

ΕΘΝΙΚΟ ΚΑΙ ΚΑΠΟΔΙΣΤΡΙΑΚΟ ΠΑΝΕΠΙΣΤΗΜΙΟ ΑΘΗΝΩΝ

«ΗΛΕΚΤΡΟΝΙΚΕΣ ΨΗΦΟΦΟΡΙΕΣ ΑΝΘΕΚΤΙΚΕΣ ΣΕ ΕΚΒΙΑΣΜΟΥΣ»

ΔΙΠΛΩΜΑΤΙΚΗ ΕΡΓΑΣΙΑ ΣΤΑ ΠΛΑΙΣΙΑ ΤΟΥ ΜΕΤΑΠΤΥΧΙΑΚΟΥ ΠΡΟΓΡΑΜΜΑΤΟΣ ΣΠΟΥΔΩΝ

ΜΠΛΑ

ΝΟΕΜΒΡΙΟΣ 2017

ΚΑΛΟΓΕΡΟΠΟΥΛΟΣ ΠΑΝΑΓΙΩΤΗΣ

Η παρούσα Διπλωματική Εργασία
εκπονήθηκε στα πλαίσια των σπουδών
για την απόκτηση του
Μεταπτυχιακού Διπλώματος Ειδίκευσης
στη
Λογική και Θεωρία Αλγορίθμων και Υπολογισμού
που απονέμει το
Τμήμα Μαθηματικών
του
Εθνικού και Καποδιστριακού Πανεπιστημίου Αθηνών

Εγκρίθηκε την από Εξεταστική Επιτροπή
αποτελούμενη από τους:

| <u>Όνοματεπώνυμο</u> | <u>Βαθμίδα</u> | <u>Υπογραφή</u> |
|----------------------|----------------|-----------------|
| 1. | | |
| 2. | | |
| 3. | | |

- Ευχαριστώ θερμά τον επιβλέποντα της διπλωματικής εργασίας Κο Παγουριζή για την πολύτιμη βοήθεια του.
- Ευχαριστώ τους κυρίους Φωτάκη Δημήτρη και Κιαγιά Άγγελο που με τίμησαν με τη συμμετοχή τους στην τριμελή επιτροπή.
- Ευχαριστώ τον συνάδελφο Γροντά Παναγιώτη και τον Ζαχαράκη Αλέξανδρο για την πολύτιμη βοήθεια και υπομονή τους.

Περιεχόμενα

| | | |
|----------|--|-----------|
| 1 | Εισαγωγή | 7 |
| 1.1 | Εισαγωγικά | 7 |
| 1.2 | Συνήθως... | 8 |
| 1.3 | <i>Cryptographic Primitives</i> | 11 |
| 1.3.1 | Scheme. | 11 |
| 1.3.2 | <i>Pedersen Commitments</i> | 11 |
| 1.3.3 | Digital Signatures. | 12 |
| 1.3.3.1 | <i>Security Digital Signatures.</i> | 13 |
| 2 | Plaintext Equation Test | 15 |
| 2.1 | Εισαγωγή. | 15 |
| 2.2 | <i>Computational Hardness Assumptions.</i> | 16 |
| 2.3 | Groth-Sahai (GS) Non-Interactive Zero Knowledge Proofs. | 16 |
| 2.4 | Smooth Projective Hash Functions | 19 |
| 2.5 | <i>Sign and Encrypt and Prove Paradigm</i> | 20 |
| 2.6 | All-or-Nothing Public Key Encryption With Equality Tests | 20 |
| 2.7 | Non-Interactive Plaintext (In-)Equality Proofs | 21 |
| 2.8 | Υλοποίηση | 22 |
| 3 | Towards a Publicly-Verifiable Mix-Net Providing Everlasting Privacy | 25 |
| 3.1 | Εισαγωγή | 25 |
| 3.2 | Δομικοί Λίθοι | 26 |
| 3.2.1 | Commitment Scheme | 26 |
| 3.2.2 | Encryption Scheme | 26 |
| 3.2.3 | Proofs of correct reencryption | 27 |
| 3.2.4 | Proofs of consistency | 27 |
| 3.3 | <i>Security</i> | 28 |
| 3.4 | Mixing Process | 28 |
| 3.4.1 | Υποβολή Μηνύματος | 28 |
| 3.5 | Mixing | 29 |
| 3.5.1 | Αποκρυπτογράφηση και Δημοσίευση | 30 |
| 3.5.2 | Πιστοποίηση | 30 |
| 3.6 | Ιδιότητες | 30 |
| 3.6.1 | Correctness | 30 |
| 3.6.2 | Robustness | 31 |
| 3.7 | Everlasting Privacy Towards the Authorities | 31 |

| | | |
|----------|--|-----------|
| 4 | Coercion-Resistant Electronic Elections | 33 |
| 4.1 | Modelling | 33 |
| 4.1.1 | Αρχές. | 33 |
| 4.1.2 | Συναρτήσεις. | 34 |
| 4.2 | Ορισμοί. | 36 |
| 4.3 | Περιγραφή Πρωτόκολλου. | 41 |
| 4.3.1 | Δομικοί Λίθοι. | 41 |
| 4.3.2 | Αρχές. | 41 |
| 4.3.3 | Εγγραφή Ψηφοφόρων. | 42 |
| 4.3.4 | Λίστα υποψήφιων. | 42 |
| 4.3.5 | Κατάθεση ηλεκτρονικών ψηφοδελτίων. | 42 |
| 4.3.6 | Καταμέτρηση | 43 |
| 4.3.7 | Αντοχή σε εκβιασμό. | 43 |
| 4.4 | Definitions of Correctness and Verifiability | 43 |
| 4.4.1 | Correctness | 43 |
| 4.4.2 | Verifiability | 44 |
| 4.4.3 | Strong Verifiability | 45 |
| 4.4.4 | Ψευδή Διαπιστευτήρια | 46 |
| 4.4.5 | Proving Coercion - Freeness. | 46 |
| 4.4.6 | Proof of Coercion Resistance | 47 |
| 5 | Coercion Resistant Internet Voting with Everlasting Privacy | 51 |
| 5.1 | Δομικοί Λίθοι. | 51 |
| 5.1.1 | Ομάδες και γεννήτορες. | 51 |
| 5.1.2 | Ομομορφικές δεσμεύσεις. | 51 |
| 5.1.3 | Πρωτόκολλο Κρυπτογράφησης. | 51 |
| 5.1.4 | Αποδείξεις Μηδενικής Γνώσης. | 52 |
| 5.1.5 | Cryptographic Shuffle | 53 |
| 5.2 | Περιγραφή Πρωτοκόλλου. | 53 |
| 5.2.1 | Εγγραφή Ψηφοφόρων. | 54 |
| 5.2.2 | Election Preparation | 54 |
| 5.2.3 | Vote Casting | 54 |
| 5.2.4 | Tallying | 55 |
| 5.3 | Ασφάλεια. | 56 |
| 5.3.1 | Correctness | 56 |
| 5.3.2 | Everlasting Privacy | 57 |
| 5.3.3 | Coercion Resistance | 57 |
| 5.4 | Συμπέρασμα. | 57 |
| 5.5 | Σχόλια. | 57 |
| 6 | Efficient coercion resistant and everlasting privacy in remote electronic elections | 59 |
| 6.1 | FOO. | 59 |
| 6.2 | Αρχές. | 61 |
| 6.3 | Δομικοί Λίθοι. | 61 |
| 6.3.1 | Blind Signatures | 62 |
| 6.4 | Conditional Blind Signatures. | 62 |

| | | |
|----------|---|-----------|
| 6.5 | Πρωτόκολλο. | 64 |
| 6.6 | Παραλλαγή ως προς την επικοινωνία. | 67 |
| 6.7 | Περιγραφή Πρωτοκόλλου | 68 |
| 6.7.1 | Αρχικοποίηση. | 68 |
| 6.7.2 | Εγγραφή Ψηφοφόρων. | 69 |
| 6.7.3 | Pre-Election. | 69 |
| 6.7.4 | Εξουσιοδότηση. | 69 |
| 6.7.5 | Voting. | 70 |
| 6.7.6 | Tallying. | 70 |
| 6.7.7 | Threshold Protocol for Authorisation. | 71 |
| 6.7.8 | Πολυπλοκότητα. | 72 |
| 6.8 | Ασφάλεια. | 72 |
| 6.8.1 | Verifiability. | 72 |
| 6.8.2 | Eligibility. | 72 |
| 6.8.3 | Everlasting Privacy. | 72 |
| 6.8.4 | Coercion Resistance. | 72 |
| 7 | Adding Everlasting Privacy in JCJ | 81 |
| 7.1 | Αρχές | 81 |
| 7.1.1 | Επιμέρους Στοιχεία. | 82 |
| 7.2 | Δομικοί Λίθοι. | 82 |
| 7.2.1 | Bulletin Board | 82 |
| 7.3 | Primitives | 82 |
| 7.4 | Voting | 85 |
| 7.5 | Credential Checking | 86 |
| 7.6 | Tallying | 89 |
| 7.7 | Coercion Resistance | 89 |
| 7.8 | Everlasting privacy | 89 |

Κεφάλαιο 1

Εισαγωγή

1.1 Εισαγωγικά

Στο πρώτο κεφάλαιο συνοψίζονται οι βασικές αρχές που θα πρέπει να ικανοποιεί ένα πρωτόκολλο διεξαγωγής εκλογών καθώς και η βασική ορολογία η οποία θα χρησιμοποιηθεί. Αν και τα οφέλη των ηλεκτρονικών εκλογών είναι πολλά, υψηλότερα ποσοστά συμμετοχής, χαμηλό κόστος διεξαγωγής, σχεδόν άμεση καταμέτρηση αποτελεσμάτων πολλά είναι και τα ανοιχτά ερωτήματα στα οποία πρέπει να δοθούν προσεκτικές απαντήσεις.

Στην παρούσα διπλωματική εργασία θα ασχοληθούμε με πρωτόκολλα που εξασφαλίζουν **Everlasting Privacy**. Συνοπτικά θα λέγαμε ότι τα πρωτόκολλα αυτά εξασφαλίζουν την προστασία του ψηφοφόρου ανεξάρτητα από την υπολογιστική δύναμη του αντιπάλου ακόμη κι αν μετά από μερικές δεκαετίες τα κρυπτογραφικά εργαλεία σπάσουν.

Ένα πρωτόκολλο ηλεκτρονικών εκλογών παρέχει **information-theoretic privacy** αν ένας αντίπαλος με απεριόριστη υπολογιστική ισχύ δεν αποκτά καμία πληροφορία για τις ψήφους, εκτός φυσικά από το τελικό αποτέλεσμα. Αν η μυστικότητα των ψήφων συνδέεται με κάποιο υπολογιστικό πρόβλημα (διακριτό λογάριθμο κτλ) τότε το πρωτόκολλο παρέχει **computational privacy**. Θα μπορούσαμε να πούμε ότι η **information-theoretic privacy** είναι **everlasting privacy**.

Στην πλειοψηφία των πρωτοκόλλων ηλεκτρονικών εκλογών με το τέλος της διαδικασίας υποβολής ψηφοδελτίου δημοσιοποιείται ένα μεγάλο κομμάτι πληροφορίας απαραίτητο για τον έλεγχο από κάθε ενδιαφερόμενο της ορθότητας και της τιμότητας της εκτέλεσης του πρωτοκόλλου. Η αποθήκευση των πληροφοριών αυτών πλέον απαιτεί αμελητέο οικονομικό κόστος. Κάθε ενδιαφερόμενος μπορεί να αποθηκεύσει την πληροφορία αυτή με σκοπό την επεξεργασία της μετά από μερικές δεκαετίες διαθέτοντας πολλαπλάσια υπολογιστική ισχύ.

Στο δεύτερο κεφάλαιο της εργασίας περιγράφεται η ιδέα των Olivier Blazy, David Derler, Daniel Slamanig, Raphael Spreitzer με τίτλο "NonInteractive Plaintext (In) Equality Proofs and Group Signatures with Verifiable Controllable Linkability." Παρουσιάζουν κρυπτοσύστημα το οποίο επιτρέπει έλεγχο ισότητας και ανισότητας κρυπτοκειμένων από ουδέτερη αρχή η οποία διαθέτει μία επιπλέον πληροφορία γνωστή ως *trapdoor*. Η αρχή δεν είναι σε θέση να αποκρυπτογραφεί κρυπτοκειμένα παρά μόνο να ελέγχει αν αυτά προέρχονται από το ίδιο αρχικό κείμενο ή όχι και να παρέχει αποδείξεις μηδενικής γνώσης.

Στο τρίτο κεφάλαιο περιγράφεται *mix net* το οποίο παρέχει *everlasting privacy*. Δυστυχώς απαιτεί την ύπαρξη ασφαλών καναλιών μετάδοσης, υπόθεση ίσως μη ρεαλιστική για ηλεκτρονικά εκλογικά πρωτόκολλα. Ακόμη κι αντίπαλος με απεριόριστη υπολογιστική ισχύ δεν μπορεί να σχετίσει τα μηνύματα της εισόδου με αυτά στην έξοδο αρκεί να υπάρχει ένα ειλικρινές *mix*.

Στο τέταρτο κεφάλαιο παρουσιάζεται το εκλογικό πρωτόκολλο *JCJ*. Το πρωτόκολλο αυτό παρέχει *Coercion Resistance* και είναι κορμός της ιδέας που θα σας παρουσιάσουμε στο τέλος της εργασίας.

Στο πέμπτο κεφάλαιο παρουσιάζεται το εκλογικό πρωτόκολλο των Philipp Locher, Rolf Haenni και Reto E. Koenig. Οι συγγραφείς ισχυρίζονται ότι το πρωτόκολλο παρέχει *Everlasting Privacy* καθώς και *Coercion Resistance*. Το πρωτόκολλο αποτυγχάνει αν ο αντίπαλος είναι σε θέση να υποβάλλει ψηφοδέλτιο τις τελευταίες στιγμές της ψηφοφορίας.

Στο έκτο κεφάλαιο παρουσιάζεται το εκλογικό πρωτόκολλο των Panagiotis Grontas, Aris Ragourtzis και Alexandros Zacharakis. Πρόδρομος της ιδέας το εκλογικό πρωτόκολλο *FOO*.

Στο έβδομο και τελευταίο κεφάλαιο περιγράφεται η προσπάθεια να ενισχύσουμε το εκλογικό πρωτόκολλο *JCJ* με την επιπλέον ιδιότητα *Everlasting Privacy*. Βασική ιδέα είναι ο διαμερισμός σε σύνολο αρχών τυχαιότητας που αποκρύπτει το διαπιστευτήριο του ψηφοφόρου. Ο δε έλεγχος εγκυρότητας γίνεται τυφλά χωρίς να αποκαλύπτεται η πραγματική ταυτότητα του ψηφοφόρου με την προϋπόθεση ότι πλειοψηφία αρχών παραμένουν ειλικρινείς και ο αντίπαλος μπορεί να παρακολουθεί συγκεκριμένο το πλήθος καναλιών.

1.2 Συνήθως...

Κατά την διεξαγωγή μίας εκλογικής διαδικασίας, ηλεκτρονικής ή όχι, παρατηρούνται συχνά τα ακόλουθα επιμέρους στάδια :

1. Μία αρχή κηρύσσει την έναρξη της εκλογικής διαδικασίας.
2. Καθορίζονται οι παράμετροι των εκλογών (δημοτικές, εθνικές, κτλ).
3. Καθορίζονται οι νόμιμοι ψηφοφόροι.
4. Οι νόμιμοι ψηφοφόροι εγγράφονται σε εκλογικές λίστες (σε διάφορα πρωτόκολλα η εγγραφή γίνεται μία φορά και ισχύει για διαδοχικές εκλογές).
5. Οι νόμιμοι ψηφοφόροι αποκτούν μέσω κάποιας διαδικασίας κατάλληλα διαπιστευτήρια (ανάλογα με το πρωτόκολλο που ακολουθείται ψηφοφόροι πιθανόν να τροποποιούν τα είδη υπάρχοντα).
6. Κατασκευάζονται και διανέμονται τα ψηφοδέλτια.
7. Ο εξοπλισμός και το προσωπικό προετοιμάζονται.

-
-
8. Τα διαπιστευτήρια των ψηφοφόρων ελέγχονται (sign in).
 9. Ο ψηφοφόρος συμπληρώνει – ελέγχει – καταθέτει το ψηφοδέλτιο του.
 10. Τα ψηφοδέλτια συγκεντρώνονται.
 11. Καταμέτρηση ψήφων.
 12. Δημοσιοποιούνται τα αποτελέσματα και η λίστα των ψηφοφόρων (πιθανόν προστατεύοντας την ταυτότητα τους) .
 13. Ελέγχονται τα αποτελέσματα.
 14. Αν κριθεί απαραίτητο καταμετρούνται οι ψήφοι ξανά.

Κάθε πρωτόκολλο εκλογών πρέπει:

1. Να είναι δημοκρατικό.
 - (α) Μόνο οι νόμιμοι ψηφοφόροι επιτρέπεται να ψηφίζουν.
 - (β) Οι νόμιμοι ψηφοφόροι επιτρέπεται να ψηφίζουν μία μόνο φορά.
2. Να προστατεύει την μυστικότητα της ψήφου.
 - (α) Κανείς δεν είναι σε θέση να γνωρίζει το περιεχόμενο άλλου ψηφοδελτίου.
 - (β) Δημοσιοποιεί λίστα με όσους ψήφισαν (πιθανόν προστατεύοντας την ταυτότητα τους).
3. Να είναι ανθεκτικό σε εκβιασμούς των ψηφοφόρων.
 - (α) Ο ψηφοφόρος θα πρέπει να προστατευθεί από πιθανούς εκβιασμούς ακόμη κι αν του ζητηθούν τα διαπιστευτήρια του.
 - (β) Ο ψηφοφόρος θα πρέπει να μην μπορεί να αποδείξει σε κανένα την επιλογή του (receipt free).
 - (γ) Ανθεκτικότητα στο χρόνο ακόμη κι αν τα κρυπτογραφικά πρωτόκολλα που χρησιμοποιούνται μετά από δεκαετίες σπάσουν.
4. Να παράγει ακριβές εκλογικό αποτέλεσμα.
 - (α) Μόνο οι νόμιμοι ψήφοι καταμετρούνται ενώ ψηφοδέλτια μη εξουσιοδοτημένων ψηφοφόρων δεν καταμετρούνται.
 - (β) Το περιεχόμενο των ψηφοδελτίων δεν παραποιείται, είναι αδύνατη η διαγραφή ή αντικατάσταση ψηφοδελτίων.
5. Το πρωτόκολλο πρέπει να παρέχει δυνατότητες επαλήθευσης της σωστής καταμέτρησης καθώς και της τιμιότητας των αρχών που το εκτελούν (verifiability).
 - (α) Κάθε ψηφοφόρος πρέπει να είναι σε θέση να επαληθεύσει την ορθή καταμέτρηση της επιλογής του.

-
- (β') Κάθε ψηφοφόρος πρέπει να μπορεί να αναθέτει την διαδικασία επαλήθευσης της ψήφου του σε κάποιο ενδιαφερόμενο τρίτο μέρος χωρίς να αποκαλύπτει το περιεχόμενο της.
- (γ') Όλοι πρέπει να είναι σε θέση να επαληθεύουν την ορθότητα του εκλογικού αποτελέσματος.

6. Να είναι ανθεκτικό σε κακόβουλες μειοψηφίες.

7. Να μην αποκαλύπτει αποτελέσματα πριν την ολοκλήρωση του πρωτοκόλλου.

Συχνά στα εκλογικά πρωτόκολλα υπάρχουν:

1. Ένα σύνολο από n_V ψηφοφόρους, $\mathcal{V} = \{V_1, V_2, \dots, V_{n_V}\}$. Ο δείκτης i αντιστοιχεί σε μοναδικό ψηφοφόρο.
2. Αρχές έκδοσης διαπιστευτηρίων (Registrars).

Ένα σύνολο από $n_{\mathcal{R}}$ οντότητες $\mathcal{R} = \{R_1, R_2, \dots, R_{n_{\mathcal{R}}}\}$ επιφορτισμένες με την έκδοση διαπιστευτηρίων στους ψηφοφόρους.

3. Αρχές Επεξεργασίας ψηφοδελτίων (Talliers)

Ένα σύνολο από $n_{\mathcal{T}}$ οντότητες, $\mathcal{T} = \{T_1, T_2, \dots, T_{n_{\mathcal{T}}}\}$ αρχές επιφορτισμένες με την διαδικασία επεξεργασίας των ψηφοδελτίων, καταμέτρησης των ψήφων και έκδοσης τελικών αποτελεσμάτων.

Σχεδόν όλα τα πρωτόκολλα ηλεκτρονικών εκλογών μπορούν να ταξινομηθούν στις ακόλουθες 3 κατηγορίες:

- **Mix - Type.** Τα πρωτόκολλα αυτά βασίζονται κυρίως στην ύπαρξη *mixes*.
- **Blind Signatures.** Βασική ιδέα πίσω από τις λεγόμενες τυφλές υπογραφές είναι ότι μια έμπιστη αρχή θα υπογράψει τυφλά τα ψηφοδέλτια των ψηφοφόρων.
- **Homomorphic Voting.** Τα πρωτόκολλα αυτά βασίζονται σε συναρτήσεις με την ιδιότητα

$$E(x + y) = E(x) + E(y).$$

Κάθε ψηφοφόρος κρυπτογραφεί το ψηφοδέλτιο χρησιμοποιώντας το δημόσιο κλειδί που έχει παραχθεί στην αρχή της εκλογικής διαδικασίας. Κάθε ψηφοφόρος αποδεικνύει ότι το κρυπτοκείμενο που κατάθεσε προέρχεται από την κρυπτογράφηση του ψηφοδελτίου του. Τα κρυπτοκείμενα αθροίζονται χωρίς να έχουν αποκρυπτογραφηθεί. Ένα σύνολο έμπιστων αρχών συνεργάζονται για την από κοινού αποκρυπτογράφηση και επεξεργασία του αποτελέσματος

1.3 Cryptographic Primitives

1.3.1 Scheme.

Κρυπτογραφική δέσμευση (commitment scheme) είναι ένα κρυπτογραφικό εργαλείο που επιτρέπει σε κάποιον χρήστη \mathcal{A} να δεσμευτεί σε μία επιλογή x αφαιρώντας του την δυνατότητα να αλλάξει γνώμη μετέπειτα.

$$c := Comm(x, open)$$

Η τιμή c καλείται δέσμευση του \mathcal{A} στο x .

Η επιλογή x από την χρήστη στην διαδικασία της δέσμευσης είναι μυστική. Η επιλογή αποκαλύπτεται στην τελική φάση του πρωτοκόλλου με την εκμυστήρευση της τυχαιότητας $open$.

Επιθυμούμε τα σχήματα αυτά να έχουν τις εξής ιδιότητες:

- **Binding:** Ο \mathcal{A} δεν μπορεί να αλλάξει την επιλογή του x μετέπειτα.
- **Hiding:** Το c δεν αποκαλύπτει καμία πληροφορία για το x .

Τα πρωτόκολλα αυτά δεν μπορούν ταυτόχρονα να είναι **perfectly hiding** και **perfectly binding**.

Ανάλογα με τη χρήση κανείς μπορεί να επιλέξει σε πρωτόκολλα που:

- Είναι **perfectly binding** και **computationally hiding**. Στα πρωτόκολλα αυτά ο χρήστης είναι αδύνατο να βρει δύο διαφορετικά x_1, x_2 ώστε $Comm(x_1, r_1) = Comm(x_2, r_2)$ ακόμα κι αν διαθέτει απεριόριστη υπολογιστική ισχύ. Κάποιος που έχει απεριόριστη υπολογιστική ισχύ μπορεί να υπολογίσει την τιμή x .
- Είναι **computationally binding** και **perfectly hiding**. Αντίστοιχα εδώ κάποιος με απεριόριστη υπολογιστική ισχύ μπορεί να υπολογίσει διαφορετικά x_1, x_2 ώστε $Comm(x_1, r_1) = Comm(x_2, r_2)$ ενώ κανείς δεν μπορεί να υπολογίσει την τιμή x ακόμα κι αν διαθέτει απεριόριστη υπολογιστική ισχύ.

Ακολουθεί ένα πρωτόκολλο που είναι **computationally binding** και **perfectly hiding**.

1.3.2 Pedersen Commitments

Set Up:

1. Επιλέγει p, q πρώτους αριθμούς ώστε $q|(p-1)$.
2. Επιλέγει g γεννήτορα της υποομάδας G τάξεως q της \mathbb{Z}_p^* .
3. Επιλέγει τυχαίο α .

-
- Υπολογίζει το $h = g^\alpha \pmod p$.
 - Επιστρέφει ως δημόσια τα $\langle p, q, g, h \rangle$

Commit:

Για να δεσμευτεί στην τιμή x :

- Επιλέγει τυχαίο $r \in Z_q$.
- Υπολογίζει το $c = g^x h^r \pmod p$.

Open:

Αποκαλύπτονται τα x, r και ελέγχεται αν $c = g^x h^r \pmod p$.

1.3.3 Digital Signatures.

Σε αναλογία με την χειρόγραφη προσωπική υπογραφή τα πρωτόκολλα ψηφιακών υπογραφών επιχειρούν να καλύψουν τις αντίστοιχες ανάγκες στον ψηφιακό κόσμο. Ένα πρωτόκολλο ψηφιακής υπογραφής φιλοδοξεί να παρέχει στον χρήστη ένα μοναδικό ψηφιακό αποτύπωμα το οποίο αφενός δεν παραχαράσσεται και αφετέρου σηματοδοτεί την ταυτότητα του.

Ένα πρωτόκολλο ψηφιακής υπογραφής δημοσίου κλειδιού αποτελείται από μια τριάδα (G, σ, V) ώστε:

1. Ο τυχαιοκρατικός πολυωνυμικού χρόνου αλγόριθμος G με είσοδο μια παράμετρο ασφαλείας 1^k παράγει ζευγάρι κλειδιών (P, S) όπου P είναι το δημόσιο κλειδί και S το ιδιωτικό κλειδί του χρήστη. $((P, S) \in G(1^k))$.
2. Ο αλγόριθμος υπογραφής σ , τυχαιοκρατικός, πολυωνυμικού χρόνου που με είσοδο μια παράμετρο ασφαλείας 1^k , μυστικό κλειδί S αντίστοιχης ασφάλειας και μηνύματος $m \in \{0, 1\}^k$ παράγει υπογραφή s στο m . Συμβολίζουμε $s \in \sigma(1^k, S, m)$ αν ο αλγόριθμος είναι τυχαιοκρατικός και $s = \sigma(1^k, S, m)$ διαφορετικά. Απλούστερα, $s = \sigma(S, m)$.
3. Τυχαιοκρατικός, πολυωνυμικού χρόνου αλγόριθμος επαλήθευσης V ο οποίος με είσοδο το δημόσιο κλειδί P , μια υπογραφή s και το μήνυμα m , επιστρέφει 1 αν η επαλήθευση επιτύχει και 0 διαφορετικά. Προφανώς, $V(P, s, m) = 1$ αν $s \in \sigma(m)$ και 0 διαφορετικά. Απλούστερα $V(s, m)$ χωρίς να δίνεται έμφαση στο δημόσιο κλειδί.
4. Μία υπογραφή οφείλει να είναι ανθεκτική σε προσπάθειες παραχάραξης από PPT αντιπάλους.

Αν ο αλγόριθμος V είναι τυχαιοκρατικός η συνθήκη αποδοχής και απόρριψης υπογραφών μπορεί να χαλαρώσει ώστε να αποδέχεται έγκυρες υπογραφές και να απορρίπτει πλαστές με μεγάλη πιθανότητα για όλα τα μηνύματα m , επαρκώς μεγάλη παράμετρο ασφαλείας 1^k και ζευγάρια κλειδιών $(P, S) \in G(1^k)$. Επισημαίνουμε ότι τα μηνύματα δύναται να είναι κρυπτογραφημένα. Ο χώρος των μηνυμάτων είναι κάθε υποσύνολο του $\{0, 1\}^*$.

1.3.3.1 Security Digital Signatures.

Διακρίνουμε τρία βασικά είδη επιθέσεων στα πρωτόκολλα ψηφιακών υπογραφών. Με σειρά σημαντικότητας:

1. **Key - Only Attack:** Ο αντίπαλος γνωρίζει μόνο το δημόσιο κλειδί του χρήστη και μπορεί απλά να ελέγχει την εγκυρότητα των υπογραφών που λαμβάνει.
2. **Known Signature Attack:** Ο αντίπαλος γνωρίζει το δημόσιο κλειδί του χρήστη και έχει στη διάθεση του ζευγάρια με μηνύματα και έγκυρες υπογραφές. Είναι και το ελάχιστο που αντίπαλος διαθέτει σε ρεαλιστικά σενάρια.
3. **Chosen Message Attack:** Στον αντίπαλο επιτρέπεται να ζητά μηνύματα της επιλογής του καθώς και υπογραφές από τον υπογράφων. Η επιλογή των μηνυμάτων εξαρτάται πιθανώς από τα προηγούμενα μηνύματα και υπογραφές.

Διακρίνουμε διάφορα επίπεδα επιτυχίας στην προσπάθεια του αντιπάλου να παραχαράξει την υπογραφή και να σπάσει το πρωτόκολλο. Με φθίνουσα σειρά σημαντικότητας:

1. **Existential Forgery:** Ο αντίπαλος επιτυγχάνει να υπογράψει έγκυρα κάποιο μήνυμα όχι απαραίτητα της επιλογής του.
2. **Selective Forgery:** Ο αντίπαλος επιτυγχάνει να υπογράψει έγκυρα κάποια μηνύματα της επιλογής του.
3. **Universal Forgery:** Ο αντίπαλος αν και δεν καταφέρνει να υπολογίσει το ιδιωτικό κλειδί του χρήστη δύναται να υπογράψει κάθε μήνυμα έγκυρα.
4. **Total Break:** Ο αντίπαλος υπολόγισε το ιδιωτικό κλειδί.

Η επιθυμητή ασφάλεια είναι ανάλογη φυσικά με την χρήση του πρωτοκόλλου. Σε κάποιες εφαρμογές είναι αρκετό ο αντίπαλος που επιτίθεται με *known signature attack* να μην μπορεί να επιτύχει *selective forgery*. Σε άλλες εφαρμογές πρέπει το πρωτόκολλο να είναι εξαιρετικά ανθεκτικό ώστε αντίπαλος που επιτίθεται με *chosen signature attack* να επιτυγχάνει ακόμα και στην *existential forgery* με αμελητέα μόνο πιθανότητα.

Ένα πρωτόκολλο ψηφιακών υπογραφών είναι ασφαλές αν ο αντίπαλος χρησιμοποιώντας σαν μαντείο τον χρήστη δεν μπορεί σε πολυωνυμικό χρόνο ως προς το μέγεθος του δημόσιου κλειδιού να πλαστογραφήσει υπογραφή σε μήνυμα που δεν έχει υπογραφεί από τον χρήστη. Αυστηρότερα, έστω B ένα *blackbox* που παράγει έγκυρες υπογραφές σε μηνύματα m , $V(P, B(m), m) = 1$ για όλα τα μηνύματα m και ο αλγόριθμος πλαστογράφησης F με είσοδο το δημόσιο κλειδί P έχει πρόσβαση στο B (συμβολικά $F^B(P)$). Ο αλγόριθμος τρέχει σε δύο στάδια. Αρχικά ζητά μηνύματα και υπογραφές και έπειτα παράγει έγκυρη υπογραφή σε νέο μήνυμα (σε μήνυμα για το οποίο δεν έχει λάβει υπογραφή από το B).

