UNIVERSITÀ DEGLI STUDI DI MILANO
FACOLTÀ DI SCIENZE E TECNOLOGIE

Laurea triennale in sicurezza dei sistemi
e delle reti informatiche

Analisi della tecnologia blockchain: protocolli e sicurezza

Relatore
Prof.ssa Chiara Braghin

Candidato
Matteo Rota
Matricola 941561

Anno Accademico 2021-2022

Prima di tutto, vorrei ringraziare la mia famiglia e i miei cari per il costante sostegno fornito in questi anni e per i sacrifici sostenuti per rendere questo obiettivo una realtà.
Ci tengo anche a ringraziare la professoressa *Chiara Braghin* per l'assistenza fornita durante il periodo di stesura dell'elaborato finale.
Infine, vorrei esprimere la mia gratitudine a tutti gli amici che mi hanno aiutato durante i momenti difficili di questo splendido viaggio.
Senza il vostro aiuto, questo traguardo non sarebbe stato raggiungibile.
Grazie di cuore.

Matteo

Indice

Figure

Capitolo 1 - Introduzione

Blockchain e criptovalute sono termini che negli ultimi anni vengono utilizzati spesso, ma il cui significato non è sempre noto ad un'alta percentuale della popolazione, che comunque non ha ben chiara la tecnologia sottostante.
In termini informali, la blockchain è definita come un registro condiviso ed immutabile, dove qualunque cosa venga scritta non è più cancellabile.
Fino a qualche anno fa, l'accesso e l'utilizzo della tecnologia blockchain, e relative criptovalute, era molto ristretto e necessitava di competenze, a livello informatico, avanzate. Negli ultimi anni però abbiamo assistito ad una crescita esponenziale della tecnologia Blockchain, pilastro portante alla base di ogni criptovaluta, oramai applicabile in qualsiasi ambito e sotto svariate forme. Complice la pandemia in corso, stiamo attraversando un'epoca dove la società si sta digitalizzando sempre più, sotto tutti gli aspetti, come per esempio passando da menù cartacei nei bar e ristoranti a quelli digitali tramite codice QR, oppure ancora da riunioni di presenza a conferenze in videochiamata, per non parlare dell'attenzione maggiore allo *status symbol* sulle piattaforme digitali piuttosto che la presenza nella vita quotidiana.

Un cambiamento radicale, che magari non è fin da subito evidente, riguarda la transizione in corso per passare da denaro contante a metodi di pagamento elettronici. La principale differenza tra i due, escludendo la modalità effettiva fisica di pagamento, riguarda la tracciabilità delle operazioni: con le vecchie banconote, non è possibile consultare uno storico di tutte le transazioni effettuate, né tantomeno chi le ha possedute; in contrapposizione, i pagamenti elettronici ci permettono di avere dettagliate informazioni su ogni singola transazione, tra cui il beneficiario e l'emittente.
Se ci fermassimo qui, non vi sarebbero evidenti problematiche riguardo l'utilizzo di mezzi per effettuare pagamenti elettronici.
Tutto questo è possibile grazie alla presenza di un ente centrale, in questo caso la banca, che agisce da governante del sistema.
Ma quindi, cosa succederebbe in caso di compromissione, o addirittura del fallimento della banca stessa?
Ci sono problematiche irrisolte anche a livello temporale, poiché non è possibile inviare del capitale da un conto ad un altro istantaneamente, ma è necessario attendere qualche giorno. Per non parlare delle festività o anche fine settimana, durante i quali è impossibile logisticamente effettuare transazioni. Il classico problema, che tutti hanno riscontrato almeno una volta nella vita, riguarda la necessità di ricaricare carte prepagate da un altro conto: spesso non è istantaneo e avviene solo quando le banche sono aperte, limitando di fatto il potere di acquisto di un singolo.
Un altro problema comune può riguardare l'approvazione delle transazioni da parte della banca, che per svariati motivi, può bloccarle o addirittura rifiutarle.

Infine, sono presenti forti limiti territoriali: non tutti gli stati utilizzano la stessa valuta. Di conseguenza, nel caso ipotetico di invio di denaro da un continente ad un altro, sarebbe necessario cambiare valuta, operazione che comporta dei costi, oltre che ritardi. Analizzando più nel dettaglio le problematiche appena esternate, ci si rende conto di alcuni forti limiti invalicabili.
Analogamente alla definizione informale di blockchain fornita, il termine criptovaluta è identificabile come una rappresentazione digitale di valore, decentralizzata, ed utilizzabile come mezzo di scambio. Le monete virtuali vengono incontro a queste necessità e cercano di colmare le lacune di sistemi consolidati, ma troppo chiusi e restrittivi al giorno d'oggi. L'obiettivo principale che viene posto è quello di restituire un alto grado di libertà sul proprio capitale, senza controlli da enti centrali o limiti di carattere temporale e spaziale.
La tecnologia blockchain ha permesso lo sviluppo di innumerevoli criptovalute, alcune ben consolidate ed altre in via di sviluppo, che molto probabilmente andranno ad integrarsi con i sistemi ai quali siamo abituati al giorno d'oggi.

L'altra grande rivoluzione, alla quale stiamo assistendo, riguarda la virtualizzazione del mondo in cui viviamo. L'obiettivo è quello di creare una realtà virtuale condivisa in tre dimensioni, nella quale è possibile possedere una propria abitazione, beni di valori, opere d'arte, animali e tutto ciò che è possibile immaginare. Come anticipato, l'importanza della propria apparenza nel digitale sta acquisendo sempre più valore. Di conseguenza, la creazione e il mantenimento di una seconda vita virtuale diverrà un aspetto crescente nei prossimi anni, al quale risulterà complicato sottrarsi. Le persone inizieranno ad investire nel proprio mondo virtuale, acquistando beni e servizi ad esempio, proprio come siamo abituati a fare nella realtà. Per molti questo può risultare un'idea inutile ed irrealizzabile, ma lo stesso pensiero c'è stato anche con il commercio dei primi computer, l'avvento di Internet, l'utilizzo dei primi cellulari mobili e social media. É solo una questione di tempo.

Progettata dall'uomo, questa struttura dati innovativa non è esente da problemi di sicurezza. Se da una parte la libertà di gestione dei beni torni sotto il controllo del proprietario, dall'altra si va incontro a responsabilità maggiori. Attualmente sono presenti molteplici tipologie di attacco: alcune di esse sfruttano falle nella progettazione della blockchain, altre si avvalgono della carenza di attenzione da parte degli utenti.

Considerando le premesse sopra citate, questa tesi si pone come obiettivo l'analisi dettagliata degli aspetti e strumenti principali basati sulla tecnologia blockchain, per permettere un'adeguata comprensione delle potenzialità, dei vantaggi ma anche dei rischi annessi.

Capitolo 2 - Tecnologia blockchain

2.1 Architettura e caratteristiche

La *blockchain*, letteralmente “catena a blocchi, è identificabile come un registro digitale, immutabile e condiviso. Come intuibile dal termine, questa struttura è formata da un insieme di voci, dette blocchi, concatenati tra loro in ordine cronologico. La peculiarità di tale tecnologia risiede nel fatto che l’autenticità viene sempre garantita: una volta che viene aggiunto un blocco alla struttura, non è più possibile modificarlo o eliminarlo, a meno di non invalidare tutta la catena. Per quanto riguarda invece l’integrità, essa è assicurata dalle funzioni crittografiche utilizzate e da una marca temporale, come per esempio il *timestamp*. Di fatto, l’esistenza della blockchain permette a un individuo qualunque di poter verificare che una specifica transazione sia avvenuta in un determinato momento.
La grande differenza che contraddistingue la blockchain da altre reti è la modalità di conservazione dei dati. Per comprendere meglio questo aspetto, è opportuno ricordare ed evidenziare le differenze delle tre topologie di rete principali: centralizzati, distribuiti e decentralizzati. Un sistema centralizzato prevede che tutti i dati passino da un nodo centrale, il quale autorizza le operazioni in modo autonomamente. Si parla invece di ambienti distribuiti quando il sistema è progettato su un numero di macchine definito, collocate geograficamente lontane tra loro, che necessitano di cooperare per raggiungere un risultato. Solitamente ogni nodo mantiene una parte dei dati, come per esempio nel caso di un database distribuito su diverse macchine. Invece, un sistema decentralizzato non richiede un determinato numero di nodi a priori, poiché la rete è formata da un insieme di host, interconnessi, che danno vita ad una rete peer-to-peer, la quale non richiede un’autorità centrale. L’unicità di questa architettura risiede nel fatto che viene creata una rete nella quale ogni nodo contiene la stessa copia esatta dei dati. Di fatto quindi c’è anche una distribuzione a livello spaziale delle singole macchine.
La tecnologia in questione sfrutta proprio quest'ultima architettura per la gestione dei dati e del potere decisionale governativo.

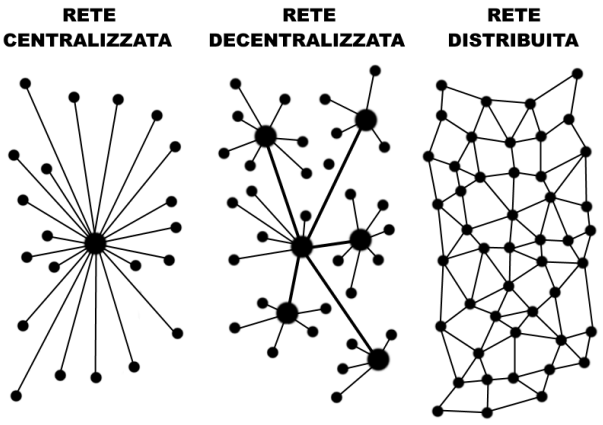

Figura 1 - *Rappresentazione grafica delle varie topologie di reti*

Inoltre, utilizzando una rete decentralizzata e distribuita, tutti i nodi sono paritari e non necessitano di conoscere le varie identità reciproche. Si può quindi assicurare un alto livello di sicurezza della rete: la mancanza di un ente centrale impedisce la necessità di dover riporre la fiducia in un singolo nodo. Nessuno è in grado di prevalere sull'altro e tutti hanno gli stessi ruoli e gli stessi diritti di accesso ai dati.
Le altre due caratteristiche fondamentali che accomunano i sistemi basati su blockchain sono riconducibili alla trasparenza e tracciabilità. Esse necessitano un certo grado di coesione, poiché l'una senza l'altra risulterebbe vana. Infatti, la trasparenza è garantita dalla topologia della rete stessa: decentralizzata e distribuita. Di fatto, i dati sono accessibili da chiunque. L'altra proprietà complementare, la tracciabilità, consente di verificare in qualunque momento le informazioni relative ad una determinata transazione. Importante è ribadire che non si corre il rischio di analizzare dati fraudolenti, siccome questa struttura non è mutabile.
L'aggiunta di un nuovo blocco nella catena è regolamentata da un protocollo di validazione ben definito, specifico per ogni blockchain. Nel capitolo successivo, andrò ad approfondire dettagliatamente la struttura di ogni singolo blocco, la logica di concatenamento e il funzionamento effettivo della struttura. Per adesso, possiamo rappresentare la blockchain come un insieme di blocchi concatenati, ognuno dei quali punta a quello precedente, includendo nel suo contenuto anche l'impronta digitale del blocco antecedente.

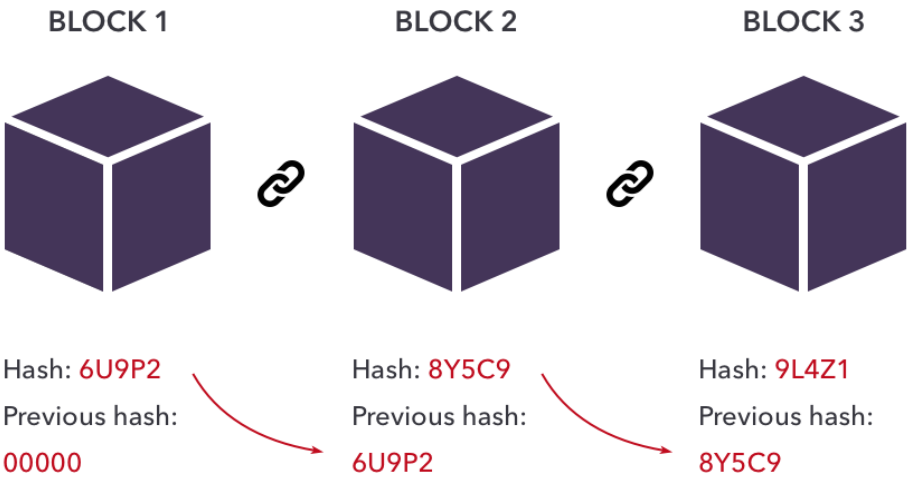

Figura 2 - *Rappresentazione schematica della struttura blockchain*

2.2 Funzioni hash

Il marchio univoco, citato in precedenza, è calcolato attraverso una funzione crittografica di *hash*. Questi algoritmi sono predisposti per mappare dati di lunghezza arbitraria in stringhe di dimensioni fisse, dette anche *digest*. La peculiarità principale di questa famiglia di funzioni risiede nel fatto che esse sono unidirezionali, ovvero difficili da invertire. Dato un determinato *digest*, non è possibile risalire ai valori di input. A livello di computazione, questi algoritmi permettono una verifica del risultato immediata, noti i dati in ingresso.

Riassumendo, una funzione ideale deve godere di quattro proprietà per essere collocata nella famiglia *hash*: deve identificare il messaggio in maniera univoca, due messaggi differenti non possono avere lo stesso risultato; deve essere prevedibile, gli stessi dati devono fornire lo stesso *digest*; deve essere rapida da verificare e garantire la quasi impossibilità del risalire al messaggio originale, partendo dal solo valore di hash.
La problematica principale riscontrata in questa tipologia di funzioni è chiamata collisione.

Da un punto di vista puramente concettuale, possiamo vedere questi algoritmi come vere e proprie funzioni matematiche. Come ognuna di esse, anche gli algoritmi di hash hanno un proprio dominio e codominio [Figura 3], i quali sono rappresentati rispettivamente dall'insieme dei messaggi di lunghezza variabile (M), e dei digest di dimensione fissa (h).

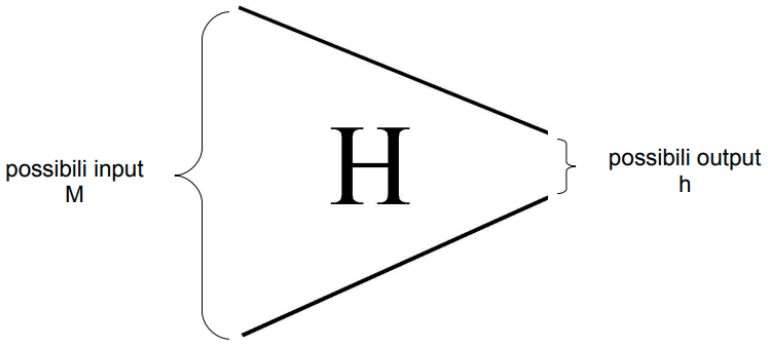

Figura 3 - *Schema concettuale funzione hash*

Dalla rappresentazione grafica, si evince facilmente che il numero di possibili casi in ingresso è nettamente maggiore della quantità dei possibili digest ottenibili.
La lunghezza fissa del risultato è un fattore determinante nella sicurezza e robustezza della funzione: maggiore è il numero di bit che lo compongono, minori saranno le possibilità di avere una collisione. Quest'ultima viene a crearsi quando due messaggi differenti producono lo stesso risultato. Tutte le funzioni hash, che prendono in ingresso messaggi di dimensione arbitraria, dovranno necessariamente fronteggiare alcune collisioni, anche se solitamente sono molto difficili da trovare.
Il problema è arginabile ricorrendo a funzioni hash che producono digest maggiori di 256 bit. Per sostenere questa tesi, prendiamo in esame un algoritmo che produce risultati composti da pochi bit, come 32 o 64, ed anche il noto "paradosso del compleanno", definito da Richard Von Mises [1].
Quest'ultimo afferma: "La probabilità che almeno due persone in un gruppo compiano gli anni lo stesso giorno è largamente superiore a quanto potrebbe dire l'intuito: infatti, in un gruppo di 23 persone la probabilità è già del 51% circa e con 30 si supera il 70%".
L'omonimo "attacco del compleanno" sfrutta la teoria matematica alla base del paradosso appena citato. Esso tenta di trovare due messaggi indipendenti, solitamente si parla di originale e di contraffatto, che però producano lo stesso digest. In termini pratici, viene generato un gran numero di varianti del messaggio autentico e anche di quello contraffatto, si applica la funzione hash su tutti gli elementi ed infine si cercano eventuali digest uguali, nonostante i dati in input differenti. Stiamo parlando a tutti gli effetti di un attacco a forza bruta, che però permette di ridurre i tempi in maniera notevole: definita *n* come lunghezza

del risultato prodotto, ci basta calcolare $2^{n/2}$ hash per avere un'alta probabilità di trovare due messaggi che collidono, senza dover esaminare tutti i 2^n casi possibili.

2.3 Struttura del blocco

L'unità più piccola nella struttura blockchain è rappresentata dal singolo blocco. Esso archivia un insieme di transazioni, le correla da un marcatore temporale e da un hash, identificativo del blocco.
Principalmente, un blocco è composto da due parti distinte: l'*intestazione* e il *corpo.*
Nell'intestazione, detta anche *header*, sono presenti i dati di gestione del blocco stesso:

- Versione
- Radice di Merkle
- Timestamp
- Target
- Nonce
- Numero transazione
- Hash blocco precedente

Il primo campo, *versione*, serve per specificare la versione del protocollo blockchain che il blocco deve seguire, per essere ritenuto valido.
Il campo *radice di Merkle* identifica, con un singolo valore di hash, tutte le transazioni presenti all'interno del blocco. Esso sfrutta la struttura dati definibile come albero hash, o di Merkle, la quale ha lo scopo di facilitare la verifica di grandi quantità di dati. Per realizzare l'albero binario, il valore di *digest* di ogni singola transazione rappresenta logicamente una foglia. Viene quindi applicato l'algoritmo ricorsivo per il calcolo della *radice di Merkle*: ogni nodo padre è etichettato dal risultato di una funzione hash, che prende in ingresso i digest delle foglie sottostanti. La radice si raggiunge quando il nodo si trova nel livello più alto, quindi in assenza di padre. Grazie a questa struttura, è possibile verificare velocemente se anche solo una transazione nel blocco è stata alterata, quindi garantire integrità.

