

AMI Backup Setup

Steps involved in Auto AMI backup setup process

- 1, SSM agent installation
- 2, IAM Role
- 3, Attach IAM Role
- 4, System manager

1, SSM agent installation:

- Login to the server using ssh
- cat /etc/os-release (Check the OS of the server)

```
[[root@3-109-218-183 centos]# cat /etc/os-release
NAME="CentOS Linux"
VERSION="7 (Core)"
ID="centos"
ID_LIKE="rhel fedora"
VERSION_ID="7"
PRETTY_NAME="CentOS Linux 7 (Core)"
ANSI_COLOR="0;31"
CPE_NAME="cpe:/o:centos:centos:7"
HOME_URL="https://www.centos.org/"
BUG_REPORT_URL="https://bugs.centos.org/"

CENTOS_MANTISBT_PROJECT="CentOS-7"
CENTOS_MANTISBT_PROJECT_VERSION="7"
REDHAT_SUPPORT_PRODUCT="centos"
REDHAT_SUPPORT_PRODUCT_VERSION="7"
```

- Now go the link <https://go.aws/3patdM5>
- Select the correct OS. Here I'm selecting centos and then centos 7.x
- Copy the Intel 64-bit (x86_64) instances: `sudo yum install -y https://s3.region.amazonaws.com/amazon-ssm-region/latest/linux_amd64/amazon-ssm-agent.rpm`
- Replace the region with region where you have hosted the instance. Here I'm using ap-south-1 since I have hosted my EC2 instance in Mumbai region
- Run the updated in the terminal/putty `sudo yum install -y https://s3.ap-south-1.amazonaws.com/amazon-ssm-ap-south-1/latest/linux_amd64/amazon-ssm-agent.rpm`
- Now the amazon-ssm-agent is installed
- `sudo systemctl enable amazon-ssm-agent`
- `sudo systemctl start amazon-ssm-agent`
- `sudo systemctl status amazon-ssm-agent`

```

Installing:
amazon-ssm-agent      x86_64      3.1.338.0-1      /amazon-ssm-agent      111 M

Transaction Summary
=====
Install 1 Package

Total size: 111 M
Installed size: 111 M
Downloading packages:
Running transaction check
Running transaction test
Transaction test succeeded
Running transaction
  Installing : amazon-ssm-agent-3.1.338.0-1.x86_64                                1/1
Created symlink from /etc/systemd/system/multi-user.target.wants/amazon-ssm-agent.service to /etc/systemd/system/amazon-ssm-agent.service.
  Verifying  : amazon-ssm-agent-3.1.338.0-1.x86_64                                1/1

Installed:
amazon-ssm-agent.x86_64 0:3.1.338.0-1

Complete!
[root@3-109-218-183 centos]#

```


2, IAM Role:


- Go to the IAM Role console in the AWS console
- Click Role
- Now click create role
- Now Select type of trusted entity AWS services and choose use case as EC2
- Click Next:Permissions


Create role


1 2 3 4

Select type of trusted entity


AWS service
 EC2, Lambda and others


Another AWS account
 Belonging to you or 3rd party


Web identity
 Cognito or any OpenID provider


SAML 2.0 federation
 Your corporate directory

Allows AWS services to perform actions on your behalf. [Learn more](#)

Choose a use case

Common use cases

EC2

Allows EC2 instances to call AWS services on your behalf.

Lambda

Allows Lambda functions to call AWS services on your behalf.

Or select a service to view its use cases

API Gateway	CloudWatch Events	EMR	IoT SiteWise	RDS
AWS Backup	CodeBuild	EMR Containers	IoT Things Graph	Redshift
AWS Chatbot	CodeDeploy	ElastiCache	KMS	Rekognition
AWS Marketplace	CodeGuru	Elastic Beanstalk	Kinesis	RoboMaker

* Required

Cancel

Next: Permissions

- In the search field search for SSM and select AmazonSSMMaintenanceWindowRole, AmazonSSMManagedInstanceCore
- Click Next:Tags

Choose one or more policies to attach to your new role.