Για όλους τους πιθανούς αλγόριθμους πλαστογράφησης F , για όλα τα πολυώνυμα Q , για αρκετά μεγάλο k ,

$$\text{Prob}(V(P, s, m) = 1 : (P, S) \leftarrow G(1^k); (m, s) \leftarrow F^B(P)) \leq \frac{1}{Q(k)}.$$

Chapter 2

Plaintext Equation Test

Ένα κρυπτογραφικό εργαλείο που εξετάζει αν δύο κρυπτοκείμενα προέρχονται από το ίδιο αρχικό κείμενο ή όχι χωρίς να διαρρέει καμία περαιτέρω πληροφορία για τα αρχικά κείμενα.

Ακολουθούν εν συντομία οι ιδέες που παρουσιάζονται στο [25] των Olivier Blazy, David Derler, Daniel Slamanig και Raphael Spreitzer με τίτλο "Non-Interactive Plaintext (In-)Equality Proofs and Group Signatures with Verifiable Controllable Linkability."

Ο έλεγχος για την ισότητα των αρχικών κειμένων γίνεται από prover που δεν έχει γνώση του ιδιωτικού κλειδιού του κρυπτοσυστήματος ούτε και της τυχαιότητας που χρησιμοποιήθηκε.

2.1 Εισαγωγή.

ΑΣ περιγράψουμε εν συντομία βασικούς ορισμούς και εργαλεία. Με $x \leftarrow_R X$ συμβολίζεται η τυχαία επιλογή στοιχείου x από το σύνολο X . Μια συνάρτηση $\epsilon : \mathbb{N} \rightarrow \mathbb{R}^+$ καλείται αμελητέα αν για κάθε $c > 0$ υπάρχει k_0 τέτοιο ώστε $\epsilon(k) < \frac{1}{k^c}$ για όλα τα $k > k_0$. Στο υπόλοιπο της παραγράφου με ϵ συμβολίζονται τέτοιες συναρτήσεις. Με έντονα κεφαλαία γράμματα συμβολίζονται διανύσματα $\mathbf{X} = (X_1, X_2, \dots, X_n)$.

Έστω $\mathbb{G}_1 = \langle g \rangle$, $\mathbb{G}_2 = \langle \hat{g} \rangle$ και \mathbb{G}_T ομάδες με τάξη πρώτο αριθμό p . Τα στοιχεία της ομάδας \mathbb{G}_2 συμβολίζονται ως \hat{g}, \hat{h} κτλ.

Ορισμός.

Μία απεικόνιση $e : \mathbb{G}_1 \times \mathbb{G}_2 \rightarrow \mathbb{G}_T$ ονομάζεται bilinear αν:

1. Για κάθε $(u, \hat{v}, a, b) \in \mathbb{G}_1 \times \mathbb{G}_2 \times \mathbb{Z}_p^2$ ισχύει ότι $e(u^a, \hat{v}^b) = e(u, \hat{v})^{ab}$.
2. $e(g, \hat{g}) \neq 1$.
3. Η e υπολογίζεται με αποδοτικό τρόπο.

Υποθέτουμε ότι $\mathbb{G}_1 \neq \mathbb{G}_2$.

2.2 Computational Hardness Assumptions.

Ας δούμε τις απαιτούμενες προϋποθέσεις που πρέπει να εκπληρώνουν οι ομάδες που θα χρησιμοποιηθούν στο πρωτόκολλο.

Decisional Diffie-Hellman Assumption (DDH).

Έστω $\mathbb{G} = \langle g \rangle$ ομάδα με τάξη πρώτο αριθμό p ώστε $\log_2 p = k$. Τότε για κάθε PPT αντίπαλο \mathcal{A} υπάρχει αμελητέα συνάρτηση $\epsilon(\cdot)$ ώστε:

$$\Pr[b \leftarrow_R \{0, 1\}, r, s, t \leftarrow_R \mathbb{Z}_p^*, b^* \leftarrow \mathcal{A}(g, g^r, g^s, g^{b(rs)+(1-b)t}) | b = b^*] \leq \frac{1}{2} + \epsilon(k).$$

Decision Linear Assumption (DLIN).

Έστω $\mathbb{G} = \langle u \rangle = \langle v \rangle = \langle h \rangle$ ομάδα με τάξη πρώτο αριθμό p ώστε $\log_2 p = k$. Τότε για κάθε PPT αντίπαλο \mathcal{A} υπάρχει αμελητέα συνάρτηση $\epsilon(\cdot)$ ώστε:

$$\Pr[b \leftarrow_R \{0, 1\}, r, s, t \leftarrow_R \mathbb{Z}_p, b^* \leftarrow \mathcal{A}(u, v, h, u^r, v^s, h^{b(r+s)+(1-b)t}) | b = b^*] \leq \frac{1}{2} + \epsilon(k).$$

Computational co-Diffie-Hellman Assumption (co-CDH).

Έστω $\mathbb{G}_1 = \langle g \rangle$ και $\mathbb{G}_2 = \langle \hat{g} \rangle$ διακεκριμένες ομάδες με τάξη πρώτο αριθμό p , ώστε $\log_2 p = k$. Τότε για κάθε PPT αντίπαλο \mathcal{A} υπάρχει αμελητέα συνάρτηση $\epsilon(\cdot)$ ώστε:

$$\Pr[r \leftarrow_R \mathbb{Z}_p, \hat{h} \leftarrow (g, g^r, \hat{g}) | \hat{h} = \hat{g}^r] \leq \epsilon(k).$$

External Diffie-Hellman Assumption (XDH).

Έστω $\mathbb{G}_1, \mathbb{G}_2$ και \mathbb{G}_T κυκλικές ομάδες με τάξη πρώτο αριθμό p και $e : \mathbb{G}_1 \times \mathbb{G}_2 \rightarrow \mathbb{G}_T$. Το XDH δηλώνει ότι το DDH ισχύει στην \mathbb{G}_1 .

Symmetric External Assumption (SXDH).

Έστω $\mathbb{G}_1, \mathbb{G}_2$ και \mathbb{G}_T κυκλικές ομάδες με τάξη πρώτο αριθμό p και $e : \mathbb{G}_1 \times \mathbb{G}_2 \rightarrow \mathbb{G}_T$. Το $SXDH$ δηλώνει ότι το DDH ισχύει στις \mathbb{G}_1 και \mathbb{G}_2 .

2.3 Groth-Sahai (GS) Non-Interactive Zero Knowledge Proofs.

Οι Groth-Sahai στο [26] δημιούργησαν το απαραίτητο πλαίσιο για αποδοτικές Non-Interactive Zero Knowledge (NIZK) και Non-Interactive Witness Indistinguishable (NIWI) αποδείξεις.

Μεταξύ άλλων επιτρέπει την απόδειξη ισχυρισμών που αφορούν την satisfiability των λεγόμενων pairing product equations (PPEs).

Αν και είναι ανεξάρτητο το πλαίσιο των αποδείξεων από τις ιδιότητες ασφάλειας που πρέπει να ικανοποιεί η ομάδα \mathbb{G} θεωρούμε ότι το $SXDH$ ικανοποιείται.

Μία PPE (pairing product equation). έχει την ακόλουθη μορφή:

$$\prod_{i=1}^n e(A_i, \hat{Y}_i) \cdot \prod_{i=1}^m e(X_i, \hat{B}_i) \cdot \prod_{i=1}^m \prod_{j=1}^n e(X_i, \hat{Y}_j)^{\gamma_{ij}} = t_T,$$

όπου :

- Τα $\mathbf{X} \in \mathbb{G}_1^m, \mathbf{Y} \in \mathbb{G}_2^n$ τα μυστικά διανύσματα που πρέπει να αποδείξεις ότι έχεις γνώση.
- Τα $\mathbf{A} \in \mathbb{G}_1^n, \hat{\mathbf{B}} \in \mathbb{G}_2^m$ και $\Gamma = (\gamma_{ij})_{i \in [m], j \in [n]} \in \mathbb{Z}_p^{n \cdot m}$ και t_T δημόσιες σταθερές.

Σε μια χαλαρή περιγραφή οι αποδείξεις GS ακολουθούν την ακόλουθη στρατηγική :

1. Αντί των διανυσμάτων $\mathbf{X}, \hat{\mathbf{Y}}$ χρησιμοποιούνται στην PPE εξίσωση δεσμεύσεις επί των διανυσμάτων αυτών.
2. Η απόδειξη π αποκαλύπτει την τυχαιότητα που χρησιμοποιείται στις δεσμεύσεις.
3. Επειδή δεν είναι δυνατόν να εργαστούμε στις ομάδες $\mathbb{G}_1, \mathbb{G}_2$ και \mathbb{G}_T τα στοιχεία προβάλλονται στον διανυσματικούς χώρους $\mathbb{G}_1^2, \mathbb{G}_2^2$ και \mathbb{G}_T^4 .
4. Αποδεικνύεται η satisfiability της PPE εξίσωσης χρησιμοποιώντας την επαγόμενη bilinear απεικόνιση $F : \mathbb{G}_1^2 \times \mathbb{G}_2^2 \rightarrow \mathbb{G}_T^4$.

Αυστηρότερα, μία GS απόδειξη για εξίσωση PPE επιτρέπει την απόδειξη γνώσης για κάποιον witness $w = (\mathbf{X}, \hat{\mathbf{Y}})$ τέτοιον ώστε η PPE εξίσωση που καθορίζεται με μοναδικό τρόπο από $x = (\mathbf{A}, \hat{\mathbf{B}}, \Gamma, t_T)$ να ικανοποιείται. Με BG συμβολίζεται η bilinear group και R είναι σχέση ώστε $(BG, x, w) \in R$ αν και μόνο αν w είναι satisfying witness για το x ως προς την BG . Με L_R συμβολίζεται η επαγόμενη γλώσσα.

Ένα μη διαδραστικό σύστημα αποδείξεων στο bilinear group πλαίσιο ορίζεται ως εξής :

Ορισμός.

Ένα μη διαδραστικό σύστημα αποδείξεων Π αποτελείται από τους ακόλουθους PPT αλγόριθμους $(BGGen, CRSGen, Proof, Verify)$:

1. $BGGen(1^k)$: Με είσοδο μια παράμετρο ασφάλειας k δίνει στην έξοδο την περιγραφή μιας bilinear ομάδας BG .
2. $CRSGen(BG)$: Με είσοδο την περιγραφή μιας bilinear ομάδας BG δίνει στην έξοδο μία common reference string crs .
3. $Proof(BG, crs, x, w)$: Με είσοδο την περιγραφή μιας bilinear ομάδας BG , μία common reference string crs , μια δήλωση x και witness w δίνει στην έξοδο μια απόδειξη π .
4. $Verify(BG, crs, x, \pi)$: Με είσοδο την περιγραφή μιας bilinear ομάδας, μία common reference string, μια δήλωση x και απόδειξη π δίνει στην έξοδο 1 αν η απόδειξη είναι έγκυρη και 0 διαφορετικά.

Ως προς τις ιδιότητες που οφείλουν να πληρούν τα μη διαδραστικά συστήματα αποδείξεων (Non Interactive Proof Systems) ακολουθούν οι σχετικοί ορισμοί:

Perfect Completeness.

Ένα μη διαδραστικό σύστημα αποδείξεων είναι *perfectly complete* αν για κάθε αντίπαλο \mathcal{A} ισχύει ότι:

$$Pr \left[\begin{array}{l} BG \leftarrow BGGen(1^k) \\ crs \leftarrow CRSGen(BG) \\ (x, w) \leftarrow \mathcal{A}(BG, crs) \\ \pi \leftarrow Proof(BG, crs, x, w) \end{array} \quad \begin{array}{l} Verify(BG, crs, x, \pi) = 1 \wedge \\ (BG, x, w) \in R \end{array} \right] = 1$$

Perfect Soundness.

Ένα μη διαδραστικό σύστημα αποδείξεων είναι *perfectly sound* αν για κάθε αντίπαλο \mathcal{A} ισχύει ότι:

$$Pr \left[\begin{array}{l} BG \leftarrow BGGen(1^k) \\ crs \leftarrow CRSGen(BG), \\ (x, \pi) \leftarrow \mathcal{A}(BG, crs) \end{array} \quad \begin{array}{l} Verify(BG, crs, x, \pi) = 1 \wedge \\ x \notin L_R \end{array} \right] = 0$$

Witness Indistinguishability.

Ένα μη διαδραστικό σύστημα αποδείξεων είναι *composably witness indistinguishable* αν υπάρχει εξομοιωτής \mathcal{S} τέτοιος ώστε για όλους τους *PPT* αντίπαλους \mathcal{A} υπάρχει αμελητέα συνάρτηση $\epsilon(\cdot)$ τέτοια ώστε:

$$\left| \begin{array}{l} Pr[BG \leftarrow BGGen(1^k), crs \leftarrow CRSGen(BG) : \mathcal{A}(BG, crs) = 1] - \\ Pr[BG \leftarrow BGGen(1^k), crs \leftarrow S(BG) : \mathcal{A}(BG, crs) = 1] \end{array} \right| \leq \epsilon(k)$$

και

$$Pr \left[\begin{array}{l} b \leftarrow_R \{0, 1\}, BG \leftarrow BGGen(1^k) \\ crs \leftarrow SBG), (x, w_0, w_1, st) \leftarrow \mathcal{A}(BG, crs) \\ \pi \leftarrow Proof(BG, crs, x, w_b), b^* \leftarrow \mathcal{A}(\pi, st) \end{array} \quad \begin{array}{l} b = b^* \wedge \\ (BG, x, w_0) \in R \wedge \\ (BG, x, w_1) \in R \end{array} \right] = \frac{1}{2}$$

Zero Knowledge.

Ένα μη διαδραστικό σύστημα αποδείξεων είναι *composably zero knowledge* αν υπάρχουν εξομοιωτές $\mathcal{S}_1, \mathcal{S}_2$ τέτοιοι ώστε για όλους τους *PPT* αντίπαλους \mathcal{A} υπάρχει αμελητέα συνάρτηση $\epsilon(\cdot)$ τέτοια ώστε:

$$Pr \left[\begin{array}{l} Pr[BG \leftarrow BGGen(1^k), crs \leftarrow CRSGen(BG) : \mathcal{A}(BG, crs) = 1] - \\ Pr[BG \leftarrow BGGen(1^k), (crs, T) \leftarrow \mathcal{S}_1(BG) : \mathcal{A}(BG, crs) = 1] \end{array} \right| \leq \epsilon(k)$$

και

$$Pr \left[\begin{array}{l} BG \leftarrow BGGen(1^k), (crs, T) \leftarrow S_1(BG) \\ (x, w, st) \leftarrow \mathcal{A}(BG, crs, T) \\ \pi \leftarrow Proof(BG, crs, x, w) \end{array} : \mathcal{A}(\pi, st) = 1 \right] =$$

$$Pr \left[\begin{array}{l} BG \leftarrow BGGen(1^k), (crs, T) \leftarrow S_1(BG) \\ (x, w, st) \leftarrow \mathcal{A}(BG, crs, T) \\ \pi \leftarrow S_2(BG, crs, x, T) \end{array} : \mathcal{A}(\pi, st) = 1 \right]$$

Οι αποδείξεις GS είναι perfectly complete, perfectly sound, witness indistinguishable. Όταν $t_T = 1_{\mathbb{G}_T}$ και η εξίσωση δεν περιέχει κάποιο ταίριασμα δύο δημόσιων σταθερών είναι και composable zero knowledge.

2.4 Smooth Projective Hash Functions

Οι λείες προβολικές Hash Functions (SPHF) είναι οικογένειες ζευγών συναρτήσεων ($Hash, ProjHash$) πάνω σε μια γλώσσα L . Συνοδεύονται από ένα ζευγάρι κλειδιών (hk, hp). Το hk (hashing key) μπορεί να θεωρηθεί ως το ιδιωτικό κλειδί ενώ το hp (projection key) ως το δημόσιο. Οι εικόνες των συναρτήσεων πάνω σε μια λέξη $W \in L$ πρέπει να ταυτίζονται

$$Hash(hk, L, W) = ProjHash(hp, L, W, w).$$

Για τον δεύτερο υπολογισμό απαραίτητη είναι η ύπαρξη ενός witness w τέτοιου ώστε $W \in L$.

Ορισμός.

Μια SPHF για γλώσσα L αποτελείται από τους PPT αλγόριθμους

1. **Setup(1^k)**: Με είσοδο μία παράμετρο ασφαλείας k παράγει παραμέτρους pp . Θεωρούμε πως όλοι ο αλγόριθμοι έχουν πρόσβαση στους pp .
2. **HashKG(L)**: Με είσοδο μία γλώσσα L παράγει ένα hashing κλειδί hk για την L .
3. **ProjKG(hk, L, W)**: Με είσοδο ένα hashing κλειδί hk , την γλώσσα L , μία λέξη W παράγει ένα κλειδί projection hp που πιθανόν να εξαρτάται από την W .
4. **Hash(hk, L, W)**: Με είσοδο ένα hashing κλειδί hk , την γλώσσα L , μία λέξη W παράγει ένα Hash H' .
5. **ProjHash(hp, L, W, w)**: Με είσοδο ένα projecting κλειδί hp , την γλώσσα L , μία λέξη W και witness w για την $W \in L$ παράγει ένα Hash H .

2.5 Sign and Encrypt and Prove Paradigm

Σχήματα υπογραφών που βασίζονται σε ομάδες και ακολουθούν το παράδειγμα Sign and Encrypt and Prove Paradigm είναι δημοφιλή και υπάρχουν αρκετές αποδοτικές υλοποιήσεις. Ένα τέτοιο σχήμα απαρτίζεται από τα ακόλουθα επιμέρους στοιχεία :

1. Ένα ασφαλές σχήμα υπογραφών $DS = (KeyGen, Sign, Vrfy)$.
2. Ένα τουλάχιστον $IND-CPA$ ασφαλές κρυπτοσύστημα δημοσίου κλειδιού $\mathcal{AE} = (KeyGen_e, Enc, Dec)$.
3. Ένα μη διαδραστικό σύστημα αποδείξεων μηδενικής γνώσης $NIZKPK$.

Το κλειδί gpk αποτελείται από το δημόσιο κλειδί του κρυπτοσυστήματος pk_e και το δημόσιο κλειδί του πρωτόκολλου των υπογραφών pk_s . Το master opening key mok , περιέχει το ιδιωτικό κλειδί του κρυπτοσυστήματος sk_e καθώς και το master issuing key mik , δηλαδή το ιδιωτικό κλειδί των υπογραφών sk_s .

Ο χρήστης i στέλνει το $f(x_i)$ στον εκδότη υπογραφών. Η f είναι μια One way function και το x_i κάποιο μυστικό του χρήστη. Ο Εκδότης επιστρέφει μια υπογραφή $cert \leftarrow Sign(sk_s, f(x_i))$ που ο χρήστης χρησιμοποιεί ως διαπιστευτήριο.

Μία group signature $\sigma = (T, \pi)$ σε κάποιο μήνυμα M αποτελείται από ένα κρυπτοκείμενο $T \leftarrow Enc(pk_e, cert)$ καθώς και

$$\pi \leftarrow SoK\{(x_i, cert) : cert = Sign(sk_s, f(x_i)) \wedge T = Enc(pk_e, cert)\}(M).$$

Στην παρούσα εργασία ο τρόπος κατασκευής του T δεν είναι σημαντικός.

2.6 All-or-Nothing Public Key Encryption With Equality Tests

Η ιδέα πίσω από τα κρυπτοσυστήματα $AoN - PKEET$ είναι να επιτρέπει σε τρίτο που έχει στην κατοχή του μία επιπλέον πληροφορία (trapdoor) να εκτελεί ελέγχους ισότητας μεταξύ κρυπτοκειμένων χωρίς να είναι σε θέση να ανακτά τα αρχικά κείμενα. Οι έλεγχοι συνοδεύονται από αποδείξεις μηδενικής γνώσης.

Ένα $AoN - PKEET$ είναι συνηθισμένο κρυπτοσύστημα δημοσίου κλειδιού το οποίο παρέχει τουλάχιστον $IND - CPA$ ασφάλεια και επιτρέπει αποδείξεις μηδενικής γνώσης. Αποτελείται από $(KeyGen, Enc, Dec, Aut, Com)$ όπου :

- $Aut(sk_e)$: Ο αλγόριθμος με είσοδο το ιδιωτικό κλειδί του κρυπτοσυστήματος sk_e επιστρέφει trapdoor tk που επιτρέπει τον έλεγχο ισότητας κρυπτοκειμένων.
- $Com(T, T', tk)$: Ο αλγόριθμος με είσοδο δύο κρυπτοκείμενα T, T' και trapdoor tk επιστρέφει 1 αν τα κρυπτοκείμενα προκύπτουν από το ίδιο αρχικό (άγνωστο) κείμενο και 0 διαφορετικά.

Ορισμός.

Ένα $AoN - PKEEET$ σχήμα θεωρείται ασφαλές αν είναι `sound` και παρέχει

- $OW - CPA$ ασφάλεια απέναντι σε αντιπάλους που γνωρίζουν την επιπλέον πληροφορία `trapdoor tk`.
- Το κρυπτοσύστημα παρέχει $IND - CPA/IND - CCA$ ασφάλεια.

Κατασκευή με χρήση ElGamal.

Σε `bilinear` ομάδα που η υπόθεση $SXDH$ ισχύει η κρυπτογράφηση στην ομάδα \mathbb{G}_1 με ElGamal αρκεί. Ας υποθέσουμε ότι το ιδιωτικό κλειδί είναι τυχαίο στοιχείο $\xi \leftarrow \mathbb{Z}_p$ και το δημόσιο κλειδί είναι το $h \leftarrow g^\xi \in \mathbb{G}_1$. Η κρυπτογράφηση T του μηνύματος m γίνεται επιλέγοντας τυχαιότητα $r \leftarrow_R \mathbb{Z}_p$ ώστε $T = (T_1, T_2) = (g^r, mh^r)$. Η `trapdoor` και ο αλγόριθμος σύγκρισης ακολουθούν.

- $\text{Aut}(\xi)$: Επιστρέφει την `trapdoor tk` $= (\hat{r}, \hat{t} = \hat{r}^\xi) \in \mathbb{G}_2^2$ για τυχαίο $\hat{r} \leftarrow_R \mathbb{G}_2$.
- $\text{Com}(T, T', tk)$: Με $T = (T_1, T_2) = (g^r, mh^r)$ και $T' = (T'_1, T'_2) = (g^{r'}, m'h^{r'})$ και `trapdoor tk` $= (\hat{r}, \hat{t} = \hat{r}^\xi)$ επιστρέφει 1 αν $e(T_2, \hat{r}) \cdot e(T_1, \hat{t})^{-1} = e(T'_2, \hat{r}) \cdot e(T'_1, \hat{t})^{-1}$ ισχύει και 0 διαφορετικά.

Παρατήρηση.

Δεδομένου ότι ισχύει το $co - CDH$ συμπέρασμα. Κρυπτοσύστημα $AoN - PKEEET$ που χρησιμοποιεί το ElGamal στην ομάδα \mathbb{G}_1 σε πλαίσιο $SXDH$ είναι ασφαλές.

2.7 Non-Interactive Plaintext (In-)Equality Proofs

Ενδιαφέρον παρουσιάζουν οι αποδείξεις ισότητας ή και ανισότητας κρυπτοκειμένων στις οποίες ο `prover` δεν έχει γνώση του ιδιωτικού κλειδιού ούτε της τυχαιότητας που χρησιμοποιήθηκε για την κρυπτογράφηση. Όπως έγινε προφανές από την προηγούμενη παράγραφο τα $AoN - PKEEET$ κρυπτοσυστήματα προσφέρουν αυτή την δυνατότητα εφοδιάζοντας τον `prover` με μια `trapdoor` η οποία του επιτρέπει τέτοιους ελέγχους χωρίς να μπορεί να αποκρυπτογραφήσει.

Αν $\mathcal{PKES} = (KeyGen, Enc, Dec, Aut, Com)$ ένα ασφαλές $AoN - PKEEET$ σχήμα. Ορίζουμε επί του \mathcal{PKES} ένα μη διαδραστικό σύστημα αποδείξεων Π ώστε αν T και T' δύο κρυπτοκειμένα που προκύπτουν με χρήση του ίδιου ιδιωτικού κλειδιού να επιτρέπει την απόδειξη γνώσης `trapdoor tk` που ελέγχει αν (T, T', pk) ανήκει σε γλώσσα $L_{R \in}$ ή $L_{R \notin}$.

$$((T, T', pk), tk) \in R_{\in} \iff Com(T, T', tk) = 1 \wedge tk \equiv pk,$$

$$((T, T', pk), tk) \in R_{\notin} \iff Com(T, T', tk) = 1 \wedge tk \equiv pk,$$

Το σύστημα αποδείξεων Π προκύπτει από τη σύνθεση δύο επιμέρους συστημάτων αποδείξεων των Π_ϵ και Π_ζ . Το Π_ϵ καλύπτει τα αιτήματα για τη γλώσσα L_{R_ϵ} ενώ το Π_ζ τα αιτήματα για τη γλώσσα L_{R_ζ} . Από την *soundness* ιδιότητα του κρυπτοσυστήματος κάθε τριάδα ανήκει είτε στην R_ϵ είτε στην R_ζ .

Συνοπτική Περιγραφή του Σχήματος Αποδείξεων.

- **BGGen(1^k)** : Με είσοδο μια παράμετρο ασφαλείας k επιστρέφει $BG \leftarrow \Pi_{\epsilon,\zeta}.BGGen(1^k)$. Τα δύο συστήματα αποδείξεων χρησιμοποιούν την ίδια *bilinear* ομάδα.
- **CRSGen(BG)** : Με είσοδο την περιγραφή μιας *bilinear* ομάδας BG παράγει $crs_\epsilon \leftarrow \Pi_\epsilon.CRSGen(BG)$ και $crs_\zeta \leftarrow \Pi_\zeta.CRSGen(BG)$. Στην έξοδο δίνει ένα *common reference string* $crs \leftarrow (crs_\epsilon, crs_\zeta)$.
- **Proof(BG, crs, (T, T', pk), tk)** : Με είσοδο την περιγραφή μιας *bilinear* ομάδας BG , έναν *common reference string* crs , δήλωση (T, T', pk) και *witness* tk ελέγχει
 - Αν $((T, T', pk), tk) \in R_\epsilon$ επιστρέφει $\pi_\epsilon \leftarrow \Pi_\epsilon.Proof(BG, crs_\epsilon, (T, T', pk), tk)$.
 - Διαφορετικά επιστρέφει $\pi_\zeta \leftarrow \Pi_\zeta.Proof(BG, crs_\zeta, (T, T', pk), tk)$.
- **Verify(BG, crs, (T, T', pk), π)** : Με είσοδο την περιγραφή μιας *bilinear* ομάδας BG , έναν *common reference string* crs , δήλωση (T, T', pk) και απόδειξη π
 - Αν η απόδειξη π αφορά τη γλώσσα L_{R_ϵ} επιστρέφει $\Pi_\epsilon.Verify(BG, crs_\epsilon, (T, T', pk), \pi)$.
 - Αν η απόδειξη π αφορά τη γλώσσα L_{R_ζ} επιστρέφει $\Pi_\zeta.Verify(BG, crs_\zeta, (T, T', pk), \pi)$.

Ένα μη διαδραστικό σύστημα ελέγχου ισότητας και ανισότητας κρυπτοκειμένων με παροχή απόδειξης καλείται *ασφαλές* αν είναι *perfectly sound*, *perfectly complete* και τουλάχιστον *computationally zero knowledge*.

Παρατήρηση.

Αν Π_ϵ και Π_ζ ασφαλή συστήματα αποδείξεων μηδενικής γνώσης τότε και το Π είναι επίσης ασφαλές. Το Π κληρονομεί τις ασθενέστερες ιδιότητες των Π_ϵ, Π_ζ .

2.8 Υλοποίηση

Θα παρουσιάσουμε υλοποίηση που βασίζεται στο κρυπτοσύστημα *ElGamal* στην ομάδα \mathbb{G}_1 σε πλαίσιο που η υπόθεση *SXDH* ισχύει.

Υπενθυμίζουμε ότι το δημόσιο κλειδί είναι το $pk = g^\xi \in \mathbb{G}_1$, η *trapdoor* $tk = (\hat{r}, \hat{t} = \hat{r}^\xi) \in \mathbb{G}_2^2$. Τα δύο κρυπτοκειμένα T, T' προκύπτουν από κοινό *plaintext* αν $e(T_2, \hat{r}) \cdot e(T_1, \hat{t})^{-1} = e(T'_2, \hat{r}) \cdot e(T'_1, \hat{t})^{-1}$.

Για έλεγχο στην R_ϵ οι ακόλουθες *PPEs* πρέπει να ικανοποιούνται

$$(((T_1, T_2), (T'_1, T'_2)), (\hat{r}, \hat{t})) \in R_{\in} \iff \begin{aligned} &e(g^{\xi}, \hat{r}) \cdot e(g^{-1}, \hat{t}) = 1_{\mathbb{G}_T} \wedge \hat{r} \neq 1_{\mathbb{G}_2} \wedge \hat{t} \neq \\ &1_{\mathbb{G}_2} \wedge e(T_2 \cdot T_2'^{-1}, \hat{r}) \cdot e(T_1^{-1} \cdot T'_1, \hat{t}) = 1_{\mathbb{G}_T} \end{aligned}$$

Για έλεγχο στην R_{\neq} οι ακόλουθες *PPEs* πρέπει να ικανοποιούνται

$$(((T_1, T_2), (T'_1, T'_2)), (\hat{r}, \hat{t})) \in R_{\neq} \iff \begin{aligned} &e(g^{\xi}, \hat{r}) \cdot e(g^{-1}, \hat{t}) \neq 1_{\mathbb{G}_T} \wedge \hat{r} \neq 1_{\mathbb{G}_2} \wedge \hat{t} \neq \\ &1_{\mathbb{G}_2} \wedge e(T_2 \cdot T_2'^{-1}, \hat{r}) \cdot e(T_1^{-1} \cdot T'_1, \hat{t}) = 1_{\mathbb{G}_T} \end{aligned}$$

Αποτέλεσμα.

Το *NIPPI* σύστημα αποδείξεων που προκύπτει από τη σύνθεση των $\Pi_i n$ και Π_{\neq} είναι ασφαλές, complete, sound και zero knowledge. Στη θέση του ElGamal μπορεί να χρησιμοποιηθούν κρυπτοσυστήματα όπως τα Cramer-Shoup, twin-ElGamal.

Chapter 3

Towards a Publicly-Verifiable Mix-Net Providing Everlasting Privacy

3.1 Εισαγωγή

Πρώτος ο David Chaum το 1981 χρησιμοποίησε Mix Nets με την πρόθεση να εξασφαλίσει ανώνυμη επικοινωνία σε δίκτυα. Βασική τους χρήση να επεξεργαστούν ένα σύνολο μηνυμάτων ώστε στην έξοδο τα περιεχόμενα να παραμείνουν αναλλοίωτα ενώ κάθε συσχέτιση μεταξύ ενός μηνύματος στην είσοδο και του αντίστοιχου μηνύματος στην έξοδο να είναι αδύνατη.

Ο Park το 1993 χρησιμοποίησε Mix Nets εφοδιασμένα με κάποιο αλγόριθμο κρυπτογράφησης (Reencryption Mix Nets) . Η ορθότητα του συγκεκριμένου αλγόριθμου είναι επαληθεύσιμη από κάθε ενδιαφερόμενο (Universally Verifiable Mix Nets) . Δηλαδή οποιοσδήποτε μπορεί να επαληθεύσει ότι μηνύματα που δέχεται ως είσοδο το πρωτόκολλο θα υπάρχουν και στην έξοδο αναλλοίωτα.