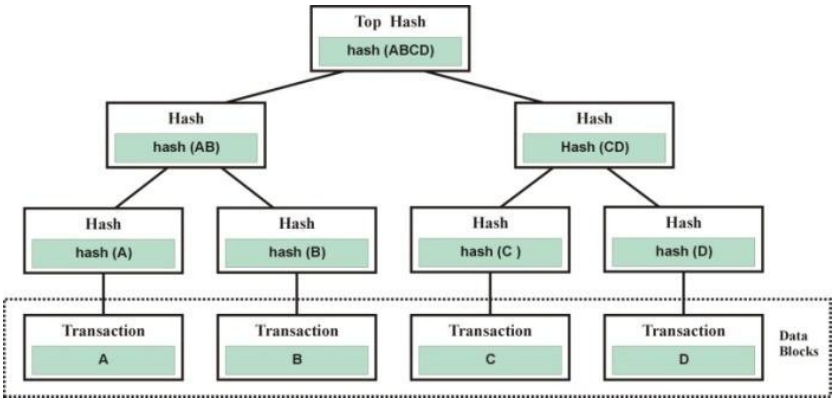

Figura 4 - *Rappresentazione albero di Merkle*

Il terzo campo, *timestamp*, rappresenta la data e l'orario dell'ultima transazione presente all'interno del blocco. Esso viene calcolato attraverso l'algoritmo *Unix timestamp* ed espresso in forma esadecimale. Il terzo dato, *target*, indica l'obiettivo di difficoltà da raggiungere: viene spesso definito anche come *bits* e specifica il valore massimo che l'hash dell'*header* del blocco può avere. Più questo numero è grande, maggiori sono le possibilità che l'hash calcolato sia minore del *target*, quindi di più facile computazione. Questo valore è variabile nel tempo ed è di fondamentale importanza per regolare la frequenza con la quale un nuovo blocco è aggiunto alla rete: se il valore è particolarmente basso, sarà necessario più tempo e potenza di calcolo per ottenere un hash inferiore al *target.* Da notare il fatto che il target viene rimodulato dopo l'aggiunta di un numero definito di blocchi, variabile a seconda del protocollo. Viene calcolato il rapporto tra il tempo effettivo di calcolo e quello stimato: se i blocchi sono stati prodotti troppo rapidamente il *target* verrà abbassato, in modo da rendere più laboriosa la verifica di un'unità. Arrivati a questo punto, sorge spontaneo domandarsi quale sia la variabile all'interno del blocco, che ci permette di ottenere i valori *digest* sufficientemente bassi, dato che le transazioni contenute sono sempre le stesse. Il prossimo campo, *nonce*, contiene proprio un valore variabile, ad 8 bit, che viene aggiunto come input nella funzione hash per cercare di ottenere un risultato inferiore al *target.* Il penultimo dato, *numero di transazione*, identifica il numero del blocco stesso all'interno della catena mentre l'ultimo campo, *hash del blocco precedente*, contiene l'hash del blocco cronologicamente antecedente. Questo agisce da vero e proprio puntatore e permette la creazione di una struttura simile ad una catena, nella quale ogni blocco è legato a quello precedente. Non vi è la possibilità di aggiungere blocchi nel mezzo oppure eliminarli, senza invalidare il resto della struttura.
Nel corpo, detto anche *body*, è racchiusa la lista degli identificativi di tutte le transazioni presenti all'interno del singolo blocco.
Le caratteristiche appena descritte accomunano tutti i blocchi presenti nell'ecosistema, ma nonostante ciò è possibile definire una categorizzazione di essi, a seconda del ruolo che ricoprono e della posizione in cui si trovano all'interno della catena. L'unità più significativa per la blockchain è chiamata *blocco genesi*, da cui il nome poiché è il primo elemento della struttura. Attraverso esso, è possibile iniziare a concatenare blocchi per la creazione della catena. A livello di parametri, esso non contiene nessun puntatore al blocco precedente. La funzione principale ed essenziale del blocco genesi è quella di permettere la sincronizzazione tra due nodi della rete: essi potranno comunicare soltanto se possiedono lo stesso blocco genesi. Questo limite apparente gioca un ruolo fondamentale nel garantire la sicurezza all'interno della struttura, assicurando che ogni nodo abbia memorizzato lo stesso registro. La seconda tipologia di unità è chiamata *blocco valido* ed rappresenta tutti i blocchi inclusi nella blockchain correttamente. Queste unità permettono il funzionamento e la registrazione delle transazioni convalidate. L'ultima categoria è detta *blocco orfano* e sostanzialmente identificano tutti quei blocchi validi, ma che non fanno parte della blockchain. Durante la validazione di un blocco, spiegata nel dettaglio nel capitolo seguente, è possibile che due nodi producano più blocchi "concorrenti", ossia collegati ad una stessa unità precedente, andando a creare una biforcazione nella catena. I diversi protocolli, unici per ogni blockchain, specificano quale strategia viene adottata in questi casi. Nel caso di Bitcoin, per esempio, viene ritenuta valida la parte di catena che contiene un numero maggiore di dati, quindi transazioni.

2.4 Transazioni

Le transazioni rappresentano la parte fondamentale della blockchain. Il sistema è progettato per garantire che ognuna possa essere creata, propagata, convalidata da altri nodi ed infine aggiunta al registro globale. Esse sono strutture dati che codificano il trasferimento di valore tra i partecipanti della rete. Ogni singola transazione è una voce pubblica nella blockchain, quindi disponibile e consultabile da chiunque.
Prima di parlare di transazioni è essenziale introdurre il concetto di *portafoglio elettronico*, che andrò ad analizzare nel dettaglio nei capitoli successivi, paragonabile ad un conto corrente presso un istituto di credito. Vige una sostanziale differenza tra i due: mentre il secondo possiede una quantità numerica rappresentativa del patrimonio complessivo, il primo non contiene tale valore, bensì tiene traccia di tutte le transazioni associate. Ciò implica che per conoscere il valore complessivo del capitale, è necessario effettuata una semplice somma algebrica di tutte le transazioni in ingresso ed uscita, relative allo specifico portafoglio. Questo rappresenta lo strumento base che permette qualunque tipo di interazione con la blockchain. Esso è dotato di un indirizzo, detto anche *chiave pubblica*, e di una parola di accesso, chiamata *chiave privata*. Dato che la chiave pubblica è composta da molti caratteri, ne viene generata una più breve univoca chiamata *indirizzo* che identifica il portafoglio, speculare al codice IBAN del conto bancario. L'indirizzo e la chiave privata hanno due scopi differenti, rispettivamente permettere la ricezione di valore ed autorizzare le transazioni.
Un'altra peculiarità riguarda il processo effettivo di una transazione: dal momento che non è presente un valore rappresentativo complessivo del portafoglio, non è possibile inviare valori arbitrari ma solamente quantità ricevute da transazioni precedenti. C'è anche la possibilità di raggruppare i quantitativi di diverse transazioni in ingresso per raggiungere una certa cifra. Il problema principale si riscontra quando il valore che desideriamo inviare è minore di qualunque transazione che abbiamo a disposizione in entrata: saremmo costretti a trasmettere più di quanto vorremmo. Per aggirare l'ostacolo, è comunque necessario spedire tutto il quantitativo di una transazione, ma specificando due indirizzi di destinazione: l'effettivo ricevente e lo stesso mittente. Così facendo, il valore in eccesso è inviato nuovamente all'indirizzo sorgente. Questo meccanismo, abbastanza laborioso, è stato implementato per garantire maggiore sicurezza e facilità a livello di programmazione.
Per agevolare la comprensione del concetto appena esposto, propongo di analizzare un semplice esempio. Il portafoglio A contiene una sola transazione in ingresso, dal valore di 10. Egli necessita di inviare 7 unità al destinatario B. Dato che l'unica transazione a sua disposizione equivale a 10 e non è possibile dividerla, il mittente specificherà due indirizzi in uscita differenti, includendo se stesso per il valore in eccesso [Figura 5].

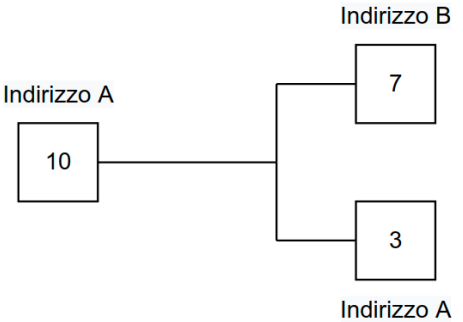

Figura 5 - *Rappresentazione grafica transazioni*

Le transazioni individuano le singole unità all'interno del corpo del blocco. La struttura dati relativa ad ogni singolo accordo tra due parti è composta da diversi campi, i cui principali sono i seguenti:

- hash
- data
- stato
- indirizzo sorgente
- indirizzo destinatario
- quantità in input
- quantità in output
- commissioni
- blocco

Ogni transazione è identificata in modo univoco da un *digest*, calcolato tramite una funzione hash su tutti i restanti campi che la compongono. La *data* specifica il momento in cui essa è stata trasmessa a tutti i nodi: ogni volta che una transazione viene immessa nella rete è necessario propagarla ai partecipanti per ottenere l'autorizzazione all'inserimento della stessa in un blocco. Tutte le transazioni contengono un campo di *stato* che comunica se sono in stato di attesa oppure se lo sono già state. Due elementi fondamentali in una transazione sono il mittente ed il destinatario: questi vengono rappresentati dagli indirizzi dei relativi portafogli degli utenti nella rete. Infine, è presente sia la quantità inviata, in *input*, che quella ricevuta, in *output*, poiché la differenza tra le due costituisce la commissione da pagare per l'aggiunta della transazione nel blocco: una sorta di ricompensa per incentivare i nodi che lo andranno a validare. Una volta che la transazione è stata associata ad un blocco, in attesa di validazione, l'hash dello stesso viene salvato nel campo *blocco* della transazione.

Note le caratteristiche della struttura dati, analizziamo nel dettaglio i passaggi per eseguirle.
La prima operazione da effettuare è andare a costruire una transazione legittima, popolata in tutti i suoi campi, e successivamente autenticarla. Più nel dettaglio, viene sfruttato il metodo crittografico di firma digitale per poter garantire autenticazione, non ripudio ed integrità. Questo processo matematico utilizza un algoritmo di cifratura asimmetrica, che per definizione è composto da due chiavi distinte: una privata ed una pubblica. La prima è strettamente riservata e permette la creazione di una firma autentica, mentre la seconda è di dominio pubblico e viene utilizzata per la verifica. Quindi, è di cruciale importanza mantenere la chiave privata in segreto, onde evitare che terzi possano emettere transazioni non autorizzate. Per ottimizzare le prestazioni della rete, la firma non viene calcolata sulla transazione grezza, bensì sul relativo digest. In questo modo viene ridotta drasticamente la dimensione dei dati in ingresso.
Con il sigillo posto sulla transazione, gli altri nodi sono in grado di verificare l'effettiva proprietà del valore. Inoltre, la firma posta sull'accordo impedisce a terzi di manipolare il contenuto senza invalidare la stessa. In seguito, essa viene aggiunta al primo blocco disponibile, per poi essere propagata sulla rete. La fase di validazione, detta anche *mining*, è atta ad aggiungere nuovi blocchi idonei alla rete. Ognuno di essi deve rispettare le regole rigide, presenti nel protocollo, per poter superare la verifica di altri nodi, quali per esempio la presenza di un hash inferiore al *target* del blocco. Quando un nodo valida un blocco, questo viene aggiunto alla blockchain e propagato a tutta la rete. Il compito degli altri nodi sarà quello di verificare la correttezza della soluzione proposta. Qualora il blocco risultasse

idoneo, ogni singolo utente all'interno del sistema dovrà aggiornare la propria versione locale del registro.
Può accadere che una transazione non confermata venga inclusa in più blocchi. Inoltre, mentre il blocco è in fase di verifica da altri nodi, è accettabile concatenare ulteriori blocchi a quest'ultimo, nonostante non sia ancora stato validato. Di conseguenza, essendo l'attività di *mining* concorrenziale, se più blocchi con le stesse transazioni vengono considerati idonei, per eleggere quale aggiungere alla rete globale, si applica la regola chiamata "*longest chain rule".* Il protocollo dichiara di mantenere come valida la catena più lunga, quindi il blocco che ha a seguito il maggiore numero di altre unità. In questo modo, si cerca di preservare la catena che ha necessitato di uno sforzo maggiore per essere generata.

2.5 Tipologie di blockchain

Nonostante tutte le reti blockchain condividano uno stesso scheletro, è bene sottolineare che sono presenti molteplici differenze sostanziali fra i diversi protocolli. Attraverso queste è possibile classificarli e quindi analizzarli, evidenziando pregi e difetti di ognuno.
Una caratteristica fondamentale della rete è l'apertura. Sin dall'inizio, abbiamo definito questa nuova tecnologia come libera ed accessibile da chiunque, ma nonostante ciò è possibile definire protocolli senza permessi oppure privati. La prima tipologia, detta anche blockchain aperta, identifica tutte quelle reti pubbliche dove non è presente nessun controllo degli accessi. Questa categoria rappresenta l'idea di rete blockchain descritta in precedenza, dove non ogni membro è posto sullo stesso piano rispetto agli altri. Data l'assenza di ulteriori controlli, le prestazioni rimangono molto elevate rispetto all'altra tipologia. D'altra parte, una rete privata prevede sia una fase di autenticazione che un rigido processo di validazione, delegato a pochi membri. Queste vengono spesso implementate in piccole realtà industriali, dove è necessario mantenere l'integrità dei dati ma al contempo governare sulla rete stessa. Proprio per questo motivo, le blockchain private sono rinnegate da molti, poiché violano alcuni dei principi fondamentali.
La tecnologia blockchain esiste da più di quindici anni e con il tempo ha subito diverse evoluzioni funzionali, ampliandone l'efficienza. Sono presenti tre versioni di blockchain, enumerate da *1.0* a *3.0*.
Il primo modello di rete è definito *"blockchain 1.0"* e permette l'introduzione del concetto di criptovaluta. Esso venne progettato da Hall Finley nel 2005, il quale implementò la prima applicazione basata su blockchain, chiamata "Distributed Ledger Technology". Attraverso una semplice interfaccia, era possibile trasferire valute da un indirizzo ad un altro, senza passare attraverso intermediari. Questo evento sancisce l'inizio di una nuova era: il mondo delle criptovalute. La prima, ed anche più famosa, rete 1.0 è la blockchain *Bitcoin.*
Successivamente, la *"blockchain 2.0"* integra i cosiddetti contratti intelligenti, definiti anche *smart contract*. Questi rappresentano un insieme di regole che permettono di creare, verificare e far rispettare un vero e proprio contratto, stipulato tra due parti. Il vantaggio notevole, apportato da questa nuova tipologia di accordo, risiede nel fatto che essi vengono stipulati e scritti sulla blockchain. In altre parole, non sono né innegabili né tantomeno ignorabili: sono composti da una serie di condizioni che, una volta verificate, attivano in maniera automatica l'azione definita. Quindi non c'è più il rischio che una delle due parti non adempia all'obbligo stabilito. Notare che, a livello giuridico, un semplice contratto intelligente

non rappresenta un dovere. Nonostante ciò, è possibile includere una serie di campi opzionali per creare un cosiddetto contratto legale intelligente, perciò eseguibile anche da un tribunale.
L'altro aggiornamento, portato dalla versione 2.0, è un notevole aumento delle prestazioni e scalabilità della rete. Questo modello di blockchain è implementato da *Ethereum*, la seconda rete ad oggi più popolare, seconda solo a *Bitcoin*.
Infine, con la versione 3.0 assistiamo alla creazione di applicazioni decentralizzate direttamente sulla blockchain, dette anche "*Decentralized App"* (DApp). Queste sono normali applicazioni che invece di risiedere in un singolo nodo, vengono suddivise ed eseguite sulla rete decentralizzata. Anche lo spazio di archiviazione gode di questa caratteristica, garantendo un'altissima disponibilità. Quindi, l'interfaccia utente può essere scritta in qualunque linguaggio in grado di interrogare il *backend.* Il vantaggio principale delle DApps riguarda l'alto livello di sicurezza dei dati e delle operazioni, siccome basate su blockchain. Infine, si sostituisce il metodo di validazione delle transazioni, descritto in seguito, presente sia nella versione 1.0 che nella 2.0, rendendolo molto più efficace. Quindi, non viene a crearsi solo un innovativo metodo di pagamento, bensì un intero ecosistema che permette di eseguire tutti gli applicativi che utilizziamo quotidianamente, ma in maniera decentralizzata.
Le applicazioni decentralizzate più conosciute sono *BitTorrent*, per lo scambio di file peer-to-peer, e *Tor*, come browser per la navigazione anonima. Le blockchain 3.0 con maggiore capitalizzazione di mercato, secondo la classifica stilata dal sito CoinMarketCap [18], sono le seguenti: *Binance Chain* [3], *Cardano* [4] e *Solana* [5].

2.6 Algoritmi di consenso

In un sistema informatico distribuito, il problema maggiormente studiato riguarda il mantenimento della coerenza tra i nodi che lo costituiscono: consideriamo come esempio un comune *social network* , in cui le persone pubblicano migliaia di nuovi contenuti al secondo e al contempo questi ultimi sono consultabili, in tempo reale, dal resto della comunità. Risulta evidente che l'architettura non si basa su un unico sistema centralizzato, per problemi di prestazioni e risorse, bensì sfrutta la scalabilità di una rete distribuita: ogni utente può collegarsi a nodo qualsiasi, senza notare differenze a livello di servizio.
Nel caso specifico della tecnologia blockchain, è essenziale che i nodi si coordinino circa lo stato complessivo della rete in ogni istante di tempo. A differenza di un protocollo distribuito, in uno decentralizzato i nodi non sono soggetti a ruoli differenti. Questo può causare l'ingresso nella rete di un nodo malevolo, il quale ha lo scopo di eseguire il protocollo in maniera volutamente errata.
Gli algoritmi di consenso servono proprio ad arginare situazioni non legittime, cercando di lasciare la rete in uno stato consistente. Per convalidare le interazioni e le transazioni sulla rete esistono diversi algoritmi di consenso, tra cui i più famosi sono chiamati *Proof of Work* ("PoW") e *Proof of Stake* ("PoS").

2.6.1 Proof of Work

L'algoritmo *Proof of Work* si basa sulla risoluzione di complessi problemi matematici per la validazione di nuovi blocchi. I *miners*, ossia nodi addetti alla verifica, mettono a disposizione la propria capacità computazionale per cercare di risolvere l'enigma crittografico in cambio di una ricompensa. Il processo è di tipo concorrenziale e l'obiettivo comune è quello di riuscire a calcolare l'hash del blocco, rispettando il grado di difficoltà e il campo *nonce*. La tipologia di operazioni da effettuare deve essere molto onerosa per il richiedente, ma di facile verifica per tutti gli altri. Per queste ragioni, le funzioni hash sono la scelta ottimale. Al giorno d'oggi. nessuna singola macchina sarebbe in grado di produrre un risultato accettabile in così poco tempo, dato il grado di difficoltà elevato. Una strategia molto diffusa ricade nel cosiddetto *mining pool*, ossia un insieme di elaboratori collaborano alla stessa soluzione, sommando le rispettive potenze computazionali. Maggiore è la capacità elaborativa, maggiori saranno le probabilità di risolvere l'enigma prima degli altri. L'unità di misura che descrive il numero di hash calcolati per secondo è definita *hashrate* (h/s). D'altra parte, la ricompensa dovrà essere spartita in modo proporzionale alla potenza fornita al gruppo.
L'algoritmo ricopre anche un importante ruolo nell'ambito della sicurezza della rete: attacchi di negazione di servizi ("*Denial of Service*") e inoltro di blocchi falsificati sono di difficile realizzazione, dato l'elevato dispendio di risorse necessarie per la scrittura sul registro. Un nodo malevolo dovrebbe essere disposto a sprecare una grandissima quantità sia di tempo che di energie per cercare di compiere l'attacco.
Il modello implementato dalla rete *Bitcoin* è chiamato *Hashcash*, il quale utilizza la funzione crittografica *SHA-256* per le computazioni. In particolare, esso è stato progettato per validare un blocco solamente se l'hash ottenuto contiene un determinato numero di zeri iniziali. La quantità di cifre a zero è definita dal grado di difficoltà ed è variabile in relazione alla capacità computazionale dell'intera rete. La potenza viene stimata ad intervalli regolari e permette l'aggiustamento della complessità, in modo da mantenere una generazione dei blocchi lineare nel tempo. Infatti, se in un intervallo di tempo vengono prodotti troppi blocchi, la difficoltà verrà aumentata. Nel caso in questione, la blockchain *Bitcoin* punta a produrre un blocco ogni 10 minuti circa.

2.6.2 Proof of Stake

In contrapposizione, l'algoritmo di consenso *Proof of Stake* è più recente e mira ad oltrepassare i limiti della famiglia *PoW*. Attraverso il nuovo protocollo è possibile validare transazioni e blocchi semplicemente dimostrando di possedere un definito quantitativo di valute, correlata alla relativa blockchain. Questa strategia è stata proposta come soluzione principale all'enorme spreco di energia elettrica necessaria per il funzionamento del modello precedente. Nello specifico, i nodi vengono definiti *validatori* ed invece che competere tra di loro per la risoluzione di un problema comune, si limitano a depositare una parte delle proprie valute come garanzia. Viene richiesto di dimostrare il possesso di una certa quantità del bene, piuttosto che fornire potenza di calcolo. Il protocollo è stato progettato per eleggere in maniera deterministica un candidato dall'intera rete, il quale avrà il compito di verificare la transazione in cambio di una ricompensa. Molti protocolli che implementano questo meccanismo prevedono necessariamente che colui che verifichi la transazione sia un'entità differente da chi l'ha creata, onde evitare problemi di sicurezza. Per incentivare i

validatori, le probabilità di essere scelti aumentano proporzionalmente al crescere della quota posta come garanzia. Il guadagno ottenuto dalla verifica del blocco è la somma di tutte le commissioni che lo compongono. Quindi, il vantaggio principale di questo algoritmo è riscontrabile nell'efficienza e risparmio energetico dell'intera rete.
Non manca dello scetticismo intorno a questo nuovo modello, provocato da un problema definito come *"nothing at stake"*. Essenzialmente, quando ci si trova di fronte a una biforcazione della blockchain, solitamente provocata da un disaccordo tra i nodi, è possibile che un nodo agisca da validatore per entrambi i rami dato che non sono richieste molte risorse. Ciò comporterebbe a potenziali azioni malevole, quali la possibilità di verificare in entrambi i rami una stessa transazione fraudolenta.

Un'evoluzione del modello *PoS* è chiamato *Delegated Proof of Stake* ("*DPoS"*): la differenza sostanziale con il suo predecessore risiede nel fatto che non è necessario coinvolgere tutta la rete per l'elezione del validatore, bensì l'onere ricade su un determinato numero di nodi fidati, eletti in precedenza dall'intera rete. Solitamente il gruppo di rappresentanti è composto dalle entità che possiedono il numero maggiore di valore. Proprio come in una democrazia rappresentativa, i delegati possono variare nel tempo e devono essere esattamente centouno. Per poter diventare un delegato, è necessario avere più consensi di uno dei membri, così da prendere il suo posto. Il sistema di elezione è basato sui voti degli utenti, dove un'unità di valore equivale ad un singolo voto. Data la scarsità del numero di delegati, le ricompense ottenute dalle validazioni sono di gran lunga maggiori rispetto a un semplice sistema *Proof of Stake*. Nonostante ciò, si può contare su un gruppo di membri fidati, eletti democraticamente dalla comunità.

