



**ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΙΡΑΙΩΣ**

**ΣΧΟΛΗ ΤΕΧΝΟΛΟΓΙΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΕΠΙΚΟΙΝΩΝΙΩΝ  
ΤΜΗΜΑ ΨΗΦΙΑΚΩΝ ΣΥΣΤΗΜΑΤΩΝ**

**ΕΡΕΥΝΗΤΙΚΗ ΕΡΓΑΣΙΑ**

## **Τεχνολογίες Διασφάλισης Ιδιωτικότητας στο Blockchain και Web 3.0**

**Δεύτερη Έκδοση**

**Δημήτρης Βαγιακάκος Ε18019  
Σταύρος Γκίνος Ε18043  
Κωνσταντίνος Καραχάλης Ε18065**

**Επιβλέποντες Καθηγητές:**

**Κωνσταντίνος Λαμπρινουδάκης, Καθηγητής Πανεπιστημίου Πειραιώς  
Στέφανος Γκριτζαλής, Καθηγητής Πανεπιστημίου Πειραιώς**

**ΠΕΙΡΑΙΑΣ**

**ΜΑΙΟΣ 2022**



## **ΕΡΕΥΝΗΤΙΚΗ ΕΡΓΑΣΙΑ**

Τεχνολογίες Διασφάλισης Ιδιωτικότητας στο Blockchain και Web 3.0

Δεύτερη Έκδοση

**Δημήτρης Βαγιακάκος**

**A.M: E18019**

**Σταύρος Γκίνος**

**A.M: E18043**

**Κωνσταντίνος Καραχάλης**

**A.M: E18065**

## Περίληψη

Με την εξέλιξη της τεχνολογίας των Αποκεντρωμένων Δικτύων Blockchain και το ενδιαφέρον που παρουσιάζεται καθημερινά από πολλούς χρήστες στο διαδίκτυο μετά την άνοδο της τιμής που σημειώνουν τα κρυπτονομίσματα τον τελευταίο καιρό, μας έδωσε το έναυσμα να συντάξουμε μία ερευνητική εργασία στα Ελληνικά μιας και δεν υπάρχει αρκετό υλικό που να παρέχει αρκετές πληροφορίες, όσο το δυνατόν πιο κατανοητές για την τεχνολογία blockchain στα Ελληνικά. Έτσι, πραγματοποιήσαμε αυτή την ερευνητική εργασία, με σκοπό να κατηγοριοποιήσουμε και να εξετάσουμε πως οι κυριότερες τεχνολογίες που χρησιμοποιούνται στο Blockchain εξασφαλίζουν την ιδιωτικότητα των χρηστών για την επερχόμενη νέα έκδοση του διαδικτύου (Web3.0).

**ΘΕΜΑΤΙΚΗ ΠΕΡΙΟΧΗ:** Δίκτυα Blockchain

**ΛΕΞΕΙΣ ΚΛΕΙΔΙΑ:**

Blockchain, Web3.0, Smart-Contracts, Decentralized-Applications, Bitcoin, Ethereum, Monero, Oracles, Cardano, Zcash, Zether

## Ευχαριστίες

Για τη διεκπεραίωση της παρούσας Ερευνητικής Εργασίας, θα θέλαμε να ευχαριστήσουμε τους καθηγητές μας του Πανεπιστημίου Πειραιώς, κ. Χρήστο Ξενάκη, κ. Κωνσταντίνο Λαμπρινουδάκη και κ. Στέφανο Γκρίτζαλη για τα ερεθίσματα που μας παρείχαν έτσι ώστε να ασχοληθούμε με την τεχνολογία Blockchain καθώς και με την ανάπτυξη smart-contracts αλλά και την ασφάλεια και ιδιωτικότητά τους.

## Περιεχόμενα

1. Εισαγωγή .....	7
1.1: Τι είναι το Blockchain .....	7
2. Τύποι Δικτύων Blockchain.....	9
2.1 Public Blockchains: .....	9
2.2 Private/Permissioned Blockchains: .....	9
2.3 Consortium Blockchain: .....	9
3. Ψηφιακά Πορτοφόλια .....	10
3.1 Hardware Wallets .....	10
3.2 Paper Wallets .....	12
3.3 Software Wallets .....	13
3.3.1 Full Node Wallet .....	13
3.3.2 Lightweight Node Wallet.....	14
3.3.3 Online Wallets .....	15
3.3.4 Web Wallets .....	16
4. Βασικοί μηχανισμοί του Blockchain .....	17
4.1 Consensus Mechanisms.....	17
4.1.1 Proof Of Work .....	17
4.1.2 Proof Of Stake.....	17
4.2 Μοντέλα Συναλλαγών στο Blockchain .....	17
4.2.1 UTXO-Unspent Transaction Output .....	17
4.2.2 ABOT-Account Based Online Transaction .....	18
5. Ψηφιακές Υπογραφές.....	19
5.1 Γενική ιδέα τρόπου λειτουργίας.....	19
5.2 Τύποι Υπογραφών.....	20
5.2.1 Aggregate Signature .....	20
5.2.2 Ring Signature .....	20
5.2.3 Blind Signature .....	21
6. Zero Knowledge Proof.....	22
6.1 Zero-Knowledge Proof Κριτήρια .....	22
6.2 Παράδειγμα για το Zero Knowledge Proof .....	22
6.3 Non-Interactive Zero-Knowledge Proof (NIZKP) .....	23
6.4 Σύνοψη ZKP VS NIZKP .....	25
7. Bitcoin.....	26
7.1 Mixers .....	27
7.1.1 Mixcoin .....	27
7.1.2 CoinJoin .....	28

8. Η λύση του κρυπτονομίσματος Monero.....	28
8.1 Stealth Addresses.....	28
8.2 Τι είναι το Monero; .....	28
8.3 Τρόπος λειτουργίας των κλειδιών στο Monero .....	29
8.4 Double Spending σε Monero .....	29
8.5 Transaction Input στο Monero .....	30
9. Web 3.0 .....	30
9.1 Ethereum .....	30
9.1 Από τα Αποκεντρωμένα Ψηφιακά Νομίσματα, στις Αποκεντρωμένες Εφαρμογές.....	30
9.2 Blockchain Oracles .....	34
9.3 Unstoppable Domains.....	35
10. Η λύση του κρυπτονομίσματος Zcash .....	36
11. Ιδιωτικότητα στα Smart-Contracts .....	38
11.1 Συναρτήσεις και μεταβλητές:.....	39
11.2 Private μεταβλητές .....	40
11.3 Internal μεταβλητές .....	42
11.4 Τεχνολογίες Διασφάλισης της Ιδιωτικότητας στα Smart-Contracts .....	43
11.4.1 Η λύση του zkay .....	43
11.4.2 Η λύση του Kachina .....	44
12. Αποκεντρωμένα Αναγνωριστικά.....	45
13. Προστασία της ιδιωτικότητας των συναλλαγών στο Ethereum.....	48
13.1 Η λύση του smart-contract Zether.....	48
14. Συμπεράσματα .....	50
15. Πίνακας Ορολογίας.....	51
16. Συντμήσεις – Αρκτικόλεξα – Ακρώνυμα.....	52
17. Βιβλιογραφικές αναφορές.....	53

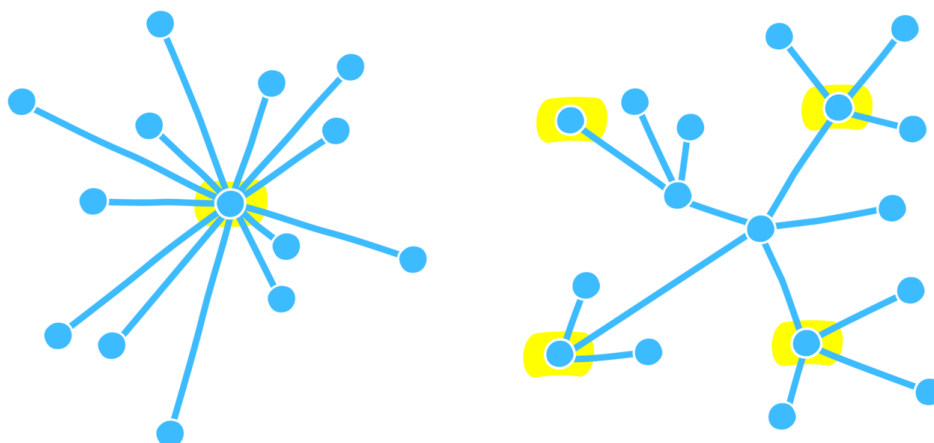
# 1. Εισαγωγή

## 1.1: Τι είναι το Blockchain

Το Blockchain είναι μία Peer-to-Peer (P2P) τεχνολογία όπου κρατάει μητρώο συναλλαγών, χωρίς κάποιον υπεύθυνο να το ελέγχει, ούτως ώστε όλες οι συναλλαγές-πληροφορίες να είναι διαθέσιμες σε όλους τους χρήστες πάνω στο ίδιο δίκτυο. Το μητρώο συναλλαγών(ledger) είναι αποθηκευμένο σε κάθε κόμβο του δικτύου (ή αλλιώς node) , είναι ένα αποκεντρωμένο σύστημα, σε αντίθεση με το κλασσικό κεντροποιημένο σύστημα, όπου το ledger είναι αποθηκευμένο π.χ. στον κεντρικό server μιας τράπεζας.

Το Blockchain αποτελεί μία επαναστατική τεχνολογία, αφού δίνει την δυνατότητα να στηθούν αποκεντρωμένες υπηρεσίες, δεν μπορούν να τροποποιηθούν από τρίτους πράγμα που καθιστά το δίκτυο αξιόπιστο από ενδεχόμενες τροποποιήσεις τρίτων όπου θα μπορούσαν να συμβούν στα παραδοσιακά κεντροποιημένα δίκτυα.

Σε σχέση με τις κεντροποιημένες υπηρεσίες όπου όλες οι πληροφορίες των συναλλαγών που λαμβάνουν χώρα μεταξύ δύο ή περισσότερων χρηστών να συσσωρεύονται σε ένα κεντρικό server τον οποίο τον διαχειρίζονται κάποια συγκεκριμένα άτομα που είναι αρμόδια όπως πχ σε τράπεζες.

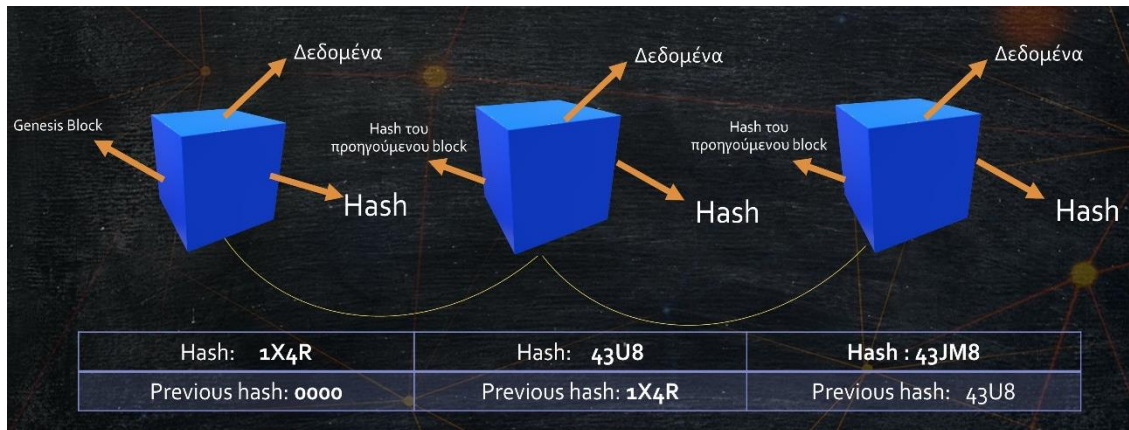


*Εικόνα 1: Η δομή ενός Κεντροποιημένου Δίκτυου και από την άλλη, η δομή ενός Αποκεντρωμένου Δικτύου.*



Στο Blockchain όπου αποτελεί ένα αποκεντρωμένο δίκτυο, το κάθε block του Blockchain αποτελείται από τα εξής κύρια συστατικά:

- ❖ Hash (συνήθως 256-bits)
- ❖ Hash του προηγούμενου block
- ❖ Τα δικά του δεδομένα



*Εικόνα 2: Βασική Δομή μίας αλυσίδας από Block (Blockchain)*

Μόλις δημιουργηθεί το block, υπολογίζεται το hash, το οποίο δεν είναι αμφίδρομο, δηλαδή (data -> hash(και όχι αντιστρόφως)). Το αρχικό block ενός δικτύου Blockchain ονομάζεται Genesis Block. Αν κάποιος προσπαθήσει να τροποποιήσει κάποιο block, θα αλλάξει το hash του και επομένως θα σπάσει η αλυσίδα και θα είναι άκυρα τα blocks. Σε περίπτωση κάποιου conflict (πχ. Αν κάποιος έχει ξοδέψει το ίδιο αριθμό κρυπτονομισμάτων σε δύο διαφορετικά άτομα στο δίκτυο), το Blockchain ακολουθεί πάντα την αλυσίδα με τα περισσότερα Block για να λυθεί το conflict.

## 2. Τύποι Δικτύων Blockchain

### 2.1 Public Blockchains:

Με τον όρο Public Blockchain, εννοούμε μία δημόσια αλυσίδα από blocks, δηλαδή ένα ανοικτό δίκτυο. Η πληροφορία που βρίσκεται σε αυτό είναι δημόσια για όλους. Ο καθένας έχει την δυνατότητα να δει, να διαβάσει και να γράψει δεδομένα σε ένα Public Blockchain. Τα δεδομένα του είναι προσβάσιμα από όλους. Ωστόσο, αυτό δεν σημαίνει ότι κάποιος μπορεί να επηρεάσει τα δεδομένα του Blockchain. Έτσι, το δίκτυο παραμένει αποκεντρωμένο μιας και ο καθένας ανώνυμα μπορεί να παράγει blocks, χωρίς κάποιος τρίτος όπως μία εταιρεία ή κάποιος οργανισμός να μπορεί να το ελέγξει. Τα Public Blockchains είναι αποκεντρωμένα και παρέχουν ασφάλεια. Αυτό σημαίνει ότι από την στιγμή που κάποια πληροφορία περάσει μέσα στο δίκτυο, δεν μπορεί να τροποποιηθεί ή να διαγραφεί, από την στιγμή που η είσοδος έχει γίνει validated στο δίκτυο του Blockchain. Σε τέτοιου τύπου Blockchain, ο καθένας μπορεί να συμμετέχει στο δίκτυο και να επαληθεύει συναλλαγές. Public Blockchain αποτελούν Blockchain Δίκτυα όπως το Bitcoin, Ethereum, Cardano.

### 2.2 Private/Permissioned Blockchains:

Σε αντίθεση με τα Public Blockchains, τα Private ή Permissioned Blockchain έχουν περιορισμούς στην συμμετοχή στο δίκτυο, καθώς και στην ανάγνωση ή στην εγγραφή των transactions μέσα στο δίκτυο. Δεν μπορεί ο καθένας να κατεβάσει ένα αντίγραφο του Blockchain και να ξεκινήσει να παράγει ή και να επιβεβαιώνει blocks. Σε αυτή την διάταξη, κεντρικές αρχές (π.χ μία εταιρεία) ελέγχουν τα μέλη/κόμβους του δικτύου ως προς τα δικαιώματα που έχουν πάνω στο συγκεκριμένο Blockchain. Αυτή η στρατηγική, διασφαλίζει την αξιοπιστία των κόμβων, ωστόσο το δίκτυο είναι κεντροποιημένο. Ο πιο γνωστός τύπος Private Blockchain είναι το Hyperledger.

### 2.3 Consortium Blockchain:

Ένας ακόμη τύπος Blockchain είναι το Consortium Blockchain, είναι μεταξύ private και Public Blockchain, δηλαδή το δίκτυο του Consortium Blockchain είναι ημί-ιδιωτικό blockchain καθώς μόνο προεπιλεγμένοι συμμετέχοντες-χρήστες έχουν πρόσβαση σε αυτό και άρα δεν είναι προσβάσιμο σε όλους αλλά μόνο στους παραπάνω χρήστες. Αυτός ο τύπος είναι απαραίτητος όταν διαφορετικές εταιρίες χρησιμοποιούν το ίδιο Blockchain. Επιπροσθέτως, αυτού του είδους blockchain παρέχει το δικαίωμα ανάγνωσης και γραφής στις πολλαπλές επιχειρήσεις που ανήκουν στο δίκτυο του consortium blockchain. Το Quorum αποτελεί ένα τέτοιου είδους Blockchain.

### 3. Ψηφιακά Πορτοφόλια

Με σκοπό να επικοινωνήσουμε με ένα δίκτυο Blockchain, κρίνεται απαραίτητη η χρήση ψηφιακών πορτοφολιών. Το πορτοφόλι στο blockchain στην πραγματικότητα είναι ψηφιακό πορτοφόλι και χρησιμοποιείται κυρίως για την διαχείριση και αποθήκευση των κρυπτονομισμάτων των διάφορων δικτύων Blockchain καθώς και άλλων περιουσιακών στοιχείων που τρέχουν ως κάποιο Smart Contract στα Blockchain αυτά. Τα πορτοφόλια των διάφορων δικτύων blockchain αξιοποιούνται επίσης για διαφορές συναλλαγές σε κρυπτονομίσματα ή για σύνδεση και λειτουργία διαφόρων smart-contracts. Η δημιουργία ενός ψηφιακού πορτοφολιού είναι δωρεάν και δεν απαιτεί κάποια προσωπικά δεδομένα από τον χρήστη. Τα ψηφιακά πορτοφόλια χωρίζονται σε δύο κατηγορίες, σε hardware wallets ( Συχνά αναφέρονται και ως cold wallets) και σε software wallets ( Συχνά αναφέρονται και ως hot wallets). Από αυτές τις κατηγορίες, προκύπτουν διάφοροι τύποι πορτοφολιών όπου θα αναλυθούν στις επόμενες σελίδες.

#### 3.1 Hardware Wallets



*Εικόνα 3: Ledger Nano S, ένα από τα πιο δημοφιλή Hardware wallets*

Το hardware wallet αποτελεί μία συσκευή που έχει σχεδιαστεί ειδικά για την ασφαλή αποθήκευση ιδιωτικών κλειδιών. Σε σχέση με τα software wallets, θεωρούνται πιο ασφαλή διότι δεν συνδέονται στο διαδίκτυο οποιαδήποτε στιγμή. Εφόσον, δεν είναι συνδεδεμένα στο διαδίκτυο ανά πάσα στιγμή, η πιθανότητα υποκλοπής μειώνεται σημαντικά καθώς δεν έχουν την

δυνατότητα να κάνουν επίθεση στο hardware wallet. Τα hardware wallets κάνουν σίγουρο πως δεν πρόκειται τα ιδιωτικά κλειδιά να αφαιρεθούν από την συσκευή (hardware wallet) και αυτό επιτυγχάνεται με την φύλαξη αυτών σε συγκεκριμένο σημείο στην συσκευή. Πρέπει να σημειωθεί το γεγονός πως αφού δεν είναι συνδεδεμένα τα hardware wallet στο internet συνεχώς θα χρειαστεί να χρησιμοποιηθεί μαζί με ένα άλλο μηχάνημα-συσκευή που θα συνδέεται με το διαδίκτυο και ακόμα και αν η συγκεκριμένη συσκευή είναι μολυσμένη από κάποιο malware, δεν πρόκειται να υπάρξει ρίσκο υποκλοπής των ιδιωτικών κλειδιών λόγω του τρόπου με τον οποίο λειτουργούν τα hardware wallets.

Μέσα λοιπόν από την άλλη συσκευή έχει πλέον την δυνατότητα ο χρήστης να πραγματοποιήσει διάφορες συναλλαγές μέσω μιας εφαρμογής που θα τρέξει στην συσκευή που είναι συνδεδεμένη με το hardware wallet. Συγκεκριμένα, ο χρήστης αφού πρώτα δημιουργήσει την συναλλαγή θα πρέπει αρχικά να την στείλει στο hardware wallet (καθώς για να ολοκληρωθεί το transaction είναι απαραίτητο να γίνει signed από το ιδιωτικό κλειδί που υπάρχει μέσα στο hardware wallet). Όταν ο χρήστης επιβεβαιώσει πως τα στοιχεία της συναλλαγής είναι έγκυρα, λαμβάνει χώρα η υπογραφή της συναλλαγής με το ιδιωτικό κλειδί (όλα αυτά γίνονται μέσα στο hardware wallet) και έπειτα το hardware wallet αποστέλλει την συναλλαγή στην συσκευή με την οποία είναι συνδεδεμένο έτσι ώστε από εκεί να μεταδοθεί στο δίκτυο του συγκεκριμένου Blockchain.