Μερικές μόνο από τις εφαρμογές των Mix Nets είναι στα πρωτόκολλα ηλεκτρονικών εκλογών, στη διεξαγωγή ηλεκτρονικών εξετάσεων κτλ.

Σημαντικό μειονέκτημα των Mix Nets είναι ακριβώς αυτή η χρήση ενός κρυπτογραφικού πρωτόκολλου. Η ασφάλεια προκύπτει ως συνέπεια ενός υπολογιστικά δύσκολου προβλήματος. Τι θα συμβεί όμως με την πάροδο του χρόνου (μετά ίσως από μερικές δεκαετίες) όταν το πρόβλημα πάψει να είναι δυσεπίλυτο; Τότε η ασφάλεια παραβιάζεται και κάθε μήνυμα στην έξοδο μπορεί να συσχετιστεί με το αρχικό του στην είσοδο.

Οι ανησυχίες αυτές δεν μπορούν να χαρακτηριστούν υπερβολικές καθώς η αποθήκευση κάθε πληροφορίας που θα δημοσιοποιήσει στην έξοδο του το πρωτόκολλο έχει πλέον αμελητέο κόστος ενώ η υπολογιστική ισχύ που διαθέτει ο μέσος χρήστης συνεχώς αυξάνεται (Moore's law).

Πριν την περιγραφή ενός πρωτόκολλου Mixing το οποίο προσφέρει Everlasting Privacy Towards the Public θα παρουσιάσουμε τους δομικούς λίθους που το απαρτίζουν.

3.2 Δομικοί Λίθοι

Το πρωτόκολλο χρησιμοποιεί δύο κρυπτογραφικά εργαλεία.

- Ένα Commitment Scheme ώστε να παρέχει Everlasting Privacy και Universal Verifiability ως προς το κοινό.
- Ένα Matching Encryption Scheme το οποίο επιτρέπει στις αρχές να ανοίξουν τις δεσμεύσεις στο τέλος του mixing χωρίς να τους αποκαλύπτει επιπλέον πληροφορία.

3.2.1 Commitment Scheme

Ένα (non interactive) Commitment scheme απαρτίζεται από τα εξής 3 πρωτόκολλα ($GenCom, Com, Unv$).

- Η συνάρτηση $GenCom$ παράγει το δημόσιο κλειδί δέσμευσης ck με είσοδο 1^k όπου k μία παράμετρος ασφάλειας. Η παράμετρος k καθορίζει τον χώρο μηνυμάτων \mathcal{M} καθώς και τον χώρο από τον οποίο αντλούνται τυχαίοι αριθμοί \mathcal{R} .
- Η συνάρτηση $Com(m, s)$ έχει ως είσοδο μήνυμα $m \in \mathcal{M}$, τυχαίο $s \in \mathcal{R}$ και παράγει μια δέσμευση $c \in \mathcal{C}$ όπου \mathcal{C} ο χώρος δεσμεύσεων. $c = Com(m, s) \in \mathcal{C}$.
- Ο αλγόριθμος $Unv(c, m, s)$ επιστρέφει m αν $c = Com(m, s)$ και \perp διαφορετικά.

Το κρυπτογραφικό αυτό πρωτόκολλο πρέπει να έχει τις ακόλουθες ιδιότητες:

1. **Correctness** . Για κάθε $m \in \mathcal{M}, r \in \mathcal{R} : Unv(Com(m, r), m, r) = m$.
2. **Non interactive** . Δεν είναι διαδραστικό. Όλη η επικοινωνία καταλήγει από τον αποστολέα στον παραλήπτη.
3. **Computationally Binding**. Για κάθε PPT αντίπαλο \mathcal{A} η πιθανότητα να υπολογίσει (m', r') με $m \neq m'$ τέτοια ώστε $Com(m, r) = Com(m', r')$ είναι αμελητέα ως προς k .
4. **Unconditionally Hiding**. Για κάθε $m, m' \in \mathcal{M}$ οι κατανομές των $Com(m', r')$ και $Com(m, r)$ είναι ταυτόσημες όταν τα $r, r' \in \mathcal{R}$ επιλέγονται ομοιόμορφα τυχαία.
5. **Homomorphic**. Για κάθε $m, m' \in \mathcal{M}$ και $r, r' \in \mathcal{R}$, $Com(m, r) \cdot c Com(m', r') = Com(m +_{\mathcal{M}} m', r +_{\mathcal{R}} r')$.

3.2.2 Encryption Scheme

Το πρωτόκολλο κρυπτογράφησης αποτελείται από τρεις επιμέρους αλγόριθμους ($GenEnc, Enc, Dec$).

1. Ο $GenEnc$ παράγει δύο χωριστά κλειδιά. Ένα δημόσιο κλειδί pk και ένα ιδιωτικό κλειδί sk το οποίο όμως διαμοιράζεται σε ένα σύνολο έμπιστων αρχών \mathcal{T} ώστε οι αποκρυπτογράφηση να είναι δυνατή μόνο αν συνεργαστούν k από τις n έμπιστες αρχές (Threshold decryption).

2. Με $Enc(t, s) = c$ συμβολίζεται η κρυπτογράφηση του μηνύματος $t \in G$ με τυχαιότητα $s \in H$ και δημόσιο κλειδί pk .
3. Με $Dec(c) = t$ συμβολίζεται η αποκρυπτογράφηση του κρυπτοκειμένου c με τη χρήση του ιδιωτικού κλειδιού sk .

Ο αλγόριθμος παρέχει *CCA* ασφάλεια (semantical security) και είναι ομομορφικός ως προς t και s , δηλαδή για όλα τα $t, t' \in G$ και για όλα τα $s, s' \in H$, $Enc(t, s) \cdot Enc(t', s') = Enc(t +_G t', s +_H s')$.

Είναι συνεπώς δυνατή η επανακρυπτογράφηση ενός μηνύματος t ($c = Enc(t, s)$) χωρίς γνώση του μηνύματος απλά πολλαπλασιάζοντας το με μια κρυπτογράφηση του O_G της ομάδας G , $ReEnc(c, s') = Enc(t, s) \cdot Enc(O_G, s') = Enc(t, s + s')$.

Το μήνυμα $m \in \mathcal{M}$ και η τυχαιότητα $r \in \mathcal{R}$ θα αποσταλούν μέσω ενός ιδιωτικού καναλιού. Είναι χρήσιμο να υπάρχουν δύο επιμέρους αλγόριθμοι κρυπτογράφησης. Ένας που θα είναι ομομορφικός στον χώρο των μηνυμάτων $Enc_{\mathcal{M}}$ και ένας ακόμη που θα είναι ομομορφικός στον χώρο τυχαιότητας, $Enc_{\mathcal{R}}$. Οι δύο αυτοί επιμέρους αλγόριθμοι συνθέτουν τον Enc .

3.2.3 Proofs of correct reencryption

Κάθε *mix* επισυνάπτει μία απόδειξη μηδενικής γνώσης ότι η είσοδος έχει επεξεργαστεί σωστά. Δηλαδή, ότι η έξοδος που προκύπτει είναι κάποιο κρυπτογραφημένο ανακάτεμα της εισόδου. Οι αποδείξεις είναι δημόσιες ώστε να μπορούν να επαληθευτούν από οποιονδήποτε ενδιαφερόμενο.

3.2.4 Proofs of consistency

Κάθε *mix* επισυνάπτει μία ιδιωτική απόδειξη μηδενικής γνώσης ότι η ίδια μετάθεση και οι ίδιες τυχαίες τιμές έχουν χρησιμοποιηθεί στο ανακάτεμα των δημοσιευμένων δεσμεύσεων και την επανακρυπτογράφηση των αντίστοιχων αρχικών τυχαίων τιμών.

Και οι δύο αποδείξεις πρέπει να είναι *Perfect zero knowledge* έτσι που αν η απόδειξη που έχει δημοσιευθεί είναι ορθή καμία επιπλέον πληροφορία να μην διαρρέει.

Ακολουθούν πρωτόκολλα *Commitment Schemes* και κρυπτοσυστήματα που ικανοποιούν τις παραπάνω προδιαγραφές.

- Για το *Split Ballot* πρωτόκολλο ηλεκτρονικών εκλογών οι Moran και Noar πρότειναν τη χρήση του κρυπτοσυστήματος Paillier και μία παραλλαγή των δεσμεύσεων Pedersen ([1]).
- Ενδιαφέρον παρουσιάζει η πρόταση του Pereira et al. ([2]) που παρουσιάζει ένα *efficient unconditionally hiding* σχήμα δέσμευσης. Το κρυπτοσύστημα που το συνοδεύει βασίζεται σε ελλειπτικές καμπύλες. Σε συνδυασμό με μία *non interactive* απόδειξη [3, 4] θα μπορούσε να οδηγήσει σε πολύ αποδοτικά *mixnet*.

3.3 Security

Η ασφάλεια ενός τυπικού Mix Net εξασφαλίζεται αν ικανοποιούνται οι ακόλουθες υποθέσεις:

1. Οι αρχές δεν μπορούν να σπάσουν το υπολογιστικό πρόβλημα στο οποίο βασίζεται η ασφάλεια του αλγόριθμου κρυπτογράφησης.
2. Τουλάχιστον ένα Mix είναι τίμιο και δεν αποκαλύπτει τον συσχετισμό μεταξύ εισόδου και εξόδου, κρατά δηλαδή μυστική την μετάθεση που χρησιμοποιεί.
3. Χρησιμοποιώντας (k, n) – threshold αποκρυπτογράφηση τουλάχιστον $n - k + 1$ έμπιστες αρχές δρουν τίμια και δεν αποκαλύπτουν το μέρος από το ιδιωτικό κλειδί που κατέχουν.
4. Όλοι οι τυχαίοι αριθμοί που χρησιμοποιούνται στο πρωτόκολλο προκύπτουν από ένα έμπιστο Random Beacon και δεν μπορούν να προβλεφθούν.
5. Οι αρχές δεν μπορούν να σπάσουν την computational binding ιδιότητα του commitment scheme για τις παραμέτρους που έχουν καθοριστεί πριν την έναρξη του πρωτοκόλλου.
6. Υπάρχουν ιδιωτικά κανάλια επικοινωνίας μεταξύ των χρηστών και του πρώτου mix, μεταξύ δύο διαδοχικών mix καθώς και μεταξύ του τελικού mix και των έμπιστων αρχών \mathcal{T} .
7. Με την λήξη του πρωτοκόλλου, όλες οι αρχές καταστρέφουν όλες τις πληροφορίες που έχουν εμπιστευτικά διαχειριστεί.

3.4 Mixing Process

Κάθε ομάδα μηνυμάτων που έχει σταλεί από έναν χρήστη παριστάνεται με ένα διάνυσμα. Κάθε διάνυσμα συμβολίζεται με ένα κεφαλαίο γράμμα.

Ορίζουμε $Perm_\pi(T) = T'$ όπου $t'(i) = t(\pi(i))$ για κάποια μετάθεση π .

3.4.1 Υποβολή Μηνύματος

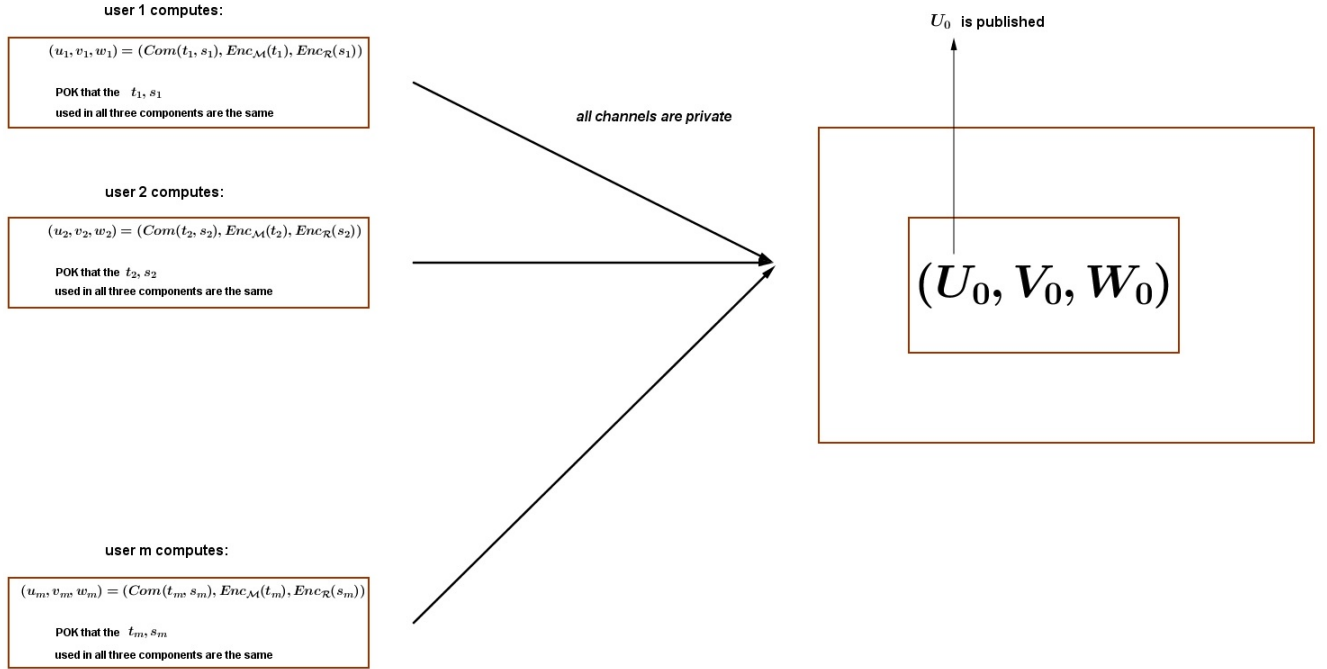
Για την υποβολή κάποιου μηνύματος $t \in \mathcal{M}$ κρυπτογραφημένο με τυχαιότητα s ο χρήστης υπολογίζει και υποβάλλει μία τριάδα

$$(u, v, w) = (Com(t, s), Enc_{\mathcal{M}}(t), Enc_{\mathcal{R}}(s))$$

που συνοδεύεται από απόδειξη γνώσης *POK* ότι τα t, s που χρησιμοποιήθηκαν είναι ίδια σε κάθε συντεταγμένη. Ο χρήστης μέσω ιδιωτικού καναλιού στέλνει την τριάδα (u, v, w) καθώς και την *POK*.

Ως είσοδο για το πρώτο «ανακάτεμα» είναι το σύνολο των τριάδων που έχει συγκεντρωθεί διατεταγμένο με κάποιο προκαθορισμένο τρόπο. Συμβολίζεται (U_0, V_0, W_0) . Το δημόσιο κομμάτι U_0

δημοσιεύεται.



3.5 Mixing

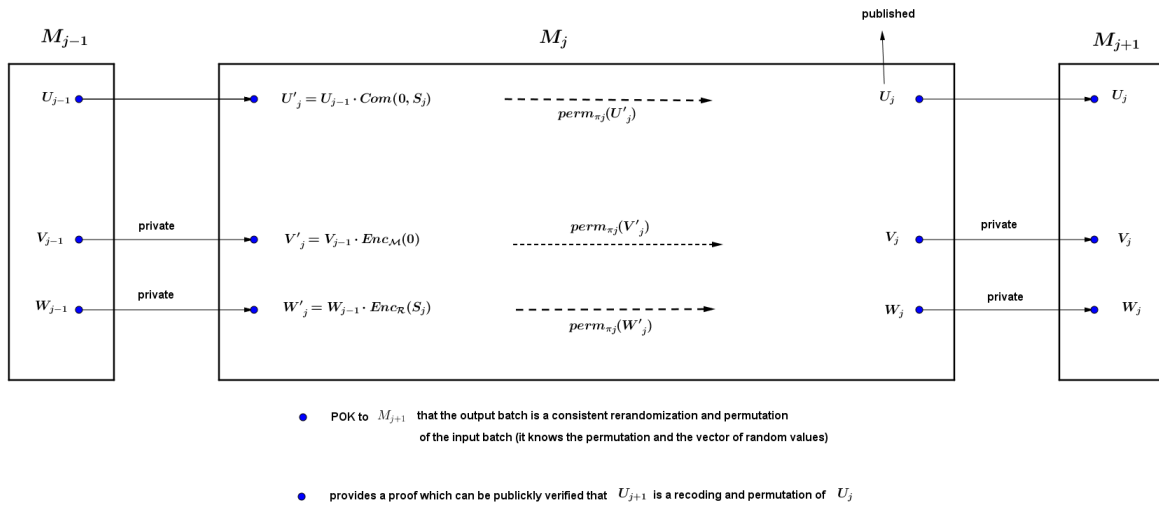
Θα περιγράψουμε τη διαδικασία για είσοδο που αποτελείται από k τριάδες και n mixes, M_1, M_2, \dots, M_n .

Είσοδος του M_j είναι η έξοδος του M_{j-1} . Προφανώς είσοδος του M_1 είναι οι τριάδες που έχουν αποσταλεί από τους χρήστες ενώ η έξοδος του M_n προωθείτε σε όποια αρχή είναι υπεύθυνη για την επεξεργασία των αποτελεσμάτων. Σε κάθε επιμέρους ανακάτεμα η διαδικασία είναι ίδια.

Έστω λοιπόν ότι το διάνυσμα που δέχεται ως είσοδο το M_j είναι $(U_{j-1}, V_{j-1}, W_{j-1})$. Τότε τα ακόλουθα συμβαίνουν:

1.
 - $U'_j = U_{j-1} \cdot Com(0, S_j)$. Από την ομομορφική ιδιότητα του σχήματος δέσμησης επηρεάζεται στο βήμα αυτό η επιλεγμένη τυχαιότητα.
 - $V'_j = V_{j-1} \cdot Enc_{\mathcal{M}}(0)$. Κρυπτογραφεί εκ νέου (χωρίς να αλλάζει το περιεχόμενο, πρακτικά επηρεάζει ξανά την τυχαιότητα).
 - $W'_j = W_{j-1} \cdot Enc_{\mathcal{R}}(S_j)$. Ενημερώνει ώστε να είναι δυνατή η αποκρυπτογράφηση.
2. Έπειτα επιλέγεται μετάθεση π_j και $(U_j, V_j, W_j) = Perm_{\pi_j}(U'_j, V'_j, W'_j)$.
3. Οι δεσμεύσεις U_j δημοσιεύονται ενώ τα (V_j, W_j) στέλνονται στο M_{j+1} μέσω ενός ιδιωτικού καναλιού.
4. Το M_j αποδεικνύει (με τρόπο που να μπορεί να επαληθευτεί δημόσια) ότι το διάνυσμα U_j είναι επανακρυπτογράφηση και μετάθεση του U_{j-1} .

5. Παρέχει *POK* ότι το διάνυσμα εξόδου είναι επανακρυπτογράφηση και μετάθεση του διανύσματος εισόδου. Αποδεικνύει ότι γνωρίζει τη μετάθεση π_j καθώς και το διάνυσμα με τυχαίες τιμές S_j . Υπεύθυνος για την επαλήθευση της εξόδου του M_n είναι η εκάστοτε αρχή.



3.5.1 Αποκρυπτογράφηση και Δημοσίευση

Η υπεύθυνη αρχή υπολογίζει και δημοσιεύει τα $T^* = Dec(V_n)$ και $S^* = Dec(W_n)$. Εφόσον οι ίδιες μεταθέσεις έχουν χρησιμοποιηθεί είτε το πρωτόκολλο εκτελείται δημόσια είτε ιδιωτικά $U_n = Com(T^*, S^*)$.

3.5.2 Πιστοποίηση

Μετά την επαλήθευση του $U_n = Com(T^*, S^*)$ και τον έλεγχο ότι όλες οι αποδείξεις γνώσης είναι αληθείς πιστοποιείται το αποτέλεσμα.

3.6 Ιδιότητες

3.6.1 Correctness

Θα αποδείξουμε ότι τα Mixes δεν έχουν παραποιήσει κάποια από τα μηνύματα. Δηλαδή, ότι $T_0 \equiv T_n$ όπου με \equiv συμβολίζεται η ύπαρξη μετάθεσης που απεικονίζει το T_0 στο T_n .

Η δυσκολία στο υπάρχων πρωτόκολλο είναι ότι ενώ επιδιώκουμε το αποτέλεσμα να είναι επαληθεύσιμο από κάθε ενδιαφερόμενο τρίτο, μόνο δημόσιες πληροφορίες μπορούν να χρησιμοποιηθούν για την επαλήθευση αυτή.

Η επαλήθευση βασίζεται στον ακόλουθο ισχυρισμό.

“Εφόσον οι έμπιστες αρχές \mathcal{T} μπορούν να ανοίξουν τις δεσμεύσεις που έχουν δημοσιευτεί από το τελευταίο mix δηλαδή T^* και S^* ώστε $U_n = Com(T^*, S^*)$ τότε $T_0 \equiv T_n$. ”

Απλούστερα, λίγη σημασία έχει το «πως» κοινοποιήθηκαν στις έμπιστες αρχές τα T^*, S^* .

Η αλήθεια του ισχυρισμού προκύπτει άμεσα από την ιδιότητα του Commitment Scheme να είναι unconditionally hiding και computationally binding.

Ας υποθέσουμε ότι με το τέλος του πρωτοκόλλου $T_0 \neq T_n$. Τότε υπάρχει ένας αποδοτικός αλγόριθμος ο οποίος σπάει την binding ιδιότητα υπολογίζοντας u που αντιστοιχεί σε δύο διαφορετικές τιμές $u = Com(\tau_1, \sigma_1) = Com(\tau_2, \sigma_2)$.

Εφόσον το κομμάτι αυτό της πληροφορίας είναι δημόσιο σε κάθε επιμέρους βήμα όλοι μπορούν να επαληθεύσουν τις αποδείξεις μηδενικής γνώσης. Να επισημανθεί ότι είναι (σε αυτό το στάδιο) σημαντική η τυχαιότητα στα challenge bit. Φυσικά, κάθε άμεσα ενδιαφερόμενος μπορεί να ελέγξει αν το μήνυμα του έχει δημοσιευτεί.

Αν τουλάχιστον ένα Mix είναι τίμιο τότε δεν μπορεί να υπάρχει συσχέτιση της εξόδου με την είσοδο. Ακόμη κι ένας αντίπαλος με απεριόριστη υπολογιστική ισχύ μπορεί να εξάγει μία μόνο πληροφορία «η έξοδος είναι μία άγνωστη μετάθεση της εισόδου...» εφόσον οι δεσμεύσεις είναι unconditional και οι αποδείξεις μηδενικής γνώσης προσφέρουν perfect zero knowledge.

3.6.2 Robustness

Αν όλοι οι συμμετέχοντες είναι τίμιοι τότε το πρωτόκολλο πάντα επιτυγχάνει. Αν εντοπιστεί κάποιο κακόβουλο mix μπορεί απλά να αντικατασταθεί ή μπορεί απλά να αντικατασταθεί ολόκληρο το mix net. Επιπρόσθετα για να αποφευχθούν κακόβουλες συμπεριφορές είναι δυνατό να απαιτηθεί από mixes και χρήστες να υπογράφουν τις εξόδους. Επίσης, οι χρήστες να υποβάλλουν αποδείξεις γνώσης ως προς τη συνέπεια των εξόδων τους αποκλείοντας εξόδους μη συνεπείς ως προς την αναμενόμενη μορφή εξασφαλίζοντας τον εντοπισμό παθογόνων συμπεριφορών.

3.7 Everlasting Privacy Towards the Authorities

Ένα μειονέκτημα του πρωτοκόλλου που παρουσιάσαμε είναι ότι αν και το κανάλι επικοινωνίας είναι ιδιωτικό ανάμεσα στον χρήστη και το πρώτο mix, πράγμα που τον προστατεύει από κακόβουλους τρίτους δεν τον προστατεύει από ένα κακόβουλο πρώτο mix. Με το πέρας δηλαδή του χρόνου, όταν το υπολογιστικό πρόβλημα στο οποίο βασίζεται η ασφάλεια της δέσμευσης ή και του αλγόριθμου κρυπτογράφησης σπάσει, το μήνυμα θα γίνει γνωστό.

Αν αυτό δεν είναι αποδεκτό μία πιθανή λύση είναι ο διαμερισμός του μηνύματος σε n επιμέρους κομμάτια και η δημιουργία n διαφορετικών mix net. Οι χρήστες στέλνουν ένα μόνο κομμάτι του μηνύματος σε κάθε mix. Το νέο πρόβλημα που προκύπτει είναι πως θα ανακτήσουμε το αρχικό μήνυμα από τις τελικές εξόδους των n διαφορετικών mix net.

Ας υποθέσουμε ότι κάθε χρήστης εφοδιάζει το μήνυμα του με ένα τυχαία επιλεγμένο r_i αρκετά μεγάλο (128 bits) για να αποφευχθούν συμπτώσεις με άλλους χρήστες. Οι αριθμοί αυτοί δεν

μπορούν φυσικά να είναι δημόσιοι, αποτρέποντας κάποιο κακόβουλο *mix* ή έναν κακόβουλο *verifier* να συνθέσει το αρχικό μήνυμα από τα επιμέρους κομμάτια του.

Ο χρήστης 1 διαμοιράζει το μήνυμα t_1 σε n επιμέρους κομμάτια $t_{1,1}, t_{1,2}, \dots, t_{1,n}$. Η διαδικασία που ακολουθεί είναι ανάλογη με την προηγούμενη. Για κάθε επιμέρους τμήμα $t_{1,i}$ ο χρήστης υποβάλλει στο i *mix net*

$$((Com(t_{1,i}, s_{1,i}), Com(r_1, w_{1,i})), Enc_{\mathcal{M}}(t_{1,i}), Enc_{\mathcal{M}}(r_1), Enc_{\mathcal{R}}(s_{1,i}), Enc_{\mathcal{R}}(w_{1,i}))$$

Τις εξόδους των n *mix nets* λαμβάνουν παραδοσιακά οι έμπιστες αρχές \mathcal{T} καθώς και μια επιπλέον αρχή \mathcal{Z} που κρατά μυστικά τα επιμέρους r_i . Σε κάθε αποκρυπτογραφημένο τμήμα ενός μηνύματος θα υπάρχει η ίδια ταυτότητα r_i την οποία και θα χρησιμοποιήσει η αρχή \mathcal{Z} για να συνθέσει το αρχικό μήνυμα και να το δημοσιεύσει. Η αρχή \mathcal{Z} επωμιζεται με το καθήκον να δημοσιεύσει απόδειξη γνώσης ότι τα αρχικά μηνύματα προκύπτουν συνθέτοντας μηνύματα που διαθέτουν τη ίδια ταυτότητα r_i .

Όσο η αρχή \mathcal{Z} δρα τίμια και δεν μοιράζεται τις πληροφορίες που διαθέτει με κάποιο από τα πρώτα *mixes*, όσο τα πρώτα *mixes* δεν συνεργάζονται μεταξύ τους ανταλλάσσοντας πληροφορίες το πρωτόκολλο παρέχει *unconditional privacy towards the authorities*. Ακόμη κι αν υπάρχουν κακόβουλα *mixes* ή κακόβουλοι *verifiers* όσο δεν έχουν την συνδρομή ενός ικανοποιητικού μέρους των έμπιστων αρχών \mathcal{T} πρέπει επίσης να σπάσουν τον αλγόριθμο κρυπτογράφησης.

Όσο όλοι οι παράγοντες δρουν τίμια το πρωτόκολλο τερματίζει με επιτυχία, εξαιρώντας ίσως την απίθανη περίπτωση που δύο συμμετέχοντες επιλέξουν το ίδιο αριθμό r_i . Ακόμη κι αν κάποιος κακόβουλος συμμετέχων επιλέξει να διαμοιράσει το μήνυμα του εφοδιάζοντας το με διαφορετικές επιμέρους ταυτότητες r_1, r_2, \dots, r_n δεν υπάρχει κάποιος προφανής τρόπος με τον οποίο αυτό θα επηρεάσει το υπόλοιπο πρωτόκολλο. Το μήνυμα απλά δεν θα προσμετρηθεί στο τελικό αποτέλεσμα χωρίς αυτό να επηρεάσει τα μηνύματα των υπόλοιπων χρηστών.

Chapter 4

Coercion-Resistant Electronic Elections

Στο παρόν κεφάλαιο περιγράφονται οι βασικές ιδέες του άρθρου Coercion-Resistant Electronic Elections των Ari Juels, Dario Catalano, Markus Jakobson ([7]).

Ως τώρα τα εκλογικά πρωτόκολλα περιορίζονταν σε μία ιδιότητα γνωστή ως Receipt Freeness. Συνοπτικά θα λέγαμε ότι ο σχεδιασμός του πρωτοκόλλου δεν επέτρεπε στον ψηφοφόρο να αποδείξει ότι ψήφισε με κάποιον συγκεκριμένο τρόπο ακόμη κι αν ο ίδιος το επιθυμούσε. Για ένα πιο αυστηρό ορισμό [5]. Δωροδοκία καθώς και προσπάθειες καθοδήγησης του ψηφοφόρου δεν έχουν νόημα.

Υπάρχουν επιθέσεις που δεν καλύπτονται από την ιδιότητα Receipt Freeness.

1. **Randomization attack:** Η συγκεκριμένη επίθεση επισημάνθηκε από τους Schoenmakers [6]. Ο αντίπαλος εξαναγκάζει τον ψηφοφόρο να υποβάλλει ψηφοδέλτιο το οποίο περιέχει τυχαία συμβολοσειρά. Στην συγκεκριμένη επίθεση πιθανόν ούτε ο ψηφοφόρος ούτε ο εκβιαστής μαθαίνουν την επιλογή του ψηφοφόρου. Αν υποθέσουμε όμως ότι στην εκλογική περιφέρεια του ψηφοφόρου υπερτερεί σημαντικά η παράταξη A έναντι της παράταξης B. Μία τέτοια επίθεση ευνοεί σημαντικά την παράταξη B.
2. **Forced abstention attack:** Ο αντίπαλος εκβιάζει τον ψηφοφόρο και του απαγορεύει την συμμετοχή στις εκλογές.
3. **Simulation attack:** Ο αντίπαλος απαιτεί από τον ψηφοφόρο να του παραδώσει τα προσωπικά του διαπιστευτήρια και ψηφίζει στην θέση του.

Μία γενικότερη ιδιότητα που θα παρέχει ασφάλεια ενάντια σε επιθέσεις τέτοιας μορφής είναι η Coercion Resistance.

4.1 Modelling

4.1.1 Αρχές.

Ένα πρωτόκολλο εκλογών απαρτίζεται από διάφορες οντότητες:

1. **Registrars.** Συμβολίζονται με $\mathcal{R} = \{R_1, R_2, \dots, R_{n_{\mathcal{R}}}\}$ και αποτελούν μία ομάδα αρχών που επωμίζονται την ευθύνη της από κοινού έκδοσης διαπιστευτηρίων στους ψηφοφόρους.
2. **Tallying Authorities.** Συμβολίζονται με $\mathcal{T} = \{T_1, T_2, \dots, T_{n_{\mathcal{T}}}\}$ και αποτελούν μια ομάδα αρχών που επωμίζονται την ευθύνη της από κοινού επεξεργασίας, καταμέτρησης των έγκυρων ψηφοδελτίων και έκδοσης του τελικού αποτελέσματος.
3. **Voters.** Το σύνολο των ψηφοφόρων $\mathcal{V} = \{V_1, V_2, \dots, V_{n_{\mathcal{V}}}\}$ που συμμετέχουν στη συγκεκριμένη εκλογική διαδικασία.

