ΤΜΗΜΑ ΜΗΧΑΝΙΚΩΝ ΠΛΗΡΟΦΟΡΙΚΗΣ ΚΑΙ ΗΛΕΚΤΡΟΝΙΚΩΝ ΣΥΣΤΗΜΑΤΩΝ

ΠΡΟΓΡΑΜΜΑ ΜΕΤΑΠΤΥΧΙΑΚΩΝ ΣΠΟΥΔΩΝ
ΕΥΦΥΕΙΣ ΤΕΧΝΟΛΟΓΙΕΣ ΔΙΑΔΙΚΤΥΟΥ - WEB INTELLIGENCE

Αποκεντρωμένα συστήματα ταυτοποίησης και ελέγχου πρόσβασης σε δεδομένα χρηστών: Ανάπτυξη μοντέλου ελέγχου, ιχνηλασίας και ειδοποιήσεων κατά τη χρήση ιατρικών δεδομένων

ΜΕΤΑΠΤΥΧΙΑΚΗ ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ

του

ΝΙΚΟΛΑΟΥ ΤΣΑΚΙΡΗ

Επιβλέπων : Χρήστος Ηλιούδης
Καθηγητής, Δι.Πα.Ε.

Θεσσαλονίκη, Ιούνιος 2024

Αποκεντρωμένα συστήματα ταυτοποίησης και ελέγχου πρόσβασης σε δεδομένα χρηστών:

Ανάπτυξη μοντέλου ελέγχου, ιχνηλασίας και ειδοποιήσεων κατά τη χρήση ιατρικών δεδομένων

Κωδικός Δ.Ε. 23246

Ονοματεπώνυμο φοιτητή: Τσακίρης Νικόλαος

Ονοματεπώνυμο εισηγητή: Ηλιούδης Χρήστος

Ημερομηνία ανάληψης Δ.Ε.: 10/10/2023
Ημερομηνία περάτωσης Δ.Ε. 21/06/2024

Βεβαιώνω ότι είμαι ο συγγραφέας αυτής της εργασίας και ότι κάθε βοήθεια την οποία είχα για την προετοιμασία της είναι πλήρως αναγνωρισμένη και αναφέρεται στην εργασία. Επίσης, έχω καταγράψει τις όποιες πηγές από τις οποίες έκανα χρήση δεδομένων, ιδεών, εικόνων και κειμένου, είτε αυτές αναφέρονται ακριβώς είτε παραφρασμένες. Επιπλέον, βεβαιώνω ότι αυτή η εργασία προετοιμάστηκε από εμένα προσωπικά, ειδικά ως διπλωματική εργασία, στο Τμήμα Μηχανικών Πληροφορικής και Ηλεκτρονικών Συστημάτων του ΔΙ.ΠΑ.Ε.

Η έγκριση της διπλωματικής εργασίας από το Τμήμα Μηχανικών Πληροφορικής και Ηλεκτρονικών Συστημάτων του Διεθνούς Πανεπιστημίου της Ελλάδος, δεν υποδηλώνει απαραιτήτως και αποδοχή των απόψεων του συγγραφέα, εκ μέρους του Τμήματος.

Περίληψη

Η παρούσα διπλωματική εργασία παρουσιάζει ένα πρωτότυπο εθνικό σύστημα αποκεντρωμένης ταυτότητας, με το σύστημα αυτό να φιλοδοξεί να ανατρέψει την παραδοσιακή, συγκεντρωτική διαχείριση ψηφιακής ταυτότητας, η οποία πάσχει από αδυναμίες όπως η ευπάθεια σε κυβερνοεπιθέσεις και η έλλειψη ελέγχου των προσωπικών δεδομένων από τους χρήστες.

Η καινοτομία του προτεινόμενου συστήματος έγκειται στην αξιοποίηση προηγμένων τεχνολογιών οι οποίες σε συνδυασμό με τους λεπτομερείς μηχανισμούς συναίνεσης, διασφαλίζουν την επεκτασιμότητα, την ασφάλεια και το απόρρητο των δεδομένων των χρηστών. Ως κεντρικός πυλώνας της αποτελεί την μηχάνευση ασφαλών μεθόδων κοινοποίησης δεδομένων μεταξύ γιατρών/τρίτων φορέων και ασθενών, χωρίς να αποκαλύπτεται το περιεχόμενο αυτών σε τρίτους, και κύριο συστατικό το δικαίωμα στην παροχή η αφαίρεση της συγκατάθεσης από αυτόν στον οποίο ανήκουν τα πρωτογενή δεδομένα.

Η αποτελεσματικότητα του συστήματος αποδεικνύεται μέσα από την αξιολόγηση ενός λειτουργικού πρωτοτύπου σε ένα απτό σενάριο πραγματικού κόσμου. Ως εκ τούτου, τα αποτελέσματα αυτής της αξιολόγησης υπογραμμίζουν τη δυνατότητα του συστήματος να διαχειρίζεται με ασφάλεια ευαίσθητα δεδομένα υγείας, διασφαλίζοντας παράλληλα την αυτονομία και το απόρρητο των ασθενών.

Συνολικά, αυτή η έρευνα καταδεικνύει τη σκοπιμότητα και τα οφέλη ενός εθνικού συστήματος αποκεντρωμένων χαρακτηριστικών, ενώ παράλληλα προσφέρει μια ρεαλιστική προσέγγιση για την υλοποίηση ενός τέτοιου συστήματος.

Λέξεις Κλειδιά: · Αποκεντρωμένα Αναγνωριστικά (DID) · Αυτοκυρίαρχη Ταυτότητα (SSI) · Διαχείριση Δεδομένων Υγείας · Ψηφιακή Ταυτότητα · Ιδιωτικότητα · Ασφάλεια · Ηλεκτρονική Διακυβέρνηση · Έλεγχος Χρήστη · Διαχείριση Συναίνεσης · Διαλειτουργικότητα Δεδομένων · Κλιμακωσιμότητα · Ψηφιακός Μετασχηματισμός

«Decentralized authentication and access control systems for user data: Development of control, tracking and notification model during usage of medical data»

«Tsakiris Nikolas»

Abstract

This thesis presents a prototype national decentralized identity system, with this system aspiring to overturn traditional, centralized digital identity management, which suffers from weaknesses such as vulnerability to cyber-attacks and lack of control over personal data by users.

The innovation of the proposed system lies in the utilization of advanced technologies which, combined with detailed consent mechanisms, ensure scalability, security and privacy of user data. As its central pillar, it is the mechanization of secure data sharing methods between doctors/third parties and patients, without disclosing their content to third parties, and the main component is the right to provide or withdraw consent from the person to whom the primary data belongs.

The effectiveness of the system is demonstrated through the evaluation of a working prototype in a tangible real-world scenario. Therefore, the results of this evaluation highlight the system's ability to securely manage sensitive health data while ensuring patient autonomy and privacy. Overall, this research demonstrates the feasibility and benefits of a national decentralized feature system, while also offering a realistic approach to implementing such a system.

Keywords: · Decentralized Identity (DID) · Self-Sovereign Identity (SSI) · Healthcare Data Management · Digital Identity · Privacy · Security · E-governance · User Control · Consent Management · Data Interoperability · Scalability · Digital Transformation

Ευχαριστίες

Ευχαριστώ εγκάρδια τους οικείους μου που με στήριξαν καθολικά σε όλη τη διάρκεια των σπουδών μου, με πρώτη και κύρια τη σύντροφό μου. Επίσης, τους καθηγητές μου για την υποστήριξή τους, και ειδικότερα τον επιβλέπων μου κ. Χρήστο Ηλιούδη, για την εμπιστοσύνη, τη στήριξη, και τη συμβολή του. Τέλος, τις κοινότητες πληροφορίας, ανοικτών δεδομένων, και εργαλείων του διαδικτύου που προσφέρουν αδιαλείπτως στην ανάπτυξη και προώθηση της πραγματικής γνώσης.

Πίνακας περιεχομένων

Πίνακας Γραφημάτων

Συντομογραφίες

DID	Decentralized Identifier
GDPR	General Data Protection Regulation, Γενικός Κανονισμός για την Προστασία των Δεδομένων
PDF	Portable Document Format
SSI	Self-Sovereign Identity
W3C	World Wide Web Consortium
VC	Verifiable Credentials
DHT	Distributed Hash Table
MPT	Multiple Party Computation
AES	Advanced Encryption Standard
RSA	Rivest-Shamir-Adleman
ECC	Elliptic Curve Cryptography
PRE	Proxy Re-Encryption
OLTP	Online Transaction Processing
ACID	Atomicity, Consistency, Isolation, και Durability
API	Application Programming Interface
KYC	Know Your Customer
IOBP	Interorganizational Business Process
ZKP	Zero Knowledge Proof
IoT	Internet of Things

1

Εισαγωγή

1.1 Υπόβαθρο & Κίνητρο

Το θέμα της παρούσας εργασίας επιλέχθηκε λόγω της διαχρονικότητας που διέπει το βασικό θέμα αυτής, δηλαδή τη ταυτότητα και τα προσωπικά δεδομένα. Αυτές οι δύο έννοιες αποτελούν αλληλένδετο κομμάτι σύστασης των σύγχρονων κοινωνιών οι οποίες χρήζουν όλο και μεγαλύτερης κατανόησης, αρχιτεκτονικής και υλοποίησης μοντέρνων συστημάτων που σέβονται τόσο την αντοχή τους στο χρόνο, όσο και τη προστασία των πολιτών. Ως απόρροια αυτών των αναγκών, οι σύγχρονες τεχνολογίες όπως οι αποκεντρωμένες βάσεις δεδομένων και τα Blockchain(s) [1] ανοίγουν τον δρόμο προς την υλοποίηση αυτών των ιδεών με τελικό προορισμό την εγκαθίδρυση ενός νέου είδους ταυτότητας, της επονομαζόμενης ως αυτο-κυριαρχούμενης (self-sovereign) [2]. Ως εκ τούτου, οι ανάπτυξη αντίστοιχων συστημάτων διαχείρισης ταυτοτικών πόρων αποτελεί μείζον ζήτημα αλλά και ευκαιρία για την εγκαθίδρυση μιας ενιαίας αρχιτεκτονικής για αυτού του είδους τα ταυτοτικά συστήματα. Σαν συνέπεια, η εργασία χωρίζεται σε ένα δίπολο μεταξύ θεωρητικής ανάλυσης και ενός προτύπου για το πως μπορεί να λειτουργήσει ένα τέτοιο πολυπαραγοντικό σύστημα, το οποίο θα εξυπηρετεί όχι μόνον τον αποκεντρωμένο χαρακτήρα αλλά και την δυνατότητα στους άμεσα εμπλεκόμενους χρήστες να κατέχουν ορατότητα και ικανότητα εξουσιοδότησης αναφορικά με το ποιός, που, πότε και γιατί αιτήθηκε τα δικά τους δεδομένα. Αναλυτικότερα, ορισμένοι λόγοι που καθόρισαν την επιλογή του θέματος περιλαμβάνουν:

- ***Ανάγκη για αυτοδιάθεση***: Η ψηφιακή εποχή δημιούργησε ανάγκη για συστήματα που επιτρέπουν στους χρήστες τον έλεγχο και τη διαχείριση των προσωπικών τους δεδομένων. Η αυτοδιάθεση, δηλαδή η δυνατότητα των χρηστών να έχουν τον πλήρη έλεγχο επί των προσωπικών τους πληροφοριών αποτελεί κρίσιμο παράγοντα για την προστασία της ιδιωτικότητας και την αποφυγή εκμετάλλευσης των δεδομένων από τρίτους και κακόβουλες οντότητες.
- ***Εξουσιοδότηση και ενημέρωση με επίκεντρο τον χρήστη***: Η μετάβαση της εξουσίας από τον πάροχο/τρίτο στον κύριο πελάτη αποτελεί τον κορμό αυτής της έρευνας. Κάθε πτυχή η οποία μπορεί να αναπλάσει το παλαιό τρόπο διαχείρισης δεδομένων αποτελεί ζήτημα έρευνας.
- ***Εξέλιξη τεχνοτροπίας***: Η συνεχής εξέλιξη των τεχνολογιών όσο και η αυξανόμενη απειλή από κυβερνοεπιθέσεις και παραβιάσεις απαιτεί την ανάπτυξη συστημάτων που εγγυώνται την ασφάλεια των προσωπικών δεδομένων.
- ***Κοινωνική Επιταγή ως Ανάγκη Προόδου***: Η ανάγκη για ασφαλή και αποτελεσματικά συστήματα διαχείρισης ταυτότητας είναι σημαντική σε διάφορους τομείς, όπως η υγεία, οι οικονομικές συναλλαγές και η διακυβέρνηση, ήτοι σε κάθε φάσμα της ζωής των ανθρώπων, οπότε η απόρροια αυτής είναι η πολιτειακή απαίτηση για ρηξικέλευθες λύσεις.

1.2 Εντοπισμός του Προβλήματος

Η συνεχώς αυξανόμενη ψηφιοποίηση των δεδομένων έχει οδηγήσει αναπόφευκτα σε μια μετάβαση νέων προκλήσεων που άπτονται της διαχείρισης, προστασίας και μεταφοράς προσωπικών δεδομένων. Τα παραδοσιακά κεντροποιημένα συστήματα έχουν αποδείξει επανειλημμένα τη χρησιμότητά τους για τουλάχιστον 15 χρόνια [3, 4], ωστόσο μαζί με αυτά ηγέρθησαν ορισμένα προβλήματα που εκ των πραγμάτων απορρέουν από τη φύση τους. Ειδικότερα, πρώτο και κύριο συστατικό αποτελεί η απαρχή της ιδιοκτησίας των δεδομένων, και ιδιοκτησία ως έννοια σημαίνει την απόλυτη ελευθερία μοιράσματος και εξουσιοδότησης αξιοποίησης αυτών σε τρίτες οντότητες. Στα κεντροποιημένα συστήματα, το ρόλο αυτόν αναλαμβάνουν οι πάροχοι που προσφέρουν την εκάστοτε υπηρεσία, καθιστώντας τους ίδιους πληρεξούσιους εντός ενός κυκλώματος δεδομένων στο οποίο ο άμεσα ενδιαφερόμενος - ο "***κατέχων ως αρχή"*** τα δεδομένα αυτά - πολίτης τίθεται σε τελευταία μοίρα αναφορικά με τις διεργασίες των δεδομένων του. Επομένως, αναδύεται αυτομάτως η ανάγκη για αποκεντροποίηση, δηλαδή το περιορισμό των εκάστοτε παρόχων στα απολύτως απαραίτητα και τη διαμόρφωση ενός ενιαίου συστήματος στο οποίο ο χρήστης μπορεί να συμμετέχει με ένα μοναδιαίο αναγνωριστικό, ετεροχρονισμένα και σε ετερόκλητους παρόχους υπηρεσιών. Γενικότερα, η κεντροποίηση μπορεί να αποβεί επικίνδυνη όταν ένας πάροχος προφασίζεται τη μη συμμόρφωση ενός χρήστη στα του πρωτοκόλλου του, με έσχατη συνέπεια τον περιορισμό η και τη πλήρη απαγόρευση της χρήσης της υπηρεσίας άνευ εξηγήσεων. Φυσικά, το τελευταίο διέπεται και από νομικές διατάξεις

οι οποίες δεν αναλύονται στη παρούσα εργασία, ωστόσο όμως το διμερές αυτό πρόβλημα αποτελεί τη κύρια έρευνα με αλληλένδετα τα δύο προαναφερθέντα συστατικά.

1.3 Στόχοι

Ως πρώτιστοι στόχοι της εργασίας μπορούν να τεθούν οι εξής παρακάτω:

Α. **Σχεδιασμός ενός Συστήματος Αυτοκυριαρχούμενης Διακριτικής Ταυτότητας (Decentralized Identifier, DID)**: Ο πρώτος στόχος είναι η σχεδίαση ενός προηγμένου συστήματος DID που θα επιτρέπει στους χρήστες να έχουν πλήρη έλεγχο επί των προσωπικών τους δεδομένων, με άμεση υλοποίηση της έννοιας της ***συγκατάθεσης από τρίτους***. Για λόγους πρακτικότητας, η υλοποίηση θα προσομοιώνει ένα απλουστευμένο σύστημα υγείας.

Β. **Υλοποίηση Τεχνολογικών Συστατικών**: ο δεύτερος σκοπός πρεσβεύει τη ανάπτυξη τεχνολογικών συστατικών που απαιτούνται για το σύστημα DID, συμπεριλαμβανομένων των μονάδων δημιουργίας και διαχείρισης DID, των μηχανισμών κρυπτογράφησης πληρεξουσίου, και των συστημάτων κατανεμημένης αποθήκευσης δεδομένων.

Γ. **Δοκιμή και Αξιολόγηση Πρότυπης Εφαρμογής**: Τρίτον πραγματοποιούνται δοκιμές του αναπτυγμένου συστήματος για να αξιολογηθεί η επίδοσή του σε όρους ασφάλειας, αποδοτικότητας και χρηστικότητας, όλα αυτά συγκριτικά με τυπικά κεντροποιημένα συστήματα.

Δ. **Εφαρμογή σε Σενάρια Χρήσης**: Τέλος, ο τέταρτος στόχος είναι η εφαρμογή του αναπτυγμένου συστήματος σε πραγματικά σενάρια χρήσης στον τομέα της υγείας υπό πρότυπη και πειραματική μορφή, προκειμένου να αξιολογηθεί η αποτελεσματικότητά του στην πράξη και να προταθούν εν δυνάμει βελτιώσεις.

1.3.1 Δυνητική Ενσωμάτωση GDPR

Ο Γενικός Κανονισμός Προστασίας Δεδομένων (GDPR) [5] είναι ένας κανονισμός που θεσπίστηκε με στόχο την προστασία των προσωπικών δεδομένων των πολιτών της Ευρωπαϊκής Ένωσης και συνδέεται αρρήκτως με το θέμα της διαχείρισης ταυτότητας και της προστασίας των αυτών όντας περιέχων σαφείς οδηγίες για το τρίπτυχο ***αποθήκευση-επεξεργασία-μεταφορά***. Ένας από τους βασικούς τρόπους με τους οποίους η εφαρμογή που προτείνεται μπορεί να συμβάλει στην τήρηση του GDPR είναι μέσω της **αυτοδιάθεσης** των δεδομένων.

Ειδικότερα, η συμμόρφωση με τον GDPR απαιτεί αφενός μεν την ύπαρξη συγκεκριμένων μέτρων και πρακτικών που εξασφαλίζουν κατ' ελάχιστον τις στοιχειώδεις αρχές του παραπάνω τριπτύχου, αφετέρου δε επιτρέπουν περιθώριο στη μελέτη καινοφανών τεχνολογικών στοιβών και τεχνοτροπιών. Στα πλαίσια εφαρμογής του προτεινόμενου συστήματος διαχείρισης ταυτότητας υφίστανται ποικίλοι τρόποι με τους οποίους δύναται να γίνει αντιληπτή η σύνδεσή του με το GDPR:

- **Διαφάνεια και Ενημέρωση** [6]: Βασική αρχή του GDPR είναι η απαίτηση για διαφάνεια στην επεξεργασία των προσωπικών δεδομένων με τους πολίτες απαραίτητα ενημερωμένου για τη συλλογή και τη χρήση των δεδομένων τους. Το σύστημα οφείλει να παρέχει διαφανή διαδικασία που εξηγεί πώς συλλέγονται, αποθηκεύονται και χρησιμοποιούνται τα δεδομένα και η έλευση αυτών.
- **Αυτοδιάθεση** [7, 8, 9] : Ένας τρόπος για να επιτευχθεί η συμμόρφωση με το GDPR είναι μέσω της υλοποίησης της αυτοδιαθέσεως των δεδομένων. Οι χρήστες πρέπει να έχουν τη δυνατότητα να ελέγχουν ποια δεδομένα συλλέγονται για αυτούς και πώς χρησιμοποιούνται. Η διαφορά με το πρώτο χαρακτηριστικό έγκειται στον ενεργό ρόλο που καλείται να αναλάβει ο χρήστης.
- **Διαγραφή Δεδομένων (Right to Erasure)** [10]: Γνωστό και ως *"the right to be forgotten"*, οι πολίτες έχουν το αναπαλλοτρίωτο δικαίωμα να αιτηθούν διαγραφής των προσωπικών τους δεδομένων από το σύστημα. Οποιαδήποτε διαδικασία διαχείρισης ταυτότητας πρέπει οπωσδήποτε να εμπεριέχει μηχανισμούς που επιτρέπουν διαγραφή δεδομένων χρήστη εφόσον αυτός το επιθυμεί.
- **Ασφάλεια Δεδομένων** [11, 12, 13]: Η ασφάλεια των προσωπικών δεδομένων είναι ζωτικής σημασίας για τη συμμόρφωση με το GDPR, με τις προσωπικές πληροφορίες να προστατεύονται από μη εξουσιοδοτημένη πρόσβαση, η και απώλεια.
- **Ευθύνη (Accountability)** [14]: Οι άμεσα υπεύθυνοι για επεξεργασία δεδομένων πρέπει να είναι εις θέσιν να επιδείξουν συμμόρφωση με τους κανονισμούς και τα αντίστοιχα νομοθετήματα, κάτι το οποίο επιτάσσει και την ανάλογη επιμόρφωσή τους επάνω στο σχετικό ζήτημα και τις αντίστοιχες διατάξεις αυτού. Ως εκ τούτου, ένα ταυτοτικό σύστημα χτίζεται εκ του μηδενός με ενσωματωμένα εργαλεία όλων των δραστηριοτήτων που αφορούν τα προσωπικά δεδομένα -από πλευράς μηχανικών- , ώστε να είναι δυνατή η ανίχνευση και η αντιμετώπιση πιθανών παραβιάσεων

Εικόνα 1: Λογότυπο του GDPR [15]

1.4 Συνεισφορά

Η παρούσα διπλωματική εργασία συμβάλλει σημαντικά στο εξελισσόμενο τοπίο της αποκεντρωμένης διαχείρισης ταυτότητας. Με το σχεδιασμό, την εφαρμογή και την αυστηρή αξιολόγηση ενός πρωτότυπου εθνικού συστήματος DID, παρέχει μια πρακτική βάση για μελλοντικές προσπάθειες σε αυτόν τον τομέα.

Βασικές συνεισφορές:

- **Νέος Σχεδιασμός Συστήματος**: Ενοποίηση ποικίλων τεχνολογιών και εκ του μή όντος αρχιτεκτονική σε ένα αποκεντρωμένο πλαίσιο ταυτότητας.
- **Υλοποίηση πρωτότυπου μηχανισμού**: Αυστηρές δοκιμές του πρωτοτύπου σε προσομοιωμένα σενάριο, παρέχοντας πολύτιμες πληροφορίες για την απόδοση, την επεκτασιμότητα, την ασφάλεια, και φυσικά τη εν γένει λειτουργικότητα.
- **Προσέγγιση με επίκεντρο τον χρήστη**: Έμφαση στον έλεγχο των χρηστών, στους λεπτομερείς μηχανισμούς συναίνεσης σε τρίτους και στη διαισθητική σχεδίαση διεπαφής.
- **Πηγή περαιτέρω έρευνας και ανάπτυξης**: Προσδιορισμός περιορισμών και συστάσεων για περαιτέρω έρευνα στους τομείς της επεκτασιμότητας, της ασφάλειας και της εμπειρίας χρήστη, προσβλέποντας στη στροφή μηχανικών και επιστημόνων προς τη νέα ταυτοτική κατάσταση, δηλαδή με επίκεντρο το χρήστη.
- **Εφαρμογές σε πραγματικό κόσμο**: Πρόταση μιας καινοτόμας ιδέας με δυνατότητα μετασχηματισμού τομέων όπως η υγειονομική περίθαλψη, τα οικονομικά και οι δημόσιες υπηρεσίες μέσω της ασφαλούς και αποτελεσματικής διαχείρισης ταυτότητας.

Αυτή η διπλωματική εργασία χρησιμεύει ως πολύτιμη πηγή για ερευνητές, προγραμματιστές και υπεύθυνους χάραξης πολιτικής που επιδιώκουν να κατανοήσουν και να προωθήσουν το πεδίο της αποκεντρωμένης ταυτότητας. Ως εκ τούτου, τα ευρήματά και οι γνώσεις που πραγματεύονται μπορούν να ενημερώσουν την ανάπτυξη μελλοντικών συστημάτων ταυτότητας που ενδυναμώνουν τα άτομα και τον απλό πολίτη ως πρώτιστο μέλημα.

1.5 Εύρος & Περιορισμοί

Το τρέχον έργο, ενώ καταδεικνύει τις βασικές λειτουργίες και τα πλεονεκτήματα ενός αποκεντρωμένου συστήματος ταυτότητας, περιορίζεται εγγενώς από τη φύση του ως πρωτοτύπου εντός των περιορισμών μιας μεταπτυχιακής εργασίας. Ως απόδειξη της ιδέας, δεν περιλαμβάνει όλες τις πτυχές μιας λύσης έτοιμη για παραγωγή. Για παράδειγμα, μέτρα ασφαλείας πλήρους κλίμακας, όπως η προηγμένη κρυπτογραφία και η ισχυρή διαχείριση κλειδιών, μαζί με τον ολοκληρωμένο χειρισμό σφαλμάτων, ενδέχεται να μην υλοποιηθούν πλήρως. Η διεπαφή χρήστη που βασίζεται σε τερματικό, αν και είναι αποτελεσματική για την επίδειξη βασικών αρχών, στερείται της φιλικότητας

προς το χρήστη και του οπτικού πλούτου που αναμένεται στις εφαρμογές του πραγματικού κόσμου. Επιπλέον, η τρέχουσα εφαρμογή εστιάζει σε συγκεκριμένους τύπους δεδομένων, όπως τα αρχεία υγείας (τα οποία χρησιμοποιήθηκαν υπό μορφής Portable Document Format, PDF), παραβλέποντας ενδεχομένως το ποικίλο φάσμα δεδομένων που θα μπορούσε να διαχειριστεί ένα ολοκληρωμένο αποκεντρωμένο σύστημα ταυτότητας. Παρόλο που η αρχιτεκτονική στοχεύει στην επεκτασιμότητα, η διεξοδική δοκιμή υπό ακραίες συνθήκες φορτίου είναι περιορισμένη λόγω των περιορισμών πόρων σε ένα ακαδημαϊκό περιβάλλον. Το πεδίο εφαρμογής του έργου αποκλείει επίσης την ολοκληρωμένη ενοποίηση με εξωτερικές υπηρεσίες και παλαιού τύπου συστήματα, απαραίτητα για την ανάπτυξη σε πραγματικό κόσμο. Αυτό περιλαμβάνει πτυχές όπως η συμμόρφωση με διάφορα ρυθμιστικά πλαίσια και η αλληλεπίδραση με τους υπάρχοντες παρόχους ταυτότητας. Η εις βάθος έρευνα των χρηστών και οι βρόχοι ανατροφοδότησης, ζωτικής σημασίας για τη βελτίωση της χρηστικότητας και την κατανόηση των αναγκών των χρηστών, περιορίζονται από χρονικούς περιορισμούς.

Επίσης, η καθιέρωση ενός ισχυρού μοντέλου διακυβέρνησης για ένα αποκεντρωμένο σύστημα ταυτότητας, που θα περιλαμβάνει διαδικασίες λήψης αποφάσεων, επίλυση διαφορών και δέσμευση της κοινότητας, είναι ένα σύνθετο έργο που μπορεί να μην διερευνηθεί πλήρως στο τρέχον πεδίο εφαρμογής του ημετέρου αυτού έργου. Αναφορικά με το τεχνικό και υπολογιστικό σκέλος, πραγματοποιήθηκαν προκαταρκτικές δοκιμές απόδοσης, αλλά η συνολική συγκριτική αξιολόγηση υπό διαφορετικές, πραγματικές συνθήκες περιορίστηκε από πόρους. Κατά συνέπεια, παραμένουν ερωτήματα σχετικά με την υπολογιστική απόδοση και την επεκτασιμότητα του συστήματος σε ακραία σενάρια. Επιπλέον, οι μηχανισμοί ασφαλείας του πρωτοτύπου λειτουργούν εκ προοιμίου με ένα απλουστευμένο μοντέλο απειλής.

Η ανάπτυξη στον πραγματικό κόσμο θα απαιτούσε μια πιο εκτεταμένη αξιολόγηση των πιθανών τρωτών σημείων, συμπεριλαμβανομένων εκείνων που μπορεί να εμφανιστούν σε μεγαλύτερα, πιο πολύπλοκα περιβάλλοντα. Οι υπολογιστικοί πόροι που είναι διαθέσιμοι κατά την ανάπτυξη (ισχύς επεξεργασίας, μνήμη, αποθήκευση) ενδέχεται να μην αντικατοπτρίζουν τις απαιτήσεις μιας εγκατάστασης σε εθνικό επίπεδο. Κατά συνέπεια, η απόδοση κάτω από μεγάλα φορτία ή η ικανότητα χειρισμού αποθήκευσης δεδομένων μεγάλης κλίμακας ενδέχεται να μην είναι πλήρως αντιπροσωπευτική. Επιπλέον, η βελτιστοποίηση του πρωτοτύπου για συγκεκριμένα περιβάλλοντα υλικού και λογισμικού θα μπορούσε να οδηγήσει σε απροσδόκητα σημεία συμφόρησης κατά την προσαρμογή σε διάφορες υποδομές. Αν και ελήφθη υπόψη η αποδοτικότητα, μια εις βάθος ανάλυση της κατανάλωσης ενέργειας για διάφορες λειτουργίες παραμένει ανεξερεύνητη, ένας κρίσιμος παράγοντας για βιώσιμη μεγάλης κλίμακας ανάπτυξη. Τέλος , η ανθεκτικότητα του συστήματος σε απροσδόκητες βλάβες ή κακόβουλη είσοδο δεν έχει ελεγχθεί διεξοδικά. Ένα σύστημα έτοιμο για παραγωγή θα απαιτούσε ολοκληρωμένους μηχανισμούς ανοχής σφαλμάτων και ελαστικότητας.

1.6 Οργάνωση της Εργασίας

1. **Εισαγωγή**: Η εργασία ξεκινά αναλύοντας τα ζητήματα που είναι εγγενή στα παραδοσιακά, κεντρικά συστήματα διαχείρισης ταυτότητας, επισημαίνοντας τόσο τα τρωτά σημεία όσο και την έλλειψη ελέγχου των προσωπικών δεδομένων από τους χρήστες. Ως συνέχεια, εισάγει τις βασικές αρχές της αυτοκυριαρχίας ταυτότητας (Self Sovereign Identity, SSI) και των DIDs, ούτως ώστε να εξηγήσει την εννοιολογική εξήγηση στο πώς προσφέρουν μια πιο ασφαλή και χρηστοκεντρική προσέγγιση.
2. **Θεωρητικό πλαίσιο**: Η μελέτη δημιουργεί μια σταθερή θεωρητική βάση εμβαθύνοντας στα υπάρχοντα μοντέλα DID και στο πρότυπο αυτών του World Wide Web Consortium (W3C). Παρέχει μια ολοκληρωμένη επισκόπηση των κρυπτογραφικών τεχνικών όπως η συμμετρική και η κρυπτογράφηση δημόσιου κλειδιού, μαζί με την καινοτόμο ιδέα της εκ νέου κρυπτογράφησης μεσολάβησης. Επιπλέον, διερευνά την επεξεργασία ροής δεδομένων και τον ρόλο των αποκεντρωμένων και κατανεμημένων βάσεων δεδομένων.
3. **Αρχιτεκτονική συστήματος**: Αυτή η ενότητα περιγράφει σχολαστικά την αρχιτεκτονική του προτεινόμενου εθνικού συστήματος DID, καλύπτοντας τις απαιτήσεις σχεδιασμού, τις περιγραφές αρθρωμάτων λογισμικού και τις ιδιαιτερότητες υλοποίησης. Συνοπτικά, εξηγεί ενδελεχώ το πώς εκδίδονται και διαχειρίζονται τα DID, πώς κρυπτογραφούνται και κοινοποιούνται με ασφάλεια τα δεδομένα μέσω επανακρυπτογράφησης και πώς το σύστημα έχει σχεδιαστεί για αποκεντρωμένη αποθήκευση και διανομή δεδομένων. Επεξηγείται επίσης η χρήση του Apache Kafka για ενορχήστρωση ροής εργασιών και παρακολούθηση δεδομένων.
4. **Μελέτη περίπτωσης υγειονομικής περίθαλψης**: Για να καταδειχθεί η δυνατότητα εφαρμογής του συστήματος στον πραγματικό κόσμο, παρουσιάζεται μια απλή μελέτη περίπτωσης στον τομέα της υγειονομικής περίθαλψης. Αυτό δείχνει πώς το σύστημα DID μπορεί να χρησιμοποιηθεί αποτελεσματικά για την ασφαλή και αποτελεσματική διαχείριση των ιατρικών δεδομένων ασθενών.
5. **Ευρήματα & Αξιολόγηση**: Σε συνέχεια, η εργασία αξιολογεί την απόδοση και την ασφάλεια του συστήματος εν λόγω. Ειδικότερα, αναλύει πλεονεκτήματα και μειονεκτήματα της προτεινόμενης προσέγγισης σε σύγκριση με άλλες λύσεις, και κυρίως με συστήματα που βασίζονται σε blockchain. Αυτή η αξιολόγηση παρέχει πολύτιμες πληροφορίες για τα δυνατά σημεία του συστήματος και τους τομείς προς βελτίωση.
6. **Συμπέρασμα και μελλοντική εργασία**: Εν κατακλείδι, η εργασία ολοκληρώνεται συνοψίζοντας τα βασικά ευρήματα, δίνοντας έμφαση στη σημαντική συμβολή του προτεινόμενου συστήματος DID στην ενίσχυση της ασφάλειας, της ιδιωτικότητας και της αυτονομίας των χρηστών στη διαχείριση ψηφιακής ταυτότητας. Τέλος, ο το τελευταίο σκέλος περιγράφει πιθανές οδούς για

μελλοντική έρευνα, συμπεριλαμβανομένης της εξερεύνησης αναδυόμενων τεχνολογιών και της περαιτέρω βελτιστοποίησης της απόδοσης και της ασφάλειας του συστήματος.

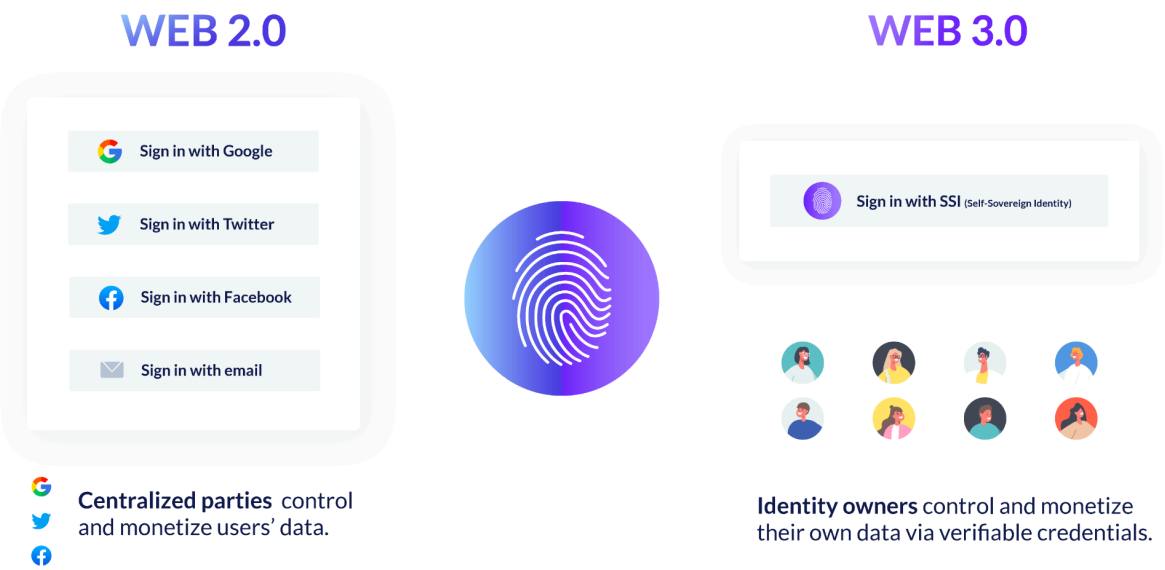
2

Βιβλιογραφική Ανασκόπηση

2.1 Επισκόπηση της Αυτο-κυριαρχούμενης Ταυτότητας

Η SSI αναφέρεται σε ένα ιδιαίτερο και συνάμα σύγχρονο πρότυπο ταυτοποίησης που επιτρέπει σε ένα άτομο, οργανισμό ή συσκευή να έχει πλήρη έλεγχο και διαχείριση της ταυτότητάς τους χωρίς την εμπλοκή τρίτων οργανισμών ή αρχών, κάτι το οποίο έρχεται σε πλήρη αντίθεση με το πως είναι εδραιωμένα τα υπάρχοντα ταυτοτικά συστήματα. Ειδικότερα, πυρήνας αυτού του μοντέλου είναι το τρίπτυχο των αυτο-υπόγραφων, αυτο-εκδιδόμενων και αυτο-επαληθεύσιμων ταυτοτήτων. Στην πράξη, ένα άτομο ή οργανισμός μπορεί να δημιουργήσει τη δική του ταυτότητα χρησιμοποιώντας κρυπτογραφικά κλειδιά, με αυτήν να περιλαμβάνει πληροφορίες όπως όνομα, διεύθυνση, δημόσιο κλειδί και άλλα προσωπικά δεδομένα. Ως εκ τούτου, οι εκάστοτε κάτοχοι των ταυτοτήτων αυτών διαχειρίζονται τις πληροφορίες τους και αποφασίζουν με ποιον τρόπο θα τις μοιραστούν, πότε και με ποιον, ενώ σε παρόμοιο φάσμα μπορούν να αποφασίζουν και ποιό μέρος από τα ταυτοτικά πεδία θα αποκαλύπτουν. Το συγκεκριμένο μοντέλο προέκυψε ως ριζοσπαστική αντίδραση στα περιοριστικά μοντέλα ταυτοποίησης που βασίζονται σε κεντρικές αρχές, όπως κυβερνητικές αρχές ή εταιρικά συστήματα, και αναλαμβάνει να προσδώσει μια νέα πνοή στο κατεστημένο σύστημα προσδοκώντας τα άτομα και τους οργανισμούς να αποκτήσουν μεγαλύτερο έλεγχο, ιδιωτικότητα και αυτοδιάθεση στη διαχείριση των προσωπικών τους δεδομένων. Φυσικά, με αυτή τη τεχνολογία μπορεί να καλυφθεί πληθώρα εφαρμογών όπως η ψηφιακή ταυτοποίηση, οι ιατρικές εγγραφές, οι ψηφιακές συναλλαγές και γενικά η διαχείριση της ταυτότητας σε οποιοδήποτε διαδικτυακό περιβάλλον. Αυτοί οι τύποι ταυτότητας προωθούνται από αρκετούς φορείς και κοινότητες όπως αυτό Ιδρύματος Αποκεντρωμένης Ταυτότητας (Decentralized Identity Foundation) [16], και τεχνολογικές εταιρείες όπως η Microsoft [17], η IBM [18] και η Evernym [19], οι οποίες μάλιστα έχουν αναπτύξει σχετιζόμενα πρωτόκολλα και πλατφόρμες.

Συνολικά, η αυτοκυριαρχούμενη ταυτότητα αντιπροσωπεύει έναν καινοτόμο τρόπο προσέγγισης της ταυτότητας και της ταυτοποίησης, θέτοντας τον ίδιο τον χρήστη στο επίκεντρο της διαδικασίας. Στον αντίποδα, τα υπάρχοντα παραδοσιακά συστήματα ψηφιακής ταυτοποίησης βασίζονται σε κεντρικές αρχές που διαχειρίζονται τις διαδικτυακές μας ταυτότητες, κάτι το οποίο μπορεί ευκόλως να οδηγήσει σε έλλειψη ελέγχου των χρηστών και σε κινδύνους για το απόρρητο με ανεξέλεγκτο αντίκτυπο. Επομένως, η SSI στοχεύει να φέρει επανάσταση σε αυτό καθιστώντας τα άτομα αυτούς που άρχουν στον πλήρη έλεγχο της ψηφιακής τους ταυτότητας. Ειδικότερα, τα άτομα πρέπει πλέον να κατέχουν και διαχειρίζονται τα δεδομένα ταυτότητάς τους, επιλέγοντας πώς και με ποιον θα κοινοποιηθούν, ήτοι οφείλουν να αποκτήσουν έναν πιό ενεργό ρόλο.



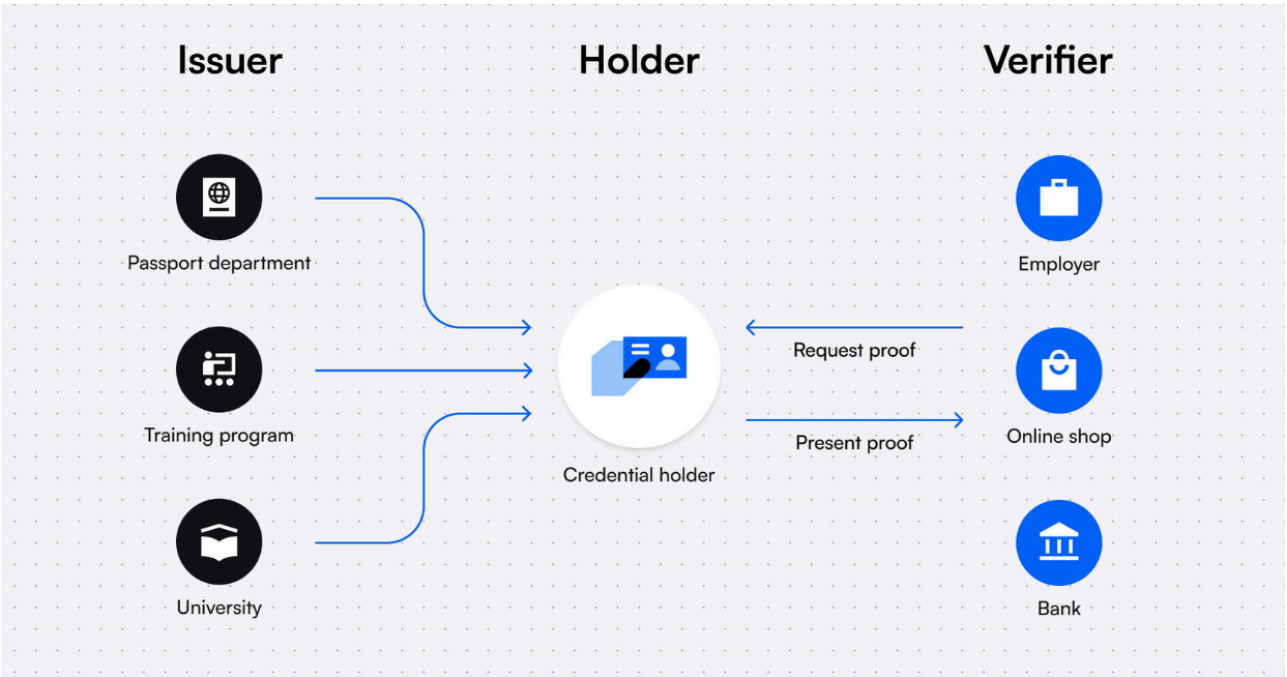
Εικόνα 2: Εφαρμογή Αυτοκυριαρχούμενης Ταυτότητας στα κοινωνικά δίκτυα [20]

Ένας από τους βασικούς πυλώνες για να χτιστεί αυτό το σύστημα αποτελεί η αποκέντρωση, δηλαδή η έννοια πως δεν ίσταται κάποια μεμονωμένη οντότητα που κατέχει τον απόλυτο έλεγχο στα δεδομένα ταυτότητας, μειώνοντας την ευπάθεια στην κατάχρηση [21], κάτι το οποίο μπορεί να επιτευχθεί με τα Αποκεντρωμένα Αναγνωριστικά που θα αναλυθούν στην αμέσως επόμενη υποενότητα. Δευτερευόντως, η έννοια των Επαληθεύσιμων Διαπιστευτηρίων (Verifiable Credentials, VC) ορίζεται ως τις υπογεγραμμένες δηλώσεις σχετικά με τα χαρακτηριστικά ενός ατόμου (π.χ. υπηκοότητα, διαπιστευτήρια) που εκδίδονται από αξιόπιστες οντότητες [22] και αναφύεται ως μια πολύ ελκυστική προσθήκη. Ακροτελεύτια πτυχή αποτελεί η αποθήκευση και τα συν αυτώ, με τα φερόμενα Πορτοφόλια SSI να λειτουργούν ως το βασικό πυλώνα επιτρέποντας στους χρήστες να διαχειρίζονται τα DID, τα VC τους και να τα μοιράζονται επιλεκτικά όπως απαιτείται [23], με τη τεχνολογία

Blockchain να παίζει τον κύριο ρόλο στο σχηματισμό ενός αποκεντρωμένου πλέγματος αποθήκευσης για συναλλαγές αυτών [22, 23]. Παρόλα αυτά, η ανάπτυξη ανθεκτικών συστημάτων SSI που μπορούν να χειριστούν μεγάλους πληθυσμούς χρηστών είναι μια πρόκληση [21], ωστόσο όμως έχει τη δυνατότητα να μεταμορφώσει τον τρόπο με τον οποίο διαχειριζόμαστε τις ψηφιακές μας ταυτότητες, ενισχύοντας το απόρρητο, την ασφάλεια και το προσωπικό δικαίωμα στην αυτοδιάθεση.

