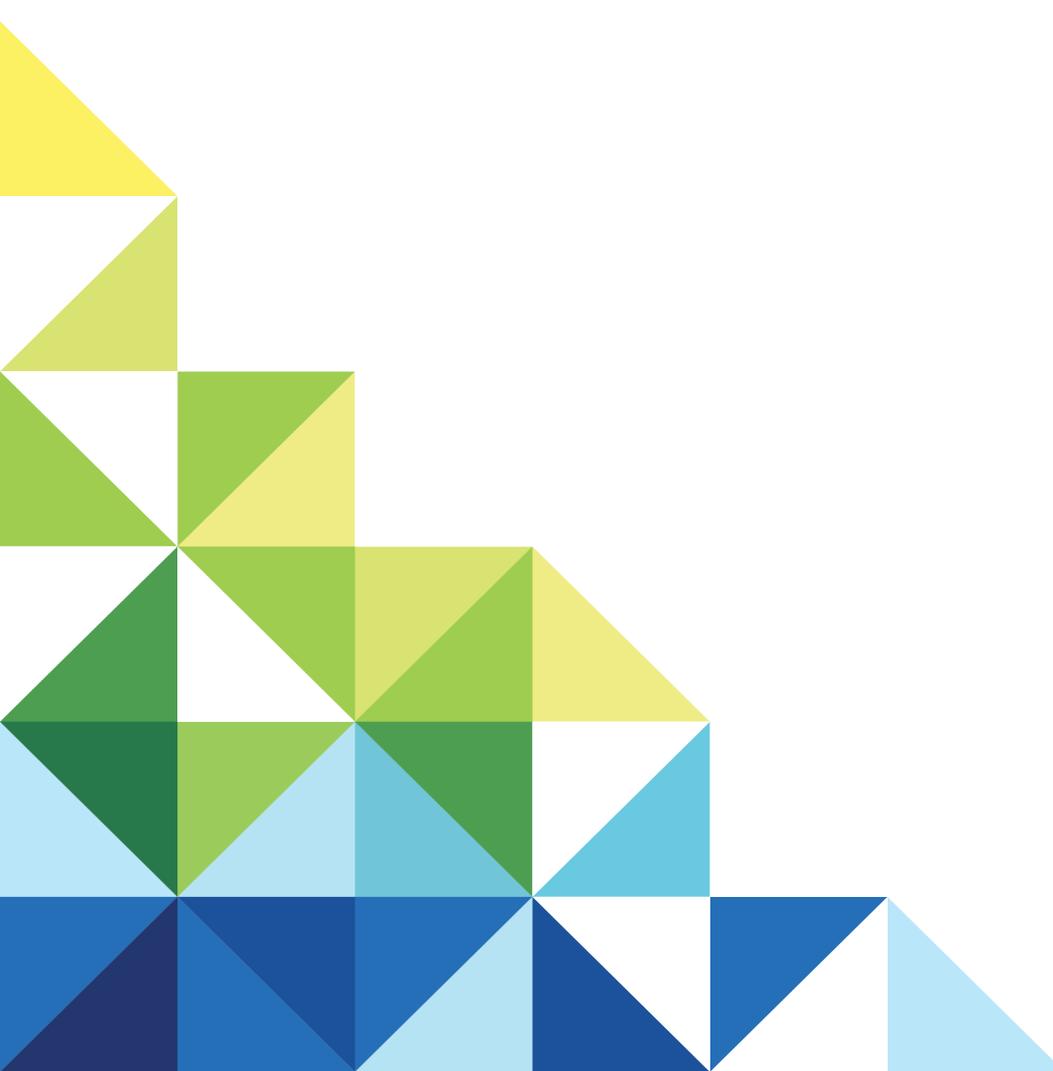


Unified Access Gateway PowerShell Deployment to Amazon Web Services

Technical Note

02 JULY 2019

Unified Access Gateway 3.6



vmware®

You can find the most up-to-date technical documentation on the VMware website at:

