Dal Vangelo secondo Giorgio

Giorgio Mecca

16 settembre 2021

Sommario

...

Indice

Capitolo 1

Introduzione

1.1 Descrizione del Progetto

1.2 Descrizione dell'azienda

Capitolo 2

Blockchain

2.1 Problema dei generali bizantini

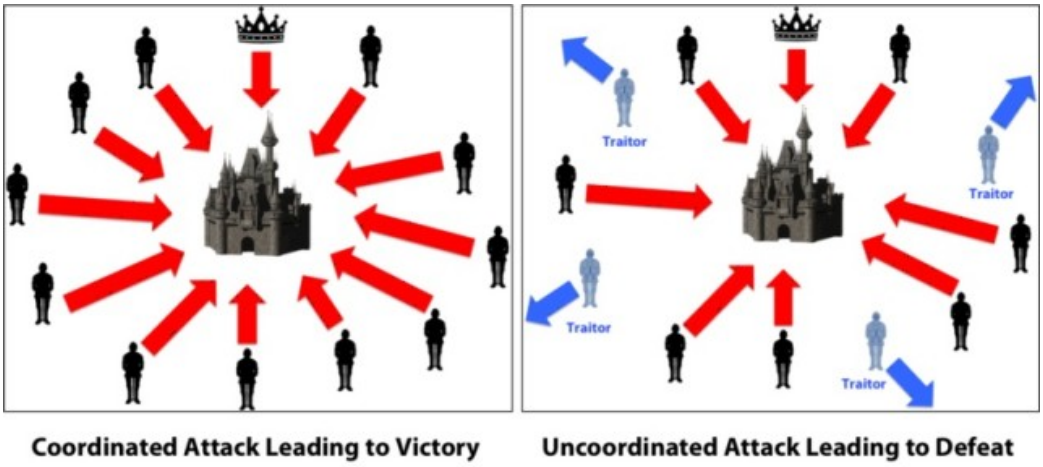

Figura 2.1: Problema dei generali bizantini

Il problema dei generali bizantini è un problema informatico su come raggiungere il consenso in situazioni in cui è possibile la presenza di errori. Il problema consiste nel trovare un accordo, comunicando solo tramite messaggi, tra componenti diversi nel caso in cui siano presenti informazioni discordanti. Il problema è stato teorizzato dai matematici Leslie Lamport,

Marshall Pease e Robert Shostak nel 1982, i quali crearono la metafora dei generali, caso di studio molto utilizzato nei sistemi basati o che comunque utilizzano una network. La metafora si basa su diversi generali che durante un assedio sono sul punto di attaccare una città nemica. Essi sono dislocati in diverse aree strategiche e possono comunicare solo mediante messaggeri al fine di coordinare l'attacco decisivo (Figura 2.1). I generali possono attaccare o ritirarsi, l'importante e che ci sia una decisione unanime, l'utilizzo di sola metà forza bellica porterebbe ad una sconfitta o una perdita. Il problema risiede quindi nell'alta probabilità che tra questi vi sia un generale traditore che mandi messaggi che vanno contro la strategia dell'esercito. La possibile soluzione punta al trovare un meccanismo secondo il quale un generale non traditore che riceva più messaggi sappia riconoscere quello veritiero. Secondo l'articolo di Lamport, Shostak e Pease non esiste una soluzione se il numero di processi non corretti è maggiore o uguale a un terzo del numero totale di processi. Una soluzione proposta è quella di Nakamoto che in una sua stesura sulla blockchain descrive un meccanisco per arrivare al consenso chiamato PoW Proof of Work.

2.2 Struttura di una blockchain

Una Blockchain come suggerisce l'etimologia della parola è una catena di blocchi o DLT - Distributed Ledger Technology. È una struttura dati formata da un insieme di blocchi(struttura prioritaria) collegati univocamente 1 ad 1 così da creare una metaforica catena. Una blockchain è considerata una struttura condivisa e immutabile in quanto il suo contenuto una volta scritto non è più né modificabile né eliminabile, a meno di non invalidare l'intera struttura. Questa tecnologia fà parte dei Distributed Legder cioè dei "libri mastro distribuiti" o registri condivisi infatti tutti i partecipanti della blockchain, detti anche nodi, posseggono lo stesso registro cioè le stesse informazioni andando a costruire il contrapposto di una struttura centralizzata come un Database, quindi una struttura Decentralizzata in cui ogni