2.7 Trilemma della scalabilità

Un aspetto di fondamentale importanza durante la progettazione di una blockchain riguarda la scalabilità. Con questo termine ci si riferisce alla capacità del sistema di gestire un numero di transazioni e nodi sempre maggiore, senza andare ad intaccare le prestazioni, aumentando anche la sicurezza. Le diverse reti sono confrontate tra loro attraverso l'unità di misura *transazioni al secondo ("TPS")*. Attraverso questa metrica, gli sviluppatori decidono quale sia la soluzione più adatta alle proprie necessità, optando per un *livello-1* oppure *livello-2*.
Il *livello-1* rappresenta l'architettura blockchain alla base, invece il *livello-2* identifica la rete realizzata al di sopra di una blockchain già esistente. Quindi, possiamo affermare che le monete native sono classificate nella prima categoria, mentre i *token* nella seconda.
Nel dettaglio, una soluzione di tipologia *livello-1* mira ad implementare delle migliorie per rendere il sistema più scalabile. I due processi più utilizzati hanno come scopo la modifica del protocollo di consenso della rete e la frammentazione.
Le soluzioni di tipologia *livello-1* sono le più antiche, risalendo anche a una decina di anni fa, e presentano spesso problemi di prestazioni. Prendendo come esempio le due reti più importanti, *Bitcoin* ed *Ethereum*, entrambe validano le transazioni con il sistema *Proof of Work*: se da una parte il livello di sicurezza è innegabile, dall'altra la velocità è nettamente trascurata, offrendo meno di 7 *TPS* nella blockchain *Bitcoin* e 20 *TPS* in quella *Ethereum.*

Questa è la ragione principale per la quale le reti con protocollo *PoW* stanno migrando a soluzioni *PoS*, come appunto *Ethereum.*
Il secondo processo per aumentare le prestazioni di una soluzione di *livello-1* è la frammentazione. Si cerca di ottenere un grado di parallelismo che sia il più alto possibile, invece che tenere l'intera rete occupata a validare transazioni in modo sequenziale. Ognuna di esse è suddivisa in piccole unità, che verranno processate in parallelo da una sola porzione di nodi.
Parlando di soluzioni di *livello-2*, il vantaggio principale risiede nella possibilità di adattare il protocollo sottostante alle proprie esigenze specifiche, senza rinunciare alla sicurezza garantita dall'alto numero di nodi della blockchain scelta. Altro aspetto da tenere in considerazione sono le commissioni: non andando ad utilizzare direttamente la rete nativa, è possibile sviluppare un sistema che gestisca microtransazioni in modo autonomo, rendendole molto efficienti, sia a livello economico che temporale. Non a caso, le reti che godono di commissioni quasi nulle sono quasi tutte di *livello-2*. Resta comunque il vincolo della blockchain alla quale ci si appoggia, quindi la tipologia di token che verrà utilizzata.

Nonostante i vantaggi citati, entrambe le soluzioni devono fronteggiare un problema noto come *"Trilemma della scalabilità"*. Le tre caratteristiche ottimali, alle quali ogni rete ambisce, vengono elencate di seguito: decentralizzazione, sicurezza e scalabilità [Figura 6].
Se ci fosse anche solamente una blockchain in grado di garantire queste tre proprietà contemporaneamente, questa probabilmente soppianterebbe tutte le altre. Infatti, il problema esplica la quasi impossibilità nel garantire tutte le qualità contemporaneamente.
Per stabilire le relazioni presenti fra le tre dimensioni, analizziamo un ipotetica rete basata su tecnologia blockchain. Come dato di fatto, tutti i membri di una rete devono essere d'accordo sulla validazione di una transazione che, in basa al numero di nodi, può richiedere parecchio tempo. Quindi, ponendo un valore fisso alla sicurezza, si deduce che la scalabilità è inversamente proporzionale alla decentralizzazione. Dopodiché, assumiamo che siano presenti due reti *Proof of Work* con il medesimo grado di decentralizzazione. Pensiamo alla sicurezza come la potenza computazionale: più questa è elevata, minore è il tempo di verifica delle transazioni. Di conseguenza, la sicurezza è direttamente proporzionale alla scalabilità. Infatti, con un livello di hashrate alto, è possibile processare molte più transazioni, a parità di difficoltà.
Nell'esempio pratico di *Bitcoin*, l'obiettivo è stato posto sulla sicurezza e la decentralizzazione, andando a penalizzare nettamente la scalabilità. In uno scenario alternativo, dove decentralizzazione e scalabilità vengono prediletti, una perdita di sicurezza è inevitabile: in una rete densa, con un alto numero di nodi, per essere in grado di garantire alte prestazioni, quindi una buona scalabilità, il processo di verifica delle transazioni deve essere particolarmente veloce, riducendo le difficoltà per le validazioni e sacrificando la sicurezza.
Potremmo definire questo problema come un gioco a somma zero, nel quale non è possibile eccellere in tutte le dimensioni senza sacrificarne una. Come mostrato graficamente dalla figura sottostante, è possibile disporre le tre proprietà nello spazio ed unirle, creando un triangolo. Quindi, durante la progettazione di un nuovo sistema è necessario posizionarsi su uno dei lati, andando a trascurare la terza caratteristica.

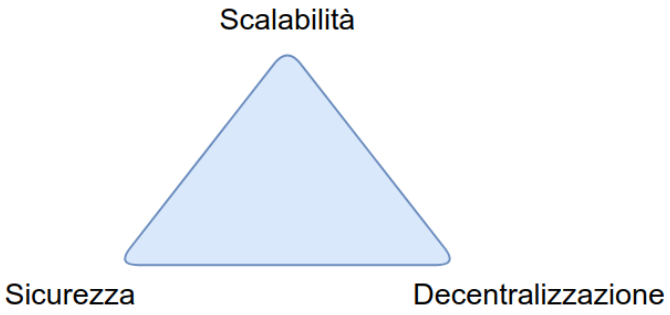

Figura 6 - *Rappresentazione problema “Trilemma della scalabilità”*

Capitolo 3 - Criptovalute

3.1 Definizione e caratteristiche

Quando parliamo di tecnologia blockchain ci riferiamo principalmente a un registro condiviso distribuito, il quale tiene traccia di tutte le transazioni effettuate nella rete. Finora ci siamo limitati ad accennare il concetto di trasferimento di valore da un nodo a un altro, senza entrare nel merito di cosa fosse effettivamente il bene in questione.
Una *criptovaluta* costituisce una rappresentazione digitale di un bene, le cui transazioni sono registrate e verificate da un sistema decentralizzato. La parola stessa deriva dalla fusione dei termini *crittografia* e *valuta*, i quali ne definiscono le caratteristiche principali: una moneta digitale che sfrutta funzioni matematiche per garantire integrità e sicurezza durante le transazioni. Ogni ecosistema possiede la propria moneta nativa, utilizzabile solo all'interno di esso. Al momento della stesura, secondo il sito CoinMarketCap [19], sono presenti più di 8000 monete differenti, costantemente in aumento.
Il termine moneta può essere sufficiente per racchiudere tutte le criptovalute, ma non tiene conto della distinzione tra monete native e derivate. Infatti, ogni blockchain è progettata in simbiosi con la propria valuta, la quale è principalmente utilizzata per eseguire pagamenti ed è necessaria per sostenere le commissioni della rete. Ogni moneta nativa agisce da valuta nel sistema in questione, come l'euro all'interno dell'Unione Europea. D'altra parte, una valuta digitale derivata, solitamente chiamata *token*, può essere qualunque tipo di bene creato sopra una blockchain già esistente, come per esempio token di utilità, governo e sicurezza oppure contratti intelligenti. Inoltre, la categoria dei token include anche tutte quelle risorse scambiabili all'interno del sistema chiamate *Non-Fungible Token ("NFT")*: questi sono atti a provare la proprietà e fungono da certificato di autenticità di un bene, che si tratti di un'opera d'arte, una semplice immagine oppure un brano musicale.
Come accennato in precedenza, non tutte le criptovalute sono monete native: accade frequentemente che venga utilizzata una blockchain già esistente sulla quale creare la propria valuta digitale. Questa situazione si verifica principalmente quando non è presente un sostegno economico sufficientemente elevato per andare a sviluppare la propria catena di blocchi. Oltre all'aspetto finanziario, progettare e sviluppare da zero una nuova blockchain richiede un alto grado di specializzazione nel settore, spesso assente. Inoltre, data la presenza di blockchain affermate nel settore risulta più conveniente appoggiare la propria moneta su di esse, soprattutto per garantire agli utenti futuri una buona sicurezza e l'assenza di falle grossolane nel sistema. La maggior parte dei token in circolazione sono stati creati sopra la rete *Ethereum* e vengono definiti *token ERC20.* Questi non sono altro che contratti intelligenti prestabiliti che facilitano il compito di uno sviluppatore nel creare i componenti del progetto.
Le criptovalute hanno mantenuto la promessa di permettere un trasferimento di beni diretto, senza bisogno di una terza entità fidata come la banca. Le transazioni vengono eseguite istantaneamente, senza i limiti temporali imposti dagli istituti di credito.
Il problema principale ricade sulla natura semi anonima delle criptovalute, poiché le rendono appetibili ed ideali per eseguire operazioni illecite, come il riciclaggio di denaro oppure l'evasione fiscale. Un falso mito, di uso comune, riporta come le valute digitali siano completamente anonime e non tracciabili. Questo non rispecchia in nessun modo la realtà: sono presenti valute con diversi gradi di anonimia, proprio perché sono state concepite e

sviluppate con scopi differenti. *Bitcoin*, per esempio, non vanta uno dei migliori protocolli per mantenere anonime le attività sulla rete. Nel passato, alcuni strumenti per l'analisi forense hanno contribuito a portare alla luce alcune attività illecite. Una rete particolarmente orientata alla privatezza è *Monero*, nella quale non è possibile tracciare le transazioni e collegarle ad individui, poiché offuscate.
L'obiettivo principale delle criptovalute rimane quello di eliminare qualunque barriera spaziale e temporale nel mondo finanziario, permettendo trasferimenti istantanei con commissioni ridotte, senza necessità di un terzo. Con questo sistema, verrebbero a mancare anche i limiti sulle operazioni introdotte dalle banche, come prelievi e bonifici. Infine, a differenza delle comuni basi di dati, le criptovalute si basano su funzioni matematiche crittografiche e decentralizzazione, motivo per il quale è presente una netta differenza nella sicurezza tra le metodologie di conservazione dati.

3.2 Portafoglio elettronico

Le criptovalute, essendo beni di tipologia virtuale, non sono tangibili né tanto meno conservabili fisicamente. Non esiste alcuna moneta o banconota fisica che rappresenti una valuta digitale. La domanda che sorge spontanea, a questo punto, risulta essere: come è possibile accedere e gestire le proprie criptovalute?
Essendo esse completamente digitali, per interagirci è necessario creare un portafoglio elettronico, chiamato anche semplicemente *wallet.* Questo è lo strumento più basilare nel mondo delle criptovalute, paragonabile ad un conto bancario. Paradossalmente, un *wallet* è un programma molto semplice, in grado di eseguire un numero limitato di operazioni. Il cuore del software consiste nella gestione di una coppia di chiavi. Come accennato in precedenza, una è pubblica mentre la seconda privata: la prima funge da indirizzo del portafoglio e permette la ricezione di valute, quella privata invece entra in gioco nel momento in cui si vuole autorizzare una transazione, firmandola con la propria chiave personale. Quando si effettua la scelta di un *wallet* è necessario tenere in considerazione le procedure di messa in sicurezza della chiave privata: chiunque la possieda è in controllo dell'intero patrimonio. In sostanza, il portafoglio contiene solamente le chiavi per l'accesso alle monete, e non le criptovalute stesse. Per stabilire il bilancio complessivo di una monete, all'interno di un singolo *wallet*, è sufficiente una somma algebrica di tutte le transazioni in ingresso ed uscita.
Durante l'atto di creazione di un nuovo portafoglio non è richiesto nessun tipo di dato sensibile: a procedura risulta anonima e permette di avere un *wallet* operativo in pochi minuti. Bisogna solamente prestare particolare attenzione alla cosiddetta *seed phrase*, la quale è un'insieme di parole semplici, solitamente da 12 a 24, che permette il ripristino del portafoglio in caso di smarrimento delle chiavi o danneggiamento del dispositivo dove è contenuto il portafoglio. L'insieme delle parole vengono generate automaticamente dal software e sono mostrate una sola volta, durante la creazione. Analizzando più a fondo il codice sorgente di un qualunque *wallet* è evidente che la *seed phrase* non è effettivamente costituita dalle 12/24 parole, bensì da una lunghissima stringa numerica casuale. I termini mostrati rappresentano una modalità più efficace e semplice per la memorizzazione, che permette di evitare errori di trascrittura dovuti alla complessità della stringa numerica casuale. Quindi, a partire dalle parole è possibile risalire alla chiave numerica iniziale.

Tutti i portafogli elettronici si basano sullo standard "*Bitcoin Improvement Proposal 39",* chiamato anche BIP39, che consiste in un 'insieme di 2048 parole uniche. La peculiarità di questo insieme finito riguarda i caratteri iniziali dei singoli elementi: prendendo qualunque coppia di parole, queste avranno al più i tre caratteri iniziali identici. Quindi, sarebbe possibile esaminare la lista considerando solamente i primi quattro caratteri di ogni parola, senza trovare due elementi uguali. Al contrario della comune logica informatica, l'indicizzazione parte dal numero 1 fino ad arrivare al 2048.
A livello teorico la *seed phrase* è associata ad un'unica stringa numerica *seme*, che agisce da superchiave del portafoglio. Infatti, a partire da essa è possibile generare le chiavi pubbliche e private. Dal momento in cui un *wallet* è in grado di contenere diverse tipologie di criptovalute, appartenenti a reti diverse, risulta necessario generare una coppia di chiavi per ogni moneta presente. Attraverso la *seed phrase* è possibile recuperare tutte le chiavi private, poiché esse sono derivate proprio dalla superchiave.

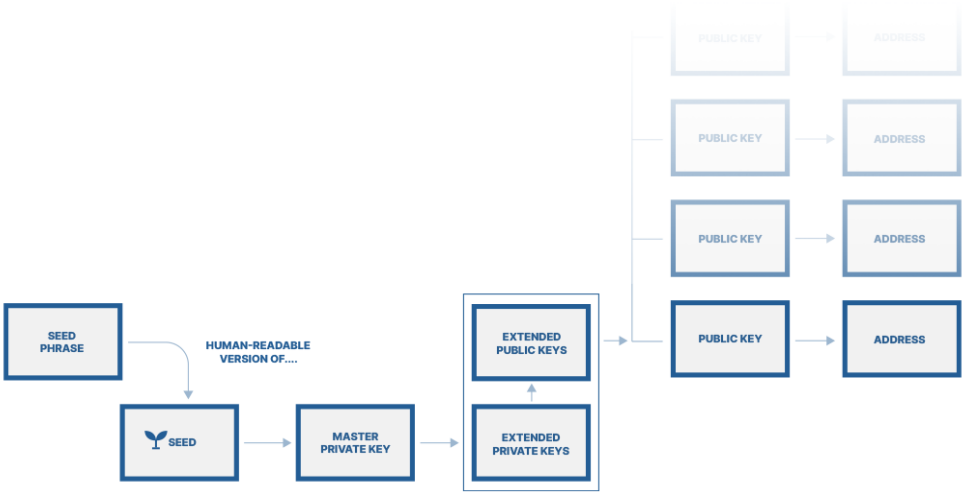

Figura 7 - *Processo logico di generazione delle chiavi*
[sorgente] *https://unchained.com/blog/what-is-a-bitcoin-seed-phrase/*

Il processo di generazione della chiave e quindi mappatura con le parole è molto sofisticato e necessita di un'approfondita analisi. Di seguito andrò ad illustrare questo procedimento, prendendo in considerazione un *wallet* con *seed phrase* a 12 parole.
Di primo impatto, sembrerebbe quasi plausibile riuscire ad indovinare la *seed phrase* di qualcuno attraverso attacchi a forza bruta. Fortunatamente, questo non è realizzabile in un tempo finito utile: non basta essere a conoscenza delle parole, ma è indispensabile anche l'ordine nel quale sono disposte. Avendo un insieme di 2048 elementi, le possibili combinazioni sono ben 2048^{12}, equivalente a 2^{132}. Inoltre, se il dispositivo è in grado di generare un adeguato livello di entropia, la possibilità di indovinare la sequenza sarebbe pari a prevedere un numero nell' intervallo tra 1 e 2^{132}.
Tutte le parole contenute nell'insieme sono indicizzabili da soli 11 bit, dato che 2^{11} equivale proprio a 2048. I bits necessari per la superchiave saranno 11*12 = 132 bits. Come in

qualunque protocollo informatico, il controllo sull'integrità è fondamentale. La sequenza di cifre casuali generate al principio non è composta esattamente da 132 bits, bensì da soli 128. Questo accade per dare spazio proprio alla somma di controllo, che andrà ad occupare lo spazio rimanente, quindi 4 bits. La dimensione del codice di controllo è specificata nel protocollo e definisce 4 bits per 12 parole mentre 8 per le *seed phrase* da 24.
Il primo passo da eseguire consiste nella generazione di una stringa binaria di 128 bits, utilizzando funzioni con un livello di entropia estremamente basso, estremamente complicate. Paradossalmente, la casualità più assoluta si otterrebbe semplicemente lanciando una moneta 128 volte ed annotando tutti i risultati. Una volta ottenuta la stringa, la si suddivide ogni 11 cifre, creano gli 11 gruppi da 11 bits e l'ultimo di soli 7.
Per mappare i gruppi creati con le parole nel dizionario BIP39, è indispensabile convertire i valori binari in numeri decimali. Bisogna però tenere in considerazione la particolare indicizzazione proposta dallo standard BIP39 nella quale il primo elemento ricopre la posizione numero 1: se un gruppo di bit fosse composto da soli zeri, non ci sarebbe mappatura, poiché il valore decimale sarebbe semplicemente zero. La problematica è stata aggirata andando a sommare ad ogni gruppo binario il valore 1, eccetto al gruppo formato da 7 cifre. Solamente dopo questo procedimento tutti i valori garantiscono una corretta mappatura nell'insieme delle parole.
Ogni gruppo binario viene convertito in valore decimale, ad eccezione dell'ultimo, e quindi stabilita la parola corrispondente, utilizzando il numero con funzione di indice all'interno dell'insieme, come fosse l'indice di un vettore. Manca solo il calcolo dell'ultimo indice per ottenere la *seed phrase* completa, in particolare del codice di controllo. L'algoritmo sfrutta la funzione hash *SHA-256* e calcola il valore di *digest* a partire dall'intera stringa di 128 bits. I primi 4 bits del risultato ottenuto rappresentano il controllo d'integrità, quindi inseriti al termine della stringa di 128, così da avere anche l'ultimo gruppo binario di 11 cifre. Manca solo da calcolare l'indice intero dell'ultima parola, quindi si procede a sommare 1 anche all'ultimo gruppo e successivamente a convertirlo in formato decimale, per ricavare anche l'ultima delle 12 parole.
Il risultato finale dell'intero processo è un elenco di 12 parole ordinate, le quali vanno custodite con estrema cautela, tenendo a mente che chiunque abbia accesso alla *seed phrase* avrebbe diretto controllo sul portafoglio. Buona norma sarebbe conservare le parole in modalità cartacea, escludendo il pericolo di tutte le minacce digitali.
La fase di ripristino di un *wallet* già esistente consiste nell'esecuzione del processo in sequenza inversa, partendo dalla *seed phrase* fino alla superchiave binaria.

