

# 北京邮电大学 冯力全

年龄: 22 岁	手机: 18810687830	2013.09-2018.06	北京邮电大学
性别: 男	邮箱: <a href="mailto:flyq951@gmail.com">flyq951@gmail.com</a>	专业: 通信工程	学历: 本科
Github: <a href="https://github.com/flyq">https://github.com/flyq</a>		微信公众号地址:	<a href="#">Solidity 学习</a>

## 项目经历:

- ◆ 2016.03-2016.08                      全国高校软件定义网络应用创新开发大赛                      项目负责人
- 在 Ubuntu 下部署 MININET + Ryu + OpenDaylight + sFlow + Wireshark/Tshark + Postman, 通过在 MININET 网络仿真环境下, 模拟多节点网络, 并进行流量控制。通过这个过程, 了解了 Ryu 的北向编程接口, 即在启动 Ryu 时可以通过参数指定自带的网络规则或者使用 Python 编写的网络规则。然后通过 sFlow 实时查看流量的变化, 用 Wireshark 工具抓包并用 Tshark 工具将包解析为 CSV 文件格式, 以及使用 OpenFlow 的 RESTAPI 来查看、添加、修改和删除流表项。
- 在一个网络仿真环境中, 利用 Hping3 模拟 DDoS 攻击, 利用 Python 的 socket 库来抓取并解析包的 TCP 包头能够得到 ack, seq 等 TCP 包头信息。然后下发对应的流表。在这个过程中熟悉了 TCP 协议格式, 理解了网络通信过程。

## 实习经历:

- 2017.10-2017.12                      对 Hyperledger 项目进行环境搭建以及实践                      公司: 泰康在线
- 环境搭建: 在 Ubuntu 16.04 下安装 Golang 1.9.2 以及 Node 6.9.5。安装 Docker 17.09.1-ce, 并从 Docker hub 下载相关镜像。
- Demo 实验: 1、通过一步步输入命令, 实现了官方的 [Building Your First Network](#), 能够进入到客户端容器中进行交易和查询操作, 理解了 fabric 网络建立的过程。2、在 1 基础上, 通过安装不同链码, 实现不同智能合约, 并成功实现了官方 marbles 智能合约。3、在 2 的基础上, 启动 couchDB, 实现对资产的富查询。4、通过 Node-SDK, 在本机环境下调用链码实现交易和查询操作。5、根据 [IBM-marbles 项目](#), 实现其 demo, 通过在网页上对资产 marble 进行拖拽实现交易并生成区块。6、安装运行 blockchain-explorer, 实现在网页查看网络节点、生成交易和区块具体信息、交易速度和区块生成速度。
- 项目理解: 通过进行 Demo 实验和阅读区块链原理设计与应用以及查看 fabric 源码, 对 fabric 整个项目有个整体认识, 如交易生成过程, 共识的机制以及用户链码与底层账本的交互接口 shim 层的作用。

## 以太坊合约开发:

- 2018.01-2018.05                      Ethereum 合约开发
- 根据[官网的发行教程](#), 安装了 Ethereum Wallet, 并同步了 Rinkeby 测试网络的区块。然后[获取 rinkeby ETH](#), 并在测试网络上部署了 [ZCC 代币](#), 熟悉测试链上智能合约的部署流程。并通过对 ERC20 token 的 [Solidity 源码分析](#), 增进 ERC20 理解。
- 刷了一遍 [CryptoZombies](#) 教程, 并在过程中查阅了以太坊的[相关资料](#), 熟悉了如基于 keccak256 的 hash 算法, 以及结构, 映射和函数被调用权限控制等。深度参与各类 CryptoGame, 包括 [CryptoKitties](#), [以太水浒](#), [CryptoPlanet](#), [Alchemygod](#) 等, 并通过查看他们的 Etherscan 地址, 分析其基于 ERC721 token 的 [Solidity 源码分析](#), 其中熟悉 ERC721 token 标准。并参与准备上线的[对赌合约](#)的编写, 负责其调试与部署。参与即将上线主网的[以太三国](#)的合约的 review, 测试。
- 深度参与了 [DaoOne 社区](#), 一个刚起步的去中心化学习协作社区。现在和别人一同负责技术培训和[教程编写](#), 并且是 GitHub 上 [DaoOne Organization](#) 的成员之一。

## 获奖情况:

- 2016.08                      2016 全国高校软件定义网络(SDN)应用创新开发大赛                      互联网应用创新开放平台联盟
- 2013.9-2016.6                      三一集团圆梦奖学金                      三一基金会
- 2014.11                      北邮高数竞赛二等奖                      中国数学会普及工作组
- 2013.9-2014.6                      三等奖学金                      北京邮电大学教务处

## 专业技能:

- 熟悉常见数据结构, 基本算法。熟悉计算机网络。
- 了解 Python, Javascript, shell 脚本语言, 写过相关的脚本。了解 C++ 和 Java, 用 C++写过俄罗斯方块, 扩展成多人操作
- 了解 Golang, 了解 Go 轻量级框架 Beego, 并用 Beego 进行[博客网站](#)的搭建, 了解 Go 并发编程的原理。
- 熟悉区块链相关技术, 如 PoW, PoS, DPoS, PBFP 共识机制, Ethereum 的 solidity, BTC 的 UTXO 交易机制等。
- 熟悉 Linux 操作和基本命令。熟悉 Git 操作; 熟悉常见 IDE 的使用, 如 VS, Eclipse; 熟悉 Emacs 基本操作。

---

◆ 2017.6-2017.8

**基于 Golang 的网络爬虫实践**

**项目负责人**

- 将爬虫框架分成网页下载模块，分析清洗模块，条目处理管道和调度模块。
  - 网页下载模块接受作为输入的请求，并将该请求转换成 HTTP 请求发送至远程服务器，之后，立即等待相应的 HTTP 响应，将响应封装后返回给网页下载模块的调用方。分析模块接受作为输入的响应，并将该响应还原成 HTTP 响应，并对其进行检查，条目处理管道接受作为输入的条目，并对其执行一系列的处理。调度模块调度各个处理模块的运行，其中包括维护各个处理模块的实例、在不同处理模块实例之间传递的数据和监控这些被调度者的状态。
  - 通过这个项目，初步从需求分析、总体设计、详细设计和编码实现等几个步骤体验了软件工程的开发流程。体会了 Go 语言在并发编程方面的威力，以及在开发效率和编译运行效率上的优势。
-