



OSS in the Cloud – Laboratorio 2

Nombre del Laboratorio:
Scaling on Azure

Preparado para:
Microsoft

Fecha:
[02/28/2018]

Versión: **[1.0]**

Preparado por:
Manuel Alejandro Peña Sánchez
oss@bside.com.mx

bSide
Expertos en Soluciones TI, S.C.



Contenido

Metas y objetivos del Laboratorio	3
Alcance del Laboratorio	3
Desarrollo del Laboratorio.....	4
Despliegue de Ubuntu en Windows 10	4
Despliegue de Azure CLI 2.0	5
Despliegue de Ambiente Elástico Open Source Software.....	7
Preparativos Imagen de Máquina Virtual	11
Despliegue de AutoScaling Open Source Software	13
Demostración Escalamiento:	14
Eliminar Ambiente.....	15
Conclusiones	15
Contacto	15



Metas y objetivos del Laboratorio

Esta sección describe las metas y los objetivos que se pretende lograr con este laboratorio.

A continuación se enlistan las metas

- Microsoft Loves Linux.
- Windows 10 + Linux.
- Conocer la facilidad de despliegue de Servicios en Azure por medio de Azure CLI 2.0.
- Automatización en Azure por medio de Azure CLI 2.0.
- Conocer la facilidad de despliegue en Azure de un ambiente elástico y de autoescalamiento con Open Source Software .

A continuación se enlistan los objetivos:

- Que el cliente conozca las bondades y facilidad de despliegue de ambientes Open Source Software elásticos en Azure con Virtual Machine Scale Set.

Alcance del Laboratorio

Como parte de esta demo solo se cubre:

1. Despliegue de Azure Virtual Machine Scale Set con Sistema Operativo Linux
2. Creación de Imágenes Open Source Software para Máquinas Virtuales en Azure.

.

Desarrollo del Laboratorio

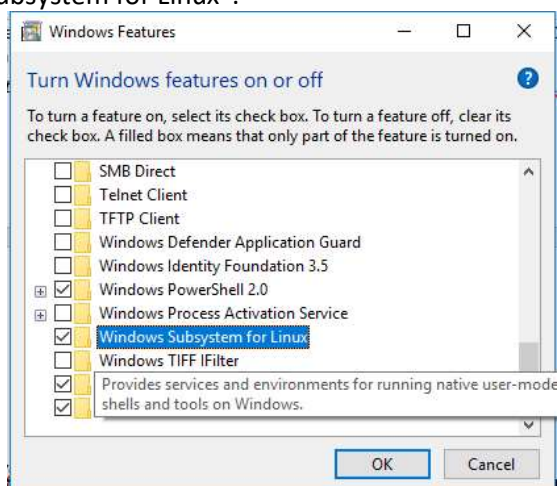
Despliegue de Ubuntu en Windows 10

En este momento se instalan los **Pre-requisitos** para poder ejecutar el script de despliegue de elementos de Azure por medio de Azure Cli 2.0.

La primera aportación (2009) de Microsoft al mundo Open Source Software fue la contribución de 20,000 líneas de código al Kernel de Linux.

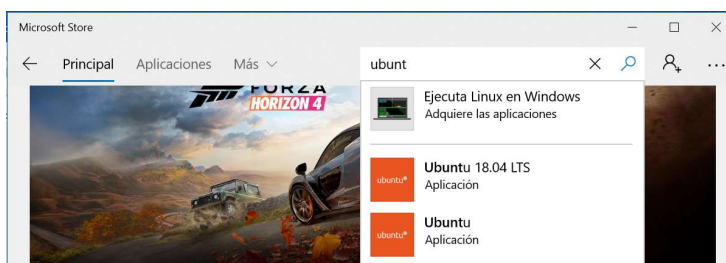
En esta sección se muestra la facilidad de instalar un ambiente Linux dentro de Windows 10 para poder gestionar centralmente dos grandes mundos Windows / Linux.

1. Ingresar a *Control Panel -> Programs -> Programs and Features -> Turn Windows features on or off*.
2. Habilitar "Windows Subsystem for Linux".