Esistono due tipologie di portafogli elettronici, distinti per caratteristiche ed utilizzi: *software* e *hardware wallet*.
Un *software wallet* è un programma, disponibile sia per piattaforme *desktop* che *mobile*, che permette un facile accesso alle proprie monete. Questa tipologia gode di un'interfaccia grafica minimale, alla quale è possibile accedere con una password impostata dall'utente. Per non complicare troppo la vita a possibili individui non troppo esperti, questi software solitamente mettono a disposizione un limitato insieme di funzioni basilari, quali la creazione dello stesso, l'invio e la ricezione di criptovalute. Come per ogni portafoglio elettronico, è essenziale annotare la *seed phrase* proposta in fase di creazione per permettere il ripristino del *wallet* in caso di problemi al dispositivo che lo ospita. Il vantaggio principale di un *software wallet* risiede nella sua semplicità d'utilizzo ed accessibilità. Infatti, questi sono indicati per utenti che necessitano un'alta interazione con le proprie valute e sono molto semplici da impostare, spesso già pronti all'uso. Il problema maggiore è riscontrato invece

nel grado di sicurezza offerto: essendo programmi in esecuzione su dispositivi in rete, la probabilità di subire un attacco od un'infezione di un malware è elevata. Questo fattore è il motivo principale per il quale molti utenti diffidano da un portafoglio *software*, optando per la prossima scelta.
La seconda tipologia di portafoglio è definita *hardware wallet*. Questi sono dispositivi hardware che hanno il compito di generare e memorizzare le chiavi private del nostro portafoglio. Di primo impatto, essi potrebbero essere scambiati per un'unità USB date le forme e dimensioni. L'enorme beneficio portato da questi dispositivi riguarda il processo di generazione delle chiavi e la sicurezza intrinseca. Ogni portafoglio hardware è dotato di un componente fisico per la generazione di numeri casuali ed uno per le operazioni di crittografiche. Inoltre, nella fase di inizializzazione del dispositivo è richiesta la creazione di un codice numerico di accesso. Ogni volta che bisogna interagire con il portafoglio è necessario inserire il PIN scelto: in caso di superamento della soglia limite d'errore, le chiavi private contenute saranno automaticamente eliminate. Questa precauzione è atta a scoraggiare furti del dispositivo fisico. Notare che l'utente legittimo sarebbe comunque in grado di recuperare il portafoglio, utilizzando la *seed phrase*.
L'altro grande vantaggio di un *hardware wallet* riguarda l'interazione con Internet: è possibile tenere il dispositivo disconnesso, rendendolo inattaccabile a livello di rete.
Questi apparecchi hardware hanno un costo, che non tutti sono disposti a sostenere inizialmente. Inoltre, il processo di inizializzazione non è alla portata di tutti e se non effettuato correttamente, potrebbe portare all'apertura di falle nel sistema. La tipologia di portafoglio in questione è raccomandata per la conservazione di criptovalute nel lungo periodo, piuttosto che scambi giornalieri.

Un'ulteriore classificazione dei portafogli elettronici può essere basata sul reale detentore delle chiavi private. Possiamo distinguere i cosiddetti *wallet custodial* e *non-custodial.* Spesso non viene dato molto peso alla differenza presente tra i due, poichè superficialmente risulta invisibile. Bensì, sempre ponendo la sicurezza delle valute al primo posto, la distinzione è abissale.
Un *wallet custodial* viene solitamente utilizzato da individui con poca dimestichezza nel settore delle criptovalute, in quanto risulta di gestione facilitata. Per fare uso di un esempio pratico, è necessario introdurre il concetto di *Exchang*e. Questo è lo strumento finanziario principe nel mondo delle criptomonete, poiché il suo scopo principale è consentire lo scambio di monete digitali fra loro ed anche con valute legali. Dopo la procedura di registrazione, ogni utente dispone di un portafoglio elettronico e di un'interfaccia di scambio. Quando ci si riferisce a *wallet custodial*, l'esempio lampante sono proprio i portafogli presenti negli *Exchange*: la società privata agisce intermediario tra l'utente e il reale portafoglio, agendo da vero e proprio custode di tutte le chiavi private dei propri clienti. Mentre per molti un *wallet custodial* è meno sicuro di uno personale, per il mancato controllo diretto delle chiavi, per altrettanti è un'opzione più che valida, comparando benefici e rischi, poiché non richiede un grado di responsabilità diretta elevato. Infatti, anche se la password di accesso all'*Exchange* fosse smarrita, sarebbe possibile recuperarla con una semplice procedura. Questa tipologia di portafoglio si avvicina molto al concetto di conto bancario attuale, nel quale esiste sempre un metodo di ripristino dell'accesso.
Se da una parte si predilige la comodità sulle responsabilità, dall'altra troviamo la categoria dei *wallet non-custodial*, che esprimono a pieno la reale nozione di proprietà di un bene.
La differenza principale consiste nell'avere controllo completo sulle proprie monete, eliminando enti di terze parti. La procedura consiste nel creare un portafoglio, come spiegato

sopra, e memorizzare la *seed phrase*. Il livello di responsabilità aumenta notevolmente, poiché una dimenticanza od un banale errore potrebbero portare ad un disastro finanziario. A livello di sicurezza, un wallet personale rappresenta un'alternativa più consistente ma anche difficile da gestire. Un'analogia nella vita di tutti i giorni potrebbe essere la comparazione tra un istituto di credito oppure una cassaforte personale. Le persone che sfruttano la tecnologia blockchain sono alla ricerca di una decentralizzazione completa, che restituisca potere ai singoli individui. Ecco perché questa scelta risulta ottimale, se prese le giuste precauzioni riguardo i metodi di ripristino.
Il mantra più famoso recita *"Not your keys, not your coins"*: ciò significa che un individuo non possiede realmente le sue monete a meno che esso non sia l'unico a conoscenza delle chiavi. Per contestualizzare la citazione, in caso di attacco ai danni di una piattaforma *Exchange*, le chiavi di tutti gli utenti registrati sarebbero a rischio, di conseguenza anche le relative criptovalute possedute.

In una realtà priva di entità centrale, nella quale non è possibile denunciare un furto né tantomeno recuperare i beni, è necessario scegliere la soluzione più adatta alle proprie necessità e di conseguenza prendere tutte le accortezze atte a scongiurare un possibile attacco.

Capitolo 4 - Esempi di criptovalute

4.1 Bitcoin

Bitcoin rappresenta la prima criptovaluta nella storia. L'obiettivo primario è la creazione di un sistema finanziario decentralizzato, universale ed economico, nel quale ogni utente è il reale ed unico possessore dei propri beni. La rete Bitcoin è stata sviluppata nel 2009 da un programmatore anonimo, noto con lo pseudonimo di Satoshi Nakamoto. Solitamente, quando ci si riferisce alla blockchain si parla di Bitcoin con la "B" maiuscola, mentre per la moneta si predilige il carattere in minuscolo. Il simbolo della moneta bitcoin è "BTC".
Questa infrastruttura getta le fondamenta per tutte le successive monete virtuali e definisce i principi e le caratteristiche della struttura dati su cui si basa: la blockchain.
Il sistema sfrutta la validazione *proof-of-Work*, nel quale tutti i nodi presenti nella rete sono detti *miners.* L'ammontare di monete totali è pari a 21 milioni e questa cifra non è modificabile. Il quantitativo massimo previsto è scritto direttamente nel codice sorgente della rete e per cambiarlo, servirebbe la collaborazione di più della metà dei nodi. Non sono presenti evidenti vantaggi ad aumentare l'offerta di monete, poiché si andrebbe solamente a ridurre il valore di una singola unità. Stiamo parlando di una moneta deflazionistica e destinata ad apprezzarsi nel tempo proprio per la sua rarità, proprio come l'oro. Per questi motivi, in molti gli hanno attribuito l'appellativo di "oro digitale".
Complice il forte sviluppo tecnologico degli ultimi anni, l'acquisizione di Bitcoin ha seguito un andamento esponenziale, così come tutte le metriche principali della rete. Al momento della scrittura, il numero di bitcoin estratti è circa 19 su 21 milioni totali, circa il 90%.
Per mantenere sotto controllo e costante la quantità di monete estratte, Satoshi Nakamoto ha avuto l'idea di sviluppare una procedura chiamato *halving*. Con questo termine si definisce un evento programmato eseguito ogni 210.000 blocchi, atto a dimezzare le ricompense ricevute dai miner per la validazione di transazioni. Stando alle potenze di calcolo attuali, il fenomeno accade ogni 4 anni circa.
La tabella sottostante evidenzia l'importanza dell'*halving* e la quantità di monete immesse ogni 4 anni, dal 2009 in avanti. Un'eccezione è rappresentata dall'intervallo temporale prima dell'*halving* del 2012*,* poiché è di soli 3 anni. Attualmente siamo situati nella riga colorata verde, in attesa del prossimo *halving* previsto per il 2024. Per facilitare la comprensione, ho riportato solamente i dati fino al 2030.

HALVING BITCOIN					
Anno	**Numero blocco**	**BTC/blocco**	**Quantità iniziali**	**Quantità prodotta**	**% circolazione**
2009	0.00	50.00	0.00	10,500,000.00	50.00%
2012	210,000.00	25.00	10,500,000.00	5,250,000.00	75.00%
2016	420,000.00	12.50	15,750,000.00	2,625,000.00	87.50%
2020	630,000.00	6.25	18,375,000.00	1,312,500.00	93.75%
2024	840,000.00	3.13	19,687,500.00	656,250.00	96.88%
2028	1,050,000.00	1.56	20,343,750.00	328,125.00	98.43%

Figura 8 - *Tabella con dati relativi ad ogni halving, fino al 2028*

Se non ci fosse stata l'introduzione dell'*halving*, tutti i bitcoin sarebbero già stati minati tempo fa. Inoltre, supponendo una domanda di mercato costante, dopo l'evento il valore delle unità è destinato a crescere. L'estrazione dell'ultima unità bitcoin è stimata per il 2140, dopodichè i *miners* riceveranno ricompense solamente accumulando le commissioni dal processo di validazione dei blocchi.

La conservazione di bitcoin avviene tramite l'utilizzo di *wallet* anonimi, quindi che non contengono nessun tipo di informazione riguardo il proprietario. La generazione di un portafoglio è molto semplice e veloce, paragonabile alla velocità per la creazione di una coppia di chiavi crittografiche asimmetriche. Come spiegato nel capitolo precedente, il *wallet* è identificato da un indirizzo unico che nel caso della rete Bitcoin inizia sempre con una delle seguenti combinazioni di caratteri, a seconda della versione della rete: "1","3","bc1q","bc1p".
La generazione della chiave pubblica e privata avviene tramite l'algoritmo appartenente alla famiglia dei cifrari ellittici, chiamato *Elliptic Curve Digital Signature Algorithm* (*ECDSA*).
Questo deriva direttamente dal *Digital Signature Algorithm (DSA)* e sfrutta la funzione ellittica del cifrario *Elliptic Curve Cryptography* (ECC)*,* dal quale eredita la lunghezza della firma equivalente a *4t* bit, dove *t* rappresenta la sicurezza misurata in bit. Il motivo principale per il quale si è prediletto *ECDSA* rispetto ad un *RSA*, ben più diffuso, riguarda l'algoritmo ellittico su cui si basa il primo, poichè richiede una chiave molto più piccola per ottenere lo stesso livello di sicurezza del secondo.
Se i *wallet* possono godere della proprietà di anonimato, non si può dire lo stesso delle transazioni. Esse sono definite pseudo anonime, dal momento che sono registrate ed accessibili a tutti pubblicamente. Infatti, è possibile tracciare tutti i movimenti di un determinato portafoglio, senza però conoscerne il proprietario.

Ogni volta che una nuova transazione è immessa nella rete viene ricevuta da un nodo che la conserva, insieme a tutte le altre in stato di attesa. Da questo istante, ognuna di esse deve sperare di essere inclusa il prima possibile in un blocco detto *candidato*. I *miner* sono soliti a prediligere transazioni con commissioni elevate, poiché quello sarà il loro guadagno effettivo. Quindi, più è alto il prezzo da pagare, più veloce questa verrà portata a termine.
Il *blocco candidato* è una tipologia di unità temporanea creata da un singolo nodo, contenente alcune tra le proprie transazioni in attesa. Infine, ogni *miner* cerca di aggiungere il proprio blocco alla catena tramite il processo di validazione. Con lo scopo di mantenere sotto controllo la quantità di blocchi prodotti, in media viene validato un'unità ogni 10 minuti.
Il riferimento interno al blocco utile a controllare la velocità di produzione è chiamato *target*, spiegato dettagliatamente del capitolo precedente.

La domanda a questo punto sorge spontanea: come è possibile analizzare nel dettaglio il contenuto di una blockchain, nello specifico Bitcoin?
Ogni rete possiede un'interfaccia grafica web per la consultazione dei blocchi, delle transazioni e di tutti i loro dettagli, dette *explorer*.
Prendiamo sotto analisi uno degli eventi più importanti e storici della rete: la creazione del blocco genesi. Per conoscerne tutti i dettagli ed analizzare i dati, ci avvaliamo del sito web https://blockchain.com.
Il primo blocco è stato aggiunto esattamente il *3 gennaio 2009,* il quale conteneva una sola emblematica transazione. Questa è unica nel suo genere, siccome non contiene nessun valore in ingresso, quindi non sono presenti né indirizzo sorgente né tantomeno quantità da

inviare. Come uscita, sono stati accreditati 50 BTC ad uno specifico indirizzo, tutt'oggi ancora attivo.
I dati tecnici rilevanti sono i seguenti:

- hash blocco: *000000000019d6689c085ae165831e934ff763ae46a2a6c172b3f1b60a8ce26f*
- indirizzo wallet destinatario: *1A1zP1eP5QGefi2DMPTfTL5SLmv7DivfNa*
- Difficoltà blocco: 1
- Dimensione: 285 byte

Satoshi Nakamoto voleva provare a livello temporale la validità e la correttezza del sistema ed ha introdotto nell'unica transazione del primo blocco un messaggio nascosto: come accennato in precedenza, tutti i campi in ingresso della transazione sono nulli, tranne uno.
Questo contiene la seguente stringa esadecimale:

> *5468652054696d65732030332f4a616e2f32303039204368616e63656c6c6f7220 6f6e206272696e6b206f66207365636f6e64206261696c6f757420666f722062616 e6b73 (HEX)*

che se convertita in ASCII restituisce il messaggio seguente:

> *The Times 03/Jan/2009 Chancellor on brink of second bailout for banks (ASCII)*

Il testo ottenuto non è altro che il titolo della copertina del giornale "London Times" del giorno 3 gennaio 2009. Questa testata ha un significato speciale, siccome rappresenta il fallimento dell'intero sistema finanziario inglese nel 2009 che ha alimentato l'interesse verso blockchain e sistemi decentralizzati.

All'interno della rete Bitcoin gli individui sono classificati in base al ruolo svolto. La percentuale maggiore è ricoperta da *consumatori*, utenti che svolgono operazioni basilari quali inviare e ricevere valute. Essi sostanzialmente usufruiscono di un servizio offerto da altri e per questo motivo sono anche detti *clienti*. La fascia di persone rimanenti ricopre il ruolo di *produttore*, colui che mette a disposizione risorse ed energie in cambio di una ricompensa.
Ogni produttore, in gergo, è detto nodo (o *miner*) siccome ricopre il ruolo di validatore della rete, assicurando la riuscita delle transazioni degli utenti. Entrando più nel pratico, per attivare un nuovo nodo è necessario eseguire il programma di collegamento alla blockchain, chiamato *Bitcoin Core.* Questo si occupa principalmente di sincronizzare la macchina con la rete, scaricando una copia della blockchain in locale. Allo stato attuale, la dimensione totale su disco si aggira intorno ai 324 GB. In modo inversamente proporzionale alla distribuzione di unità bitcoin, questa è destinata a crescere esponenzialmente.
La catena di blocchi è salvata fisicamente in diversi file binari, il cui nome radice è *blk.dat*. Ogni singolo file può contenere al più dati per una dimensione complessiva di 128 MB. Successivamente, si crea il file seguente con un indice incrementale, come per esempio *blk1.dat, blk2.dat, …*

I file sono ordinati e contengono i blocchi così come appaiono nella blockchain ed ognuno è identificato da un insieme di cinque parametri:

- byte magico
- dimensione
- testa del blocco
- numero transazioni
- dati

I primi due campi permettono di capire dove iniziano e finiscono i dati per ciascun blocco, la testa specifica i campi di riferimento obbligatori ed infine è presente una variabile per il numero delle transazioni, contenute nella parte dati seguente.
Il *blocco genesi*, analizzato precedentemente, è leggibile andando a recuperare i primi 285 byte del primo file. Essendo salvati in binario, con l'ausilio di un semplice programma è possibile estrarre il contenuto in codifica ASCII e confrontarlo con quello ottenuto in precedenza dall'*explorer.*
La differenza sostanziale consiste nell'aiuto che offriamo alla rete, in termini di sicurezza e scalabilità. Inoltre, se dovessimo analizzare la rete a basso livello in modo automatizzato, la presenza dell'intera blockchain memorizzata in file binari ci assicura la correttezza della stessa ed una maggiore rapidità di lettura, evitando problemi di fiducia provenienti da siti web di terze parti.

Bitcoin, come la maggior parte delle criptovalute, è scambiabile con monete a corso legale. Inizialmente il valore economico di una singola unità era pari a zero, finché nel luglio del 2010 il prezzo si alzò fino a 0,09 euro. Da allora, la moneta ha avuto un'altissima volatilità trovando il picco a 59.000 euro. La crescita esponenziale del valore di bitcoin è direttamente proporzionale all'adozione della criptovaluta stessa, ed è destinata a crescere nel tempo. Si stima che solamente il 3.5% della popolazione mondiale detiene bitcoin, e nonostante ciò la capitalizzazione dell'intera rete è attualmente di 600 miliardi, equivalente a quella dell'azienda *Berkshire Hathaway,* la quale è inclusa nelle dieci aziende più grandi al mondo.
Un dato non indifferente è la quantità di unità possedute da un singolo, non equamente distribuite: secondo una ricerca pubblicata dal giornale americano *"Fortune"*, una minuscola parte dei detentori controlla gran parte della rete. Nello specifico, si stima che lo 0.01% degli utenti è in possesso di più del 27% dell'offerta totale. Lo studio dichiara che circa 10.000 *wallet* contengono più di 5 milioni di unità complessivamente. Inoltre, se vengono aggiunte al conteggio anche le unità smarrite, ossia l'insieme di monete contenute in *wallet* di cui si sono perse le chiavi, la rete Bitcoin rischia di non essere più così tanto decentralizzata.

Questo è uno dei motivi principali per cui sono state create nuove reti ed introdotte nuove criptomonete, tutte quante con l'obiettivo di superare le limitazioni presenti in Bitcoin.

4.2 Ethereum

Nello spazio delle criptovalute la moneta regina è bitcoin, la quale ricopre oltre il 40% della capitalizzazione totale di mercato. Il restante 60% è occupato dalle cosiddette *monete alternative*, nomenclatura derivata dal termine inglese "*altcoins"*.

L’insieme delle monete alternative è convenzionalmente suddiviso tra *Ethereum*, il quale occupa quasi il 20% totale, e tutte le altre valute minori, le quali coprono il 40% rimanente.
Non a caso, la rete Ethereum è stata la seconda blockchain ad essere sviluppata, nonché la prima vera rivale di Bitcoin. Di seguito andrò a proporre un’analisi dei principali punti di forza e debolezze riguardo *altcoins* più significative.

Andando per ordine di volume, Ethereum è una piattaforma decentralizzata basata sulla tecnologia blockchain di tipo *Proof-of-Work*, nata nel 2015 ad opera dello sviluppatore russo *Vitalik Buterin*, che si prefigge due principali obiettivi: agire da valuta di scambio e supportare i contratti intelligenti. Infatti, la rete possiede una criptovaluta nativa chiamata *Ether* (ETH), utilizzabile come valuta di scambio. La differenza sostanziale con Bitcoin riguarda il numero di unità estraibili: l’offerta massima è illimitata, con la singola restrizione di al più 18 milioni di ETH annui. La motivazione principale della scelta di progettazione risiede nel secondo obiettivo della piattaforma: i contratti intelligenti. Riprendendo in breve, i cosiddetti *smart contract* funzionano in modo automatico, senza la necessità di introdurre terze persone umane. I vantaggi principali sono identificati a livello di sicurezza, poichè sono cifrati e memorizzati sulla blockchain, ed economico, sia a livello di persone coinvolte che di risorse utilizzate.
Questi contratti sono programmi scritti inizialmente in linguaggi di programmazione specifici quali *Serpent*, derivato da *Python* ma con funzioni di basso livello; *Mutan*, ispirato dal linguaggio *Go*, ed infine *LLL*, ricavato da *Lips.* Tuttavia, nel corso degli anni questi vengono rimpiazzati da un linguaggio più completo e sicuro, in sviluppo attivo dal 2014, chiamato *Solidity*. Esso estrapola i punti di forza da tre dei pilastri della programmazione odierna: *C++, Python* e *JavaScript*. Il risultato è un linguaggio di alto livello a tipizzazione statica, appositamente studiato ed ottimizzato per l’esecuzione all’interno della blockchain in questione. Per di poter essere eseguiti, i contratti vengono compilati per produrre il bytecode, interpretabile successivamente dalla rete Ethereum
Ethereum Virtual Machine (“EVM”), questo è il nome dell’ambiente virtuale che permette agli utenti di creare una serie di funzionalità aggiuntive (tra cui i contratti intelligenti) sulla blockchain, eseguibili ad istanti di tempo ben definiti. Si parla di una vera e propria macchina virtuale, in grado di emulare un computer fisico, attraverso processi di virtualizzazione delle risorse, messe a disposizione dai nodi della rete. Questo permette di controllare al meglio situazioni critiche, quali attacchi malevoli o interruzioni di sistema, senza andare ad intaccare le risorse fisiche.
Ogni volta che un contratto viene creato od eseguito, o comunque avviene un'interazione con la blockchain, viene richiesta una certa capacità computazionale da parte della rete, e per tale motivo è necessario adempiere al pagamento di una commissione. All’interno di questa specifica blockchain, questa imposta è definita con l’inglesismo *gas fee*, proprio ad indicare il ruolo di alimentazione che ricopre, come fosse il carburante per un’automobile. I nodi della rete sono più incentivati a prestare la propria potenza di calcolo ad individui disposti a pagare un prezzo maggiore. Analogicamente alla validazione delle transazioni, un *gas* maggiore implica un servizio prioritario, quindi più rapido.
L’ammontare delle commissioni non è casuale, ma è calcolato sulla base di due parametri: la complessità dell’operazione e la congestione della rete. Per esempio, l’invio di ETH in un momento del giorno in cui la rete è poco utilizzata costa sicuramente meno che l’utilizzo di un complesso contratto intelligente in un orario di punta. Per analizzare in termini tecnici il calcolo delle commissioni sulla rete Ethereum è necessario introdurre i concetti di *limite* e *prezzo del gas*. Il primo è un numero che rappresenta il limite massimo di risorse

computazionali che possono essere utilizzate per il conseguimento dell'operazione; il *prezzo del gas* invece indica il costo di un'unità di alimentazione. La denotazione per l'importo del *gas* è detta *Gwei*, la cui singola unità vale esattamente 10^{-9} *ETH.* Solitamente il prezzo varia tra 0 fino ad oltre 500 *Gwei,* a seconda del livello di congestione della rete. Tenendo a mente questi concetti, la commissione da pagare è calcolabile con una semplice equazione riportata di seguito:

$$Commissioni = limite * prezzo\ del\ gas$$

L'analogia con una casistica reale è immediata, basti immaginare il *limite* come la capacità del serbatoio della vettura ed *il prezzo del gas* come il costo della singola unità di carburante.
La struttura dati blockchain, essendo progettata dall'uomo, non può essere priva di errori. Può accadere che una transazione fallisca, ma ciò nonostante le commissioni non verrebbero rimborsate. Infatti, i validatori durante la fase di *mining* del blocco possono riscontrare due problemi principali: la mancanza di *gas*, caso in cui il *limite* stimato sia inferiore alle risorse richieste e la non correttezza di un'istruzione presente nel contratto, che può portare al fallimento dello stesso.