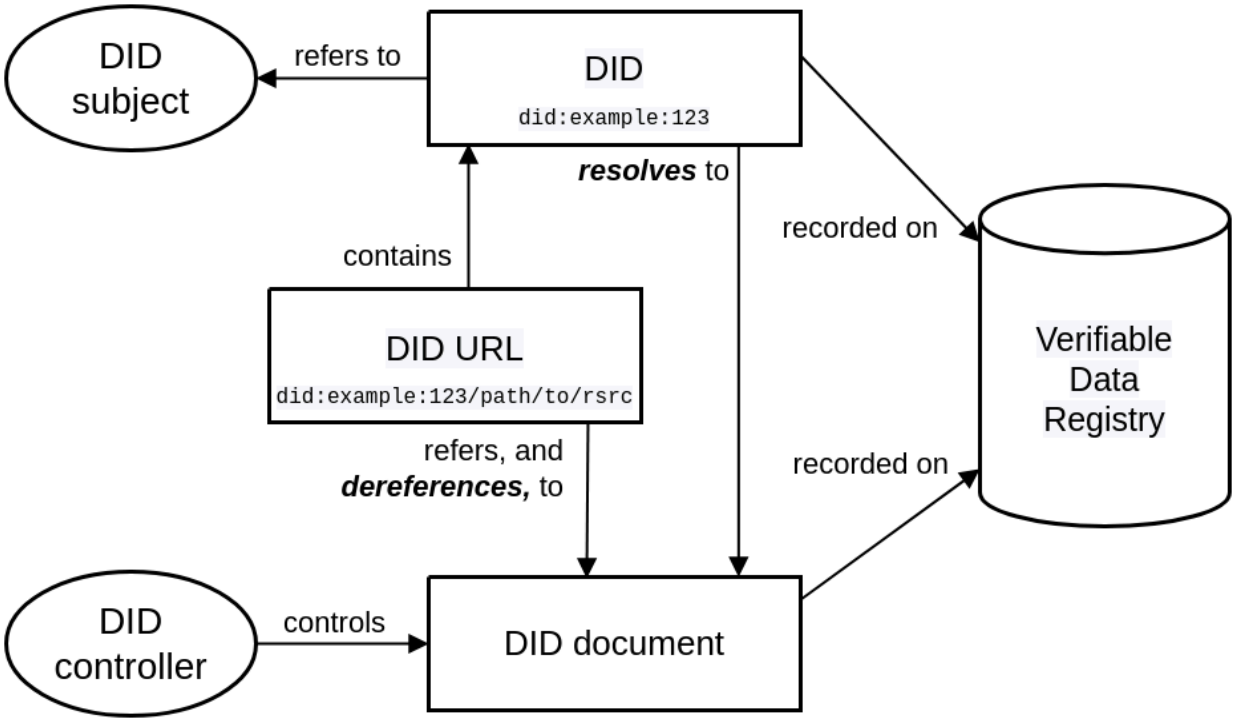
2.2 Αποκεντρωμένα Αναγνωριστικά (DIDs) & Πρότυπα DID του W3C

Τα Decentralized Identifiers (DID) είναι ένας νέος τύπος ψηφιακού αναγνωριστικού που σχεδιάστηκε συναρτήσει της SSI με σκοπό να παρέχει σε άτομα και οργανισμούς μεγαλύτερο έλεγχο των διαδικτυακών ταυτοτήτων τους [24, 25, 26]. Σε αντίθεση με τα παραδοσιακά αναγνωριστικά, τα οποία συνήθως εκδίδονται και διαχειρίζονται κεντρικές αρχές (όπως κυβερνήσεις ή μεγάλες εταιρείες), τα DID έχουν τις ρίζες τους σε αποκεντρωμένες τεχνολογίες όπως το Blockchain[26, 27, 28, 29, 30, 31] και οι κατανεμημένοι πίνακες κατακερματισμού (DHTs) [32]. Ως εκ τούτου, αυτή η τάση αποκέντρωσης αποσκοπεί στην πάγια εξάλειψη μεμονωμένων σημείων αποτυχίας και ευπάθειας, ενώ παράλληλα προσβλέπει στην θέσπιση του χρήστη vw πυρήνα διαχείρισης των δικών του ταυτοτικών δεδομένων



Εικόνα 3: Βασική Φιλοσοφία Αποκεντρωμένων Αναγνωριστικών [33]

Μάλιστα, ο οργανισμός W3C έχει προχωρήσει σε μια πρώιμη προτυποποίηση των αποκεντρωμένων χαρακτηριστικών [34] ούτως ώστε να θέσει τις βάσεις για τη διαλειτουργική υφή που πρέπει να τα χαρακτηρίζει.



Εικόνα 4: Πρότυπη Αρχιτεκτονική Αποκεντρωμένου Μηχανισμού Αναγνωριστικών [35]

```
{
  "@context": [
    "https://www.w3.org/ns/did/v1",
    "https://w3id.org/security/suites/ed25519-2020/v1"
  ]
  "id": "did:example:123456789abcdefghi",
  "verificationMethod": [{
    "id": "did:example:123456789abcdefghi#key-1",
    "type": "Ed25519VerificationKey2020", // external (property value)
    "controller": "did:example:123456789abcdefghi",
    "publicKeyMultibase": "zH3C2AVvLMv6gmMNam3uVAjZpfkcJCwDwnZn6z3wXmqPV"
  }, ...],
  "authentication": [
    // a relative DID URL used to reference a verification method above
    "#key-1"
  ]
}
```

Εικόνα 5: Παράδειγμα Αρχείου Αποκεντρωμένου Αναγνωριστικού [36]

Βασικά εξαρτήματα και λειτουργικότητα

- **Έγγραφα DID**: Τα DID συνδέονται άρρηκτα με έγγραφα DID και περιέχουν πληροφορίες που επιτρέπουν στις οντότητες να ελέγχουν την ταυτότητα τους και να αλληλεπιδρούν με άλλους [g, k], ενώ αρκετά συχνά περιλαμβάνουν κρυπτογραφικά κλειδιά και τελικά σημεία επικοινωνίας.
- **Επαληθεύσιμα διαπιστευτήρια (VC)**: Τα VC είναι ψηφιακά υπογεγραμμένες βεβαιώσεις ή δηλώσεις σχετικά με ετερόκλητα και ετεροχρονισμένα χαρακτηριστικά ενός ατόμου (π.χ. πτυχία, πιστοποιήσεις, ιατρικά αρχεία) που εκδίδονται από αξιόπιστες αρχές [27, 31, 32]. Αυτά τα διαπιστευτήρια συνδέονται με ένα ή πολλά DID, και γενικά πρεσβεύουν ρόλο επίσημου μάρτυρα για τους ισχυρισμούς του κατόχου σχετικά με την ταυτότητά του, ενισχύοντας την αξιοπιστία του.
- **Αυτοκυριαρχία**: Τα DID είναι κεντρικά στο μοντέλο SSI [26, 29]. Αυτό το μοντέλο δίνει σε άτομα ή οργανισμούς πλήρη ιδιοκτησία και έλεγχο της ψηφιακής τους ταυτότητας, επιτρέποντάς τους να προσδιορίζουν πώς και πότε μοιράζονται τα δεδομένα τους [31, 32]. Τουτέστιν, ένας ο εκδότης, πρωταγωνιστής ο κάτοχος, ποικίλες οι πλατφόρμες που το δέχονται.

Οφέλη των DID

- **Ενισχυμένο απόρρητο**: Τα DID προασπίζουν την εκλεκτική αποκάλυψη ελαχιστοποιώντας την περιττή κοινή χρήση πληροφοριών [27, 31]. Έτσι, αντί να παρουσιάζουν ολόκληρες εγγραφές ταυτότητας, με ό,τι συνεπάγεται αυτό, τα DID επιτρέπουν στους χρήστες να παρουσιάζουν μόνο τα συγκεκριμένα διαπιστευτήρια που σχετίζονται με μια δεδομένη συναλλαγή, προστατεύοντας τόσο τους ίδιους όσο και τη πλατφόρμα με την οποία σχετίζονται, και ιδίως από άποψη νομικού δικαίου.
- **Ασφάλεια**: Η αποκεντρωμένη φύση των DID και η χρήση κρυπτογραφίας καθιστούν δυσκολότερο για κακόβουλους παράγοντες να εκμεταλλευτούν ή να διακυβεύσουν δεδομένα ταυτότητας σε μεγάλη κλίμακα [27, 28, 29, 30].
- **Ανθεκτικότητα**: Ερήμην εξαρτήσεως από απόλυτες μεμονωμένες κεντρικές οντότητες, τα συστήματα DID αποφεύγουν μεμονωμένα σημεία αστοχίας, κακόβουλης χρήσεως, επιθέσεων, αλλοτρίωσης και εκποίησης, καθιστώντας τα πιο στιβαρά και ανθεκτικά στη λογοκρισία ή τον έλεγχο [6, 10].
- **Διαλειτουργικότητα**: Βασισμένα σε ανοιχτά πρότυπα, τα DID μπορούν δυνητικά να λειτουργήσουν σε διαφορετικές πλατφόρμες και τομείς, προάγοντας μεγαλύτερη απρόσκοπτη ψηφιακή αλληλεπίδραση [26].

Εφαρμογές DID

- **Υγειονομική περίθαλψη**: Τα DID επιτρέπουν στους ασθενείς να διαχειρίζονται με ασφάλεια τα αρχεία υγείας τους σε όλους τους παρόχους και να παρέχουν επιλεκτική πρόσβαση σε επαγγελματίες όπως απαιτείται [27, 30, 31].

- **Δημόσιες Υπηρεσίες**: Η διαφάνεια και η μη απόρριψη που παρέχεται από τα DIDs μπορεί να βελτιώσει την εμπιστοσύνη και τη λογοδοσία στην παροχή κυβερνητικών υπηρεσιών, και υπηρεσιών δημοσίου τομέα [28].
- **Δικτύωση**: Τα DID μπορούν ενδεχομένως να αντιμετωπίσουν ζητήματα διαχείρισης ταυτότητας μεταξύ τομέων και προστασίας της ιδιωτικής ζωής σε μελλοντικά τηλεπικοινωνιακά δίκτυα [36], όσο και ιδιαιτέρως ανεπτυγμένες τεχνικές λογισμικού όπως αυτή του Data Mesh [37].
- **Απομακρυσμένη εργασία και συνεργασία**: Τα DID βοηθούν στον ασφαλή έλεγχο ταυτότητας σε αποκεντρωμένα περιβάλλοντα εργασίας, όπου οι ταυτότητες μπορεί να επεκταθούν σε διαφορετικές πλατφόρμες και οργανισμούς [24, 25]

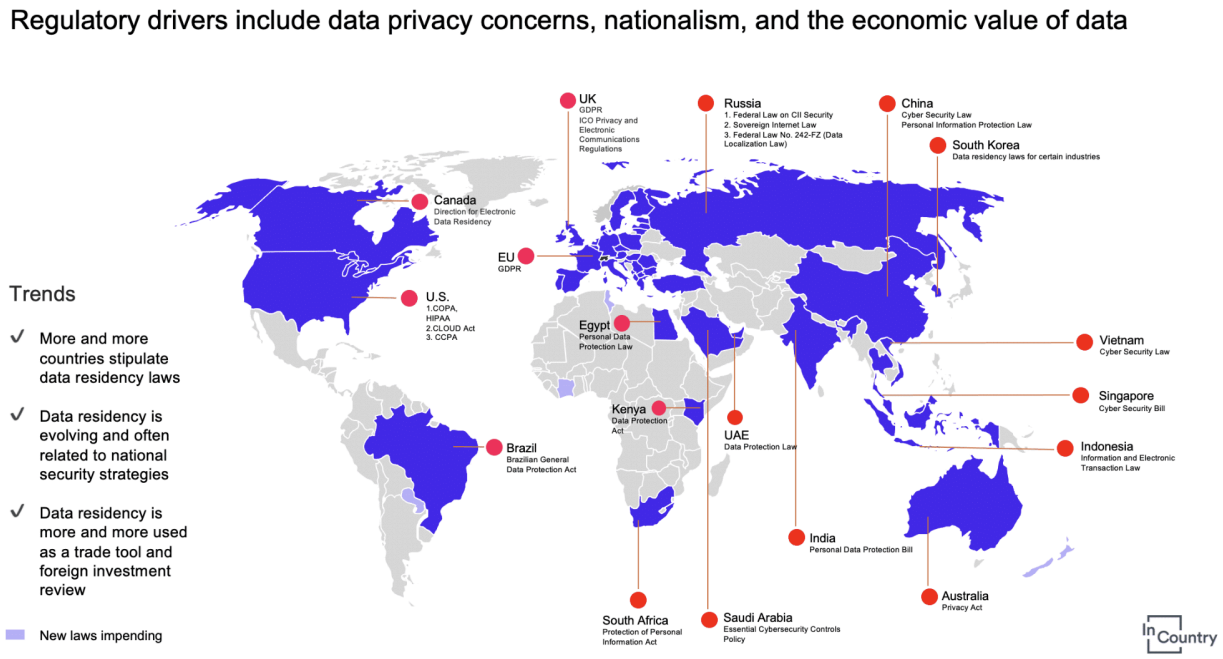
Προκλήσεις που πρέπει να ληφθούν υπόψη

❖ Ενώ τα DID υπόσχονται σημαντικά, η επιτυχής εφαρμογή θα απαιτήσει την αντιμετώπιση ορισμένων προκλήσεων:

- ➢ **Ευχρηστία**: Τα συστήματα DID και SSI πρέπει να είναι φιλικά προς τον χρήστη για να υιοθετηθούν ευρέως, με αυτό να παίζει τον καθοριστικό παράγοντα ολικής υιοθέτησης η απόρριψης.
- ➢ **Απόδοση και επεκτασιμότητα**: Ορισμένοι ερευνητές εκφράζουν ανησυχίες σχετικά με την απόδοση των συστημάτων DID σε καταστάσεις που απαιτούν μεγάλους όγκους συναλλαγών και αλληλεπιδράσεις με τους χρήστες .
- ➢ **Ρυθμιστική Αρμονία**: Τα DID πρέπει να ευθυγραμμίζονται ει δυνατόν με τους εξελισσόμενους κανονισμούς προστασίας δεδομένων και απορρήτου σε όλο τον κόσμο και αυτό είναι επιτακτική ανάγκη όταν η αποκέντρωση επιτάσσει καθολικότητα, άρα και καθολική νομοταγή.

2.3 Πλαίσιο Απορρήτου Δεδομένων & Ασφάλειας

Ένα σταθερό ζήτημα που άπτεται της σχετικής έρευνας αποτελεί η εύρεση μιας ισορροπίας μεταξύ των τεράστιων πιθανών πλεονεκτημάτων των δεδομένων —ιδιαίτερα ευαίσθητων δεδομένων όπως βιομετρικά στοιχεία και αυτά της υγείας— και της επιτακτικής ανάγκης προστασίας του ιδιωτικού απορρήτου και της ασφάλειας των προσωπικών πληροφοριών [38, 39, 40, 41, 42] , θέτοντας την αποστολή για μια εξισορρόπηση ανάμεσα σε δύο λεπτές έννοιες.



Εικόνα 6: Κρατική Κινητοποίηση Νομολογίας Ανά την Υφήλιο [43]

- Βιομετρικά Δεδομένα

Η χρήση βιομετρικών δεδομένων (π.χ. αναγνώριση προσώπου, δακτυλικά αποτυπώματα, μοτίβα βάδισης) ενισχύει τον έλεγχο ταυτότητας και ελαχιστοποιεί την απάτη [38, 39, 44, 45, 46, 47, 48]. Ωστόσο, εκθέτει επίσης πιθανές ευπάθειες, και χωρίς επαρκείς διασφαλίσεις, τα βιομετρικά δεδομένα μπορούν να κάλλιστα κλαπούν, οδηγώντας σε κλοπή ταυτότητας και παραβιάσεις του απορρήτου [38, 44, 45, 47]. Δυστυχώς, οι υπάρχουσες τεχνικές ανωνυμοποίησης αποδεικνύονται συχνά ανεπαρκείς, και αυτό διαφαίνεται εντόνως όταν τα αντίπαλα μοντέλα αποτυγχάνουν να λάβουν υπόψη το πλήρες φάσμα των πιθανών επιθέσεων [38]. Έτσι, ως άμεσο αντίκρισμα αναφύεται η απαίτηση για πιο αυστηρές μεθοδολογίες αξιολόγησης, ισχυρότερες τεχνικές διατήρησης της ιδιωτικής ζωής και ισχυρούς μηχανισμούς συναίνεσης [38, 44, 45, 47] ουτώς ώστε να δημιουργηθεί ένας ασφαλέστερο οικοσύστημα.

- Δεδομένα Υγείας

Τα δεδομένα υγείας, που περιλαμβάνουν ιατρικά αρχεία, αποτελέσματα δοκιμών και άλλες σχετικές πληροφορίες, είναι εξόχως ευαίσθητα [40, 49, 50]. Ως εκ τούτου, τα συστήματα υγειονομικής περίθαλψης που βασίζονται σε δεδομένα μπορεί να προσφέρουν συναρπαστικές ευκαιρίες, ωστόσο όμως τα σχετιζόμενα άτομα και δη οι ασθενείς τακτικά αντιμετωπίζουν έλλειψη ελέγχου σχετικά με τον τρόπο χρήσης των δεδομένων τους – τόσο από παρόχους υγειονομικής περίθαλψης όσο και από τρίτους. Αξίζει να σημειωθεί πως η ώθηση προς προσβάσιμες ψηφιακές υπηρεσίες υγείας - που

επιταχύνθηκε από τη COVID-19 [48] - ενέτεινε περαιτέρω την ανάγκη για ισχυρά πλαίσια διακυβέρνησης δεδομένων και προσεγγίσεις με επίκεντρο τον χρήστη στη διαχείριση δεδομένων υγείας.

- Λύσεις με γνώμονα την τεχνολογία

Οι αλγόριθμοι κρυπτογράφησης και αποκρυπτογράφησης σαφέστατα μπορούν να βελτιώσουν την ασφάλεια του βιομετρικού προτύπου [41] και πάσης φύσεως συναλλαγής δεδομένων. Ένας εξ αυτών είναι ο υπολογισμός πολλαπλών μερών (Multiple Party Computation, MPC) που εξασφαλίζει ένα μοντέλο κατανεμημένου ελέγχου ταυτότητας με εκλεκτική αποκάλυψη πληροφορίας [45]. Ακόμα, το Blockchain και οι όποιες εξελίξεις αυτού και των παρελκομένων του ως παρακλάδια είναι ένα πολλά υποσχόμενο εργαλείο για την παροχή διαφάνειας στην προέλευση των δεδομένων, αποκεντρωμένη διαχείριση συναίνεσης και λογοδοσία δεδομένων [39, 40, 49]. Αν και αυτές οι λύσεις προσφέρουν δυνατότητες, είναι σημαντικό να διασφαλιστεί ότι είναι φιλικές προς το χρήστη και επεκτάσιμες [44, 49].

- Κανονισμοί και Διακυβέρνηση Δεδομένων

Αγγίζοντας στη νομική πτυχή του θέματος, οι τεχνολογικές λύσεις από μόνες τους δεν αποτελούν μονολιθική πανάκεια για τη διαφύλαξη της ιδιωτικής ζωής και του προσωπικού απορρήτου, αφού πλαίσια διακυβέρνησης δεδομένων, οι κανονισμοί και ο σχεδιασμός πολιτικής με επίκεντρο τον χρήστη διαδραματίζουν ζωτικό ρόλο [41, 42, 50, 51, 52]. Αρκετές εργασίες ζητούν ισχυρούς κανονισμούς προστασίας δεδομένων, όπως ο GDPR [50] της ΕΕ και τονίζουν τη σημασία της ατομικής συναίνεσης και της διαφάνειας για την οικοδόμηση ηθικών και αξιόπιστων συστημάτων. Έτσι, οφείλει να αποτελεί λαϊκή απαίτηση για ορισμό διεθνών πλαισίων και συνεργασία για την αντιμετώπιση της χωρίς σύνορα φύσης ανταλλαγής δεδομένων και πιθανής εξωεδαφικής εμβέλειας ορισμένων κανονισμών δεδομένων.

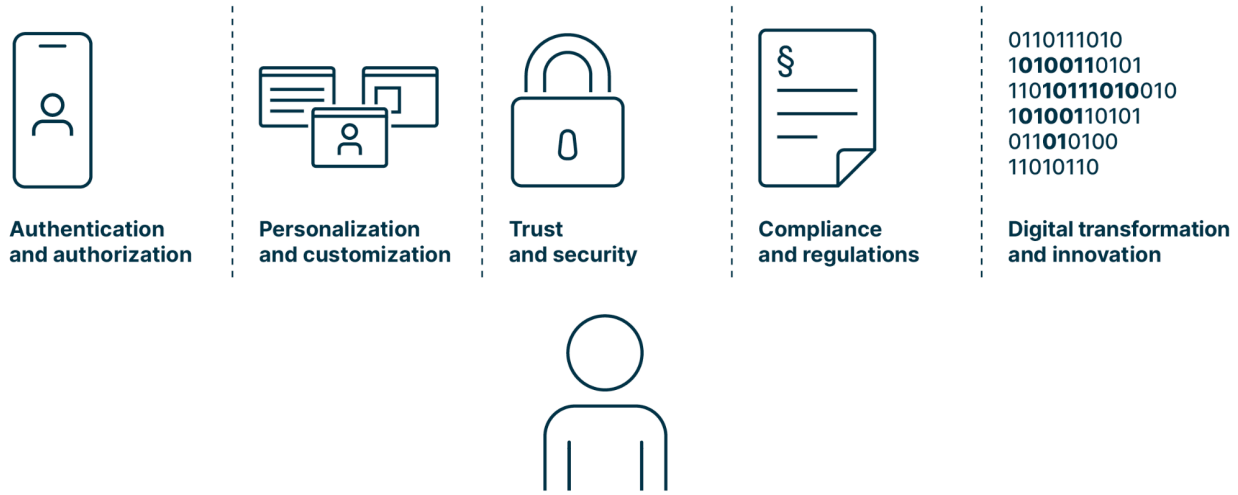
3

Θεωρητικό Πλαίσιο

3.1 Αρχές Αυτο-Κυριαρχούμενης Ταυτότητας, Μοντέλα DID και Πρότυπο W3C

Τα DID αποτελούν έναν σύγχρονο μηχανισμό ταυτοποίησης, αυθεντικοποίησης και κατ' επέκταση αναγνώρισης οντοτήτων ερήμην κάποιας συγκεντρωτικής εξουσίας. Στο συγκεκριμένο πλαίσιο, συγκεντρωτική εξουσία μπορεί να οριστεί ως ένας οργανισμός ή σύνολο αυτών οι οποίοι διαχειρίζονται όλες τις σχετικές παραπάνω λειτουργίες υπό το δικό τους ταυτοτικό πλέγμα, ήτοι όλα τα δεδομένα εισέρχονται και εξέρχονται δια των δικών τους βάσεων δεδομένων και πρωτοκόλλων. Εντούτοις, σε ένα αποκεντρωμένο σύστημα αυτό εκλείπει μεταφέροντας την εξουσία στις ίδιες τις οντότητες οι οποίες είναι και οι άμεσα ενδιαφερόμενες, με τις παρεμβαίνουσες εξουσίες τρίτων να παρακάμπτονται εξ' ολοκλήρου ή σε μεγάλο βαθμό. Κάθε DID, όπως ακριβώς και κάθε τυπικό (ID), αναπαριστά ένα μοναδικό και αναγνωρίσιμο αποτύπωμα ως ταυτότητα το οποίο μπορεί να χρησιμοποιηθεί ως αυθεντικοποίηση, ανταλλαγή πληροφοριών και διαχείριση προσωπικών δεδομένων. Επομένως, ως ειδοποιός διαφορά λογίζεται η πλατφόρμα επί της οποίας λειτουργούν τα DID, η οποία συνήθως διέπεται εκ των πραγμάτων από ανεξαρτησία παρόχων και συγκεντρωτικών βάσεων δεδομένων ούτως ώστε να μην υπάρχει ουδεμία ανάγκη εμπιστοσύνης από ενδιάμεσους. Ως εκ τούτου, τα DID χρησιμοποιούν έναν μητρώο κατανεμημένων αναγνωριστικών (DID Registry) για την αποθήκευση και τη διαχείριση των αναγνωριστικών. Η αποκέντρωση αυτή των ταυτοτήτων μπορεί να παρέχει μεγαλύτερη ασφάλεια και απόρρητο στους κατόχους των DID, ενώ επίσης επιτρέπει τη διαλειτουργικότητα μεταξύ ετερόκλητων συστημάτων και εφαρμογών, δηλαδή μία ταυτότητα για πολλές υπηρεσίες. Αυτό δημιουργεί ένα πιο ανοιχτό και ευέλικτο οικοσύστημα, ελευθερώνοντας τους χρήστες από την κλειστή φύση των κεντροποιημένων συστημάτων τα οποία μπορούν ευκόλως να θέσουν εκτός οποιονδήποτε δεν συνάδει με το θεσπισμένο πρωτόκολλο, κατά το

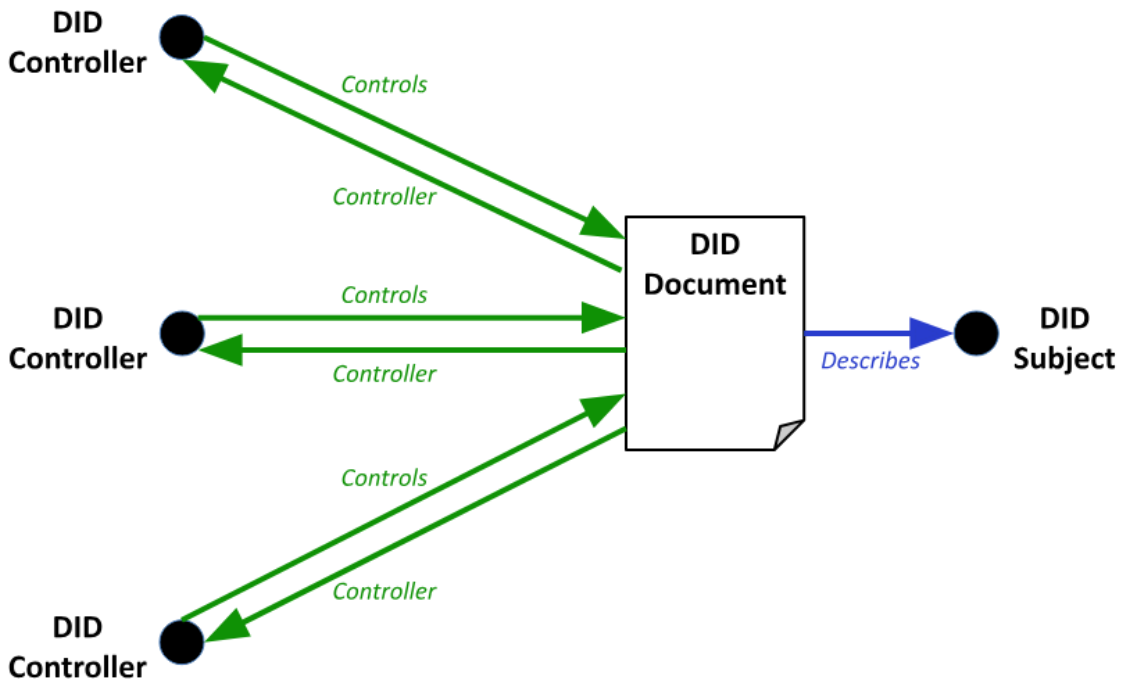
δοκούν. Επιπλέον, η χρήση ενιαίων μητρώων κατανεμημένων αναγνωριστικών (DID Registrie(s)) αντί για κεντρικές βάσεις δεδομένων παρέχει μεγαλύτερη ανθεκτικότητα και ασφάλεια στο σύστημα, καθώς δεν είναι ευάλωτο σε μονόπλευρες αποτυχίες ή κακόβουλες επιθέσεις. Αξίζει να σημειωθεί πως πραγματοποιείται εντοπισμένη προσπάθεια ώστε να θεσπιστεί *de-facto* μηχανισμός με σκοπό τα DID ως τεχνολογία να ενισχύουν τον έλεγχο των χρηστών επί των προσωπικών τους δεδομένων, χωρίς την απαραίτητη ολική αποκάλυψη ευαίσθητων δεδομένων σε τρίτους, ή ακόμα και την εκλεκτική αποκάλυψη μέρους αυτών.



Εικόνα 7: Ανακεφαλαίωση Μοντέλου [53]

Ένα απτό παράδειγμα μεταξύ κεντροποιημένης και αποκεντρωμένης αναγνωριστικής ταυτότητας για έναν πολίτη μπορεί να είναι η ψηφιακή ταυτότητα ατόμου που χρησιμοποιείται για πρόσβαση σε υπηρεσίες διακυβέρνησης. Σε μια κεντροποιημένη αναγνωριστική ταυτότητα, οι πάσης φύσεως εργασίες περί ταυτότητος εξαρτώνται από έναν κεντρικό φορέα - όπως μια κυβέρνηση ή εταιρεία - όπως για παράδειγμα η έκδοση και διαχείριση. Ο φορέας διαχειρίζεται τα προσωπικά δεδομένα του πολίτη και είναι υπεύθυνος για την επαλήθευση της ταυτότητάς του σε ποικίλες υπηρεσίες οι οποίες όμως άπτονται αποκλειστικά στα οικεία του συστήματα και μόνον. Ωστόσο, αυτό ενέχει κίνδυνο κατάχρησης δεδομένων και εξάρτησης του πολίτη από αποφάσεις του κεντρικού φορέα ως πλατφόρμα και παρόχου. Αντίθετα, σε ένα σύστημα με αποκεντρωμένη αναγνωριστική ταυτότητα, ο πολίτης διαθέτει ένα DID που του επιτρέπει να δημιουργήσει και να διαχειριστεί τη δική του ψηφιακή ταυτότητα. Αυτό το DID αποθηκεύεται στο ενιαίο μητρώο κατανεμημένων αναγνωριστικών και ο πολίτης έχει πλήρη έλεγχο επί των προσωπικών του δεδομένων, και αυτό επειδή τα δεδομένα είναι διασκορπισμένα σε αποκεντρωμένες βάσεις αντιγραφής (δηλαδή οι κόμβοι επαναλαμβάνουν ως αποθήκευση αντίγραφο των δεδομένων) ανά διαφόρους παρόχους. Όταν ο πολίτης χρειάζεται να επιβεβαιώσει την ταυτότητά του για πρόσβαση σε μια υπηρεσία, μπορεί να παρέχει το DID **(και**

μόνον) χωρίς την ανάγκη να αποκαλύψει τα προσωπικά του δεδομένα ενόσω η ίδια υπηρεσία επικυρώνει τον χρήστη με μόνη προϋπόθεση η έκδοσή του αναγνωριστικού να είναι έγκυρη. Τότε και μόνον ο εκάστοτε φορέας μπορεί να λάβει ορατότητα σχετικών δεδομένων ζητώντας εξουσιοδότηση από τη σχετιζόμενη βάση κάθε φορά (αφού ο ίδιος πλέον γίνεται από κατέχων των δεδομένων σε απλός κοινωνός αυτών). Αυτό επιτυγχάνεται απολύτως με την επικύρωση αυτού του αναγνωριστικού δια του ενιαίου μητρώου. Τουτέστιν, στα κεντροποιημένα συστήματα έχουμε το μοντέλο "*Για Έναν - Δια μέσου* ***Ενός*** *-* *Προς* ***Οποιονδήποτε"***, όπου τα δεδομένα και η πρόσβαση είναι κεκλεισμένα σε μη ελεγχόμενα στεγανά με πρωταγωνιστή τη πλατφόρμα και το χρήστη να αιτείται, ενώ στα αποκεντρωμένα επικρατεί το ***"****Για Έναν* ***-*** *Δια μέσου* ***Πολλών*** - Προς ***Ορισμένους",*** όπου τα προαναφερθέντα δεν περιορίζονται μονομερώς αλλά αποκαλύπτονται από το χρήστη με τον κάθε φορέα στη θέση του αιτούντος.



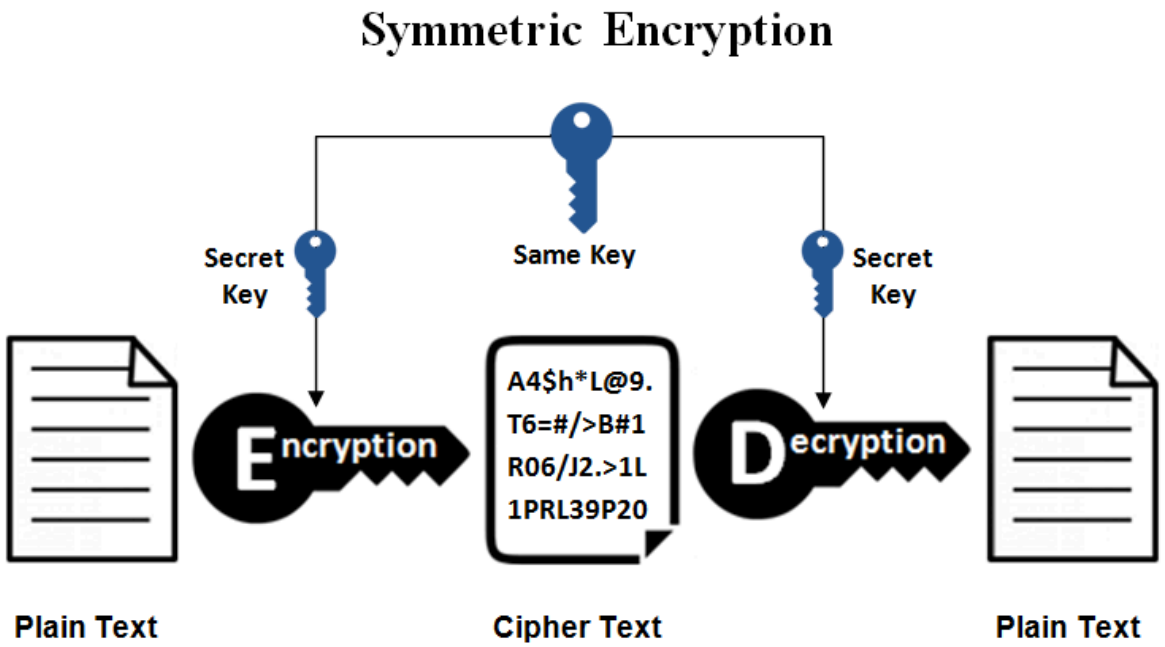
Εικόνα 8: Λογική Ροή Οικοσυστήματος DID [54]

3.2 Βασικές Τεχνικές Κρυπτογράφησης

3.2.1 Συμμετρική Κρυπτογράφηση

Η συμμετρική κρυπτογράφηση [55] είναι τεχνική κρυπτογράφησης κατά την οποία χρησιμοποιείται ένα μοναδιαίο κλειδί τόσο για την κρυπτογράφηση όσο και για την αποκρυπτογράφηση των δεδομένων. Αυτό σημαίνει ότι και ο αποστολέας και ο παραλήπτης χρησιμοποιούν το ίδιο κλειδί για

να ασφαλίσουν και να αποκρυπτογραφήσουν τα δεδομένα. Η συμμετρική κρυπτογράφηση είναι γρήγορη και αποδοτική για μεταφορά μεγάλων όγκων δεδομένων, και είναι κοινώς χρησιμοποιούμενη σε πολλές εφαρμογές κρυπτογράφησης, με ένα από τα γνωστότερα παραδείγματα τον αλγόριθμο Advanced Encryption Standard (AES) [56], ο οποίος χρησιμοποιείται ευρέως. Για παράδειγμα, αν ο αποστολέας θέλει να αποστείλει ένα μήνυμα στον παραλήπτη με χρήση συμμετρικής κρυπτογράφησης, αυτός και ο παραλήπτης συμφωνούν εκ των προτέρων σε ένα κοινό κλειδί. Ο αποστολέας χρησιμοποιεί αυτό το κλειδί για να κρυπτογραφήσει το μήνυμά του πριν την αποστολή με τον παραλήπτη να χρησιμοποιεί το ίδιο κλειδί για να αποκρυπτογραφήσει το μήνυμα και να διαβάσει τα περιεχόμενά του. Φυσικά, ως τεχνική αποτελεί μια απλούστευση της διαδικασίας και η χρήση της άπτεται πολύ συγκεκριμένων περιπτώσεων λόγω της επικινδυνότητας που τη διέπει, με τη



Εικόνα 9: Συμμετρική Κρυπτογράφηση [57]

χρήση της συνήθως να απαντάται συναρτήσει άλλων τεχνικών κρυπτογράφησης.

Έστω ότι ο Bob επιθυμεί να αποστείλει ένα κρυπτογραφημένο μήνυμα στην Alice χρησιμοποιώντας συμμετρική κρυπτογράφηση. Αρχικά, ας ορίσουμε τη σημειογραφία:

1) K_{sym}: Αποτελεί κοινό κλειδί που χρησιμοποιούν αμφότεροι ο Bob όσο και η Alice τόσο για κρυπτογράφηση όσο και αποκρυπτογράφηση του μηνύματος.
2) M: Το αρχικό μήνυμα που θέλει να αποστείλει ο Bob στην Alice.
3) C: Λογίζεται ως το κρυπτογραφημένο μήνυμα που παράγεται από τη κρυπτογράφηση.
4) M': Το αποκρυπτογραφημένο μήνυμα που λαμβάνει η Alice.

Στη συνέχεια, η διαδικασία περιγράφεται ως εξής:

I. Ο Bob χρησιμοποιεί το κοινό κλειδί K_{sym} για να κρυπτογραφήσει το αρχικό μήνυμα M, παράγοντας το κρυπτογραφημένο μήνυμα C. Αυτή η διαδικασία περιγράφεται με τον τύπο:

$C = Encrypt(K_{sym}, M)$

II. Ο Bob στέλνει το κρυπτογραφημένο μήνυμα C στην Alice. Η Alice λαμβάνει το κρυπτογραφημένο μήνυμα C και χρησιμοποιεί το κοινό κλειδί K_{sym} για να αποκρυπτογραφήσει το μήνυμα, παράγοντας το αποκρυπτογραφημένο μήνυμα M′. Αυτή η διαδικασία περιγράφεται με τον τύπο:

$M' = Decrypt(K_{sym}, C)$

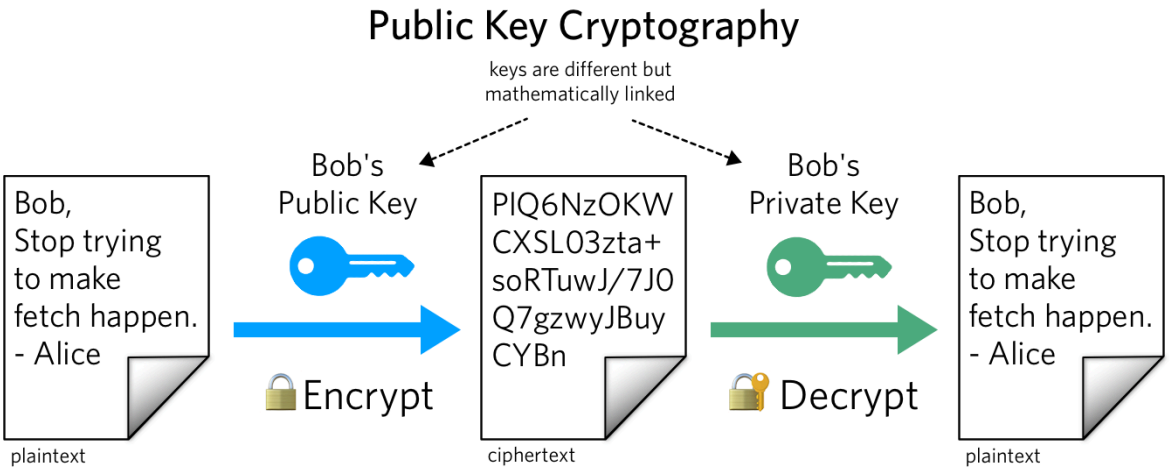
Έτσι, το μήνυμα M′ που λαμβάνει η Alice είναι το αρχικό μήνυμα που απέστειλε ο Bob.

3.2.2 Κρυπτογράφηση Δημοσίου Κλειδιού

Η κρυπτογράφηση δημοσίου κλειδιού [58] - επίσης γνωστή και ως ασύμμετρη κρυπτογράφηση - αποτελεί ένα κρυπτογραφικό σύστημα που στηρίζεται στην ύπαρξη ενός διπόλου κλειδιών, δηλαδή το:

1. **Δημόσιο Κλειδί**, το οποίο είναι είναι ευρέως διαθέσιμο και δύναται να χρησιμοποιηθεί για την κρυπτογράφηση πληροφοριών, και το

2. **Ιδιωτικό Κλειδί** που διατηρείται οπωσδήποτε μυστικό από τον εκάστοτε κάτοχό του και χρησιμεύει στην αποκρυπτογράφηση πληροφοριών που έχουν κρυπτογραφηθεί με το **αντίστοιχο** δημόσιο κλειδί, ήτοι το μοναδικό του ζεύγος. Ειδικότερα, η αυξημένη ασφάλεια που προσδίδει η συγκεκριμένη τεχνική άπτεται της ασύμμετρης φύσεως των κλειδιών έναντι της χρήσης ενός μοναδικού, και αυτό επειδή η αποκρυπτογράφησή τίθεται εντός περιορισμένου εύρους και δεν λειτουργεί αν δεν χορηγηθεί το αντίστοιχο ιδιωτικό. Σε αντίθεση με αυτό, το δημόσιο μπορεί να διαμοιραστεί οποτεδήποτε με τον οποιονδήποτε ούτως ώστε να εξυπηρετηθούν με ασφαλή τρόπο διάφορες εφαρμογές, με κύριες αυτών τις ψηφιακές υπογραφές και τον έλεγχο ταυτότητας. Δύο από τους δημοφιλέστερους αλγορίθμους αποτελούν ο RSA [59] και ο Elliptic Curve Cryptography (ECC) [60].



Εικόνα 10: Κρυπτογράφηση Δημοσίου Κλειδιού [61]

Παρέχοντας ένα σχετικό παράδειγμα, θεωρούμε ότι η Alice θέλει να στείλει ένα μήνυμα προς τον Bob. Ο Bob έχει ένα ζευγάρι κλειδιών, ένα δημόσιο K_{pubBob} και ένα ιδιωτικό $K_{privBob}$. Η Alice χρησιμοποιεί το δημόσιο κλειδί του Bob για να κρυπτογραφήσει το μήνυμά της.

Μήνυμα (Message) : M

Δημόσιο Κλειδί του Bob (Bob's Public Key) : K_{pubBob}

Κρυπτογραφημένο Μήνυμα (Encrypted Message) : $E = Encrypt(M, K_{pubBob})$

Το κρυπτογραφημένο μήνυμα Ε στέλνεται στον Bob. Ο Bob χρησιμοποιεί το ιδιωτικό κλειδί του $K_{privBob}$ για να αποκρυπτογραφήσει το μήνυμα και εν τέλει να το διαβάσει.

