

Configuration Manual

MSc Research Project
MSc Cloud Computing

Vaishnavi Waghmare
Student ID: 21172846

School of Computing
National College of Ireland

Supervisor: Prof. Shivani Jaswal

National College of Ireland
MSc Project Submission Sheet
School of Computing



Student Name: Vaishnavi Sarjerao Waghmare
Student ID: 21172846
Programme: MSc Cloud Computing **Year:** 2024
Module: Research Project
Lecturer: Prof. Shivani Jaswal
Submission Due Date: 14th January 2024
Project Title: A Comparative Analysis of Metaheuristic Algorithms for Optimizing Tasks in Serverless Frameworks for MapReduce Applications
Word Count: 500 **Page Count:** 10

I hereby certify that the information contained in this (my submission) is information pertaining to research I conducted for this project. All information other than my own contribution will be fully referenced and listed in the relevant bibliography section at the rear of the project.

ALL internet material must be referenced in the bibliography section. Students are required to use the Referencing Standard specified in the report template. To use other author's written or electronic work is illegal (plagiarism) and may result in disciplinary action.

Signature: Vaishnavi Sarjerao Waghmare

Date: 12th January 2024

PLEASE READ THE FOLLOWING INSTRUCTIONS AND CHECKLIST

Attach a completed copy of this sheet to each project (including multiple copies)	<input type="checkbox"/>
Attach a Moodle submission receipt of the online project submission, to each project (including multiple copies).	<input type="checkbox"/>
You must ensure that you retain a HARD COPY of the project, both for your own reference and in case a project is lost or mislaid. It is not sufficient to keep a copy on computer.	<input type="checkbox"/>

Assignments that are submitted to the Programme Coordinator Office must be placed into the assignment box located outside the office.

Office Use Only	
Signature:	
Date:	
Penalty Applied (if applicable):	

Configuration Manual

Vaishnavi Sarjerao Waghmare
21172846

1 Introduction

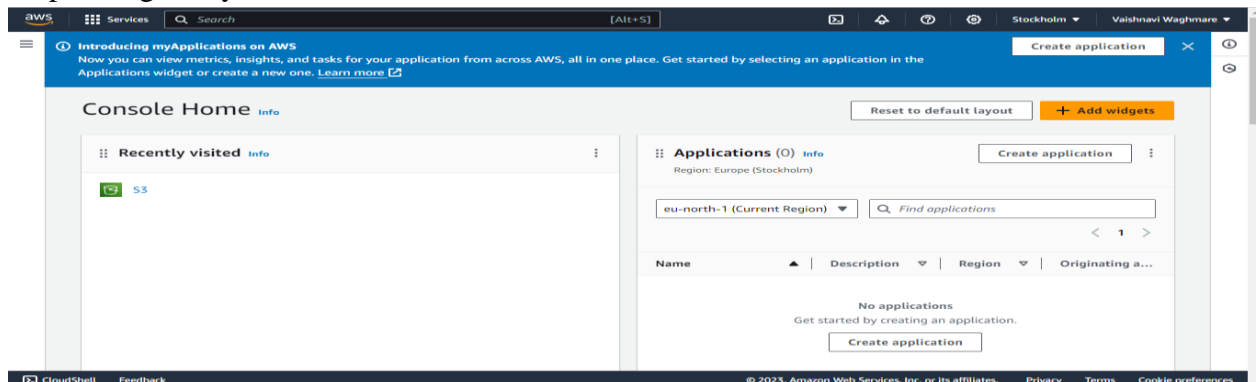
The configuration manual demonstrates running of the integrated code.

2 Requirements

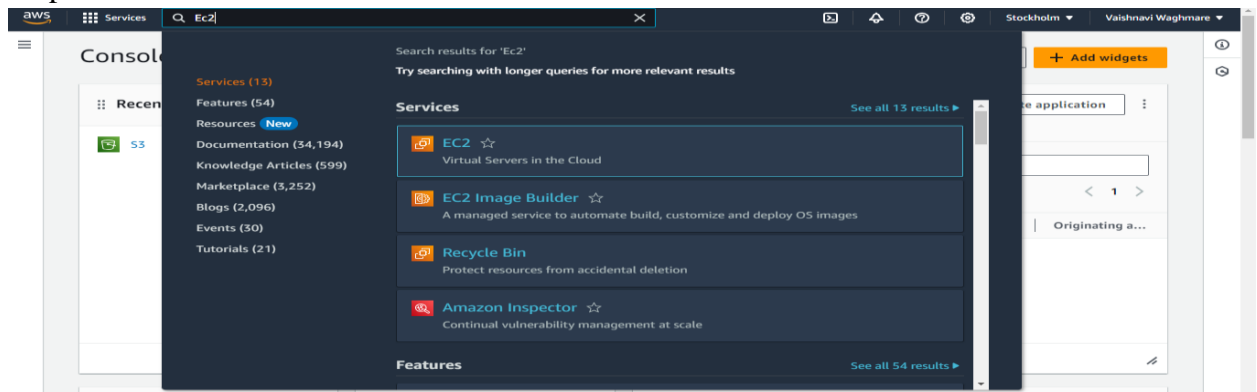
- AWS Account
- Windows/Mac Laptop
- Hadoop Installation
- AWS CLI Installation
- OpenSSH installation

