



# Deploy and scale Microsoft Azure infrastructures and applications with Red Hat Ansible Automation

Stuart R. Kirk, RHCA  
Azure Global Black Belt / Open Source  
[stuart.kirk@microsoft.com](mailto:stuart.kirk@microsoft.com)

Zim Kalinowski  
Senior Software Engineer  
[zikalino@microsoft.com](mailto:zikalino@microsoft.com)

Wednesday May 8<sup>th</sup>, 2019

# STUART KIRK

Azure Global Black Belt / Open Source  
Microsoft Corporation

- Open Source Technology Solutions Professional
- Home: Ann Arbor, Michigan, Originally from Canada
- Entirety of Career in Open Source: Dell, Cisco, Red Hat
- Joined Microsoft in 2016; First RHCA at Microsoft
- Twitter: @StuartAtMSFT
- GitHub: <https://github.com/stuartatmicrosoft>
- LinkedIn: <https://www.linkedin.com/in/stuartkirk>



# ZIM KALINOWSKI

Senior Software Engineer  
Microsoft Corporation


- Lives in Shanghai, China.
- Currently making Ansible the ultimate tool to deploy resources on Azure. Passionate about Azure REST API.
- Before joining Microsoft worked for Nokia, Opera Software, Qualcomm and created own startup.
- Also likes American craft beer and all kinds of BBQ
- LinkedIn:  
<https://www.linkedin.com/in/smarterphone/>



# CONNECTING

# OBTAINING YOUR LAB DESKTOP

- Your lab VM is provided using a content delivery environment from Spektra Systems
- To obtain your Linux desktop you will need to register on the Spektra website:



## Deploy & scale Microsoft Azure infrastructures and applications with Red Hat Ansible Automation

By : Microsoft

Immerse yourself in the world of **Microsoft Azure** and **Ansible by Red Hat**. Explore the vast Ansible module landscape for Azure and obtain hands-on experience deploying IaaS and PaaS infrastructures to Azure entirely using Ansible Playbooks. You will use example playbooks that provide the building blocks to extend your existing enterprise Ansible deployment to Azure and taking advantage of both infrastructure and platform services. You'll perform the following activities entirely from Ansible Playbooks:

- Connect an Ansible infrastructure to Azure.
- Deploy a Red Hat Enterprise Linux virtual machine in Azure using the Azure Image Gallery / Azure Marketplace.
- Create and configure an Azure managed MySQL Database.
- Install and configure an application for the Red Hat Enterprise Linux IaaS virtual machine and the MySQL PaaS database resources.

