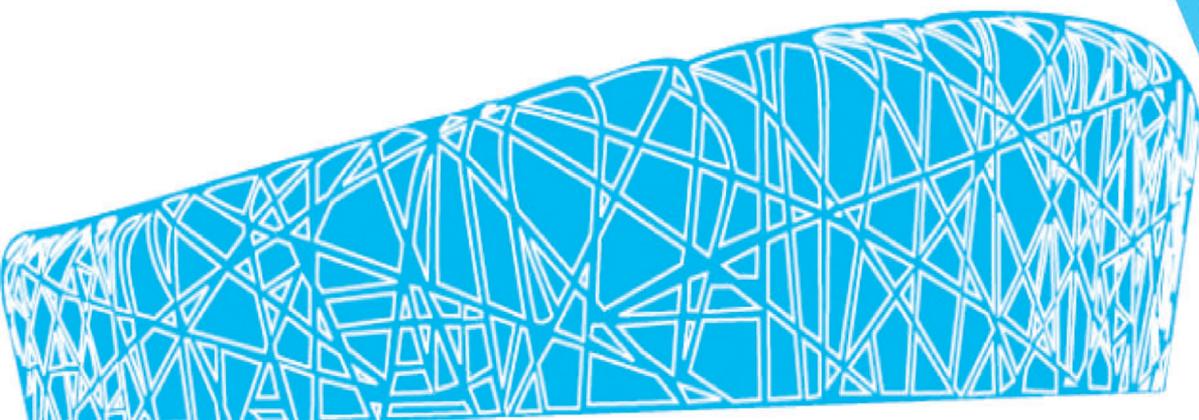


第一部分

基础篇

- 第1章 Hyperledger Fabric认知与环境搭建
- 第2章 Hyperledger Fabric架构体系详解
- 第3章 创建Hyperledger Fabric应用网络



1.1 Hyperledger Fabric 概述

1.1.1 Hyperledger 简介

从比特币开源以后,区块链逐渐广为人知,区块链技术也被广大技术人员所熟知。在多年的发展过程中,区块链技术已经逐渐成熟、完善,并已实际应用在众多不同的领域中。各行业的机构、企业也明确表态,大力支持区块链技术的发展,这使其快速形成相应的产业链,逐步推广至各个行业。

普通人员可能只会将区块链定义为一种虚拟数字货币,但对于技术人员而言,区块链实际上是多种相关的成熟技术结合之后产生的一种比较特殊的分布式数据库,其主要使用的核心技术就是点对点技术(peer-to-peer, P2P)、密码学及共识机制等。其在整个过程中能够实现去中心化、达成多方共识、实现数据的分布式存储且不可篡改。根据区块链应用范围的不同可以将其分类如下。

(1) 公有链: 全球范围内的所有组织、机构、个人均可参与,也被大众所熟知。

(2) 私有链: 某一个组织或机构范围内可以参与。由于其范围较窄,因此应用较少。

(3) 联盟链: 指定并经过审核的多个组织、机构均可参与,目前也被各大公司普遍应用。

超级账本(Hyperledger)是在 2015 年 12 月由 Linux 基金会主导并牵头建立的,由 IBM、Intel、Cisco 等金融、银行、物联网、供应链、制造和科技等行业的巨头共同宣布了 Hyperledger 联合项目成立。作为透明、公开、去中心化的企业级分布式账本技术,超级账本项目提供开源参考实现。目前已加入的成员超过 200 多家知名企业或机构。

国外成员有 IBM、Intel、Cisco、Oracle、Swift、RedHat、VISA、FUJITSU 及 NEC 等知名互联网、制造、金融和服务企业。

国内有小米、腾讯、联想、华为、中国信通院、房掌柜及浙江大学等互联网、金融、房地产和教育等行业的相关企业及机构成为 Hyperledger 企业会员。

超级账本项目是区块链技术中第一个面向企业级应用场景的开源分布式账本平台。在 Linux 基金会推出的 70 多个开源项目中,超级账本是增长最快的,目前已经成为企业级区块链技术的开放式全球生态系统。

作为 Linux 基金会的一部分,超级账本基金会(Hyperledger Foundation)采用模块化的方法来托管项目。超级账本基金会负责开发商业区块链项目,专注于为企业级区块链部署开发一套稳定的框架、工具和库。超级账本基金会邀请所有人能够为 Hyperledger 项目和