Κρυπτογραφημένο Μήνυμα (Encrypted Message) : E

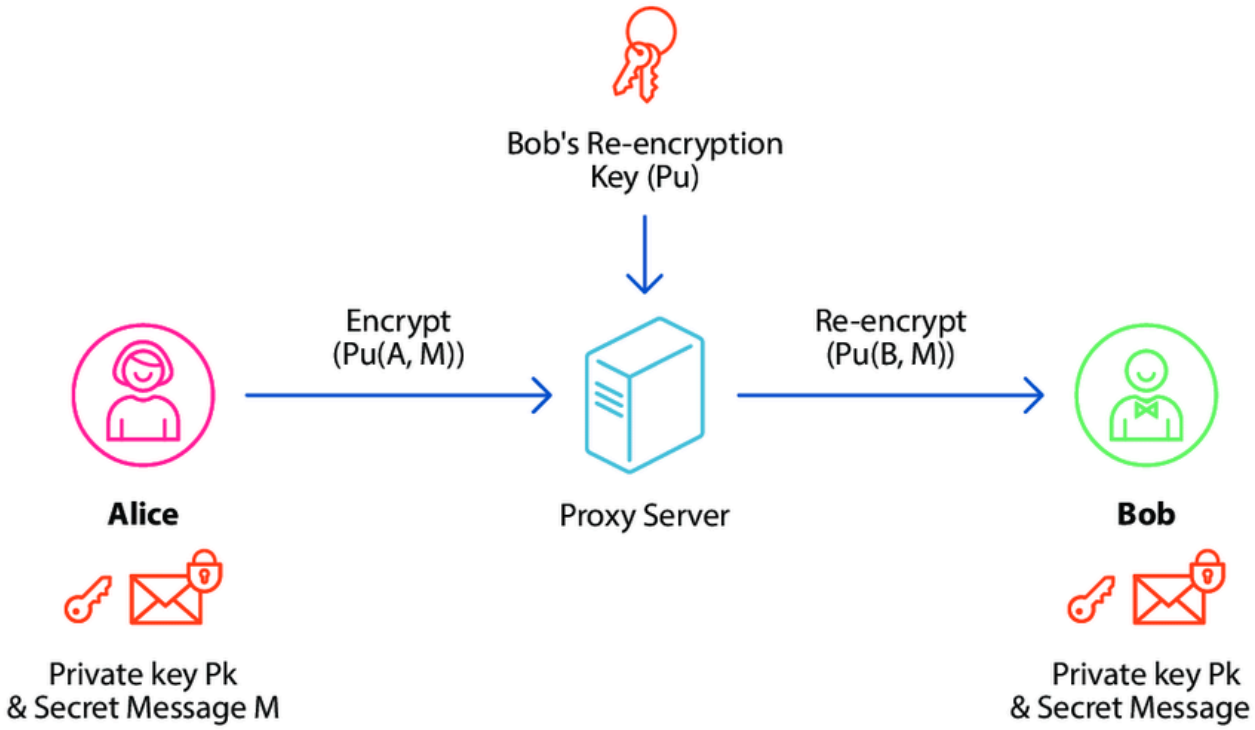
Αποκρυπτογραφημένο Μήνυμα (Decrypted Message) : $D = Decrypt(E, K_{privBob})$

Έτσι, ο Bob μπορεί τώρα να διαβάσει το μήνυμα Μ που έστειλε η Alice.

3.2.3 Επανακρυπτογράφησης Πληρεξουσίου

Η επανακρυπτογράφηση πληρεξουσίου (Proxy Re-Encryption, PRE) [62] αποτελεί τεχνική κρυπτογράφησης που τοποθετεί έναν οργανισμό ή μεσάζοντα (proxy) σε πρωταγωνιστικό ρόλο ούτως ώστε να μετασχηματίσει ένα κρυπτογραφημένο μήνυμα από το κρυπτογράφημα που έχει δημιουργηθεί με το κλειδί μιας οντότητας Α σε κρυπτογράφημα που μπορεί να αποκρυπτογραφηθεί από το κλειδί μιας δεύτερης Β. Γενικότερα, στη τυπική κρυπτογράφηση δημοσίου κλειδιού αυτό δεν

δύναται να συμβεί, αφού το ζεύγος αποκρυπτογράφησης διέπεται από μια συγκεκριμένη ροή όπου όποιος κρυπτογραφεί με το ιδιωτικό του κλειδί επιτρέπει την αποκρυπτογράφηση με το αντίστοιχο δημόσιο, καταρρίπτοντας έτσι την έννοια της ιδιωτικότητας. Εντούτοις, σε περίπτωση που κρυπτογραφήσει εν πρώτοις με το ιδιωτικό αποκόπτει οποιονδήποτε από το να πράξει οτιδήποτε πέρα της επικύρωσης πως αυτό το μήνυμα προήλθε αληθώς από τον υποφαινόμενο κρυπτογράφο. Οπότε, η PRE λύνει αυτό το πρόβλημα επιτρέποντας τη μεταφορά του κρυπτογραφημένου μηνύματος από τον Α στον Β διασφαλίζοντας την αποκρυπτογράφηση με το ιδιωτικό κλειδί του Β και μάλιστα δίχως να αποκαλύπτεται το πραγματικό περιεχόμενο του μηνύματος στον μεσάζοντα. Η PRE έχει εφαρμογές σε πολλούς τομείς, μεταξύ αυτών συστημάτων διαμεσολάβησης δεδομένων, των συστημάτων εξυπηρέτησης ασφαλών αποθετηρίων, μικροελεγκτών και διαδικτύου των πραγμάτων [63].



Εικόνα 11: Επανακρυπτογράφηση Πληρεξουσίου [64]

Η PRE έχει πολλές πρακτικές εφαρμογές και προσφέρει αρκετά οφέλη, μερικά από τα οποία περιλαμβάνουν:

- ***Ελεγχόμενη μετακίνηση δεδομένων***: μπορεί να χρησιμοποιηθεί για την ασφαλή μεταφορά κρυπτογραφημένων δεδομένων από μια οντότητα Α σε μια δεύτερη Β, χωρίς να αποκαλύπτεται το αυτούσιο περιεχόμενο αυτών. Επιπροσθέτως, στη σειρά οντοτήτων μπορούν κάλλιστα να προστίθενται έτι περισσότερες (Γ, Δ,...***ν***) αφού η κύρια οντότητα Α δύναται να εξουσιοδοτήσει ελεγχόμενη πρόσβαση ή και αποκοπή αυτής.

- ***Πολυδιάστατη διακίνηση δεδομένων άνευ αποκάλυψης***: μπορεί να χρησιμοποιηθεί για τη δημιουργία ασφαλών συστημάτων εξυπηρέτησης δεδομένων, όπου ένας ενδιάμεσος μπορεί να επανακρυπτογραφήσει και να μεταφέρει δεδομένα ασφαλώς μεταξύ πολλών συμμετεχόντων και ποικίλων συνδέσεων χωρίς να αποκαλύπτει το πραγματικό περιεχόμενο.
- ***Αποκρυπτογράφηση μόνο από εξουσιοδοτημένους παραλήπτες***: ο πρωτεύων κάτοχος του κρυπτογραφημένου μηνύματος μπορεί να επιτρέψει μόνο σε συγκεκριμένους παραλήπτες να αποκρυπτογραφήσουν το μήνυμα, διασφαλίζοντας εντοπισμένη εξουσιοδότηση.
- ***Προστασία από μεσάζοντες***: Η PRE μπορεί να προστατεύσει τα δεδομένα από τυχόν κακόβουλη δράση από μεσάζοντες, καθώς οι μεσάζοντες δεν έχουν πρόσβαση στα πραγματικά δεδομένα, ήτοι η όποια δυνητική επίθεση κατοχυρώνει την μηδενική πρόσβαση σε απτά, αυθεντικά δεδομένα.

Κατά επεξηγηματικό παράδειγμα, έστω ότι ο Bob θέλει να μεταφέρει ένα κρυπτογραφημένο μήνυμα στην Alice χρησιμοποιώντας επανακρυπτογράφηση πληρεξουσίου. Αυτή τη φορά, οι όροι θεσπίζονται ως εξείς:

Δημόσιο κλειδί του Bob: K_{PubBob}

Ιδιωτικό κλειδί του Bob: $K_{PrivBob}$

Δημόσιο κλειδί της Alice: $K_{PubAlice}$

Ιδιωτικό κλειδί της Alice: $K_{PrivAlice}$

Αρχικό Μήνυμα: M

Κλειδί Μετασχηματισμού: $K_{Transform}$

Προϊόν πρώτης κρυπτογράφησης: C

Προϊόν επανακρυπτογράφησης: C’

Οπότε:

Ο Bob χρησιμοποιεί το δημόσιο κλειδί K_{PubBob} για να κρυπτογραφήσει το αρχικό μήνυμα M, παράγοντας το κρυπτογραφημένο μήνυμα C. Αυτή η διαδικασία περιγράφεται με τον τύπο:

$$C = Encrypt(K_{PubBob}, M)$$

Ο Bob στέλνει το κρυπτογραφημένο μήνυμα C στην αντίστοιχη μονάδα πληρεξουσίου - τον *Proxy* - η οποία χρησιμοποιεί το ιδιωτικό του Bob $K_{PrivBob}$ μαζί με το δημόσιο κλειδί της Alice $K_{AlicePub}$ για να τα μετουσιώσει σε ένα παραγόμενο κλειδί μετασχηματισμού $K_{Transform}$:

$$K_{Transform} = \text{Transform}(K_{PrivBob}, K_{AlicePub})$$

Τέλος, η μονάδα χρησιμοποιεί ξανά αυτό το κλειδί μετασχηματισμού ώστε να κρυπτογραφήσει το αρχικό μήνυμα C:

$$C' = \text{ReEnrypt}(C, K_{Transform})$$

δίνοντας έτσι τη δυνατότητα στην Alice να μπορεί να το αποκρυπτογραφήσει με το δικό της ιδιωτικό και άμεση ορατότητα στο αρχικό μήνυμα M:

$$M = \text{Decrypt}\ (C', K_{PrivAlice})$$

3.4 Επεξεργασία Ροής Συμβάντων Δεδομένων

Στον προγραμματισμό, η επεξεργασία ροής συμβάντων (event stream processing) [65] αναφέρεται σε μια κατηγορία λογισμικού που χειρίζεται τη συνεχή ροή δεδομένων σε πραγματικό χρόνο και αυτά δύνανται να είναι μηνύματα, συμβάντα, μετρήσεις ή οποιαδήποτε άλλη μορφή πληροφορίας που παράγεται από συστήματα, αισθητήρες, ανθρώπινες δραστηριότητες και υπολογιστική ανταλλαγή. Ειδικότερα, η επεξεργασία ροής συμβάντων διαφέρει από την παραδοσιακή επεξεργασία δεδομένων σε διάφορα βασικά σημεία:

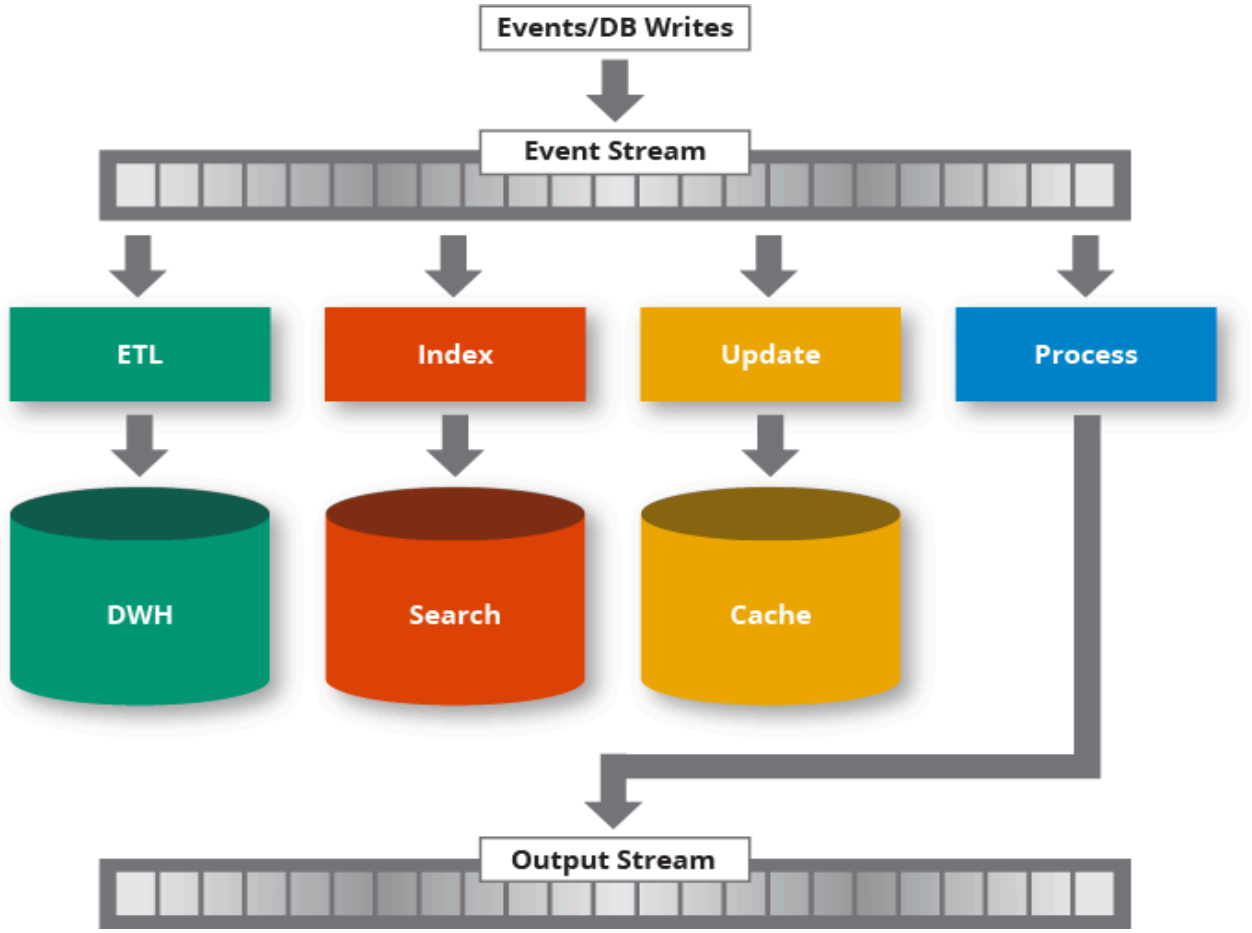
1. **Ροή δεδομένων σε πραγματικό χρόνο** [66]: Τα δεδομένα λαμβάνονται και επεξεργάζονται ακατάπαυστα δίχως να αποθηκεύονται απαραίτητα σε μια βάση δεδομένων προ αναλύσεως.

2. **Υψηλός όγκος δεδομένων** [67]: Η ροή δεδομένων μπορεί να είναι τεράστια, με εκατομμύρια ή δισεκατομμύρια συμβάντα να καταγράφονται κάθε δευτερόλεπτο, και αυτά μπορούν να μεταφράζονται από μερικά *Megabyte* εως αρκετά *Terabyte*.

3. **Χαμηλή λανθάνουσα κατάσταση** [68]: Το γνωστό *latency* που αποτελεί σημαντική παράμετρο ως δείκτη επιτυχίας αποτελεί αντίπαλο σε κάθε επεξεργασία και η ανάλυση των δεδομένων που πρέπει να γίνονται ταχέως, με ελάχιστη καθυστέρηση, για να ανταποκριθούν στις ανάγκες πραγματικού χρόνου.

4. **Ελαστικότητα** [69]: Το σύστημα επεξεργασίας ροής συμβάντων πρέπει να είναι σε θέση να προσαρμόζεται σε αλλαγές στον όγκο ή τη μορφή των δεδομένων, χωρίς να επηρεάζεται η απόδοση.

5. **Διανεμημένη επεξεργασία** [69]: Συχνά, η επεξεργασία ροής συμβάντων υλοποιείται σε ένα κατανεμημένο σύστημα, με ποικιλία διακριτών κόμβων - συνήθως και κατά γεωγραφική διάκριση - να συνεργάζονται για να χειριστούν τον όγκο δεδομένων.



Εικόνα 12: Event Streaming Processing [70]

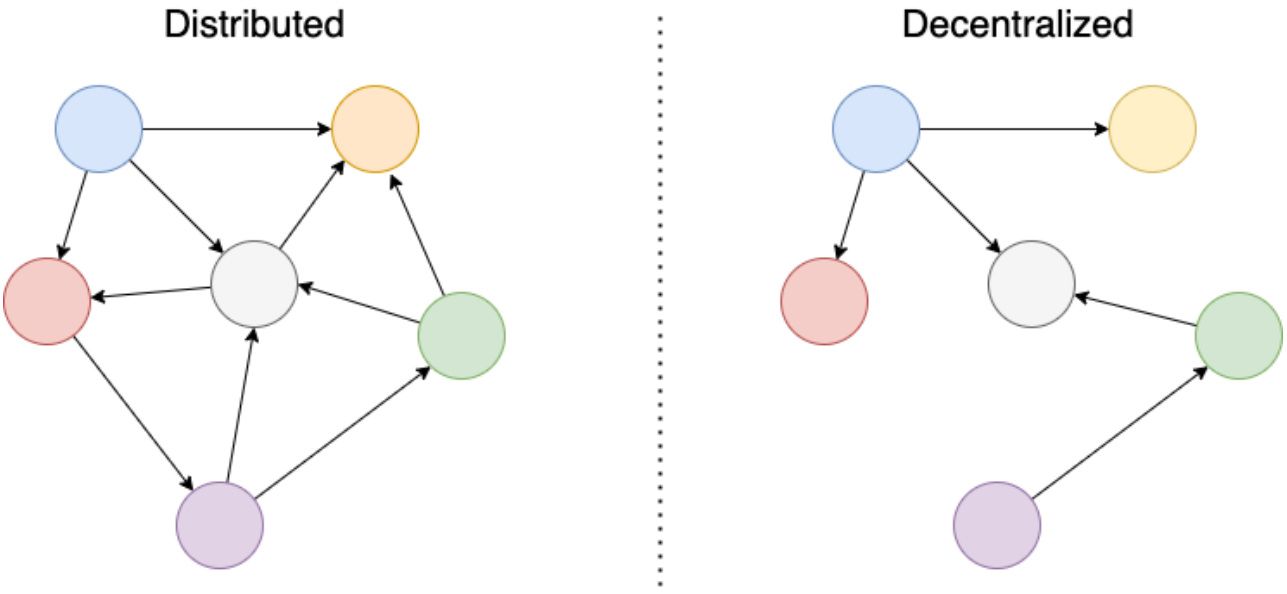
Η επεξεργασία ροής συμβάντων έχει πολλές εφαρμογές σε διάφορους τομείς, όπως:

- Ανάλυση δεδομένων σε πραγματικό χρόνο: Χρησιμοποιείται για εξαγωγή χρήσιμων πληροφοριών από ρεύματα δεδομένων σε πραγματικό χρόνο, όπως ανάλυση αγοράς, ανίχνευση απάτης ή παρακολούθηση συστημάτων.
- Εντοπισμός σφαλμάτων: Βοηθά στην ανίχνευση και την αντιμετώπιση σφαλμάτων σε συστήματα και εφαρμογές, εντοπίζοντας ανωμαλίες στη ροή δεδομένων.

- Προσαρμογή σε πραγματικό χρόνο: Δυνατοποιεί την προσαρμογή συγχρονισμένες χρονικές εναλλαγές όπως η αναπροσαρμογή τιμών ή η αδιάβλητη παροχή εξατομικευμένων προτάσεων στους χρήστες.
- Διαχείριση ροής δεδομένων: Συνιστά τη διαχείριση μεγάλων όγκων δεδομένων σε ζωντανή ροή διασφαλίζοντας ομαλή και αξιόπιστη ροή σε συστήματα όπου οι ετερόκλητοι χρήστες αιτούνται λειτουργιών με ποικιλία διαβαθμίσεως στη δυσκολία εξυπηρέτησης.

3.5 Έννοιες Αποκεντρωμένης & Κατανεμημένης Αποθήκευσης

Σε μια κεντρική βάση δεδομένων, όλα τα δεδομένα βρίσκονται σε μια ενιαία τοποθεσία, συχνά σε έναν μόνο διακομιστή ή δίκτυο που ελέγχεται από μια ενιαία οντότητα, και θα μπορούσε να λογιστεί ως ένα μεγάλο ντουλάπι αρχειοθέτησης σε ένα κλειδωμένο δωμάτιο όπου μόνο όσοι έχουν το κλειδί (π.χ. ο διαχειριστής της βάσης δεδομένων) μπορούν να έχουν πρόσβαση ή να τροποποιήσουν τα περιεχόμενα. Αν και αυτή η ρύθμιση προσφέρει ορισμένα πλεονεκτήματα σε ορισμένα περιβάλλοντα, έχει επίσης μειονεκτήματα:



Εικόνα 13: Αποκεντρωμένες και Κατανεμημένες Βάσεις Δεδομένων [71]

1. **Single Point of Failure** [72]: Μια κεντρική βάση δεδομένων αντιπροσωπεύει ένα μόνο σημείο αποτυχίας. Διαταραχές όπως δυσλειτουργίες υλικού, επιθέσεις στον κυβερνοχώρο ή ακόμα και φυσικές καταστροφές σε αυτήν την τοποθεσία μπορούν να καταστρέψουν ολοσχερώς το σύστημα, καθιστώντας έτσι τα δεδομένα μερικώς ή πλήρως απρόσιτα, και δη μη ανακτήσιμα.
2. **Περιορισμοί επεκτασιμότητας** [73]: Καθώς ο όγκος δεδομένων αυξάνεται, οι κεντρικές βάσεις δεδομένων μπορεί να αντιμετωπίσουν προβλήματα απόδοσης και αυτό ιδίως στις περιπτώσεις

ζωντανής συναλλαγής δεδομένων (Online Transaction Processing, OLTP), θέτοντας ανάχωμα στην ικανότητά τους να χειρίζονται μεγάλους αριθμούς συναλλαγών ή χρηστών. Η αναβάθμιση σε μεγαλύτερο, πιο ισχυρό υλικό μπορεί να είναι μια βραχυπρόθεσμη λύση, αλλά αυτή η προσέγγιση δεν είναι πάντα βιώσιμη ή οικονομικά αποδοτική.

3. **Ευπάθεια στη λογοκρισία και τον έλεγχο** [74]: Η κεντρική αρχή που ελέγχει τη βάση δεδομένων διαθέτει σημαντική ισχύ. Μπορούν να περιορίσουν την πρόσβαση, να χειραγωγήσουν δεδομένα ή ενδεχομένως ακόμη και να υποχρεωθούν από κυβερνήσεις ή άλλα ιδρύματα να παραδώσουν ιδιωτικές πληροφορίες.

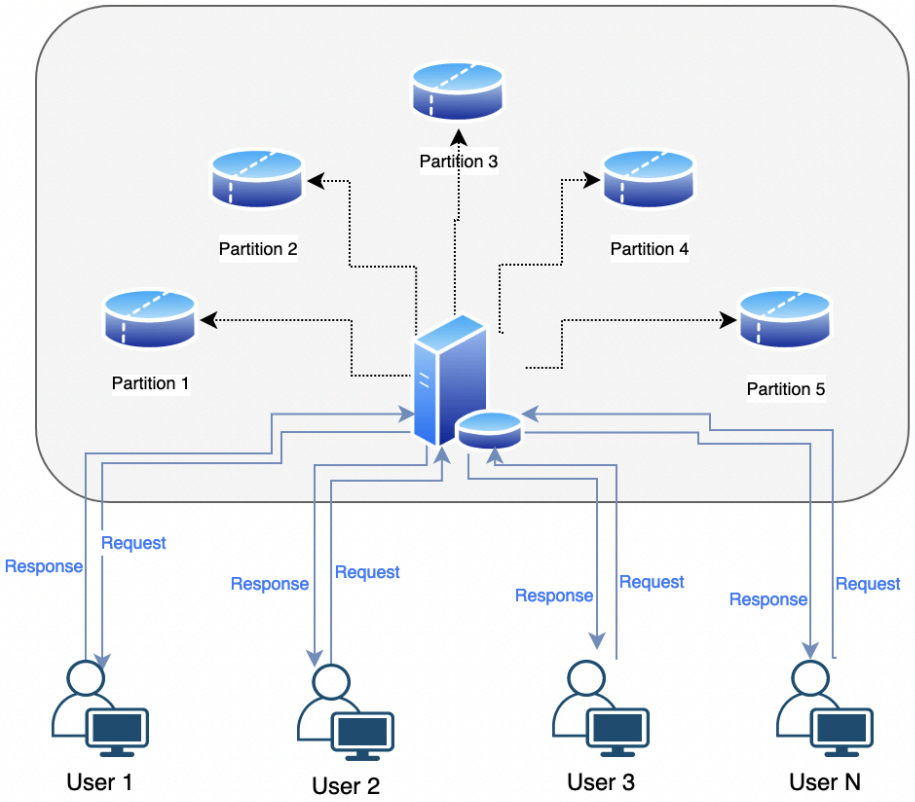
Αναλύοντας δύο έννοιες, η πρώτη αφορά αυτή των **αποκεντρωμένων βάσεων δεδομένων** [75], που στοχεύουν στην αντιμετώπιση αυτών των ελλείψεων. Έτσι, αντί για μια ενιαία τοποθεσία, τα δεδομένα διανέμονται ενιαίως σε ένα δίκτυο κόμβων (υπολογιστές ή διακομιστές). Βασικά στοιχεία αυτής της προσέγγισης περιλαμβάνουν:

- *Χωρίς Κεντρική Αρχή*: Δεν υπάρχει ενιαία οντότητα με απόλυτο έλεγχο των δεδομένων, μειώνοντας έτσι τον κίνδυνο λογοκρισίας, χειραγώγησης ή στοχευμένων επιθέσεων.
- *Πλεονασμός και ανθεκτικότητα*: Με την αναπαραγωγή δεδομένων σε πολλούς κόμβους του δικτύου, το σύστημα γίνεται ανεκτικό σε σφάλματα. Εάν ένας ή ακόμα και πολλοί κόμβοι αποτύχουν, το υπόλοιπο δίκτυο έχει εξασφαλισμένη την ακατάσχετη λειτουργική του πορεία.
- *Δυνατότητα για μεγαλύτερη επεκτασιμότητα*: Επειδή οι πόροι υπολογιστών και αποθήκευσης κατανέμονται σε ένα δίκτυο από υπολογιστές που κοινωνούν σε έναν κοινό σκοπό, οι αποκεντρωμένες βάσεις δεδομένων μπορούν θεωρητικά να κλιμακωθούν πιο εύκολα για να φιλοξενήσουν αυξανόμενους όγκους δεδομένων και βάσεις χρηστών, αφού οι υπολογιστικοί πόροι δεν απαντώνται σε ένα μοναδιαίο σύστημα αλλά σε σύμπλεγμα υπολογιστών με διακριτούς πόρους, ήτοι "άπειρη" επεκτασιμότητα.

Σε παρόμοιο φάσμα, μια **κατανεμημένη βάση δεδομένων** [76, 77, 78] είναι ένα σύστημα βάσης δεδομένων όπου τα δεδομένα κατατέμνονται και έπειτα διατίθενται σε πολλές φυσικές τοποθεσίες. Αυτές οι τοποθεσίες μπορεί να είναι διακομιστές εντός του ίδιου κέντρου δεδομένων, γεωγραφικά διασκορπισμένες τοποθεσίες ή ακόμη και ένας συνδυασμός επιτόπιου υλικού και υποδομής που βασίζεται σε σύννεφο. Φυσικά, εν αντιθέσει με τις κεντροποιημένες, δεν υπάρχει ένα μοναδικό «κύριο» αντίγραφο των δεδομένων, και ο πρωταρχικός στόχος μιας κατανεμημένης βάσης δεδομένων είναι να δομηθεί μια ενοποιημένη προβολή αυτών, επιτρέποντας στους χρήστες να αλληλεπιδρούν μαζί τους σαν να ήταν όλα αποθηκευμένα σε ένα μέρος. Τουτέστιν, η φιλοσοφία που υπηρετεί η εξής αρχιτεκτονική μπορεί να οριστεί ως: "*Ο χρήστης δε πρέπει να αντιλαμβάνεται πως τα δεδομένα που προβάλλει δεν αποτελούν ενιαίο κομμάτι, αλλά αποσπάσματα που συνενώνονται με υπολογιστικές διαδικασίες για να αναδυθούν συγκεντρωτικά ως τελικό προϊόν"*.

Κατά αυτό το τρόπο, οι κατανεμημένες βάσεις δεδομένων εισάγουν πολυπλοκότητες που δεν υπάρχουν σε κεντρικά συστήματα, με βασικές προκλήσεις να περιλαμβάνουν:

- *Συνέπεια δεδομένων*: Η διασφάλιση ότι όλοι οι κόμβοι στο κατανεμημένο σύστημα έχουν την πιο ενημερωμένη και ακριβή έκδοση των δεδομένων.
- *Έλεγχος συγχρονισμού*: Η αποτροπή αντικρουόμενων ενημερώσεων και η διασφάλιση της ακεραιότητας των δεδομένων όταν πολλοί χρήστες προσπαθούν να αποκτήσουν πρόσβαση ή να τροποποιήσουν τα ίδια δεδομένα ταυτόχρονα απαιτεί συντονισμό.
- *Κατανομή και αναπαραγωγή δεδομένων*: Η απόφαση για τον διαχωρισμό και την αναπαραγωγή δεδομένων σε κόμβους είναι ένα περίπλοκο πρόβλημα βελτιστοποίησης. Παράγοντες όπως ο όγκος δεδομένων, τα μοτίβα ανάγνωσης/εγγραφής και η γεωγραφική κατανομή των χρηστών πρέπει να λαμβάνονται προσεκτικά υπόψη.



Εικόνα 14: Μοντέλο Κατανεμημένης Βάσεως Δεδομένων [79]

Η θεωρία των κατανεμημένων βάσεων δεδομένων εδράζεται στο πεδίο των κατανεμημένων υπολογιστών, υιοθετώντας βασικές έννοιες όπως ο διαμερισμός, η αναπαραγωγή και η διαφάνεια. Ο διαμερισμός ως βασική πιλοτική αρχή αναφέρεται στην τεχνική διάσπασης ενός μεγάλου συνόλου

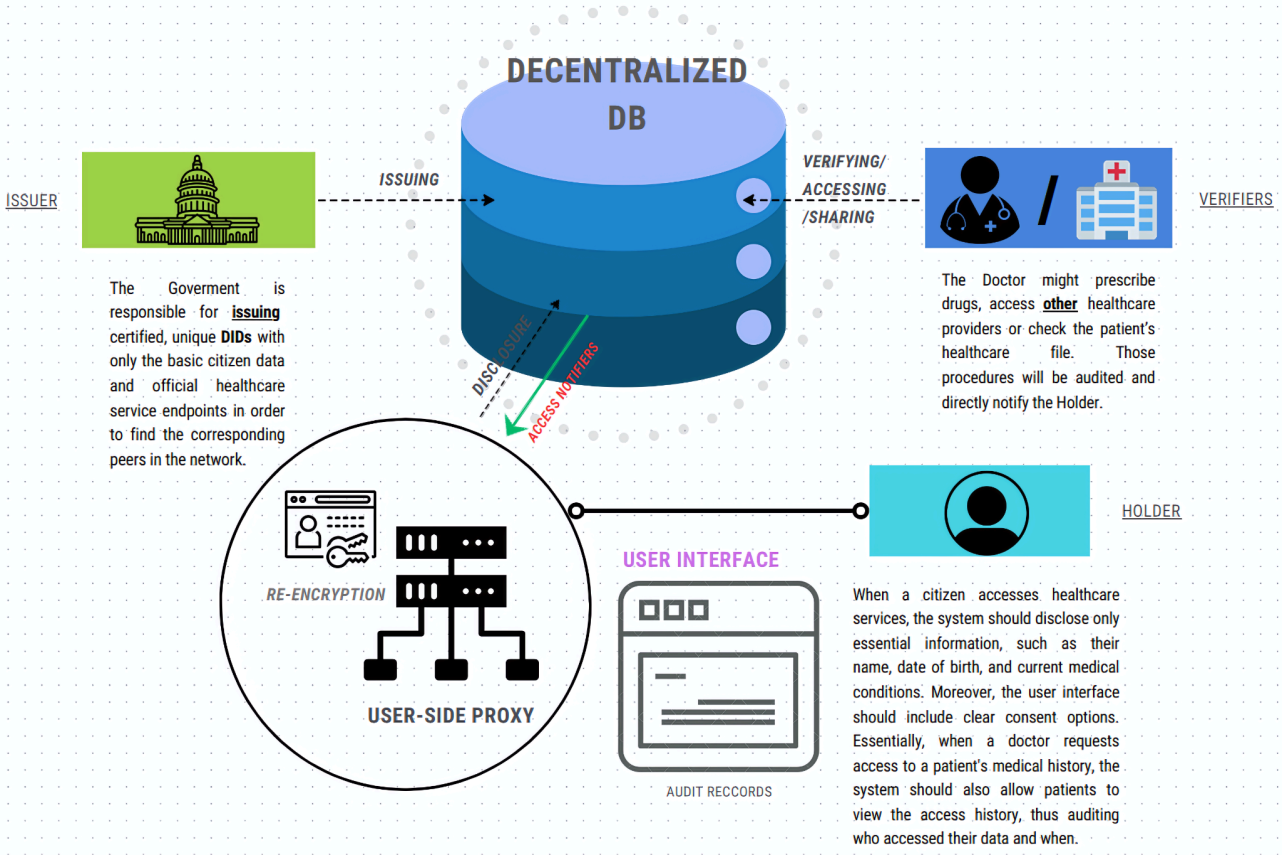
δεδομένων σε μικρότερα, διαχειρίσιμα κομμάτια. Αυτά τα κομμάτια κατανέμονται σε διαφορετικούς κόμβους ενός δικτύου, επιτρέποντας παράλληλη επεξεργασία και βελτιώνοντας την απόδοση. Έπειτα, αναπαραγωγή διασφαλίζει την ακεραιότητα και διαθεσιμότητα των δεδομένων. Δημιουργώντας πολλαπλά αντίγραφα δεδομένων και αποθηκεύοντάς τα σε διαφορετικούς κόμβους - τα οποία όμως λειτουργούν ασυγχύτως και αδιαιρέτως - η βάση δεδομένων γίνεται ανθεκτική σε αστοχίες κόμβων. Σε περίπτωση απώλειας ενός κόμβου, τα δεδομένα παραμένουν διαθέσιμα από άλλους κόμβους. Τέλος, η διαφάνεια αποτελεί τη τρίτη θεμελιώδη αρχή καθώς από την οπτική γωνία του χρήστη, καθώς η βάση οφείλει να λειτουργεί ως ένα ενιαίο, ολοκληρωμένο σύστημα, κρύβοντας την πολυπλοκότητα της κατανεμημένης αρχιτεκτονικής. Οι χρήστες αλληλεπιδρούν με τη βάση δεδομένων με τον ίδιο τρόπο, ανεξάρτητα από τον τόπο αποθήκευσης ή την κατανομή δεδομένων.

Ενώ τόσο η αποκέντρωση όσο και η διανομή μοιράζονται την έννοια της διάδοσης δεδομένων, η βασική διαφορά έγκειται στον έλεγχο. Σε μια κατανεμημένη βάση δεδομένων, ο οργανισμός που κατέχει την υποδομή εξακολουθεί να έχει έναν βαθμό κεντρικού ελέγχου σχετικά με τον τρόπο διαμόρφωσης και διαχείρισης του συστήματος. Μια αποκεντρωμένη βάση δεδομένων, από την άλλη πλευρά, συνήθως περιλαμβάνει περισσότερους συμμετέχοντες με λιγότερη κεντρική εξουσία. Η διακυβέρνηση και η λήψη αποφάσεων εξαπλώνονται μεταξύ των κόμβων του δικτύου. Η θεωρία και η εφαρμογή των κατανεμημένων βάσεων δεδομένων συνεχίζουν να εξελίσσονται. Η ανάπτυξη του cloud computing παρέχει νέους τρόπους για τη δημιουργία επεκτάσιμων και ανθεκτικών κατανεμημένων συστημάτων. Οι βάσεις δεδομένων NewSQL στοχεύουν να παρέχουν πλεονεκτήματα επεκτασιμότητας και διανομής, διατηρώντας παράλληλα τις σχεσιακές ιδιότητες και τις εγγυήσεις *Atomicity*, *Consistency*, *Isolation*, και *Durability* (ACID) των παραδοσιακών συστημάτων βάσεων δεδομένων.

4

Δόμηση Συστήματος

4.1 Επισκόπηση Εθνικού Συστήματος DID



Εικόνα 15: Εθνικής εμβέλειας σύστημα DID

Το εθνικό σύστημα DID, όπως απεικονίζεται στην εικόνα, φιλοδοξεί να διαμορφώσει το μέλλον μια ψηφιακής ταυτότητας, υιοθετώντας τις αρχές της αποκέντρωσης, της αυτονομίας, της ασφάλειας, της

διαφάνειας και της διαλειτουργικότητας, με στόχο τη δημιουργία ενός ασφαλούς και αξιόπιστου πλαισίου για την έκδοση, διαχείριση και χρήση ενός πιο εκλεπτυσμένου, ανεπτυγμένου ταυτοτικού μοντέλου, που βασίζεται σε πέντε θεμελιώδεις αρχές:

- Αποκέντρωση: Ο έλεγχος των DID παραμένει στα χέρια των πολιτών, μειώνοντας κατά το δυνατότερο την εξάρτηση από κεντρικές αρχές.
- Αυτονομία: Οι πολίτες έχουν αυτονομία στην εκλογή δεδομένων που μοιράζονται και με ποιους. Παράλληλα, εγκαθιδρύονται τα δικαιώματα στη συγκατάθεση και δη στη παρακολούθηση ως ακρογωνιαίος λίθος αναφαιρέτων ψηφιακών δικαιωμάτων της σύγχρονης εποχής.
- Ασφάλεια: Τα DID και τα δεδομένα προστατεύονται εισάγοντας την έννοια της επανακρυπτογράφησης.

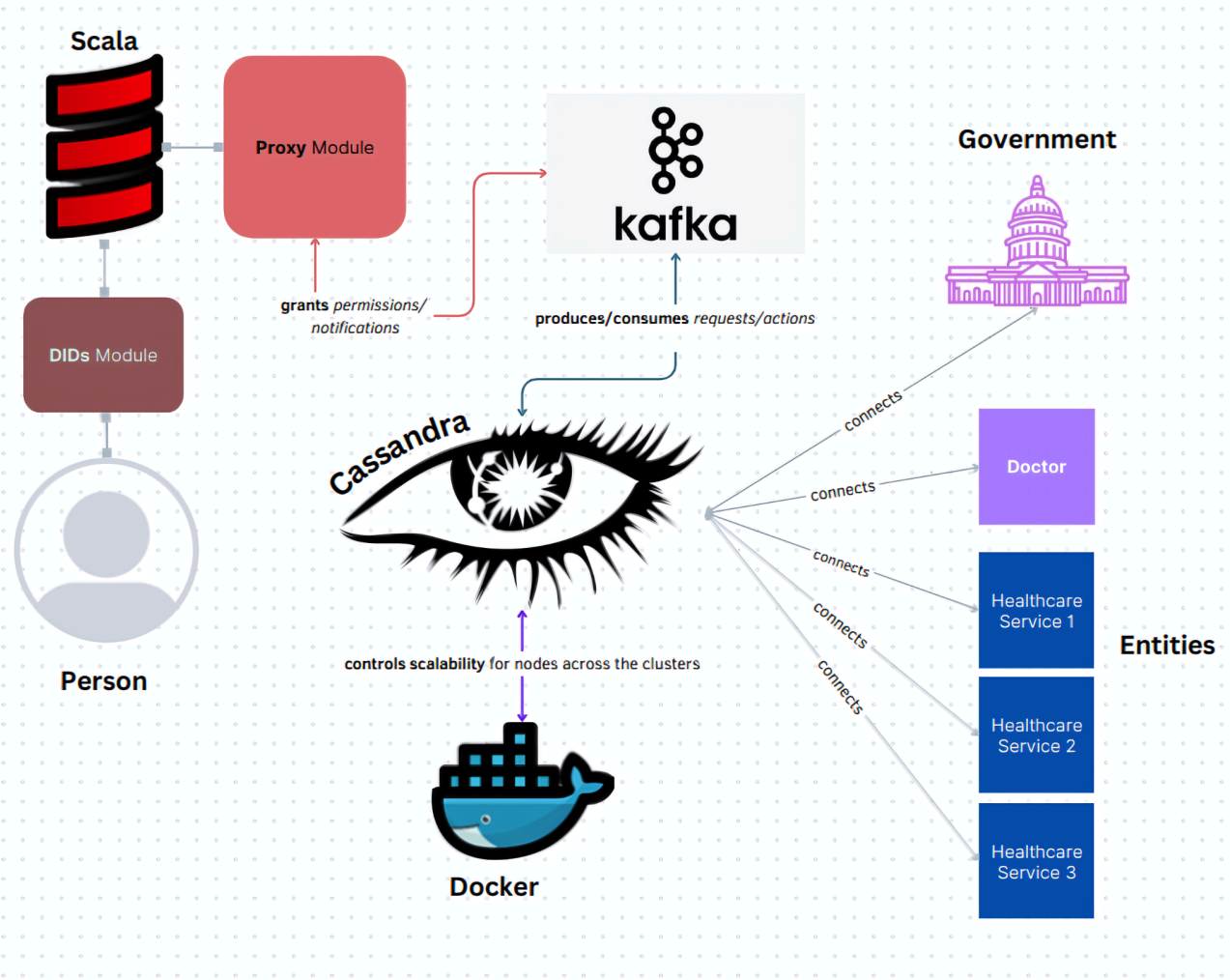
Η υλοποίηση του συστήματος DID βασίζεται σε συνεργασία μεταξύ κυβέρνησης, εγκεκριμένων παρόχων DID και πολιτών. Η πρώτη διαδραματίζει αποκλειστικό ρόλο στην έκδοση των αναγνωριστικών και μόνον, και αυτό ούτως ώστε να παρέχεται αυτόματη επικύρωση ως ύψιστη αρχή εκδόσεως. Δευτερευόντως, οι άμεσα εμπλεκόμενοι ως υπηρεσίες/ιατροί συμμετέχουν ως οι λεγόμενοι "τρίτοι", και η συμμετοχή τους αυτή έγκειται στην αντιστοίχιση των δεδομένων που ζητούν με την ακριβή τοποθεσία αυτών στο κατανεμημένο σύστημα βάσεων. Τέλος οι πολίτες ελέγχουν πλήρως τα δεδομένα τους και με ποιους τα μοιράζονται. Η διαδικασία λειτουργίας του συστήματος DID είναι απλή και ασφαλής. Ο χρήστης υποβάλλει αίτηση για DID, και διεξάγεται έλεγχος ταυτότητας για να διασφαλίσει την ακρίβεια των πληροφοριών. Ακολούθως, μετά την έκδοση παρέχεται ζευγάρι δημοσίου και ιδιωτικού κλειδιού. Το DID και η αντιστοίχιση του δημοσίου κλειδιού αποθηκεύονται σε αποκεντρωμένο αποθετήριο, οπότε ο κάθε χρήστης είναι έτοιμος να χρησιμοποιει το επικυρωμένο του αποκεντρωμένο αναγνωριστικό ώστε να το ταυτίζει με συγκεκριμένα αποκεντρωμένα DID έγγραφα που "δείχνουν" προς διακριτά του ιατρικά, κρατικά, ασφαλιστικά και πάσης φύσεως δεδομένα εντός του δακτυλίου των βάσεων. Οπότε, το ένα μοναδιαίο αναγνωριστικό αποτελεί δίαυλο του πολίτη για πληθώρα υπηρεσιών οι οποίες συμμετέχουν από κοινού στην κατανεμημένη βάση, χωρίς όμως αυτές να δύνανται προβολής, καταγραφής και επεξεργασίας δεδομένων άνευ συγκαταθέσεως.

4.1.1 Απαιτήσεις Σχεδιασμού

Ένα ποικιλόμορφο σύστημα DID τέτοιου βεληνεκούς οφείλει οπωσδήποτε να λάβει υπόψη πτυχές που συνδέονται με την προσβασιμότητα, τη διαχείριση ασφαλών προσβάσεων και την προστασία του ιδιωτικού απορρήτου. Αρχικά, η πολυγλωσσική υποστήριξη είναι απαραίτητη για τη διευκόλυνση των χρηστών με διαφορετικό γλωσσικό υπόβαθρο συνάμα της συμβατότητας με διάφορες συσκευές, όπως *smartphones*, *tablets* και υπολογιστές. Στη συνέχεια, η διαχείριση ασφαλών προσβάσεων απαιτεί τη χρήση πολλαπλών παραγόντων επαλήθευσης ταυτότητας για την αύξηση της ασφάλειας και τη

διαχείριση κινδύνου για την ανίχνευση και πρόληψη κακόβουλων ενεργειών, και αυτό απαιτεί τόσο τον ψηφιακό αλφαβητισμό όσο και τη προσωπική εγρήγορση των πολιτών. Επιπλέον, η ανάκτηση λογαριασμού πρέπει να παρέχει ασφαλείς διαδικασίες σε περίπτωση απώλειας ή κλοπής, ενώ η προστασία του ιδιωτικού απορρήτου περιλαμβάνει την ελάχιστη αποκάλυψη δεδομένων και τη δυνατότητα χρήσης ψευδωνύμων για την προστασία της ανωνυμίας. Εν κατακλείδι, το ακροτελεύτιο σκέλος απαντάται στη διακυβερνητική συνεργασία η οποία απαιτεί συνεργασία μεταξύ κυβερνητικών φορέων, παρόχων DID και άλλων ενδιαφερόμενων μερών (όπως των συμμετεχόντων υπηρεσιών), κοινές οδηγίες και πρότυπα για την ομαλή λειτουργία του συστήματος, καθώς και διαλειτουργικότητα μεταξύ κυβερνητικών υπηρεσιών μέσω της χρήσης DID για πρόσβαση σε διάφορες υπηρεσίες με ένα μόνο DID.

