

CLOUD WATCH AGENT IN WINDOWS EC2 SERVER

Step 1 – Create IAM Role for ssm & cloudwatch(policy required are attached below)

<input type="checkbox"/>	Policy name ↗	Type	Attached entities
<input type="checkbox"/>	AmazonSSMManagedInstanceCore	AWS managed	1
<input type="checkbox"/>	CloudWatchAgentAdminPolicy	AWS managed	1
<input type="checkbox"/>	CloudWatchAgentServerPolicy	AWS managed	1

Step 2- Create Your Ec2 Instance (windows server)& attach IAM to it

Instances (1) Info

Refresh

Connect

Instance state ▾

Actions ▾

Launch instances ▾

Find Instance by attribute or tag (case-sensitive)

All states ▾

< 1 >

⚙

<input type="checkbox"/>	Name ↗ ▾	Instance ID	Instance state ▾	Instance type ▾	Status check	Alarm status	Availability Zone ▾	Public IPv4 D
<input type="checkbox"/>	Web Server	i-05b9849cb8ed2a094	<div><div>Running</div><div>🔍</div></div>	t2.micro	<div><div>2/2 checks passed</div><div>View alarms +</div></div>		us-east-1b	ec2-18-212-6

Wait till session manager comes online

[EC2](#) > [Instances](#) > [i-05b9849cb8ed2a094](#) > [Connect to instance](#)

Connect to instance [Info](#)

Connect to your instance i-05b9849cb8ed2a094 (Web Server) using any of these options

[Session Manager](#) | [RDP client](#) | [EC2 serial console](#)

Session Manager usage:

- Connect to your instance without SSH keys, a bastion host, or opening any inbound ports.
- Sessions are secured using an AWS Key Management Service key.
- You can log session commands and details in an Amazon S3 bucket or CloudWatch Logs log group.
- Configure sessions on the Session Manager [Preferences](#) [↗](#) page.

[Cancel](#) [Connect](#)

Step 3- connect your server through RDP Client

[AWS](#) [Services](#) [Alt+S] N. Virginia Archit

[EC2](#) > [Instances](#) > [i-05b9849cb8ed2a094](#) > [Connect to instance](#)

Connect to instance [Info](#)

Connect to your instance i-05b9849cb8ed2a094 (Web Server) using any of these options

[Session Manager](#) | [RDP client](#) | [EC2 serial console](#)

Instance ID

[i-05b9849cb8ed2a094](#) (Web Server)

Connection Type

☒ **Connect using RDP client**
Download a file to use with your RDP client and retrieve your password.

☐ **Connect using Fleet Manager**
Connect to your instance using Fleet Manager Remote Desktop.

You can connect to your Windows instance using a remote desktop client of your choice, and by downloading and running the RDP shortcut file below:

[Download remote desktop file](#)

When prompted, connect to your instance using the following username and password:

Public DNS
[ec2-18-212-63-53.compute-1.amazonaws.com](#)

Username [Info](#)
[Administrator](#)

[CloudShell](#) [Feedback](#) © 2024, Amazon Web Services, Inc. or its affiliates. [Privacy](#) [Terms](#) [Cookie preferences](#)

Step 4 -Install CloudWatch Agent using Systems Manager

1. Verify the instance is up and running and passed both status checks.
2. Navigate to Run Command in Node Management (AWS Systems Manager > Run Command).
3. In the Command document list, choose **AWS-ConfigureAWSPackage** with “**Latest version at runtime**”.

Command document
Select the type of command that you want to run.

< 1 >

Search: AWS-ConfigureAWSPackage X Clear filters

Name	Owner	Platform types
AWS-ConfigureAWSPackage	Amazon	Windows, Linux, MacOS

Description
Install or uninstall a Distributor package. You can install the latest version, default version, or a version of the package you specify. Packages provided by AWS such as AmazonCloudWatchAgent, AwsEnaNetworkDriver, and AWSPVDriver are also supported.

Document version
Choose the document version you want to run.

Latest version at runtime

Choose the appropriate command parameters for installing the CloudWatch agent.