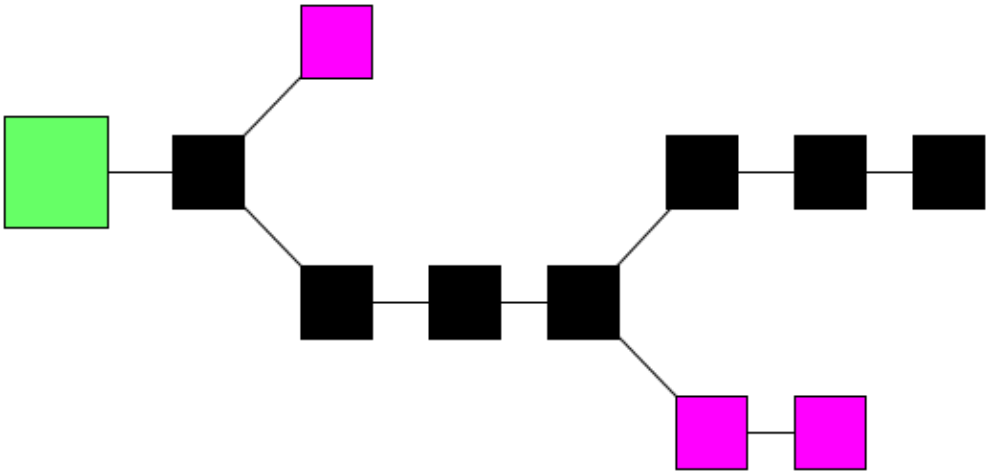

Figura 2.2: Rappresentazione struttura di una blockchain

nodo ha la possibilità di leggere autonomamente le informazioni contenute. Nella figura 2.2 viene visualizzata una semplice blockchain in cui sono presenti tre tipologie di blocchi quali, il blocco verde visto come il blocco di genesi, i blocchi neri che vanno a costituire la catena principale e i blocchi viola considerati blocchi orfani. L'aggiunta di un nuovo blocco è globalmente regolata da un protocollo condiviso e se autorizzata ogni nodo aggiorna la propria copia privata del registro così da evitare manipolazioni future. In una blockchain i nodi partecipanti vengono anche chiamati minatori- miner o validatori, riferendosi al loro compito nella rete rispetto ai blocchi.

2.3 Hashing

Un codice hash è una qualunque sequenza di caratteri alfanumerici generati da una particolare funzione di hash. Questa funzione prende in input un qualunque tipo di informazione e restituisce una stringa di lunghezza prefissata, questo rende la funzione one-way o non invertibile in quanto conoscendo il digest(codice hash restituito) non è possibile risalire all'informazione che

lo ha generato. In una blockchain l'Hash viene utilizzato per la costruzione della catena, viene calcolato l'hash di un blocco e il blocco che lo succederà avrà come parametro questo hash. In questo modo ogni blocco è legato univocamente al blocco precedente e siccome il codice hash di un blocco viene calcolato utilizzando anche il codice hash precedente modificando un singolo blocco verrà invalidata tutta la struttura blockchain immediatamente successiva.

