CC Mini Project Report
File Sharing System on Cloud using AWS EC2 and EFS

Introduction

The aim of this project is to create a cloud file sharing system using the services of Amazon AWS EC2 and EFS. Along with that we aim to provide additional security by creating security groups allowing only specific instances to access those files at the discretion of the owner. The technologies used in this project are EC2 for storage, EFS for file sharing systems and NFS. We also use cloudwatch to monitor the usage and keep a track of resources utilised. PuTTY private keys are used for authentication

Problem Definition

- Create a File Sharing system with storage on cloud
- Create security groups to only allow access to security groups
- Use Cloudwatch to keep a tab on the resources being utilised

Amazon Elastic File System(EFS)

We use EFS to provide a simple file storage interface. It offers users a simple interface that allows you to create and configure file systems quickly and easily. When mounted on Amazon EC2 instances, an Amazon EFS file system provides a standard file system interface and file system access semantics, allowing you to seamlessly integrate Amazon EFS with your existing applications and tools. Multiple Amazon EC2 instances can access an Amazon EFS file system at the same time, allowing Amazon EFS to provide a common data source for workloads and applications running on more than one Amazon EC2 instance.

Mounting an EFS file system on EC2

Use the mount helper in the amazon-efs-utils package. The amazon-efs-utils package is an open-source collection of Amazon EFS tools.

To use the mount helper, you need the following:

- **An Amazon EFS file system ID** – After you create an Amazon EFS file system, you can get that file system's ID from the console or programmatically through the Amazon EFS API. This ID is in this format: fs-12345678.
- **An Amazon EFS mount target** – You create mount targets in your VPC. If you create your file system in the console, you create your mount targets at the same time.
- **An Amazon EC2 instance running a supported distribution of Linux** – The supported Linux distributions for mounting your file system with the mount helper are Amazon

Linux 2, Amazon Linux 2017.09 and newer, Red Hat Enterprise Linux (and derivatives such as CentOS) version 7 and newer, and Ubuntu 16.04 LTS and newer.

- **The Amazon EFS mount helper installed** – The mount helper is a tool in amazon-efs-utils.

To mount your Amazon EFS file system with the mount helper

- Access the terminal for your instance through Secure Shell (SSH), and log in with the appropriate user name.Run the following command to mount your file system.
- sudo mount -t efs *fs-12345678*:/ */mnt/efs*
- Alternatively, if you want to use encryption of data in transit, you can mount your file system with the following command.
- sudo mount -t efs -o tls *fs-12345678*:/ */mnt/efs*

You also have the option of mounting automatically by adding an entry to your /etc/fstab file. When you mount automatically using /etc/fstab, you must add the _netdev mount option.

Note

Mounting with the mount helper automatically uses the following mount options that are optimized for Amazon EFS:

- nfsvers=4.1
- rsize=1048576
- wsize=1048576
- hard
- timeo=600
- retrans=2

To use the mount command, the following must be true:

- The connecting EC2 instance must be in a VPC and must be configured to use the DNS server provided by Amazon.
- The VPC of the connecting EC2 instance must have DNS host names enabled.

EC2 Security Group

A *security group* acts as a virtual firewall that controls the traffic for one or more instances. When you launch an instance, you associate one or more security groups with the instance. You add rules to each security group that allow traffic to or from its associated instances. You can modify the rules for a security group at any time; the new rules are automatically applied to all instances that are associated with the security group after a short period. When we decide whether to allow traffic to reach an instance, we evaluate all the rules from all the security groups that are associated with the instance.

For each rule, you specify the following:

- **Protocol**: The protocol to allow. The most common protocols are 6 (TCP) 17 (UDP), and 1 (ICMP).
- **Port range** : For TCP, UDP, or a custom protocol, the range of ports to allow. You can specify a single port number (for example, 22), or range of port numbers (for example,7000-8000).
- **ICMP type and code**: For ICMP, the ICMP type and code.
- **Source or destination**: The source (inbound rules) or destination (outbound rules) for the traffic. Specify one of these options:
    - An individual IPv4 address. You must use the /32 prefix length; for example, 203.0.113.1/32.
    - (VPC only) An individual IPv6 address. You must use the /128 prefix length; for example 2001:db8:1234:1a00::123/128.
    - A range of IPv4 addresses, in CIDR block notation, for example, 203.0.113.0/24.
    - (VPC only) A range of IPv6 addresses, in CIDR block notation, for example, 2001:db8:1234:1a00::/64.
    - Another security group. This allows instances associated with the specified security group to access instances associated with this security group. This does not add rules from the source security group to this security group. You can specify one of the following security groups:
        - The current security group.
        - EC2-Classic: A different security group for EC2-Classic in the same region.

