Building the AiB KafkaBlockchain AMI on Amazon Web Services

- Login to the AWS console.
- Select the N. Virginia AWS region.
- Create VPC.
    - Navigate to the VPC service.
    - Launch the VPC wizard.
    - Select *VPC with a Single Public Subnet*.
    - IPv4 CIDR Block: *10.0.0.0/16*
    - VPC name: *KafkaBlockchain*
    - Otherwise accept the defaults for VPC with a Single Public Subnet
    - Click Create VPC
- Navigate to the EC2 service to Launch an Instance.
- Select the *Ubuntu Server 18.04 LTS (HVM), SSD Volume Type 64-bit (x86)* AMI
- Select the *t2.xlarge General Purpose instance type* - having 4 vCPUs and 16 GB RAM.
- Configure Instance details.
    - Network: *KafkaBlockchain*
    - Auto-assign Public IP: *Enable*
    - Otherwise accept the defaults for Instance Details.
- Add Storage.
    - Size (GiB): *30*.
    - Otherwise accept the defaults for Add Storage.
- Tag Spot Request.
    - Skip this.
- Configure Security Group.
    - Security group name: *KafkaBlockchain*
    - Description: *KafkaBlockchain security group*
    - Otherwise accept the defaults and ignore Warning for Configure Security Group.
- Review Instance Launch.
    - accept the summary and click Launch.
- Launch.
    - Choose to Create a new key pair.
    - Key pair name: *KafkaBlockchain*
    - Download the Key Pair to the X509 directory on the development workstation.
    - Launch the instance, and navigate to the link indicated by the text "The following instance launches have been initiated:".
    - Optionally edit the instance name to: *KafkaBlockchain*
- Wait until the Instance State is "running", and Status Checks indicate "2/2…".
- Perform the following steps from a terminal session in the developer's workstation.
- `$ chmod 400 X509/KafkaBlockchain.pem`

- `$ ssh -i "X509/KafkaBlockchain.pem" ubuntu@ec2-54-174-115-225.compute-1.amazonaws.com` (Public DNS (IPv4))
- `$ sudo apt update`
- `$ sudo apt upgrade -y`
    - Keep the local version currently installed (if prompted)
- `$ sudo reboot` (closes ssh session, then afterwords wait a minute or so for the instance to restart with the apt software updates)
- `$ ssh -i "X509/KafkaBlockchain.pem" ubuntu@ec2-54-174-115-225.compute-1.amazonaws.com`
- Install Java JDK 14
    - `$ wget -qO - https://adoptopenjdk.jfrog.io/adoptopenjdk/api/gpg/key/public | sudo apt-key add -`
    - `$ sudo add-apt-repository --yes https://adoptopenjdk.jfrog.io/adoptopenjdk/deb/`
    - `$ sudo apt update`
    - `$ sudo apt-get install -y adoptopenjdk-14-hotspot`
    - `$ java --version`

      ```
      openjdk 14.0.1 2020-04-14
      OpenJDK Runtime Environment AdoptOpenJDK (build 14.0.1+7)
      OpenJDK 64-Bit Server VM AdoptOpenJDK (build 14.0.1+7, mixed mode, sharing)
      ```
- Following instructions from Kafka Quickstart… https://kafka.apache.org/quickstart
    - `$ curl --output kafka_2.12-2.5.0.tgz http://apache.mirrors.pair.com/kafka/2.5.0/kafka_2.12-2.5.0.tgz`
    - `$ tar -xzf kafka_2.12-2.5.0.tgz`
- Clone the KafkaBlockchain library from GitHub
    - `$ mkdir git`
    - `$ cd git`
    - `$ git clone https://github.com/ai-coin/KafkaBlockchain.git`
- Install Maven and compile the KafkaBlockchain source code, and run the unit tests.
    - `$ sudo apt install -y maven`
    - `$ cd ~/git/KafkaBlockchain`
    - `$ mvn install`
- In the AWS console, stop the running instance.
- Create the KafkaBlockchain AMI from the stopped EC2 instance. https://docs.aws.amazon.com/toolkit-for-visual-studio/latest/user-guide/tkv-create-ami-from-instance.html
    - Image name: *KafkaBlockchain*
    - Image description: *KafkaBlockchain*
    - Click Create Image.
    - Click view pending image to await the completion of the KafkaBlockchain AMI.
- Navigate to the AMIs page and ensure that the KafkaBlockchain AMI is listed.