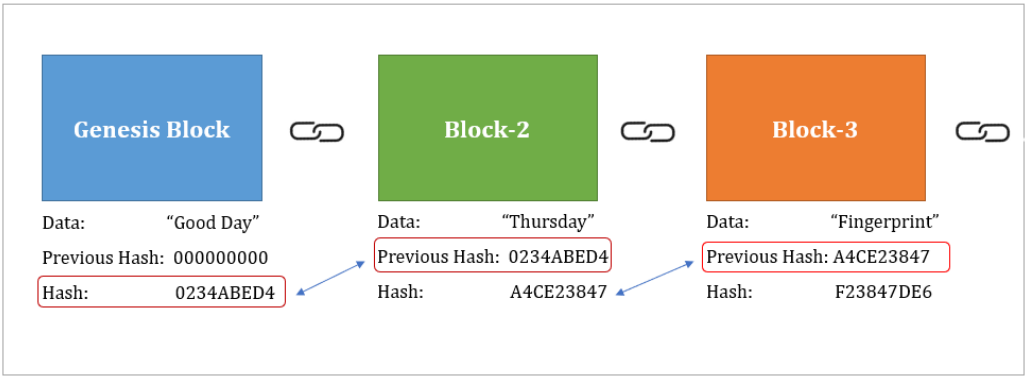

Figura 2.3: Catena di una blockchain

2.4 Transazioni

In una blockchain i dati vengono scritti sotto forma di Transazioni in seguito contenute in vari blocchi. L'uso più comune delle transazioni è l'invio di denaro o in qualche modo una moneta equivalente. Le transazioni devono quindi avere un mittente, un destinatario, un 'value' cioè il valore trasmesso, vengono quindi considerate come un cambio di stato riferendosi alle informazioni nella blockchain e saranno identificate da un Transaction Hash. Una volta inviata, la transazione entra in un transaction pool, da dove i miner andranno a selezionare randomicamente transazioni da includere nel prossimo blocco. Una transazione per essere considerata valida deve essere accettata

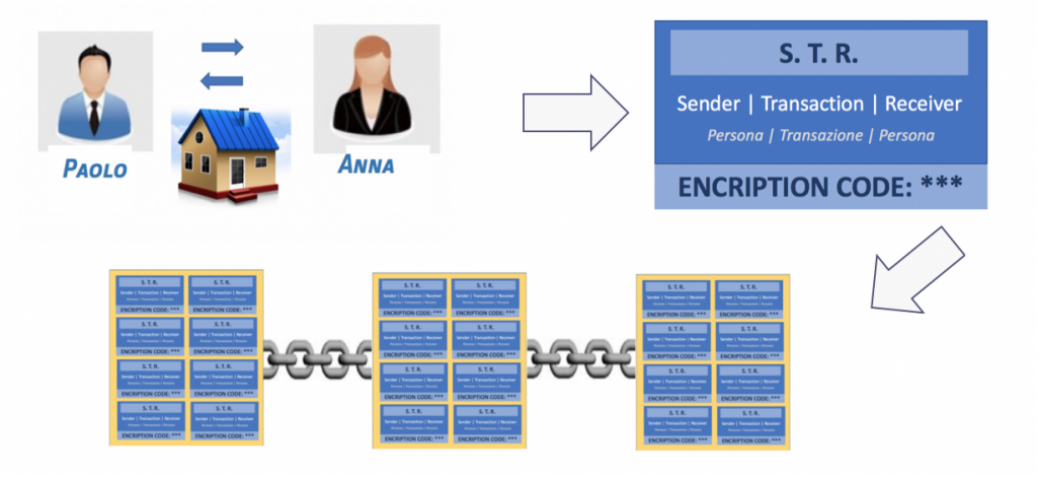

Figura 2.4: Flusso di esecuzione di una Transazione

da un nodo che la inserirà nel blocco che sta minando, non è certo che due nodi che minano lo stesso blocco la inseriscano nella stessa posizione.

2.5 Blocchi

La blockchain è una sequanza di blocchi che contengono una collezione di Transazioni. il numero di transazioni all'interno di ognuno di questi blocchi varia in base alla dimensione della transazione stessa.I blocchi sono prodotti dai nodi validators e vengono generati in un lasso di tempo definito dalle regole della blockchain(Es: 15 secondi per Ethereum, 10 min per BitCoin), nel momento in cui il blocco viene completato i dati contenuti diventano verificati.

Un Blolcco è diviso in due parti, l'header e il body. Le transazioni sono racchiuse nel body del blocco e nell'header sono presenti i campi di gestione del blocco stesso come descritto nella figura 2.5.

