e-ISSN: 2582-5208
International Research Journal of Modernization in Engineering Technology and Science
(Peer-Reviewed, Open Access, Fully Refereed International Journal)
Volume:04/Issue:04/April-2022 Impact Factor- 6.752 www.irjmets.com

SUPPORT CLIQUE CONSENSUS FOR BESU DLT NETWORK ON HYBERLEDGER LABS

Roshan Raut[*1], Kedar Kutuskar[*2], Aadesh Jadhav[*3], Atharva Dhamke[*4], Prof. Satish Kamble[*5]

[*1,2,3,4]Student, DPT of Information Technology, PVGCOET, Pune, Maharashtra, India

[*5] Professor, DPT of Information Technology, PVGCOET, Pune, Maharashtra, India

ABSTRACT

Blockchain Automation Framework is an existing open source tool to deploy different Distributed Ledger Technology (DLT) platform automatically on a given Kubernetes cluster. BAF support multi-cloud and multi-DLT deployments and have a support of DLT network like Hyperledger (HL) Fabric, HL Besu, Quorum, R3 Corda. For HL Besu, previously only IBFT2 Consensus mechanism was supported by BAF. The task is to add a support for the Clique consensus for HL Besu, so that BAF can be used to deploy and operate a HL Besu network with Clique consensus. This will also include upgrading BAF to support the latest stable Besu version.

Keywords: Clique, Consensus, Hyperledger, Blockchain Automation Framework, Hyperledger, Blockchain.

I. INTRODUCTION

Besu implements the Clique, IBFT2.0, and QBFT Proof of Authority (POA) consensus protocols. POA consensus protocols works when participants know each other and there is a level of trust between them, For example, in a permissioned consortium network. POA consensus protocols have faster block times and a much greater transaction throughput than the Ethash Proof of Work (POW) consensus protocol used on the Ethereum Main Net. In Clique, IBFT2.0, or QBFT, a group of nodes in the network act as signers (Cique) or validators (IBST2.0 and QBFT). The existing nodes in the signers/validators pool vote to add nodes to or remove nodes from the pool.

Properties to consider while comparing Clique, IBFT2.0 and QBFT are: Immediate finality, minimum number of validators required to keep the network active, Liveness, Speed.

Immediate Finality:

IBFT2.0 and QBFT have immediate finality, there are no forks and all valid blocks get included in the main chain. Clique consensus does not have immediate finality. The implementation using the Clique consensus must be aware of forks and the occurrence of chain reorganizations.

Minimum Validators:

Byzantine Fault Tolerant (BFT) is the ability to function correctly and reach consensus despite nodes failing or propagating incorrect information to peers. To be Byzantine Fault Tolerant, IBFT2.0 and QBFT require minimum of four validators while clique can operate with a single validator but operating with a single validator offers no redundancy if the validator fails.

Liveness:

Clique is more fault-tolerant than IBFT2.0 and QBFT. Clique consensus can still work even if up to half of the validators fail. IBFT2.0 and QBFT networks require more than or equal to two-thirds of validators nodes to be in operation to create blocks.

Speed:

Reaching consensus and adding blocks to the network is faster in Clique consensus. For Clique the probability of a fork increases as the number of validators increases.

II. METHODOLOGY

Followed the “Code-Test-Document” methodology. The first step while developing any software is to design the process flow by understanding the inner architecture of Besu DLT network and Blockchain Automation Framework. During the initial phase of development, did some planning of the implementation and set the important milestones for the same as represented in the schedule for the work of the respective clique consensus feature. Once developed each feature, Tested the implementation of code to make sure whatever wrote is working correctly. At the end of the testing, documented the entire implementation. Below are the important tasks set to add the Clique consensus feature for the BAF.

Understand the concepts of Besu and BAF

The architecture of BAF and Besu includes the use of Hashicorp Vault, Kubernetes, Docker, Helm, Ansible and Cloud platforms.

Setup local environment for Development

Installed Docker, Vault, Minikube, Helm, Ansible, Git, Linux machine on virtual box.

Setup GKE environment for Development

Minikube creates memory issues on the local machines so shifted the entire development environment to the google cloud platform.

Complete local Besu network with clique consensus

Manually performed the deployment step of the local Besu DLT network.