1. In the **Action** list, choose **Install**
2. Select **Uninstall and reinstall** from the **Installation Type**
3. In the **Name** box, enter **AmazonCloudWatchAgent**
3. Keep **Version** set to **latest** to install the latest version of the agent

Command parameters

Action
(Required) Specify whether or not to install or uninstall the package.

1 Install

Installation Type
(Optional) Specify the type of installation. Uninstall and reinstall: The application is taken offline until the reinstallation process completes. In-place update: The application is available while new or updated files are added to the installation.

2 Uninstall and reinstall

Name
(Required) The package to install/uninstall.

3 AmazonCloudWatchAgent

Version
(Optional) The version of the package to install or uninstall. If you don't specify a version, the system installs the latest published version by default. The system will only attempt to uninstall the version that is currently installed. If no version of the package is installed, the system returns an error.

4 latest

Additional Arguments
(Optional) The additional parameters to provide to your install, uninstall, or update scripts.

5 {}

In the **Targets** area, choose the instance on which to install the CloudWatch agent.

Target selection

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.

You haven't selected any instances.

Instances

< 1 > ⚙

<input type="checkbox"/>	Node ID	Source type	Source ID	Name	Ping status
<input type="checkbox"/>	i-02440606354f18ba7	AWS::EC2::Instance	i-32440606354f18ba7	windows-poc	Online

Choose **Run** and ensure that it is successfully executed

Command ID: f55ee434-3156-4536-bf38-826baff08218

Command status

Overall status	Detailed status	# targets	# completed	# error	# delivery timed out
Success	Success	1	1	0	0

Targets and outputs

< 1 >

	Instance ID	Instance name	Status	Detailed Status	Start time	Finish time
<input type="radio"/>	i-144800834f1bea7	EC2AMAZ- H5DQFUS.WORKGROUP	Success	Success	Tue, 30 May 2023 05:46:09 GMT	Tue, 30 May 2023 05:46:21 GMT

Command execution is Completed

Step 5- open PowerShell(PS)terminal to check if cloud watch agent is Installed or not

```
Get-Service -Name "AmazonCloudWatchAgent"
```

it can be Installed in stopped state

```
PS C:\Users\Administrator> Get-Service -Name "AmazonCloudWatchAgent"

Status  Name              DisplayName
-----
Stopped AmazonCloudWatc... Amazon CloudWatch Agent
```

Cloudwatch Agent Service Status

Step 6: Configure the CloudWatch Agent

1. Log in to the **Windows EC2 Instance** using the **RDP client** and launch **PowerShell**. Execute the below commands:

```
> cd "C:\Program Files\Amazon\AmazonCloudWatchAgent"

> .\amazon-cloudwatch-agent-config-wizard.exe
```

This will start the CloudWatch Agent configuration wizard

```
PS C:\Program Files\Amazon\AmazonCloudWatchAgent> .\amazon-cloudwatch-agent-config-wizard.exe
=====
= Welcome to the Amazon CloudWatch Agent Configuration Manager =
=
= CloudWatch Agent allows you to collect metrics and logs from =
= your host and send them to CloudWatch. Additional CloudWatch =
= charges may apply. =
=====
On which OS are you planning to use the agent?
1. linux
2. windows
3. darwin
default choice: [2]:
```

Launch the wizard

Answer the questions as per requirement.

```
PS C:\Program Files\Amazon\AmazonCloudWatchAgent> .\amazon-cloudwatch-agent-config-wizard.exe
```

```
=====
= Welcome to the Amazon CloudWatch Agent Configuration Manager =
=
= CloudWatch Agent allows you to collect metrics and logs from =
= your host and send them to CloudWatch. Additional CloudWatch =
= charges may apply. =
=====
```

On which OS are you planning to use the agent?

- 1. linux
- 2. windows
- 3. darwin

default choice: [2]:

2

Trying to fetch the default region based on ec2 metadata...

I! imds retry client will retry 1 timesAre you using EC2 or On-Premises hosts?

- 1. EC2
- 2. On-Premises

default choice: [1]:

1

Do you want to turn on StatsD daemon?

1. yes

2. no

default choice: [1]:

2

Do you have any existing CloudWatch Log Agent configuration file to import for migration?

