ΠΑΝΕΠΙΣΤΗΜΙΟ ΔΥΤΙΚΗΣ ΑΤΤΙΚΗΣ
ΣΧΟΛΗ ΜΗΧΑΝΙΚΩΝ
Τμήμα Ηλεκτρολόγων & Ηλεκτρονικών Μηχανικών
www.eee.uniwa.gr
Θηβών 250, Αθήνα-Αιγάλεω 12241
Τηλ. +30 210 538-1225, Fax. +30 210 538-1226

UNIVERSITY of WEST ATTICA
FACULTY OF ENGINEERING
Department of Electrical & Electronics Engineering
www.eee.uniwa.gr
250, Thivon Str., Athens, GR-12241, Greece
Tel:+30 210 538-1225, Fax:+30 210 538-1226

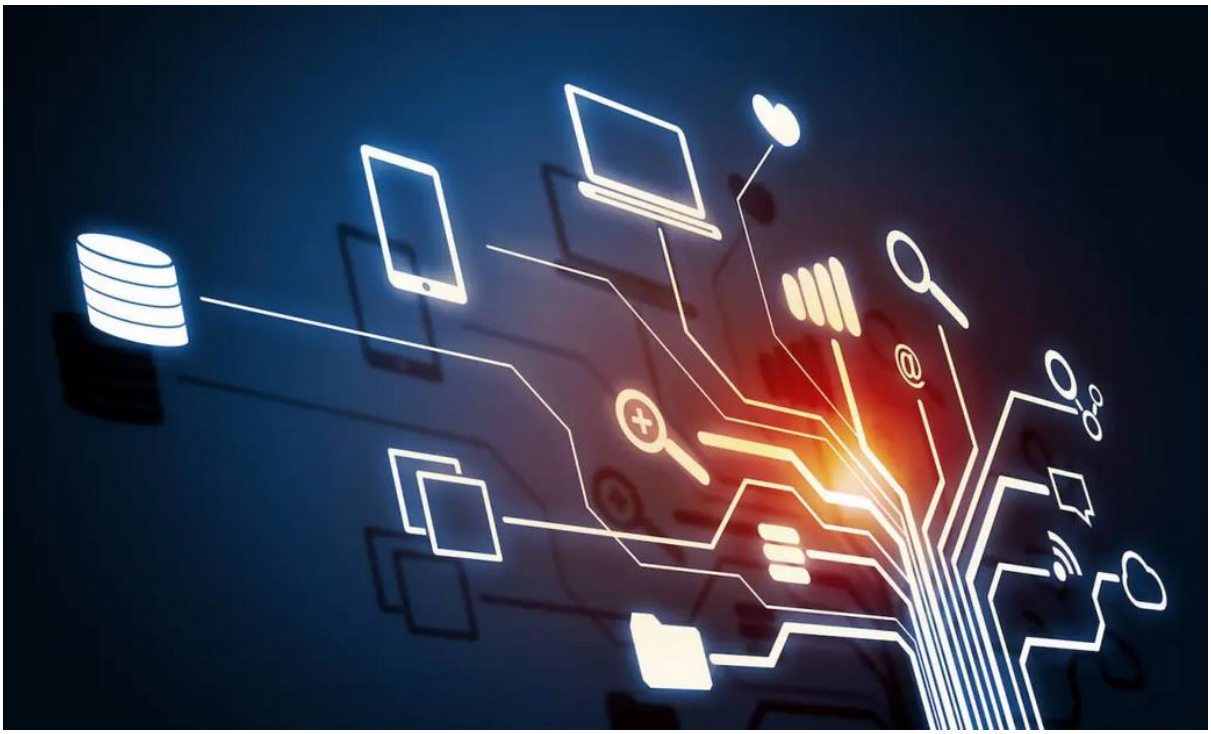
Πρόγραμμα Μεταπτυχιακών Σπουδών
Ηλεκτρικές & Ηλεκτρονικές Επιστήμες μέσω Έρευνας

Master of Science By Research in
Electrical & Electronics Engineering

ΜΕΤΑΠΤΥΧΙΑΚΗ ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

Τίτλος (ελληνικά):

Διαχείριση Πρόσβασης και Εξουσιοδότησης Πραγμάτων και Χρηστών στο Διαδίκτυο των Πραγμάτων με χρήση υπηρεσιών βασισμένων στην τεχνολογία Blockchain



Μεταπτυχιακή Φοιτήτρια: Μαρία Πολυχρονάκη, MSCRES-0052
Επιβλέπων: Χαράλαμπος Ζ. Πατρικάκης, Καθηγητής

ΑΘΗΝΑ-ΑΙΓΑΛΕΩ, 05/09/2022

ΠΑΝΕΠΙΣΤΗΜΙΟ ΔΥΤΙΚΗΣ ΑΤΤΙΚΗΣ
ΣΧΟΛΗ ΜΗΧΑΝΙΚΩΝ
Τμήμα Ηλεκτρολόγων & Ηλεκτρονικών Μηχανικών
www.eee.uniwa.gr
Θηβών 250, Αθήνα-Αιγάλεω 12241
Τηλ. +30 210 538-1225, Fax. +30 210 538-1226

UNIVERSITY of WEST ATTICA
FACULTY OF ENGINEERING
Department of Electrical & Electronics Engineering
www.eee.uniwa.gr
250, Thivon Str., Athens, GR-12241, Greece
Tel:+30 210 538-1225, Fax:+30 210 538-1226

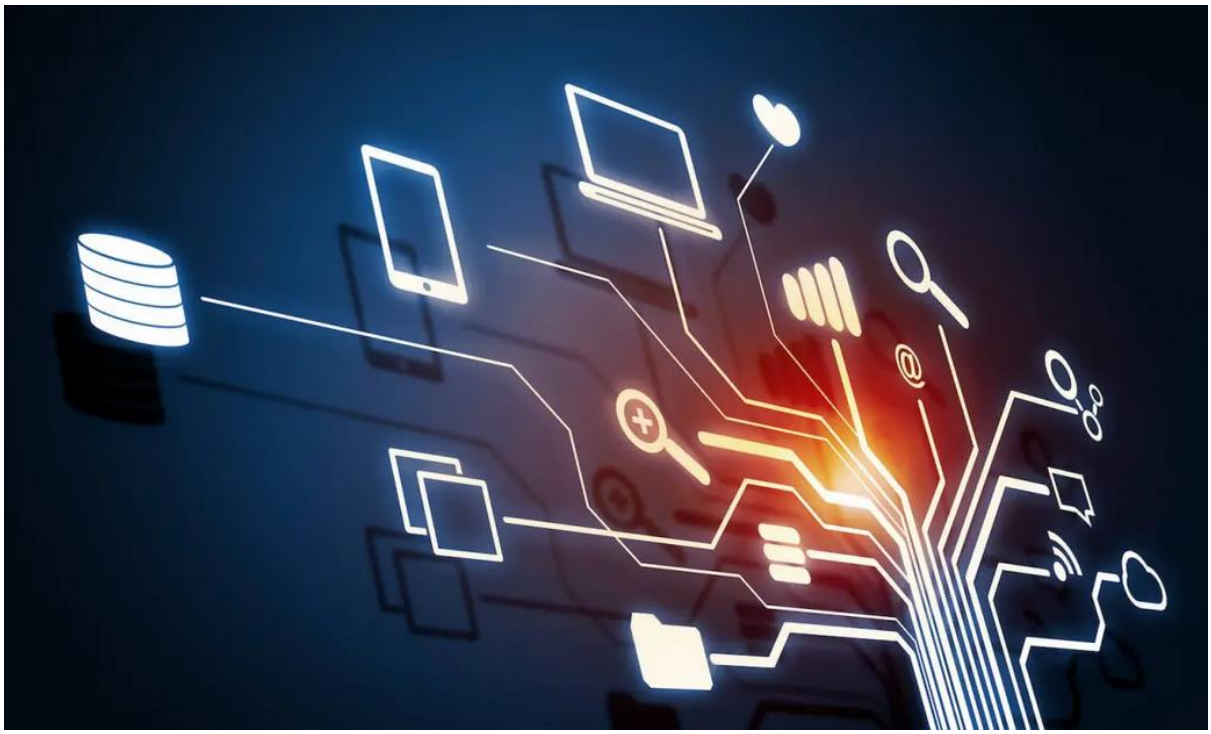
Πρόγραμμα Μεταπτυχιακών Σπουδών
Ηλεκτρικές & Ηλεκτρονικές Επιστήμες μέσω Έρευνας

Master of Science By Research in
Electrical & Electronics Engineering

MSc Thesis

Title (in English):

Access and Authorization Management for the Internet of Things using Services based on Blockchain Technology



Student: Polychronaki, Maria, Registration Number MSCRES-0052
MSc Thesis Supervisor: Patrikakis, Charalambos, Professor

ATHENS-EGALEO, 09-05-2022

Η Μεταπτυχιακή Διπλωματική Εργασία έγινε αποδεκτή, εξετάστηκε και βαθμολογήθηκε από την εξής τριμελή εξεταστική επιτροπή:

Επιβλέπων	Μέλος	Μέλος
Digitally signed by Charalampos Patrikakis Date: 2022.10.10 12:05:53 +03'00'	Digitally signed by Grigorios Kaltsas Date: 2022.10.10 19:53:35 +03'00'	Evangelos Zervas Digitally signed by Evangelos Zervas Date: 2022.10.11 19:42:23 +03'00'
Χαράλαμπος Ζ. Πατρικάκης	Γρηγόριος Καλτσάς	Ευάγγελος Ζέρβας
Καθηγητής	Καθηγητής	Καθηγητής

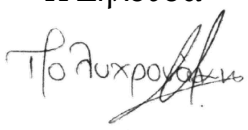
ΔΗΛΩΣΗ ΣΥΓΓΡΑΦΕΑ ΜΕΤΑΠΤΥΧΙΑΚΗΣ ΔΙΠΛΩΜΑΤΙΚΗΣ ΕΡΓΑΣΙΑΣ

Η κάτωθι υπογεγραμμένη Πολυχρονάκη Μαρία του Γεωργίου, με αριθμό μητρώου MSCRES-0052 φοιτήτρια του Προγράμματος Μεταπτυχιακών Σπουδών «Ηλεκτρικές και Ηλεκτρονικές Επιστήμες μέσω Έρευνας» του Τμήματος Ηλεκτρολόγων και Ηλεκτρονικών Μηχανικών της Σχολής Μηχανικών του Πανεπιστημίου Δυτικής Αττικής, δηλώνω ότι:

«Είμαι συγγραφέας αυτής της μεταπτυχιακής διπλωματικής εργασίας και κάθε βοήθεια την οποία είχα για την προετοιμασία της, είναι πλήρως αναγνωρισμένη και αναφέρεται στην εργασία. Επίσης, οι όποιες πηγές από τις οποίες έκανα χρήση δεδομένων, ιδεών ή λέξεων, είτε ακριβώς είτε παραφρασμένες, αναφέρονται στο σύνολό τους, με πλήρη αναφορά στους συγγραφείς, τον εκδοτικό οίκο ή το περιοδικό, συμπεριλαμβανομένων και των πηγών που ενδεχομένως χρησιμοποιήθηκαν από το διαδίκτυο. Επίσης, βεβαιώνω ότι αυτή η εργασία έχει συγγραφεί από μένα αποκλειστικά και αποτελεί προϊόν πνευματικής ιδιοκτησίας τόσο δικής μου, όσο και του Ιδρύματος. Τέλος, βεβαιώνω ότι η εργασία αυτή δεν έχει κατατεθεί στο πλαίσιο των απαιτήσεων για τη λήψη άλλου τίτλου σπουδών ή επαγγελματικής πιστοποίησης πλην του παρόντος.

Παράβαση της ανωτέρω ακαδημαϊκής μου ευθύνης αποτελεί ουσιώδη λόγο για την ανάκληση του πτυχίου μου».

Η Δηλούσα



Πολυχρονάκη Μαρία

ΠΕΡΙΛΗΨΗ

Η χρησιμότητα και τα πλεονεκτήματα του Διαδικτύου των Πραγμάτων είναι πλέον γνωστά και αποδεδειγμένα, ενώ ταυτόχρονα αποτελούν ακόμη και σήμερα ένα από τα κυρίως θέματα του ερευνητικού τομέα αλλά και της Βιομηχανίας. Από την άλλη πλευρά, το Blockchain έχει σχεδόν μονοπωλήσει το ενδιαφέρον της κοινότητας της ψηφιακής οικονομίας. Ωστόσο, ο ακαδημαϊκός τομέας βρίσκεται σε μία διαρκή «πάλη» τα τελευταία χρόνια, προκειμένου να αποδείξει την χρησιμότητα της πρωτοπόρας και ριζοσπαστικής σχεδόν τεχνολογίας σε ορίζοντες πέραν της κρυπτο-οικονομίας και μάλιστα στοχεύοντας το θέμα της ασφάλειας. Τι συμβαίνει όταν προσπαθήσουμε να εμπλέξουμε αυτές τις δύο τεχνολογίες, όχι μόνο σε θεωρητικό επίπεδο αλλά και σε πρακτικό; Πόσο εφαρμόσιμη μπορεί να είναι μία λύση βασισμένη σε blockchain κατά τη διάρκεια της ενσωμάτωσής του σε ήδη λειτουργικά συστήματα Διαδικτύου των Πραγμάτων; Στην εργασία αυτή επιχειρείται να απαντηθούν κάποια ερευνητικά ερωτήματα, τα οποία περιστρέφονται γύρω από το Διαδίκτυο των Πραγμάτων και την τεχνολογία Blockchain, στην προσπάθεια να καλυφθούν ζητήματα ασφάλειας του πρώτου.

ΛΕΞΕΙΣ – ΚΛΕΙΔΙΑ: Διαδίκτυο των Πραγμάτων, Blockchain, Ασφάλεια, Εξουσιοδότηση, Προσβασιμότητα, Σύστημα Διαχείρισης και Ταυτοποίησης, Αποκεντρωμένα Αναγνωριστικά, Έξυπνα Συμβόλαια

ABSTRACT

The Internet of Things nowadays not only has it proven its worth and value, but it has driven itself to integrate with every aspect of our everyday lives and more. Both the academic sector as well as the industry has greatly benefited from its applications, while there is no doubt that it will continue to make our life and work even more flexible as well as more efficient. On the other hand, over the last decade we have seen one technology to emerge and be the bearer of almost a radical and revolutionary transformation happening to the internet and web technologies. Blockchain has brought back the concept of decentralization, while it has successfully solidified itself in the digital economy sector with its very well-known crypto-economy applications. What is the result of the combination of these two radical technologies? Is it possible to integrate a blockchain based solution into already up and running Internet of Things systems in the effort to evade the difficulties of redesigning and redeveloping whole systems and applications? During this master thesis project these questions are to be answered, not only studying theoretical and literature material, but also developing a proof of concept of a practical blockchain-based implementation for the Internet of Things targeting to improve security matters of the latter.

KEYWORDS: Internet of Things, Blockchain, Security, Accessibility, Authorization, Identity and Access Management, Decentralized Identities, Smart Contracts

ΠΙΝΑΚΑΣ ΣΥΜΒΟΛΩΝ-ΑΚΡΩΝΥΜΙΩΝ-ΣΥΝΤΟΜΟΓΡΑΦΙΩΝ

ΑΚΡΩΝΥΜΙΟ	ΟΡΟΛΟΓΙΑ ΣΤΑ ΑΓΓΛΙΚΑ	ΟΡΟΛΟΓΙΑ ΣΤΑ ΕΛΛΗΝΙΚΑ
ΔΤΠ	Internet of Things	Διαδίκτυο των Πραγμάτων
API	Application Programmable Interface	Προγραμματιζόμενη Διεπαφή Εφαρμογής
IAM	Identity & Access Management	Διαχείριση Ταυτοποίησης & Πρόσβασης
DLT	Distributed Ledger Technologies	Τεχνολογίες Κατανεμημένων Κατάστιχων
DID	Decentralized Identifier	Αποκεντρωμένο Αναγνωριστικό
EVM	Ethereum Virtual Machine	Εικονική Μηχανή Ethereum
JSON	Javascript Object Notation	Σημειογραφία Αντικειμένων Javascript
JSON-LD	Json Link Data	Json για Διασυνδεδεμένα Δεδομένα
PKI	Public Key Infrastructure	Υποδομή Ψηφιακού Κλειδιού
RFID	Radio Frequency IDentification	Ταυτοποίηση μέσω Ραδιοσυχνοτήτων
SSI	Self-Sovereign Identity	Αυτοκυριαρχούμενη Ταυτότητα
VC	Verifiable Credential	Επαληθεύσιμο Πιστοποιητικό
ZKP	Zero Knowledge Proof	Απόδειξη Μηδενικής Γνώσης

ΚΑΤΑΛΟΓΟΣ ΕΙΚΟΝΩΝ

ΚΑΤΑΛΟΓΟΣ ΠΙΝΑΚΩΝ

ΠΙΝΑΚΑΣ ΠΕΡΙΕΧΟΜΕΝΩΝ

ΕΙΣΑΓΩΓΗ:
Αντικείμενο, ερευνητικά ερωτήματα και διάρθρωση της εργασίας

Το Διαδίκτυο των Πραγμάτων (ΔτΠ) είναι μία έννοια, η οποία έχει εδραιωθεί στον κόσμο της τεχνολογίας τα τελευταία 20 χρόνια. Η εφαρμογή του έχει βρει υψηλή απήχηση στον καταναλωτικό τομέα, κυρίως σε αυτοματισμούς σπιτιών, επιχειρήσεων αλλά και πόλεων, φορετές συσκευές, στον τομέα της υγείας, στην αυτοκινητοβιομηχανία. Ωστόσο, τα τελευταία χρόνια, η ανάπτυξη της επόμενης γενιάς της βιομηχανίας (industry 4.0 revolution), έχει παρουσιάσει πολλές ανάγκες αλλά και απαιτήσεις ως προς την ενσωμάτωση της αυτοματοποίησης και της διασυνδεσιμότητας. Το ΔτΠ φαίνεται να έχει κάθε προοπτική να ανταποκριθεί σε αυτές τις ανάγκες, ενσωματώνοντας κάθε καινοτόμα τεχνολογία που μπορεί να προσφέρει στην τέταρτη βιομηχανική επανάσταση [1] – [5].

Αξίζει, επίσης, να σημειωθεί, πως η κρίση του COVID-19 από το έτος 2019 έως το τρέχον 2022, ενώ έφερε αδιαπέραστα προβλήματα στην προσφορά πρώτων υλών για την κατασκευή ηλεκτρικών και ηλεκτρονικών ειδών, έχει επίσης επιφέρει μία απότομη αύξηση στην ανάπτυξη και την σχεδίαση καινοτόμων λύσεων, η πλειονότητα των οποίων χρησιμοποιεί με τρόπο άμεσο ή έμμεσο το ΔτΠ [6].

Από την άλλη πλευρά, η τεχνολογία του Blockchain βρίσκεται στο προσκήνιο τόσο ερευνητικά όσο και βιομηχανικά. Ωστόσο, ενώ είναι γνωστό για την κρυπτο-οικονομία, αμφισβητείται από πολλούς για την περαιτέρω χρήση του, καθώς είναι ακόμη αμφίβολο το πώς μπορεί πρακτικά να ενσωματωθεί ώστε να προσφέρει καινοτόμες λύσεις. Το γεγονός ότι το Blockchain κατάφερε να εδραιώσει ένα αποκεντρικοποιημένο σύστημα σε έναν κόσμο βασισμένο σε κεντρικοποιημένες τεχνολογίες και αρχές, έχει προκαλέσει σύγχυση ως προς την χρήση του σε ήδη υπάρχουσες λύσεις [7], [8], [9]. Τα πλεονεκτήματα του είναι αδιαμφησβήτητα, ξεκινώντας από την ίδια την αποκεντρικοποίηση, η οποία μπορεί να προσφέρει επεκτασιμότητα στο ΔτΠ, έως και την βαθμιαία χρήση κρυπτογραφικών τεχνικών για την αύξηση της ιδιωτικότητας αλλά και της ασφάλειας δεδομένων και πρόσβασης [10] – [13].

Με βάση τα παραπάνω, το ερώτημα το οποίο τέθηκε και οδήγησε σε αυτήν την διπλωματική εργασία είναι το πώς μπορεί η τεχνολογία του Blockchain να ενσωματωθεί με ομαλό τρόπο σε ήδη υπάρχοντα συστήματα ΔτΠ. Το ερώτημα αυτό, επιχειρείται να απαντηθεί με την σχεδίαση και την ανάπτυξη μίας λύσης βασισμένης στο Blockchain, η οποία επικεντρώνεται στην ασφάλεια του ΔτΠ και πιο συγκεκριμένα για την διαχείριση πρόσβασης και εξουσιοδότησης πραγμάτων και χρηστών σε ένα περιβάλλον ΔτΠ.

Η εργασία διαρθρώνεται σε 6 κεφάλαια. Το κεφάλαιο 1 περιέχει το θεωρητικό υπόβαθρο το οποίο συγκεντρώθηκε μετά από τη μελέτη σχετικής βιβλιογραφίας σχετικά με την ασφάλεια του διαδικτύου των πραγμάτων αλλά και την τεχνολογία του blockchain. Στη συνέχεια το κεφάλαιο 2 αναλύει τη μεθοδολογία της έρευνας που

επιλέχθηκε να εφαρμοστεί, ενώ τα κεφάλαια 3, 4 και 5 περιγράφουν και παρουσιάζουν τη λύση της διαχείρισης προσβασιμότητας και εξουσιοδότησης με χρήση blockchain, αλλά και τα αποτελέσματα που προκύπτουν μέσα από το πείραμα αυτό. Τέλος, η εργασία κλείνει με το κεφάλαιο 6 όπου αναφέρονται τα τελικά συμπεράσματα του πειράματος και κάποιες προτάσεις από πιθανές επεκτάσεις που μπορεί να πάρει το ανεπτυγμένο θέμα.

ΚΕΦΑΛΑΙΟ 1:
Θεωρητικό πλαίσιο του θέματος – Ανασκόπηση του πεδίου

Στο κεφάλαιο αυτό δίνεται το θεωρητικό υπόβαθρο του αντικειμένου της εργασίας. Μέσα από την θεωρητική ανασκόπηση του ΔτΠ, παρουσιάζονται οι αδυναμίες σχετικές με την ασφάλεια, τις οποίες μπορεί να καλύψει μία λύση βασισμένη στο Blockchain. Επίσης, γίνεται μια βιβλιογραφική επισκόπηση σχετικά με τους διαφορετικούς τρόπους ενσωμάτωσης του Blockchain σε ένα σύστημα ΔτΠ, ενώ, εν συνεχεία, παρουσιάζεται συνοπτικά η κατηγορία τεχνολογιών κατανεμημένων κατάστιχων, της οποίας υποκατηγορία είναι το Blockchain. Το θεωρητικό υπόβαθρο της εργασίας ολοκληρώνεται με μία ανασκόπηση των πρότυπων των Decentralized Identifiers (DID) και Verifiable Credentials (VC), τα οποία χρησιμοποιούνται για την ταυτοποίηση και εξουσιοδότηση

1.1. Το Διαδίκτυο Των Πραγμάτων – ΔτΠ

Το Διαδίκτυο των Πραγμάτων αποτελεί στην ουσία την σύγκλιση πολλών και διαφορετικών τεχνολογιών, πρωτοκόλλων και εφαρμογών [1] ώστε να παραχθεί ένα ενιαίο σύστημα με στόχο όχι μόνο την επικοινωνία μεταξύ μηχανών (αυτοματισμός) αλλά και την αμφίδρομη αλληλεπίδραση μεταξύ ανθρώπων και μηχανών. Ξεκίνησε να αναπτύσσεται σαν ιδέα από την δεκαετία του 1980 όπου διάφοροι ερευνητές και μηχανικοί βρήκαν τρόπους να απλοποιήσουν τη χρήση καθημερινών συσκευών χρησιμοποιώντας την ασύρματη επικοινωνία για τον έλεγχο αυτών. Το 1999, ο Kevin Ashton έδωσε σε αυτήν την ιδέα το όνομα του Διαδικτύου των Πραγμάτων όταν βρήκε τρόπο να διαχειριστεί πιο έξυπνα και αυτοματοποιημένα μία αλυσίδα παραγωγής χρησιμοποιώντας Radio Frequency Identification (RFID) και αισθητήρες.

1.1.1. Ασφάλεια στο Διαδίκτυο των Πραγμάτων

Το ΔτΠ αξιοποιεί και συγκεντρώνει κυρίως τεχνολογίες που χρησιμοποιούν κεντρικοποιημένη προσέγγιση και αρχιτεκτονική (π.χ. μοντέλο client - server). Ωστόσο, παρά την έρευνα και την ανάπτυξη του ΔτΠ τις τελευταίες δύο δεκαετίες, εξακολουθεί να τίθεται το πολύ σοβαρό ζήτημα της ασφάλειας. Έτσι, στον ερευνητικό χώρο γίνονται μελέτες για την εφαρμογή αποκεντρικοποιημένων ή κατανεμημένων προσεγγίσεων και πώς επηρεάζουν τη λειτουργικότητα και την αποδοτικότητα αυτού.

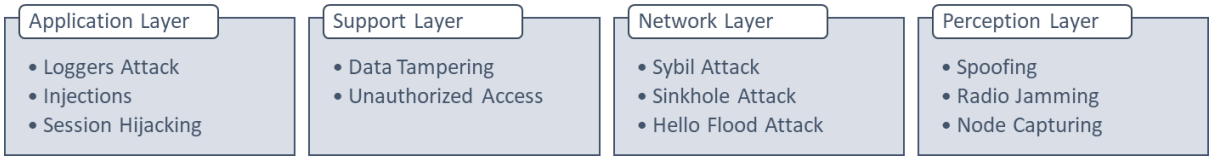
Ο Lin κ.ά. [2], μελετώντας την ενσωμάτωση της Υπολογιστικής στα Άκρα στο ΔτΠ, έκαναν μία ανάλυση για τη σημασία της Ασφάλειας και της Ιδιωτικότητας σε αυτό διακρίνοντας τα εξής στοιχεία Ασφάλειας που πρέπει να το χαρακτηρίζουν:

- *Εμπιστευτικότητα Δεδομένων* αποκλειστικά και μόνο σε εγκεκριμένους χρήστες
- *Ακεραιότητα Δεδομένων* κατά την επικοινωνία δικτύου
- *Συνεχής Διαθεσιμότητα Υπηρεσιών* και δεδομένων κατ' απαίτηση

- *Ταυτοποίηση* εγκεκριμένων συσκευών και εφαρμογών και *Αυθεντικοποίηση* των εισερχόμενων δεδομένων στο δίκτυο ως «νόμιμα»
- *Ιδιωτικότητα Δεδομένων* και έλεγχος αυτών μόνο από εγκεκριμένους χρήστες
- *Εμπιστοσύνη* στην τήρηση των παραπάνω μεταξύ των διαφορετικών «πραγμάτων», διαφορετικών επιπέδων και διαφορετικών εφαρμογών.