Setup a small Besu DLT network using local/GKE Kubernetes network using BAF

Created the three validator nodes and set up the small Besu network on GKE using Bevel.

Created the Helmcharts required to setup clique consensus with Besu

Helm is used as a package manager for K8s. Helm charts are configuration files designed for K8s to help define, install and upgrade complex K8s applications. BAF used Helm Charts for designing and configuring the architecture of each DLT/Blockchain platform for its network set-up.

Wrote Ansible scripts to automate the generation of helm value files and deployment of Clique-Besu

Ansible playbooks were designed to automate each of the manual deployment steps, and the roles make the whole DLT/Blockchain network set up to happen automatically defined in the playbooks in a specific order.

Testing the scripts and successfully deploying Besu network using Clique on GKE cluster

Blackbox testing pattern was used to test the Besu network deployment using Clique consensus with all the pods successfully deployed on the Kubernetes cluster and with all the roles working properly without any failure.

III. MODELING AND ANALYSIS

Below is the activity diagram for the Besu Network Deployment using the Clique consensus mechanism.

e-ISSN: 2582-5208
International Research Journal of Modernization in Engineering Technology and Science
(Peer-Reviewed, Open Access, Fully Refereed International Journal)
Volume:04/Issue:04/April-2022 Impact Factor- 6.752 www.irjmets.com