4

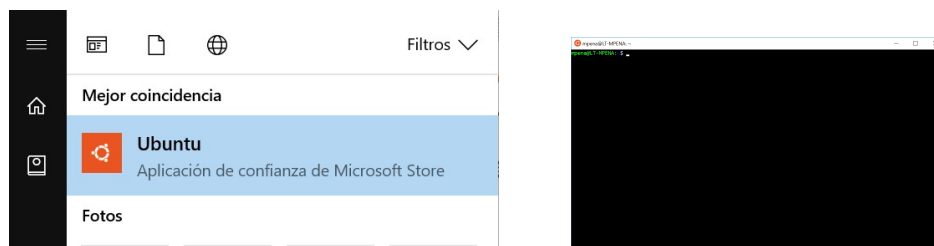
3. Solicitará el reinicio del equipo.
4. Después de Reiniciar el equipo hay que abrir el Microsoft Store y buscar Ubuntu



5. Dar clic en obtener:



6. Una vez instalado ir a menú de inicio y buscar Ubuntu para abrir la terminal:



Despliegue de Azure CLI 2.0

Es importante mencionar que gran parte de la base de Azure, no como todo mundo piensa, esta basado en Open Source, uno de los ejemplos es Azure CLI. Este CLI sirve para gestionar todo el mundo Azure.

5

- Ejecución:

En este punto realizaremos el despliegue de Azure CLI 2.0 dentro del bash instalado en Windows 10.

1. Agregar los repositorios de Microsoft para Azure Cli:

```
# echo "deb [arch=amd64] https://packages.microsoft.com/repos/azure-cli/ wheezy main"
| \ sudo tee /etc/apt/sources.list.d/azure-cli.list
```

```
# sudo apt-key adv --keyserver packages.microsoft.com --recv-keys
52E16F86FEE04B979B07E28DB02C46DF417A0893
```
2. Instalar soporte de repositorios por medio de HTTPS. (Mencionar punto de seguridad).

```
# sudo apt-get install -y apt-transport-https
```

```
psmapad@LTMPENA:/mnt/c/Users/mpena$ sudo apt-get install apt-transport-https
[sudo] password for psmapad:
Leyendo lista de paquetes... Hecho
Creando árbol de dependencias
Leyendo la información de estado... Hecho
```

3. Actualizar listado de repositorios e instalar azure-cli.

sudo apt-get update && sudo apt-get install -y azure-cli

```
psmapad@LTMPEA:~/mnt/c/Users/mpena$ sudo apt-get update && sudo apt-get install azure-cli
Obj:1 http://archive.ubuntu.com/ubuntu xenial InRelease
Des:2 http://security.ubuntu.com/ubuntu xenial-security InRelease [102 kB]
Des:3 http://archive.ubuntu.com/ubuntu xenial-updates InRelease [102 kB]
Des:4 http://security.ubuntu.com/ubuntu xenial-security/main amd64 Packages [384 kB]
Des:5 http://archive.ubuntu.com/ubuntu xenial-updates/main amd64 Packages [657 kB]
Des:6 http://security.ubuntu.com/ubuntu xenial-security/main Translation-en [170 kB]
Des:7 http://security.ubuntu.com/ubuntu xenial-security/restricted amd64 Packages [7 472 B]
Des:8 http://security.ubuntu.com/ubuntu xenial-security/restricted Translation-en [2 412 B]
```

```
Creando árbol de dependencias
Leyendo la información de estado... Hecho
El paquete indicado a continuación se instaló de forma automática y ya no es necesario.
snap-confine
Utilice «sudo apt autoremove» para eliminarlo.
Se actualizarán los siguientes paquetes:
azure-cli
1 actualizados, 0 nuevos se instalarán, 0 para eliminar y 80 no actualizados.
```

4. Después de esto se puede utilizar Azure CLI 2.0:

```
psmapad@LTMPEA:~/mnt/c/Users/mpena$ az login --username mpena@bSide.com.mx
Password:
[
  {
    "cloudName": "AzureCloud",
    "id": " ",
    "isDefault": true,
    "name": " ",
    "state": "Enabled",
    "tenantId": " ",
    "user": {
      "name": " ",
      "type": "user"
    }
  }
]
```

5. Ejemplo de Listado de Resource Group

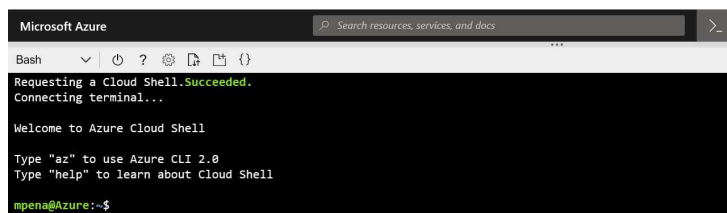
az group list -o table

```
mpena@LT-MPEA:~$ az group list -o table
Name                                Location    Status
-----
eastus                             Succeeded
eastus                             Succeeded
southcentralus                    Succeeded
eastus                             Succeeded
southcentralus                    Succeeded
southcentralus                    Succeeded
southcentralus                    Succeeded
mpena@LT-MPEA:~$
```

