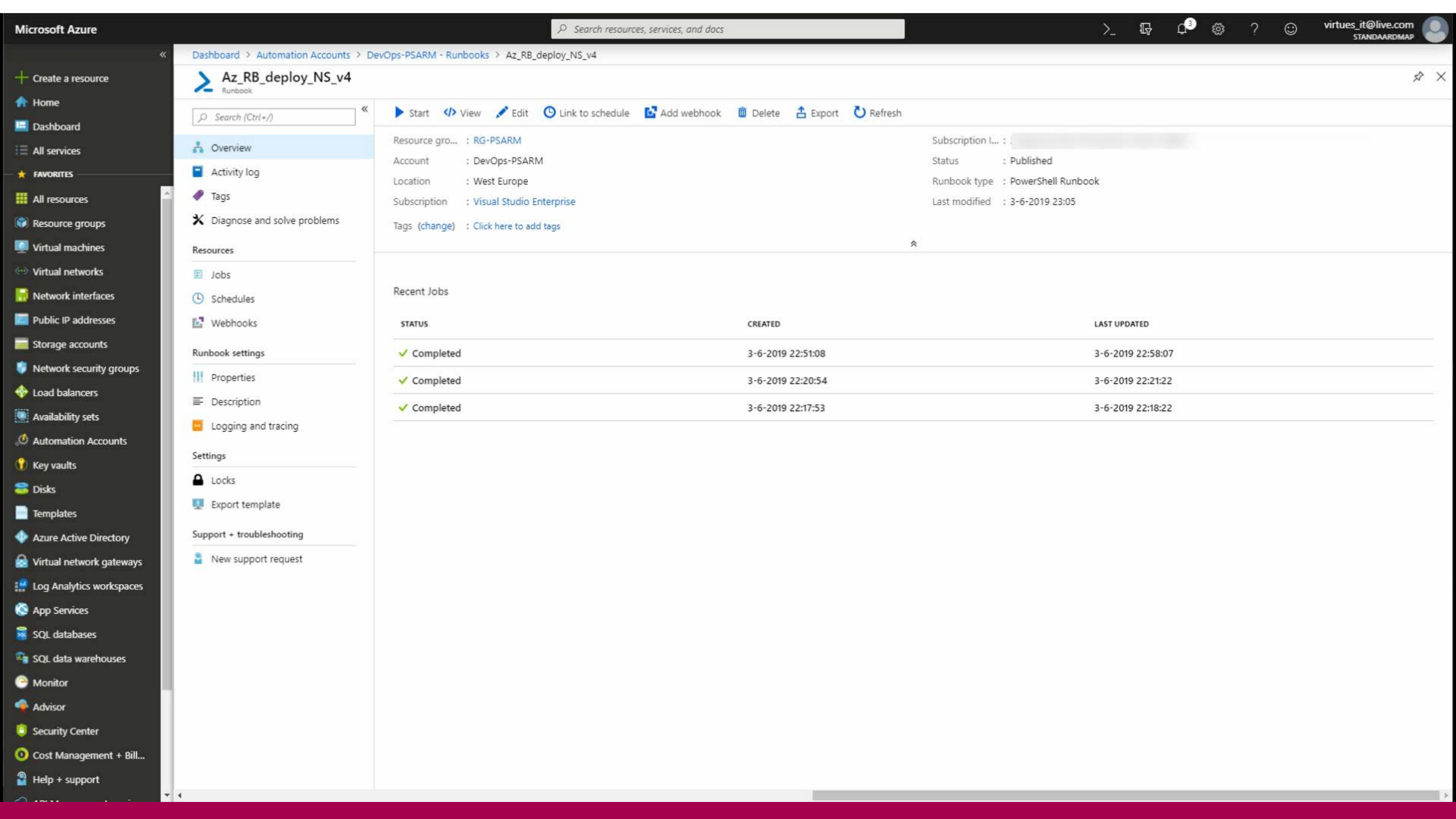
Azure Automation in action

Section 4 – Configure my VM using REST API calls (25 actions)

```
Invoke-RestMethod -Method Post -Uri $strURI -Body $payload -ContentType "application/json" `
-WebSession $NetScalerSession
```

