# Configuration Structures

Overall structure:

<https://blockchain-automation-framework.readthedocs.io/en/latest/operations/fabric_networkyaml.html>

Crypto-Config structure:

<https://hyperledger-fabric.readthedocs.io/en/release-1.1/commands/cryptogen-commands.html>

Run test network:

<https://hyperledger-fabric.readthedocs.io/en/latest/test_network.html>

Hyperledger Releases:

<https://github.com/hyperledger/fabric/tags>

# Steps to create hlf-network with 2 orgs:

* Create crypto-configs for the orderer and peer orgs.
* Install the binary tools

Refer this link: <https://hyperledger-fabric.readthedocs.io/en/latest/install.html>

Open git bash

And run install-fabric.sh script in git bash with appropriate arguments.

* Generate crypto materials for orgs and orderers
* Generate ccp files for all orgs

**Port in networking:**

A port in networking is a software-defined number associated to a network protocol that receives or transmits communication for a specific service.

An IP address identifies a machine in an IP network and is used to determine the destination of a data packet. Port numbers identify a particular application or service on a system.

**Raft**

New as of v1.4.1, Raft is a crash fault tolerant (CFT) ordering service based on an implementation of Raft protocol in etcd. Raft follows a “leader and follower” model, where a leader node is elected (per channel) and its decisions are replicated by the followers. Raft ordering services should be easier to set up and manage than Kafka-based ordering services, and their design allows different organizations to contribute nodes to a distributed ordering service.

**BFT (New as of v3.0)**

A Byzantine Fault Tolerant (BFT) ordering service, as its name implies, can withstand not only crash failures, but also a subset of nodes behaving maliciously. It is now possible to run a BFT ordering service with the SmartBFT library as its underlying consensus protocol. Consider using the BFT orderer if true decentralization is required, where up to and not including a third of the parties running the orderers may not be trusted due to malicious intent or being compromised.

**Gossip Protocol:**

<https://hyperledger-fabric.readthedocs.io/en/latest/gossip.html>

.