Fabric WebAssembly Chaincode
August, 2019
Fabric WebAssembly Chaincode

Introduction

Name: Shubham Aggarwal, @shubham_aggarwal
Location: Singapore
University: National University of Singapore
Mentor(s):
  Jay Guo, @guoger, IBM
  Morgan Bauer, @MHBauer, IBM
  Swetha Repakula, @swetha, IBM
Hyperledger project: Hyperledger Fabric
Fabric WebAssembly Chaincode

Project Description

- Hyperledger Fabric chaincode shims are only available in Golang, nodejs, and java.
- WebAssembly exposes lots of languages for development such as C, Rust, and many more
- This project aims to integrate WebAssembly based smart contract in Hyperledger Fabric
Fabric WebAssembly Chaincode

Project Objectives:

- Research on WebAssembly and do a feasibility study
- Create an integration layer for deploying WebAssembly based chaincodes on Hyperledger Fabric
- The integration layer should have none to minimal affect on Hyperledger Fabric core code
Fabric WebAssembly Chaincode

Project Deliverables:

- Integration layer for WebAssembly based chaincodes
- Essential shim wrapper functions to expose Hyperledger Fabric shim functionalities for WebAssembly based chaincode
- End to End example using one of the language other than java, js, golang
Fabric WebAssembly Chaincode

› How it all works
How it all works
Fabric WebAssembly Chaincode

DEMO
Fabric WebAssembly Chaincode

Project Execution & Accomplishments:
- Learned about WebAssembly, Golang
- Using github issues for new feature requests and bug reports.
- Using github actions for CI pipeline
- Most challenging part would be looking for feasible WebAssembly based execution engine
- My main accomplishment would be getting a good grasp on how to work in open source community.
Fabric WebAssembly Chaincode

Recommendations for future work:

› Create minimal interfaces for languages that can be compiled to WebAssembly
› Expose all other shim functions such as acl for WebAssembly chaincodes
› Easier exception feedback mechanism from host module to WebAssembly chaincode
› Try other WebAssembly execution engines as this area is evolving daily
› Create more examples using languages such as python, dotnet etc.
› Benchmarking