Το κύριο αρνητικό χαρακτηριστικό των hardware wallets είναι το supply chain attack που όπως έχει αποδειχτεί μπορεί να γίνει αρκετά αποτελεσματική στην μείωση της αξίας ενός hardware wallet. Δηλαδή, η συγκεκριμένη επίθεση λαμβάνει χώρα όταν ένας malicious third user αποκτήσει την συσκευή hardware wallet πριν από τον χρήστη με σκοπό να «πειράξει» την συσκευή και να μειώσει την ασφάλεια της έτσι ώστε όποτε ο χρήστης πραγματοποιεί συναλλαγές να μπορούν να κλέβουν οτιδήποτε υπόλοιπο απομένει στο πορτοφόλι του (hardware wallet).

### 3.2 Paper Wallets



*Εικόνα 4: Template από Bitcoin Paper Wallet*

Το paper wallet αποτελεί ένα χαρτί στο οποίο μία crypto address και το ιδιωτικό της κλειδί είναι εκτυπωμένα σε μορφή QR code. Στην συνέχεια μπορούν να γίνουν scan οι συγκεκριμένοι κωδικοί για να γίνει η πραγματοποίηση συναλλαγών κρυπτονομισμάτων. Σε μερικές ιστοσελίδες επιτρέπεται στον χρήστη να κατεβάσει τον κωδικό του για την παραγωγή νέας διεύθυνσης και συνεπώς νέου private key ενώ είναι offline. Αυτός, είναι ο κυριότερος λόγος για τον οποίο ο συγκεκριμένος τύπος wallet είναι πολύ ασφαλής απέναντι σε επιθέσεις που μπορεί να γίνουν online από malicious third parties.

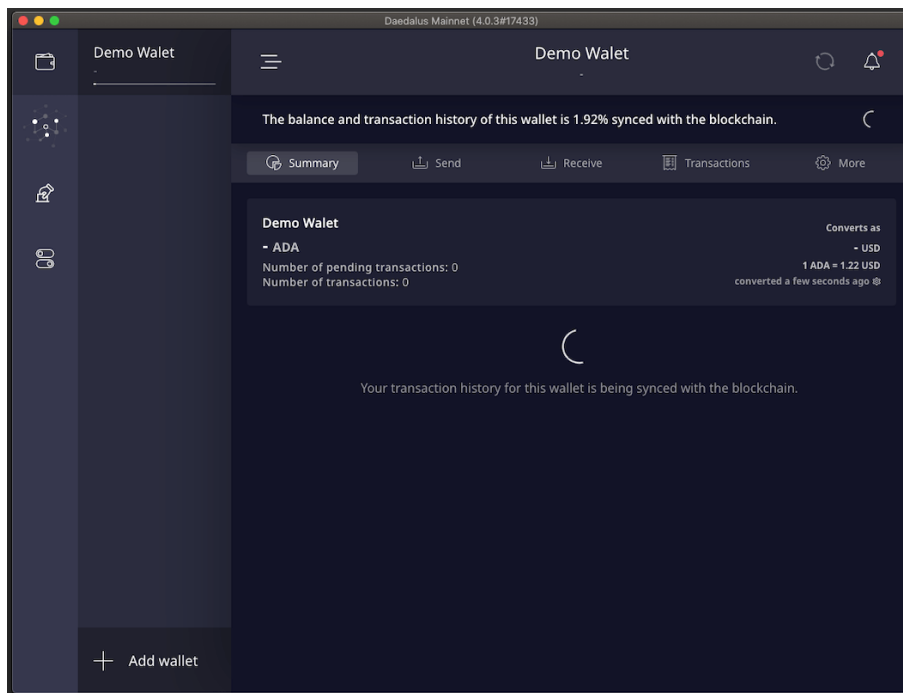
Παρόλα αυτά, τα paper wallets έχουν πολλά μειονεκτήματα όπως:

- ❖ Δεν είναι δυνατόν να στείλει ο χρήστης συγκεκριμένα ποσά αλλά πρέπει να στείλει ολόκληρο το υπόλοιπο που υπάρχει στο πορτοφόλι του. Για παράδειγμα, έστω πως ο χρήστης Σταύρος έχει δημιουργήσει ένα paper wallet και έχει σκοπό να πραγματοποιήσει διαδοχικές συναλλαγές για να βάλει bitcoins στο πορτοφόλι του, έστω συνολικά 26BTC. Αν τυχόν ξοδέψει 6BTC τότε θα αναγκαστεί να στείλει τα 26BTC σε ένα άλλο wallet και μετά να ξοδέψει τα 6BTC και εν τέλει να επιστρέψει τα υπόλοιπα 20BTC σε ένα καινούργιο paper wallet. Αξίζει να σημειωθεί πως, όταν γίνεται η αλλαγή του τύπου wallet που αναφέρθηκε παραπάνω θα πρέπει να χρησιμοποιηθεί το private key του paper wallet και να γίνει import σε ένα desktop wallet. Όμως τα υπόλοιπα bitcoins που μπορεί να προκύψουν από μία συναλλαγή αυτόματα στέλνονται σε μία νέα address που παράγεται από το bitcoin protocol. Άρα υπάρχει πιθανότητα, αν αυτή η διεύθυνση δεν ελέγχεται από τον χρήστη, να χάσει ο χρήστης όλα τα κρυπτονομίσματα που είχε μέσα στο paper wallet του.

- ❖ Ακριβώς επειδή το paper wallet είναι ένα χαρτί που έχει εκτυπωμένο πάνω του τους QR codes είναι πάρα πολύ εύκολο να φθαρεί και συνεπώς είναι πολύ εύκολο ένας χρήστης να χάσει το ποσό των κρυπτονομισμάτων όπου έχει αποθηκευμένα μέσα σε αυτό.

### 3.3 Software Wallets

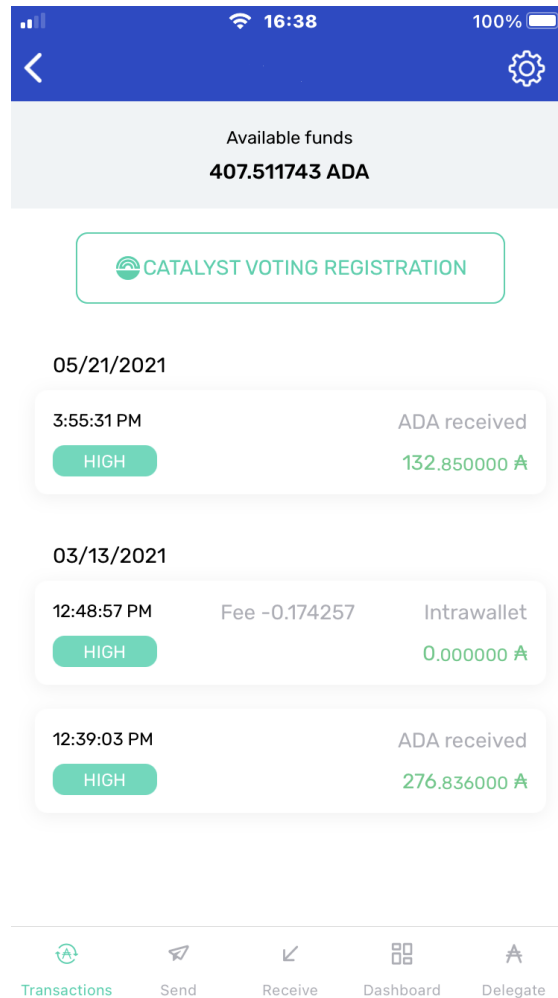
#### 3.3.1 Full Node Wallet



*Εικόνα 5: Daedalus Wallet. Ένα Full Node Wallet για το Blockchain του Cardano*

Ο συγκεκριμένος τύπος wallet του blockchain αξιοποιεί ένα πρόγραμμα που ονομάζεται full node. Τα Full Node wallets πρακτικά, κατεβάζουν τοπικά ένα ολόκληρο αντίγραφο του Blockchain. Έτσι, τρέχοντας οι ίδιοι ένα full node ως πορτοφόλι, αρχικά μπορούμε να είμαστε σίγουροι ότι το transaction μας μεταδόθηκε στο δίκτυο. Δεν χρειάζεται να εμπιστευόμαστε και να χρησιμοποιούμε το full node κάποιου άλλου (όπως θα δούμε παρακάτω, συμβαίνει στην περίπτωση του Lightweight Wallet) για να κάνει την δουλεία για εμάς. Μεταδίδουμε οι ίδιοι το transaction μας και είμαστε σίγουροι ότι το transaction θα φτάσει στους miners ή στους validators (Ανάλογα τι Consensus Mechanism χρησιμοποιεί το Blockchain που χρησιμοποιούμε), εξασφαλίζοντας έτσι στον χρήστη την αυτονομία και την ιδιωτικότητά του.

### 3.3.2 Lightweight Node Wallet

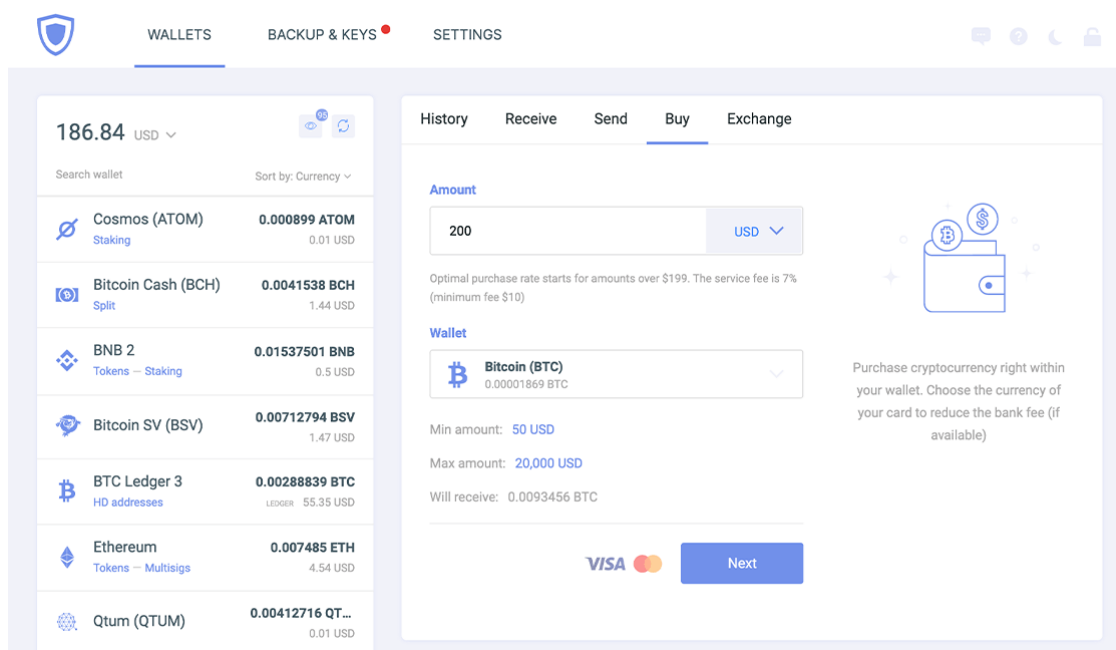


*Εικόνα 6: Yoroi Wallet, ένα Lightweight Node Wallet για το Blockchain του Cardano*

Σε αντίθεση με τον παραπάνω τύπο digital wallet, το lightweight node wallet δεν κατεβάζει ολόκληρο το blockchain, αλλά κατεβάζει αποκλειστικά τους headers των block ώστε να επαληθεύσει την εγκυρότητα των συναλλαγών. Τα Lightweight nodes συνδέονται σε full nodes έτσι ώστε να συνδεθούν στο ανάλογο δίκτυο Blockchain. Συνήθως, στέλνουν διευθύνσεις σε trusted third parties και δέχονται στην συνέχεια πληροφορίες για το wallet όπως είναι πχ το ιστορικό, το υπόλοιπο. Έτσι όμως, δεν μεταδίδουμε οι ίδιοι το transaction μας στο δίκτυο του Blockchain, αφού βασιζόμαστε σε κάποιον τρίτο και έτσι, δεν είμαστε σίγουροι ότι το transaction θα φτάσει στους miners ή στους validators (Ανάλογα το Blockchain που χρησιμοποιούμε).



### 3.3.3 Online Wallets



*Εικόνα 7: Guadra Online Wallet*

Το online wallet αποτελεί είναι μία υπηρεσία όπου αποθηκεύει τα κλειδιά του κρυπτονομίσματος ενός δικτύου Blockchain σε έναν (online) server, και σε σχέση με τους άλλους τύπους wallet είναι πιο εύκολα στην χρήση τους καθώς το μόνο που χρειάζεται ο χρήστης για να συνδεθεί σε αυτό από οποιαδήποτε συσκευή ή web browser είναι το username και το password του. Επιπρόσθετα, παρέχουν πολλά features που επιτρέπουν την αγοραπωλησία και ανταλλαγή cryptocurrencies.

Παρόλα αυτά, έχουν σημαντικά μειονεκτήματα ως προς την ιδιωτικότητα μιας και αποθηκεύονται όλα τα ιδιωτικά κλειδιά σε ένα κεντρικό server με συνέπεια να μοιάζουν με κεντροποιημένες υπηρεσίες όπως για παράδειγμα οι τράπεζες. Δηλαδή, τα κλειδιά αυτά περνάνε στα χέρια ενός τρίτου που τα διαχειρίζεται για εμάς. Ο κύριος κίνδυνος που φέρνει ένα Online wallet είναι ότι το Online Wallet όπου θα χρησιμοποιήσουμε, μπορεί να μην είναι αξιόπιστο και έτσι ο διαχειριστής του, κατέχοντας το private key μας, να μας κλέψει ο ίδιος τα κρυπτονομίσματά μας ή να πείσει ο ίδιος θύμα διαρροής δεδομένων και με αυτό τον τρόπο να διαρρεύσουν τα ιδιωτικά μας κλειδιά.

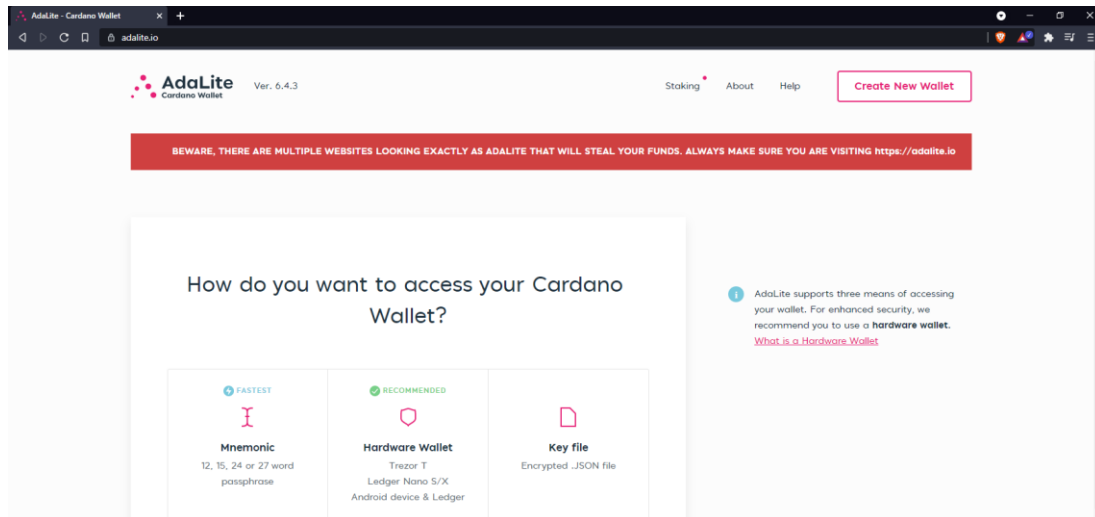
Για νέους χρήστες ωστόσο που δεν τους ενδιαφέρει ιδιαίτερα το θέμα της ιδιωτικότητας, αυτός ο τύπος πορτοφολιού, μιας και ο χρήστης δεν διαχειρίζεται ο ίδιος το πορτοφόλι του, μπορεί η υπηρεσία όπου του τα διαχειρίζεται να επαληθεύσει με κάποιον άλλο τρόπο (πχ. με έλεγχο ταυτότητας πολλών παραγόντων) και να επαναφέρει την πρόσβαση στο πορτοφόλι του, σε περίπτωση όπου ο κάτοχος ξεχάσει τον κωδικό του.

Αξίζει να αναφερθεί πως σε όλους τους υπόλοιπους τύπους πορτοφολιών που έχουμε αναφερθεί, σε περίπτωση όπου ο χρήστης χάσει τα ιδιωτικά του



κλειδιά ή ξεχάσει τον κωδικό των ιδιωτικών του κλειδιών, τα κρυπτονομίσματά του μένουν χαμένα για πάντα. Δεν γίνεται με κανέναν τρόπο να τα επαναφέρει με κάποιον τρόπο.

### 3.3.4 Web Wallets



*Εικόνα 8: AdaLite, Ένα Web Wallet για το Cardano*

Η πρόσβαση σε ένα web wallet γίνεται μέσω ενός web browser και το web wallet αποθηκεύει το πορτοφόλι του χρήστη σε ένα server. Συνήθως, ο χρήστης έχει την δυνατότητα να φτιάξει έναν νέο πορτοφόλι και να θέσει ένα δικό του secret key για να έχει πρόσβαση σε αυτό. Όμως, σε μερικές περιπτώσεις υπάρχουν service providers (specialized organizations) που κρατάνε και διαχειρίζονται τα ιδιωτικά κλειδιά του χρήστη σε έναν third party server. Αλλά είναι πιο ασφαλές ο χρήστης να έχει αυτός στην κατοχή του τα ιδιωτικά του κλειδιά για να μην παρέχεται πρόσβαση στην διαχείριση των χρημάτων του σε κάποιο third party.

## 4. Βασικοί μηχανισμοί του Blockchain

### 4.1 Consensus Mechanisms

#### 4.1.1 Proof Of Work

Στο Proof Of Work, το block προσπαθεί να μας αποδείξει ότι όντως δούλεψε. Για να προστεθεί ένα block στην αλυσίδα, οι miners πρέπει να λύσουν επιτυχώς ένα «puzzle», χρησιμοποιώντας την υπολογιστική τους ισχύ. Ο πρώτος miner που θα επιλύσει το puzzle, ανταμείβεται για την δουλειά του, ενώ οι υπόλοιποι δεν θα ανταμειφθούν. Όπως είναι κατανοητό, όσο περνάει ο καιρός, τόσο πιο δαπανηρή γίνεται η διαδικασία, κάνοντας εξαιρετικά δύσκολο το Mining χωρίς αρκετή υπολογιστική ισχύ, χαρακτηριστικό που αποτελεί και μειονέκτημα, μιας και σπαταλούνται εκατομμύρια κιλοβατώρες κάθε χρόνο. Για να καταφέρει κάποιος να επηρεάσει το αποτέλεσμα του δικτύου, πρέπει να ελέγχει πάνω από το 51% της ισχύς του δικτύου. Το Bitcoin, το Ethereum (στην έκδοση 1) και το Monero είναι παραδείγματα Blockchain όπου λειτουργούν χρησιμοποιώντας τον αλγόριθμο Proof Of Work.

#### 4.1.2 Proof Of Stake

Στο Proof Of Work, είδαμε πρακτικά ότι υπάρχει ένας διαγωνισμός, όπου αμείβει αποκλειστικά μόνο αυτόν όπου θα επιλύσει πρώτος το μαθηματικό παζλ. Στο Proof Of Stake, δεν υπάρχει διαγωνισμός στο ποιος θα επιλύσει το puzzle πρώτος, μιας και ο αλγόριθμος βασίζεται στο stake. Δηλαδή, ο αλγόριθμος επιλέγει ψευδοτυχαία έναν validator. Μόλις επιλεγθεί ένας Validator, έχει το αποκλειστικό δικαίωμα να παράξει αυτός το block. Έτσι, οι υπόλοιποι Validators δεν σπαταλούν ενέργεια να κάνουν την οποιαδήποτε πράξη μιας και δεν επιλέχθηκαν. Επιπλέον, αν ο Validator προσπαθήσει να κάνει κάτι ακατάλληλο στο δίκτυο, τότε θα χάσει το stake του. Έτσι, για να καταφέρει κάποιος να επηρεάσει το δίκτυο, πρέπει να έχει την κατοχή του το 51% του κρυπτονομίσματος του δικτύου, κάτι που είναι εξαιρετικά δύσκολο να συμβεί. Blockchains όπως του Cardano και του Ethereum 2.0 χρησιμοποιούν το Proof Of Stake.