3 Code Implementation

Step 1. Login to your AWS Account



Step 2. Choose EC2 Instance



Step 3. Choose an Instance

The screenshot shows the AWS Management Console interface. On the left, there is a navigation menu with options like 'EC2 Dashboard', 'Instances', 'Images', and 'Elastic Block Store'. The main content area displays the details for the 'c5n.18xlarge' instance type. The 'Details' section includes information about the instance type, family, size, and hypervisor. The 'Compute' section lists features like 'Free-Tier eligible', 'Free Trial available', 'Bare metal', and 'vCPUs'.

Details			
Instance type	c5n.18xlarge	Instance family	c5n
Instance size	18xlarge	Hypervisor	nitro
Auto Recovery support	true	Supported root device types	ebs
Dedicated Host support	true	On-Demand Hibernation support	-
Burstable Performance support			

Compute			
Free-Tier eligible	false	Free Trial available	false
Bare metal	false	vCPUs	72

Step 4. Launch instance

The screenshot shows the 'Launch instance' wizard in the AWS Management Console. The left sidebar contains navigation options. The main area is divided into several sections: 'Instances' (showing 0 instances), 'Key pairs' (0), 'Load balancers' (0), 'Placement groups' (0), 'Security groups' (1), 'Snapshots' (0), and 'Volumes' (0). The 'Launch instance' section has a 'Launch instance' button and a 'Migrate a server' button. The 'Service health' section shows the 'AWS Health Dashboard'. The 'Account attributes' section displays the 'Default VPC' and 'Settings'.

Launch instance

To get started, launch an Amazon EC2 instance, which is a virtual server in the cloud.

Launch instance (button)

Migrate a server (button)

Note: Your instances will launch in the Europe (Stockholm) Region

Service health

AWS Health Dashboard (button)

Region: Europe (Stockholm)

Zones

Zone name	Zone ID
-----------	---------

Account attributes

Default VPC

vpc-05d6ca3599402a381

Settings

- Data protection and security
- Zones
- EC2 Serial Console
- Default credit specification
- Console experiments

Step 5. Choose kernel and instance type

The screenshot shows the 'Choose kernel and instance type' step in the AWS Management Console. The left sidebar contains navigation options. The main area is divided into several sections: 'Amazon Linux 2023 AMI 2023.2.20231113.0 x86_64 HVM kernel-6.1', 'Architecture' (64-bit (x86)), 'AMI ID' (ami-0416c18e75bd69567), and 'Instance type' (t3.2xlarge). The 'Instance type' section shows the 't3.2xlarge' instance type with its specifications and pricing. The 'Summary' section shows the 'Number of instances' (1), 'Software Image (AMI)', 'Virtual server type (instance type)', 'Firewall (security group)', and 'Storage (volumes)'.

Amazon Linux 2023 AMI 2023.2.20231113.0 x86_64 HVM kernel-6.1

Architecture: 64-bit (x86)

AMI ID: ami-0416c18e75bd69567

Instance type

t3.2xlarge

Family: t3 8 vCPU 32 GiB Memory Current generation: true

On-Demand RHEL base pricing: 0.4756 USD per Hour

On-Demand Linux base pricing: 0.3456 USD per Hour

On-Demand SUSE base pricing: 0.4706 USD per Hour

On-Demand Windows base pricing: 0.4928 USD per Hour

Summary

Number of instances: 1

Software Image (AMI)

Amazon Linux 2023 AMI 2023.2.2...read more

ami-0416c18e75bd69567

Virtual server type (instance type)

t3.2xlarge

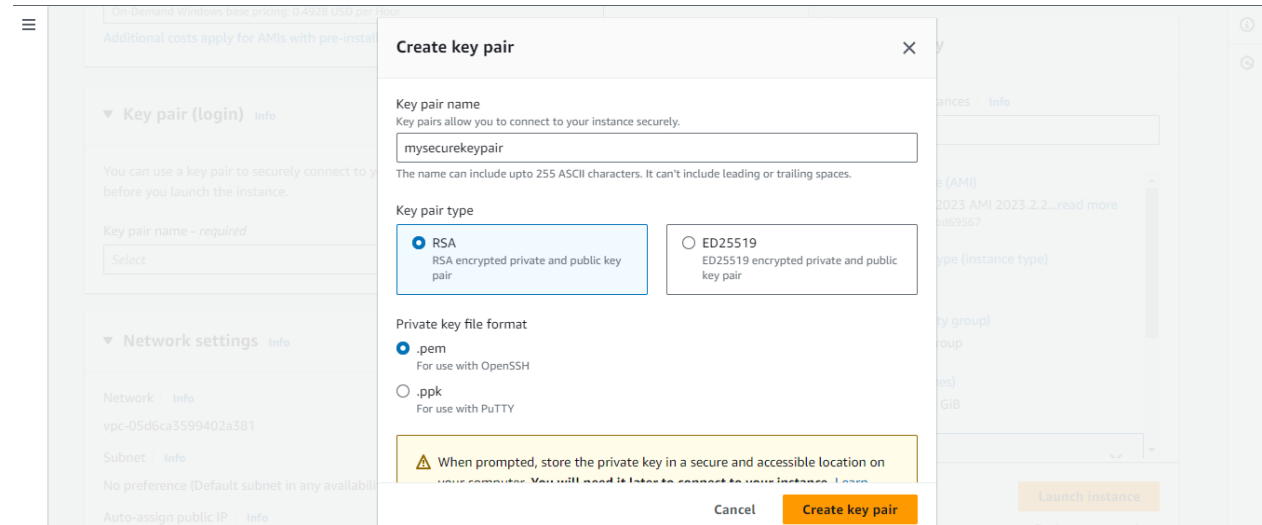
Firewall (security group)