### Register Now

First Name\*

Last Name\*

Work email\*

Organization\*

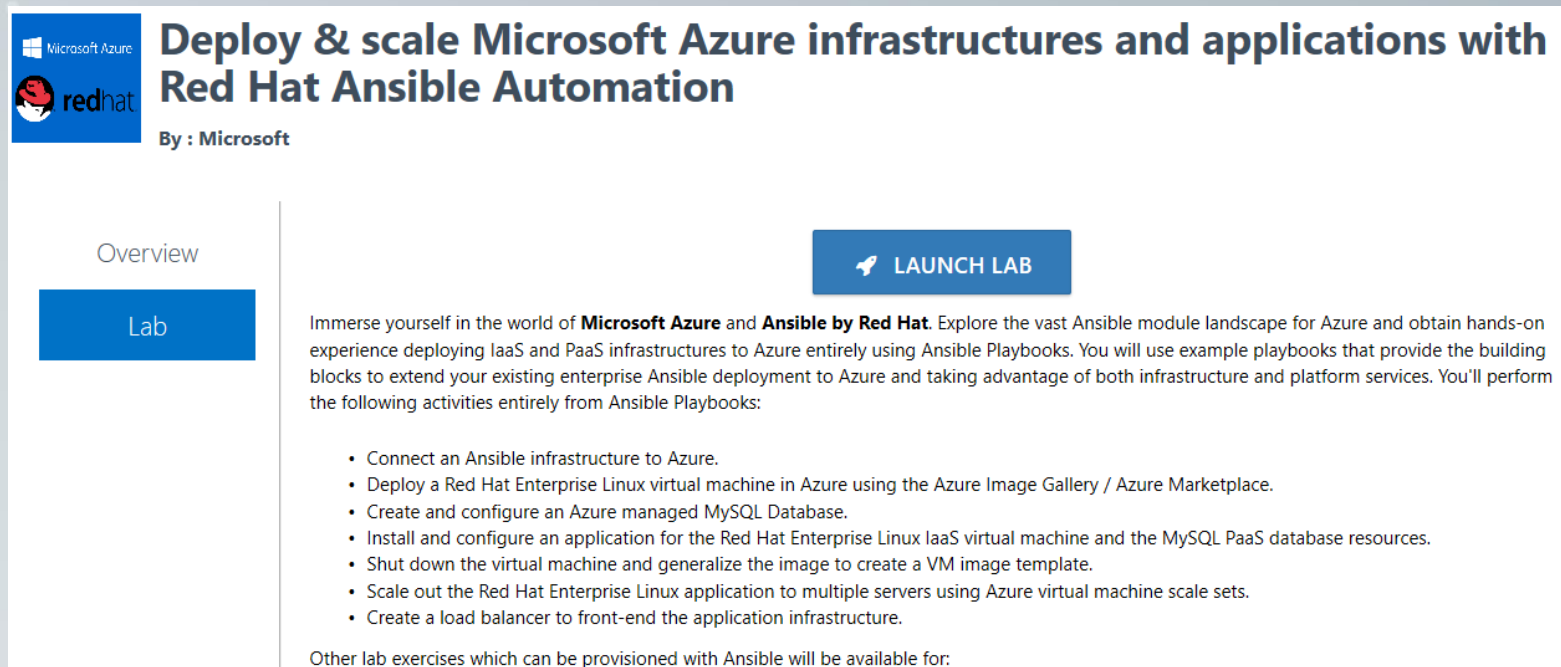
Country \* ▾

Microsoft or training providers may use your contact information to provide updates and special offers about Microsoft Azure and other Microsoft products and services. You can unsubscribe at any time. To learn more you can read the [Privacy Policy](#).

SUBMIT

# OBTAINING YOUR LAB DESKTOP

- The login credential information for your lab will be available immediately
- Click the “LAUNCH LAB” button as shown:



**Deploy & scale Microsoft Azure infrastructures and applications with Red Hat Ansible Automation**

By : Microsoft

Overview

Lab

**LAUNCH LAB**

Immerse yourself in the world of **Microsoft Azure** and **Ansible by Red Hat**. Explore the vast Ansible module landscape for Azure and obtain hands-on experience deploying IaaS and PaaS infrastructures to Azure entirely using Ansible Playbooks. You will use example playbooks that provide the building blocks to extend your existing enterprise Ansible deployment to Azure and taking advantage of both infrastructure and platform services. You'll perform the following activities entirely from Ansible Playbooks:

- Connect an Ansible infrastructure to Azure.
- Deploy a Red Hat Enterprise Linux virtual machine in Azure using the Azure Image Gallery / Azure Marketplace.
- Create and configure an Azure managed MySQL Database.
- Install and configure an application for the Red Hat Enterprise Linux IaaS virtual machine and the MySQL PaaS database resources.
- Shut down the virtual machine and generalize the image to create a VM image template.
- Scale out the Red Hat Enterprise Linux application to multiple servers using Azure virtual machine scale sets.
- Create a load balancer to front-end the application infrastructure.

Other lab exercises which can be provisioned with Ansible will be available for:



# OBTAINING YOUR LAB DESKTOP

- Access to the lab VM is via noVNC (HTML5 Web-Based VNC)
- You will need the following:  
**VNCSERVERURL**  
**PASSWORD**  
**RESOURCE GROUP**
- Connect to VNCSERVERURL via http / https
- All VNC & account passwords are **Microsoft**
- All other data is available on the lab VM

✓ Your On Demand Lab is ready (23 hour(s), 59 minute(s) remaining)

Environment Details Virtual Machines

**Azure Credentials**

Here are your credentials to login to Microsoft Azure and access the On Demand Lab