4.3 Δομικά Συστατικά



Εικόνα 16: Αλληλεπίδραση εργαλείων

Η παραπάνω εικόνα παρουσιάζει ένα διάγραμμα ροής του ταυτοτικού μοντέλου και εμπεριέχει ένα συνονθύλευμα τεχνικών εργαλείων, ενώ παράλληλα καταδεικνύει το πως κάθε ένα από αυτά διαδραματίζει ενεργό ρόλο εν σχέσει με τα υπόλοιπα.

- **Κυβέρνηση**:
    - Η κυβέρνηση διαδραματίζει τον ρόλο της εκδότριας οντότητας DIDs. Είναι υπεύθυνη για:
    - Την υλοποίηση μίας πλατφόρμας έκδοσης DIDs.
    - Την τήρηση αρχείου DIDs και των σχετικών μεταδεδομένων.
    - Την εφαρμογή κανόνων και διαδικασιών για την έκδοση DIDs.
    - Την διασφάλιση της ασφάλειας και της ακεραιότητας του συστήματος DIDs.
- **Μονάδα Επανακρυπτογράφησης:**
    - Η μονάδα επανακρυπτογράφησης λειτουργεί ως μεσάζων των πολιτών και των εκάστοτε ενδιαφερομένων - ιατροί και πάροχοι υπηρεσιών - υπολοίπων παραγόντων. Διαχειρίζεται τις επικοινωνίες, εξασφαλίζοντας την ομαλή ροή πληροφοριών και την προστασία των δεδομένων, και μπορεί κάλλιστα να υλοποιηθεί με διάφορες μορφές, όπως λογισμικό, υπηρεσίες cloud ή φυσικές υποδομές.
- **Μονάδα DID:** διαχειρίζεται τα DIDs (και τα σχετικά DID έγγραφα του καθενός) και σχετικές αυτών λειτουργίες, και είναι υπεύθυνη για:
    - Την αποθήκευση και την ανάκτηση DIDs.
    - Την επαλήθευση της αυθεντικότητας των DIDs.
    - Την διαχείριση κρυπτογραφικών κλειδιών και πιστοποιητικών.
    - Την υποστήριξη λειτουργιών όπως η ανάκληση και η μεταφορά DIDs.
- **Πολίτης (ασθενής)**:
    - Ο άνθρωπος αντιπροσωπεύει τον κάτοχο του DID.
    - Μπορεί να αλληλεπιδρά με το σύστημα DIDs για:
        - Να λάβει DID.
        - Να διαχειριστεί το DID του (π.χ. να αλλάξει κρυπτογραφικά κλειδιά).
        - Να χορηγήσει εξουσιοδοτήσεις σε τρίτους.
        - Να ανακτήσει πληροφορίες σχετικά με το DID του.
- **Γιατρός/Πάροχος**:
    - Ο γιατρός αντιπροσωπεύει τον τρίτο που λαμβάνει εξουσιοδότηση από τον κάτοχο του DID.
    - Μπορεί να αλληλεπιδρά με το σύστημα DIDs για:
        - Να λάβει εξουσιοδοτημένο DID από τον κάτοχο.
        - Να επαληθεύσει την αυθεντικότητα του εξουσιοδοτημένου DID.

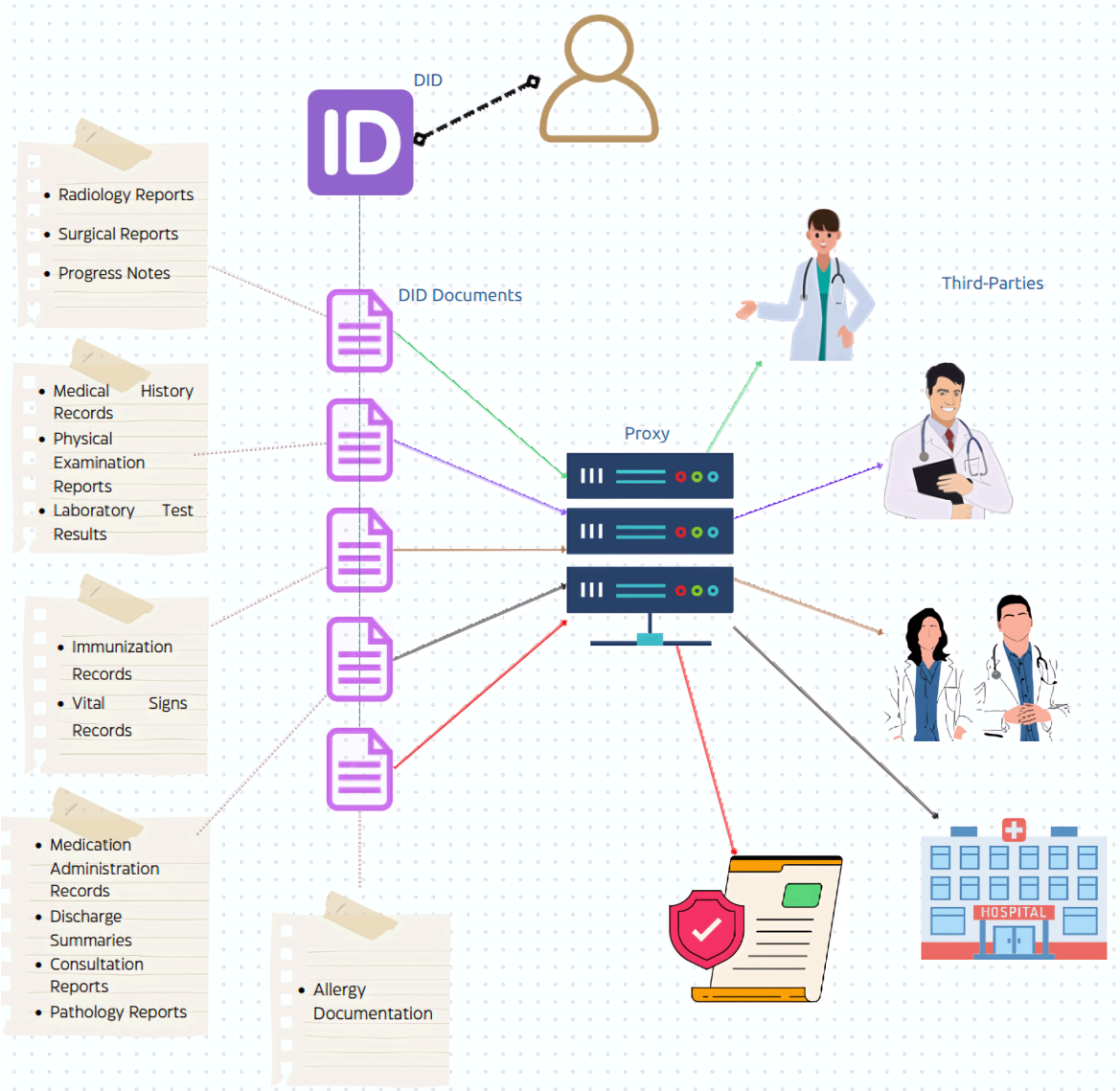
        - Να αποκτήσει πρόσβαση σε δεδομένα ή υπηρεσίες μέσω του εξουσιοδοτημένου DID.

- **Cassandra**:
    - Η Cassandra αποτελεί μια βάση δεδομένων NoSQL που χρησιμοποιείται για την αποθήκευση και την ανάκτηση δεδομένων [80, 81].
    - Παρέχει υψηλή διαθεσιμότητα, κλιμακωσιμότητα και ανθεκτικότητα, καθιστώντας την ιδανική για διαχείριση μεγάλων όγκων δεδομένων.
    - Η μονάδα DID και άλλα στοιχεία της εργασίας μπορούν να αλληλεπιδρούν με την Cassandra για να αποθηκεύσουν και να ανακτήσουν δεδομένα.
- **Docker**:
    - Το Docker [82] μπορεί να χρησιμοποιηθεί για την υλοποίηση ενός ελεγχόμενου περιβάλλοντος για την εκτέλεση του DID Module και άλλων σχετιζόμενων στοιχείων.
    - Κατά αυτό το τρόπο εξασφαλίζεται η απομόνωση και φορητότητα του κώδικα, καθιστώντας εύκολη τη οποιαδήποτε επέκταση.
    - Μπορούν να δημιουργηθούν Docker containers για κάθε στοιχείο, όπως το DID Module, το Proxy Module, η Cassandra, κ.λπ., διευκολύνοντας την διαχείριση και την κλιμάκωση του συστήματος. Στο παρόν σύστημα, η κύρια χρήση έγκειται στην διαμόρφωση πολυδιάστατου αποκεντρωμένου συστήματος βάσεων δεδομένων.
- **Scala**:
    - Η Scala [83] χρησιμοποιήθηκε ως γλώσσα προγραμματισμού για την ανάπτυξη της μονάδας DID, πληρεξουσίου και διασύνδεσης με τη Cassandra.
    - Αποτελεί λειτουργική γλώσσα προγραμματισμού που προσφέρει δυνατότητες όπως η ασφάλεια τύπων, καθιστώντας την ιδανική για την ανάπτυξη σύνθετων και **αξιόπιστων** συστημάτων, ωστόσο όμως μπορεί να αποτελέσει λύση ακόμα και για εφαρμογές υλισμικού [84], παρόλο που κυμαίνεται σε υψηλότερο επίπεδο ως γλώσσα.
- **Kafka** [85]:
    - Το Kafka δύναται να χρησιμεύσει ως σύστημα διαμεσολάβησης μηνυμάτων για την ασύγχρονη επικοινωνία.
    - Αυτό μπορεί να βελτιώσει την απόδοση και την κλιμακωτότητα του συστήματος, επιτρέποντας την παράλληλη επεξεργασία μηνυμάτων, κάτι απόλυτα χρήσιμο σε ένα σύστημα με αρκετούς μεσάζοντες.

4.3.1 Έκδοση & Διαχείριση DID

Δοθείσες οι δύο έννοιες των DID και DID Εγγράφων, ένας πολίτης (ασθενής) κατέχει την ευχέρεια να εκδίδει αυτοβούλως τόσα Έγγραφα όσα και τα αντίστοιχα ιατρικά του, με κάθε Έγγραφο να αποτελεί τον συνδετικό κρίκο μεταξύ ενός ιατρικού δεδομένου και του τελικού τρίτου ως εξουσιοδοτηθέντα να

το προβάλλει, αφού φυσικά ο μεσάζων ως μονάδα πληρεξούσιας επανακρυπτογράφησης εκδώσει το απαραίτητο κλειδί για τον εκάστοτε εξουσιοδοτηθέντα. Επομένως, τα χρωματιστά βελάκια υποδηλώνουν τη μοναδιαία ροή εξουσιοδότησης, δηλαδή ένα ιατρικό δεδομένο ξεκλειδώνεται δια μέσου ενός εγγράφου DID και η απόληξη αυτού καταλήγει σε έναν μοναδικό ιατρό/υπηρεσία, αποτρέποντας τυχόν ασάφειες ή αθέμιτες παρεμβάσεις. Επιπροσθέτως, εάν για το ίδιο ιατρικό δεδομένο απαιτείται πρόσβαση από περισσότερους από έναν τρίτους, τότε ο πολίτης προχωρά στην έκδοση αντίστοιχων εγγράφων DID, αποτρέποντας την οποιαδήποτε ανάμειξη αναμεταξύ εγγράφων, κλειδιών και ιατρικών δεδομένων. Η σαφής αυτή αντιστοίχιση αποτρέπει την ασάφεια και την πιθανότητα μη εξουσιοδοτημένης πρόσβασης, διασφαλίζοντας την ακεραιότητα και την εμπιστευτικότητα των δεδομένων.



Εικόνα 17: Διακριτοποίηση ροών εξουσιοδότησης πληροφορίας

Πλαισιώνοντας καλύτερα τη ροή σε μια πιο συμπυκνωμένη μορφή, τα βήματα έχουν ως εξής:

1. **Έκδοση DID**: Ο ασθενής λαμβάνει ένα DID από μια αξιόπιστη οντότητα, όπως ένας πάροχος υγείας ή μια κυβερνητική υπηρεσία.
2. **Δημιουργία DID Εγγράφου**: Ο ασθενής δημιουργεί ένα DID Έγγραφο που συνδέει το DID του με ιατρικά δεδομένα και το κρυπτογραφεί με το δημόσιο κλειδί του εξουσιοδοτημένου τρίτου.
3. **Ανταλλαγή DID Εγγράφου**: Ο ασθενής μοιράζεται το κρυπτογραφημένο DID Έγγραφο με τον εξουσιοδοτημένο τρίτο.
4. **Πρόσβαση σε Δεδομένα**: Ο εξουσιοδοτημένος τρίτος αποκρυπτογραφεί το DID Έγγραφο με το ιδιωτικό του κλειδί και λαμβάνει πρόσβαση στα ιατρικά δεδομένα.

4.3.2 Κρυπτογράφηση Δεδομένων & Επανακρυπτογράφηση Πληρεξουσίου

Ένας διακομιστής μεσολάβησης (Proxy, ή μονάδα επανακρυπτογράφησης πληρεξουσίου) διαδραματίζει κρίσιμο ρόλο στη διευκόλυνση της ασφαλούς κοινής χρήσης δεδομένων και τη διατήρηση του απορρήτου. Καταρχάς, λειτουργεί ως αξιόπιστος ενδιάμεσος μεταξύ ενός κατόχου DID (όπως ένας ασθενής) και ενός εκπροσώπου (όπως ένας γιατρός). Όταν ένας ασθενής παραχωρεί περιορισμένη πρόσβαση σε δεδομένα υγείας, ο διακομιστής μεσολάβησης δεν λαμβάνει ο ίδιος τα δεδομένα, αλλά λαμβάνει κλειδιά εκ νέου κρυπτογράφησης. Χρησιμοποιώντας αυτά τα κλειδιά, ο διακομιστής μεσολάβησης μπορεί να μετατρέψει το κρυπτογραφημένο κείμενο από το αρχικό σχήμα κρυπτογράφησης που χρησιμοποιούσε ο ασθενής σε ένα σχήμα κατανοητό από τον γιατρό - όλα αυτά χωρίς ποτέ να αποκρυπτογραφήσει τα ευαίσθητα δεδομένα υγείας. Αυτό διασφαλίζει ότι οι πληροφορίες του ασθενούς παραμένουν προστατευμένες ακόμη και όταν κοινοποιούνται προσωρινά με άλλο μέρος ή μέλος συστήματος, και φυσικά διαφοροποιημένη τοποθέτηση αναφορικά με βάσεις δεδομένων, όντως κατανεμημένες [86, 87]. Επιπλέον, ο διακομιστής μεσολάβησης μπορεί να επιβάλει λεπτομερείς ελέγχους πρόσβασης. Εάν ο ασθενής εξουσιοδοτήσει μόνο τον γιατρό του να δει τα αρχεία προϊστορίας ιατρικού δρώμενού του, ο πληρεξούσιος μπορεί να βοηθήσει επιλεκτικά να αποκαλύψει μόνο αυτά τα συγκεκριμένα δεδομένα, διασφαλίζοντας ότι άλλες ιατρικές πληροφορίες παραμένουν απρόσιτες (αυτό επιτυγχάνεται με τη διακριτή έκδοση εγγράφων DID, καθένα με το σχετικό του ιατρικό δεδομένο). Συνοπτικά, ο διακομιστής μεσολάβησης εξουσιοδοτεί τους κατόχους DID να διατηρούν τον έλεγχο των δεδομένων τους, επιτρέποντας την ασφαλή, λεπτομερή κοινή χρήση χωρίς να διακυβεύεται η εμπιστευτικότητα.

Συμπληρώνοντας την προηγούμενη ανάλυση, ας δούμε μερικά παραδείγματα και επιπλέον πληροφορίες που ερμηνεύουν τον κρίσιμο ρόλο της μονάδας επανακρυπτογράφησης πληρεξουσίου:

Παραδείγματα Χρήσης:

- ➔ Ασθενής με Πολλαπλούς Ιατρούς: Ένας ασθενής μπορεί να εξουσιοδοτήσει τη μονάδα να εκδώσει εντοπισμένα κλειδιά πρόσβασης για τον θεράποντα ιατρό, τον καρδιολόγο και

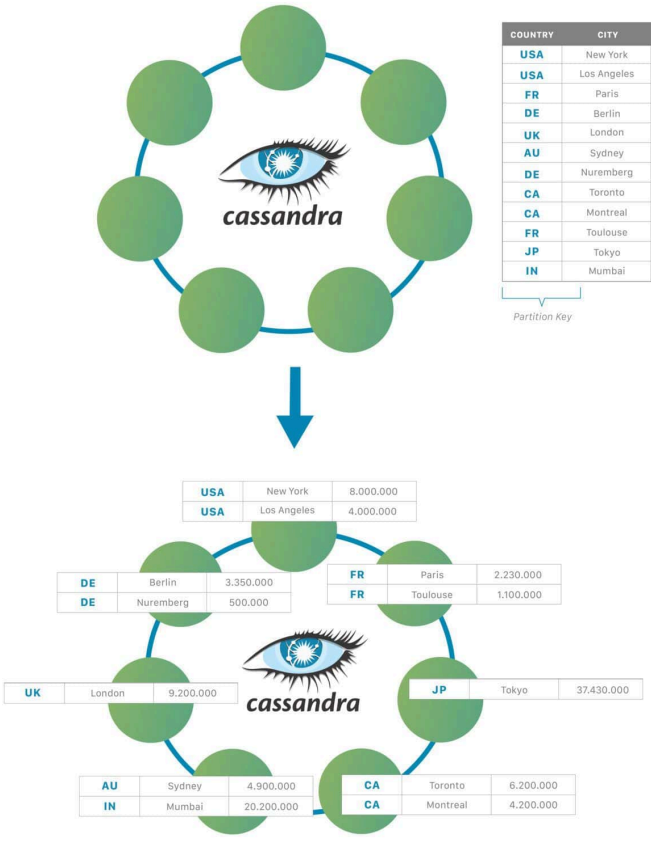
τον οδοντίατρο του, και γενικά προς κάθε έναν φορέα που ενδιαφέρεται να λάβει πρόσβαση. Έτσι, κάθε ένας λαμβάνει αντιστοιχίζεται DID Έγγραφο που του επιτρέπει πρόσβαση στα αντίστοιχα ιατρικά δεδομένα.

➔ Κλινική Έρευνα: Αναφορικά με τη δυνητική ενσωμάτωση του διαμοιρασμού μεταξύ τρίτων, σε μια κλινική μελέτη, η μονάδα δύναται να χρησιμοποιηθεί για την ασφαλή κοινοποίηση δεδομένων ασθενών στους ερευνητές. Ο ασθενής μπορεί να λαμβάνει σχετική ειδοποίηση για αυτή την έγκριση διαμοιρασμού, οπότε και να εξουσιοδοτεί τη μονάδα να εκδώσει εκ νέου ξεχωριστά κλειδιά πρόσβασης για τους ερευνητές, οι οποίοι λαμβάνουν κρυπτογραφημένα DID Έγγραφα με πρόσβαση στα απαραίτητα δεδομένα.

Ως επί το πλείστον, η επανακρυπτογράφηση επιλέχθηκε ούτως ώστε να εξυπηρετήσει έναν και μόνον σκοπό, δηλαδή το τρόπο αυτό που θα χαρίσει ακατάπαυστη για τον πολίτη/ασθενή προβολή της πορείας των δεδομένων του, και ταυτόχρονα την άμεση εξουσιοδότηση του δικαιώματος στη συγκατάθεση προς επιλεγμένους αποδέκτες, κάτι που επιτρέπει εκ φύσεως η συγκεκριμένη τεχνική. Στο παρόν σύστημα υλοποιούνται αμφότερα η περίπτωση προβολής της ροής δεδομένων, όπως επίσης και το μοντέλο ελεγχόμενης συγκατάθεσης.

4.3.3 Αποθήκευση Δεδομένων & Αποκεντρωμένη Διανομή

Για να εξασφαλιστεί υψηλή διαθεσιμότητα και επεκτασιμότητα, ένα σύστημα που υποστηρίζει ένα αποκεντρωμένο σύστημα ταυτότητας (DID) πιθανότατα θα αξιοποιήσει έναν συνδυασμό συμπλεγμάτων, διακομιστών, κατατμήσεων και αναπαραγωγής για να δημιουργήσει μια ισχυρή υποδομή διαχείρισης δεδομένων.



Εικόνα 18: Κατανομή και Αντιγραφή Δεδομένων σε Κόμβους Συμπλεγμάτων [88]

Εν προκειμένω, η εκλογή της συγκεκριμένης τεχνολογίας δεν έγινε τυχαία, καθώς αυτή αποτελεί μία ώριμη τεχνική με αποδεδειγμένες επιδόσεις ασφαλείας και ταχύτητος

έναντι άλλων λύσεων [89], παρόλο που και το Blokchain έχει ξεκινήσει να παρεισφρύει εντός του ταυτοτικού οικοσυστήματος, και δη στην υγεία [90]. Εν πρώτοις, οι συστάδες Cassandra θα μπορούσαν να δημιουργηθούν για συγκεκριμένους τομείς δεδομένων: ένα "Σύμπλεγμα Ιατρών/Φορέων" που θα περιέχει διαπιστευτήρια, πληροφορίες αδειοδότησης και ιστορικό πρακτικής, και ένα "Σύμπλεγμα Πολιτών/Ασθενών" που περιέχει αρχεία υγείας, δημογραφικά δεδομένα και προτιμήσεις συναίνεσης. Αυτός ο λογικός διαχωρισμός προάγει την ασφάλεια διαχωρίζοντας ευαίσθητες πληροφορίες με κάθε σύμπλεγμα να περιλαμβάνει πολλούς διακομιστές, παρέχοντας πλεονασμό και ανοχή σφαλμάτων. Τα δεδομένα μέσα σε ένα σύμπλεγμα διαμερίζονται, διανέμοντας κομμάτια δεδομένων στρατηγικά σε διακομιστές για βελτιστοποίηση της διανομής και της απόδοσης ερωτημάτων. Η αναπαραγωγή - replication - διασφαλίζει τη διαθεσιμότητα δεδομένων, με αντίγραφα δεδομένων που υπάρχουν σε πολλούς κόμβους. Τουτέστιν, εάν ένας κόμβος αποτύχει στο "Cluster Citizen/Patient Cluster", για παράδειγμα, το σύστημα μπορεί ακόμα να εξυπηρετήσει αρχεία ασθενών από άλλα αντίγραφα. Οι αλληλεπιδράσεις μεταξύ συμπλεγμάτων πιθανότατα θα ενορχηστρώνονται μέσω API και θα διέπονται από συμφωνίες κοινής χρήσης δεδομένων που βασίζονται στο DID ενός χρήστη. Για παράδειγμα, όταν ένας πολίτης/ασθενής παραχωρεί σε έναν γιατρό προσωρινή πρόσβαση για να δει τα αρχεία υγείας του, το DID του θα μπορούσε να ενεργοποιήσει μια ασφαλή ανταλλαγή δεδομένων μεταξύ του "Cluster Citizen/Patient" και του "Doctor Cluster". Σε σύνοψη αυτών, για να επιτευχθεί το όραμα ενός εθνικά αναγνωρισμένου DID, το σύστημα θα πρέπει να φιλοξενήσει τη δυναμική συμμετοχή στις υπηρεσίες. Οπότε, υπηρεσίες όπως μια ιατρική αρχή αδειοδότησης, ένα σύστημα νοσοκομειακών αρχείων ή μια πύλη πολιτών/ασθενών ενδέχεται να αλληλεπιδράσουν με τα συμπλέγματα δεδομένων υπό διαφορετικές συνθήκες βάσει αδειών που βασίζονται στο DID. Για παράδειγμα, μια υπηρεσία της αρχής αδειοδότησης θα αλληλεπιδράσει με το "Cluster Doctor" κατά την επαλήθευση διαπιστευτηρίων αλλά όχι απαραίτητα με το "Cluster Citizen/Patient Cluster". Η ευελιξία είναι το κλειδί: οι υπηρεσίες θα πρέπει να μπορούν να ενώνουν τα σχετικά clusters ή διακομιστές εντός συμπλέγματος ανάλογα με τις ανάγκες, να πιστοποιούνται και να εξουσιοδοτούνται με βάση τις αλληλεπιδράσεις που ξεκινούν μέσω του συστήματος DID υπηρεσίες που αλληλεπιδρούν απρόσκοπτα γύρω από την βασική ιδέα ενός DID ελεγχόμενου από τον χρήστη.

4.3.4 Ενορχήστρωση Ροής Εργασιών & Παρακολούθηση Δεδομένων

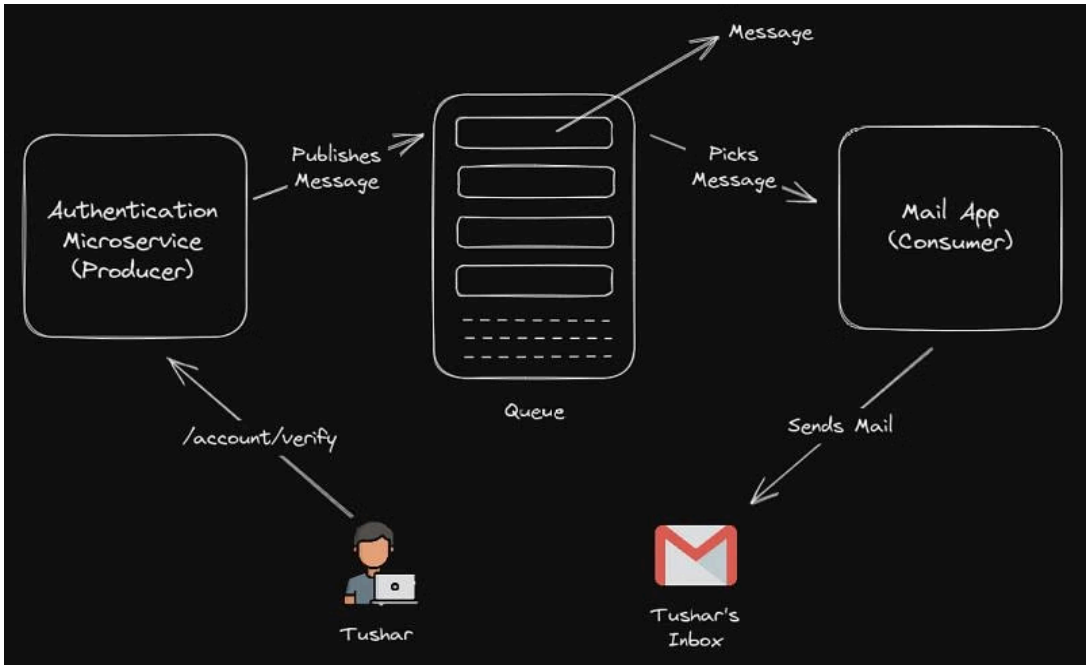
Το Apache Kafka διαδραματίζει κρίσιμο ρόλο ως ισχυρός, επεκτάσιμος και ανθεκτικός σε σφάλματα κορμός μηνυμάτων, διευκολύνοντας τον ακατάσχετο συντονισμό και διασφαλίζοντας τη συνοχή δεδομένων σε όλο το οικοσύστημα DID, κάτι αδιαμφισβήτητα θεσπισμένο ως χαρακτηριστικό ενός οικοσυστήματος που καλείται να αναλάβει τέτοιες διαστάσεις από άποψη όχι μόνο όγκου πληροφορίας, αλλά και διαφοροποίησης στη πηγή αυτής. Ας εμβαθύνουμε στο γιατί το Kafka είναι στρατηγική επιλογή:

➢ *Αρχιτεκτονική βάσει συμβάντων σε πραγματικό χρόνο*: Καθώς οι χρήστες αλληλεπιδρούν με τα DID τους, δημιουργούνται συμβάντα – ενημερώσεις συναίνεσης, ανακλήσεις πρόσβασης ή αλλαγές σε προσωπικά δεδομένα. Το μοντέλο δημοσίευσης-εγγραφής του Κάφκα επιτρέπει στα σχετικά στοιχεία να εγγραφούν σε συγκεκριμένα θέματα εκδηλώσεων. Αυτό σημαίνει ότι οι υπηρεσίες ειδοποιούνται αμέσως, επιτρέποντάς τους να αντιδρούν άμεσα και να διατηρούν μια συνεπή κατάσταση σε όλο το σύστημα.

➢ *Ανθεκτική προσμονή μηνυμάτων*: Σε ένα κατανεμημένο περιβάλλον όπως σε αυτό που προτείνεται στην ημέτερη εργασία, οι προσωρινές διακοπές λειτουργίας ή οι διακοπές δικτύου είναι πραγματικότητα. Η ικανότητα του Κάφκα να αποθηκεύει διαρκώς μηνύματα, ακόμη και όταν τα συστήματα είναι εκτός σύνδεσης, προστατεύει την ακεραιότητα του οικοσυστήματος DID.

➢ *Επεκτασιμότητα και απόδοση*: Το δίκτυό DID είναι πιθανό να συνεχίζει την ανάπτυξη με την πάροδο του χρόνου. Ο σχεδιασμός του Kafka το καθιστά εξαιρετικά επεκτάσιμο, δίνοντάς του τη δυνατότητα να χειρίζεται αυξανόμενους όγκους μηνυμάτων και έναν αυξανόμενο αριθμό συνδρομητικών υπηρεσιών. Αυτή η προσαρμοστικότητα αποτρέπει τα σημεία συμφόρησης που θα μπορούσαν να εμποδίσουν την ανταπόκριση του αποκεντρωμένου συστήματος ταυτότητάς σας.

➢ *Υπηρεσίες αποσύνδεσης*: Ο Kafka λειτουργεί ως ενδιάμεση μνήμη μεταξύ διαφορετικών στοιχείων. Αυτή η αποσύνδεση προάγει την ευελιξία – οι υπηρεσίες μπορούν να προστεθούν, να ενημερωθούν ή να αντικατασταθούν χωρίς να διαταραχθεί η συνολική ροή πληροφοριών. Αυτή η χαλαρή σύζευξη απλοποιεί την ανάπτυξη και τη συντήρηση, επιτρέποντας σε διαφορετικά μέρη του οικοσυστήματος DD να εξελιχθούν ανεξάρτητα.



Εικόνα 19: Αναπαράσταση σειριακής καταγραφής και αποστολής μηνυμάτων ενορχήστρωσης διεργασιών [91]

5

Υλοποίηση

5.1 Μονάδα Δημιουργίας & Διαχείρισης DID (Scala)

Σε αυτή τη υποενότητα θα πραγματοποιηθεί μια εκτενής ανάλυση των κυρίων παραγόντων που απαρτίζουν τη δομική μονάδα κώδικα αναφορικά με τα αποκεντρωμένα χαρακτηριστικά. Βασικό κορμό της υλοποίησης περιλαμβάνει η δημιουργία και διαχείριση των Decentralized Identifiers (DIDs) για τους χρήστες όπου οι λειτουργίες αναπαριστώνται από διακριτές κλάσεις της γλώσσας ούτως ώστε να επιτευχθεί τόσο η καλή συντήρηση του κώδικος όσο και η διευκόλυνση μελλοντικών κλιμακούμενων επεκτάσεων, τεχνική η οποία συμβάλλει οπωσδήποτε δια του κατακερματισμού του κώδικα σε μπλοκ και όχι σε μονολιθική συγγραφή αυτού.

Το πακέτο κώδικα Scala που αφορά τα Decentralized Identifiers (DIDs) περιλαμβάνει τις ακόλουθες κλάσεις και λειτουργίες:

➔ ***Controller*** ***(Εμπεριέχεται στο DIDDocument.scala)***: Αυτή η κλάση αντιπροσωπεύει τον ρόλο ενός *controller* σε ένα DID. Περιλαμβάνει πληροφορίες όπως το *ID* του controller, τον τύπο, το όνομα και το *public key* του, ήτοι τον κύριο κάτοχο στον οποίο γίνεται η οποιαδήποτε αναφορά περί δεδομένων.

➔ ***KeyAgreement*** ***(Εμπεριέχεται στο DIDDocument.scala)***: Αντιπροσωπεύει μια συμφωνία κλειδιών σε ένα DID. Περιλαμβάνει πληροφορίες όπως το ID της συμφωνίας, το *public key*, τον τύπο και τον *delegatee* (δηλαδή αυτής της οντότητας στην οποία θα πραγματοποιηθεί εν γένει εξουσιοδότηση προβολής δεδομένων).

➔ ***VerificationMethod*** ***(Εμπεριέχεται στο DIDDocument.scala)***: Αντιπροσωπεύει έναν τρόπο επαλήθευσης σε ένα DID. Περιλαμβάνει πληροφορίες όπως το *ID* της μεθόδου, τον τύπο, τον *controller* και το *public key*.

➔ ***Authentication*** ***(Εμπεριέχεται στο DIDDocument.scala)***: Αντιπροσωπεύει μια μέθοδο πιστοποίησης σε ένα DID. Περιλαμβάνει πληροφορίες όπως τον τύπο και το *public key*.

➔ ***Service*** ***(Εμπεριέχεται στο DIDDocument.scala)***: Αποτελεί σταθερό πεδίο που που συνιστά το εγκαθιδρυμένο πρότυπο του W3C, και περιλαμβάνει πληροφορίες όπως το ID της υπηρεσίας, τον τύπο και το service endpoint που θα μπορούσε να συσχετιστεί ο κάτοχος του αναγνωριστικού.

➔ ***DIDDocument***: Αυτή η κλάση αναφύεται ως ο ακρογωνιαίος λίθος του συστήματος και λογίζεται ως ένα Έγγραφο DID περιλαμβάνοντας πληροφορίες όπως ο controller, οι συμφωνίες κλειδιών, οι μέθοδοι επαλήθευσης και οι υπηρεσίες που συσχετίζονται με το συγκεκριμένο ***DID***. Επίσης, περιλαμβάνει μεθόδους κωδικοποίησης και αποκωδικοποίησης για τη σειριοποίηση του αντικειμένου σε JSON και αντίστροφα. Αυτή η κλάση παίζει καθοριστικό ρόλο στη δόμηση του όλου συστήματος και είναι αυτή στην οποία στηρίζεται ολάκερη η αρχιτεκτονική όπως θα παρουσιαστεί παρακάτω, στην αναπαράσταση του προγράμματος.

➔ ***DIDVerifier***: Εμπεριέχει λειτουργίες που αφορούν τους πιστοποιητές DID, όπως την επαλήθευση υπογραφών και την ενημέρωση των DID *documents* τους.

➔ ***VerifierStorage***: Αυτό το αντικείμενο αποθηκεύει τα DID documents των πιστοποιητών DID στη προσωρινή μνήμη του προγραμματιστικού περιβάλλοντος (*cache*)

➔ ***Holder***: Αντιπροσωπεύει έναν κάτοχο DID και περιλαμβάνει μεθόδους για την ανάκτηση του αντίστοιχου DID document.

➔ ***HolderStorage***: Αποθηκεύει τα DID *documents* των κατόχων DID κατά τον ίδιο τρόπο με τη προηγούμενη κλάση

➔ ***Issuer***: Αυτή η κλάση παρέχει λειτουργίες για την έκδοση DIDs και DID *documents*.

Προχωρώντας στην ανάλυση των δύο κυρίων συνιστωσών DIDDocument και Issuer θα πραγματοποιηθεί μια εμβάθυνση στο πως συνδέουν την έκδοση ενός αναγνωριστικού με τα υπόλοιπα στάδια της αποθήκευσης και κρυπτογράφησης περιεχομένου. Ας δούμε για αρχή πιο αναλυτικά τη λειτουργία του πρώτου:

I. **Κλάση** DIDDocument:

A. Αυτή η κλάση περιγράφει ένα DID document σύμφωνα με το πρότυπο του W3C. Η κλάση περιέχει πεδία για το ID του document, τους ελεγκτές (controllers), τις μεθόδους συμφωνίας κλειδιών (key agreement methods), τις μεθόδους επαλήθευσης (verification methods) και τις υπηρεσίες (services) που παρέχονται από το DID document.

B. Υπάρχουν επίσης περιορισμοί στους τύπους των πεδίων, όπως για παράδειγμα ο τύπος του κλειδιού και ο τύπος της υπηρεσίας. Συμπεριλαμβάνονται μεθόδοι κωδικοποίησης (Encoder) και αποκωδικοποίησης (Decoder) για κάθε τύπο που χρησιμοποιείται στο DID document, ώστε να είναι δυνατή η μετατροπή του σε JSON και αντίστροφα.

1. **Μέθοδος** generateDIDDocument:

a) Αυτή η μέθοδος δημιουργεί ένα DID document για έναν προσδιορισμένο χρήστη (συνήθως έναν ασθενή).

b) Δέχεται το DID, το κλειδί και το όνομα του χρήστη, καθώς επίσης και το DID του αντιπροσώπου (delegateeDID). Δημιουργεί διάφορα στοιχεία του DID document όπως ελεγκτές, μέθοδους συμφωνίας κλειδιών και υπηρεσίες και επιστρέφει ένα αντικείμενο DIDDocument που αντιπροσωπεύει το DID document του συγκεκριμένου χρήστη (δηλαδή ταυτίζεται άμεσα με ένα και μόνο συγκεκριμένο αποκεντρωμένο αναγνωριστικό). Το αρχείο αυτό είναι υπεύθυνο για τη δημιουργία των DID documents που απαιτούνται για την αυθεντικοποίηση και τον έλεγχο πρόσβασης στο σύστημα.

I. **Κλάση** Issuer:

A. Το αρχείο Issuer.scala παρέχει λειτουργίες για την έκδοση των DIDs και DID documents και προσομοιώνει τον εκδότη των αποκεντρωμένων αναγνωριστικών θέτοντας την επίσημη στάμπα επικύρωσης. Ειδικότερα, είναι υπεύθυνο για τη δημιουργία και την αρχικοποίηση DIDs και τα επακόλουθα αυτών DID documents στο σύστημα. Επιτρέπει στους χρήστες να δημιουργούν νέα DIDs και να παρέχουν τα αντίστοιχα documents που απαιτούνται για την αυθεντικοποίηση και τον έλεγχο πρόσβασης.

1. **Μέθοδος** issueDID:

a) Η μέθοδος δέχεται ένα προαιρετικό όνομα name, δημιουργεί ένα DID χρησιμοποιώντας ένα τυχαίο UUID και στη συνέχεια δημιουργεί έναν κάτοχο (Holder) για το συγκεκριμένο DID με βάση το όνομα που δόθηκε.

b) Εκδίδει ένα ζεύγος κλειδιών (public-private) χρησιμοποιώντας τη μέθοδο Asymmetric.generatePPK() αποθηκεύοντας το ιδιωτικό κλειδί σε ένα αρχείο με βάση το DID και το όνομα του κατόχου.

2. **Μέθοδος** issueDIDDocument:

a) Αυτή η μέθοδος δέχεται πολλά ορίσματα, συμπεριλαμβανομένου του DID, του encoded public key, του ονόματος και του delegateeDID. Χρησιμοποιεί την generateDIDDocument από την κλάση DIDDocument για να εκμαιεύσει ένα νέο DID document.

Καταλήγοντας, η ανωτέρα τεκμηρίωση του πηγαίου κώδικα που αφορά έκδοση/διαχείριση των αναγνωριστικών και των συν αυτώ συνοψίζει επαρκώς ό,τι θα χρειαστεί ένας μελετητής ή προγραμματιστής ούτως ώστε να λάβει καλύτερη σχετική επίγνωση.

5.2 Μονάδα Πληρεξούσιας Επανακρυπτογράφησης Δεδομένων (Scala)

Ως επί το πλείστον, η υλοποίηση βασίζεται σε βιβλιοθήκες και πλατφόρμες που παρέχουν κρυπτογραφικές λειτουργίες και ασφάλεια, όπως η tsec για την κρυπτογράφηση και αποκρυπτογράφηση δεδομένων, καθώς και η com.ironcorelabs.recrypt για τη διαχείριση κλειδιών και την υλοποίηση του σχήματος proxy re-encryption. Η επιλογή του proxy re-encryption έγινε γιατί προσφέρει μια αποτελεσματική λύση για την ασφαλή διαβίβαση δεδομένων ανάμεσα σε δύο μεριές, όπου η μία πλευρά Α επιθυμεί να επιστρέψει στη δεύτερη Β, ελεγχόμενη πρόσβαση σε κρυπτογραφημένα δεδομένα χωρίς να αποκαλυφθεί η το ιδιωτικό κλειδί της πρώτης. Η υλοποίηση της συγκεκριμένης τεχνικής πραγματοποιείται διά μέσου ενός ενδιάμεσου proxy/μεσάζοντα ο οποίος στο εν λόγω σύστημα λογίζεται ως υπολογιστικός πόρος που ανήκει στο πελάτη. Αυτός έχει τη δυνατότητα να επανακρυπτογραφεί δεδομένα μεταξύ των εμπλεκόμενων μερών διατηρώντας τόσο την εχεμύθεια - αφού δε μπορεί να διαβάσει τα αρχικά μηνύματα των αποστολέων - όσο και την εντοπισμένη ανάγκη της ιδιωτικότητας . Τεχνικά, η υλοποίηση απαιτεί τη χρήση κρυπτογραφικών μεθόδων ελλειπτικής καμπύλης - ECC - καθώς αυτή επιτάσσει και η αντίστοιχη βιβλιοθήκη των ironcorelabs.

Ειδικότερα, ο κώδικας εντοπίζεται αποκλειστικά σε μια λιτή και λειτουργική κλάση ονόματι ProxyPipeline με μία μέθοδο ορισμένη ως proxyJobs, στην οποία προσδιορίζεται η ροή επανακρυπτογράφησης ως εξής:

Παράμετροι:

- *coreApi*: Αντικείμενο που παρέχει πρόσβαση σε διάφορες κρυπτογραφικές υπηρεσίες της σχετικής βιβλιοθήκης του ironcorelabs.
- *symmetric_key*: Το μήνυμα προς διαμεσολάβηση, σε μορφή ευανάγνωστου κειμένου. Λόγω περιορισμού της βιβλιοθήκης χρήσεως, το μήνυμα οφείλει να είναι ενός ειδικού τύπου και όχι αυθαίρετο, ήτοι περιορισμός στο της μπορεί να αποτελέσει αντικείμενο κρυπτογράφησης. Τουτέστι, η αρχιτεκτονική μεταβλήθηκε ώστε το μήνυμα να αποτελεί ένα **συμμετρικό κλειδί κρυπτογράφησης** το οποίο είναι ικανό να ξεκλειδώσει εντέλει τα πραγματικά δεδομένα.
- *senderPrivateKey*: Το κλειδί αποκρυπτογράφησης του αποστολέα, απαραίτητο για την υπογραφή και αποκρυπτογράφηση των δικών του μηνυμάτων.
- *senderPublicKey*: Τέλος, ο κλειδί κρυπτογράφησης του αποστολέα, που επιτρέπει σε τρίτους να κρυπτογραφούν μηνύματα προς αυτόν και να επικυρώνουν τις υπογραφές του.
- *receiverPublicKey*: Το κλειδί κρυπτογράφησης του αποδέκτη που θα χρησιμοποιηθεί για την τελική κρυπτογράφηση του μηνύματος. Αξίζει να σημειωθεί πως αυτό αποτελεί συνδετικό κρίκο

ώστε ο μετασχηματισμός να ξεκινήσει από κάτι που αποκρυπτογραφείται από τον Α (αποστολέας) και να μεταβληθεί σε κάτι που αποκρυπτογραφείται μόνον από τον Β (αποδέκτης).

Ως εκ τούτου, με το πέρας της παραπάνω διαδικασίας επιστρέφεται η παράμετρος transformedValue (μετασχηματισμένη τιμή) η οποία και αποθηκεύεται σε αντίστοιχη θέση της αποκεντρωμένης βάσης, εγκαθιδρύοντας έτσι έναν δεσμό αναμεταξύ αποστολέα και αποδέκτη. Συνεπώς, τη συγκεκριμένη παράμετρο μπορεί να ξεκλειδώσει μόνον ο εξουσιοδοτημένος αποδέκτης κάθε φορά, παρέχοντας φυσικά το ιδιωτικό του κλειδί.