<https://docs.vmware.com/>

If you have comments about this documentation, submit your feedback to

docfeedback@vmware.com

VMware, Inc.
3401 Hillview Ave.
Palo Alto, CA 94304
www.vmware.com

Copyright © 2019 VMware, Inc. All rights reserved. [Copyright and trademark information.](#)

Contents

- 1** Introduction 4
- 2** Prepare the Windows Client Machine for Powershell 5
- 3** Prepare the AWS EC2 Environment 6
- 4** Uploading the Unified Access Gateway Image with PowerShell 9
- 5** Prepare an INI File 11
- 6** Run the Unified Access Gateway Deploy Command uagdeployec2.ps1 15

Introduction

This technical note describes the use of PowerShell command to deploy Unified Access Gateway 3.5 or later to Amazon Web Services Elastic Compute Cloud (EC2). It describes the steps needed to prepare the EC2 environment before creating any Unified Access Gateway instances. It also provides the details of the .INI file containing the configuration settings and shows how to run the deployment PowerShell command. These are general guidelines. For more detailed information, see *Amazon AWS Documentation*.

Prepare the Windows Client Machine for Powershell

2

Prepare your Windows client for Powershell deployment

Prerequisites

Ensure that you are running this from Windows 10 machine with access to the internet.

Note Other Windows operating systems may also be supported but these instructions are for Windows 10.

Procedure

- 1 Open the Powershell command window with administrative rights.
- 2 Run the command

```
Install-Module -Name AWSPowerShell -Force  
Install-Package 7Zip4PowerShell
```

What to do next

Prepare the Amazon AWS EC2 environment.

Prepare the AWS EC2 Environment

3

Please refer to the official Amazon AWS PowerShell documentation for full details of the steps outlined in this section.

Prerequisites

Create an Amazon AWS account if you don't already have one.

Procedure

- 1 In the AWS Console, create an Access Key and obtain the Access Key ID and Secret Access Key. Set them in the default profile.

This step is applicable only if you don't have an access key ID and Secret Access Key

```
Set-AWSCredential -AccessKey AKIAI6428NKYOEXAMPLE `
-SecretKeybvfhkvvfhsbvhsdbhfbvfhfhvfhdskvbhfvbhfhEXAMPLE `
-StoreAs default
```

- 2 Create a bucket in Amazon S3 to store Unified Access Gateway .vmdk images if one doesn't already exist.

```
$bucket="uag-images"
New-S3Bucket -BucketName $bucket -Region us-east-2
```