4.1.2 Συναρτήσεις.

Γίνεται επίσης χρήση ενός πίνακα \mathcal{BB} ο οποίος είναι γνωστός ως **Bulletin Board**. Ο πίνακας είναι δημόσιος, όλοι οι χρήστες έχουν δικαίωμα εγγραφής κανείς δεν έχει δικαίωμα διαγραφής. Όλοι μπορούν να δουν τα περιεχόμενα του πίνακα όταν η κατάθεση των ψηφοδελτίων έχει ολοκληρωθεί. Για λόγους ευκολίας θεωρούμε ότι η πληροφορία στον πίνακα είναι γραμμένη σε μ -bit πακέτα για κατάλληλο μ . Αν χρειάζεται κομμάτια πληροφορίας μπορούν να αποκτήσουν το κατάλληλο μήκος με *padding*.

Ορίζουμε ως **Candidate Slate** μία διατεταγμένη λίστα από $n_{\mathcal{C}}$ δείκτες $\{c_1, c_2, \dots, c_{n_{\mathcal{C}}}\}$ που αντιστοιχούν στους $n_{\mathcal{C}}$ υποψήφιους. Η επιλογή c_j καθορίζεται πλήρως από τον δείκτη j συνεπώς μπορούμε να θεωρήσουμε το Candidate Slate ως $\{1, 2, \dots, n_{\mathcal{C}}\}$ και καθορίζεται πλήρως από τον $n_{\mathcal{C}}$.

Ορίζουμε ως **Tally** του Candidate Slate \mathcal{C} το διάνυσμα X των $n_{\mathcal{C}}$ θετικών ακέραιων $\{x_1, x_2, \dots, x_{n_{\mathcal{C}}}\}$ ώστε ο x_j να φανερώνει το πλήθος των ψήφων του c_j .

Ακολουθούν οι βασικές συναρτήσεις του πρωτοκόλλου.

1. **Registering.** Η συνάρτηση $register(SK_{\mathcal{R}}, i, k_1) \rightarrow (sk_i, pk_i)$ με είσοδο το ιδιωτικό κλειδί των αρχών \mathcal{R} , $SK_{\mathcal{R}}$, μία παράμετρο ασφάλειας k_1 και τον δείκτη i που υποδεικνύει τον αντίστοιχο ψηφοφόρο παράγει ζεύγος κλειδιών sk_i, pk_i για τον ψηφοφόρο i . Ο υπολογισμός γίνεται από κοινού με πιθανή συνεργασία και του ψηφοφόρου i .
2. **Voting.** Η συνάρτηση $vote(sk, PK_{\mathcal{T}}, n_{\mathcal{C}}, \beta, k_2) \rightarrow ballot$ με είσοδο το ιδιωτικό κλειδί του ψηφοφόρου sk , το δημόσιο κλειδί των αρχών \mathcal{T} , $PK_{\mathcal{T}}$, τον $n_{\mathcal{C}}$ που καθορίζει το Candidate Slate, την επιλογή των υποψήφιων β και μία παράμετρο ασφάλειας k_2 παράγει ψηφοδέλτιο μήκους μ -bit το πολύ. Η μορφή του ψηφοδελτίου ποικίλει ανάλογα με το εκλογικό πρωτόκολλο.
3. **Tallying.** Η συνάρτηση $tally(SK_{\mathcal{T}}, \mathcal{BB}, n_{\mathcal{C}}, \{pk_i\}_{i=1}^{n_{\mathcal{V}}}, k_3) \rightarrow (X, P)$ με είσοδο το ιδιωτικό κλειδί των αρχών \mathcal{T} , $SK_{\mathcal{T}}$ τα περιεχόμενα του Bulletin Board, το μέγεθος του Candidate Slate, τα δημόσια κλειδιά των ψηφοφόρων και μια παράμετρο ασφάλειας k_3 παράγει το εκλογικό αποτέλεσμα tally X καθώς και μη διαδραστική απόδειξη P ότι έχει υπολογιστεί σωστά.

-
-
4. **Verifying.** Η συνάρτηση $verify(PK_{\mathcal{T}}, \mathcal{BB}, n_C, X, P) \rightarrow \{0, 1\}$ με είσοδο το δημόσιο κλειδί των αρχών, τα περιεχόμενα του Bulletin Board , το μέγεθος του Candidate Slate , το αποτέλεσμα X , και μια μη διαδριστική απόδειξη P της ορθότητας του υπολογισμού επιστρέφει 1 αν ο υπολογισμός είναι σωστός και 0 διαφορετικά.

Ένα εκλογικό πρωτόκολλο \mathcal{ES} είναι το σύνολο αυτών των συναρτήσεων.

$$\mathcal{ES} = \{register, vote, tally, verify\}$$

Θεωρούμε ότι μια εκλογική διαδικασία απαρτίζεται από τις ακόλουθες επιμέρους διαδικασίες:

1. **Setup.** Ζευγάρια κλειδιών παράγονται από ή για τις \mathcal{R}, \mathcal{T} . Το Candidate Slate δημοσιεύεται από τις \mathcal{R} .
2. **Registration.** Οι αρχές \mathcal{R} ελέγχουν τους συμμετέχοντες στην εκλογική διαδικασία. Μετά το πέρας του ελέγχου αποδίδονται διαπιστευτήρια που επιτρέπουν στους ψηφοφόρους τη συμμετοχή τους στην εκλογική διαδικασία. Ψηφοφόροι που έχουν διαπιστευτήρια από προηγούμενες εκλογές μπορούν να τα χρησιμοποιήσουν. Οι αρχές \mathcal{R} δημοσιεύουν την λίστα L με τα κρυπτογραφημένα διαπιστευτήρια των ψηφοφόρων (voter roll).
3. **Voting.** Οι ψηφοφόροι καταθέτουν την ψήφο τους χρησιμοποιώντας τα προσωπικά τους διαπιστευτήρια.
4. **Tallying.** Οι αρχές \mathcal{T} επεξεργάζονται τα περιεχόμενα του Bulletin Board και παράγουν το εκλογικό αποτέλεσμα X μαζί με απόδειξη P για την ορθότητα του.
5. **Verification.** Οποιοσδήποτε, είτε συμμετέχει είτε όχι στην εκλογική διαδικασία, μπορεί να δει το περιεχόμενο του Bulletin Board , την απόδειξη P και τη λίστα ψηφοφόρων L για να επαληθεύσει την ορθότητα του αποτελέσματος που εκδόθηκε από τις αρχές \mathcal{T} .

Ακολουθούν υποθέσεις που αφορούν τις ικανότητες ενός αντιπάλου στις φάσεις που προηγούνται.

1. **Setup.** Οι υποθέσεις του πρωτοκόλλου επιτρέπουν static, active διαφθορά μειοψηφίας συμμετεχόντων στις αρχές \mathcal{R}, \mathcal{T} . Η ασφάλεια τότε στηρίζεται στην παραγωγή των κλειδιών $(SK_{\mathcal{T}}, PK_{\mathcal{T}}), (SK_{\mathcal{R}}, PK_{\mathcal{R}})$ από κάποια έμπιστη αρχή ή με ένα διαδραστικό πρωτόκολλο παραγωγής κλειδιών μεταξύ των συμμετεχόντων στις \mathcal{R}, \mathcal{T} .
2. **Πριν από την εγγραφή του ψηφοφόρου.** Οι υποθέσεις του πρωτοκόλλου επιτρέπουν τον εκβιασμό του ψηφοφόρου πριν από την εγγραφή του στους εκλογικούς καταλόγους. Ο εκβιαστής μπορεί να απαιτήσει από τον ψηφοφόρο αντίγραφα από τη διαδικασία εγγραφής ή και να καθοδηγήσει τον ψηφοφόρο προσφέροντας του έγγραφα που θα χρησιμοποιήσει κατά την εγγραφή του.
3. **Κατά την εγγραφή του ψηφοφόρου.** Οι υποθέσεις του πρωτοκόλλου ΔΕΝ επιτρέπουν τον εκβιασμό του ψηφοφόρου κατά την διαδικασία εγγραφής του. Η υπόθεση αυτή είναι προφανώς προαπαιτούμενη καθώς αν ο εκβιαστής μπορεί σε αυτή τη διαδικασία να εκδιώξει τον ψηφοφόρο μπορεί απλά να αποκτήσει τα διαπιστευτήρια του και να ψηφίσει ο ίδιος στη θέση του. Ειδικότερα μία από τις ακόλουθες υποθέσεις πρέπει να ισχύουν:

- (α') Η διαγραφή των δεδομένων από την επικοινωνία του ψηφοφόρου με τις αρχές \mathcal{R} είναι αναγκαστική και επιβάλλεται με κάποιο μέσο (smart cards). Αυτό εμποδίζει τον ψηφοφόρο να παρέχει αντίγραφα της επικοινωνίας του στον εκβιαστή.
- (β') Ο αντίπαλος δεν μπορεί να διαφθείρει κανένα από τους συμμετέχοντες στις αρχές \mathcal{R} .
- (γ') Οι ψηφοφόροι γνωρίζουν ποιοι από τους συμμετέχοντες στις αρχές \mathcal{R} είναι διεφθαρμένοι.

Μία από τις τρεις υποθέσεις που προηγούνται πρέπει να ισχύει. Διαφορετικά ο εκβιαστής μπορεί να απαιτήσει αντίγραφα της επικοινωνίας του ψηφοφόρου με τις αρχές και να επαληθεύσει την ορθότητα των εγγράφων με μεγάλη πιθανότητα. Αυτό οδηγεί σε αποτελεσματικό εκβιασμό του ψηφοφόρου ο οποίος παραδίδει τα (πραγματικά) αντίγραφα της επικοινωνίας του και ο εκβιαστής ψηφίζει στη θέση του.

Μετά την διαδικασία εγγραφής ο αντίπαλος μπορεί να διαφθείρει μειοψηφία συμμετεχόντων στις αρχές \mathcal{T} και οποιοδήποτε αριθμό ψηφοφόρων με static, active τρόπο. Επειδή οι αρχές \mathcal{R} δεν συμμετέχουν στη φάση αυτή δεν έχει σημασία η συμπεριφορά τους. Ο αντίπαλος μπορεί επίσης να επιχειρήσει τον εκβιασμό ψηφοφόρων που δεν ελέγχει είτε ζητώντας να του παραδώσουν τα διαπιστευτήρια τους είτε αναγκάζοντας τους να συμπεριφερθούν με προκαθορισμένο τρόπο.

Αναγκαία προϋπόθεση είναι η ύπαρξη ανώνυμων καναλιών επικοινωνίας. Διαφορετικά αν ο εκβιαστής μπορεί να διαπιστώσει αν κάποιος ψηφοφόρος έλαβε μέρος στην εκλογική διαδικασία θα του απαγορεύσει απλά τη συμμετοχή του (forced abstention attack) .

4.2 Ορισμοί.

Στην παράγραφο αυτή θα στρέψουμε την προσοχή μας στους βασικούς ορισμούς ασφάλειας καθώς και στις ιδιότητες που πρέπει να έχει το πρωτόκολλο, correctness, verifiability, coercion resistance σύντομα *corr, ver, c – resist*. Οι ορισμοί προκύπτουν με τη βοήθεια πειραμάτων μεταξύ ενός αντιπάλου \mathcal{A} και του εκλογικού πρωτοκόλλου \mathcal{ES} . Η κατασκευή των πειραμάτων είναι τέτοια ώστε ο αντίπαλος επιτυγχάνει όταν η έξοδος του πειράματος είναι 1.

Για παράδειγμα στο πείραμα $EXP_{\mathcal{ES}, \mathcal{A}}^E(\cdot)$ όπου E μία από τις ιδιότητες $E \in (ver, corr, c-resist)$ ορίζουμε ως

$$Succ_{\mathcal{ES}, \mathcal{A}}^E(\cdot) = Pr[EXP_{\mathcal{ES}, \mathcal{A}}^E(\cdot) = 1].$$

Coercion Resistance: Μπορεί να θεωρηθεί ως επέκταση της βασικής ιδιότητας της μυστικότητας που πρέπει να ικανοποιεί ένα εκλογικό πρωτόκολλο.

Ένα πρωτόκολλο ηλεκτρονικών εκλογών διαφυλάσσει την μυστικότητα αν ο αντίπαλος δεν μπορεί να μαντέψει την ψήφο οποιουδήποτε ψηφοφόρου καλύτερα από ένα αλγόριθμο που έχει ως είσοδο το εκλογικό αποτέλεσμα.

Η αντοχή σε εκβιασμούς είναι μία επιπλέον ασφαλιστική δικλείδα που αφορά την μυστικότητα της ψήφου. Ο αντίπαλος μπορεί να χρησιμοποιήσει τα διαπιστευτήρια ψηφοφόρων για να ψηφίσει στη θέση τους ή ακόμα να αναγκάσει ψηφοφόρους να καταθέσουν ψηφοδέλτια συγκεκριμένης

μορφής. Αν ο εκβιαστής αντίπαλος μπορεί να διαπιστώσει αν οι ψηφοφόροι έχουν υπακούσει στις εντολές του τότε είναι σε θέση να εκβιάσει ψηφοφόρους και ασκεί ανεπιθύμητη επίδραση στο εκλογικό σώμα.

Συνεπώς, ανθεκτικό σε εκβιασμούς είναι το εκλογικό πρωτόκολλο στο οποίο ο ψηφοφόρος μπορεί να παραπλανήσει τον εκβιαστή ότι έπραξε κατά τα λεγόμενα του ενώ στην πραγματικότητα κατάθεσε ψηφοδέλτιο με τις προσωπικές του επιλογές.

Είναι στο σημείο αυτό αναγκαία η εισαγωγή μιας νέας συνάρτησης

$$fakekey(PK_{\mathcal{T}}, sk, pk) \rightarrow \overline{sk}$$

η οποία έχει ως είσοδο το δημόσιο κλειδί των αρχών \mathcal{T} , $PK_{\mathcal{T}}$, τα κλειδιά του ψηφοφόρου pk, sk και παράγει ένα ψεύτικο κλειδί \overline{sk} . Για να συνδράμει η συνάρτηση $fakekey$ στην αύξηση της ανθεκτικότητας σε εκβιασμούς πρέπει ο αντίπαλος \mathcal{A} να μην μπορεί να ξεχωρίσει το κλειδί \overline{sk} από το πραγματικό. Η διάκριση αληθινού και ψευδούς διαπιστευτηρίου είναι δυνατή μόνο από μία πλειοψηφία αρχών \mathcal{T} . Για να απλουστεύσουμε τη διαδικασία ο υπολογισμός του αποτελέσματος γίνεται από μαντείο που έχει γνώση του μυστικού κλειδιού των αρχών $SK_{\mathcal{T}}$. Είναι αρκετή ωστόσο μια διαδικασία που να εξάγει σωστό αποτέλεσμα και να μπορεί να προσομοιωθεί από τον αντίπαλο \mathcal{A} (ο οποίος μπορεί να διαφθείρει μειοψηφία από τις αρχές \mathcal{T}).

Ακολουθεί η περιγραφή ενός παιχνιδιού μεταξύ του εκβιαστή \mathcal{A} και του υποψήφιου για εκβιασμό ψηφοφόρου. Ο ψηφοφόρος ρίχνει ένα νόμισμα. Το αποτέλεσμα της ρίψης είναι $b \in \{0, 1\}$.

- Αν $b = 0$ ο ψηφοφόρος καταθέτει ψηφοδέλτιο με την επιλογή του β και παραδίδει στον εκβιαστή \mathcal{A} ψευδές διαπιστευτήριο \overline{sk} . Ο ψηφοφόρος προσπαθεί να αποφύγει τον εκβιασμό.
- Αν $b = 1$ ο ψηφοφόρος υποκύπτει στον εκβιασμό, παραδίδει το πραγματικό διαπιστευτήριο του και απέχει από την εκλογική διαδικασία.

Ο αντίπαλος καλείται να μαντέψει την τιμή b , δηλαδή να διακρίνει ποιες από τις δύο συμπεριφορές υιοθέτησε ο ψηφοφόρος. Επιτρέπουμε ακόμη και τον καθορισμό της ψήφου β από τον αντίπαλο \mathcal{A} .

Αν ο αντίπαλος γνωρίζει τις προθέσεις όλων των υπόλοιπων ψηφοφόρων τότε ο εκβιασμός είναι αναπόφευκτος. Αν για παράδειγμα ο αντίπαλος \mathcal{A} εκβιάζει ψηφοφόρο και γνωρίζει ότι όλοι οι υπόλοιποι ψηφοφόροι θα ψηφίσουν τον υποψήφιο B τότε είναι προφανές πως μία μοναδική ψήφος για τον υποψήφιο A θα ανήκει στον εκβιαζόμενο ψηφοφόρο. Η αντοχή σε εκβιασμούς έχει νόημα όταν υπάρχει αρκετή ασάφεια ως προς τις προθέσεις των ψηφοφόρων.

Σε κάθε πραγματικό εκλογικό σενάριο ο αντίπαλος εκβιαστής δεν μπορεί παρά να έχει μόνο μερική εικόνα για τις προθέσεις των ψηφοφόρων. Συνεπώς η αντοχή σε εκβιασμούς είναι ρεαλιστικό σενάριο. Υποθέτουμε την ύπαρξη n ψηφοφόρων που δεν εκβιάζονται.

Συμβολίζουμε με ϕ την αποχή και με λ την ψήφο με ψευδή διαπιστευτήρια. D_{n,n_C} είναι η κατανομή επί των $(\beta_1, \beta_2, \dots, \beta_n) \in (n_C \cup \phi \cup \lambda)^n$ δηλαδή επί του συνόλου των έγκυρων ψήφων, των αποχών και των ψήφων με ψευδή διαπιστευτήρια. Η κατανομή D_{n,n_C} περιγράφει τον απαραίτητο θόρυβο που αποκρύπτει την πραγματική συμπεριφορά των ψηφοφόρων που εκβιάζονται. Για ένα σύνολο από διαπιστευτήρια $\{sk_i\}$ ορίζουμε ως $vote(\{sk_i\}, PK_{\mathcal{T}}, n_C, D_{n,n_C}, k_2)$ την διαδικασία υποβολής ηλεκτρονικών ψηφοδελτίων που ακολουθεί την κατανομή D_{n,n_C} . Απλούστερα, επιλέγουμε

διάνυσμα $(\beta_1, \beta_2, \dots, \beta_n)$ από την κατανομή D_{n, n_C} και η ψήφος β_i αντιστοιχεί στο διαπιστευτήριο sk_i .

Θα περιγράψουμε το πείραμα *c-resist* που περιγράφει το παιχνίδι μεταξύ εκβιαστή και ψηφοφόρου. Υπενθυμίζουμε ότι οι k_1, k_2 και k_3 είναι παράμετροι ασφάλειας του πρωτοκόλλου, n_V ο συνολικός αριθμός των ψηφοφόρων και n_C το πλήθος των υποψήφιων. Με n_A συμβολίζουμε τον αριθμό των ψηφοφόρων που ελέγχονται από τον αντίπαλο. Με $n_U = n_V - n_A - 1$ συμβολίζουμε το πλήθος των αβέβαιων ψήφων. Δηλαδή τη διαφορά των ψήφων που προέρχονται από τους ψηφοφόρους που ελέγχει ο αντίπαλος συν μία ακόμη που προέρχεται από τον ψηφοφόρο που προσπαθεί να εκβιάσει, από το συνολικό αριθμό των ψήφων. Συνεπώς, n_U ψήφοι συνεισφέρουν στον αναγκαίο θόρυβο.

Θεωρούμε έναν στατικό αντίπαλο που επιλέγει τους διεφθαρμένους ψηφοφόρους στην αρχή του πρωτοκόλλου. Με \leftarrow συμβολίζουμε την ανάθεση και με \Leftarrow συμβολίζουμε την πράξη της προσάρτησης. Μετά από το σύμβολο % ακολουθεί σχόλιο. Το πείραμα που θα περιγράψουμε αφορά την προσπάθεια του αντιπάλου να εκβιάσει έναν μόνο ψηφοφόρο. Η γενίκευση έπεται εύκολα.

```

ExpES,A,Hc-resist(k1, k2, k3, nV, nA, nC)
V ← A(voter names, control voters);                               %A corrupts voters
{(ski, pki) ← register(SKR, i, k2)}i=1nV;                       % Voters are registered
(j, β) ← A({ski}i∈V, set target voter and vote);                   %A sets coercive target

If |V| ≠ nA or j ∉ {1, 2, ..., nV} – V or β ∉ {1, 2, ..., nC} ∪ φ
Then output "0";                                                  % Outputs of A checked for validity

b ∈U {0, 1};                                                    % A coin is flipped

If b = 0 Then                                                    % Voter evades coercion
s̄k ← fakekey(PKT, skj, pkj);
BB ← vote(skj, PKT, nC, β, k2);

else                                                            % voter submits to coercion
s̄k ← skj;

BB ← vote({ski}i≠j, i∈V, PKT, nC, DnU, nC, k2);                % ballots posted for honest players

BB ← A(s̄k, BB, cast ballots);                                    %A posts to BB

(X, P) ← tally(SKT, BB, nC, {pki}i=1nV, k3);                    % election results are tallied

b' ← A(X, P, guess b);                                          %A guesses coin flip

If b' = b then                                                  % experimental output determined
output"1"
else
output"0";

```

Ο αντίπαλος \mathcal{A} όταν $b = 1$ επιβάλλει πλήρως την θέληση του στον εκβιαζόμενο ψηφοφόρο. Είναι ίσως ρεαλιστικότερο να συγκρίνουμε τον \mathcal{A} με αντίπαλο \mathcal{A}' . Οι δυνατότητες του \mathcal{A}' να επηρεάζει τους ψηφοφόρους περιορίζονται από το ιδεατό εκλογικό πρωτόκολλο $c - resist - ideal$ στο οποίο δρα. Απλούστερα ο \mathcal{A}' χαρακτηρίζει την ασφάλεια που επιθυμούμε να έχει το εκλογικό πρωτόκολλο \mathcal{ES} .

Βασικός σκοπός του ιδεατού πειράματος $c - resist - ideal$ είναι ο \mathcal{A}' να μην αποκτά καμία γνώση από τα διαπιστευτήρια που έχει αποκτήσει από τους διεφθαρμένους ψηφοφόρους καθώς και από το διαπιστευτήριο του εκβιαζόμενου ψηφοφόρου. Συγκεκριμένα ο \mathcal{A} δεν μπορεί να χρησιμοποιήσει τα διαπιστευτήρια αυτά σε κάποιου είδους επίθεση. Δεν μπορεί να ψηφίσει ο ίδιος μπορεί όμως να καθορίσει τις επιλογές των ψηφοφόρων που είναι στον έλεγχο του. Η γνώση των διαπιστευτηρίων δεν του αποκαλύπτει τις επιλογές των ψηφοφόρων. Η μόνη πληροφορία που αποκτά είναι το τελικό εκλογικό αποτέλεσμα \mathcal{X} .

Στο ιδεατό αυτό πείραμα είναι μάλλον αναμενόμενο στον \mathcal{A} να δίνεται πάντα το \overline{sk} ως διαπιστευτήριο εφόσον δεν είναι σε θέση να εξακριβώσει αν ο εκβιασμός ήταν επιτυχής.

Ορίζουμε μία καινούρια συνάρτηση *ideal – tally*. Η καινούρια συνάρτηση καταμετρά τις ψήφους που υπάρχουν στον \mathcal{BB} με συγκεκριμένο τρόπο. Δηλαδή, καταμετρά με προφανή τρόπο όλες τις ψήφους των τίμιων ψηφοφόρων (πριν ξεκινήσει δηλαδή η κατάθεση ψήφων διεφθαρμένων ψηφοφόρων). Όσοι ψήφοι προέρχονται από τον αντίπαλο \mathcal{A}' καταμετρώνται διαφορετικά. Συγκεκριμένα για κάθε ψήφο η συνάρτηση *ideal – tally* εξακριβώνει ποιο είναι το μυστικό διαπιστευτήριο sk_i . Αν $i \notin V$ αν δηλαδή το διαπιστευτήριο δεν ανήκει σε κάποιον από τους ψηφοφόρους που ελέγχει ο αντίπαλος δεν προσμετράται. Αν υπάρχουν ψήφοι που αντιστοιχούν στο ίδιο διαπιστευτήριο δεν προσμετρώνται.

Ανάλογα με το αποτέλεσμα b της ρίψης του νομίσματος η συνάρτηση πράττει τα εξής. Αν $b = 0$ η συνάρτηση δεν προσμετρά τις ψήφους που έχουν κατατεθεί από τον αντίπαλο με χρήση του ψευδούς κλειδιού \overline{sk} . Αν $b = 1$ η συνάρτηση προσμετρά την ψήφο που έχει κατατεθεί με το \overline{sk} .

Στα περισσότερα εκλογικά πρωτόκολλα κάθε ψήφος συνοδεύεται από κάποιο διαπιστευτήριο. Να σημειωθεί ότι αν και ο \mathcal{A}' γνωρίζει τα διαπιστευτήρια όλων των διεφθαρμένων ψηφοφόρων στο ιδεατό αυτό πείραμα η γνώση αυτή δεν του προσφέρει καμία χρήσιμη πληροφορία. Η συνάρτηση *ideal – tally* προστατεύει το πρωτόκολλο από κακοπροαίρετη χρήση των διαπιστευτηρίων. Ο \mathcal{A}' επίσης δεν αποκτά καμία πληροφορία για τις ψήφους εφόσον δεν βλέπει τον \mathcal{BB} .

$Exp_{\mathcal{ES}, \mathcal{A}, \mathcal{H}}^{c-resist-ideal}(k_1, k_2, k_3, n_V, n_A, n_C)$

$\mathcal{V} \leftarrow \mathcal{A}'(\text{voter names, control voters});$ % \mathcal{A}' corrupts voters

$\{(sk_i, pk_i) \leftarrow register(SK_{\mathcal{R}}, i, k_2)\}_{i=1}^{n_V};$ % Voters are registered

$(j, \beta) \leftarrow \mathcal{A}'(\text{ set target voter and vote});$ % \mathcal{A}' sets coercive target

If $|\mathcal{V}| \neq n_A$ or $j \notin \{1, 2, \dots, n_V\} - \mathcal{V}$ or $\beta \notin \{1, 2, \dots, n_C\} \cup \phi$
Then output "0"; % Outputs of \mathcal{A}' checked for validity

$b \in_U \{0, 1\};$ % A coin is flipped

If $b = 0$ Then % Voter evades coercion

$\mathcal{BB} \leftarrow vote(sk_j, PK_{\mathcal{T}}, n_C, \beta, k_2);$

$\overline{sk}_j \leftarrow sk_j;$

$\mathcal{BB} \leftarrow vote(\{sk_i\}_{i \neq j, i \notin V}, PK_{\mathcal{T}}, n_C, D_{n_U, n_C}, k_2);$ % ballots posted for honest players

$\mathcal{BB} \leftarrow \mathcal{A}'(\overline{sk}, \{sk_i\}_{i \in V}, \text{ cast ballots})$ % \mathcal{A}' specifies vote choices

$(X, P) \leftarrow ideal - tally(SK_{\mathcal{T}}, \mathcal{BB}, n_C, \{pk_i\}_{i=1}^{n_V}, k_3);$ % election results are tallied

$b' \leftarrow \mathcal{A}'(X, \text{ guess } b);$ % \mathcal{A}' guesses coin flip

If $b' = b$ then % experimental output determined

output "1"

else

output "0";

4.3 Περιγραφή Πρωτόκολλου.

4.3.1 Δομικοί Λίθοι.

Παρουσιάζουμε συνοπτικά τους δομικούς λίθους που απαρτίζουν το πρωτόκολλο.

Threshold cryptosystem with re-encryption:

Αρχικός δομικός λίθος είναι κάποιο Threshold κρυπτοσύστημα δημοσίου κλειδιού που να επιτρέπει επανακρυπτογράφηση κρυπτοκειμένων με γνώση μόνο δημοσίου κλειδιού. Στο παρόν πρωτόκολλο η γνώση του ιδιωτικού κλειδιού μοιράζεται μεταξύ των αρχών \mathcal{T} . Σκοπός μας είναι από τα κρυπτοκείμενα E να μην διαρρέει καμία πληροφορία. Η αποκρυπτογράφηση να είναι δυνατή μόνο από πλειοψηφία συμμετεχόντων στις αρχές \mathcal{T} . Η αποκρυπτογράφηση μοντελοποιείται για ευκολία από ένα μαντείο $D\tilde{E}C$. Κάθε αποκρυπτογράφηση από το μαντείο $D\tilde{E}C$ είναι δημόσια επαληθεύσιμη.

Θα χρησιμοποιήσουμε μία παραλλαγή του El Gamal. Έστω \mathcal{G} ομάδα με τάξη q . Το δημόσιο κλειδί είναι (g_1, g_2, h) όπου $g_1, g_2, h \in \mathcal{G}$. Ιδιωτικό κλειδί $x \in \mathbb{Z}_q$ τέτοιο ώστε $h = g_1^x$. Για την κρυπτογράφηση μηνύματος m στέλνει την τριάδα $(h^r m, g_1^r, g_2^r)$ για τυχαίο r . Συμβολίζουμε με ϵ_U την ομοιόμορφη επιλογή από σύνολο. Με $E_h[m]$ συμβολίζεται η κρυπτογράφηση του κειμένου m με χρήση του δημοσίου κλειδιού h . Για την αποκρυπτογράφηση χρησιμοποιείται μόνο ο όρος g_1^r

$$\frac{h^r m}{(g_1^r)^x} = \frac{(g_1^x)^r m}{(g_1^r)^x} = m.$$

Σημαντική ιδιότητα του κρυπτοσυστήματος El Gamal είναι ότι το ιδιωτικό κλειδί x μπορεί εύκολα να μοιραστεί μεταξύ των αρχών \mathcal{T} ώστε η αποκρυπτογράφηση να είναι δυνατή μόνο με τη συνεργασία πλειοψηφίας. Η διαδικασία μπορεί να προσομοιωθεί από ένα μαντείο $D\tilde{E}C$.

Plaintext Equivalence Test (PET):

Ως είσοδο αυτός ο δομικός λίθος δέχεται δύο κρυπτοκείμενα. Στην έξοδο το αποτέλεσμα μπορεί να είναι 1 αν τα δύο κρυπτοκείμενα προέρχονται από το ίδιο αρχικό κείμενο ή μηδέν αν τα αρχικά κείμενα είναι διαφορετικά. Σημαντικό είναι ότι ο δομικός λίθος δεν αποκαλύπτει καμία πληροφορία για τα αρχικά κείμενα ενώ η έξοδος είναι δημόσια επαληθεύσιμη. Ξανά η διαδικασία προσομοιώνεται από κάποιο μαντείο $P\tilde{E}T$.

Proofs of knowledge:

Γίνεται επίσης χρήση σε διάφορα σημεία μη διαδραστικών αποδείξεων μηδενικής γνώσης $NIZK$.

4.3.2 Αρχές.

Τα ζευγάρια κλειδιών $(SK_{\mathcal{R}}, PK_{\mathcal{R}})$ και $(SK_{\mathcal{T}}, PK_{\mathcal{T}})$ δημιουργούνται και τα $PK_{\mathcal{R}}, PK_{\mathcal{T}}$ δημοσιεύονται μαζί με τις λοιπές παραμέτρους του πρωτοκόλλου.

4.3.3 Εγγραφή Ψηφοφόρων.