- Versione del blocco: indica le regole di validazione del blocco da rispettare

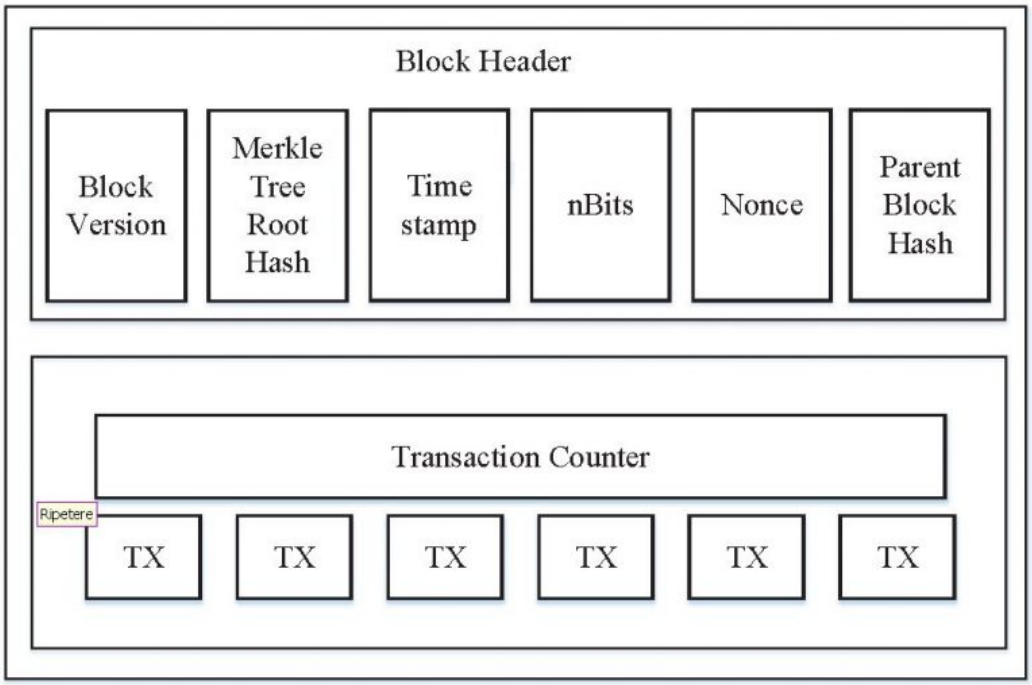

Figura 2.5: Struttura di un Blocco

- Merkle Tree Root Hash valore della radice del Merkle Tree in cui sono salvate le transazioni del blocco
- TimeStamp: Marca temporale salvata come Timestamp UNIX che indica l'inserimento del blocco nella blockchain
- nBits: è la soglia target di un hash di blocco valido
- Nonce: è un campo il cui valore è settato dai miner così che l'hash del blocco calcolato sia minore o uguale al target attuale della rete(difficoltà). Dato che non è possibile prevedere la combinazione di bit che risulterebbero nell'hash voluto, numerosi valori di nonce sono calcolati fino a quando l'hash risultante rispetti i requisiti attuali della rete.
- Parent Block Hash: segna l'hash del blocco a cui verrà agganciato

Il Body è composto da un contatore di transazioni (transaction counter) e dalle transazioni (TX), queste vengono memorizzate e organizzate tramite un Merkle Tree cioè una struttura ad albero in cui le foglie contengono i digest hash delle informazioni mentre i nodi contengono i digest hash dei nodi sottostanti(figli)

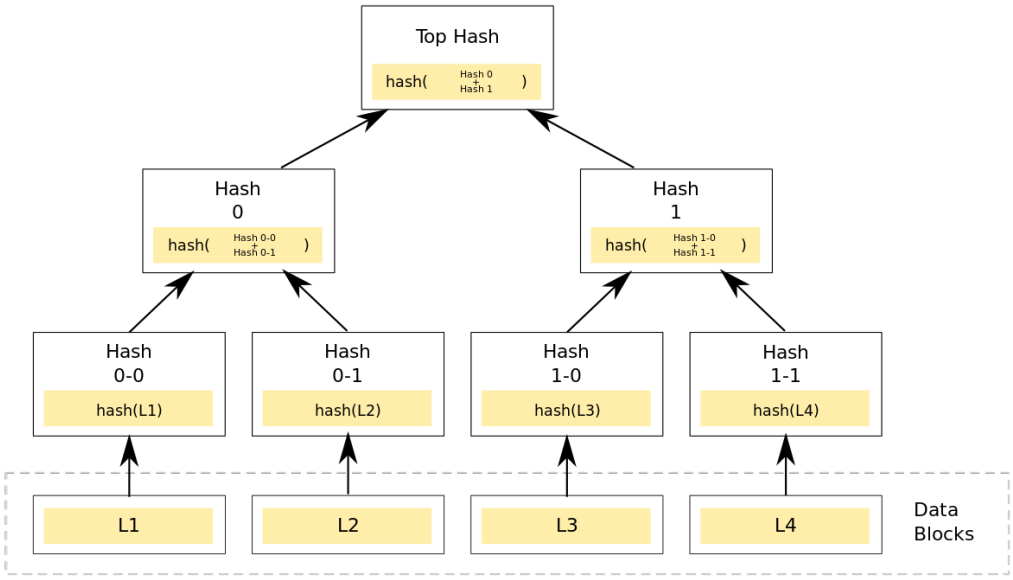

Figura 2.6: Merkle Tree

2.6 Mining e Meccanismi del consenso

I miner sono i nodi partecipanti alla rete che ascoltano le transazioni inviate, verificano che non siano malevole e compongono un blocco organizzando le transazioni in un Merkle Tree. I minatori che 'calcolano' un blocco valido vengono premiati con un incentivo (Es: criptovaluta) È anche possibile che più miner producano lo stesso blocco e la scelta del blocco da seguire per la catena spetta al meccanismo del consenso cioè il set di regole che permette

