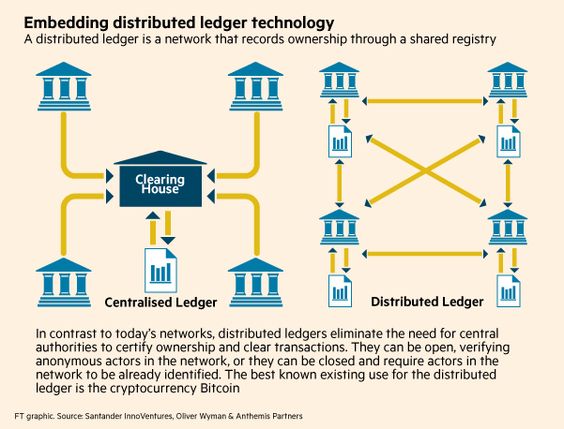
## horizontal line

Distributed Ledger Security Practice for Startups

15th August, 2017



Shahid Sharif

[Security, Privacy, & Risk Management Inc.](https://www.securityprivacyrisk.com)

Toronto, ON

# Table of contents

[**Table of contents**](#_w149twjnzs57) **1**

[**Change Log:**](#_i8ntfef8ovbp) **3**

[**Overview**](#_au51mny0sx6) **4**

[**Goals**](#_o4tpdrxmcfzu) **5**

[**Security Organization**](#_f38ndwr8iz6n) **5**

[**Privacy & Data Protection**](#_96460ze86v9) **6**

[**Security Policy**](#_gmxpdi63xufe) **6**

[**Security Manual**](#_5jed058ai6gf) **6**

[**Security Standards**](#_v5qabkhkrsdi) **6**

[Application Security](#_q3usoe91qckl) 6

[Code Analysis](#_uu0c04wwt9o7) 6

[Key Management](#_5lpik1wur5qf) 6

[System Security](#_viwet48vbt57) 7

[**Security Guidelines**](#_cki320u085vb) **7**

[**Security Architecture**](#_jwjzxub91fk0) **7**

[Network Architecture](#_mmojfp2zhbi7) 7

[Application Architecture](#_3ot7pyhrvq3r) 7

[System Architecture](#_s3ohj7kf68rz) 7

[**Security Operations**](#_bmz0nq16i3b2) **7**

[**Governance, Risk, & Compliance**](#_2mziguvlfuxz) **7**

[**Milestones**](#_yyrhu7ml5bea) **8**

[Lorem ipsum](#_buwz1tcz7y35) 8

[Dolor sit amet](#_1q8rddq5uxlj) 8

[**Consensus Protocols**](#_y1tlblviua2u) **9**

[**References**](#_viboqd4y02mj) **10**

# 

# Change Log:

|  |  |  |  |
| --- | --- | --- | --- |
| **Date** | **Editing time** | **Editor** | **Update Notes** |
| 15Aug2017 | 3hours | Shahid Sharif | First draft |
| 3Sep2017 | 3mins | Shahid Sharif | Renamed the document from “Distributed Ledger Security” to “Distributed Ledger Security Practice for Startups” |
| 10Sep2017 | 1500→ 1700 | Shahid Sharif | Added Distributed Ledger Components |

# 

# 

# Glossary of Terms

|  |  |
| --- | --- |
| **Term** | **Definition** |
| EVM | Ethereum Virtual Machine |
| SIEM | Security Information & Event Management |
| IDS | Intrusion Detection System |
| IPS | Intrusion Prevention System |

# 

# Overview

Relational databases were first introduced in 1970’s, prior to that, databases existed in one form or the other. When relational databases became mainstream, their availability became a concern, and as a result various database vendors provided data replication capabilities amongst the databases.

Distributed ledgers are pretty much repositories that rely on a consensus algorithm to replicate data across geographically disparate ledgers.

Blockchain is one of the implementations of distributed ledger technology.

Bitcoin was the very first use case of the distributed ledger technology using a specific consensus algorithm, proof of work.