Despliegue de Ambiente Elástico Open Source Software

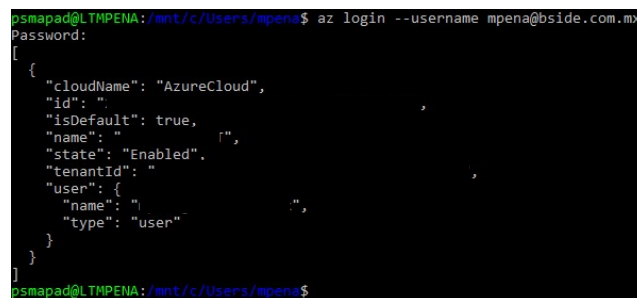
Microsoft es uno de los contribuyentes más grande de Github, esto desde antes de su adquisición. Existen miles de templates para el despliegue de recursos de Azure en este proyecto.

Se mostrará el despliegue de una VM por medio de Azure CLI 2.0. Hay que hacer énfasis en las posibilidades que esto conlleva. Como creación de scripts para despliegue de VMs Masivas. Adicional que este mismo CLI se encuentra en el portal de Azure para su uso sin necesidad de instalar nada.



- Ejecución:

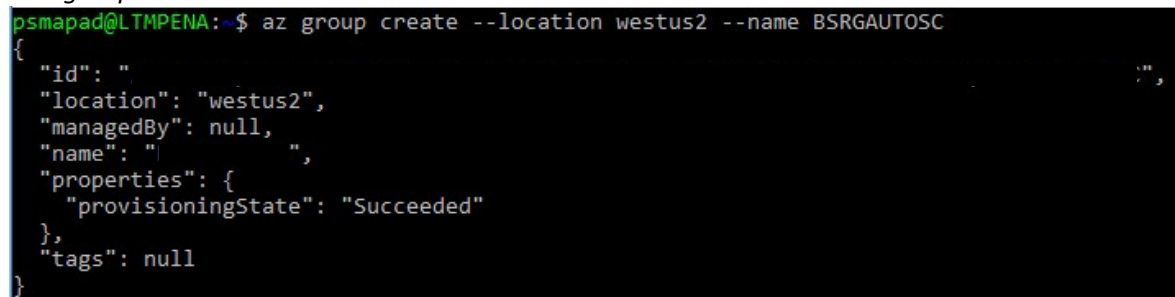
1. Acceder a Tenant por medio de Azure CLI 2.0



```
# az login --username <usuario@dominio>
```

7. Crear Resource Group desde Azure CLI.

```
# az group create --location westus2 --name BSRAUTOSC
```



8. Crear Public IP (Guardar el campo fqdn)

az network public-ip create --resource-group BSRGAUTOSC --name BSPIPAUTOSC --dns-name bsdautosc

```
psmapad@LTMPEA: $ az network public-ip create --resource-group BSRGAUTOSC --name BSPIPAUTOSC --dns-name bsdautosc
{
  "publicIp": {
    "dnsSettings": {
      "domainNameLabel": "bsdautosc",
      "fqdn": "bsdautosc.westus2.cloudapp.azure.com",
      "reverseDns": null
    },
    "etag": "W/\"00ad1daa-1fb1-4345-949a-7846ebdfb5c2\"",
    "id": "/subscriptions/.../resourceGroups/BSRGAUTOSC/providers/Microsoft.Network/publicIPAddresses/BSPIPAUTOSC",
    "idleTimeoutInMinutes": 4,
    "ipAddress": null,
    "ipConfiguration": null,
    "location": "westus2",
    "name": "BSPIPAUTOSC",
    "provisioningState": "Succeeded",
    "publicIpAddressVersion": "IPv4",
    "publicIpAllocationMethod": "Dynamic",
    "resourceGroup": "BSRGAUTOSC",
    "resourceGuid": "...",
    "sku": {
      "name": "Basic"
    },
    "tags": null,
    "type": "Microsoft.Network/publicIPAddresses"
  }
}
```

9. Crear Subred

az network vnet create --resource-group BSRGAUTOSC --name BSVNETAUTOSC --address-prefix 10.0.0.0/24 --subnet-name BSVNETINTAUTOSC --subnet-prefix 10.0.0.0/24

```
psmapad@LTMPEA: $ az network vnet create --resource-group BSRGAUTOSC --name BSVNETAUTOSC --address-prefix 10.0.0.0/24 --subnet-name BSVNETINTAUTOSC --subnet-prefix 10.0.0.0/24
{
  "newVNet": {
    "addressSpace": {
      "addressPrefixes": [
        "10.0.0.0/24"
      ]
    },
    "dhcpOptions": {
      "dnsServers": []
    },
    "etag": "W/\"...",
    "id": "/subscriptions/.../resourceGroups/BSRGAUTOSC/providers/Microsoft.Network/virtualNetworks/BSVNETAUTOSC",
    "location": "westus2",
    "name": "BSVNETAUTOSC",
    "provisioningState": "Succeeded",
    "resourceGroup": "BSRGAUTOSC",
    "resourceGuid": "...",
    "sku": {
      "name": "Basic"
    },
    "tags": null,
    "type": "Microsoft.Network/virtualNetworks"
  }
}
```