Quindi, data la vastità dell'ecosistema ed il grande utilizzo della blockchain Ethereum, la scelta progettuale di non imporre un tetto massimo al numero di unità *ETH* assicura di non escludere nessuno dallo sviluppo all'interno della rete.

Ethereum rappresenta un ecosistema in grado di promuovere l'interoperabilità tra diverse monete e miglioramenti della rete. Oltre agli *smart contract*, attraverso il linguaggio *Solidity* è perfino possibile la realizzazione di un proprio *token* di livello-2, senza preoccuparsi di progettare una rete dedicata. Una struttura predefinita è stata messa a punto dagli sviluppatori di Ethereum ed è indicata con la sigla *ERC-20.* Viene definito un vero e proprio standard riguardo le funzionalità e proprietà che un *token* può avere. Attraverso l'*explorer* della rete Ethereum, consultabile al seguente indirizzo https://etherscan.io, è possibile notare l'enorme numero di *token ERC-20* in circolazione, attualmente quasi 500.000. Ogni volta che si interagisce con queste monete di secondo livello, è comunque necessario pagare delle commissioni nella valuta nativa della blockchain, supportando indirettamente il token ETH.
Analizzando più tecnicamente la struttura di un contratto *ERC-20*, si può notare come esso funga da interfaccia per facilitare l'interoperabilità intera alla rete. Qualunque servizio voglia dialogare con un *token ERC-20* necessita di conoscere solamente un'insieme di funzioni standard, senza preoccuparsi di possibili divergenze o convenzioni tra sviluppatori di progetti differenti.

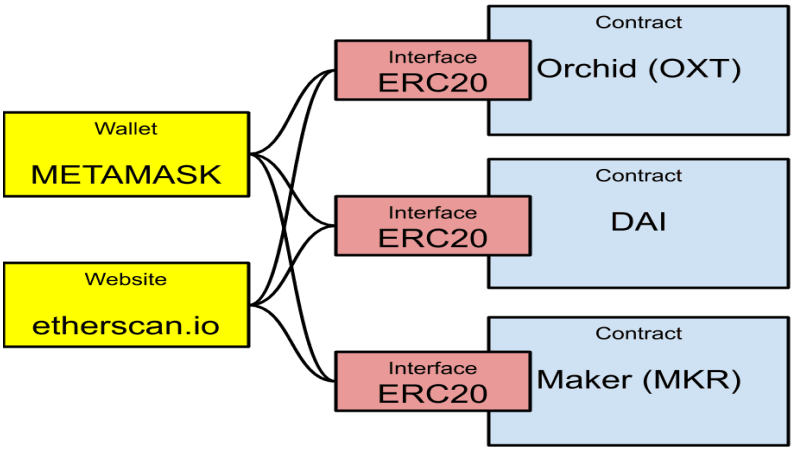

Figura 9 - *Schema interfaccia token ERC-20*

L'interfaccia, a livello di codice, risulta molto più snella di quello che ci si possa aspettare. Le operazioni sono limitate alle seguenti funzioni:

- totalSupply()
- balanceOf(address)
- transfer(recipient, amount)
- approve(spender, amount)
- allowance(owner, spender)

Le prime due operazione permettono rispettivamente di conoscere l'offerta totale di *token* in circolazione ed il quantitativo contenuto in un determinato portafoglio. La terza consiste nell'azione più comune all'interno della rete, ossia il trasferimento di monete da un indirizzo ad un altro. Invece, le ultime due funzioni potrebbero essere di comprensione meno immediata. Il metodo *approve(spender, amount)* abilita un determinato indirizzo a spendere una definita quantità di *token* posseduti dal chiamante, mentre *allowance(owner, spender)* fornisce il numero di monete che possono essere ancora spese dal portafoglio indicato.
Le ultime due funzioni lavorano in simbiosi, poiché è possibile richiamare *allowance* solo se prima si è evocata *approve*. Questi metodi sono particolarmente utili nell'ottica di *smart contract* che agiscono da venditori per il seguente motivo: i contratti non sono in grado di ascoltare e gestire gli eventi. Immaginiamo la situazione nella quale si volesse pagare per un certo servizio: l'acquirente non potrebbe trasferire direttamente le monete al destinatario, poichè esso non verrebbe notificato dell'evento. Per aggirare l'ostacolo, tramite il metodo *approve* si fornisce il permesso al ricevente di spendere una determinata quantità di *token* posseduti dal chiamante. L'operazione di trasferimento può essere frammentata ed è possibile conoscere il quantitativo di monete spendibili rimanente, tramite il metodo *allowance.*
Sorge un problema non indifferente riguardo i permessi attribuiti a terzi, il quale se non gestito adeguatamente porterebbe a conseguenze disastrose. Il presupposto iniziale è il seguente: ognuno è in pieno controllo dell'ordine cronologico delle proprie transazioni, ma non di quelle altrui. Chiamando due volte il metodo *allowance* è possibile modificare la quantità originale di monete che un altro indirizzo può spendere. Può accadere che due transazioni concorrenti, rispettivamente la spesa e la modifica del quantitativo, vengano

validate nello stesso istante, abilitando a tutti gli effetti il portafoglio esterno a spendere anche la nuova cifra impostata. Una volta impostato, la modifica del suddetto valore è altamente sconsigliata per la problematica appena affrontata.

Essendo una rete storica siccome progettata più di sette anni fa, Ethereum soffre di alcune problematiche che potrebbero risultare sempre più evidenti con l'aumento degli utenti.
Il primo grande ostacolo riguarda la scalabilità: attualmente il meccanismo di consenso è *PoW*, il quale è altamente inefficiente per caratteristiche di progettazione, a favore di una maggiore sicurezza e decentralizzazione (ricordando il *trilemma della scalabilità*). Anche il numero di transazioni al secondo risente di queste scelte progettuali, attestandosi mediamente intorno alle 15-20 TPS. Con lo sviluppo e l'utilizzo di contratti sempre più complessi, essere costretti ad attendere minuti (se non addirittura ore) non incentiva lo sviluppo all'interno della suddetta piattaforma.
Il secondo problema riguarda i costi di utilizzo, infatti per interagire con la rete sono necessarie decine di euro. In uno scenario di alcune semplici transazioni giornaliere, le commissioni dovute non sarebbero sostenibili a lungo termine. Al momento della stesura, in accordo con il portale *Messari.io* [5], la rete ha generato un volume di 6.7 miliardi di euro nelle ultime 24 ore, dato da correlare ad un totale di 30 milioni in commissioni. Il rapporto percentuale tra imposte dovute e volume è pari allo 0.45%. Numero privo di significato, finchè non lo si compara ad una qualunque blockchain 3.0 con validazione *PoS*.
Confrontando Ethereum con la rete *Cardano*, la quale vanta 8.3 miliardi di euro di volume giornaliero con soli 55 mila di commissioni ed un rapporto tra i due pari a 0.0006627%, si evince che sia l'efficienza che i costi non favoriscono la storica blockchain.
Il terzo ed ultimo grande ostacolo riguarda le modalità di aggiornamento della rete. Il processo viene definito in inglese *hard fork* e si verifica quando è necessaria un'importante modifica del protocollo, sia per motivi progettuali che di sicurezza. Tramite questo evento, tutte le transazioni valide vengono marcate come invalide e viceversa. A tutti gli effetti, ciò consiste in un'azione irreversibile sulla rete. Successivamente, tutti i nodi sono costretti ad installare una nuova versione del software. L'aggiornamento causa disservizi per un certo lasso temporale, durante il quale non è possibile utilizzare la rete. A seguito di un *hard fork*, il nuovo protocollo non riconosce più quello precedente. In una situazione dove non tutti i nodi decidono di aggiornarsi è possibile che si verifichi sia la creazione della nuova rete, che la persistenza di quella precedente. In questo caso, entrambe proseguono il proprio ciclo di vita, come se fossero due perfette reti indipendenti.

Attualmente, gli sviluppatori della rete Ethereum hanno in programma la modifica del meccanismo di consenso passando da *PoW* a *PoS,* oltre al rimpiazzamento dell'ambiente virtuale EVM con una macchina chiamata *eWASM* [6]*,* in grado di gestire lo standard *WebAssembly* [7] e garantire maggiore velocità di esecuzione*.*
Il nuovo protocollo prenderà il nome di *Ethereum 2.0*, portando l'intera infrastruttura da una blockchain di tipo *2.0* a *3.0*. Il vantaggio principale aggiunto sarà l'integrazione di applicazioni decentralizzate all'interno della rete.

Ethereum ha giocato un ruolo fondamentale nello sviluppo dell'architettura blockchain poiché essa, per prima, ha dimostrato i benefici della decentralizzazione, aggiungendo la possibilità di creare del contenuto utile da parte degli utenti. Non ci troviamo più di fronte ad un semplice sistema di scambio non governato, bensì ad un innovativo ecosistema multifunzione decentralizzato.

4.3 Cardano

Cardano è una blockchain *3.0* pubblica nata nel 2017 dalla collaborazione tra la società americana *Input Output* (IOHK) [8], guidata da *Charles Hoskinson*, e due entità legali, quali *Cardano Foundation* ed *Emurgo*. *Hoskinson* nel 2013 partecipò attivamente allo sviluppo della piattaforma Ethereum, per poi abbandonarlo a fine 2014 a favore dello sviluppo di una propria rete, in grado di colmare i problemi progettuali riscontrati nella struttura dati pensata da *Buterin*. Gli obiettivi principali riguardano la scalabilità, l'interoperabilità e la sostenibilità.
Il linguaggio scelto per la realizzazione del progetto è chiamato *Haskell,* il quale è basato interamente sul paradigma di programmazione funzionale e la politica di tipizzazione statica. Oltre che vantare un'eccellente velocità di esecuzione, le risorse, in particolare la memoria, vengono sfruttate a pieno grazie all'utilizzo di strutture dati molto semplici, predilette rispetto ad aggregati. Il suddetto linguaggio si predispone bene alla modularizzazione, permettendo di utilizzare nuovamente frammenti di codice scritti in precedenza. Rispetto a linguaggi orientati ad oggetti, l'enorme vantaggio risiede nella possibilità di verificare matematicamente ogni funzione, riducendo drasticamente la probabilità di introdurre errori nel codice. D'altra parte, le operazioni di stesura e verifica sono molto onerose e richiedono esperti specializzati. Negli ultimi due anni è stato introdotto anche il linguaggio *Plutus*, derivato dal *Haskell*, sviluppato appositamente per questa rete.
Analizzando la struttura interna della blockchain Cardano, è possibile individuare due componenti principali:

- *Cardano Settlement Layer* (CSL)
- *Cardano Computation Layer* (CCL)

Il CSL funge da libro mastro, quindi si occupa di registrare tutte le transazioni avvenute all'interno della rete. Questo strato gestisce anche tutte le informazioni relative ai *wallet* ed anche i rispettivi bilanci. L'idea è quella di creare una rete avanzata, partendo dalle reti già esistenti, quindi sfruttando fin da subito l'algoritmo di consenso *PoS* per la validazione di transazioni e la generazione di nuovi blocchi. In particolare, l'algoritmo implementato è chiamato *Ouroboros* [9] e rappresenta il primo meccanismo *PoS* matematicamente verificato e sicuro. Infatti, è dimostrabile che se almeno il 51% di monete depositate come garanzia sono detenute da nodi non malevoli, la rete è sicura e non può subire modifiche, a livello di protocollo, non autorizzate. L'altro componente è chiamato CLL ed ha il compito di svolgere tutte le attività computazionali, dalla creazione di un contratto all'esecuzione di una *DApp*.
In caso di fallimento, la processazione delle transazioni continuerebbe comunque, senza subire interferenze.
La decisione di separare i due elementi è stata presa semplificare un possibile cambiamento progettuale futuro. L'esempio lampante è individuabile nel confronto con la rete Ethereum, dove un aggiornamento potrebbe portare alla divisione della rete (*fork*), mentre nella blockchain Cardano viene ricercato un alto livello di flessibilità, onde evitare la rottura del protocollo.

Come ogni rete nativa di primo livello, Cardano possiede una propria moneta che, oltre ad avere valore di scambio, attribuisce potere decisionale. La valuta onora il nome della persona che viene definita la prima programmatrice della storia: *Ada Lovelace*. Sotto il simbolo *"ADA"*, il gettone permette di contribuire alla sicurezza e la governazione della rete attraverso la delegazione degli stessi. Il processo viene definito, dall'inglese, *staking* e sta

alla base dell'omonimo algoritmo di consenso *PoS*. Il fondatore fissa l'offerta massima di monete a 45 miliardi, di cui il 75% è già in circolazione. A differenza della pianificazione programmata proposta da Bitcoin, nuove unità ADA vengono assegnate come ricompensa ai validatori della rete fino al raggiungimento della soglia massima. La moneta ha avuto un fortissimo innalzamento del prezzo, sintomo di grande richiesta da parte degli utenti. Nel 2017 il prezzo iniziale si attestava intorno ai circa 12 centesimi, per poi registrare un valore unitario massimo di 2.47 euro ad agosto 2021. Stiamo parlando di un aumento di più del 2000% in meno di 5 anni.
La possibilità di sviluppare applicativi e contratti interni alla rete porta con sé alcuni costi di gestione, irrisori se comparati alle altre reti in circolazione. Le commissioni dovute possono venire saldate solamente con la moneta nativa, nel caso sotto esame ADA.
La tipologia di applicazioni decentralizzate su cui focalizzare l'attenzione sono sicuramente i cosiddetti *Exchange decentralizzati* (DEX). Questi rappresentano piattaforme di scambio di criptovalute *peer-to-peer*, quindi senza intermediari. Un altro termine spesso associato ai DEX è *finanza decentralizzata* (DeFi), il quale sta ad indicare la possibilità di utilizzare strumenti finanziari in maniera del tutto indipendente.Il primo DEX presente sulla rete Cardano si chiama *SundaeSwap* [10] ed è stato lanciato il 20 gennaio 2022. Se con gli exchange tradizionali è necessaria una registrazione con dati personali e le chiavi private non sono in possesso dell'utente, attraverso i DEX è sufficiente collegare il *wallet* all'applicativo ed eseguire la conversione delle monete con la controparte desiderata. Data l'assenza di terze parti, la responsabilità ricade interamente sull'utente finale. Inoltre, l'interazione con valute a corso legale non è più necessaria, poiché è possibile scambiare tra loro solamente criptovalute.
Il problema maggiore dei servizi DEX riguarda lo slittamento dei prezzi, ossia la differenza di prezzo tra l'immissione della transazione e l'effettiva esecuzione. Siccome non è presente un ente centrale, le transazioni non vengono processate immediatamente. La conseguenza riguarda il lasso di tempo trascorso tra la conferma dell'utente e la validazione della transazione all'interno della rete. Nello scenario in cui questo intervallo sia di qualche minuto, il valore di una moneta potrebbe essere cambiato drasticamente.
Indicando con *X* la quantità del token A, con Y la quantità del token B, il rapporto K = X * Y deve rimanere costante. Se prendiamo come esempio la coppia ETH-ADA, e si volesse convertire un'unità ethereum, la quantità di monete ricevute si aggirerebbe intorno a 2570 ADA (considerando i valori correnti). Ora ipotizziamo la conferma da parte dell'utente della transazione appena citata, con l'aggiunta di una forte congestione all'interno della rete. Per avere la validazione dello scambio potrebbero volerci diversi minuti e qualora il valore variasse, anche la quantità delle unità verrebbe ricalcolata. Il numero di ADA ricevuti potrebbe quindi essere sia maggiore che minore.
Resta sempre possibile impostare una percentuale di tolleranza riguardo una variazione di prezzo sfavorevole, sopra la quale la transazione non viene processata. Notare bene che se questa viene impostata ad un valore tendente a zero, si potrebbe attendere un tempo indefinito prima di riuscire a validare la transazione. Viceversa, con una percentuale elevata vi è la concreta possibilità di perdere potenziale valore, a favore della velocità di processazione.

Il fondatore della rete, *Charles Hoskinson,* si distacca dallo stereotipo classico di programmatore, senza emozioni né tatto. In un video pubblicato su internet [11], egli ha recitato *"... se crei un prodotto di valore, allora il valore tornerà indietro anche alla persona che l'ha distribuito inizialmente."* A differenza di molti progetti a solo scopo speculativo,

Hoskinson ambisce a creare un ecosistema finanziario pubblico, che possa aiutare le popolazioni del mondo più in difficoltà. Se un individuo non possiede documenti relativi alla propria identità e possedimenti personali, diventa complicato assicurarsi una vita prospera. Anche nel caso di autorità giudiziarie, provare la detenzione di un certo bene, quale un'immobile, diventa un'impresa ardua.
La missione di Cardano è quella mettere a disposizione un insieme di strumenti decentralizzati che possano agevolare la vita ad individui in gravi condizioni finanziarie. Nello scenario in cui tutta la documentazione di una persona sia memorizzata all'interno della blockchain, verificare un'asserzione diverrebbe un operazione semplice ed immediata.
L'attenzione è rivolta verso il continente africano [12], il quale non possiede una moneta legale condivisa e pecca di un'infrastruttura digitale all'avanguardia. Dato anche l'importante numero di giovani, assicurare un'educazione uniforme e riconosciuta nel mondo diventa un obiettivo da porre in primo piano.
Il progetto di maggior rilievo, presente sotto forma di applicazione decentralizzata, si chiama *Atala Prism* [13]. Questo rappresenta un portale per la creazione di identità digitali e la gestione della condivisione degli stessi. Come accade per qualsiasi tipologia di transazione, i dati vengono cifrati e non sono accessibili da nessuno, al di fuori del possessore della chiave privata. Attraverso l'utilizzo della piattaforma, è possibile verificare velocemente i dati anagrafici di un individuo, il suo grado di istruzione, possedimenti, contratti ed eventuali assicurazioni. Ciò garantisce ai governi di svolgere attività burocratiche in maniera ben più rapida ed affidabile.
L'accordo più significativo è stato siglato con lo stato etiope, sotto l'aspetto educativo dei giovani. Il sistema permetterebbe di organizzare i dati accademici di circa 5 milioni di studenti. Il passo successivo riguarda l'integrazione dell'identità digitale in tutto il paese, il quale conta più di 115 milioni di individui.

Cardano si prepara a guidare una delle rivoluzioni tecnologiche di maggiore spessore che si siano mai verificate nel continente africano.