Επιπλέον, ο Vashi κ.ά. [3], κατέληξαν στα πέντε σημεία που θα πρέπει να επικεντρώνεται η Ασφάλεια του ΔτΠ:

- *Κρυπτογράφηση* σε όλα τα επίπεδα αρχιτεκτονικής
- *Εμπιστευτικότητα* πρόσβασης στους διακεκριμένους χρήστες
- *Αυθεντικοποίηση* πηγών δεδομένων και από συσκευές
- *Εξουσιοδότηση* ελέγχου συσκευών στα πλαίσια του δικτύου
- *Πιστοποίηση και Έλεγχος Πρόσβασης* μόνο από εγκεκριμένους χρήστες ή συσκευές.



Εικόνα 1: Είδη επιθέσεων στο Διαδίκτυο των Πραγμάτων (Αναδημιουργία από [4])

Η αποτυχία στη διατήρηση αυτών των έξι σημείων μπορεί να έχει ως συνέπεια πολλές και διαφορετικών ειδών επιθέσεις μεταξύ των οποίων οι πιο γνωστές και συνήθεις είναι η Άρνηση Υπηρεσίας (Denial-of-Service), η εμπλοκή Ενδιάμεσου Ατόμου (Man-in-the-Middle) και «Ψάρεμα» Πληροφορίας (Phishing Attack) [1],[2] Παράλληλα, η ετερογένεια που χαρακτηρίζει το ΔτΠ όσον αφορά το συνδυασμό των διαφορετικών πρωτοκόλλων και μηχανισμών που μπορεί να χρησιμοποιηθούν αυξάνει κατακόρυφα και το είδος των επιθέσεων που μπορεί να συμβούν [4], [5].

Συμπερασματικά, μετά από την παραπάνω μελέτη σχετικά με την δομή και τις ανάγκες του ΔτΠ ειδικά στο ζήτημα της ασφάλειας, μπορούμε να διακρίνουμε πως οι κεντρικοποιημένες προσεγγίσεις στην ανάπτυξη συστημάτων ΔτΠ μπορούν μεν να υποστηρίξουν την πολυπλοκότητα και την ετερογένεια που χαρακτηρίζει αυτήν την τεχνολογία, ωστόσο παρουσιάζουν περιορισμούς. Η υποστήριξη υπηρεσιών και υποδομών για αυξημένη ασφάλεια μπορεί να αυξήσει σημαντικά το κόστος ενός συστήματος. Για αυτόν τον λόγο αυτό έρχονται στο προσκήνιο του ερευνητικού τομέα αποκεντρικοποιημένες ή κατανεμημένες προσεγγίσεις προκειμένου να καλυφθούν ζητήματα όπως αυτό της ασφάλειας. Μία τέτοια προσέγγιση αποτελεί η χρήση του Blockchain, του οποίου η κατανόηση και η αξιοποίηση από τον τεχνολογικό κόσμο βρίσκεται ακόμα σε πολύ αρχικά στάδια.

1.1.2. Διαχείριση Ταυτοποίησης και Πρόσβασης – IAM

Ένα σημαντικό βήμα στην κάλυψη των απειλών εκείνων που αφορούν τα ευαίσθητα προσωπικά δεδομένα χρηστών και την πρόσβασή τους, είναι η ανάπτυξη μοντέλων

Διαχείρισης Ταυτοποίησης και Πρόσβασης (Identity and Access Management - IAM). Πρακτικά πρόκειται για την εφαρμογή τεχνολογιών και κανονισμών έτσι ώστε να διασφαλιστεί η σωστή και ασφαλής κατ' απαίτηση πρόσβαση στο πλαίσιο ενός οργανισμού ή ενός συστήματος ΔτΠ. Η χρήση τεχνολογιών όπως η Υποδομή Ψηφιακών Πιστοποιητικών (Pubic Key Infrastructure - PKI) και η Αρχή Έκδοσης Πιστοποιητικών (Certificate Authorities - CA) είναι η πιο συνήθης περίπτωση για την διαχείριση πρόσβασης σε υπολογιστικά περιβάλλοντα.

Το 2019, οι Kettani, Houssain και Carnley [14], με την πρόθεση να φτιάξουν ένα σύστημα IAM με τη χρήση RFID, PKI και Blockchain, παρουσίασαν τις τέσσερις αρχές που πρέπει να διατηρεί ένα μοντέλο IAM ως εξής:

Αυθεντικοποίηση

Διασφαλίζει την ταυτότητα ενός χρήστη ή συσκευής μέσα στα πλαίσια ενός οργανισμού και επιβεβαιώνει τα στοιχεία αυθεντικοποίησής του (π.χ. κωδικός πρόσβασης, δακτυλικό αποτύπωμα, κ.α.) σε υπηρεσίες και πόρους του συστήματος, όταν ζητά πρόσβαση.

Εξουσιοδότηση

Πρόκειται για τον έλεγχο και την επαλήθευση του δικαιώματος ενός χρήστη να χρησιμοποιεί και να επικαλείται τις υπηρεσίες του συστήματος ΔτΠ, μέσω της ταυτότητάς του. Σε αυτό το σημείο λαμβάνουν δράση οι πολιτικές κανόνων που έχουν τεθεί σε ισχύ και προσδιορίζουν τα επίπεδα πρόσβασης ανάλογα την ιεραρχία και τους ρόλους που έχουν προδιαγραφεί.

Διαχείριση Ταυτοτήτων

Είναι το σύστημα που αναλαμβάνει την εισαγωγή αλλά και την διαχείριση μίας ταυτότητας ενός χρήστη ή συσκευής μέσα στο ΔτΠ με την εγγραφή του σε αυτό. Η ταυτότητα αυτή περιλαμβάνει το όνομα του χρήστη, τον κωδικό αυθεντικοποίησής του καθώς και οποιαδήποτε άλλη πληροφορία είναι απαραίτητη για το σύστημα IAM.

Ομοσπονδία Διαχείρισης Ταυτοτήτων

Συνήθως αποτελείται από τρίτες υπηρεσίες που δρουν ανεξάρτητα του συστήματος ΔτΠ και παρέχουν την εγγύηση ή πιστοποίηση ενός χρήστη σε πολλαπλά συστήματα ή οργανισμούς για την πρόσβασή του σε υπηρεσίες και πλατφόρμες. Γνωστά παραδείγματα τέτοιων είναι το Single Sign On (SSO), Open Authorization (OAuth) και το OpenID.

1.2. Κατανεμημένα Καθολικά Κατάστιχα – DLT

Ο όρος Κατανεμημένα Καθολικά Κατάστιχα (Distributed Ledger Technologies - DLT) χρησιμοποιείται για να περιγράψει μία ευρεία κατηγορία τεχνολογικών εφαρμογών που κάνουν χρήση προχωρημένων τεχνικών κρυπτογράφησης προκειμένου να αναπτυχθούν κατανεμημένα δίκτυα που σκοπό έχουν τον συντονισμό και την ομοφωνία μεταξύ άγνωστων μελών χωρίς την παρέμβαση τρίτων [7]**Error! Reference source not found.**. Ο

συντονισμός επέρχεται με την εφαρμογή αλγορίθμων συναίνεσης ενώ τα αποτελέσματα της εφαρμογής αυτής γνωστοποιούνται σε όλους τους ενδιαφερόμενους χρήστες.

Το 2018 οι Ioini και Paul [15] κατέληξαν στην κατηγοριοποίηση των DLT βάσει ήδη υπαρχόντων πλατφορμών και της δομής του κατάστιχου σε τέσσερα διαφορετικά μοντέλα, εκ των οποίων τα δύο έχουν τη μορφή διασυνδεδεμένης λίστας και τα άλλα δύο τη μορφή κατευθυνόμενων και άκυκλων γράφων. Η έρευνα στον τομέα αυτόν δεν έχει προχωρήσει αρκετά ώστε να υπάρξει κάποιο συμφωνημένο πρότυπο στην αρχιτεκτονική DLT συστημάτων, ή ακόμα και στην κατηγοριοποίηση των διαφορετικών ειδών αυτών.

1.2.1. Χαρακτηριστικά των DLT

Σε γενικές γραμμές υπάρχουν κάποια κοινά χαρακτηριστικά και εργαλεία που χρησιμοποιούνται σε ένα DLT είτε πρόκειται για Blockchain είτε όχι.

Κατανεμημένο ή Αποκεντρικοποιημένο Δίκτυο

Το θεμέλιο χαρακτηριστικό των DLT είναι ότι αξιοποιούν αποκλειστικά και μόνο αρχιτεκτονικές ομότιμων κόμβων (Peer-2-Peer) για την υποστήριξη κατανεμημένων ή αποκεντρικοποιημένων δικτύων με σκοπό τον διαμοιρασμό της υπολογιστικής ισχύς τόσο γεωγραφικά όσο και από θέμα διακυβέρνησης του δικτύου.

Συναλλαγές

Οι συναλλαγές είναι το μέσο με το οποίο καταγράφεται κάθε σημαντική πληροφορία που αφορά κάποια ή όλα τα μέλη του δικτύου. Προκύπτουν από την εκτέλεση κώδικα από τους κόμβους του δικτύου, ο οποίος περιέχει και τους κανόνες τήρησης συναλλαγών ανάλογα με το περιβάλλον και τους λόγους κατασκευής ενός DLT.

Διακυβέρνηση και Συναίνεση

Προτού καταγραφούν οποιεσδήποτε αλλαγές που αφορούν το δίκτυο ή τα δεδομένα αυτού, θα πρέπει οι κόμβοι να βεβαιωθούν ότι πρόκειται για αλλαγές που έχουν προέλθει από έγκυρες συναλλαγές. Η χρήση Αλγόριθμων Συναίνεσης (Consensus Algorithms) σε συνδυασμό με το μοντέλο διακυβέρνησης του δικτύου αποτελούν το φίλτρο εμπιστοσύνης μεταξύ των κατά κόρον ισότιμων μελών του.

Κρυπτογράφηση και Ψηφιακά Πορτοφόλια

Τα ψηφιακά πορτοφόλια αποτελούν την διεπαφή των χρηστών ενός DLT με το ίδιο το δίκτυο. Είναι λογισμικό που χρησιμοποιείται για την διατήρηση της ψηφιακής ταυτότητας ενός χρήστη, ενώ παράλληλα εκτελεί την προώθηση μιας συναλλαγής στο δίκτυο. Στην παρούσα φάση ανάπτυξης των DLT, η ψηφιακή ταυτότητα είναι ένα ζευγάρι κλειδιών που έχουν παραχθεί με τη χρήση Υποδομής Ψηφιακών Πιστοποιητικών (PKI) και διατηρούν την ανωνυμία των χρηστών. Οι συναλλαγές πάντα υπογράφονται και κρυπτογραφούνται με το δημόσιο κλειδί του αντίστοιχου χρήστη.

1.3. Πρότυπα Αποκεντρωμένων Ταυτοτήτων

Είναι σημαντικό να τονίσουμε την λειτουργία σε αποκεντρωμένο (ή και κατανεμημένο) περιβάλλον. Σε κεντρικοποιημένα περιβάλλοντα, η δημιουργία αλλά και η επαλήθευση ψηφιακών ταυτοτήτων ή πιστοποιητικών πραγματοποιείται από έναν κεντρικό εξυπηρετητή. Ωστόσο, μεταβαίνοντας σε ένα αποκεντρωμένο ή κατανεμημένο περιβάλλον, απαιτείται όλοι οι συμμετέχοντες κόμβοι (ή τουλάχιστον ένα ποσοστό αυτών) να βρίσκονται σε πλήρη συμφωνία μεταξύ τους σχετικά την κατάσταση των ταυτοτήτων. Το Blockchain θεωρείται μία από τις ιδανικές λύσεις για την υλοποίηση ενός συστήματος διαχείρισης αποκεντρωμένων ταυτοτήτων, με το κατάστιχο να αποτελεί τον πλήρη και αδιάσειστο κατάλογο που περιγράφει τα επίπεδα πρόσβασης των οντοτήτων στις υπηρεσίες και τις εφαρμογές ενός συστήματος ΔτΠ.

1.3.1. Αποκεντρωμένα Αναγνωριστικά – DID

Τα Αποκεντρωμένα Αναγνωριστικά (DIDs) [16] είναι ένα πρότυπο το οποίο αναπτύχθηκε από το World Wide Web Consortium (W3C) προκειμένου να μπορεί να χρησιμοποιηθεί σε ένα αποκεντρωμένο περιβάλλον. Το πρότυπο αυτό περιγράφει το είδος της πληροφορίας που απαιτείται να είναι συγκεντρωμένη σε JSON μορφή ώστε να μπορεί να πιστοποιήσει την οντότητα στην οποία αντιστοιχεί (π.χ. άνθρωπος, αντικείμενο, συσκευή, υπηρεσία). Κάθε DID επιτρέπεται να αντιστοιχεί σε μία και μόνο οντότητα μοναδική σε όλο τον κύκλο ζωής του, ενώ θα πρέπει να χαρακτηρίζεται από τις εξής ιδιότητες:

- ✓ *Αποκέντρωση*: η έκδοσή τους θα πρέπει να πραγματοποιείται από ένα αποκεντρωμένο ή κατανεμημένο σύστημα και όχι από μία ενιαία αρχή ή εξυπηρετητή.
- ✓ *Ανθεκτικότητα*: τα DIDs θα πρέπει να διατηρούνται στην μνήμη του συστήματος χωρίς να απαιτείται κανενός είδους συντήρηση ή ενέργεια από τους διαχειριστές του συστήματος.
- ✓ *Κρυπτογραφικά Επαληθεύσιμα*: θα πρέπει να μπορεί να επαληθευτεί ότι η εκάστοτε οντότητα έχει τον έλεγχο του αντίστοιχου DID με χρήση κρυπτογραφικών μηχανισμών για την διατήρηση της ιδιωτικότητας.
- ✓ *Επιλύσιμα*: θα πρέπει να υπάρχουν μεταδεδομένα τα οποία σχετίζονται με το κάθε DID, προκειμένου να μπορούν να ανακαλυφθούν από άλλες οντότητες ώστε να αλληλοεπιδράσουν.

Εικόνα 2: Δομή και μορφή ενός Αποκεντρωμένου Αναγνωριστικού (DID) **Error! Reference source not found.**

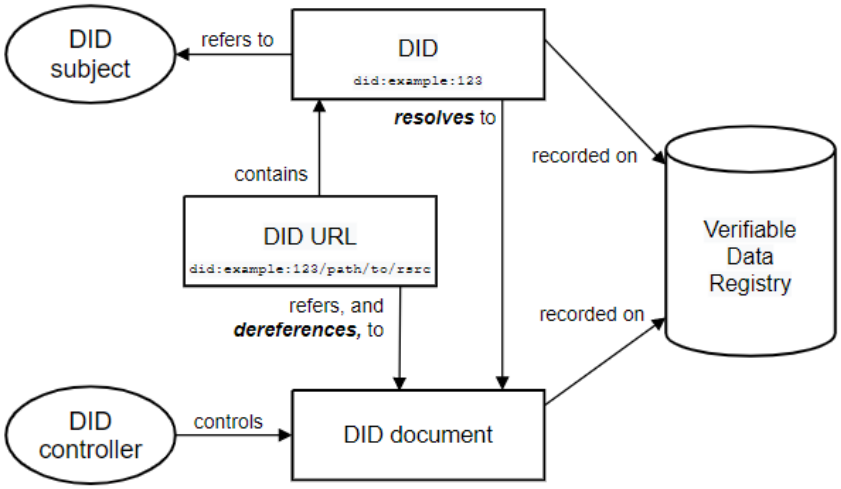
Ένα DID έχει συγκεκριμένη δομή και μορφή (*Εικόνα 2*). Στην ουσία είναι ένα απλό αλφαριθμητικό το οποίο αποτελείται από 3 μέρη χωρισμένα με άνω-κάτω τελείες. Τα μέρη αυτά είναι τα εξής:

- DID Scheme: Το μοντέλο δομής δεδομένων που ακολουθεί.
- DID Method: Η μέθοδος με τη οποία η συγκεκριμένη ταυτότητα δημιουργήθηκε, επιλύεται, ανανεώνεται και απενεργοποιείται
- DID Method-Specific Identifier: Ένα μοναδικό αλφαριθμητικό το οποίο χαρακτηρίζει το συγκεκριμένο DID.

Τα τρία αυτά μέρη θα πρέπει να συμπληρώνουν ένα μοναδικό DID, το οποίο μπορεί να δείξει την ψηφιακή τοποθεσία ενός Εγγράφου Αποκεντρωμένου Αναγνωριστικού (DID Document). Το DID Έγγραφο περιέχει όλες τις απαραίτητες πληροφορίες σχετικά με την οντότητα στην οποία ανήκει (π.χ. άνθρωπος, συσκευή, κ.α.), όπως επίσης και όλους τους τρόπους με τους οποίους μπορεί να ταυτοποιηθεί και αποδείξει ότι έχει τον έλεγχο του συγκεκριμένου DID (π.χ. κρυπτογραφικά κλειδιά). Επίσης, θα πρέπει να είναι διαθέσιμο σε JSON αλλά και JSON-LD μορφή προκειμένου να μπορεί να χρησιμοποιηθεί στα πλαίσια ενός ευρύτερου συστήματος.

Τα DID είναι σχεδιασμένα με τέτοιο τρόπο ώστε να λειτουργούν σε ένα περιβάλλον το οποίο προϋποθέτει την ύπαρξη συγκεκριμένων εξαρτημάτων (*Εικόνα 3*). Αυτά μπορούν να διαχειριστούν και να χρησιμοποιήσουν ένα DID και είναι τα εξής:

- DID Subject: Η οντότητα στην οποία ανήκει και αναφέρεται το DID (π.χ. άνθρωπος, συσκευή, οργανισμός κ.α.).
- DID Controller: Η οντότητα η οποία ελέγχει τις πληροφορίες του αντίστοιχου DID Εγγράφου (π.χ. άνθρωπος, συσκευή, αυτόνομο λογισμικό).
- Verifiable Data Registry: Ένα σύστημα ή δίκτυο, το οποίο μπορεί να λειτουργήσει ως ένα αρχείο δεδομένων και μπορεί να εκτελέσει τις απαραίτητες DID Μεθόδους.



Εικόνα 3: Η αρχιτεκτονική του Αποκεντρωμένου Αναγνωριστικού (DID)

Error! Reference source not found.

Σε πολλές περιπτώσεις, οι οντότητες των DID Subject και DID Controller μπορεί να ταυτίζονται, καθώς η οντότητα την οποία χαρακτηρίζει το DID έχει και την ικανότητα όπως και την δικαιοδοσία να διαχειρίζεται όποια πληροφορία σχετίζεται μαζί της. Ένα τέτοιο παράδειγμα είναι ο άνθρωπος. Ωστόσο υπάρχουν και περιπτώσεις στις οποίες αυτοί οι δύο ρόλοι δεν ταυτίζονται. Οι συσκευές, οι οποίες παράγουν δεδομένα, πολύ συχνά έχουν υπερβολικά χαμηλή υπολογιστική ισχύ για να εκτελέσουν τις διεργασίες μιας συναλλαγής προς ένα Blockchain. Σε αυτές τις περιπτώσεις, θα πρέπει να υπάρχει μία οντότητα, με την δικαιοδοσία να εκτελεί συναλλαγές εκ μέρους των συσκευών αυτών (που αποτελούν και τα DID Subjects) και η οποία είναι ο DID Controller.

1.3.2. Επαληθεύσιμα Πιστοποιητικά - VC

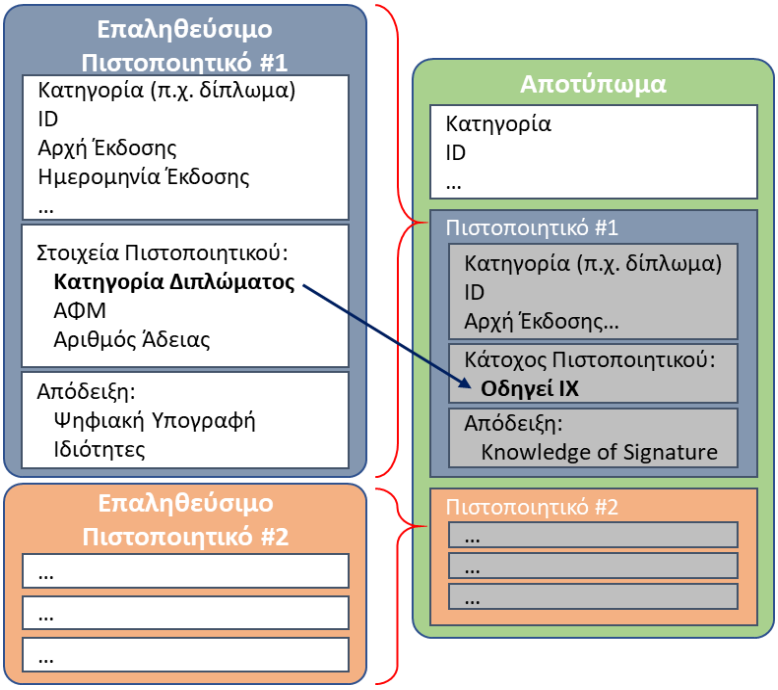
Τα Επαληθεύσιμα Πιστοποιητικά (VCs) [17] είναι ακόμη ένα πρότυπο ανεπτυγμένο από την W3C και αποσκοπεί στον προσδιορισμό των απαραίτητων χαρακτηριστικών που θα πρέπει να έχουν τα ψηφιακά πιστοποιητικά. Η δημιουργία τους είναι συνυφασμένη με τα DIDs, καθώς αποτελούν την κρυπτογραφημένη πληροφορία που αποδεικνύει τις ιδιότητες της οντότητας στην οποία ανήκει το αντίστοιχο DID. Τα VCs παρομοιάζονται με τα φυσικά πιστοποιητικά, όπως η αστυνομική ταυτότητα, ή το δίπλωμα οδήγησης.

Με τον ίδιο τρόπο που ένα δίπλωμα οδήγησης αποδεικνύει την ιδιότητα του οδηγού, έτσι και τα VCs στοχεύουν να προσφέρουν την ίδια βαρύτητα της απόδειξης μίας ιδιότητας, χωρίς ωστόσο να αποκαλύπτονται περεταίρω ευαίσθητες πληροφορίες. Χρησιμοποιώντας ψηφιακά μέσα, το διαδίκτυο, καθώς και τα απαραίτητα εργαλεία κρυπτογραφίας σκοπός είναι να διατηρείται η ιδιωτικότητα και η ασφάλεια της πληροφορίας αλλά και της οντότητας στην οποία αναφέρεται το εκάστοτε VC.

Προκειμένου να επιτευχθεί αυτό, το πρότυπο των VCs διαχωρίζει δύο ξεχωριστές εκδοχές ενός ίδιου πιστοποιητικού, όπως φαίνεται και στο Παράδειγμα της *Εικόνα 4:*

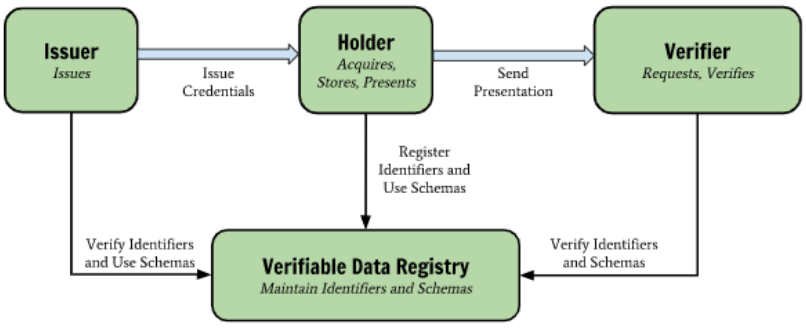
- *Το Επαληθεύσιμο Πιστοποιητικό*, το οποίο περιέχει όλες τις ευαίσθητες (και μη) πληροφορίες που απαιτεί ένα πιστοποιητικό (π.χ. το DID στο οποίο αντιστοιχεί, τον αριθμό διπλώματος οδήγησης, ημερομηνία έκδοσης, αρχή έκδοσης, κ.α.), και
- *Το Αποτύπωμα του Πιστοποιητικού*, το οποίο είναι μία κωδικοποιημένη παρουσίαση του πιστοποιητικού και έχει προκύψει από αυτό μετά από κρυπτογραφικές διαδικασίες που το καθιστούν ψηφιακά αδιαμφησβήτητο. Περιέχει μόνο την ιδιότητα προς απόδειξη, καθώς και την μέθοδο με την οποία μπορεί να επαληθευτεί η κρυπτογράφησή του.

Ο διαχωρισμός αυτός θα πρέπει να προκύπτει με εντολή του κατόχου του αντίστοιχου πιστοποιητικού, ενώ θα πρέπει να έχει και το δικαίωμα να αποκρύπτει την πληροφορία κατά τη δική του βούληση. Έτσι, με αυτόν τον τρόπο και σε συνδυασμό με την κρυπτογράφηση που απαιτείται, εξασφαλίζεται η ιδιωτικότητα του χρήστη.



