How to onboard your Azure servers on InfraGuard

Prerequisite

Once you are ready to onboard Servers to InfraGuard from Azure, you will need to create a role and Activation on an AWS account to install agents on the servers. If you don't have an AWS account, contact InfraGuard support to create an Activation. You are going to need one IAM role for each Azure account.

Creating a Role for communicating with InfraGuard

- **Log into your AWS account console**
- **Go to IAM → Roles → Create Role**
- **Choose EC2 as the service that will use this role**
- **Attach Policy AmazonSSMFullAccess and skip to Review**
- **Enter Role Name "Infraguard-azure" & Press "Create Role"**
- **Again, Click on the role "Infraguard-azure" from the list**
- **Click on "Add Inline Policy"**
- **Click on "JSON"**
- **Replace the content with the JSON below:**

```
{
	"Version": "2012-10-17",
	"Statement": [
		{
			"Action": [
				"ec2:DescribeInstances",
				"ec2:DescribeInstanceAttribute",
				"ec2:DescribeTags",
				"ec2:ModifyInstanceAttribute",
				"ec2:StartInstances",
				"ec2:StopInstances",
				"ec2:rebootInstances",
				"ec2:DescribeImages",
				"ec2:CreateImage",
				"ec2:DeregisterImage",
			"iam:ListInstanceProfilesForRole",
			"iam:GetInstanceProfile"
			  ],
			  "Resource": "*",
			  "Effect": "Allow",
			  "Sid": "Stmt1434989080227"
		}
	]

	}
```

- **Click on "Review Policy"**
- **Enter the name "InfraGuard-azure-policy "**
- **Click on "Create Policy"**
- **Now, Click on "Trust Relationships" and replace Trust Relationship JSON with:**

```
{
  "Version": "2012-10-17",
  "Statement": [
  {
    "Sid": "foriamuser",
    "Effect": "Allow",
    "Principal": {
    "AWS": "arn:aws:iam::028197385767:role/infraguardswitchrole"
    },
    "Action": "sts:AssumeRole",
    "Condition": {
    "StringEquals": {
      "sts:ExternalId": "InfraGuardApp"
    }
    }
  },
  {
    "Sid": "forssmec2",
    "Effect": "Allow",
    "Principal": {
    "Service": [
      "ec2.amazonaws.com",
      "ssm.amazonaws.com"
    ]
    },
    "Action": "sts:AssumeRole"
  }
  ]

      }
```

- **Click on “Update Trust Policy”**

Create an Activation on an AWS account

- **Open the AWS Systems Manager console**
- **In the navigation pane, choose Hybrid Activations and choose Create activation.**
- **(Optional) In the Activation description field, enter a description for this activation.**
- **In the Instance limit field, specify the total number of on-premises servers or VMs that you want to register with AWS as part of this activation.**
- **In the IAM role name section, choose a service role option that enables your servers and VMs to communicate with AWS Systems Manager in the cloud:**
    1. **Choose Use the system created default command execution role to use a role and managed policy created by AWS.**
    2. **Choose Select an existing custom IAM role that has the required permissions to use the optional custom role you created earlier.**
- **In the Activation expiry date field, specify an expiration date for the activation.**
- **(Optional) In the Default instance name field, specify a name.**
- **Choose Create activation. Systems Manager immediately returns the Activation Code and ID to the console.**
- **Note down the Activation Code and Activation ID in a safe place.**

Install the agent in your Azure Window Environment:

- **Log on to a server or VM in your Window environment**
- **Open Windows PowerShell**

- **Copy and paste the following command block after replacing the placeholder values with the Activation Code and Activation ID and AWS Region code:**

```
        $code = "activation-code"
         $id = "activation-id"
         $region = "region"
         $dir = $env:TEMP + "\ssm"
         New-Item -ItemType directory -Path $dir -Force
         cd $dir
         (New-Object System.Net.WebClient).DownloadFile("https://amazon-ssm-
$region.s3.amazonaws.com/latest/windows_amd64/AmazonSSMAgentSetup.exe", $dir +
"\AmazonSSMAgentSetup.exe")
         Start-Process .\AmazonSSMAgentSetup.exe -ArgumentList @("/q", "/log",
"install.log", "CODE=$code", "ID=$id", "REGION=$region") -Wait
         Get-Content ($env:ProgramData + "\Amazon\SSM\InstanceData\registration")
                 Get-Service -Name "AmazonSSMAgent"
```

Install the agent in your Azure Linux Environment:

- **Log on to a server or VM in your Linux environment**
- **Copy and paste one of the following command blocks into SSH. Replace the placeholder values with the Activation Code and Activation ID and AWS Region code.**