- 3 Create an IAM role in Amazon AWS called `vmimport` and apply a policy to the role.

```
$importPolicyDocument = @"
{
  "Version":"2012-10-17",
  "Statement":[
    {
      "Sid": "",
      "Effect": "Allow",
      "Principal": {
        "Service": "vmie.amazonaws.com"
      },
      "Action": "sts:AssumeRole",
      "Condition": {
        "StringEquals": {
          "sts:ExternalId": "vmimport"
        }
      }
    }
  ]
}
```

```

]
}
"@

New-IAMRole -RoleName vmimport -AssumeRolePolicyDocument $importPolicyDocument

$bucket="uag-images"
$rolePolicyDocument = @"
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Effect": "Allow",
      "Action": [
        "s3:GetBucketLocation",
        "s3:GetObject",
        "s3:ListBucket"
      ],
      "Resource": [
        "arn:aws:s3:::$bucket",
        "arn:aws:s3:::$bucket/*"
      ]
    },
    {
      "Effect": "Allow",
      "Action": [
        "ec2:ModifySnapshotAttribute",
        "ec2:CopySnapshot",
        "ec2:RegisterImage",
        "ec2:Describe*"
      ],
      "Resource": "*"
    }
  ]
}
"@

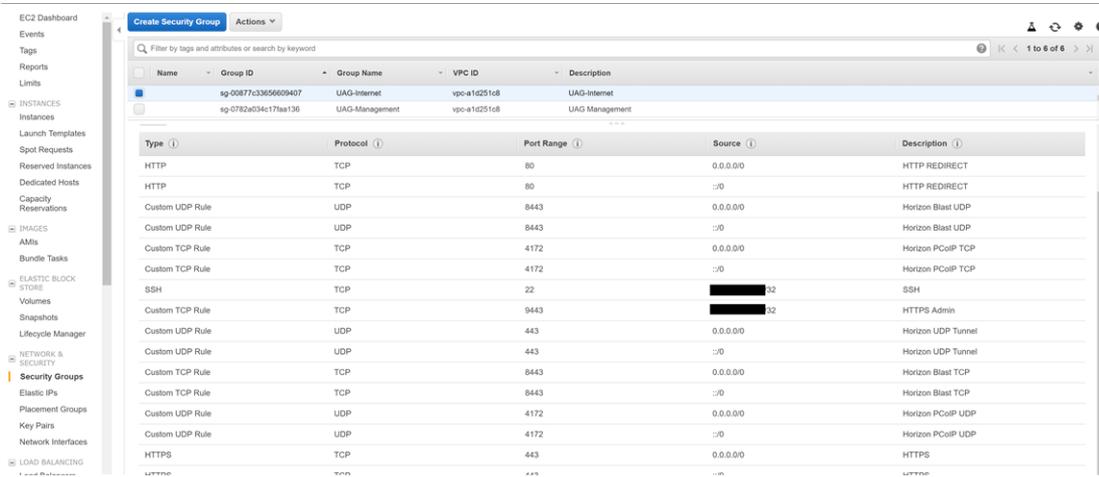
Write-IAMRolePolicy -RoleName vmimport -PolicyName vmimport -PolicyDocument $rolePolicyDocument

```

- 4 Prepare the network environment in EC2. These steps can be performed from the EC2 Management Console or with PowerShell. They just need to be done once to prepare the EC2 environment for Unified Access Gateway deployments. For this, at least one subnet is needed. For multi NIC Unified Access Gateway deployments, each NIC can either be on the same subnet or on different subnets.
- 5 Create a Security Group for each type of NIC.

A security group contains a set of firewall rules to restrict TCP and UDP port access. A security group can be shared among multiple Unified Access Gateway appliances. For example you can create a security group called **UAG-Internet** for eth0 and associate with the first NIC automatically when the Unified Access Gateway appliance is created. For Horizon use, the first (UAG-Internet) could allow TCP ports 80, 443, 8443, 4172 and UDP ports 443, 8443, 4172 from any client. If you want to allow

ssh access to Unified Access Gateway then you must specify `sshEnabled=true` in the General section of each `.ini` file. SSH should generally only be enabled for testing purposes and not for a production deployment. You should also make sure that access to ssh on TCP port 22 is restricted in the security group to individual source IP addresses so that it is not open to all.



- 6 If the Unified Access Gateway appliance is directly accessible from the Internet, then each NIC requiring access must also have an associated public IP address known as Elastic IPs.
- 7 For each NIC, determine the Subnet ID, the Security Group ID and the Public IP Allocation ID. If you do not specify a Security Group ID for any NIC then the default Security Group will be used. If you don't specify a Public IP ID then there won't be a public IP address for that NIC and it won't be directly accessible from the Internet. This may be the case if a load balancer is used in front of a group of Unified Access Gateway appliances.

Uploading the Unified Access Gateway Image with PowerShell

4

You can upload the Unified Access Gateway image with PowerShell. The image can be imported and registered to other regions as well if required.