Ο ψηφοφόρος V_i αποδεικνύει την ταυτότητα του στις αρχές \mathcal{R} και λαμβάνει τυχαία συμβολοσειρά $\sigma_i \in_U \mathcal{G}$ ως διαπιστευτήριο για την συμμετοχή στην εκλογική διαδικασία. Τα διαπιστευτήρια αυτά είναι δυνατόν να δημιουργηθούν με κατανομημένο τρόπο ώστε κάθε ενεργός server των \mathcal{R} να στέλνει στον ψηφοφόρο V_i τμήμα του διαπιστευτηρίου του. Οι αρχές \mathcal{R} προσθέτουν το $S_i = E_{PK_T}[\sigma_i]$ στην λίστα των ψηφοφόρων L . Η λίστα L είναι δημόσια στον \mathcal{BB} και υπογεγραμμένη ψηφιακά από τις αρχές \mathcal{R} .

Υποθέτουμε ότι η πλειοψηφία των \mathcal{R} είναι ειλικρινείς ώστε κάθε ψηφοφόρος να λαμβάνει τελικά το προσωπικό του διαπιστευτήριο. Είναι επίσης δυνατό οι αρχές \mathcal{R} να στείλουν απόδειξη στον ψηφοφόρο V_i ότι το S_i είναι όντως η κρυπτογράφηση του σ_i .

4.3.4 Λίστα υποψήφιων.

Οι αρχές \mathcal{R} ή κάποια άλλη αρμόδια αρχή εκδίδει και δημοσιεύει λίστα με τους υποψήφιους της εκλογικής διαδικασίας \mathcal{C} (candidate slate) η οποία περιέχει n_C δείκτες έναν για κάθε υποψήφιο. Η αρχή επίσης δημοσιεύει ένα μοναδικό δείκτη ϵ που αφορά την συγκεκριμένη εκλογική διαδικασία.

4.3.5 Κατάθεση ηλεκτρονικών ψηφοδελτίων.

Κάθε ψηφοφόρος V_i καταθέτει ψηφοδέλτιο με την επιλογή του υποψηφίου c_j το οποίο απαρτίζεται από δύο κομμάτια

$$E_1^{(i)} = (g_1^{a_1}, g_2^{a_1}, c_j h^{a_1}) = (a_1, a'_1, \beta_1) \text{ με } a_1 \in_U \mathbb{Z}_q$$

$$E_2^{(i)} = (g_1^{a_2}, g_2^{a_2}, \sigma_i h^{a_2}) = (a_2, a'_2, \beta_2) \text{ με } a_2 \in_U \mathbb{Z}_q$$

Το πρώτο κομμάτι περιέχει την επιλογή του ψηφοφόρου και το δεύτερο το διαπιστευτήριο του.

Ο ψηφοφόρος περιλαμβάνει επίσης στο ηλεκτρονικό ψηφοδέλτιο :

- *NIZK* ότι γνωρίζει τα σ_i, c_j .
- *NIZK* ότι το $c_j \in \mathcal{C}$.
- *NIZK* ότι τα a_i, a'_i έχουν το ίδιο διακριτό λογάριθμο ως προς τις βάσεις g_1, g_2

Οι αποδείξεις αυτές συνθέτουν το σύνολο Pf . Ο ψηφοφόρος δημοσιεύει στον \mathcal{BB} το $B_i = (E_1, E_2, Pf)$ μέσω ανώνυμου καναλιού.

4.3.6 Καταμέτρηση

Για την καταμέτρηση των ψήφων οι αρχές \mathcal{T} εκτελούν τα ακόλουθα:

1. **Έλεγχος αποδείξεων.** Οι αρχές \mathcal{T} ελέγχουν την ορθότητα των αποδείξεων στον \mathcal{BB} . Κάθε ψηφοδέλτιο με μη έγκυρες αποδείξεις απορρίπτεται. Για τα υπόλοιπα ψηφοδέλτια έστω A_1 η λίστα που περιέχει τα κρυπτοκείμενα με τις επιλογές και B_1 η λίστα που περιέχει τα κρυπτογραφημένα διαπιστευτήρια.
2. **Απαλοιφή πολλαπλών ψηφοδελτίων.** Οι αρχές \mathcal{T} εκτελούν έλεγχο PET σε όλα τα κρυπτοκείμενα της λίστας B_1 διαγράφοντας τα πολλαπλά ψηφοδέλτια βάσει συγκεκριμένης πολιτικής (π.χ. διατηρείται το τελευταίο σε χρονολογική σειρά). Όταν ένα στοιχείο διαγράφεται από τη λίστα B_1 το ίδιο συμβαίνει και με το αντίστοιχο στοιχείο της λίστας A_1 . Έστω B'_1, A'_1 οι λίστες που δημιουργούνται μετά τις διαγραφές.
3. **Mixing.** Οι αρχές περνούν τις λίστες A'_1, B'_1 μέσα από το `mixnet`. Έστω A_2, B_2 οι δύο νέες λίστες.
4. **Έλεγχος Διαπιστευτηρίων.** Οι αρχές \mathcal{T} εφαρμόζουν ξανά το `mixnet` στη λίστα των διαπιστευτηρίων L . Έπειτα με PET ελέγχουν αν κάθε κρυπτοκείμενο στη λίστα B_2 υπάρχει στην λίστα L . Παράλληλα διατηρεί μία λίστα A_3 για τα κρυπτοκείμενα της A_2 που τα αντίστοιχα στοιχεία της λίστας B_2 έχουν ταυτιστεί με κάποιο στοιχείο της λίστας L . Στο τέλος της συγκεκριμένης διαδικασίας ψηφοδέλτια με μη έγκυρα διαπιστευτήρια έχουν διαγραφεί.
5. **Καταμέτρηση.** Οι αρχές \mathcal{T} αποκρυπτογραφούν τα στοιχεία της λίστας A_3 .

4.3.7 Αντοχή σε εκβιασμό.

Η συνάρτηση `fakekey` παρέχει στον ψηφοφόρο V_i ψευδές διαπιστευτήριο $\bar{\sigma}_i$. Αν ο ψηφοφόρος εκβιαστεί πολλαπλές φορές τότε παρέχει σε κάθε εκβιαστή το ίδιο ψευδές διαπιστευτήριο. Μπορεί επίσης να παρέχει στον εκβιαστή και αντίγραφο της διαδικασίας.

4.4 Definitions of Correctness and Verifiability

4.4.1 Correctness

Στην ιδιότητα `Correctness` περιλαμβάνονται δύο σκέλη. Αφενός ο αντίπαλος \mathcal{A} δεν γνωρίζει τις ψήφους εκ των προτέρων και δεν μπορεί να αλλάξει ή να ακυρώσει ψήφους ειλικρινών ψηφοφόρων. Αφετέρου δεν μπορεί να επιτύχει διπλή καταμέτρηση ψήφου που να προέρχεται από ένα διαπιστευτήριο.

Στο πείραμα μας επιτρέπουμε στον αντίπαλο να έχει δυνατότητες που υπό φυσιολογικές συνθήκες δεν θα είχε. Μπορεί να επιλέξει το σύνολο των ψηφοφόρων που θα διαφθείρει, να επιλέξει το μέγεθος n_c του `Candidate Slate`, όπως και να καθορίσει τις επιλογές των ψηφοφόρων που δεν ελέγχει. Οι τελευταίοι για τους σκοπούς του πειράματος θα ψηφίσουν όπως θα υποδειξει ο αντίπαλος. Αν παρόλα αυτά ο αντίπαλος δεν μπορεί να προκαλέσει λανθασμένη καταμέτρηση του τελικού αποτελέσματος τότε το πρωτόκολλο έχει όντως την ιδιότητα `Correctness` ακόμη και στο πραγματικό ρεαλιστικό σενάριο στο οποίο ο αντίπαλος έχει λιγότερες δυνατότητες.

Σκοπός του αντίπαλου είναι να καταμετρηθούν στο τελικό αποτέλεσμα περισσότεροι από \mathcal{V} ψήφοι των ψηφοφόρων που ελέγχει ή να αλλάξει την ψήφο ενός τουλάχιστον έντιμου ψηφοφόρου. Πρέπει στο τελικό στάδιο δηλαδή το πρωτόκολλο να επιβεβαιώσει ότι η καταμέτρηση των ψήφων είναι σωστή ενώ είτε μία ψήφος δεν καταμετρήθηκε είτε μία ψήφος προστέθηκε.

$Exp_{\mathcal{E}\mathcal{S},\mathcal{A}}^{corr}(k_1, k_2, k_3, n_{\mathcal{V}}, n_{\mathcal{C}})$

$\{(sk_i, pk_i) \leftarrow register(SK_{\mathcal{R}}, i, k_2)\}_{i=1}^{n_{\mathcal{V}}};$

% Voters are registered

$\mathcal{V} \leftarrow \mathcal{A}(\{pk_i\}_{i=1}^{n_{\mathcal{V}}}, \text{choose controlled voter set});$

% \mathcal{A} corrupts voters

$\{\beta_i\}_{i \notin \mathcal{V}} \leftarrow \mathcal{A}(\text{choose votes for uncontrolled voters});$

% \mathcal{A} chooses votes for honest voters

$\mathcal{B}\mathcal{B} \leftarrow \{vote(sk_i, PK_{\mathcal{T}}, n_{\mathcal{C}}, \beta_i, k_2)\}_{i \notin \mathcal{V}};$

% honest voters cast ballots

$(X, P) \leftarrow tally(SK_{\mathcal{T}}, \mathcal{B}\mathcal{B}, n_{\mathcal{C}}, \{pk_i\}_{i=1}^{n_{\mathcal{V}}}, k_3);$

% honest ballots are tallied

$\mathcal{B}\mathcal{B} \leftarrow \mathcal{A}(\text{cast ballots}, \mathcal{B}\mathcal{B})$

% \mathcal{A} posts ballots to $\mathcal{B}\mathcal{B}$

$(X', P') \leftarrow tally(SK_{\mathcal{T}}, \mathcal{B}\mathcal{B}, n_{\mathcal{C}}, \{pk_i\}_{i=1}^{n_{\mathcal{V}}}, k_3);$

% all ballots are tallied

If $verify(PK_{\mathcal{T}}, \mathcal{B}\mathcal{B}, n_{\mathcal{C}}, X', P') = 1$ and
 $(\{\beta_i\} \not\subseteq \langle X' \rangle \text{ or } |\langle X' \rangle| - |\langle X \rangle| > |\mathcal{V}|)$ then

% does function verify accept?

% did \mathcal{A} successfully tamper?

output"1"

else

output"0";

Το εκλογικό πρωτόκολλο $\mathcal{E}\mathcal{S}$ κατέχει την ιδιότητα *Correctness* αν για κάθε πολυωνυμικό αντίπαλο \mathcal{A} η $Succ_{\mathcal{E}\mathcal{S},\mathcal{A}}^{corr}(k_1, k_2, k_3, n_{\mathcal{V}})$ είναι αμελητέα.

4.4.2 Verifiability

Συνοπτικά, ένα εκλογικό πρωτόκολλο έχει την ιδιότητα *Correctness* αν η συνάρτηση *tally* επιστρέφει την αληθινή καταμέτρηση των ψήφων. Ο αντίπαλος \mathcal{A} έχει την ικανότητα να διαφθείρει μέρος των αρχών \mathcal{T} συνεπώς δεν μπορούμε να είμαστε βέβαιοι ότι ο υπολογισμός είναι ακριβής. Το πρωτόκολλο έχει την ιδιότητα *Verifiability* αν επιτρέπει σε κάθε ενδιαφερόμενο να εξετάσει αν το εκλογικό αποτέλεσμα X έχει υπολογιστεί σωστά. Αν δηλαδή κάθε απόκλιση των αρχών \mathcal{T} από την πιστή εκτέλεση του πρωτοκόλλου μπορεί να εντοπιστεί.

Ο αντίπαλος \mathcal{A} ελέγχει το σύνολο των ψηφοφόρων και των αρχών \mathcal{T} . Επιδιώκει την κατασκευή ενός συνόλου ψήφων που θα δημοσιευτούν στον $\mathcal{B}\mathcal{B}$ και του αποτελέσματος X και απόδειξης P ώστε η συνάρτηση *verify* να αποδεχθεί την P ενώ το αποτέλεσμα X είναι ψευδές. Αυτό μπορεί να συμβεί για παράδειγμα αν όλες οι ψήφοι ενός ψηφοφόρου απορριφθούν ή αν προσμετρηθούν πολλαπλοί ψήφοι ενός ψηφοφόρου.

$Exp_{\mathcal{ES},\mathcal{A}}^{ver}(k_1, k_2, k_3, n_V, n_C)$

$\{(sk_i, pk_i) \leftarrow register(SK_{\mathcal{R}}, i, k_2)\}_{i=1}^{n_V};$

% Voters are registered

$(\mathcal{BB}, X, P) \leftarrow \mathcal{A}(SK_{\mathcal{T}}, \{(sk_i, pk_i)\}_{i=1}^{n_V}, \text{forge election});$

% \mathcal{A} concocts full election

$(X', P') \leftarrow tally(SK_{\mathcal{T}}, \mathcal{BB}, n_C, \{pk_i\}_{i=1}^{n_V}, k_3);$

% tally is taken on \mathcal{BB}

if $X \neq X'$

% does \mathcal{A} tally differ from correct \mathcal{BB} tally?

and $verify(PK_{\mathcal{T}}, \mathcal{BB}, n_C, X, P) = 1$ then

% does function accept?

output "1"

else

output "0";

Το εκλογικό πρωτόκολλο \mathcal{ES} κατέχει την ιδιότητα *Verifiability* αν για κάθε πολυωνυμικό αντίπαλο \mathcal{A} και κάθε θετικό ακέραιο n_V η $Succ_{\mathcal{ES},\mathcal{A}}^{ver}(k_1, k_2, k_3, n_V)$ είναι αμελητέα.

4.4.3 Strong Verifiability

Οι ορισμοί που προηγήθηκαν για *Correctness* και *Verifiability* περιέχουν τις ελάχιστες απαιτήσεις ώστε το εκλογικό πρωτόκολλο να είναι δίκαιο. Οι ορισμοί είναι επαρκείς για την πλειονότητα των εκλογικών σεναρίων δεν περιλαμβάνουν όμως όλα τα πιθανά σενάρια συμπεριφοράς του αντίπαλου.

Για παράδειγμα ψηφοφόρος που ελέγχεται από τον αντίπαλο \mathcal{A} καταθέτει το ψηφοδέλτιο με την επιλογή του για τον υποψήφιο β πλην όμως η ψήφος του προσμετράτε στον υποψήφιο β' . Εφόσον ο \mathcal{A} επιλέγει την ψήφο του διεφθαρμένου ψηφοφόρου αυτό σημαίνει ότι η ψήφος άλλαξε κατά την διάρκεια εκτέλεσης του πρωτοκόλλου. Επιθυμητό σενάριο αν ο αντίπαλος επιθυμεί την νίκη υποψηφίου με την ελάχιστη δυνατή πλειοψηφία. Στην περίπτωση αυτή αν ο \mathcal{A} ελέγχει πλειοψηφία αρχών \mathcal{T} είναι δυνατό να προσμετρά και να αλλοιώνει ψήφους ώστε να επιτύχει το επιθυμητό αποτέλεσμα.

Μπορούμε πλέον να επεκτείνουμε τον ορισμό ώστε να συμπεριλάβουμε και τέτοιου είδους επιθέσεις. Ανάλογες τροποποιήσεις μπορούν να γίνουν και για τον ορισμό της ιδιότητας *Correctness*.

Αρκεί συνοπτικά να επεκταθεί η απόδειξη P ώστε να συμπεριλάβει ότι κάθε ψήφος που προσμετράτε αντιστοιχεί σε ένα μοναδικό διαπιστευτήριο με το οποίο έχει κατατεθεί έγκυρο ψηφοδέλτιο.

Έστω $\langle vote(\cdot) \rangle$ το σύνολο των δυνατών αποτελεσμάτων σε συγκεκριμένη είσοδο. Η έξοδος πρέπει να είναι απόλυτα καθορισμένη από την επιλογή β και το διαπιστευτήριο sk .

$$\langle vote(sk_0, PK_{\mathcal{T}}, n_C, \beta_0, k_2) \rangle \cap \langle vote(sk_1, PK_{\mathcal{T}}, n_C, \beta_1, k_2) \rangle = \emptyset$$

αν $\beta_0 \neq \beta_1$ ή $sk_0 \neq sk_1$.

Μεταβάλουμε τον ορισμό του πειράματος της *Verifiability*, $Exp_{\mathcal{ES},\mathcal{A}}^{ver}(k_1, k_2, k_3, n_V)$ ώστε αν μία από τις δύο ακόλουθες συνθήκες 1 και 2 ισχύει στην έξοδο να λαμβάνουμε 1. Διαφορετικά λαμβάνουμε 0.

1. $verify(PK_{\mathcal{T}}, \mathcal{BB}, n_C, X, P) = 1$

2. Για κάθε μονομορφισμό $f : \langle X \rangle \rightarrow Z_{n\gamma}$ μία από τις ακόλουθες συνθήκες ισχύουν:

(α) $\exists B : B \in \mathcal{BB}, B \in \langle \text{vote}(sk_i, PK_{\mathcal{T}}, n_C, \beta, k_2) \rangle, \forall j f(j) \neq i$

(β) $\exists \beta \in X : f(\beta) = i, \forall B \in \mathcal{BB}, B \notin \langle \text{vote}(sk_i, PK_{\mathcal{T}}, n_C, \beta, k_2) \rangle$

Οι συνθήκες 2(α) και 2(β) περιγράφουν την ικανότητα του αντίπαλου είτε να προκαλέσει ακύρωση όλων των ψήφων που προέρχονται από ένα διαπιστευτήριο είτε να επιτρέψει πολλαπλές ψήφους σε μοναδικό διαπιστευτήριο.

4.4.4 Ψευδή Διαπιστευτήρια

Υποθέτουμε πως ο ψηφοφόρος προσπαθεί να αντισταθεί στον εκβιασμό δίνοντας ψευδή διαπιστευτήρια $\bar{\sigma}_i$. Ο αντίπαλος προσπαθεί να διαπιστώσει αν ο ψηφοφόρος έχει υποκύψει στον εκβιασμό. Θυμίζουμε ότι ο αντίπαλος μπορεί να διαφθείρει μόνο μία μειοψηφία από τις αρχές \mathcal{T} συνεπώς δεν μπορεί να αποκρυπτογραφήσει τα κρυπτογραφημένα διαπιστευτήρια.

1. Αν υποθέσουμε ότι υπάρχει κάποιος μηχανισμός που υποχρεώνει τον ψηφοφόρο να διαγράφει τα έγγραφα που αντάλλαξε με τις αρχές \mathcal{R} στο τέλος της μεταξύ τους επικοινωνίας. Επίσης μόνο μια μειοψηφία αρχών \mathcal{R} είναι διεφθαρμένη. Δεδομένου ότι δεν υπάρχουν αντίγραφα της επικοινωνίας ο αντίπαλος δεν είναι σε θέση να επαληθεύσει αν το διαπιστευτήριο $\bar{\sigma}_i$ είναι το πραγματικό.
2. Αν ο αντίπαλος δεν έχει διαφθείρει κανένα μέλος των αρχών \mathcal{R} . Οι αρχές μπορούν να αποδείξουν ότι το τμήμα από το διαπιστευτήριο που έστειλαν στον ψηφοφόρο είναι το πραγματικό συνοδεύοντας την αποστολή αυτή με την αντίστοιχη απόδειξη (designated verifier proofs.)
Ενώ ο ψηφοφόρος πείθεται για την ειλικρινή λειτουργία των αρχών ο αντίπαλος αδυνατεί. Ο ψηφοφόρος μπορεί προσομοιώνοντας την λειτουργία των αρχών να συνοδεύσει το ψευδές διαπιστευτήριο $\bar{\sigma}_i$ που παραδίδει στον εκβιαστή με αντίστοιχες πλαστές αποδείξεις.
3. Αν ο ψηφοφόρος γνωρίζει ποια μειοψηφία αρχών \mathcal{R} έχει διαφθαρεί δεν μπορεί όμως να διαγράψει τα έγγραφα τις μεταξύ τους επικοινωνίας. Τότε ο ψηφοφόρος παραδίδει με συνέπεια τα αντίγραφα της επικοινωνίας που είχε με τους server των αρχών που είναι διεφθαρμένοι και παραποιεί τα υπόλοιπα. Τα τελευταία συνοδεύονται από ψευδείς αποδείξεις που παράγονται από τον ψηφοφόρο. Εφόσον ο αντίπαλος διαφθείρει μόνο μειοψηφία των αρχών υπάρχει κομμάτι του διαπιστευτηρίου σ_i το οποίο και να μπορεί να παραποιηθεί από τον ψηφοφόρο ώστε να δημιουργηθεί ψευδές διαπιστευτήριο $\bar{\sigma}_i$ με $\sigma_i \neq \bar{\sigma}_i$ χωρίς να είναι σε θέση ο αντίπαλος να τα διακρίνει.

4.4.5 Proving Coercion - Freeness.

Στην παράγραφο αυτή θα αποδείξουμε ότι το πρωτόκολλο είναι Coercion Resistant. Δεν θα ασχοληθούμε με τις ιδιότητες Correctness και Verifiability. Για την απόδειξη υποθέτουμε τη χρήση του M-El Gamal επί ομάδας \mathcal{G} τάξης q στην οποία ισχύει η Decision Diffie Hellman υπόθεση. Η απόδειξη βασίζεται στην ισχύ της υπόθεσης Decision Diffie Hellman επί της ομάδας \mathcal{G} .

Συνοπτικά σε ομάδες που η υπόθεση Decision Diffie Hellman ισχύει δεν υπάρχει αλγόριθμος πολυωνυμικού χρόνου που να μπορεί να διακρίνει με μη αμελητέα πιθανότητα τις κατανομές:

D οι τετράδες (y_1, g_1, y_2, g_2) με $g_1, g_2 \in_U \mathcal{G}, y_1 = g_1^x, y_2 = g_2^x$ για $x \in_U \mathbb{Z}_q$

D' οι τετράδες (y_1, g_1, y_2, g_2) με $y_1, g_1, y_2, g_2 \in_U \mathcal{G}$.

Απλουστεύοντας την περιγραφή υποθέτουμε την ύπαρξη διάφορων μαντείων που αντικαθιστούν διαδικασίες του εκλογικού πρωτοκόλλου.

1. Το μαντείο $\tilde{M}N$ αντικαθιστά το `mix net`. Στην είσοδο λαμβάνει μία διατεταγμένη λίστα $E = \{E_1, E_2, \dots, E_d\}$ και το δημόσιο κλειδί $PK_{\mathcal{T}}$ των αρχών \mathcal{T} . Στην έξοδο λαμβάνουμε μία διατεταγμένη λίστα $E' = \{E'_{\pi(1)}, E'_{\pi(2)}, \dots, E'_{\pi(d)}\}$ όπου π κάποια μυστική μετάθεση και το $E'_{\pi(i)}$ συμβολίζει επανακρυπτογράφηση του E_i .
2. Το μαντείο $P\tilde{E}T$ λαμβάνει στην είσοδο ένα ζευγάρι κρυπτοκείμενα (E, E') . Στην έξοδο λαμβάνουμε 1 αν τα κρυπτοκείμενα προέρχονται από το ίδιο αρχικό κείμενο και 0 διαφορετικά.
3. Το μαντείο $D\tilde{E}C$ αποκρυπτογραφεί τα κρυπτοκείμενα.
4. Το μαντείο $O\tilde{W}$ λαμβάνει στην είσοδο μία συμβολοσειρά από $\{0, 1\}^*$ και στην έξοδο λαμβάνουμε μία ακολουθία $\{0, 1\}^{k_4}$ όπου k_4 είναι μία παράμετρος ασφάλειας (που πιθανόν να εξαρτάται από τα k_1, k_2, k_3). Σε ίδιες εισόδους λαμβάνουμε ίδιες εξόδους. Αντικαθιστά `hash function`.

Η είσοδος κάθε μαντείου λαμβάνεται από όλες τις αρχές και είναι δημόσια. Κάθε αρχή μπορεί να θεωρηθεί ότι έχει μία ταινία για κάθε μαντείο στην οποία γράφει την αντίστοιχη είσοδο. Κάθε ταινία είναι χωρισμένη σε αντίστοιχα κομμάτια για κάθε βήμα του πρωτοκόλλου το οποίο εκτελείται συγχρόνως. Αν οι εισοδοί των αρχών στην αρχή του πρωτοκόλλου είναι κατά πλειοψηφία μη κενές το μαντείο παράγει την αναμενόμενη έξοδο, διαφορετικά παράγει το σύμβολο \perp . Εξασφαλίζοντας ότι η λειτουργία του μαντείου καθορίζεται από το αδιάφθορο κομμάτι των αρχών.

Η παραγωγή κλειδιών και η εγγραφή των ψηφοφόρων γίνεται από κάποια έμπιστη αρχή.

4.4.6 Proof of Coercion Resistance

Η απόδειξη Coercion Resistance βασίζεται σε ένα παιχνίδι μεταξύ του εκβιαστή και του ψηφοφόρου. Ο εκβιαστής καλείται να μαντέψει ποιες από τις ακόλουθες δύο συμπεριφορές έχει υιοθετήσει ο ψηφοφόρος κατά την εκτέλεση του εκλογικού πρωτοκόλλου \mathcal{ES}

1. Παρέδωσε αληθινά διαπιστευτήρια και απείχε από την εκλογική διαδικασία.
2. Παρέδωσε ψευδή διαπιστευτήρια και συμμετείχε στην εκλογική διαδικασία.

Για να αποδείξουμε ότι όντως το πρωτόκολλο είναι Coercion Resistance αρκεί να αποδείξουμε ότι ο αντίπαλος \mathcal{A} μπορεί να μαντέψει με πιθανότητα αμελητέα μεγαλύτερη από ένα πολυωνυμικό αντίπαλο \mathcal{A}' που επιδρά με ένα ιδεατό εκλογικό σύστημα στο οποίο μπορεί να λάβει γνώση μόνο του τελικό εκλογικό αποτέλεσμα \mathcal{X} καθώς και το πλήθος Γ των ψηφοδελτίων που απορρίφθηκαν λόγω ψευδών διαπιστευτηρίων.

Θα κατασκευάσουμε πολυωνυμικού χρόνου αλγόριθμο ο οποίος με είσοδο σύνολο ψήφων \mathcal{W} ειλικρινών ψηφοφόρων, προσομοιώνει το εκλογικό πρωτόκολλο \mathcal{ES} στο πείραμα `c-resist`. Αν ο \mathcal{A} δεν μπορεί να διακρίνει την προσομοίωση από μία πραγματική εκτέλεση του πρωτοκόλλου και

επίσης δεν μπορεί να επηρεάσει την λειτουργία της προσομοίωση προς όφελος του, τότε μαθαίνει μόνο το τελικό αποτέλεσμα \mathcal{X} και το πλήθος – των ψήφων με ψευδή διαπιστευτήρια. Συνεπώς ο \mathcal{A} μαντεύει με όση επιτυχία μαντεύει και ο αντίπαλος \mathcal{A}' στο πείραμα c -resist-ideal και το εκλογικό πρωτόκολλο \mathcal{ES} είναι Coercion Resistant.

Η ανικανότητα του αντιπάλου να επηρεάσει τη λειτουργία της προσομοίωσης οφείλεται στην λειτουργία των μαντειών. Τα μαντεία για να εκτελέσουν την λειτουργία τους απαιτούν πλειοψηφία αρχών να συμφωνούν στις εισόδους. Με αυτόν τον περιορισμό μπορούμε να δείξουμε ότι η προσομοίωση εκτελείται χωρίς παρεκκλίσεις. Ο \mathcal{A} δεν μπορεί να τη διακρίνει από μία πραγματική εκτέλεση του πρωτοκόλλου c -resist αρκεί η υπόθεση DDH να ικανοποιείται στην ομάδα \mathcal{G} .

Ένας πολυωνυμικός αντίπαλος που επιλέγει μήνυμα m δεν μπορεί να διακρίνει μεταξύ των κατανομών D μηνυμάτων M-El Gamal (A_1, A_2, B) και D' τριάδων της μορφής (a_1, a_2, β) με $\beta \in_U \mathcal{G}$ όταν τα a_1, a_2 ακολουθούν την κατανομή των (A_1, A_2) (με μη αμελητέα πιθανότητα στις παραμέτρους ασφάλειας του πρωτοκόλλου).

Συνέπεια της προηγούμενης παραγράφου είναι ο \mathcal{S} να έχει τη δυνατότητα να αντικαθιστά με τυχαία κρυπτοκείμενα (τριάδες στοιχείων της ομάδας \mathcal{G}) τα κρυπτοκείμενα που θα προέκυπταν από μία εκτέλεση του πρωτοκόλλου c -resist. Ο \mathcal{S} μπορεί να προσομοιώσει τις ψήφους ελικρινών ψηφοφόρων με μια λίστα τυχαίων κρυπτοκειμένων. Επίσης, ο \mathcal{S} μπορεί να προσομοιώσει το μαντείο \tilde{MN} θέτοντας στην έξοδο του τυχαία λίστα κρυπτοκειμένων. Από την υπόθεση DDH ο αντίπαλος \mathcal{A} δεν μπορεί να διακρίνει μεταξύ των κρυπτοκειμένων που παράγονται από τον αλγόριθμο \mathcal{S} και από μια πραγματική εκτέλεση του πειράματος \mathcal{ES} .

Περιγράφουμε τα στάδια της προσομοίωσης του c -resist από τον αλγόριθμο \mathcal{S} . Έστω $W \in D_{n_u, n_c}$ ένα σύνολο που αντικαθιστά τις ψήφους των ελικρινών ψηφοφόρων. Αν $d = 1$ ο προσομοιωτής λαμβάνει μία τετράδα (g_1, g_2, h_1, h_2) που είναι Diffie - Hellman. Αν $d = 0$ η τετράδα δεν είναι Diffie - Hellman. Ο προσομοιωτής καλείται να μαντέψει τι από τα δύο συμβαίνει.