4.4 Monete stabili

Ad oggi, il mercato delle criptovalute è altamente volatile. In termini di potere d'acquisto, non è immaginabile adottare una moneta digitale che potrebbe variare drasticamente di valore in un lasso di tempo ristretto. Per chiarire quanto appena esposto, supponiamo che si possa utilizzare la moneta ADA per acquistare beni e servizi e fissiamo un valore unitario intorno ad 1 euro. Indicativamente, con una singola moneta è possibile acquistare un'ipotetica bottiglia di latte. Nello scenario fortuito in cui, nel giro di pochi giorni, il valore della moneta cresca, allora sarebbe possibile acquistare più bottiglie con la stessa unità. Da prendere in considerazione è anche la casistica opposta, nella quale il valore della moneta potrebbe precipitare e di conseguenza ridurre il potere d'acquisto dell'utente finale.
Il 99% delle criptovalute in circolazione non si appoggiano a nessuna risorsa materiale, e di conseguenza il prezzo di mercato è determinato unicamente dal punto di incontro tra domanda ed offerta.

Le monete stabili, dette *stablecoins*, sono valute digitali legate ad attività di riserva stabili come l'oro od il dollaro statunitense. Esse sono state progettate per ridurre la volatilità della moneta rispetto alle altre criptovalute e tendono a replicare l'andamento del bene sottostante. Questa categoria di montere ricopre un ruolo fondamentale nello spazio digitale, poiché permette di collegare il sistema finanziario tradizionale, e le relative valute a corso legale, con i vantaggi e le innovazioni delle criptovalute. Anche nell'ottica di introdurre le monete digitali come metodo di pagamento quotidiano, le *stablecoins* rappresentano il compromesso migliore.
Le monete come l'*USD Coin* (USDC) oppure *Theter* (USDT) sono radicate al dollaro americano e la seguente equivalenza è sempre rispettata: $1\,USD = 1\,USDC = 1\,USDT$
L'idea di base è garantire che per ogni dollaro digitale ne sia presente in riserva uno fisico.
Quindi, viene a fondersi il potere d'acquisto del dollaro classico con la tecnologia blockchain, ed il risultato è una moneta in grado di preservare il proprio valore nel tempo, senza rinunciare ai benefici digitali. Infatti, è possibile trasferire valore 24 ore al giorno, 7 giorni alla settimana, senza vincoli temporali né spaziali. Inoltre, l'invio di denaro è pressoché istantaneo e verificabile pubblicamente. Quasi tutte le *stablecoins* sono token creati sulla rete Ethereum, di conseguenza necessitano di un portafoglio adatto per la conservazione e gestione.
I tre possibili beni a cui le monete stabili possono legarsi sono i seguenti:

- valute a corso legale
- materie prime
- criptovalute

Nel primo caso, la moneta viene associata ad una valuta di stato quale in dollaro o la sterlina. Solitamente si tende a prediligere questa tipologia di ancoraggio poiché permette di controllare più facilmente il potere di acquisto. La seconda categoria regola il proprio valore a seconda del prezzo di mercato di materie prime, solitamente metalli preziosi quali oro ed argento. La criptovaluta *Digix Gold Tokens* (DGX), per esempio, replica l'andamento azionario dell'oro e permette l'acquisto di esso tramite il token stesso. L'ultima categoria opera esattamente come la prima ma invece che ancorarsi a valute legali, si lega a criptovalute esistenti. L'esempio più lampante riguarda la rete Ethereum ed il suo token nativo. ETH non segue il protocollo ERC-20, di conseguenza non è in grado di interagire con numerosi contratti presenti in rete. Mantenendo una proporzione 1:1, è possibile scambiare ETH con *Wrapped Ethereum* (WETH) [14]. Quest'ultimo è a tutti gli effetti un token ERC-20, con valore dipendente dalla moneta nativa.

Lo svantaggio intrinseco all'utilizzo di tali criptovalute riguarda le mancate regolazioni governative. Attualmente, trattandosi di beni digitali, non è presente nessuna garanzia né assicurazione atta alla tutela del capitale. Le *stablecoins* sono uno strumento finanziario ad alto rischio. Non vi è la certezza assoluta che USDC rimanga tale e quale in un tempo indefinito, né tantomeno l'assoluta sicurezza dell'assenza di errori all'interno del contratto.

Capitolo 5 - Sicurezza e vulnerabilità

L'innovazione portata dalla tecnologia blockchain è innegabile. Nonostante ciò, risulta ancora abbastanza complesso esaltarne i vantaggi e interagirci. Per essere compresa a fondo, la struttura dati in questione richiede conoscenze avanzate in materia. Un altro aspetto da tenere in considerazione è la poca chiarezza e trasparenza delle informazioni a riguardo: è facile imbattersi in spiegazioni non alla portata di tutti ed opinioni espresse senza un dovuto studio pregresso.
Per natura della struttura dati, le operazioni sono definite irreversibili ed immutabili. La mancanza di conoscenza di concetti basilari può portare l'individuo a compiere azioni contro la propria volontà, con conseguenze spesso irrimediabili.
L'altro aspetto fondamentale, spesso sottovalutato, riguarda la sicurezza dei beni. Ricordiamo che la persona in possesso delle chiavi di accesso al portafoglio ha pieno controllo sul capitale. In caso di furto, nessun ente di terze parti sarebbe in grado di prestare aiuto. Siamo abituati a poter rimediare a ogni nostra azione, che sia lo smarrimento di una password oppure la perdita dell'accesso al conto corrente. In caso di frode su Internet, la banca provvede al risarcimento del danno. Tutto ciò non è contemplato nel mondo digitale, poiché ognuno è l'unico responsabile delle proprie azioni, quindi delle conseguenze.
In questo capitolo affronteremo le tipologie di attacco più diffuse, con relative contromisure da adottare per tutelarsi.

5.1 Attacchi alla rete

Gli attacchi che ricadono sotto questa nomenclatura riguardano esclusivamente la rete ed i nodi. Non sono coinvolte azioni malevole direttamente sui portafogli degli utenti. Attualmente le tre tipologie di attacchi alla rete più significative sono le seguenti:

- attacco "51%"
- attacco dell'eclissi e Sybil
- attacco al timestamp

5.1.1 Attacco "51%"

L'attacco soprannominato "51%" ha lo scopo di prendere il controllo sull'intera rete, in particolare della potenza computazionale del 51% di tutti i nodi, quindi compromettere l'integrità dell'intera struttura dati. Questa tipologia offensiva è realizzabile solo in reti che utilizzano *Proof-of-Work* come algoritmo di validazione.
Nello specifico, l'obiettivo ricade nell'acquisizione della maggioranza di *hash rate* presente per realizzare una pratica nota come *doppia spesa*. Tramite quest'azione è possibile utilizzare due volte la stessa unità di monete, senza invalidare la catena.
La monopolizzazione della rete da parte di un singolo o di un'organizzazione permette la manomissione delle transazioni presenti all'interno dei blocchi, così come il rifiuto di alcune di esse. Quando un'operazione non viene accettata di proposito, viene riscontrato un mancato servizio, chiamato con il termine inglese *transaction denial of service*. La vittima

rimane in uno stato di attesa indefinita, attendendo un risultato che potrebbe non arrivare mai. Importante notare che dopo un attacco “51%” vengono a crearsi due catene parallele, mantenute rispettivamente da nodi genuini e malevoli. Il problema principale risiede nel fatto che un utente malevolo potrebbe utilizzare un quantitativo di monete sulla catena originale, per poi includere la stessa quantità in un’altra transazione all’interno della rete malevola, privata e non diffusa in tempo reale con il resto dei nodi. Per natura del protocollo, i nodi classificano come fidata la struttura dati con più blocchi, e dato che il 51% della potenza computazionale è in possesso di enti malevoli, la catena maligna verrà ritenuta da tutti come valida.
Per chiarire il processo, prendiamo sotto esame un utente malizioso che intende spendere due volte il proprio capitale di 100 BTC. La prima fase consiste nel creare una propria versione della blockchain attuale, da tenere privata. Successivamente, si realizza la transazione all’interno della rete originale, ottenendo un determinato beneficio. La controparte che ha fornito il servizio all’utente malevolo però non vedrà accreditati i 100 BTC, poiché nel mentre la transazione è stata invalidata e diffusa dall’altra catena, ora pubblica. Infatti, siccome la seconda rete gode del 51% dell’*hashrate* totale, questa è in grado di validare i blocchi più velocemente, risultando più lunga di quella originale. Quindi, tutti i nodi accetteranno quest’ultima come fidata, andando a invalidare la spesa effettuata nella rete originale [figura 10].

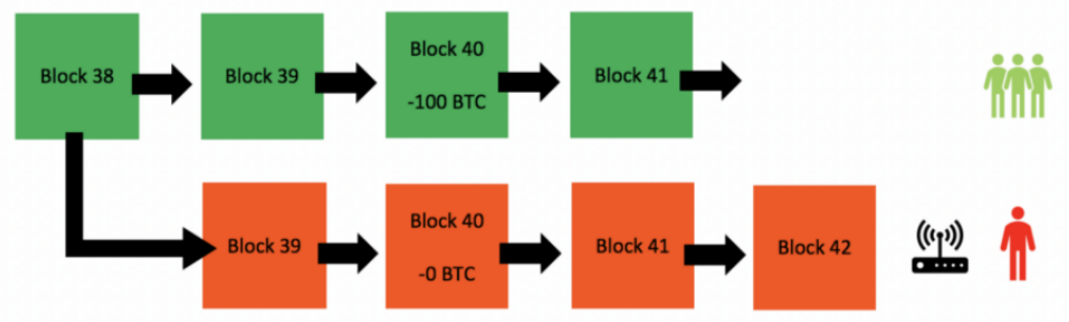

Figura 10 - *Rappresentazione grafica attacco “51%”*

L’attacco appena descritto non aggredisce i *wallet*, bensì solo la rete. Questo significa che tutte le criptovalute possedute, se non inviate all’interno della rete sotto attacco, non risentono dell’accaduto. I danni maggiori, nonostante le ingenti perdite di beni, sono riscontrati a livello di credibilità ed affidabilità della rete. A conferma di ciò, nel 2020 una sottorete di Ethereum denominata *Ethereum Classic* ha subito molteplici attacchi di questo tipo, andando ad evidenziare le problematiche della versione alternativa ed allontanando la maggior parte degli utenti.

La probabilità di subire un attacco “51%” risulta essere inversamente proporzionale alla decentralizzazione della rete. Più alto è il numero di nodi presenti, meno probabile è la riuscita dell’attacco. Compariamo due piattaforme, relativamente composte da 10 e 1000 validatori: nel primo caso sarebbe sufficiente convincere 6 enti per sferrare l’attacco, mentre per il secondo ne servirebbero almeno 501.

Quindi, utilizzando reti note e diffuse, un attacco di questa tipologia è da considerare altamente improbabile.

Da sottolineare è lo scopo dell'attacco: tale azione è volta a distruggere un rete, non manipolarla a proprio favore. Nello scenario in cui si abbia effettivamente il controllo della maggior parte della potenza computazionale operativa, a livello remunerativo sarebbe più redditizio sfruttare tale energia per validare blocchi legittimamente, piuttosto che eseguire operazioni rischiose di doppia spesa. Il guadagno ottenuto dal processo di verifica sarebbe ingente, costante e sostenibile nel lungo periodo. La continua presenza di blocchi orfani insospettirebbe gli utenti che, analizzando il registro, si accorgerebbero della manomissione del corretto funzionamento. A tal punto, cadrebbe la fiducia riposta nel sistema ed il valore della criptovaluta precipiterebbe.

5.1.2 Attacco dell'eclissi e Sybil

L'attacco dell'eclissi (*eclipse attack*) ha lo scopo di isolare un singolo nodo all'interno di una rete *peer-to-peer* immaginaria, al fine di oscurare la visione sull'intera rete. Solitamente questo attacco avviene in preparazione di azioni malevole, quali la doppia spesa. Il requisito fondamentale è il controllo di numerosi nodi all'interno della rete.
L'attaccante cerca di indirizzare le comunicazioni in entrata e uscita verso questi, facendo credere alla vittima di stare comunicando con legittimi enti. La riuscita dell'attacco si basa proprio sul concetto alla base di una rete peer-to-peer: non tutti i nodi possono essere connessi tra loro contemporaneamente, principalmente per problemi di larghezza di banda.
Inizialmente, sfruttando la tecnica *Distributed Denial of Service* (DDos), si obbliga la vittima alla disconnessione dalla rete, per poi successivamente intercettare il nuovo tentativo di connessione e dirottarlo sulle macchine malevole controllate. A questo punto, l'attaccante può fornire versioni del registro contraffatte, senza destare sospetto.
Le conseguenze di questo attacco riguardano la doppia spesa, esplicata in precedenza, e l'interruzione del processo di validazione. Quest'ultimo ha l'obiettivo di impedire la validazione dei blocchi a un nodo, in maniera trasparente, provocando uno spreco di potenza. Una volta che il validatore è circondato da soli enti malevoli, è possibile inoltrare versioni della catena precedenti, omettendo blocchi già risolti. In questo modo, la vittima utilizza energia per aggiungere unità già presenti, senza ricevere alcuna ricompensa. Ciò si traduce in uno spreco di tempo ed energia non indifferente.

Una versione distribuita di questo attacco consiste nella realizzazione di molteplici situazioni analoghe su diversi nodi, con l'obiettivo di impersonificare più utenti possibili ed ottenere un reale *hash rate* del 51%, al fine di portare a termine l'attacco spiegato in precedenza. Questa versione viene denominata come *attacco di Sybil*, nome derivato dal caso di studio di una donna omonima che soffriva di disturbo dissociativo d'identità. Il successo della suddetta azione è fortemente condizionato dai meccanismi di convalida delle identità dei nodi. Se i requisiti per diventare un nodo non sono elevati, sia a livello di risorse che di monete possedute, la probabilità di riuscita dell'attacco aumenta.

La contromisura attuata da alcune reti, tra cui Bitcoin, riguarda l'introduzione di una lista di nodi fidati e la possibilità di effettuare solo connessioni in uscita. La cosiddetta *whitelist* garantisce un insieme di punti verificati, ai quali è possibile connettersi senza incorrere in nodi alterati. Inoltre, viene a mancare la possibilità di effettuare interrogazioni su altri nodi, così da impedire l'adescamento di utenti incoscienti.

5.1.3 Attacco al timestamp

L'attacco al *timestamp*, noto anche come *timejacking*, prevede l'alterazione dello stesso all'interno di un determinato nodo per impedire la corretta sincronizzazione dell'unità con il resto della rete, forzandolo ad accettare una versione della catena differente.
Ogni nodo mantiene un contatore che identifica la data e l'orario esatto corrente, ottenuto attraverso la media dei valori comunicati dai punti della rete adiacenti. Nello specifico protocollo Bitcoin, se i messaggi ricevuti contengono un valore che dista più di 70 minuti da quello posseduto, allora questi vengono scartati. Da questa asserzione si evince facilmente che è presente una finestra temporale valida di massimo 140 minuti, considerando la tolleranza sia per valori passati che futuri. La messa in atto dell'attacco prevede il rallentamento, o velocizzazione, del contatore di un singolo nodo, attraverso comunicazioni imprecise da parte di un elevato numero di utenti malevoli.
Durante il processo di validazione di un nuovo blocco, il timestamp ricopre un ruolo fondamentale. Per l'aggiunta dell'unità alla catena è necessario che il *timestamp* dell'unità non sia maggiore di 120 minuti dal valore interno del nodo. Analogamente, un valore minore della media degli ultimi 11 blocchi presenti nella struttura causa il rifiuto dell'unità.
L'attacco si concretizza con la creazione di un apposito blocco fittizio, con *timestamp* ottenuto aggiungendo 190 minuti al vero orario attuale. Il numero non è casuale: viene utilizzata il limite massimo di validità di un blocco di 120 minuti con l'aggiunta di 70, ossia il valore di discrepanza massima tra l'orario reale e quello interno ad un nodo.
Il requisito fondamentale risiede nell'abilità di rallentare il contatore interno di un nodo del tempo massimo, 70 minuti. A questo punto, la vittima è costretta a rifiutare il blocco fasullo siccome il valore temporale risulta superiore a 120 minuti. Al contempo, gli altri nodi non infetti della rete sarebbero in grado di validare l'unità, poiché il *timestamp* risulterebbe avanti di soli 120 minuti, quindi dentro i limiti previsti dal protocollo.
Il risultato evidenzia l'aggiunta del blocco alla catena, senza la consapevolezza di ciò da parte della vittima, che possiede un registro non aggiornato. La conseguenza dell'attacco può portare ad operazioni di doppia spesa, vista l'ampia finestra temporale a disposizione.
Inoltre, l'attaccante è in grado di tenere occupato il nodo vittima tramite la continua elaborazione di blocchi già validati, provocando uno spreco di energia non indifferente.

L'identificazione di tale attacco non è affatto semplice. Al contempo, una volta confermata è possibile riavviare il nodo infetto ed utilizzare liste di entità sicure alle quali connettersi. Sebbene sia un attacco teoricamente molto efficace, considerando la grandezza della rete Bitcoin ad oggi e la potenza di calcolo necessaria, il *timejacking* risulta quasi impossibile da mettere in atto.

5.1.4 Validazione blocchi *selfish*

La validazione dei blocchi "*selfish*" (*selfish mining*) descrive un tentativo di attacco all'integrità della blockchain. Questa pratica riguarda esclusivamente protocolli basati sull'algoritmo di consenso *Proof-of-Work*, in particolare i gruppi di validazione. Questi non sono altro che un insieme di nodi che collaborano alla risoluzione dei blocchi, mettendo a disposizione la propria potenza di calcolo. In caso di successo, la ricompensa viene distribuita in proporzione all'*hashrate* fornita.
L'attacco prevede che un nodo dopo aver validato un blocco correttamente non lo diffonda nella rete pubblica, bensì lo tenga segreto in una propria catena privata, passando direttamente al *mining* dell'unità successiva. Infine, avviene il rilascio contemporaneo di tutti i blocchi completati da parte del singolo, con lo scopo di dimostrare al gruppo il possesso di una potenza computazionale maggiore a quella realmente utilizzata. Utilizzando questa strategia, il singolo nodo è in grado di ottenere ricompense ben maggiori.
La pratica appena descritta può essere messa in atto da molteplici nodi contemporaneamente, anche situati all'interno dello stesso gruppo. Solitamente i nodi che godono di maggiore potenza tendono ad inglobare le ricompense spettate ai piccoli partecipanti del gruppo, senza destare troppo sospetto. Sebbene l'attacco sembri altamente improbabile, la curva di successo può essere descritta attraverso il *processo di Markov* [15]. Questo enuncia sinteticamente che la probabilità di transizione di stato, da uno iniziale ad uno finale, dipende esclusivamente dallo stato immediatamente precedente del sistema.
Non viene considerata alcuna memoria presente nel sistema stesso.
Disponendo in un piano cartesiano la ricompensa dei blocchi sull'asse delle ordinate, e la potenza di calcolo del singolo rispetto al gruppo in percentuale su quello delle ascisse, è possibile individuare il punto esatto nel quale l'attacco risulta profittevole. In caso di validazione lecita, il guadagno è proporzionale alla potenza fornita. Geometricamente parlando questo è rappresentato dalla retta $y = (1/4) * x$. Invece, l'attività illecita di un nodo è graficamente rappresentabile attraverso una curva esponenziale, simile all'andamento della funzione $y = e^{x-2}$.

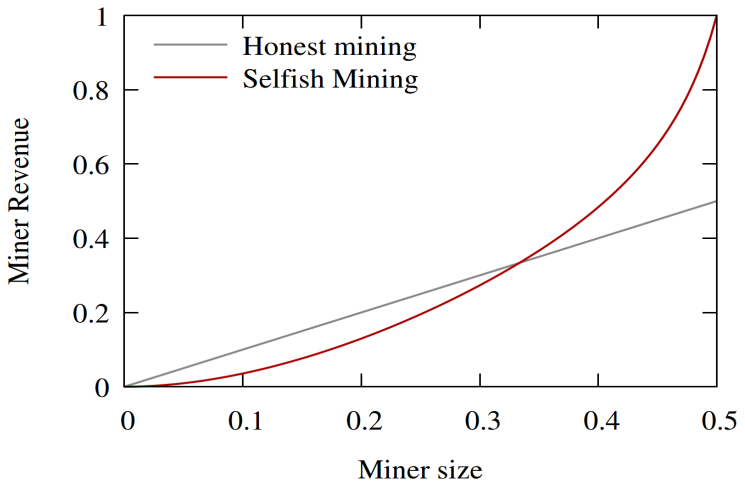

Figura 11 - *Grafico rapporto dimensione nodo e ricompense*

Il risultato, evidenziato dal grafico, suggerisce che un nodo malevolo di potenza maggiore del 30% dell'intero gruppo otterrebbe ricompense più elevate se attuasse la suddetta procedura malevola.