NetScaler Load Balancing Configuration




Microsoft Azure
Search resources, services, and docs
virtues_it@live.com
STANDAARDMAP
Create a resource
Home
Dashboard
All services
FAVORITES
All resources
Resource groups
Virtual machines
Virtual networks
Network interfaces
Public IP addresses
Storage accounts
Network security groups
Load balancers
Availability sets
Automation Accounts
Key vaults
Disks
Templates
Azure Active Directory
Virtual network gateways
Log Analytics workspaces
App Services
SQL databases
SQL data warehouses
Monitor
Advisor
Security Center
Cost Management + Bill...
Help + support
Dashboard > Automation Accounts > DevOps-PSARM - Runbooks > Az_RB_deploy_NS_v4
Az_RB_deploy_NS_v4
Runbook
Search (Ctrl+/)
Overview
Activity log
Tags
Diagnose and solve problems
Resources
Jobs
Schedules
Webhooks
Runbook settings
Properties
Description
Logging and tracing
Settings
Locks
Export template
Support + troubleshooting
New support request
Start
View
Edit
Link to schedule
Add webhook
Delete
Export
Refresh
Resource gro... : RG-PSARM
Account : DevOps-PSARM
Location : West Europe
Subscription : Visual Studio Enterprise
Tags (change) : Click here to add tags
Subscription I... :
Status : Published
Runbook type : PowerShell Runbook
Last modified : 3-6-2019 23:05
Recent Jobs
STATUS
CREATED
LAST UPDATED
Completed
3-6-2019 22:51:08
3-6-2019 22:58:07
Completed
3-6-2019 22:20:54
3-6-2019 22:21:22
Completed
3-6-2019 22:17:53
3-6-2019 22:18:22