1. **Set Up:** Ο \mathcal{S} επιλέγει ομοιόμορφα τυχαία δύο στοιχεία $x_1, x_2 \in_U \mathbb{Z}_q$ και θέτει $h = g_1^{x_1} g_2^{x_2} \pmod p$. Το δημόσιο κλειδί (g_1, g_2, h) μαζί με Candidate Slate της μορφής $C = \{c_i\}_{i=1}^{n_c}$ τέτοιο ώστε $c_i = g_1^{r_i}$ για $r_i \in_U \mathbb{Z}_q$ γίνονται γνωστά. (Για τεχνικούς λόγους στην απόδειξη θεωρούμε ότι οι δείκτες των υποψηφίων είναι τυχαίοι και όχι οι $\{1, 2, \dots, n_c\}$.)
2. **Registration:** Ο \mathcal{S} προσομοιώνει τη λειτουργία των αρχών \mathcal{R} και εκδίδει διαπιστευτήρια $\{\sigma_i = g_1^{s_i}\}$ για $s_i \in_U \mathbb{Z}_q$. Η λίστα L_0 που αποτελείται από τα n_V κρυπτοκείμενα των διαπιστευτηρίων δημοσιεύεται.
3. **Adversarial Corruption:** Ο \mathcal{A} επιλέγει υποσύνολο V από n_A ψηφοφόρους υπό τον έλεγχο, ψηφοφόρο j που θα εκδιόσει και την επιθυμητή ψήφο β που θα του υποδείξει. Αν κάποιος από τους ακόλουθους ισχυρισμούς $|V| \neq n_A$ ή $j \notin V$ ή $\beta \notin C \cup \phi$ αληθεύει η προσομοίωση τερματίζεται.
4. **Coin Flip:** Επιλέγουμε τυχαία $b \in_U \{0, 1\}$.
5. **Credential Release:** Ο \mathcal{S} δίνει στον \mathcal{A} την λίστα με τα διαπιστευτήρια $\{\sigma_i\}_{i \in V}$ των ψηφοφόρων που ελέγχει καθώς και διαπιστευτήριο σ του ψηφοφόρου j που εκδιόζεται. Αν $b = 1$ ο \mathcal{S} προσφέρει το πραγματικό διαπιστευτήριο $\sigma = \sigma_j$ διαφορετικά μία τυχαία συμβολοσειρά.
6. **Honest Voter Simulation:** Για κάθε ψήφο στην λίστα W ο προσομοιωτής \mathcal{S} δημοσιεύει ψήφο που αποτελείται από δύο κρυπτοκείμενα $(a_{i,1}, a'_{i,1}, \beta_{i,1}), (a_{i,2}, a'_{i,2}, \beta_{i,2})$. Ο \mathcal{S} εφοδιαζόμενες με NIZK αποδείξεις. Έστω A_0 η λίστα με τις ψήφους αυτές και A^* η λίστα των αρχικών

ψήφων στην W για τις οποίες το διαπιστευτήριο είναι σωστό (διαγράφονται λ ψήφοι). Για κάθε ψήφο στο W ο προσομοιωτής \mathcal{S} επιλέγει $r_i, k_i \in_U \mathbb{Z}_q$ και $(a_{i,1} = h_1^{r_i}, a'_{i,1} = h_2^{r_i}, \beta_{i,1} = h_1^{r_i x_1} h_2^{r_i x_2} c_j), (a_{i,2} = h_1^{k_i}, a'_{i,2} = h_2^{k_i}, \beta_{i,2} = h_1^{k_i x_1} h_2^{k_i x_2} \sigma_i)$

7. **Adversarial ballot posting:** Ο αντίπαλος \mathcal{S} δημοσιεύει λίστα με ψήφους B_0 και τις αντίστοιχες $NIZK$ αποδείξεις.
8. **Decryption of ballots posted by the adversary:** Ο \mathcal{S} ελέγχει τις αποδείξεις $NIZK$ στην λίστα B_0 . Έστω B_1 η λίστα των ψήφων με ορθές αποδείξεις. Για κάθε ψήφο στην B_1 και κάθε διαπιστευτήριο $\{\sigma_i\}_{i \in V} \cup \sigma_j$ ο προσομοιωτής αποκρυπτογραφεί με χρήση του ιδιωτικού κλειδιού.
9. **Tallying simulation:** Ο \mathcal{S} προσομοιώνει την λειτουργία των αρχών \mathcal{T} . Κάθε αποκλίνουσα συμπεριφορά των αρχών μπορεί να αγνοηθεί καθώς ο αντίπαλος ελέγχει την μειοψηφία αυτών.
 - (α) **Proof checking:** Έστω E_0 η λίστα που προκύπτει από την ένωση των A_0, B_0 . Ο \mathcal{S} προσομοιώνει τη λειτουργία των αρχών και απορρίπτει όλες τις ψήφους με ψευδείς αποδείξεις. Έστω E_1 η νέα λίστα που προκύπτει.
 - (β) **Eliminating duplicates:** Ο \mathcal{S} μπορεί να διαγράψει τις διπλές ψήφους χρησιμοποιώντας το ιδιωτικό του κλειδί. Αποκρυπτογραφεί και διαγράφει τις διπλές εμφανίσεις. Έστω E_2 η νέα λίστα που προκύπτει.
 - (γ) **Mixing:** Ο \mathcal{S} προσομοιώνει τη λειτουργία του \tilde{MN} στην λίστα E_2 και στην έξοδο προκύπτει λίστα ίδιου μήκους E_3 με τυχαία κρυπτοκείμενα. Ομοίως προκύπτει με είσοδο την L_0 νέα λίστα L_1 .
 - (δ) **Checking Credentials:** Ο \mathcal{S} προσομοιώνει τον έλεγχο των διαπιστευτηρίων χρησιμοποιώντας το ιδιωτικό κλειδί. Η λίστα E_4 προκύπτει.
 - (ε) **Decryption:** Με χρήση του ιδιωτικού κλειδιού αποκρυπτογραφεί.

Αν ο αντίπαλος μαντέψει b' στην έξοδο τότε και ο προσομοιωτής επιστρέφει στην έξοδο του b' ως απάντηση στο Diffie - Hellman ερώτημα.

Αν στην είσοδο της προσομοίωσης $d = 1$ έχουμε δηλαδή Diffie - Hellman τριάδα τότε η προσομοίωση είναι πρακτικά το πείραμα $Exp_{\mathcal{E}, \mathcal{S}, \mathcal{A}, \mathcal{H}}^{c-resist}$.

Αν υποθέσουμε $g_1 = g, g_2 = g^a, h_1 = g^b, h_2 = g^{ab}$ για κάποιο g κάθε κρυπτοκείμενο θα έχει την ακόλουθη μορφή $(a_{i,1} = h_1^{r_i}, a'_{i,1} = h_2^{r_i}, \beta_{i,1} = h_1^{r_i x_1} h_2^{r_i x_2} m)$.

- $h_1^{r_i} = g^{br_i} = g_1^{br_i}$
- $h_2^{r_i} = g^{abr_i} = g_2^{br_i}$
- $h_1^{r_i x_1} h_2^{r_i x_2} m = g^{br_i x_1} g^{abr_i x_2} m = g_1^{br_i x_1} g_2^{br_i x_2} m = h^{br_i} m.$

Συνεπώς ο αντίπαλος \mathcal{A} θα γνώριζε την μορφή που θα έχουν τα κρυπτοκείμενα. Θα έλεγε κανείς ότι βλέπει τον Bulletin Board.

$$Pr[\mathcal{S} = 1 | d = 1] = Pr[Exp_{\mathcal{E}, \mathcal{S}, \mathcal{A}, \mathcal{H}}^{c-resist}(\mathcal{V}) = 1] = Succ_{\mathcal{E}, \mathcal{S}, \mathcal{A}}^{c-resist}(\mathcal{V})$$

όπου με \mathcal{V} η εικόνα που έχει ο αντίπαλος.

Διαφορετικά αν στην είσοδο της προσομοίωσης η τριάδα δεν είναι Diffie - Hellman (δηλαδή $d = 0$) τότε καμία πληροφορία δεν διαρρέει για της ψήφους των έντιμων ψηφοφόρων. Αν υποθέσουμε ότι $g_1 = g, g_2 = g^a, h_1 = g^b, h_2 = g^c$ για κάποιο τυχαίο $c \in_U \mathbb{Z}_q$ το κρυπτοκείμενο $(a_{i,1} = h_1^{r_i}, a'_{i,1} = h_2^{r_i}, \beta_{i,1} = h_1^{r_i x_1} h_2^{r_i x_2} m)$ αποκρύπτει πλήρως το μήνυμα m . Πράγματι,

- $h_1^{r_i} = g^{br_i} = g_1^{br_i}$
- $h_2^{r_i} = g^{cr_i} = g_2^{c'r_i}$
- $h_1^{r_i x_1} h_2^{r_i x_2} m = g^{br_i x_1} g^{cr_i x_2} m = g_1^{br_i x_1} g_2^{c'r_i x_2} m = g_1^{br_i x_1} g_2^{br_i x_2} g_2^{c''r_i x_2} m.$

Η πιθανότητα η προσομοίωση να επιστρέψει ως αποτέλεσμα 1 είναι ίση με την πιθανότητα ο αντίπαλος να επιστρέψει ως αποτέλεσμα στο πείραμα $Exp^{c-resist-ideal}$. Δηλαδή,

$$Pr[\mathcal{S} = 1 | d = 0] = Pr[Exp_{\mathcal{E}, \mathcal{S}, \mathcal{A}, \mathcal{H}}^{c-resist-ideal}(\mathcal{V}) = 1] = Succ_{\mathcal{E}, \mathcal{S}, \mathcal{A}}^{c-resist-ideal}(\mathcal{V})$$

και προκύπτει ότι

$$Adv_{\mathcal{S}}^{ddh} = Pr[\mathcal{S} = 1 | d = 1] - Pr[\mathcal{S} = 1 | d = 0] = Adv_{\mathcal{E}, \mathcal{S}, \mathcal{A}}^{c-resist}$$

Από το decisional Diffie - Hellman η ποσότητα αυτή είναι αμελητέα.

Chapter 5

Coercion Resistant Internet Voting with Everlasting Privacy

Στο παρόν κεφάλαιο παρουσιάζεται το άρθρο των Philipp Locher, Rolf Haenni και Reto E. Koenig [8]

5.1 Δομικοί Λίθοι.

Θα παρουσιάσουμε συνοπτικά τους επιμέρους δομικούς λίθους που απαρτίζουν το πρωτόκολλο.

5.1.1 Ομάδες και γεννήτορες.

Έστω \mathcal{G}_p κυκλική ομάδα με τάξη πρώτο αριθμό p και γεννήτορες g_0, g_1 για την οποία το πρόβλημα διακριτού λογαρίθμου είναι δυσεπίλυτο. Έστω επίσης, $\mathbb{G}_q \subset \mathbb{Z}_p^*$ με q πρώτο (αρκετά μεγάλο) και γεννήτορες h, h_0, h_1, \dots

Δεν είναι γνωστός ο $\log_{g_0} g_1$. Ομοίως για κάθε πιθανό συνδυασμό γεννητόρων.

5.1.2 Ομομορφικές δεσμεύσεις.

Το πρωτόκολλο χρησιμοποιεί δεσμεύσεις Pedersen επί των ομάδων \mathcal{G}_p και \mathbb{G}_q .

- Για δέσμευση $u \in \mathbb{Z}_p$ με τυχαιότητα $r \in \mathbb{Z}_p$ συμβολίζουμε $com_p(u, r) = g_0^r g_1^u$.
- Για δέσμευση $v \in \mathbb{Z}_q$ με τυχαιότητα $s \in \mathbb{Z}_q$ συμβολίζουμε $com_q(v, s) = h_0^s h_1^v$.
- Για δέσμευση σε n τιμές $v_1, v_2, \dots, v_n \in \mathbb{G}_q$ με τυχαιότητα $s \in \mathbb{Z}_q$ συμβολίζουμε $com_q(v_1, \dots, v_n, s) = h_0^s h_1^{v_1} \dots h_n^{v_n}$.

5.1.3 Πρωτόκολλο Κρυπτογράφησης.

Ως συνήθως, ElGamal επί της ομάδας \mathbb{G}_q με x ιδιωτικό κλειδί που μοιράζεται από κοινού στις αρχές (ώστε αποκρυπτογράφηση να είναι δυνατή μόνο με συνεργασία πλειοψηφίας) και $y = h^x \in \mathbb{G}_q$

το δημόσιο κλειδί.

Συμβολίζουμε με $E = enc_y(m, r) = (h^r, my^r) \in \mathbb{G}_q \times \mathbb{G}_q$ την κρυπτογράφηση μηνύματος $m \in \mathbb{G}_q$ με τυχαιότητα $r \in \mathbb{Z}_q$. Συμβολίζουμε με $m = dec_x(E) = ba^{-x}$ την αποκρυπτογράφηση του $E = (a, b)$ με καταναμημένο τρόπο από πλειοψηφία αρχών.

Συμβολίζουμε με $\mathbf{M} = dec_x(\mathbf{E}) = (m_1, \dots, m_n)$ την αποκρυπτογράφηση λίστας μηνυμάτων $\mathbf{E} = (E_1, \dots, E_n)$. Για την επανακρυπτογράφηση κρυπτοκειμένου E με νέα τυχαιότητα $r' \in \mathbb{Z}_q$ ως συνήθως $\mathbf{E}' = reEnc_y(\mathbf{E}, r') = E \cdot enc_y(1, r')$ πολλαπλασιάζοντας με την κρυπτογράφηση της μονάδας. Συμβολίζουμε με $\mathbf{E}' = reEnc_y(\mathbf{E}, \mathbf{r}') = (E'_1, E'_2, \dots, E'_n)$ την κρυπτογράφηση της λίστας $\mathbf{E} = (E_1, \dots, E_n)$ με νέες τυχαιότητες $\mathbf{r}' = (r'_1, r'_2, \dots, r'_n)$.

5.1.4 Αποδείξεις Μηδενικής Γνώσης.

Το πρωτόκολλο στηρίζεται σε διάφορες μη διαδραστικές αποδείξεις μηδενικής γνώσης.

Παρατήρηση. Σημαντικό εργαλείο στην κατασκευή των αποδείξεων είναι η αντίστροφη εικόνα μέσω one way ομομορφισμού ομάδων $\phi : X \rightarrow Y$ ([9]). Να σημειώσουμε δε, ότι δεν είναι αναγκαίο ο ομομορφισμός να είναι one way για να υπάρχει απόδειξη μηδενικής γνώσης για την αντίστροφη εικόνα μιας δημόσιας πληροφορίας.

$NIZK[(a) : b = \phi(a)]$ όπου το $a = \phi^{-1}(b)$ είναι η αντίστροφη εικόνα μιας δημόσιας πληροφορίας $b \in Y$.

Παραδείγματα τέτοιων αποδείξεων μηδενικής γνώσης είναι

- $NIZKP[(u, r) : C = com_p(u, r)]$ για την απόδειξη γνώσης της opening value της δέσμευσης Pedersen.
- $NIZKP[(m, r) : E = enc_y(m, r)]$ για την απόδειξη γνώσης του αρχικού κειμένου και της τυχαιότητας στο κρυπτοσύστημα ElGamal.
- $NIZKP[(x) : \mathbf{M} = dec_x(\mathbf{E}) \wedge y = h^x]$ για την απόδειξη γνώσης του ιδιωτικού κλειδιού που χρησιμοποιείται στην αποκρυπτογράφηση λίστας μηνυμάτων.

Set Membership Proof.

Έστω $U = \{u_1, u_2, \dots, u_N\}$ πεπερασμένο σύνολο με τιμές $u_i \in \mathbb{Z}_p$ και $C = com_p(u, r)$ δέσμευση στο $u \in U$. Τα U, C είναι δημόσια. Με την ακόλουθη

$$NIZKP[(u, r) : C = com_p(u, r) \wedge u \in U],$$

αποδεικνύει γνώση του $u \in U$ καθώς και του $r \in \mathbb{Z}_p$. Μπορεί να κατασκευαστεί απόδειξη μηδενικής γνώσης για το Set Membership με χρήση του πολυωνύμου

$$P(X) = \prod_{i=1}^N (X - u_i) \text{ και } P(u) = 0.$$

Συμβολίζεται με

$$NIZKP[(u, r) : C = \text{com}_p(u, r) \wedge P(u) = 0].$$

Proof of Known Representation.

Σε κυκλική ομάδα \mathbb{G}_q με γεννήτορες h_1, h_2, \dots, h_n μία n -ιάδα $(v_1, v_2, \dots, v_n) \in \mathbb{Z}_q^n$ καλείται *DL-αναπαράσταση* (ή απλά αναπαράσταση) του $u \in \mathbb{G}_q$ αν $u = h_1^{v_1} \cdot h_2^{v_2} \cdot \dots \cdot h_n^{v_n}$. Για κάθε τέτοια $u \in \mathbb{G}_q \subset \mathbb{Z}_p$ έστω $C = \text{com}_p(u, r)$ και $D = \text{com}_q(v_1, v_2, \dots, v_n, s)$ δημόσιες δεσμεύσεις. Απόδειξη γνώσης αναπαράστασης υπάρχει από το [28].

$$NIZKP[(u, r, v_1, v_2, \dots, v_n, s) : C = \text{com}_p(u, r) \wedge D = \text{com}_q(v_1, \dots, v_n, s) \wedge u = h_1^{v_1} \cdot \dots \cdot h_n^{v_n}].$$

5.1.5 Cryptographic Shuffle

Ως είσοδο το "κρυπτογραφικό ανακάτεμα" δέχεται μια λίστα $\mathbf{Z} = (z_1, z_2, \dots, z_n)$ με $z_i \in Z$. Το ανακάτεμα εφαρμόζει μία one-way συνάρτηση που χρησιμοποιεί κάποιο ιδιωτικό κλειδί $f : Z \times K \rightarrow Z$ σε κάθε επιμέρους είσοδο z_i και μεταθέτει το αποτέλεσμα με χρήση κάποιας μετάθεσης $\phi : [1, n] \rightarrow [1, n]$. Η έξοδος συνεπώς είναι μία λίστα $\mathbf{Z}' = (z'_1, \dots, z'_n)$ με $z'_j = f(z_i, k_i)$ με δείκτες που καθορίζονται από την μετάθεση $j = \phi(i)$ και κλειδιά $k_i \in K$. Στην έξοδο επισυνάπτεται απόδειξη μηδενικής γνώσης $\pi_{\mathbf{Z}}$.

$$(\mathbf{Z}', \pi_{\mathbf{Z}}) = \text{shuffle}_f^\phi(\mathbf{Z}, k_1, k_2, \dots, k_n).$$

Αρκεί ένα ειλικρινές *mix* ώστε η έξοδος να μην μπορεί να συνδεθεί με την είσοδο.

Το πρωτόκολλο χρησιμοποιεί σε δύο σημεία το Cryptographic Shuffle.

- Την πρώτη φορά ως είσοδο έχει την λίστα $\mathbf{E} = (E_1, \dots, E_n)$ από κρυπτογραφημένα μηνύματα ElGamal. $E_i \in \mathbb{G}_q \times \mathbb{G}_q$. Για τυχαίες τιμές $\gamma_i \in_R \mathbb{Z}_q \setminus \{0\}$ η συνάρτηση $\text{exp}(E_i, \gamma_i) = E_i^{\gamma_i}$ λαμβάνει ως είσοδο κάθε μήνυμα E_i . Στην έξοδο, $(\mathbf{E}', \pi_{\mathbf{E}}) = \text{shuffle}_{\text{exp}}^\phi(\mathbf{E}, \gamma_1, \dots, \gamma_n)$. Με το ανακάτεμα αυτό όλα τα κρυπτοκείμενα αποσυνδέονται από τα αρχικά στην λίστα \mathbf{E} με εξαίρεση το κρυπτοκείμενο "1".
- Στην δεύτερη περίπτωση ως είσοδο λαμβάνει μία λίστα $\mathbf{EE} = (\mathbf{E}_1, \dots, \mathbf{E}_n)$ όπου $\mathbf{E}_i = \{E_{i,1}, \dots, E_{i,n}\}$ με κρυπτοκείμενα ElGamal. Συνεπώς n^2 κρυπτοκείμενα $E_{i,j} \in \mathbb{G}_q \times \mathbb{G}_q$ περιέχονται στην λίστα \mathbf{EE} . Με τυχαιότητες $\mathbf{r}'_i = (r'_{i,1}, \dots, r'_{i,n}) \in_R \mathbb{Z}_q^n$ η συνάρτηση $\text{reEnc}_y(\mathbf{E}_i, \mathbf{r}'_i)$ εφαρμόζεται σε κάθε λίστα \mathbf{E}_i .

$$(\mathbf{EE}', \pi_{\mathbf{EE}}) = \text{shuffle}_{\text{reEnc}_y}^\phi(\mathbf{EE}, \mathbf{r}'_1, \dots, \mathbf{r}'_n)$$

και κρυπτογραφεί τα n^2 κρυπτοκείμενα μεταθέτοντας μόνο τις γραμμές κι όχι τις στήλες.

5.2 Περιγραφή Πρωτοκόλλου.

Το πρωτόκολλο αποτελείται από 4 επιμέρους διαδικασίες.

5.2.1 Εγγραφή Ψηφοφόρων.

Ο ψηφοφόρος:

1. Επιλέγει τυχαία το ιδιωτικό του κλειδί, $(\alpha, \beta) \in \mathbb{Z}_q \times \mathbb{Z}_q$.
2. Υπολογίζει το δημόσιο διαπιστευτήριο του $u = h_1^\alpha \cdot h_2^\beta \in \mathbb{G}_q$.
3. Στέλνει μέσω ασφαλούς καναλιού το u στις αρχές.

5.2.2 Election Preparation

1. Οι αρχές δημοσιεύουν την λίστα $U = ((V_1, u_1), \dots, (V_N, u_N))$ με τα στοιχεία που έχουν αποσταλεί από τους ψηφοφόρους. Κάθε ζεύγος (V_i, u_i) συνδέει ένα δημόσιο διαπιστευτήριο u_i με τον ψηφοφόρο V_i .
2. Υπολογίζουν τη λίστα $A = (\alpha_0, \dots, \alpha_N)$, $\alpha_i \in \mathbb{Z}_p$ των συντελεστών του πολυωνύμου $P(X) = \prod_{i=1}^N (X - u_i) \in \mathbb{Z}_p[X]$. Το πολυώνυμο χρησιμοποιείται από τους ψηφοφόρους για την δημιουργία απόδειξης μηδενικής γνώσης κατά την διαδικασία υποβολής της ψήφου.
3. Ανεξάρτητος γεννήτορας $\hat{h} \in \mathbb{G}_q$ καθορίζεται.
4. Τα στοιχεία (U, A, \hat{h}) δημοσιεύονται.

5.2.3 Vote Casting

Οι ψηφοφόροι επιλέγουν από το σύνολο $\mathbb{V} \subset \mathbb{G}_q$ τον υποψήφιο της επιλογής τους. Η κωδικοποίηση των υποψήφιων στο σύνολο \mathbb{V} καθώς και το δημόσιο κλειδί των αρχών y έχουν δημοσιευτεί.

Για την υποβολή της ψήφου ο ψηφοφόρος υπολογίζει τα ακόλουθα:

1. Δέσμευση $C = \text{com}_p(u, r)$ του δημόσιου διαπιστευτηρίου καθώς και απόδειξη μηδενικής γνώσης $\pi_1 = \text{NIZKP}[(u, r) : C = \text{com}_p(u, r) \wedge P(u) = 0]$ ώστε το u να αντιστοιχεί σε διαπιστευτήριο της λίστας U .
2. Δέσμευση $D = \text{com}_q(\alpha, \beta, s)$ των ιδιωτικών διαπιστευτηρίων καθώς και απόδειξη μηδενικής γνώσης $\pi_2 = \text{NIZKP}[(u, r, \alpha, \beta, s) : C = \text{com}_p(u, r) \wedge D = \text{com}_q(\alpha, \beta, s) \wedge u = h_1^\alpha \cdot h_2^\beta]$ ώστε να απαγορεύεται η χρήση διαπιστευτηρίου χωρίς γνώση των ιδιωτικών κλειδιών.
3. Κρυπτογραφεί $E = \text{enc}_y(\hat{h}^\beta, \rho)$ και $F = \text{enc}_y(v, \sigma)$ καθώς και απόδειξη μηδενικής γνώσης $\pi_3 = \text{NIZKP}[(\alpha, \beta, s, \rho, v, \sigma) : D = \text{com}_q(\alpha, \beta, s) \wedge E = \text{enc}_y(\hat{h}^\beta, \rho) \wedge F = \text{enc}_y(v, \sigma)]$ ώστε οι δύο δεσμεύσεις D, E έχουν χρησιμοποιήσει το ίδιο β και η τιμή v είναι γνωστή στον ψηφοφόρο.

Το ψηφοδέλτιο $B = (C, D, E, F, \pi_1, \pi_2, \pi_3)$ μέσω ανώνυμου καναλιού δημοσιεύεται. Ο ψηφοφόρος μπορεί να υποβάλλει πολλαπλά ψηφοδέλτια κατά τη διάρκεια της εκλογικής διαδικασίας. Φυσικά μόνο ένα θα καταμετρηθεί.

5.2.4 Tallying

Με το πέρας της διαδικασίας υποβολής τα περιεχόμενα του Bulletin Board επεξεργάζονται από τις αρχές. Χάρη ευκολίας θα θεωρήσουμε ότι υπάρχει μία έμπιστη αρχή που επεξεργάζεται τα περιεχόμενα ενώ στην πραγματικότητα πλειοψηφία έντιμων αρχών είναι απαραίτητη.

Η έμπιστη αρχή:

1. Ως είσοδο δέχεται τη λίστα \mathbf{B} με τα ψηφοδέλτια από τον Bulletin Board. Θεωρούμε ότι τα ψηφοδέλτια είναι τοποθετημένα με την χρονολογική σειρά που υποβλήθηκαν.
2. Ελέγχουν τις αποδείξεις μηδενικής γνώσης και διαγράφουν ψηφοδέλτια με μη έγκυρες αποδείξεις.
3. Από τα ψηφοδέλτια με έγκυρες αποδείξεις μηδενικής γνώσης επιλέγουν τα κρυπτοκείμενα (E, F) .
4. Η εναπομείνουσα λίστα συμβολίζεται με $\mathbf{E} = ((E_1, F_1), \dots, (E_n, F_n))$ και θεωρούμε ότι έχει διατηρηθεί η διάταξη της \mathbf{B} . Δηλαδή για $j > i$ το (E_j, F_j) έχει υποβληθεί μετά από το (E_i, F_i) . Ότι έχει απομείνει στη λίστα προέρχεται από ψηφοφόρο που διαθέτει έγκυρο διαπιστευτήριο. Προφανώς δύο $(E_i, F_i), (E_j, F_j) \in \mathbf{E}$ ανήκουν στον ίδιο ψηφοφόρο αν προκύπτουν από το ίδιο αρχικό κείμενο.
5. Για κάθε E_i υπολογίζουν λίστα $\mathbf{E}_i = (E_{i,1}, \dots, E_{i,n-1})$ με κρυπτοκείμενα ως εξής

$$E_{i,j} = \begin{cases} E_j & \text{for } j < i; \\ E_{j+1}/E_i & \text{if } j \geq i. \end{cases}$$

Η \mathbf{E}_i πιθανόν να περιέχει πολλαπλές κρυπτογραφήσεις της μονάδας. Στην περίπτωση αυτή υπάρχει $E_j \in \{E_{i+1}, \dots, E_n\}$ που να περιέχει το ίδιο αρχικό κείμενο με το E_i . Στην περίπτωση αυτή το (E_i, F_i) πρέπει να διαγραφεί καθώς υπάρχει μεταγενέστερη ψήφος από το ίδιο διαπιστευτήριο.

6. Για τον προσδιορισμό διπλών ψηφοδελτίων χωρίς την αποκρυπτογράφηση των \mathbf{E}_i οι αρχές καταρχάς δίνουν τη λίστα ως είσοδο στο Cryptographic Shuffle

$$(\mathbf{E}', \pi_{\mathbf{E}'}) = shuffle_{exp}^{\phi_i}(\mathbf{E}_i, \gamma_{i,1}, \dots, \gamma_{i,n-1})$$

σε κάθε \mathbf{E}_i με $\phi_i \in_R \Phi_{n-1}$ τυχαία μετάθεση και $\gamma_{i,j} \in_R \mathbb{Z}_q \setminus \{0\}$ οι τυχαιότητες. Σκοπός του ανακατέματος η πλήρης απόκρυψη των κρυπτοκειμένων που δεν είναι 1. Έστω $\mathbf{E}_i = (E'_{i,1}, \dots, E'_{i,n-1})$ η λίστα που προκύπτει και $\mathbf{F}_i = (F_i, E'_{i,1}, \dots, E'_{i,n-1})$ η λίστα που προκύπτει με την προσθήκη του F_i .

7. Οι αρχές δίνουν τη λίστα $\mathbf{FF} = (\mathbf{F}_1, \dots, \mathbf{F}_n)$ ως είσοδο στο Cryptographic Shuffle

$$(\mathbf{FF}', \pi_{\mathbf{FF}'}) = shuffle_{reEncy}^{\phi}(\mathbf{FF}, r'_1, \dots, r'_n)$$

για τυχαία μετάθεση $\phi \in_R \Phi_n$ και τυχειότητες $\mathbf{r}'_i = (r'_{i,1}, \dots, r'_{i,n}) \in \mathbb{Z}_q^n$. Σκοπός να μην υπάρχει συσχέτιση με τα αρχικά ψηφοδέλτια. Έστω $\mathbf{FF}' = (\mathbf{F}'_1, \dots, \mathbf{F}'_n)$ η λίστα που προκύπτει και $\mathbf{F}'_i = (F'_i, E''_{i,1}, \dots, E''_{i,n-1})$ κάθε στοιχείο του \mathbf{FF}' .

8. Για να προσδιορίσουν οι αρχές αν το στοιχείο F'_i πρέπει να διαγραφεί ελέγχουν αν $\text{dec}_x(E''_{i,j}) = 1$ για κάποιο $j \in [1, n-1]$. Έστω $U \subseteq [1, n]$ το υποσύνολο με δείκτες i για τους οποίους αυτο συμβαίνει και $V = [1, n-1] \setminus U$.
9. Για κάθε $i \in U$ οι αρχές επιλέγουν από το $\mathbf{F}'_i = (F'_i, E''_{i,1}, \dots, E''_{i,n-1})$ μία από τις κρυπτογραφήσεις της μονάδας $E''_{i,j}$ και κατασκευάζουν $NIZKP$

$$\hat{\pi}_i = NIZKP[(x) : 1 = \text{dec}_x(E''_{i,j}) \wedge y = h^x].$$

10. Για κάθε $i \in V$ οι αρχές υπολογίζουν $\mathbf{V}_i = \text{dec}_x(\mathbf{F}'_i)$ καθώς και απόδειξη μηδενικής γνώσης

$$\tilde{\pi}_i = NIZKP[(x) : \mathbf{V}_i = \text{dec}_x(\mathbf{F}'_i) \wedge y = h^x].$$

11. Για την ολοκλήρωση της καταμέτρησης ελέγχουν αν τα \mathbf{V}_i είναι στοιχεία του \mathbb{V} και αθροίζουν.
12. Δημοσιεύουν

$$(\mathbf{E}, \{\mathbf{E}_i, \mathbf{E}'_i, \pi_{\mathbf{E}_i}\}_{i=1}^n, \mathbf{FF}, \mathbf{FF}', \pi_{\mathbf{FF}}, U, \{E''_{i,j}, \hat{\pi}_i\}_{i \in U}, V, \{\mathbf{V}_i, \tilde{\pi}_i\}_{i \in V})$$

5.3 Ασφάλεια.

5.3.1 Correctness

Αν ο αντίπαλος δεν συνεργάζεται με τις αρχές και δεν διαθέτει κάποιο ιδιωτικό κλειδί υπάρχουν δύο μόνο εκδοχές με τις οποίες μπορεί να κατασκευάσει κάποιο ψηφοδέλτιο το οποίο να μπορεί να καταμετρηθεί.

- Επιχειρεί να υπολογίσει α', β' τέτοια ώστε $u = h_1^{\alpha'} h_2^{\beta'}$ για κάποιο $u \in U$. Τότε μπορεί να λύσει το πρόβλημα του διακριτού λογάριθμου.
- Επιχειρεί να κατασκευάσει απόδειξη χωρίς να γνωρίζει κάποιο ζευγάρι (α', β') . Αδύνατο από το computational soundness των π_1, π_2, π_3 .

Αν ο αντίπαλος είναι ψηφοφόρος και υποβάλλει πολλαπλά ψηφοδέλτια μόνο το τελευταίο θα καταμετρηθεί. Χωρίς να χρησιμοποιήσει το ιδιωτικό του διαπιστευτήριο δεν διαφέρει από κάθε άλλο αντίπαλο.