社区做出贡献,共同推进分布式账本和智能合约的行业目标。

 **超级账本基金会:** Hyperledger Foundation 是一个开源社区,是一个由 Linux 基金会主办的全球合作项目,包括金融、银行、物联网、供应链、制造业和技术领域的领导者。在技术治理和开放合作的基础上,个人开发者、服务和解决方案提供商、政府协会、企业成员与最终用户都被邀请参与这些改变规则的技术的开发和推广。

1.1.2 超级账本项目分类

1. 框架项目 (framework project)

超级账本(Hyperledger)项目包含了多个不同的区块链框架(分布式账本应用开发平台),主要提供企业级应用程序开发环境及相关的软件开发工具包(software development kit,SDK),用来满足不同行业的需求。

Besu: Hyperledger Besu 是一款以太坊客户端,以前称为 Pantheon,被设计作为公共和私有许可网络,对企业友好。它也可以在 Rinkeby、Ropsten 和 Görli 等测试网络上运行。Hyperledger Besu 包括几种共识算法:工作量证明(proof of work,PoW)和权威证明(proof of authority,PoA)。其综合许可方案被专门设计用于联合体环境。

Fabric: 用模块化架构作为开发区块链程序或解决方案的基础,允许一些部件(如共识算法和成员服务)实现即插即用。

Iroha: 由 Soramitsu、Hitachi、NTT Data 和 Colu 提供,使用 C++ 编写,采用拜占庭容错(byzantine fault tolerance,BFT)一致性算法实现共识,是为了将分布式账本技术简单容易地与基础架构型项目集成而设计的一个区块链框架项目。其应用程序可以使用 Python、Java、JavaScript、C++、Android 和 iOS 移动平台编写。

Sawtooth: 一个创建、部署和运行分布式账本的模块化平台。它包含一个新奇的共识算法,叫作消耗时间证明(proof of elapsed time,PoET),其面向大型分布式验证器群时能够消耗最少的资源。

Indy: 特别为去中心化的身份而建立的一种分布式账本,用于提供植根于区块链或其他分布式账本的数字身份,以便可以跨管理域、应用程序和其他孤岛进行互操作。

2. 超级账本实用工具(hyperledger tool)

为了能够对基于 Hyperledger 框架项目开发的区块链应用程序进行测试、部署、监控及管理,Hyperledger 提供了多个不同的实用工具来实现相关的功能。

Bevel: Hyperledger Bevel 是一个加速器工具,开发人员可以通过它在公共和私有云提供商之间一致地部署生产就绪的分布式网络。

Cacti: Hyperledger Cacti 是一个区块链集成工具,是超级账本目前孵化中的一个项目,旨在提供分散的、安全的和自适应的区块链网络之间的集成。Hyperledger Cacti 能够允许用户安全地集成不同的区块链。

Caliper: 由华为、Hyperchain、Oracle、Bitwise、Soramitsu、IBM 和布达佩斯技术与经济大学的开发人员提供;是一个区块链基准工具,内置一套预定义的测试用例,让使用者可以测试特定区块链执行的性能。

Cello: 由 IBM 提供,赞助商为 Soramitsu、华为和英特尔(Intel);旨在给区块链生态系统带来按需部署服务的模式,减少创建、管理和终止区块链的难度。

FireFly: Hyperledger FireFly 是第一个开源的超级节点是企业构建并扩展安全 Web3 应用程序的完整堆栈。FireFly API 用于数字资产、数据流和区块链交易,使得在流行的链和协议上构建应用程序的速度大幅加快。

 **提示:** 如果读者需要更加详细地了解超级账本的相关内容,可以通过访问 Hyperledger 官方网站获取相关的详细信息。各项目也可以在 GitHub 仓库中获取对应项目的说明及源代码。

1.1.3 Fabric 概述

Hyperledger Fabric 是一个由 IBM、Intel、DAH 等企业机构共同提供的区块链分布式账本的具体实现平台,也是 Hyperledger 旗下最优秀的项目之一。因其目的明确,专为企业设计,所以 Fabric 具有高度模块化和可灵活配置的架构,实现了组件的即插即用,可为各行各业的不同业务提供多样性的服务和优化服务,其中包括银行、金融、保险、医疗保健、人力资源、供应链,甚至数字音乐分发。

Fabric 是第一个支持使用通用编程语言(如 Java、Golang 和 Node.js)编写智能合约的分布式账本平台,不会受限于特定领域的编程语言(domain-specific languages, DSL)。这种情况说明大多数企业已经拥有开发智能合约所需的技能,并不需要额外花费时间、精力及费用去培训学习新的语言或特定领域的编程语言。