8

10. Crear Network Security Group con los permisos de administración.

az network nsg create --resource-group BSRGAUTOSC --name BSNSGAUTOSC

```
psmapad@LTMPEA: $ az network nsg create --resource-group BSRGAUTOSC --name BSNSGAUTOSC
{
  "NewNSG": {
    "defaultSecurityRules": [
      {
        "access": "Allow",
        "description": "Allow inbound traffic from all VMs in VNET",
        "destinationAddressPrefix": "VirtualNetwork",
        "destinationAddressPrefixes": [],
        "destinationPortRange": "*",
        "destinationPortRanges": [],
        "direction": "Inbound",
        "etag": "W/\"3...",
        "id": "/subscriptions/.../resourceGroups/BSRGAUTOSC/providers/Microsoft.Network/networkSecurityGroups/BSNSGAUTOSC/defaultSecurityRules/AllowVnetInBound",
        "name": "AllowVnetInBound",
        "provisioningState": "Succeeded",
        "resourceGroup": "BSRGAUTOSC",
        "resourceGuid": "...",
        "sku": {
          "name": "Basic"
        },
        "tags": null,
        "type": "Microsoft.Network/networkSecurityGroups"
      }
    ],
    "etag": "W/\"...",
    "id": "/subscriptions/.../resourceGroups/BSRGAUTOSC/providers/Microsoft.Network/networkSecurityGroups/BSNSGAUTOSC",
    "location": "westus2",
    "name": "BSNSGAUTOSC",
    "provisioningState": "Succeeded",
    "resourceGroup": "BSRGAUTOSC",
    "resourceGuid": "...",
    "sku": {
      "name": "Basic"
    },
    "tags": null,
    "type": "Microsoft.Network/networkSecurityGroups"
  }
}
```



```
# az network nsg rule create --resource-group BSRGAUTOSC --nsg-name BSNSGAUTOSC --name BSSSHAUTOSC --protocol tcp --priority 1000 --destination-port-range 22 --access allow
```

```
psmapad@LTMPEA: $ az network nsg rule create --resource-group BSRGAUTOSC --nsg-name BSNSGAUTOSC --name BSSSHAUTOSC --priority 1000 --destination-port-range 22 --access allow
{
  "access": "Allow",
  "description": null,
  "destinationAddressPrefix": "*",
  "destinationAddressPrefixes": [],
  "destinationPortRange": "22",
  "destinationPortRanges": [],
  "direction": "Inbound",
  "etag": "W/\\",
  "id": "/subscriptions/ /resourceGroups/BSRGAUTOSC/providers/Microsoft.Network/networkSecurityGroups/BSNSGAUTOSC",
  "name": "BSSSHAUTOSC",
  "priority": 1000,
  "protocol": "Tcp",
  "provisioningState": "Succeeded",
  "resourceGroup": "BSRGAUTOSC",
  "sourceAddressPrefix": "*",
  "sourceAddressPrefixes": [],
  "sourcePortRange": "*",
  "sourcePortRanges": []
}
```

11. Crear Load Balancer, para gestionar el autoscaling y el tráfico de entrada hacia los webserver.

```
# az network lb create --resource-group BSRGAUTOSC --name BSLBAUTOSC --public-ip-address BSIPIAUTOSC --frontend-ip-name BSFIPIAUTOSC --backend-pool-name BSBIPAUTOSC
```

```
psmapad@LTMPEA: $ az network lb create --resource-group BSRGAUTOSC --name BSLBAUTOSC --public-ip-address BSIPIAUTOSC --frontend-ip-name BSFIPIAUTOSC --backend-pool-name BSBIPAUTOSC
{
  "loadBalancer": {
    "backendAddressPools": [
      {
        "etag": "W/\\",
        "id": "/subscriptions/ /resourceGroups/BSRGAUTOSC/providers/Microsoft.Network/loadBalancers/BSLBAUTOSC/backendAddressPools/BSBIPAUTOSC",
        "name": "BSBIPAUTOSC",
        "properties": {
          "provisioningState": "Succeeded"
        }
      }
    ]
  }
}
```

