

# Task 1: Generate Inventory Lists for Managed Instances

You can use **AWS Systems Fleet Manager** to collect operating system, application, and instance metadata from your Amazon EC2 instances and your on-premises servers or virtual machines in your hybrid environment. You can query the metadata to quickly understand which instances are running the software and configurations required by your software policy, and which instances need to be updated.

In this task, you will use the AWS Systems Fleet Manager to gather inventory from an Amazon EC2 instance.

5. In the **AWS Management Console**, on the Services menu, click **Systems Manager**.
6. In the left navigation pane, click **Fleet Manager**.
7. Click Get Started if it appears.

If the webpage is redirected to documentation page then scroll down through the webpage and go back to AWS console by clicking the back button on your browser window.

8. Under **Managed Instance**, on the Account management menu, click **Set up inventory**

You will now create an association that will collect information about software and settings for your managed instance.

- **Name:** Inventory-Association
  - **Targets:** Manually selecting instances
    - Select **Managed Instance**
9. Click Setup Inventory (at the bottom of the page).

AWS Systems Manager Inventory will now regularly inventory the instance for the selected properties.

10. Click the **Instance ID** link displayed in the **Managed Instance** row.
11. Click the **Inventory** tab.

A list of all of the applications on the instance will be displayed. If nothing appears, wait a minute, then refresh the page.

Take a moment to review the installed applications and other options in the **Inventory type** pull-down menu.

You have successfully created an AWS Systems Manager inventory association for your instance. Using AWS Systems Manager Inventory, you can review and validate software configurations on your instances without needing to SSH into each instance.

## Task 2: Install a Custom Application using Run Command

In this task, you will install a custom web application (*Widget Manufacturing Dashboard*) using the AWS Systems Manager Run Command without logging into the instance.

12. In the left navigation pane, click **Run Command**.
13. Click Run command

You will see a list of pre-configured documents for running common commands. You will run a pre-configured command.

14. Click then:
  - *Owner*
  - *Owned by me*

A document will appear.

15. Click the name of the document.

A new tab will open with the description of the document. Verify the description is *Install Dashboard App*

16. Click the **Content** tab.

The contents will be displayed. It contains a script that:

- Installs an Apache web server and PHP
- Activates the web server
- Installs the AWS SDK for PHP
- Installs an application

These commands will install an application and all the support files required by the application.

17. Close the current web browser tab to return to the *Run a command* tab.
18. Select the document (click the circle, not the name).
19. For **Targets**, select Choose instances manually
20. Select **Managed Instance**.

The *Managed Instance* has the Systems Manager agent installed. The agent has registered the instance to the service, which allows it to be selected for the Run Command.

It is also possible to identify target instances by using Tags. This makes it easy to run a single command on a whole fleet of matching instances.

21. Expand **AWS command line interface command** (at the bottom of the page).

This section displays the CLI command that will trigger the Run Command. You could copy this command and use it in future within a script rather than having to use the management console.

## 22. Click Run

The progress of your command will be displayed. You will see an **In Progress** message in the Overall status column.

## 23. Wait for the **Overall status** to change to *Success*. You can occasionally click refresh in the top right to update the status.

You will now validate that the custom application was installed.

## 24. Copy the **ServerIP** value by clicking on the Details drop down menu above these instructions, and then click Show"

## 25. Open a new web browser tab, paste the IP address you just copied and press Enter.

The Widget Manufacturing Dashboard that you installed will now appear.

You have successfully run a Run Command via AWS Systems Manager that has installed a custom application onto your instance without needing to remotely access the instance via SSH.

# Task 3: Use Parameter Store to Manage Application Settings

AWS Systems Manager Parameter Store provides secure, hierarchical storage for configuration data management and secrets management. You can store data such as passwords, database strings, and license codes as parameter values. You can store values as plain text or encrypted data. You can then reference values by using the unique name that you specified when you created the parameter.

In this task you will use AWS Systems Manager Parameter Store to store a parameter that will be used to activate a feature in an application.

## 26. Keep the Dashboard tab open, but return to the Management Console tab.

## 27. In the left navigation pane, under **Application Management**, click **Parameter Store**.

## 28. Click Create parameter:

- **Name:** /dashboard/show-beta-features
- **Description:** Display beta features
- **Value:** True
- Click Create parameter

The parameter can be specified as a hierarchical path, such as: */dashboard/<option>*

The application that is running on the EC2 instance will automatically check for the existence of this parameter. If it finds the parameter, then additional features will be displayed.

29. Return to the web browser tab displaying the application.

If the tab is not available, copy the **ServerIP** value by clicking on the Details drop down menu above these instructions, and then click Show"

Notice that the application is only displaying two charts.

30. Refresh the web page.

You should now notice that *three* charts are displayed. This is because the application is checking the Parameter Store to determine whether the additional chart (which is still in beta) should be displayed. This is a common method that applications can be configured to display "dark features" that are installed but not yet activated.

**Optional:** Remove the parameter, then refresh the application. The third chart should disappear again!

## Task 4: Use Session Manager to Access Instances

AWS Systems Manager Session Manager lets you manage your Amazon EC2 instances through an interactive one-click browser-based shell or through the AWS CLI. Session Manager provides secure and auditable instance management without the need to open inbound ports, maintain bastion hosts, or manage SSH keys. Session Manager also makes it easy to comply with corporate policies that require controlled access to instances, strict security practices, and fully auditable logs with instance access details, while still providing end users with simple one-click cross-platform access to your Amazon EC2 instances.

When used with Microsoft Windows, the AWS Systems Manager Session Manager provides access to a PowerShell console on the instance.

In this task, you will access the Amazon EC2 instance via Session Manager.

31. In the Management Console, in the left navigation pane, click **Session Manager**.
32. Click Start Session
33. Select **Managed Instance**.
34. Click Start session

A session window will open in your browser.

35. Click in the session to activate the cursor.
36. Run this command in the session window:



```
ls /var/www/html
```

You will see application files that were installed on the instance.

37. Run this command in the session window:



```
# Get region
AZ=`curl -s http://169.254.169.254/latest/meta-
data/placement/availability-zone`
export AWS_DEFAULT_REGION=${AZ::-1}

# List information about EC2 instances
aws ec2 describe-instances
```

This demonstrates how AWS Systems Manager Session Manager can be used to login to an instance *without using SSH*. In fact, this instance does not have SSH port 22 opened in its Security Group. *Optional* you could verify the same by browsing to the Security Group used by the instance.

Access to the Session Manager can be restricted via IAM policies and usage is logged in AWS CloudTrail. This provides much better security and auditing than traditional SSH access.