
# NetScaler Automation

## Using the PS-NITRO Module

**Esther Barthel, MSc**

**Solutions Architect**


**@virtuEs_IT**


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


**http://www.virtues.it**

# Agenda

- Introduction
  - RESTful APIs & NITRO
  - PowerShell NITRO Module
- Automated NetScaler Configurations
  - Basic Settings
  - NetScaler Load Balancing
- Bonus Slides
  - Deploy NetScaler VPX w/ fixed IP-address
  - Monitoring

# Introduction

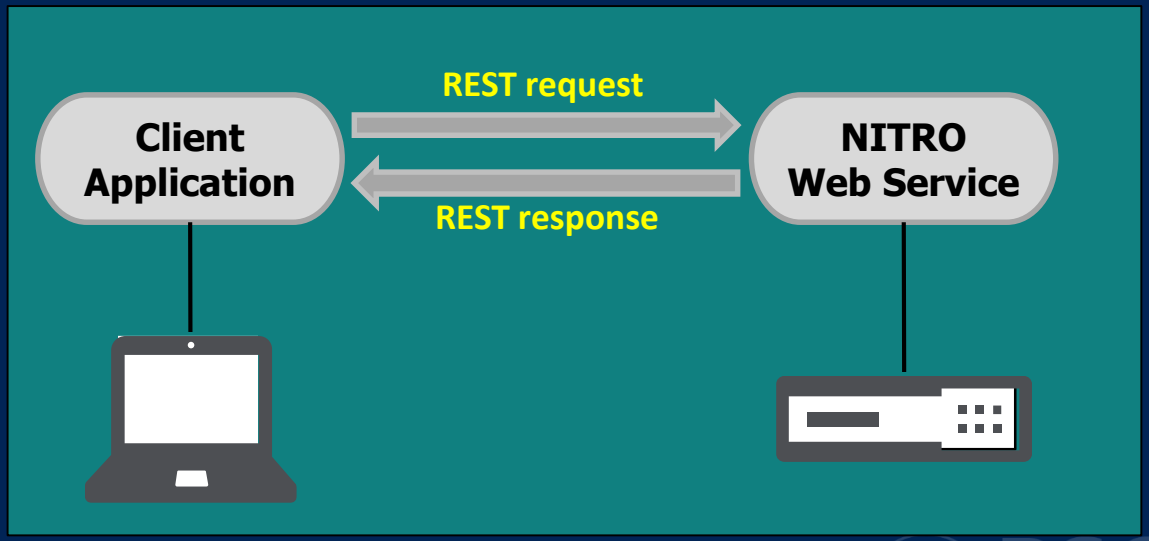
REST API & NITRO

# Restful APIs

- REpresentational State Transfer (REST)
  - **Client-Server**
  - Stateless
  - Standards-based (runs on top of **HTTP protocol**)
  - Easily used in presence of **firewalls** (port 80 or 443)

# Restful APIs

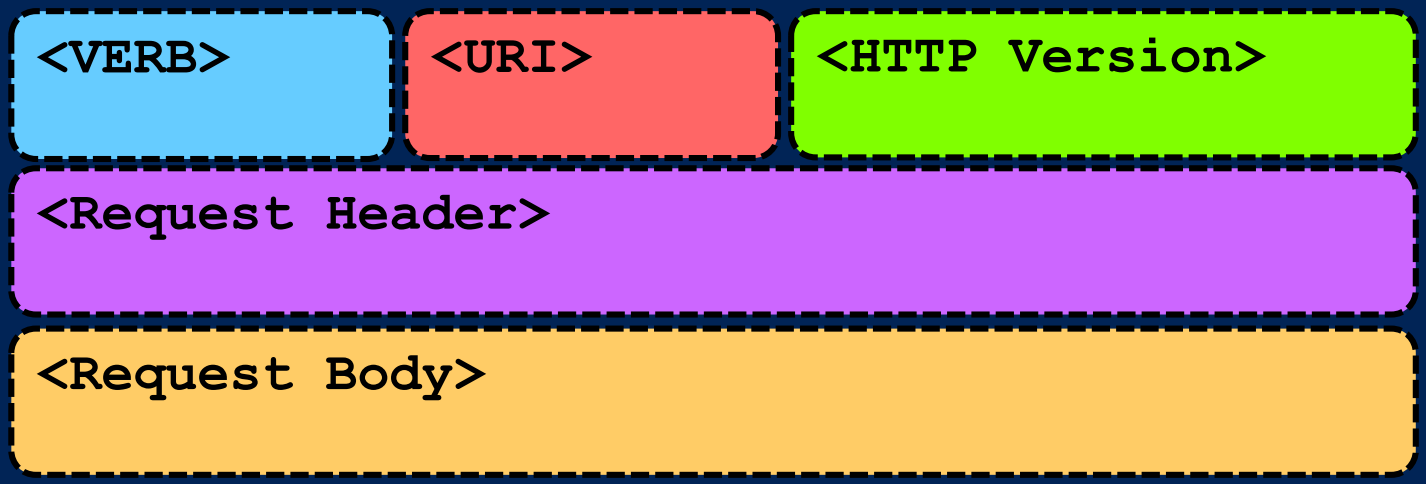
- NITRO (Restful service on NetScaler)
  - Communicate with NetScaler programmatically
  - Human readable HTTP-based interaction
  - **NITRO SDKs** available for multiple languages (Java, Python, .NET, REST)