Αν ο αντίπαλος συνεργάζεται με τις αρχές ίσως προσπαθήσει να διαγράψει, να αλλοιώσει ψηφοδέλτια σε κάποιο από τα στάδια της καταμέτρησης. Αυτό δεν επιτρέπεται από τις αποδείξεις $\pi_{\mathbf{E}_i}, \pi_{\mathbf{FF}}, \hat{\pi}_i$ και $\tilde{\pi}_i$ οι οποίες μπορούν δημόσια να επαληθευτούν.

5.3.2 Everlasting Privacy

Κάθε ψηφοδέλτιο που έχει υποβληθεί μέσω ανώνυμου καναλιού δεν διαθέτει καμία πληροφορία για την ταυτότητα του ψηφοφόρου. Φυσικά στο μέλλον ο αντίπαλος μπορεί να υπολογίσει το ιδιωτικό κλειδί x συνεπώς και το β από το \hat{h}^β . Ακόμη και τότε όμως για κάθε διαπιστευτήριο $u' \in U$ υπάρχει α' τέτοιο ώστε $u' = h_1^{\alpha'} h_2^\beta$ συνεπώς καμία πληροφορία δεν διαρρέει για το u δεδεδεμένου ότι οι αποδείξεις π_1, π_2, π_3 είναι μηδενικής γνώσης.

5.3.3 Coercion Resistance

Ο ψηφοφόρος είτε εθελοντικά είτε επειδή υπέκυψε σε εκβιασμό μπορεί να παραδώσει τις τυχαιότητες που χρησιμοποίησε. Ο αντίπαλος μπορεί τότε να υπολογίσει τα \hat{h}^β καθώς και το u . Πρέπει επίσης να αποδείξει ότι δεν έχει υποβάλλει άλλη ψήφο ή διαφορετικά ότι κάθε άλλη ψήφος που έπεται έχει υποβληθεί από άλλον ψηφοφόρο. Αδύνατο καθώς πως μπορεί να αποδείξει ότι δεν γνωρίζει τα διαπιστευτήρια που ακολουθούν; Διαφορετικά μπορεί να επιχειρήσει να αποδείξει ότι τα E_{i+1}, \dots, E_n δεν είναι κρυπτογραφήσεις του \hat{h}^β ή ισοδύναμα ότι το \mathbf{E}_i δεν περιέχει κρυπτογράφιση του 1 ή να βρει κάποιου είδους συσχέτιση ώστε $\mathbf{F}_i \in \mathbf{FF}$ και $\mathbf{F}'_{\phi(i)} \in \mathbf{FF}'$. Ισοδύναμα σε δυσκολία με τα DDH και DL .

Επιθέσεις εκβιασμού μπορούν να αντιμετωπιστούν καθώς ο ψηφοφόρος δεν μπορεί να αποδείξει ότι δεν έχει υποβάλλει την τελευταία στιγμή κάποιο ψηφοδέλτιο. Ακόμη κι αν ο ψηφοφόρος υποκύψει στον εκβιασμό και υποβάλλει για παράδειγμα τυχαία ψήφο μπορεί αργότερα να υποβάλλει νέο ψηφοδέλτιο που θα αναιρεί το πρώτο (randomization attack).

Δεδομένου ότι το πρωτόκολλο παρέχει Everlasting Privacy δηλαδή δεν υπάρχει συσχέτιση μεταξύ ψηφοφόρου και διαπιστευτηρίου δεν έχει αποτέλεσμα και καμία forced abstention attack.

Ακόμη και στην περίπτωση που τα προσωπικά διαπιστευτήρια αποκτηθούν από τον αντίπαλο ο ψηφοφόρος μπορεί να καταθέσει ψηφοδέλτιο την τελευταία στιγμή με την επιλογή της αρεσκείας του. (simulation attack).

5.4 Συμπέρασμα.

Οι συγγραφείς ισχυρίζονται ότι είναι το πρώτο πρωτόκολλο που παρουσιάστηκε και παρέχει Everlasting Privacy καθώς και Coercion Resistance. Η πολυπλοκότητα είναι τετραγωνική συνεπώς είναι κατάλληλο μόνο για εκλογές με περιορισμένο πλήθος ψηφοφόρων.

5.5 Σχόλια.

1. Αν ο ψηφοφόρος εκβιαστεί να παραδώσει τα προσωπικά του διαπιστευτήρια δεν έχει άλλη επιλογή από το να υποκύψει. Ακόμη κι αν προσπαθήσει να παραδώσει κάποιο $u = h_1^\alpha \cdot h_2^\beta \in U$ με διαφορετικά α', β' ώστε να αποτύχει ο αντίπαλος στις αποδείξεις μηδενικής γνώσης αυτό είναι αδύνατο καθώς είναι ισοδύναμο με τον υπολογισμό διακριτού λογαρίθμου.
2. Υποθέτουμε ότι ο αντίπαλος διαθέτει αρκετή υπολογιστική ισχύ ώστε να μπορεί να υποβάλλει ψηφοδέλτια λίγες μόνο στιγμές πριν από το χρονικό όριο λήξης του πρωτοκόλλου. Δεδομένου

ότι διαθέτει τα προσωπικά διαπιστευτήρια των ψηφοφόρων που υπέκυψαν στον εκβιασμό είναι δυνατό να αποφύγουν τον εκβιασμό; Είναι ανθεκτικό δηλαδή το πρωτόκολλο σε Simulation Attack ;

3. Το πρωτόκολλο θεωρεί ότι ο αντίπαλος δεν είναι σε θέση να καταγράψει και να αποθηκεύσει για μελλοντική χρήση όλη την πληροφορία στο ανώνυμο κανάλι που χρησιμοποιεί ο ψηφοφόρος.
4. Το πρωτόκολλο θεωρεί ότι ο αντίπαλος δεν είναι σε θέση να επηρεάσει την λειτουργία των συσκευών που χρησιμοποιούνται από το πρωτόκολλο.

Chapter 6

Efficient coercion resistant and everlasting privacy in remote electronic elections

Στο παρόν κεφάλαιο παρουσιάζεται ένα εκλογικό πρωτόκολλο που παρέχει **Coercion-Resistance** καθώς και **Everlasting Privacy**. Το κεφάλαιο βασίζεται στο άρθρο [11] των Panagiotis Grontas, Aris Pagourtzis, Alexandros Zacharakis. Θα ξεκινήσουμε το κεφάλαιο με το εκλογικό πρωτόκολλο των Atsushi Fujioka, Tatsuaki Okamoto, Kazuo Ohta ([24]).

6.1 FOO.

Επιμέρους δομικοί λίθοι του πρωτόκολλου είναι φυσικά οι ψηφοφόροι, μία αρχή που ελέγχει τα διαπιστευτήρια καθώς και ένας δημόσιος πίνακας Bulletin Board ή ένας counter ο οποίος καταμετρά τα ψηφοδέλτια. Οι ψηφοφόροι χρησιμοποιούν ανώνυμα κανάλια για την επικοινωνία τους με τον counter.

Κάθε ψηφοφόρος διαθέτει κάποιο σχήμα ψηφιακής υπογραφής, η αρχή διαθέτει σχήμα τυφλών υπογραφών. Οι ψηφοφόροι δεσμεύουν την επιλογή τους με σχήμα δέσμευσης.

Σε ότι ακολουθεί με

1. V_i συμβολίζεται ο ψηφοφόρος i .
2. A συμβολίζεται η αρχή.
3. C ο counter.
4. $\xi(v, k)$ το σχήμα δέσμευσης για το μήνυμα v με τυχαιότητα k .
5. $\sigma_i(m)$ το σχήμα υπογραφών του ψηφοφόρου V_i .
6. $\sigma_A(m)$ το σχήμα υπογραφών της Αρχής.
7. $\chi_A(m, r)$ η blinding τεχνική για το μήνυμα m με τυχαιότητα r .

-
-
8. $\delta_A(s, r)$ αποκάλυψη της τυφλής υπογραφής.
 9. ID_i το διαπιστευτήριο του ψηφοφόρου V_i .
 10. v_i η ψήφος του ψηφοφόρου V_i .

Περιγραφή Πρωτοκόλλου.

1. Προετοιμασία:

- Ο ψηφοφόρος V_i επιλέγει την ψήφο του v_i και δεσμεύεται σ' αυτήν $x_i = \xi(v_i, k_i)$ με τυχαιότητα k_i .
- Ο ψηφοφόρος *blinds* x_i με τυχαιότητα r_i υπολογίζοντας το $e_i = \xi(x_i, r_i)$
- Ο ψηφοφόρος υπογράφει το e_i , $s_i = \sigma_i(e_i)$ και στέλνει στην αρχή A τα $\langle ID_i, e_i, s_i \rangle$.

2. Αρχή:

- Η αρχή ελέγχει αν ο ψηφοφόρος έχει δικαίωμα συμμετοχής. Αν όχι απορρίπτει το ψηφοδέλτιο.
- Η αρχή ελέγχει αν ο ψηφοφόρος έχει αιτηθεί είδη υπογραφή. Αν ναι απορρίπτει το αίτημα.
- Η αρχή ελέγχει την υπογραφή s_i του ψηφοφόρου στο μήνυμα e_i . Αν είναι έγκυρη υπογράφει και στέλνει στο ψηφοφόρο $d_i = \sigma_A(e_i)$. Αν η υπογραφή είναι πλαστή απορρίπτει το αίτημα.
- Όταν η διαδικασία περατωθεί η αρχή ανακοινώνει το πλήθος των ψηφοφόρων που έχουν θετική απάντηση στα αιτήματά τους. Δημοσιεύει λίστα $\langle ID_i, e_i, s_i \rangle$.

3. Κατάθεση ψηφοδελτίων:

- Ο ψηφοφόρος αποκτά την υπογραφή της αρχής $y_i = \delta(d_i, r_i)$.
- Ελέγχει αν η υπογραφή που έλαβε είναι έγκυρη. Διαφορετικά διαμαρτύρεται παρουσιάζοντας τα $\langle x_i, y_i \rangle$.
- Ο ψηφοφόρος στέλνει τα $\langle x_i, y_i \rangle$ στον counter μέσω ανώνυμου καναλιού.

4. Έλεγχος υπογραφών:

- Ο counter ελέγχει την υπογραφή y_i της αρχής στο x_i . Αν ο έλεγχος επιτύχει τοποθετεί σε λίστα δίπλα από αύξοντα αριθμό l , τα $\langle x_i, y_i \rangle$. Σε κάθε γραμμή η λίστα έχει $\langle l, x_i, y_i \rangle$.
- Όταν όλοι οι ψηφοφόροι καταθέσουν τα ψηφοδέλτια τους. Η λίστα δημοσιεύεται.

5. Συγκέντρωση ιδιωτικών κλειδιών:

- Ελέγχεται αν ο αριθμός των ψηφοδελτίων είναι ίσος με τον αριθμό των υπογραφών που εξέδωσε η αρχή.

-
- Ο ψηφοφόρος V_i ελέγχει αν η ψήφος του υπάρχει στην λίστα. Αν όχι διαμαρτύρεται παρουσιάζοντας τα $\langle x_i, y_i \rangle$.
 - Ο ψηφοφόρος στέλνει τα $\langle l, k_i \rangle$ στον counter μέσω ανώνυμου καναλιού.

6. Καταμέτρηση:

- (α) Ο counter ανοίγει την δέσμευση αποκαλύπτοντας την ψήφο v_i . Ελέγχει αν η ψήφος είναι έγκυρη.
- (β) Ανακοινώνει το τελικό αποτέλεσμα.

6.2 Αρχές.

Το πρωτόκολλο απαρτίζεται από τις ακόλουθες αρχές:

1. **Ψηφοφόροι.** Θεωρούμε n το πλήθος ψηφοφόρους. Αν ανήκουν σε κάποιο συγκεκριμένο σύνολο C συμβολίζουμε το πλήθος τους ως n_C . Θεωρούμε ότι οι ψηφοφόροι καταθέτουν πολλαπλά ψηφοδέλτια ώστε να συμβάλλουν στη δημιουργία αβεβαιότητας ως προς τα διαπιστευτήρια και τις προθέσεις του εκλογικού σώματος.
2. **Registration Authorities RA.** Οι αρχές επωμίζονται την ευθύνη της επαλήθευσης των στοιχείων των ψηφοφόρων και της παραγωγής διαπιστευτηρίων. Διαδικασία που γίνεται πριν την έναρξη της εκλογικής διαδικασίας μέσω untappable channel.
3. **Tallying Authorities TA.** Οι αρχές με χρήση τυφλών υπογραφών και ενός ιδιωτικού *bit* εξουσιοδοτούν τους εγγεγραμμένους ψηφοφόρους. Είναι επίσης υπεύθυνες για την καταμέτρηση και την έκδοση του τελικού αποτελέσματος.

Τόσο οι Registration Authorities όσο και οι Tallying Authorities αποτελούνται από διάφορες οντότητες πιθανόν με αντικρουόμενα συμφέροντα. Υποθέτουμε ότι οι πλειοψηφία των αρχών είναι ειλικρινείς. Για την κρυπτογράφηση και υπογραφή ψήφων χρησιμοποιούν κάποιο Threshold Cryptosystem.

6.3 Δομικοί Λίθοι.

Συνοπτικά αναφέρουμε ότι χρησιμοποιούνται τα ακόλουθα:

- Bulletin Board (BB).
- Homomorphic Encryption Scheme.
- Plaintext Equivalence Test.

6.3.1 *Blind Signatures*

Θα επιμείνουμε λίγο παραπάνω στις τυφλές υπογραφές.

Οι τυφλές υπογραφές προτάθηκαν από τον David Chaum στο [12]. Επιτρέπουν στον signer την υπογραφή μηνυμάτων χωρίς να γνωρίζει το περιεχόμενό τους. Ο χρήστης αποκρύπτει (blinds) τα περιεχόμενα του μηνύματος και ο υπογράφων υπογράφει χωρίς να γνωρίζει το περιεχόμενο. Έπειτα ο χρήστης επαναφέρει το μήνυμα στην αρχική του μορφή (unblinds) και έχει πλέον στη διάθεση του μία υπογραφή για το αρχικό μήνυμα.

Η ασφάλεια τους έχει μελετηθεί εκτεταμένα. Οι σημαντικότερες ιδιότητες που πρέπει να διαθέτουν είναι οι blindness ή unlinkability ώστε ο υπογράφων να μην μπορεί να αντλήσει το αρχικό κείμενο από την υπογραφή του και η Unforgeability ώστε ο χρήστης να μην μπορεί να παράγει έγκυρες υπογραφές μόνος παρά μόνο όσες του παρέχει ο υπογράφων.

6.4 Conditional Blind Signatures.

Το πρωτόκολλο βασίζεται σε ένα νέο primitive τις τυφλές υπογραφές υπό συνθήκη CBS. Οι τυφλές υπογραφές υπό συνθήκη επιτρέπουν σε υπογράφων \mathcal{S} να παράγει υπογραφές σε μηνύματα που έχει υποβάλει ο χρήστης \mathcal{U} . Η επαλήθευση μπορεί να πραγματοποιηθεί μόνο από συγκεκριμένο Verifier \mathcal{V} . Η επαλήθευση της υπογραφής πραγματοποιείται με χρήση ενός μυστικού *bit* καθώς και με ιδιωτικό κλειδί που κατέχει ο Verifier. Η επαλήθευση της υπογραφής εξαρτάται από το μυστικό *bit*. Αυτό παρέχει στον υπογράφων την ικανότητα να υποδείξει στον Verifier αν θα αποδεχθεί ή όχι την υπογραφή. Το μυστικό *bit* παραμένει μυστικό κι από τον χρήστη καθώς είναι απαραίτητο να μην μπορεί ούτε ο ίδιος να διακρίνει μεταξύ των δύο περιπτώσεων.

Ακολουθούν οι βασικοί ορισμοί.

Ορισμός 1.

Μία τυφλή υπογραφή υπό συνθήκη αποτελείται από τρία επιμέρους πρωτόκολλα ($Gen, Sign, Vrfy$) με τις ακόλουθες ιδιότητες.

1. Gen είναι ένας αλγόριθμος που δέχεται ως είσοδο μία παράμετρο ασφαλείας 1^λ και στην έξοδο παράγει
 - Δύο ζευγάρια κλειδιών (sk_S, pk_S) για την υπογραφή και (sk_V, pk_V) για την επαλήθευση.
 - Τον χώρο μηνυμάτων \mathbb{M} .
 - Τον χώρο υπογραφών \mathbb{S} .

και οι δύο χώροι περιγράφονται από κάποιες παραμέτρους π.χ. γεννήτορες ομάδων. Συμβολίζουμε με $pk = (pk_S, pk_V)$ τα δημόσια κλειδιά και με $sk = (sk_S, sk_V)$ τα ιδιωτικά αντίστοιχα.

2. $Sign$ είναι το πρωτόκολλο που εκτελείται μεταξύ του χρήστη \mathcal{U} και του υπογράφων \mathcal{S} . Η δημόσια είσοδος αποτελείται από τις παραμέτρους και τα δημόσια κλειδιά. Τα μυστικά

κομμάτια της εισόδου είναι το ιδιωτικό κλειδί sk_S του υπογράφων καθώς και το μυστικό bit ενώ ο χρήστης συμμετέχει με το μήνυμα m στο οποίο ζητά υπογραφή. Στην έξοδο το πρωτόκολλο παράγει υπογραφή sig του m στον \mathcal{U} .

3. $Vrfy$ είναι αλγόριθμος που με είσοδο $(sk_V, params, pk, m, sig)$ παράγει στην έξοδο bit με το οποίο αποδέχεται ή απορρίπτει την υπογραφή.

Αν το μυστικό bit είναι 1 ο αλγόριθμος $Vrfy(sk_V, params, pk, m, sig)$ αποδέχεται την υπογραφή. Σε διαφορετική περίπτωση την απορρίπτει.

Η ασφάλεια του πρωτοκόλλου βασίζεται στην ασφάλεια που παρέχουν οι υπογραφές δηλαδή το blindness και One More Forgery.

Για να περιγράψουμε αυστηρότερα την ιδέα πως ένα bit καθορίζει την εγκυρότητα της υπογραφής ορίζουμε μία νέα ιδιότητα την Conditional Verifiability με χρήση του παιχνιδιού $CondVerExp$ που παρουσιάζεται στον αλγόριθμο 1.

Στο συγκεκριμένο παιχνίδι ο αντίπαλος \mathcal{A} δέχεται έγκυρες κι άκυρες υπογραφές της επιλογής του και παράγει ένα μήνυμα το οποίο θα στείλει για υπογραφή. Ανάλογα με το αποτέλεσμα της ρίψης ενός νομίσματος είτε θα του παραδοθεί μία έγκυρη υπογραφή είτε μία μη έγκυρη. Καλείται να μαντέψει το αποτέλεσμα της ρίψης. Επιτρέπεται να συνεχίσει να παίρνει υπογραφές της επιλογής του.

Ορισμός 2.

Ένα σχήμα τυφλών υπογραφών υπό συνθήκη Π διαθέτει την ιδιότητα Conditional Verifiability αν για κάθε PPT αντίπαλο \mathcal{A} υπάρχει negligible συνάρτηση $negl$ τέτοια ώστε

$$Pr[CondVerExp(\lambda)_{\mathcal{A}, \Pi} = 1] \leq \frac{1}{2} + negl(\lambda).$$

Algorithm 1: $CondVerExp_{\mathcal{A}, \Pi}$

Input: security parameter λ

Output: $x \in \{0, 1\}$

$(sk, pk, params) \leftarrow Gen(1^\lambda)$

$\{(m_i, sig_i) \leftarrow Sign \langle \mathcal{S}(params, sk_S, b_i), \mathcal{A}(params, pk, \{m_j, sig_j\}_{j=1}^{i-1}, b_i) \rangle\}_{i=1}^{l_1}$

$m_c \leftarrow \mathcal{A}(pk, params, \{(m_i, sig_i)\}_{i=1}^{l_1}, Challenge)$

$b \leftarrow_R \{0, 1\}$

$(\epsilon, sig_c) \leftarrow Sign \langle \mathcal{S}(params, sk, b), \mathcal{A}(params, pk, m_c) \rangle$

$\{(m_i, sig_i) \leftarrow Sign \langle \mathcal{S}(params, sk_S, b_i), \mathcal{A}(params, pk, \{m_j, sig_j\}_{j=1}^{i-1}, b_i) \rangle\}_{i=l+1}^{l_2}$

$b' = \mathcal{A}(\{m_i, sig_i\}_{i=1}^{l_1+l_2}, m_c, sig_c, Guess)$

return 1 **iff** $b = b'$

Ορισμός 3.

Ένα σχήμα τυφλών υπογραφών υπό συνθήκη Π είναι ασφαλές αν ικανοποιεί τις ιδιότητες **Perfect Blindness, Strong One More Forgery, Conditional Verifiability**.

6.5 Πρωτόκολλο.

Στην λεπτομερή περιγραφή του πρωτοκόλλου θα χρησιμοποιήσουμε μία παραλλαγή των τυφλών υπογραφών υπό συνθήκη CBS που βασίζεται στις τυφλές υπογραφές των Okamoto-Schnorr.

Βασική διαφορά είναι ότι αντικαθίσταται το ζεύγος (y_1, y_2) της τυφλής υπογραφής του πρωτοκόλλου που παρουσιάζεται στο [13] με το $(Enc_h(k^{y_1}), y_2)$ όπου ο πρώτος παράγοντας έχει υψωθεί σε δύναμη και κρυπτογραφηθεί με $M\text{-ElGamal}$. Το k είναι κάποιο στοιχείο με διακριτό λογάριθμο γνωστό μόνο στον Verifier.

Για το πρωτόκολλο υπογραφής υποθέτουμε ότι υπάρχει hash function $\mathcal{H} : \{0, 1\}^* \rightarrow \mathbb{Z}_q$ που μοντελοποιείται από τυχαίο μαντείο.

Ο αλγόριθμος ξεκινά όπως και στο πρωτόκολλο [13]. Ο υπογράφων δεσμεύεται σε μια τυχαία επιλογή. Ο χρήστης επιλέγει τους παράγοντες με τους οποίους θα αποκρύψει τη δέσμευση και την έξοδο της συνάρτησης $Hash$ που θα υπογραφεί. Η παραλλαγή ξεκινά όταν ο υπογράφων είναι έτοιμος να υπογράψει.

- Αν το μυστικό *bit* του \mathcal{S} είναι 1, μετά τη δημιουργία του ζεύγους (y_1, y_2) υψώνει το δημόσιο κλειδί k του Verifier στον πρώτο παράγοντα, κρυπτογραφεί και παράγει το $(Enc_h(k^{y_1}), y_2)$.
- Αν το μυστικό *bit* του \mathcal{S} είναι 0, απλά στέλνεται ένα τυχαίο κρυπτοκείμενο.

Σε κάθε περίπτωση η δεύτερη συντεταγμένη δεν αλλάζει. Το unblinding στην πρώτη συντεταγμένη πραγματοποιείται με χρήση των ομομορφικών ιδιοτήτων του κρυπτοσυστήματος.

Algorithm 2: Key Generation Algorithm for CBS

Input: security parameter λ
Output: $(sk_S, vk_S), (sk_V, pk_V), (sk_E, pk_E), params$
% Select group \mathbb{G} of prime order $q > 2^\lambda$ with hard DDH problem
 $(q, \mathbb{G}) \leftarrow GroupGen(1^\lambda)$
% Select the appropriate generators
 $(g_1, g_2) \leftarrow_R \mathbb{G}$
 $(h_1, h_2) \leftarrow_R \mathbb{G}$
 $params \leftarrow (q, \mathbb{G}, g_1, g_2, h_1, h_2)$
% Select secret sk_S and public vk_S Signature Keys for \mathcal{S}
 $s_1, s_2 \leftarrow_R \mathbb{Z}_q$
 $v \leftarrow g_1^{-s_1} g_2^{-s_2}$
 $(sk_S, vk_S) \leftarrow ((s_1, s_2), v)$
% Select secret sk_V and public pk_V verification keys for \mathcal{V}
 $s \leftarrow_R \mathbb{Z}_q$
 $k \leftarrow g_1^s$
 $(sk_V, pk_V) \leftarrow (s, k)$
% Select secret sk_E and public pk_E encryption keys for \mathcal{V}
 $z \leftarrow_R \mathbb{Z}_q$
 $h \leftarrow h_1^z$
 $(sk_E, pk_E) \leftarrow (z, h)$

Ο αλγόριθμος 2 περιγράφει τη δημιουργία των κλειδιών. Ας περιγράψουμε το πρωτόκολλο.

1. Στην είσοδο έχουμε,
 - Οι παράμετροι $params$ και το δημόσιο κλειδί vk_S .
 - Το ιδιωτικό κλειδί του υπογράφων s_k και το μυστικό bit .
 - Το μήνυμα του χρήστη m .
2. Ο υπογράφων επιλέγει τυχαία $r_1, r_2 \leftarrow_r \mathbb{Z}_q$ και υπολογίζει το $x \leftarrow g_1^{r_1} g_2^{r_2}$. Στέλνει το x στον ψηφοφόρο.
3. Ο ψηφοφόρος επιλέγει τυχαία $u_1, u_2, d \leftarrow_R \mathbb{Z}_q$ και υπολογίζει το $x^* \leftarrow x g_1^{u_1} g_2^{u_2} v^d$, $e^* \leftarrow \mathcal{H}(m, x^*)$ και $e \leftarrow e^* - d$. Στέλνει το e στον υπογράφων.
4. Ο υπογράφων υπολογίζει $y_1 \leftarrow r_1 + e s_1$, $y_2 \leftarrow r_2 + e s_2$ και στέλνει $(bsig_1, bsig_2)$ όπου $bsig_2 \leftarrow y_2$ και $bsig_1 \leftarrow Enc_h(k^{y_1})$ αν $b = 1$ και $bsig_1 \leftarrow_R \mathbb{G}^3$ διαφορετικά.
5. Ο ψηφοφόρος υπολογίζει $sig_1 \leftarrow bsig_1 \cdot Enc_h(k^{u_1})$ και $sig_2 \leftarrow bsig_2 + u_2$.
6. Στην έξοδο ο ψηφοφόρος λαμβάνει $(m, x^*, e^*, sig_1, sig_2)$.

Signature Protocol

Singer

Input: $(params, vk_S, sk_S, b)$

$$r_1, r_2 \leftarrow_R \mathbb{Z}_q$$

$$x \leftarrow g_1^{r_1} g_2^{r_2}$$

$$y_1 \leftarrow r_1 + es_1$$

$$y_2 \leftarrow r_2 + es_2$$

$$bsig_1 \leftarrow Enc_h(k^{y_1}) \text{ (if } b = 1 \text{ else } bsig_2 \leftarrow_R \mathbb{G}$$

$$bsig_2 \leftarrow y_2$$

\xrightarrow{x}

Recipient

Input: $(params, pk_S, m)$

$$u_1, u_2, d \leftarrow_R \mathbb{Z}_q$$

$$x^* \leftarrow x g_1^{u_1} g_2^{u_2} v^d$$

$$e^* \leftarrow \mathcal{H}(m, x^*)$$

$$e \leftarrow e^* - d$$

\xleftarrow{e}

$\xrightarrow{(bsig_1, bsig_2)}$

$$sig_1 \leftarrow bsig_1 \cdot Enc_h(k^{u_1})$$

$$sig_2 \leftarrow bsig_2 + u_2$$

Output: $(m, x^*, e^*, sig_1, sig_2)$

Για την επαλήθευση ο Verifier ελέγχει αν $x^{*s} = y_1' g_2^{y_2' \cdot s} v^{e^* \cdot s}$ και $m = m'$.

$$x^{*s} = y_1' g_2^{y_2' \cdot s} v^{e^* \cdot s}$$

$$(x g_1^{u_1} g_2^{u_2} v^d)^s = k^{y_1 + u_1} g_2^{(y_2 + u_2)s} v^{(e+d)s}, k = g_1^s$$

$$x g_1^{u_1} g_2^{u_2} v^d = g_1^{y_1 + u_1} g_2^{y_2 + u_2} v^{e+d}$$

$$x g_1^{u_1} g_2^{u_2} = g_1^{y_1 + u_1} g_2^{y_2 + u_2} v^e$$

$$g_1^{r_1} g_2^{r_2} g_1^{u_1} g_2^{u_2} = g_1^{y_1 + u_1} g_2^{y_2 + u_2} (g_1^{-s_1} g_2^{-s_2})^e$$

$$g_1^{r_1} g_2^{r_2} = g^{y_1 - es_1} g_2^{y_2 - es_2}$$

$$g_1^{r_1} g_2^{r_2} = g_1^{r_1} g_2^{r_2}$$

Αν η μυστική είσοδος είναι $b = 1$ τότε η επαλήθευση θα είναι καταφατική διαφορετικά η επαλήθευση θα αποτύχει με μεγάλη πιθανότητα. Ο Verifier ενημερώνεται για το μυστικό *bit* του υπογράφων και απορρίπτει το ψηφοδέλτιο.

Algorithm 3: Signature Verification

Input: $s, z, params, \mathcal{H}, m, sig = (x^*, e^*, sig_1, sig_2)$

Output: $b \in \{0, 1\}$

$e^* \leftarrow \mathcal{H}(m, x^*)$

$y'_1 \leftarrow Dec_z(sig_1)$

$y'_2 \leftarrow sig_2$

Return 1 iff $x^{*s} = y'_1 y'_2 v^{e^* \cdot s}$ and $m = m'$

Το πρωτόκολλο ικανοποιεί τον ορισμό 3.

6.6 Παραλλαγή ως προς την επικοινωνία.

Παρουσιάζουμε μία περισσότερο αποδοτική παραλλαγή του πρωτοκόλλου μειώνοντας την επικοινωνία μεταξύ Signer και του ψηφοφόρου. Το κλειδί επαλήθευσης μπορεί να χρησιμοποιηθεί για τις υπογραφές ώστε στη συγκεκριμένη περίπτωση υπογράφων και Verifier δύναται να είναι το ίδιο πρόσωπο. Αν η τυχαιότητα για τη δέσμευση στο μήνυμα x προκαθοριστεί με κάποιο τρόπο, οι δύο συμμετέχοντες μπορούν να την υπολογίσουν μειώνοντας τους γύρους επικοινωνίας.

Παρουσιάζουμε το νέο πρωτόκολλο.

1. Ως είσοδο έχουμε τις παραμέτρους $params$, το δημόσιο κλειδί, το v και το x . Ο υπογράφων χρησιμοποιεί το ιδιωτικό κλειδί s καθώς και το μυστικό *bit*, b . Ο ψηφοφόρος το μήνυμα m στο οποίο αιτείται υπογραφή.
2. Το *blinding* βήμα είναι όπως και πριν.
3. Ο υπογράφων υπολογίζει $(bsig_1, bsig_2)$ ανάλογα με το b . Αν $b = 1$ τότε επιλέγει τυχαίο $bsig_2 \leftarrow_R \mathbb{Z}_q$ και υπολογίζει $bsig_1 \leftarrow Enc_h((x \cdot g_2^{-y_2} \cdot v^{-e})^s)$. Αν $b = 0$ τότε $(bsig_1, bsig_2) \leftarrow_R \mathbb{G}^3 \times \mathbb{Z}_q$.
4. Το *unblinding* βήμα είναι όπως και πριν.

Η υπογραφή επαληθεύει την εξίσωση επαλήθευσης του αλγόριθμου 3. Η εικόνα του \mathcal{U} παραμένει ίδια για τυχαίο x ώστε η ασφάλεια του πρωτοκόλλου δεν αλλάζει.