5.3 Ενσωμάτωση Αποκεντρωμένης Αποθήκευσης (Cassandra)

Η κατασκευή του αποκεντρωμένου συστήματος βάσης δεδομένων ακολούθησε μια απλή προσέγγιση με βασικές παραμέτρους ως πίνακες, οι οποίοι θα εξυπηρετούν σε εύρος τις αντίστοιχες ανάγκες των DID εγγράφων, τους διαύλους σύζευξης μεταξύ αποστολέα και αποδέκτη κρυπτογραφημένων δεδομένων, και φυσικά τη καταγραφή ροής συγκατάθεσης και προβολής αυτών. Ειδικότερα, το σύστημα απαρτίζεται από τα παρακάτω χαρακτηριστικά (πίνακες/προγραμματιστικές κλάσεις διεπαφής):

➔ Διαχείριση DID:

- Προσθήκη DID: Προσθέτει έναν Αναγνωριστικό Αποκεντρωμένης Διαχείρισης (DID) μαζί με λεπτομέρειες όπως ο τύπος, το δημόσιο κλειδί, το hash του ιδιωτικού κλειδιού και το όνομα στον πίνακα did_registry. Αυτό λειτουργεί σαν ένας δημόσιος κατάλογος DID, ήτοι το DID Registry.
- Προσθήκη Εγγράφου DID: Αποθηκεύει ένα έγγραφο DID που σχετίζεται με ένα DID στον πίνακα did_documents_registry. Το έγγραφο DID μπορεί να περιέχει περισσότερες πληροφορίες για το DID, όπως πληροφορίες επικοινωνίας ή περιγραφές υπηρεσιών που προσφέρει. Όπως έχει προαναφερθεί στην ενότητα περί δόμησης συστήματος, το κάθε έγγραφο **DID** παράγεται ως αντίστοιχο ζευγάρι ενός πραγματικού εγγράφου προσωπικών - στη περίπτωση μελέτης ιατρικών - δεδομένων.
- Ανάκτηση Δημοσίου Κλειδιού από DID: Εξάγει το δημόσιο κλειδί από το έγγραφο DID που είναι αποθηκευμένο στην Cassandra. Το δημόσιο κλειδί είναι απαραίτητο για την επαλήθευση των υπογραφών και τον έλεγχο της ταυτότητας, οπότε δε θα μπορούσε να λείπει ως βασική μέθοδος.
- Ανάκτηση Ονόματος από DID: Εξάγει το όνομα που σχετίζεται με το έγγραφο DID αποθηκευμένο στην Cassandra. Το όνομα μπορεί να είναι πιο φιλικό προς τον χρήστη από μια διεύθυνση DID όπως ακριβώς αποτυπώνεται και στα δικτυακά συστήματα με τη χρήση του DNS έναντι των αμιγώς αριθμητικών διευθύνσεων διαδικτυακού πρωτοκόλλου (IP).

- Ανάκτηση Αποθηκευμένου Hash Ιδιωτικού Κλειδιού: Ανακτά το αποθηκευμένο hash του ιδιωτικού κλειδιού για ένα DID από την Cassandra (αν υπάρχει). Εδώ χρειάζεται ειδική μνεία ώστε να ξεκαθαριστεί πως συνήθως, ένα ιδιωτικό κλειδί δεν πρέπει ποτέ να αποθηκευτεί τοιουτοτρόπως σε μια βάση δεδομένων ανεξαρτήτως της ασφάλειάς στης. Εντούτοις, αυτό το κλειδί άπτεται της προσωπικής αποθήκευσης του χρήστη είτε σε ηλεκτρονική συσκευή, είτε σε κλειδί υλισμικού (Hardware Key)

➔ Διαχείριση Δεδομένων Υγείας:

- Προσθήκη Δεδομένων Υγείας: Προσθέτει μια εγγραφή για δεδομένα υγείας που σχετίζονται με έναν χρήστη DID στον πίνακα healthcare_data. Αυτά τα δεδομένα είναι κρυπτογραφούνται και αναπαρίστανται ως πίνακας Byte από πρώτιστη μετατροπή του (PDF).
- Ανάκτηση Κρυπτογραφημένων Δεδομένων: Αντανακλώντας τη παραπάνω μέθοδο, αυτή ανακτά τα κρυπτογραφημένα δεδομένα υγείας που σχετίζονται με ένα DID και ένα αναγνωριστικό εγγραφής.

➔ Έλεγχος Πρόσβασης και Διαχειρισμός Ιστορικού:

- Προσθήκη Πολιτικής Ελέγχου Πρόσβασης: Δημιουργεί μια καταχώρηση ελέγχου πρόσβασης που καθορίζει ποιοι χρήστες έχουν πρόσβαση στα δεδομένα ενός άλλου χρήστη στον πίνακα access_control_policies (ως DID προς DID).
- Ανάκτηση Ελέγχου Πρόσβασης: Στο ίδιο φάσμα, εδώ πραγματοποιείται ανάκτηση για καταχωρήσεις ελέγχου πρόσβασης που καθορίζουν τους χρήστες που επιτρέπεται να έχουν πρόσβαση στα δεδομένα ενός συγκεκριμένου DID, και αυτό συνδέεται άμεσα με τη προβολή ροής συγκατάθεσης.
- Προσθήκη Ιστορικού Ελέγχων: Προσθέτει μια καταχώρηση στον πίνακα audit_trails για να παρακολουθεί τις ενέργειες που εκτελούνται σε ένα DID, όπως ποιος χρηστές πρόσβασε σε ποια δεδομένα. Αυτό επιτρέπει την παρακολούθηση και τον εντοπισμό τυχόν κακόβουλων ενεργειών, και πέραν αυτών αποτελεί τη πεμπτουσία του σκοπού αυτού του έργου, τουλάχιστον στο πρώιμό του στάδιο.
- Εκτύπωση Ιστορικού Ελέγχων: Εμφανίζει τις καταχωρήσεις ιστορικού που σχετίζονται με ένα συγκεκριμένο DID με συνέπεια αυτό μπορεί να χρησιμοποιηθεί για την εξέταση των δραστηριοτήτων σε ένα DID.

➔ Διαχείριση Εξουσιοδότησης Δεδομένων:

- Προσθήκη Δεδομένων Εξουσιοδότησης: Αποθηκεύει μετασχηματισμένες τιμές και σχετικές πληροφορίες για την εξουσιοδότηση δεδομένων στον πίνακα delegation_data. Ειδικότερα, η παρούσα μέθοδος αντικατοπτρίζει τον αντίστοιχο πίνακα στον οποίο αποτυπώνεται πλήρως το ποιό έγγραφο DID συνδέεται με ποιά ιατρικά δεδομένα. Αξίζει να σημειωθεί πως η αποκεντρωμένη φύση της Cassandra εξασφαλίζει πως ο

προαναφερθείς αντικατοπτρισμός τείνει να βρίσκεται πάντα στο προσκήνιο για όποιον ενδιαφέρεται να τον προβάλλει ως πληροφορία αφού οι αντιγραφές της βάσεως είναι πάντα (θεωρητικά) κατανεμημένες στους κόμβους - τουτέστιν αδιαλείπτως διαθέσιμες - διασφαλίζοντας έτσι την άρρηκτη διαθεσιμότητα ως κεντρικό πυλώνα του μηχανισμού.

- Ανάκτηση Μετασχηματισμένης Τιμής: Τέλος, η αντίθετη της προηγούμενης μέθοδος αφορά ως είθισται την ανάκτηση δεδομένων εξουσιοδότησης που σχετίζονται με έναν εξουσιοδοτούντα και εξουσιοδοτημένο χρήστη από τον πίνακα delegation_data.

5.4 Συγχρονισμός Κατανεμημένων Κόμβων (Docker)

Δια μέσου του Docker δόθηκε η ευκαιρία για ένα ιδανικό περιβάλλον οργάνωσης και εκτέλεσης συμπλεγμάτων Cassandra, επιτρέποντας την εύκολη διαχείριση και κλιμάκωση του συστήματος.

Συγχρονισμός Δεδομένων και Διαχείριση:

1) Εκκίνηση Κόμβων Cassandra με Docker:
   a) Χρησιμοποιούνται εντολές Docker για την εκκίνηση και διαχείριση των κόμβων Cassandra.
   b) Κάθε κόμβος εκτελείται σε ένα μεμονωμένο container Docker, διασφαλίζοντας την απομόνωση και την ευελιξία.
   c) Δημιουργούνται Docker volumes για την αποθήκευση των δεδομένων Cassandra, διασφαλίζοντας την ανθεκτικότητα και την ασφάλεια των δεδομένων.
2) Συγχρονισμός Πινάκων σε Κάθε Κόμβο:
   a) Χρησιμοποιήθηκαν scripts CQL για τη δημιουργία των πινάκων σε κάθε κόμβο του Cassandra cluster.
   b) Το docker-cqlsh χρησιμοποιείται για την εκτέλεση των scripts CQL σε κάθε container κόμβου.
   c) Ως εκ τούτου, αυτό εξασφαλίζει ότι οι πίνακες και το keyspace μοιράζονται ομοιόμορφα σε όλο το σύμπλεγμα, διασφαλίζοντας την ακεραιότητα και τη συνέπεια των δεδομένων.

Προσδιορίζοντας την εκκίνηση του συμπλέγματος, το Docker χωρίστηκε σε δύο διακριτά αρχεία, docker.sh και το scripts.sh. Κατά το πρώτο παρέχεται ένας αυτοματοποιημένος τρόπο για την ανάπτυξη ενός συμπλέγματος Cassandra χρησιμοποιώντας Docker containers.

- Δημιουργία Docker Volumes για Δεδομένα Cassandra (Για Κάθε Κόμβο, ***docker.sh***):
  - docker volume create cassandra_data_node_1
  - docker volume create cassandra_data_node_2
  - docker volume create cassandra_data_node_3

- Το script ξεκινά δημιουργώντας ξεχωριστά Docker volumes για κάθε κόμβο Cassandra.
- Αυτά τα volumes αποθηκεύουν τα δεδομένα της βάσης δεδομένων Cassandra για κάθε κόμβο.
- Χρησιμοποιώντας ξεχωριστά volumes διασφαλίζεται η απομόνωση των δεδομένων μεταξύ των κόμβων και η ανθεκτικότητα του συστήματος σε περίπτωση αποτυχίας ενός κόμβου.
- Τα ονόματα των volumes (cassandra_data_node_1, cassandra_data_node_2, κ.λπ.) βοηθούν στον διαχωρισμό των δεδομένων κάθε κόμβου.

Εν κατακλείδι, η συγκεκριμένη δόμηση έχει ως απώτερο σκοπό τη μελλοντική κλιμάκωση του συστήματος σε δυνητικά μεγάλο αριθμό κόμβων από ετερόκλητα συμπλέγματα. Αμέσως μετά περνά η εκτέλεση του ***scripts.sh,*** κατά το οποίο εκτελείται κατά συρροήν η εντολή docker run και αυτομάτως *"ανεβαίνουν"* σε λειτουργία οι τρεις ορισθέντες κόμβοι.

Οι παράμετροι που χρησιμοποιούνται στην εντολή docker run έχουν ως εξής:

- ☐ **--rm**: Αφαιρεί αυτόματα το container μετά την ολοκλήρωση της εκτέλεσης. Αυτό σημαίνει ότι τα containers δημιουργούνται μόνο για την εκτέλεση του CQL script και δεν χρειάζεται να διαχειριστούμε χειροκίνητα τον κύκλο ζωής τους.
- ☐ **--network cassandra**: Τοποθετεί κάθε container στο δίκτυο των αποκεντρωμένων βάσεων, εξασφαλίζοντας ότι τα containers μπορούν να επικοινωνήσουν με τους όλους τους σχετικούς εντός του δακτυλίου κόμβους.
- ☐ **-v "$(pwd)/data.cql:/scripts/data.cql"**: Συνδέει το CQL script data.cql από τον τρέχοντα κατάλογο με τον κατάλογο /scripts/data.cql μέσα στον container. Αυτό τοποθετεί το script εντός του περιβάλλοντος του container για εκτέλεση.
- ☐ **-v cql_scripts:/scripts**: Συνδέει το Docker volume cql_scripts που δημιουργήθηκε στο βήμα 1 με τον κατάλογο /scripts μέσα στον container. Αυτό επιτρέπει στο script data.cql να προσπελάσει τυχόν πρόσθετα scripts ή αρχεία διαμόρφωσης που βρίσκονται στο volume.
- ☐ **-e CQLSH_HOST=cassandra-node-1, -e CQLSH_HOST=cassandra-node-2, -e CQLSH_HOST=cassandra-node-3**: Ορίζει το hostname του αντίστοιχου Cassandra node στον οποίο θα συνδεθεί το container (ανάλογα με την εκτέλεση). Αυτό επιτρέπει στο script να εκτελέσει τις εντολές CQL στον συγκεκριμένο κόμβο.
- ☐ **-e CQLSH_PORT=9042**: Ορίζει την θύρα CQLSH (9042) για επικοινωνία με τον Cassandra node.

- ☐ **-e CQLVERSION=3.4.6**: Ορίζει την έκδοση CQL (3.4.6) που θα χρησιμοποιηθεί από το docker-cqlsh.
- ☐ **nuvo/docker-cqlsh**: Εκτελεί την εικόνα Docker nuvo/docker-cqlsh, η οποία περιέχει το εργαλείο cqlsh για την εκτέλεση CQL scripts.

Με αυτά, το αποκεντρωμένο δίκτυο τίθεται σε *"ζωντανή"* εκτέλεση και ο νοητός δακτύλιος με τους κόμβους έχει σχηματισθεί. Εφεξής υπάρχει κάθε δυνατότητα παρεμβολής και προβολής στη βάση ανεξαρτήτως κόμβου, γρήγορα, μέσω της ενσωματωμένης δυνατότητας χρήσης τερματικού με την ειδική εντολή docker exec -it [**όνομα κόμβου**] cqlsh.

5.5 Ροή Συμβάντων (Kafka)

Αρχικά ορίζεται ένας Kafka Producer ο οποίος στέλνει μηνύματα σε ένα Kafka topic με το όνομα "did-issuing-topic", με τα μηνύματα να σχετίζονται πρωτίστως και άμεσα με τις διαδικασίες εκκίνησης έκδοσης και επανακρυπτογράφησης DIDs.

Αναφορικά με τον Παραγωγό (*Producer)*:

1. Σύνδεση: Ο παραγωγός συνδέεται με έναν Kafka broker στη διεύθυνση "localhost:9092" (τοπική εγκατάσταση).
2. Ορισμός Θέματος: Το καθορισμένο topic είναι "did-issuing-topic".
3. Serializers: Χρησιμοποιούνται StringSerializers για κλειδιά και τιμές. Αυτό εξασφαλίζει πως τα επικείμενα μηνύματα θα μετατραπούν σε strings πριν την αποστολή.
4. Αποστολή Μηνύματος: Ο παραγωγός στέλνει ένα μήνυμα στο topic. Το κλειδί του μηνύματος είναι null.
5. Κλείσιμο του Producer: Τέλος, κλείνει η σύνδεση του Kafka producer.

Στον αντίποδα, ο καταναλωτής (*Consumer*) είναι υπεύθυνος για την κατανάλωση μηνυμάτων από ένα θέμα (topic) στο Apache Kafka, εκτελώντας διάφορες ενέργειες ανάλογα με το περιεχόμενο των μηνυμάτων, τα οποία έχουν δομηθεί τοιουτοτρόπως ώστε να εξυπηρετούν όλο το φάσμα του ταυτοτικού συστήματος, από τα απλούστερα μέχρι τα πιο νευραλγικά σημεία.

Ας αναλύσουμε ενδελεχέστερα τον κώδικα του:

1. Δήλωση των περιπτώσεων ***ConsumerResult***:
   - Η κλάση ConsumerResult ορίζει το αποτέλεσμα της κατανάλωσης μηνυμάτων από τον καταναλωτή.
   - Υποστηρίζονται τρία διαφορετικά είδη αποτελεσμάτων: TypeIssueResult, TypeIssueDIDDOCResult, και TypeInvokeResult.

2. Κλάση ***Consumer***:

- Αρχικώς, η πρωταρχική κλάση Consumer συγκεντρώνει μια σειρά από μεθόδους για την κατανάλωση μηνυμάτων από το Kafka και την επεξεργασία τους.
- Δευτερευόντως, η μέθοδος consume είναι υπεύθυνη για την πραγματική κατανάλωση των μηνυμάτων από το Kafka, με αυτή να καθορίζει τις ρυθμίσεις του Kafka consumer και καταναλώνει μηνύματα από το θέμα "did-issuing-topic", τουτέστιν τυπικός κώδικας εδραίωσης συστήματος.
- Κατά την κατανάλωση ελέγχεται το περιεχόμενο κάθε μηνύματος και καλεί τις αντίστοιχες μεθόδους επεξεργασίας, ανάλογα με το είδος του μηνύματος το οποίο και εξυπηρετείται από αφιερωμένη μέθοδο που θα αναλυθεί επακόλουθα.
- Η μέθοδος typeIssue είναι υπεύθυνη για την επεξεργασία μηνυμάτων τύπου "IssueDID" και παράγει ένα TypeIssueResult με πληροφορίες σχετικά με το δημιουργούμενο DID.
- Η μέθοδος typeIssueDIDDOC επεξεργάζεται μηνύματα τύπου "IssueDIDDocument" και δημιουργεί ένα TypeIssueDIDDOCResult που περιέχει το DID document. Ουσιαστικά, το συγκεκριμένο μήνυμα καλείται κάθε φορά όπου αιτείται έκδοση αναγνωριστικού.
- Η μέθοδος typeINV αναλαμβάνει την επεξεργασία μηνυμάτων τύπου "InvokeProxyPipeline" ή "InvokeAccess" και δημιουργεί ένα TypeInvokeResult με πληροφορίες σχετικά με την εκτέλεση τους. Αυτές οι μέθοδοι αφορούν το “ξύπνημα” άλλων μεθόδων, με τη πρώτη να αφορά την εκκίνηση της επανακρυπτογράφησης, ενώ η δεύτερη την εκκίνηση διαδικασιών περί συγκατάθεσης και προβολής δεδομένων.

Τέλος, η κλάση και η σχετική μέθοδος *ResultHandler* είναι υπεύθυνη για την επεξεργασία των αποτελεσμάτων του καταναλωτή και την εκτέλεση των αντίστοιχων ενεργειών βάσει του τύπου του κάθε αποτελέσματος. Αξίζει να σημειωθεί πως ο τρόπος κατά τον οποίο σχηματίστηκε ο παραπάνω κώδικας προσβλέπει σε μια εν γένει κλιμάκωση του εν λόγω συστήματος ούτως ώστε να είναι καθαρογραμμένος, ευνόητος, και ευέλικτος, ενώ στη παρούσα φάση η χρήση του είναι υπεραρκετή στα πλαίσια ενός ελεγχόμενου πειραματικού μοντέλου όπως αυτού της εργασίας.

Εξετάζοντας της μορφή του:

1. Η μέθοδος processResult δέχεται ένα αντικείμενο τύπου ConsumerResult και επεξεργάζεται το αποτέλεσμα ανάλογα με τον τύπο του.
   - Λαμβάνει επιπλέον οποιαδήποτε άλλα ορίσματα που απαιτούνται για την εκτέλεση συγκεκριμένων ενεργειών.

2. Περιπτώσεις Επεξεργασίας:

- Στην περίπτωση TypeIssueResult, εξάγονται οι πληροφορίες σχετικά με το DID, το ιδιωτικό κλειδί και το δημόσιο κλειδί.
- Αναφορικά με τις περιπτώσεις TypeIssueDIDDOCResult, επιστρέφεται το DID document, εξού και η διάκριση μεταξύ **ενός** DID και **πολλών** DID που αναφέρονται σε αυτό.
- Αν το αποτέλεσμα είναι τύπου TypeInvokeResult και το trigger είναι "InvokeProxyPipeline", τότε εκτελείται η μέθοδος ProxyPipeline.proxyJobs για την επεξεργασία των εργασιών που σχετίζονται με το proxy, όπως ακριβώς έχει προαναφερθεί και στην εξήγηση του καταναλωτή.

Ως εκ τούτου, η συγκεκριμένη μέθοδος παρέχει τη δυνατότητα να διαχειριστούμε δυναμικά τα αποτελέσματα του καταναλωτή βάσει του τύπου τους. Όπως αποδείχθηκε, η λειτουργικότητα είναι ευκόλως επεκτάσιμη, καθώς μπορούμε να προσθέσουμε νέες περιπτώσεις επεξεργασίας για να υποστηρίξουμε νέους τύπους αποτελεσμάτων.

5.6 Προκλήσεις

Κατά την ανάπτυξη, προέκυψαν αρκετές βασικές προκλήσεις, υπογραμμίζοντας την πολυπλοκότητα που είναι εγγενής στην κατασκευή ενός τέτοιου μοντέλου.

1. Ενοποίηση διαφορετικών τεχνολογιών: Μία από τις κύριες προκλήσεις ήταν η εναρμόνιση των διαφορετικών τεχνολογιών που αποτελούν το σύστημα. Ο συντονισμός των αλληλεπιδράσεων μεταξύ Cassandra, Kafka, του μηχανισμού επανακρυπτογράφησης μεσολάβησης και της διεπαφής τερματικού απαιτούσε σχολαστική σχεδίαση και προσπάθειες ενσωμάτωσης, με χρήσης πλήθους βιβλιοθηκών που εντοπίζονται στη γλώσσα Scala. Η διασφάλιση απρόσκοπτης επικοινωνίας και ροής δεδομένων μεταξύ αυτών των στοιχείων απαιτούσε την αντιμετώπιση προβλημάτων συμβατότητας, μετατροπών μορφών δεδομένων και εμποδίων συγχρονισμού.
2. Ανησυχίες για την ασφάλεια και το απόρρητο: Η εξισορρόπηση της ισχυρής ασφάλειας με τη φιλική προς τον χρήστη λειτουργικότητα ήταν μια συνεχής πρόκληση. Η εφαρμογή ισχυρών κρυπτογραφικών μηχανισμών, όπως η εκ νέου κρυπτογράφηση μεσολάβησης, με παράλληλη διατήρηση της βέλτιστης απόδοσης του συστήματος απαιτούσε προσεκτική εξέταση και συμβιβασμούς. Αυτό φυσικά ισχύει αν λάβει κανείς υπόψη πως η εφαρμογή είναι σχεδιασμένη να λειτουργεί με μηνύματα συμβάντων, οπότε η σύζευξη αυτών με κατάτμηση βάσεων είναι ένα μείζον ζήτημα τόσο υλοποίησης, όσο και απόδοσης.
3. Σχεδιασμός διεπαφής χρήστη: Ο σχεδιασμός μιας διεπαφής που βασίζεται σε τερματικό που επικοινωνεί αποτελεσματικά σύνθετες έννοιες, ενώ παραμένει διαισθητικός για τους χρήστες, αποδείχθηκε πρόκληση. Η εξισορρόπηση της σαφήνειας με την ολοκληρωμένη λειτουργικότητα

απαιτούσε προσεκτική εξέταση των ροών εργασίας των χρηστών, των πιθανών σεναρίων σφαλμάτων και των περιορισμών ενός περιβάλλοντος που βασίζεται σε κείμενο.

4. Δοκιμή και επικύρωση: Ο ενδελεχής έλεγχος της λειτουργικότητας, της ασφάλειας και της απόδοσης του συστήματος αποδείχθηκε πολύπλοκο εγχείρημα. Η ανάπτυξη ρεαλιστικού σεναρίου δοκιμής συνάμα της προσομοίωση φορτίων του πραγματικού κόσμου, δηλαδή ταυτόχρονες εκδόσεις αναγνωριστικών, επανακρυπτογραφήσεις και διασφάλιση της εγκυρότητας των αποτελεσμάτων υπό διάφορες συνθήκες δε παύει να είναι ένα ακανθώδες θέμα που δεν επιλύεται απλά εντός του περιορισμένου πλαισίου της συγκεκριμένης διπλωματικής εργασίας.

6

Σχεδιασμός της Διεπαφής Χρήστη

6.1 Σκέψεις για την Εμπειρία Χρήστη

Η διεπαφή χρήστη, η οποία για την συγκεκριμένη εργασία αξιοποιεί τερματικό, εξυπηρετεί πολλαπλούς κρίσιμους σκοπούς εντός του έργου. Κατά πρώτον, παρέχει τα πρακτικά μέσα στους χρήστες να αλληλεπιδρούν με τις βασικές λειτουργίες του συστήματος: δημιουργία DID, συσχέτιση δεδομένων, διαχείριση συναίνεσης (χορήγηση και ανάκληση αδειών) και ελεγχόμενη πρόσβαση σε δεδομένα. Δεύτερον, η σκόπιμη επιλογή ενός περιβάλλοντος γραμμής εντολών δίνει προτεραιότητα στη σαφήνεια έναντι της οπτικής πολυπλοκότητας, εστιάζοντας την προσοχή στους ουσιαστικούς μετασχηματισμούς και τις αλλαγές στη δυναμική ισχύος που είναι εγγενείς σε ένα πραγματικά αποκεντρωμένο σύστημα. Κατά αυτό το τρόπο η εστίαση έγκειται επί της ουσίας στο τί και πως γίνεται ούτως ώστε να αποσαφηνιστεί πλήρως η αναπαράσταση των λειτουργιών. Τρίτον, η διεπαφή καθιστά απτές τις συχνά αφηρημένες έννοιες της αποκεντρωμένης ταυτότητας μεταφράζοντας τις σε μια σαφή ακολουθία αλληλεπιδράσεων που καθοδηγούνται από τον χρήστη. Η εμφάνιση δεδομένων χρησιμοποιεί απλές μεθόδους τύπωσης στην οθόνη, διατηρώντας τη σαφήνεια εντός των περιορισμών ενός περιβάλλοντος απλού κειμένου. Επίσης, ο χειρισμός σφαλμάτων υλοποιείται σε στοιχειώδη μορφή για να διασφαλίσει ότι η είσοδος του χρήστη ευθυγραμμίζεται με τις αναμενόμενες ενέργειες και για να καθοδηγήσει τους χρήστες προς έγκυρες ενέργειες εντός του συστήματος. Ενώ τα ισχυρά μέτρα ασφαλείας είναι πέρα από το πεδίο εφαρμογής αυτού του πρωτοτύπου (κάτι που ούτως ή άλλως άπτεται σε μεταγενέστερο στάδιο τέτοιων έργων), η διεπαφή περιλαμβάνει σκόπιμα προτροπές και στοιχεία κειμένου που υπογραμμίζουν την κρίσιμη σημασία του ελέγχου ταυτότητας και της κρυπτογράφησης ή παραπλήσιων αυτής ενεργειών. Κατά αυτό το τρόπο, οι χρήστες δέχονται μια πρώτη γεύση για το πως λειτουργεί πραγματικά το εν λόγω σύστημα εν σχέσει με όλα τα στάδια, δηλαδή την έκδοση αναγνωριστικών, είσοδο στο σύστημα, αποκρυπτογραφήσεις, συγκαταθέσεις, και προβολή διαμοιρασμένων αρχείων.

Βασικοί Στόχοι και Σκεπτικό Σχεδιασμού:

- *Λειτουργική επίδειξη*: Το πρωτότυπο στοχεύει να παρουσιάσει τους βασικούς μηχανισμούς ενός αποκεντρωμένου συστήματος ταυτότητας. Οι χρήστες πρέπει να μπορούν να δημιουργούν DID, να συσχετίζουν επιλεγμένα στοιχεία δεδομένων, να διαχειρίζονται λεπτομερώς τη συναίνεση για πρόσβαση σε δεδομένα και να παρατηρούν τα αποτελέσματα αυτών των ενεργειών.
- *Εικονογράφηση της αποκέντρωσης*: Η διεπαφή αποφεύγει ενεργά τη συσκότιση των αρχών της αποκέντρωσης, με αυτό το χαρακτηριστικό να αναδεικνύεται πλήρως κατά την εκκίνηση πολλαπλών τερματικών διαφορετικών χρηστών που όμως συμμετέχουν στον ίδιο δακτύλιο βάσης δεδομένων, όπου και κάθε τους ενέργεια έχει το ίδιο αντίκτυπο που συγχρονίζεται σε αυτή. Η επιλογή και ο έλεγχος των χρηστών είναι πρωταρχικής σημασίας – οι ασθενείς επιλέγουν ενεργά ποια δεδομένα θα μοιραστούν και οι πάροχοι υγειονομικής περίθαλψης πρέπει να ζητήσουν ρητά πρόσβαση. Αυτό έρχεται σε έντονη αντίθεση με τα παραδοσιακά μοντέλα δεδομένων υγειονομικής περίθαλψης, υπογραμμίζοντας εμφανώς τη μετατόπιση της δυναμικής ισχύος προς τη μεγαλύτερη αντιπροσωπεία των ασθενών.
- *Προσβασιμότητα για αξιολόγηση*: Η σαφήνεια μιας διεπαφής απλού κειμένου βοηθά στη σχολαστική αξιολόγηση, επιτρέποντας την εστίαση στη λογική του συστήματος και όχι στην ίδια την οπτική διεπαφή. Επιτρέπει επίσης πιο προσιτή τεκμηρίωση και ανάλυση, καθώς οι βασικές ενέργειες και οι ροές δεδομένων είναι εύκολα ορατές.
- *Εκπαιδευτικό εργαλείο*: Το πρωτότυπο τοποθετείται όχι μόνο ως τεχνική επίδειξη αλλά ως εργαλείο μάθησης, παρέχοντας ένα συγκεκριμένο πλαίσιο για την κατανόηση των αρχών και των επιπτώσεων της αποκεντρωμένης διαχείρισης ταυτότητας στον τομέα της υγειονομικής περίθαλψης και όχι μόνον. Αυτή η προσέγγιση επιτρέπει την ενημερωμένη συζήτηση σχετικά με τον πιθανό μετασχηματιστικό αντίκτυπο ενός τέτοιου συστήματος.

Πλαισιώνοντας το Πρωτότυπο:

Είναι σημαντικό να αναγνωριστεί ότι αυτό το πρωτότυπο αποδίδει σκόπιμα προτεραιότητα στη χρηστικότητα για την επίδειξη βασικών εννοιών σε αντίθεση με στοιχεία όπως η εξαιρετικά εκλεπτυσμένη οπτική σχεδίαση (Γραφικό Περιβάλλον) ή τα πλήρως εφαρμοσμένα μέτρα ασφαλείας. Μια επιτυχημένη επίδειξη θα επικοινωνήσει με σαφήνεια αυτούς τους περιορισμούς ενώ ταυτόχρονα θα ανοίξει συζήτηση σχετικά με πιθανές μελλοντικές επαναλήψεις, προηγμένες υλοποιήσεις ασφάλειας και πλουσιότερες μορφές αλληλεπίδρασης με τον χρήστη κατάλληλες για ανάπτυξη σε πραγματικό κόσμο.

6.2 Λεπτομέρειες Υλοποίησης

Προκειμένου να επιτευχθεί σχεδόν σε πραγματικό χρόνο συγχρονισμός μεταξύ των κατά προσομοίωση γεωγραφικά διαχωρισμένων τερματικών, πολλά βασικά χαρακτηριστικά πρέπει να ενσωματωθούν προσεκτικά στη σχεδίαση της διεπαφής. Το πρώτο βήμα περιλαμβάνει έναν ασφαλή μηχανισμό σύνδεσης, άμεσα ενσωματωμένο στο σύστημα αποκεντρωμένης ταυτότητας (DID) του έργου. Αυτό το βήμα ελέγχου ταυτότητας όχι μόνο επικυρώνει την ταυτότητα του χρήστη, αλλά ανακτά επίσης το μοναδικό DID του και τυχόν σχετικά μεταδεδομένα, συμπεριλαμβανομένων των καθιερωμένων αδειών πρόσβασης για προσωπικές πληροφορίες. Μετά την επιτυχή σύνδεση, η διεπαφή καθορίζει αυτόματα με ποιον κόμβο στο σύμπλεγμα Cassandra θα αλληλεπιδράσει κυρίως ο χρήστης. Αξίζει να σημειωθεί πως αυτή η αντιστοίχιση μπορεί να είναι προκαθορισμένη με βάση τους ρόλους των χρηστών και μια στρατηγική κατάτμησης ή θα μπορούσε να εκχωρηθεί δυναμικά με βάση παράγοντες όπως η διαθεσιμότητα κόμβων και οι μηχανισμοί εξισορρόπησης φορτίου συστήματος. Μια κρίσιμη πτυχή του σχεδιασμού της διεπαφής είναι ο μηχανισμός που διασφαλίζει την επίγνωση σε πραγματικό χρόνο (ή σχεδόν σε πραγματικό χρόνο) των σχετικών αλλαγών - κυρίως συγκαταθέσεων και προβολής από τρίτους - σε όλο το αποκεντρωμένο σύστημα. Οι ενέργειες χρήστη, όπως η προβολή αρχείων υγείας, η χορήγηση ή η ανάκληση αδειών σε άλλους χρήστες ή οι τροποποιήσεις των δικών τους δεδομένων που σχετίζονται με το DID, μεταφράζονται από τη διεπαφή σε αντίστοιχες κλήσεις προγραμματιστικής διεπαφής εφαρμογή, γνωστής και ως *Application Programming Interface (*API), ή ερωτήματα βάσης δεδομένων. Αυτές οι ενέργειες ενεργοποιούν αλλαγές που διαδίδονται σε όλο το σύμπλεγμα Cassandra, επηρεάζοντας τελικά τα υπόλοιπα συνδεδεμένα τερματικά. Ως εκ τούτων, η κύρια πρόκληση έγκειται στη προδήλως εμφανή αντανάκλαση αυτών των δυναμικών ενημερώσεων, διατηρώντας παράλληλα μια απρόσκοπτη και ξεκάθαρη εμπειρία χρήστη. Τέλος, μία δυνητική διαχείριση προσδοκιών των χρηστών σε ένα ιδιαίτερα εκτεταμένο κατανεμημένο σύστημα, η διεπαφή θα μπορούσε να χρησιμοποιήσει οπτικές ενδείξεις ή δείκτες κειμένου (όπως "Ενημέρωση..." ή "Σε εκκρεμότητα...") για να σηματοδοτήσει ότι ο συγχρονισμός μπορεί να διαρκέσει μερικά λεπτά, ενισχύοντας τη κατανόηση της συμπεριφοράς του συστήματος.

7

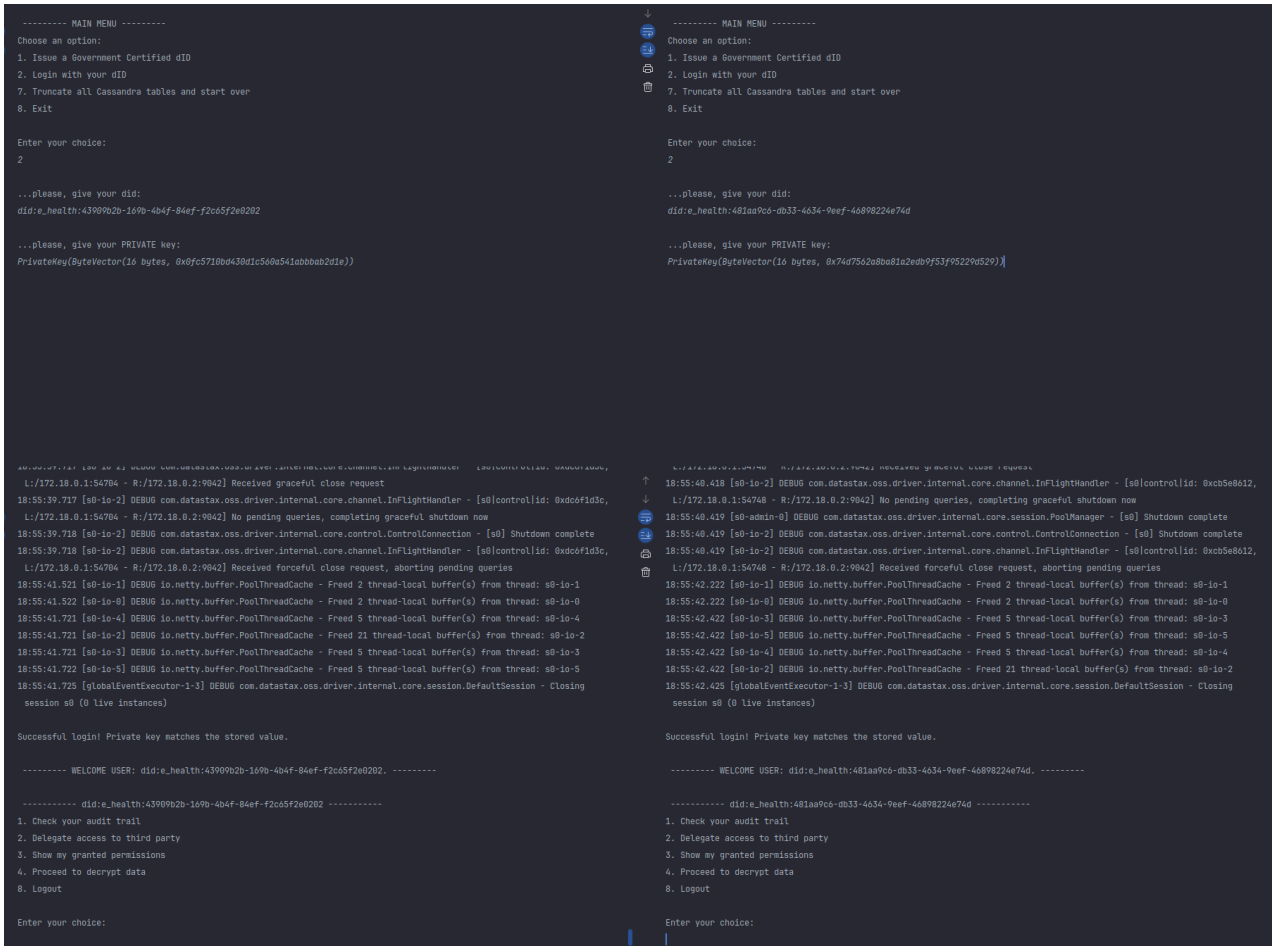
Μελέτη Περίπτωσης: Υπηρεσίες Υγείας

7.1 Σενάριο Χρήσης

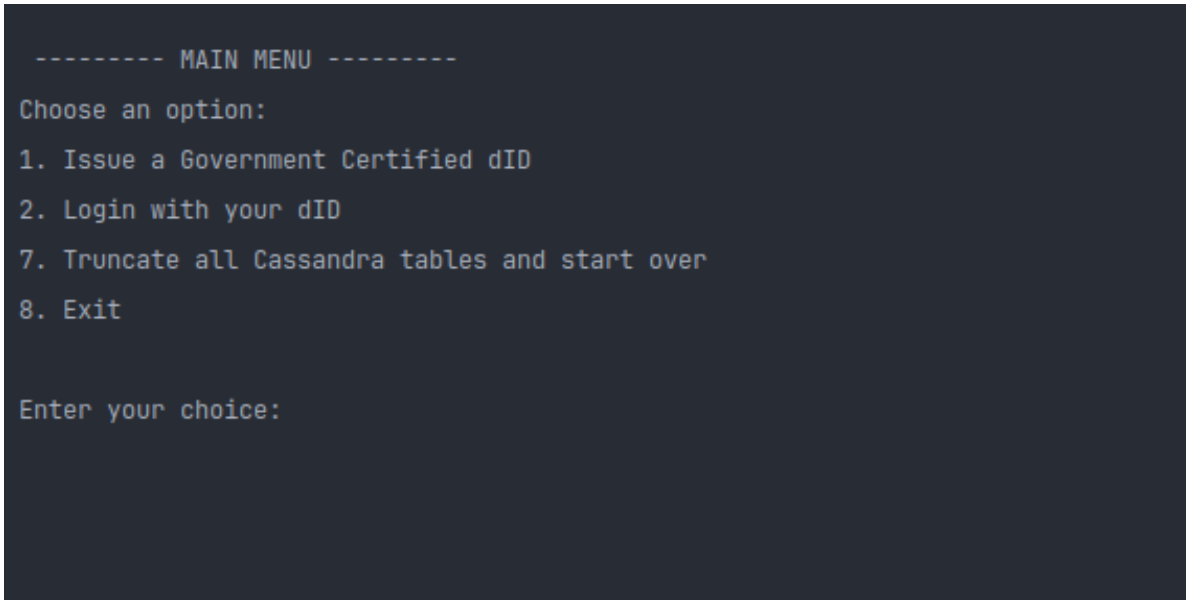
Το σενάριο αφορά τον ασθενή ως πρωταγωνιστή και το τρίτο μέρος ως δευτερεύων ρόλο (όπως ένας γιατρός) να διαθέτουν καθιερωμένα DID και απαραίτητα διαπιστευτήρια πρόσβασης στο αποκεντρωμένο σύστημα ταυτότητας. Η διεπαφή δίνει προτεραιότητα στη διαφάνεια, εμφανίζοντας ευδιάκριτα τόσο τις συγκεκριμένες κατηγορίες δεδομένων που ζητήθηκαν όσο και την ταυτότητα του τρίτου μέρους που ζητά. Το σύστημα πριμοδοτεί τον ασθενή, παρέχοντάς του διαισθητικές και εύκολα προσβάσιμες επιλογές για έγκριση, απόρριψη ή ακόμη και άμεση προβολή σχετιζόμενων με αυτόν στοιχείων. Αυτό υπογραμμίζει τη στροφή προς τα ελεγχόμενα από τον ασθενή μοντέλα κοινής χρήσης δεδομένων που ενεργοποιούνται από αποκεντρωμένα συστήματα ταυτότητας. Μετά την έγκριση του ασθενούς, η μονάδα μεσολάβησης λειτουργεί ως προστατευτικός ενδιάμεσος. Ανακτά τα σχετικά κρυπτογραφημένα δεδομένα υγείας και, μέσω κρυπτογραφικών τεχνικών, τα κρυπτογραφεί εκ νέου. Αυτή η εκ νέου κρυπτογράφηση βασίζεται σε κλειδιά και λεπτομερείς άδειες, διασφαλίζοντας ότι μόνο το σύστημα του τρίτου μπορεί να αποκρυπτογραφήσει τα δεδομένα, διατηρώντας το απόρρητο ακόμη και κατά τη μετάδοση. Επιπλέον, το σύστημα διατηρεί ένα διαφανές και αμετάβλητο αρχείο καταγραφής αυτού του συμβάντος πρόσβασης. Το αρχείο καταγραφής, με δυνατότητα προβολής από τον ασθενή, αναφέρει λεπτομερώς τα συγκεκριμένα στοιχεία δεδομένων που μοιράζονται, σφραγίζει χρονικά την πρόσβαση και επιβεβαιώνει την ταυτότητα του τρίτου μέρους που έχει πρόσβαση. Αυτή η δυνατότητα ελέγχου ενισχύει την εμπιστοσύνη στο σύστημα παρέχοντας συγκεκριμένη λογοδοσία. Κατά αυτό τον τρόπο αναφύεται ένας άρρηκτος συνδετικός κρίκος μεταξύ ασθενών και τρίτων - και όχι τρίτων ερήμην ασθενών - κάτι το οποίο διαφαίνεται προδήλως στα παρακάτω στιγμιότυπα που ακολουθούν. Κάθε κίνηση είναι ελεγχόμενη, ενώ η απρόσκοπτη και ακατάσχετη λειτουργία των αποκεντρωμένων κόμβων προσδίδουν στην αξιοπιστία, τη διαθεσιμότητα, και φυσικά την πολυδιάστατη λειτουργία χωρίς κεντρικούς τοποτηρητές.

Σύμφωνα με τα παρακάτω στιγμιότυπα, το απλό αλλά πρακτικό παράδειγμα της υλοποίησης του έργου κάνει εμφανή το τρόπο λειτουργίας και της φιλοσοφίας του συστήματος. Τα βήματα είναι συγκεκριμένα, με τους ασθενής και τρίτους να είναι δέσμιοι ενός αυστηρού πρωτοκόλλου που δεν αφήνει περιθώριο για παρεκκλίνουσες και εν κρυπτώ δραστηριότητες. Το αρχείο που χρησιμοποιήθηκε από τη πλευρά του ασθενούς είναι σε μορφή *pdf* και τοποθετείται χειροκίνητα στο πρόγραμμα ούτως ώστε να εξυπηρετήσει τους σκοπούς της πρωτότυπης αναπαράστασης. Ωστόσο όμως, έχει προαναφερθεί πως η χρήση του ιατρικού κύκλου στη συγκεκριμένη εργασία αποτελεί απλώς υποκείμενο αναφοράς, καθώς το εν λόγω σύστημα είναι εκ φύσεως κατασκευασμένο για να λειτουργεί και να συνδέει πολλούς κλάδους, χώρους και συστήματα έτσι ώστε να συγκλίνουν σε ένα ενιαίο σύστημα όπου ο κάθε ένας εξ αυτών διατηρεί μεν την ακεραιότητά του ως μοναδική οντότητα, εκπονεί δε την εξωστρεφή συμπεριφορά με τη μετοχή του.