Le misure di prevenzione per questa tipologia di attacco riguardano principalmente l'assegnazione casuale dei nodi alle diverse sottocatene presenti. Ogni singolo lavora per un ipotetica struttura principale, non sapendo a priori se il proprio personale sforzo sarà utile alla comunità. Tenere privati alcuni blocchi risolti perde di significato e guadagno, siccome non si ha la certezza che la catena sia quella più lunga, quindi valida. L'altra contromisura al *selfish mining* prevede l'imposta di un tempo massimo entro il quale è necessario fornire una possibile soluzione, eliminando la possibilità di accumulare blocchi validati in catene private.
Sfortunatamente, questa tipologia di attacco ha avuto una percentuale di successo elevata nel passato ed è applicabile a qualunque rete con algoritmo di consenso *PoW*.

5.2 Attacchi al wallet

Gli attacchi ai wallet si verificano di gran lunga con più frequenza. Il comune denominatore è quello di ottenere l'accesso alle chiavi del portafoglio. Per essere attuati è necessario un minore sforzo computazionale, ragione per cui sono preferiti da un numero maggiore di malintenzionati. Inoltre, essi vanno anche a fare leva sulla scarsa preparazione di alcuni nuovi utenti, includendo un fattore psicologico. Di seguito sono riportati i nomi di alcuni tra i più noti attacchi appartenenti a questa categoria:

- Vulnerabilità delle chiavi private
- Phishing
- Attacchi al software wallet
- Attacchi all'hardware wallet

5.2.1 Vulnerabilità delle chiavi private

La firma digitale è stata progettata per proteggere l'integrità e l'autenticità dei dati, nel caso specifico interni della blockchain. Attraverso l'utilizzo di tale meccanismo è possibile verificare l'identità di colui che ha prodotto il dato e anche l'assenza di modifiche da quel preciso istante in poi. Prendendo come esempio la rete Bitcoin ed il cifrario asimmetrico sfruttato ECDSA, attualmente non esiste calcolatore in grado di intaccare la sicurezza dell'algoritmo.
Nonostante ciò, questi sono da considerare efficaci solamente se utilizzati correttamente.
La maggior parte degli algoritmi asimmetrici richiede, come parametro, un numero ottenuto in modo totalmente imprevedibile. Di fondamentale importanza è la presenza di un generatore di numeri casuali che goda di un alto livello di entropia, quindi di una bassa probabilità di prevedere il risultato.
Uno studio pubblicato nel gennaio del 2013 [16] ha dimostrato le problematiche intrinseche ad un'errata scelta dei parametri dell'algoritmo ECDSA, utilizzato all'interno della blockchain Bitcoin.

La funzione ellittica prende in ingresso le variabili racchiuse nella tupla *(CURVA, G, n)*, che identificano rispettivamente l'equazione ellittica utilizzata, il generatore della curva ed un ordine intero di *G*, tale per cui $n * G = O$. Oltre alla tupla, è necessario fornire il *digest* del messaggio, la chiave privata (*PK*) ed un numero generato casualmente (*k*) compreso nell'intervallo *1* ed *n-1*. In un passaggio intermedio si compone anche una stringa (*z*), estraendo i primi *L* bit dell'*hash* del messaggio, dove *L* equivale alla lunghezza in bit del gruppo di ordine *n*. Il risultato della computazione restituisce una tupla (*r, s*), la quale costituisce la firma digitale del messaggio.
Per l'analisi dello studio, risulta essenziale chiarire come il valore finale di *r* è strettamente dipendente da *G* e *k*. Ma se il parametro *G* è di dominio pubblico, utilizzato successivamente per la verifica della firma, il numero casuale *k* ricopre un ruolo fondamentale nella sicurezza dell'algoritmo. Inoltre, la seconda componente della firma digitale viene computata attraverso l'equazione $s = k^{-1}(z + r * pk) \bmod n$. Lo standard impone che vengano selezionati *k* differenti per firme diverse, altrimenti l'equazione per il calcolo di *s*, secondo campo della firma digitale, può essere invertita a favore della chiave privata.
Per concretizzare la teoria esposta, ipotizziamo lo scenario in cui due transazioni differenti abbiano firme digitali equivalenti a (*r, s1*) ed (*r, s2*). Data l'uguaglianza della prima componente di ognuna, è deducibile che il numero casuale utilizzato sia il medesimo per entrambe. Invertendo la formula per il calcolo della seconda componente è possibile risalire alla chiave privata, utilizzando la seguente formula:

$$PK = (z1 * s2 - z2 * s1)/(r * (s1 - s2))$$

Gli elementi presenti *z1* e *z2* rappresentano le stringhe estratte a partire dal *digest* del messaggio originale, mentre *s1* ed *s2* sono la seconda componente della firma generata.
L'attaccante in possesso della chiave privata del portafoglio è in grado di gestire tutto il relativo contenuto. Questa procedura è realizzabile esclusivamente se il generatore di numeri casuali risulta difettoso, non garantendo un adeguato livello di entropia. Utenti malintenzionati sono in costante monitoraggio delle transazioni interne alla blockchain, alla ricerca di possibili firme vulnerabili.
Quindi è buona norma assicurarsi che ogni transazione, autorizzata dallo stesso *wallet*, abbia entrambe le componenti della firma digitale differenti. Infatti, l'algoritmo crittografico non rappresenta una minaccia, bensì è fondamentale conoscerne le fondamenta per utilizzarlo nel modo corretto.

5.2.2 Attacco al software wallet

Un software *wallet* è identificato da un programma che necessita di una connessione Internet per interagire con la blockchain. Questo rappresenta la via più veloce gestire le proprie criptovalute, rimanendo sempre in possesso delle chiavi. Generalmente un software wallet può essere un vero e proprio applicativo da installare sul dispositivo oppure anche una semplice estensione del web browser. Data la facilità d'installazione e configurazione, solitamente viene prediletta la seconda opzione. Le operazioni preliminari da effettuare riguardano la creazione, od importazione tramite *seed phrase*, del portafoglio e la configurazione di una password personale di accesso.

Questa tipologia di portafoglio, come ogni altro programma sul dispositivo, è soggetta a *malware* con la finalità comune di sottrarre criptovalute agli utenti.
Il programma malevolo denominato *Bitcoin Clipper* è stato rilevato per la prima volta nel 2017 in ambiente *Windows*, per poi evolversi e colpire anche piattaforme mobili *Android*. Questo opera con lo scopo di agire durante la fase di creazione di una transazione: sostituisce il vero indirizzo di destinazione con quello dell'attaccante.
Come si evince dal nome del programma, questo opera sulla rete Bitcoin e sfrutta la struttura formale degli indirizzi. Infatti, come già esposto nel capitolo precedente, ognuno di essi deve necessariamente avere lunghezza compresa tra 26 e 35 caratteri inclusi e deve iniziare con una delle seguenti stringhe di caratteri, a seconda della versione della rete: "1", "3", "bc1q", "bc1p". Vi è la possibilità di comporre un'espressione regolare per controllare se una determinata stringa potrebbe rappresentare un indirizzo appartenente alla rete Bitcoin.
Con il fine popolare i campi di recapito di una transazione, è buona pratica avvalersi del cosiddetto *copia ed incolla* per evitare di incappare in errori di battitura. L'area di memoria temporanea, chiamata comunemente *appunti*, contenente le stringhe appena copiate è per natura ad accesso pubblico, sia in modalità di lettura che scrittura.
Note le suddette premesse, non risulta difficile immaginare la funzione eseguita dall'eseguibile *Bitcoin Clipper*. Questo si occupa di monitorare periodicamente gli *appunti* alla ricerca di un indirizzo valido, da andare a sostituire con quello dell'attaccante. Nel momento della compilazione del campo destinazione della transazione il valore incollato risulterà essere la stringa modificata precedentemente. L'operazione verrebbe finalizzata con successo, inviando il capitale prestabilito al malintenzionato.
La contromisura più efficace è sicuramente il controllo manuale a posteriori, prima dell'immissione della transazione in rete. Solitamente è buona norma controllare che i primi ed ultimi caratteri dell'indirizzo coincidano con la reale destinazione desiderata. In caso di discrepanze tra l'originale e quello incollato, probabilmente la macchina risulta infetta.

Un altro vettore d'attacco, diffuso attraverso servizi di posta elettronica, prevede l'interazione del portafoglio con pagine web sospette. Si tratta a tutti gli effetti di tentativi di *phishing*, atti ad adescare gli utenti meno esperti e convincerli a fornire la *seed phrase* in cambio di un ipotetico servizio quale un controllo di sicurezza, una ricompensa o un accredito di criptovalute gratuito.
L'attacco consiste nel fare leva sull'aspetto psicologico della persona, cercando di convincerla dei vantaggi del servizio offerto. Solitamente viene visualizzata una scadenza temporale, così da porre ulteriore pressione mentale. L'obiettivo è garantire un innegabile guadagno in cambio di un'informazione "banale", agli occhi di utenti neofiti. I messaggi ricevuti inoltre contengono informazioni riconducibili al portafoglio, quali una lista degli ultimi movimenti. Infatti, data la natura pubblica della blockchain, non risulta affatto complicato ottenere una sorta di estratto conto del wallet. Ma ciò nonostante, queste informazioni sono mirate ad ottenere una maggiore fiducia da parte dell'utente.
Anche l'indirizzamento della vittima su pagine apparentemente innocue può provocare danni irreparabili. Quando un sito web intende interagire con l'estensione che funge da portafoglio, viene visualizzato un messaggio di richiesta di autorizzazione. Attraverso un semplice click del mouse e con una buona dose di distrazione, è possibile delegare l'accesso al nostro wallet all'attaccante. La tipologia di pagine in questione spesso replica la richiesta di aggiornamento del portafoglio stesso, nel modo più fedele possibile all'originale. Le conseguenze dell'attacco risultano evidenti nel momento in cui si apre il wallet, dal momento che il saldo risulterà nullo. Nello specifico, attraverso l'autorizzazione del contratto proposto

dal sito web, l'utente ha acconsentito implicitamente a tutte le sue clausole. Questi contengono una funzione di prelievo totale, che permette di trasferire il saldo del portafoglio ad un indirizzo prestabilito, solitamente quello dell'attaccante.
La vulnerabilità, in questo caso, risiede sia nella disattenzione dell'utente che nella facilità con la quale è possibile approvare un'operazione. Quest'ultima proprietà funge da lama a doppio taglio, che risulta utile solamente con un'adeguata conoscenza ed esperienza. L'introduzione di *hardware wallet* ha imposto l'approvazione di qualunque operazione attraverso un dispositivo fisico, riducendo drasticamente la possibilità di compiere errori provocati da disattenzioni.

5.2.3 Attacco all'hardware wallet

Senza ombra di dubbio, il metodo migliore per conservare la chiave privata del portafoglio è l'utilizzo di un hardware wallet. In sintesi, il processo di approvazione di una transazione deve passare obbligatoriamente per il dispositivo fisico, imponendo l'interazione dell'utente con esso. Inoltre, questi apparecchi sono molto apprezzati per la mancata comunicazione con la rete, fattore che li rende immuni dalla maggior parte degli attacchi software.
Nonostante ciò, gli hardware wallet espongono l'utente ad un intrinseco svantaggio inevitabile: la sicurezza dei componenti fisici. Infatti, sebbene il software che viene eseguito all'interno del dispositivo sia sviluppato con particolare attenzione ai dettagli, non si può dire lo stesso per quanto riguarda i microcontrollori presenti. Per abbattere i costi di produzione, molte aziende tendono ad adottare componenti già sviluppati in passato, adattandoli alle proprie esigenze. Questo può rendere alcuni modelli di hardware wallet vulnerabili a modifiche dei componenti non autorizzate, con l'obiettivo di estrarre la chiave privata da essi.
L'attacco che andremo ad analizzare è tratto da una storia reale e sfrutta una vulnerabilità presente sul dispositivo *Trezor One* [17], prodotto nel 2018. L'ingegnere informatico ed hacker etico *Joe Grand* [18] ha rivelato come ha realizzato l'attacco, andando a recuperare criptovalute per un valore superiore a due milioni di euro. Il lavoro è stato commissionato da un individuo che ha accidentalmente dimenticato il PIN di accesso al dispositivo. Infatti per autorizzare qualsiasi transazione, il portafoglio richiede l'immissione di un codice con lunghezza superiore a 4 cifre, scelto arbitrariamente dall'utente durante la procedura di inizializzazione. Se il limite massimo di tentativi errati consecutivi, fissato a 16, viene oltrepassato, il dispositivo è programmato per eseguire la cancellazione forzata di qualunque dato presente all'interno.
La vulnerabilità scoperta dall'ingegnere si presenta durante la fase di aggiornamento del *firmware* interno. Il comportamento ideale per lo svolgimento di questa operazione prevede che il dispositivo passi in modalità di manutenzione, effettui uno spostamento della chiave privata e del PIN in un'area della memoria RAM protetta, installi la nuova versione del software ed infine ripristini i dati sensibili presenti nella memoria temporanea. Il problema del dispositivo *Trezor One* riguarda la gestione dei dati tra le memorie: la chiave privata ed il PIN non vengono spostati, bensì copiati. Il principale rischio di attuare un attacco su di un dispositivo fisico riguarda l'alta probabilità di eliminare accidentalmente tutti i dati contenuti in esso. Però, nel caso preso sotto esame, l'ingegnere punta all'accesso di una mera copia delle informazioni, senza preoccuparsi di una possibile alterazione di esse.

Questo introduce la possibilità di realizzare un attacco hardware noto con il nome di *glitching*, o *iniezione dell'errore* (fault injection). Come suggerisce il termine *glitch* stesso, il elettronica questo evento identifica un picco breve ed improvviso di una forma d'onda, causato da un errore non prevedibile, che evidenzia un difetto del sistema.
Per la corretta comprensione della procedura d'attacco, è necessario accennare i tre livelli di sicurezza utilizzati dal dispositivo *Trezor*: RDP2, RDP1 ed RDP0. Il primo non permette la lettura della memoria, mentre gli altri due la consentono per svolgere le operazioni di aggiornamento del *firmware*. Il livello impostato durante il normale utilizzo è RDP2.
Note tutte le caratteristiche del dispositivo ed il relativo funzionamento, l'attacco consiste in un *glitching* sul microcontrollore, il quale altera il voltaggio fornito al processore interno. Quindi, agendo nell'esatto istante del malfunzionamento del dispositivo, è possibile abbassare il livello di sicurezza da RDP2 ad RDP1. Successivamente viene imposta la modalità di manutenzione, la quale copia sia la chiave privata che il PIN nella memoria RAM. Infine, dato il livello RDP1, è possibile leggere il contenuto presente e stamparlo a video.
L'unico problema dell'attacco risulta trovare i parametri giusti per eseguire correttamente il *glitch* sul processore, quindi innescare la procedura per la riduzione del livello di sicurezza. L'operazione realizzata da *Joe* impiegò quasi quattro ore prima di ottenere con successo le informazioni dimenticate, permettendo il recupero di tutto il capitale.
L'azienda ha tempestivamente rilasciato un aggiornamento per la copertura del difetto presente. Nonostante ciò, il microcontrollore generico utilizzato non è stato revisionato. Ad oggi, la *patch* software dovrebbe essere sufficiente a garantire un grado di sicurezza sufficientemente elevato.

Non esistono contromisure specifiche in questo caso, semplicemente una serie di accortezze da controllare prima dell'utilizzo di qualunque hardware wallet. L'acquisto di questi dispositivi deve avvenire solo da rivenditori di fiducia, con successiva verifica di eventuali segni di manipolazione della confezione del prodotto. Inoltre, la conservazione del portafoglio in un luogo sicuro e non alla portata di tutti è un aspetto di fondamentale importanza. Minore è il numero di individui che sono a conoscenza del wallet, minore è la probabilità di subire un attacco.

5.3 Attacchi tramite *smart contract*

Gli *smart contracts* vengono utilizzati per garantire la corretta esecuzione operazioni complesse, così come per la gestione di token all'interno di una specifica blockchain. Nonostante gli innumerevoli vantaggi, questi possono presentare alcuni difetti di sviluppo ed essere utilizzati per scopi non previsti. Inoltre, i contratti prevedono l'implementazione obbligatoria di funzioni standard considerate critiche, dal momento che, se attuate in determinate circostanze, potrebbero causare danni non indifferenti agli utenti.
Le azioni malevole più significative, che sfruttano contratti, sono riportate di seguito:

- prelievo multiplo attraverso token ERC-20
- rimozione della liquidità

5.3.1 Prelievo multiplo attraverso token ERC-20

Lo standard ERC-20 mette a disposizione un insieme di funzioni per la gestione del contratto. In particolare, l'attacco sfrutta la concorrenza tra la chiamata di due procedure quali *approve(spender, amount)* e *transferFrom(source, recipient, amount)*. Quest'ultima viene derivata dalla funzione originale *transfer(recipient, amount)* ed esegue il trasferimento di monete dalla sorgente alla destinazione. La differenza con l'originale risiede nel chiamante: mentre *transfer* viene invocata dal mittente, *transferFrom* deve essere chiamata dal destinatario. La procedura derivata può essere eseguita solamente dopo il metodo *approve(spender, amount)*, il quale abilita un determinato indirizzo a spendere una quantità prestabilita di token, posseduti dal chiamante.
Le due funzioni vengono sfruttate nell'attacco per permettere al destinatario di spendere più monete di quelle attese. Questo è possibile siccome il metodo *approve* sovrascrive la quantità di token spendibili, indipendentemente se questi sono già stati utilizzati o meno.
Il conflitto viene a crearsi quando il mittente stabilisce una cifra che il destinatario può utilizzare, quindi decide di modificare questo numero. Se la transazione di trasferimento avviene prima della modifica, allora il destinatario sarà in grado di spendere anche il quantitativo impostato la seconda volta. Per chiarire il processo e la condizione di concorrenza critica, andiamo ad analizzare il seguente esempio.
L'utente A permette a B di trasferire N token, attraverso la chiamata *approve(B, N)*. Successivamente, A intende modificare la cifra spendibile da N a M, eseguendo *approve(B,M)*. L'utente B rileva la seconda transazione di A prima che essa sia confermata, quindi esegue immediatamente *transferFrom(A, B, N)* per prelevare N token. Per assicurarsi la precedenza sulla modifica dell'importo utilizzabile, B imposta un delle commissioni molto elevate, rendendo la sua transazione più profittevole per i validatori. Successivamente, il limite delle monete spendibili viene modificato come richiesto e B è in grado di trasferire altri M token nel proprio wallet. Al termine, B è stato in grado di ottenere N + M unità totali.
L'attacco è possibile dal momento che è responsabilità del primo utente assicurarsi che i fondi non siano già stati utilizzati, prima di immettere un'altra transazione.
Per mitigare i potenziali danni causabili da questa vulnerabilità, è altamente consigliato di verificare la seguente condizione prima di permettere la modifica di un valore già impostato.

require((amount == 0) || (allowed[msg.sender][spender] == 0))

La funzione assicura che l'indirizzo destinazione non abbia alcun permesso di gestire i token del chiamante, così da evitare problemi legati all'ordine cronologico di esecuzione delle transazioni.

5.3.2 Criptovalute truffa

Il processo di sviluppo di un nuovo token si traduce nella semplice implementazione di un contratto, il quale contiene tutte le funzioni specificate nello standard ERC-20 (nel caso di blockchain Ethereum). Il protocollo prevede, come nel caso di una classe astratta, il

mantenimento invariato della dichiarazione del metodo, dando comunque la possibilità di modificare il comportamento interno.
Terminato lo sviluppo, il contratto viene pubblicato all'interno della rete e la nuova criptovaluta diventa scambiabile attraverso exchange decentralizzati (DEX). Come accade per ogni operazione all'interno della blockchain, il codice diventa visibile pubblicamente. Nonostante ciò, gli individui che si soffermano all'analisi minuziosa del codice sono ben pochi, data la difficoltà del linguaggio. La mancanza d'attenzione è spesso sfruttata da sviluppatori malintenzionati tramite l'implementazione di funzioni apparentemente legittime, ma che contengono clausole nascoste.
L'alterazione di un contratto più comune agisce sulla modifica delle commissioni applicate nel momento di acquisto e di vendita. L'attacco consiste nell'attendere che il token inizi ad essere scambiato in grandi quantità, quindi portare la percentuale di commissione pari al 99%. Ogni transazione successiva porterà alla perdita quasi totale del capitale, rendendo impossibile sia la vendita che l'acquisto. Il contratto è progettato per inviare le commissioni ad un gruppo di indirizzi, solitamente in possesso degli sviluppatori.
Il contratto malevolo del token denominato *M3*, utilizzando l'explorer di Ethereum [20], è consultabile all'indirizzo *0x8ed9C7E4D8DfE480584CC7EF45742ac302bA27D7.*
Il codice risulta alquanto ambiguo, siccome contiene una funzione che cerca di impersonare la dichiarazione di *approve(spender, amount)*. La procedura sospetta è definita come *aprove(a)* ed ha lo scopo di impostare la nuova percentuale di commissioni.

```
120     function approve(address spender, uint256 amount) public virtual override returns (bool) {
121         _approve(_msgSender(), spender, amount);
122         return true;
123     }
124
125     function aprove(uint256 a) public externelBurn {
126         _setTaxFee( a);
127       (_msgSender());
128     }
```

Figura 12 - *Comparazione delle due funzioni di prelievo*

Gli sviluppatori hanno cercato di mascherare la chiamata della funzione, attribuendo un nome pressoché identico. In un codice sorgente legittimo, il valore delle commissioni è statico, non modificabile tramite una funzione dagli sviluppatori.