Username

Password

**Service Principal Details**

Application/Client Id

Application Display Name

Application Secret Key

Subscription Id

Tenant Id

Tenant Domain Name

**Environment Details**

**Resource Group : 01-62317**

VNCSERVERURL

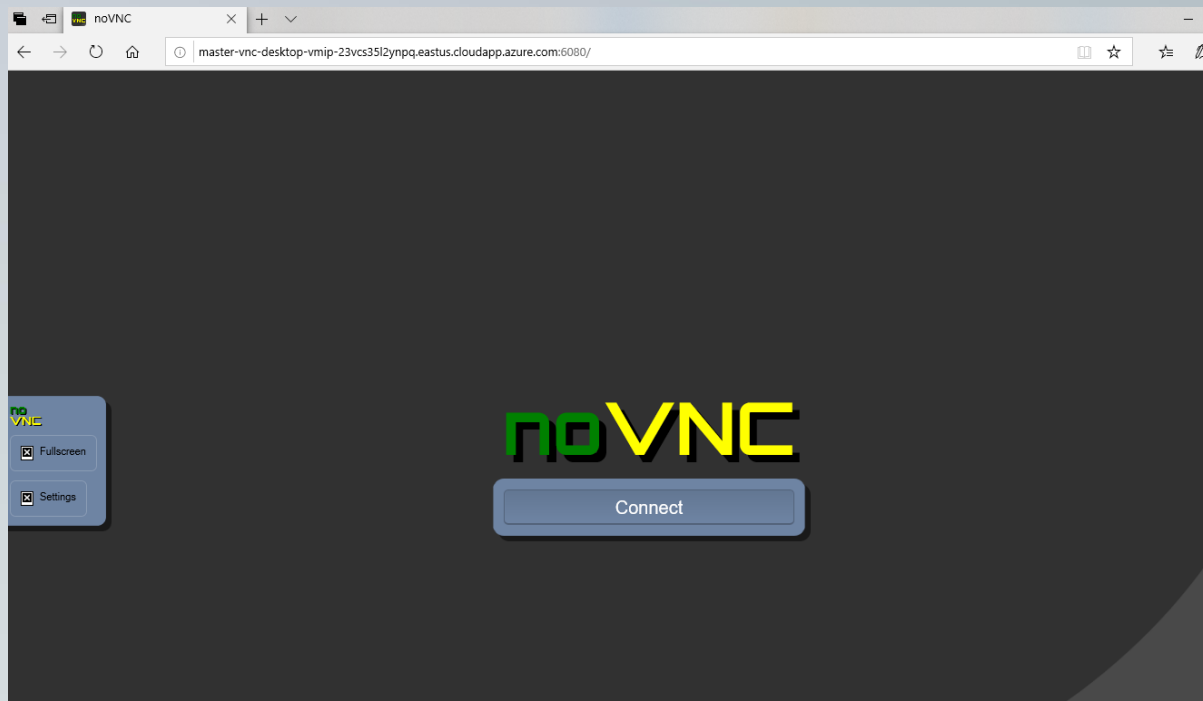
PASSWORD

Lab Guide : <https://github.com/stuartatmicrosoft/RedHatSummit2019>

Help Document URL : <https://github.com/stuartatmicrosoft/RedHatSummit2019>

# OBTAINING YOUR LAB DESKTOP

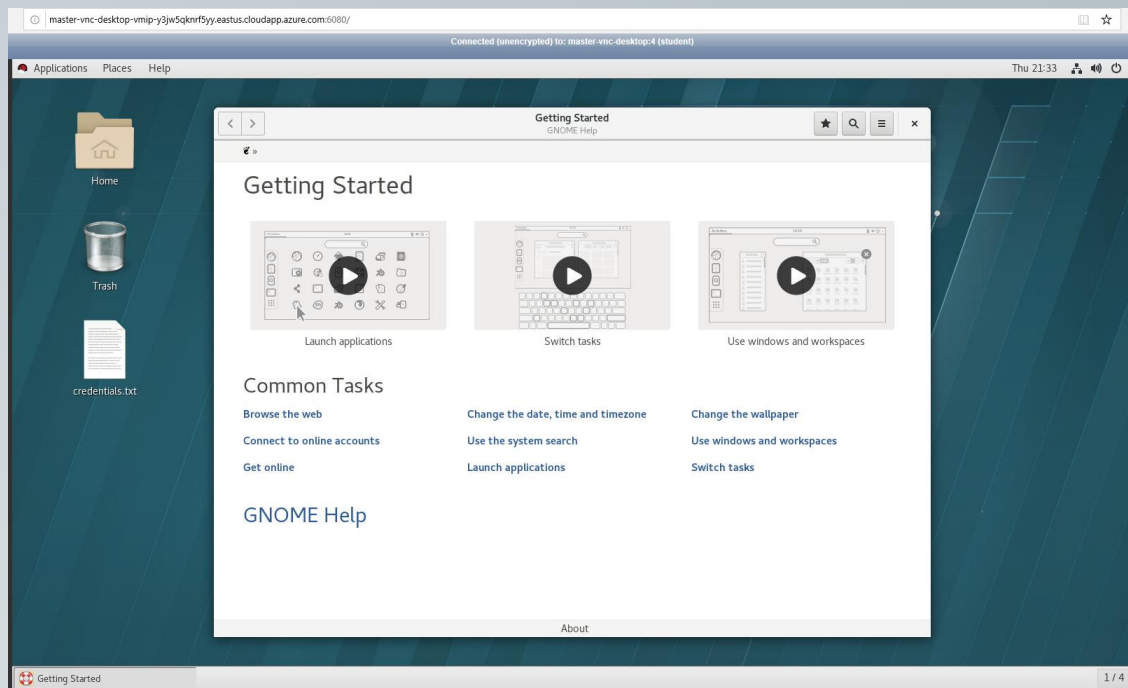
- Your VNCSERVERURL should produce a login page similar to the illustration below
- All VNC & account passwords are **Microsoft**