la finalizzazione delle transazioni e il funzionamento del sistema così che tutti i nodi della rete convergano ad una sola versione condivisa della catena, questo andrà a creare una biforcazione o Fork da cui si avranno una sequenza valida(quella che verrà continuata) e una sequenza(anche un singolo nodo) orfana(Figura 2.2).

2.6.1 Consenso Trustless

La blockhain è considerata Trustless, letteralmente "senza fiducia" poiché a differenza di un sistema centralizzando non esiste nessun ente centrale in cui riporre la fiducia(come ad esempio una Banca,governi o istituti finanziari). Utilizzando un sistema decentralizzato la fiducia non viene eliminata definitivamente ma viene suddivisa tra tutti i partecipanti, in questo modo più sarà alto il numero dei partecipanti meno fiducia si dovrà affidare ad ogni singolo nodo. La fiducia viene quindi riposta nei singoli nodi e che questi rispettino le regole considerate valide.

2.6.2 Proof of Work

L'algoritmo del consenso più famoso ed utilizzato(come da Bitcoin, Ethereum e Monero) nonché il primo algoritmo del consenso mai creato è il PROOF OF WORK abbreviato PoW. Questo si basa sulla "Prova del Lavoro" svolto dai miner che dovranno risolvere unauna serie di operazioni considerate u puzzle matematico per poter creare un blocco valido. Il primo miner che costruisce il blocco lo aggiunge alla catena, notifica in broadcast il resto della rete e di conseguenza tutte le transazioni in esso presenti vengono validate, infine il miner viene ricompensato con un incentivo come una moneta che sarà relativa quello che gli utenti pagano per effettuare le transazioni, cioè le "tasse" - fees. I miner quindi utilizzano la loro potenza e le risorse computazionali per provare che hanno svolto del lavoro(questo produce anche un massivo consumo di elettricità), talvolta è possibile che due o più nodi producano lo stesso blocco creando un fork e quindi due differenti catene. Il sistema è incentivato

a scrivere nuovi blocchi sulla catena più lunga così da eliminare la biforcazione orfana e ricondurre tutta la rete a un'unica catena, questo implica che se si voglia creare un blocco "falso" e farlo accettare dalla blockchain bisognerà possedere una potenza di calcolo maggiore di tutta la rete. Il mining diventa sempre più competitivo con la partecipazione di sempre più gente e con l'aumentare della difficoltà, perciò si è vista la creazione delle Mining Pool cioè un insieme di persone che raggruppa la propria potenza di calcolo per il mining di criptomoneta, in questo modo la probabilità di costruire un blocco valido aumenta e di conseguenza aumenta anche il guadagno suddiviso tra i partecipanti

2.6.3 Proof of Stake

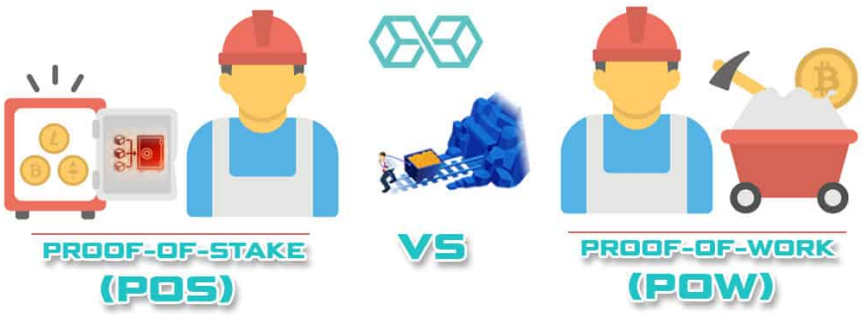

Figura 2.7: PoW-PoS

Nel 2011 basandoci sui problemi del PoW come il dispendio di energia elettrica o la creazione di grosse Mining Pool che elimina la decentralizzazione si è sviluppata l'idea del PoS Proof of Stake. Il PoS sostituisce i miner con i validatori o coniatore, questi per far si che il loro blocco venga considerato valido devono depositare della moneta come "cauzione" che viene appunto chiamata Stake. In questo modo se con il PoW era possibile inserire transa-