Procedure

- 1 Download the Unified Access Gateway .ova image file from VMware. The version of this file must be 3.5 or later.
- 2 Extract the .vmdk image from the .ova file.

```
expand-7zip C:\uag\euc-unified-access-gateway-x.y.0.0-12345678_OVF10.ova C:\uag\
```

- 3 Upload the .vmdk image into the S3 bucket

```
$vmdkImage="euc-unified-access-gateway-x.y.0.0-12345678-system.vmdk"
$bucket="uag-images"
$region="us-east-2"

$params = @{
    "BucketName"=$bucket
    "File"="C:\uag\"+$vmdkImage
    "key"="/"+$vmdkImage
    "Region"="us-east-2"
}
Write-S3Object @params
```

- 4 Import the EC2 snapshot

```
$params = @{
    "DiskContainer_Format"="VMDK"
    "DiskContainer_S3Bucket"=$bucket
    "DiskContainer_S3Key"=$vmdkImage
    "Region"=$region
}
$impId=Import-EC2Snapshot @params
```

- 5 To track the import, periodically run the following command to obtain progress status.

Note The import will take several minutes.

```
(Get-EC2ImportSnapshotTask -ImportTaskId `
$impId.ImportTaskId).SnapshotTaskDetail
```

- 6 Once complete, the following command should show the SnapshotId

```
(Get-EC2ImportSnapshotTask -ImportTaskId `
$impId.ImportTaskId).SnapshotTaskDetail.SnapshotId
```

- 7 Register the Image as an Amazon Machine Image (AMI)

```
$bdm=New-Object Amazon.EC2.Model.BlockDeviceMapping
$bd=New-Object Amazon.EC2.Model.EbsBlockDevice
$bd.SnapshotId=(Get-EC2ImportSnapshotTask `
-ImportTaskId $impId.ImportTaskId).SnapshotTaskDetail.SnapshotId
$bd.DeleteOnTermination=$true
$bdm.DeviceName="/dev/sda1"
$bdm.Ebs=$bd
$params = @{
"BlockDeviceMapping"=$bdm
"RootDeviceName"="/dev/sda1"
"Name"=$vmdkImage
"Architecture"="x86_64"
"VirtualizationType"="hvm"
}
Register-EC2Image @params
```

In AWS Console you should see your imported image in EC2 AMI Images.

Prepare an INI File

Most sections of the INI file are identical to the standard INI settings for Unified Access Gateway as supported for vSphere, Hyper-V and Azure deployments.

Refer to <https://communities.vmware.com/docs/DOC-30835>.

- Create an Amazon AWS account if you do not have one.
- Create an access key and obtain the values of the Access Key ID and Secret Access Key. See <https://docs.aws.amazon.com/powershell/latest/userguide/pstools-appendix-sign-up.html>
- For security reasons, the .INI file will not contain the Access Key ID or Secret Access Key so they must be stored in a named or default profile. These AWS credentials are used to cryptographically sign the corresponding web service requests used by the PowerShell script. They should be stored in a named profile which is then referenced from the .INI file. See <https://docs.aws.amazon.com/powershell/latest/userguide/specifying-your-aws-credentials.html>. Use the following PowerShell example command to store these values in a profile named `awsCredentialProfile`:

```
Set-AWSCredential-AccessKey AKIAIOSFODNN7EXAMPLE `
-SecretKeyJa1rXUtnFEMI/K7MDENG/bPxrFiCYEXAMPLEKEY `
-StoreAsawsCredentialProfile
```

For AWS EC2 deployments, the following settings in the General section are not used.

- `diskMode`
- `ds`
- `folder`
- `netInternet`
- `netManagementNetwork`
- `netmask0`
- `netmask1`
- `netmask2`
- `netBackendNetwork`
- `source`
- `target`
- All of the IPv4 settings

- All of the IPv6 settings

For AWS EC2 there is a new group called AmazonEC2 that contains all of the settings specific to AWS EC2.