Εικόνα 4: Παράδειγμα του διαχωρισμού ενός Επαληθεύσιμου Πιστοποιητικού και του Αποτυπώματός του

Το παραπάνω πρότυπο είναι πραγματοποιήσιμο σε ένα συγκεκριμένο περιβάλλον Αποκεντρωμένων Ταυτοτήτων, το οποίο θα πρέπει να διαθέτει τα προγραμματιστικά και υλιστικά εργαλεία τόσο για την έκδοση, όσο και την επαλήθευση αλλά και κτήση των πιστοποιητικών. Το διάγραμμα ροής της *Εικόνα 5*, παρουσιάζει τόσο τους απαραίτητους ρόλους, τους οποίους θα πρέπει να διαθέτει το περιβάλλον των Αποκεντρωμένων Ταυτοτήτων, όπως επίσης και τη διαχείριση των VCs. Όπως φαίνεται, οι απαραίτητοι ρόλοι είναι οι εξής:



Εικόνα 5: Διάγραμμα Ροής Διαχείρισης Επαληθεύσιμων Πιστοποιητικών
Error! Reference source not found.

- ✓ Εκδότης (Issuer): Είναι ο ρόλος του συστήματος που είναι υπεύθυνος για την έκδοση πιστοποιητικών αλλά και την απόδοσή τους στις αντίστοιχες οντότητες, οι οποίες χαρακτηρίζονται από ένα μοναδικό DID. (π.χ. Πανεπιστήμιο, ιδιωτικοί οργανισμοί, το κράτος κ.α.)

- ✓ Κάτοχος (Holder): Είναι ο ρόλος εκείνος, ο οποίος έχει στην κατοχή του το VC και μπορεί να δώσει την εντολή για την παραγωγή ενός Αποτυπώματος (π.χ. άνθρωπος, συσκευή κ.α.).
- ✓ Ελεγκτής (Verifier): Είναι ο ρόλος εκείνος που μπορεί να ζητήσει ένα Αποτύπωμα και να επαληθεύσει την εγκυρότητά του μέσα από το σύστημα (π.χ. λογισμικό και εφαρμογές περιορισμένης χρήσης, προσωπικό ασφαλείας, εργοδότες κ.α.).
- ✓ Verifiable Data Registry: Ακριβώς όπως και στην αρχιτεκτονική των DID, έτσι και για τα VCs απαιτείται η ύπαρξη ενός συστήματος ή δικτύου, το οποίο μπορεί να λειτουργήσει ως ένα αρχείο δεδομένων και μπορεί να εκτελέσει τις απαραίτητες επαληθεύσεις για τα VCs καθώς και τα συσχετιζόμενα DIDs.

1.3.3. Μοντέλο Ιδιόκτητης Ταυτότητας – SSI

Τα συστήματα Διαχείρισης Ταυτοτήτων και Πρόσβασης (IAM) έχουν μία μεγάλη ιστορία εξέλιξης, καθώς ο έλεγχος σχετικά με το ποιος αλλά και σε ποιο βαθμό έχει πρόσβαση σε φυσικούς ή και τεχνολογικούς/ψηφιακούς πόρους ήταν πάντα ένα μείζον ζήτημα διοίκησης. Τα συστήματα IAM (π.χ. Auth, GRNET κ.α.) συνήθως βασίζονται σε κάποιο θεωρητικό αρχιτεκτονικό μοντέλο, το οποίο καθορίζει τους τρόπους με τους οποίους διακρίνονται οι διαφορετικοί ρόλοι (π.χ. εργοδότης, εργαζόμενος, καθηγητής, φοιτητής) αλλά και τον τρόπο με τον οποίο αναγνωρίζεται ο ρόλος τους μέσα στο σύστημα (ψηφιακά κλειδιά, όνομα χρήστη/κωδικός πρόσβασης, κ.α.). Έτσι, μπορούμε να πούμε ότι υπάρχουν δύο αρμοδιότητες στις οποίες πρέπει να ανταποκρίνεται επιτυχώς ένα τέτοιο σύστημα:

- ✓ τη διαχείριση ταυτοτήτων των επιμέρους μελών (π.χ. εγγραφή, αλλαγή στοιχείων, διανομή ρόλων, κ.α.),
- ✓ τη διαχείριση πρόσβασης στους πόρους προς χρήση (π.χ. ιστοσελίδες, APIs, εφαρμογές κ.α.)

Η πρώτη αρμοδιότητα, είναι και αυτή στην οποία μπορεί να ανταποκριθεί πιο αποτελεσματικά το μοντέλο της Ιδιόκτητης Ταυτότητας (SSI). Ωστόσο, αυτό απαιτεί ένα κατανεμημένο ή αποκεντρωμένο περιβάλλον προκειμένου να υλοποιηθεί και κατά συνέπεια αλλάζει και τον τρόπο με τον οποίο μπορεί να πραγματοποιηθεί η διαχείριση πρόσβασης στους διάφορους πόρους. Τη τελευταία δεκαετία, το μοντέλο SSI, έχει τραβήξει την προσοχή των ερευνητών, όχι μόνο λόγω των πλεονεκτημάτων που έχει να προσφέρει στο ΔτΠ αλλά και σε οποιοδήποτε σύγχρονο τεχνολογικό σύστημα που χρήζει ανάγκης για διαχείριση πρόσβασης [18]. Το μοντέλο, ενώ δεν έχει προτυποποιηθεί ακόμη, καταφέρνει να τοποθετήσει τον χρήστη στο επίκεντρο ενός συστήματος IAM.

Ο Allen [19] θεωρεί ότι το μοντέλο SSI είναι το τέταρτο και τελευταίο εξελικτικό στάδιο της ταυτότητας ως έννοια, μετά τα στάδια της κεντρικοποιημένης, ομοσπονδιακής και χρήστο-κεντρικής ταυτότητας. Τα τελευταία τρία αποτυγχάνουν τα προσφέρουν πλήρη ιδιωτικότητα και αυτονομία στον χρήστη σχετικά με την ταυτότητά του και για αυτόν τον λόγο ο Allen επιχειρεί να λύσει αυτό το πρόβλημα με τον προσδιορισμό δέκα αξιωμάτων, τα οποία θα πρέπει να τηρεί ένα IAM σύστημα ανεπτυγμένο βάσει του SSI μοντέλου. Αυτά περιστρέφονται γύρω από το δικαίωμα του

χρήστη να έχει τον πλήρη έλεγχο, πρόσβαση αλλά και δικαίωμα μετακίνησης της ταυτότητάς του. Ταυτόχρονα θα πρέπει να έχει τη δυνατότητα να επιτρέψει ή να απαγορεύσει τον οποιοδήποτε διαμοιρασμό των προσωπικών του δεδομένων, ενώ το δίκτυο, στο οποίο πραγματοποιούνται αυτά, θα πρέπει να χαρακτηρίζεται από πλήρη διαφάνεια στους αλγορίθμους που χρησιμοποιεί και να επιτρέπει στον χρήστη να «ξεχαστεί» αν το επιθυμεί.

Σε αυτό το σημείο, μπορεί κανείς να παρατηρήσει, ότι τα ζητούμενα αξιώματα του Allen μπορούν να διατηρηθούν και να πραγματοποιηθούν πλήρως από τα δύο πρότυπα των DIDs και των VCs που περιεγράφηκαν στις προηγούμενες ενότητες του κεφαλαίου. Παράλληλα, η τεχνολογία Blockchain είναι ίσως η πιο εφαρμόσιμη στην παρούσα φάση ανάπτυξης των DLT.

Αξίζει να σημειωθεί, πως αυτό που καθιστά το Blockchain κατάλληλο για χρήση σε εφαρμογές με ψηφιακές ταυτότητες, είναι το γεγονός πως βασίζεται σε κατανεμημένα ή αποκεντρωμένα συστήματα, ενώ παράλληλα αξιοποιεί κρυπτογραφικούς αλγορίθμους και αλγορίθμους συναίνεσης προκειμένου να διατηρήσει ένα αδιαμφησβήτητο και κωδικοποιημένο αρχείο γεγονότων, το οποίο ωστόσο μπορεί να επαληθευτεί εύκολα. Αυτός ο συνδυασμός προσφέρει τα εξής πλεονεκτήματα, καίρια για ένα σύστημα ταυτοποίησης:

- ✓ Αποφυγή μοναδικού σημείου αποτυχίας (single point of failure) και επιθέσεων τύπου DOS.
- ✓ Δεν μπορεί να πλαστογραφηθεί ή να παραλλαχθεί, ενώ αν μπορεί απαιτεί απίστευτα ποσά επεξεργαστικής ισχύος.
- ✓ Μπορεί να διαβαστεί και να επαληθευτεί μόνο από μέρη του συστήματος που γνωρίζουν τα κατάλληλα κρυπτογραφικά κλειδιά.

1.3.4. Κρυπτογραφία

Από την αρχή της παρούσας εργασίας τονίζεται πολύ έντονα ο σημαντικός ρόλος της κρυπτογραφίας τόσο στην λειτουργία της τεχνολογίας του Blockchain αλλά και στην περίπτωση χρήσης αυτής σε ένα σύστημα Ταυτοποίησης. Συνοπτικά, ας αναφερθεί ότι υπάρχουν δύο βασικά σημεία, στα οποία ήδη το Blockchain κάνει χρήση κρυπτογραφικών αλγορίθμων:

- Αλγόριθμοι Κατακερματισμού (Hashing Algorithms)
- Ασύμμετρη Κρυπτογράφηση (Κρυπτογραφία Ιδιωτικού - Δημόσιου Κλειδιού)

Το πρώτο σημείο αφορά την διατήρηση και συμφωνία ενός κοινού κατάστιχου (ledger) σε όλους τους κόμβους. Οι αλγόριθμοι κατακερματισμού (Hashing) διασφαλίζουν την αλυσιδωτή σχέση των δεδομένων που καταχωρούνται μέσα στο κατάστιχο, καθώς κάθε block πληροφορίας περιέχει το hash ολόκληρου του ιστορικού του [20], [21].

Το δεύτερο σημείο είναι αυτό του *ψηφιακού πορτοφολιού* που αλληλοεπιδρά με το αντίστοιχο Blockchain δίκτυο και είναι το λογισμικό εκείνο υπεύθυνο για την

δημιουργία ενός γεγονότος, το οποίο πρέπει να καταγραφεί ως *συναλλαγή*. Ένας χρήστης θα πρέπει πρώτα να έχει αποκτήσει το ζεύγος του ιδιωτικού και δημόσιου κλειδιού του από το εκάστοτε δίκτυο, προκειμένου να μπορέσει να εκτελέσει συναλλαγές. Ασύμμετρη κρυπτογράφηση χρησιμοποιείται ώστε να κωδικοποιηθούν αλλά και να υπογραφούν οι συναλλαγές από τους χρήστες και να περάσουν από έλεγχο εγκυρότητας από το δίκτυο πριν καταχωρηθούν.

Το ψηφιακό πορτοφόλι είναι το μόνο σημείο στο οποίο ένα χρήστης διατηρεί τα κλειδιά του, τα οποία αποτελούν και ένα είδος ταυτότητας των χρηστών στις υπάρχουσες εφαρμογές Blockchain. Για τον λόγο αυτό, είναι ένα από τα πιο κατάλληλα εργαλεία για να διαφυλάξει μία ψηφιακή ταυτότητα αποκεντρωμένου τύπου. Ωστόσο, μένει ακόμα να προσδιορίσουμε πώς, μέσω ενός ψηφιακού πορτοφολιού που περιέχει ευαίσθητες πληροφορίες, θα μπορέσει ένας χρήστης να παράγει τα επαληθεύσιμα πιστοποιητικά. Λύση σε αυτό το πρόβλημα δίνουν οι Αλγόριθμοι Απόδειξης Μηδενικής Γνώσης (Zero Knowledge Proofs - ZKPs), μία από τις εφαρμογές των οποίων θεωρείται πως είναι σε συστήματα ταυτοποίησης [22].

1.4. Η Ενσωμάτωση του Blockchain στο ΔτΠ

Λόγω της πολυπλοκότητας του ΔτΠ αλλά και της ετερογένειάς του, συνεχίζουν και υπάρχουν πολλές και υπαρκτές απειλές, οι οποίες μπορεί να έχουν σοβαρές συνέπειες τόσο στην λειτουργία του συστήματος όσο και στους χρήστες του εκθέτοντας τα ευαίσθητα δεδομένα που τους αφορούν. Το μόνο που έχει αποτελέσει την βέλτιστη λύση είναι η χρήση τρίτων και έμπιστων υπηρεσιών για την ύπαρξη εμπιστοσύνης μέσα στο πλαίσιο λειτουργίας του εκάστοτε συστήματος.

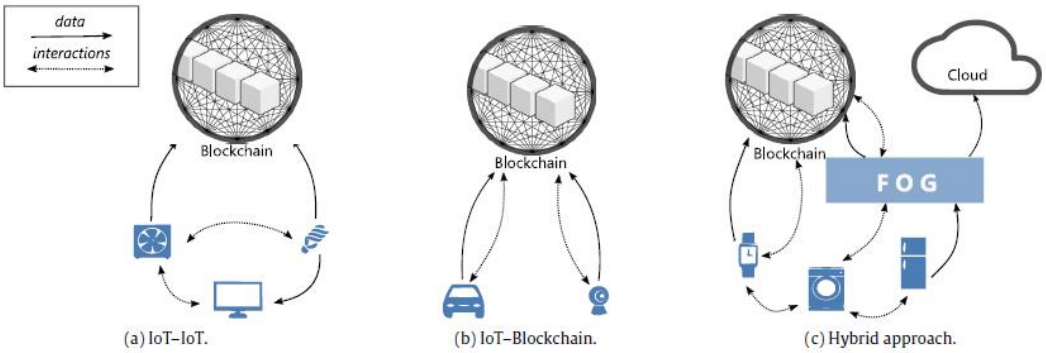
Από την άλλη πλευρά, η απότομη ανάπτυξη των DLT και του Blockchain βασίζεται στην ανάγκη της σύγχρονης τεχνολογίας να αποκεντρικοποιηθεί, όπως επίσης και την απομάκρυνση των ενδιάμεσων και των τρίτων επιτρέποντας στους χρήστες να αλληλοεπιδρούν μεταξύ τους όσο πιο άμεσα γίνεται. Βασική προϋπόθεση για την πραγματοποίηση αυτών είναι η χρήση εξαιρετικά ισχυρών και προχωρημένων τεχνικών κρυπτογράφησης ώστε να διασφαλιστούν η ανωνυμία και η ασφάλεια των χρηστών.

Δεν υπάρχει αμφιβολία πως ο συνδυασμός αυτών των δύο τεχνολογιών μπορούν να αποφέρουν βελτιστοποίηση στην υπάρχουσα οργάνωση και λειτουργία του ΔτΠ. Το σημείο της Διαχείρισης Ταυτοποίησης και Πρόσβασης είναι η τομή της βελτιστοποίησης αυτής, χωρίς αυτό να σημαίνει πως τα DLT δεν μπορούν να προσφέρουν περεταίρω πλεονεκτήματα.

Στο [9] παρουσιάζονται τρεις διαφορετικοί τρόποι με τους οποίους μπορεί να ενσωματωθεί το Blockchain στο ΔτΠ, εστιάζοντας στην αλληλεπίδραση του πρώτου με το δεύτερο.

Ο πρώτος τρόπος (Εικόνα 3a) εμπλέκει το Blockchain καθαρά ως μία βάση δεδομένων (που προσφέρει όμως ακεραιότητα δεδομένων), ενώ οι συσκευές επικοινωνούν ανεξάρτητα μεταξύ τους και το Blockchain αποτελεί μία Υπηρεσία για αυτές. Ο δεύτερος τρόπος (Εικόνα 3b) είναι το Blockchain να αποτελεί το μέσο Επικοινωνίας μεταξύ των συσκευών προκειμένου να πραγματοποιείται πλήρης έλεγχος και Ταυτοποίηση

συσκευών. Ο τρίτος τρόπος (Εικόνα 3c) αποτελεί μία υβριδική λύση των προηγούμενων δύο, κατά την οποία ρυθμίζεται ποιες είναι οι κατάλληλες αλληλεπιδράσεις των συσκευών με το Blockchain προκειμένου να διασφαλιστεί η ασφάλεια χωρίς να επηρεαστεί σημαντικά η απόδοση και η ταχύτητα του συστήματος.



Εικόνα 6: Σημεία ενσωμάτωσης Blockchain στο ΔτΠ [9]

1.5. Η Διαχείριση Ταυτοποίησης και Πρόσβασης με χρήση Blockchain

Οι Rayna και λοιποί στο [9]**Error! Reference source not found.**, στοχεύοντας στην ανάλυση των αδυναμιών αλλά και των προτερημάτων της ένωσης των δύο τεχνολογιών παρουσιάζουν τα σημεία του ΔτΠ που μπορούν να βελτιστοποιηθούν με τη ενσωμάτωση των δύο τεχνολογιών. Πιο συγκεκριμένα, κάποια από αυτά τα σημεία είναι η ταυτοποίηση συσκευών και χρηστών μέσω της αποκεντρικοποίησης του συστήματος, η αυτονομία των συσκευών ώστε να επικοινωνούν μεταξύ τους χωρίς τη χρήση κεντρικών διακομιστών καθώς και η ασφάλεια όσον αφορά την επικοινωνία συσκευών εκμεταλλευόμενο την ισχυρή κρυπτογράφηση που χρησιμοποιεί το Blockchain.

Επιπλέον, μετά από πειραματικές διατάξεις και αρχιτεκτονικές που έχουν σχεδιαστεί με σκοπό την διαχείριση πρόσβασης συσκευών σε ένα σύστημα ΔτΠ, όπως αυτές στα [23] και [24], είναι φανερό πως η χρήση Blockchain με κατάλληλο τρόπο μπορεί να εμποδίσει κάποια συχνά ήδη επιθέσεων. Κάποια από αυτά είναι η Άρνηση Υπηρεσίας (DoS / DDoS attack), όπως επίσης και Επίθεση Ιχνηλάτησης (Link Attack) κατά την οποία εκτίθενται τα προσωπικά δεδομένα ενός ανώνυμου χρήστη καθώς ο επιτιθέμενος επιχειρεί να ανιχνεύσει το δημόσιο κλειδί αυτού σε όλη την αλυσίδα δεδομένων, για να ανακαλύψει την πραγματική του ταυτότητα.

Σε διατάξεις όπως αυτές που βασίζονται σε κατανεμημένες αρχιτεκτονικές, το πλήθος των κόμβων που το αποτελούν μπορεί να αποτελέσει πλεονέκτημα, έναντι της κεντρικοποιημένης αρχιτεκτονικής όπου η επεκτασιμότητα του συστήματος είναι πεπερασμένη. Όσο περισσότεροι κόμβοι έχουν αντίγραφο της κρυπτογραφημένης πληροφορίας τόσο πιο δύσκολο είναι να παραποιηθεί αυτή από κάποιον κακόβουλο χρήστη.

Στο [25], ο Novo παρουσιάζει μία αναλυτική αρχιτεκτονική και υλοποίηση αυτής όπου επιτυγχάνεται η Διαχείριση Πρόσβασης των συσκευών με τη χρήση Management Hubs. Αυτά αποτελούν ουσιαστικά το Blockchain δίκτυο ενώ παράλληλα δρουν και ως

διεπαφές για τις υπόλοιπες συσκευές. Τόσο σε αυτήν την περίπτωση όσο και σε αυτές των [23] και [25], αναφέρεται πως κατά την εκτέλεση των συναλλαγών από τις συσκευές υπάρχουν καθυστερήσεις τόσο λόγω της φύσης του κατανεμημένου δικτύου έναντι του κεντρικοποιημένου (της τάξης των milisecond), όσο και λόγω των αλγόριθμων ομοφωνίας προκειμένου να επιβεβαιώσουν την εγκυρότητα αυτών (της τάξης των second).

Ωστόσο, επισημαίνουν πως ο πρώτος τύπος καθυστερήσεων μπορεί να γίνει αποδεκτός λόγω των πλεονεκτημάτων ασφάλειας που προσφέρει το Blockchain. Ο δεύτερος τύπος καθυστερήσεων μπορεί να βελτιωθεί με την χρήση διαφορετικών μοντέλων διακυβέρνησης και αλγορίθμων ομοφωνίας μεταξύ των κόμβων του δικτύου.

ΚΕΦΑΛΑΙΟ 2:
Μεθοδολογία της έρευνας

2.1. Σκοπός της Έρευνας – Ερωτήματα

Σκοπός της έρευνας που οδήγησε στην ολοκλήρωση της διπλωματικής εργασίας, ήταν να διερευνηθεί *«αν είναι εφικτό αλλά και καρποφόρο να εφαρμοστούν οι Τεχνολογίες Κατανεμημένων Καθολικών Κατάστιχων σε λύσεις βασισμένες στο ΔτΠ προκειμένου να ενισχυθεί ή ασφάλεια, την αξιοπιστία αλλά και η ιδιωτικότητα»*. Το εν λόγω ερώτημα, γεννήθηκε μέσα από τις πολλαπλές αναφορές που υπάρχουν τόσο στο διαδίκτυο όσο και έναν μεγάλο αριθμό επιστημονικών δημοσιεύσεων, ισχυριζόμενες πως το blockchain μπορεί να ενισχύσει τις παραμέτρους αυτές στο ΔτΠ. Ωστόσο, μπορεί κανείς να παρατηρήσει το οξύμωρο, πως σχεδόν μία δεκαετία μετά την εμφάνιση της τεχνολογίας Blockchain, η τελευταία δεν έχει εδραιωθεί επίσημα σε κανένα άλλο πεδίο πέραν αυτού της κρυπτο-οικονομίας.

Προκειμένου το ερώτημα αυτό να απαντηθεί, υπήρξαν τρία στάδια εκ των οποίων τα πρώτα δύο παρουσιάστηκαν στις Τεχνικές Αναφορές Α και Β του συγκεκριμένου ΠΜΣ. Στη συνέχεια, μετά από την επιτυχή εξέταση αυτών, διαμορφώθηκε και σχεδιάστηκε μία λύση, η οποία επιχειρεί με τρόπο πρακτικό να αποδείξει το εν λόγω ερώτημα αλλά και να το επεκτείνει, απαντώντας ακόμη στο δευτερεύον ερώτημα του «*αν και πώς είναι εφικτό να εφαρμοστεί και να ενσωματωθεί το Blockchain ομαλά σε ήδη υπάρχοντα συστήματα ΔτΠ*».

Τεχνικές Αναφορές Α & Β

Οι τεχνικές αναφορές, οι οποίες προηγήθηκαν της παρούσας διπλωματικής εργασίας, έθεσαν το θεωρητικό υπόβαθρο του Κεφαλαίου 1, αλλά και σημαντικό μέρος της σχεδίασης του Κεφαλαίου 3. Αποτέλεσαν τη βάση μίας σημαντικής βιβλιογραφικής ανασκόπησης, όχι μόνο των θεωρητικών αρχών που διέπουν τόσο το Διαδίκτυο των Πραγμάτων αλλά και των Τεχνολογιών Κατανεμημένων Κατάστιχων, και πολύ περισσότερο της τεχνολογίας Blockchain.

Αξίζει να αναφερθεί πως κατά τη διάρκεια της Τεχνικής Αναφοράς Β', έγινε ο σχεδιασμός μίας τεχνικής αρχιτεκτονικής, η οποία έδειξε από τη μία πλευρά την αξία και την χρησιμότητα των προτύπων DID και VC, αλλά και των Zero Knowledge Proof (ZKP) αλγορίθμων. Ωστόσο από την άλλη πλευρά, το σενάριο υλοποίησης (έκδοση διπλώματος οδήγησης νέου οδηγού και επαλήθευση από έξυπνα οχήματα) δεν βασιζόταν σε κάποιο σύστημα ΔτΠ. Έτσι, ενώ η τεχνική αρχιτεκτονική παραμένει η ίδια στη βάση της και παρουσιάζεται αναλυτικά στο επόμενο κεφάλαιο, πρέπει να αναφερθεί πως το σενάριο και το πλαίσιο υλοποίησης ανασχηματίστηκαν, προκειμένου να προσαρμοστούν στα πλαίσια του ΔτΠ.

2.2. Μέθοδοι Έρευνας που επιλέχθηκαν

Η κύρια μέθοδος που χρησιμοποιήθηκε για την διεκπεραίωση της επικείμενης έρευνας είναι η πειραματική, με σκοπό να προκύψει μία απόδειξη αρχής (proof-of-concept). Η τελευταία επιχειρεί να αποδείξει, όχι μόνο ότι όντως μπορεί η τεχνολογία του blockchain να χρησιμοποιηθεί με ωφέλιμο τρόπο για το ΔτΠ, αλλά και να επιδείξει έναν τρόπο με τον οποίο μπορεί να επιτευχθεί η ενσωμάτωση της τεχνολογίας με ομαλότητα σε ήδη ανεπτυγμένα συστήματα ΔτΠ.

Σε αυτό το σημείο χρήζει διευκρίνισης το γεγονός πως πρόκειται για μία έρευνα μεικτού τύπου και όχι αμιγώς ποιοτικού ή ποσοτικού τύπου. Η μέθοδος του πειράματος που επιλέχθηκε κατευθύνει στην ποσοτική έρευνα, καθώς είναι ευρέως γνωστό ότι στην επιστήμη της μηχανικής η έρευνα ολοκληρώνεται συνήθως με την εξαγωγή στατιστικών και αριθμητικά μετρήσιμων αποτελεσμάτων. Ωστόσο, στην προκειμένη περίπτωση, ναι μεν θα χρησιμοποιηθεί μία ποσοτικού τύπου μέθοδος, τα ερευνητικά ερωτήματα δε, είναι ταυτόχρονα ανοιχτού αλλά και κλειστού τύπου, ενώ τα συμπεράσματα που θα παρουσιαστούν δεν προκύπτουν βάσει στατιστικών ή αριθμητικών δεδομένων.