La seconda manipolazione effettuabile sulle funzioni standard riguarda il potere di vendita. In un contratto malevolo è possibile inserire una condizione che permette la transazione solo ad un ristretto numero di indirizzi. Adottando questa politica, l'utente non sarà in grado di scambiare i token con la moneta nativa della rete. L'evento porta inevitabilmente ad un drastico aumento del valore della singola unità, considerata la mancata possibilità di vendita. L'atto di scambio attraverso DEX prevede che ci sia un gruppo di liquidità (*liquidity pool*) che contiene tutte le controparti del token scambiato. Andando ad acquistare la criptovaluta A,

attraverso B, quest'ultima viene depositata nel gruppo di liquidità rimanendo disponibile per l'operazione inversa di vendita.
Quando la *liquidity pool* diventa consistente, gli sviluppatori possono decidere arbitrariamente di prelevare tutto il contenuto, lasciando tutti gli acquirenti con token dal valore nullo. Negli scambi all'interno di un DEX, il prezzo della moneta viene stabilito dal rapporto tra i due gruppi di liquidità presenti. Se uno di questi è vuoto, il valore dell'altro ricade a zero. L'operazione di drenaggio della liquidità viene chiamata *rug pull*.
La truffa più recente ha sfruttato l'euforia portata dalla nuova serie televisiva *Squid Game* [21]. In data 15 ottobre 2021, il token omonimo identificato tramite la sigla SQUID [22] è stato lanciato all'interno di un DEX, sulla rete *Binance Chain* [2], divenendo scambiabile per pochi centesimi con la moneta nativa della blockchain: *Binance Coin* (BNB). Gli utenti si accorsero presto di possedere un token che non era abilitato alla vendita. L'1 novembre 2021 gli sviluppatori hanno rimosso tutti i *Binance Coin* dalla liquidità, portando il prezzo di SQUID da quasi 3000 euro a zero in pochi minuti. L'evento è stato videoregistrato e commentato in diretta, per poi essere pubblicato sul canale *YouTube* dell'utente americano *Sykogene* [23].

5.4 Altre problematiche

5.4.1 Limiti attuali

La struttura blockchain porta un'innovazione non indifferente nel mondo tecnologico attuale, dall'integrità dei dati fino al consenso distribuito, rimuovendo la necessità di enti centrali.
Nonostante ciò, ci sono altrettanti aspetti da tenere in considerazione quando si parla di reti decentralizzate: alcuni di essi riguardano direttamente la figura umana, altri sono identificabili come problematiche non ancora risolte, fino ad arrivare a veri e propri svantaggi.
Ad oggi, lo sviluppo di servizi ed applicativi decentralizzati non è ancora molto diffuso, così come i tecnici in grado di studiare, realizzare e mantenere un progetto basato su blockchain. Sebbene la figura dello sviluppatore sia ben affermata nella società attuale, l'analista e progettista blockchain è alquanto raro. Di conseguenza, risulta ancora troppo difficile pensare di offrire un prodotto finale che necessiti esclusivamente dell'uso di tale struttura dati. Inoltre, ogni linguaggio di programmazione è altamente specifico per una determinata blockchain. Questo implica un basso livello di possibile cooperazione tra applicativi appartenenti a catene di blocchi differenti.
Un'altra problematica non indifferente riguarda la responsabilità acquisita da ogni singolo individuo. All'interno di un ecosistema decentralizzato, ogni transazione risulta irreversibile. La possibilità di sbagliare non è contemplata. Nello scenario in cui l'utente compili in modo errato il campo di destinazione di una transazione, la quantità di monete risulterebbe persa per sempre. Oppure ancora, lo smarrimento della chiave privata comporterebbe l'inevitabile perdita di tutto il capitale contenuto nel wallet. Inoltre, non esiste ancora nessun tipo di assicurazione né tantomeno regolazione a livello legislativo, atta alla tutela del cliente.
Questo fattore non incentiva molti individui ad adottare tale tecnologia come strumento finanziario, specialmente se con carenti nozioni in materia.

In un sistema centralizzato l'aumento della capacità computazionale è direttamente proporzionale all'incremento delle risorse. In una rete decentralizzata, la scalabilità può rappresentare un ostacolo non indifferente. La velocità con cui una transazione viene confermata dipende dalla congestione di rete, il che si traduce in una latenza maggiore in presenza di più individui. Inoltre, la blockchain utilizzata è determinante per la stima delle conseguenze di una carenza di scalabilità. La rete bitcoin è in grado di gestire circa 5-7 TPS, d'altra parte un attuale sistema di pagamento centralizzato può raggiungere le 1700 TPS.
Non tutte le strutture soffrono di questa problematica, anche se è evidente che ancora non si è ancora raggiunta una soglia di scalabilità sufficiente a sostenere la mole di transazioni attuali, nella maggior parte dei casi.
Le infrastrutture utilizzate sia nei servizi pubblici che privati non possono essere rimpiazzate completamente da un giorno all'altro con un'ipotetica piattaforma decentralizzata. Sorge un problema che riguarda la compatibilità e l'interoperabilità tra le due categorie di sistemi. Come analizzato finora, non è presente una tecnologia immune da vulnerabilità e problemi. Di conseguenza, sfruttando la coesistenza di un entrambe le tipologie di rete, centralizzate e decentralizzate, si potrebbero raggiungere risultati non ottenibili altrimenti. Infatti, non è facile individuare un caso d'uso pratico nel quale l'utilizzo della struttura blockchain sia vantaggioso ed efficiente sotto tutti i punti di vista. Il registro distribuito è di dominio pubblico, quindi consultabile da chiunque. Non è pensabile il salvataggio di dati sensibili in una struttura composta da nodi pubblici. D'altra parte, la creazione e l'utilizzo di una rete privata di piccole dimensioni, con sole unità autorizzate, limiterebbe sia la decentralizzazione che la scalabilità e non sarebbe immune da possibili attacchi interni.
Per natura, la catena a blocchi è una struttura dati immutabile nel tempo. Ciò significa che una volta eseguita un'operazione, questa risulta irreversibile. La medesima proprietà è applicata ai dati contenuti. Questa caratteristica è stata pensata per preservare e garantire l'integrità delle transazioni presenti. Nonostante ciò, l'immutabilità può non rispecchiare le necessità di tutti i casi d'uso. Un generico sistema è progettato per evolvere nel tempo, consentendo l'aggiunta e la modifica di informazioni utili. La possibilità di rettifica dei dati, in alcuni casi, è imprescindibile.
L'ultimo aspetto critico, che affligge maggiormente alcune blockchain, riguarda i costi di utilizzo della piattaforma. Come spiegato dettagliatamente in precedenza, ogni transazione deve includere una commissione per essere processata. Il pagamento dell'imposta identifica il guadagno del validatore. Dato che una rete decentralizzata è composta da molteplici nodi, ognuno di essi dovrà sopperire a costi di manutenzione ed utilizzo, sia a livello di componenti dell'elaboratore che di risorse energetiche. Confrontando una piccola blockchain privata, composta da qualche decina di macchine, ed un sistema centralizzato risulta abissale la differenza delle spese da sostenere. Inoltre, operando all'interno di una rete privata decentralizzata, pochi individui sarebbero disposti al pagamento di una commissione per utilizzare un proprio sistema.

5.4.2 Impatto ambientale

Negli ultimi anni, il consumo di energia globale è cresciuto esponenzialmente. Inoltre, il progresso tecnologico, avvenuto negli ultimi dieci anni, ha alimentato in maniera non indifferente il consumo di energia elettrica. In particolare, vorrei focalizzare l'attenzione sulle

reti decentralizzate ed eseguire un'analisi dettagliata riguardo i consumi attuali, comparati con quelli dei tradizionali sistemi informatici e finanziari.
Inizialmente è possibile fare una macro distinzione tra le reti a seconda dell'algoritmo di validazione adottato. *Proof-of-Work* viene utilizzato principalmente dalle blockchain più datate, quali *Bitcoin* ed *Ethereum*. In estrema sintesi, *PoW* prevede la risoluzione di complessi problemi matematici, da parte di migliaia di elaboratori. Inoltre, più blocchi vengono estratti e più la difficoltà aumenta, richiedendo maggiore energia. Esso garantisce un maggiore livello di sicurezza, a discapito dell'efficienza energetica. Senza ombra di dubbio, questo algoritmo risulta essere il meno sostenibile.
D'altra parte, *Proof-of-Stake* è stato sviluppato negli ultimi anni, con lo scopo di alleggerire il carico computazionale e promuovere la scalabilità e l'efficienza energetica, a scapito della sicurezza. Per l'operazione di verifica delle transazioni, *PoS* prevede che i nodi validatori depositino le monete all'interno della rete. L'algoritmo in questione è stato pensato anche con l'obiettivo di ridurre la quantità di risorse energetiche, favorendo uno sviluppo sostenibile.
L'unità di misura *wattora* (Wh*)* identifica il consumo di energia (W) nell'arco di tempo di un'intera ora. A seconda delle dimensioni dell'evento analizzato, vengono spesso utilizzati i multipli del *wattora*. Per lo studio sottostante l'unità di misura presa in considerazione è preceduta dal prefisso del sistema internazionale *Tera* (T), equivalente a 10^{12}. L'unità di misura *Terawattora* è identificata dal simbolo *TWh*.
La sola rete Bitcoin, nell'anno solare 2020, ha realizzato un consumo stimato di 120 TWh mentre Ethereum si è attestata intorno ai 26 TWh, in accordo con il giornale *Medium* [24].
Questi numeri, se non contestualizzati, non forniscono un'informazione tangibile riguardo la quantità di risorse utilizzate. L'unione dell'energia consumata dalle due blockchain *proof-of-work* appena nominate è maggiore di tutte le altre reti esistenti sommate.
Per rendere l'idea di quanto sia inefficiente, parlando di risorse energetiche, l'algoritmo *PoW* basti pensare che la rete *Proof-of-Stake* più estesa risulta essere *Cardano*, la quale ha un consumo annuo stimato di 6 GWh, inferiore allo 0.01% di Bitcoin. Un'altra blockchain emergente, di tipologia 3.0, chiamata *Tezos* ha un impatto ancora minore di circa 0.06 GWh annui [24].

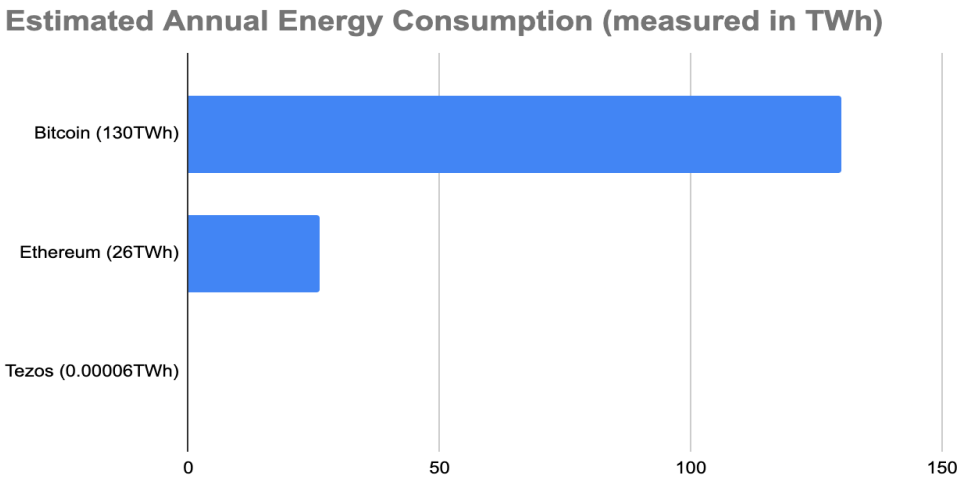

Figura 13 - *Istogramma dei consumi energetici blockchain*

Il confronto, riguardo le risorse necessarie, all'interno dell'universo blockchain evidenzia in maniera inequivocabile quanto l'algoritmo di consenso *Proof-of-Work* sia dispendioso.
Nonostante ciò, questo rimane il metodo di verifica delle transazioni più sicuro ed affidabile.
Per dare una reale dimensione all'impatto ambientale portato dalla rete Bitcoin, relazioniamo il suo consumo energetico con quello di alcune nazioni mondiali.
Nel 2020 la richiesta di risorse energetiche da parte della Norvegia, in accordo con il giornale tedesco *Deutsche Welle* [25], è stata pari a 124 TWh, solo quattro unità in più rispetto a Bitcoin. Gli Emirati Arabi Uniti hanno registrato un consumo di 119 TWh, mentre altri stati europei quali Olanda e Belgio si aggirano intorno ai 100 TWh. Attualmente, ai vertici della classifica è presente la Cina, con 6453 TWh consumati. Il valore stimato per la rete decentralizzata è destinato a crescere negli anni, in proporzione all'aumento della difficoltà dei blocchi. Mentre la rete Ethereum ha in programma per il 2022 la transizione da *PoW* a *PoS*, Bitcoin non intende modificare il protocollo attuale.
Tutta l'energia utilizzata dalla rete non proviene da combustibili fossili, bensì la maggior parte sfrutta fonti rinnovabili. Secondo uno studio pubblicato dall'università di *Cambridge* [26], il 62% dei validatori si affida ad energia idroelettrica mentre il 38% rimanente utilizza sia carbone che anche fonti alternative, quali solare e geotermico. Di fondamentale importanza è ricordare che i quantitativi di energia utilizzati da ogni blockchain sono stime approssimate, dipendenti sia dal valore del bene che dalla complessità del processo di verifica.
Sebbene il sistema finanziario offerto da Bitcoin risulti oneroso a livello energetico, è bene paragonarlo all'attuale sistema bancario mondiale. Infatti, uno studio pubblicato da *FridaysForFutures* [27] evidenzia come la rete decentralizzata globale utilizzi un ammontare di risorse pari solo al 7% di quelle impiegate dall'attuale sistema finanziario centralizzato. Quest'ultimo include tutte le fasi di generazione e gestione delle valute legali, dalla stampa alla distribuzione, fino al mantenimento delle sedi fisiche operative.

L'impatto ambientale di Bitcoin non è certamente indifferente ma l'utilizzo di reti decentralizzate, supportate da algoritmi di consenso *Proof-of-Stake,* potrebbe portare ad una reale rivoluzione dell'attuale sistema economico e digitale globale.

Capitolo 6 - Conclusioni

L'elaborato proposto ha lo scopo di introdurre ed analizzare la struttura dati blockchain, proponendo una visione completa dei benefici apportati e dei protocolli basati su di essa. In secondo luogo, la concentrazione si è spostata sulle attuali vulnerabilità e problematiche di sicurezza presenti.
Ad oggi, la blockchain è ancora ritenuta una struttura ancora molto giovane e di poca utilità. La mancata conoscenza in materia porta all'allontanamento da questa nuova tecnologia. Infatti, i riferimenti bibliografici sono spesso da filtrare, dato il basso livello di chiarezza in merito all'argomento. La carenza di materiale affidabile rende difficile un'istruzione da autodidatta, in quanto è facile incappare in documenti non originali o poco chiari. La maggior parte dei testi analizzati preclude la lettura a persone senza studi passati in campo informatico, visto l'alto grado di tecnicismi presenti.
I vantaggi apportati dalle reti decentralizzate sono evidenti ed innegabili, ciò nonostante risultano ancora poco evidenti alla maggior parte delle persone. Oltretutto, per sfruttare a pieno tale tecnologia è necessaria una certa consapevolezza da parte dell'utente, siccome il livello di responsabilità non è più indifferente. Attualmente le criptovalute sono classificate come investimenti ad alto rischio, proprio per l'elevata volatilità del valore e l'assenza di regolazioni consistenti.
Nella vita di tutti i giorni ci siamo abituati ad accettare alcuni limiti imposti dall'alto, pensando che essi siano parte della normalità. L'esempio lampante riguarda le restrizioni temporali e spaziali presenti nell'attuale sistema finanziario: le diverse valute internazionali e l'impossibilità di operare in maniera continuativa sono caratteristiche che rendono alquanto inefficace la produttività del singolo individuo. La famosa citazione anonima, in circolazione nel web da diversi anni, *"Don't trust, verify!"* esprime in poche parole il vero significato di un sistema decentralizzato. Non è necessario riporre la fiducia in un ente di terze parti quanto è possibile verificare matematicamente l'insieme delle operazioni interne all'ecosistema.

Personalmente credo fermamente in una possibile integrazione futura di tale tecnologia nella vita quotidiana, ma al contempo riconosco l'impossibilità di un cambiamento immediato per i motivi citati. Negli ultimi anni l'adozione ha seguito un andamento esponenziale che non accenna a rallentare, anche con la legalizzazione di *Bitcoin* come valuta in alcuni stati. Ciò indica che sempre più persone riconoscono il potenziale, ancora inespresso, celato in una rete decentralizzata. Per natura intrinseca, non è fisicamente possibile impedire il funzionamento di una struttura con tali caratteristiche. Come *Internet* negli anni '90, la blockchain riscontra ancora molta incertezza e scetticismo da parte delle persone.

La vera rivoluzione da attuare non riguarda esclusivamente l'aspetto tecnologico, ma soprattutto quello sociale. L'ostacolo principale da superare è collocato nella mente delle persone. Il giorno in cui la maggior parte della popolazione comprenderà i reali benefici apportati da un sistema decentralizzato, l'adozione delle relative tecnologie diverrà una semplice conseguenza naturale ed inevitabile.

"Bitcoin è l'inizio di qualcosa di grande: una moneta senza un governo, qualcosa di necessario e imperativo."
[Peter Thiel, co-fondatore di *Paypal*]

Capitolo 7 - Riferimenti bibliografici, sitografia e figure

Riferimenti bibliografici e sitografia

1. Paradosso del compleanno, Giulia Bon, 31 novembre 2021, https://www.thedifferentgroup.com/2021/11/23/paradosso-del-compleanno/
2. Binance Chain, https://www.binance.org/
3. Cardano, https://cardano.org/
4. Solana, https://solana.com/
5. Messari.io Ethereum, https://messari.io/asset/ethereum/metrics/
6. eWASM, https://ewasm.readthedocs.io/en/mkdocs/
7. WebAssembly, https://webassembly.org/
8. IOHK, https://iohk.io/
9. Ouroboros, https://cardano.org/ouroboros/
10. SundaeSwap, https://sundaeswap.finance/
11. Price and value (Cardano); Charles Hoskinson; 30 marzo 2021; https://www.youtube.com/watch?v=AGWZ2HBSI7E
12. Cardano Africa, https://africa.cardano.org/
13. Atala Prism, https://atalaprism.io/app
14. WETH, https://weth.io/it/
15. Processo di Markov, Carlo Consoli, 10 ottobre 2001, https://www.vialattea.net/content/37
16. Recupero chiave privata Bitcoin; Nils Schneider; 28 dicembre 2013; https://web.archive.org/web/20160308014317/http://www.nilsschneider.net/2013/01/28/recovering-bitcoin-private-keys.html
17. Trezor, https://trezor.io/
18. Joe Grand, https://twitter.com/joegrand
19. CoinMarketCap, https://coinmarketcap.com
20. Etherscan, https://etherscan.io/
21. Squid Game; Hwang Dong-hyuk; 17 settembre 2021; https://www.netflix.com/it/title/81040344
22. SQUID, https://bscscan.com/token/0x87230146E138d3F296a9a77e497A2A83012e9Bc5
23. SQUID rug pull; Sykogene; 1 novembre 2021; https://www.youtube.com/watch?v=FFRhm4CZpR0
24. Medium impatto ambientale; TQ Tezos, 16 marzo 2021; https://medium.com/tqtezos/proof-of-work-vs-proof-of-stake-the-ecological-footprint-c58029faee44
25. DW, https://www.dw.com/
26. Cambridge global cryptocurrency benchmarking study; Apolline Blandin, Dr. Gina Pieters, Yue Wu, Thomas Eisermann, Anton Dek, Sean Taylor, Damaris Njoki; settembre 2020; https://www.jbs.cam.ac.uk/wp-content/uploads/2021/01/2021-ccaf-3rd-global-cryptoasset-benchmarking-study.pdf,
27. FridaysForFutures impatto ambientale; 19 giugno 2021; https://fridaysforfutureitalia.it/criptomonete-e-nft-quanto-inquinano-davvero/
28. Bitcoin Whitepaper, *Satoshi Nakamoto*, 2009, https://bitcoiner.guide/bitcoin.pdf
29. Bitcoin glossario, https://bitcoin.org/it/glossario#indirizzo