12. Para que el balanceador sepa qué servicio es el que está gestionando se crea una prueba de disponibilidad del servicio.

```
# az network lb probe create --resource-group BSRGAUTOSC --lb-name BSLBAUTOSC --name BSLBHPAUTOSC --protocol tcp --port 80
```

```
psmapad@LTMPEA: $ az network lb probe create --resource-group BSRGAUTOSC --lb-name BSLBAUTOSC --name BSLBHPAUTOSC --protocol tcp --port 80
{
  "etag": "W/\\81",
  "id": "/subscriptions/ /resourceGroups/BSRGAUTOSC/providers/Microsoft.Network/loadBalancers/BSLBAUTOSC/probes/BSLBHPAUTOSC",
  "intervalInSeconds": 15,
  "loadBalancingRules": null,
  "name": "BSLBHPAUTOSC",
  "numberOfProbes": 2,
  "port": 80,
  "protocol": "Tcp",
  "provisioningState": "Succeeded",
  "requestPath": null,
  "resourceGroup": "BSRGAUTOSC"
}
```



13. Después de tener la prueba de disponibilidad tenemos que crear una regla de balanceo, esto significa que y como vamos a balancear.

```
# az network lb rule create --resource-group BSRGAUTOSC --lb-name BSLBAUTOSC --name BSLBRWEBAUTOSC --protocol tcp --frontend-port 80 --backend-port 80 --frontend-ip-name BSFIPAUTOSC --backend-pool-name BSBIPAUTOSC --probe-name BSLBHPAUTOSC
```

```
psmapad@LTMPEA: $ az network lb rule create --resource-group BSRGAUTOSC --lb-name BSLBAUTOSC --name BSLBRWEBAUTOSC --p
rotocol tcp --frontend-port 80 --backend-port 80 --frontend-ip-name BSFIPAUTOSC --backend-pool-name BSBIPAUTOSC --probe-
name BSLBHPAUTOSC
{
  "backendAddressPool": {
    "id": "/subscriptions/2.../resourceGroups/BSRGAUTOSC/providers/Microsoft.Network/loadBalancers/BSLBAUTOSC/backendAddressPools/BSBIPAUTOSC",
    "resourceGroup": "BSRGAUTOSC"
  },
  "backendPort": 80,
```

14. Crear Equipo para Imagen que se desplegará en el VM Scale Set.

```
# az vm create --resource-group BSRGAUTOSC --nsg BSNSGAUTOSC --vnet-name BSVNETAUTOSC --subnet BSVNETINTAUTOSC --name BSVAUTOSCBASE --size Standard_A1_v2 --image Debian --admin-username bsvmssadmin --admin-password '..Pa$$w0rd..'
```

```
psmapad@LTMPEA: $ az vm create --resource-group BSRGAUTOSC --nsg BSNSGAUTOSC --vnet-name BSVNETAUTOSC --subnet BSVNETIN
TAUTOSC --name BSVAUTOSCBASE --location westus2 --size Standard_A1_v2 --image Debian --admin-username --generate-s
sh-keys
{
  "fqdns": "",
  "id": "/subscriptions/...:/resourceGroups/BSRGAUTOSC/providers/Microsoft.Compute/virtu
alMachines/BSVAUTOSCBASE",
  "location": "westus2",
  "macAddress": "00-0D-3A-F9-5F-07",
  "powerState": "VM running",
  "privateIpAddress": "10.0.0.4",
```

NOTA: Nos proporcionara una IP Publica, favor de anotarla.

Preparativos Imagen de Máquina Virtual

En este punto prepararemos la VM que nos servirá de imagen para el laboratorio.

1. Ingresar a la VM por medio de cliente de SSH y convertirse en root.

`# sudo su -`

2. Actualizar

`# apt-get update`

```
root@BSVAUTOSCBASE:~# apt-get update
Ign http://debian-archive.trafficmanager.net jessie InRelease
Hit http://debian-archive.trafficmanager.net jessie/updates InRelease
Hit http://debian-archive.trafficmanager.net jessie-backports InRelease
Hit http://debian-archive.trafficmanager.net jessie Release.gpg
Hit http://debian-archive.trafficmanager.net jessie Release
Hit http://debian-archive.trafficmanager.net jessie/updates/main Sources
Hit http://debian-archive.trafficmanager.net jessie/updates/main amd64 Packages
Hit http://debian-archive.trafficmanager.net jessie/updates/main Translation-en
Get:1 http://debian-archive.trafficmanager.net jessie-backports/main Sources/DiffIndex [27.8 kB]
Get:2 http://debian-archive.trafficmanager.net jessie-backports/main amd64 Packages/DiffIndex [27.8 kB]
Get:3 http://debian-archive.trafficmanager.net jessie-backports/main Translation-en/DiffIndex [27.8 kB]
Hit http://debian-archive.trafficmanager.net jessie/main Sources
Hit http://debian-archive.trafficmanager.net jessie/main amd64 Packages
Hit http://debian-archive.trafficmanager.net jessie/main Translation-en
Fetched 83.4 kB in 4s (19.2 kB/s)
Reading package lists... Done
```