Table 5-1. Settings specific to AWS EC2

Group	Value	Example	Description
AmazonEC2	amiId	amiId=ami-1986bb7c	The ID of the registered Amazon Machine Image (AMI). This represents the Unified Access Gateway appliance image uploaded to Amazon S3. Note This is a mandatory setting.
	credentialProfileName	credentialProfileName=MyUAGProfile	The name of the credential profile containing the Access Key ID and Secret Access Key. This must be setup first. See https://docs.aws.amazon.com/powershell/latest/userguide/specifying-your-aws-credentials.html . If this is not set, the deployment will attempt to use the default credential profile.
	instanceType	instanceType=c4.large	AWS EC2 instance type. Default is c4.large.
	region	region=us-east-2	The AWS EC2 region name. Note This is a mandatory setting.
	privateIPAddress0 privateIPAddress1 privateIPAddress2	privateIPAddress1=172.31.7.222	Optional fixed IP address used by EC2 DHCP for eth0, eth1, or eth2. Normally this is not required but can be used to set a static private IP address instead of a dynamic one.
	publicIPId0 publicIPId1 publicIPId2	publicIPId0=eipalloc-027afa45f34984c87	AWS EC2 Elastic Public IP address ID associated with eth0, eth1 or eth2. This setting is optional for each NIC.

Group	Value	Example	Description
	securityGroupId0 securityGroupId1 securityGroupId2	securityGroupId0=sg-00877c33656609407	<p>AWS EC2 Security Group ID associated with eth0, eth1, or eth2. The same Security Group can be used by multiple Unified Access Gateway instances.</p> <p>Note This setting is optional.</p> <p>If this setting is not specified, the default EC2 Security Group will be used.</p>
	subnetId0 subnetId1 subnetId2	subnetId1=subnet-5c980935	<p>AWS EC2 Subnet ID associated with eth0, eth1 or eth2.</p> <ul style="list-style-type: none"> ■ For one NIC subnetId0 is mandatory. ■ For two NIC subnetId0 and subnetId2 are mandatory. ■ For three NIC subnetId0, subnetId1 and subnetId2 are mandatory.

INI File Definition Example

```
[General]
name=UAG12
deploymentOption=twnic
licenseEdition=Advanced
honorCipherOrder=true

[AmazonEC2]

# authentication

credentialProfileName=awsCredentialProfile

# type, region and image

instanceType=c4.large
region=us-east-2
amiId=ami-1986bb7c

# eth0 settings
subnetId0=subnet-5c980935
securityGroupId0=sg-00877c33656609407
```

```
publicIPId0=eipalloc-027afa45f34984c87
```

```
# eth1 settings
```

```
subnetId1=subnet-5c980935
```

The screenshot displays the AWS Management Console interface for an EC2 instance named 'UAG3'. The instance is in a 'running' state. The console shows various details including instance type (o4.large), availability zone (us-east-2a), and public DNS (ec2-13-59-224-71.us-east-2.compute.amazonaws.com). The 'Description' tab is selected, showing a comprehensive list of instance attributes such as AMI ID, platform, key pair name, and network interfaces.

Category	Attribute	Value
Instance	Instance ID	i-0bd1f0a6592cc8f9
Instance	Instance state	running
Instance	Instance type	o4.large
Instance	Elastic IPs	13.59.224.71*
Instance	Availability zone	us-east-2a
Security	Security groups	UAG-Internet, view inbound rules, view outbound rules
Scheduled events	Scheduled events	No scheduled events
AMI	AMI ID	ami-1986bb07c
Platform	Platform	-
IMM	IMM role	-
Key pair	Key pair name	-
Owner	Owner	189704117342
Launch time	Launch time	November 5, 2018 at 9:00:15 AM UTC (less than one hour)
Termination protection	Termination protection	False
Lifecycle	Lifecycle	normal
Monitoring	Monitoring	basic
Alarm status	Alarm status	None
Kernel ID	Kernel ID	-
Capacity Reservation	Capacity Reservation	-
Public DNS (IPv4)	Public DNS (IPv4)	ec2-13-59-224-71.us-east-2.compute.amazonaws.com
IPv4 Public IP	IPv4 Public IP	13.59.224.71
IPv6 IPs	IPv6 IPs	-
Private DNS	Private DNS	ip-172-31-9-65.us-east-2.compute.internal
Private IPs	Private IPs	172.31.9.65
Secondary private IPs	Secondary private IPs	-
VPC ID	VPC ID	vpc-a1d251c8
Subnet ID	Subnet ID	subnet-5c980935
Network interfaces	Network interfaces	eth0
Source/dest. check	Source/dest. check	True
T2/T3 Unlimited	T2/T3 Unlimited	-
EBS-optimized	EBS-optimized	False
Root device type	Root device type	efs
Root device	Root device	/dev/vda1
Stock devices	Stock devices	/dev/vda1
Elastic CPU	Elastic CPU	-
Elastic GPU type	Elastic GPU type	-
Elastic GPU status	Elastic GPU status	-
RAM disk ID	RAM disk ID	-
Placement group	Placement group	-