### 4.2 Μοντέλα Συναλλαγών στο Blockchain

#### 4.2.1 UTXO-Unspent Transaction Output

Το μοντέλο του UTXO (Unspent Transaction Output) εμφανίστηκε για πρώτη φορά στο Bitcoin. Αυτό το μοντέλο φέρνει αρκετές ομοιότητες με το παραδοσιακό τραπεζικό σύστημα καταγραφής των συναλλαγών, δηλαδή, των ιδιοκτητών κάθε λογαριασμού καθώς και το υπόλοιπο του κάθε λογαριασμού. Τα UTXOs έχουν πρόσβαση συνεχόμενα και είναι υπεύθυνα για τον ορισμό της αρχής και του τέλους κάθε transaction. Το output των unspent transaction είναι το αποτέλεσμα του ποσού όπου ο χρήστης θα ξοδεύει και θα λαμβάνει στο μέλλον. Κάθε UTXO μπορεί να σταλθεί αποκλειστικά μία φορά. Αυτό σημαίνει ότι από την στιγμή που χρησιμοποιηθεί, δεν μπορεί να ξαναχρησιμοποιηθεί στο μέλλον. Το Validation της κάθε συναλλαγής είναι απαραίτητη για να διατηρηθεί

η ασφάλεια και η ιδιωτικότητα στο Blockchain. Στο UTXO, κάθε συναλλαγή θεωρείται έγκυρη αν τηρεί τις παρακάτω προϋποθέσεις:

- Κάθε αναφερόμενο input στην συναλλαγή, πρέπει να υπογράφεται από τον ιδιοκτήτη της και να μην ξοδεύεται ακόμη.
- Αν το transaction έχει πολλαπλά inputs, τότε κάθε ένα από τα inputs πρέπει να έχει υπογραφή όπου θα ταιριάζει στον κάθε ιδιοκτήτη του input.
- Ένα transaction είναι έγκυρο μόνο και μόνο αν το value του κάθε input του ισούνται ή υπερβαίνει το συνολικό value των inputs.

Τα προτερήματα της χρήσης του μοντέλου του UTXO στα Blockchain δίκτυα είναι τα εξής:

- Επεκτασιμότητα: Το UTXO επιτρέπει σε παράλληλα transactions να επεξεργαστούν πολλαπλά UTXOs ταυτόχρονα.
- Ιδιωτικότητα: Το UTXO μπορεί να διατηρήσει υψηλά ποσοστά ιδιωτικότητας, όσο χρησιμοποιεί σε κάθε transaction διαφορετική address.

#### 4.2.2 ABOT-Account Based Online Transaction

Το μοντέλο του ABOT (Account based Online Transaction) εμφανίστηκε για πρώτη φορά στο Ethereum. Σε σύγκριση με το UTXO, το ABOT αποτελεί ένα πιο απλό μοντέλο. Το ABOT λειτουργεί εξαντλητικά σε όλες τις συναλλαγές έτσι ώστε να βελτιώσει το consensus αποδοτικά καθώς και να επιτύχει γρηγορότερα το block, ωστόσο με το κόστος ενός υψηλότερου βαθμού κινδύνου.

Κάθε ένα transaction με κάποιος value στο token του, επαληθεύεται μόνο και μόνο αν:

- ❖ Το token είναι υπογεγραμμένο από τον αποστολέα.
- ❖ Η ιδιοκτησία του token του αποστολέα μπορεί να επαληθευτεί.
- ❖ Ο λογαριασμός του αποστολέα έχει επαρκή υπόλοιπο για να πληρωμή του transaction.

Έπειτα από την επικύρωση του transaction, γίνεται η χρέωση στον αποστολέα και η αξία πιστώνεται στον λογαριασμό του παραλήπτη. Ιδιαίτερα, στο δίκτυο του Ethereum, το υπόλοιπο του χρήστη αποτελεί ένα σύνολο από Ether (το κρυπτονόμισμα του δικτύου του Ethereum) όπου ο χρήστης διαθέτει το ιδιωτικό κλειδί έτσι ώστε να παραχθεί μία έγκυρη υπογραφή.

Τα προτερήματα της χρήσης του μοντέλου του ABOT στα Blockchain δίκτυα είναι τα εξής:

- Απλότητα: Το μοντέλο Account/Balance δεν εξαναγκάζει τα transactions να κρατάνε καταστάσεις, ενώ παράλληλα κρατάνε το μοντέλο απλό.
- Αποδοτικότητα: Το μοντέλο Account/Balance είναι αποδοτικό επειδή κάθε transaction χρειάζεται αποκλειστικά να επικυρώσει ότι ο λογαριασμός προς αποστολή έχει επαρκή υπόλοιπο στον λογαριασμό για την πληρωμή του Transaction.

## 5. Ψηφιακές Υπογραφές

Ένας από τους βασικότερους μηχανισμούς (τεχνολογίες) για την διασφάλιση της ιδιωτικότητας στο Blockchain αποτελεί η χρήση ψηφιακών υπογραφών. Αρχικά, η ψηφιακή υπογραφή είναι απαραίτητη στο blockchain καθώς για να μπορούν οι χρήστες μεταξύ τους να ανταλλάσσουν δεδομένα θα χρειαστεί να επαληθεύσουν πως τα δεδομένα που λαμβάνουν είναι από τον επιθυμητό χρήστη, δηλαδή πως η ακεραιότητα των δεδομένων θα είναι σίγουρη και συνεπώς, με την χρήση του κρυπτογραφικού αλγορίθμου εξασφαλίζεται η εγκυρότητα των δεδομένων => data privacy. Στις περισσότερες υλοποιήσεις, υπάρχει ένας αλγόριθμος παραγωγής κλειδιών που παράγει δύο κλειδιά, ένα private και ένα public key. Το public key παράγεται από το private key. Το public key αποτελεί την address του χρήστη. Παρακάτω γίνεται κατανοητή η αναγκαιότητα και η χρησιμότητα των κλειδιών αυτών.

### 5.1 Γενική ιδέα τρόπου λειτουργίας

Ως γενική εικόνα, ένα δίκτυο blockchain δημιουργεί hashes, που στην πραγματικότητα πρόκειται για μία αλληλουχία από αριθμούς κωδικών(σειρά αλφαριθμητικών string πχ δεδομένα -> \$aS0Rsfgm57kf90v3m45k#\$) που χρησιμοποιείται για την ταυτοποίηση δεδομένων. Δηλαδή, περνάνε τα δεδομένα μέσα από μία hash function και παράγεται ένα hash όπως στο παραπάνω παράδειγμα. Στην συνέχεια θα πρέπει ο χρήστης να χρησιμοποιήσει το ιδιωτικό του κλειδί για να κρυπτογραφηθεί το hash (digital signature).

Έστω, πως έχουμε 2 χρήστες που θέλουν να πραγματοποιήσουν μία συναλλαγή. Ο 1<sup>ος</sup> χρήστης θα στείλει το αρχείο με την ψηφιακή υπογραφή στον 2<sup>ο</sup> χρήστη. Ο 2<sup>ος</sup> χρήστης για την αποκρυπτογράφηση της ψηφιακής υπογραφής θα χρησιμοποιήσει το public key του 1<sup>ου</sup> χρήστη από το οποίο παίρνουμε το hash του 1<sup>ου</sup> εγγράφου. Έπειτα, ο 2<sup>ος</sup> χρήστης με τον ίδιο αλγόριθμο hash στο αρχείο που έλαβε υπολογίζει την hash τιμή του εγγράφου που λήφθηκε και έπειτα με την σύγκριση των τιμών αυτών γίνεται η επαλήθευση ότι το έγγραφο που έλαβε ο 2<sup>ος</sup> χρήστης είναι όντως από τον 1<sup>ο</sup> χρήστη. Δηλαδή, αν η τιμή του 1<sup>ου</sup> hash και η τιμή του 2<sup>ου</sup> hash δεν είναι ίδιες τότε αυτό συνεπάγεται πως το έγγραφο που έλαβε ο 2<sup>ος</sup> χρήστης πιθανόν να μην είναι το έγγραφο που έστειλε ο 1<sup>ος</sup>. Διαφορετικά, επαληθεύεται πως το λαμβανόμενο έγγραφο είναι αυτό που έστειλε ο 1<sup>ος</sup> χρήστης.

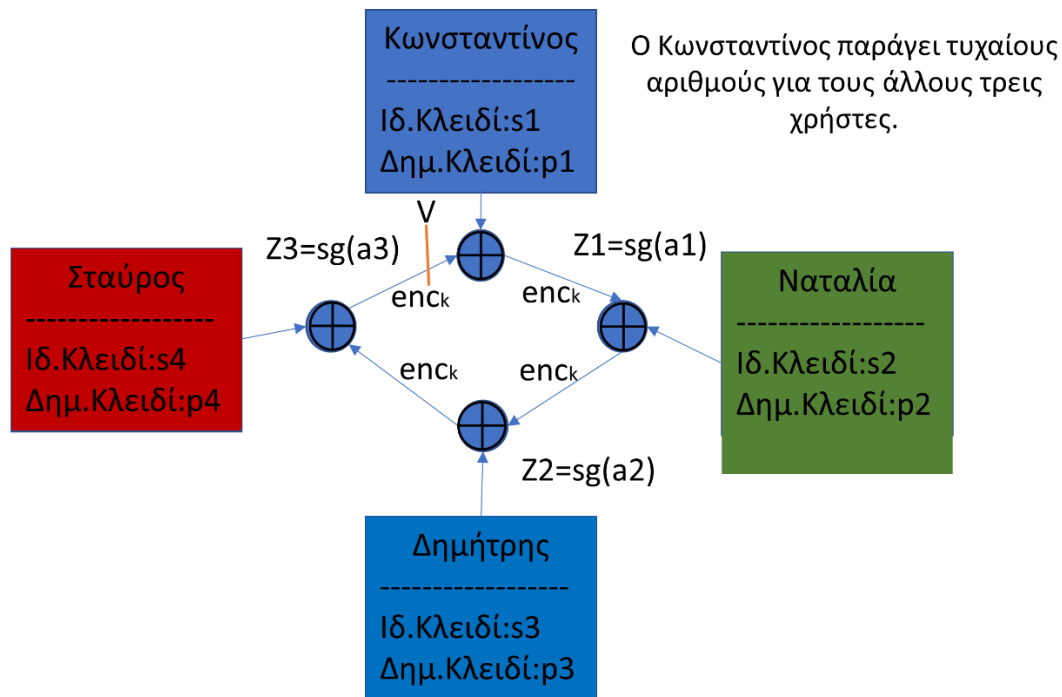
## 5.2 Τύποι Υπογραφών

### 5.2.1 Aggregate Signature

Ο συγκεκριμένος τύπος ψηφιακής υπογραφής αποτελεί ένα συνδυασμό από τέσσερις αλγόριθμους: *keygen*, *sign*, *combine* & *verify*. Οι δύο αρχικοί αλγόριθμοι λειτουργούν ακριβώς με τον ίδιο τρόπο που λειτουργεί μία ψηφιακή υπογραφή και από αυτούς τους δύο αλγορίθμους παίρνουμε την ψηφιακή υπογραφή *sk*. Ο τρίτος αλγόριθμος δέχεται ως είσοδο ένα διάνυσμα με *n* τριπλάσια όπου κάθε ένα από αυτά αποτελείται από ένα δημόσιο κλειδί (*public key ak*), ένα μήνυμα *mesk* και μία υπογραφή *sk*. Ο αλγόριθμος *combine* επομένως παράγει ως αποτέλεσμα μια υπογραφή *s1* για όλα τα μηνύματα (*mes1, ..., mesn*) και αποτελεί την *aggregate signature* που έχει ίδιο μήκος με την υπογραφή ενός μόνο μηνύματος. Ο τελευταίος αλγόριθμος που δέχεται ως είσοδο το δημόσιο κλειδί *ak* και *mesk* μήνυμα. Ως στόχο έχει την επαλήθευση της υπογραφής που παράχθηκε δηλαδή να επιβεβαιώσει αν η υπογραφή που παράχθηκε αποτελεί *aggregation* πολλών *n* έγκυρων υπογραφών. Σκοπός του συγκεκριμένου τύπου είναι η συνάθροιση πολλών ψηφιακών υπογραφών από πολλαπλά διαφορετικά δεδομένα σε μία όσο το δυνατόν μικρότερη συγκεντρωτική υπογραφή. Αυτό, θα έχει ως αποτέλεσμα την μείωση του φόρτου πάνω σε πόρους κυρίως υπολογιστικούς καθώς και για το χώρο των υπογραφών με αποτέλεσμα να αξιοποιούνται λιγότεροι πόροι για την απόπειρα εξασφάλισης ιδιωτικότητας δεδομένων (συναλλαγών).

### 5.2.2 Ring Signature

Με τον παραδοσιακό τρόπο της ψηφιακής υπογραφής, ξέρουμε και εμπιστευόμαστε ένα συγκεκριμένο άτομο, γνωρίζουμε δηλαδή κάποια στοιχεία για αυτό. Ειδικά αν πάρουμε ως επιλογή την απλή μας υπογραφή, θα μπορούσε κανείς να πει ότι όποιος γνωρίζει την υπογραφή μας θα μπορούσε να πει ότι αυτή η υπογραφή ανήκει σε εμάς, κάτι που μειώνει την ιδιωτικότητα μας. Εδώ έρχονται οι *Ring Signatures*. Οι *Ring Signatures*, παίρνουν πολλαπλές Ψηφιακές Υπογραφές από διαφορετικά τυχαία άτομα και τις συνδυάζουν έτσι ώστε να παραχθεί μία μοναδική υπογραφή. Κανείς δεν μπορεί να ανακαλύψει σε ποιόν ανήκουν αυτές οι υπογραφές. Δηλαδή, αν είχαμε 4 διαφορετικά άτομα, θα συνδυάζοντουσαν οι υπογραφές αυτών των 4 ατόμων και θα προέκυπτε μία ενιαία, χωρίς να είναι δυνατό να δει κανείς ποιοι βρίσκονται πίσω από αυτή την ενιαία υπογραφή. Έτσι αυξάνεται η ιδιωτικότητα των ατόμων. Το *Monero* χρησιμοποιεί *Ring Signatures* για τις συναλλαγές του.



Εικόνα 9: Η δομή ενός ring signature μεταξύ πολλαπλών χρηστών

### 5.2.3 Blind Signature

Βασίζεται πάνω σε αλγορίθμους επίλυσης προβλημάτων διακριτών μαθηματικών, ελλειπτικής καμπύλης. Συνήθως, όταν πραγματοποιείται η χρήση αυτής της ψηφιακής υπογραφής (blind) τα δεδομένα του μηνύματος αποκρύπτονται, και έπειτα υπογράφονται χωρίς ωστόσο να αποκαλύπτεται η πληροφορία όπου εμπεριέχεται στα μηνύματα! Εφόσον ο στόχος είναι η εξασφάλιση της ιδιωτικότητας της πληροφορίας του μηνύματος κατά την μετάδοσή του, χρησιμοποιείται σε καταστάσεις όπου ο συντάκτης του μηνύματος και αυτός που έκανε την υπογραφή θεωρούνται διαφορετικοί και όχι το ίδιο άτομο. Για παράδειγμα, σε εφαρμογές-τεχνολογίες που χρησιμοποιούν blockchain αξιοποιείται η blind digital signature για συστήματα εκλογών που έχουν κρυπτογραφηθεί, για εφαρμογές τραπεζικές.

Οπωσδήποτε, για να γίνει η έκδοση του μηνύματος είναι αναγκαίο να έχει γίνει επαλήθευση πως δεν μπορεί να ανιχνευθεί το μήνυμα από έναν κακόβουλο χρήστη και πως αυτός που υπέγραψε το μήνυμα να μην μπορεί να φανερωθεί. Δηλαδή, να μην γνωρίζει το περιεχόμενο του μηνύματος που υπέγραψε εξού και ονομάζεται τυφλή ψηφιακή υπογραφή.



## 6. Zero Knowledge Proof

### 6.1 Zero-Knowledge Proof Κριτήρια

- ❖ 1<sup>st</sup> Correctness (**Prover** και **Verifier** θεωρούμε ότι είναι έντιμοι)
  - ❖ 2<sup>nd</sup> Soundness (Ο **Prover** πρέπει να γνωρίζει το μυστικό καθώς δεν θα μπορεί να αποδείξει έναν ισχυρισμό για το μυστικό αυτό)
  - ❖ 3<sup>rd</sup> True Zero-Knowledge (Μετά το τέλος της διαδικασίας του zero-knowledge πρωτοκόλλου ο **Verifier** δεν θα πρέπει να μάθει καμία νέα πληροφορία πέρα της πληροφορίας του **Prover** που αποδεικνύει τον ισχυρισμό του)
- Web3.0 Applications → Zero-Knowledge proof
  - With Privacy → More data sharing
  - Prover, Verifier

### 6.2 Παράδειγμα για το Zero Knowledge Proof



Ας υποθέσουμε ότι έχουμε την παραπάνω εικόνα και μια εταιρεία θέλει να βρει που βρίσκεται ο Δημήτρης στην συγκεκριμένη εικόνα, όμως ο αλγόριθμος αναζήτησης που χρησιμοποιεί η εταιρεία δεν είναι ικανός να βρει τον Δημήτρη οπότε η εταιρεία απευθύνεται σε έναν εξωτερικό συνεργάτη. Σε αυτό το σημείο, η εταιρεία θα ήθελε να βεβαιωθεί ότι ο αλγόριθμος που χρησιμοποιεί ο εξωτερικός συνεργάτης λειτουργεί σωστά και βρίσκει τον Δημήτρη και παράλληλα ο εξωτερικός συνεργάτης θέλει να αποδείξει ότι μπορεί να βρει τον Δημήτρη χωρίς όμως να δώσει παραπάνω πληροφορίες πέρα από το γεγονός ότι μπορεί να βρει το ζητούμενο.

Οπότε ο εξωτερικός συνεργάτης στέλνει στην εταιρεία το εξής :



Αυτή αποτελεί η εικόνα του Δημήτρη που η εταιρεία θέλει να βρει, όμως ο εξωτερικός συνεργάτης έχει αποκρύψει το περιβάλλον γύρω-γύρω από τον Δημήτρη ώστε να μην προδώσει που βρίσκεται μέσα στην εικόνα αλλά παράλληλα αποδεικνύει στην εταιρεία ότι έχει καταφέρει να βρει τον Δημήτρη χωρίς όμως να φανερώνει παραπάνω πληροφορίες πέρα από αυτές που είναι αναγκαίες για να ικανοποιηθούν το ζητούμενο.

### 6.3 Non-Interactive Zero-Knowledge Proof (NIZKP)

Η κλασσική λειτουργία του ZKP όμως δεν μπορεί να είναι αποδοτική σε τεχνολογίες όπως είναι το IoT (Internet Of Things) όπου σε αυτή την περίπτωση υπάρχουν για παράδειγμα τα “έξυπνα” οχήματα τα οποία κινούνται σε μεγάλη ταχύτητα οπότε η ανταλλαγή τεράστιου όγκου μηνυμάτων είναι ακατόρθωτη καθώς θα υπάρχουν πιθανές αποτυχίες σύνδεσης του καναλιού επικοινωνίας. Για να λυθεί αυτό το θέμα δημιουργήθηκε το NIZKP. Στο NIZKP όλα τα μηνύματα challenges ενός κλασσικού ZKP είναι συμπυκνωμένα σε ένα μόνο πακέτο το οποίο στέλνεται σε ένα μόνο μήνυμα και για αυτό τον λόγο το NIZKP είναι πιο αποδοτικό και χρειάζεται λιγότερο χρόνο για να πραγματοποιηθεί.

Το NON - Interactive Zero Proof Knowledge οριζόταν προηγουμένως μόνο ως ένα σύστημα απόδειξης ενός θεωρήματος. Κάθε απόδειξη σε ένα τέτοιο σύστημα απαιτεί το δικό της reference string, το οποίο δεν αποτελεί ένα τυχαίο string. Θα μπορούσε, για παράδειγμα, να αποτελείται από τυχαία στοιχεία ομάδων που χρησιμοποιούνται από όλα τα μέρη του πρωτοκόλλου. Τα στοιχεία των ομάδων είναι τυχαία αλλά αυτό δεν ισχύει για το reference string καθώς περιέχει δομή (π.χ. τα στοιχεία της ομάδας) η οποία διακρίνεται από τυχειότητα. Έπειτα από αυτό το γεγονός, οι Feige, Lapidot και Shamir πρότειναν το Multi-theorem zero-knowledge proofs ως μια πιο ευέλικτη έννοια για τα Non-Interactive Zero Knowledge Proofs.

Τα NON - Interactive Zero Knowledge Proof είναι ZKPs που δεν απαιτούν αλληλεπίδραση μεταξύ του prover και του verifier. Πρόσφατα, αυτές οι κρυπτογραφικές τεχνικές έχουν προκαλέσει το ενδιαφέρον σε εφαρμογές Blockchain, με την εμφάνιση εφαρμογών όπως τα NIZK, ZK-SNARK, ZK-STARK.