New security group

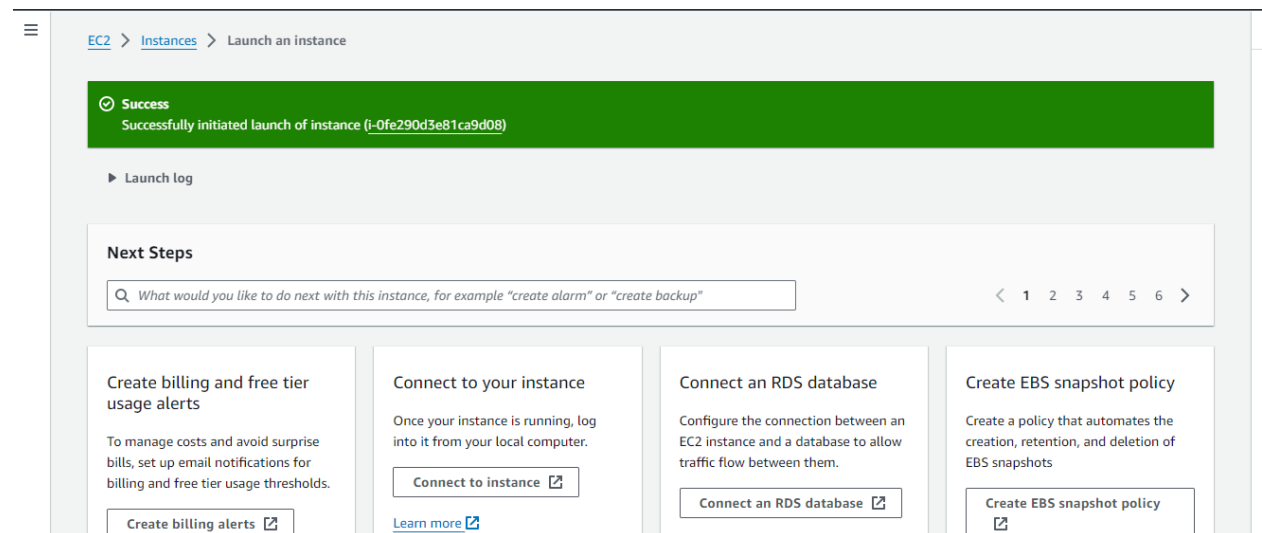
Storage (volumes)