1. yes

2. no

default choice: [2]:

2

Do you want to monitor any host metrics? e.g. CPU, memory, etc.

1. yes

2. no

default choice: [1]:

1

Do you want to monitor cpu metrics per core?

1. yes

2. no

default choice: [1]:

1

Do you want to add ec2 dimensions (ImageId, InstanceId, InstanceType, AutoScalingGroupName) into all of your metrics if the info is available?

1. yes

2. no

default choice: [1]:

1

Do you want to aggregate ec2 dimensions (InstanceId)?

1. yes

2. no

default choice: [1]:

1

Would you like to collect your metrics at high resolution (sub-minute resolution)? This enables sub-minute resolution for all metrics, but you can customize for specific metrics in the output json file.

1. 1s

2. 10s

3. 30s

4. 60s

default choice: [4]:

4

Which default metrics config do you want?

1. Basic

2. Standard

3. Advanced

4. None

default choice: [1]:

2

Current config as follows:

```
{
  "metrics": {
    "aggregation_dimensions": [
      "InstanceId"
    ],
    "append_dimensions": {
      "AutoScalingGroupName":
"${aws:AutoScalingGroupName}",
      "ImageId": "${aws:ImageId}",
      "InstanceId": "${aws:InstanceId}",
      "InstanceType": "${aws:InstanceType}"
    },
    "metrics_collected": {
      "LogicalDisk": {
        "measurement": [
          "% Free Space"
        ],
        "metrics_collection_interval": 60,
        "resources": [
          "*"
        ]
      },
      "Memory": {
```

```

        "measurement": [
            "% Committed Bytes In Use"
        ],
        "metrics_collection_interval": 60
    },
    "Paging File": {
        "measurement": [
            "% Usage"
        ],
        "metrics_collection_interval": 60,
        "resources": [
            "*"
        ]
    },
    "PhysicalDisk": {
        "measurement": [
            "% Disk Time"
        ],
        "metrics_collection_interval": 60,
        "resources": [
            "*"
        ]
    },
    "Processor": {
        "measurement": [
            "% User Time",
            "% Idle Time",
            "% Interrupt Time"
        ],
        "metrics_collection_interval": 60,
        "resources": [
            "*"
        ]
    }
}

```

Are you satisfied with the above config? Note: it can be manually

customized after the wizard completes to add additional items.

1. yes

2. no

default choice: [1]:

1

Do you want to monitor any customized log files?

1. yes

2. no

default choice: [1]:

2

Do you want to monitor any Windows event log?

1. yes

2. no

default choice: [1]:

2

Do you want the CloudWatch agent to also retrieve X-ray traces?

1. yes

2. no

default choice: [1]:

2

Existing config JSON identified and copied to:

C:\Users\Administrator\AppData\Roaming\Amazon\CloudWatchAgent\etc\backup-configs

Saved config file to config.json successfully.

Current config as follows:

```
{
  "metrics": {
    "aggregation_dimensions": [
      "InstanceId"
    ],
    "append_dimensions": {
      "AutoScalingGroupName":
"${aws:AutoScalingGroupName}",
      "ImageId": "${aws:ImageId}",
      "InstanceId": "${aws:InstanceId}",
      "InstanceType": "${aws:InstanceType}"
    }
  }
}
```

```
    },
    "metrics_collected": {
      "LogicalDisk": {
        "measurement": [
          "% Free Space"
        ],
        "metrics_collection_interval": 60,
        "resources": [
          "*"
        ]
      },
      "Memory": {
        "measurement": [
          "% Committed Bytes In Use"
        ],
        "metrics_collection_interval": 60
      },
      "Paging File": {
        "measurement": [
          "% Usage"
        ],
        "metrics_collection_interval": 60,
        "resources": [
          "*"
        ]
      },
      "PhysicalDisk": {
        "measurement": [
          "% Disk Time"
        ],
        "metrics_collection_interval": 60,
        "resources": [
          "*"
        ]
      },
      "Processor": {
        "measurement": [
          "% User Time",
```

```
        "% Idle Time",  
        "% Interrupt Time"  
    ],  
    "metrics_collection_interval": 60,  
    "resources": [  
        "*" ]  
    }  
}  
}
```

Please check the above content of the config.