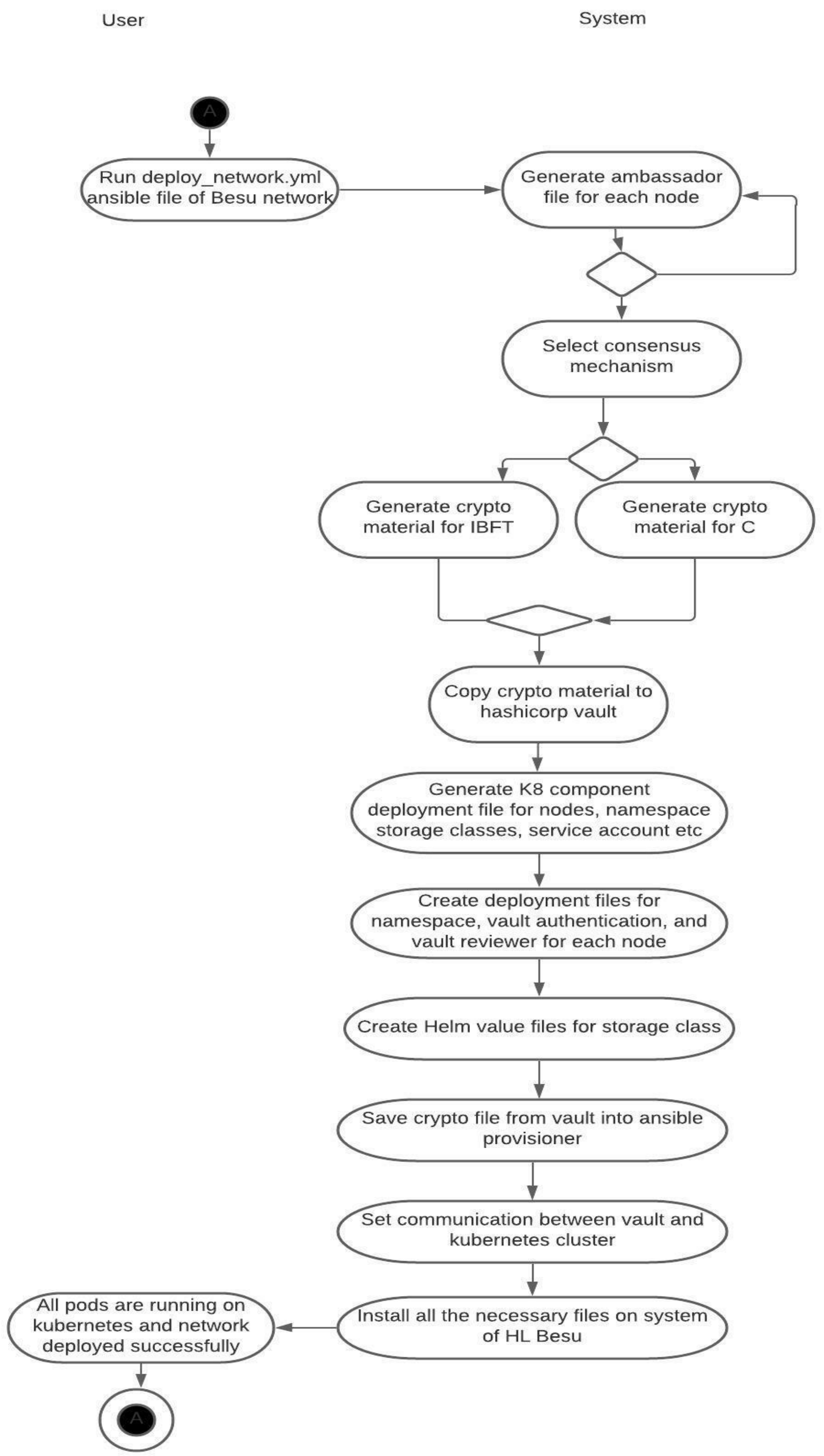

Figure 1: Activity Diagram of Besu Network Deployment using Clique Consensus.

IV. RESULTS AND DISCUSSION

Adding the Clique consensus mechanism for the HL Besu DLT Blockchain network increased the Liveness and Speed. Improved the Byzantine Fault Tolerance of the Besu DLT Network, Tested the complete BAF code on GCP and successfully deployed the Besu DLT network on given Kubernetes using Clique.