Create policy

↺

Filter policies ▾

SSM

Showing 19 results

	Policy name ▾	Used as
<input type="checkbox"/>	▶  AmazonEC2RoleforSSM	None
<input type="checkbox"/>	▶  AmazonSSMAutomationApproverAccess	None
<input type="checkbox"/>	▶  AmazonSSMAutomationRole	None
<input type="checkbox"/>	▶  AmazonSSMDirectoryServiceAccess	None
<input type="checkbox"/>	▶  AmazonSSMFullAccess	None
<input checked="" type="checkbox"/>	▶  AmazonSSMMaintenanceWindowRole	None
<input checked="" type="checkbox"/>	▶  AmazonSSMManagedInstanceCore	None
<input type="checkbox"/>	▶  AmazonSSMPatchAssociation	None

▶ Set permissions boundary

* Required

Cancel

Previous



Next: Tags

- Give Name in the key field and SSM in Value
- Click Next:Review
- Give the role name as SSM_AMI
- Then click create role
- Now select the role SSM_AMI
- Click add inline policy

Roles > SSM_AMI

Summary

Delete role

Role ARN	arn:aws:iam::447236384338:role/SSM_AMI 
Role description	Allows EC2 instances to call AWS services on your behalf. Edit
Instance Profile ARNs	arn:aws:iam::447236384338:instance-profile/SSM_AMI 
Path	/
Creation time	2021-10-15 14:47 UTC+0530
Last activity	Not accessed in the tracking period
Maximum session duration	1 hour Edit

Permissions

Trust relationships

Tags (1)

Access Advisor

Revoke sessions

▼ Permissions policies (2 policies applied)

Attach policies

+ Add inline policy

Policy name ▾	Policy type ▾	
▶  AmazonSSMManagedInstanceCore	AWS managed policy	✕
▶  AmazonSSMMaintenanceWindowRole	AWS managed policy	✕

- Click Json
- Remove the default lines and add the below lines and click Review Policy

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "VisualEditor0",
      "Effect": "Allow",
      "Action": [
        "ec2:DescribeImages",
        "ec2:CreateImage"
      ],
      "Resource": "*"
    }
  ]
}
```

- In the name field enter Create_ami
- Click create policy

A policy defines the AWS permissions that you can assign to a user, group, or role. You can create and edit a policy in the visual editor and using JSON. [Learn more](#)

Visual editor

JSON

Import managed policy

```

1 {
2   "Version": "2012-10-17",
3   "Statement": [
4     {
5       "Sid": "VisualEditor0",
6       "Effect": "Allow",
7       "Action": [
8         "ec2:DescribeImages",
9         "ec2:CreateImage"
10      ],
11      "Resource": "*"
12    }
13  ]
14 }

```

Security: 0
Errors: 0
Warnings: 0
Suggestions: 0

Character count: 144 of 10,240.

The current character count includes character for all inline policies in the role: SSM_AMI.

Cancel

Review policy

- Now Click Trust Relationships which is next to the permission tab
- Click edit Trust Relationships
- Remove the default line and add the below give line
- Now Click Update Trust Policy

Note: Trust Relationship Policy is given at the end of the document

Edit Trust Relationship

You can customize trust relationships by editing the following access control policy document.

Policy Document