- EC2-Classic: A security group for another AWS account in the same region (add the AWS account ID as a prefix; for example, 111122223333/sg-edcd9784).
- EC2-VPC: A different security group for the same VPC or a peer VPC in a VPC peering connection.

AMAZON CLOUDWATCH

What is Amazon CloudWatch?

Amazon CloudWatch monitors your Amazon Web Services (AWS) resources and the applications you run on AWS in real time. You can use CloudWatch to collect and track metrics, which are variables you can measure for your resources and applications. CloudWatch alarms send notifications or automatically make changes to the resources you are monitoring based on rules that you define. For example, you can monitor the CPU usage and disk reads and writes of your Amazon EC2 instances and then use this data to determine whether you should launch additional instances to handle increased load. You can also use this data to stop under-used instances to save money. In addition to monitoring the built-in metrics that come with AWS, you can monitor your own custom metrics. With CloudWatch, you gain system-wide visibility into resource utilization, application performance, and operational health.

Getting Started: Installing the CloudWatch Agent on your First Instance

To download and install the CloudWatch agent on a running Amazon EC2 instance, you can use either AWS Systems Manager or the command line. With either method, you must first create an IAM role and attach it to the instance.

1. Attach an IAM role to the Instance

The first procedure creates the IAM role that you need to attach to each Amazon EC2 instance that runs the CloudWatch agent. This role provides permissions for reading information from the instance and writing it to CloudWatch.

The second procedure creates the IAM role that you need to attach to the Amazon EC2 instance being used to create the CloudWatch agent configuration file, if you are going to store this file in

Systems Manager Parameter Store so that other servers can use it. This role provides permissions for writing to Parameter Store, in addition to the permissions for reading information from the instance and writing it to CloudWatch. This role includes permissions sufficient to run the CloudWatch agent as well as to write to Parameter Store.

To create the IAM role necessary for each server to run CloudWatch agent

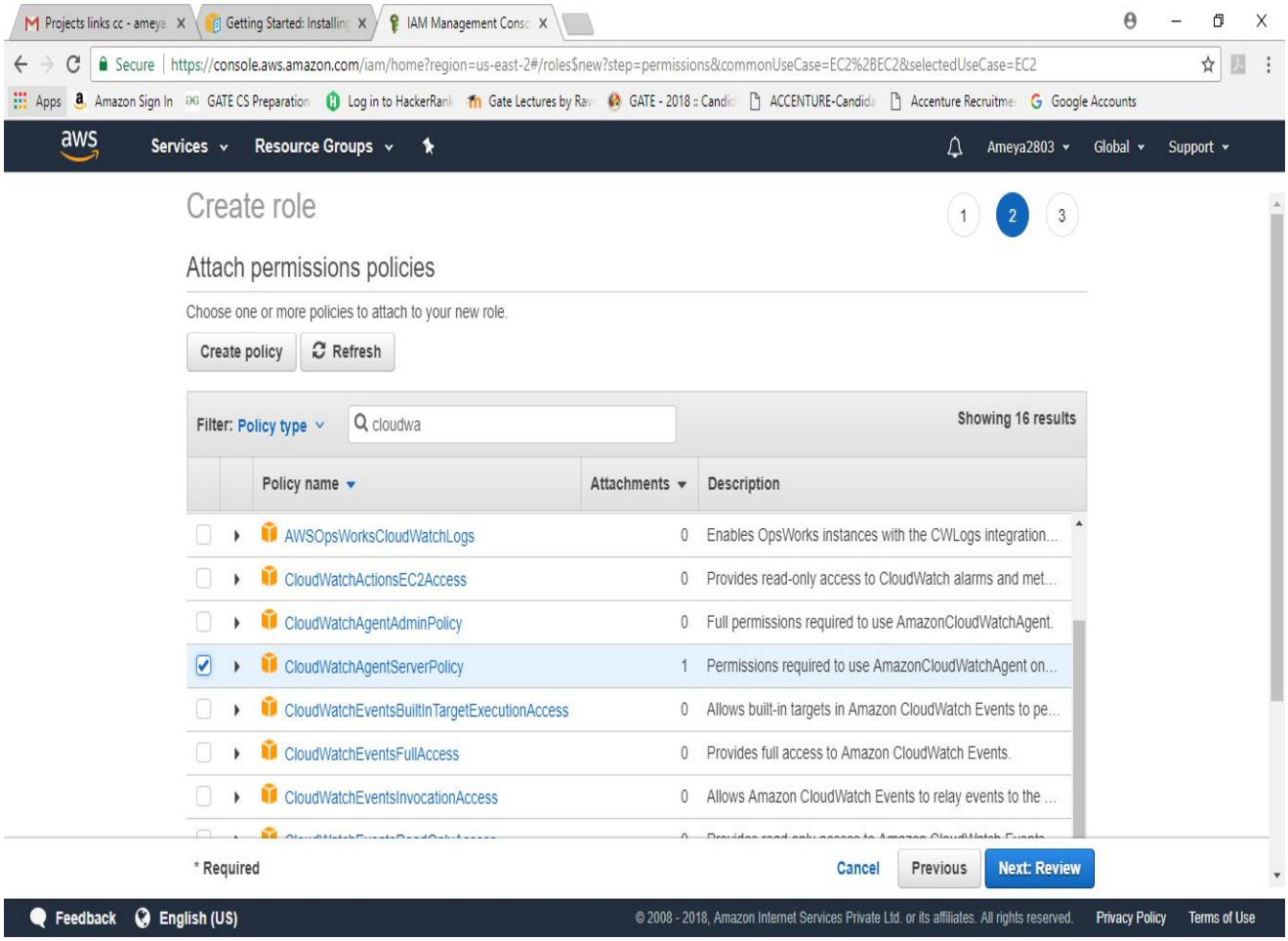
1. Sign in to the AWS Management Console and open the IAM console at https://console.aws.amazon.com/iam/.
2. In the navigation pane on the left, choose **Roles**, **Create role**.
3. For **Choose the service that will use this role**, choose **EC2 Allows EC2 instances to call AWS services on your behalf**. Choose **Next: Permissions**.
4. In the list of policies, select the checkbox next to **CloudWatchAgentServerPolicy**. Use the search box to find the policy, if necessary.
5. If you will use SSM to install or configure the CloudWatch agent, select the check box next to **AmazonEC2RoleforSSM**. Use the search box to find the policy, if necessary. This policy is not necessary if you will start and configure the agent only through the command line.
6. Choose **Next: Review**
7. Confirm that **CloudWatchAgentServerPolicy** and optionally **AmazonEC2RoleforSSM**appear next to **Policies**. In **Role name**, type a name for the role, such as CloudWatchAgentServerRole. Optionally give it a description, and choose **Create role**.
8. The role is now created.