# Restful APIs

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

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



# NITRO


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

- **HTTP Method** (GET, PUT, POST, DELETE)
- **URL:** http://**<ns-ipaddress>**/nitro/v1**/config /nsfeature?action=enable**
  - Add a **basic URL stem** to use with NITRO:
    - ~~http://<NSIP>/nitro/v1/**stat/**~~ ~~–> entity and system statistics~~
    - http://<NSIP>/nitro/v1/**config/** –> configuration operations
  - Add the **resource type** to the **URL** (and in some cases also the **resource name**)
  - Specify an **action** for the **URL** (bind, unset, enable, disable)

# NITRO

<Request Header>

<Request Body>

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

- Specify the **Content-Type** in the **Request Header**:
  - Content-Type: application/vnd.com.citrix.netscaler.**nsfeature+json**

(or generic content-type: **application/json**)

- Add the **JSON payload** to the **Request Body**:

```
{
   "login":
   {
      "username":"admin",
      "password":"verysecret"
   }
}
```

```
{
   "lbvserver":[
      {"name":"lbvserver1","servicetype":"http"},
      {"name":"lbvserver2","servicetype":"ssl"},
      {"name":"lbverver3","servicetype":"ftp"}
   ]
}
```

# NITRO

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

**url:** http://<ns-ipaddress>/nitro/v1/config/nsfeature?action=enable **method:** POST

**header:** Content-Type: application/json

**payload:**

```
{
   "nsfeature":
   {
      "feature":
      [
         "LB",
         "SSL",
         "SSLVPN"
      ]
   }
}
```

# NITRO API reference

CITRIX
Product Documentation

Browse

Search

NetScaler 11.1
NITRO API
REST Web Services
API Reference
Configuration

NS
- nsconfig
- nsip
- nsfeature
- nshostname
- nshttpparam

## NS

System/Global level configuration.

| Name | Description |
|---|---|
| nsconfig | |
| nsfeature | |
| nshostname | |
| nsip | |
| nslicense | |
| nsmode | |

### Operations (click to see Properties)

ENABLE | DISABLE | GET (ALL)

enable

**URL:** http://<netscaler-ip-address>/nitro/v1/config/nsfeature?**action=enable**

**HTTP Method:** POST

**Request Headers:**

```
Cookie:NITRO_AUTH_TOKEN=<tokenvalue>
Content-Type:application/json
```

**Request Payload:**

```
{"nsfeature":{
      "feature":<String[]_value>
}}
```

**Response:**

HTTP Status Code on Success: 200 OK
HTTP Status Code on Failure: 4xx <string> (for general HTTP errors) or 5xx <string> (for NetScaler-specific errors). The response payload provides details of the error