# OBTAINING YOUR LAB DESKTOP

- Your lab environment will be delivered from a RHEL 7.6, X-Windows based host
- **All credential information is stored in the credentials.txt file**



# QUICK START

- Add service principal data from `credentials.txt` into `~/.bashrc` then `source .bashrc`
- As root, install the Azure modules for Ansible:

```
pip install ansible[azure]
```

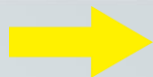
- Clone the RedHatSummit 2019 Repository:

```
git clone https://github.com/stuartatmicrosoft/RedHatSummit2019
```

- Edit the “vars.yml” file in the “playbooks” directory to provide values for:

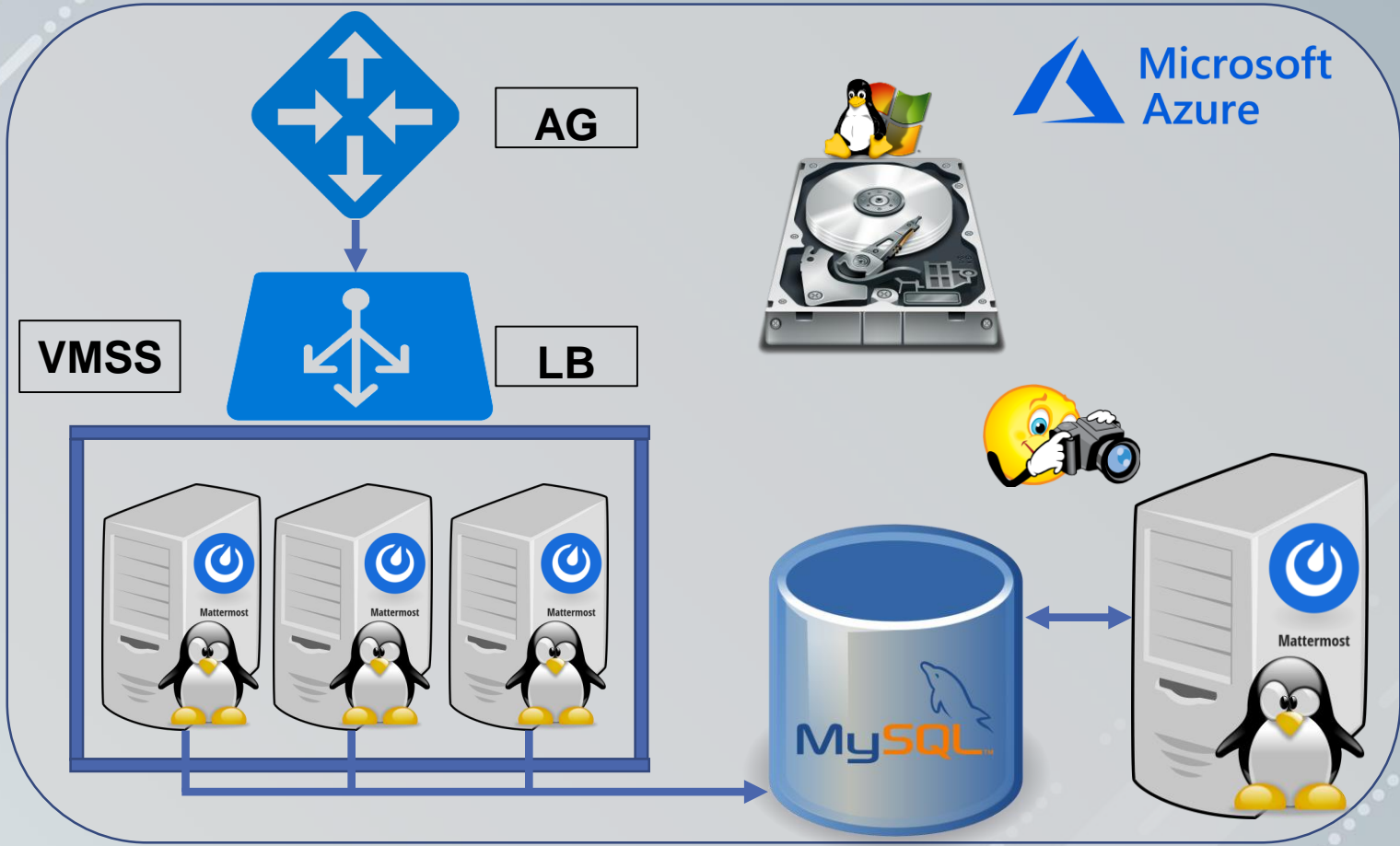
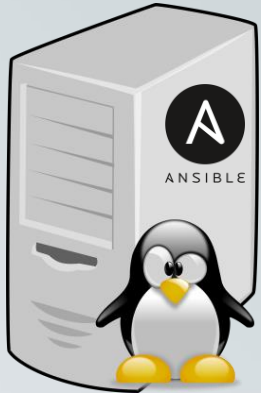
<code>resource_group</code>	The resource group you were assigned
<code>vm_name</code>	A prefix for naming your Azure resources
<code>location</code>	The name of your Azure data center (eastus / northcentralus)
<code>github_id</code>	Your Github ID
<code>github_token</code>	A personal access token for your Github account