The following procedure creates the IAM role that can also write to Parameter Store. You need to use this role if you are going to store the agent configuration file in Parameter Store so that other servers can use it. This role provides permissions for writing to Parameter Store, in addition to the permissions for reading information from the instance and writing it to CloudWatch. The permissions for writing to Parameter Store provide broad powers, and should not be attached to all your servers, and should be used only by administrators. After you are

finished creating the agent configuration file and copying it to Parameter Store, you should detach this role from the instance and use the **CloudWatchAgentServerPolicy** instead.

To create the IAM role necessary for an administrator to save an agent configuration file to Systems Manager Parameter Store

1. Sign in to the AWS Management Console and open the IAM console at https://console.aws.amazon.com/iam/.
2. In the navigation pane on the left, choose **Roles**, **Create role**.
3. For **Choose the service that will use this role**, choose **EC2 Allows EC2 instances to call AWS services on your behalf**. Choose **Next: Permissions**.
4. In the list of policies, select the check box next to **CloudWatchAgentAdminPolicy**. Use the search box to find the policy, if necessary.
5. If you will use SSM to install or configure the CloudWatch agent, select the check box next to **AmazonEC2RoleforSSM**. Use the search box to find the policy, if necessary. This policy is not necessary if you will start and configure the agent only through the command line.
6. Choose **Next: Review**
7. Confirm that **CloudWatchAgentAdminPolicy** and optionally **AmazonEC2RoleforSSM**appear next to **Policies**. In **Role name**, type a name for the role, such as CloudWatchAgentAdminRole. Optionally give it a description, and choose **Create role**.
8. The role is now created.



Download the CloudWatch Agent Package using Run Command

Systems Manager Run Command enables you to manage the configuration of your instances. You specify a Systems Manager document, specify parameters, and execute the command on one or more instances. The SSM Agent on the instance processes the command and configures the instance as specified.