1 volume(s) - 8 GiB

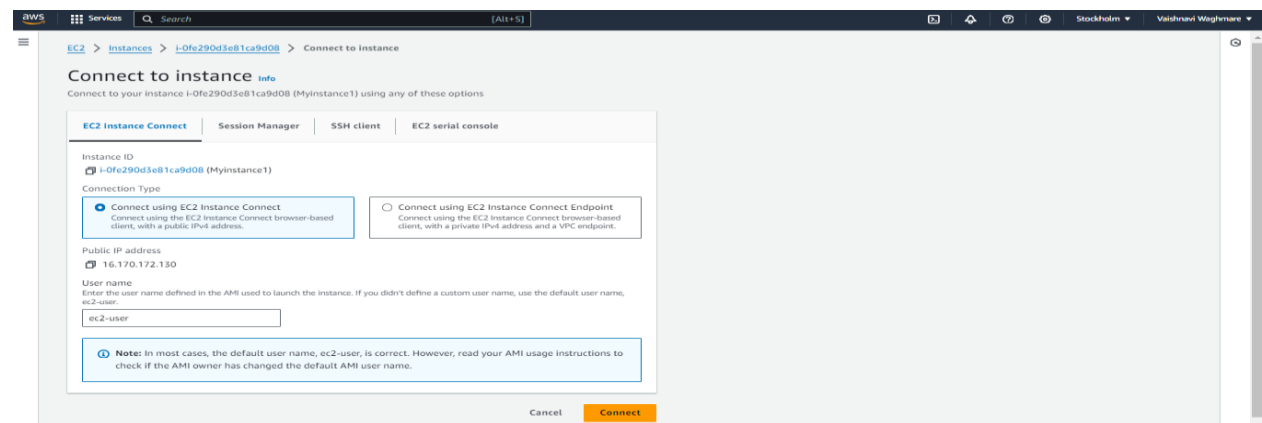
Step 6. Create a key – pair



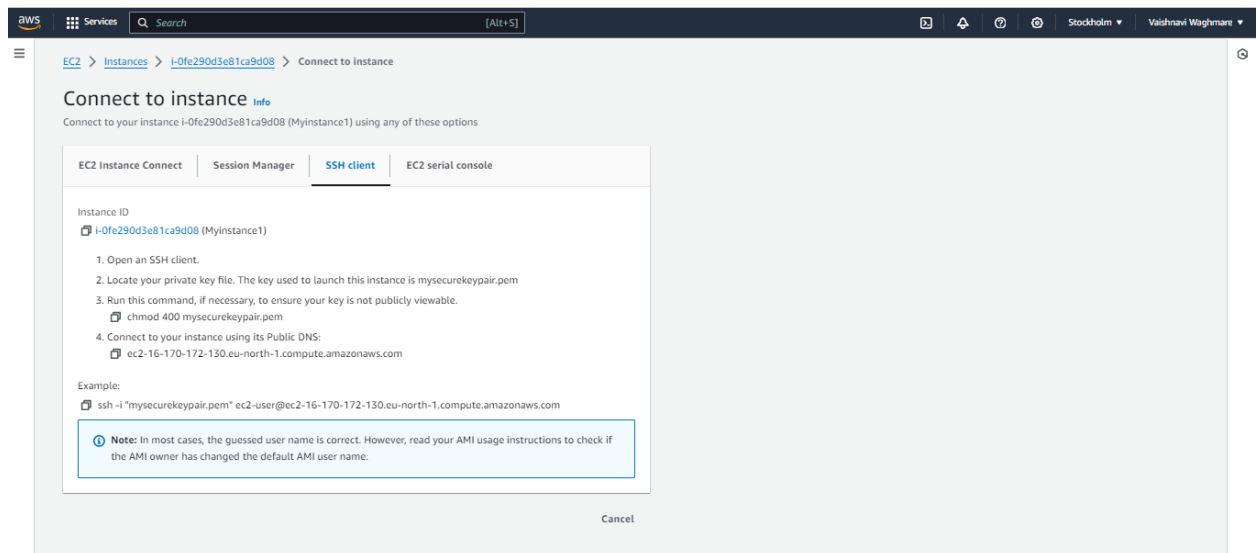
Step 7. Launch Instance



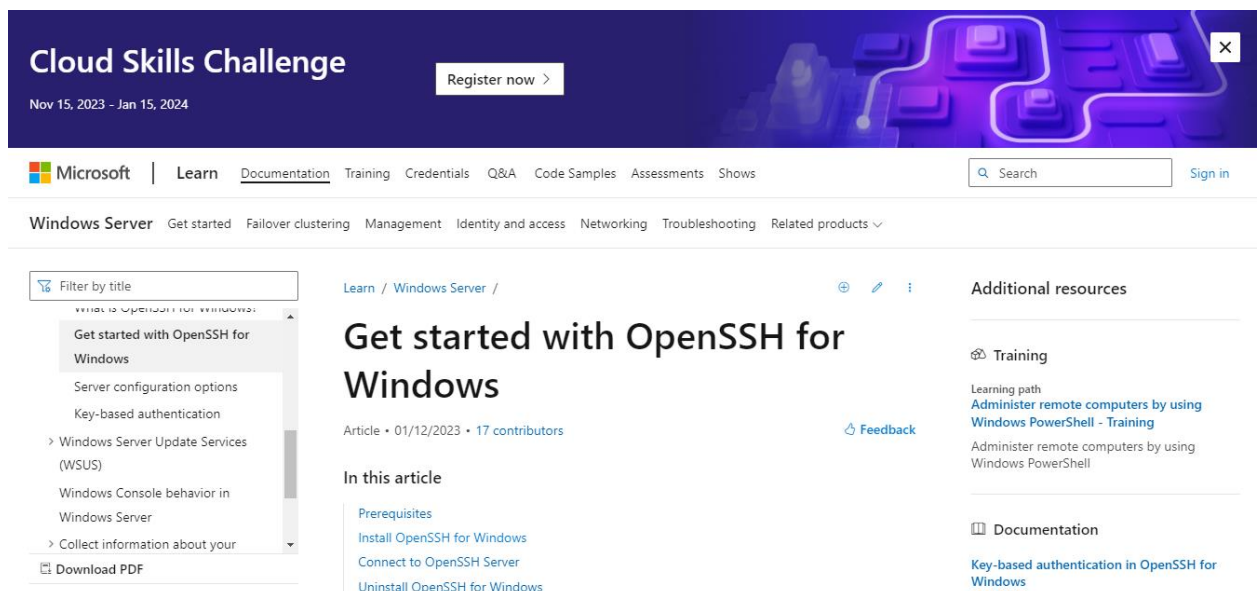
Step 8. Connecting your AWS instance



Step 9. configure sshclient



Step 10. Install OpenSSH client



Step 11. Check for prerequisites

Step 12. Open Powershell as administrator and Check powershell version

```
Windows PowerShell
PS C:\Users\pcc> $PSVersionTable

Name                           Value
----                           -
PSVersion                      5.1.19041.3693
PSEdition                      Desktop
PSCompatibleVersions           {1.0, 2.0, 3.0, 4.0...}
BuildVersion                   10.0.19041.3693
CLRVersion                     4.0.30319.42000
WSManStackVersion              3.0
PSRemotingProtocolVersion      2.3
SerializationVersion           1.1.0.1

PS C:\Users\pcc>
```