# NITRO requirements

**NetScaler 9.2 or later**

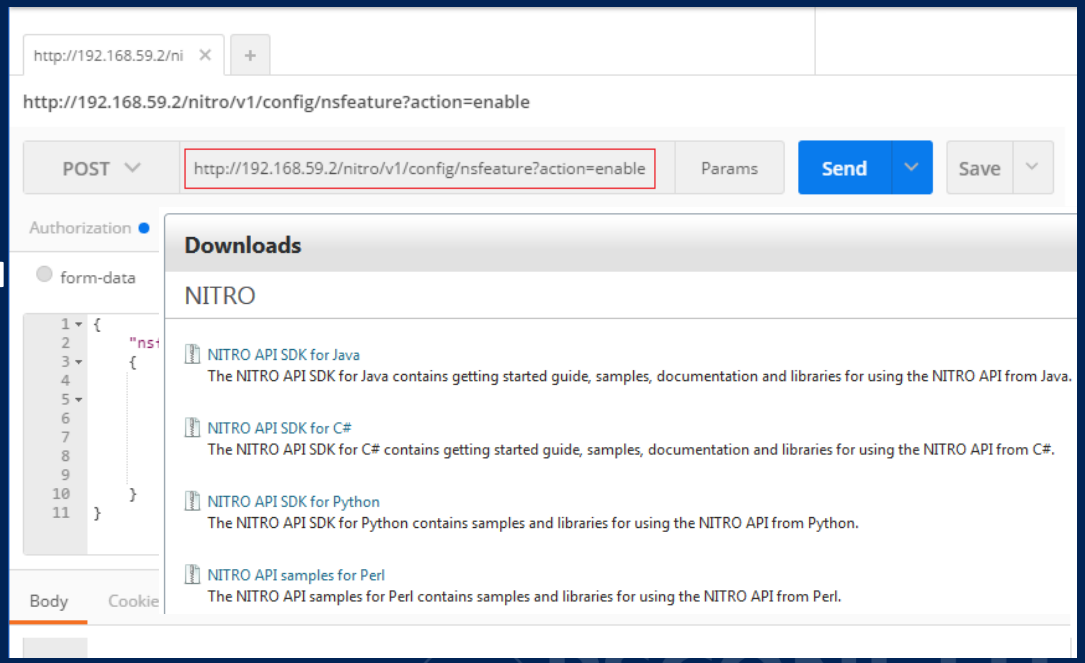
- Account with **execute** permissions
- NetScaler **NSIP** (or SNIP) with Management Access enabled

**REST Client**

- Use a client like cURL or **POSTMAN**

**Programming language w/ REST su**

- Download the **NITRO SDK**



# Automating NetScaler Configurations

## A Basic Example

# NetScaler Automation w/ NITRO

## Global NetScaler Features

**url:** http://<ns-ipaddress>/nitro/v1/config/nsfeature?action=enable **method:** POST

**header:** Content-Type: application/json

**payload:**

```
{
   "nsfeature":
   {
      "feature":
      [
         "LB",
         "SSL",
         "SSLVPN"
      ]
   }
}
```