Αναλυτικότερα, τα ερευνητικά ερωτήματα αναλύονται στα εξής:

1. Είναι εφικτό αλλά και καρποφόρο να εφαρμοστούν οι Τεχνολογίες Κατανεμημένων Καθολικών Κατάστιχων σε λύσεις βασισμένες στο ΔτΠ προκειμένου να ενισχυθεί ή ασφάλεια, την αξιοπιστία αλλά και η ιδιωτικότητα;
2. Είναι εφικτό να εφαρμοστεί και να ενσωματωθεί το Blockchain ομαλά σε ήδη υπάρχοντα συστήματα ΔτΠ;
3. Αν η απάντηση είναι θετική στο ερώτημα 2, τότε πώς είναι εφικτό να πραγματοποιηθεί;

Παρατηρούμε, ότι τα πρώτα δύο ερωτήματα είναι κλειστού τύπου και η απάντησή τους είναι δυαδικής φύσης, ανεξάρτητα του ενδεχομένου να προκύψουν προϋποθέσεις για την απάντησή τους. Αντίθετα, η τελευταία ερώτηση είναι ανοιχτού τύπου και πιθανώς μπορεί να απαντηθεί με ποικίλους τρόπους, ένας από τους οποίους επιχειρείται να δοθεί στην παρούσα εργασία. Συμπερασματικά, η έρευνα της παρούσας διπλωματικής εργασίας, χαρακτηρίζεται ως μεικτού τύπου.

2.3. Συλλογή Δεδομένων και Παράμετροι

Η συνέχεια της έρευνας που παρουσιάζεται στα επόμενα κεφάλαια περιλαμβάνεται από την σχεδίαση και θεμελίωση μίας τεχνικής αρχιτεκτονικής για ένα ενδιάμεσο σύστημα διαχείρισης της πρόσβασης και της εξουσιοδότησης συσκευών και χρηστών. Τα αποτελέσματα της εφαρμογής αυτού, δεν περιλαμβάνουν αριθμητικές μετρήσεις, αλλά εξαγωγή ποιοτικών συμπερασμάτων. Συμπερασματικά, δεν υπάρχει συλλογή αριθμητικών δεδομένων, αλλά ποιοτικών, ενώ αφορούν παραμέτρους σχετικές με την ασφάλεια του ΔτΠ, οι οποίες αναφέρθηκαν και αναλύθηκαν στο Κεφάλαιο 1.

Πιο συγκεκριμένα, θα εξεταστεί η επιρροή (θετική, αρνητική ή ουδέτερη) της τεχνολογίας Blockchain στην πειραματική διάταξη ΔτΠ που θα παρουσιαστεί στα επόμενα κεφάλαια. Η επιρροή αυτή θα εξεταστεί με βάση τους εξής άξονες ασφάλειας:

- *Ταυτοποίηση* εγκεκριμένων συσκευών και εφαρμογών
- *Έλεγχος και Διαχείριση Πρόσβασης μόνο σε* εγκεκριμένους χρήστες
- *Διαχείριση Εξουσιοδότησης* ελέγχου συσκευών στα πλαίσια του δικτύου
- *Ακεραιότητα Δεδομένων*
- *Ιδιωτικότητα Δεδομένων*
- *Αυθεντικοποίηση* πηγών δεδομένων
- *Συνεχής Διαθεσιμότητα* Δεδομένων κατ' απαίτηση
- *Συνεχής Διαθεσιμότητα Υπηρεσιών* κατ' απαίτηση
- *Επεκτασιμότητα*
- *Κρυπτογράφηση* Δεδομένων και Επικοινωνίας

Έχοντας τις παραμέτρους αυτές ως άξονες, θα παρουσιαστεί μία λύση βασισμένη σε Blockchain, η οποία θα είναι εφικτό να ενσωματωθεί σε ήδη υπάρχοντα συστήματα ΔτΠ, ενώ ταυτόχρονα επιδρά θετικά στην πλειονότητα των παραμέτρων, αν όχι όλων.

2.4. Βιβλιογραφική Αναζήτηση

Το πρώτο βήμα για την απάντηση των ερωτημάτων που τέθηκαν καθ' όλη τη διάρκεια της παρούσας έρευνας, ήταν μία ανασκόπηση της βιβλιογραφίας πάνω στο θέμα του Blockchain, της ασφάλειας του ΔτΠ αλλά και των τρόπων ενσωμάτωσης του πρώτου, προκειμένου να αντιμετωπιστούν δυσκολίες που υπάρχουν στην δεύτερη. Αναλυτικότερα, τα αποτελέσματα της αναζήτησης που παρουσιάζονται βασίστηκαν στις λέξεις και φράσεις κλειδιά: "IoT and blockchain", "identity and access management" ή επί γραμματικά "iam", "Blockchain", καθώς και κάθε δυνατού συνδυασμού μεταξύ αυτών.

Οι πηγές βιβλιογραφίας που χρησιμοποιήθηκαν ήταν:

- **Google Scholar**: μία ελεύθερα προσβάσιμη μηχανή αναζήτησης ιστού που εντοπίζει το πλήρες κείμενο ή τα μεταδεδομένα των ακαδημαϊκών δημοσιεύσεων που έχουν υλοποιηθεί σε μία εκτεταμένη σειρά επιστημονικών και τεχνικών εκδόσεων, σε παγκόσμια κλίμακα.
- **IEEE Xplore**: μία από τις μεγαλύτερες ψηφιακές βιβλιοθήκες για αξιόπιστη έρευνα. Περιλαμβάνει περιοδικά, συνέδρια, πρότυπα, ηλεκτρονικά βιβλία και εκπαιδευτικά μαθήματα και αριθμεί περισσότερα από 5 εκατομμύρια έγγραφα.
- **ResearchGate**: ένας ιστότοπος κοινωνικής δικτύωσης για ακαδημαϊκά προφίλ με στόχο την κοινή χρήση ακαδημαϊκών δημοσιεύσεων.
- **Αποθετήριο arXiv**: ένα ηλεκτρονικό αποθετήριο ελεύθερης πρόσβασης για επιστημονικές μελέτες πριν την ακαδημαϊκή δημοσίευση τους, για θέματα που αφορούν τα μαθηματικά, φυσική, αστρονομία, επιστήμη υπολογιστών, βιολογία, στατιστικά, και μαθηματικό-οικονομικές μελέτες.
- **ACM Digital Library**: αποτελεί την πιο ολοκληρωμένη βάση δεδομένων στον κόσμο με άρθρα πλήρους κειμένου και επιστημονική βιβλιογραφία που καλύπτει τους υπολογιστές και την τεχνολογία της πληροφορίας.

- **ScienceDirect**: παρέχει έναν μεγάλο αριθμό δημοσιεύσεων πλήρους κειμένου σε τομείς όπως οι φυσικές επιστήμες και η μηχανική, οι επιστήμες υγείας, κοινωνικές και ανθρωπιστικές επιστήμες.

2.5. Εργαλεία που χρησιμοποιήθηκαν

Μετά από μελέτη σχετικής βιβλιογραφίας [26] – [29], οι πιο δημοφιλείς πλατφόρμες για την ανάπτυξη Blockchain εφαρμογών πέραν του τομέα της κρυπτό-οικονομίας και πιο συγκεκριμένα στο ΔτΠ, αποτελούν οι πλατφόρμες Ethereum, Hyperledger Fabric, IBM Blockchain και R3 Corda. Ωστόσο, την πιο ολοκληρωμένη λύση την δίνει το Ethereum καθώς προσφέρει εργαλεία από άκρη σε άκρη για την ανάπτυξη μιας ολοκληρωμένης blockchain λύσης. Τελικά, για αυτό το πείραμα επιλέχθηκε το οικοσύστημα εργαλείων του Ethereum. Τα επιμέρους εργαλεία που χρησιμοποιήθηκαν είναι τα εξής:

- Ethereum Blockchain Platform[1]: Αποτελεί ίσως την πιο δημοφιλή επιλογή Blockchain πλατφόρμας. Υπάρχουν πολλά δίκτυα βασισμένα στο Ethereum, ωστόσο η αλυσίδα που διατηρεί το κρυπτονόμισμα με την πραγματική του ανταλλακτική αξία είναι μία, αυτή στο κυρίως δίκτυο (main net). Το Ethereum προσφέρει όλα τα απαραίτητα εργαλεία για την ανάπτυξη έξυπνων συμβολαίων σε γλώσσα solidity, καθώς επίσης και εργαλεία για την ανάπτυξη αποκεντρωμένων εφαρμογών.
- Solidity[2]: Η solidity είναι μία γλώσσα αρκετά ευέλικτη και ανεπτυγμένη, ώστε αφενός έχει αποδειχθεί Turing-complete, αφετέρου δίνει στον προγραμματιστή τη δυνατότητα να χτίζει ολόκληρη προγραμματιστική λογική, η οποία σκοπό έχει τη διαχείριση της κατάστασης του κατάστιχου. Στην ουσία, εκμεταλλευόμαστε τη δυνατότητα των έξυπνων συμβολαίων να διατηρούν τον έλεγχο και την διαχείριση μεταβλητών, ωστόσο η τρέχουσα αλλά και η ιστορική αναδρομή αυτών βρίσκεται στο κατάστιχο.
- Remix IDE[3]: Είναι ένα ολοκληρωμένο περιβάλλον ανάπτυξης ανοιχτού κώδικα που μπορεί να χρησιμοποιηθεί για τη σύνταξη, τη μεταγλώττιση και τον εντοπισμό σφαλμάτων κατά την ανάπτυξη έξυπνων συμβολαίων σε solidity.
- Metamask[4]: Πρόκειται για ένα ανοιχτού κώδικα πορτοφόλι κρυπτονομισμάτων και μπορεί να χρησιμοποιηθεί για την αποθήκευση κλειδιών μόνο για κρυπτονομίσματα Ethereum. Μπορούμε να πούμε ότι λειτουργεί σαν γέφυρα μεταξύ των κανονικών προγραμμάτων περιήγησης και του blockchain Ethereum.

[1] https://ethereum.org/en/
[2] https://soliditylang.org/
[3] https://remix-project.org/
[4] https://metamask.io/

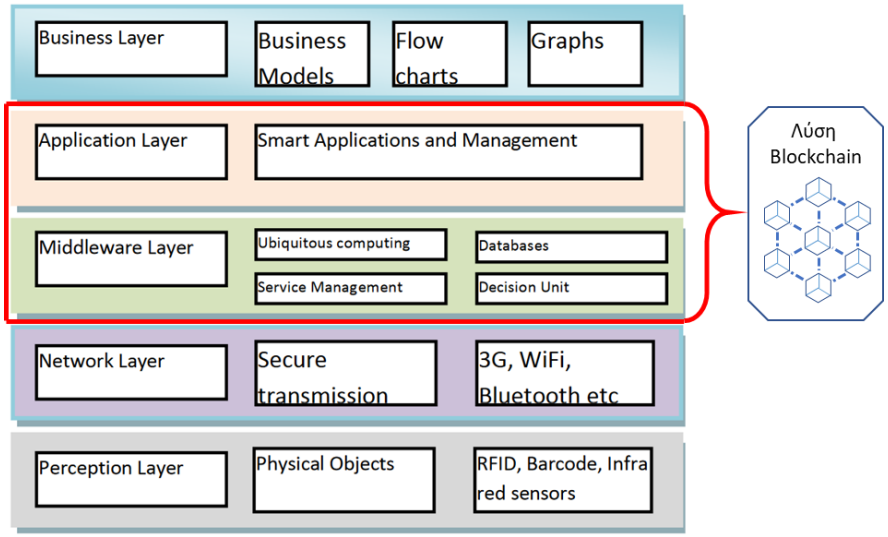
ΚΕΦΑΛΑΙΟ 3:
Η προτεινόμενη μέθοδος – Θεμελίωση, Σχεδίαση, Ανάπτυξη

Στο κεφάλαιο αυτό θα παρουσιαστεί η αρχιτεκτονική ενός συστήματος εξουσιοδότησης βασισμένη στην τεχνολογία Blockchain, κατάλληλη να ενσωματωθεί σε συστήματα ΔτΠ και που θα έχει ως σκοπό την επαλήθευση πιστοποιητικών χρηστών αλλά και συσκευών. Οι επόμενες υπό-ενότητες περιλαμβάνουν την περιγραφή της μεθοδολογίας που ακολουθήθηκε στην παρούσα έρευνα, αλλά και της προτεινόμενης αρχιτεκτονικής μέσω γραφικών διαγραμμάτων.

3.1. Πλαίσιο Υλοποίησης

Το πλαίσιο υλοποίησης μέσα στο οποίο έχει σχεδιαστεί η παρούσα λύση είναι αυτό ενός ήδη υλοποιημένου συστήματος ΔτΠ. Αξίζει να αναφερθεί πως υπάρχει μεγάλη ετερογένεια αλλά και ποικιλία όσον αφορά πρωτόκολλα, συσκευές και εφαρμογές που μπορούν να αξιοποιηθούν σε ένα ΔτΠ σύστημα. Ωστόσο, για τις ανάγκες της προς παρουσίαση λύσης, υποθέτουμε την ύπαρξη ενός συστήματος 5 επιπέδων (*Εικόνα 7*), ξεκινώντας από κάτω προς τα πάνω με το Φυσικό επίπεδο (συσκευές αισθητήριες αλλά και) και φτάνοντας στο επίπεδο Business, όπου περιέχονται όλα τα μοντέλα και η επιχειρησιακή λογική ενός συστήματος.

Είναι σημαντικό το γεγονός ότι είναι αναγκαία η χρήση διαδικτύου προκειμένου να επιτευχθεί η οποιαδήποτε σύνδεση με κάποιο δίκτυο Blockchain. Στόχος της λύσης εξουσιοδότησης συσκευών με χρήση Blockchain είναι να μπορεί να ενσωματωθεί τουλάχιστον στο επίπεδο Εφαρμογών (Application) και Ενδιάμεσων (Middleware), τα επίπεδα δηλαδή που οφείλουν να έχουν πρόσβαση στο διαδίκτυο.



Εικόνα 7: Αρχιτεκτονική 5 Επιπέδων (βασισμένο από το [30])

Στην περίπτωση, ωστόσο, όπου χρησιμοποιείται το Διαδίκτυο και από το Φυσικό επίπεδο και το Δικτυακό επίπεδο, δεν υπάρχει καμία ένσταση στο να ενσωματώσουν και αυτά την εν λόγω λύση. Επιπλέον, καθώς η λύση στοχεύει τον έλεγχο της εξουσιοδότησης συσκευών και υπηρεσιών, οποιοσδήποτε φορέας προσφέρει αυτό το σύστημα ΔτΠ μπορεί να επιλέξει αν θα ενσωματώσει την λύση και στο επίπεδο Business, με το να περιορίζει τους χρήστες που τη χρησιμοποιούν, ανάλογα με κάποιο πακέτο υπηρεσιών. Παρ' όλ' αυτά, από την τεχνική σκοπιά του ζητήματος, τα έξυπνα συμβόλαια που θα παρουσιαστούν δεν προνοούν για την ενσωμάτωση στο επίπεδο Business.

Στην ουσία, η παρούσα λύση αποσκοπεί να μετατρέψει τις διαδικτυακές και κινητές εφαρμογές που ήδη έχουν σχεδιαστεί και υλοποιηθεί για συστήματα ΔτΠ, σε μερικώς αποκεντρωμένες εφαρμογές (D-Apps), καθώς οι κεντρικοποιημένες λειτουργίες δεν θα αλλαχθούν, προκειμένου να κάνουν άμεση χρήση έξυπνων συμβολαίων. Με αυτόν τον τρόπο, η εφαρμογή, και κατ' επέκταση οι χρήστες αλλά και οι συσκευές με πρόσβαση στο διαδίκτυο, θα έχουν τη δυνατότητα να επαληθεύσουν και να επαληθευτούν μέσα από ένα αποκεντρωμένο σύστημα διαχείρισης πρόσβασης.

Πρακτικά, τα αποτελέσματα που αναμένονται από αυτό το πείραμα είναι η παροχή τριών τύπων αποδείξεων από την βασισμένη σε blockchain υλοποίηση. Αυτές είναι:

1. Απόδειξη συμμετοχής στο συγκεκριμένο σύστημα ΔτΠ (είτε συσκευής, είτε χρήστη)
2. Απόδειξη εξουσιοδότησης ελέγχου συστήματος ΔτΠ (κυρίως όταν πρόκειται για διαχείριση του ίδιου του συστήματος – π.χ. αλλαγή ρυθμίσεων)
3. Απόδειξη ελέγχου συσκευής από χρήστη ή άλλη συσκευή

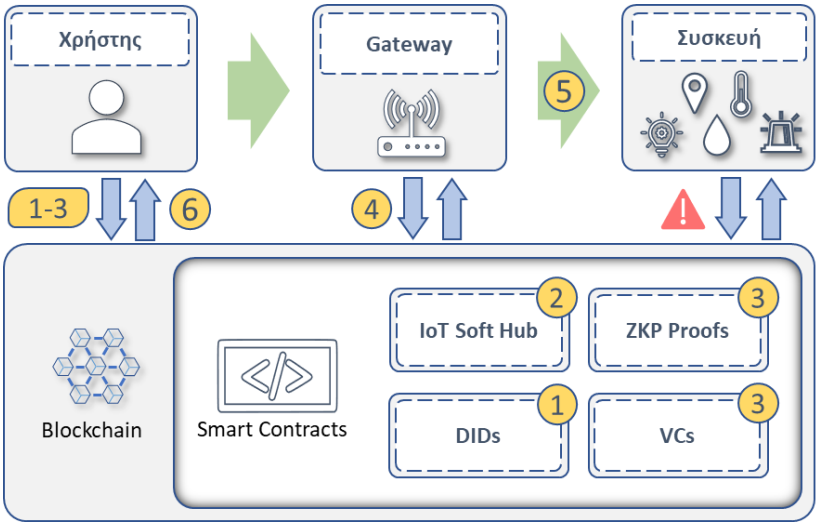
3.2. Τεχνική - Αρχιτεκτονική

Σε συνέχεια της περιγραφής του σεναρίου παραπάνω, η *Εικόνα 8* παρουσιάζει ένα διάγραμμα ροής με τα βήματα τα οποία πρόκειται να ακολουθηθούν στην περίπτωση του πειράματος της εργασίας. Υποθέτουμε πως υπάρχει ένα σύστημα ΔτΠ με Blockchain, το οποίο για λόγους απλότητας στην προκειμένη περίπτωση, αποτελείται από τα εξής 4 μέρη:

A. Χρήστες: Πρόκειται για τους τελικούς χρήστες, οι οποίοι αλληλεπιδρούν με το σύστημα είτε για να δώσουν μία εντολή (αλληλεπίδραση με συσκευή ή υπηρεσία), είτε για να ειδοποιηθούν για κάποιο συμβάν (π.χ. υπέρβαση κατωφλίου κάποιας μέτρησης).
B. Gateway (Πύλες): Πρόκειται για υπηρεσίες διαδικτύου που συμμετέχουν (π.χ. βάση δεδομένων, επεξεργασία δεδομένων, κ.α.), όπως επίσης και συσκευές που κατηγοριοποιούνται στο επίπεδο των Ενδιάμεσων. Συνήθως λειτουργούν ως «πύλες» για συσκευές χαμηλού επιπέδου που δεν μπορούν να ολοκληρώσουν κάποια διεργασία μόνες τους (π.χ. αποστολή raw δεδομένων).

C. Συσκευή: Οποιαδήποτε συσκευή που δρα ως αισθητήρας ή ενεργοποιητής και συμμετέχει με οποιονδήποτε τρόπο στο σύστημα, είτε δίνοντας είτε λαμβάνοντας πληροφορία. Είναι σημαντικό να σημειωθεί πως δεν έχουν όλες οι συσκευές τη δυνατότητα επικοινωνίας με το Blockchain απευθείας. Σε αυτήν την περίπτωση θα πρέπει να οι πύλες με τις οποίες επικοινωνούν να εκτελέσουν τις αντίστοιχες διεργασίας εκ μέρους τους.

D. Blockchain: Αποτελείται από ένα σύνολο έξυπνων συμβολαίων, τα οποία εδρεύουν και εκτελούνται σε ένα συγκεκριμένο δίκτυο Blockchain, ενώ ταυτόχρονα περιέχει και το κατανεμημένο κατάστιχο (distributed ledger). Στο τελευταίο καταγράφεται ολόκληρο το ιστορικό εξουσιοδότησης με τρόπο κρυπτογραφημένο ώστε μόνο όσοι έχουν την απαραίτητη άδεια μπορούν να διαβάσουν από αυτό.



Εικόνα 8: Διάγραμμα Ροής Προτεινόμενης Λύσης

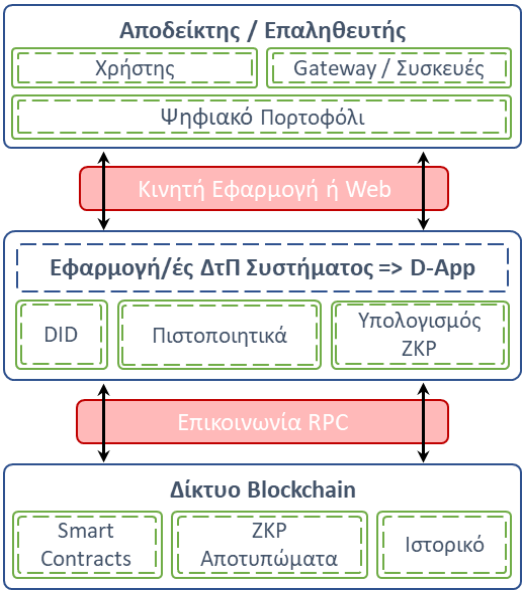
Επιπρόσθετα, το διάγραμμα ροής της Εικόνας 8 δείχνει και την πορεία των βημάτων που θα παρουσιαστούν αργότερα, για την ολοκλήρωση του πειράματος. Τα βήματα αυτά αποσκοπούν στην δημιουργία των απαραίτητων Αποκεντρωμένων Αναγνωριστικών για τα τρία μέρη του συστήματος ΔτΠ (Χρήστης, Gateway, Συσκευές), τη δημιουργία ενός SoftHub, το οποίο αποτελεί σημασιολογική οντότητα μέσα στο Blockchain, και τέλος τη δημιουργία των απαραίτητων πιστοποιητικών και ZKPs για την λειτουργία της εξουσιοδότησης και επαλήθευσης. Πιο αναλυτικά, τα βήματα είναι τα εξής:

1. Ο χρήστης δημιουργεί για τον εαυτό του ένα μοναδικό Αποκεντρωμένο Αναγνωριστικό DID.
2. Ο χρήστης δημιουργεί ένα SoftHub, το οποίο αποτελεί μία σημασιολογική οντότητα μέσα στο Blockchain και συγκρατεί την πληροφορία των συμμετεχόντων χρηστών και συσκευών του ΔτΠ συστήματος. Στην ουσία αποτελεί την ψηφιακή ταυτότητα και περιγραφή ενός πραγματικού συστήματος, μέσα στο Blockchain.
3. Ο χρήστης δημιουργεί DIDs για τις συσκευές και θέτει ρόλους, τόσο στις συσκευές, όσο και σε άλλους χρήστες που συμμετέχουν στο SoftHub (αφού φυσικά έχουν οι

ίδιοι ήδη δημιουργήσει τα δικά τους DIDs). Αυτό επιτυγχάνεται μέσω των Πιστοποιητικών VC και την παραγωγή ZKP όταν είναι απαραίτητο.

4. Αν υπάρχουν Gateways, τα οποία αντιπροσωπεύουν περιορισμένης δυνατότητας συσκευές, επαναλαμβάνουν τη διαδικασία του βήματος 3 εκ μέρους αυτών των συσκευών.
5. Οποτεδήποτε πρέπει να ολοκληρωθεί κάποια διεργασία μέσω του συστήματος ΔτΠ, τότε το μέλος που δέχεται την εντολή θα πρέπει να επαληθεύει την εξουσιοδότηση του μέλους που την δίνει.
6. Τέλος, όπου είναι απαραίτητο, η εκτέλεση συγκεκριμένων συναρτήσεων κώδικα των Έξυπνων Συμβολαίων, μπορεί να επιστρέψει ειδοποίηση στον ή στους χρήστες προς ενημέρωσή τους για την νέα κατάσταση του συστήματος (π.χ. κάποιος μη εξουσιοδοτημένος χρήστης προσπάθησε να εκτελέσει μία συνάρτηση ή η κατάσταση κάποιας συσκευής άλλαξε).

Η τεχνική αρχιτεκτονική με τα επιμέρους μέρη που συμμετέχουν και την επικοινωνία μεταξύ τους φαίνεται στην *Εικόνα 9*. Μπορεί κανείς να παρατηρήσει ότι η αρχιτεκτονική αυτή έχει τη μορφή οριζόντιων επιπέδων και αποτελείται από 3 από αυτά, ενώ ενδιάμεσά τους παρεμβάλλεται ο τρόπος επικοινωνίας. Πιο συγκεκριμένα, ξεκινώντας από κάτω προς τα πάνω, τα επίπεδα αυτά είναι τα εξής:



Εικόνα 9: Τεχνική Αρχιτεκτονική Λύσης

1. Δίκτυο Blockchain:
2. Βασικές Εφαρμογές και Υπηρεσίες Ελέγχου του ΔτΠ συστήματος
3. Αποδείκτες και Επαληθευτές (συμμετέχουσες οντότητες στο σύστημα ΔτΠ)

Δίκτυο Blockchain

Το πρώτο επίπεδο αποτελείται από ένα δίκτυο blockchain, το οποίο περιέχει τα κατάλληλα smart contracts, καθώς επίσης και το κατανεμημένο κατάστιχο που μπορεί να λειτουργήσει ως ιστορικό κινήσεων. Στο τελευταίο καταγράφονται όλες οι κινήσεις ως συναλλαγές, μαζί και τα ZKP αποτυπώματα, τα οποία μπορούν να επαληθεύσουν μία αληθή δήλωση. Επίσης ένα δίκτυο Blockchain χαρακτηρίζεται από το είδος του (δημόσιο/ιδιωτικό), την προσβασιμότητα σε αυτό (permissioned / permissionless), όπως επίσης και από τον αλγόριθμο συναίνεσης που χρησιμοποιεί, καθώς αυτό μπορεί να επηρεάσει σημαντικά το χρόνο και την απόδοση εκτέλεσης συναλλαγών.