Step 13. Check for other prerequisites

```
Administrator: Windows PowerShell

Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

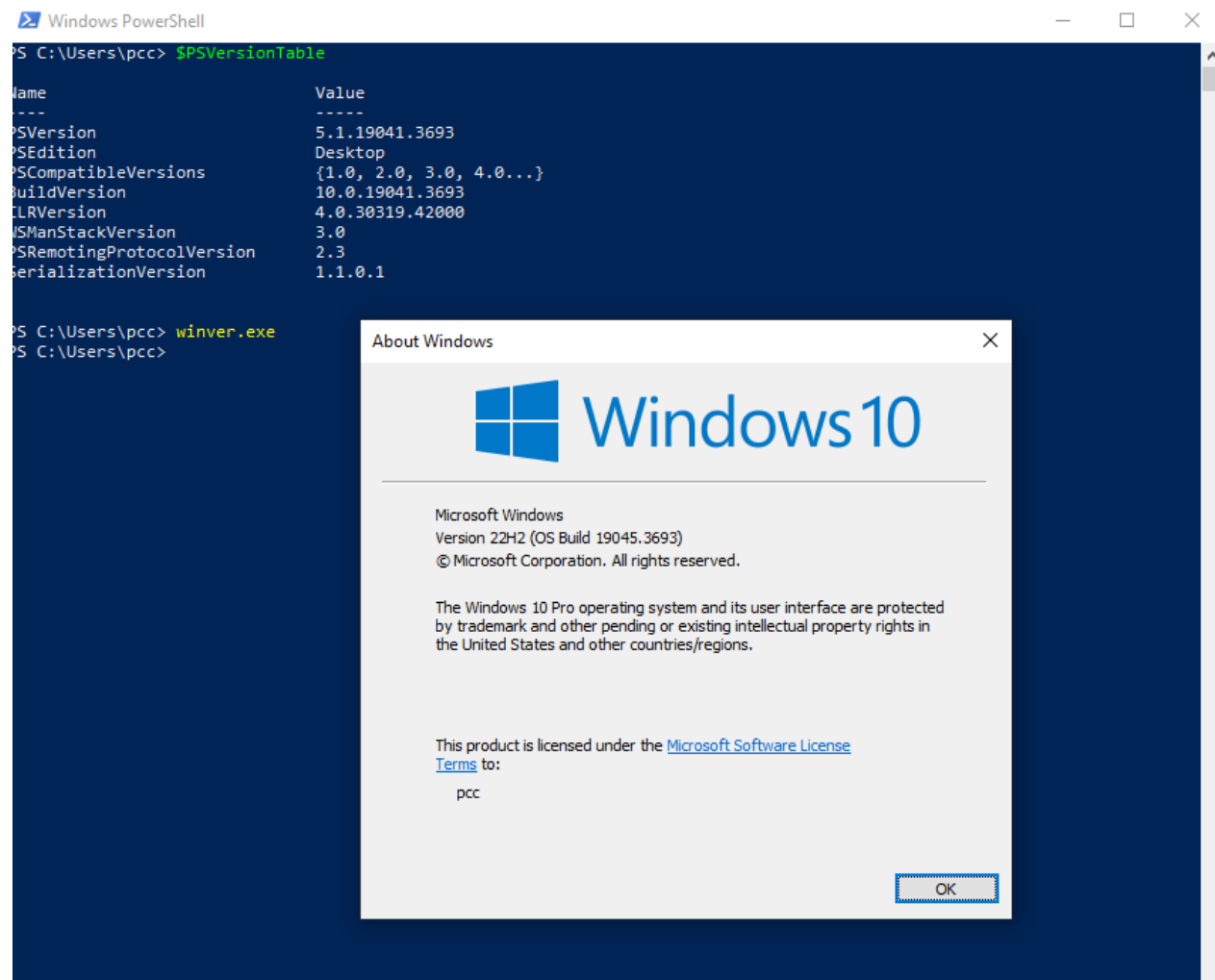
Try the new cross-platform PowerShell https://aka.ms/pscore6

S C:\WINDOWS\system32> (New-Object Security.Principal.WindowsPrincipal([Security.Principal.WindowsIdentity]::GetCurrent
)).IsInRole([Security.Principal.WindowsBuiltInRole]::Administrator)
true
S C:\WINDOWS\system32> $PSVersionTable.PSVersion

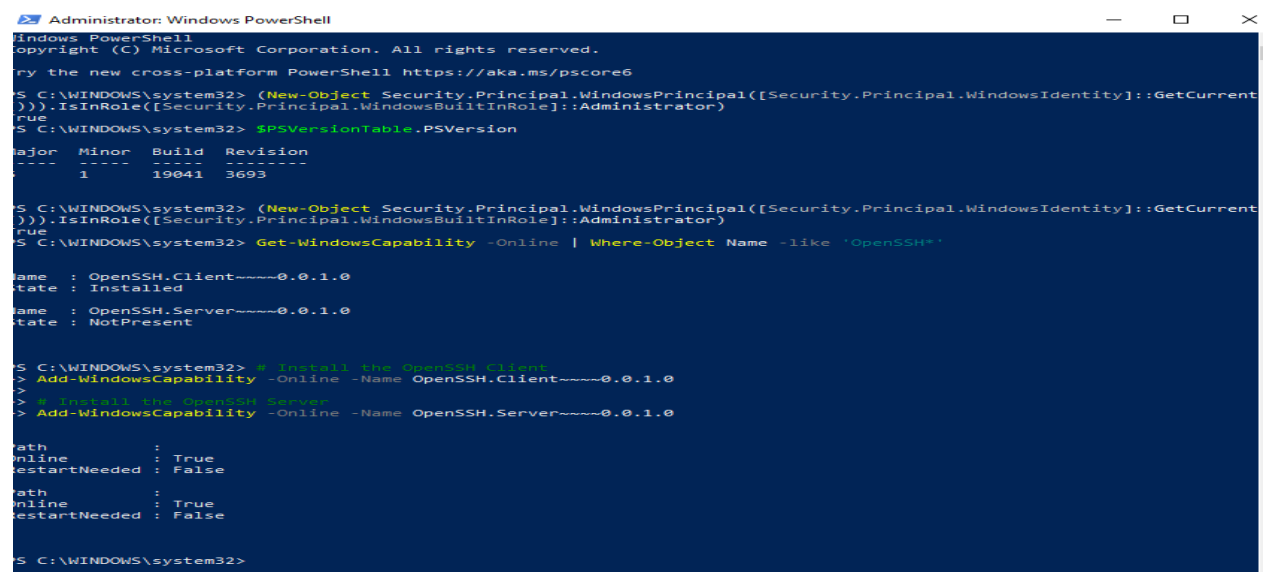
Major Minor Build Revision
-----
5      1      19041 3693

S C:\WINDOWS\system32> (New-Object Security.Principal.WindowsPrincipal([Security.Principal.WindowsIdentity]::GetCurrent
)).IsInRole([Security.Principal.WindowsBuiltInRole]::Administrator)
true
S C:\WINDOWS\system32>
```