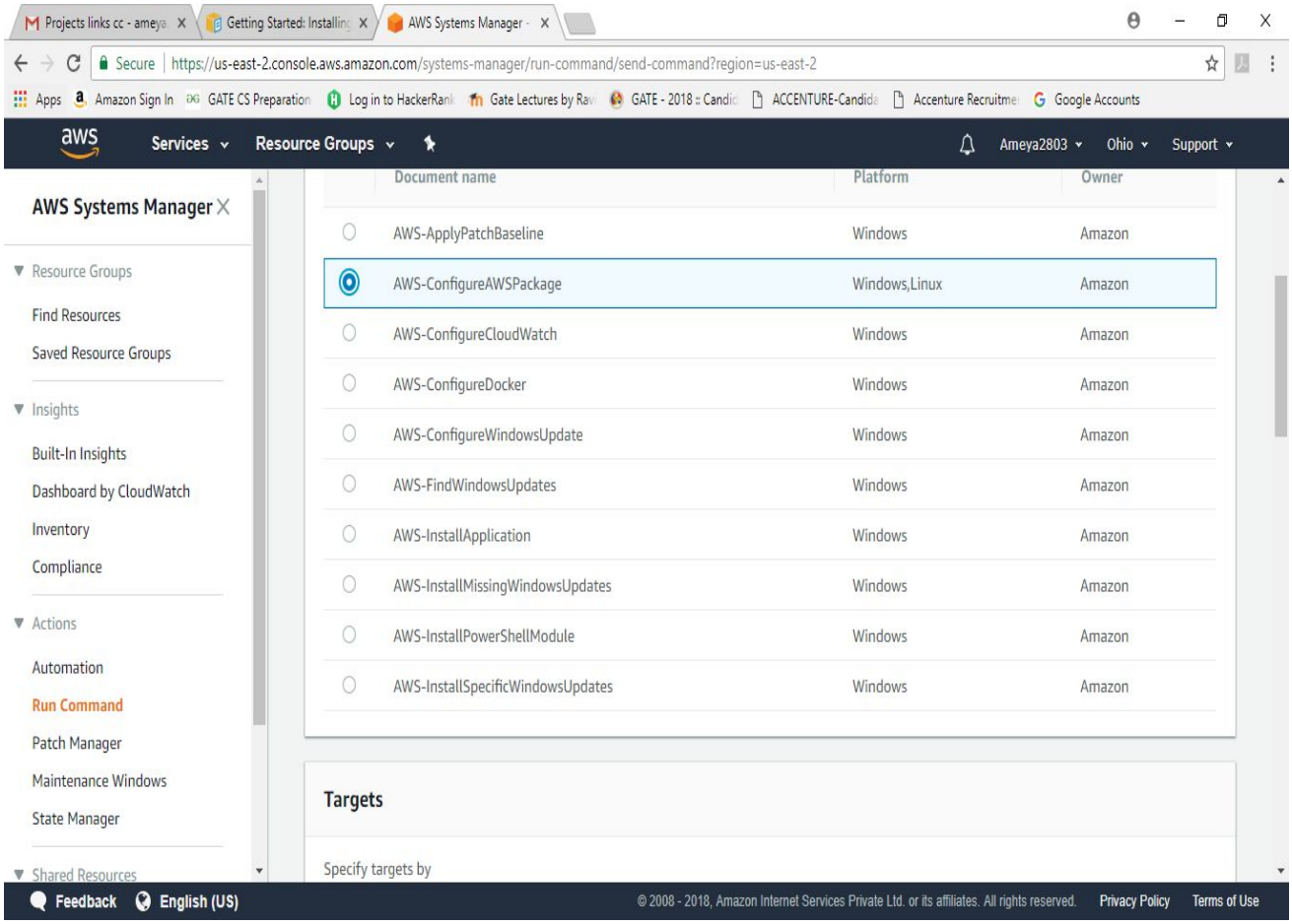
Steps:

1. Open the Systems Manager console at https://console.aws.amazon.com/systems-manager/.
2. In the navigation pane, choose Run Command.

   -or-

If the AWS Systems Manager home page opens, scroll down and choose Explore Run Command.