zione fraudolente qui si andrebbe a perdere la somma congelata. Il criterio di scelta dei validatori si basa sulla quantità di moneta bloccata la durata del blocco, con il secondo parametro si va in contro al problema generato dal primo parametro cioè che solo i più ricchi possono essere scelti e quindi diventare più ricchi. L'elezione del blocco leader (da aggiungere) avviene tramite l'algoritmo VFR – Verifiable random function che utilizza l'algoritmo "follow-the-coin" - "più denaro blocchi più hai fiducia". Il PoS porta molteplici vantaggi rispetto al PoW come un minimo dispendio energetico e una maggiore sicurezza e decentralizzazione dovuta all'assenza di mining pool.

2.6.4 Proof of Authority

2.7 Attacchi

2.7.1 Selfish Mining Attack

2.7.2 Double Spending Attack

2.8 Blockchain Pubbliche/Private

2.9 Etherium

2.9.1 Smart Contract

2.9.2 Solidity

2.9.3 Gas

2.9.4 Dapps

Capitolo 3

Tecnologie utilizzate

3.1 Besu

3.1.1 IBFT

IBFT Methods

3.1.2 Free Gas Network

3.1.3 API Methods

3.2 Truffle

3.2.1 Compile

3.2.2 Test

3.2.3 Deploy

3.3 Node js

3.3.1 Web3

Capitolo 4

Caso d'uso

4.1 Problema Iniziale

Il progetto blockchain è stato ideato e sviluppato come proposta di soluzione ai problemi nella gestione del tracciamento, invio, certificazione e mantenimento di dati riguardanti "spostamenti". Questi spostamenti sono informazioni(LOG) inviate da un qualunque ente che metta a disposizione della società o di un qualunque servizio che preveda dei mezzi pubblici o privati quali ad esempio autobus, treni, taxi etc... Queste informazioni vengono ora raccolte, analizzate e utilizzate su di modelli di storage a fogli di calcolo (Ad Esempio EXCEL - Programma Micrsoft). I fogli di calcolo offrono alcuni vantaggi come la semplicità con cui vengo creati, scritti e salvati benché offrano un'interfaccia poco user friendly('facilmente utilizzabile') guardando tutti i possibili attori, mentre il progetto si focalizza sui difetti come la pubblicazione/condivisione delle informazioni o l'interrogazione di queste in quanto utilizzando semplici fogli non vengono proposte alcune regole di struttura e organizzazione,e ci si pone particolare importanza alla sicurezza e all'affidabilità di queste informazioni e che non vengano modificate durante la condivisione, quindi la possibile certificazione di essi.

4.2 Soluzione

La soluzione proposta si offre di risolvere tutti i problemi sopra elencati come la certificazione, salvataggio e interrogazioni di informazioni. Viene ideata una blockchain privata che avrà funzione di ente (decentralizzato) certificatore, questa non utilizza nessuna moneta creando una Free Gas Network e che con l'ausilio di appositi smart contract(scritti e caricati autonomamente) ci permette di salvare un codice che andrà ad identificare un determinato gruppo di spostamenti come un codice Hash che usufruendo della struttura e utilizzo della blockchain non potrà essere modificato, ciò implica che si potrà sempre verificare la correttezza del gruppo di spostamenti richiesti ricreando e controllando il loro codice.

La memorizzazione dei dati viene invece affidata ad un database relazionale, utilizzando nel progetto il DBMS(DataBase Managenent System) MySql, che ci permette di salvare grandi quantità di dati con una efficiente organizzazione gestita con la creazione di tabelle così da essere facilmente interrogabile in futuro.

L'interfaccia comune è gestita con un server sviluppato tramite tecnologia node js che con una single Page Application avrà la funzione di interfaccia user friendy con funzioni di memorizzazione per i log degli spostamenti su Database, calcolo e salvataggio dei loro codici hash sulla blockchain al tempo stimato e quando necessario, cioè quando e ovunque verranno richiesti dei dati e avverrà l'interrogazione del DataBase sarà reso obbligatorio il controllo di questi con il codice sulla blockchain. Inserendo questo WebServer intermedio o server proxy si andrà ad eliminare il passaggio di dati non propriamente protetto e rende partecipi tutti i singoli attori dell'attività.