Βασικές Εφαρμογές και Υπηρεσίες Ελέγχου του ΔτΠ Συστήματος

Στο δεύτερο επίπεδο ανήκουν όλες οι εφαρμογές ή/και υπηρεσίες, οι οποίες συμμετέχουν στο σύστημα ΔτΠ και μπορούν να παραμετροποιηθούν ή να επαναπρογραμματιστούν

ως ένα σημείο. Σκοπός αυτού του επιπέδου είναι να ενσωματώσει την χρήση του πρώτου επιπέδου, δίνοντας την δυνατότητα στις εφαρμογές και τις υπηρεσίες, με τις οποίες έρχονται σε επαφή οι χρήστες και οι συσκευές, να κάνουν κλήσεις των smart contracts. Αυτό θα έχει ως άμεσο αποτέλεσμα τη δυνατότητα επαλήθευσης εξουσιοδότησης των χρηστών και των συσκευών πριν την εκτέλεση κάποια εντολής στα πλαίσια του ΔτΠ. Από την άλλη πλευρά, μέσω αυτού του επιπέδου θα μπορούν και οι χρήστες να αποκτούν και να παρέχουν εξουσιοδότηση σε άλλους χρήστες και συσκευές. Αξίζει να αναφερθεί πως σε αυτό το επίπεδο ανήκει και η διαχείριση των DID, των πιστοποιητικών αλλά και η "off-chain" διαδικασία του υπολογισμού των ZKP.

Αποδείκτες Επαληθευτές

Στο τρίτο και ανώτερο επίπεδο ανήκουν όλες οι συσκευές και οι χρήστες που αλληλεπιδρούν με κάποιο τρόπο με το σύστημα ΔτΠ, είτε δίνοντας κάποια εντολή, είτε δίνοντας δεδομένα, είτε λαμβάνει μία εντολή προς εκτέλεση. Σε κάθε περίπτωση, κάθε οντότητα θα πρέπει αντιστοιχίζεται με ένα λογαριασμό, καταγεγραμμένο στο αντίστοιχο δίκτυο Blockchain. Ο τρόπος με τον οποίο μπορεί μία οντότητα να αποκτήσει αλλά και να διατηρήσει στην κατοχή του τον έλεγχο του λογαριασμού του είναι να χρησιμοποιήσει ένα ψηφιακό πορτοφόλι. Το τελευταίο ουσιαστικά αποθηκεύει τόσο τον αριθμό λογαριασμού, όσο και το δημόσιο και ιδιωτικό κλειδί που αντιστοιχούν σε αυτό, αλλά είναι και το λογισμικό εργαλείο που δίνει την τελική εντολή εκτέλεσης μίας συναλλαγής, μετά την κρυπτογραφική υπογραφή αυτής με το αντίστοιχο ιδιωτικό κλειδί.

3.3. Έξυπνα Συμβόλαια (Smart Contracts)

Τα έξυπνα συμβόλαια αποτελούν την πηγή της λογικής μίας blockchain λύσης. Είναι σημαντικό να αναφερθεί πως δεν υποστηρίζουν όλα τα δίκτυα blockchain την εκτέλεση προγραμματιζόμενων έξυπνων συμβολαίων. Ωστόσο, ακόμα και τα δίκτυα τα οποία χρησιμοποιούνται αποκλειστικά στον τομέα της κρυπτο-οικονομίας, χρειάζονται τα βασικά έξυπνα συμβόλαια για την εκτέλεση συναλλαγών κρυπτονομισμάτων.

Στην προκειμένη περίπτωση, τα έξυπνα συμβόλαια αναπτύσσονται με τέτοιο τρόπο ώστε να δίνουν στους χρήστες τη δυνατότητα να σχηματίζουν ένα σύστημα διαχείρισης πρόσβασης και ταυτοποίησης στα πλαίσια του ΔτΠ, ενώ ταυτόχρονα αναπτύσσονται συναρτήσεις επαλήθευσης αυτού. Αυτό επιτυγχάνεται με την ανάπτυξη τεσσάρων έξυπνων συμβολαίων, των οποίων ο πλήρης κώδικας βρίσκεται στο Παράρτημα της παρούσας εργασίας. Τα συμβόλαια αυτά είναι τα εξής:

DidFactory.sol

Το πρώτο έξυπνο συμβόλαιο είναι υπεύθυνο για την δημιουργία των Αποκεντρωμένων Αναγνωριστικών (Decentralized Identifiers). Κάθε οντότητα που συμμετέχει ή χρησιμοποιεί την λύση αυτή, είναι προαπαιτούμενο να διαθέτει ένα προσωπικό DID που αντιστοιχεί αυστηρά σε ένα και μόνο ένα λογαριασμό ψηφιακού πορτοφολιού. Το DID σύμφωνα με το θεωρητικό υπόβαθρο του Κεφαλαίου 1, πρέπει να έχει συγκεκριμένη δομή αλλά και πληροφορία, η οποία μπορεί να είναι δημόσια χωρίς να εκθέτει τον κάτοχό του.

Συνοπτικά, η πληροφορία την οποία διαχειρίζεται το συμβόλαιο αυτό είναι η παρακάτω:

- ***struct Did {scheme, method, path, property, registrationNo, did}***
- ***mapping (address => uint) identityToAccount***
- ***mapping (address => bool) accountIsRegistered***

Η παραπάνω *struct Did* δομή παρουσιάζει την πληροφορία της οντότητας DID, με τα επιμέρους πεδία που περιέχει. Χαρακτηριστικό είναι το πεδίο registrationNo, το οποίο είναι μοναδικό για κάθε DID και παράγεται μέσω τυχαιότητας. Επιπλέον το συμβόλαιο διαχειρίζεται και δύο πίνακες *mapping*, οι οποίοι κρατάνε την αντιστοίχιση:

- του λογαριασμού ψηφιακού πορτοφολιού με το αντίστοιχο Did αντικείμενο και
- το αν ο συγκεκριμένος λογαριασμός διαθέτει ήδη κάποιο Did.

Τέλος, η μοναδική βασική συνάρτηση που περιέχει το έξυπνο συμβόλαιο είναι αυτή για τη δημιουργία νέου DID, ενώ απαιτεί και μία παράμετρο ως είσοδο, η οποία αντιπροσωπεύει την ιδιότητα της αντίστοιχης οντότητας (administrator/user – device – softhub).

IothubFactory.sol

Στην ίδια λογική με το προηγούμενο έξυπνο συμβόλαιο, λειτουργεί και το δεύτερο, με τη διαφορά ότι δεν επικεντρώνεται στα DID αλλά στις οντότητες των softhubs που αναφέρθηκαν νωρίτερα στην τεχνική αρχιτεκτονική της λύσης, αλλά και των συσκευών. Η λογική της πληροφορίας αυτής είναι, για κάθε σύστημα ΔτΠ που ενσωματώνει τη λύση αυτή, αντιστοιχίζεται ένα softhub, ενώ ταυτόχρονα για κάθε συσκευή που υπάρχει στο πραγματικό σύστημα ΔτΠ, αντιστοιχίζεται και ένα *struct Device* αντικείμενο.

Πιο συγκεκριμένα, η πληροφορία αυτού του συμβολαίου είναι η εξής:

- ***struct Device {deviceID, hubID, deviceType}***
- ***struct Iothub {DID, ownerDID, name}***
- ***mapping (uint => uint[]) deviceHubHistory***
- ***mapping (address => uint) ownerToHub***
- ***mapping (string => uint) rnToDev***
- ***mapping (uint => uint) devToHub***

Παρατηρούμε τα δύο *structs Device και Iothubs*, τα οποία χρησιμοποιούνται για την αντιστοίχιση των softhubs και των συσκευών, ενώ στη συνέχεια υπάρχουν και τρεις πίνακες *mapping*, οι οποίοι χρησιμοποιούνται για:

- την αντιστοίχιση της κάθε συσκευής με το softhub στο οποίο ανήκει.
- την αντιστοίχιση των μοναδικών σειριακών κωδικών τύπου registration number των συσκευών με το αντίστοιχο DID της κάθε συσκευής
- τη διατήρηση του ιστορικού κάθε συσκευής για τα softhubs στα οποία έχει συμμετάσχει

Αξίζει να σημειωθεί πως ο λόγος για τον που χρησιμοποιείται ενός χρήστη να χρησιμοποιήσει ένα QR code με αυτήν την πληροφορία, παρά να θυμάται το μοναδικό

DID του, το οποίο είναι αρκετά ψηφία σε μήκος. Ωστόσο, αυτή η πληροφορία είναι ιδιωτική και προσβάσιμη μόνο από το ίδιο το έξυπνο συμβόλαιο. Έτσι, δεν μπορεί να υπάρξει διαρροή της πληροφορίας αυτής, που θεωρείται ευαίσθητη, προς τα έξω.

Τέλος, υπάρχουν δύο συναρτήσεις, οι οποίες δημιουργούν τις οντότητες των softhub και device, αφού φυσικά δημιουργήσουν και το μοναδικό DID τους κάθε φορά.

DeviceManagement.sol

Το τρίτο έξυπνο συμβόλαιο είναι υπεύθυνο για τις διεργασίες διαχείρισης των συσκευών. Έχοντας θέσει όλες τις απαραίτητες δομές δεδομένων και πίνακες στα προηγούμενα έξυπνα συμβόλαια, το τρίτο από τα τέσσερα αποτελείται μόνο από τέσσερις μεθόδους. Αυτές είναι:

- ***function claimDevice (_deviceRN)***
- ***function removeFromHub (_deviceRN)***
- ***function changeHub (_deviceRN, _targetOwner)***
- ***function claimTransferedDevice (_deviceRN)***

Με τις δύο πρώτες συναρτήσεις και είσοδο τον μοναδικό σειριακό κωδικό της συσκευής, μπορεί μόνο ο ιδιοκτήτης ενός softhub να προσθέσει ή να αφαιρέσει μία συσκευή από αυτό. Οι επόμενες δύο συναρτήσεις σκοπό έχουν την μεταφορά κάποια συσκευής βάσει σειριακού αριθμού σε κάποιο άλλο softhub.

Ο λόγος για τον οποίο έχουν φτιαχτεί ξεχωριστές συναρτήσεις είναι για ασφάλεια. Αν για κάποιο λόγο υπάρξει καθυστέρηση χρόνου ανάμεσα στην αφαίρεση και την προσθήκη, μπορεί να μεσολαβήσει κάποιος χρήστης κακόβουλα και να διεκδικήσει μία ή πολλές συσκευές «χτυπώντας» σειριακούς κωδικούς συσκευών. Αντ' αυτού, αυτό που συμβαίνει με το δεύτερο ζευγάρι μεθόδων είναι ότι ο διαχειριστής του πρώτου softhub, αφού αφαιρέσει τη συσκευή, την δεσμεύει για διεκδίκηση από κάποιον συγκεκριμένο χρήστη, με βάση τον μοναδικό αριθμό λογαριασμού του. Στη συνέχεια, η συσκευή μπορεί να προστεθεί αποκλειστικά και μόνο στο softhub του οποίου ιδιοκτήτης είναι ο λογαριασμός για τον οποίο έχει δεσμευτεί η συσκευή.

Verification.sol

Τέλος, το τέταρτο και τελευταίο έξυπνο συμβόλαιο είναι αυτό της επαλήθευσης μεταξύ όλων των οντοτήτων. Ξανά, και αυτό το συμβόλαιο βασίζεται στην προσπέλαση των δομών και πινάκων των πρώτων δύο συμβολαίων, κάνοντάς και αυτό να αποτελείται αποκλειστικά και μόνο από συναρτήσεις. Αυτές είναι οι ακόλουθες:

- ***function verifyDeviceOwnership (_deviceDID, _address)***
- ***function verifySofthubOwnership (_hubDID, _address)***
- ***function verifyDeviceToSofthub (_softhubDID, _deviceDID)***
- ***function verifyAdminToSofthub (_softhubDID)***

Και οι τέσσερις αυτές συναρτήσεις μπορούν να κληθούν αποκλειστικά και μόνο από εξωτερικές πηγές και όχι από άλλα συμβόλαια. Επίσης είναι σημαντικό να παρατηρηθεί πως οι παράμετροι, τις οποίες απαιτούν αυτές οι συναρτήσεις είναι μονάχα

αποκεντρωμένα αναγνωριστικά και αριθμός λογαριασμού, πληροφορίες που είναι ήδη δημόσιες και δεν χρειάζεται με αυτόν τον τρόπο να διαρρεύσει κανενός άλλου είδους πληροφορία. Με την σειρά που αναγράφονται παραπάνω, οι συναρτήσεις επιστρέφουν μία δυαδική μεταβλητή Boolean για τις εξής δηλώσεις:

- Η συσκευή με το αναγνωριστικό *_deviceDID* ανήκει (και άρα μπορεί να εκτελέσει εντολές από) στον χρήστη που έχει αριθμό λογαριασμού *_address*.
- Το softhub με το αναγνωριστικό *_hubDID* ανήκει (και άρα μπορεί να εκτελέσει εντολές από) στον χρήστη που έχει αριθμό λογαριασμού *_address*.
- Η συσκευή με το αναγνωριστικό *_deviceDID* είναι μέρος του softhub με το αναγνωριστικό *_hubDID*.
  Ο αριθμός λογαριασμού που εκτέλεσε τη συγκεκριμένη συνάρτηση έχει δικαιώματα διαχείρισης του softhub με το αναγνωριστικό *_hubDID*.

3.4. Ενσωμάτωση στο Επίπεδο Εφαρμογών ΔτΠ – Web3

Σε αυτό το κεφάλαιο παρουσιάστηκε μία λύση για την διαχείριση προσβασιμότητας και εξουσιοδότησης χρηστών και εφαρμογών στα πλαίσια ενός συστήματος ΔτΠ. Ωστόσο, ανάμεσα στα ερευνητικά ερωτήματα που τέθηκαν στην παρούσα ερευνητική εργασία ήταν και το αν είναι εφικτό αυτή η λύση να μπορεί να ενσωματωθεί σε ήδη λειτουργικά συστήματα ΔτΠ. Η απάντηση σε αυτό το ερώτημα βρίσκεται στον τρόπο σχεδίασης της παραπάνω λύσης. Σχεδιάστηκε με τέτοιο τρόπο, ώστε να μην χρησιμοποιείται από τους τελικούς χρήστες ενός συστήματος ΔτΠ, αλλά από τους σχεδιαστές/προγραμματιστές αυτού, προκειμένου να ενισχύσουν τις εφαρμογές τους με ένα επίπεδο Blockchain.

Αυτό είναι εφικτό ανάλογα με τον τύπο προγραμματισμού των εφαρμογών αυτών. Το πλέον κατάλληλο και πιο εύχρηστο πλαίσιο για την ενσωμάτωση των έξυπνων συμβολαίων είναι το Web3[5]. Πρόκειται για έναν όρο που δημιουργήθηκε κυρίως λόγω της εμφάνισης του Ethereum καθώς γέννησε την ανάγκη για την ανάπτυξη εφαρμογών σε αποκεντρωμένα περιβάλλοντα.

Χρησιμοποιώντας το Web3 είναι εφικτό να ενισχυθεί μία εφαρμογή, είτε διαδικτυακή, είτε κινητή, ώστε να προσφέρει στον τελικό χρήστη την διασύνδεση που απαιτείται για να εκτελεί συναλλαγές με το προσωπικό του Ethereum account. Στα πλαίσια της εργασίας δεν πραγματοποιήθηκε τέτοια εφαρμογή, καθώς το ζητούμενο είναι να αποδειχθεί η ενσωμάτωση σε κάποια ήδη έτοιμη, και όχι η ανάπτυξη αυτής από την αρχή. Ωστόσο, είτε πρόκειται για διαδικτυακή εφαρμογή, είτε για κινητή, είναι απόλυτα σίγουρο πως η ενσωμάτωση είναι εφικτή, τουλάχιστον σε προγραμματιστικό επίπεδο. Βασικό ρόλο σε αυτό παίζει η χρήση του πρωτόκολλου WalletConnect[6], το οποίο αφού ενσωματωθεί, επιτρέπει την σύνδεση οποιουδήποτε εξωτερικού πορτοφολιού που χρησιμοποιεί ο τελικός χρήστης με την οποιαδήποτε διαδικτυακή ή κινητή εφαρμογή για επικοινωνία με κάποιο blockchain.

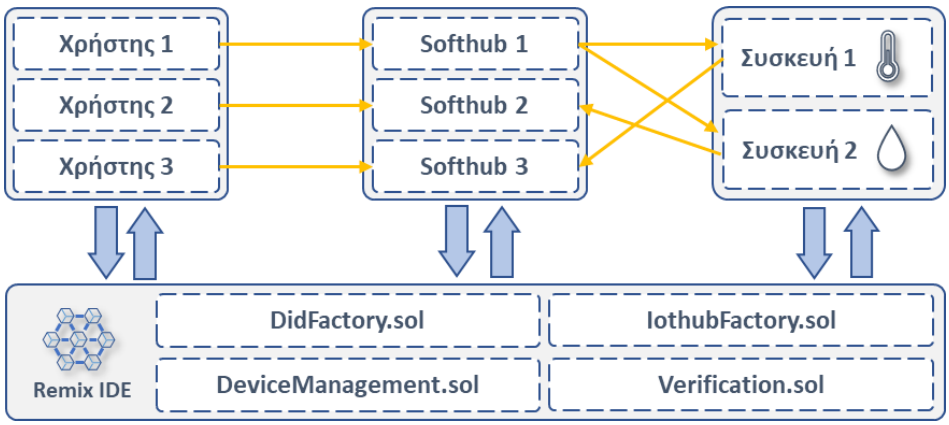
[5] https://web3.foundation/

[6] https://walletconnect.com/

ΚΕΦΑΛΑΙΟ 4:
Εφαρμογή και Αποτελέσματα

Στο κεφάλαιο αυτό θα παρουσιαστεί η εκτέλεση των έξυπνων συμβολαίων που παρουσιάστηκαν στο προηγούμενο κεφάλαιο με τη βοήθεια του λογισμικού Remix. Η Εικόνα 10 απεικονίζει την πειραματική διάταξη που θα ακολουθηθεί χρησιμοποιώντας την υλοποίηση του κεφαλαίου 3. Στο σενάριο αυτό υποθέτουμε ότι έχουμε 3 χρήστες, οι οποίοι έχουν ο καθένας από ένα softhub, έτοιμο να δεχτεί συσκευές για διαχείριση. Επίσης, υποθέτουμε δύο συσκευές τύπου αισθητήρα. Τόσο οι χρήστες και τα softhubs όσο και οι συσκευές θα αποκτήσουν προσωπικό DID αναγνωριστικό με βάση το οποίο θα πραγματοποιείται η ταυτοποίηση και η επαλήθευσή τους σχετικά με τα δικαιώματά εξουσιοδότησης και πρόσβασής τους.

Πιο συγκεκριμένα θα εκτελέσουμε μία σειρά συναλλαγών για την απόκτηση των DID αλλά και την δημιουργία softhubs. Στη συνέχεια θα εξετάσουμε τι συμβαίνει αν κάποιος αποφασίσει να φερθεί «ανέντιμα» και να μην ακολουθήσει τους κανόνες που έχουν οριστεί από την λύση διαχείρισης προσβασιμότητας και εξουσιοδότησης.

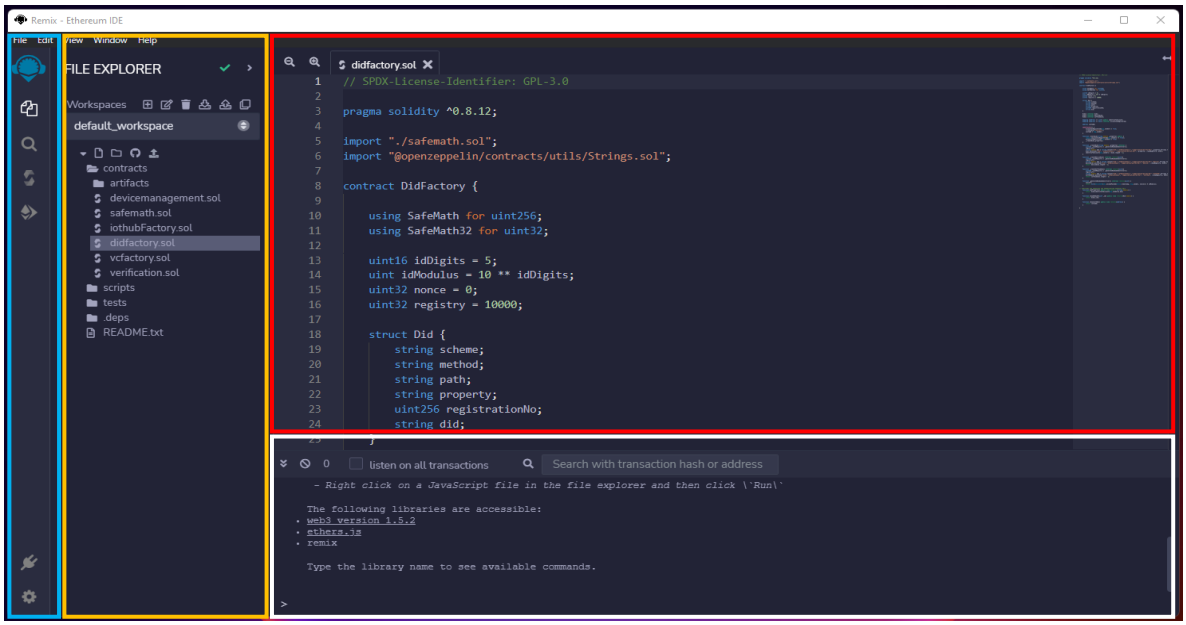


Εικόνα 10: Σενάριο Πειραματικής Διάταξης

4.1. Remix

Το Remix IDE, εκτός από την ικανότητα να μεταγλωττίζει και να τεστάρει τον κώδικα των έξυπνων συμβολαίων, μας παρέχει και την ικανότητα να προσομοιώσουμε την εκτέλεση συναλλαγών αυτών, χωρίς να προαπαιτείται κάποιο πραγματικό blockchain δίκτυο. Στην ουσία, ως εναλλακτικό του blockchain δικτύου χρησιμοποιεί έναν τοπικό προσομοιωτή της εικονικής μηχανής EVM που λειτουργεί ως θεμέλιο εργαλείο των κόμβων ενός δικτύου Ethereum blockchain. Μπορεί να εκτελεί τον κώδικα των συμβολαίων προσπερνώντας τις διαδικασίες του διαμοιρασμού και της συναίνεσης. Αυτό επιτυγχάνεται χρησιμοποιώντας δέκα εικονικούς λογαριασμούς Ethereum.

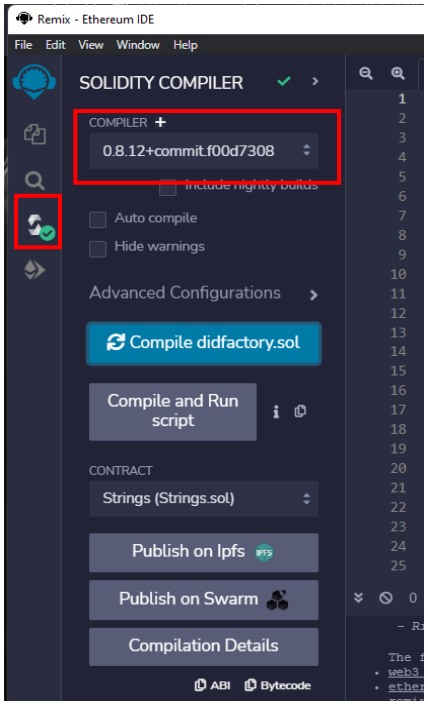
Στην *Εικόνα 11* βλέπουμε ένα στιγμιότυπο του Remix. Στο κέντρο (κόκκινο περίγραμμα) βρίσκεται ο βασικός επεξεργαστής κειμένου, ο οποίος και χρησιμοποιείται στην συγγραφή κώδικα solidity. Ακριβώς από κάτω (λευκό περίγραμμα) βρίσκεται η κονσόλα



Εικόνα 11: Στιγμιότυπο του Remix IDE

στην οποία φαίνονται όλες οι διεργασίες που εκτελούνται, τόσο κατά τη μεταγλώττιση, όσο και κατά την εκτέλεση των συναλλαγών.

Στα αριστερά του στιγμιότυπου της Εικόνας 11, μπορούμε να διακρίνουμε μία στενή στήλη επιλογών (γαλάζιο περίγραμμα), μέσα στις οποίες συμπεριλαμβάνεται η Μεταγλώττιση, η Εκτέλεση των έξυπνων συμβολαίων όπως επίσης και ο περιηγητής αρχείων. Ο τελευταίος είναι η στήλη (κίτρινο περίγραμμα) με τις πληροφορίες των διάφορων απαραίτητων αρχείων, μαζί με τα συμβόλαια, προκειμένου να γίνει η προσομοίωση αυτών.



Εικόνα 12: Στιγμιότυπο Μεταγλώτισης

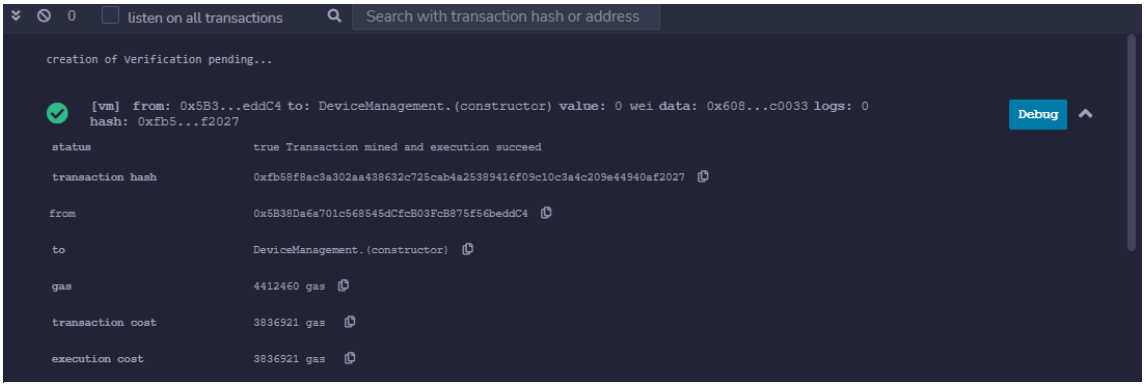
Αν επιλέξουμε την επιλογή της μεταγλώττισης από τη στενή στήλη (*Εικόνα 12*), είναι σημαντικό να παρατηρήσουμε ότι πρέπει να επιλέξουμε την κατάλληλη έκδοση μεταγλωττιστή για τον κώδικα των συμβολαίων που έχουμε γράψει. Στην προκειμένη περίπτωση χρειαζόμαστε μεταγλωττιστή έκδοσης 0.8.12 και πάνω.