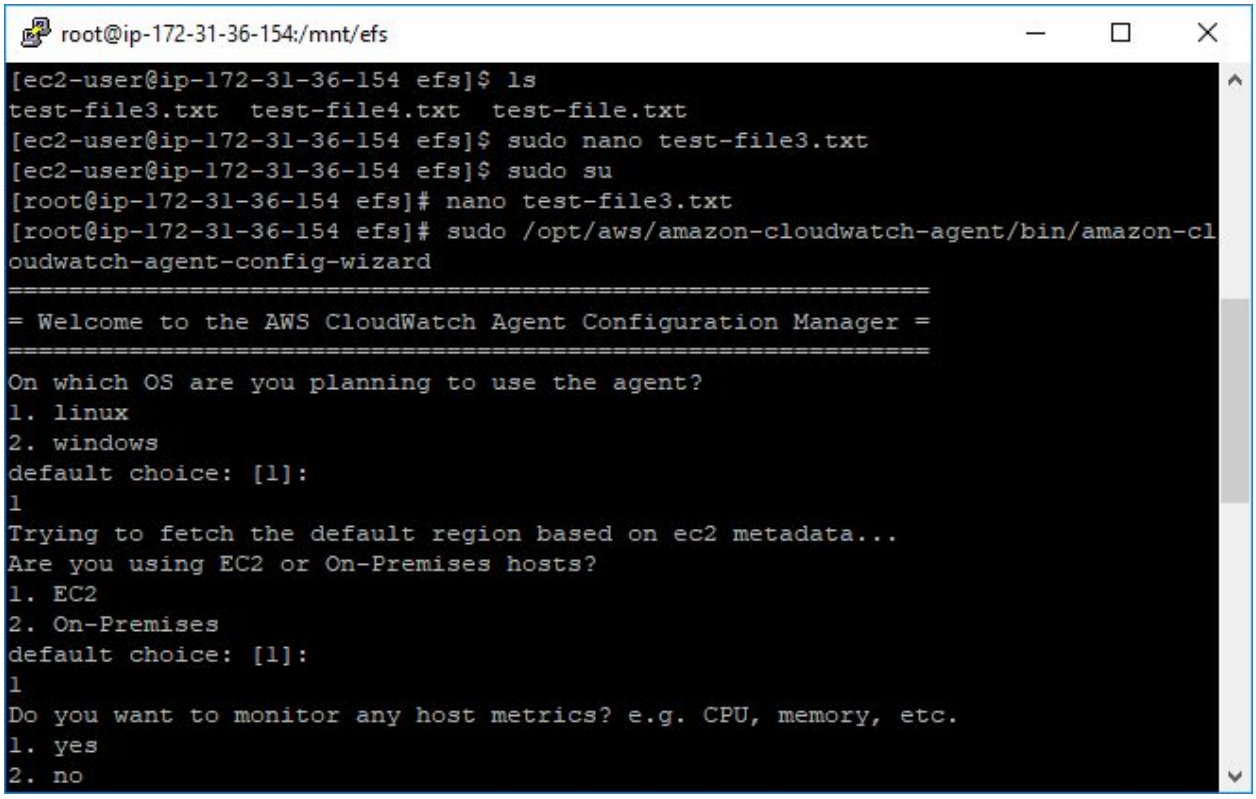
3. Choose Run command.
4. In the Command document list, choose AWS-ConfigureAWSPackage.
5. In the Targets area, choose the instance on which to install the CloudWatch agent. If you do not see a specific instance, it might not be configured for Run Command. For more information, see Systems Manager Prerequisites in the *Amazon EC2 User Guide for Windows Instances*.
6. In the Action list, choose Install.
7. In the Name field, type AmazonCloudWatchAgent.
8. Leave Version set to latest to install the latest version of the agent.
9. Choose Run.
10. Optionally, in the Targets and outputs areas, select the button next to an instance name and choose View output. Systems Manager should show that the agent was successfully installed.