COGNITION

Resources

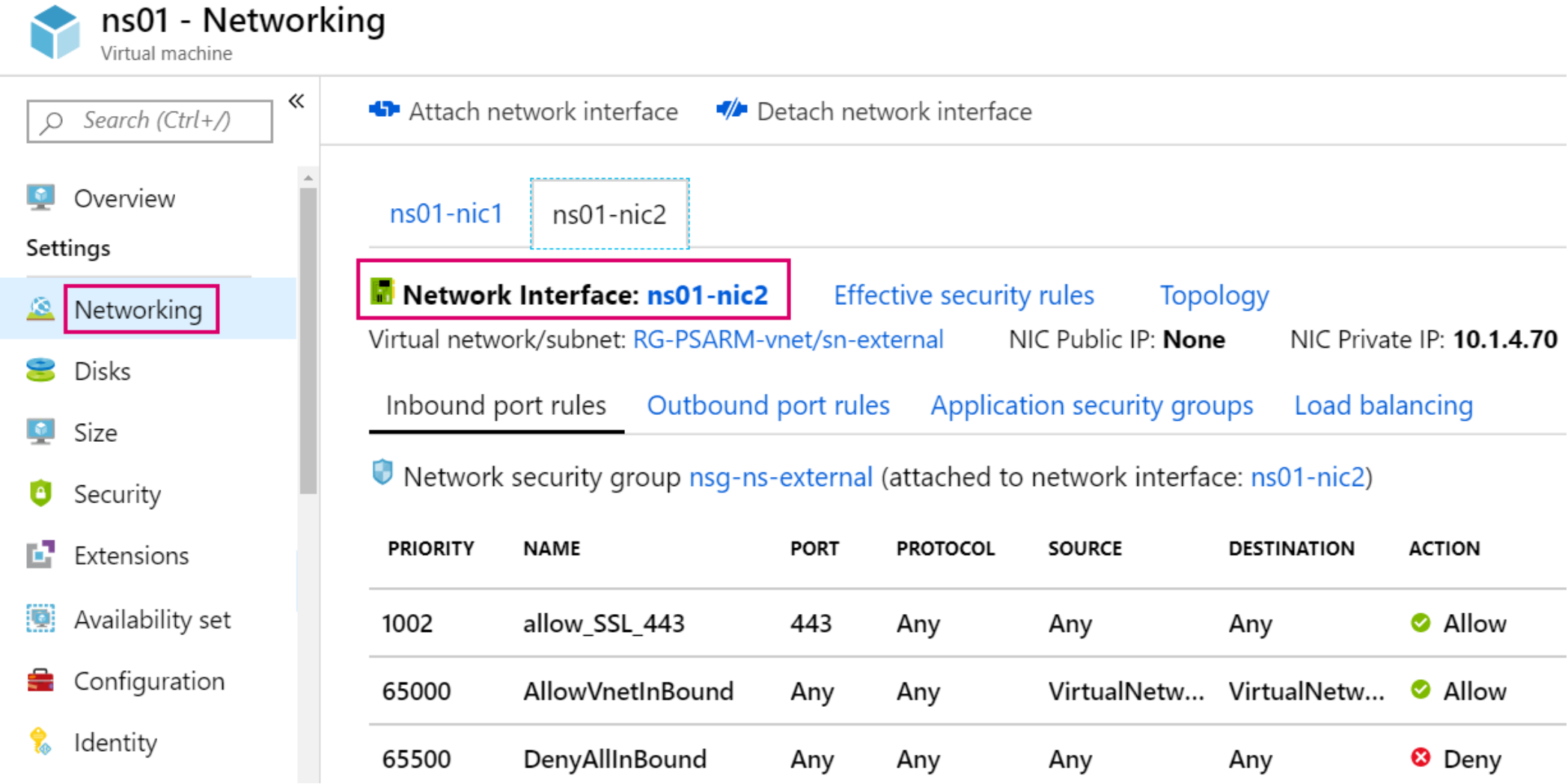
All resources

Standaardmap

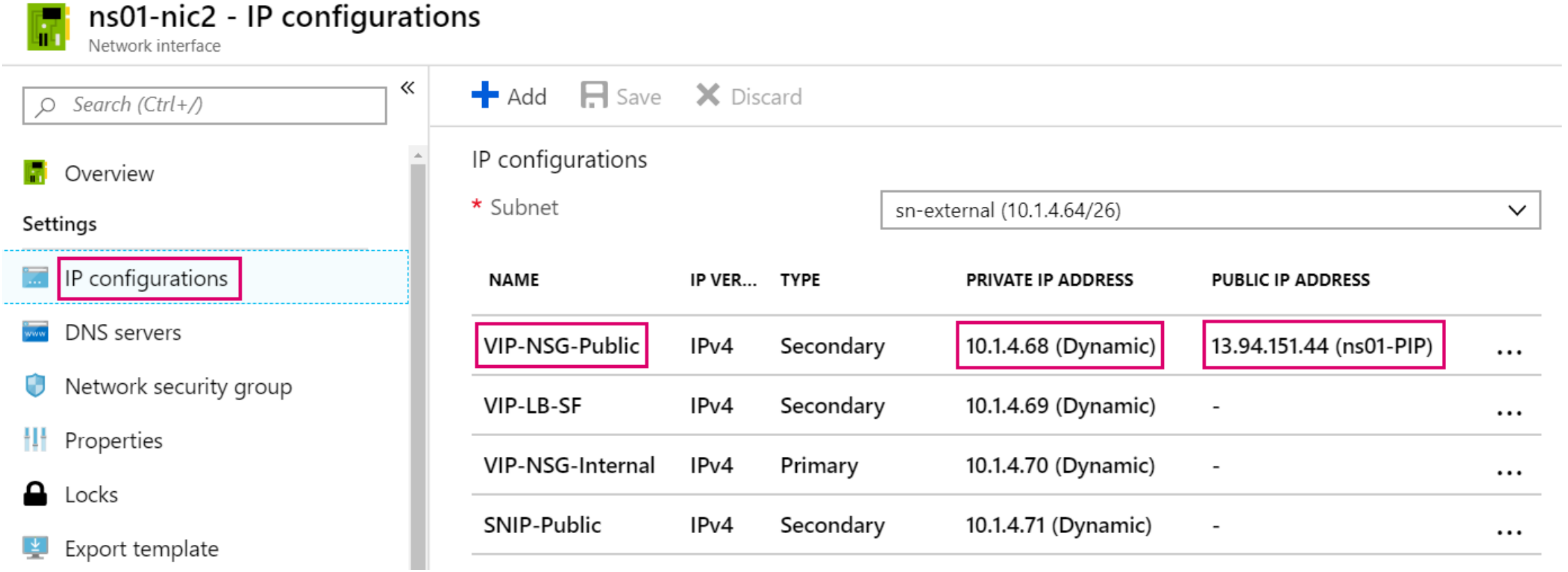
Add | Edit columns | Refresh | Export to CSV | Assign tags | Delete | Try preview