Hyperledger Fabric 是一个开源的企业级许可分布式账本技术(distributed ledger technology, DLT)平台,Hyperledger Fabric 与其他流行的分布式账本或区块链平台最大的不同主要体现在以下几个方面。

1. 隐私和保密性

Hyperledger Fabric 是一个需要经过许可的平台,通过其通道架构(channel architecture)和私有数据(private data)的特性来实现对数据的保密性。Hyperledger Fabric 提供了建立通道(channel)的功能,允许参与者为交易新建一个单独的账本。参与者并不希望所有的交易信息(如提供给部分客户的特定价格信息)都对网络中所有参与者公开。只有在同一个通道中的参与者,才会拥有该通道中的账本,而其他不在此通道中的参与者则无权查看到这个账本的相关信息。同时 Hyperledger Fabric 还可以支持多通道(multi channel)的实现。

2. 许可

几乎任何人都可以参与非许可区块链,并且每个参与者都可以是匿名的。与开放且无须许可的网络(公有区块链)系统允许未知身份的参与者加入不同网络(需要通过工作量证明协议来保证交易有效并维护网络的安全)不同,Hyperledger Fabric 通过成员服务提供商(membership service provider, MSP)来登记所有的成员,并且在许可的情况下,降低了参与者故意通过智能合约引入恶意代码的风险。

3. 可插拔的共识协议

Hyperledger Fabric 支持可插拔的共识协议,能够让平台根据不同的应用场景进行定

制,以适应特定的业务场景和信任模型。例如,当多台服务器部署在单个企业内或由可信任的权威机构管理时,会大大降低其性能和吞吐量的需求,所以完全拜占庭容错的共识机制不是必要的。在这样的情况下,使用崩溃容错(crash fault-tolerance,CFT)共识协议可能就足够了,而在去中心化的场景中,则需要使用更传统的拜占庭容错共识协议。

Hyperledger Fabric 与其他区块链平台相比较,最大的优势在于其性能更好。一个区块链平台的性能可能会受到诸多因素的影响,如交易大小、区块大小、网络情况及硬件限制等,但 Hyperledger Fabric 最新的性能测试已经达到了 20000 笔/s。

综上所述,大部分企业公司在开发区块链应用程序时,都会考虑并优先使用 Hyperledger Fabric 平台,来适应不同的应用场景及需求。

1.2 搭建 Hyperledger Fabric 环境

1.2.1 环境要求说明

“工欲善其事,必先利其器。”要学习使用超级账本开发区块链应用程序,首先必须了解相应的开发环境,掌握需要哪些开发工具,以及对应工具的获取、安装及设置。

(1) 操作系统: Ubuntu/CentOS/macOS/Windows。

(2) 所需工具: Git、cURL、Docker、Docker-Compose、Golang。

(3) fabric-samples: bootstrap.sh。

(4) 相关二进制文件: bin 和 config 文件夹及其所包含的可执行二进制工具及配置文件。

下面向大家逐一介绍各工具的安装及相关配置说明。

1.2.2 操作系统的要求

推荐使用的操作系统为 64 位的 Ubuntu v16.04 LTS,系统内核为 GNU/Linux 4.13.0-36-generic x86_64。

硬件资源要求: 内存最低为 2GB,最好 3GB 以上;磁盘空间为 30GB 或更高。

如果计算机默认安装的是 Windows 操作系统,则可以在 Windows 系统中安装一个 VMware 或 VirtualBox 虚拟机应用程序,然后在虚拟机中安装 Ubuntu v16.04 LTS 系统,最后在 Windows 操作系统中安装一个远程连接工具(如 XShell 或 SecureCRT),以便对 Ubuntu 系统进行远程连接并操作。后面所有的操作都基于 Ubuntu v16.04 LTS 的操作系统。

 **提示:** 如果你使用的是 Mac 计算机,同样建议使用安装虚拟机的方式来学习。

1.2.3 工具的安装

1. 安装 Git

Git 是一个非常优秀的免费、开源的版本管理控制工具,用户可以使用 Git 工具方便地

下载 Golang、Hyperledger Fabric 等官方在 GitHub 网站上发布的相关源代码或其他内容。在系统中安装 Git 工具可以使用以下命令。

```
$ sudo apt update
$ sudo apt install git
```

Git 工具在 macOS 系统中默认已安装。如未安装,可以通过浏览器进入 git-scm 官方网站的下载页面,选择 Mac 系统的安装包下载并安装。

2. 安装 cURL

如果操作系统中还没有安装 cURL 工具,可以使用如下命令安装 cURL。

```
$ sudo apt install curl
```

3. 安装 Docker

首先需要检查一下当前系统中是否已经安装了 Docker 工具,如果已经安装,需要将其升级至 v18.03+ 或最新版本;如果未安装,则可以直接使用相关命令进行安装。步骤如下。