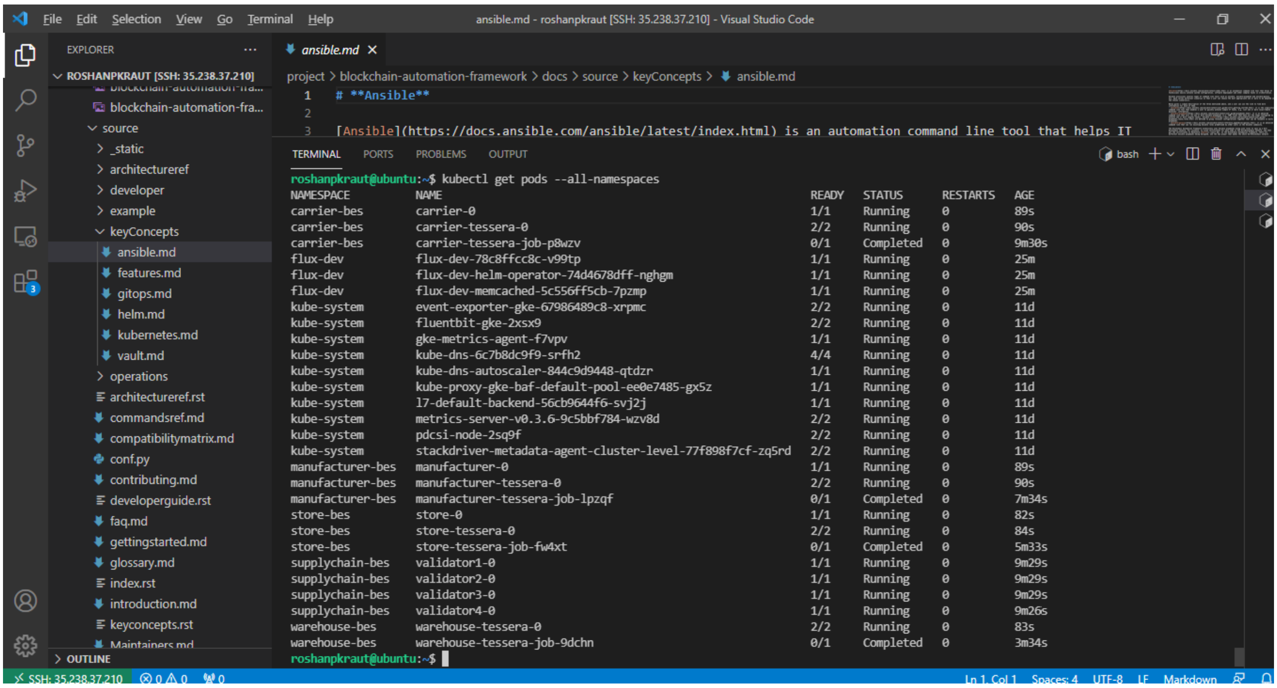

Figure 2: All pods running successfully to deploy Besu using Clique consensus

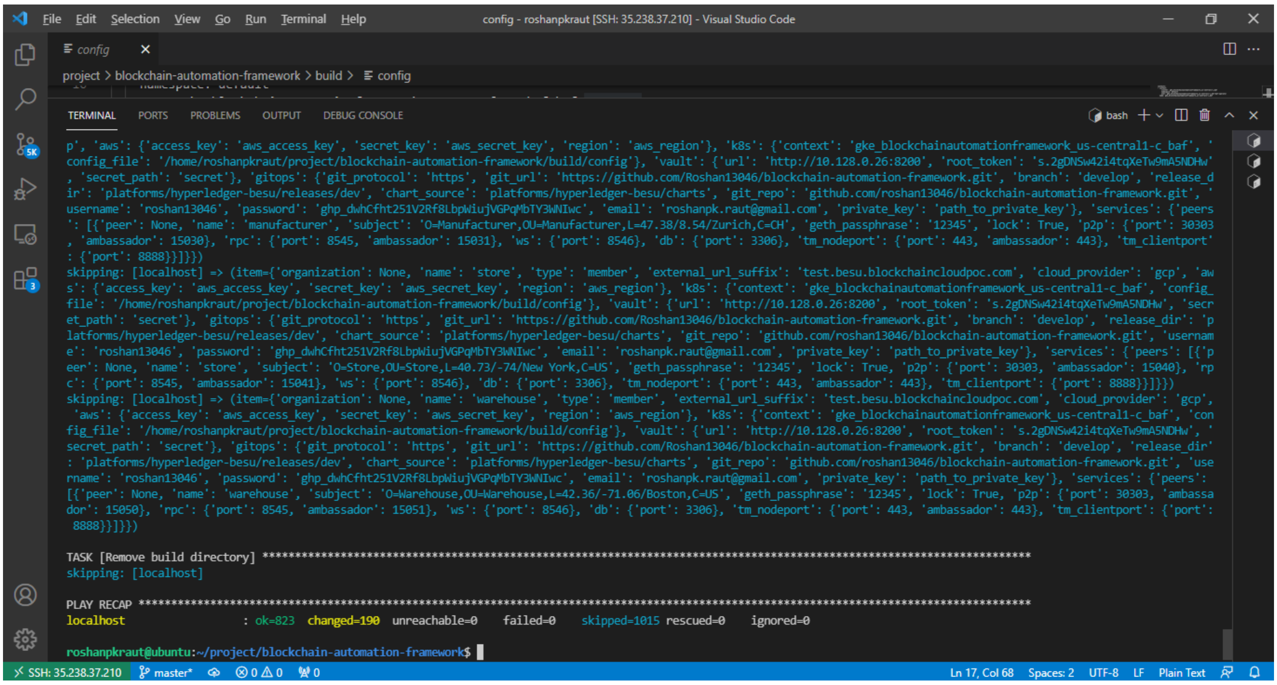

Figure 3: All pods running successfully to deploy Besu using Clique consensus

V. CONCLUSION

Initially, only IBFT2.0 consensus was supported for the Besu DLT network that required 4 validators, to keep the network live required >=2/3 validators, and the relative speed of validating is Slower compared to Clique. The addition of the Clique consensus mechanism requires only 1 validator, improved the Liveness with ½ validators live and made the relative speed faster than IBFT2.0. Thus, the addition of the Clique consensus feature improved the overall performance of the Besu DLT network.

ACKNOWLEDGEMENTS

Roshan Raut, Atharva Dhamke, Aadesh Jadhav, Kedar Kutaskar (PVGCOET-Pune students from Dept of Information Technology), Prof Satish Kamble (Project Guide), Sownak Roy (Project Maintainer and Code Reviewer)

VI. REFERENCES

[1] Blog on Blockchain consensus mechanisms [consensys.net]

[2] Review of Blockchain Architecture and Consensus protocols [mdpi.com]

[3] Blockchain Automation Framework GitHub Repo [BAF]

[4] Blockchain Automation Framework Official Documentation [Doc]