# NetScaler Automation w/ PowerShell

## PowerShell NITRO Module

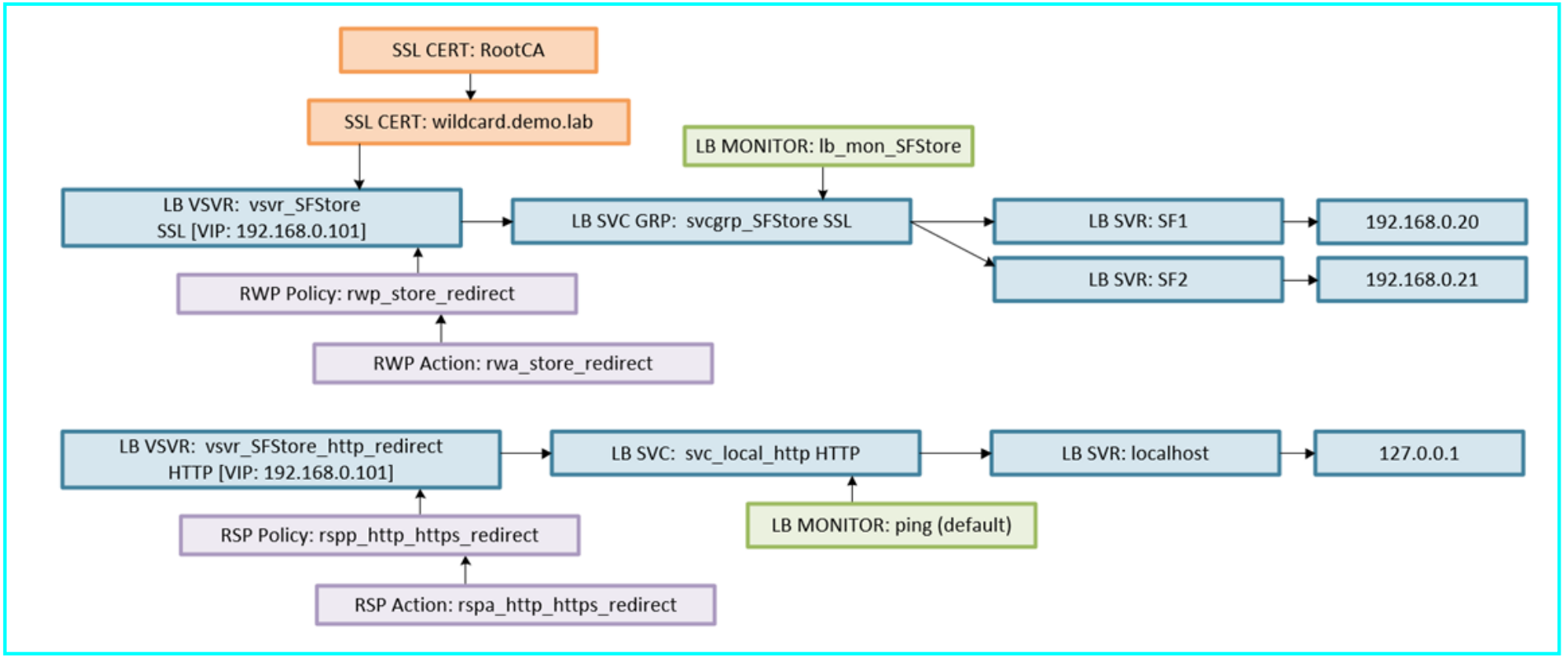
```
# Load the NetScaler PowerShell Module
Import-Module "$RootFolder\NitroConfigurationFunctions" -Force

# Connection protocol for the NetScaler
Set-NSMgmtProtocol -Protocol http

# Connect to the NetScaler (Start the session)
$NSSession = Connect-NSAppliance -NSAddress $NSAddress `
                -NSUserName $NSUserName - NSPAssword $PNUserPW

# Enable NetScaler Basic $ Advanced Features
Enable-NSFeature -NSSession $NSSession -Feature "lb ssl sslvpn"

#Disconnect from the NetScaler (End the session)
Disconnect-NSAppliance -NSSession $NSSession
```

# Demo

Basic NetScaler Configuration

# Automating NetScaler Configurations

## VPX Load Balancing



# NetScaler Automation w/ NITRO

## NetScaler Load Balancing – Rewrite & Responder policies

```
#region Configure Rewrite Actions & Policies
    Add-NSRewritePolicy -NSSession $NSSession -PolicyName "rwp_store_redirect" `
        -PolicyAction "rwa_store_redirect" -PolicyRule "HTTP.REQ.URL.EQ(""/"")"

    Add-NSRewriteAction -NSSession $NSSession -ActionName "rwa_store_redirect" `
        -ActionType "replace" -TargetExpression "HTTP.REQ.URL" `
        -Expression """/Citrix/StoreWeb"""
#endregion

#region Configure Responder Actions & Policies
    Add-NSResponderPolicy -NSSession $NSSession -Name "rspp_http_https_redirect" `
        -Rule "HTTP.REQ.IS_VALID" -Action "rspa_http_https_redirect"

$TargetExpression = """https://"" + HTTP.REQ.HOSTNAME.HTTP_URL_SAFE + " + `
                    "HTTP.REQ.URL.PATH_AND_QUERY.HTTP_URL_SAFE"

    Add-NSResponderAction -NSSession $NSSession -ActionName "rspa_http_https_redirect" `
    -ActionType "redirect" -TargetExpression $TargetExpression -ResponseStatusCode 302
#endregion
```

# NetScaler Automation w/ NITRO

## NetScaler Load Balancing – Servers, Services & Service Groups

```
#region Configure Load Balancing - Servers
    Add-NSServer -NSSession $NSSession -Name "SF1" -IPAddress "$SubnetIP.21"
#endregion

#region Configure Load Balancing - Services
    Add-NSService -NSSession $NSSession -Name "svc_local_http" -ServerName "localhost" `
        -Protocol HTTP -Port 80
#endregion

#region Configure Load Balancing - Service Groups
    Add-NSServiceGroup -NSSession $NSSession -Name "svcgrp_SFStore" -Protocol HTTP `
        -CacheType SERVER -Cacheable -State ENABLED -HealthMonitoring -AppflowLogging `
        -AutoscaleMode DISABLED

    New-NSServicegroupServicegroupmemberBinding -NSSession $NSSession -Name "svcgrp_SFStore" `
        -ServerName "SF1" -Port 80 -State ENABLED
#endregion
```