`# apt-get -y upgrade`

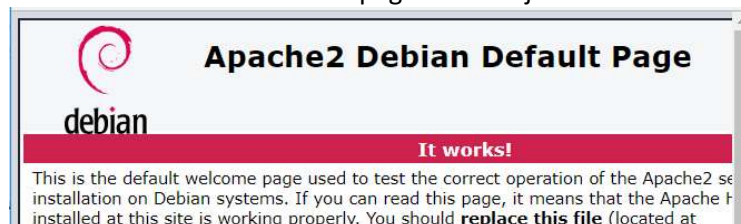
```
root@BSVAUTOSCBASE:~# apt-get -y upgrade
Reading package lists... Done
Building dependency tree
Reading state information... Done
Calculating upgrade... Done
The following packages will be upgraded:
  libssl1.0.0 openssl wget
3 upgraded, 0 newly installed, 0 to remove and 0 not upgraded.
Need to get 2207 kB of archives.
After this operation, 367 kB of additional disk space will be used.
```

3. *Instalar Apache HTTPD con PHP.* Hacer mención de que es el webserver y lenguaje de OSS más utilizado.

`# apt-get -y install apache2 php5`

```
root@BSVAUTOSCBASE:~# apt-get -y install apache2 php5
Reading package lists... Done
Building dependency tree
Reading state information... Done
The following extra packages will be installed:
  apache2-bin apache2-data apache2-utils libapache2-mod-php5 libapr1 libaprutil1 libaprutil1-dbd-sqlite3 libaprutil1-ldap liblua5.1-0
  libonig2 libperl4-corelibs-perl libqdbm14 libsof php5-cli php5-common php5-json php5-readline psmisc ssl-cert
```

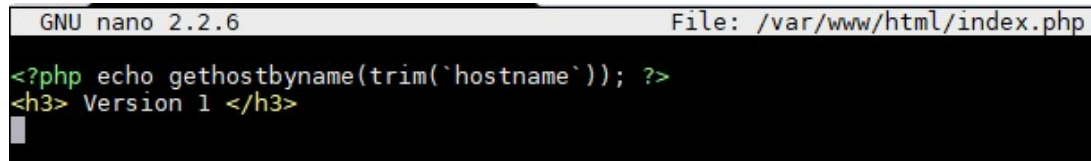
4. Validar funcionamiento, en un navegador abrimos por IP Publica para visualizar la página de Apache, debe de visualizarse similar a la página de abajo.



- Después creamos un index.php dentro del DOCUMENTROOT para validar funcionamiento e identificación de nodos en scaling.

```
# nano -w /var/www/html/index.php
```

```
<?php echo gethostname(trim(`hostname`)); ?>
<h3> Version 1 </h3>
```



```
GNU nano 2.2.6 File: /var/www/html/index.php
<?php echo gethostname(trim(`hostname`)); ?>
<h3> Version 1 </h3>
```

- Después de grabar el archivo validamos su correcto funcionamiento en el navegador.



Dentro de nuestra terminal como root realizamos un logout, en la consola de SSH, para regresar al usuario sin privilegios y poder preparar la máquina para convertirla en imagen.

```
# exit
```

- De aprovisionamos para preparar la máquina para convertirla en imagen:

```
# sudo waagent -deprovision
```

```
mpena@BSVAUTOSCBASE:~$ sudo waagent -deprovision
WARNING! The waagent service will be stopped.
WARNING! All SSH host key pairs will be deleted.
WARNING! Cached DHCP leases will be deleted.
WARNING! root password will be disabled. You will not be able to login as root.
WARNING! /etc/resolv.conf will be deleted.
Do you want to proceed (y/n)y
```