Αφού επιλέξουμε να γίνει μεταγλώττιση, αν δεν υπάρχουν σφάλματα, θα παρατηρήσουμε ένα πράσινο σήμα έγκρισης στο σύμβολο επιλογής του μεταγλωττιστή.

4.2. Αλληλεπίδραση με τα Έξυπνα Συμβόλαια

Μετά την επιτυχή μεταγλώττιση, προχωράμε στην εκτέλεση πριν την οποία θα πρέπει να επιλέξουμε έναν από τους δέκα εικονικούς λογαριασμούς. Αξίζει σε αυτό το σημείο να αναφερθεί, πώς χάρη στην ιδιότητα της κληρονομικότητας, δεν χρειάζεται να εκτελέσουμε καθένα από τα τέσσερα συμβόλαια ξεχωριστά. Μπορούμε απλώς να εκτελέσουμε το τελευταίο της

«γενιάς» και θα έχουμε πρόσβαση σε όλες τις συναρτήσεις αλλά και μεταβλητές των προηγούμενων.



Εικόνα 13: Επιτυχής Αρχικοποίηση Έξυπνου Συμβολαίου ως Συναλλαγή

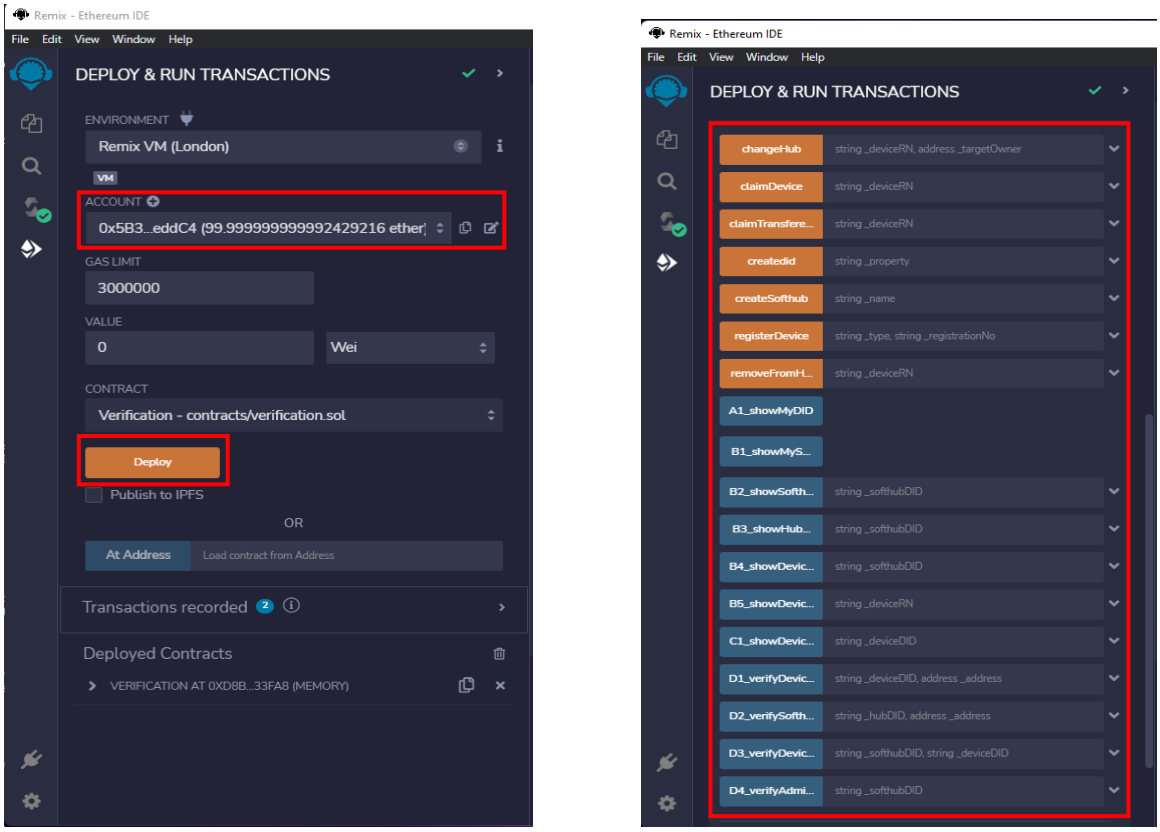
Έτσι, έχοντας αφήσει τον πρώτο λογαριασμό Ethereum επιλεγμένο (*0x5B38Da6a701c568545dCfcB03FcB875f56beddC4*), πατάμε "Deploy" και αφού ολοκληρωθεί η διαδικασία βλέπουμε ακριβώς από κάτω στα διαθέσιμα και προς εκτέλεση συμβόλαια, το τελευταίο από τα τέσσερα, το οποίο είναι το Verification.sol. Στην *Εικόνα 13* μπορούμε να δούμε ότι η ακόμα και η αρχικοποίηση ενός συμβολαίου θεωρείται συναλλαγή για το Ethereum και καταγράφεται ως επιτυχής στην κονσόλα.

Συνεχίζοντας, στην *Εικόνα 14* μπορούμε να δούμε τη δυνατότητα εκτέλεσης και προσομοίωσης όλων των συναρτήσεων που περιλαμβάνονται στα τέσσερα έξυπνα συμβόλαια και που αναλύθηκαν στην προηγούμενη ενότητα. Ο λόγος για τον οποίο κάποια από αυτά εμφανίζονται με πορτοκαλί χρώμα ενώ άλλα με μπλε είναι ότι οι πορτοκαλί συναρτήσεις προκαλούν αλλαγή στην κατάσταση του κατάστιχου ενώ οι μπλε απλώς ζητούν πληροφορία από αυτό και δεν προκαλούν καμία αλλαγή.

Πρέπει να αναφερθεί πως όλες οι μπλε συναρτήσεις, με εξαίρεση τις τέσσερις τελευταίες που αποτελούν τις συναρτήσεις επαλήθευσης, υπό πραγματικές συνθήκες δεν θα είναι διαθέσιμες για την αποφυγή διαρροής πληροφοριών εκτός αλυσίδας για λόγους ιδιωτικότητας. Εδώ, υπάρχουν για λόγους παρουσίασης της λύσης.

Συνεχίζοντας, η διαδικασία που θα ακολουθηθεί για την ολοκλήρωση του πειράματος περιλαμβάνει την εκτέλεση των εξής συναρτήσεων με την ακόλουθη σειρά και παραμέτρους:

1. Επιλογή του δεύτερου λογαριασμού από τους δέκα εικονικούς (0xAb8483F64d9C6d1EcF9b849Ae677dD3315835cb2) και δημιουργία προσωπικού DID με την ιδιότητα του διαχειριστή (Administrator).
2. Δημιουργία ενός softhub με το ψευδώνυμο testhub1.
3. Εγγραφή 2 συσκευών ΔτΠ:
    i. Μία συσκευή τύπου αισθητήρα θερμοκρασίας με σειριακό αριθμό abc1234
    ii. Μία συσκευή τύπου αισθητήρα καπνού με σειριακό αριθμό def5678
4. Διεκδίκηση και των δύο συσκευών από το testhub1 χρησιμοποιώντας τον αντίστοιχο λογαριασμό.



Εικόνα 14: Εκτέλεση Έξυπνων Συμβολαίων και Προσομοίωση των Συναρτήσεών τους

5. Δημιουργία δύο ακόμα προσωπικών DID με την ιδιότητα του διαχειριστή (administrator) χρησιμοποιώντας δύο νέους λογαριασμούς:
   i. 0x4B20993Bc481177ec7E8f571ceCaE8A9e22C02db
   ii. 0x78731D3Ca6b7E34aC0F824c42a7cC18A495cabaB
6. Δημιουργία δύο νέων softhub για τους προαναφερθέντες λογαριασμούς αντίστοιχα:
   i. testhub2
   ii. testhub3
7. Επαλήθευση ελέγχου softhub από τον λογαριασμό που αντιστοιχεί στο softhub2 και για τα τρία softhubs που έχουν δημιουργηθεί.
8. Εκκίνηση μεταφοράς πρώτης συσκευής (με σ.α. abc1234) προς τον λογαριασμό που αντιστοιχεί στο softhub2.
9. Δοκιμή διεκδίκησης συσκευής με σ.α. abc1234 από τον λογαριασμό που ανήκει στο softhub3.
10. Ολοκλήρωση μεταφοράς συσκευής με σ.α. abc1234 από τον λογαριασμό που ανήκει το softhub2.
11. Αφαίρεση συσκευής με σ.α. def5678 από το softhub1.
12. Διεκδίκηση συσκευής με σ.α. def5678 από το softhub3.
13. Επαλήθευση ότι η συσκευή με σ.α. abc1234 ανήκει στο softhub1.
14. Επαλήθευση ότι η συσκευή με σ.α. abc1234 ανήκει στο softhub2.
15. Επαλήθευση ότι η συσκευή με σ.α. def5678 ανήκει στο softhub1.

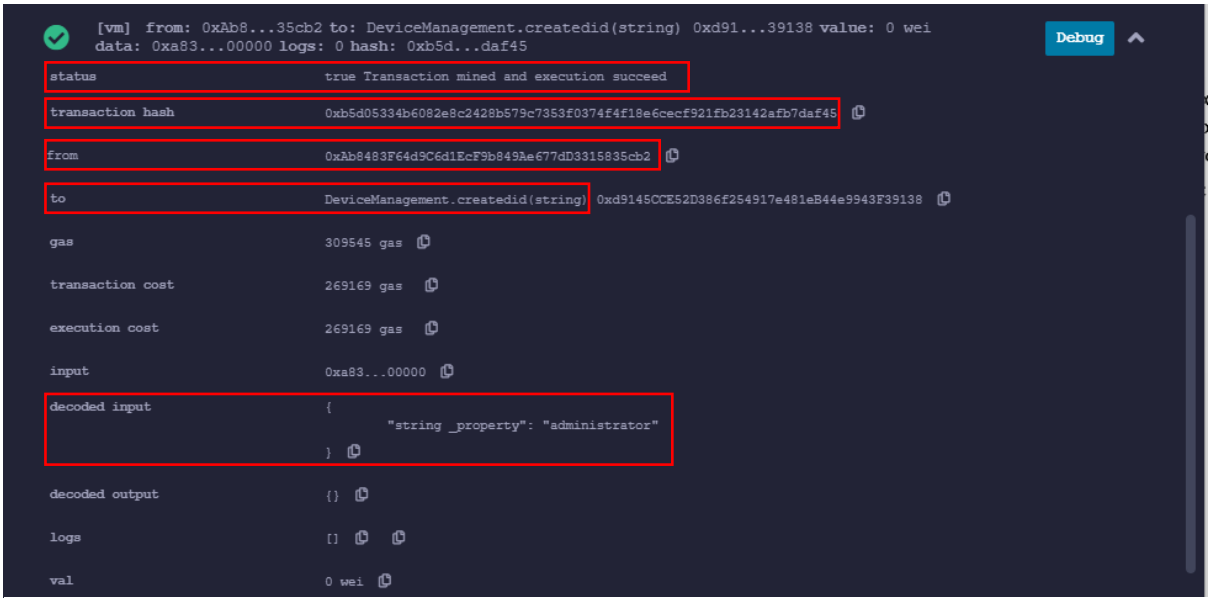
16. Εμφάνιση ιστορικού συσκευής με σ.α. def5678.
17. Επαλήθευση ότι η συσκευή με σ.α. abc1234 μπορεί να ελεγχθεί από τον αριθμό λογαριασμού του softhub1
18. Επαλήθευση ότι η συσκευή με σ.α. abc1234 μπορεί να ελεγχθεί από τον αριθμό λογαριασμού του softhub2
19. Επαλήθευση ότι το softhub1 μπορεί να ελεγχθεί από τον αριθμό λογαριασμού του softhub2
20. Επαλήθευση ότι το softhub1 μπορεί να ελεγχθεί από τον αριθμό λογαριασμού του softhub3

Το αποτέλεσμα όλων των παραπάνω βημάτων θα παρουσιαστεί και θα ερμηνευτεί στην επόμενη ενότητα, μέσω της κονσόλας όπου φαίνεται η επιτυχής ή αποτυχημένη εκτέλεση συναλλαγών.

4.3. Αποτελέσματα – Transactions

Μετά την ολοκλήρωση των έξυπνων συμβολαίων, ξεκινάμε να εκτελούμε τη σειρά των συναλλαγών που προαναφέρθηκαν, με σκοπό να παρατηρήσουμε το αποτέλεσμά τους και τις συνθήκες υπό τις οποίες είναι δυνατόν να αλλάξει η κατάσταση του κατάστιχου. Μετά την εκτέλεση κάθε συναλλαγής θα παρατηρούμε την απόκριση του προσομοιωτή Blockchain του Remix. Η απόκριση αυτή μοιάζει με την απόκριση ενός μηνύματος HTTP μέσω διαδικτύου, περιέχοντας κάποια βασικά στοιχεία της αλληλεπίδρασης.

Ξεκινώντας με την συναλλαγή 1, παρακάτω στην *Εικόνα 15* μπορούμε να δούμε την απόκριση της εκτέλεσής της. Καταρχάς, παρατηρούμε πως στο πεδίο "status" αναγράφεται *"Transaction minded execution succeed"*, το οποίο προφανώς δείχνει την επιτυχή εκτέλεσή της. Στη συνέχεια, παρατηρούμε το γνωστό "transaction hash", το οποίο χαρακτηρίζει τη συναλλαγή μέσα στο block που παράγεται. Ακριβώς από κάτω υπάρχει και ο αριθμός λογαριασμού που εκτέλεσε την συγκεκριμένη συναλλαγή, όπως επίσης ο



Εικόνα 15: Εκτέλεση συναλλαγής 1: Δημιουργία προσωπικού DID με την ιδιότητα διαχειριστή

προορισμός της συναλλαγής, ο οποίος αποτελείται από το συμβόλαιο και τη συνάρτηση που εκτελείται.

Τέλος, λίγο πιο κάτω μπορούμε να δούμε την είσοδο και την έξοδο της συνάρτησης. Όπως παρατηρούμε, στο σύνολό της, η εκτέλεση της συναλλαγής φανερώνει ότι ο συγκεκριμένος αριθμός λογαριασμού δημιούργησε ένα προσωπικό DID με την ιδιότητα του διαχειριστή.

Συνεχίζοντας με τις συναλλαγές 2 έως και 6, η απόκριση του Remix είναι επιτυχής και σχεδόν πανομοιότυπη με αυτή της *Εικόνα 15*, με τη μόνη διαφορά ότι αλλάζουν οι τιμές στα πεδία "transaction hash", "from", "to" και "decoded input" με την αντίστοιχη πληροφορία. Οι αποκρίσεις των συναλλαγών αυτών παρατίθενται αναλυτικά στο Παράρτημα Β.

Ο *Πίνακας 1* δείχνει την κατάσταση του κατάστιχου μετά την εκτέλεση των συναλλαγών 1 έως και 6. Όπως βλέπουμε, ολόκληρη η πληροφορία περιλαμβάνει τις struct δομές που αντιπροσωπεύουν χρήστες, softhubs και συσκευές, όπως επίσης και την αντιστοίχιση των τριών αυτών μεταξύ τους με σχέσεις ιδιοκτησίας, διατηρώντας όσο το δυνατόν την ανωνυμία.

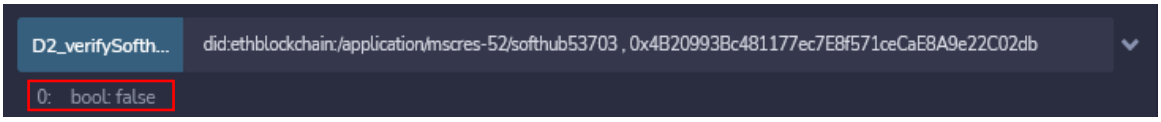
Πίνακας 1: Κατάσταση κατάστιχου μετά τις συναλλαγές 1 έως 6

Δομή Δεδομένων	Πληροφορία που περιέχεται
Struct DID	did:ethblockchain:/application/mscres-52/administrator91101
Struct DID	did:ethblockchain:/application/mscres-52/administrator98868
Struct DID	did:ethblockchain:/application/mscres-52/administrator54989
Struct DID	did:ethblockchain:/application/mscres-52/softhub53703
Struct DID	did:ethblockchain:/application/mscres-52/softhub37899
Struct DID	did:ethblockchain:/application/mscres-52/softhub92253
Struct DID	did:ethblockchain:/application/mscres-52/device89873
Struct DID	did:ethblockchain:/application/mscres-52/device47235
Struct Softhub	Testhub1, did:…/softhub53703
Struct Softhub	Testhub2, did:…/softhub37899
Struct Softhub	Testhub3, did:…/softhub92253
Struct Device	temp_sensor, abc1234
Struct Device	Smoke_detect, def5678
Mapping identityToAccount	0xAb8483F…835cb2 => …/administrator91101 0x4B20993…2C02db => …/administrator98868 0x78731D3…5cabaB => …/administrator54989
Mapping OwnerToHub	did:…/administrator91101 => Testhub1 did:…/administrator98868 => Testhub2 did:…/administrator54989 => Testhub3

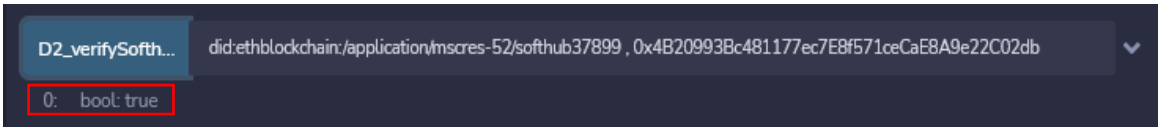
Mapping rnToDev	abc1234 => did:…/device89873 def5678 => did:…/device47235
Mapping DevToHub	did:…/device89873 => Testhub1 did:…/device47235 => Testhub1

Στη συνέχεια, η συναλλαγή 7 (επαλήθευση ελέγχου softhub από τον λογαριασμό που αντιστοιχεί στο softhub2 και για τα τρία softhubs που έχουν δημιουργηθεί) δεν είναι συναλλαγή που θα καταγραφεί στο κατάστιχο, καθώς δεν αλλάζει την κατάστασή του με κανένα τρόπο. Πρόκειται για μία συναλλαγή τύπου ερωτήματος (query) και πιο συγκεκριμένα από συνάρτηση του έξυπνου συμβολαίου Verification, το οποίο χρησιμοποιείται για την επαλήθευση προσβασιμότητας και εξουσιοδότησης. Λόγω της φύσης της συναλλαγής, δεν θα δούμε το αποτέλεσμα αυτής στην κονσόλα, αλλά ακριβώς κάτω από το πεδίο της κλήσης της αντίστοιχης συνάρτησης (*Εικόνα 14*). Έτσι, δίνοντας ως είσοδο κάθε φορά, το DID του softhub1 με τους 3 διαφορετικούς αριθμούς λογαριασμού, λαμβάνουμε την απάντηση της *Εικόνα 16*. Όπως παρατηρούμε, τα αποτελέσματα ορθά δείχνουν ότι ο λογαριασμός που αντιστοιχεί στο softhub2 μπορεί να ελέγξει αποκλειστικά και μόνο το softhub2, ενώ η απάντηση του blockchain για τα softhub1 και softhub3 είναι ότι δεν μπορεί να τα ελέγξει.

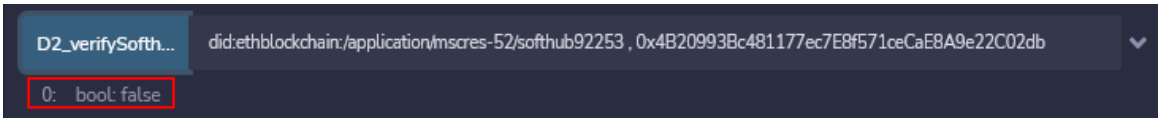
α



β



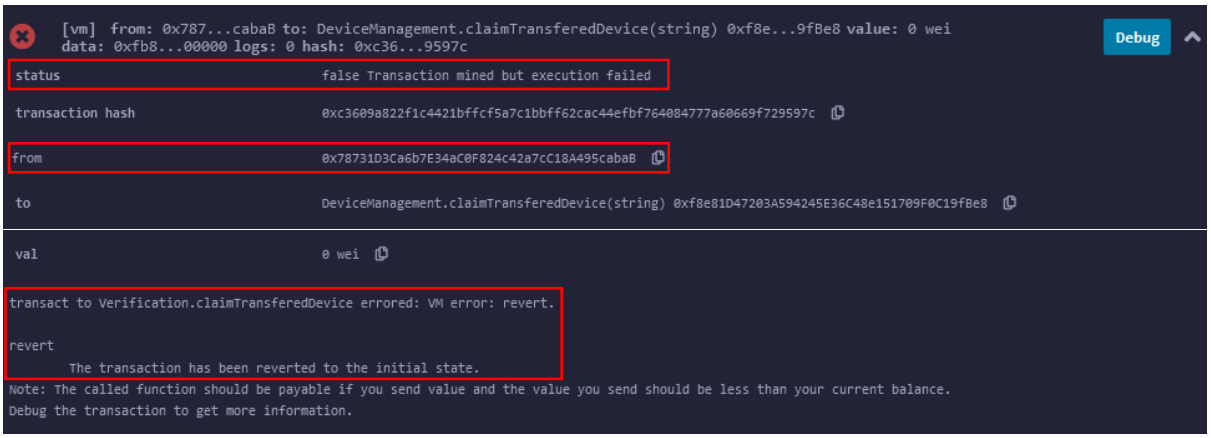
γ



Εικόνα 16: Επαλήθευση Ελέγχου: α) του softhub1 από τον δεύτερο αριθμό λογαριασμού
β) του softhub2 από τον δεύτερο αριθμό λογαριασμού
γ) του softhub2 από τον δεύτερο αριθμό λογαριασμού

Συνεχίζοντας, οι συναλλαγές 8 έως και 12 έχουν σκοπό να αλλάξουν την κατάσταση του κατάστιχου, εκτελώντας όλες τις δυνατές επιλογές διαχείρισης συσκευής από το έξυπνο συμβόλαιο DeviceManagement.sol. Επίσης οι αναλυτικές αποκρίσεις αυτών των συναλλαγών υπάρχουν στο Παράρτημα Β της εργασίας αυτής. Ωστόσο, αξίζει να σταθούμε στην συναλλαγή 9, κατά την οποία ουσιαστικά επιχειρεί ο λογαριασμός του softhub3 να διεκδικήσει μία συσκευή ενώ αυτή έχει δεσμευτεί για μεταφορά από το softhub1 προς το softhub2 αποκλειστικά.

Η απόκριση της εν λόγω συναλλαγής φαίνεται στην *Εικόνα 17.* Είναι εμφανές ότι η συναλλαγή έχει αποτύχει, ενώ ταυτόχρονα καταγράφεται και ο αριθμός λογαριασμού από τον οποίο εκτελέστηκε. Επιπλέον, στο κάτω μέρος της απόκρισης, αναγράφεται πως



Εικόνα 17: Αποτυχία Συναλλαγής 9

από τη στιγμή που η συναλλαγή απορρίφθηκε, οποιαδήποτε αλλαγή επιχείρησε να προκαλέσει η αντίστοιχη συνάρτηση στο κατάστιχο ανατράπηκε και γύρισε στην κατάσταση που βρισκόταν ακριβώς πριν.

Τέλος, οι συναλλαγές 13 έως και 20, είναι τύπου ερωτήματος ακριβώς όπως η συναλλαγή 7 και αποσκοπεί στην παρουσίαση της λειτουργίας του έξυπνου συμβολαίου Verification.sol. Έτσι, δοκιμάζοντας διάφορους συνδυασμούς παραμέτρων εξετάζουμε τη λειτουργικότητα της διαχείρισης προσβασιμότητας και εξουσιοδότησης. Το αποτέλεσμα των συναλλαγών αυτών σε συνάρτηση με τον αριθμό συναλλαγής φαίνονται στον *Πίνακας 2*, ενώ η τελική κατάσταση του κατάστιχου, στον *Πίνακας 3*.

Πίνακας 2: Εκτέλεση συναλλαγών 13 - 20

Απόδειξη	Παράμετρος 1	Παράμετρος 2	Έξοδος
verifyDeviceToSofthub	did:… /softhub53703	… /device89873	false
verifyDeviceToSofthub	did:… /softhub37899	… /device89873	true
verifyDeviceToSofthub	did:…/softhub53703	…/device47235	false
verifyDeviceOwnership	did:… /device89873	0xAb8483…835cb2	false
verifyDeviceOwnership	did:… /device89873	0x4B2099…2C02db	true
verifySofthubOwnership	did:… /softhub53703	0x4B2099…2C02db	false
verifySofthubOwnership	did:… /softhub53703	0x78731D…5cabaB	false

Πίνακας 3: Τελική κατάσταση κατάστιχου μετα την εκτέλεση των συναλλαγών 1 - 20

Δομή Δεδομένων	Πληροφορία που περιέχεται
Struct DID	did:ethblockchain:/application/mscres-52/administrator91101
Struct DID	did:ethblockchain:/application/mscres-52/administrator98868
Struct DID	did:ethblockchain:/application/mscres-52/administrator54989
Struct DID	did:ethblockchain:/application/mscres-52/softhub53703
Struct DID	did:ethblockchain:/application/mscres-52/softhub37899
Struct DID	did:ethblockchain:/application/mscres-52/softhub92253

Struct DID	did:ethblockchain:/application/mscres-52/device89873
Struct DID	did:ethblockchain:/application/mscres-52/device47235
Struct Softhub	Testhub1, did:…/softhub53703
Struct Softhub	Testhub2, did:…/softhub37899
Struct Softhub	Testhub3, did:…/softhub92253
Struct Device	temp_sensor, abc1234
Struct Device	Smoke_detect, def5678
Mapping identityToAccount	0xAb8483F…835cb2 => did:…/administrator91101 0x4B20993…2C02db => did:…/administrator98868 0x78731D3…5cabaB => did:…/administrator54989
Mapping OwnerToHub	did:…/administrator91101 => Testhub1 did:…/administrator98868 => Testhub2 did:…/administrator54989 => Testhub3
Mapping rnToDev	abc1234 => did:…/device89873 def5678 => did:…/device47235
Mapping DevToHub	**did:…/device89873 => Testhub2** **did:…/device47235 => Testhub3**

ΚΕΦΑΛΑΙΟ 5:
Ανάλυση Αποτελεσμάτων

Τα αποτελέσματα ή τα ευρήματα που περιγράφηκαν στο προηγούμενο κεφάλαιο, εδώ αναλύονται, σχολιάζονται, κρίνονται και αποτιμώνται. Στο Κεφάλαιο 2 όπου περιγράφεται η μεθοδολογία της έρευνας της παρούσας εργασίας, τέθηκαν κάποιες παράμετροι υπό εξέταση, σχετικά με την ασφάλεια του ΔτΠ. Μετά από την σχεδίαση και την ανάπτυξη της πειραματικής λύσης, σε αυτό το κεφάλαιο θα εξαχθούν πιο γενικευμένα συμπεράσματα, τα οποία ωστόσο επικεντρώνονται στις προς μέτρηση παραμέτρους της έρευνας. Επίσης θα αναφερθούν δυσκολίες αλλά και προβλήματα που εμπόδισαν την ποιότητα και την εγκυρότητα ή τη γενίκευση των αποτελεσμάτων.