Since the inception of Bitcoin, which is also called a cryptocurrency, various other implementations of blockchain distributed ledger came into existence and are called Alternate Coins or Alt Coins, some examples are:

1. Litecoin
2. Darkcoin
3. Dogecoin
4. IOTA

After the cryptocurrency craze, the business users started thinking of other use cases such as the need to implement some form of logic into blockchain, this would allow decisions based on parameters. As a result, Ethereum introduced this ability to implement logic as smart contracts. Then after Ethereum other implementations came to light which were private or public. This stack allowed users the ability to implement Distributed Applications (Dapp) on the distributed ledger. Some of the implementations that seem to improve on Ethereum are:

1. Steem
2. EOS
3. Peerplays
4. Hyperledger
5. Tezos

While the industry dabbles with various technologies, and implementation of smart contracts , the key to implementation would be the security of the underlying platform.

# Goals

This paper will present security framework that can be leveraged for private blockchain implementations of the Distributed Ledger technology. Some of the areas to consider are:

1. Security Organization
2. Privacy & Data Protection
3. Security Policy
4. Security Manual
5. Security Standards
6. Security Guidelines
7. Security Architecture
8. Security Operations
9. Governance, Risk, & Compliance

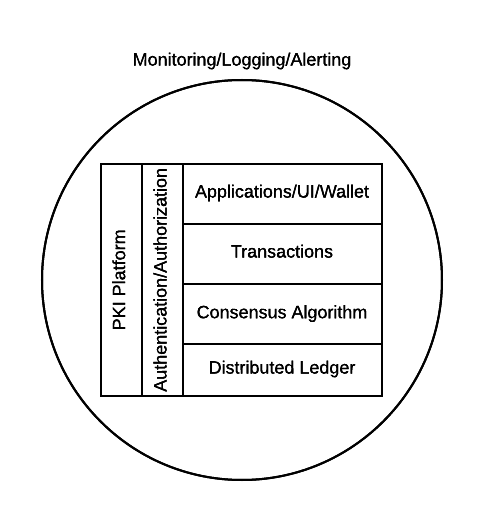
# Distributed Ledger

Blockchain is an implementation of a DL, and in my opinion, the key ingredients are:

1. **Users**: Entities generating transactions when they send/receive bitcoin.
2. **Distributed Ledger**: All transactions are written all active nodes participating in the network
3. **Malleability**: Transactions written to the Blockchain cannot be changed as everyone has the same copy, no one party can change a transaction on their own, unless they own 51% of the hashing power.
4. **Proof of Work:** To generate a block, the miners are given a range, within which the hash of their block should be in, if the hash is within the range, the miner gets a reward, which is a Bitcoin. This activity of finding the hash value within a certain range is very resource intensive and requires multiple miners working together to solve this problem. Once this block is generated, the miner announces it to the network of nodes running the distributed ledgers that the problem has been solved and they can move to solving another block. Proof of work algorithm addresses the issues with multiple miners announcing that they have solved the problem.
5. **Miners**: Entities with distributed ledgers who are using their computers to package x number of transactions into blocks.

## Distributed Ledger Framework

The following framework outlines the DL platform environment.



## Other Implementations

Other DL implementations include 1,2, and 3, but depending on what the consensus algorithm has implemented as in 4 above, 5 may or may not even exist.

As more features get added to the DL platforms, they become more complex, and so does the security.

For instance Ethereum uses a scripting engine to translate smart contracts written in Solidity Ethereum Virtual Machine has been developed by a group of developers but a thorough code review has not be done on it. Nobody knows if it has any vulnerabilities exists in the code or an independent review [[1]](#footnote-0)has been completed.

# Security Organization

For Security to be taken seriously a security organization is mandatory, while the non operational team would report to the CISO , the operational structure can be based on two models:

1. Centralized
2. Distributed

In centralized operational model, all aspects of security operations are within the security organization, such as:

1. User provisioning/deprovisioning
2. Security Information & Event Management (SIEM)
3. Security Operations Center
4. Incident Response
5. Firewall & Network Operations

In a decentralized operations model, the above aspects are managed by different business units with the security organization providing oversight in form of GRC.