NAME	TYPE	RESOURCE GROUP	LOCATION	SUBSCRIPTION
ns01	Virtual machine	RG-PSARM	West Europe	Visual Studio Enterprise
ns01_OsDisk_1_7d59669c···	Disk	RG-PSARM	West Europe	Visual Studio Enterprise
ns01-nic1	Network interface	RG-PSARM	West Europe	Visual Studio Enterprise
ns01-nic2	Network interface	RG-PSARM	West Europe	Visual Studio Enterprise
ns01-PIP	Public IP address	RG-PSARM	West Europe	Visual Studio Enterprise
nsg-ns-external	Network security group	RG-PSARM	West Europe	Visual Studio Enterprise
nsg-ns-internal	Network security group	RG-PSARM	West Europe	Visual Studio Enterprise
RG-PSARM-vnet	Virtual network	RG-PSARM	West Europe	Visual Studio Enterprise

Virtual Machine



Virtual Machine



Combining Dev and Ops

UPGRADING OPS TO DEVOPS

COGNITION

Next steps

Introducing Azure DevOps

Azure Boards

Plan, track, and discuss work across teams, deliver value to your users faster.

Azure Repos

Unlimited cloud-hosted private Git repos. Collaborative pull requests, advanced file management, and more.

Azure Pipelines

CI/CD that works with any language, platform, and cloud. Connect to GitHub or any Git provider and deploy continuously to any cloud.

Azure Test Plans

The test management and exploratory testing toolkit that lets you ship with confidence.

Azure Artifacts

Create, host, and share packages. Easily add artifacts to CI/CD pipelines.

The Scripts

WHERE TO GO NEXT?

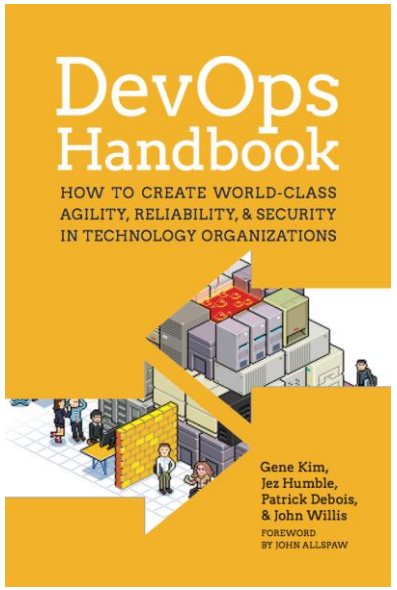
COGNITION

GitHub

Check out the scripts and get started with NITRO after the presentation

Sharing the **Azure Automation fun** and demo **scripts** with the **Community**:

https://github.com/cognitionIT/AzureAutomation



Presented by: **Esther Barthel** **Solutions Architect**

@virtuEs_IT http://nl.linkedin.com/in/ebarthel http://www.virtues.it

COGNITION
LEARN IT. KNOW IT. SHARE IT.

Thank You

COGNITION

LEARN IT. KNOW IT. SHARE IT.