5.1. Εξέταση Παραμέτρων

Παρατηρώντας την διεκπεραίωση του πειράματος του κεφαλαίου 4, πρέπει να αναφερθεί ό,τι ο τρόπος υλοποίησης μίας λύσης βασισμένης σε Blockchain δεν είναι απλή υπόθεση. Απαιτεί πολύ καλή γνώση των προγραμματιστικών εργαλείων που περιλαμβάνονται κάτω από την ομπρέλα της τεχνολογίας, ενώ ταυτόχρονα χρειάζεται και εμπειρία όσον αφορά τις σχεδιαστικές και αρχιτεκτονικές επιλογές.

Στον *Πίνακας 4* φαίνονται οι παράμετροι που τέθηκαν κατά την επιλογή της μεθοδολογίας της έρευνας σε συνάρτηση με την επιρροή που μπορεί να παρατηρηθεί μετά το πείραμα του κεφαλαίου 4. Η επιρροή, καθώς δεν είναι εφικτό να μετρηθεί αριθμητικά, χαρακτηρίζεται με τους εξής όρους:

- Θετική: Παρατηρείται θετική επιρροή στην παράμετρο με την βελτίωσή της να υπάρχει στην περίπτωση που χρησιμοποιηθεί η λύση, έναντι της περίπτωσης που δεν χρησιμοποιηθεί.
- Μερικώς Θετική: Παρατηρείται θετική επιρροή αλλά όχι σε κάθε περίπτωση χρήσης. Δεν παρουσιάζεται αρνητική επιρροή ούτε εκτίθεται από θέμα ασφάλειας το σύστημα, εξ' αιτίας της παραμέτρου
- Ουδέτερη: Η παράμετρος δεν επηρεάζεται ή αγγίζεται από την χρήση της προταθείσας λύσης.
- Θετική Προοπτική: Μπορεί να επηρεαστεί θετικά σε μεγάλο ή μικρό βαθμό αν αναπτυχθεί περεταίρω η προταθείσα λύση.

Υποθέτοντας την ύπαρξη δύο ταυτόσημων και ολοκληρωμένων συστημάτων ΔτΠ, η αξιολόγηση της επιρροής πραγματοποιείται μέσω παρατήρησης και σύγκρισης μεταξύ αυτών των δύο. Ωστόσο, το πρώτο σύστημα κάνει χρήση της προταθείσας λύσης, ενώ αντίθετα το δεύτερο δεν συμπεριλαμβάνει μία λύση βασισμένη σε blockchain για την επαλήθευση εξουσιοδότησης και πρόσβασης χρηστών ή συσκευών.

Πίνακας 4: Ερευνητικές Παράμετροι προς Εξέταση

Παράμετρος	Επιρροή
Ταυτοποίηση εγκεκριμένων συσκευών και εφαρμογών	Θετική
Έλεγχος και Διαχείριση Πρόσβασης μόνο σε εγκεκριμένους χρήστες	Θετική
Διαχείριση Εξουσιοδότησης ελέγχου συσκευών στα πλαίσια του δικτύου	Θετική
Ακεραιότητα Δεδομένων	Ουδέτερη / Θετική προοπτική
Ιδιωτικότητα Δεδομένων	Μερικώς Θετική
Κρυπτογράφηση Δεδομένων και Επικοινωνίας	Ουδέτερη / Θετική προοπτική
Αυθεντικοποίηση πηγών δεδομένων	Θετική
Συνεχής Διαθεσιμότητα Δεδομένων κατ' απαίτηση	Ουδέτερη
Συνεχής Διαθεσιμότητα Υπηρεσιών κατ' απαίτηση	Μερικώς Θετική
Επεκτασιμότητα	Θετική

5.2. Ανάλυση Παραμέτρων

Πιο αναλυτικά, είναι προφανές πως η ταυτοποίηση και η διαχείριση/έλεγχος πρόσβασης και εξουσιοδότησης επηρεάζονται θετικά και άμεσα από την λύση, καθώς παρουσιάζονται προγραμματιστικές μεταβλητές που ανταποκρίνονται ακριβώς σε αυτές τις παραμέτρους. Η εκτέλεση έξυπνων συμβολαίων αφήνει τους χρήστες να έχουν τον πλήρη έλεγχο στην αδειοδότηση συσκευών και χρηστών στα πλαίσια του συστήματος ΔτΠ.

Σχετικά με τα δεδομένα που διακινούνται σε ένα σύστημα ΔτΠ, η επιρροή παραμένει θετική σε μικρότερο βαθμό, καθώς η παραπάνω λύση μπορεί να επαληθεύσει την πηγή των δεδομένων αυτών εφόσον είναι αληθής η πληροφορία της συσκευής που δρα ως πηγή. Επίσης, μπορεί να επηρεαστεί μερικώς θετικά και η ιδιωτικότητα δεδομένων, αποκλειστικά και μόνο από την πλευρά του ελέγχου προσβασιμότητας. Σε αυτό το σημείο, η Ακεραιότητα των δεδομένων έχει πολλές προοπτικές να επηρεαστεί θετικά με τη χρήση Blockchain Oracle, εργαλείο που εξειδικεύεται στην ασφάλεια δεδομένων που εισέρχονται στο blockchain από οντότητες εκτός αλυσίδας.

Επιπλέον, η Ιδιωτικότητα δεδομένων μπορεί να εξασφαλιστεί στον μέγιστο βαθμό αν ενσωματωθεί ένα επίπεδο ZKP κρυπτογράφησης. Με αυτόν τον τρόπο, αντί να καταγράφονται τα στοιχεία εξουσιοδότησης των συσκευών και των χρηστών στο κατάστιχο, θα καταγράφονται οι κρυπτογραφικές αποδείξεις αυτών, κάνοντας τη διαρροή ευαίσθητης πληροφορίας σχεδόν αδύνατη. Φυσικά, για τον ίδιο λόγο, μπορεί να επηρεαστεί θετικά και η παράμετρος της Κρυπτογράφησης Δεδομένων και Επικοινωνίας.

Η Διαθεσιμότητα Δεδομένων δεν επηρεάζεται σε κανένα βαθμό, καθώς η παρουσιαζόμενη λύση δεν εμπλέκεται με τα δεδομένα που διακινούνται μέσα στο ΔτΠ. Από την άλλη πλευρά, η συνεχής διαθεσιμότητα υπηρεσιών επηρεάζεται σε μικρό βαθμό καθώς η λύση αυτή που παρουσιάζεται δεν βασίζεται σε κεντρικοποιημένη υποδομή,

όπως αυτές που χρησιμοποιούνται ευρέως σήμερα και είναι αρκετά πιθανό να «πέσουν», αφήνοντας για κάποιο χρονικό διάστημα ολόκληρο το σύστημα εκτεθειμένο σε πολλούς κινδύνους. Ωστόσο, η λύση δεν μπορεί να επηρεάσει θετικά της διαθεσιμότητα άλλων υπηρεσιών που μπορεί να απαιτούνται για την λειτουργία του ΔτΠ (π.χ. brokers, διακομιστές http, rest services κ.α.).

Τέλος, η τελευταία παράμετρος της Επεκτασιμότητας μπορεί μόνο θετικά να επηρεαστεί, αν σκεφτεί κανείς πως για την ενσωμάτωση μία λύσης βασισμένης σε blockchain δεν απαιτεί την υποστήριξη ξεχωριστής υποδομής, καθώς υπάρχει μία ποικιλία από ανοιχτά blockchain δίκτυα που μπορούν να χρησιμοποιηθούν. Με αυτόν τον τρόπο μάλιστα, είναι εφικτό να σχεδιαστεί και να δημιουργηθεί ένα ολόκληρο πρωτόκολλο Διαχείρισης Ταυτοποίησης και Πρόσβασης, το οποίο μπορεί να προσφέρεται ως υπηρεσία σε κάποιο δημόσιο δίκτυο blockchain.

5.3. Περιορισμοί – Προβλήματα

Το επίπεδο των προγραμματιστικών εργαλείων είναι ακόμα σχετικά χαμηλό, καθώς δεν αποτελούν έτοιμες λύσεις, οι οποίες είναι εφικτό απλώς να ενσωματωθούν. Για παράδειγμα, η παράμετρος της ιδιωτικότητας θα επηρεαζόταν σε πολύ διαφορετικό και θετικό βαθμό αν ήταν εφικτή η ενσωμάτωση των κρυπτογραφικών ZKP αποδείξεων. Σε αυτήν την περίπτωση, η παρουσιαζόμενη λύση θα ήταν κατάλληλη ακόμα και για ΔτΠ συστήματα που διαχειρίζονται εξαιρετικά ευαίσθητα δεδομένα, για παράδειγμα στον τομέα της υγείας ή της ψηφιακής ταυτοποίησης.

Ένας ακόμη περιορισμός που στάθηκε εμπόδιο στην περεταίρω ανάπτυξη της λύσης ήταν η υψηλή απαίτηση όσον αφορά τις προγραμματιστικές γνώσεις που απαιτούνται. Τα έξυπνα συμβόλαια έχουν τη δική τους αποκλειστική γλώσσα προγραμματισμού, σε αντίθεση με το ψηφιακό πορτοφόλι και τις αποκεντρωμένες εφαρμογές που απαιτούν γνώσεις προγραμματισμού κινητών αλλά και διαδικτυακών εφαρμογών (React/ HTML – Javascript – CSS / PHP κ.λ.π.). Για παράδειγμα, η ενσωμάτωση των ZKP δεν απαιτεί μόνο την εκμάθηση ειδικής γλώσσας προγραμματισμού εξατομικευμένων ZKP κυκλωμάτων, αλλά επίσης απαιτεί και γνώσεις προγραμματισμού υποδομών και υπηρεσιών, καθώς απαιτεί ξεχωριστή και αποκλειστική υποδομή για την διαδικασία της επαλήθευσης ενός αποδεικτικού ZKP.

ΚΕΦΑΛΑΙΟ 6:
Συμπεράσματα – Προτάσεις

Συμπερασματικά, μετά το πέρας της παρούσας εργασίας, τα ερευνητικά ερωτήματα που τέθηκαν στην αρχή της έρευνας, μπορούν πλέον, ως έναν ικανοποιητικό βαθμό να απαντηθούν. Επιπλέον, μέσα από την βιβλιογραφική αναζήτηση ως προαπαιτούμενο της εργασίας αυτής, προέκυψε μία δημοσίευση ενός κεφαλαίου με τίτλο "Identity management in Internet of Things with Blockchain" [31] του διαδικτυακού βιβλίου "*Blockchain based Internet of Things*", το οποίο ανήκει στην σειρά βιβλίων "*Lecture Notes on Data Engineering and Communications Technologies*" των εκδόσεων Springer.

Για αρχή, χρήζει διευκρίνισης το γεγονός ότι η τεχνολογία Blockchain προκειμένου να αξιοποιηθεί σε βέλτιστο βαθμό, θα πρέπει να σχεδιαστεί σε συνδυασμό με το υπόλοιπο οικοσύστημα στο οποίο θα συμμετάσχει. Δεν αποτελεί ένα απλό εργαλείο, το οποίο μπορεί απλώς να «στηθεί» και να ξεκινήσει να προσφέρει τα πλεονεκτήματά του. Απαιτεί αποκλειστικό software engineering και πολύ καλή γνώση τουλάχιστον της γλώσσας solidity (ή αντίστοιχης γλώσσας προγραμματισμού έξυπνων συμβολαίων), προκειμένου να αξιοποιηθούν οι ιδιότητές της στο έπακρο.

Έχοντας τονίσει το παραπάνω γεγονός, μπορούμε να πούμε ότι με περεταίρω ανάπτυξη, τόσο σχεδιαστικά / αρχιτεκτονικά όσο και προγραμματιστικά, το blockchain μπορεί να αντιμετωπίσει μία λίστα από ζητήματα που υπάρχουν στο ΔτΠ και ταλαιπωρούν τους μηχανικούς λογισμικού. Κάποια από αυτά τα ζητήματα εμφανίζονται στην παρακάτω λίστα.

- ✓ Ενίσχυση ασφάλειας μηχανισμών για την ενημέρωση συσκευών
- ✓ Διαχείριση Συσκευών και Πρόσβασης ειδικότερα για μεγαλύτερης κλίμακας δίκτυα με μεγάλο αριθμό πανομοιότυπων συσκευών
- ✓ Ενίσχυση ασφάλειας στην προσβασιμότητα των εκτός αλυσίδας δεδομένων
- ✓ Σχεδιασμός πιο ασφαλών εφαρμογών υψηλού επιπέδου (ειδικά στις περιπτώσεις όπου οι συσκευές δεν έχουν υψηλή ασφάλεια)
- ✓ Αντιμετώπιση του προβλήματος της κλοπής ταυτοτήτων συσκευών (impersonating)
- ✓ Χρήση των ιδιοτήτων της solidity, ειδικότερα των Events για σχεδιασμό αυθεντικοποίησης πολλαπλών παραγόντων
- ✓ Χρήση των ZKP προκειμένου να μην εκτίθενται σε καμία περίπτωση ευαίσθητες ή προσωπικές πληροφορίες

Εν κατακλείδι, είναι ασφαλές να συμπεράνουμε πλέον ότι μπορούν όλα τα ερευνητικά ερωτήματα να δεχθούν θετική απάντηση. Ναι, το Blockchain μπορεί να συμβάλλει θετικά στην ενίσχυση της ασφάλειας του ΔτΠ. Ναι, είναι εφικτό να σχεδιαστεί με τέτοιο τρόπο ώστε να ενσωματωθεί σε ήδη λειτουργικά συστήματα. Ο τρόπος ώστε να επιτευχθεί αυτό κρύβεται στην υιοθέτηση μίας συγκεκριμένης λογικής κατά το σχεδιασμό της λύσης. Η λογική αυτή μπορεί να παρομοιαστεί με αυτή των API, με τη διαφορά πως αντί να δίνεται απομακρυσμένη πρόσβαση σε δεδομένα, δίνεται πρόσβαση σε εκτελέσιμο κώδικα.

ΒΙΒΛΙΟΓΡΑΦΙΑ - ΠΗΓΕΣ

Βιβλιογραφία – Πηγές σε ξένες γλώσσες

[1]. Al-Fuqaha, M. Guizani, M. Mohammadi, M. Aledhari and M. Ayyash, "Internet of Things: A Survey on Enabling Technologies, Protocols, and Applications", IEEE Communications Surveys & Tutorials, vol. 17, no. 4, pp. 2347-2376, 2015. Available: 10.1109/comst.2015.2444095 [Accessed 15 February 2022].

[2]. J. Lin, W. Yu, N. Zhang, X. Yang, H. Zhang and W. Zhao, "A Survey on Internet of Things: Architecture, Enabling Technologies, Security and Privacy, and Applications", IEEE Internet of Things Journal, vol. 4, no. 5, pp. 1125-1142, 2017. Available: 10.1109/jiot.2017.2683200.

[3]. S. Vashi, J. Ram, J. Modi, S. Verma and C. Prakash, "Internet of Things (IoT): A vision, architectural elements, and security issues", 2017 International Conference on I-SMAC (IoT in Social, Mobile, Analytics and Cloud) (I-SMAC), 2017. Available: 10.1109/i-smac.2017.8058399.

[4]. J. Granjal, E. Monteiro and J. Sa Silva, "Security for the Internet of Things: A Survey of Existing Protocols and Open Research Issues", IEEE Communications Surveys & Tutorials, vol. 17, no. 3, pp. 1294-1312, 2015. Available: 10.1109/comst.2015.2388550.

[5]. H. Atlam, A. Alenezi, M. Alassafi, A. Alshdadi and G. Wills, "Security, Cybercrime and Digital Forensics for IoT", Intelligent Systems Reference Library, pp. 551-577, 2019. Available: 10.1007/978-3-030-33596-0_22.

[6]. Polychronaki M., Patrikakis C.Z., Social Distancing: Technology to the Rescue, In International Conference on Strategic Innovative Marketing and Tourism, ISCIMAT 2020, Zante, Kefallonia – Ionian Islands, Greece, September 2020

[7]. B. Bodo and A. Giannopoulou, "The logics of technology decentraliziation - the case of distributed ledger technologies", Blockchain and Web 3.0: Social, Economic, and Technological Challenges, New York: Routledge, 2020, p. Chapter 8.

[8]. F. Casino, T. Dasaklis and C. Patsakis, "A systematic literature review of blockchain-based applications: Current status, classification and open issues", Telematics and Informatics, vol. 36, pp. 55-81, 2019. Available: 10.1016/j.tele.2018.11.006.

[9]. A. Reyna, C. Martín, J. Chen, E. Soler and M. Díaz, "On blockchain and its integration with IoT. Challenges and opportunities", Future Generation Computer Systems, vol. 88, pp. 173-190, 2018. Available: 10.1016/j.future.2018.05.046.

[10]. W. Wu, E. Liu, X. Gong and R. Wang, "Blockchain Based Zero-Knowledge Proof of Loca-tion in IoT," ICC 2020 - 2020 IEEE International Conference on Communications (ICC), 2020, pp. 1-7, doi: 10.1109/ICC40277.2020.9149366

[11]. Chuang, B. Guo, J. Tsai and Y. Kuo, "Multi-graph Zero-knowledge-based authentication system in Internet of Things", IEEE International Conference on Communications (ICC), 2017, pp. 1-6, doi: 10.1109/ICC.2017.7996820
[12]. S. Muthamilselvan, N. Praveen, S. Suresh and V. Sanjana, "E-DOC Wallet Using Blockchain," 2018 3rd International Conference on Communication and Electronics Systems (ICCES), 2018, pp. 989-993, doi: 10.1109/CESYS.2018.8724054.
[13]. N. Naik and P. Jenkins, "Self-Sovereign Identity Specifications: Govern Your Identity Through Your Digital Wallet using Blockchain Technology," 2020 8th IEEE International Conference on Mobile Cloud Computing, Services, and Engineering (MobileCloud), 2020, pp. 90-95, doi: 10.1109/MobileCloud48802.2020.00021.
[14]. P. Carnley and H. Kettani, "Identity and Access Management for the Internet of Things", International Journal of Future Computer and Communication, vol. 8, no. 4, pp. 129-133, 2019. Available: 10.18178/ijfcc.2019.8.4.554.
[15]. N. El Ioini and C. Pahl, "A Review of Distributed Ledger Technologies", Lecture Notes in Computer Science, pp. 277-288, 2018. Available: 10.1007/978-3-030-02671-4_16.
[16]. W3, "Decentralized Identifiers (DIDs) v1.0", W3.org, 2021, [Online], Available: https://www.w3.org/TR/did-core/ [accessed 18/08/2021]
[17]. W3, "Verifiable Credentials Data Model 1.0", W3.org, 2021, [Online], Available: https://www.w3.org/TR/vc-data-model/ [accessed 18/08/2021]
[18]. K. C. Toth and A. Anderson-Priddy, "Self-Sovereign Digital Identity: A Paradigm Shift for Identity", IEEE Security & Privacy, vol. 17, no. 3, pp. 17-27, May-June 2019, doi: 10.1109/MSEC.2018.2888782.
[19]. C. Allen, “The path for self-sovereign identity”, http://www.lifewithalacrity.com , Apr 25, 2016, [online], Available: http://www.lifewithalacrity.com/2016/04/the-path-to-self-soverereign-identity.html [accessed 18/08/2021]
[20]. D. Dasgupta, J. Shrein, and K. Gupta, “A survey of blockchain from security perspective”, Journal of Banking and Financial Technology, vol. 3, no. 1, pp.1-17, 2019, doi: 10.1007/s42786-018-00002-6
[21]. D. Dasgupta, J. Shrein, and K. Gupta, “A survey of blockchain from security perspective”, Journal of Banking and Financial Technology, vol. 3, no. 1, pp.1-17, 2019, doi: 10.1007/s42786-018-00002-6
[22]. Α.Παγουρτζής, Ε. Ζάχος, Αποδείξεις Μηδενικής Γνώσης, Κεφάλαιο Συγγράμματος, Υπολογιστική κρυπτογραφία, 2015 [ηλεκτρ. βιβλ.] Αθήνα: Σύνδεσμος Ελληνικών Ακαδημαϊκών Βιβλιοθηκών, κεφ 10, Διαθέσιμο στο: http://hdl.handle.net/11419/5449
[23]. A. Dorri, S. S. Kanhere, R. Jurdak and P. Gauravaram, "Blockchain for IoT security and privacy: The case study of a smart home," 2017 IEEE International Conference on Pervasive Computing and Communications Workshops (PerCom Workshops), Kona, HI, 2017, pp. 618-623, Available: 10.1109/PERCOMW.2017.7917634.

[24]. S. Huh, S. Cho and S. Kim, "Managing IoT devices using blockchain platform," 2017 19th International Conference on Advanced Communication Technology (ICACT), Bongpyeong, 2017, pp. 464-467, doi: 10.23919/ICACT.2017.7890132.
[25]. O. Novo, "Blockchain Meets IoT: An Architecture for Scalable Access Management in IoT", IEEE Internet of Things Journal, vol. 5, no. 2, pp. 1184-1195, 2018. Available: 10.1109/jiot.2018.2812239.
[26]. G. Lawton, "Top 9 blockchain platforms to consider in 2022", SearchCIO, 2022. [Online]. Available: https://www.techtarget.com/searchcio/feature/Top-9-blockchain-platforms-to-consider.
[27]. Macdonald, Millie & Liu-Thorrold, Lisa & Julien, R. (2017). The Blockchain: A Comparison of Platforms and Their Uses Beyond Bitcoin. 10.13140/RG.2.2.23274.52164.
[28]. L. Lao, Z. Li, S. Hou, B. Xiao, S. Guo and Y. Yang, "A Survey of IoT Applications in Blockchain Systems", ACM Computing Surveys, vol. 53, no. 1, pp. 1-32, 2021. Available: 10.1145/3372136.
[29]. M. Suvitha and R. Subha, "A Survey on Smart Contract Platforms and Features," 2021 7th International Conference on Advanced Computing and Communication Systems (ICACCS), 2021, pp. 1536-1539, doi: 10.1109/ICACCS51430.2021.9441970.
[30]. Ara, Tabassum & Shah, Pritam & Prabhakar, M. (2016). Internet of Things Architecture and Applications: A Survey. Indian Journal of Science and Technology. 9. 10.17485/ijst/2016/v9i45/106507.
[31]. M. Polychronaki, D. Kogias and C. Patrikakis, "Identity Management in Internet of Things with Blockchain", Blockchain based Internet of Things, pp. 209-236, 2022. Available: 10.1007/978-981-16-9260-4_9.