Προηγουμένως, τα Zero Knowledge Proofs συστήματα επαλήθευσης ήταν interactive. Για να εκτελεστούν επιτυχώς οι πράξεις, ο proofer και ο witness ή



verifier, έπρεπε να είναι ταυτοχρόνως σε απευθείας σύνδεση για να είναι εφικτή η επαλήθευση.

Ως αποτέλεσμα, η όλη διαδικασία κατέστη δύσκολη και μη κλιμακούμενη. Οι Fiat και Shamir ανέπτυξαν την ευριστική Fiat-Shamir το 1986, η οποία μετέτρεψε το Interactive Zero Knowledge Proof σε Non-Interactive Zero Knowledge Proof.

Η ευριστική των Fiat-Shamir είναι μια τεχνική για τη δημιουργία μιας ψηφιακής υπογραφής που βασίζεται πάνω στο Non-Interactive Proof Of Knowledge. Με αυτόν τον τρόπο, ένας prover ή ένα γεγονός μπορεί να επαληθευτεί δημοσίως χωρίς ο prover να είναι συνεχώς συνδεδεμένος στο διαδίκτυο.

Βηματική Περιγραφή του Non-Interactive Zero-Knowledge Proof με Discrete Αλγόριθμο:

- Ο Σταύρος θέλει να αποδείξει στον Δημήτρη ότι γνωρίζει τιμή τέτοια ώστε  $y = g^a$ .
- Ο Σταύρος επιλέγει τυχαία τιμή  $v$  από το σύνολο τιμών  $Z$  και υπολογίζει  $t = g^v$ .
- Ο Σταύρος υπολογίζει  $c = H(g, y, t)$  όπου  $H()$  είναι συνάρτηση κατακερματισμού.
- Ο Σταύρος υπολογίζει  $d = v - c \cdot a$ .
- Ο Δημήτρης ή οποιοσδήποτε άλλος μπορεί στη συνέχεια να ελέγξει αν  $t = g^d \cdot y^c$ .

Η ευριστική Fiat-Shamir μας επιτρέπει να αντικαταστήσουμε το interactive βήμα 3 με ένα μη interactive τυχαίο Oracle Access, αλλά στην πράξη χρησιμοποιείται μία hash συνάρτηση.

Στο Interactive ZKP, ο Δημήτρης θα επέλεγε μια τυχαία τιμή  $c$  από το  $Z$  και θα την έστελνε στον Σταύρο.

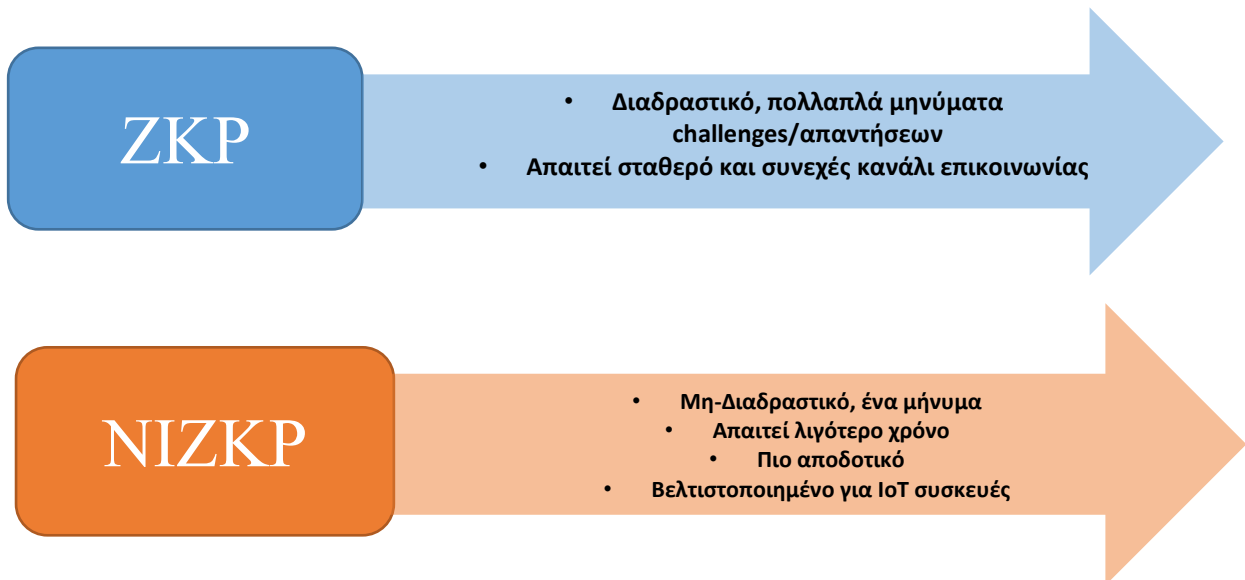
Εάν η τιμή κατακερματισμού που χρησιμοποιείται, δεν εξαρτάται από τη (δημόσια) τιμή του  $y$ , η ασφάλεια του συστήματος αποδυναμώνεται, καθώς ο κακόβουλος ελεγκτής μπορεί στη συνέχεια να επιλέξει συγκεκριμένη τιμή  $x$  έτσι ώστε το γινόμενο  $c \cdot a$  να είναι γνωστό.

#### Πλεονεκτήματα:

- **Επεκτασιμότητα:** Δεν απαιτεί ο prover ή ο verifier να είναι συνεχώς online έτσι ώστε να πραγματοποιηθεί ο έλεγχος.
- **Μεταβιβασιμότητα:** Εάν ο prover αποδείξει το proof of witness μία φορά, μπορεί να δημοσιοποιηθεί και η ίδια διαδικασία και δεν χρειάζεται να επαναληφθεί ξανά για διαφορετικό verifier.

Ως προς το Blockchain, ποίκιλες εφαρμογές είναι βασισμένες στο Non Interactive ZKP για να χρησιμοποιηθούν για την επαλήθευση συναλλαγών σε δημόσια Blockchain, ακόμη και αν οι πληροφορίες του αποστολέα, του παραλήπτη και της συναλλαγής παραμένουν ανώνυμες.

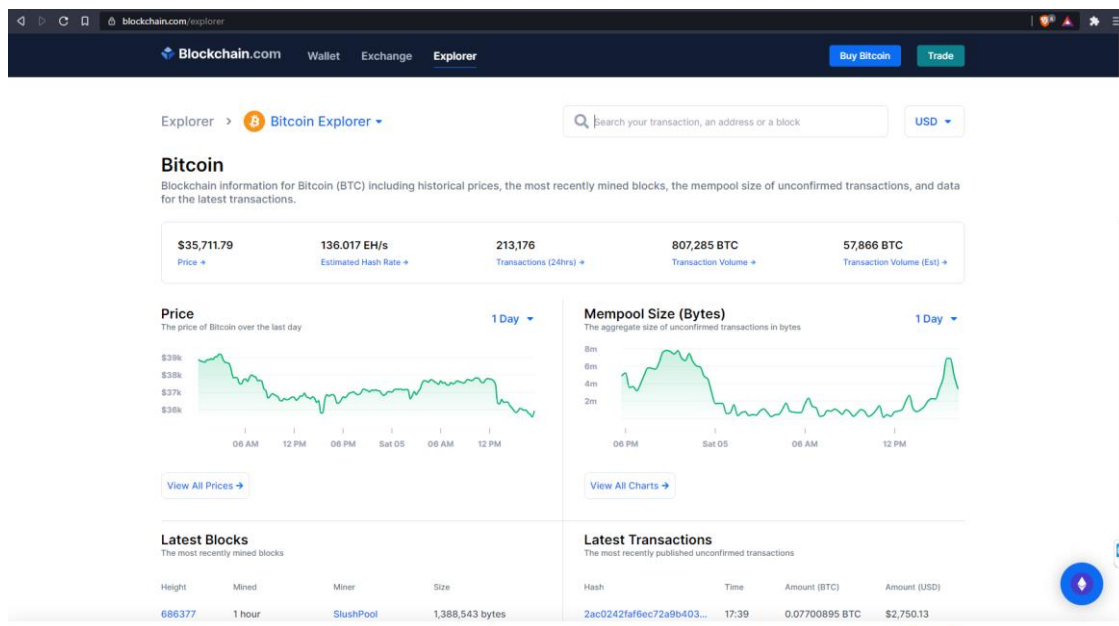
#### 6.4 Σύνοψη ZKP VS NIZKP



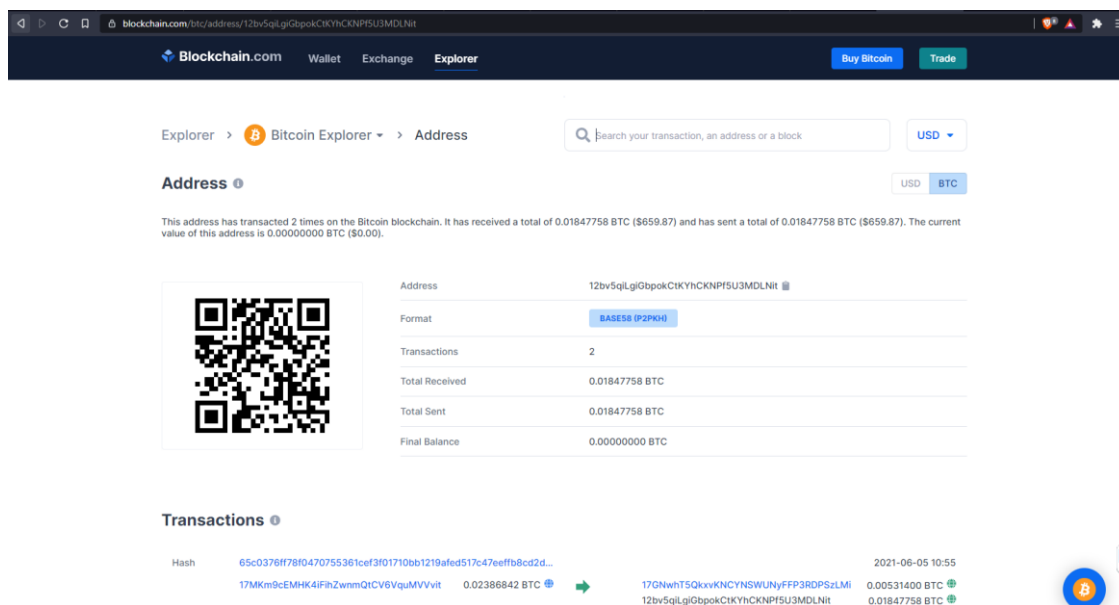
## 7. Bitcoin

Το Bitcoin αποτελεί το πρώτο Blockchain που φτιάχτηκε ποτέ. Το bitcoin εφευρέθηκε το 2008 από μία άγνωστη ομάδα χρησιμοποιώντας το ψευδώνυμο Satoshi Nakamoto και ξεκίνησε το 2009, όταν ο πηγαίος κώδικάς του δόθηκε στο διαδίκτυο ως ελεύθερο λογισμικό.

Ως δίκτυο, το Bitcoin αποτελεί ένα Public Blockchain, άρα ο καθένας μπορεί να διαβάσει το περιεχόμενό του δηλαδή ποιος έστειλε σε ποιόν καθώς και το ποσό των bitcoin που αποστάληκε σε ένα transaction.



*Εικόνα 10: Bitcoin Explorer. Παρουσίαση κάθε συναλλαγής και block στο δίκτυο του Blockchain του Bitcoin.*



*Εικόνα 11: Παράδειγμα transaction στο Bitcoin.*

Όπως μπορούμε να πούμε συμπερασματικά από τις παραπάνω εικόνες, το Bitcoin είναι ψευδο-ανώνυμο, μιας και οι διευθύνσεις φαίνονται δημόσια.

Στα πρώτα χρόνια λειτουργίας του δικτύου του Bitcoin, οι χρήστες αποκτούσαν τα bitcoins τους κυρίως εξορίζοντάς τα οι ίδιοι (Mining) . Αυτά τα bitcoins, κανείς δεν θα μπορούσε να ήξερε σε ποιον ανήκουν. Ωστόσο, όσο εξελισσόταν το δίκτυο και αυξανόταν η πολυπλοκότητα εξόρυξής του αλλά και η δημιουργία Mining Pools (Άτομα που ενώνουν τους εξοπλισμούς τους ώστε να έχουν μεγαλύτερη πιθανότητα να εξορύξουν οι ίδιοι κάποιο block έχοντας αθροιστικά μεγαλύτερη υπολογιστική ισχύ), το προσωπικό Mining έγινε ασύμφορο λόγω κόστους του εξοπλισμού και του ρεύματος που κατανάλωνε αυτό το hardware. Έτσι, ξεκίνησε η άνοδος των ανταλλακτηρίων, στα οποία μπορούσες να μετατρέψεις το εθνικό νόμισμά σου σε άλλα κρυπτονομίσματα. Ωστόσο σε συνδυασμό με το KYC (Know Your Customer) που έχει επιβάλει η νομοθεσία όπου έχει αναγκάσει τα ανταλλακτήρια να ζητάνε προσωπικά στοιχεία για τους πελάτες τους όπως αστυνομική ταυτότητα, διαβατήριο, δίπλωμα κλπ., μπορεί να γίνει συσχέτιση μεταξύ των addresses και αληθινών προσώπων. Έτσι, ξεκίνησαν να κάνουν την εμφάνισή τους υπηρεσίες όπου θα προσπαθήσουν να κάνουν τα bitcoins των χρηστών πραγματικά ανώνυμα.

## 7.1 Mixers

Οι υπηρεσίες Mixing (ή tumblers) έχουν σχεδιαστεί για να εμποδίζουν την παρακολούθηση των χρηστών μέσω της address του. Οι πλατφόρμες που προσφέρουν Mixing είναι ανταλλακτήρια όπου λαμβάνουν τα bitcoins των χρηστών και τα στέλνουν σε άλλους χρήστες. Ως αποτέλεσμα, για τον παρατηρητή του Blockchain Explorer του Bitcoin, είναι αδύνατον να διαπιστώσει σε ποιον ανήκουν τα bitcoins μιας και τα bitcoins είναι obfuscated.

Ωστόσο, αξίζει να επισημανθεί πώς αυτές οι υπηρεσίες mixing δεν προσφέρουν προστασία από τις κλοπές των bitcoins. Σε αυτή την φάση, θα περιγράψουμε 2 διάσημα mixing services και θα αναφερθούν οι δυνατότητές τους ως προς την ασφάλεια και την ιδιωτικότητα.

### 7.1.1 Mixcoin

Το Mixcoin είναι ένα ακόμη bitcoin mixing πρωτόκολλο που προσφέρει ανώνυμες πληρωμές σε κρυπτονομίσματα βασισμένα στο bitcoin. Για να ενισχύσει την ιδιωτικότητά των χρηστών του, το Mixcoin επιτρέπει στους χρήστες να στείλουν τα κρυπτονομίσματά τους σε έμπιστους τρίτους όπου θα αναλάβουν την διαδικασία του Mixing. Έπειτα, θα λάβουν πίσω το ίδιο ποσό που απέστειλαν. Αυτοί οι trusted third parties χρησιμοποιούν mixing servers, όπου για συντομία, αναφέρονται και ως mix. Έπειτα, τα mix φτιάχνουν νέες τυχαίες διευθύνσεις εμπλέκοντας τις και έπειτα, αποστέλλονται πίσω στον κάθε ένα κάτοχο τους. Έτσι, επιτυγχάνεται πραγματικά η ανωνυμία στα bitcoin transactions. Αξίζει να επισημανθεί πως το Mixcoin έχει προβλέψει και σε περίπτωση προβλήματος στον third party να κρατάει signed warranties στους συμμετέχοντες ώστε να μπορούν να κάνουν επαναφορά στα κρυπτονομίσματά τους οι κάτοχοι. Επίσης, το mixcoin επιτρέπει ανωνυμία και σε εξωτερικούς συμμετέχοντες, ωστόσο από αυτή την προσέγγισή, οι ενδιαφερόμενοι πρέπει να εμπιστεύονται το mix.

### 7.1.2 CoinJoin

Το CoinJoin ξεκίνησε στο 2013 ως μία εναλλακτική μέθοδος για την διεξαγωγή των bitcoin transactions όπου θα επιλύει το μειονέκτημα του MixCoin, συνδυάζοντας τα inputs από πολλαπλούς χρήστες σε μόλις ένα transaction, προστατεύοντας έτσι την ιδιωτικότητα των κατόχων τους, καθώς διεξάγονται οι συναλλαγές μεταξύ τους. Το CoinJoin, προσφέρει ανωνυμία χρησιμοποιώντας multi-signature transactions.

Τα Multi-Signature transactions απαιτούν την συμμετοχή από περισσότερα από έναν σέρβερ στο transaction. Στο CoinJoin, οι συμμετέχοντες κάνουν μίξη τα κρυπτονομίσματά τους, δημιουργώντας μόλις μια mixed συναλλαγή. Η συναλλαγή αυτή με τα πολλαπλά inputs θεωρείται έγκυρη αποκλειστικά αν και μόνο αν έχει γίνει signed από όλα τα κλειδιά που σχετίζονται με τις διευθύνσεις εισαγωγής. Επιπλέον, το CoinJoin προσφέρει εξωτερική unlinkability. Είναι μία διαδικασία από την οποία κανένα εξωτερικό party δεν μπορεί να προσδιορίσει σε ποιόν χρήστη ανήκει κάθε input. Ωστόσο, το μειονέκτημα του CoinJoin είναι ότι ένας από τους συμμετέχοντες στο mix, μπορεί να μάθει την διαδικασία και να συνδέσει την σύνδεση μεταξύ εσόδων και εξόδων με τις συναλλαγές.

## 8. Η λύση του κρυπτονομίσματος Monero

### 8.1 Stealth Addresses

Οι Stealth Addresses χρησιμοποιούνται σε συναλλαγές σε ένα δημόσιο Blockchain δίκτυο όπου ο αποστολέας και ο παραλήπτης θέλουν να διατηρήσουν την ιδιωτικότητα τους. Οι Stealth Addresses λειτουργούν δημιουργώντας μία μιας χρήσης διεύθυνση για κάθε transaction, ακόμα κι αν πραγματοποιούνται πολλαπλές συναλλαγές με τον ίδιο παραλήπτη. Έτσι, οι stealth addresses είναι χρήσιμες για την διασφάλιση του απορρήτου των παραληπτών. Ωστόσο, οι stealth addresses αντιμετωπίζουν πολλαπλά νομοθετικά προβλήματα από τις αρχές, μιας και μπορούν να χρησιμοποιηθούν για ανωνυμία σε παράνομες δραστηριότητες.

### 8.2 Τι είναι το Monero;

Το Monero (XMR) είναι ένα privacy-focused ανοικτού κώδικα κρυπτονομίσμα το οποίο κυκλοφόρησε το 2014. Το Monero είναι ένας τύπος Public Blockchain που χρησιμοποιεί obfuscated public ledger, δηλαδή, ο καθένας μπορεί να στείλει transactions ωστόσο κανένας εξωτερικός παρατηρητής δεν μπορεί να διαπιστώσει τον αποστολέα, τον αριθμό των Monero που στέλνονται καθώς και τον παραλήπτη. Το Monero χρησιμοποιεί κι αυτό όπως και το Bitcoin, το Proof Of Work ως μηχανισμό για να κρατάει το δίκτυο ασφαλή και για να κάνει validate τα transactions.

Η διαφορά του Monero σε σχέση με το Bitcoin είναι ως προς τις privacy-enhancing τεχνολογίες που χρησιμοποιεί το Monero.

Όπως ήδη αναφέρθηκε, το Monero έχει κατασκευαστεί ώστε να είναι privacy by default. Χρησιμοποιεί διαφορετικές τεχνολογίες για να επιτύχει την ανωνυμία και την ιδιωτικότητα των συναλλαγών και των ατόμων που περιπλέκονται σε αυτές.

Όλος ο σχεδιασμός του είναι έτσι ώστε να ικανοποιεί 2 βασικά κριτήρια: Untraceability και Unlinkability.

Στην περίπτωση του Untraceability, το Monero χρησιμοποιεί Ring Signatures έτσι ώστε να εμπλέκει πολλαπλούς πιθανούς αποστολείς σε ένα transaction. Το Unlinkability προστατεύει ότι ο παραλήπτης έχει stealth address.

### 8.3 Τρόπος λειτουργίας των κλειδιών στο Monero

Μία από τις πιο ενδιαφέρουσες λειτουργίες του Monero είναι ότι χρησιμοποιεί πολλαπλά κλειδιά. Στο Bitcoin, στο Ethereum ή/και στο Cardano, κάθε άτομο έχει το public key του καθώς και το private key του. Ωστόσο, στο Monero, τα πράγματα είναι λίγο πιο περίπλοκα.

Το Monero έχει το public view key του και το private view key.