(1) 查看系统中是否已经安装 Docker。

```
$ docker -- version
```

如果未安装,则使用以下命令安装 Docker 的最新版本。

```
$ sudo apt update
$ sudo apt install docker.io
```

(2) 安装完成,使用以下命令查看 Docker 的版本信息。

```
$ docker -- version
```

命令执行后会在屏幕中输出以下的 Docker 版本信息。

```
Docker version 20.10.12, build 20.10.12 - 0ubuntu2~20.04.1
```

如果输出上面的信息,则证明 Docker 工具已经安装成功。

为了确保 Docker daemon(Docker 守护进程)是在运行状态,可以使用以下命令启动 Docker 服务。

```
$ sudo systemctl start docker
```

 **提示:** 如果希望 Docker daemon 在系统启动的时候会自动启动的话,则可以使用如下命令实现。

```
$ sudo systemctl enable docker
```

最后,使用以下命令将系统中指定的用户添加至 docker 组。

```
$ sudo usermod -a -G docker <username>
```

 **注意：**添加至 docker 组的命令执行完毕之后必须注销或退出当前终端之后再重新进入。

4. 安装 Docker-compose

与安装 Docker 类似,需要先检查一下系统中是否已安装 Docker-compose 工具。可以使用以下命令进行检查。

```
$ docker - compose -- version
```

如系统提示未安装,则使用以下命令安装 Docker-compose 工具。

```
$ sudo apt install docker - compose
```

安装成功后,查看 Docker-compose 版本信息。

```
$ docker - compose -- version
```

命令执行后会在屏幕中输出以下 Docker-compose 版本信息。

```
docker - compose version 1.25.0, build unknown
```

5. 安装 Golang

Hyperledger Fabric v1. 4. x 要求 Golang 版本必须为 v1. 11. x 以上,Hyperledger Fabric v2. 2. 0 版本要求 Golang 版本必须为 v1. 14. 4 以上。本书使用 Hyperledger Fabric v2. 2. 9 环境,选择使用 Golang 版本为 v1. 18. 7。

1) 下载 Golang

首先在当前用户目录下创建一个名为 download 的目录,用来保存从网络中下载的文件。然后使用 wget 工具下载 Golang 的指定版本压缩包文件 go1. 18. 7. linux-amd64. tar. gz:

```
$ cd ~
$ mkdir download && cd download
$ wget https://golang.google.cn/dl/go1.18.7.linux-amd64.tar.gz
```

下载 Golang 压缩包文件,需要保证操作系统能够正常访问 Golang 官方网站。下载过程可能耗时较长(取决于具体网络情况),需耐心等待。

其他系统可以在 Golang 官方下载页面中查找相应的安装包下载并安装。下载完成后文件会保存在当前的 download 目录下,可以使用 ll 命令查看,如图 1-1 所示。

2) 解压文件

压缩包文件成功下载至本地后,使用 tar 命令将下载后的压缩包文件解压到指定的 /usr/local/ 路径下:

```
$ sudo tar -zxvf go1.18.7.linux-amd64.tar.gz -C /usr/local/
```

```

kevin@example:~$ cd download/
kevin@example:~/download$ wget https://golang.google.cn/dl/go1.18.7.linux-amd64.tar.gz
--2022-11-19 00:01:23-- https://golang.google.cn/dl/go1.18.7.linux-amd64.tar.gz
Resolving golang.google.cn (golang.google.cn)... 180.163.151.34
Connecting to golang.google.cn (golang.google.cn)[180.163.151.34]:443... connected.
HTTP request sent, awaiting response... 302 Found
Location: https://dl.google.com/go/go1.18.7.linux-amd64.tar.gz [following]
--2022-11-19 00:01:23-- https://dl.google.com/go/go1.18.7.linux-amd64.tar.gz
Resolving dl.google.com (dl.google.com)... 203.208.39.225
Connecting to dl.google.com (dl.google.com)[203.208.39.225]:443... connected.
HTTP request sent, awaiting response... 200 OK
Length: 141906548 (135M) [application/x-gzip]
Saving to: 'go1.18.7.linux-amd64.tar.gz'

go1.18.7.linux-amd64.tar.gz          100%[=====] 135.33M  1.16MB/s   in 1m 58s
2022-11-19 00:03:22 (1.15 MB/s) - 'go1.18.7.linux-amd64.tar.gz' saved [141906548/141906548]

kevin@example:~/download$ ll
total 138596
drwxrwxr-x 2 kevin kevin  4096 Nov 19 00:01 ./
drwxr-xr-x 5 kevin kevin  4096 Nov 19 00:03 ../
-rw-rw-r-- 1 kevin kevin 141906548 Oct  5 01:44 go1.18.7.linux-amd64.tar.gz
kevin@example:~/download$