# NetScaler Automation w/ NITRO

## NetScaler Load Balancing – Monitors & vServers

```
#region Configure Load Balancing - Servers
    Add-NSServer -NSSession $NSSession -Name "SF1" -IPAddress "$SubnetIP.21"
#endregion

#region Configure Load Balancing - Services
    Add-NSService -NSSession $NSSession -Name "svc_local_http" -ServerName "localhost" `
        -Protocol HTTP -Port 80
#endregion

#region Configure Load Balancing - Service Groups
    Add-NSServiceGroup -NSSession $NSSession -Name "svcgrp_SFStore" -Protocol HTTP `
        -CacheType SERVER -Cacheable -State ENABLED -HealthMonitoring -AppflowLogging `
        -AutoscaleMode DISABLED

    New-NSServicegroupServicegroupmemberBinding -NSSession $NSSession -Name "svcgrp_SFStore" `
        -ServerName "SF1" -Port 80 -State ENABLED
#endregion
```

# Demo

Load Balancing Configuration

# The Scripts

Where to go next?

# GitHub

**Check out the PS-NITRO repository**

Sharing the **extended** PowerShell **Module** w/ **Community**:

**https://github.com/cognitionIT/PS-NITRO**

# Summary

- Use the NITRO PowerShell module to automate your NetScaler configuration
- Get the latest version on GitHub: https://github.com/cognitionIT/PS-NITRO
- Feel free to reach out for questions, feature requests and code contributions

# Questions?

# Bonus Slides

Automate Deployments

# NetScaler VPX w/ fixed IP

## CTX128236: How To Auto-Provision NetScaler VPX on XenServer

From an SSH console, type the following command:

**# xe vm-import filename=nsvpx-9.2-50.4_nc.xva**

where the filename of the NetScaler VPX virtual machine depends on the release and build that you wish to install.

5. From the SSH session console, type:
   **# xe vm-param-add uuid=<vm uuid> param-name=xenstore-data vm-data/ip=<ip address>**
   **# xe vm-param-add uuid=<vm uuid> param-name=xenstore-data vm-data/netmask=<netmask>**
   **# xe vm-param-add uuid=<vm uuid> param-name=xenstore-data vm-data/gateway=<gateway>**

6. Start the NetScaler VPX virtual machine by typing:
   **# xe vm-start vm=NetScaler\ Virtual\ Appliance**

   Type xe vm-start vm= and then press the tab key to see a full list of virtual machine names with backslashes preceding special characters. You can
7. When the NetScaler
   parameters. If the vn
   machine console, ac
8. Following NetScaler
   From XenCenter, sel
   command
   **> sh ns ip**

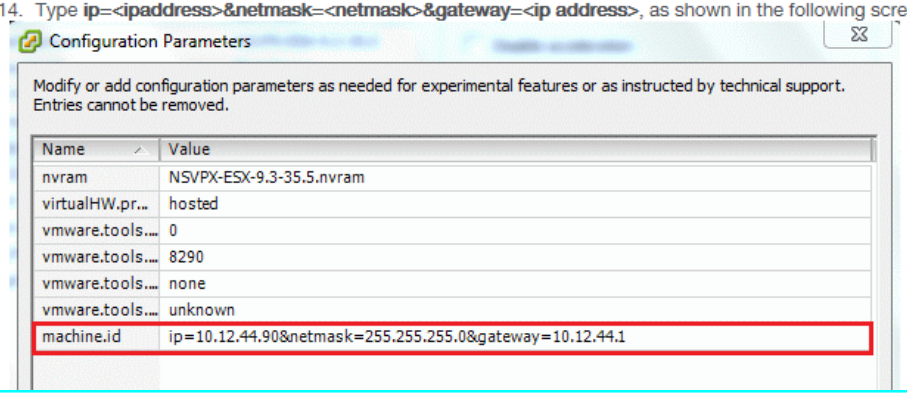
```
 Ipaddress

1) 10.217.147.145
Done
```

```powershell
# Import the NetScaler image to the default SR
Import-XenVm -XenHost $XSServer -Path $sourcePath

# Get the imported NS VPX VM uuid by it's default name
$oVM = Get-XenVM -Name "NetScaler Virtual Appliance"

# Get current VM XenStoreData values
$newHash = $oVM.xenstore_data

# Add required values for fixed IP settings for the NetScaler VPX
$newHash.add("vm-data/ip",$nsIPAddress)
$newHash.add("vm-data/netmask",$nsNetmask)
$newHash.add("vm-data/gateway",$nsGateway)

# Add new values to current VM XenStoreData (works only once, before NS is booted)
Set-XenVM -VM $oVM -XenstoreData $newHash

# Start the NS Appliance
Invoke-XenVM $oVM -XenAction Start
```