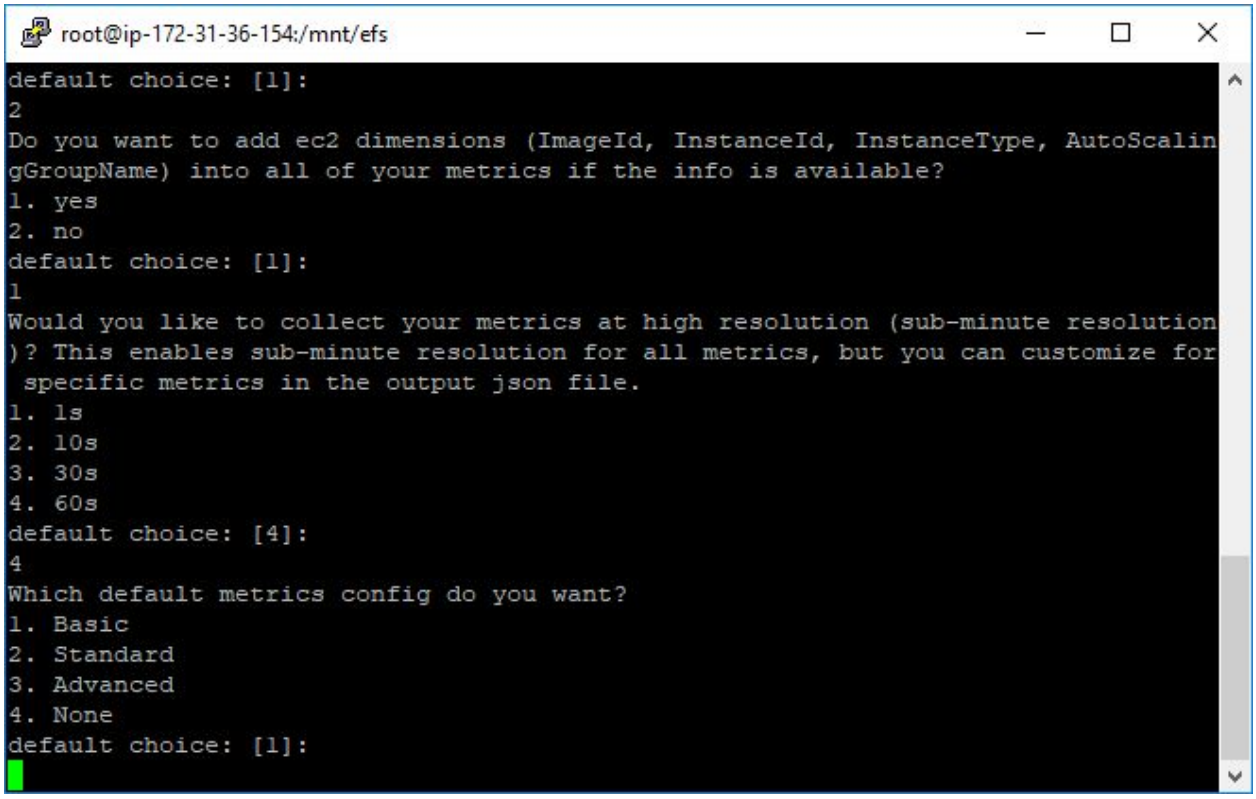


Create the CloudWatch Agent Configuration File with the Wizard:

1. Start the CloudWatch agent configuration wizard by typing the following:
   `sudo /opt/aws/amazon-cloudwatch-agent/bin/amazon-cloudwatch-agent-config-wizard`
2. On a server running Windows Server, type the following:
   `cd "C:\Program Files\Amazon\AmazonCloudWatchAgent"`
   `Amazon-cloudwatch-agent-config-wizard.exe`
3. Answer the questions to customize the configuration file for your server.
4. The agent configuration file wizard, amazon-cloudwatch-agent-config-wizard, asks a series of questions, including the following:
   - Are you installing the agent on an Amazon EC2 instance or an on-premises server?
   - Is the server running Linux or Windows Server?

- Do you want the agent to also send log files to CloudWatch Logs? If so, do you have an existing CloudWatch Logs agent configuration file? If yes, the CloudWatch agent can use this file to determine the logs to collect from the server.
- If you are going to collect metrics from the server, do you want to monitor one of the default sets of metrics, or customize the list of metrics that you collect?
- Are you migrating from an existing SSM Agent?
- If you are going to use Systems Manager to install and configure the agent, be sure to answer **Yes** when prompted whether to store the file in Systems Manager Parameter Store. You can also choose to store the file in Parameter Store even if you aren't using the SSM Agent to install the CloudWatch agent. To be able to store the file in Parameter Store, you must use an IAM role with sufficient permissions.


root@ip-172-31-36-154:/mnt/efs
[ec2-user@ip-172-31-36-154 efs]$ ls
test-file3.txt test-file4.txt test-file.txt
[ec2-user@ip-172-31-36-154 efs]$ sudo nano test-file3.txt
[ec2-user@ip-172-31-36-154 efs]$ sudo su
[root@ip-172-31-36-154 efs]# nano test-file3.txt
[root@ip-172-31-36-154 efs]# sudo /opt/aws/amazon-cloudwatch-agent/bin/amazon-cl
oudwatch-agent-config-wizard
==
= Welcome to the AWS CloudWatch Agent Configuration Manager =
==
On which OS are you planning to use the agent?
1. linux
2. windows
default choice: [1]:
1
Trying to fetch the default region based on ec2 metadata...
Are you using EC2 or On-Premises hosts?
1. EC2
2. On-Premises
default choice: [1]:
1
Do you want to monitor any host metrics? e.g. CPU, memory, etc.
1. yes
2. no


root@ip-172-31-36-154:/mnt/efs
default choice: [1]:
2
Do you want to add ec2 dimensions (ImageId, InstanceId, InstanceType, AutoScalin
gGroupName) into all of your metrics if the info is available?
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]:

Start the CloudWatch Agent on the Amazon EC2 Instance using the Command Line

Follow these steps to use the command line to install the CloudWatch agent on an Amazon EC2 instance.

