CS205: Computing Foundations for Computational Science, Spring 2020

IACS INSTITUTE FOR APPLIED COMPUTATIONAL SCIENCE AT HARVARD UNIVERSITY

HARVARD School of Engineering and Applied Sciences

Guide: MPI on AWS

Ignacio M. Llorente, Simon Warchol

v3.0 - 4 February 2021

Abstract

This is a guideline document to show the necessary actions to set up and use a MPI cluster on AWS Ubuntu (18.04) instances. It includes its configuration on a single node for prototyping and debugging, and its deployment on multiple nodes for production.

Requirements

- **First you should have followed the Guide "First Access to AWS"**. It is assumed you already have an AWS account and a key pair, and you are familiar with the AWS EC2 environment.
- We strongly recommend you use the same instance type used for the shared memory (OpenMP) infrastructure guide so you can compare performance results. The results in this guide have been obtained on a **t2.2xlarge** instance with 8 vCPUs, which is the instance type recommended in the homework assignment.
- The files needed to do the exercises are available for download from **Canvas**.
- There are a wide variety of MPI implementations out there. You are free to use any implementation you wish. **MPICH** is a widely-used implementation of MPI that is developed primarily by Argonne National Laboratory in the United States.

Acknowledgments

The author is grateful for constructive comments and suggestions from David Sondak, Charles Liu, Matthew Holman, Keshavamurthy Indireshkumar, Kar Tong Tan, Zudi Lin,Nick Stern, and Dylan Randle.

In this lab, we will install an implementation of MPI (Message Passing Interface) onto a single EC2 instance. MPI is used for parallel computing.

Later on in the course, we'll install it to run programs in parallel across multiple hosts. However, in this case, we'll use it to run multiple processes on a single host.

0. Launch a VM

1. Using Lab 1 as a reference, please spin up a **t2.2xlarge** EC2 instance running Ubuntu 18.04, with your CS205-key
2. Connect to this instance w/ `ssh` (or putty on Windows), again using your CS205-key.

1. MPI in a Single Node

Here we install the MPI library within a single system. This is used for prototyping, development, debugging, and testing.

All of the following instructions should be executed on the EC2 instance, so make sure you're using the terminal window you SSHed in from or are in putty.

- Install `mpich`

```
$ sudo apt-get update
```

```
$ sudo apt-get install libcr-dev mpich mpich-doc
```

- To check the `mpich` installation is successful run following command in the terminal

```
$ mpiexec --version
```

- Upload to the VM the `mpi_sc.c` code.

The mpi_sc.c file is available on canvas.

Remember to scp from your local machine to the host, you'll need to be in another terminal window that isn't already SSHed into the EC2 instance.

```
$ scp -i ~/.ssh/cs205-key.pem mpi_sc.c
ubuntu@ec2-18-218-17-64.us-east-2.compute.amazonaws.com:
```

Windows users please follow the instructions in part 5 of Lab 1 to copy files to your instance.

- Compile it with `mpicc`, and run the code with multiple processes (on the same host).

```
$ mpicc mpi_sc.c -o mpi_sc
$ mpirun -np 2 ./mpi_sc
Hello world from process 0 of 2
Hello world from process 1 of 2
```

Horray, you’re now running two processes in parallel on the same host. Note that if you do

```
$ mpirun -np 2 ./mpi_sc
```

A few more times, the order that the processes print may change i.e.

```
Hello world from process 1 of 2
Hello world from process 0 of 2
```

as they are running in parallel and not sequentially.

Stop your instances when are done for the day to avoid incurring charges
Terminate them when you are sure you are done with your instance