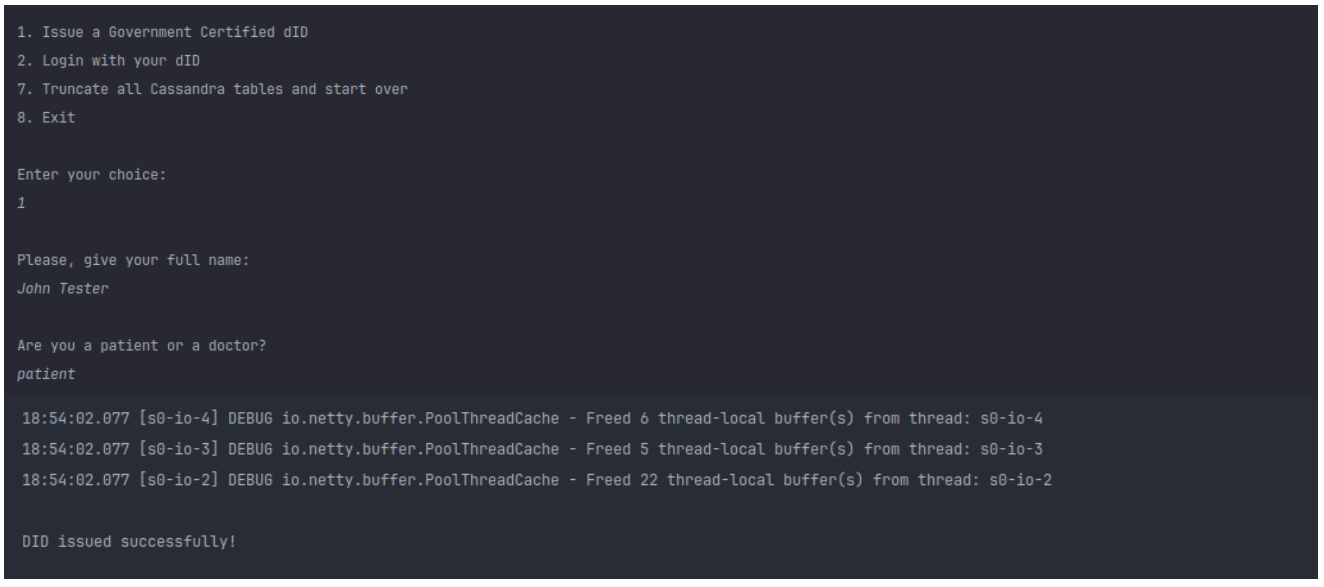
7.2 Εκτέλεση



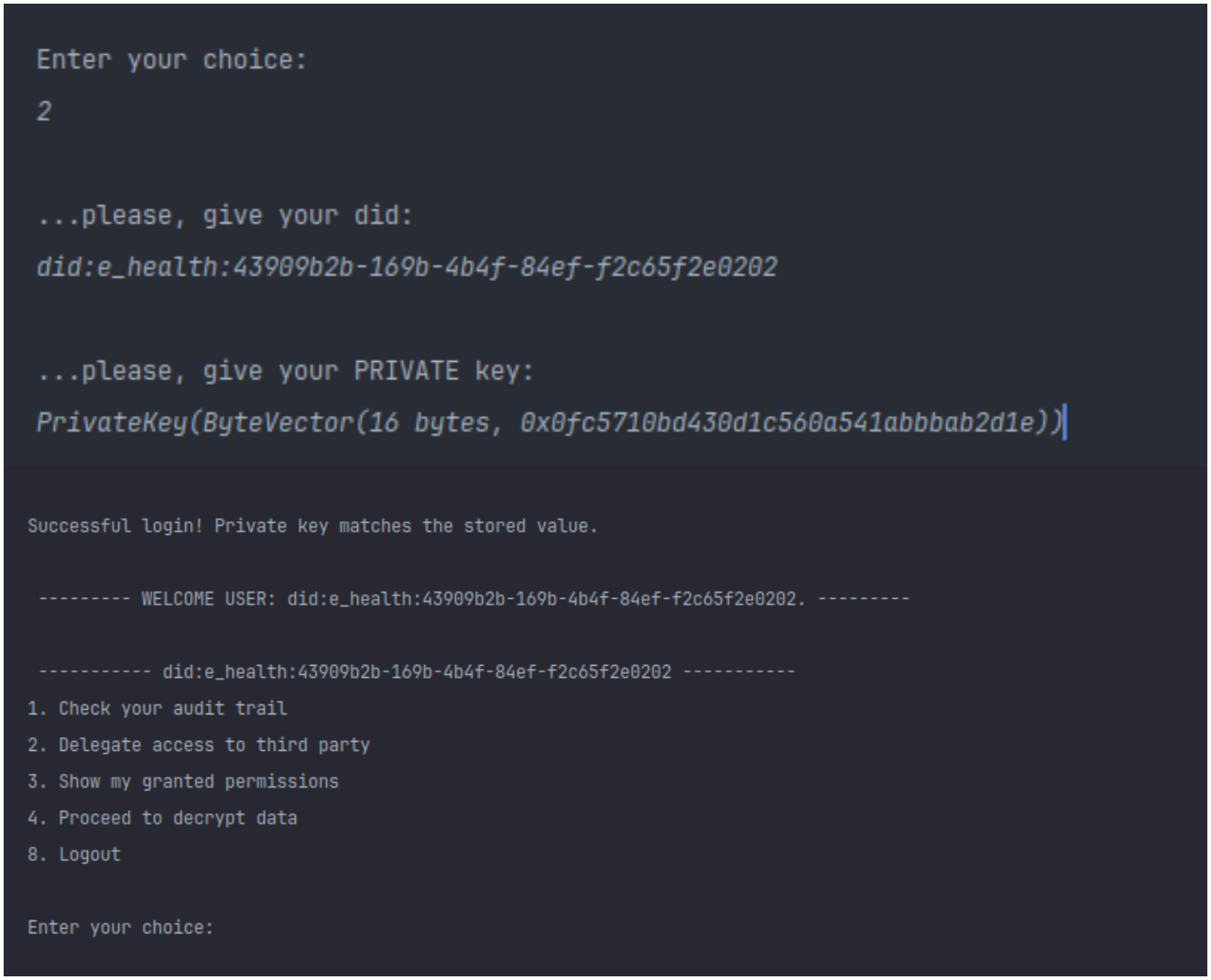
Εικόνα 20: Παράλληλη εκτέλεση Ασθενή και Τρίτου



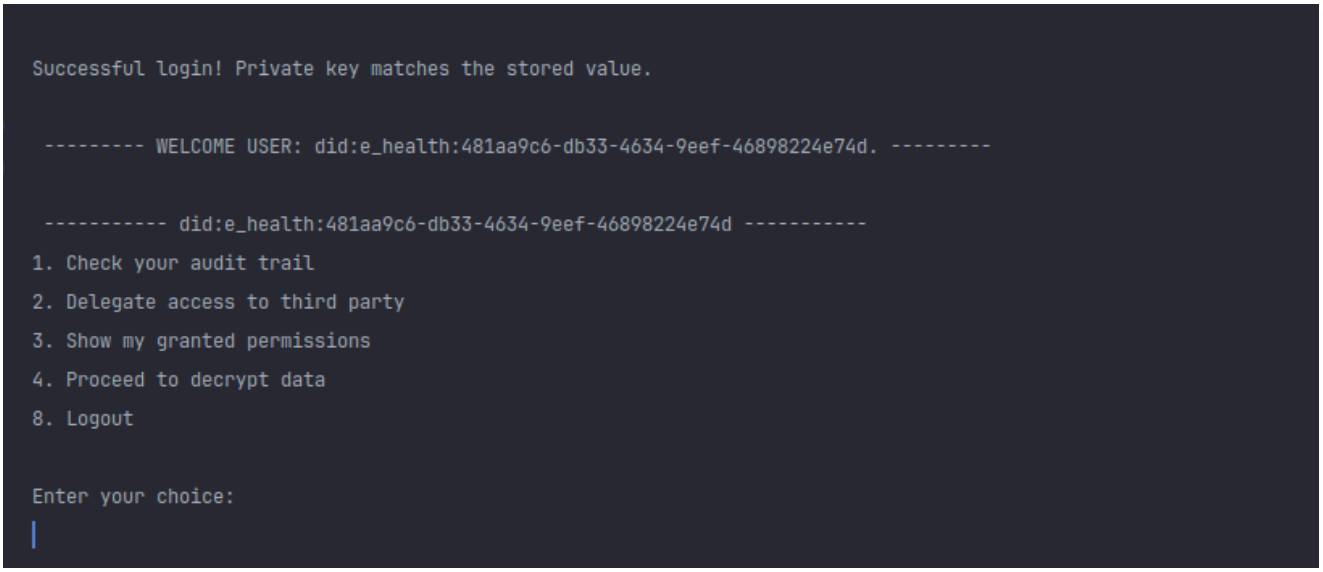
Εικόνα 21: Εισαγωγική οθόνη



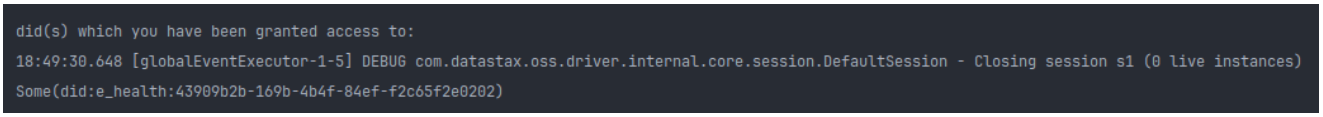
Εικόνα 22: Έκδοση Αναγνωριστικού



Εικόνα 23: Ο Ασθενής εισέρχεται στο σύστημα



Εικόνα 24: Ο Τρίτος (Ιατρός/Φορέας) εισέρχεται στο σύστημα



Εικόνα 25: Ο Τρίτος προβάλλει τις τρέχουσες συγκαταθέσεις του

```
Audit Trail for did: did:e_health:43909b2b-169b-4b4f-84ef-f2c65f2e0202
18:46:09.188 [globalEventExecutor-1-5] DEBUG com.datastax.oss.driver.internal.core.session.DefaultSession - Closing session s1 (0 live instances)

------------------------------------------------------------
Action: READ
Timestamp: 2024-03-26T20:21:27.252Z
Source did: did:e_health:481aa9c6-db33-4634-9eef-46898224e74d
Target did: did:e_health:43909b2b-169b-4b4f-84ef-f2c65f2e0202
------------------------------------------------------------
Action: READ
Timestamp: 2024-03-26T20:28:43.887Z
Source did: did:e_health:481aa9c6-db33-4634-9eef-46898224e74d
Target did: did:e_health:43909b2b-169b-4b4f-84ef-f2c65f2e0202
```

Εικόνα 26: Ο Ασθενής προβάλλει τις τελευταίες κινήσεις τρίτων επί των δεδομένων του

```
Enter the DID you wish to delegate access to:
18:46:58.460 [globalEventExecutor-1-7] DEBUG com.datastax.oss.driver.internal.core.session.DefaultSession - Closing session s2 (0 live instances)
did:e_health:481aa9c6-db33-4634-9eef-46898224e74d

DID document was created successfully.
18:48:47.095 [globalEventExecutor-1-19] DEBUG com.datastax.oss.driver.internal.core.session.DefaultSession - Closing session s8 (0 live instances)

Access to user did:e_health:481aa9c6-db33-4634-9eef-46898224e74d from user did:e_health:43909b2b-169b-4b4f-84ef-f2c65f2e0202 granted successfully.

Proxy initiated successfully.
```

Εικόνα 27: Ο Ασθενής παραδίδει εξουσιοδότηση και τελική συγκατάθεση σε Τρίτο

Εικόνα 28: Ο Τρίτος αποκρυπτογραφεί και έχει πρόσβαση στα σχετικά δεδομένα

```
case "2" ⇒
  val pdfFilePath: String = "/home/nikolai/Documents/M.Sc./Διπλωματική/"
  val pdfFileName: String = "PISTOPOIHTIKO-YGEIAS.pdf"

  val pdfFile: File = new File(pdfFilePath, pdfFileName)
  val pdfByteArray: Array[Byte] = new FileInputStream(pdfFile).readAllBytes()

  val message: Plaintext = coreApi.generatePlaintext.unsafeRunSync
  val symmetricKey: Array[Byte] = coreApi.deriveSymmetricKey(message).bytes.toArray

  val symEncrypted: Array[Byte] = Symmetric.symEncrypt(pdfByteArray, symmetricKey).unsafeRunSync().toConcatenated

  DIDRegistry.addEncryptedData(currentDID, symEncrypted, pdfFileName)

  println("\nEnter the DID you wish to delegate access to: ")
  val target: String = scala.io.StdIn.readLine()
```

Εικόνα 29: Απόσπασμα κώδικα ψηφιοποίησης ιατρικού εγγράφου

ΑΔΑ: Β4Λ0Θ-Ο5Υ

ΠΙΣΤΟΠΟΙΗΤΙΚΟ ΥΓΕΙΑΣ
(εργαζομένων σε καταστήματα, εργαστήρια ή επιχειρήσεις υγειονομικού ενδιαφέροντος)

Ο/Η Επώνυμο……
Όνομα……
Όνομα Πατρός……
Α.Δ.Τ ……
ή Α.Μ.Κ.Α……
ή Αριθμός Διαβατηρίου……
Δ/νση.Κατοικίας ……
Υποβλήθηκε σε Ιατρική κλινική εξέταση και στις παρακλινικές εξετάσεις

Ακτινογραφία Θώρακος: ☐ Ναι

Καλλιέργεια *, Παρασιτολογική κοπράνων (για χειριστές τροφίμων * *) ☐ Ναι ☐ Οχι

Βρέθηκε να μην πάσχει από κάποιο λοιμώδες ή μεταδοτικό νόσημα για τη Δημόσια Υγεία και μπορεί να εργαστεί σε κατάστημα, εργαστήριο ή επιχείρηση υγειονομικού ενδιαφέροντος.
Το παρόν ισχύει για πέντε (5) χρόνια, πλην των εργαζομένων σε παιδικούς, βρεφονηπιακούς σταθμούς κ.λπ. που ισχύει για 2 χρόνια ***

ΗΜΕΡΟΜΗΝΙΑ ……/……/20……

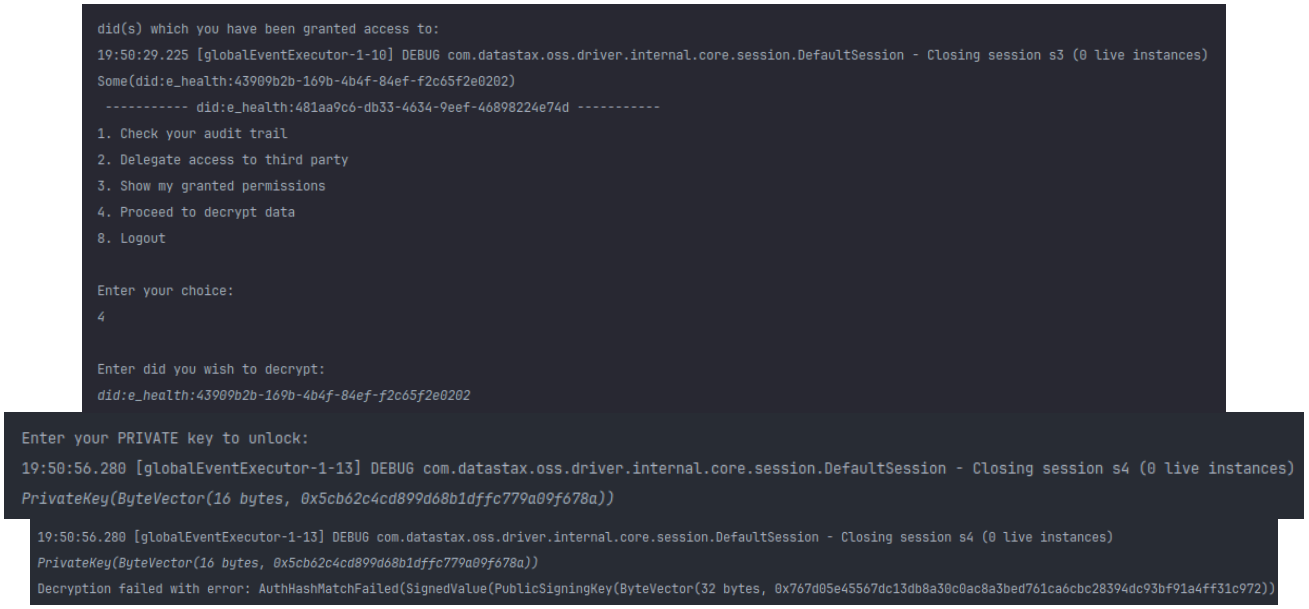
Ο ΙΑΤΡΟΣ
(υπογραφή, σφραγίδα)

*Για salmonella .
** (Χειριστής τροφίμων είναι το άτομο που απευθείας χειρίζεται συσκευασμένα ή μη συσκευασμένα τρόφιμα, εξοπλισμό και εργαλεία για τρόφιμα ή επιφάνειες που έρχονται σε επαφή με τα τρόφιμα).
*** Τα Πιστοποιητικά υγείας του προσωπικού θα φυλάσσονται στις Επιχειρήσεις υγειονομικού ενδιαφέροντος και θα επιδεικνύονται όταν ζητούνται, στα αρμόδια όργανα υγειονομικού ελέγχου. Θα ανανεώνονται μετά τη λήξη τους επαναλαμβάνοντας όλες τις εξετάσεις (Ιατρική κλινική εξέταση

Εικόνα 30: Για λόγους απλότητος, το αρχείο που χρησιμοποιείται στις λειτουργίες της εργασίας είναι το παρόν, με μετατροπή αυτού από PDF σε ByteArray, και τελική είσοδο - ή έξοδο όταν πραγματοποιείται αποκρυπτογράφηση - στη βάση δεδομένων

7.2.1 Μέτρα Ασφαλείας

Εικόνα 31: Ο γενικός χρήστης δέχεται **απόρριψη** εισόδου λόγω λανθάνοντος ιδιωτικού κλειδιού.



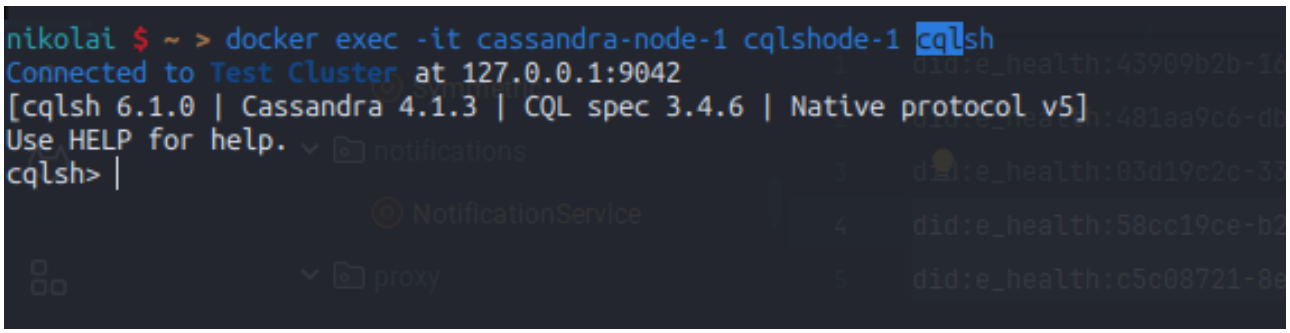
Εικόνα 32: Ο Τρίτος δέχεται **απόρριψη** αποκρυπτογραφησης λόγω λανθάνοντος ιδιωτικού κλειδιού.

8

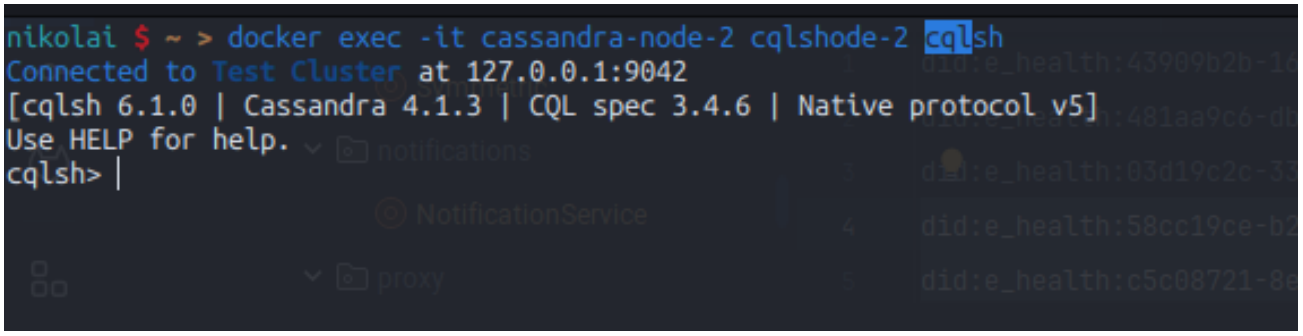
Αποτελέσματα & Ανάλυση

8.1 Αξιολόγηση Απόδοσης

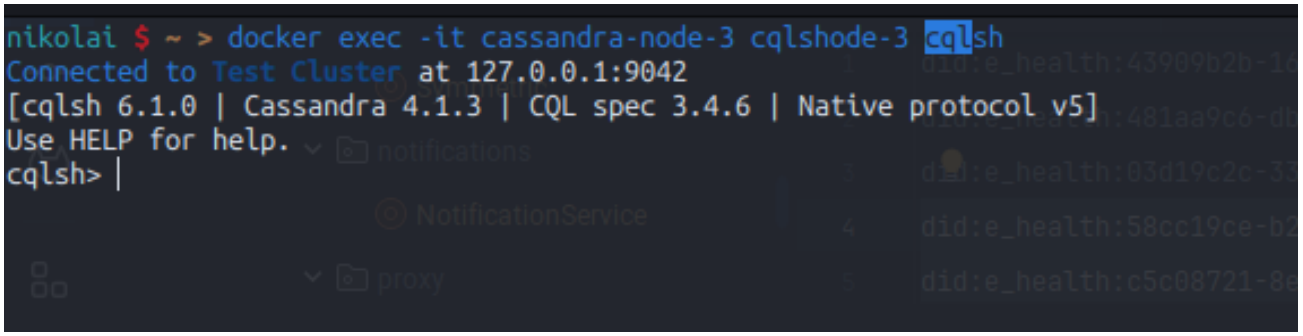
Βασικός πυλώνας αξιολόγησης της απόδοσης για τη παρούσα φάση είναι εφόσον και αν οι αποκεντρωμένοι κόμβοι υπακούν στο δόγμα της διαθεσιμότητας. Βάσει της αρχιτεκτονικής της Cassandra, εφόσον ένας κόμβος διακοπεί ακαριαία, τότε ο δακτύλιος αντιλαμβάνεται το σφάλμα και συνεχίζει την αντιγραφή των δεδομένων στους υπόλοιπους μετέχοντες κόμβους. Παράλληλα με αυτό, τοποθετεί μια ετικέτα εν σχέσει με τον εσφαλμένο κόμβο δηλώνοντάς τον ως “Εκτός”, με την ετικέτα να παραμένει έως ότου ο κόμβος επανέλθει αισίως σε λειτουργία. Θεωρητικά, με το πέρας του προβλήματος και την επανένταξή του στο δακτύλιο τα δεδομένα που αγνόησε θα του προστεθούν εμβόλιμα. Ως εκ τούτου, στα παρακάτω στιγμιότυπα διαφαίνεται η εκκίνηση επανακρυπτογράφησης με το κόμβο νούμερο δύο να τίθεται εκουσίως εκτός, ενώ στο τέλος επιτυγχάνεται η επανένταξη αυτού με τη πλήρη αντιγραφή των χαμένων δεδομένων.



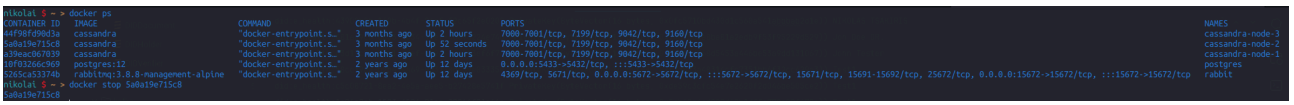
Εικόνα 33: 1ος Κόμβος



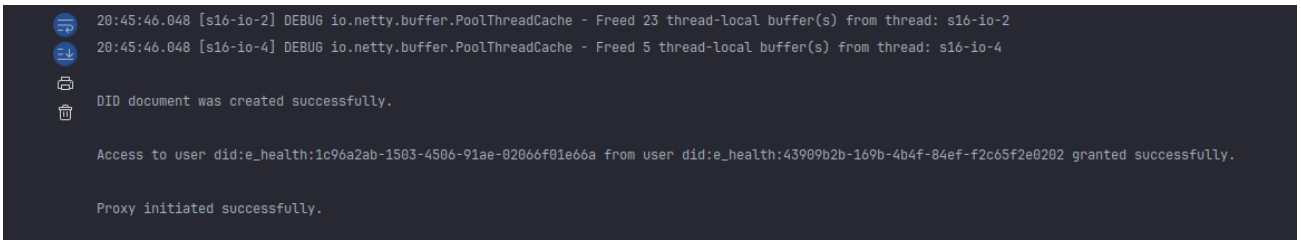
Εικόνα 34: 2ος Κόμβος



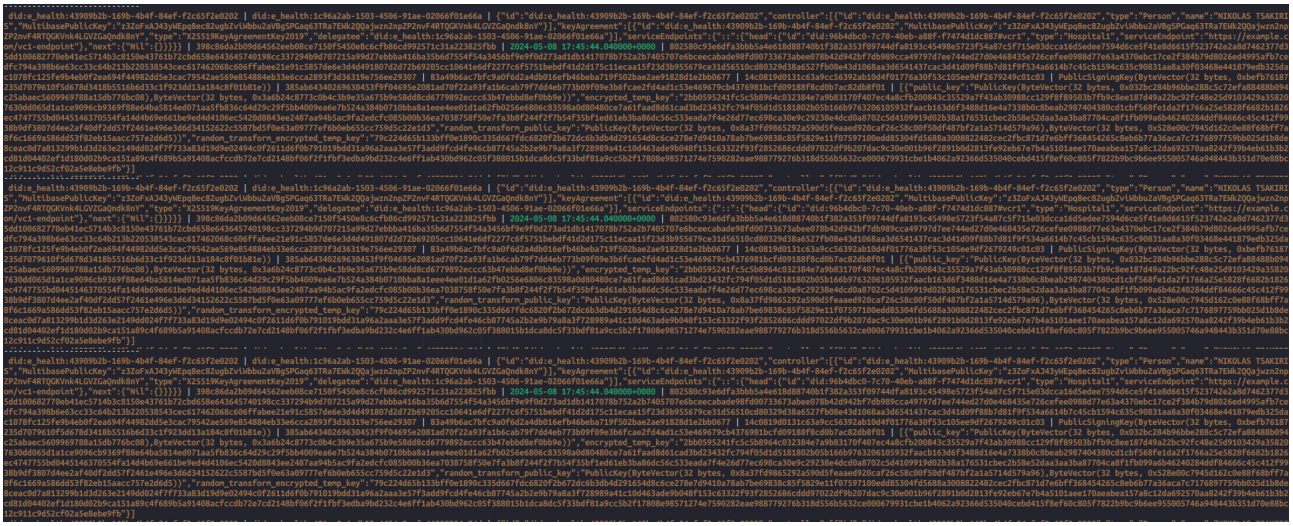
Εικόνα 35: 3ος Κόμβος



Εικόνα 36: Εκούσιος Τερματισμός δευτέρου κόμβου



Εικόνα 37: Επανακρυπτογράφηση



Εικόνα 38: Οι τρεις κόμβοι ταυτίζονται, με τη πράσινη χρονοσφραγίδα να συγκλίνει στην ίδια καταγραφή.

8.2 Αξιολόγηση Ασφαλείας

Ως κεντρικός αξιολογητής ασφαλείας είναι τα περί κρυπτογράφησης, επανακρυπτογράφησης και τα συν αυτώ, τουλάχιστον στην έκδοση και μορφή που η συγκεκριμένη χώρα υλοποιήθηκε μέχρι στιγμής. Το έργο αξιοποιεί τη βιβλιοθήκη επανακρυπτογράφησης μεσολάβησης του οργανισμού IronCoreLabs για ισχυρή εφαρμογή στο αποκεντρωμένο σύστημα ταυτότητάς. Αυτή η βιβλιοθήκη έχει υποβληθεί σε αυστηρό έλεγχο ασφαλείας από την NCC Group, μια κορυφαία κρυπτογραφική αρχή. Η ανασκόπηση [92] περιλάμβανε και τα δύο, ήτοι την καταλληλότητα της επιλεγμένης ελλειπτικής καμπύλης φιλικής προς το ζευγάρωμα και την ασφαλή εφαρμογή του πρωτοκόλλου εντός της βιβλιοθήκης. Κυρίως, ο έλεγχος εντόπισε πιθανές ευπάθειες σε προηγούμενες εκδόσεις της βιβλιοθήκης, που σχετίζονται ειδικά με την εξαγωγή κλειδιού και την επικύρωση σημείου καμπύλης. Αυτό ώθησε το έργο να υιοθετήσει προληπτικά ενημερωμένες εκδόσεις όπου αντιμετωπίζονταν αυτά τα ζητήματα. Επιπλέον, ακολουθώντας τις συστάσεις του NCC Group, ενσωματώθηκε μια καμπύλη Barreto-Naehrig, διασφαλίζοντας ασφάλεια 128-bit σε ευθυγράμμιση με τις τρέχουσες βέλτιστες πρακτικές κρυπτογράφησης. Ενώ η αναφορά της ομάδας NCC πρότεινε περαιτέρω βελτιστοποιήσεις απόδοσης (όπως προσαρμοσμένες εφαρμογές λειτουργίας πεδίου), η κύρια εστίαση ήταν στον καθορισμό της ορθότητας και της βασικής ασφάλειας της διαδικασίας επανακρυπτογράφησης του διακομιστή μεσολάβησης. Έτσι, η ολοκληρωμένη αναθεώρηση, μαζί με την υιοθέτηση των προτεινόμενων επιδιορθώσεων, ενισχύει την αποτελεσματικότητα της ασφάλειας του , μειώνει τον κίνδυνο εκμεταλλεύσιμων τρωτών σημείων και προστατεύει ευαίσθητα δεδομένα κατά την επανακρυπτογράφηση. Η ενσωμάτωση αυτής της ελεγμένης βιβλιοθήκης δείχνει μια δέσμευση στις

αρχές του απορρήτου των χρηστών και της ακεραιότητας των δεδομένων στο πλαίσιο του αποκεντρωμένου μοντέλου ταυτότητας.

8.3 Περί Σχετικών Εργασιών

Αυτή η εργασία συμβάλλει ουσιαστικά στον τομέα της αποκεντρωμένης διαχείρισης ταυτότητας σχεδιάζοντας, εφαρμόζοντας και αξιολογώντας αυστηρά ένα πρωτότυπο εθνικό σύστημα DID, ξεχωρίζοντας από προηγούμενες εργασίες όπως [93], [94], [95] και [96] . Σε αντίθεση με αυτές τις μελέτες, οι οποίες επικεντρώνονται κυρίως σε συγκεκριμένες πτυχές όπως το Know Your Customer (KYC) ή τα πλαίσια εμπιστοσύνης, η παρούσα εργασία ακολουθεί μια ολιστική προσέγγιση, σχεδιάζοντας ένα ολοκληρωμένο εθνικό σύστημα DID που περιλαμβάνει όχι μόνο επαλήθευση ταυτότητας αλλά και λεπτομερείς μηχανισμούς συναίνεσης, πρωτόκολλα κοινής χρήσης δεδομένων και πληρεξούσιο επανακρυπτογράφηση, προσφέροντας έτσι μια απτή, πιο ολοκληρωμένη λύση για ανάπτυξη σε πραγματικό κόσμο. Η ανάπτυξη ενός πρωτόλειου λειτουργικού πρωτοτύπου, σε συνδυασμό με προσομοιώσεις και αξιολογήσεις χρησιμοποιώντας σενάρια πραγματικού κόσμου, ξεχωρίζει αυτή την εργασία από τις καθαρά θεωρητικές εξερευνήσεις που βρέθηκαν σε δημοσιευμένες εργασίες όπως το [94]. Η παρούσα έρευνα εμβαθύνει στις πρακτικές δυνατότητες εφαρμογής ενός εν δυνάμει εθνικού συστήματος DID, παρέχοντας εις βάθος ανάλυση των προκλήσεων και των αντισταθμίσεων που εμπλέκονται στην επίτευξη επεκτασιμότητας, ασφάλειας και ελέγχου από τον χρήστη, σε αντίθεση με τη θεωρητική επισκόπηση που παρουσιάζεται στο [94]. Ξεπερνά τις βασικές αρχές του SSI που εκεί περιγράφονται, ενσωματώνοντας ποικιλία τεχνολογιών που βελτιώνουν την ιδιωτικότητα, επιδεικνύοντας μια ισχυρότερη δέσμευση για την προστασία των δεδομένων χρήστη σε αναλυτικό επίπεδο. Επιπλέον, στο [95] διερευνάται η εφαρμογή του SSI σε Διοργανωτικές Επιχειρηματικές Διαδικασίες (Interorganizational Business Process, IOBP), κάτι το οποίο προσιδιάζει στο όραμα και μηχάνευση της ημέτερης εκπονηθείσας εργασίας αναφορικά με την ευρύτητα της εφαρμογής, ωστόσο η τελευταία προσβλέπει σε καθολική, παν-επιχειρηματική και παν-οργανωσιακή υλοποίηση, με αρκετά προσιτή την υλοποίηση τόσο από άποψης τεχνογνωσίας, όσο και ώριμων εργαλείων.

Επιπροσθέτως, η ενσωμάτωση τριών διακριτών εργαλείων, δηλαδή κατανεμημένης βάσης δεδομένων (Cassandra), ροής συμβάντων σε πραγματικό χρόνο (Kafka), και επανακρυπτογράφηση διακομιστή μεσολάβησης είναι μια νέα προσέγγιση που δεν έχει διερευνηθεί στις αναφερόμενες εργασίες. Αυτός ο συνδυασμός τεχνολογιών αντιμετωπίζει τα προβλήματα επεκτασιμότητας, απόδοσης και ασφάλειας πιο αποτελεσματικά από τις παραδοσιακές λύσεις που βασίζονται σε Blockchain, παρέχοντας μια πιο ισχυρή και αποτελεσματική υποδομή για ένα εθνικό σύστημα DID. Παραταύτα, ενώ το [96] εισάγει ένα πολύτιμο πολυεπίπεδο πλαίσιο αξιοπιστίας για το SSI, αυτή η εργασία περιλαμβάνει ένα ολοκληρωμένο σύστημα DID καλύπτοντας εκατέρωθεν τόσο θεωρητικές όσο και πρακτικές

αναζητήσεις, αντιμετωπίζοντας όχι μόνο τη διαχείριση εμπιστοσύνης αλλά και την **κοινή χρήση δεδομένων, τη συναίνεση και τα σύγχρονα ζητήματα απορρήτου**. Η λεπτομερής εφαρμογή και αξιολόγηση παρέχουν μια πιο ολιστική κατανόηση των προκλήσεων και των ευκαιριών που συνεπάγεται η ανάπτυξη ενός συστήματος DID σε εθνικό επίπεδο. Η διπλωματική διακρίνεται επίσης αντιμετωπίζοντας το εξελισσόμενο νομικό τοπίο και τους ηθικούς λόγους που περιβάλλουν την εφαρμογή του DID, μια πτυχή που συχνά παραβλέπεται σε καθαρά τεχνικές εξερευνήσεις όπως διαφαίνεται στο [96]. Διασφαλίζοντας ότι το σύστημα ευθυγραμμίζεται με τους σχετικούς κανονισμούς και τηρεί τις ηθικές αρχές στη διαχείριση δεδομένων, η έρευνα ανοίγει το δρόμο για υπεύθυνη ανάπτυξη και ανάπτυξη αποκεντρωμένων συστημάτων ταυτότητας. Επιπλέον, η ενδελεχής ανάλυση των αλληλεπιδράσεων των χρηστών με τη διεπαφή που βασίζεται σε τερματικό παρέχει πολύτιμες πληροφορίες για μελλοντικές υλοποιήσεις, υπογραμμίζοντας τη σημασία του σχεδιασμού με επίκεντρο τον χρήστη για την επίτευξη ευρείας υιοθέτησης αποκεντρωμένων λύσεων ταυτότητας.

Πτυχή	Σύστημα DID Διπλωματικής	Άλλες υφιστάμενες λύσεις
Τύπος συστήματος	Αποκεντρωμένο DID	Κεντρικό DID, Blockchain-based DID
Βάση Δεδομένων	Cassandra	MongoDB, CouchDB, Ethereum, Hyperledger Fabric
Κρυπτογράφηση	Επανακρυπτογράφηση μεσολάβησης με βιβλιοθήκη IronCoreLabs	Απευθείας κρυπτογράφηση, κρυπτογράφηση με κλειδί DID
Συνοχή δεδομένων	Κατανεμημένη βάση δεδομένων Cassandra	Κεντρική βάση δεδομένων, Blockchain
Ευελιξία	Δυνατότητα προσαρμογής μοντελοποίησης δεδομένων	Περιορισμένη ευελιξία, αμετάβλητα δεδομένα στο Blockchain
Κόστος	Χαμηλότερο κόστος ανάπτυξης και συντήρησης, χρήση ώριμων τεχνολογιών	Υψηλότερο κόστος ανάπτυξης και συντήρησης, χρεώσεις ανά συναλλαγή στο Blockchain
Συμμόρφωση	Δυνατότητα τροποποίησης δεδομένων	Δυσκολία τροποποίησης δεδομένων
Γνωριμία ρυθμιστικών αρχών	Γνωστές αρχές κατανεμημένων βάσεων δεδομένων	Νέες έννοιες Blockchain
Πλεονεκτήματα για χρήστες	Έλεγχος δεδομένων, υψηλή ακρίβεια, ισχυρή ασφάλεια, διαφάνεια, καινοτόμες υπηρεσίες, ικανότητα εθνικών και δια-υπηρεσιακών επεκτάσεων	Περιορισμένος έλεγχος, κατακερματισμός, αβεβαιότητα, κίνδυνοι ασφάλειας, περιορισμένη δια-υπηρεσιακή χρήση

Εικόνα 39: Διαφοροποιήσεις ημέτερης διπλωματικής από υφιστάμενες λύσεις

8.4 Περί Πλεονεκτημάτων & Μειονεκτημάτων

8.4.1 Πλεονεκτήματα

Στην καρδιά του, αυτό το αποκεντρωμένο σύστημα ταυτότητας αλλάζει θεμελιωδώς τη δυναμική ισχύος, θέτοντας τους χρήστες στον άμεσο έλεγχο της ψηφιακής τους ταυτότητας (DID) και των δεδομένων που σχετίζονται με αυτές. Σε αντίθεση με τα παραδοσιακά μοντέλα όπου οι κεντρικές πλατφόρμες λειτουργούν ως φύλακες, οι χρήστες εδώ διαχειρίζονται τις δικές τους πληροφορίες. Οι λεπτομερείς μηχανισμοί συναίνεσης παρέχουν πρωτοφανή ακρίβεια, δίνοντας τη δυνατότητα στους χρήστες (είτε ασθενείς είτε τρίτους) να προσδιορίζουν ποια συγκεκριμένα δεδομένα μοιράζονται με ποιες οντότητες και για πόσο χρονικό διάστημα – ένα επίπεδο ελέγχου που απουσιάζει στα κεντρικά συστήματα. Αυτή η εστίαση στην επιλεκτική αποκάλυψη και τη συναίνεση με γνώμονα τον χρήστη ενισχύει δραματικά το απόρρητο, περιορίζοντας τη δυνατότητα τρίτων οντοτήτων και οργανισμών να συντάσσουν εκτεταμένα προφίλ ή να παρακολουθούν την πλήρη δραστηριότητα ενός χρήστη, συγκλίνοντας σε ένα ενιαίο προφίλ. Η αποκεντρωμένη αρχιτεκτονική του ταυτοτικού συστήματος συνάμα της κατάτμησης της βάσεως αυξάνει εγγενώς την ασφάλεια, εξαλείφοντας την ευπάθεια ενός μόνο σημείου αστοχίας που θα μπορούσε να στοχεύσει ή να τεθεί σε κίνδυνο, οπότε σε συνδυασμό με ισχυρά πρωτόγονα κρυπτογραφικά στοιχεία, η ακεραιότητα των δεδομένων ενισχύεται, καθιστώντας πιο δύσκολο για τους κακόβουλους παράγοντες να παραβιάσουν ευαίσθητες πληροφορίες. Είναι σημαντικό ότι ο μηχανισμός επανακρυπτογράφησης μεσολάβησης επιτρέπει την ασφαλή ανταλλαγή δεδομένων χωρίς να χρειάζεται ποτέ οι μεσάζοντες να δουν το αρχικό απλό κείμενο, διασφαλίζοντας περαιτέρω την εμπιστευτικότητα. Η έμφαση του συστήματος στη διαφάνεια και τη δυνατότητα ελέγχου χτίζει εμπιστοσύνη. Οι βασικές λειτουργίες – από τη δημιουργία DID έως τις τροποποιήσεις αδειών – μπορούν να καταγραφούν και να επαληθευτούν, μειώνοντας την εξάρτηση από αδιαφανείς κεντρικές αρχές και ενισχύοντας μεγαλύτερη εμπιστοσύνη στη δικαιοσύνη του συστήματος. Επομένως, αυτή η προσέγγιση έχει τη δυνατότητα εν πολλοίς να φέρει επανάσταση στον τρόπο διαχείρισης της ταυτότητας. Ανοίγει την πόρτα σε καινοτόμες υπηρεσίες που αλληλεπιδρούν απρόσκοπτα με δεδομένα ελεγχόμενα από τον χρήστη, προσφέροντας νέες εμπειρίες με επίκεντρο αυτόν. Για βιομηχανίες που εξαρτώνται σε μεγάλο βαθμό από την επαλήθευση ταυτότητας και τη συμμόρφωση, αυτό το σύστημα θα μπορούσε να εξορθολογίσει τις διαδικασίες ενσωμάτωσης και ελέγχου ταυτότητας, εξοικονομώντας τελικά χρόνο και πόρους.

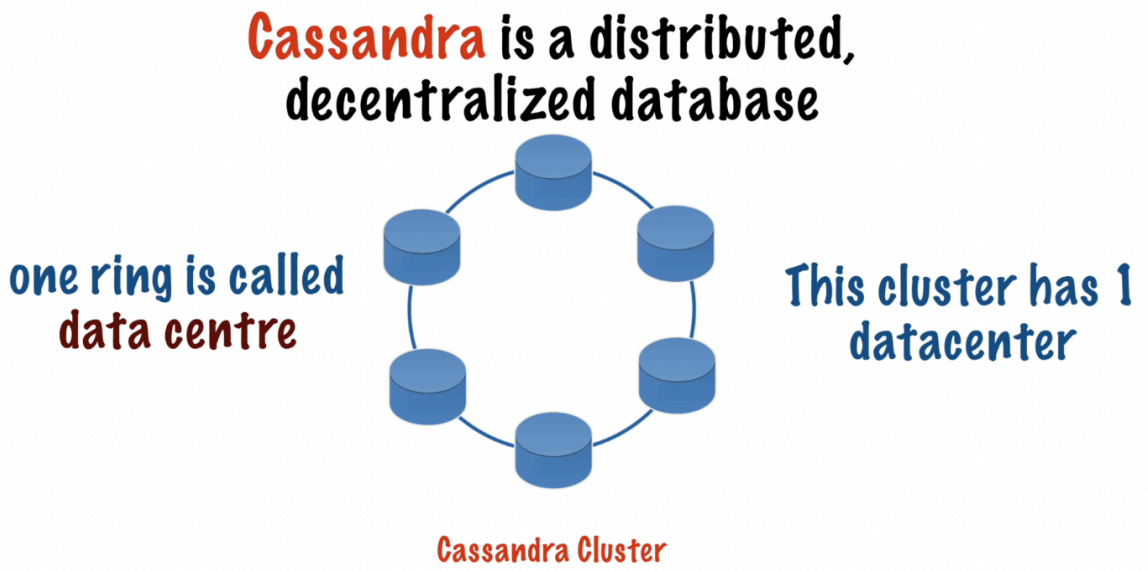
8.4.2 Μειονεκτήματα

Τα αποκεντρωμένα συστήματα ταυτότητας, ενώ προσφέρουν πολλά πλεονεκτήματα, αντιμετωπίζουν εμπόδια στην εμπειρία του χρήστη. Έτσι και στο παρόν σύστημα, η διαχείριση DID, διαπιστευτηρίων και αδειών μπορεί να είναι τεχνικά πολύπλοκη για μη τεχνικούς χρήστες, εμποδίζοντας την

υιοθέτηση. Ως εκ τούτου, η έλλειψη τυποποίησης θα μπορούσε εν γένει να οδηγήσει σε κατακερματισμένα οικοσυστήματα, περιορίζοντας τη φορητότητα των διαπιστευτηρίων σε όλες τις πλατφόρμες, αφού αυτά τα συστήματα βασίζονται συχνά σε μια άρρηκτη διαλειτουργική σχέση μεταξύ πολλών παραγόντων, οπότε ο μη ορθά χρονοπρογραμματισμός των συναλλαγών να λειτουργεί απαγορευτικά για συχνές ενημερώσεις. Συν τοις άλλοις, το εξελισσόμενο ρυθμιστικό τοπίο, ειδικά όσον αφορά την ιδιοκτησία δεδομένων και τη διαχείριση ταυτότητας, προσθέτει αβεβαιότητα για τις επιχειρήσεις που σκέφτονται να υιοθετήσουν. Παραταύτα, η ασφάλεια παραμένει ένα πολύπλευρο ζήτημα, με τους χρήστες να φέρουν την ευθύνη της προστασίας των ιδιωτικών κλειδιών τους, με την απώλεια να οδηγεί σε μόνιμη απώλεια πρόσβασης. Τα αποκεντρωμένα συστήματα μπορούν επίσης να ανοίξουν νέους φορείς επίθεσης που στοχεύουν έξυπνα συμβόλαια ή πρωτόκολλα DID. Τέλος, η ενοποίηση με την υπάρχουσα κεντρική υποδομή παρουσιάζει προκλήσεις, που επιδεινώνονται από την έλλειψη ευρέως διαδεδομένων προτύπων για αποκεντρωμένη επικοινωνία ταυτότητας και μορφές δεδομένων.