The config file is also located at `config.json`.

Edit it manually if needed.

Do you want to store the config in the SSM parameter store?

1. yes
2. no

```
default choice: [1]:
```

1

What parameter store name do you want to use to store your config? (Use 'AmazonCloudWatch-' prefix if you use our managed AWS policy)

```
default choice: [AmazonCloudWatch-windows]
```

```
Trying to fetch the default region based on ec2 metadata...
```

```
I! imds retry client will retry 1 timesWhich region do you want to store
the config in the parameter store?
```

```
default choice: [us-east-1]
```

Which AWS credential should be used to send json config to parameter store?

1. ASIAUR4XXXXXXXXXX (From SDK)
2. Other

```
default choice: [1]:
```

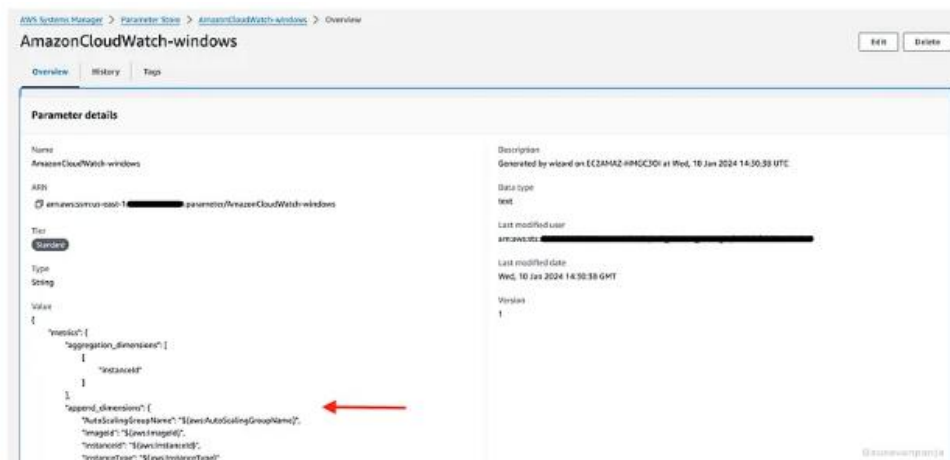
1

Successfully put config to parameter store AmazonCloudWatch-windows.

```
Please press Enter to exit...
```

```
Program exits now.
```

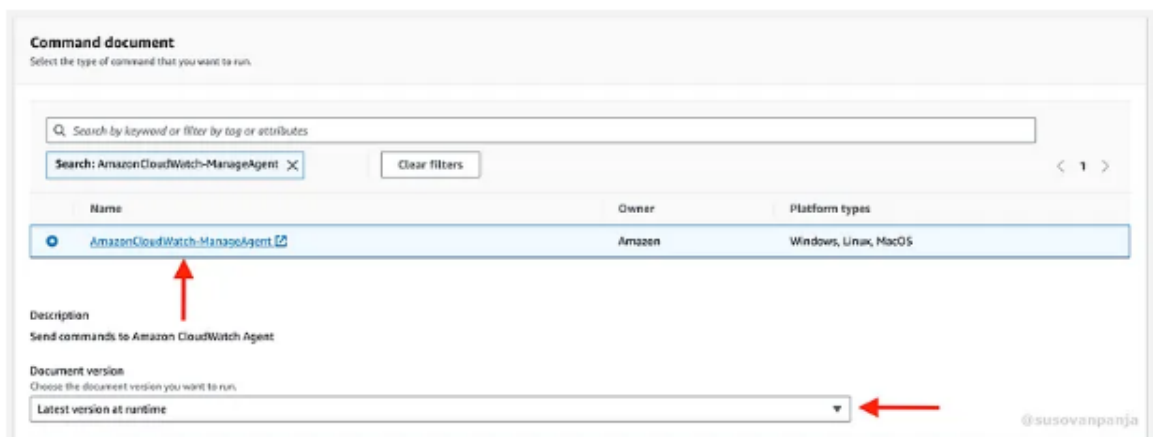
Verify the CloudWatch Agent configuration file by going to **Systems Manager -> Parameter store -> 'AmazonCloudWatch-windows'**