4.2.1 Problematiche

Utilizzando delle nuove tecnologie sorgono comunque nuove problematiche che non sono state affrontate nello sviluppo in quanto non inerenti ai fini del progetto.

Una prima problematica si sviluppa utilizzano un server proxy. Avendo un singolo server di accesso al database a alla blockchain sequesto non viene correttamente protetto e costantemente controllato è soggetto ai classici attacchi come un DDOS - Distributed Denial of Service in cui si utilizzano molteplici messaggi fittizi (come un inizio di HandShake per una connessione TCP) per far si che il server non possa sostenere tutti i servizi e essendo l'unico punto di accesso bloccherebbe l'intero accesso alla rete blockchain.

Una caratteristica che rende sicura la blockchain pubblica è la molteplicità di nodi, questa con una blockchain privata come la nostra va a decadere con il discendere del numero di nodi; utilizzando un meccanisco di conseso basato su PoA(Proof of Autority) si ha infatti bisogno di un minimo di 4 nodi per essere resistente al problema bizantino.

Per evitare l'appesantimento della Blockchain si è pensato di salvare su di essa solo un codice identificativo(codice Hash) per un gruppo di Log. Questo implica che con l'aumentare dei log identificati da un singolo codice hash diminuisca la sicurezza che questo apporta infatti sarà più facilmente utilizzabile un attacco come l'attacco del compleanno che ha come obiettivo quello di generare una collisione cioè di trovare dei dati fittizi ai Log originari che però generano lo stesso codice Hash, questi dati fittizi potranno essere quindi sostituiti nel DB ma verranno comunque considerati certificati dal sistema in quanto produrranno lo stesso codice.

4.3 Attori

Il caso d'uso per il progetto blockchain prevede la partecipazione di diversi attori quali:

Un Terminal User o utente finale è un comune dipendente di un ente che partecipa alla blockchain il quale ha il compito di comunicare i propri spostamenti/Log o qualunque informazione di cui si preveda il salvataggio;

Gli Admin sono dei dipendenti di enti partecipanti che vengono segnati dagli stessi come amministratori che quindi posseggono particolari oneri come

il possesso e la trasmissione di una chiave privata;

Il proprietario/gestore della blockchain avrà il compito di gestire l'intera blockchain privata con l'amministrazione che ne segue, come la supervisione dei nodi presenti, il loro funzionamento e la loro caratterizzazione come validatori.

4.4 Scenario di utilizzo

Il Progetto prevede uno scenario di utilizzo diverso seguendo la distinzione degli attori. Per l'utilizzo si prevede che ad ogni ente partecipante al progetto gli venga assegnato un account, cioè una copia di chiavi privata e pubblica che serviranno per interagire con la blockchain, inoltre ogni ente dovrà inserire i propri dipendenti nel Database e specificare il ruolo di essi, se admin o Terminal User.

Un Terminal User, una volta effettuato l'accesso, viene portato ad un'interfaccia in cui può inserire la città che sarà selezionata come Start dello spostamento e in seguito viene spostato in una seconda interfaccia da cui può terminare lo spostamento o annullarlo, se annullato potrà cominciare un nuovo spostamento dalla precedente interfaccia, il completamento di questo avverrà solo se compila i campi necessari quali la città di Termine e la distanza percorsa indicata in Kilometri.

Un admin, una volta effettuato l'accesso, potrà a differenza di un Terminal User effettuare delle query/ interrogazioni riguardo gli spostamenti compiuti, inserendo una data otterrà tutti gli spostamenti che sono stati certificati da una transazione inserita in quella determinata data, da qui potrà anche accedere ai dettagli della transazione o del blocco che la contiene riferendoci alla blockchain, inoltre, quando il sistema lo richiede, ha il compito di inserire la Private Key dell'utente(ente) che verrà utilizzata per la scrittura su blockchain.

Capitolo 5

Sviluppo

5.1 Schema Progetto

5.2 Blockchain Ibrida

5.3 Smart Contract

5.3.1 Boxing

5.4 WebApp

5.4.1 Single Page Application

5.4.2 Input

Inserimento in un DB

Inserimento nella Blockchain

5.4.3 Output

Report di Controllo

Monitor Blockchain

Capitolo 6

Sviluppi futuri

6.1 Analisi costi

6.2 Immissione nella blockchain pubblica

6.3 Blockchain pubblica come certificazione

6.4 Svilupo full Blockchain

Elenco delle figure