- Run the `./randomize.sh` script to create your personal variables file `vars-myvars.yml`



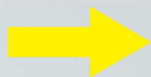
# Mattermost Overview

- 00-prerequisites
- 01-mm-vm-deploy
- 02-create-mysql
- 03-mm-setup
- 04-create-vm-image
- 05-vmss-create
- 06-appgateway-create



# TIPS

- Log in to the Azure Linux CLI and set your default output to “TABLE” format using the `az configure` command
- It is recommended that you stay ENTIRELY within the noVNC environment
- If you need to access a web browser, use the Firefox web browser already loaded on your lab VM host
- Copying and pasting content in/out of the lab VM is very cumbersome
- Visit Zim & I at the Microsoft booth after this session ends
- Do not forget to pick up some swag at the front of the room!
- **DO NOT DELETE** any assets in your resource group beginning with **master-vnc** – or you will lose your Linux VM and your lab will be over!
- HAVE FUN!



# LAB CONTENT OVERVIEW

# LAB OVERVIEW – IaaS / PaaS / Application Deploy

**Summary:** Deploy a highly available application using IaaS / PaaS / Auto Scaling

## Lab Goals:

- Configuring Infrastructure and Platform services on Azure with applications
- Creating a disk image to scale the infrastructure node
- Using virtual machine scale sets and Azure application gateway for resiliency

## Playbooks:

- mm-00-prerequisites.yml (16m25s)
- mm-01-vm-deploy.yml (3m24s)
- mm-02-create-mysql.yml (3m10s)
- mm-03-setup-mattermost.yml (1m44s)
- mm-04-create-vm-image.yml (1m11s)
- mm-05-vmss-create.yml (3m37s)
- mm-06-appgateway-attach.yml (6m2s)

# LAB OVERVIEW – High-Performance Computing

**Summary:** Create an Infiniband enabled HPC cluster and test intra-VM latency

## Lab Goals:

- Create a master node to serve as an NFS master for the HPC compute nodes
- Create three compute nodes utilizing Azure's Infiniband VM interconnects
- Measure the latency between TCP and DAPL-based connections using Intel MPI