Verify config in the SSM parameter store

Step 7: Start the CloudWatch Agent via the Systems Manager

1. Go to **Systems Manager -> Fleet manager**, select the instance/node, and click on **'Execute run command'**.



Select **AmazonCloudWatch-ManageAgent** from the command documents

In the **Command parameters** section select the below options:

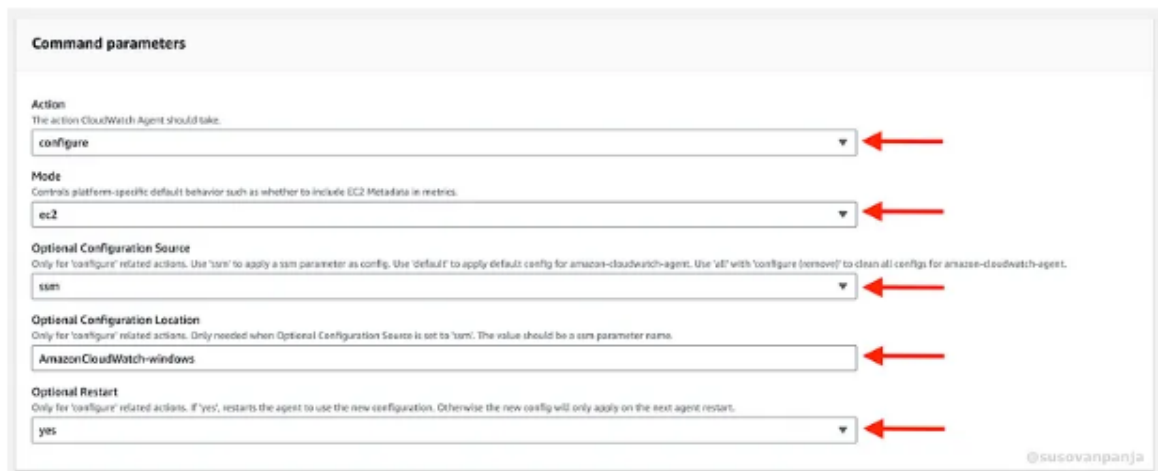
- a) Action -> **configure**
- b) Mode -> **ec2**
- c) Optional Configuration Source -> **ssm** (to pick up the config)

file from ssm parameters)

d) Optional Configuration Location -

> **AmazonCloudWatch-windows** (paste as it is, name of the ssm parameter we created in **Step 5**)

e) Optional Restart -> **yes**



The screenshot shows the 'Command parameters' section of the AWS CloudWatch Agent configuration console. It contains five dropdown menus, each with a red arrow pointing to it from the right:

- Action:** The action CloudWatch Agent should take. Set to `configure`.
- Mode:** Controls platform-specific default behavior such as whether to include EC2 Metadata in metrics. Set to `ec2`.
- Optional Configuration Source:** Only for 'configure' related actions. Use 'ssm' to apply a ssm parameter as config. Set to `ssm`.
- Optional Configuration Location:** Only for 'configure' related actions. Only needed when Optional Configuration Source is set to 'ssm'. The value should be a ssm parameter name. Set to `AmazonCloudWatch-windows`.
- Optional Restart:** Only for 'configure' related actions. If 'yes', restarts the agent to use the new configuration. Otherwise the new config will only apply on the next agent restart. Set to `yes`.

At the bottom right of the form, there is a watermark: `@susovanpanja`.

Select the appropriate options

. Select the instance in the **Target selection**.

. Click on the '**Run**' button to execute the command and make sure it gets executed successfully.

Step 8: Verify that CloudWatch Agent is sending metrics to the CloudWatch Console

```
Get-Service -Name "AmazonCloudWatchAgent"
```

```
PS C:\Users\Administrator> Get-Service -Name "AmazonCloudWatchAgent"

Status      Name                DisplayName
-----
Running     AmazonCloudWatc...  Amazon CloudWatch Agent
```

@susovanpanja

CloudWatch Agent is running

Conclusion

We have created a **Windows EC2 instance**, installed and configured **CloudWatch Agent** on that instance using the **SSM console**, and viewed the system-level metrics on the **CloudWatch Dashboard**.