- Nos salimos de la máquina, **sin apagarla dentro del sistema operativo**, y creamos la imagen desde Azure CLI 2.0

```
# az vm deallocate --resource-group BSRGAUTOSC --name BSVAUTOSCBASE
```

```
psmapad@LTPENA:~$ az vm deallocate --resource-group BSRGAUTOSC --name BSVAUTOSCBASE
{
  "endTime": "2018-01-01T10:00:00",
  "error": null,
  "name": "BSVAUTOSCBASE",
  "startTime": "2018-01-01T10:00:00",
  "status": "Succeeded"
}
```

```
# az vm generalize --resource-group BSRGAUTOSC --name BSAUTOSCBASE
# az image create --resource-group BSRGAUTOSC --name BSAUTOSCIMG --source
BSAUTOSCBASE
```

```
psmapad@LTPENA: $ az vm generalize --resource-group BSRGAUTOSC --name BSAUTOSCBASE
psmapad@LTPENA: $ az image create --resource-group BSRGAUTOSC --name BSAUTOSCIMG --source BSAUTOSCBASE
{
  "id": "/subscriptions/2.../resourceGroups/BSRGAUTOSC/providers/Microsoft.Compute/image
s/BSAUTOSCIMG",
  "location": "westus2",
  "name": "BSAUTOSCIMG",
  "provisioningState": "Succeeded",
}
```

Despliegue de AutoScaling Open Source Software

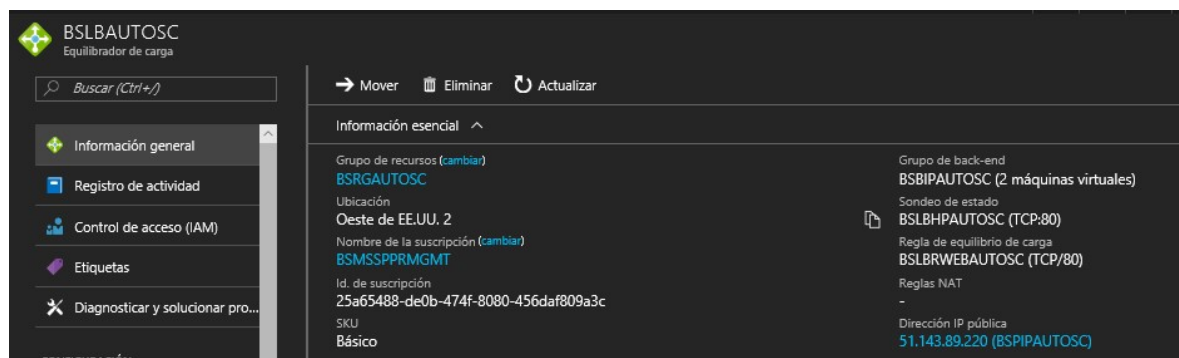
Aquí realizaremos el despliegue de un ambiente Web Open Source Software con posibilidad de autoescalamiento en Azure.

2. Crear VM Scale utilizando la imagen que creamos:

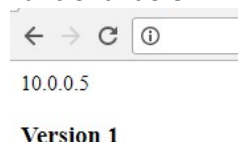
```
# az vmss create --lb BSLBAUTOSC --location westus2 --public-ip-address BSIPIAUTOSC --nsg
BSNSGAUTOSC --vnet-name BSVNETAUTOSC --subnet BSVNETINTAUTOSC --resource-group
BSRGAUTOSC --name BSSSAUTOSC --image BSAUTOSCIMG --vm-sku Standard_A1_v2 --
admin-username bsvmssadmin --admin-password '..Pa$$w0rd..'
```

```
psmapad@LTPENA: $ az vmss create --lb BSLBAUTOSC --location westus2 --public-ip-address BSIPIAUTOSC --nsg BSNSGAUTOSC --vnet-name BSVNETAUTOSC --subnet BSVNETINTAUTOSC --resource-group BSRGAUTOSC --name BSSSAUTOSC --image BSAUTOSCIMG --vm-sku Standard_A1_v2 --admin-username bsvmssadmin --admin-password '..Pa$$w0rd..' --generate-ssh-keys
{
  "vmss": {
    "overprovision": true,
    "provisioningState": "Succeeded",
    "singlePlacementGroup": true,
    "uniqueId": "{...}",
    "upgradePolicy": {
      "automaticOSUpgrade": false,
      "mode": "Manual"
    },
  },
  "virtualMachineScaleSet": {
}
```

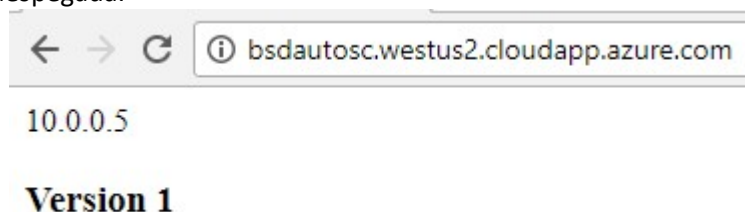
3. Una vez creado e inicializado el ambiente de scale set podemos extraer por medio de ARM los datos de conexión para los servicios web.



