Hyperledger Fabric v1.4: Architecting, Designing and Deploying a Network

Course Summary

Description
This is designed for Architects, and System Administrators interested in gathering a deeper understanding on how Hyperledger Fabric Networks are Architected and operated throughout. You will learn about the integral base files in which Fabric are built upon. You will learn about how identity and permissions are administered throughout Membership Only networks. Additionally, you will gather the experience necessary to enter into and manage nodes/peer needs (i.e. chaincode installation, securing communication, etc).

Objective
Upon completion of this course, Students will:

- Gain an understanding of the Hyperledger Fabric network topology
- Learn about Chaincode, its purpose, and how to develop it using Go.
- Learn how to handle chaincode operations such as making chaincode packages accessible to peers, invoking and interacting with the chaincode logic from the CLI, and much more.

- Learn about how to create and develop an initial client Side solution (Using Node) to interact with the network.
- Learn about how to scale the network organizations, peers, and orderer nodes.
- Perform Certificate Authority Related operations & setup necessary for initial network Identity relations.

Topics

- Introduction
- What you should know:
- Blockchain Architecture
- Use Case Walkthrough
- Hyperledger Fabric Architecture
- Demystifying the Blockchain in HLF
- Fabric Network Definition Topology Walkthrough
- Bootstrapping the Backend
- Writing Chaincode
- Packaging and Distributing Chaincode
- Administering Identity
- Channel Management
- Network Scaling
- Securing Network
- Hosting OptionsAdding the SDK
- Writing an SDK (Lab)
- Everything’s a Database
- Smaller Companies Vs. Bigger Companies
- Consensus
- Everything about Orderers
- Network Security
- TLS implementation
- Bash Scripts to automate the Dev to deployment process.

Audience
This course is designed for System Administrators or anyone who wants to know how to run day-to-day network operations on a stood up Hyperledger Fabric network.
Hyperledger Fabric v1.4: Architecting, Designing and Deploying a Network

Course Summary (cont.)

Prerequisites

- Familiarity with Hyperledger Fabric Component Structure & purposes
- A minimal understanding of YAML Structure
- Knowledgeable on how to navigate through the Command Line

Duration

Three Days
Hyperledger Fabric v1.4: Architecting, Designing and Deploying a Network

Course Outline

I. Introduction
   A. Why Blockchain
   B. Why Hyperledger Fabric?

II. What you should know:
   A. Hardware Prerequisites
   B. Knowledge Prerequisites
   C. Blockchain Prerequisites

III. Blockchain Architecture
   A. Public Vs. Private
   B. Nodes
   C. Permissions

IV. Use Case Walkthrough

V. Hyperledger Fabric Architecture
   A. Component Walkthrough
      1. Orderer
      2. Certificate Authority
         a. Server
         b. Client
      3. Ledger
      4. Database
         a. Database Options
   B. Roles & Actors
   C. Transaction Flow
   D. Fabric Capabilities
   E. Identity
   F. New Features

VI. Demystifying the Blockchain in HLF
   A. Standard System vs. Blockchain System
   B. Integration with Existing Systems
   C. Examples

VII. Fabric Network Definition
     Topology Walkthrough
     A. What the purpose of each file
        1. File Content Walkthrough

VIII. Bootstrapping the Backend
      A. Specifying our Volumes

B. Writing our container definitions from scratch
C. Network Artifact
   1. What is it?
   2. Genesis Block
   3. Initial Channel Config
   4. Creation & Generation
D. Initial Certificate Creation & Walkthrough

IX. Writing Chaincode
   A. Chaincode Overview (what is it, purpose, etc.)
   B. Chaincode writing with GO, and Node

X. Packaging and Distributing Chaincode

XI. Administering Identity
   A. Identity In Hyperledger Fabric
      1. X509 certificates
   B. Certificate Authority Autonomy
      1. Purposes
      2. Capabilities (Register, enroll, etc)
      3. Best Practices (RootCA→ICA, etc)
      4. Certificate Authority Server vs. Client
         a. CLI
         b. SDK

XII. Channel Management
     A. What is a Channel?
        1. Purpose
        B. Creating Channels
        C. Permissioning Peers and Organizations to join channels
        D. Creating initial Channel

XIII. Network Scaling
      A. Horizontal Scaling vs. Vertical using Docker
      B. Organization Scaling
      C. Orderer Adjustment
Hyperledger Fabric v1.4: Architecting, Designing and Deploying a Network

Course Outline (cont.)

XIV. Securing Network
   A. Role Control with ACL’s

XV. Hosting Options
   A. AWS
   B. IBM Bluemix
   C. Digital Ocean

XVI. Adding the SDK
   A. What the SDK does
   B. SDK options
   C. SDK API’s

XVII. Writing an SDK (Lab)

XVIII. Everything’s a Database
   A. Ledger
   B. WorldState Database (Repository)
   C. Database options
   D. Database Differences

XIX. Smaller Companies Vs. Bigger Companies

XX. Consensus
   A. Modularity
   B. Endorsement Policy

XXII. Everything about Orderers
   A. What are they
   B. Node Difference in Orderers vs. other Peers
   C. Kafka vs. Solo

XXIII. Network Security

XXIV. TLS implementation

XXV. Bash Scripts to automate the Dev to deployment process.

Labs:
- Installing Fabric
- Bootstrapping the Network
- Building the Network Artifacts, and creating the base
- Building and Starting the Network
- Creating and Implementing Organizations
- Scaling the network & Using Modularity
- Creating the Channels and Implementing Permissions
- Adding Pearson Vue
- CA/MSP Operations