Presented by: Esther Barthel — Solutions Architect

@virtuEs_IT

http://nl.linkedin.com/in/ebarthel

http://www.virtues.it

Bonus Scripts

AUTOMATE EVERYTHING

Bonus: Automate Automation

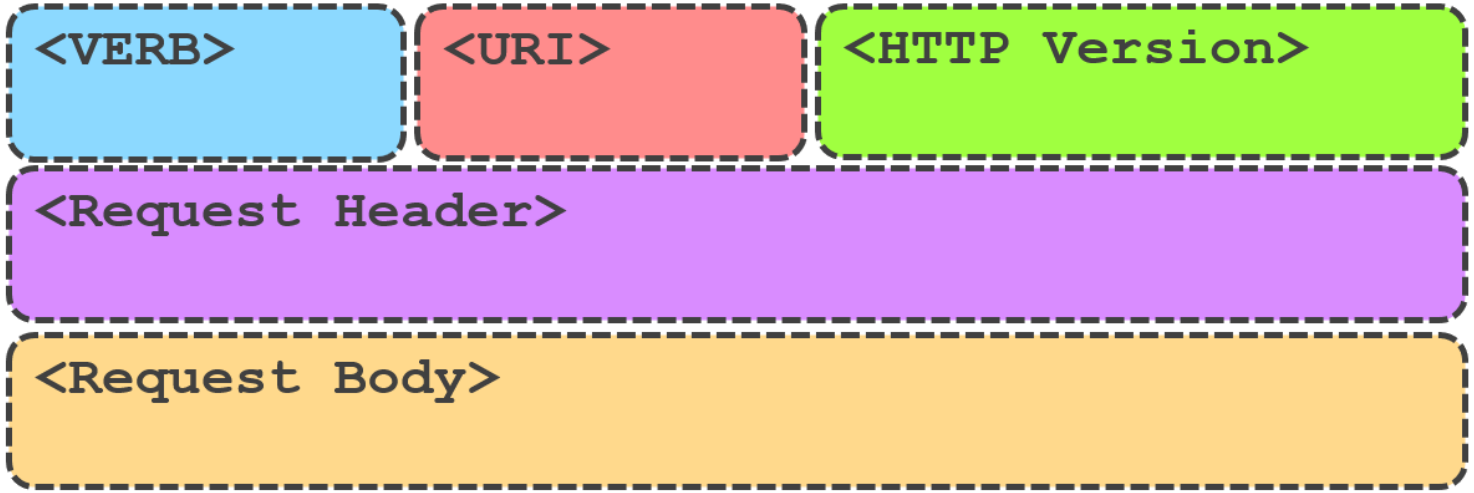
AUTOMATICALLY START A RUNBOOK

COGNITION

Bonus: Automatically start a Runbook

A way to interact with an API via series of HTTP calls

- **VERB:** HTTP Method (GET, PUT, POST, DELETE, OPTIONS)
- **URI:** the URI of the resource on which the operation is performed
- **HTTP Version:** usually "HTTP v1.1"
- **Request Header:** contains metadata (formatting, etc.)
- **Request Body:** actual message content



COGNITION

Bonus: Automatically start a Runbook

```
#region Start specified Runbook
    $Uri = 'https://management.azure.com/subscriptions/'
    $Uri = $Uri + '{0}/resourceGroups/{1}/providers/Microsoft.Automation/automationAccounts/{2}/jobs/{3}?api-version={4}' `
    -f $SubscriptionID, $ResourceGroupName, $AutomationAccountName, $((New-Guid).guid), $APIVersion

    $body = ConvertTo-Json @{
        "properties" = @{
            "runbook"   = @{"name" = $RunbookName}
            "parameters" = @{"NetScalerName" = "ns01";"NSPassword" = "NetScalerDemo!";"NSUsername" = "adm-demo"}
            "runon" = $HybridWorkerGroup
        }
        "tags" = @{}
    } -Depth 5

    # Invoke-RestMethod parameters
    $params = @{
        ContentType = "application/json"
        Headers = @{"authorization" = "Bearer $($token.Access_Token)"}
        Method = "Put"
        URI = $Uri
        Body = $body
    }

    # Make the REST API call
    $oRunbook = Invoke-RestMethod @params
#endregion
```

COGNITION

PowerShell in Action

AutoStartAzureAutomationRunbook_v0_4.ps1

```powershell
#requires -Version 3
# ------------------------------------------------------
# Based upon the script shared by MS employee Stefan Stranger
#               https://blogs.technet.microsoft.com/stefan_stranger/2017/08/09/triggering-azure-automation-runbooks-using-the-azure-arm-rest-api/
# ... and the script from Laurie Rhodes
#             http://www.laurierhodes.info/?q=node/118
# ------------------------------------------------------


## -------------------------------------------------------------------
## | Section 1 - Create a Access Token for Azure with REST API call |
## -------------------------------------------------------------------
Write-Output ""
Write-Host "* Creating an Access Token for Azure, using REST API" -ForegroundColor Yellow

# Azure Automation account information
    $ResourceGroupName = "RG-PSARM"
    $AutomationAccountName = "DevOps-PSARM"
    $APIVersion = "2015-10-31"
    $RunbookName = "Az_RB_deploy_NS_v4"
    $HybridWorkerGroup = "HWG-PSARM"

#region Read App secrets from CSV file
    #source: https://docs.microsoft.com/en-us/powershell/module/microsoft.powershell.utility/import-csv?view=powershell-6
    $AppSecrets = Import-Csv -Path "C:\`$_Sources\Azure_App_Secrets.csv" -Delimiter ","
    $ClientID = $AppSecrets.AppID
    $ClientSecret = $AppSecrets.AppKey
    $TenantID = $AppSecrets.TenantID
    $SubscriptionID = $AppSecrets.SubscriptionID
#endregion

$TokenEndpoint = {https://login.windows.net/{0}/oauth2/token} -f $TenantID
$ARMResource = "https://management.core.windows.net/";

# Create the JSON payload
$Body = @{
        'resource'= $ARMResource
        'client_id' = $ClientID
        'grant_type' = 'client_credentials'
        'client_secret' = $ClientSecret
}

```

```
PS C:\_Scripts\PoSH\AzureAutomation>
```

Completed

Ln 1 Col 38

145%