1. Το Public View Key χρησιμοποιείται για να παραχθεί η μίας χρήσης stealth address, από την οποία τα Monero θα σταλούν στον παραλήπτη.
2. Το Private View key όπου χρησιμοποιείται από τον παραλήπτη για να αναζητήσει στο Blockchain του Monero για να βρει αν υπάρχει κάποιος αριθμός από Monero που στάλθηκε σε αυτόν.

Το Public View Key αποτελεί το πρώτο μέρος μίας Monero address. Έπειτα έχουμε τα Spend Keys, τα οποία έχουν να κάνουν με τον αποστολέα.

1. Το Public Spend Key θα βοηθήσει τον αποστολέα να είναι μέρος των Ring Transactions όπου τελικά θα κάνουν verify την υπογραφή από το key image.
2. Το Private Spend key όπου θα βοηθήσει στην δημιουργία του key image, το οποίο το ενεργοποιεί για να στείλει transactions.

Το Public Spend key αποτελεί το δεύτερο μέρος μίας Monero address. Τελικά, θα προκύψει μία 95 χαρακτήρων address η οποία έχει παραχθεί από το public spend και public view key.

### 8.4 Double Spending σε Monero

Το πρόβλημα του Double-Spending έχει περιγράψει ήδη από το Bitcoin, ωστόσο στην πλευρά του Monero έχει να αντιμετωπίσει και την ιδιαιτερότητα της ιδιωτικότητας στα transaction. Πώς το σύστημα θα κρίνει αν κάποιο Monero δεν έχει ήδη ξοδευτεί, μιας και όλα γίνονται κρυφά;

Την απάντηση στο πρόβλημα θα μας την δώσει πάλι η κρυπτογραφία.

Κάθε transaction στο Monero έχει το δικό του μοναδικό key image. Έτσι, αφού το κάθε key image είναι μοναδικό για κάθε transaction, οι Miners μπορούν απλά να ελέγξουν και να καταλάβουν αν ένα Monero έχει ξοδευτεί 2 φορές ή όχι. Με αυτό τον τρόπο το Monero επιτυγχάνει να ενισχύσει την ιδιωτικότητα του αποστολέα με την χρήση Ring Transaction.

## 8.5 Transaction Input στο Monero

Το Monero για να επιτύχει να διακινεί στο δίκτυο privacy enchanted transactions. Υποθέτοντας ότι η Alice θέλει να στείλει 1000 Monero στον Bob, το blockchain του Monero αυτόματα θα χρησιμοποιήσει Ring Signatures για να αποκρύψει την ταυτότητα της Alice.

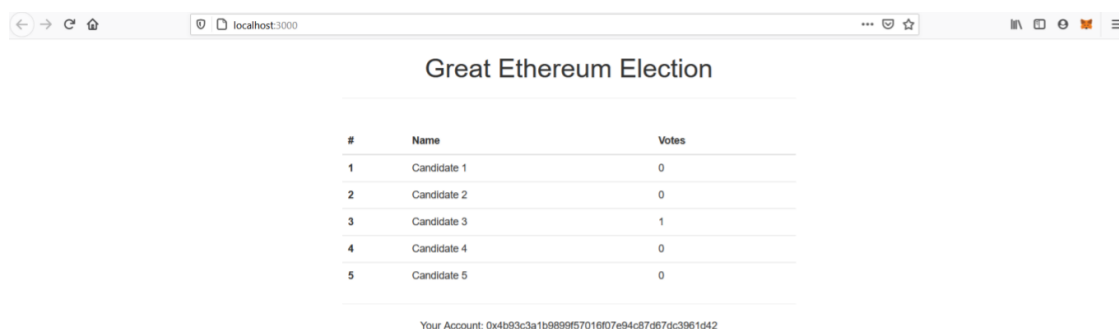
Αυτό θα προκύψει από το Ring size, το οποίο αποτελεί ένα τυχαίο output από το δίκτυο του Monero, το οποίο θα είναι ίδιο με την αξία από το output των 1000 Monero. Όσο πιο μεγάλο είναι το Ring Size, τόσο μεγαλύτερο θα είναι και το transaction μαζί με τα transaction fees. Η Alice τότε θα υπογράψει. Δεν χρειάζεται η Alice να ζητήσει από τους ιδιοκτήτες από τα προηγούμενα transactions την άδεια τους για να χρησιμοποιήσει τα outputs.

## 9. Web 3.0

### 9.1 Ethereum

Το Ethereum είναι ένα ανοικτού κώδικα (open-source) Blockchain, το οποίο αποτελεί μία αποκεντρωμένη δομή, η οποία είναι κάτι πολύ περισσότερο από απλά ένα αποκεντρωμένο ψηφιακό νόμισμα. Το Ethereum αποτελεί την δεύτερη γενιά των Blockchain δικτύων και πρακτικά πρόκειται για έναν αποκεντρωμένο υπολογιστή. Ως ένας αποκεντρωμένος υπολογιστής, ο οποιοσδήποτε χρήστης όπου χρειάζεται να τον χρησιμοποιήσει, πληρώνει για να δεσμεύσει αυτούς τους πόρους. Το Ethereum έχει δικιά του μνήμη αλλά και “δίσκο”, όπως δηλαδή θα είχε και ένας παραδοσιακός υπολογιστής. Τα αποκεντρωμένα προγράμματα που τρέχουν στο δίκτυο του Ethereum, ονομάζονται Smart-Contracts (έξυπνα συμβόλαια).

#### 9.1 Από τα Αποκεντρωμένα Ψηφιακά Νομίσματα, στις Αποκεντρωμένες Εφαρμογές



#	Name	Votes
1	Candidate 1	0
2	Candidate 2	0
3	Candidate 3	1
4	Candidate 4	0
5	Candidate 5	0

Your Account: 0x4b93ca1b989f5701607e94c87d679c3961d42

*Εικόνα 12: Μία αποκεντρωμένη εφαρμογή εκλογών που δουλεύει στο Blockchain του Ethereum*

Το σημαντικότερο χαρακτηριστικό των Smart Contracts είναι ότι είναι αμετάβλητα! Από την στιγμή που θα γίνουν deployed, ο κώδικας τους δεν μπορεί να τροποποιηθεί ή να αλλάξει. Τα Smart Contracts στο Ethereum, γράφονται



στην γλώσσα προγραμματισμού Solidity, η οποία έχει σχεδιαστεί για να τρέχει στο EVM (Ethereum Virtual Machine). Ο μόνος τρόπος για να αλλάξει ή να τροποποιηθεί ο κώδικας, είναι να γίνει ξανά deploy από την αρχή το smart contract με τις νέες αλλαγές. Επιπλέον, τα αποτελέσματα του smart contract, είναι ίδια για όλους τους χρήστες όπου το εκτελούν.

Το Ethereum έρχεται ως μία νέα έκδοση του Web, το Web 3.0. Στην πρώτη έκδοση του Web, οι χρήστες μπορούσαν μόνο να προβάλουν ιστοσελίδες στον φυλλομετρητή τους. Επικρατούσε μία στασιμότητα. Στην 2<sup>η</sup> έκδοση του Web, διάφορες υπηρεσίες έκαναν την εμφάνιση τους, κάνοντας έτσι το internet πιο διαδραστικό και δίνοντας την δυνατότητα σε χρήστες να αλληλεπιδρούν με αυτό, χωρίς την ανάγκη εξειδικευμένων γνώσεων σε θέματα πληροφορικής και δικτύων. Πλέον, ο χρήστης μπορεί και πραγματοποιεί αγορές μέσω του διαδικτύου, προβάλλει οπτικοακουστικό υλικό, συνομιλεί με τους φίλους του, οργανώθηκαν κοινωνικά δίκτυα κ.α. Όλες οι υπηρεσίες του διαδικτύου ωστόσο, βρίσκονται σε κάποιους κεντρικούς servers που διαχειρίζεται κάποιο πραγματικό ή νομικό πρόσωπο. Αυτό αυτόματα δημιουργεί προβλήματα στην ιδιωτικότητα, μιας και δεν υπάρχει διαφάνεια στο τι τρέχει από πίσω και αφού οι διάφορες υπηρεσίες με σκοπό το κέρδος, εκμεταλλεύονται τα cookies των υπηρεσιών τους, έτσι ώστε να είναι σε θέση να γνωρίζουν τα ενδιαφέροντα των χρηστών και να προωθούν στενευμένες διαφημίσεις ανάλογα των ενδιαφερόντων αυτών. Αυτή η χρήση των προσωπικών δεδομένων τελικά κατέληξε να χρησιμοποιηθεί ακόμη και για την χειραγώγηση των ψηφοφόρων των εκλογών των ΗΠΑ το 2016 από την Cambridge Analytica μέσω των δεδομένων που συλλέγονταν από το Facebook. Το Web2 όπου είναι κυρίως client-based έχει δείξει δεκάδες φορές τις αδυναμίες του ως προς την προστασία των προσωπικών δεδομένων. Είναι συχνό φαινόμενο να γίνονται αναφορές για διαρροές δεδομένων από βάσεις δεδομένων διάφορων εταιρειών.

Το Web3 αποτελεί την Τρίτη γενιά του διαδικτύου, φέρνοντας στην επιφάνεια νέες τεχνολογίες όπως: edge computing, Αποκεντρωμένα δίκτυα Δεδομένων και την Τεχνητή νοημοσύνη.



*Εικόνα 13: Λογισμικό επαλήθευσης από αποκεντρωμένη Blockchain Camera που δουλεύει στο Blockchain του Ethereum .*

Όπως αναφέρθηκε αρχικά στο Web2, όλες οι υπηρεσίες βρίσκονται σε κάποιες κεντρικές εταιρείες. Η εμπιστοσύνη ως προς την ακεραιότητα και την ιδιωτικότητα των δεδομένων των χρηστών, εξαρτάται αποκλειστικά από την εταιρεία ιδιοκτήτη του server. Αυτό σημαίνει ότι οι χρήστες πρέπει να



εμπιστεύονται την υπηρεσία όπου θα χρησιμοποιήσουν τυφλά. Κανείς δεν μπορεί να τους εγγυηθεί την ιδιωτικότητα.

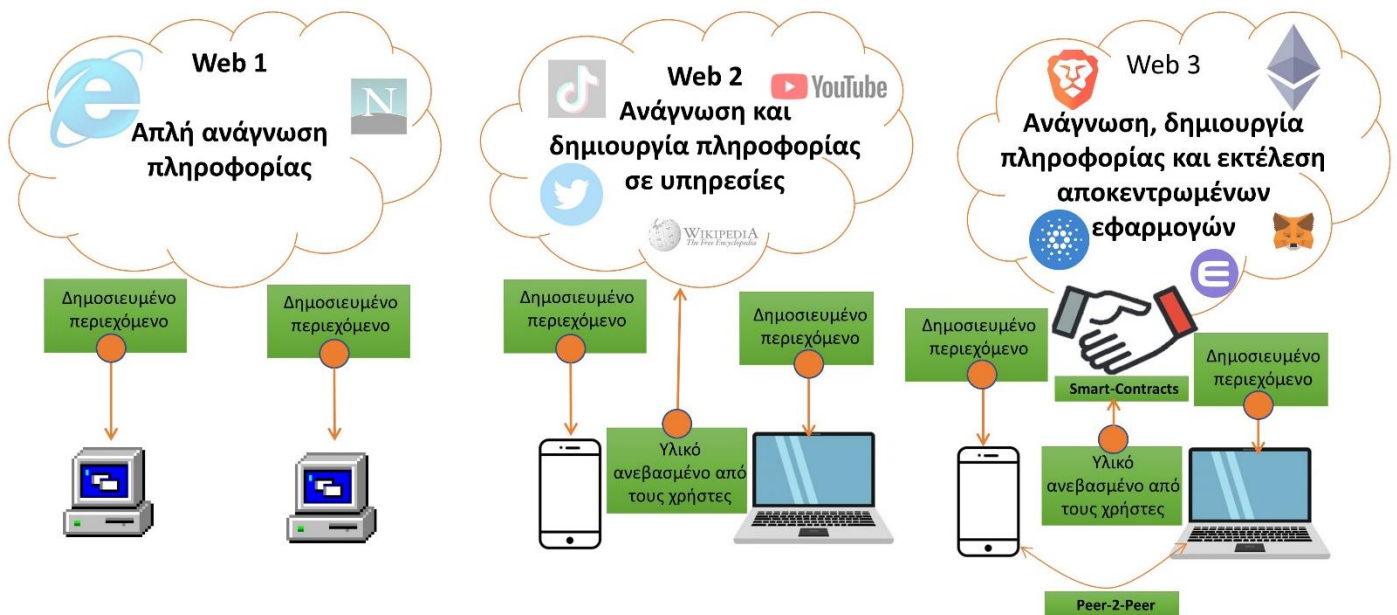
Το Web3 ως κύρια χαρακτηριστικά του, αποτελεί νέα ανοικτά, trustless και permissionless δίκτυα. Με το Web3, άνθρωποι, εταιρείες και μηχανές μπορούν να ανταλλάξουν, πληροφορίες και να δουλέψουν μαζί με άλλους ανθρώπους, εταιρείες και μηχανές όπου δεν εμπιστεύονται απαραίτητα. Αναλυτικότερα:

- **Ανοικτά** μιας και ο κώδικάς του είναι ανοικτού κώδικα, κάτι που σημαίνει πώς ο κώδικας είναι προσβάσιμος από τον καθένα για ανάγνωση, κάτι που επιτρέπει σε ολόκληρες κοινότητες από προγραμματιστές να ελέγξουν τον κώδικα για ενδεχόμενα κενά ασφαλείας. Γνωρίζοντας το τι ακριβώς τρέχει, υπάρχει το προτέρημα ότι είναι πολύ πιο εύκολο να βρεθεί και να επιλυθεί άμεσα κάποιο πρόβλημα, αλλά και δεν βασίζεται σε τυφλή εμπιστοσύνη. Σε κάθε περίπτωση, το λογισμικό αυτό μπορεί να εκτελεστεί από τον καθένα.
- **Trustless** μιας και επιτρέπει στα μέλη του δικτύου να αλληλεπιδρούν με άλλους συμμετέχοντες δημόσια ή ιδιωτικά χωρίς να χρειάζεται κάποιος trusted third party.
- **Permissionless** όπου ο καθένας από τους χρήστες και τους suppliers, μπορεί να συμμετέχει σε αυτά χωρίς authorization από κάποιον κυβερνόντα.

Τα αποκεντρωμένα δίκτυα δεδομένων δίνουν δυνατότητες στους χρήστες να διαχειριστούν τα δεδομένα τους (από ατομική χρήση σε δεδομένα υγείας, δεδομένα καλλιέργειας ενός αγρότη και άλλες εφαρμογές) όπως την πώληση ή την ανταλλαγή τους, χωρίς όμως να χάσουν τον έλεγχο ιδιοκτησίας, να μεταβληθεί η ιδιωτικότητα ή να βασίζονται σε trusted third parties. Επομένως, καταλαβαίνουμε ότι το Web3 φέρνει αποκεντρωμένα δίκτυα όπου μπορούν να αλλάξουν ριζικά τον τρόπο όπου λειτουργεί το διαδίκτυο σήμερα.

Το Web3 ενεργοποιεί την δυνατότητα όπου κατανεμημένοι χρήστες και μηχανές έχουν την δυνατότητα να αλληλεπιδράσουν με δεδομένα από τιμή σε τιμή, χωρίς να εμπλέκεται κάποιος τρίτος. Άρα, το δίκτυο καταλήγει πιο ανθρωποκεντρικό βελτιώνοντας την ιδιωτικότητα, κατασκευάζοντας έτσι ένα νέο web.

# Η ιστορία του Διαδικτύου



## Αποκεντρωμένη Ιδιωτικότητα:

Όπως έχει επισημανθεί παραπάνω, τα παραδοσιακά δίκτυα φέρνουν συχνά περιστατικά παραβιάσεων της ασφαλείας τους, με αποτέλεσμα να υπάρχουν διαρροές δεδομένων.

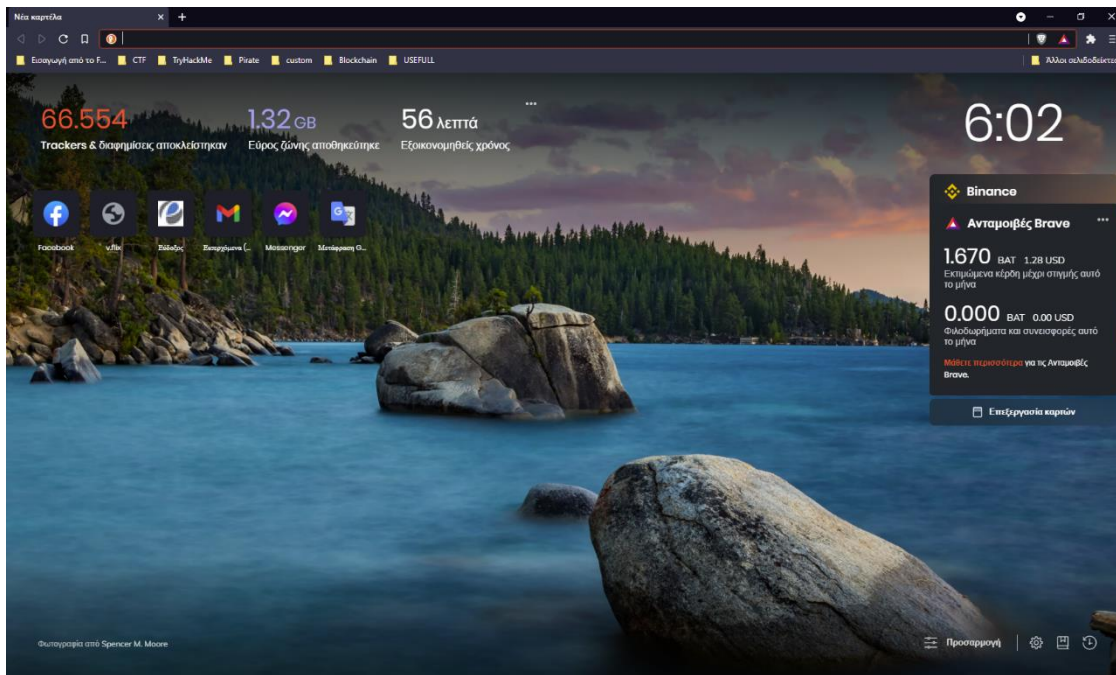
Το Blockchain εξαλείφει την ανάγκη για κεντροποιημένες αρχές, βελτιώνοντας έτσι την ιδιωτικότητα του χρήστη.

Το απόρρητο των δεδομένων μπορεί να διαμορφωθεί εφαρμόζοντας τα παρακάτω μέτρα:

- Αποκεντρωμένη αποθήκευση και μεταφορά δεδομένων.
- Ενσωμάτωση της αποκέντρωσης με καινοτομίες όπως η κρυπτογραφία, τα multiparty computation καθώς και με περιβάλλοντα αξιόπιστης εκτέλεσης (Trusted Execution Environments).

Το Blockchain τροφοδότησε την ανάπτυξη μίας νέας γενιάς ιδεών, όπου επιτρέπουν στην ιδιωτικότητα να επιστρέψει στα χέρια των χρηστών. Για να επιτευχθεί όμως αυτό, ο μηχανισμός των Blockchain transactions έχει πολλαπλά εμπόδια μπροστά του να ξεπεράσει από το να καταλήξει κεντροποιημένος. Οι τρεις κύριοι λόγοι είναι:

- Να αποτρέπει την αναπαραγωγή (replicate) ταυτότητας μέσω από επαλήθευση δεδομένων από όλους τους χρήστες του δικτύου με χρήση timestamps στις καταγραφές των transaction.
- Να αποτρέπει την αλλοίωση δεδομένων με χρήση αλγορίθμων κατακερματισμού.
- Να αποτρέπει την χειραγώγηση της επεξεργασίας δεδομένων με την επίτευξή πλειοψηφίας συναίνεσης με διάφορους μηχανισμούς (Proof Of Work, Proof Of Stake).