Step 14. Check windows version



Step 15. Install OpenSSH



Step 16. Restart the powershell as administrator

Step 17. Configure SSH

```
Administrator: Windows PowerShell
Windows PowerShell
Copyright (C) Microsoft Corporation. All rights reserved.

Try the new cross-platform PowerShell https://aka.ms/powershell

S C:\WINDOWS\system32> Start-Service sshd
S C:\WINDOWS\system32> Set-Service -Name sshd -StartupType 'Automatic'
S C:\WINDOWS\system32> if (!(Get-NetFirewallRule -Name "OpenSSH-Server-In-TCP" -ErrorAction SilentlyContinue | Select-Object Name, Enabled)) {
>     Write-Output "Firewall Rule 'OpenSSH-Server-In-TCP' does not exist, creating it..."
>     New-NetFirewallRule -Name 'OpenSSH-Server-In-TCP' -DisplayName 'OpenSSH Server (sshd)' -Enabled True -Direction Inbound -Protocol TCP -Action Allow -LocalPort 22
> } else {
>     Write-Output "Firewall rule 'OpenSSH-Server-In-TCP' has been created and exists."
> }
Firewall rule 'OpenSSH-Server-In-TCP' has been created and exists.
S C:\WINDOWS\system32>
```

Step 18. Access SSH

[illegible]