ΠΑΡΑΡΤΗΜΑ Α – Έξυπνα Συμβόλαια

DidFactory.sol

```
// SPDX-License-Identifier: GPL-3.0

pragma solidity ^0.8.12;

import "./safemath.sol";
import "@openzeppelin/contracts/utils/Strings.sol";

contract DidFactory {

    using SafeMath for uint256;
    using SafeMath32 for uint32;

    uint16 idDigits = 5;
    uint idModulus = 10 ** idDigits;
    uint32 nonce = 0;
    uint32 registry = 10000;

    struct Did {
        string scheme;
        string method;
        string path;
        string property;
        uint256 registrationNo;
        string did;
    }

    Did[] internal dids;
    Did[] internal devicedids;
    Did[] internal softhubdids;

    mapping (address => uint) internal identityToAccount;
    mapping (address => bool) internal accountIsRegistered;

    constructor() {
        accountIsRegistered[msg.sender] = true;
        _createDid("contract");
    }

    function createdid(string memory _property) public {
        require(accountIsRegistered[msg.sender] != true);
        accountIsRegistered[msg.sender] = true;
        _createDid(_property);
    }
```

```
    function _createDid(string memory _property) internal {
        uint256 _randRegistry = _generateRandomIdentifier();
        registry++;
        string memory did =
string.concat("did:","ethblockchain:","/application/mscres-
52/",_property,Strings.toString(_randRegistry));
        dids.push(Did("did", "ethblockchain", "/application/mscres-52/",
_property, _randRegistry, did));
        identityToAccount[msg.sender] = dids.length - 1;
    }

    function _createDeviceDid() internal returns(uint){
        uint256 _randRegistry = _generateRandomIdentifier();
        registry++;
        string memory did =
string.concat("did:","ethblockchain:","/application/mscres-
52/","device",Strings.toString(_randRegistry));
        devicedids.push(Did("did", "ethblockchain", "/application/mscres-52/",
"device", _randRegistry, did));
        return devicedids.length - 1;
    }

    function _createSofthubDid() internal returns(uint){
        uint256 _randRegistry = _generateRandomIdentifier();
        registry++;
        string memory did =
string.concat("did:","ethblockchain:","/application/mscres-
52/","softhub",Strings.toString(_randRegistry));
        softhubdids.push(Did("did", "ethblockchain", "/application/mscres-52/",
"softhub", _randRegistry, did));
        return softhubdids.length - 1;
    }

    function _generateRandomIdentifier() internal returns(uint) {
        nonce++;
        return uint(keccak256(abi.encodePacked(block.timestamp, msg.sender,
nonce))) % idModulus;
    }

// FUNCTIONS FOR DEBUGGING AND DEMONSTRATION PURPOSES ONLY
    function A1_showMyDID() public view returns (string memory) {
        return dids[identityToAccount[msg.sender]].did;
    }

}
```

IothubFactory.sol

```
// SPDX-License-Identifier: GPL-3.0

pragma solidity ^0.8.12;
import "./safemath.sol";
import "./didfactory.sol";
import "@openzeppelin/contracts/utils/Strings.sol";

contract IothubFactory is DidFactory{

    using SafeMath16 for uint16;
    using SafeMath32 for uint32;

    struct Device {
        uint32 deviceID;
        uint32 hubID;
        string deviceType;
    }

    struct Iothub {
        string DID;
        string ownerDID;
        string name;
    }

    Iothub[] internal softhubs;
    Device[] internal devices;

    mapping (uint32 => uint32[]) internal deviceHubHistory; //history of a
device's hub
    mapping (address => uint32) internal ownerToHub; // owner account to hub from
softhubs array
    mapping (string => uint32) internal rnToDev; //registration Number to Device
from devices array
    mapping (uint32 => uint32) internal devToHub; // device appointed to a hub
    mapping (uint32 => uint16) internal hubDeviceCount; // measure how many
devices a hub has claimed

    mapping (string => uint32) internal didToSofthub;
    mapping (string => uint32) internal didToDevice;

    constructor() {
        //create hub 0 of the contract
        require(keccak256(abi.encodePacked(dids[identityToAccount[msg.sender]].pr
operty)) == keccak256(abi.encodePacked("contract")));
        uint didIndex = _createSofthubDid();
```

```
            softhubs.push(Iothub(softhubdids[didIndex].did,dids[identityToAccoun
t[msg.sender]].did, "contracthub"));
        uint softhubIndex = 0;
        ownerToHub[msg.sender] = softhubIndex;
        didToSofthub[softhubdids[didIndex].did] = softhubIndex;
        hubDeviceCount[softhubIndex] = 0;
    }

    //create a new softHub
    function createSofthub(string memory _name) external {
        //check account's identity to be administrator
        require(keccak256(abi.encodePacked(dids[identityToAccount[msg.sender]].pr
operty)) == keccak256(abi.encodePacked("administrator")));
        uint didIndex = _createSofthubDid();
        softhubs.push(Iothub(softhubdids[didIndex].did,
dids[identityToAccount[msg.sender]].did, _name));
        uint softhubIndex = softhubs.length - 1;
        ownerToHub[msg.sender] = softhubIndex;  // !!Important!! each account can
only have one softhub registered as administrator
        didToSofthub[softhubdids[didIndex].did] = softhubIndex;
        hubDeviceCount[softhubIndex] = 0;
    }

    //register a new device to the contract
    function registerDevice(string memory _type, string memory _registrationNo)
external {
        uint didIndex = _createDeviceDid();
        devices.push(Device(didIndex, 0, _type));
        uint deviceIndex = devices.length - 1;
        rnToDev[_registrationNo] = deviceIndex;
        didToDevice[devicedids[didIndex].did] = deviceIndex;
        deviceHubHistory[deviceIndex].push(0);
        hubDeviceCount[0]++;
    }


// FUNCTIONS FOR DEBUGGING AND DEMONSTRATION PURPOSES ONLY
    function B1_showMySofthub() public view returns(Iothub memory){
        return softhubs[ownerToHub[msg.sender]];
    }

    function B2_showSofthubByDID(string memory _softhubDID) public view
returns(Iothub memory) {
        return softhubs[didToSofthub[_softhubDID]];
    }
```

```
    function B3_showHubDeviceCount(string memory _softhubDID) public view
returns(uint) {
        return hubDeviceCount[didToSofthub[_softhubDID]];
    }

    function B4_showDevicesBySofthub(string memory _softhubDID) public view
returns(string[] memory){
        string[] memory _devices = new string[]
(hubDeviceCount[didToSofthub[_softhubDID]]);
        uint counter = 0;
        for(uint i=0; i<devices.length; i++){
            if (devToHub[devices[i].deviceID] == didToSofthub[_softhubDID]) {
                _devices[counter] = devicedids[i].did;
                counter++;
            }
        }
        return _devices;
    }

    function B5_showDeviceDIDByRN(string memory _deviceRN) public view
returns(string memory){
        return devicedids[rnToDev[_deviceRN]].did;
    }

}
```

DeviceManagement.sol

```
// SPDX-License-Identifier: GPL-3.0

pragma solidity ^0.8.12;

import "./iothubFactory.sol";
import "@openzeppelin/contracts/utils/Strings.sol";

contract DeviceManagement is IothubFactory{

    function claimDevice(string memory _deviceRN) external {
        // check to see if device is unclaimed.
        require(devices[rnToDev[_deviceRN]].hubID == 0);
        require(devToHub[rnToDev[_deviceRN]] == 0);

        // complete the claim process.
        devices[rnToDev[_deviceRN]].hubID = identityToAccount[msg.sender];
        devToHub[rnToDev[_deviceRN]] = ownerToHub[msg.sender];

        // and change the device count for the hub
        hubDeviceCount[ownerToHub[msg.sender]] =
hubDeviceCount[ownerToHub[msg.sender]] + 1;

        // Log it to the History mapping
        deviceHubHistory[rnToDev[_deviceRN]].push(ownerToHub[msg.sender]);
    }

    function changeHub(string memory _deviceRN, address _targetOwner) external {
        // ckeck if the device belongs to the hub the sender is administrator of.
        require(ownerToHub[msg.sender] == devToHub[rnToDev[_deviceRN]]);

        //bind the device to the next owner
        devices[rnToDev[_deviceRN]].hubID = identityToAccount[_targetOwner];
        devToHub[rnToDev[_deviceRN]] = 0; // remember, hub 0 is always the one of
the contract.

        // and change the device count for the hub
        hubDeviceCount[ownerToHub[msg.sender]] =
hubDeviceCount[ownerToHub[msg.sender]] - 1;
    }

    function claimTransferedDevice(string memory _deviceRN) external {
        // check if the hub which the sender is administrator of is the same with
the one the device belongs to.
        require(keccak256(abi.encodePacked(dids[identityToAccount[msg.sender]].pr
operty)) == keccak256(abi.encodePacked("administrator")));
```

```
        // check if the binded address of the device is the same as the sender.
        require(identityToAccount[msg.sender] ==
devices[rnToDev[_deviceRN]].hubID);

        //complete the transfer.
        devices[rnToDev[_deviceRN]].hubID = identityToAccount[msg.sender];
        devToHub[rnToDev[_deviceRN]] = ownerToHub[msg.sender];

        // and change the device count for the hub
        hubDeviceCount[ownerToHub[msg.sender]] =
hubDeviceCount[ownerToHub[msg.sender]] + 1;

        // Log it to the History mapping
        deviceHubHistory[rnToDev[_deviceRN]].push(ownerToHub[msg.sender]);
    }

    function removeFromHub(string memory _deviceRN) external {
        // ckeck if the device belongs to the hub the sender is administrator of.
        require(ownerToHub[msg.sender] == devToHub[rnToDev[_deviceRN]]);

        // unclaim the device.
        devices[rnToDev[_deviceRN]].hubID = 0; //the did 0 is the one of the
contract
        devToHub[rnToDev[_deviceRN]] = 0; //hub 0 is always the one of the
contract

        // and change the device count for the hub
        hubDeviceCount[ownerToHub[msg.sender]] =
hubDeviceCount[ownerToHub[msg.sender]] - 1;

        // Log it to the History mapping
        deviceHubHistory[rnToDev[_deviceRN]].push(0);
    }

    // FUNCTIONS FOR DEBUGGING AND DEMONSTRATION PURPOSES ONLY
    function C1_showDeviceHistory(string memory _deviceDID) public view
returns(uint[] memory){
        return deviceHubHistory[didToDevice[_deviceDID]];
    }
}
```

Verfication.sol

```
// SPDX-License-Identifier: GPL-3.0

pragma solidity ^0.8.12;

import "./devicemanagement.sol";

contract Verification is DeviceManagement {

    function D1_verifyDeviceOwnership(string memory _deviceDID, address _address)
external view returns(bool) {
        if(devToHub[didToDevice[_deviceDID]] == ownerToHub[_address]){
            return true;
        } else { return false; }
    }

    function D2_verifySofthubOwnership(string memory _hubDID, address _address)
external view returns(bool) {
        if(ownerToHub[_address] == didToSofthub[_hubDID]){
            return true;
        } else { return false; }
    }

    function D3_verifyDeviceToSofthub(string memory _softhubDID, string memory
_deviceDID) external view returns(bool) {
        if(devToHub[didToDevice[_deviceDID]] == didToSofthub[_softhubDID]) {
            return true;
        } else { return false; }
    }

    function D4_verifyAdminToSofthub(string memory _softhubDID) external view
returns(bool) {
        if(ownerToHub[msg.sender] == didToSofthub[_softhubDID]){
            if(keccak256(abi.encodePacked(dids[identityToAccount[msg.sender]].pro
perty)) == keccak256(abi.encodePacked("administrator"))){
                return true;
            } else { return false; }
        }else { return false; }
    }

}
```

SafeMath.sol

```
// SPDX-License-Identifier: GPL-3.0

pragma solidity ^0.8.0;

/**
 * @title SafeMath
 * @dev Math operations with safety checks that throw on error
 */
library SafeMath {

  /**
  * @dev Multiplies two numbers, throws on overflow.
  */
  function mul(uint256 a, uint256 b) internal pure returns (uint256) {
    if (a == 0) {
      return 0;
    }
    uint256 c = a * b;
    assert(c / a == b);
    return c;
  }

  /**
  * @dev Integer division of two numbers, truncating the quotient.
  */
  function div(uint256 a, uint256 b) internal pure returns (uint256) {
    // assert(b > 0); // Solidity automatically throws when dividing by 0
    uint256 c = a / b;
    // assert(a == b * c + a % b); // There is no case in which this doesn't hold
    return c;
  }

  /**
  * @dev Subtracts two numbers, throws on overflow (i.e. if subtrahend is greater
than minuend).
  */
  function sub(uint256 a, uint256 b) internal pure returns (uint256) {
    assert(b <= a);
    return a - b;
  }

  /**
  * @dev Adds two numbers, throws on overflow.
  */
  function add(uint256 a, uint256 b) internal pure returns (uint256) {
```

```
    uint256 c = a + b;
    assert(c >= a);
    return c;
  }
}

/**
 * @title SafeMath32
 * @dev SafeMath library implemented for uint32
 */
library SafeMath32 {

  function mul(uint32 a, uint32 b) internal pure returns (uint32) {
    if (a == 0) {
      return 0;
    }
    uint32 c = a * b;
    assert(c / a == b);
    return c;
  }

  function div(uint32 a, uint32 b) internal pure returns (uint32) {
    // assert(b > 0); // Solidity automatically throws when dividing by 0
    uint32 c = a / b;
    // assert(a == b * c + a % b); // There is no case in which this doesn't hold
    return c;
  }

  function sub(uint32 a, uint32 b) internal pure returns (uint32) {
    assert(b <= a);
    return a - b;
  }

  function add(uint32 a, uint32 b) internal pure returns (uint32) {
    uint32 c = a + b;
    assert(c >= a);
    return c;
  }
}

/**
 * @title SafeMath16
 * @dev SafeMath library implemented for uint16
 */
library SafeMath16 {
```

```
  function mul(uint16 a, uint16 b) internal pure returns (uint16) {
    if (a == 0) {
      return 0;
    }
    uint16 c = a * b;
    assert(c / a == b);
    return c;
  }

  function div(uint16 a, uint16 b) internal pure returns (uint16) {
    // assert(b > 0); // Solidity automatically throws when dividing by 0
    uint16 c = a / b;
    // assert(a == b * c + a % b); // There is no case in which this doesn't hold
    return c;
  }

  function sub(uint16 a, uint16 b) internal pure returns (uint16) {
    assert(b <= a);
    return a - b;
  }

  function add(uint16 a, uint16 b) internal pure returns (uint16) {
    uint16 c = a + b;
    assert(c >= a);
    return c;
  }
}
```

Συναλλαγή 2: Δημιουργία ενός softhub με το ψευδώνυμο testhub1

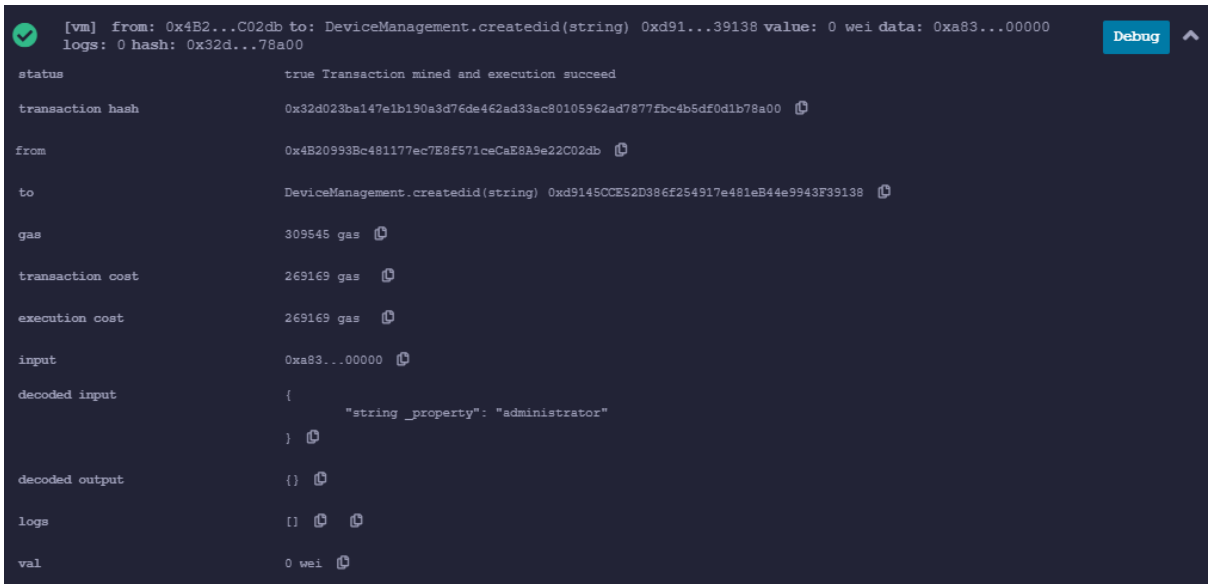
Συναλλαγή 3α: Εγγραφή συσκευής τύπου αισθητήρα θερμοκρασίας με σειριακό αριθμό abc1234

Συναλλαγή 3β: Εγγραφή συσκευής τύπου αισθητήρα καπνού με σειριακό αριθμό def5678

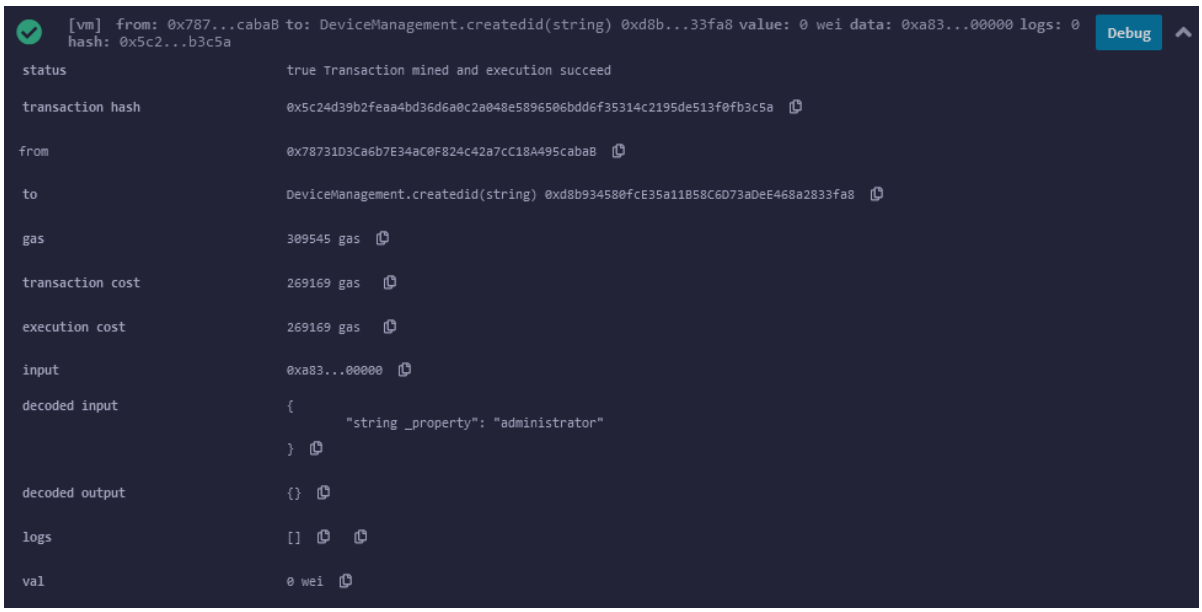
Συναλλαγή 4α: Διεκδίκηση της συσκευής με σειριακό αριθμό abc1234 από το testhub1 χρησιμοποιώντας τον αντίστοιχο λογαριασμό

Συναλλαγή 4β: Διεκδίκηση της συσκευής με σειριακό αριθμό abc1234 από το testhub1 χρησιμοποιώντας τον αντίστοιχο λογαριασμό

Συναλλαγή 5α: Δημιουργία προσωπικού DID με την ιδιότητα του διαχειριστή (administrator) χρησιμοποιώντας τον λογαριασμό 0x4B20993Bc481177ec7E8f571ceCaE8A9e22C02db



Συναλλαγή 5β: Δημιουργία προσωπικού DID με την ιδιότητα του διαχειριστή (administrator) χρησιμοποιώντας τον λογαριασμό 0x78731D3Ca6b7E34aC0F824c42a7cC18A495cabaB



Συναλλαγή 6α: Δημιουργία νέου softhub για τους προαναφερθέντες λογαριασμούς αντίστοιχα: testhub2

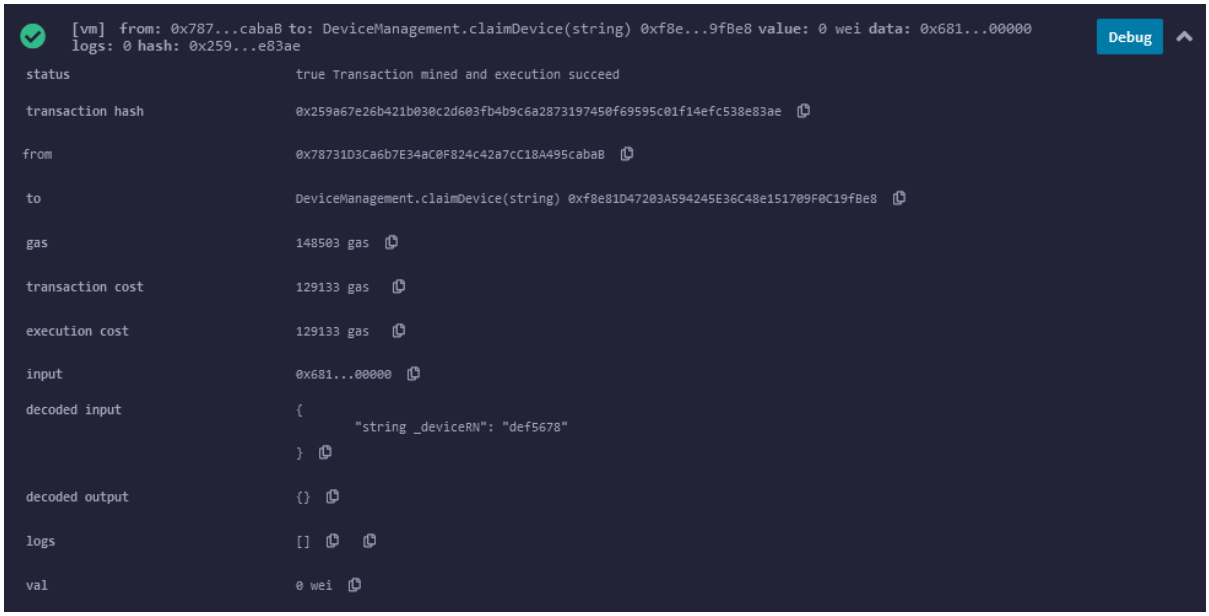
Συναλλαγή 6β: Δημιουργία νέου softhub για τους προαναφερθέντες λογαριασμούς αντίστοιχα: testhub3

Συναλλαγή 8: Εκκίνηση μεταφοράς πρώτης συσκευής (με σειριακό αριθμό abc1234) προς τον λογαριασμό που αντιστοιχεί στο softhub2

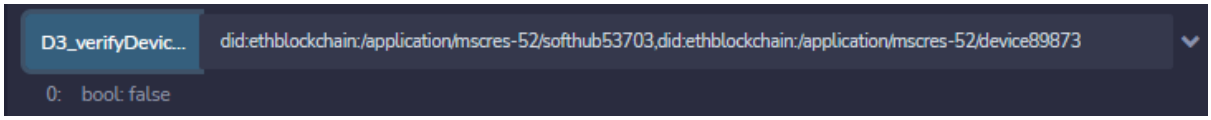
Συναλλαγή 10: Ολοκλήρωση μεταφοράς συσκευής με σειριακό αριθμό abc1234 από τον λογαριασμό που ανήκει το softhub2

Συναλλαγή 11: Αφαίρεση συσκευής με σειριακό αριθμό def5678 από το softhub1

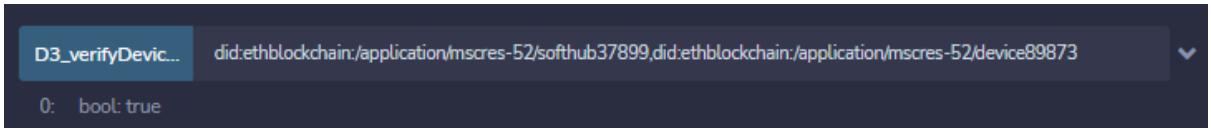
Συναλλαγή 12: Διεκδίκηση συσκευής με σειριακό αριθμό def5678 από το softhub3.



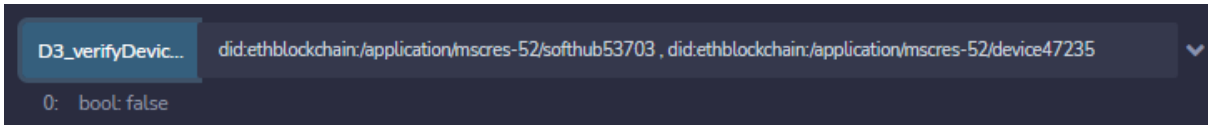
Συναλλαγή 13: Επαλήθευση ότι η συσκευή με σειριακό αριθμό abc1234 ανήκει στο softhub1



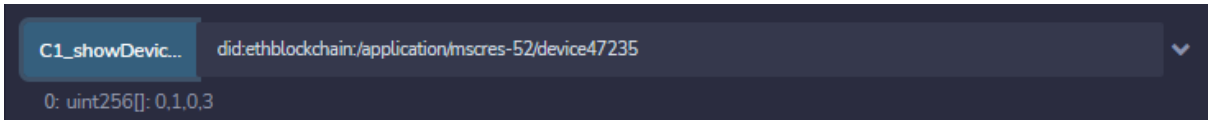
Συναλλαγή 14: Επαλήθευση ότι η συσκευή με σειριακό αριθμό abc1234 ανήκει στο softhub2



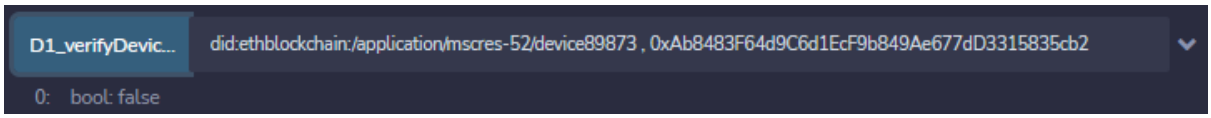
Συναλλαγή 15: Επαλήθευση ότι η συσκευή με σειριακό αριθμό. def5678 ανήκει στο softhub1



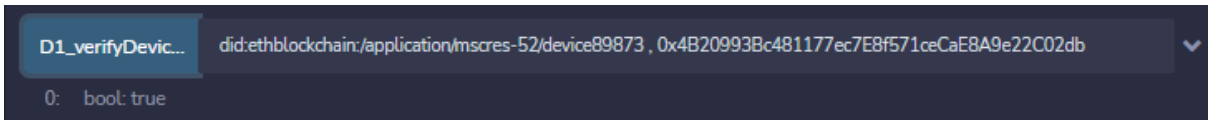
Συναλλαγή 16: Εμφάνιση ιστορικού συσκευής με σειριακό αριθμό def5678



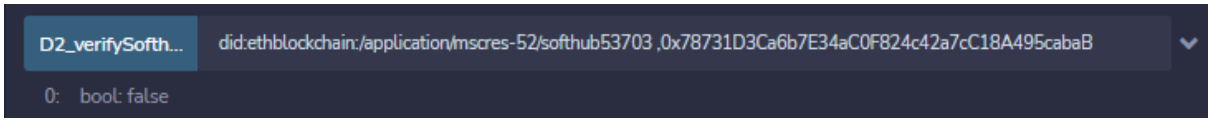
Συναλλαγή 17: Επαλήθευση ότι η συσκευή με σειριακό αριθμό abc1234 μπορεί να ελεγχθεί από τον αριθμό λογαριασμού του softhub1



Συναλλαγή 18: Επαλήθευση ότι η συσκευή με σειριακό αριθμό abc1234 μπορεί να ελεγχθεί από τον αριθμό λογαριασμού του softhub2



Συναλλαγή 19: Επαλήθευση ότι το softhub1 μπορεί να ελεγχθεί από τον αριθμό λογαριασμού του softhub2



Συναλλαγή 20: Επαλήθευση ότι το softhub1 μπορεί να ελεγχθεί από τον αριθμό λογαριασμού του softhub3