On RHEL 6.x, and CentOS 6.x

```
        mkdir /tmp/ssm
         curl https://s3.amazonaws.com/ec2-downloads-
windows/SSMAgent/latest/linux_amd64/amazon-ssm-agent.rpm -o /tmp/ssm/amazon-ssm-
agent.rpm
         sudo yum install -y /tmp/ssm/amazon-ssm-agent.rpm
         sudo stop amazon-ssm-agent
         sudo amazon-ssm-agent -register -code "activation-code" -id "activation-id"
-region "region"
         sudo start amazon-ssm-agent
```

- ## On RHEL 7.x and CentOS 7.x

```
        mkdir /tmp/ssm
         curl https://s3.amazonaws.com/ec2-downloads-
windows/SSMAgent/latest/linux_amd64/amazon-ssm-agent.rpm -o /tmp/ssm/amazon-ssm-
agent.rpm
         sudo yum install -y /tmp/ssm/amazon-ssm-agent.rpm
         sudo systemctl stop amazon-ssm-agent
         sudo amazon-ssm-agent -register -code "activation-code" -id "activation-id"
-region "region"
         sudo systemctl start amazon-ssm-agent
```

- ## On SLES [SUSE Linux Enterprise Server]

```
        mkdir /tmp/ssm
         sudo wget https://s3.amazonaws.com/ec2-downloads-
windows/SSMAgent/latest/linux_amd64/amazon-ssm-agent.rpm
         sudo rpm --install amazon-ssm-agent.rpm
         sudo systemctl stop amazon-ssm-agent
```

```
        sudo amazon-ssm-agent -register -code "activation-code" -id "activation-id"
-region "region"
        sudo systemctl enable amazon-ssm-agent
        sudo systemctl start amazon-ssm-agent
```

- On Ubuntu, Debian

```
       mkdir /tmp/ssm

        curl https://s3.amazonaws.com/ec2-downloads-
windows/SSMAgent/latest/debian_amd64/amazon-ssm-agent.deb -o /tmp/ssm/amazon-ssm-
agent.deb
        sudo dpkg -i /tmp/ssm/amazon-ssm-agent.deb
        sudo service amazon-ssm-agent stop
        sudo amazon-ssm-agent -register -code "activation-code" -id "activation-id"
-region "region"

        sudo service amazon-ssm-agent start
```

- On Raspbian

```
       mkdir /tmp/ssm
        sudo curl https://s3.amazonaws.com/ec2-downloads-
windows/SSMAgent/latest/debian_arm/amazon-ssm-agent.deb -o /tmp/ssm/amazon-ssm-
agent.deb
        sudo dpkg -i /tmp/ssm/amazon-ssm-agent.deb
        sudo service amazon-ssm-agent stop
        sudo amazon-ssm-agent -register -code "activation-code" -id "activation-id"
-region "region"
        sudo service amazon-ssm-agent start
```

Attach AWS IAM role with managed Azure instance

- **Go to AWS Systems Manager and select “Managed instances” under Instances & Nodes**
- **Select your instance**
- **Go to Actions -> Change IAM Role**
- **From the drop-down, choose “Infraguard-azure” & press “Save IAM role”**
- **Make sure that Managed Instances is in running state**

Get account information from Azure's portal

- **Log in to your Azure account**
- **Search "Azure Active Directory" from the top search text box and select it**
- **Copy the tenant id and keep it at a safe place. We will need it while adding information on InfraGuard**
- **On the side menu of the same page click on "App registrations" and select "New registration"**
- **Enter the name as `Infraguard app`**
- **Make sure "Single Tenent" is selected in Supported Account types**
- **In Redirect URI select "Web" from the list and enter `https://app.infraguard.io` in the text box next to it**
- **Click on Register**
- **Now your application is registered. Copy the "Application (Client) ID" and keep it at a safe place**
- **Go to certificates and secrets and click on "New Client Secret" button**

- Enter description as `Infraguard key` and select "Never" in "EXPIRES" option radio button
- Click on Add
- This will generate a new client secret key. Copy the value column item and keep it at a safe place with the name `Client Secret`
- Now search "Subscription" on the top search text box and go to your current subscription and copy the subscription id and put it at a safe place
- Select "Access control(IAM)" and go to add a role assignment card on right and click on add
- In the "Add role Assignment" pop-up select role as a contributor
- Assign role to Azure AD user, Group, or Service principal
- Now enter "Infraguard app" in the select text box and select Infraguard app from the search list and click on save

Onboard your servers to InfraGuard

- Log onto app.infraguard.io Account
- Select CLUSTER from side-menu
- Click on “Create Azure cluster”
- Add any relevant Name
- Add your Role ARN (AWS IAM -> Roles -> Infraguard-azure)
- Add Tenant ID, Subscription ID, Client ID and Client Secret as created and copied in previous section
- Click ‘Sync’ to make your newly added server appear in list of servers
- Wait for some time before you click on ‘Servers’ to get your list of servers for that role ARN

Backup Image Creation via InfraGuard is not available for Azure Instances.
Azure on-board Youtube Playlist link - Click Here