8.4.3 Κατανεμημένες Βάσεις Εναντίον Blockchain

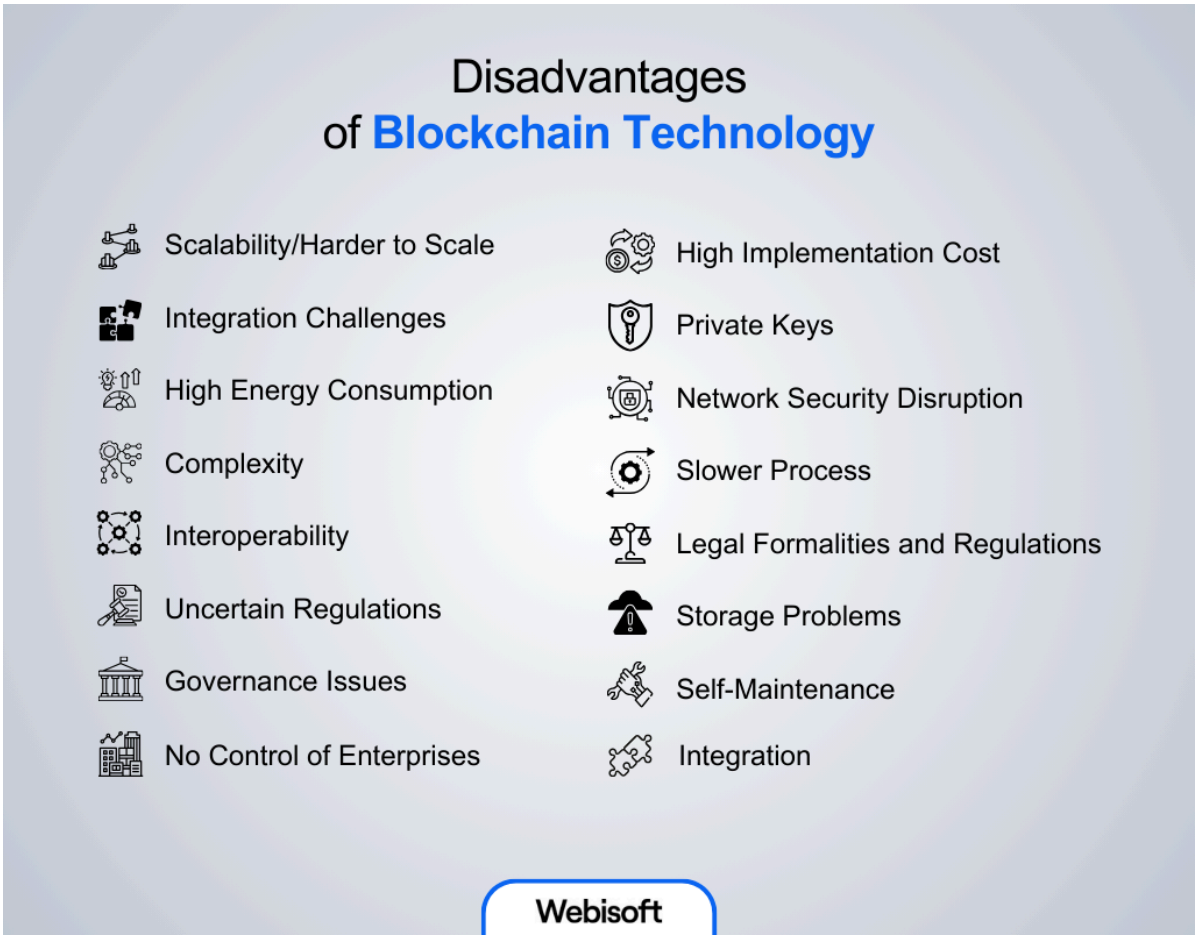
Η χρήση της Cassandra για το παρόν σύστημα δεν έγινε τυχαία, αφού η τεχνολογία της είναι εγγενώς σχεδιασμένη για υψηλή απόδοση εγγραφής και γρήγορες αναγνώσεις, καθιστώντας την ιδανικό για ένα αποκεντρωμένο σύστημα ταυτότητας. Οι συχνές λειτουργίες όπως ενημερώσεις αδειών ή τροποποιήσεις δεδομένων αντιμετωπίζονται αποτελεσματικά, διασφαλίζοντας μια ανταποκρινόμενη εμπειρία χρήστη. Οι παραδοσιακές δημόσιες αλυσίδες μπλοκ συχνά δυσκολεύονται να ταιριάξουν με αυτήν την επεκτασιμότητα, ιδιαίτερα εκείνες που βασίζονται σε πολύπλοκους μηχανισμούς



Εικόνα 40: Δόμηση αποκεντρωμένης φιλοσοφίας στη Cassandra [97]

συναίνεσης που εμποδίζουν τον όγκο συναλλαγών. Εντούτοις, με τη Cassandra παρακάμπτεται αυτό το κώλυμα αφού ο μηχανικός δύναται εύκολα να κλιμακώσει το σύμπλεγμα οριζόντια για να φιλοξενήσει μια αυξανόμενη βάση χρηστών χωρίς περίπλοκες λύσεις όπως το sharding ή sidechains που χρησιμοποιούνται συχνά για την αντιμετώπιση των περιορισμών κλιμάκωσης της αλυσίδας μπλοκ. Η ευελιξία που προσφέρει η Cassandra το καθιστά φυσικό κατάλληλο εργαλείο για ένα αποκεντρωμένο σύστημα ταυτότητας.

Η μοντελοποίηση δεδομένων είναι προσαρμόσιμη, επιτρέποντάς δόμηση με ακρίβεια εν σχέσει με δεδομένα χρήστη και DID. Παράλληλα, διατηρείται η δυνατότητα να εφαρμόζονται λεπτομερείς μηχανισμοί συναίνεσης και λεπτομερείς έλεγχοι πρόσβασης με τις πλούσιες δυνατότητες ερωτημάτων της Cassandra. Αντίθετα, οι αλυσίδες μπλοκ μπορεί να είναι άκαμπτες όταν αντιμετωπίζουν σύνθετες απαιτήσεις δεδομένων λόγω της εστίασής τους στο δόγμα του αμετάβλητου (Immutability). Για ένα έργο που εξελίσσεται με την πάροδο του χρόνου, η Cassandra διευκολύνει την προσαρμογή των σχημάτων ή την επέκταση της λειτουργικότητας, εκσυγχρονίζοντας τη συντήρηση του συστήματος σε σύγκριση με την πολυπλοκότητα της τροποποίησης της βασικής λογικής Blockchain.



Εικόνα 41: Μειονεκτήματα του Blockchain [98]

Στο μέτωπο του κόστους, η Cassandra έχει ένα πλεονέκτημα. Οι δημόσιες αλυσίδες μπλοκ συνήθως επιβαρύνονται με χρέωση ανά συναλλαγή, προσθέτοντας δυνητικά σημαντικά γενικά έξοδα σε ένα σύστημα με συχνές αλληλεπιδράσεις και ενημερώσεις DID. Η Cassandra αποφεύγει αυτές τις χρεώσεις ανά συναλλαγή. Ενώ και τα δύο συνεπάγονται κόστος υποδομής, η ωριμότητα του αναπτυξιακού οικοσυστήματος της συναρτήσει των καθιερωμένων εργαλείων, τις βιβλιοθήκες και τους έμπειρους προγραμματιστές, μπορεί να οδηγήσει σε χαμηλότερο συνολικό κόστος ανάπτυξης και συντήρησης σε σύγκριση με ένα πιο εξειδικευμένο σύνολο δεξιοτήτων του Blockchain.

Εν κατακλείδι, η Cassandra διαφαίνεται να ταιριάζει καλύτερα από κανονιστική άποψη. Σε τομείς με υψηλή ρύθμιση, όπως η υγειονομική περίθαλψη, η δυνατότητα τροποποίησης δεδομένων (εάν απαιτείται νομικά) μπορεί να είναι κρίσιμη, κάτι που η Cassandra υποστηρίζει εύκολα. Το Immutability, ένα βασικό χαρακτηριστικό του Blockchain, μπορεί να δημιουργήσει προκλήσεις σε αυτά τα περιβάλλοντα. Οι ρυθμιστικές αρχές είναι επίσης πιο εξοικειωμένες με τις αρχές των κατανεμημένων βάσεων δεδομένων, καθιστώντας τις συζητήσεις συμμόρφωσης δυνητικά ευκολότερες σε σύγκριση με τις νέες έννοιες που διέπουν την τεχνολογία Blockchain.

9

Συμπεράσματα & Μελλοντικές Εργασίες

9.1 Περίληψη Βασικών Ευρημάτων & Συμπεράσματα

Το προτεινόμενο εθνικό σύστημα DID φιλοδοξεί να φέρει ριζικές αλλαγές στον τρόπο διαχείρισης της ψηφιακής ταυτότητας, υιοθετώντας μια προσέγγιση αποκεντρωμένης, αυτόνομης, ασφαλούς, διαφανούς και διαλειτουργικής φύσης. Τοποθετώντας τον έλεγχο στα χέρια των πολιτών, το σύστημα πέτυχε να αναδείξει σε ένα πρώιμο στάδιο πως οι πολίτες και οι καθημερινοί χρήστες πράγματι μπορούν να ανακαταλάβουν ένα μέρος ή ολόκληρη την ευθύνη των δεδομένων τους, με απόλυτη αυτονομία. Παράλληλα, οι λεπτομερείς μηχανισμοί συναίνεσης που αναπτύχθηκαν κατέδειξαν εμφανώς τη διασφάλιση και τον έλεγχο της επιλεκτικής κοινής χρήσης δεδομένων, και η ρητή εξουσιοδότηση έδεσε αισίως και ομοιόμορφα ως ακροτελεύτια βασική λειτουργία.

Η αποκεντρωμένη υποδομή του συστήματος, βασισμένη σε βάση δεδομένων Cassandra, εξαλείφει τυχόν μεμονωμένα σημεία αστοχίας. Ασφαλώς, η εκ προοιμίου φύση της ανήκει στις κατανεμημένες και όχι αποκεντρωμένες βάσεις, και οφείλει να σημειωθεί πως η αποκέντρωση αφορά τη χρήση των DID και επιτυγχάνεται μέσω της κατάτμησης και αντιγραφής των βάσεων, με τα κρυπτογραφημένα δεδομένα να διανέμονται σε πολλαπλούς κόμβους, ενισχύοντας την ανθεκτικότητα έναντι κυβερνοεπιθέσεων. Επίσης, η επεκτασιμότητα αποτελεί βασικό στοιχείο σχεδιασμού ο οποίος μπορεί να επιδεχθεί πολλαπλές περαιτέρω προσεγγίσεις και επαυξήσεις επιτρέποντας στο σύστημα να προσαρμοστεί σε μια αυξανόμενη βάση χρηστών και σε ολοένα και μεγαλύτερους όγκους δεδομένων, παρόχων και εμπλεκομένων. Η επανακρυπτογράφηση που βασίζεται σε διακομιστή μεσολάβησης διευκολύνει την ασφαλή ανταλλαγή δεδομένων χωρίς να διακυβεύεται η εμπιστευτικότητα ευαίσθητων πληροφοριών. Η πλατφόρμα διαχειρίζεται με άνεση διάφορους τύπους δεδομένων, όπως ιατρικά αρχεία και διοικητικά έγγραφα, και οι σχολαστικές καταγραφές ελέγχου να διασφαλίζουν την υπευθυνότητα και να επιτρέπουν τη πλήρη ιχνηλάτηση των συμβάντων πρόσβασης δεδομένων.

Η διαλειτουργικότητα αποτελεί θεμέλιο λίθο του συστήματος, αφού σε δυνητική του παραγωγική υλοποίηση θα κληθεί να εξυπηρετήσει κυβερνητικές, υγειονομικές, ασφαλιστικές, και πολλές άλλες υπηρεσίες. Έτσι, τυποποιημένα πρωτόκολλα DID διευκολύνουν την ομαλή επικοινωνία μεταξύ διαφορετικών υπηρεσιών και πλατφορμών, ενώ παράλληλα, τα κοινά μητρώα DID και καλά καθορισμένα πλαίσια διακυβέρνησης βελτιστοποιούν τη συνεργασία μεταξύ των φορέων. Επομένως, οι χρήστες απολαμβάνουν τα οφέλη ενός ενιαίου συστήματος, αξιοποιώντας τα DID τους σε τομείς όπως η υγειονομική περίθαλψη, τα οικονομικά και η ηλεκτρονική διακυβέρνηση. Ως εκ τούτου, το προτεινόμενο σύστημα δύναται να ενδυναμώσει τα άτομα στον ψηφιακό κόσμο, ενισχύοντας την εμπιστοσύνη και την ευρύτερη συμμετοχή σε διαδικτυακές υπηρεσίες. Οι κίνδυνοι παραβίασης δεδομένων και μη εξουσιοδοτημένης πρόσβασης μειώνονται σημαντικά, θωρακίζοντας το απόρρητο και την ασφάλεια των χρηστών, ενώ η διαφάνεια και η λογοδοσία ενσωματώνονται επιτέλους στις πρακτικές διαχείρισης δεδομένων του συστήματος. Συνολικά, αυτό το πλαίσιο DID παρουσιάζεται ως μια αξιοσημείωτη λύση για τον επαναπροσδιορισμό της διαχείρισης ψηφιακής ταυτότητας, δίνοντας προτεραιότητα τόσο στον ατομικό έλεγχο όσο και στην προστασία δεδομένων στο διασυνδεδεμένο ψηφιακό μας τοπίο.

9.2 Προτάσεις για Μελλοντικές Επεκτάσεις

Σε πρώτη φάση, οι επεκτάσεις που θα μπορούσαν να εγκιβωτιστούν άμεσα στο σύστημα αποτελούν σίγουρα η **από τρίτο σε τρίτο** υλοποίηση, δηλαδή η δημοσιοποίηση του εγγράφου, δεδομένων και λοιπών προς έναν άλλο τρίτο κατευθείαν. Κατά αυτή τη περίπτωση, η διαδικασία παραμένει η ίδια, δηλαδή πραγματοποιείται αίτηση προς το πολίτη ούτως ώστε να προχωρήσει στην έκδοση νέου DID εγγράφου, δημιουργώντας έτσι τρίγωνο Τρίτος-Πολίτης-Τρίτος. Δευτερευόντως, η ολοκλήρωση της πιλοτικής υλοποίησης θα μπορούσε κάλλιστα να συμπεριλάβει την αξιοποίηση **και** της λειτουργίας **WRITE** εκ μέρους των εμπλεκομένων που παρεμβαίνουν δυναμικά στα δεδομένα. Παραταύτα, παρακάτω αναπτύσσεται μια ρητορική που καταδεικνύει ένα φάσμα επεκτάσεων που κυμαίνεται σε ένα πιο μακροσκοπικό επίπεδο.

- **Προηγμένο απόρρητο και ασφάλεια**

Το έργο θα μπορούσε να ενισχύσει περαιτέρω τη δέσμευσή του για την προστασία της ιδιωτικής ζωής με την εφαρμογή αποδείξεων μηδενικής γνώσης (Zero Knowledge Proof, ZKP). Αυτό θα επιτρέπει στους χρήστες να επαληθεύουν χαρακτηριστικά ταυτότητας (όποιος είναι άνω των 18 ετών) χωρίς να αποκαλύπτουν ευαίσθητα υποκείμενα δεδομένα. Επιπλέον, η εξερεύνηση της Ομομορφικής Κρυπτογράφησης θα παρείχε ευκαιρίες για ασφαλή αναλυτικά στοιχεία και επικύρωση σε κρυπτογραφημένα δεδομένα, ενισχύοντας το απόρρητο ακόμη και κατά την επεξεργασία δεδομένων. Επιπροσθέτως, η ενσωμάτωση επιλογών βιομετρικού ελέγχου ταυτότητας (δακτυλικό αποτύπωμα,

αναγνώριση προσώπου) θα μπορούσε να προσθέσει ένα ισχυρό επίπεδο ασφάλειας και άνεσης για την **πρόσβαση** στο DID.

☐ **Εμπειρία χρήστη**

Οι επενδύσεις στην ανάπτυξη εύχρηστων διεπαφών χρήστη και πορτοφολιών είναι μείζονος σημασίας για την υιοθέτηση του κοινού. Οι σαφείς απεικονίσεις και η απλοποιημένη διαχείριση συναίνεσης μπορούν να κάνουν την αλληλεπίδραση DID προσβάσιμη σε μη τεχνικούς χρήστες. Θα πρέπει να εφαρμόζονται διασφαλίσεις, όπως μέθοδοι ανάκτησης κοινωνικής δικτύωσης (έμπιστες επαφές) ή εφεδρικές μέθοδοι, για την αντιμετώπιση πιθανής απώλειας ιδιωτικών κλειδιών. Η ενσωμάτωση με προσωπικές συσκευές ή πορτοφόλια υλικού (Hardware Keys) θα μπορούσε να απλοποιήσει τον έλεγχο ταυτότητας.

☐ **Επεκτάσεις μεταξύ τομέων**

Η επέκταση του συστήματος DID για διασύνδεση με συσκευές Διαδικτύου των Πραγμάτων (Internet of Things, IoT) ανοίγει νέες δυνατότητες στους χρήστες να διαχειρίζονται συνδεδεμένα οικοσυστήματα (έξυπνα σπίτια, οχήματα). Επίσης, η εφαρμογή συστημάτων φήμης που συνδέονται με DIDs θα μπορούσε να διευκολύνει την εμπιστοσύνη σε διαδικτυακές αγορές ή επαγγελματικά δίκτυα.

☐ **Ρυθμιστική & Διακυβέρνηση**

Το έργο θα χρειαστεί να συμμετάσχει ενεργά στις προσπάθειες παγκόσμιας τυποποίησης, ενισχύοντας τη διεθνή συνεργασία για τη διαλειτουργικότητα. Μάλιστα, η προληπτική δέσμευση με τους υπεύθυνους χάραξης πολιτικής είναι καίριας σημασίας για τη διαμόρφωση των εξελισσόμενων νομικών πλαισίων που αφορούν την ευθύνη και την προστασία δεδομένων. Οπότε, η θέσπιση κατευθυντήριων γραμμών δεοντολογίας θα διασφαλίσει την υπεύθυνη ανάπτυξη και θα μετριάσει τους κινδύνους κακής χρήσης.

☐ **Πιλοτικά Έργα**

Η συνεργασία με κρατικούς φορείς σε πιλοτικά έργα για την έκδοση επίσημων εγγράφων (διαβατήρια κ.λπ.) μπορεί να αποδείξει την αποτελεσματικότητα του συστήματος στη μείωση της απάτης. Κοινώς, η συνεργασία με παρόχους υγειονομικής περίθαλψης για τη διαχείριση συναίνεσης ασθενών με βάση το DID μπορεί να ενισχύσει την αυτονομία των ασθενών και να υποστηρίξει ερευνητικές πρωτοβουλίες. Η χρήση DID στις αλυσίδες εφοδιασμού μπορεί να βελτιώσει τη διαφάνεια και να διασφαλίσει την ηθική προμήθεια.

10

Επίλογος

Εν κατακλείδι, η εργασία έκανε προσπάθεια να παράσχει μια ολοκληρωμένη έρευνα για την ανάπτυξη και αξιολόγηση ενός πρωτότυπου αποκεντρωμένου συστήματος ταυτότητας (DID). Ως άμεσο αντίκρισμα, η πορεία της έρευνας και της εφαρμογής έχει αποκαλύψει τόσο τις πολλά υποσχόμενες δυνατότητες όσο και την εγγενή πολυπλοκότητα ενός τέτοιου συστήματος. Έτσι, μέσω αυστηρής ανάλυσης και ανάπτυξης πρωτοτύπων, αυτή η εργασία πραγματοποίησε ειδική μνεία στις τεχνικές, κανονιστικές και ηθικές προκλήσεις που πρέπει να αντιμετωπιστούν για ευρεία υιοθέτηση. Ωστόσο όμως, ενώ μεν το πρωτότυπο σύστημα καταδεικνύει τη σκοπιμότητα και τα πλεονεκτήματα της αποκεντρωμένης ταυτότητας, είναι σημαντικό να αναγνωρίσουμε τους περιορισμούς του ως ερευνητικό τεχνούργημα. Απαιτείται περαιτέρω έρευνα για την αντιμετώπιση των προκλήσεων επεκτασιμότητας, ασφάλειας και χρηστικότητας που προκύπτουν σε πραγματικές υλοποιήσεις. Εντούτοις, αυτή η διπλωματική χρησιμεύει ως σκαλοπάτι προς την υλοποίηση του οράματος μιας χρηστοκεντρικής, διαφύλαξης της ιδιωτικής ζωής και αποτελεσματικής υποδομής ψηφιακής ταυτότητας. Τα ευρήματα αυτής της έρευνας όχι μόνο συμβάλλουν στη συνεχή ακαδημαϊκή συζήτηση, αλλά παρέχουν επίσης πρακτικές γνώσεις για τους υπεύθυνους χάραξης πολιτικής, τους τεχνολόγους και τους επαγγελματίες υγείας που επιδιώκουν να περιηγηθούν στην πολυπλοκότητα της διαχείρισης ψηφιακής ταυτότητας. Συνοψίζοντας, με γνώμονα την ενθάρρυνση προς τη βαθύτερη κατανόηση των παραμέτρων που εμπλέκονται, η εργασία ελπίζει να τονώσει την περαιτέρω καινοτομία και τη συνεργασία προς ένα μέλλον όπου τα άτομα θα έχουν μεγαλύτερο έλεγχο της ψηφιακής τους ταυτότητας, προστατεύοντας παράλληλα το απόρρητο και την ασφάλειά τους κατά έναν πλήρως καινοφανή, μοντέρνο και σύγχρονο τρόπο.

Βιβλιογραφία

[1] Z. Zheng, S. Xie, H. Dai, X. Chen, and H. Wang, “An Overview of Blockchain Technology: Architecture, Consensus, and Future Trends,” *2017 IEEE International Congress on Big Data (BigData Congress)*, Jun. 2017.

[2] Frederico Schardong and Ricardo Felipe Custódio, “Self-Sovereign Identity: A Systematic Review, Mapping and Taxonomy,” *arXiv (Cornell University)*, Aug. 2021, doi: https://doi.org/10.48550/arxiv.2108.08338.

[3] D. Pöhn and W. Hommel, “An overview of limitations and approaches in identity management,” *Proceedings of the 15th International Conference on Availability, Reliability and Security*, Jul. 2020, doi: https://doi.org/10.1145/3407023.3407026.

[4] Yuan Cao and Lin Yang, “A survey of Identity Management technology,” *2010 IEEE International Conference on Information Theory and Information Security*, Dec. 2010, doi: https://doi.org/10.1109/icitis.2010.5689468.

[5] European Union, “Regulation (EU) 2016/679 of the European Parliament and of the Council of 27 April 2016 on the protection of natural persons with regard to the processing of personal data and on the free movement of such data, and repealing Directive 95/46/EC (General Data Protection Regulation) (Text with EEA relevance),” *Europa.eu*, 2016. https://eur-lex.europa.eu/eli/reg/2016/679/oj

[6] GDPR, “Art. 12 GDPR – Transparent information, communication and modalities for the exercise of the rights of the data subject | General Data Protection Regulation (GDPR),” *General Data Protection Regulation (GDPR)*, 2013. https://gdpr-info.eu/art-12-gdpr/

[7] GDPR, “Art. 13 GDPR – Information to be provided where personal data are collected from the data subject | General Data Protection Regulation (GDPR),” *General Data Protection Regulation (GDPR)*, 2013. https://gdpr-info.eu/art-13-gdpr/

[8] GDPR, “Art. 14 GDPR – Information to be provided where personal data are collected from the data subject | General Data Protection Regulation (GDPR) | General Data Protection Regulation (GDPR),” *General Data Protection Regulation (GDPR)*, 2013. https://gdpr-info.eu/art-14-gdpr/

[9] GDPR, “Art. 15 GDPR – Right of access by the data subject | General Data Protection Regulation (GDPR),” *General Data Protection Regulation (GDPR)*, 2013. https://gdpr-info.eu/art-15-gdpr/

[10] GDPR, “Art. 17 GDPR – Right to erasure (‘right to be forgotten’) | General Data Protection Regulation (GDPR),” *General Data Protection Regulation (GDPR)*, 2013. https://gdpr-info.eu/art-17-gdpr/

[11] GDPR, “Art. 32 GDPR – Security of processing | General Data Protection Regulation (GDPR),” *General Data Protection Regulation (GDPR)*, 2013. https://gdpr-info.eu/art-32-gdpr/

[12] GDPR, “Art. 33 GDPR – Notification of a personal data breach to the supervisory authority | General Data Protection Regulation (GDPR),” *General Data Protection Regulation (GDPR)*, 2013. https://gdpr-info.eu/art-33-gdpr/

[13] GDPR, “Art. 34 GDPR – Communication of a personal data breach to the data subject | General Data Protection Regulation (GDPR),” *General Data Protection Regulation (GDPR)*, 2013. https://gdpr-info.eu/art-34-gdpr/

[14] GDPR, “Art. 6 GDPR – Lawfulness of processing | General Data Protection Regulation (GDPR),” *General Data Protection Regulation (GDPR)*, 2013. https://gdpr-info.eu/art-6-gdpr/

[A15] “Έναρξη Ισχύος του Κανονισμού GDPR για τα Προσωπικά Δεδομένα,” *esm.com.gr*, May 23, 2018. Available: http://esm.com.gr/enarksi-isxios-kanonismou-gdpr/. [Accessed: Jun. 20, 2024]

[16] H. van Cann and S. L. von Gohren Edwin, “DIF FAQ,” *identity.foundation*. https://identity.foundation/faq/ (accessed May 9, 2024).

[17] M. Maynes, “Why decentralization is the future of digital identities,” *Microsoft Security Blog*, Mar. 10, 2022. https://www.microsoft.com/en-us/security/blog/2022/03/10/why-decentralization-is-the-future-of-digital-identities/ (accessed May 9, 2024).

[18] IBM, “Decentralized identity,” *IBM Security Verify Documentation Hub*. https://docs.verify.ibm.com/verify/docs/use-cases-decentralized-identity (accessed May 10, 2024).

[19] DIDecentral, “Evernym,” *Verifiable Credentials and Self Sovereign Identity Web Directory*, Nov. 02, 2020. https://decentralized-id.com/companies/evernym/ (accessed May 10, 2024).

[20] “What Is Self-Sovereign Identity?,” *www.onchainid.com*. Available: https://www.onchainid.com/post/what-is-self-sovereign-identity

[21] L. A. Neilson, “Digital ID, Surveillance, and the Value of Privacy - Part One,” *Justice Centre for Constitutional Freedoms*. https://www.jccf.ca/published_reports/digital-id-surveillance-and-the-value-of-privacy/ (accessed May 10, 2024).

[22] F. Wang and P. De Filippi, “Self-Sovereign Identity in a Globalized World: Credentials-Based Identity Systems as a Driver for Economic Inclusion,” *Frontiers in Blockchain*, vol. 2, Jan. 2020, doi: https://doi.org/10.3389/fbloc.2019.00028.

[23] M. Schäffner, “DEPARTMENT OF INFORMATICS Analysis and Evaluation of Blockchain-based Self-Sovereign Identity Systems.” Available: https://wwwmatthes.in.tum.de/file/1oua980d9tgxx/Sebis-Public-Website/-/Master-s-Thesis-Martin-Schaeffner/Masters%20Thesis%20Martin%20Scha%CC%88ffner.pdf

[24] “Decentralized Identity Own and control your identity.” Available: https://query.prod.cms.rt.microsoft.com/cms/api/am/binary/RE2DjfY

[25] C. Farmer, S. Pick, and A. Hill, “Decentralized identifiers for peer-to-peer service discovery,” *2021 IFIP Networking Conference (IFIP Networking)*, Jun. 2021, doi: https://doi.org/10.23919/ifipnetworking52078.2021.9472201.

[26] A. Othman and J. H. Callahan, “The Horcrux Protocol: A Method for Decentralized Biometric-based Self-sovereign Identity,” *International Joint Conference on Neural Network*, Jul. 2018, doi: https://doi.org/10.1109/ijcnn.2018.8489316.

[27] A. A. Torongo and M. Toorani, “Blockchain-based Decentralized Identity Management for Healthcare Systems,” *arXiv.org*, Jul. 30, 2023. https://arxiv.org/abs/2307.16239 (accessed Dec. 10, 2023).

[28] S. Wadhwa, “Decentralized digital identity management using blockchain and its implication on public sector,” *Semantic Scholar*, 2019. https://www.semanticscholar.org/paper/Decentralized-digital-identity-management-using-and-Wadhwa/9b6e9bd362dfa6182e833358c34b2ec60cf170cd (accessed May 28, 2024).

[29] S. R. Garzon, H. Yildiz, and A. Küpper, “Decentralized Identifiers and Self-sovereign Identity in 6G,” *arXiv.org*, Aug. 04, 2022. https://arxiv.org/abs/2112.09450 (accessed May 28, 2024).

[30] I. T. Javed, F. Alharbi, B. Bellaj, T. Margaria, N. Crespi, and K. N. Qureshi, “Health-ID: A Blockchain-Based Decentralized Identity Management for Remote Healthcare,” *Healthcare*, vol. 9, no. 6, p. 712, Jun. 2021, doi: https://doi.org/10.3390/healthcare9060712.

[31] A. M. Alnour and K. H. Kim, “Decentralized Identifiers (DIDs)-Based Authentication Scheme for Smart Health Care System,” *2022 Thirteenth International Conference on Ubiquitous and Future Networks (ICUFN)*, pp. 433–438, Jul. 2022, doi: https://doi.org/10.1109/ICUFN55119.2022.9829562.

[32] M. Alizadeh, K. Andersson, and O. Schelen, “Comparative Analysis of Decentralized Identity Approaches,” *IEEE Access*, vol. 10, pp. 92273–92283, 2022, doi: https://doi.org/10.1109/access.2022.3202553.

[33] “Decentralized Identifiers (DIDs): The Ultimate Beginner’s Guide 2023,” *www.dock.io*. Available: https://www.dock.io/post/decentralized-identifiers

[34] “Decentralized Identifiers (DIDs) v1.0,” *www.w3.org*. Available: https://www.w3.org/TR/did-core/#architecture-overview

[35] “Decentralized Identifiers (DIDs) v1.0,” *W3.org*, Jul. 19, 2022. Available: https://www.w3.org/TR/did-core/#example-an-example-of-a-relative-did-url. [Accessed: Jun. 20, 2024]

[36] M. Sporny, D. Longley, M. Sabadello, D. Reed, O. Steele, and Christopher Allen, “Decentralized Identifiers (DIDs) v1.0,” *www.w3.org*, Jul. 19, 2022. https://www.w3.org/TR/did-core/

[37] A. Wider, S. Verma, and A. Akhtar, “Decentralized Data Governance as Part of a Data Mesh Platform: Concepts and Approaches,” *arXiv (Cornell University)*, Jan. 2023, doi: https://doi.org/10.48550/arxiv.2307.02357.

[38] S. Hanisch, J. Todt, J. Patino, N. Evans, and Thorsten Strufe, “A False Sense of Privacy: Towards a Reliable Evaluation Methodology for the Anonymization of Biometric Data,” *Proceedings on Privacy Enhancing Technologies*, vol. 2024, no. 1, pp. 116–132, Jan. 2024, doi: https://doi.org/10.56553/popets-2024-0008.

[39] X. Mao, Y. Chen, C. Deng, and X. Zhou, “A novel privacy-preserving biometric authentication scheme,” vol. 18, no. 5, pp. e0286215–e0286215, May 2023, doi: https://doi.org/10.1371/journal.pone.0286215.

[40] M. M. Merlec, Y. K. Lee, S.-P. Hong, and H. P. In, “A Smart Contract-Based Dynamic Consent Management System for Personal Data Usage under GDPR,” *Sensors (Basel, Switzerland)*, vol. 21, no. 23, p. 7994, Nov. 2021, doi: https://doi.org/10.3390/s21237994.

[41] T. Habibu, E. Talina, and A. Elikana, “Developing an Algorithm for Securing the Biometric Data Template in the Database,” *International Journal of Advanced Computer Science and Applications*, vol. 10, no. 10, 2019, doi: https://doi.org/10.14569/ijacsa.2019.0101051.

[42] United Nations, “E-Government Survey 2020 Digital Government in the Decade of Action for Sustainable Development With addendum on COVID-19 Response,” 2020. Available: https://publicadministration.un.org/egovkb/Portals/egovkb/Documents/un/2020-Survey/2020%20UN%20E-Government%20Survey%20(Full%20Report).pdf

[43] I. Staff, “The 2021 Data Regulation Recap,” *InCountry*, Dec. 14, 2021. Available: https://incountry.com/blog/the-2021-data-regulation-recap/

[44] A. Jain and U. Uludag, “Hiding Biometric Data.” Accessed: Feb 12, 2024. [Online]. Available: https://biometrics.cse.msu.edu/Publications/SecureBiometrics/JainUludag_HidingBiometrics_PAMI03.pdf

[45] C.-A. Toli, A. Aly, and B. Preneel, “Privacy-Preserving Multibiometric Authentication in Cloud with Untrusted Database Providers,” *Cryptology ePrint Archive (eprint.iacr.org)*, 2018. https://eprint.iacr.org/2018/359 (accessed May 28, 2024).

[46] Juliano Paulo Menzen, K. Farias, and V. Bischoff, “Using biometric data in software engineering: a systematic mapping study,” vol. 40, no. 9, pp. 880–902, Mar. 2020, doi: https://doi.org/10.1080/0144929x.2020.1734086.

[47] B. Moriarty *et al.*, “Utility-Preserving Biometric Information Anonymization,” *Lecture notes in computer science*, pp. 24–41, Jan. 2022, doi: https://doi.org/10.1007/978-3-031-17146-8_2.

[48] “[White Paper] Implementing a national digital ID program based on biometrics,” *IDEMIA*. https://www.idemia.com/public_download/white-paper-implementing-national-digital-id-program-based-biometrics (accessed May 28, 2024).

[49] R. D. Garcia, G. S. Ramachandran, R. Jurdak, and J. Ueyama, “A Blockchain-based Data Governance Framework with Privacy Protection and Provenance for e-Prescription,” *arXiv.org*, Dec. 27, 2021. https://arxiv.org/abs/2112.13956 (accessed May Jan 22, 2024).

[50] Ελένη Κ. Ντούρου, “Η ΤΗΡΗΣΗ ΤΩΝ ΙΑΤΡΙΚΩΝ ΑΡΧΕΙΩΝ ΚΑΙ Η ΠΡΟΣΤΑΣΙΑ ΠΡΟΣΩΠΙΚΩΝ ΔΕΔΟΜΕΝΩΝ ΤΩΝ ΑΣΘΕΝΩΝ ΥΠΟ ΤΟ ΠΡΙΣΜΑ ΤΟΥ ΕΥΡΩΠΑΪΚΟΥ ΚΑΝΟΝΙΣΜΟΥ (ΕΕ) 2016/679 ΓΙΑ ΤΗΝ ΠΡΟΣΤΑΣΙΑ ΤΩΝ ΠΡΟΣΩΠΙΚΩΝ ΔΕΔΟΜΕΝΩΝ,” 2021, Τμήμα Ιατρικής Σχολής Επιστημών Υγείας, Πανεπιστήμιο Θεσσαλίας, Πρόγραμμα Μεταπτυχιακών Σπουδών: Δεοντολογία και Ηθική στις Βιοϊατρικές Επιστήμες, https://ir.lib.uth.gr/xmlui/bitstream/handle/11615/55542/22697.pdf?sequence=1 (accessed Feb. 12, 2024).

[51] Stefanus Van Staden and N. J. Bidwell, “Localised Trust in a Globalised Knot: Designing Information Privacy for Digital-ID,” Aug. 2023, doi: https://doi.org/10.1145/3616024.

[52] S. Hoffman, “Social credit : technology-enhanced authoritarian control with global consequences,” *Trove*, 2018. https://nla.gov.au/nla.obj-752973059 (accessed April 12, 2024).

[53] “Self-Sovereign Identity: How it Works and How it Impacts our Lives,” *www.adnovum.com*. Available: https://www.adnovum.com/blog/self-sovereign-identity-ssi-switzerland. [Accessed: Jun. 20, 2024]

[54] “Decentralized Identifiers (DIDs) v1.0,” *W3.org*, Jul. 19, 2022. Available: https://www.w3.org/TR/did-core/#independent-control. [Accessed: Jun. 20, 2024]

[55] SSL2BUY, “Symmetric vs. Asymmetric Encryption – What Are differences?,” *SSL2BUY Wiki - Get Solution for SSL Certificate Queries*, Feb. 07, 2019. Available: https://www.ssl2buy.com/wiki/symmetric-vs-asymmetric-encryption-what-are-differences

[56] S. Chandra, S. Bhattacharyya, S. Paira, and S. S. Alam, “A study and analysis on symmetric cryptography,” *2014 International Conference on Science Engineering and Management Research (ICSEMR)*, Nov. 2014, doi: https://doi.org/10.1109/icsemr.2014.7043664.

[57] A. M. Abdullah , “(PDF) Advanced Encryption Standard (AES) Algorithm to Encrypt and Decrypt Data,” *ResearchGate*. https://www.researchgate.net/publication/317615794_Advanced_Encryption_Standard_AES_Algorithm_to_Encrypt_and_Decrypt_Data

[58] M. E. Hellman, “An overview of public key cryptography,” *IEEE Communications Magazine*, vol. 40, no. 5, pp. 42–49, May 2002, doi: https://doi.org/10.1109/mcom.2002.1006971.

[59] M. Preetha and M. Nithya, “A STUDY AND PERFORMANCE ANALYSIS OF RSA ALGORITHM,” *IJCSMC*, vol. 2, no. 6, pp. 126–139, 2013, Available: https://www.ijcsmc.com/docs/papers/June2013/V2I6201330.pdf

[60] Y. Yan, “The Overview of Elliptic Curve Cryptography (ECC),” *Journal of Physics: Conference Series*, vol. 2386, no. 1, p. 012019, Dec. 2022, doi: https://doi.org/10.1088/1742-6596/2386/1/012019.

[61] “What is Public Key Cryptography?,” *Twilio*. Available: https://www.twilio.com/en-us/blog/what-is-public-key-cryptography

[62] D. Nuñez, I. Agudo, and J. Lopez, “Proxy Re-Encryption: Analysis of constructions and its application to secure access delegation,” *Journal of Network and Computer Applications*, vol. 87, pp. 193–209, Jun. 2017, doi: https://doi.org/10.1016/j.jnca.2017.03.005.

[63] K. O.-B. O. Agyekum, Q. Xia, E. B. Sifah, C. N. A. Cobblah, H. Xia, and J. Gao, “A Proxy Re-Encryption Approach to Secure Data Sharing in the Internet of Things Based on Blockchain,” *IEEE Systems Journal*, vol. 16, no. 1, pp. 1685–1696, Mar. 2022, doi: https://doi.org/10.1109/jsyst.2021.3076759.

[64] M. Alsayegh, T. Moulahi, A. Alabdulatif, and P. Lorenz, “Towards Secure Searchable Electronic Health Records Using Consortium Blockchain,” *Network*, vol. 2, no. 2, pp. 239–256, Apr. 2022, doi: https://doi.org/10.3390/network2020016

[65] D. Loshin, “Quick Reference Guide - Event Stream Processing,” p. 347, Jan. 2013, doi: https://doi.org/10.1016/b978-0-12-385889-4.00021-1.

[66] Z. Milosevic, W. Chen, A. Berry, and F. A. Rabhi, “Real-Time Analytics,” *Big Data*, pp. 39–61, 2016, doi: https://doi.org/10.1016/b978-0-12-805394-2.00002-7.

[67] D. Loshin, “Quick Reference Guide - Big Data/Metadata Analysis,” pp. 348, 349–350, Jan. 2013, doi: https://doi.org/10.1016/b978-0-12-385889-4.00021-1.

[68] D. Loshin, “Quick Reference Guide - Analytics Appliance ,” pp. 333, 334, Jan. 2013, doi: https://doi.org/10.1016/b978-0-12-385889-4.00021-1.

[69] S. Bonner, I. Kureshi, J. Brennan, and G. Theodoropoulos, “Exploring the Evolution of Big Data Technologies,” *Software Architecture for Big Data and the Cloud*, pp. 253–283, 2017, doi: https://doi.org/10.1016/b978-0-12-805467-3.00014-4.

[70] “What is Event Stream Processing? How & When to Use It,” *Hazelcast*. Available: https://hazelcast.com/glossary/event-stream-processing/

[71] F. Doglio, “What is a Decentralized Database?,” *Medium*, Jun. 22, 2022. Available: https://blog.bitsrc.io/what-is-a-decentralized-database-8ed4edfdc743

[72] R. Li, X. Dong, X. Gu, Z. Xue, and K. Li, “System Optimization for Big Data Processing,” *Elsevier eBooks*, p. 217, Jan. 2016, doi: https://doi.org/10.1016/b978-0-12-805394-2.00009-x.

[73] D. Agrawal, A. El Abbadi, S. Das, and A. J. Elmore, “Database Scalability, Elasticity, and Autonomy in the Cloud,” *Database Systems for Advanced Applications*, pp. 2–15, 2011, doi: https://doi.org/10.1007/978-3-642-20149-3_2.

[74] T. I. Martiny, “Privacy in Centralized Systems,” *scholar.colorado.edu*. https://scholar.colorado.edu/concern/graduate_thesis_or_dissertations/2801ph58q (accessed May 17, 2024).

[75] M. Pincheira, E. Donini, M. Vecchio, and S. Kanhere, “A Decentralized Architecture for Trusted Dataset Sharing Using Smart Contracts and Distributed Storage,” *Sensors*, vol. 22, no. 23, p. 9118, Nov. 2022, doi: https://doi.org/10.3390/s22239118.

[76] H. Rababaah and H. Hakimzadeh, “DISTRIBUTED DATABASES FUNDAMENTALS AND RESEARCH Advanced Database -B561. Spring 2005.” Available: https://clas.iusb.edu/computer-science-informatics/research/reports/TR-20050525-1.pdf

[77] P. Tomar, “An Overview of Distributed Databases,” *International Journal of Information and Computation Technology*, vol. 4, no. 2, pp. 207–214, 2014, Available: https://www.ripublication.com/irph/ijict_spl/ijictv4n2spl_15.pdf

[78] G. Kuntal Saroha Swati, “Fundamental Research of Distributed Database.,” 2011. https://www.semanticscholar.org/paper/Fundamental-Research-of-Distributed-Database-Gupta-Saroha/f9351f0cf3c4307dd76c85d6815e2a1b8095324b#citing-papers

[79] “Distributed Databases: A brief introduction and a high level walk through,” *www.linkedin.com*. Available: https://www.linkedin.com/pulse/distributed-databases-brief-introduction-high-level-walk-chakrabarti

[80] A. Lakshman, F. Prashant, and M. Facebook, “Cassandra -A Decentralized Structured Storage System.”

[81] D. Featherston, “Cassandra: Principles and Application.”

[6882] A. Mrs, Namrata Khandhar, and M. Shah, “Docker -The Future of Virtualization,” 2019.

[83] M. Odersky *et al.*, “An Overview of the Scala Programming Language.”