# Privacy & Data Protection

Data that is being processed by the solution should care fully considered for:

1. Data protection: Based on data classification ensure data is protected in storage and in transit.
2. Data retention: Based on regulatory, and industry requirements, data should be retained for
3. Data access & update: Allow end users ability to access their data and modify it as required.

# Security Policy

Before any solution is implemented, a security policy must be created to ensure all the industry, regulatory, and other compliance requirements are documented. This document will provide the security requirements to ensure the deployment is secure. All the requirements should be very high level without delving into implementation details.

# Security Manual

This document goes into details on the how the security policy requirements must be implemented. It can be one document, or multiple depending on the size of implementation.

# Security Standards

Standards are mandatory requirements that must be adhered to. Some of the standards to be considered are:

## Application Security

The scope of application security would be the

### Code Analysis

### Key Management

## System Security

All the servers that the distributed ledger platform runs on must be:

1. Hardened
2. Monitored for availability
3. Intrusion Detection/Intrusion Protection mechanisms deployed to protect from attacks coupled with SIEM for proactive defense.
4. Based on the security be in a DMZ

# Security Guidelines

# Security Architecture

## Network Architecture

## Application Architecture

## System Architecture

# Security Operations

A robust mechanism must be instituted to ensure all systems are baselined and any deviation from the baseline is reported to the SOC [[2]](#footnote-1)for action.

All critical components must be monitored at all times, and SIEM [[3]](#footnote-2) leveraged to discover anomalies and ensure they are addressed in a timely fashion.

# Governance, Risk, & Compliance

A documented process must be followed to ensure compliance to security policy and to highlight risks that might be introduced when security policy requirements are not adhered to.

# Milestones

## Lorem ipsum

Lorem ipsum dolor sit amet, consectetuer adipiscing elit, sed diam nonummy nibh euismod tincidunt ut laoreet dolore magna aliquam erat volutpat.

## Dolor sit amet

Lorem ipsum dolor sit amet, consectetuer adipiscing elit, sed diam nonummy nibh euismod tincidunt ut laoreet dolore magna aliquam erat volutpat.

# 

# 

# Consensus Protocols

Following are the predominant consensus protocols that are use by the distributed ledger implementations.

|  |  |
| --- | --- |
| **CONSENSUS PROTOCOL** | **OVERVIEW** |
| Proof of Work | Uses computational power to validate new blocks of data.  To participate in this scheme, participants are required to collate transactions within a single block and then apply a hash function with the use of some additional metadata. |
| Proof of Stake | Validators (special nodes) voting on valid blocks whilst posting collateral in order to be able to participate in the validation process.  Unlike Proof of Work, Proof of Stake relies on proving the user is invested in the underlying token of value of the network being mined rather than being the owner of a large amount of computing power |
| Ripple Protocol | In order to validate new transactions, servers amalgamate outstanding transactions into a “candidate list.”  All participants then vote on valid transactions to be included in the ledger.  Transactions that meet the 80% threshold of “yes” votes are included within the following last closed ledger state. |
| Proof of Elapsed Time | As part of its Intelledger proposal, Intel has devised a means of establishing a validation lottery that takes advantage of the capability of its CPUs to produce a timestamp cryptographically signed by the hardware.  Whoever in the chain has the next soonest timestamp will be the one to decide which transactions will be a part of the next block in the chain.  This consensus method is extremely energy efficient compared to Proof of Work and therefore more adapted to IoT devices. |

# References

Writing Position Papers: <http://www.studygs.net/wrtstr9.htm>

Cover Page graphic: <https://www.pinterest.com/pin/352758583290504850/>

1. <https://www.ideals.illinois.edu/bitstream/handle/2142/97207/hildenbrandt-saxena-zhu-rodrigues-guth-daian-rosu-2017-tr.pdf> [↑](#footnote-ref-0)
2. Security Operations Center [↑](#footnote-ref-1)
3. SIEM - Security Information & Event Management [↑](#footnote-ref-2)