Step 19. Verify response if Hadoop is installed in your instance

```

C:\Users\pcch>hdfs namenode -format
Microsoft Windows [Version 10.0.19045.3603]
(c) Microsoft Corporation. All rights reserved.

C:\Users\pcch>hdfs namenode -format
2023-12-16 12:24:07,026 INFO namenode.NameNode: STARTUP_MSG:
/*****
STARTUP_MSG: Starting NameNode
STARTUP_MSG: host = DESKTOP-V4VK0DC/172.18.192.1
STARTUP_MSG: args = [-format]
STARTUP_MSG: version = 3.3.6
STARTUP_MSG: classpath = C:\hadoop-3.3.6\etc\hadoop;C:\hadoop-3.3.6\share\hadoop\common;C:\hadoop-3.3.6\share\hadoop\common\lib\animal-sniffer-annotations-1.17.jar;C:\hadoop-3.3.6\share\hadoop\common\lib\audience-annotations-0.5.0.jar;C:\hadoop-3.3.6\share\hadoop\common\lib\avro-1.7.7.jar;C:\hadoop-3.3.6\share\hadoop\common\lib\checker-qual-2.5.2.jar;C:\hadoop-3.3.6\share\hadoop\common\lib\commons-beanutils-1.9.4.jar;C:\hadoop-3.3.6\share\hadoop\common\lib\commons-cli-1.2.jar;C:\hadoop-3.3.6\share\hadoop\common\lib\commons-codec-1.15.jar;C:\hadoop-3.3.6\share\hadoop\common\lib\commons-collections-3.2.2.jar;C:\hadoop-3.3.6\share\hadoop\common\lib\commons-compress-1.21.jar;C:\hadoop-3.3.6\share\hadoop\common\lib\commons-configuration2-2.8.0.jar;C:\hadoop-3.3.6\share\hadoop\common\lib\commons-daemon-1.0.13.jar;C:\hadoop-3.3.6\share\hadoop\common\lib\commons-io-2.8.0.jar;C:\hadoop-3.3.6\share\hadoop\common\lib\commons-lang3-3.12.0.jar;C:\hadoop-3.3.6\share\hadoop\common\lib\commons-logging-1.1.3.jar;C:\hadoop-3.3.6\share\hadoop\common\lib\commons-math3-3.1.1.jar;C:\hadoop-3.3.6\share\hadoop\common\lib\commons-net-3.9.0.jar;C:\hadoop-3.3.6\share\hadoop\common\lib\commons-text-1.10.0.jar;C:\hadoop-3.3.6\share\hadoop\common\lib\curator-client-5.2.0.jar;C:\hadoop-3.3.6\share\hadoop\common\lib\curator-framework-5.2.0.jar;C:\hadoop-3.3.6\share\hadoop\common\lib\curator-recipes-5.2.0.jar;C:\hadoop-3.3.6\share\hadoop\common\lib\dnsjava-2.1.7.jar;C:\hadoop-3.3.6\share\hadoop\common\lib\fail0safe-core-5.1.0.jar;C:\hadoop-3.3.6\share\hadoop\common\lib\gson-2.9.0.jar;C:\hadoop-3.3.6\share\hadoop\common\lib\guava-27.0-jre.jar;C:\hadoop-3.3.6\share\hadoop\common\lib\hadoop-annotations-3.3.6.jar;C:\hadoop-3.3.6\share\hadoop\common\lib\hadoop-auth-3.3.6.jar;C:\hadoop-3.3.6\share\hadoop\common\lib\hadoop-shaded-guava-1.1.1.jar;C:\hadoop-3.3.6\share\hadoop\common\lib\hadoop-shaded-protobuf-3.7.1-1.1.jar;C:\hadoop-3.3.6\share\hadoop\common\lib\httpclient-4.5.13.jar;C:\hadoop-3.3.6\share\hadoop\common\lib\httpcore-4.4.13.jar;C:\hadoop-3.3.6\share\hadoop\common\lib\j2objc-annotations-1.1.jar;C:\hadoop-3.3.6\share\hadoop\common\lib\jackson-annotations-2.12.7.jar;C:\hadoop-3.3.6\share\hadoop\common\lib\jackson-core-2.12.7.jar;C:\hadoop-3.3.6\share\hadoop\common\lib\jackson-mapper-asl-1.9.13.jar;C:\hadoop-3.3.6\share\hadoop\common\lib\jakarta.activation-api-1.2.1.jar;C:\hadoop-3.3.6\share\hadoop\common\lib\javax-servlet-api-1.0.jar;C:\hadoop-3.3.6\share\hadoop\common\lib\jaxb-api-2.2.11.jar;C:\hadoop-3.3.6\share\hadoop\common\lib\jaxb-impl-2.2.3-1.1.jar;C:\hadoop-3.3.6\share\hadoop\common\lib\jcip-annotations-1.0.1.jar;C:\hadoop-3.3.6\share\hadoop\common\lib\jersey-core-1.19.4.jar;C:\hadoop-3.3.6\share\hadoop\common\lib\jersey-json-1.20.jar;C:\hadoop-3.3.6\share\hadoop\common\lib\jersey-server-1.19.4.jar;C:\hadoop-3.3.6\share\hadoop\common\lib\jersey-servlet-1.19.4.jar;C:\hadoop-3.3.6\share\hadoop\common\lib\jettison-1.5.4.jar;C:\hadoop-3.3.6\share\hadoop\common\lib\jetty-http-9.4.51.v20230217.jar;C:\hadoop-3.3.6\share\hadoop\common\lib\jetty-io-9.4.51.v20230217.jar;C:\hadoop-3.3.6\share\hadoop\common\lib\jetty-security-9.4.51.v20230217.jar;C:\hadoop-3.3.6\share\hadoop\common\lib\jetty-server-9.4.51.v20230217.jar;C:\hadoop-3.3.6\share\hadoop\common\lib\jetty-servlet-9.4.51.v20230217.jar;C:\hadoop-3.3.6\share\hadoop\common\lib\jetty-webapp-9.4.51.v20230217.jar;C:\hadoop-3.3.6\share\hadoop\common\lib\jetty-xml-9.4.51.v20230217.jar;C:\hadoop-3.3.6\share\hadoop\common\lib\jsch-0.1.55.jar;C:\hadoop-3.3.6\share\hadoop\common\lib\jsp-api-2.1.jar;C:\hadoop-3.3.6\share\hadoop\common\lib\jsr305-3.0.2.jar;C:\hadoop-3.3.6\share\hadoop\common\lib\jsr311-api-1.1.1.jar;C:\hadoop-3.3.6\share\hadoop\common\lib\jul-to-slf4j-1.7.36.jar;C:\hadoop-3.3.6\share\hadoop\common\lib\kerb-asm-1.0.1.jar;C:\hadoop-3.3.6\share\hadoop\common\lib\kerb-client-1.0.1.jar;C:\hadoop-3.3.6\share\hadoop\common\lib\kerb-core-1.0.1.jar;C:\hadoop-3.3.6\share\hadoop\common\lib\kerb-crypto-1.0.1.jar;C:\hadoop-3.3.6\share\hadoop\common\lib\kerb-identity-1.0.1.jar;C:\hadoop-3.3.6\share\hadoop\common\lib\kerb-server-1.0.1.jar;C:\hadoop-3.3.6\share\hadoop\common\lib\kerb-simpleclient-1.0.1.jar;C:\hadoop-3.3.6\share\hadoop\common\lib\kerb-util-1.0.1.jar;C:\hadoop-3.3.6\share\hadoop\common\lib\kerby-asm-1.0.1.jar;C:\hadoop-3.3.6\share\hadoop\common\lib\kerby-config-1.0.1.jar;C:\hadoop-3.3.6\share\hadoop\common\lib\kerby-pkix-1.0.1.jar;C:\hadoop-3.3.6\share\hadoop\common\lib\kerby-util-1.0.1.jar;C:\hadoop-3.3.6\share\hadoop\common\lib\kerby-xdr-1.0.1.jar;C:\hadoop-3.3.6\share\hadoop\common\lib\listenablefuture-9999.0-empty-to-avoid-conflict-with-guava.jar;C:\hadoop-3.3.6\share\hadoop\common\lib\metrics-core-3.2.4.jar;C:\hadoop-3.3.6\share\hadoop\common\lib\netty-all-4.1.89.Final.jar;C:\hadoop-3.3.6\share\hadoop\common\lib\netty-buffer-4.1.89.Final.jar;C:\hadoop-3.3.6\share\hadoop\common\lib\netty-codec-4.1.89.Final.jar;C:\hadoop-3.3.6\share\hadoop\common\lib\netty-codec-dns-4.1.89.Final.jar;C:\hadoop-3.3.6\share\hadoop\common\lib\netty-codec-haproxy-4.1.89.Final.jar;C:\hadoop-3.3.6\share\hadoop\common\lib\netty-codec-http-4.1.89.Final.jar;C:\hadoop-3.3.6\share\hadoop\common\lib\
*****/

