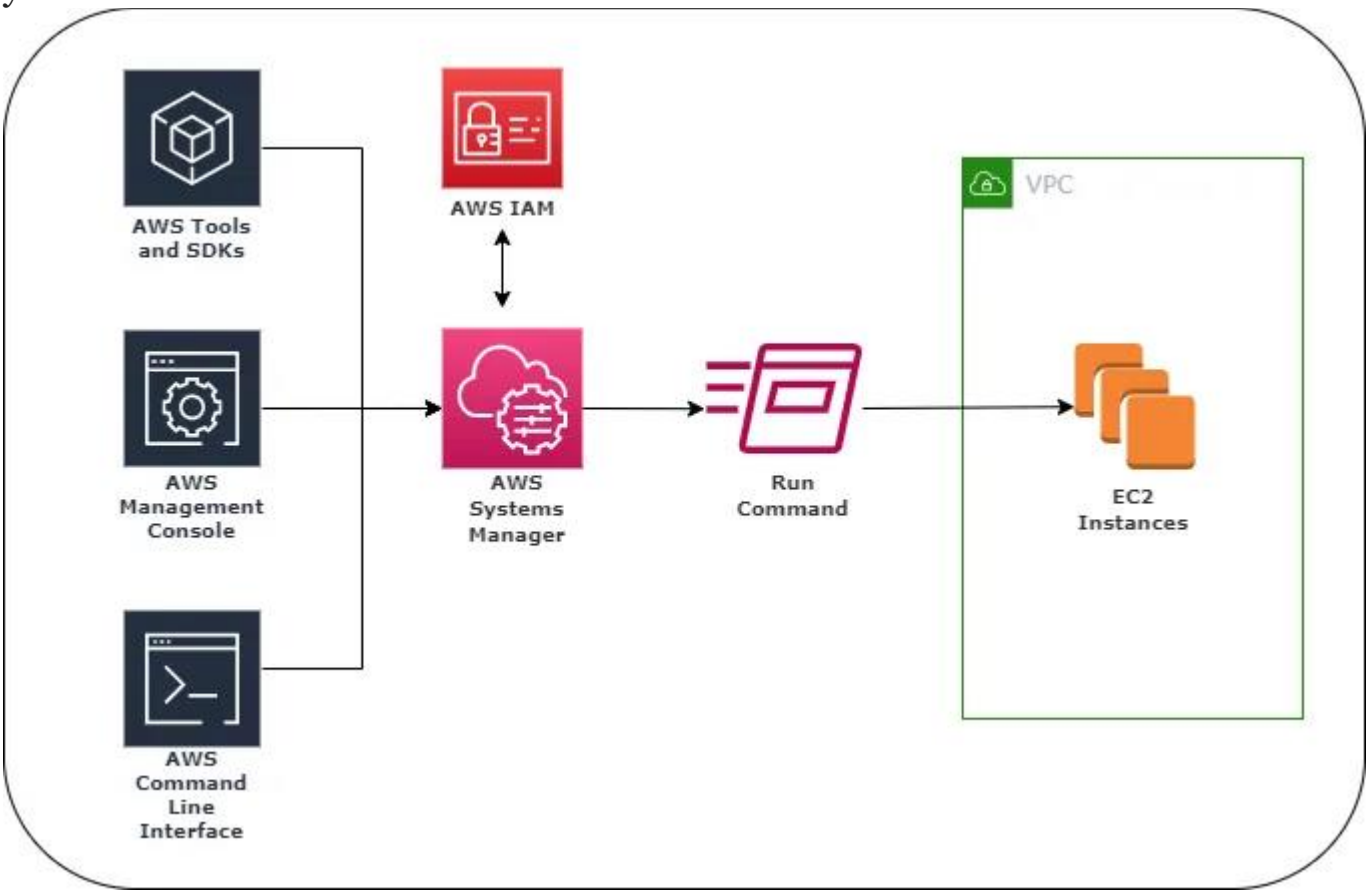
What is AWS Systems Manager?

Systems Manager is a Management Tool that enables you to gain operational insights and take action on AWS resources safely and at scale. Using the run command, one of the automation features of Systems Manager, you can simplify management tasks by eliminating the need to use bastion hosts, SSH, or remote PowerShell.