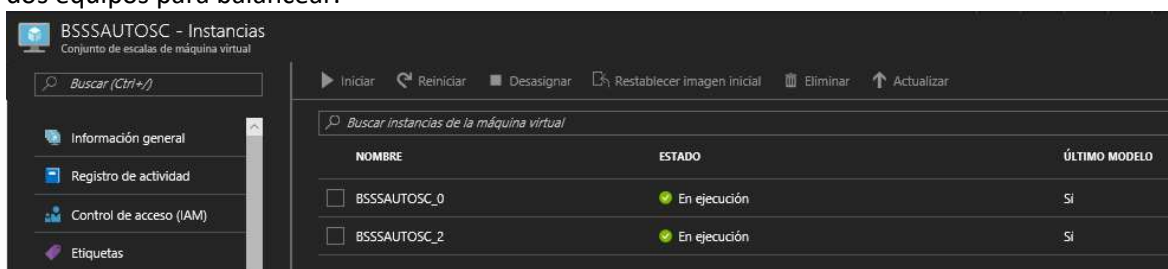
Confirmar que el Servicio Web está funcionando en VM Scale Set.



- Podemos validar el dominio creado por medio de la IP Publica, creada con anterioridad, que la pagina es despegada.



- Validamos dentro de ARM que los equipos ya se encuentran en ejecución, por defecto crea dos equipos para balancear:

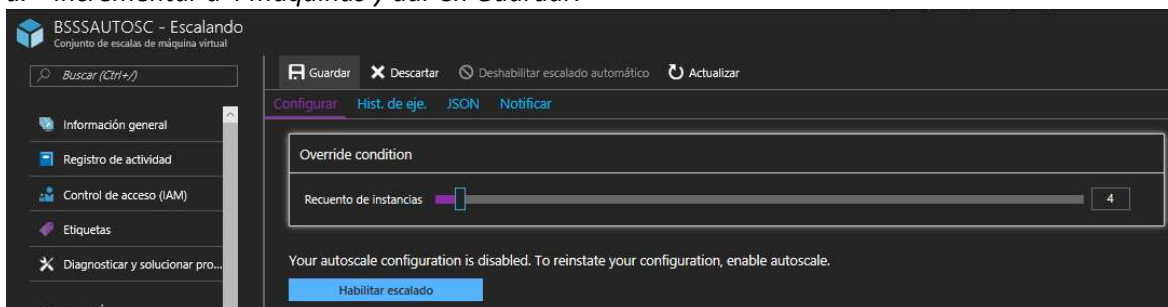


Demostración Escalamiento:

En este punto realizaremos la demo del funcionamiento de Scale Up y Scale Down de forma manual.

- Ir a Azure Resource Manager (ARM) e incrementar a dos máquinas más, para ver la distribución de carga (**Scale UP Manual**).

a. Incrementar a 4 máquinas y dar en Guardar.



- b. Validar el estado del despliegue de las VMs. (Normalmente crea un equipo adicional el cual después desaparece cuando se termina el Scale UP)



c. Validar que ya están en ejecución las 2 máquinas adicionales que solicitamos.

▶ Iniciar ↺ Reiniciar ■ Desasignar 🔄 Restablecer imagen inicial 🗑 Eliminar ⬆ Actualizar		
🔍 <i>Buscar instancias de la máquina virtual</i>		
NOMBRE	ESTADO	ÚLTIMO MODELO
<input type="checkbox"/> BSSSAUTOSC_0	✓ En ejecución	Si
<input type="checkbox"/> BSSSAUTOSC_2	✓ En ejecución	Si
<input type="checkbox"/> BSSSAUTOSC_5	✓ En ejecución	Si
<input type="checkbox"/> BSSSAUTOSC_6	✓ En ejecución	Si

2. Regresamos al navegador para validar que nos conteste otro equipo.

Eliminar Ambiente

Al terminar la ejecución del laboratorio es importante eliminar todos los recursos, hay que recordar que parte de los beneficios de la nube es que se paga por el tiempo que se usa.

```
# az group delete --name BSRGAUTOSC
```

Conclusiones

En este laboratorio se incluye las actividades para cubrir los requerimientos de Microsoft. En esto se incluye:

- Microsoft Loves Linux.
- Integración de Linux en Windows 10.
- Creación y despliegue de Imágenes Open Source Software en Azure por medio de Azure CLI 2.0.
- Automatización en Azure por medio de Azure CLI 2.0.
- Creación de ambiente web Open Source Software por medio de Azure CLI 2.0.
- Adicional se comprobó la seguridad a nivel comunicación en Azure.

Contacto