```

Step 20. Configure your Hadoop on our Ec2 instance and run the .sh functions

```

1  #!/bin/bash
2
3  # Run Teragen job
4  yarn jar $HADOOP_HOME/share/hadoop/mapreduce/hadoop-mapreduce-examples-3.3.6.jar teragen 500000 ./I
5
6  # Check if the Teragen job was successful
7  if [ $? -eq 0 ]; then
8  | echo "Teragen Hadoop job completed successfully."
9  else
10 | echo "Error: Teragen Hadoop job failed."
11 | exit 1
12 fi
13
14
15 # Run Terasort job
16 yarn jar $HADOOP_HOME/share/hadoop/mapreduce/hadoop-mapreduce-examples-3.3.6.jar terasort ./INPUT_I
17
18 # Check if the Terasort job was successful
19 if [ $? -eq 0 ]; then

```

Step 21. Execute the script successfully

```
2023-12-18 10:43:09,815 INFO mapreduce.Job: Job job_local460332852_0001 running in uber mode : false
2023-12-18 10:43:09,816 INFO mapreduce.Job: Map 100% reduce 100%
2023-12-18 10:43:09,817 INFO mapreduce.Job: Job job_local460332852_0001 completed successfully
2023-12-18 10:43:09,824 INFO mapreduce.Job: Counters: 30
    File System Counters
      FILE: Number of bytes read=101344528
      FILE: Number of bytes written=1846222
      FILE: Number of read operations=0
      FILE: Number of large read operations=0
      FILE: Number of write operations=0
    Map-Reduce Framework
      Map input records=500000
      Map output records=3
      Map output bytes=81
      Map output materialized bytes=93
      Input split bytes=113
      IO_ERROR=0
      WRONG_LENGTH=0
      WRONG_MAP=0
      WRONG_REDUCE=0
    File Input Format Counters
      Bytes Read=5039636
    File Output Format Counters
      Bytes Written=35
Teravallidate Hadoop job completed successfully.
All Hadoop jobs completed successfully
[ec2-user@ip-172-31-31-158 ~]$ nano run_benchmarking.sh
[ec2-user@ip-172-31-31-158 ~]$ ./client_run.py send disconnect: Connection reset
PS C:\Users\poc\Desktop> cd ..\SanaVaishanavi\Code
```