When you need to update the packages on your EC2 instances, but your security team does not allow you to directly access the instances via SSH or does not allow you to use bastion hosts, you can use

Systems Manager to remotely run commands, like update packages, on your EC2 instances.



architecture diagram

Pre-requisites:

- An AWS Account
- An AWS CLI
- Boto3

Steps:

Step-1 Create an IAM Role

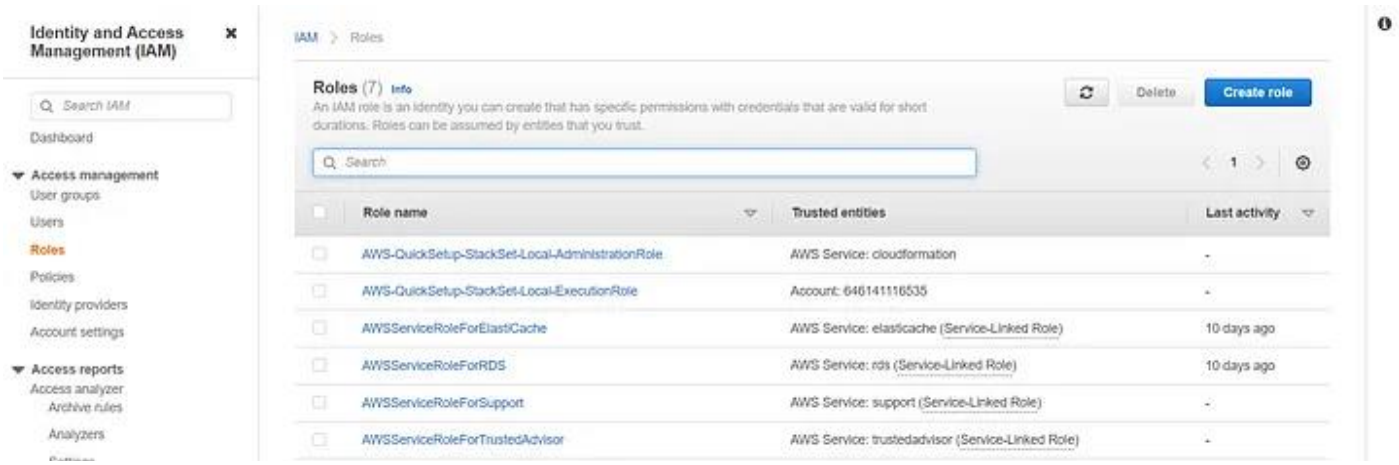
In this step, we'll create an IAM role that will be used to give AWS Systems Manager permissions to perform actions on your instances.

First, log in to your AWS Account and Open your IAM console at: https://console.aws.amazon.com/iam/. Then click on the Create Role.

On the 'select type of trusted entity page, under the AWS Service, choose EC2 and then choose Next: Permissions.

On the Attached permissions policy page, in the search bar type AmazonEC2RoleforSSM then from the policy list select AmazonEC2RoleforSSM, and then choose Next: Review.

On the Review page, in the Role name box type in EC2RoleForSSM. In the Role description box type in Allow EC2 instances to call AWS services on your behalf. Choose to Create a role.



Step-2: Install SSM Agent into your EC2 instance

SSM Agent makes it possible for the Systems Manager to update, manage, and configure these resources. The agent processes requests from the Systems Manager service in the AWS Cloud, and then run them as specified in the request. SSM Agent then sends status and execution information back to the Systems Manager service by using the Amazon Message Delivery Service (service prefix: ec2messages).

In most cases, SSM Agent is preinstalled, by default, on the following Amazon Machine Images (AMIs):

Amazon Linux

2. Amazon Linux 2

3. Amazon Linux 2 ECS-Optimized Base AMIs

4. SUSE Linux Enterprise Server (SLES) 12 and 15

5. Ubuntu Server 16.04, 18.04, and 20.04

→ Install & Configure the SSM agent on an EC2 instance for Linux

→ https://docs.aws.amazon.com/systems-manager/latest/userguide/sysman-install-ssm-win.html

Now if your EC2 instances are already running and do not have an IAM role for SSM then you've to attach the IAM Role (EC2RoleForSSM)that we've just created in STEP-1.

Step-3: Run the boto3 script

By running Boto script you can execute multiple commands onto your multiple EC2 instances from your CLI.

Now open the AWS System Manager dashboard to check the details of our command.

Use Cases:

Patch Management:

Use Case: Regularly update software packages and apply patches across multiple EC2 instances.

Benefit: Ensures all instances are up-to-date with security patches and software updates without manual intervention.

Configuration Management:

Use Case: Configure application settings or update configuration files across multiple instances.

Benefit: Maintains consistency in application settings across the fleet of instances, reducing configuration drift.

Automated Deployment:

Use Case: Deploy new versions of applications or scripts simultaneously to multiple instances.

Benefit: Streamlines the deployment process, ensuring all instances are running the latest version of the application or script.

Monitoring and Troubleshooting:

Use Case: Execute diagnostic commands or retrieve logs from multiple instances to investigate issues.

Benefit: Facilitates quick diagnosis and resolution of problems across the environment, improving overall system reliability.

Conclusion:

Using AWS SSM Agent for remote command execution on multiple EC2 instances enhances operational efficiency by enabling simultaneous command execution across large fleets, ensuring consistency in configurations and updates, automating routine tasks for DevOps efficiency, and bolstering security through timely patching and compliance audits. This approach supports a more robust and manageable cloud infrastructure aligned with best practices for operational excellence.