*Εικόνα 14: Brave Browser: Ένας web browser με ενσωματωμένο lightweight wallet, έτοιμος για πρόσβαση σε σελίδες Web3.0*

## 9.2 Blockchain Oracles

Τα Blockchain και τα smart-contracts όπου τρέχουν σε αυτά δεν μπορούν να έχουν πρόσβαση σε δεδομένα έξω από το κάθε δίκτυο του Blockchain. Για να λειτουργήσει ομαλά ένα smart-contract, όπου χρειάζεται δεδομένα να επεξεργαστεί από τον έξω κόσμο (δηλαδή, έξω από το δίκτυο του Blockchain), χρειάζεται μία «συμβατική» συμφωνία, με την μορφή ψηφιακών δεδομένων, όπου ονομάζονται Oracles. Τα oracles είναι υπηρεσίες οι οποίες στέλνουν και επιβεβαιώνουν δεδομένα από τον πραγματικό κόσμο και υποβάλουν αυτές τις πληροφορίες σε smart-contracts, προκαλώντας έτσι αλλαγές στο blockchain. Τα Oracles προσφέρουν στα smart contracts εξωτερικές πληροφορίες οι οποίες μπορούν να προκαλέσουν προκαθορισμένες ενέργειες σε ένα smart-contract. Αυτά τα εξωτερικά δεδομένα συνήθως προέρχονται είτε από εφαρμογές με Big-Data (Από κλάδους υγείας, Βιβλιοθήκες, Θερμοκρασίες κλπ.) είτε από υλικό από το Διαδίκτυο των πραγμάτων (αισθητήρες θερμοκρασίας, διοξειδίου του άνθρακα κλπ).

Η κύρια πρόκληση με τα Oracles είναι ότι οι άνθρωποι πρέπει να εμπιστεύονται αυτές τις εξωτερικές πηγές πληροφοριών, είτε προέρχονται από κάποιον ιστότοπο ή κάποιον αισθητήρα. Έχοντας ως δεδομένο ότι τα oracles είναι third party υπηρεσίες και δεν αποτελούν κομμάτι του μηχανισμού συναίνεσης του Blockchain (Consensus Mechanism), δεν υπόκεινται στους μηχανισμούς ασφαλείας που υπάρχουν στο κάθε σύστημα Blockchain. Έτσι θα μπορούσε κάποιος να επιχειρήσει μία “man-in-the-middle” επίθεση, μεταξύ του smart contract και oracle.

### 9.3 Unstoppable Domains

Τα Unstoppable Domains πρόκειται για blockchain based domains. Αυτά τα domains αντικαθιστούν μία address με κάποιο όνομα και δουλεύουν σε διάφορα Blockchain όπως του Ethereum και Zilliqa. Έτσι, τα Unstoppable Domains είναι αποκεντρωμένα. Από την στιγμή που κατοχυρωθούν από κάποιον χρήστη, αυτός ο χρήστης έχει την αποκλειστική κατοχή αυτού του domain και άρα μπορεί να τα μεταφέρει, να τα ενημερώσει και να τα συνδέσει με άλλες υπηρεσίες χωρίς να έχει κάποιος άλλος την δυνατότητα να τον περιορίσει.

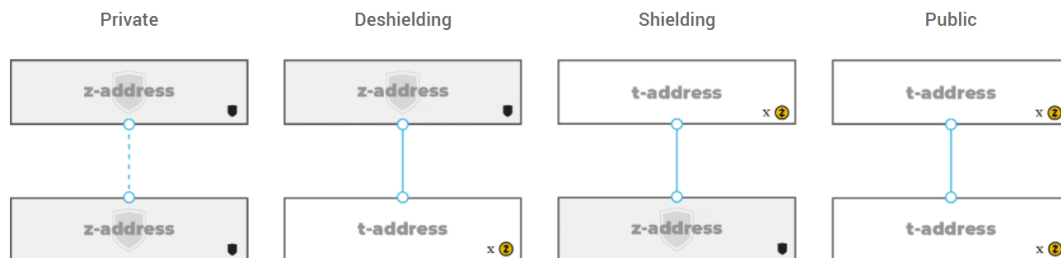
Έτσι, ο κάτοχος ενός unstoppable domain είναι ο αληθινός κάτοχος του, κανείς δεν μπορεί να του το κατάσχει ή να ελεγχθεί από κάποια εξουσία.

## 10. Η λύση του κρυπτονομίσματος Zcash

Το Zcash αποτελεί ένα fork του Bitcoin, δηλαδή μία εξ ολοκλήρου νέα και διαφορετική αλυσίδα από αυτή του Bitcoin, από το οποίο μάλιστα χρησιμοποιεί ένα μέρος από τον κώδικά του καθώς και παρόμοιους μηχανισμούς πάνω στο Proof Of Work και UTXO από αυτό. Το Zcash έχει διαμορφωθεί πάνω στον κώδικα του Bitcoin με τέτοιο τρόπο, έτσι ώστε να επιτρέπει την διασφάλιση της ιδιωτικότητας των συναλλαγών αλλά και των δεδομένων τους, χρησιμοποιώντας Zero-Knowledge Proofs.

Το κρυπτονομίσμα στο Zcash Blockchain είναι το ZEC. Ως fork του Bitcoin, και στο ZEC μπορούν να παραχθούν συνολικά 21.000.000 ZEC, δεν θα παράγονται καινούρια δηλαδή, όπως στο Ethereum και Monero για πάντα. Αντίστοιχα, η μικρότερη υποδιαίρεση του ZEC είναι το zatoshi, δηλαδή 0.00000001 ZEC. Ο μέσος όρος για να παραχθεί κάθε block είναι 75 δευτερόλεπτα και το μέγεθός τους δεν μπορεί να ξεπερνάει τα 2MB. Τέλος, το Zcash δίνει την δυνατότητα στις συναλλαγές όπου μετά το πέρας ενός χρονικού ορίου δεν έχουν επαληθευτεί (50 λεπτά της ώρας, δηλαδή να έχουν περάσει 40 blocks χωρίς να έχει γίνει Validate) να λήγουν επιστρέφοντας έτσι τα χρήματα πίσω στο αρχικό πορτοφόλι χωρίς να χάνονται.

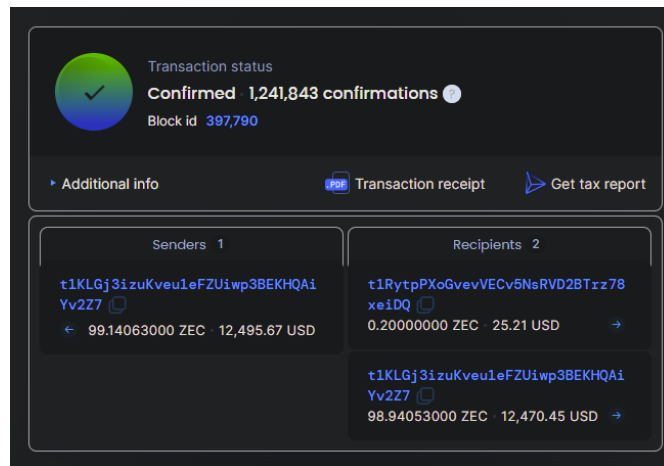
Το Zcash χρησιμοποιεί το ZK-SNARKs για να προστατέψει την ιδιωτικότητα των συναλλαγών, το οποίο θα παρουσιαστεί σε επόμενο κεφάλαιο. Χαρακτηριστικό του Zcash είναι ότι υποστηρίζει και ιδιωτικά transactions, συμπεριλαμβανομένου απόκρυψης στοιχείων από παραλήπτη, αποστολέα, καθώς και κανονικές συναλλαγές όπως θα συνέβαινε με το Bitcoin αλλά και συνδυασμό των παραπάνω διαδικασιών.



*Εικόνα 15: Διάγραμμα παρουσίασης των διευθύνσεων του Zcash όπου μπορούν να στείλουν κάποιο υπόλοιπο Zcash σε κάποια άλλη διεύθυνση.*

Στο Zcash συναντάμε 2 τύπων διευθύνσεις, τις Private (z-addresses) καθώς και τις transparent (t-addresses), έτσι ώστε να μπορούμε να πραγματοποιήσουμε τις συναλλαγές μας.

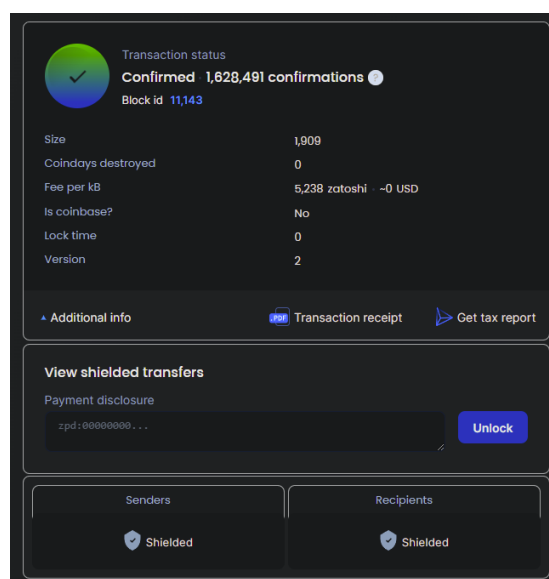
Οι z-addresses ξεκινάνε με το χαρακτηριστικό “z”, ενώ οι t-addresses με το χαρακτηριστικό “t” αντίστοιχα. Οι διευθύνσεις που ξεκινάνε με “t” λειτουργούν όπως στο Bitcoin, δηλαδή πραγματοποιούνται κανονικά transactions στο δίκτυο χωρίς χρήση κάποιο μηχανισμού απόκρυψης των διευθύνσεων. Έτσι, ο καθένας μπορεί να παρακολουθήσει τις διευθύνσεις των εμπλεκόμενων καθώς και τον αριθμό των Zcash που αποστάλθηκαν, όπως θα συνέβαινε και στο δίκτυο του Bitcoin. Από την άλλη, οι συναλλαγές μεταξύ των z-addresses αποκρύπτουν τα αληθινά στοιχεία των εμπλεκόμενων με την χρήση Zero-Knowledge-Proofs και έτσι δεν είναι εφικτό να διαπιστωθεί ποιος έστειλε σε ποιόν κάποια Zether καθώς και τον αριθμό των Zcash που αποστάλθηκαν.



*Εικόνα 16: Συναλλαγή στο Zcash μεταξύ «t» διευθύνσεων. Οι διευθύνσεις των εμπλεκόμενων φαίνονται κανονικότητα.*

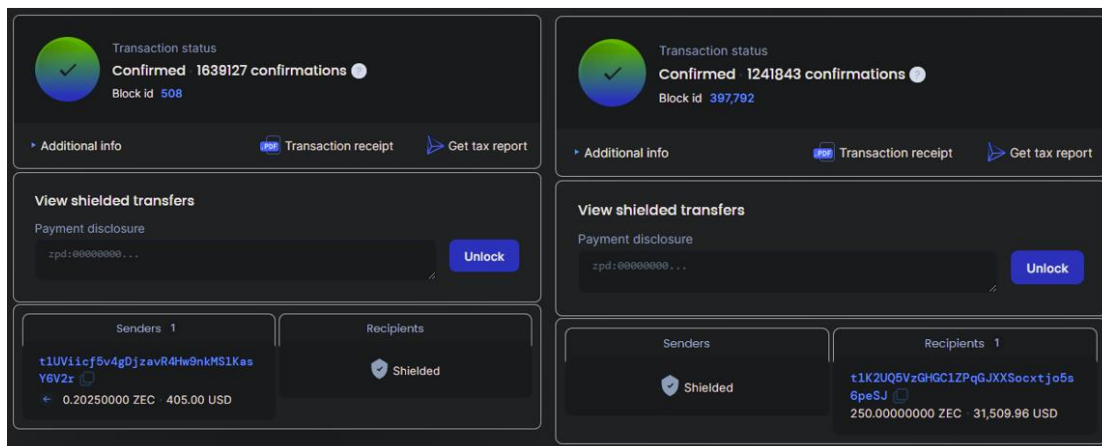
Επιπλέον, αξίζει να επισημανθεί ότι το πρωτόκολλο επιτρέπει να πραγματοποιηθούν συναλλαγές μεταξύ t και z addresses. Δηλαδή, ένας αποστολέας που δεν θέλει να φανερωθεί η ταυτότητα του ατόμου που στέλνει χρήματα, μπορεί να αποστείλει από την «t» διεύθυνσή του την «z» διεύθυνση του παραλήπτη. Η διαδικασία αυτή ονομάζεται Shielding. Αντίστροφα, αν ένας αποστολέας οπότε δεν θέλει να φανερώσει την ταυτότητα του, μπορεί να στείλει από την «z» διεύθυνση του σε κάποια public «t» διεύθυνση. Η διαδικασία αυτή ονομάζεται Dshielding.

Συνεπώς, αν θέλουμε να εκμεταλλευτούμε όλες τις δυνατότητες του Zcash για ιδιωτικές συναλλαγές, θα πρέπει να χρησιμοποιούμε αποκλειστικά τις «z» διευθύνσεις. Σε αυτό το σημείο επίσης αξίζει να αναφερθεί ότι μία «z» προς «z» συναλλαγή εμφανίζεται στο Public Blockchain, και είναι γνωστό τα transaction fees τα οποία πληρώθηκαν, ωστόσο οι διευθύνσεις και ο αριθμός των Zcash που αποστάλθηκαν είναι κρυπτογραφημένος και δεν μπορεί να εμφανιστεί.



*Εικόνα 17: Συναλλαγή στο Zcash μεταξύ «z» διευθύνσεων. Κανένα στοιχείο για τους ιδιοκτήτες καθώς και τον αριθμό των Zcash οπου ανταλλάχτηκαν δεν φανερώνεται.*





*Εικόνα 18: Συναλλαγή με Shielding και Deshielding αντίστοιχα. Το Zcash μας επιτρέπει να αποκρύψουμε την διεύθυνση από την μία πλευρά της συναλλαγής όπου επιθυμούμε.*

Άλλες ενδιαφέρουσες λειτουργίες του Zcash αποτελούν τα:

- **Memos:** Το Zcash υποστηρίζει την επισύναψη πληροφοριών και οδηγιών σε μία συναλλαγή (ή κάποια metadata), με χρήση κρυπτογραφίας έτσι ώστε να μπορεί ο αποστολέας να μοιραστεί κάποιες επιθυμητές πληροφορίες με τον παραλήπτη με ασφάλεια.
- **Viewing keys:** Άλλη μία ενδιαφέρουσα λειτουργία του Zcash είναι η ύπαρξη Viewing Keys. Όπως αναφέρθηκε και προηγουμένως, οι “z” διευθύνσεις χρησιμοποιούνται για να διασφαλίσουν την ιδιωτικότητα του ατόμου. Τι συμβαίνει όμως αν, για παράδειγμα για φορολογικούς λόγους, θέλουμε σε μία έμπιστη Τρίτη οντότητα να της φανερώσουμε επιλεκτικά κάποια στοιχεία έτσι ώστε να μπορεί να κάνει τους ανάλογους ελέγχους; Αυτό το δικαίωμα επιλεκτικής ανάγνωσης μπορεί να συμβεί με χρήση των viewing keys. Με την χρήση των Viewing Keys μπορούμε να δείξουμε κάποιες πληροφορίες ή και memos της συναλλαγής, χωρίς να φανερωθούν οι διευθύνσεις.
- **MultiSignature Transactions:** transactions όπου την έγκριση τους την δίνουν πολλαπλές διαφορετικές “i” διευθύνσεις, ωστόσο δεν είναι δυνατή η απόκρυψη των στοιχείων τους.

## 11. Ιδιωτικότητα στα Smart-Contracts

Η ιδιωτικότητα σε επίπεδο τρόπου λειτουργίας του Blockchain είναι εξαιρετικά σημαντική. Ωστόσο, μπορεί να υπάρξει ιδιωτικότητα και σε επίπεδο εφαρμογών (Layer2) όπου τρέχουν σε κάποιο Public Blockchain όπως το Ethereum; Σε αυτό το σημείο θα εξετάσουμε τους τύπους μεταβλητών που υποστηρίζει το Ethereum καθώς και θα εξετάσουμε κατά το πόσο πραγματικά υπάρχει ιδιωτικότητα στο Blockchain του Ethereum και το πώς μπορούμε να βελτιώσουμε την ιδιωτικότητα των δεδομένων.

Το Ethereum τρέχει τις αποκεντρωμένες εφαρμογές του εντός ενός ενοποιημένου περιβάλλοντος εντός του Ethereum Blockchain, όπου ονομάζεται

Ethereum Virtual Machine (EVM). Τα Smart-Contracts στο οικοσύστημα του Ethereum γράφονται στην γλώσσα προγραμματισμού Solidity. Η Solidity, όπως και κάθε γλώσσα προγραμματισμού, διαθέτει κάποια ορισμένα χαρακτηριστικά κατά την συγγραφή των smart-contracts.

### 11.1 Συναρτήσεις και μεταβλητές:

Οι μεταβλητές και οι συναρτήσεις οπου μπορούν να οριστούν σε ένα smart-contract είναι ένα από τα σημαντικότερα σημεία σε ένα smart-contract. Ανάλογα με τον τρόπο οπου δηλωθεί μία μεταβλητή ή συνάρτηση, θα μπορεί να καλείται, είτε ως προς ανάγνωση είτε ως προς εγγραφή στο Blockchain, από διαφορετικά άτομα.

Για παράδειγμα, έχουμε τον παρακάτω κώδικα:

```
public owner;  
function change_owner() public payable {  
    owner= msg.sender;  
}
```

Στον συγκεκριμένο κώδικα, βλέπουμε μία public μεταβλητή owner και μία public payable function change\_owner(). Ας εξετάσουμε αρχικά τι σημαίνει public μεταβλητή. Μία public μεταβλητή είναι ένας τύπος μεταβλητής που ο καθένας μπορεί να δει το περιεχόμενο της χωρίς κάποιον περιορισμό. Όταν μία μεταβλητή δηλωθεί ως public, τότε παράγεται μία συνάρτηση που μπορεί να καλεστεί και απευθείας κάποιος χρήστης ή άλλο smart-contract να δει τι εμπεριέχεται στην μεταβλητή αυτή. Για παράδειγμα, για να δούμε εντός της owner θα δίναμε μέσα από μία κονσόλα με σύνδεση στο Ethereum Blockchain:

```
await contract.owner()
```

Και θα εμφανιζόταν το περιεχόμενο της μεταβλητής owner. Σε αυτό το σημείο, πρέπει να αναφερθεί ότι η ανάγνωση δεδομένων από το Blockchain του Ethereum δεν έχει κάποιο κόστος. Ο καθένας μπορεί να διαβάσει το περιεχόμενο ενός Public Blockchain χωρίς κάποιο οικονομικό κόστος.