On a Linux server, type the following if you saved the configuration file in the Systems Manager Parameter Store:

```
sudo /opt/aws/amazon-cloudwatch-agent/bin/amazon-cloudwatch-agent-ctl -a fetch-config -m ec2 -c ssm:configuration-parameter-store-name -s
```

On a Linux server, type the following if you saved the configuration file on the local computer:

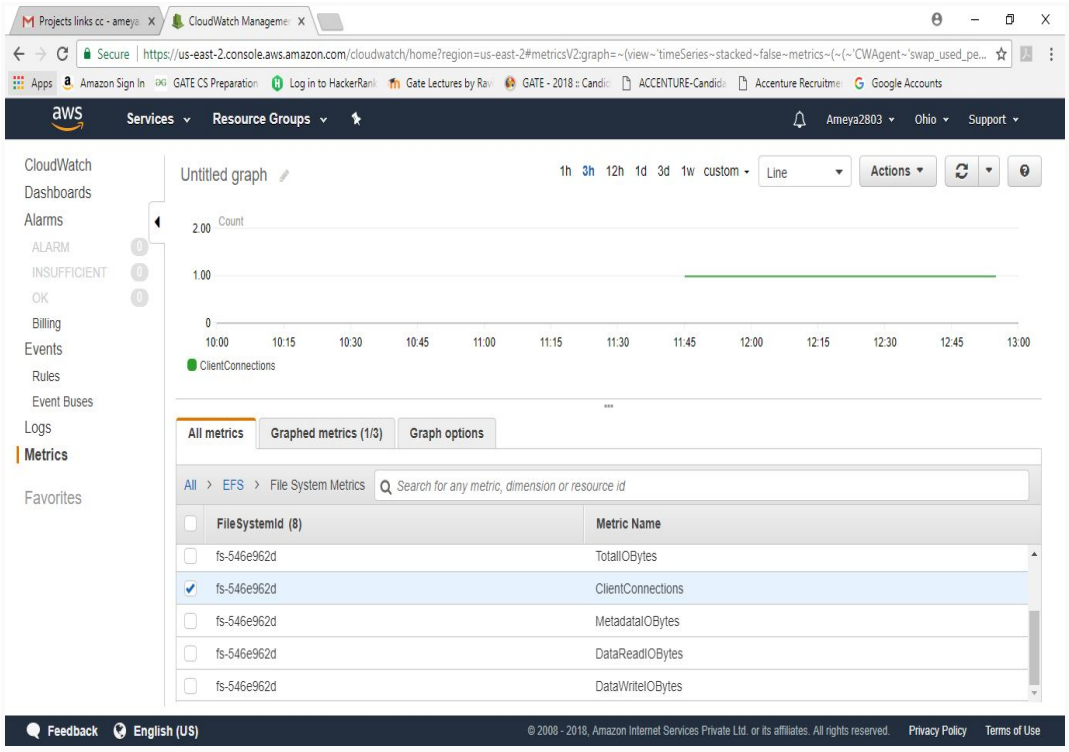
```
sudo /opt/aws/amazon-cloudwatch-agent/bin/amazon-cloudwatch-agent-ctl -a fetch-config -m ec2 -c file:configuration-file-path -s
```

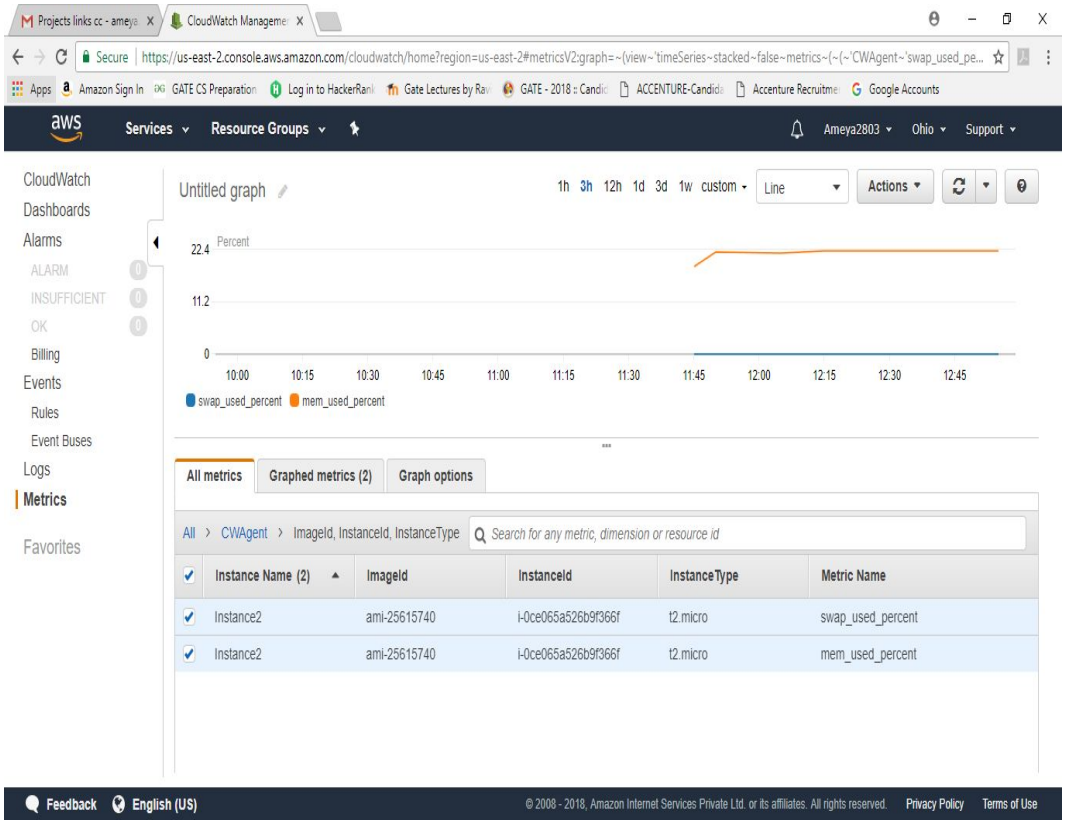
On a server running Windows Server, type the following if you saved the configuration file in Systems Manager Parameter Store:

```
amazon-cloudwatch-agent-ctl.ps1 -a fetch-config -m ec2 -c ssm:configuration-parameter-store-name -s
```