# NetScaler VPX w/ fixed IP

## CTX128250: How To Auto-Provision NetScaler VPX on ESX or ESXi Host

12. Click **Add Row** in the Configuration Parameters dialog box.
13. Type **machine.id** as the name in the new row.
14. Type **ip=<ipaddress>&netmask=<netmask>&gateway=<ip address>**, as shown in the following screenshot:




```
# Import OVF on specified ESXi Host
Import-vApp -Source $vAppSource -VMHost $vmHost -Force -WarningAction SilentlyContinue

# Get VM (based upon default name after import)
$VM = Get-VM -Name "NSVPX-ESX" -Server $ESXiHostName

#Configure fixed IP for NetScaler
New-AdvancedSetting -Entity $VM -Name "machine.id" `
    -Value "ip=$IPaddress&netmask=$SubnetMask&gateway=$DefaultGW" -Confirm:$false -Force:$true

# Start VM
Start-VM $VM
```

# NetScaler VPX w/ fixed IP

## Citrix Product Documentation: SDX 11.1 – NITRO API Documentation

**url:** http://<ns-ipaddress>/nitro/v2/config/ns **method: POST**

**header:** Content-Type: **application/ json**

**payload:**

```
{
   "ns":
   {
      "name":"VPX01",
      "ip_address":"192.168.59.2",
      "netmask":"255.255.255.0",
      "gateway":"192.168.59.1",
      "imag
      "vm_m
      "thro
      "pps"
      "lice
      "prof
      "user
      "pass
      "netw
   }
}
```

```powershell
# Specifying the correct URL
$strURI = "http://10.42.0.51/nitro/v2/config/ns?action=add"
$ContentType = "application/json"

# Creating the right payload formatting (mind the Depth for the nested arrays)
$payload = @{
"ns"= @{
    "name"="testVPX";
    "ip_address"="192.168.59.2";
    "netmask"="255.255.255.0";
    "gateway"="192.168.59.1";
    "image_name"="NSVPX-XEN-11.1-50.10_nc.xva";
    "vm_memory_total"=2048;
    "throughput"=1000;
    "pps"=1000000;
    "license"="Platinum";
    "profile_name"="ns_nsroot_profile";
    "username"="admin";
    "password"="admin";
    "network_interfaces"=@(@{"port_name"="10/1"},@{"port_name"="10/2"})
    }
} | ConvertTo-Json -Depth 5

# Making the REST API call to the NetScaler SDX to create a NetScaler VPX Instance
Invoke-RestMethod -Method Post -Uri $strURI -Credential $MyCreds -Body ("object=" + $payload) `
    -ContentType $ContentType -Verbose
```

# Bonus Slides

Monitoring

# Demo

Monitoring

# Next Steps...

- Now: 15 min break
- Grab a coffee
- Next up:
  - Part 2: **NITRO Deep Dive with Invoke-RestMethod**
  - Or change track & switch to another room
- Ask me questions or meet me in a breakout session room afterwards


# psconf.eu 2018

scheduled to be in the week of

## April 16-20, 2018

details on www.psconf.eu as they become available

# About_Author

- 15+ years of Technical Consulting
- 5+ years of DevOps scripting fun

- Citrix CTP (2015 -2017)
- Microsoft MVP for Enterprise Mobility (2017)

**Esther Barthel, MSc**

**Solutions Architect**


**@virtuEs_IT**


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


**http://www.virtues.it**