Τώρα θα αναλύσουμε την συνάρτηση change\_owner(). Όπως βλέπουμε, η συνάρτηση change\_owner είναι μία public payable συνάρτηση οπου σημαίνει ότι έχει πρόσβαση ο καθένας ως public και μιας και είναι payable, μπορεί ο καθένας να δημιουργήσει κάποιο transaction έτσι ώστε να την καλέσει. Για να την καλέσουμε δίνουμε:

```
contract.change_owner({value:1})
```

Και πληρώνουμε τα ανάλογα transaction fees. Στην παραπάνω εντολή, καλούμε την συνάρτηση change\_owner και στέλνουμε value 1. Το value 1 αποτελεί 1 Wei, δηλαδή την μικρότερη ποσότητα Ether οπου μπορεί να σταλεί και ισοδυναμεί με 0,000000000000000001 Ether στο δίκτυο του Ethereum. Μόλις



επικυρωθεί η συναλλαγή, ο κώδικας θα θέσει αυτόν που πραγματοποίησε την συναλλαγή ως owner και έτσι αν ξαναδώσουμε την εντολή:

```
await contract.owner()
```

Και με αυτό τον τρόπο βλέπουμε την νέα διεύθυνση όπου αποθηκεύτηκε στην owner, όπου είναι η δικιά μας διεύθυνση.

```
await contract.owner()  
'0x9CB391dbcD447E645D6Cb55dE6ca23164130D008'
```

*Εικόνα 19: Προβολή του περιεχομένου της public μεταβλητής owner από την γραμμή εντολών.*

## 11.2 Private μεταβλητές

Και αφού ελέγξαμε τι συμβαίνει με τις public μεταβλητές, ας δούμε τώρα τι συμβαίνει με τις private μεταβλητές. Σε αυτό το παράδειγμα, θα ελέγξουμε το συγκεκριμένο smart-contract, όπου δεν είναι τίποτα περισσότερο από ένα απλό κλειδωμένο λουκέτο.

```
// SPDX-License-Identifier: MIT  
pragma solidity ^0.8.11;  
contract Vault {  
    bool public locked;  
    bytes32 private password;  
    constructor(bytes32 _password) public {  
        locked = true;  
        password = _password;  
    }  
    function unlock(bytes32 _password) public {  
        if (password == _password) {  
            locked = false;  
        }  
    }  
}
```

Αρχικά, να αναφερθεί ότι ο constructor() είναι μία συνάρτηση όπου εκτελείται αποκλειστικά μία φορά, κατά το deploy του smart-contract (δηλαδή την δημιουργία του smart-contract και ένταξη του στο Blockchain του Ethereum) και δεν μπορεί να καλεστεί ξανά. Ας δούμε τώρα τις μεταβλητές του smart-contract. Παρατηρούμε στον κώδικα ότι υπάρχει μία public Boolean μεταβλητή locked όπου αποτελεί το λουκέτο, καθώς και μία private byte32 μεταβλητή password. Η public locked, όπως είδαμε και προηγουμένως, μπορούμε να την καλέσουμε κατευθείαν δίνοντας:

```
await contract.locked()
```

Και να ελέγξουμε τι υπάρχει μέσα. Ωστόσο, δεν μπορούμε να κάνουμε το ίδιο και με την password, μιας και είναι private. Μία private μεταβλητή δεν διαθέτει συνάρτηση προβολής, άρα δεν μπορούμε απευθείας να ελέγξουμε τι υπάρχει μέσα σε αυτήν. Αυτό μπορούμε να το επαληθεύσουμε επίσης, δίνοντας:

```
contract.abi
```

Με αυτή την εντολή, μπορούμε να δούμε όλες τις συναρτήσεις και μεταβλητές όπου μπορούμε να καλέσουμε, είτε για να δούμε τι υπάρχει μέσα είτε για να πραγματοποιήσουμε transactions.

```
contract.abi
▼ (3) [{...}, {...}, {...}] ⓘ
  ▶ 0: {inputs: Array(1), stateMutability: 'nonpayable', type: 'cc
  ▶ 1: {inputs: Array(0), name: 'locked', outputs: Array(1), state
  ▶ 2: {inputs: Array(1), name: 'unlock', outputs: Array(0), state
      length: 3
  ▶ [[Prototype]]: Array(0)
```

*Εικόνα 20: Αποτέλεσμα από την εκτέλεση του contract.abi. Δεν φαίνεται η private μεταβλητή password άρα δεν μπορούμε να την καλέσουμε.*

Ούτε εκεί βλέπουμε την password. Μία λύση θα ήταν να κάνουμε bruteforce attack, δηλαδή δοκιμή τυχαίων συνδυασμών από λίστες γνωστών κωδικών ή τυχαίες δοκιμές διάφορων συνδυασμών, μέχρι να βρούμε τον κωδικό. Κάτι τέτοιο όμως απαιτεί χρόνο και μιας και οι έλεγχοι γίνονται εντός Blockchain, εξαιρετικά κοστοβόρο. Εδώ πρέπει να υπενθυμίσουμε ότι το Main δίκτυο του Ethereum πρόκειται για ένα Public Blockchain όπου ο καθένας μπορεί να συμμετάσχει και να διαβάσει τι υπάρχει εντός δικτύου. Η private μεταβλητή μας περιορίζει στο να δούμε απευθείας την πληροφορία, αλλά δεν μας περιορίζει από το να διαβάσουμε απευθείας εντός των slots του αποθηκευτικού χώρου του συγκεκριμένου smart-contract. Το Ethereum αποθηκεύει σε slots με την σειρά τα δεδομένα όπου εισάγονται σε ένα smart-contract. Έτσι, αν ελέγξουμε το πρώτο slot (slot 0), θα δούμε την τιμή 00001, που σημαίνει true και είναι δηλαδή η μεταβλητή locked όπου είναι true. Αν δούμε το ακριβώς επόμενο slot, δηλαδή το slot1, θα μας εμφανίσει το περιεχόμενο της private μεταβλητής password. Δίνουμε σε μία γραμμή εντολών:

```
var pwd // Δημιουργούμε μία μεταβλητή pwd
web3.eth.getStorageAt(contract.address, 1, function( err, result){pwd = result})
//Αποθηκεύουμε στην μεταβλητή pwd το περιεχόμενο του slot1 από το
συγκεκριμένο smart-contract
```

Και τώρα μας εμφανίζεται ο κωδικός σε δεκαεξαδική μορφή. Αν το βάλουμε σε έναν μετατροπέα από δεκαεξαδικό σε ASCII, βλέπουμε τον κωδικό και είναι:

```
A very strong secret password :)
```

Όπως καταλαβαίνουμε, η έννοια του `private` δεν υφίσταται πραγματικά στο Blockchain. Έτσι, δεν θα έπρεπε να αποθηκεύουμε εντός αλυσίδας κάποια ευαίσθητη πληροφορία αποκρυπτογραφημένη. Γι' αυτό τον λόγο, συνιστάται η αποθήκευση των απόρρητων αρχείων να γίνεται σε κάποιο εξωτερικό κεντρικό server (π.χ. κάποιο cloud) και εντός Blockchain να αποθηκεύεται ένα hash από το αρχείο, έτσι ώστε να μπορεί να αποδειχθεί ότι το αρχείο δεν έχει τροποποιηθεί. Όταν ένα αρχείο βρίσκεται σε κάποιο Blockchain, όλα τα άτομα τα οποία συμμετέχουν στο δίκτυο έχουν δικαίωμα ανάγνωσης στο αρχείο. Τέλος, αν πραγματικά είναι απαραίτητη κάποια πληροφορία να αποθηκευτεί εντός του Blockchain, συνιστάται να καταχωρηθεί κρυπτογραφημένη με ένα αρκετά ισχυρό κλειδί κρυπτογράφησης.

### 11.3 Internal μεταβλητές

Εκτός από τις `public` και `private` που επεξηγήθηκαν παραπάνω, στην Solidity συναντάμε και τις `internal` μεταβλητές. Μία `internal` μεταβλητή επιτρέπει πρόσβαση στις εσωτερικές μεταβλητές κατάστασης μόνο εσωτερικά από το `smart-contract` ή άλλα `smart-contracts` που κληρωνόμουν τα χαρακτηριστικά της με την χρήση πολυμορφισμού. Δεν είναι δηλαδή δυνατή η πρόσβαση σε αυτές εξωτερικά από άλλα `smart-contract` ή κάποιο `smart-contract` από το οποίο επεκτείνεται τις δυνατότητες του.

Για παράδειγμα, έχουμε το εξής `smart-contract`:

```
pragma solidity ^0.8;

contract Parent {
    bool internal internalProperty;
    bool private privateProperty;
}

contract Child is Parent {
    function foo() external {
        // ok
        internalProperty = true;

        // error, not visible
        privateProperty = true;
    }
}
```

Εδώ έχουμε 2 `smart-contracts`, το `Parent` και `Child`. Το `Child` αποτελεί μία προέκταση του `smart-contract` `Parent`. Η μεταβλητή `internalProperty` είναι μία μεταβλητή `internal` που σημαίνει ότι μπορεί το περιεχόμενό της είναι φανερό μόνο εντός του `smart-contract` της και των `smart-contracts` που την επεκτείνουν, άρα το `smart-contract` `Child` μπορεί να αλλάξει η τιμή του `internalProperty`. Δεν συμβαίνει το ίδιο όμως για την `privateProperty`, η οποία μιας και είναι `private` δεν είναι φανερό το τι εμπεριέχεται σε αυτή στην `Child` και δεν μπορεί να μεταβάλει την τιμή του. Ωστόσο όπως και στο προηγούμενο παράδειγμα, αν κάποιος

εξερευνήσει το storage και των 2 τύπων μεταβλητών, πάλι θα είναι σε θέση να δει τι εμπεριέχεται μέσα στις μεταβλητές αυτές.

#### 11.4 Τεχνολογίες Διασφάλισης της Ιδιωτικότητας στα Smart-Contracts

Τα smart-contracts χρησιμοποιούν γλώσσες προγραμματισμού υψηλού επιπέδου για την έκφραση σύνθετων απαιτήσεων και προτύπων, επιτρέποντας τόσο στους προγραμματιστές όσο και στους απλούς χρήστες να εκφράζουν σύνθετες απαιτήσεις και πρότυπα. Ταυτόχρονα, ο δημόσιος χαρακτήρας των περισσότερων δημοφιλών Blockchain αλλά και smart-contract που τρέχουν μέσα σε αυτά, περιορίζει την εκφραστικότητα των smart-contracts. Όλοι μπορούν να δουν τα περιεχόμενα των smart-contracts και οι συναλλαγές αποκαλύπτουν πολλές ιδιωτικές πληροφορίες. Πολλά πρότυπα επικοινωνίας με διασφάλιση της ιδιωτικότητας που απαιτούν αυτές οι πληροφορίες να είναι ιδιωτικές και προσβάσιμες μόνο σε συγκεκριμένα μέρη αντιμετωπίζουν περιορισμούς πόρων ή πολυπλοκότητας σχεδιασμού, παρά το γεγονός ότι είναι θεωρητικά δυνατή η υλοποίησή τους. Επιπλέον, τα security proofs για αυτούς τους τύπους συστημάτων είναι χρονοβόρα λόγω των πρόσθετων μη τυποποιημένων και μη ενοποιημένων επιπέδων αφαίρεσης που εισάγονται με την εφαρμογή της λογικής διατήρησης της ιδιωτικότητας πάνω στη λειτουργικότητα των public smart-contracts και όχι με τον σχεδιασμό περιβαλλόντων έξυπνων συμβολαίων με γνώμονα την υποστήριξη της ιδιωτικότητας.

Σε αυτό το κεφάλαιο, θα εξετάσουμε δύο περιπτώσεις για ιδιωτικά συστήματα smart-contract, το Zkay και το Kachina, τα οποία παρέχουν έναν τρόπο έκφρασης και συλλογισμού για διαφορετικές ιδιότητες ιδιωτικότητας σε smart-contract.

Ενώ το Zkay είναι ένα ενδεικτικό παράδειγμα ενός πρακτικού πλαισίου smart-contracts, που δείχνει την οπτική γωνία του τελικού χρήστη ή του προγραμματιστή, το Kachina παρέχει μια πιο θεωρητική προσέγγιση της ιδιωτικότητας, επιτρέποντας λεπτομερή ανάλυση ασφάλειας των λειτουργιών του smart-contract.

##### 11.4.1 Η λύση του zkay

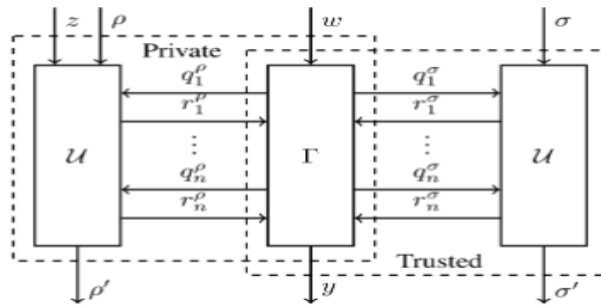
Το zkay αποτελεί μια γλώσσα προγραμματισμού, για την ακρίβεια είναι μια επέκταση της Solidity (που χρησιμοποιούμε στο Ethereum για την ανάπτυξη smart-contracts), που όμως λειτουργεί και σαν compiler οπου, ουσιαστικά, μετασχηματίζει τον κώδικα του smart-contract σε δύο ξεχωριστά κομμάτια. Το Solidity κομμάτι που είναι το εκτελέσιμο και μπορεί να τρέξει στο Ethereum Blockchain, για παράδειγμα. Το δεύτερο κομμάτι αποτελεί μια περιγραφή της λειτουργίας του κώδικα του smart-contract το οποίο προορίζεται για να τρέξει μέσα σε ένα zk-SNARK.

Η γλώσσα zkay διαφέρει από τη βασική γλώσσα Solidity στο ότι διαθέτει λεπτομερείς privacy annotations που καθορίζουν σε ποιον ανήκουν οι εσωτερικές μεταβλητές του συμβολαίου. Αυτές οι privacy annotations επιτρέπουν τον περιορισμό της πρόσβασης στη διεύθυνση του συγκεκριμένου μεμονωμένου χρήστη, έτσι ώστε η μεταβλητή να μπορεί να διαβαστεί μόνο από τον ιδιοκτήτη αυτής της διεύθυνσης. Ένα άλλο σημαντικό χαρακτηριστικό της γλώσσας είναι ο προσεκτικός χειρισμός των δηλώσεων που επαναπροσδιορίζουν τις μυστικές πληροφορίες – όποτε κάποιος θέλει να

δημοσιεύσει ή να δώσει μια secret value, ένα call στη συνάρτηση 'reveal' του επιτρέπει να το κάνει. Ως αποτέλεσμα, η σημασιολογία της γλώσσας διαχωρίζει τους ιδιωτικούς υπολογισμούς από τους δημόσιους υπολογισμούς, απαιτώντας από τον συγγραφέα του smart-contract να δηλώσει ρητά αυτό το όριο.

#### 11.4.2 Η λύση του Kachina

Το Kachina προσεγγίζει διαφορετικά αυτά τα ζητήματα. Αντί να ορίζει μια επέκταση της γλώσσας προγραμματισμού, παρέχει μια πιο γενική αφαίρεση για το διαχωρισμό της public και private state, καθώς και ένα πλαίσιο για την απόδειξη της ασφάλειας των smart-contract. Στο επίκεντρο του Kachina βρίσκεται το πλαίσιο Universal Composability (UC), το οποίο είναι μια συλλογή τυπικών προδιαγραφών και τεχνικών απόδειξης για την απόδειξη της ασφάλειας κρυπτογραφικών πρωτοκόλλων. Το Kachina επεκτείνει το UC και προτείνει τη ροή σχεδιασμού smart-contract που παρουσιάζεται παρακάτω.



Εικόνα 21: Ροή σχεδιασμού smart-contract από Kachina.

Πρώτον, πρέπει να προσδιορίσουμε επίσημα τη συνάρτηση μετάβασης του contract ( $w$ ) στη γλώσσα ψευδοκώδικα UC, η οποία αποτελεί την καρδιά της αλληλεπίδρασης μεταξύ των χρηστών και του contract. Αυτή η συνάρτηση, ενώ μπορεί να φαίνεται ότι είναι πανομοιότυπη με οποιοδήποτε άλλο contract, έχει έναν ξεχωριστό περιορισμό: πρέπει να είναι γραμμένη με τέτοιο τρόπο ώστε οι αλληλεπιδράσεις με τη public state του smart-contract και ένα σύνολο private states (μία ανά χρήστη) να αφηρηθούν ως επικοινωνία με ξεχωριστά Oracles public ή private state, αντίστοιχα. Αυτό μας αναγκάζει να διαχωρίσουμε τις δημόσια προσβάσιμες πληροφορίες, όπως οι public μεταβλητές του smart-contract, από τις users' private states, οι οποίες γενικά αποθηκεύονται στην πλευρά του χρήστη και μπορεί να περιλαμβάνουν μυστικά κλειδιά, nonces και ό,τι άλλο απαιτείται για να ενεργήσει ως συμβαλλόμενο μέρος του smart-contract. Επιπλέον, επειδή τα πάντα ανατίθενται στα Oracles, η ίδια η συνάρτηση transition δεν διατηρεί κανένα state. Αυτή η προσέγγιση state-oracle μπορεί να ακούγεται αρκετά οικεία στους προγραμματιστές, καθώς χρησιμοποιείται ευρέως στη software engineering (σκεφτόμαστε τα Oracles ως εσωτερικά αντικείμενα του αντικειμένου λειτουργικότητας, με πρόσβαση σε αυτά μόνο με κλήση), αλλά τα δύσκολα προβλήματα που επιλύονται από το Kachina βρίσκονται περισσότερο στους ακριβείς ορισμούς και τις αφαιρέσεις που μας επιτρέπουν να σκεφτούμε αργότερα για αυτή τη συνάρτηση μετάβασης.

Στο Σχήμα 21 απεικονίζεται η συνάρτηση μετάβασης  $\Gamma$  που μετατρέπει την είσοδο  $w$  του smart-contract (η οποία μπορεί να θεωρηθεί ως ερώτημα ή RPC με επιχειρήματα) σε έξοδο  $y$ , ενώ επικοινωνεί με ένα δημόσιο στη δεξιά πλευρά, το οποίο μετατρέπει τη δημόσια κατάσταση  $\sigma$  σε νέα  $\sigma'$ , και με ένα σύνολο ιδιωτικών Oracles, καθένα από τα οποία αλλάζει την ιδιωτική κατάσταση  $\rho$  σε  $\rho'$ . (Βλέπε Παραπάνω Σχήμα).

Ως αποτέλεσμα, το Kachina χρησιμεύει ως ένα γενικό πλαίσιο στο οποίο οι ιδιότητες ασφαλείας μπορούν να εκφραστούν συνοπτικά και η αντιστοιχία του συγκεκριμένου συστήματος smart-contract μπορεί να αποδειχθεί ασφαλής από την άποψη αυτών των ιδιοτήτων. Ως θεωρητικό έργο, το Kachina δεν περιλαμβάνει γλώσσα ή compiler, αλλά βασίζεται στη γλώσσα UC. Ταυτόχρονα, η εκφραστική του δύναμη είναι αρκετά υψηλή: αποτυπώνει εύκολα τις ιδιωτικές πληρωμές τύπου Zcash (που είναι το κύριο παράδειγμα στο σώμα της εργασίας) και η αφαίρεση της ιδιωτικότητας zkay, με τις μεταβάσεις κρυπτογραφημένων τιμών, μπορεί επίσης να εκφραστεί αρκετά απλά στο Kachina, τοποθετώντας κρυπτογραφημένες τιμές σε public state, ενώ έχει μυστικά κλειδιά στα private states.

## 12. Αποκεντρωμένα Αναγνωριστικά

Τα αποκεντρωμένα Αναγνωριστικά (Decentralized Identifiers ή πιο σύντομα DIDs) είναι ένας νέος τύπος αναγνωριστικών όπου επιτρέπει την επαλήθευση στοιχείων, αναπαριστώντας έτσι μία αποκεντρωμένη ψηφιακή ταυτότητα. Ένα αποκεντρωμένο αναγνωριστικό προορίζεται για την επαλήθευση τόσο φυσικών προσώπων όσο και νομικών προσώπων, οργανισμούς, ή οποιαδήποτε άλλη οντότητα η οποία ορίζεται από τον υπεύθυνο επεξεργασίας των αποκεντρωμένων αναγνωριστικών. Χαρακτηριστικά των DIDs είναι ότι δίνουν την δυνατότητα απόδειξης ότι κάποια οντότητα είναι πράγματι αυτή που ισχυρίζεται, χωρίς να γίνεται αποκάλυψη περαιτέρω πληροφοριών που σχετίζονται με ένα DID. Δηλαδή, ένα αποκεντρωμένο αναγνωριστικό σχεδιάζεται με τέτοιο τρόπο έτσι ώστε να επιτρέπει στον ελεγκτή να επικυρώνει τα στοιχεία χωρίς να γίνεται αποκάλυψη περαιτέρω στοιχείων αλλά και χωρίς να απαιτείται κάποιο επιπρόσθετο δικαίωμα από κάποια άλλη οντότητα με τέτοιο τρόπο ώστε να δημιουργείται μία έμπιστη και αξιόπιστη σύνδεση.

Στην πράξη, ένα DID αναπαρίσταται ως ένα String το οποίο χωρίζεται σε 3 μέρη:

- Τον DID identifier
- Την DID μέθοδο όπου χρησιμοποιείται
- Τον DID method-specific identifier



Στόχοι ενός DID είναι να επιτύχει:

- **Αποκέντρωση:** Εξάλειψη της απαίτησης για κεντρικές αρχές ή και του προβλήματος ύπαρξης single point of failure, δηλαδή την διακοπή των υπηρεσιών σε περίπτωση κατάρρευσης μέρους της υποδομής, στη διαχείριση αναγνωριστικών.
- **Έλεγχος των δεδομένων:** Οι οντότητες θα έχουν οι ίδιες τον έλεγχο των δεδομένων τους, δηλαδή να μπορούν οι ίδιες να χειρίζονται τα ψηφιακά αναγνωριστικά τους χωρίς να χρειάζεται κάποια τρίτη κεντρική αρχή.
- **Ιδιωτικότητα:** Επιτρέπει στις ίδιες τις οντότητες να διαχειρίζονται και να ελέγχουν την ιδιωτικότητα των πληροφοριών τους συμπεριλαμβανομένου και της αποκάλυψης αποκλειστικά μόνο των πληροφοριών που απαιτούνται κάθε φορά για να γίνει ο έλεγχος. Σε αυτό το σημείο βέβαια αξίζει να επισημανθεί ότι το βάρος της προστασίας των δεδομένων αυτών πέφτει στην ίδια την οντότητα οπού της ανήκουν.
- **Ασφάλεια:** Χρησιμοποιούνται έγγραφα DID με περιγραφή των κρυπτογραφικών proofs για την χρήση καθώς και για την διασφάλιση των επιπέδων ασφαλείας οπού απαιτούνται.
- **Proof-Based:** Επιτρέπει στους DID controllers να παρέχουν κρυπτογραφικά proofs όταν αλληλοεπιδρούν με άλλες οντότητες.
- **Ανακάλυψη:** Δίνει την δυνατότητα στις οντότητες να ανακαλύπτουν και να εξερευνούν DIDs από άλλες οντότητες.
- **Διαλειτουργικότητα:** Δημιουργία και χρήση προτύπων, ώστε να υπάρχει μια ευρέως αποδεκτή υποδομή στον τρόπο λειτουργίας αλλά και των εργαλείων που χρησιμοποιούνται για να λειτουργήσουν τα DIDs.
- **Φορητότητα:** Να είναι ανεξάρτητο από συστήματα και δίκτυα και να επιτρέπει στις οντότητες να χρησιμοποιούν τα ψηφιακά τους αναγνωριστικά με οποιοδήποτε σύστημα που υποστηρίζει DIDs και μεθόδους DIDs.
- **Επεκτασιμότητα και απλότητα:** Ανάπτυξη με τέτοιο έτσι ώστε η τεχνολογία να είναι απλή, εύκολη και κατανοητή ενώ παράλληλα θα επιτρέπει την μελλοντική επέκταση των δυνατοτήτων του.