[84] J. Bachrach *et al.*, “Chisel: Constructing Hardware in a Scala Embedded Language.”

[85] Z. Wang *et al.*, “Kafka and Its Using in High-throughput and Reliable Message Distribution,” *2015 8th International Conference on Intelligent Networks and Intelligent Systems (ICINIS)*, Nov. 2015, doi: https://doi.org/10.1109/icinis.2015.53.

[86] J. Kan, J. Zhang, D. Liu, and X. Huang, “Proxy Re-Encryption Scheme for Decentralized Storage Networks,” *Applied Sciences*, vol. 12, no. 9, p. 4260, Apr. 2022, doi: https://doi.org/10.3390/app12094260.

[87] G. Ateniese, K. Fu, M. Green, and S. Hohenberger, “Improved proxy re-encryption schemes with applications to secure distributed storage,” *ACM Transactions on Information and System Security*, vol. 9, no. 1, pp. 1–30, Feb. 2006, doi: https://doi.org/10.1145/1127345.1127346.

[88] Baeldung and M. Aibin, “Cluster, Datacenters, Racks and Nodes in Cassandra,” Jan. 08, 2024. Available: https://www.baeldung.com/cassandra-cluster-datacenters-racks-nodes

[89] S. Bergman, M. Asplund, and S. Nadjm‐Tehrani, “Permissioned blockchains and distributed databases: A performance study,” *Concurrency and Computation: Practice and Experience*, vol. 32, no. 12, Mar. 2019, doi: https://doi.org/10.1002/cpe.5227.

[90] S. Syed, M. Ciampi, J. Bautista, and A. Khurshid, “OPEN ACCESS EDITED BY.”

[91] “Kafka Queue: Your 101 Solution for Efficient Message Processing,” *ProjectPro*. Available: https://www.projectpro.io/article/kafka-queue/932. [Accessed: Jun. 20, 2024]

[92] NCC Group, “Proxy Re-Encryption Protocol IronCore Labs ©2018 -NCC Group,” NCC Group, 2018. Accessed: May. 07, 2024. [Online]. Available: https://research.nccgroup.com/2018/08/17/proxy-re-encryption-protocol-ironcore-public-report/

[93] E. Bandara, X. Liang, P. Foytik, S. Shetty, and K. D. Zoysa, “A Blockchain and Self-Sovereign Identity Empowered Digital Identity Platform,” *2021 International Conference on Computer Communications and Networks (ICCCN)*, Jul. 2021, doi: https://doi.org/10.1109/icccn52240.2021.9522184.

[94] Y. Bai, H. Lei, S. Li, H. Gao, J. Li, and L. Li, “Decentralized and Self-Sovereign Identity in the Era of Blockchain: A Survey.”

[95] A. Abid, S. Cheikhrouhou, S. Kallel, and M. Jmaiel, “A Blockchain-Based Self-Sovereign Identity Approach for Inter-Organizational Business Processes,” *Annals of Computer Science and Information Systems*, Sep. 2022, doi: https://doi.org/10.15439/2022f194.

[96] A. De Salve, D. Di Francesco Maesa, P. Mori, L. Ricci, and A. Puccia, “A multi-layer trust framework for Self Sovereign Identity on blockchain,” *Online Social Networks and Media*, vol. 37–38, p. 100265, Sep. 2023, doi: https://doi.org/10.1016/j.osnem.2023.100265.

[97] Aditya Goel, “Deep Dive with Cassandra || Part-1,” *Medium*, Jan. 14, 2024. Available: https://adityagoel123.medium.com/deep-dive-with-cassandra-part-1-885d95859729. [Accessed: Jun. 20, 2024]

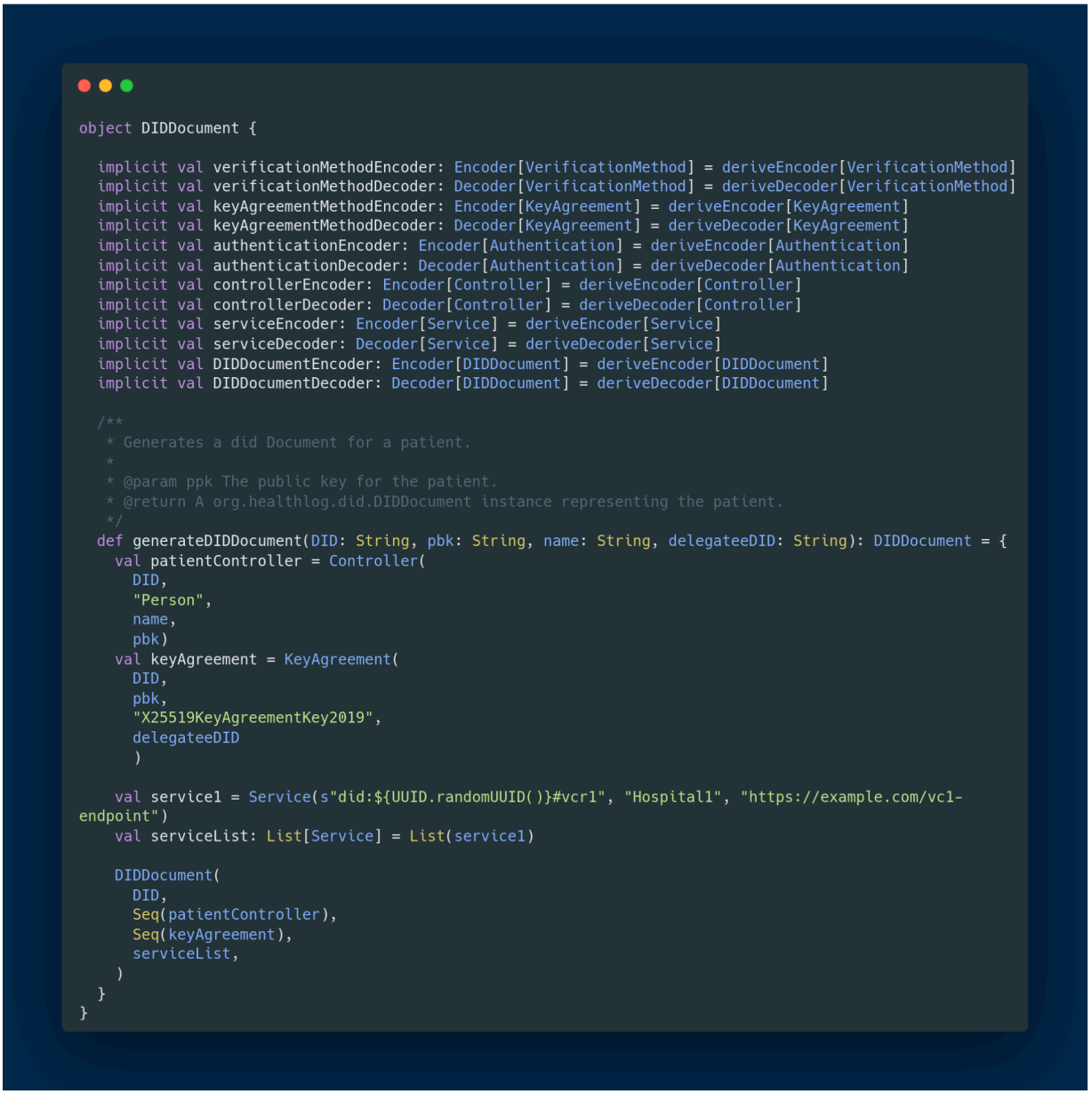
[98] Notomoro, “16 Disadvantages of Blockchain: Limitations and Challenges - Webisoft Blog,” *Webisoft*, Feb. 13, 2024. Available: https://webisoft.com/articles/disadvantages-of-blockchain/

Παραρτήματα

Αποσπάσματα Κώδικα

Κώδικας μονάδας DID

```
package org.healthlog.did

import io.circe._
import io.circe.generic.semiauto._

import java.util.UUID

/** declaration of did components */
case class Controller(id: String, `type`: String, name: String, MultibasePublicKey: String)

case class KeyAgreement(id: String, MultibasePublicKey: String, `type`: String, delegatee: String)

case class VerificationMethod(id: String, `type`: String, controller: String, MultibasePublicKey:
String)
case class Authentication(`type`: String, MultibasePublicKey: String)

case class Service(id: String, `type`: String, serviceEndpoint: String)
```



```
object DIDDocument {

  implicit val verificationMethodEncoder: Encoder[VerificationMethod] = deriveEncoder[VerificationMethod]
  implicit val verificationMethodDecoder: Decoder[VerificationMethod] = deriveDecoder[VerificationMethod]
  implicit val keyAgreementMethodEncoder: Encoder[KeyAgreement] = deriveEncoder[KeyAgreement]
  implicit val keyAgreementMethodDecoder: Decoder[KeyAgreement] = deriveDecoder[KeyAgreement]
  implicit val authenticationEncoder: Encoder[Authentication] = deriveEncoder[Authentication]
  implicit val authenticationDecoder: Decoder[Authentication] = deriveDecoder[Authentication]
  implicit val controllerEncoder: Encoder[Controller] = deriveEncoder[Controller]
  implicit val controllerDecoder: Decoder[Controller] = deriveDecoder[Controller]
  implicit val serviceEncoder: Encoder[Service] = deriveEncoder[Service]
  implicit val serviceDecoder: Decoder[Service] = deriveDecoder[Service]
  implicit val DIDDocumentEncoder: Encoder[DIDDocument] = deriveEncoder[DIDDocument]
  implicit val DIDDocumentDecoder: Decoder[DIDDocument] = deriveDecoder[DIDDocument]

  /**
   * Generates a did Document for a patient.
   *
   * @param ppk The public key for the patient.
   * @return A org.healthlog.did.DIDDocument instance representing the patient.
   */
  def generateDIDDocument(DID: String, pbk: String, name: String, delegateeDID: String): DIDDocument = {
    val patientController = Controller(
      DID,
      "Person",
      name,
      pbk)
    val keyAgreement = KeyAgreement(
      DID,
      pbk,
      "X25519KeyAgreementKey2019",
      delegateeDID
      )

    val service1 = Service(s"did:${UUID.randomUUID()}#vcr1", "Hospital1", "https://example.com/vc1-
endpoint")
    val serviceList: List[Service] = List(service1)

    DIDDocument(
      DID,
      Seq(patientController),
      Seq(keyAgreement),
      serviceList,
    )
  }
}
```

```scala
package org.healthlog.did

import org.healthlog

import scala.collection.mutable

case class Holder(name: String, DID: String)

object HolderStorage {
  private val didDocuments: mutable.Map[String, DIDDocument] =
mutable.Map.empty
  def tempStoreDIDDocument(holder: Holder, document: DIDDocument): Unit = {
    didDocuments += (holder.DID -> document)
  }

  def getDIDDocument(did: String): Option[healthlog.did.DIDDocument] = {
    didDocuments.get(did)
  }
}

object Holder {
  def requestDID(holder: Holder): Option[DIDDocument] = {
    HolderStorage.getDIDDocument(holder.DID)
  }
}
```

Κώδικας Επανακρυπτογράφησης

```scala
package org.healthlog.proxy

import cats.effect.IO
import com.ironcorelabs.recrypt.{CoreApi, EncryptedValue, Plaintext, PrivateKey, PrivateSigningKey,
PublicKey, PublicSigningKey, TransformKey}
import org.healthlog.signature.SignaturePipeline

object ProxyPipeline {

  /**
   * Perform a series of operations to proxy re-encrypt a message for a receiver using the given CoreApi
instance.
   *
   * @param coreApi           An instance of CoreApi for performing encryption, transformation, and other
cryptographic operations.
   * @param receiverPublicKey The public key of the receiver who will decrypt the transformed message.
   * @return An IO containing the re-encrypted message as an EncryptedValue.
   */
  def proxyJobs(
                 coreApi: CoreApi,
                 symmetric_key: Plaintext,
                 senderPrivateKey: PrivateKey,
                 senderPublicKey: PublicKey,
                 receiverPublicKey: PublicKey
               ): IO[EncryptedValue] = {

    //TODO move SIGNING to separate module
    val PUSK: PublicSigningKey = SignaturePipeline.senderPublicSigningKey
    val PRSK: PrivateSigningKey = SignaturePipeline.senderPrivateSigningKey

    val encryptedValue: IO[EncryptedValue] = for {
      encryptedValue <- coreApi.encrypt(symmetric_key, senderPublicKey, PUSK, PRSK)
    } yield encryptedValue

    val transformKey: IO[TransformKey] = for {
      transformKey <- coreApi.generateTransformKey(senderPrivateKey, receiverPublicKey, PUSK, PRSK)
    } yield transformKey

    val transformedValue: IO[EncryptedValue] = for {
      encryptedValue <- encryptedValue
      transformKey <- transformKey
      transformedValue <- coreApi.transform(encryptedValue, transformKey, PUSK, PRSK)
    } yield transformedValue

    transformedValue
  }
}
```

Κώδικας Ενσωμάτωσης Cassandra

```
package org.healthlog.cassandra

import cats.data.NonEmptyVector
import cats.implicits.toFoldableOps
import com.datastax.oss.driver.api.core.CqlSession
import com.datastax.oss.driver.api.core.cql.{BoundStatement, PreparedStatement, ResultSet,
SimpleStatement}
import com.ironcorelabs.recrypt._
import com.typesafe.config.ConfigFactory
import io.circe.generic.codec.DerivedAsObjectCodec.deriveCodec
import io.circe.syntax.EncoderOps
import io.circe.{Decoder, HCursor, Json, parser}
import org.healthlog.did._
import org.healthlog.key.Asymmetric
import scodec.bits.ByteVector

import java.io.IOException
import java.net.InetSocketAddress
import java.nio.ByteBuffer
import java.time.Instant
import java.util.UUID

private object CassandraConfig {
  private val config = ConfigFactory.load()

  val cassandraHost: String = config.getString("cassandra.host")
  val cassandraPort: Int = config.getInt("cassandra.port")
  val cassandraDatacenter: String = config.getString("cassandra.datacenter")
}
```

```
object DIDRegistry {

  private val cassandraHost = new InetSocketAddress(
    CassandraConfig.cassandraHost,
    CassandraConfig.cassandraPort
  )
  private val localDatacenter =
```

```scala
  /**
   * Adds a Decentralized Identifier (did) and associated information to the did Registry.
   *
   * @param subject The holder representing the entity for which the did is being added.
   * @return Unit
   */
  def addDID(DID: String, DIDType: String, privateKey: PrivateKey, publicKey: PublicKey, name: String):
Unit = {
    val keyspaceName = "decentralized_identity_keyspace"
    val insertDidCql = s"INSERT INTO $keyspaceName.did_registry (" +
      s" did," +
      s" name," +
      s" type," +
      s" public_key," +
      s" private_key_hash," +
      s" created_at" +
      s") VALUES (?, ?, ?, ?, ?, ?);"
    val session =
CqlSession.builder().addContactPoint(cassandraHost).withLocalDatacenter(localDatacenter).build()

    try {
      val preparedStatement: PreparedStatement = session.prepare(insertDidCql)

      val final_DID = s"did:e_health:$DID"
      val privateKeyHash: String = Asymmetric.haskKey(privateKey.toString)
      val encodedPublicKey: String = Asymmetric.encodePublicKey(publicKey)
      val createdAt: Instant = Instant.now()

      val boundStatement: BoundStatement = preparedStatement.bind(
        final_DID,
        name,
        DIDType,
        encodedPublicKey,
        privateKeyHash,
        createdAt
      )
      session.execute(boundStatement)
      println(s"did $DID inserted into $keyspaceName.did_registry successfully.")
    } finally {
      session.close()
    }
  }
```

```
  def addDIDDocument(DID: String, document: DIDDocument): Unit = {
    val keyspaceName = "decentralized_identity_keyspace"
    val insertDidCql = s"INSERT INTO $keyspaceName.did_documents_registry (" +
      s" did," +
      s" document," +
      s" created_at" +
      s") VALUES (?, ?, ?);"
    val session =
CqlSession.builder().addContactPoint(cassandraHost).withLocalDatacenter(localDatacenter).build()

    try {
      val preparedStatement: PreparedStatement = session.prepare(insertDidCql)

      val json: String = document.asJson.noSpaces
      val createdAt: Instant = Instant.now()

      val boundStatement: BoundStatement = preparedStatement.bind(
        DID,
        json,
        createdAt
      )
      session.execute(boundStatement)
      println(s"DID document for $DID inserted into $keyspaceName.did_documents_registry successfully.")
    } finally {
      session.close()
    }
  }
```

```
  /**
   * Adds data related to a corresponding Decentralized Identifier (did) Document to the Healthcare Data
KeySpace.
   *
   * @param subject The holder representing the entity for which the data is being added.
   * @return Unit
   */
  def addHealthcareData(subject: Holder): Unit = {

    val keyspaceName = "healthcare_data_keyspace"
    val insertHealthcareDataCql = s"INSERT INTO $keyspaceName.healthcare_data (" +
      s" did," +
      s" record_id," +
      s" last_updated_at" +
      s") VALUES (?, ?, ?);"
    val session =
CqlSession.builder().addContactPoint(cassandraHost).withLocalDatacenter(localDatacenter).build()

    try {
      val healthcareDataPreparedStatement: PreparedStatement = session.prepare(insertHealthcareDataCql)

      val retrievedDID: Option[DIDDocument] = Holder.requestDID(subject)
      val json: Json = retrievedDID.fold(Json.Null)(_.asJson)

      val did: String = json.hcursor.downField("id").as[String].getOrElse("")
      val recordId: UUID = UUID.randomUUID()
      val lastUpdatedAt: Instant = Instant.now()

      val healthcareDataBoundStatement: BoundStatement = healthcareDataPreparedStatement.bind(
        did,
        recordId,
        lastUpdatedAt
      )
      session.execute(healthcareDataBoundStatement)
      println(s"Healthcare data inserted for user $did and record $recordId into
$keyspaceName.healthcare_data successfully.")
    } finally {

      session.close()
    }
  }
```

```scala
  /**
   * Adds an access control policy to the access control keyspace.
   *
   * @param delegator The holder granting access.
   * @param delegatee The holder being granted access.
   */
  def addAccessControl(delegator: String, delegatee: String): Unit = {
    val keyspaceName = "access_control_keyspace"
    val insertAccessControlCql = s"INSERT INTO $keyspaceName.access_control_policies (" +
      s" allowed_user_did," +
      s" delegator_user_did," +
      s" last_updated_at" +
      s") VALUES (?, ?, ?);"
    val session =
CqlSession.builder().addContactPoint(cassandraHost).withLocalDatacenter(localDatacenter).build()

    try {
      val accessControlPreparedStatement: PreparedStatement = session.prepare(insertAccessControlCql)

      val lastUpdatedAt: Instant = Instant.now()

      val accessControlBoundStatement: BoundStatement = accessControlPreparedStatement.bind(
        delegatee,
        delegator,
        lastUpdatedAt
      )
      session.execute(accessControlBoundStatement)
      println(s"User $delegatee has access on data of user $delegator")
    } finally {
      session.close()
    }
  }
```

```scala
  /**
   * Adds an audit trail entry to the audit trails key space.
   *
   * @param source     The did representing the source entity of the action.
   * @param target     The did representing the target entity of the action.
   * @param actionType The type of action being audited.
   * @return Unit
   */
  def addAuditTrail(source: String, target: String, actionType: String): Unit = {
    val keyspaceName = "audit_trail_keyspace"
    val insertAuditTrailCql =
      s"INSERT INTO $keyspaceName.audit_trails (" +
        s" FROM_did," +
        s" TO_did," +
        s" action_type," +
        s" timestamp" +
        s") VALUES (?, ?, ?, ?);"
    val session =
CqlSession.builder().addContactPoint(cassandraHost).withLocalDatacenter(localDatacenter).build()

    try {
      val auditTrailPreparedStatement: PreparedStatement = session.prepare(insertAuditTrailCql)

      val timestamp: Instant = Instant.now()

      val auditTrailBoundStatement: BoundStatement = auditTrailPreparedStatement.bind(
        source,
        target,
        actionType,
        timestamp
      )
      session.execute(auditTrailBoundStatement)
      println(s"Audit trail recorded: $actionType from $source to $target at $timestamp")
    } finally {
      session.close()
    }
  }
```

```
  /**
   * Inserts delegation data into the Cassandra database.
   *
   * @param delegator        The ID of the delegator.
   * @param delegatee        The ID of the delegatee.
   * @param transformedValue The transformed value associated with the delegation.
   * @param document         The DID document associated with the delegation.
   */
  def addDelegationData(delegator: String, delegatee: String, transformedValue: TransformedValue,
document: DIDDocument): Unit = {
    val keyspaceName = "delegation_data_keyspace"
    val session =
CqlSession.builder().addContactPoint(cassandraHost).withLocalDatacenter(localDatacenter).build()

    try {
      val transformBlocksJson = transformedValue.transformBlocks.toList.map { block =>
        Json.obj(
          "public_key" -> Json.fromString(block.publicKey.toString),
          "encrypted_temp_key" -> Json.fromString(block.encryptedTempKey.bytes.toHex),
          "random_transform_public_key" -> Json.fromString(block.randomTransformPublicKey.toString),
          "random_transform_encrypted_temp_key" ->
Json.fromString(block.randomTransformEncryptedTempKey.bytes.toHex)
        )
      }
      val timestamp: Instant = Instant.now()
      val insertDelegationDataCql =
        s"INSERT INTO $keyspaceName.delegation_data (" +
          s" delegator_id," +
          s" delegatee_id," +
          s" document," +
          s" ephemeral_public_key_x," +
          s" ephemeral_public_key_y," +
          s" encrypted_message," +
          s" auth_hash," +
          s" public_signing_key," +
          s" signature," +
          s" transform_blocks," +
          s" created_at" +
          s") VALUES (?, ?, ?, ?, ?, ?, ?, ?, ?, ?, ?);"
      val insertDelegationDataPreparedStatement: PreparedStatement =
session.prepare(insertDelegationDataCql)
      val insertDelegationDataBoundStatement: BoundStatement =
insertDelegationDataPreparedStatement.bind(
        delegator,
        delegatee,
        document.asJson.noSpaces,
        transformedValue.ephemeralPublicKey.x.toHex,
        transformedValue.ephemeralPublicKey.y.toHex,
        transformedValue.encryptedMessage.bytes.toHex,
        transformedValue.authHash.bytes.toHex,
        transformedValue.publicSigningKey.toString,
        transformedValue.signature.bytes.toHex,
        Json.fromValues(transformBlocksJson).noSpaces,
        timestamp
      )

      session.execute(insertDelegationDataBoundStatement)
      println(s"Delegation data inserted for delegator $delegator and delegatee $delegatee into
$keyspaceName.delegation_data successfully.")
    } finally {
      session.close()
    }
  }
```

```
  def addEncryptedData(DID: String, encrypted_data: Array[Byte], fileName: String): Unit = {
    val keyspaceName = "healthcare_data_keyspace"
    val session =
CqlSession.builder().addContactPoint(cassandraHost).withLocalDatacenter(localDatacenter).build()

    try {

      val uuid: UUID = UUID.randomUUID()
      val timestamp: Instant = Instant.now()

      val insertPDFQuery = s"INSERT INTO $keyspaceName.healthcare_data (" +
        s" did," +
        s" record_id," +
        s" file_name," +
        s" data," +
        s" created_at) " +
        s" VALUES (?, ?, ?, ?, ?);"

      val preparedStatement = session.prepare(insertPDFQuery)
      val boundStatement = preparedStatement.bind(DID, uuid, fileName, ByteBuffer.wrap(encrypted_data),
timestamp)

      session.execute(boundStatement)
      println(s"File with name'$fileName' saved successfully in Cassandra.")
    } catch {
      case e: IOException =>
        println(s"Failed to read PDF file: ${e.getMessage}")
    } finally {
      session.close()
    }
  }
```

```
  /**
   * Extracts the public key field from inside the did document stored in the decentralized identity
keyspace.
   *
   * @param did The Decentralized Identifier (did) for which the public key is to be extracted.
   * @return An optional String containing the public key, or None if the did document or public key is
not found.
   */
  def getPublicKeyFromDID(did: String): String = {
    val keyspaceName = "decentralized_identity_keyspace"
    val selectPublicKeyCql = s"SELECT public_key FROM $keyspaceName.did_registry WHERE did = ?;"

    val session =
CqlSession.builder().addContactPoint(cassandraHost).withLocalDatacenter(localDatacenter).build()

    try {
      val selectPublicKeyPreparedStatement = session.prepare(selectPublicKeyCql)
      val selectPublicKeyBoundStatement = selectPublicKeyPreparedStatement.bind(did)
      val resultSet = session.execute(selectPublicKeyBoundStatement)

      val row = resultSet.one()
      if (row != null) {
        row.getString("public_key")
      } else {
        throw new RuntimeException(s"Public key not found for did: $did")
      }
    } finally {
      session.close()
    }
  }
```

```scala
  /**
   * Retrieves TransformedValue from the database for a given delegator and delegatee.
   *
   * This method queries the database to fetch the TransformedValue associated with the specified
delegator
   * and delegatee IDs. It constructs the TransformedValue object using the retrieved data and returns it
   * wrapped in an Option. If no matching record is found in the database, it returns None.
   *
   * @param delegator The ID of the delegator.
   * @param delegatee The ID of the delegatee.
   * @return An Option containing the TransformedValue if found, otherwise None.
   */
  def getTransformedValue(delegator: String, delegatee: String): Option[TransformedValue] = {
    val keyspaceName = "delegation_data_keyspace"
    val selectDelegationDataCql =
      s"""
         |SELECT ephemeral_public_key_x, ephemeral_public_key_y, encrypted_message, auth_hash,
public_signing_key, signature, transform_blocks
         |FROM $keyspaceName.delegation_data
         |WHERE delegator_id = ? AND delegatee_id = ?;
         """.stripMargin

    val session =
CqlSession.builder().addContactPoint(cassandraHost).withLocalDatacenter(localDatacenter).build()

    try {
      val preparedStatement: PreparedStatement = session.prepare(selectDelegationDataCql)
      val boundStatement: BoundStatement = preparedStatement.bind(delegator, delegatee)
      val resultSet: ResultSet = session.execute(boundStatement)

      val row = resultSet.one()
      if (row != null) {
        val ephemeralPublicKeyXStr = row.getString("ephemeral_public_key_x")
        val ephemeralPublicKeyYStr = row.getString("ephemeral_public_key_y")
        val encryptedMessageStr = row.getString("encrypted_message")
        val authHashStr = row.getString("auth_hash")
        val publicSigningKeyStr = row.getString("public_signing_key")
        val signatureStr = row.getString("signature")
        val transformBlocksJsonStr = row.getString("transform_blocks")

        implicit val transformBlockDecoder: Decoder[TransformBlock] = (c: HCursor) => {
          for {
            publicKey <- c.downField("public_key").as[String]
            encryptedTempKey <- c.downField("encrypted_temp_key").as[String]
            randomTransformPublicKey <- c.downField("random_transform_public_key").as[String]
            randomTransformEncryptedTempKey <-
c.downField("random_transform_encrypted_temp_key").as[String]
          } yield TransformBlock(
            PublicKey(
              ByteVector.fromValidHex(publicKey.substring(33, 97)),
              ByteVector.fromValidHex(publicKey.substring(122, 186)))
              .getOrElse(throw new RuntimeException("Error parsing transform blocks JSON for first public
key")),
            EncryptedElement(ByteVector.fromValidHex(encryptedTempKey)),
            PublicKey(
              ByteVector.fromValidHex(randomTransformPublicKey.substring(33, 97)),
              ByteVector.fromValidHex(randomTransformPublicKey.substring(122, 186)))
              .getOrElse(throw new RuntimeException("Error parsing transform blocks JSON for second
public key")),
            EncryptedElement(ByteVector.fromValidHex(randomTransformEncryptedTempKey))
          )
        }

        val transformBlocks: NonEmptyVector[TransformBlock] = parser.parse(transformBlocksJsonStr)
          .flatMap(_.as[NonEmptyVector[TransformBlock]])
          .getOrElse(throw new RuntimeException("Error parsing transform blocks JSON"))
        val ephemeralPublicKey = for {
          x <- ByteVector.fromHex(ephemeralPublicKeyXStr)
          y <- ByteVector.fromHex(ephemeralPublicKeyYStr)
        } yield PublicKey(x, y).getOrElse(throw new RuntimeException("Error constructing ephemeral public
key"))
        val encryptedMessage = EncryptedMessage(ByteVector.fromValidHex(encryptedMessageStr))
        val authHash = AuthHash(ByteVector.fromValidHex(authHashStr))
        val publicSigningKey = PublicSigningKey(ByteVector.fromValidHex(publicSigningKeyStr.substring(40,
104)))
        val signature = Signature(ByteVector.fromValidHex(signatureStr))

        Some(TransformedValue(ephemeralPublicKey.get, encryptedMessage, authHash, transformBlocks,
publicSigningKey, signature))
      } else {
        None
      }
    } finally {
      session.close()
    }
  }
```

```
  def getNameKeyFromDID(did: String): String = {
    val keyspaceName = "decentralized_identity_keyspace"
    val selectPublicKeyCql = s"SELECT name FROM $keyspaceName.did_registry WHERE did = ?;"

    val session =
CqlSession.builder().addContactPoint(cassandraHost).withLocalDatacenter(localDatacenter).build()

    try {
      val selectPublicKeyPreparedStatement = session.prepare(selectPublicKeyCql)
      val selectPublicKeyBoundStatement = selectPublicKeyPreparedStatement.bind(did)
      val resultSet = session.execute(selectPublicKeyBoundStatement)

      val row = resultSet.one()
      if (row != null) {
        row.getString("name")
      } else {
        throw new RuntimeException(s"Name key not found for did: $did")
      }
    } finally {
      session.close()
    }
  }
```

```
  /**
   * Retrieves the stored private key hash for a given did from the decentralized identity keyspace.
   *
   * @param did The Decentralized Identifier (did) for which the private key hash is to be retrieved.
   * @return An optional String containing the stored private key hash, or None if the did is not found
or the hash is null.
   */
  def getStoredPrivateKeyHash(did: String): Option[String] = {
    val keyspaceName = "decentralized_identity_keyspace"
    val selectPrivateKeyHashCql =
      s"SELECT private_key_hash FROM $keyspaceName.did_registry WHERE did = ?;"
    val session =
CqlSession.builder().addContactPoint(cassandraHost).withLocalDatacenter(localDatacenter).build()

    try {
      val selectPrivateKeyHashPreparedStatement: PreparedStatement =
session.prepare(selectPrivateKeyHashCql)

      val selectPrivateKeyHashBoundStatement: BoundStatement =
selectPrivateKeyHashPreparedStatement.bind(did)
      val resultSet = session.execute(selectPrivateKeyHashBoundStatement)

      val row = resultSet.one()
      if (row != null && !row.isNull("private_key_hash")) {
        Some(row.getString("private_key_hash"))
      } else {
        None
      }
    } catch {
      case ex: Exception =>
        println(s"Error retrieving private key hash for did $did: ${ex.getMessage}")
        None
    } finally {
      session.close()
    }
  }
```

```scala
  /**
   * Retrieves access control policies from the database.
   *
   * This method retrieves access control policies from the database for a given delegator and delegatee.
   * It constructs an SQL SELECT statement to fetch the data based on the provided delegator and
delegatee IDs.
   * After executing the statement, it prints a success message along with the retrieved access control
policies.
   *
   * @param delegatee The ID of the delegatee.
   * @return An optional tuple containing the fetched access control policies (did, allowed users, last
updated timestamp).
   */
  def getAccessControl(delegatee: String): Option[String] = {
    val keyspaceName = "access_control_keyspace"
    val selectAccessControlCql =
      s"""
         |SELECT delegator_user_did
         |FROM $keyspaceName.access_control_policies
         |WHERE allowed_user_did = ?;
         """.stripMargin

    val session =
CqlSession.builder().addContactPoint(cassandraHost).withLocalDatacenter(localDatacenter).build()

    try {
      val selectAccessControlPreparedStatement: PreparedStatement =
session.prepare(selectAccessControlCql)
      val selectAccessControlBoundStatement: BoundStatement =
selectAccessControlPreparedStatement.bind(delegatee)
      val resultSet: ResultSet = session.execute(selectAccessControlBoundStatement)

      val row = resultSet.one()
      if (row != null) {
        val delegator = row.getString("delegator_user_did")
        val accessControlTuple = (delegator)
        Some(accessControlTuple)
      } else {
        None
      }
    } finally {
      session.close()
    }
  }
```

```
  def getEncryptedData(did: String, recordId: UUID): Option[Array[Byte]] = {
    val keyspaceName = "healthcare_data_keyspace"
    val session =
CqlSession.builder().addContactPoint(cassandraHost).withLocalDatacenter(localDatacenter).build()

    try {
      val selectQuery =
        s"""
           |SELECT data
           |FROM $keyspaceName.healthcare_data
           |WHERE did = ? AND record_id = ?;
           """.stripMargin

      val preparedStatement: PreparedStatement = session.prepare(selectQuery)
      val boundStatement: BoundStatement = preparedStatement.bind(did, recordId)
      val resultSet: ResultSet = session.execute(boundStatement)

      val row = resultSet.one()
      Option(row).map { r =>
        val dataBuffer: ByteBuffer = r.getByteBuffer("data")
        val dataBytes: Array[Byte] = new Array[Byte](dataBuffer.remaining())
        dataBuffer.get(dataBytes)
        dataBytes
      }
    } catch {
      case e: Exception =>
        println(s"Failed to retrieve encrypted data: ${e.getMessage}")
        None
    } finally {
      session.close()
    }
  }
```

```
  /**
   * Prints the audit trail in a readable format for a specified did.
   *
   * @param did The Decentralized Identifier (did) for which the audit trail is to be printed.
   */
  def printAuditTrailForDID(did: String): ResultSet = {
    val keyspaceName = "audit_trail_keyspace"
    val selectAuditTrailCql =
      s"SELECT FROM_did, TO_did, action_type, timestamp " +
        s"FROM $keyspaceName.audit_trails " +
        s"WHERE TO_did = ?;"
    val session =
CqlSession.builder().addContactPoint(cassandraHost).withLocalDatacenter(localDatacenter).build()

    try {
      val selectAuditTrailPreparedStatement: PreparedStatement = session.prepare(selectAuditTrailCql)

      val selectAuditTrailBoundStatement: BoundStatement = selectAuditTrailPreparedStatement.bind(did)

      session.execute(selectAuditTrailBoundStatement)

    } catch {
      case ex: Exception =>
        println(s"Error retrieving audit trail for did $did: ${ex.getMessage}")
        throw ex
    } finally {
      session.close()
    }
  }
```

```
  /**
   * Truncate all tables in the Cassandra keyspaces related to the application.
   * This operation removes all data from the specified tables.
   */
  def truncateAll(): Unit = {
    val ksp1 = "decentralized_identity_keyspace"
    val ksp2 = "healthcare_data_keyspace"
    val ksp3 = "access_control_keyspace"
    val ksp4 = "audit_trail_keyspace"
    val ksp5 = "delegation_keyspace"
    val session =
CqlSession.builder().addContactPoint(cassandraHost).withLocalDatacenter(localDatacenter).build()

    try {
      val truncateStatement1_0: SimpleStatement = SimpleStatement.newInstance(s"TRUNCATE
$ksp1.did_registry;")
      session.execute(truncateStatement1_0)

      val truncateStatement1_1: SimpleStatement = SimpleStatement.newInstance(s"TRUNCATE
$ksp1.did_documents_registry;")
      session.execute(truncateStatement1_1)

      val truncateStatement2: SimpleStatement = SimpleStatement.newInstance(s"TRUNCATE
$ksp2.healthcare_data;")
      session.execute(truncateStatement2)

      val truncateStatement3: SimpleStatement = SimpleStatement.newInstance(s"TRUNCATE
$ksp3.access_control_policies;")
      session.execute(truncateStatement3)

      val truncateStatement4: SimpleStatement = SimpleStatement.newInstance(s"TRUNCATE
$ksp4.audit_trails;")
      session.execute(truncateStatement4)

      val truncateStatement5: SimpleStatement = SimpleStatement.newInstance(s"TRUNCATE
$ksp5.delegation_data7;")
      session.execute(truncateStatement5)

      println("All tables truncated successfully.")
    } catch {
      case ex: Exception =>
        println(s"Unexpected error: ${ex.getMessage}")
    } finally {
      session.close()
    }
  }

}
```

Κώδικας Διαμόρφωσης Docker

```
# Create Docker volumes for Cassandra data for all nodes
docker volume create cassandra_data_node_1
docker volume create cassandra_data_node_2
docker volume create cassandra_data_node_3

# Run Cassandra container for Node 1
docker run -d --name cassandra-node-1 \
  --hostname cassandra-node-1 \
  --network cassandra \
  -v cassandra_data_node_1:/var/lib/cassandra \
  -e CASSANDRA_SEEDS=cassandra-node-1,cassandra-node-2,cassandra-node-3
\ cassandra

# Run Cassandra container for Node 2
docker run -d --name cassandra-node-2 \
  --hostname cassandra-node-2 \
  --network cassandra \
  -v cassandra_data_node_2:/var/lib/cassandra \
  -e CASSANDRA_SEEDS=cassandra-node-1,cassandra-node-2,cassandra-node-3
\ cassandra

# Run Cassandra container for Node 3
docker run -d --name cassandra-node-3 \
  --hostname cassandra-node-3 \
  --network cassandra \
  -v cassandra_data_node_3:/var/lib/cassandra \
  -e CASSANDRA_SEEDS=cassandra-node-1,cassandra-node-2,cassandra-node-3
\ cassandra
```

```
#!/bin/bash

# Create a Docker volume for CQL scripts
docker volume create cql_scripts

# Run Docker container with mounted volume for CQL scripts for Node
docker run --rm --network cassandra \
  -v "$(pwd)/data.cql:/scripts/data.cql" \
  -v cql_scripts:/scripts \
  -e CQLSH_HOST=cassandra-node-1 \
  -e CQLSH_PORT=9042 \
  -e CQLVERSION=3.4.6 \
  nuvo/docker-cqlsh

# Run Docker container with mounted volume for CQL scripts for Node
docker run --rm --network cassandra \
  -v "$(pwd)/data.cql:/scripts/data.cql" \
  -v cql_scripts:/scripts \
  -e CQLSH_HOST=cassandra-node-2 \
  -e CQLSH_PORT=9042 \
  -e CQLVERSION=3.4.6 \
  nuvo/docker-cqlsh

# Run Docker container with mounted volume for CQL scripts for Node
docker run --rm --network cassandra \
  -v "$(pwd)/data.cql:/scripts/data.cql" \
  -v cql_scripts:/scripts \
  -e CQLSH_HOST=cassandra-node-3 \
  -e CQLSH_PORT=9042 \
  -e CQLVERSION=3.4.6 \
  nuvo/docker-cqlsh
```

Κώδικας Συμβάντων Kafka

```scala
package org.healthlog.kafka

import org.apache.kafka.clients.producer.{KafkaProducer, ProducerConfig, ProducerRecord}
import org.apache.kafka.common.serialization.StringSerializer

import java.util.Properties

//noinspection ScalaWeakerAccess
object Producer {

  /**
   * Produce a message to a Kafka topic.
   *
   * @param msg The message to be produced.
   * @return No specific return value; the message is sent to the Kafka topic.
   */
  def produce(msg: String): Unit = {
    val topic = "did-issuing-topic"
    val kafkaProps = new Properties()
    kafkaProps.put(ProducerConfig.BOOTSTRAP_SERVERS_CONFIG, "localhost:9092")
    kafkaProps.put(ProducerConfig.CLIENT_ID_CONFIG, "DIDIssuerProducer")
    kafkaProps.put(ProducerConfig.KEY_SERIALIZER_CLASS_CONFIG, classOf[StringSerializer].getName)
    kafkaProps.put(ProducerConfig.VALUE_SERIALIZER_CLASS_CONFIG,
classOf[StringSerializer].getName)
    val producer = new KafkaProducer[String, String](kafkaProps)

    val record = new ProducerRecord[String, String](topic, null, msg)

    producer.send(record)

    producer.close()
  }
}
```

```scala
package org.healthlog.kafka

import com.ironcorelabs.recrypt.{PrivateKey, PublicKey}
import org.apache.kafka.clients.consumer.{ConsumerConfig, ConsumerRecord, KafkaConsumer}
import org.apache.kafka.common.serialization.StringDeserializer
import org.healthlog.did.{DIDDocument, Holder, Issuer}

import java.util.{Collections, Properties}

sealed trait ConsumerResult

case class TypeIssueResult(DID: String, prvk: PrivateKey, pubk: PublicKey, sub: Holder) extends
ConsumerResult

case class TypeIssueDIDDOCResult(document: DIDDocument) extends ConsumerResult

case class TypeInvokeResult(trigger: String) extends ConsumerResult

//noinspection ScalaWeakerAccess
object Consumer {

  /**
   * Consumes messages from the Kafka topic "did-issuing-topic".
   *
   * @param name Optionally, a name to be used in message processing.
   * @return The result of consuming messages.
   */
  def consume(
               DID: Option[String] = None,
               encodedPublicKey: Option[String] = None,
               name: Option[String] = None,
               delegateeDID: Option[String] = None
             ): ConsumerResult = {
    val topic = "did-issuing-topic"
    val kafkaProps = new Properties()
    kafkaProps.put(ConsumerConfig.BOOTSTRAP_SERVERS_CONFIG, "localhost:9092")
    kafkaProps.put(ConsumerConfig.GROUP_ID_CONFIG, "DIDIssuerConsumer")
    kafkaProps.put(ConsumerConfig.KEY_DESERIALIZER_CLASS_CONFIG, classOf[StringDeserializer].getName)
    kafkaProps.put(ConsumerConfig.VALUE_DESERIALIZER_CLASS_CONFIG, classOf[StringDeserializer].getName)

    val consumer = new KafkaConsumer[String, String](kafkaProps)
    consumer.subscribe(Collections.singletonList(topic))
    var result: ConsumerResult = null

    try {
      val timeout = java.time.Duration.ofSeconds(5) // Consume messages for:...
      val startTime = java.time.Instant.now()

      while (java.time.Instant.now().isBefore(startTime.plus(timeout)))  {
        val records = consumer.poll(java.time.Duration.ofSeconds(1))
        records.forEach { record: ConsumerRecord[String, String] =>
          println(s"Received message: ${record.value()}")

          record.value() match {
            case "IssueDID" =>
              result = typeIssue(name)
            case "IssueDIDDocument" =>
              result = typeIssueDIDDOC(DID, encodedPublicKey, name, delegateeDID)
            case "InvokeProxyPipeline" =>
              result = typeINV("InvokeProxyPipeline")
            case "InvokeAccess" =>
              result = typeINV("InvokeAccess")
          }
        }
      }
    } finally {
      consumer.close()
    }
    result
  }

  /**
   * Process Type A message and generate a `TypeIssueResult`.
   *
   * @return A `TypeIssueResult` containing information related to Type A processing.
   */
  def typeIssue(name: Option[String]): TypeIssueResult = {
    Issuer.issueDID(name)
  }

  def typeIssueDIDDOC(DID: Option[String], encodedPublicKey: Option[String], name: Option[String],
delegateeDID: Option[String]): TypeIssueDIDDOCResult = {
    Issuer.issueDIDDocument(DID, encodedPublicKey, name, delegateeDID)
  }

  /**
   * Process Type INV (Function INVocation) message and generate a `TypeInvokeResult`.
   *  * @return A `TypeInvokeResult` containing information related to Type INV processing.
   */
  def typeINV(trigger: String): TypeInvokeResult = {
    trigger match {
      case "InvokeProxyPipeline" =>
        TypeInvokeResult("InvokeProxyPipeline")
      case "InvokeAccess" =>
        TypeInvokeResult("InvokeAccess")
    }
  }
}
```

```
package org.healthlog.kafka

import com.ironcorelabs.recrypt.{CoreApi, Plaintext, PrivateKey, PublicKey}
import org.healthlog.proxy.ProxyPipeline

object ResultHandler {

  /**
   * Process the given ConsumerResult and execute corresponding actions based on its
type.
   * @param result The ConsumerResult to process.
   * @param args   Additional arguments used for specific actions.
   * @return Any - The result of processing, depending on the executed action.
   */
  def processResult[T](result: ConsumerResult, args: T*): Any = {
    result match {
      case TypeIssueResult(did, privk, pubk, sub) =>
        (did, privk, pubk, sub)

      case TypeIssueDIDDOCResult(doc) =>
        doc

      case TypeInvokeResult(trigger)
        if trigger == "InvokeProxyPipeline" =>
        ProxyPipeline.proxyJobs(
          args(0).asInstanceOf[CoreApi],
          args(1).asInstanceOf[Plaintext],
          args(2).asInstanceOf[PrivateKey],
          args(3).asInstanceOf[PublicKey],
          args(4).asInstanceOf[PublicKey]
        )
    }
  }
}
```