The screenshot displays the AWS Management Console interface for a Network Interface (eni-0c06d8a3369ca57f8). The interface is in an 'in-use' state. The console shows details such as network interface ID, VPC ID (vpc-a1d251c8), MAC address (02:8b:99:5f:a5:6c), security groups (UAG-Internet), and private DNS (ip-172-31-9-65.us-east-2.compute.internal). The 'Details' tab is selected, providing a comprehensive overview of the network interface's configuration and associated resources.

Category	Attribute	Value
Network Interface	Network interface ID	eni-0c06d8a3369ca57f8
VPC ID	VPC ID	vpc-a1d251c8
MAC address	MAC address	02:8b:99:5f:a5:6c
Security groups	Security groups	UAG-Internet, view inbound rules, view outbound rules
Status	Status	in-use
Private DNS (IPv4)	Private DNS (IPv4)	ip-172-31-9-65.us-east-2.compute.internal
Secondary private IPv4 IPs	Secondary private IPv4 IPs	-
Source/dest. check	Source/dest. check	true
Instance ID	Instance ID	i-0bd1f0a6592cc8f9
Device Index	Device Index	0
Delete on termination	Delete on termination	true
Allocation ID	Allocation ID	eipalloc-027afa45f34984c87
Subnet ID	Subnet ID	subnet-5c980935
Availability Zone	Availability Zone	us-east-2a
Description	Description	UAG3-eth0 (Internet, Management and Backend)
Owner ID	Owner ID	189704117342 (my account)
Primary private IPv4 IP	Primary private IPv4 IP	172.31.9.65
IPv4 Public IP	IPv4 Public IP	13.59.224.71*
IPv6 IPs	IPv6 IPs	-
Attachment ID	Attachment ID	eni-attach-0434b676646296092
Attachment owner	Attachment owner	189704117342
Attachment status	Attachment status	attached
Owner ID	Owner ID	189704117342
Association ID	Association ID	eipassoc-0c30b665789396b13

Run the Unified Access Gateway Deploy Command `uagdeployec2.ps1`

6

You can deploy Unified Access Gateway to Amazon AWS EC2 with the PowerShell command.

Procedure

- 1 Download `uagdeployec2.ps1` and `uagdeploy.psm1` into a folder on your Windows machine.
- 2 Run the command.

`uag12.ini` is the name of your .INI file.

```
uagdeployec2.ps1 uag12.ini
```

Note If you receive an error message `Error: Failed to deploy UAG – User data is limited to 16384 bytes`, it means that the configuration data in your INI file is too large for Amazon AWS EC2 deployment. It is a known limitation which Amazon might increase in future. While this limit is in place, it might be necessary to reduce the amount of configuration data specified in your INI file. You should check the SSL certificate files to see if unnecessary root or intermediate certificates can be removed. If necessary, do not specify the SSL certificates, and upload them using the Unified Access Gateway Admin UI after deployment.