On a server running Windows Server, type the following if you saved the configuration file on the local computer:

```
amazon-cloudwatch-agent-ctl.ps1 -a fetch-config -m ec2 -c file:configuration-file-path -s
```


Projects links cc - ameya
CloudWatch Management
Secure
https://us-east-2.console.aws.amazon.com/cloudwatch/home?region=us-east-2#metricsV2:graph=~(view~'timeSeries~stacked~false~metrics~(~(~'CWAgent~'swap_used_pe...
Apps
Amazon Sign In
GATE CS Preparation
Log in to HackerRank
Gate Lectures by Ravi
GATE - 2018 :: Candidate
ACCENTURE-Candidate
Accenture Recruitment
Google Accounts
aws
Services
Resource Groups
Ameya2803
Ohio
Support
CloudWatch
Dashboards
Alarms
ALARM 0
INSUFFICIENT 0
OK 0
Billing
Events
Rules
Event Buses
Logs
Metrics
Favorites
Untitled graph
1h 3h 12h 1d 3d 1w custom
Line
Actions
Count
2.00
1.00
0
10:00 10:15 10:30 10:45 11:00 11:15 11:30 11:45 12:00 12:15 12:30 12:45 13:00
ClientConnections
All metrics
Graphed metrics (1/3)
Graph options
All > EFS > File System Metrics
Search for any metric, dimension or resource id
FileSystemId (8)
Metric Name
fs-546e962d TotalIOBytes
fs-546e962d ClientConnections
fs-546e962d MetadataIOBytes
fs-546e962d DataReadIOBytes
fs-546e962d DataWriteIOBytes
Feedback
English (US)
© 2008 - 2018, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved.
Privacy Policy
Terms of Use


Projects links cc - ameya
CloudWatch Manageme
Secure
https://us-east-2.console.aws.amazon.com/cloudwatch/home?region=us-east-2#metricsV2:graph=~(view~'timeSeries~stacked~false~metrics~(~(~'CWAgent~'swap_used_pe...
Apps
Amazon Sign In
GATE CS Preparation
Log in to HackerRank
Gate Lectures by Ra
GATE - 2018 :: Candid
ACCENTURE-Candida
Accenture Recruitme
Google Accounts
aws
Services
Resource Groups
Ameya2803
Ohio
Support
CloudWatch
Dashboards
Alarms
ALARM 0
INSUFFICIENT 0
OK 0
Billing
Events
Rules
Event Buses
Logs
Metrics
Favorites
Untitled graph
1h 3h 12h 1d 3d 1w custom
Line
Actions
Percent
22.4
11.2
0
10:00 10:15 10:30 10:45 11:00 11:15 11:30 11:45 12:00 12:15 12:30 12:45
swap_used_percent
mem_used_percent
All metrics
Graphed metrics (2)
Graph options
All > CWAgent > ImageId, InstanceId, InstanceType
Search for any metric, dimension or resource id
Instance Name (2)
ImageId
InstanceId
InstanceType
Metric Name
Instance2 ami-25615740 i-0ce065a526b9f366f t2.micro swap_used_percent
Instance2 ami-25615740 i-0ce065a526b9f366f t2.micro mem_used_percent
Feedback
English (US)
© 2008 - 2018, Amazon Internet Services Private Ltd. or its affiliates. All rights reserved.
Privacy Policy
Terms of Use