```

图 1-1 Golang 安装包下载

 **提示：**如果在解压过程中出现以下错误。

```

gzip: stdin: unexpected end of file
tar: Unexpected EOF in archive
tar: Unexpected EOF in archive
tar: Error is not recoverable: exiting now

```

则说明下载的 tar 压缩包文件有问题,如没有下载完整或压缩包数据损坏。请删除后重新下载,并重新使用解压缩命令将其解压至指定的目录中。

3) 配置环境变量

解压至指定目录后,为了使 Golang 可以被系统的所有用户正常使用,我们使用 vim 文件编辑工具打开系统的 profile 文件进行编辑。

```
$ sudo vim /etc/profile
```

将光标定位在文件的最后一行,然后在 profile 文件末尾添加以下内容。

```

export GOPATH = $ HOME/go
export GOROOT = /usr/local/go
export PATH = $ GOROOT/bin: $ PATH

```

编辑完成之后,保存退出,然后使用 source 命令,使刚刚添加的配置信息生效。

```
$ source /etc/profile
```

如果只想让当前登录用户使用 Golang,其他用户不能使用,则编辑当前用户 \$ HOME 目录下的 bashrc 或 profile 文件,在该文件中添加相应的环境变量即可。

通过 go version 命令验证是否成功。

```
$ go version
```

如果成功,则会输出以下 Golang 版本信息。

```
go version go1.18.7 linux/amd64
```

如果系统中有旧版本的 Golang,则使用以下命令卸载旧版本的 Golang,然后重新安装并配置。

```
$ su -
# apt - get remove golang - go -- purge && apt - get autoremove -- purge && apt - get clean
```

1.2.4 安装 Hyperledger Fabric

所需工具的下载和安装完成之后,我们就可以下载并安装 Hyperledger Fabric 了。安装方式可以分为两种:第一种也是经常使用的方式,使用官方提供的一个脚本文件实现自动安装;第二种是先将源码下载至本地系统中,然后手动编译源码进行安装。下面向大家介绍使用脚本安装的具体方式。

为了方便安装,Hyperledger Fabric 官方提供了一个可执行脚本,该执行脚本可以将 fabric-samples 和所需的二进制工具自动下载安装至本地系统中。

Hyperledger Fabric 的最新版本为 v2.5.x,我们将使用一个 v2.2.x 的 LTS 版本(上一个 LTS 版本为 v1.4.x)。为了方便后期的操作与管理,我们先在操作系统中创建一个目录,用来保存下载的 fabric-samples。

```
$ cd ~ && mkdir fabric && cd fabric
```

创建并进入该目录之后,创建一个名为 bootstrap.sh 的脚本文件,并将 GitHub 仓库中 hyperledger/fabric 项目的 blob/v2.2.9/scripts/bootstrap.sh 的内容复制至新创建的脚本文件中,然后保存退出(或直接将该脚本文件下载至本地系统中)。

```
$ vim bootstrap.sh
```

该 bootstrap.sh 可执行脚本文件的作用如下。

- (1) 如果当前目录中没有 hyperledger/fabric-samples 目录及其所包含的相应文件、子目录,则从 github.com 复制到 hyperledger/fabric-samples 存储库。
- (2) 使用 checkout 切换为对应指定的版本标签。
- (3) 将指定版本的 Hyperledger Fabric 平台特定的二进制文件和配置文件及 fabric-ca 下载并解压到 fabric-samples 存储库的根目录中。
- (4) 下载指定版本的 Hyperledger Fabric Docker 镜像文件。
- (5) 将下载的 Docker 镜像文件及各镜像文件标记为“latest”。

普通文件在 Linux 操作系统中是无法直接运行的,我们必须对其赋予可执行权限它们才能够运行,执行以下命令对 bootstrap.sh 脚本文件赋予可执行权限。

```
$ chmod +x bootstrap.sh
```

赋予可执行权限之后,可以直接执行此文件。

```
$ sudo ./bootstrap.sh
```

在执行此脚本文件时,网络环境必须稳定,否则会因为网络而导致各种问题的产生,例