## Playbooks:

- hpc-00-cluster-master-deploy.yml (2m59s)
- hpc-01-cluster-master-configure.yml (4m4s)
- hpc-02-cluster-compute-deploy.yml (8m44s)
- hpc-03-cluster-compute-configure.yml (3m34s)



# LAB OVERVIEW – Big Data (HDInsight)

**Summary:** Deploy a platform based Big Data solution & Perform various queries

## Lab Goals:

- Create an HDInsight Cluster
- Explore the resulting configuration
- Perform several manual queries on the sample data sets

## Playbooks:

- `hdinsight-36-create-hdinsight.yml` (13m58s)
- OR
- `hdinsight-40-create-hdinsight.yml` (18m38s)

# LAB OVERVIEW – Azure Kubernetes Service (AKS)

**Summary:** Deploy a superhero voting application using an AKS Kubernetes cluster

## Lab Goals:

- Create a managed AKS instance in Azure
- Obtain Kubernetes credentials from Azure & merge into ~/.kube
- Create a new CosmosDB account and database
- Create a Kubernetes secret to connect the voting application to CosmosDB
- Deploy the voting application definition file to the AKS Kubernetes cluster
- Wait for the load balancer to be created and test the application

## Playbooks:

- aks-00-create-aks-cluster.yml (17m47s)
- aks-01-create-cosmosdb.yml (3m47s)

## Kubernetes definition file:

- aks-k8s-hero-voting-deploy.yml

# LAB OVERVIEW – Azure Functions (Serverless)

**Summary:** Run Ansible as a container based function app on Azure. Create entire serverless pipeline triggered from GitHub, including static web page to test it using Azure Storage.

## Lab Goals:

- Create function app Docker image in Azure Container Registry using ACR task
- Create function app using image from ACR
- Create a static web page utilizing function app using Azure Storage

## Playbooks:

- fa-00-create-image.yml
- fa-01-create-function-app-from-acr.yml
- fa-02-create-website.yml

# LAB OVERVIEW – Azure Web Apps / NoSQL

**Summary:** Modernize a local infrastructure application to a platform-based service

## Lab Goals:

- Containerize a NodeJS Application running locally on your desktop
- Use Azure Container Registry to store the container
- Create an Azure CosmosDB (MongoDB)
- Export data from a local MongoDB and import into Azure CosmosDB
- Create an Azure Web Application to consume the containerized application in ACR

## Playbooks:

- todo-00-create-acr.yml (0m9s)
- todo-01-create-cosmosdb.yml (4m45s)
- todo-02-create-appservice-plan.yml (0m11s)
- todo-03-create-azure-webapp.yml (0m30s)

# REPLICATING ENVIRONMENT

# REPLICATE THE ENVIRONMENT

Deploy the same Linux desktop environment in your own Azure subscription:

- `az group create -n 'rg_name' -l 'eastus'` (Pick your closest Azure Data Center)
- `az group deployment create --template-uri  
https://raw.githubusercontent.com/stuartatmicrosoft/RedHatSummit2019/master/provision-scripts/deploy-01.json --parameters  
https://raw.githubusercontent.com/stuartatmicrosoft/RedHatSummit2019/master/provision-scripts/deploy-01.parameters.json -g rg_name`
- `az vm list -d | grep master-vnc | awk '{print $6}'`
- Connect to this FQDN via http or https on port 6080

# REPLICATE THE ENVIRONMENT

Create a service principal for your Ansible host to connect to Azure:

- `az account list` (Obtain SubscriptionID for your Azure Account)
- `az ad sp create-for-rbac --name="http://`whoami`-ansible" --role="Contributor" --scopes="/subscriptions/YOUR_SUBSCRIPTION_ID"`

Set environment variables in `~/.bashrc`:

- `AZURE_SUBSCRIPTION_ID` = SubscriptionId
- `AZURE_TENANT` = Tenant
- `AZURE_CLIENT_ID` = AppId
- `AZURE_SECRET` = Password



**RED HAT  
SUMMIT**

THANK YOU



[linkedin.com/company/Red-Hat](https://www.linkedin.com/company/Red-Hat)



[youtube.com/user/RedHatVideos](https://www.youtube.com/user/RedHatVideos)



[facebook.com/RedHatinc](https://www.facebook.com/RedHatinc)



[twitter.com/RedHat](https://twitter.com/RedHat)

**Please don't  
forget to  
complete the  
survey!**



**<https://aka.ms/RedHatSummit>**