```
1 {  
2   "Version": "2012-10-17",  
3   "Statement": [  
4     {  
5       "Effect": "Allow",  
6       "Principal": {  
7         "Service": [  
8           "ec2.amazonaws.com",  
9           "ssm.amazonaws.com"  
10        ]  
11      },  
12      "Action": "sts:AssumeRole"  
13    }  
14  ]  
15 }
```

[Cancel](#) [Update Trust Policy](#)

3, Attach IAM Role:

- Go to EC2 Console
- Select the instance and Click Actions
- Click the instance settings and click Attach/Replace IAM Role
- Now click the dropdown
- Select the Role SSM_AMI which you have created in the 3rd step
- Then click apply
- Reboot the instance for the attached IAM role to be effective

[Instances](#) > Attach/Replace IAM Role

Attach/Replace IAM Role

Select an IAM role to attach to your instance. If you don't have any IAM roles, choose Create new IAM role to create a role in the IAM console.
If an IAM role is already attached to your instance, the IAM role you choose will replace the existing role.

Instance ID I-0c93121aa855b09bd (Training) ⓘ

IAM role* SSM_AMI ⓘ [Create new IAM role](#) ⓘ

* Required

Filter by attributes

Profile Name
No Role
SSM_AMI

[Cancel](#) [Apply](#)

4, System manager:

- Go to System manager console
- Click Maintenance Windows under the change management category
- Click create Maintenance window

- In the Name field give the name Create_AMI
- Now under the Schedule select Cron schedule builder
- Select Daily, Everyday and enter the time when you want to create the AMI daily, here I have given 15:40 (3:40 pm) for demo (Note: Should be in 24 format)
- In the duration enter 2 hours
- In the Stop initiating tasks enter 1 hour
- Under Schedule timezone - optional click the dropdown
- Search Kolkata in the search field select Asia/Kolkata
- Click create maintenance window

Specify with

- ☒ Cron schedule builder
☐ Rate schedule builder
☐ CRON/Rate expression

☐ Default
 Window starts every 30 minutes

☐ Hourly
 Window starts on custom hourly rate

☒ Daily
 Window starts on custom daily rate

Every Day ▼

at 03:40

Duration

Maintenance window duration

2 hours

Value from 1 to 24.

Stop initiating tasks

Time to stop starting scheduled task before maintenance window ends

1 hour before the window closes

Value from 0 to 23.

Window start date - optional

Date time to start the maintenance window

MM/DD/YYYY



hh:mm:ss

GMT+00:00 ▼

Window end date - optional

Date time to stop the maintenance window

MM/DD/YYYY



hh:mm:ss

GMT+00:00 ▼

Q kolk



(GMT+05:30) Asia/**Kolkata**

(GMT-12:00) Etc/GMT+12 ▲

IANA timezone

Schedule offset - optional

Days to wait after the CRON expression date before running the maintenance window

- Now select the window id
- Click Actions
- Click Register Targets

- Give the target name as Target_AMI rest leave it as default
- Under Targets click choose instances manually
- Select the instance and click Register Targets

Targets

Targets are the AWS resources, such as instances, that you register with the maintenance window.

Target selection

Choose a method for selecting targets.

☐ Specify instance tags
Specify one or more tag key-value pairs to select instances that share those tags.

☒ Choose instances manually
Manually select the instances you want to register as targets.

☐ Choose a resource group
Choose a resource group that includes the resources you want to target.

i-0c93121aa855b09bd X

Instances

< 1 >

⚙

<input checked="" type="checkbox"/>	Name	Instance ID	Instance state	Availability zone	Ping
<input checked="" type="checkbox"/>	Training	i-0c93121aa855b09bd	running	ap-south-1a	Onlin

Cancel

Register target

- Now select the Window target ID
- Click Actions
- Click Register Automation task
- Give the Name as Target_automation rest leave it as default
- Under Automation document in the search field type AWS-CreateImage
- Select AWS-CreateImage
- Scroll down to target and select the created target
- Under Input parameters in the InstanceId enter the value as your running instance id
- In the NoReboot enter the value as true
- Now under Rate control in the target field enter 1 and in the errors field enter 1
- Under IAM service role choose Use a custom service role
- Click the dropdown
- Select the IAM Role which you have created in the 2nd step (SSM_AMI)
- Click Register Automation Task

Targets

Targets are the instances you would like to associate with this document. You can choose to target by both managed instance and tag.

Target by

- ☒ Selecting registered target groups
- ☐ Selecting unregistered targets
- ☐ Task target not required

e5d6c66f-19d9-47db-ad97-7fe692042514 X

< 1 >

<input checked="" type="checkbox"/>	Window target ID	Name	Owner information
<input checked="" type="checkbox"/>	e5d6c66f-19d9-47db-ad97-7fe692042514	Target_AMI	-

Input parameters

Variable name	Description	Value
InstanceId	(Required) The ID of the Amazon EC2 instance.	<input type="text" value="i-0c93121aa855b09bd"/>
NoReboot	(Optional) Do not reboot the instance before creating the image.	<input type="text" value="true"/>
AutomationAssumeRole	(Optional) The ARN of the role that allows Automation to perform the actions on your behalf.	<input type="text"/>

Rate control

Concurrency

Specify the number or percentage of targets on which to execute the task at the same time

- ☒ targets
- ☐ percentage

Error threshold

Stop the task after the task fails on the specified number or percentage of targets

- ☒ errors
- ☐ percentage

IAM service role

Select the service role that allows Maintenance Windows to interact with other AWS services on your behalf. [Learn more](#)

Service role option

- ☐ Create and use a service-linked role for Systems Manager
Create and use the service-linked role `AWSServiceRoleForAmazonSSM` to allow Maintenance Windows to manage AWS resources on your behalf. [Learn more](#)
- ☒ Use a custom service role
Select a custom service role that allows Maintenance Windows to interact with other AWS resources on your behalf.

arn:aws:iam::447236384338:role/SSM_AMI ▼

Cancel

Register Automation task

- Now the AMI will create daily as per your Cron scheduler

Trust Relationship Policy:

```
{
  "Version": "2012-10-17",
  "Statement": [
    {
      "Sid": "",
      "Effect": "Allow",
      "Principal": {
        "Service": [
          "ssm.amazonaws.com",
          "ec2.amazonaws.com"
        ]
      },
      "Action": "sts:AssumeRole"
    }
  ]
}
```