Εκτός από ένα έγγραφο DID, υπάρχει και ο DID controller ο οποίος είναι η οντότητα (πρόσωπο, οργανισμός ή λογισμικό) που έχει τη δυνατότητα - όπως ορίζεται από μια μέθοδο DID - να κάνει αλλαγές σε ένα έγγραφο DID. Η ικανότητα αυτή συνήθως βεβαιώνεται με τον έλεγχο ενός συνόλου κρυπτογραφικών κλειδιών που χρησιμοποιούνται από το πρόγραμμα που λειτουργεί ως ελεγκτής, αν και μπορεί υπάρξει έλεγχος και μέσω άλλων μηχανισμών. Σε αυτό το σημείο πρέπει να αναφερθεί ότι ένα DID μπορεί να έχει περισσότερους από έναν ελεγκτές και το υποκείμενο του DID μπορεί να είναι ο ελεγκτής του DID ή ένας από αυτούς.

Ωστόσο, για να είναι δυνατή η ανάγνωση αλλά και έλεγχος σε έγγραφο DID, τα DIDs καταγράφονται συνήθως σε κάποιο δίκτυο. Το δίκτυο δεν χρειάζεται απαραίτητα να είναι ένα Blockchain δίκτυο, μπορεί να περιορίζεται απλά ως ένα απλό peer-to-peer δίκτυο. Ανεξάρτητα από την τεχνολογία που χρησιμοποιείται, κάθε σύστημα που υποστηρίζει την καταγραφή των DIDs αλλά και την επιστροφή δεδομένων που είναι απαραίτητα για την παραγωγή εγγράφων DID



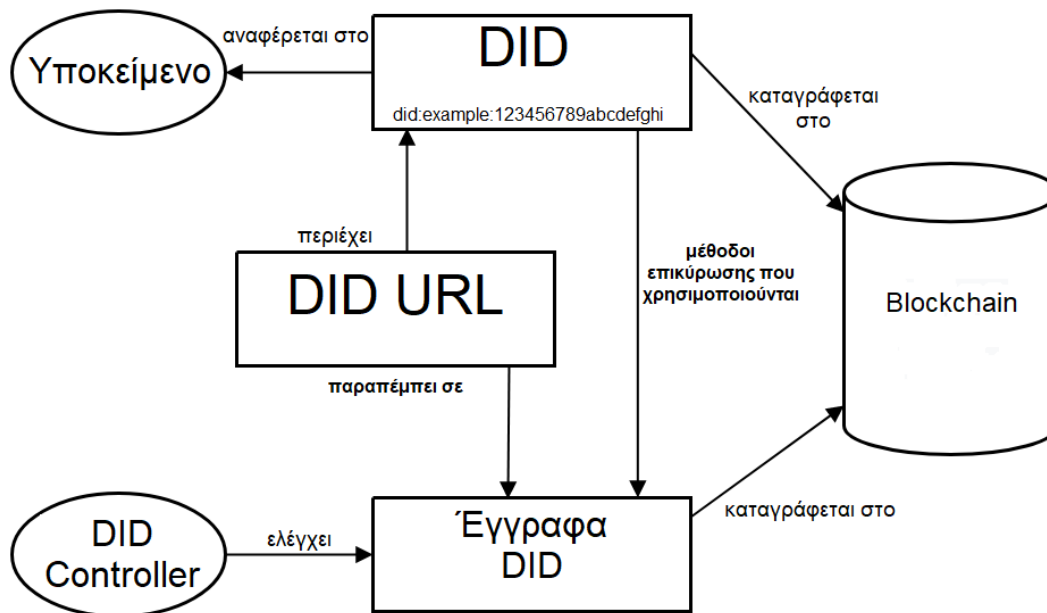
ονομάζεται μητρώο επαληθεύσιμων δεδομένων (Verifiable data registry). Μερικά παραδείγματα αποτελούν τα κατανεμημένα λογιστικά βιβλία, αποκεντρωμένα συστήματα αρχείων, βάσεις δεδομένων κάθε είδους κ.α. Τα έγγραφα DID περιέχουν πληροφορίες που σχετίζονται με ένα DID. Συνήθως εκφράζουν μεθόδους επαλήθευσης, όπως κρυπτογραφικά δημόσια κλειδιά, και υπηρεσίες σχετικές με τις αλληλεπιδράσεις με το υποκείμενο DID. Όσο αφορά τις μεθόδους DID, αποτελούν μηχανισμό με τον οποίο δημιουργούνται, επιλύονται, ενημερώνονται αλλά και απενεργοποιείται ένας συγκεκριμένος τύπος DID και το σχετικό έγγραφο DID.

```
{
  "@context": [
    "https://www.w3.org/ns/did/v1",
    "https://w3id.org/security/suites/ed25519-2020/v1"
  ]
  "id": "did:example:123456789abcdefghi",
  "authentication": [{
    // used to authenticate as did:...fghi
    "id": "did:example:123456789abcdefghi#keys-1",
    "type": "Ed25519VerificationKey2020",
    "controller": "did:example:123456789abcdefghi",
    "publicKeyMultibase": "zH3C2AVvLMv6gmMNam3uVAjZpfkcJCwDwnZn6z3wXmqPV"
  }]
}
```

*Εικόνα 22: Παράδειγμα από έγγραφο DID που προσφέρεται από την w3c.org*

Κάθε αποκεντρωμένο αναγνωριστικό μπορεί να εκφραστεί με κρυπτογραφικό υλικό, μεθόδους επιβεβαίωσης ή υπηρεσίες οι οποίες παρέχουν ένα σύνολο μηχανισμών όπου επιτρέπουν στον διαχειριστή των DIDs να αποδείξουν ότι κάποια οντότητα είναι πράγματι αυτή που ισχυρίζεται. Οι υπηρεσίες επιτρέπουν αξιόπιστες αλληλεπιδράσεις που σχετίζονται με το θέμα DID. Ένα DID μπορεί να παρέχει τα μέσα για την επιστροφή του ίδιου του υποκειμένου DID, εάν το υποκείμενο DID είναι ένας πόρος πληροφοριών όπως ένα μοντέλο δεδομένων. Αυτό το έγγραφο καθορίζει τη σύνταξη DID, ένα κοινό μοντέλο δεδομένων, βασικές ιδιότητες, σειριακές αναπαραστάσεις, λειτουργίες DID και επεξηγήσεις σχετικά με την διαδικασία επίλυσης των DID στους πόρους που αντιπροσωπεύουν. Στο Ethereum, υπάρχει η βιβλιοθήκη Ethr-DID Library όπου μπορεί να παρέχει αποκεντρωμένα αναγνωριστικά στο οικοσύστημα του Ethereum, ενώ στο Cardano το συναντάμε στο Atala Project.





*Εικόνα 23: Αναπαράσταση της αρχιτεκτονικής DID και την αλληλεπίδραση τους με τα βασικά στοιχεία που χρησιμοποιούν για να λειτουργήσουν.*

## 13. Προστασία της ιδιωτικότητας των συναλλαγών στο Ethereum

### 13.1 Η λύση του smart-contract Zether

Το Zether αποτελεί μία λύση που υπόσχεται να λύσει τα προβλήματα στην ιδιωτικότητα των συναλλαγών στο δίκτυο του Ethereum. Το Zether έρχεται ως ένα smart-contract όπου ονομάζεται Zether Smart Contract (ZSC) και μπορεί να γίνει deploy σε οποιοδήποτε Ethereum Δίκτυο (mainnet ή testnet) . Σε αντίθεση με τα μοντέλα που συζητήσαμε προηγουμένως όπου βασίζονται στο UTXO μοντέλο, το Zether ως Ethereum Based, έχει διαμορφωθεί έτσι ώστε να λειτουργεί με το Account Based μοντέλο. Το Zether προσφέρει confidentiality μιας και αποκρύπτει τον αριθμό των Zether Tokens (ZTH) όπου στέλνονται καθώς και τις διευθύνσεις συναλλαγής μεταξύ αποστολέα και παραλήπτη. Σαν μία γενική ιδέα, υποθέτουμε ότι ο Σταύρος έχει την Ethereum διεύθυνση A και δημιουργεί ένα key pair (sk,pk) και στέλνει μία συναλλαγή αποκρύπτοντας το pk και στέλνει κάποια Ether στο ZSC. Το ZSC όπου θα λάβει τα Ether, θα δημιουργήσει έναν λογαριασμό pk με τα αντίστοιχα Ether που έλαβε μετατρέποντας τα σε ZTH. Ο Σταύρος τώρα μπορεί να στείλει ZTH ανώνυμα χωρίς να μπορεί να αποκαλυφθεί κάποιο στοιχείο που να φανερώνει την ταυτότητα του. Φυσικά μπορεί όποτε επιθυμεί, να ξαναμετατρέψει τα ZTH του σε Ether.

Με την χρήση του Zether, έχουμε την δυνατότητα να πραγματοποιήσουμε τους παρακάτω τύπους συναλλαγών:

**Fund transaction** Χρησιμοποιείται για να σταλθεί ένας αριθμός από Ether σε μία Zether διεύθυνση (ή αλλιώς Elgamal public key). Ο καθένας μπορεί να στείλει Ether σε ένα Zether λογαριασμό.

**Transfer Transaction:** Χρησιμοποιείται για να σταλθεί κάποιο ποσό από ZTH από ένα Zether account σε κάποιο άλλο.

**Burn Transaction:** Χρησιμοποιούνται για να γίνει μετατροπή όλων των ZTH όπου αντιστοιχούν σε ένα Zether λογαριασμό, σε Ether για να αποσταλούν σε μία ορισμένη από τον χρήστη Ethereum διεύθυνση.

**Anonymous transaction:** Ο Σταύρος επιλέγει ένα anonymity set από  $n$  λογαριασμούς (συμπεριλαμβανομένου  $P$  και  $Q$ ). Τότε, λαμβάνει όλα τα  $tx$ s που δεν έχουν γίνει validate από όλους τους  $n$  λογαριασμούς που επέλεξε, πριν παραχθεί κάποιου είδους proof έτσι ώστε να ανταποκρίνεται στο τρέχων υπόλοιπο των λογαριασμών. Σε αυτό το σημείο πρέπει να αναφερθεί ότι το proof θα αποτύχει εάν πριν από την επεξεργασία αυτού του  $tx$ , οποιοσδήποτε από τους  $n$  λογαριασμούς λάβει ένα  $tx$  αλλάζοντας έτσι την ουρά των εκκρεμών  $tx$ . Στη συνέχεια δημιουργεί τα ciphertext  $C_1, C_2, \dots, C_n$  των διαθέσιμων χρηματικών υπόλοιπων τους και αποδεικνύει ότι:

- Όλα τα  $C_1, C_2, \dots, C_n$  είναι έγκυρα ciphertexts, με ένα από αυτά να κρυπτογραφεί με το  $x$ , ένα από αυτά να κρυπτογραφεί με  $-x$  και τα υπόλοιπα να κρυπτογραφούν με 0.
- Το  $x$  είναι θετικό σε ένα προκαθορισμένο εύρος τιμών.
- Το υπόλοιπο του ciphertext που κρυπτογραφεί το  $x$  είναι θετικό σε αυτό το εύρος τιμών.
- Χρησιμοποιείται ένα έγκυρο nonce.

Τότε, το ZSC θα χρησιμοποιήσει στη συνέχεια τα ciphertext  $C_1, C_2, \dots, C_n$  για να ενημερώσει τους  $n$  λογαριασμούς. Σε αυτό το σημείο, το υπόλοιπο μόλις των 2 λογαριασμών θα αλλάξει, καθώς τα υπόλοιπα κρυπτογραφημένα κείμενα κρυπτογραφούν 0.

## 14. Συμπεράσματα

Με το πέρας αυτής της έρευνας καταλήξαμε στο συμπέρασμα ότι το Blockchain θα αποτελέσει κύριο στοιχείο της ανάπτυξης του Web3.0, το οποίο θα αλλάξει ριζικά τον τρόπο που λειτουργεί το διαδίκτυο σήμερα προσθέτοντας μεγαλύτερη ιδιωτικότητα στους χρήστες και παράλληλα ανοίγοντας πόρτες για μεγαλύτερη κοινοποίηση πληροφορίας ανάμεσα τους. Ιδιαίτερα όσο εξελίζεται η τεχνολογία και τόσο περισσότερες τεχνολογίες διασφάλισης της ιδιωτικότητας ενσωματωθούν τόσο στα Blockchain όσο και στα υποσυστήματα που τρέχουν σε αυτά, η υλοποίηση του Web3.0 θα αλλάξει για πάντα το διαδίκτυο όπως το ξέρουμε σήμερα και θα συμβάλει στην αντιμετώπιση ακόμη περισσότερο των προβλημάτων που μαστίζουν το διαδίκτυο καθώς και την ασφάλεια και ιδιωτικότητα του.

## 15. Πίνακας Ορολογίας

Ξενόγλωσσος όρος	Ελληνικός Όρος
Malicious Third Parties	Κακόβουλοι Εξωτερικοί Παράγοντες
Malicious Third Users	Κακόβουλοι Εξωτερικοί Χρήστες
Prover	Αυτός που πρέπει να αποδείξει ότι ισχύει ο ισχυρισμός
Verifier	Αυτός που επιβεβαιώνει ότι η απόδειξη του Prover είναι αποδεκτή
Obfuscated	Δεδομένα που προέκυψαν από ένα πρόγραμμα με σκοπό να γίνονται δύσκολα readable από έναν άνθρωπο.
nonce	Τυχαίος αριθμός οπου παράγεται για αποκλειστικά μία κρυπτογράφηση σε μία επικοινωνία έτσι ώστε να αποτρέψει από replay attacks τα οποία θα μπορούσαν να συμβούν έπειτα από συλλογή δεδομένων από παλιότερες επικοινωνίες.

## 16. Συντμήσεις – Αρκτικόλεξα – Ακρώνυμα

ZKP	Zero Knowledge Proof
NIZKP	Non-Interactive Zero Knowledge Proof

## 17. Βιβλιογραφικές αναφορές

- [1] Developing a Blockchain eVoting Application using Ethereum, Dimitris Vagiakakos, Konstantinos Karahalios, Stavros Gkinos (2021) : [https://github.com/sv1sjp/eVoting\\_Elections\\_Decentralized\\_App/blob/main/eVoting\\_Smart\\_Contract\\_paper.pdf](https://github.com/sv1sjp/eVoting_Elections_Decentralized_App/blob/main/eVoting_Smart_Contract_paper.pdf)
- [2] Antonopoulos, Andreas M.; Wood, Gavin (2018). Mastering Ethereum: building smart contracts and DApps (First ed.). Sebastopol, CA: O'Reilly Media, Inc.
- [3] Antonopoulos, Andreas M.; (2017). Mastering Bitcoin: programming the open blockchain (Second ed.). CA: O'Reilly Media, Inc.
- [4] Blockchain for Cybersecurity and Pricacy (2020), Yassine Maleh, Mohammad Shojafar, Mamoun Alazab, Imed Romdhani: CRC Press
- [5] Applications of Blockchain in Healthcare (2021), Suyel Namasudra, Ganesh Chandra Deka Editors: Springer
- [6] Token Economy: How the Web3 Reinvents the Internet (2020) Shermin Voshmgir
- [7] What is Monero: <https://blockgeeks.com/guides/monero/>
- [8] Tokenized Networks: Web3, the Stateful Web: <https://blockchainhub.net/web3-decentralized-web/>
- [9] What Is Web 3.0 & Why It Matters: <https://medium.com/fabric-ventures/what-is-web-3-0-why-it-matters-934eb07f3d2b>
- [10] Bringing Privacy to Smart Contracts: <https://blog.nucypher.com/bringing-privacy-to-smart-contracts-is-nontrivial/>
- [11] Blockchain Oracles: <https://blockchainhub.net/blockchain-oracles/>
- [12] Digital Signatures (how they work): <https://www.cryptomathic.com/news-events/blog/how-digital-signatures-and-blockchains-can-work-together>
- [13] Digital Signatures (general): <https://dzone.com/articles/digital-signature-2>
- [14] How digital signatures work: <https://bisonrails.co/digital-signatures/>
- [15] Digital Signatures and blockchain: [https://www.researchgate.net/publication/339696424\\_Digital\\_signature\\_scheme\\_for\\_information\\_non-repudiation\\_in\\_blockchain\\_a\\_state\\_of\\_the\\_art\\_review](https://www.researchgate.net/publication/339696424_Digital_signature_scheme_for_information_non-repudiation_in_blockchain_a_state_of_the_art_review)
- [16] Types of digital signatures: <https://link.springer.com/article/10.1186/s13638-020-01665-w/tables/1>
- [17] Blockchain wallet (definition): <https://www.investopedia.com/terms/b/blockchain-wallet.asp>
- [18] What is a full node wallet: <https://bitcoin.org/en/full-node#what-is-a-full-node>
- [19] Importance of nodes in blockchain and what are they: <https://www.seba.swiss/research/Classification-and-importance-of-nodes-in-a-blockchain-network>
- [20] Hardware wallets (definition & how they work): <https://medium.com/radartech/hardware-wallets-explained-da8bd93ce801>

- [21] What is a hardware wallet: <https://academy.binance.com/en/articles/what-is-a-hardware-wallet>
- [22] What is a bitcoin wallet: <https://cointelegraph.com/bitcoin-for-beginners/what-is-bitcoin-wallets>
- [23] Types of wallets in blockchain: <https://academy.binance.com/en/articles/crypto-wallet-types-explained>
- [24] Zero Knowledge Proof: Blockchain Identity Management for Humans: [https://www.youtube.com/watch?v=V\\_ZWNFUHIRQ](https://www.youtube.com/watch?v=V_ZWNFUHIRQ)
- [25] Non-Interactive Zero Knowledge Proofs: <https://www.youtube.com/watch?v=cMyKGJ3yiGg>
- [26] Secure Multiparty Computation Goes Live: <https://eprint.iacr.org/2008/068.pdf>
- [27] Zero Knowledge Proofs and Their Future Applications: <https://www.youtube.com/watch?v=J3jKROwTPCs>
- [28] Smart Contract Security, Dimitris Vagiakakos, Stavros Gkinos, Ioannis Karvelas (2022): [https://github.com/sv1sjp/eVoting\\_Elections\\_Decentralized\\_App/blob/main/smartcontract\\_security\\_paper.pdf](https://github.com/sv1sjp/eVoting_Elections_Decentralized_App/blob/main/smartcontract_security_paper.pdf)
- [29] zkay: Specifying and Enforcing Data Privacy in Smart Contracts, ETH Zurich: <https://files.sri.inf.ethz.ch/website/papers/ccs19-zkay.pdf>
- [30] zkay Documentation: <https://eth-sri.github.io/zkay/>
- [31] Reasoning about privacy in smart contracts: <https://priviledge-project.eu/news/reasoning-about-privacy-in-smart-contracts>
- [32] zk-SNARKs — A Realistic Zero-Knowledge Example and Deep Dive:
- [33] [zk-SNARKs — A Realistic Zero-Knowledge Example and Deep Dive | by Adam Luciano | Coinmonks | Medium](#)
- [34] Zcash Documentation: <https://zcash.readthedocs.io/en/latest/>
- [35] Zcash website: <https://z.cash/technology/>
- [36] Zk\_SNARK introduction: <https://www.investopedia.com/terms/z/zksnark.asp>
- [37] Zether: Towards Privacy in a Smart Contract World, Benedikt Bunz, Shashank Agrawal, Mahdi Zamani, Dan Boneh: <https://crypto.stanford.edu/~buenz/papers/zether.pdf>
- [38] Eth.Research: <https://ethresear.ch/latest>
- [39] Notes on “Zether: Towards Privacy in a Smart Contract World” : <https://medium.com/@loveshharchandani/notes-on-zether-towards-privacy-in-a-smart-contract-world-6c4333f975d>
- [40] The Etheraut - A Web3 wargame played in Ethereum Virtual Machine - <https://ethernaut.openzeppelin.com/>
- [41] Internal and Private variables in solidity example: <https://stackoverflow.com/questions/70019983/what-is-difference-between-internal-and-private-in-solidity>