6.7 Περιγραφή Πρωτοκόλλου

Για να μειωθούν οι γύροι επικοινωνίας το πρωτόκολλο βασίζεται σε μια παραλλαγή του *FOO* [14]. Σε μία επιπλέον φάση εξουσιοδότησης τα διαπιστευτήρια μυστικά προσημειώνονται ως έγκυρα ή άκυρα με χρήση του σχήματος των υπό συνθήκη τυφλών υπογραφών. Η διαδικασία υπογραφής περιγράφεται από τη συνάρτηση *auth*.

$$\text{auth}(\text{Enc}_h(\sigma), \text{Enc}_h(\sigma'), \text{blinded} - \text{ballot}, sk, pk, \text{params}) = \text{Sign}_{CBS}(\sigma =? \sigma', \text{blinded} - \text{ballot}, sk, pk, \text{params})$$

όπου sk, pk, params είναι τα κλειδιά και οι παράμετροι του σχήματος *CBS* καθώς και του κρυπτογραφικού πρωτοκόλλου. Υπολογίζει $(bsig_1, bsig_2)$ έγκυρη ή άκυρη υπογραφή ανάλογα με το αν $\sigma = \sigma'$ ή όχι. Με σ συμβολίζεται το διαπιστευτήριο που δόθηκε από τις αρχές *RA* κατά την διαδικασία εγγραφής του ψηφοφόρου και με σ' το διαπιστευτήριο που παρέχεται από τον ψηφοφόρο κατά την διαδικασία εξουσιοδότησης.

Η συνάρτηση *auth* μας επιτρέπει να διαχωρίσουμε την ηλεκτρονική ψηφοφορία σε δύο στάδια ώστε να μειωθεί η τετραγωνική πολυπλοκότητα του *JCJ*. Ο έλεγχος των διαπιστευτηρίων με *PET* γίνεται στην φάση εξουσιοδότησης και όχι στην φάση καταμέτρησης.

Οι αρχές *TA* ελέγχουν την εγκυρότητα του διαπιστευτηρίου. Ο έλεγχος μπορεί να γίνει σε σταθερό χρόνο καθώς είναι γνωστή η ταυτότητα του ψηφοφόρου (όχι όμως και η ψήφος) και οι αρχές έχουν πρόσβαση στη λίστα με τα διαπιστευτήρια. Με *pet* ελέγχουν αν το διαπιστευτήριο υπάρχει στην λίστα των διαπιστευτηρίων εξακριβώνοντας αν ο ψηφοφόρος εκβιάζεται. Αν η ταυτότητα του ψηφοφόρου είναι γνωστή ο έλεγχος των διπλών ψηφοδελτίων είναι εύκολος ελέγχοντας όλα τα διαπιστευτήρια που σχετίζονται με τον ψηφοφόρο.

Αν ο ψηφοφόρος χρησιμοποιεί ένα πραγματικό διαπιστευτήριο τότε οι αρχές εκδίδουν μία έγκυρη υπογραφή ($b = 1$) που συνεπάγεται και καταμέτρηση της ψήφου που θα υποβληθεί αργότερα. Διαφορετικά ($b = 0$) η υπογραφή που εκδίδεται είναι άκυρη και το ψηφοδέλτιο δεν θα καταμετρηθεί. Από την κατασκευή του σχήματος *CBS* η εγκυρότητα ή μη της ψήφου δεν μπορεί να διαπιστωθεί από ψηφοφόρο ή εκβιαστή.

Έπειτα ο ψηφοφόρος *unblinds* την ψήφο και δημοσιεύει στο *Bulletin Board*. Οι αρχές ελέγχουν την υπογραφή και είτε καταμετρούν είτε όχι την ψήφο. Η ταυτότητα του ψηφοφόρου δεν περιέχεται στο ψηφοδέλτιο ώστε να μην μπορεί να συσχετιστεί με την ψήφο.

6.7.1 Αρχικοποίηση.

Παράμετροι και κλειδιά υπολογίζονται με κατανεμημένο τρόπο για το πρωτόκολλο κρυφών υπογραφών υπό συνθήκη *CBS* καθώς και το κρυπτογραφικό πρωτόκολλο που θα χρησιμοποιηθεί το *M - ElGamal*. Η ομάδα είναι ίδια και για τα δύο σχήματα.

- Καθορίζεται ομάδα \mathbb{G} με τάξη πρώτο αριθμό q όπου η *DDH* υπόθεση ισχύει.
- Καθορίζονται οι παράμετροι για το *CBS*. Γεννήτορες $g_1, g_2, v \leftarrow_R \mathbb{G}$ επιλέγονται. Το ζεύγος (s, k) με $s \leftarrow_R \mathbb{Z}_q$ είναι το ιδιωτικό κλειδί και το $k = g_1^s$ είναι το δημόσιο.
- Παράμετροι για το *M - ElGamal* καθορίζονται. $h_1, h_2 \leftarrow_R \mathbb{G}$ καθώς και (z, h) όπου $z \leftarrow_R \mathbb{Z}_q$ είναι το ιδιωτικό κλειδί και $h = h_1^z$ το δημόσιο.

- Δυο συναρτήσεις $hash, \mathcal{H}_1 : \{0, 1\}^* \rightarrow \mathbb{G}, \mathcal{H}_2 : \{0, 1\}^* \rightarrow \mathbb{Z}_q$ οι οποίες μοντελοποιούνται από δυο μαντεία.

6.7.2 Εγγραφή Ψηφοφόρων.

Οι αρχές RA δημιουργούν με καταναμημένο τρόπο διαπιστευτήρια σ_i τα οποία και διανέμουν στους ψηφοφόρους μέσω ασφαλούς καναλιού. Οι ψηφοφόροι μπορούν να χρησιμοποιήσουν τα διαπιστευτήρια σε διαδοχικές εκλογικές αναμετρήσεις. Τα διαπιστευτήρια είναι τυχαία στοιχεία της ομάδας όπως και στο JCJ . Έγκυρα διαπιστευτήρια παράγονται μόνο από τις αρχές ενώ κάθε άλλο τυχαίο στοιχείο μπορεί να προσομοιώσει ένα διαπιστευτήριο και να χρησιμοποιηθεί από τον ψηφοφόρο για να αποφύγει πιθανό εκβιασμό.

6.7.3 Pre-Election.

- Οι αρχές RA δημιουργούν και δημοσιεύουν λίστα με τα ID 's και τα κρυπτογραφημένα διαπιστευτήρια των ψηφοφόρων $VR = \{ID_i, Enc_h(\sigma_i)\}_{i=1}^n$. Στον ψηφοφόρο μπορεί να δοθεί απόδειξη ότι $Enc_h(\sigma_i)$ είναι όντως η κρυπτογράφιση του διαπιστευτηρίου του σ_i .
- Για κάθε υποψήφιο επιλέγεται τυχαίο στοιχείο v_i και δημοσιεύεται το Candidate Slate C .

6.7.4 Εξουσιοδότηση.

Στην αρχή κάθε εκλογικής διαδικασίας μέσω του Bulletin Board.

- Ο ψηφοφόρος υπολογίζει $Enc_h(\sigma'_i)$ και $NIZK$ απόδειξη PoK_1 που αποδεικνύει ότι γνωρίζει το σ'_i . Δημοσιεύει $\langle ID_i, Enc_h(\sigma'_i), PoK_1 \rangle$ το οποίο θα καλούμε στο εξής αίτημα.
- Μόλις ο προκαθορισμένος χρόνος περάσει, οι αρχές TA σημειώνουν τα διπλά ψηφοδέλτια με ελέγχους pet επί των ψηφοδελτίων που αντιστοιχούν στα ίδια ID . Βάσει προκαθορισμένης πολιτικής τα διπλά αιτήματα απορρίπτονται. Ψηφοδέλτια με μη έγκυρες αποδείξεις αγνοούνται. Ο ψηφοφόρος λαμβάνει το $x = \mathcal{H}_1(ID_i, j)$ όπου j είναι γραμμή του BB του αιτήματος.
- Ο ψηφοφόρος υπολογίζει $Enc_h(vote_i)$.
- Ξεκινά η εφαρμογή του CBS . Ο ψηφοφόρος υπολογίζει το x^* που προκύπτει από το $blinding$ του x . Το μήνυμα e υπολογίζεται από το $e^* = \mathcal{H}_2(Enc_h(vote_i), x^*)$ και δημοσιεύεται.
- Για κάθε είσοδο στον BB οι αρχές TA υπολογίζουν τις τυφλές υπογραφές υπό συνθήκη:

$$(bsig_1, bsig_2) \leftarrow auth(Enc_h^{VR}(\sigma_i), Enc_h(\sigma'_i), x, e, \langle s, z \rangle, \langle k, h \rangle, params)$$

- Ο ψηφοφόρος υπολογίζει την τελική υπογραφή υπολογίζοντας $sig_1 \leftarrow bsig_1 \cdot Enc_h(k^{u_1})$ και $sig_2 \leftarrow bsig_2 + u_2$.

Authorization Phase:

Singer

$$x = \mathcal{H}_1(ID_i, j)$$

Recipient

$$\xrightarrow{x}$$

$$\begin{aligned} u_1, u_2, d &\leftarrow_R \mathbb{Z}_q \\ x^* &\leftarrow x g_1^{u_1} g_2^{u_2} v^d \\ e^* &\leftarrow \mathcal{H}_2(Enc_h(vote_i), x^*) \\ e &\leftarrow e^* - d \end{aligned}$$

$$\xleftarrow{e}$$

$$\begin{aligned} y_1 &\leftarrow r_1 + es_1 \\ y_2 &\leftarrow r_2 + es_2 \\ bsig_1 &\leftarrow Enc_h(k^{y_1}) \text{ (if } b = 1 \text{ else } bsig_2 \leftarrow_R \mathbb{G}) \\ bsig_2 &\leftarrow y_2 \end{aligned}$$

$$\xrightarrow{(bsig_1, bsig_2)}$$

$$\begin{aligned} sig_1 &\leftarrow bsig_1 \cdot Enc_h(k^{u_1}) \\ sig_2 &\leftarrow bsig_2 + u_2 \end{aligned}$$

Output: (x^*, e^*, sig_1, sig_2)

6.7.5 Voting.

Κάθε ψηφοφόρος δημοσιεύει σε μέσω ανώνυμου καναλιού

$$\langle Enc_h(vote_i), \langle x^*, e^*, sig_1, sig_2 \rangle, PoK_2, PoK_3 \rangle$$

όπου PoK_2 απόδειξη γνώσης της ψήφου $vote_i$ και PoK_3 απόδειξη γνώσης ότι η $vote_i$ ανήκει στο Candidate Slate.

6.7.6 Tallying.

Οι αρχές TA ξεκινούν την καταμέτρηση των ψηφοδελτίων.

- Πολλαπλά ψηφοδέλτια του ίδιου ψηφοφόρου καθώς και ψηφοδέλτια με μη έγκυρες αποδείξεις αγνοούνται.
- Υπολογίζεται το $\mathcal{H}_2(Enc_h(\text{vote}_i), x^*)$ και ελέγχεται αν είναι ίσο με το e^* . Για κάθε ψηφοδέλτιο υπολογίζονται τα $\text{validity} = x^* g_2^{-\text{sig}_2} v^{-e^*}$ και $Enc_h(\text{validity})$.
- Υπολογίζεται $Enc_h(\text{validity})^s = Enc_h(\text{validity}^s)$.
- Οι τιμές $\langle Enc_h(\text{vote}_i), Enc_h(\text{validity}^s), \text{sig}_2 \rangle$ δίνονται σε `mixnet`.
- Πραγματοποιούνται έλεγχοι *pet* μεταξύ των $Enc_h(\text{validity}^s), \text{sig}_1$. Αν το αποτέλεσμα είναι 1 η ψήφος αποκρυπτογραφείται και προσμετρώνται. Συγκεκριμένα η ψήφος προσμετρώνται αν και μόνο αν

$$PET(Enc_h(\text{validity}^s), \text{sig}_1) = 1 \text{ δηλαδή αν } x^{*s} = Dec_z(\text{sig}_1) \cdot g_2^{\text{sig}_2^s} \cdot v^{e^*s}.$$

6.7.7 Threshold Protocol for Authorisation.

Οι υπογραφές υπολογίζονται από r οντότητες με το πολύ $t - 1$ διεφθαρμένες. Κάθε μέλος των αρχών TA έχει μερίδιο s_i ενός ιδιωτικού κλειδιού s που προκύπτει από την χρήση κάποιου Secret Sharing πρωτοκόλλου όπως στο [15]. Η εξουσιοδότηση γίνεται ως εξής:

- Είσοδος κάθε μέλους είναι η δυάδα $(Enc_h^{VR}(\sigma), Enc_h(\sigma'), x)$.
- Συμφωνούν με καταναμημένο τρόπο σε τυχαίο $bsig_2 = y_2$.
- Κάθε μέλος υπολογίζει $Enc_h(xg_2^{-y_2}v^{-e})$ και παρέχει απόδειξη ορθού υπολογισμού.
- Υπολογίζουν $(Enc_h^{VR}(\sigma)/Enc_h(\sigma'))^{z_i}$ για τυχαίο z_i και παρέχουν απόδειξη ορθού υπολογισμού.
- Οι τιμές που προέκυψαν πολλαπλασιάζονται και δίνουν $c_i = Enc_h(xg_2^{-y_2}v^{-e})$. Η αποκρυπτογράφηση του c_i συμβολίζεται με m_i .
- Κάθε μέλος υπολογίζει το $c_i^{s_i} = Enc_h(m_i)^{s_i}$ και παρέχει απόδειξη ορθού υπολογισμού.
- Από υποσύνολο T με t σωστά υπολογισμένα $Enc_h(m_i)$ έχουμε $bsig_1 = \prod_{i \in T} Enc_h(m_i)^{s_i \lambda_i(0)}$ με λ_i το πολυώνυμο *Lagrange*.

Ός προς την ορθότητα αν τα αρχικά κρυπτοκείμενα είναι διαφορετικά τότε $(Enc_h^{VR}(\sigma)/Enc_h(\sigma'))^{z_i}$ είναι ομοιόμορφα καταναμημένα για τυχαίο z_i . Αν τα αρχικά κείμενα είναι ίσα τότε $Enc_h^{VR}(\sigma)/Enc_h(\sigma') = Enc_h(1)$ ώστε για κάθε i : $Enc_h(m_i) = Enc_h(m) = Enc_h(xg_2^{-y_2}v^{-e})$ ώστε

$$bsig_1 = \prod_{i \in T} Enc_h(m_i)^{s_i \lambda_i(0)} = \prod_{i \in T} Enc_h(m^{s_i \lambda_i(0)}) = Enc_h(\prod_{i \in T} m^{s_i \lambda_i(0)}) = Enc_h(m^{\sum_{i \in T} s_i \lambda_i(0)}) = Enc_h(m^s) = Enc_h((xg_2^{-y_2}v^{-e})^s).$$

6.7.8 Πολυπλοκότητα.

Αν εξαιρεθεί η διαγραφή των διπλών ψηφοδελτίων η πολυπλοκότητα είναι γραμμική στο πλήθος των ψήφων. Αν τώρα $|ID_i|$ είναι το πλήθος των ψήφων που έχουν υποβληθεί με το ID_i και $m = \max_i |ID_i|$ η πολυπλοκότητα είναι $\mathcal{O}(m^2n)$. Η πολυπλοκότητα μπορεί περαιτέρω να μειωθεί σε $\mathcal{O}(mn)$ με κάποια μέθοδο όπως στο [16]. Σε κάθε περίπτωση αν $m = \mathcal{O}(1)$ το πλήθος των υπολογισμών είναι γραμμικό στο πλήθος των ψηφοφόρων.

6.8 Ασφάλεια.

6.8.1 Verifiability.

Ο ψηφοφόρος μπορεί να επαληθεύσει ότι η επικοινωνία του με το εκλογικό σχήμα αποτυπώθηκε ορθά [17],[18]. Πρέπει οι ακόλουθες διεργασίες των αρχών TA να είναι επαληθεύσιμες:

1. Οι αρχές αναγνωρίζουν ως έγκυρα μόνο τα ψηφοδέλτια που προέρχονται από έγκυρα διαπιστευτήρια.
2. Στην καταμέτρηση προσμετρούν μόνο τα έγκυρα ψηφοδέλτια.

Η αρχιτεκτονική του πρωτοκόλλου ανάγει την ορθότητα των διεργασιών στην εγκυρότητα της υπογραφής που παράγεται από το CBS .

Οι αρχές πιθανόν να είναι οντότητες με αντικρουόμενα συμφέροντα. Οι υπογραφές τότε παράγονται με καταναμημένο τρόπο ώστε αν πλειοψηφία τους δρα έντιμα το πρωτόκολλο να παρέχει επαληθευσσιμότητα ως προς τις αρχές.

6.8.2 Eligibility.

Για την καταμέτρηση της ψήφου χρειάζεται μία έγκυρη υπογραφή. Αν ο αντίπαλος περιοριστεί σε πολύ-λογαριθμικό το πλήθος υπογραφές η ασφάλεια του πρωτοκόλλου βασίζεται στην ιδιότητα Strong One More Forgery.

6.8.3 Everlasting Privacy.

Όπως και στο [19]. Κατά την διαδικασία εξουσιοδότησης του χρήστη η perfect blindness ιδιότητα της υπογραφής εξασφαλίζει ότι δεν διαρρέει καμία πληροφορία για την ταυτότητα του χρήστη. Η δημόσια πληροφορία που υπάρχει στον Bulletin Board έχει δημοσιευτεί μέσω ανώνυμου καναλιού και δεν περιέχει καμία πληροφορία για την ταυτότητα του χρήστη. Η κρυπτογραφημένη ψήφος καθώς και οι υπογραφές δεν μπορούν να συσχετιστούν. Συνεπώς, ο αντίπαλος δεν μπορεί να σχετίσει ψηφοφόρο και ψήφο.

6.8.4 Coercion Resistance.

Αποδεικνύουμε την ασφάλεια του πρωτοκόλλου με ανάλογες τεχνικές όπως αυτές που χρησιμοποιήθηκαν και στο JCJ . Ελαφρά τροποποιούμε τα $c - resist$ και $c - resist - ideal$ ώστε να συμπεριλάβουμε και την φάση εξουσιοδότησης του ψηφοφόρου. Η συνάρτηση $auth$ παρέχει στον

ψηφοφόρο έγκυρη ή άκυρη υπογραφή ανάλογα με την είσοδο.

Ψηφοφόρος που εκβιάζεται μπορεί να παρέχει στον αντίπαλο τυχαίο στοιχείο της ομάδας \mathcal{G} ως διαπιστευτήριο. Αν λάβει ψευδές διαπιστευτήριο θα λάβει και μη έγκυρη υπογραφή από το πρωτόκολλο CBS . Ο αντίπαλος \mathcal{A} δεν μπορεί να διακρίνει μεταξύ μιας έγκυρης και μιας μη έγκυρης υπογραφής. Ο ψηφοφόρος θα υποβάλει αίτημα εξουσιοδότησης καθώς και την ψήφο του σε χρονική στιγμή που δεν θα επιτηρείται.

Ο αντίπαλος υποβάλλει αίτημα εξουσιοδότησης χωρίς να είναι σε θέση να επιβεβαιώσει την εγκυρότητα της υπογραφής. Το `mixnet` αποκρύπτει την πορεία του ψηφοδελτίου του.

Η μόνη αλλαγή από το $c - resist$ παιχνίδι του πρωτοκόλλου JCJ είναι η διαδικασία εξουσιοδότησης του ψηφοφόρου. Σε αυτή την φάση οι ψηφοφόροι λαμβάνουν έγκυρη ή άκυρη υπογραφή ανάλογα με την εγκυρότητα του διαπιστευτηρίου που καταθέτουν. Τα μηνύματα ανταλλάσσονται μέσω του `Bulletin Board` και συνεπώς είναι γνωστά τον αντίπαλο.

Όπου λ παράμετρος που καθορίζει την ασφάλεια του πρωτοκόλλου, n το πλήθος των ψηφοφόρων και n_V το πλήθος των ψηφοφόρων που ο αντίπαλος μπορεί να διαφθείρει.

Algorithm 4: *c – resist*

Input: n, n_V, C, D, λ

Output: $\text{result} \in \{0, 1\}$

% Ο Α επιλέγει το σύνολο των ψηφοφόρων που θα διαφθείρει

1. $(V, U) \leftarrow \mathcal{A}(\text{corrupt})$

% Δίνονται διαπιστευτήρια. Ο Α λαμβάνει τα διαπιστευτήρια των διεφθαρμένων ψηφοφόρων.

2. $\{(sk_i, pk_i) \leftarrow \text{reg}(sk_{\mathcal{R}}, ID_i, \lambda)\}_{i \in [n]}$

% Ο Α επιλέγει τον ψηφοφόρο που θα εκδιώσει καθώς και την ψήφο που θα υποβάλει.

3. $(j, \beta) \leftarrow \mathcal{A}(\{sk_i\}_{i \in V}, \text{Coerce})$

% Ο Α ελέγχει αν επέλεξε σωστά.

4. if $\beta \notin C$ **or** $j \notin U$ **then**

output 0

end

% Ρίχνεται νόμισμα.

5. $b \leftarrow_R \{0, 1\}$

% Ο ψηφοφόρος ανυδρά στον εκβιασμό και παραδίδει στον αντίπαλο ψευδές διαπιστευτήριο.

6. if $b = 0$ **then**

$sk^* \leftarrow \text{fakekey}(pk_T, sk_j, pk_j)$

$\mathcal{BB} \leftarrow \text{auth}(sk_j, pk_j, sk_T, pk_T, C, \beta, \lambda)$

% Ο ψηφοφόρος υποκύπτει στον εκβιασμό και παραδίδει στον αντίπαλο το διαπιστευτήριο του.

7. else

$sk^* \leftarrow sk_j$

end

% Ψηφίζουν όλοι οι έντιμοι ψηφοφόροι. Εκτός από τον εκβιαζόμενο. **8.** $\{\mathcal{BB} \leftarrow \text{auth}(sk_i, pk_i, sk_T, pk_T, C, D, \lambda)\}_{i \in V \cup U}$

% Υποβάλει ο Α ψηφοδέλτια για τους διεφθαρμένους ψηφοφόρους και τον εκβιαζόμενο.

9. $\mathcal{BB} \leftarrow \mathcal{A}^{\text{auth}(\cdot)}(\{sk_i\}_{i \in V}, sk^*, pk_T, C, \mathcal{BB})$

% Εκδίδεται το αποτέλεσμα.

10. $(X, P) \leftarrow \text{tally}(sk_T, \mathcal{BB}, C, \{pk_i\}_{i \in V \cup U}, \lambda)$

% Ο Α μαντεύει.

11. $b' \leftarrow \mathcal{A}(X, P, \mathcal{BB}, \text{Guess})$

12. output $b == b'$

Στο παιχνίδι *c – resist* ο αντίπαλος διαφθείρει μια ομάδα V των ψηφοφόρων και αποκτά τα διαπιστευτήρια τους μετά την εγγραφή τους. Επιλέγει τον ψηφοφόρο προς εκβιασμό καθώς και την ψήφο που θα υποβάλλει. Ακολουθεί ρίψη νομίσματος.

- Αν το αποτέλεσμα της ρίψης είναι $b = 0$, ο ψηφοφόρος αντιδρά στον εκβιασμό. Παρέχει στον εκβιαστή ψευδές διαπιστευτήριο και υποβάλλει ψήφο με το αληθινό.
- Αν το αποτέλεσμα της ρίψης είναι $b = 1$, ο ψηφοφόρος υποκύπτει και παραδίδει στον αντίπαλο το αληθινό διαπιστευτήριο.

Οι ειλικρινείς ψηφοφόροι συμμετέχουν στην φάση της εξουσιοδότησης με τα διαπιστευτήρια τους. Ο αντίπαλος A συμμετέχει στην φάση της εξουσιοδότησης με τα διαπιστευτήρια των ψηφοφόρων που ελέγχει και το διαπιστευτήριο που του παρείχε ο εκβιαζόμενος ψηφοφόρος. Ακολουθεί η καταμέτρηση των ψήφων και ο αντίπαλος καλείται να μαντέψει το αποτέλεσμα της ρίψης εξετάζοντας τα δεδομένα του Bulletin Board, το τελικό αποτέλεσμα X και τις αποδείξεις μηδενικής γνώσης.

Algorithm 5: *c – resist – ideal*

Input: n, n_V, C, D, λ

Output: $\text{result} \in \{0, 1\}$

% Ο Α επιλέγει το σύνολο των ψηφοφόρων που θα διαφθείρει

1. $(V, U) \leftarrow \mathcal{A}(\text{corrupt})$

% Δίνονται διαπιστευτήρια. Ο Α λαμβάνει τα διαπιστευτήρια των διεφθαρμένων ψηφοφόρων.

2. $\{(sk_i, pk_i) \leftarrow \text{reg}(sk_{\mathcal{R}}, ID_i, \lambda)\}_{i \in [n]}$

% Ο Α επιλέγει τον ψηφοφόρο που θα εκδιάσει καθώς και την ψήφο που θα υποβάλει.

3. $(j, \beta) \leftarrow \mathcal{A}(\text{Coerce})$

% Ο Α ελέγχει αν επέλεξε σωστά.

4. if $\beta \notin C$ **or** $j \notin U$ **then**

output 0

end

% Ρίχνεται νόμισμα.

5. $b \leftarrow_R \{0, 1\}$

% Ο Α αποκτά το διαπιστευτήριο του εκδιαζόμενου ψηφοφόρου.

6. $sk^* \leftarrow sk_j$

% Το πρωτόκολλο συνεχίζεται με την *idauth*.

7. if $b = 0$ **then**

$\mathcal{BB} \leftarrow \text{idauth}(sk_j, pk_j, sk_T, pk_T, C, \beta, \lambda)$

end

$\{\mathcal{BB} \leftarrow \text{idauth}(sk_i, pk_i, sk_T, pk_T, C, D, \lambda)\}_{i \in U \setminus \{j\}}$

$\mathcal{BB} \leftarrow \mathcal{A}^{\text{idauth}(\cdot)}(\{sk_i\}_{i \in V}, sk^*, pk_T, C)$

% Το αποτέλεσμα υπολογίζεται.

8. $(X, P) \leftarrow \text{tally}(sk_T, \mathcal{BB}, C, \{pk_i\}_{i \in V \cup U}, \lambda)$

% Ο Α μαντεύει το αποτέλεσμα.

9. $b' \leftarrow \mathcal{A}(X, P, \Gamma, \text{Guess})$

10. output $b == b'$

Στο παιχνίδι *c – resist – ideal* συμβαίνουν τα ίδια με μικρές διαφορές. Τα κλειδιά που αποκτά

ο \mathcal{A} δεν τον καθοδηγούν στην επιλογή του υποψήφιου προς εκβιασμό ψηφοφόρου. Πάντα του δίνεται το αληθινό διαπιστευτήριο και η συνάρτηση $auth$ αντικαθίσταται από την $idauth$.

Η $idauth$:

- Για κάθε έγκυρο διαπιστευτήριο καταμετρά μόνο μία ψήφο. Δεν υπάρχουν δηλαδή πολλαπλά ψηφοδέλτια.
- Στους έντιμους ψηφοφόρους ανάλογα με το διαπιστευτήριο απαντά με έγκυρη ή μη έγκυρη υπογραφή.
- Αιτήματα του αντίπαλου με χρήση διαπιστευτηρίων ψηφοφόρων απαντώνται όπως συνήθως. Δηλαδή, απαντά με έγκυρη υπογραφή σε κάθε διαπιστευτήριο που αντιστοιχεί σε διεφθαρμένο ψηφοφόρο που ελέγχεται από τον αντίπαλο. Στο αίτημα με το διαπιστευτήριο του εκβιαζόμενου ψηφοφόρου απαντά ανάλογα με τη ρίψη του νομίσματος. Αν $b = 0$ δίνει έγκυρη υπογραφή διαφορετικά δίνει άκυρη. Στο ιδεατό πείραμα θεωρούμε ότι αντίπαλος λαμβάνει το πραγματικό διαπιστευτήριο του εκβιαζόμενου χρήστη.
- Η έξοδος δημοσιεύεται στον Bulletin Board.

Για να μαντέψει το αποτέλεσμα της ρίψης ο \mathcal{A} μπορεί να χρησιμοποιήσει το τελικό αποτέλεσμα X και το πλήθος των άκυρων ψήφων Γ .

Για να αποδείξουμε ότι το πρωτόκολλο είναι Coercion Resistant αρκεί να δείξουμε ότι κάθε PPT αντίπαλος έχει αμελητέα μεγαλύτερη πιθανότητα να κερδίσει το $c - resist$ παιχνίδι από το $c - resist - ideal$ το οποίο και θεωρείται ασφαλές.

1. **Input:** Ο προσομοιωτής \mathcal{S} λαμβάνει ως είσοδο τα στοιχεία g_1, g_2, h_1, h_2 ομάδας \mathcal{G} τάξης q και διάνυσμα w από κατανομή D που αντικατοπτρίζει την τυχαιότητα των ψήφων. Κάθε στοιχείο του w είναι σύνολο από έγκυρα και άκυρα ψηφοδέλτια λαμβάνοντας υπόψη ότι οι ψηφοφόροι υποβάλλουν πολλαπλές φορές ψήφο αν το επιθυμούν. Ο \mathcal{S} προσπαθεί να απαντήσει αν η τετράδα (g_1, g_2, h_1, h_2) είναι DH ή όχι. Δηλαδή αν $\log_{g_1} h_1 = \log_{g_2} h_2$.
2. **Parameter generation:** Αρχικά ο \mathcal{S} παράγει κλειδί $M - ElGamal$ επιλέγοντας τυχαία $x_1, x_2 \in \mathbb{Z}_q$ και υπολογίζοντας $h = g_1^{x_1} g_2^{x_2}$. Το δημόσιο κλειδί είναι (g_1, g_2, h) . Στη συνέχεια δημιουργεί κλειδί υπογραφών για το σχήμα CBS επιλέγοντας $g_3, g_4, y \leftarrow_R \mathbb{G}, s \in \mathbb{Z}_q$ και $k = g_3^s$. Το ιδιωτικό κλειδί είναι το s και το δημόσιο κλειδί είναι το (g_3, g_4, y, k) .
3. **Registration:** Σε κάθε ψηφοφόρο παρέχεται τυχαίο $\sigma \leftarrow_R \mathbb{G}$. Με χρήση του ιδιωτικού κλειδιού ο \mathcal{S} δημοσιεύει την Voter roll. Δημοσιεύεται επίσης το Candidate Slate.
4. **Corruption :** Ο \mathcal{A} διαφθείρει ψηφοφόρους.
5. **Coercion:** Ο \mathcal{A} επιλέγει ψηφοφόρο για να εκβιάσει και την ψήφο που θα υποβάλλει (j, β) .
6. **Coin Flip:** Ο \mathcal{S} επιλέγει $b \leftarrow_R \{0, 1\}$. Αν $b = 0$ στον \mathcal{A} δίνεται τυχαίο στοιχείο $\sigma^* \leftarrow_R \mathbb{G}$, διαφορετικά παραδίδεται το πραγματικό διαπιστευτήριο $\sigma^* \leftarrow \sigma_j$.

7. **Authorization Requests:** Ο \mathcal{S} εκδίδει έγκυρες υπογραφές για του ελικρινείς ψηφοφόρους. Για κάθε στοιχείο στο w εκδίδει $(Enc_h(\sigma_i), ID_i, PoK_1)$ όπου $Enc_h(\sigma_i) = (h_1^{u_i}, h_2^{u_i}, h_1^{u_i x_1} h_2^{u_i x_2} \sigma_i)$ για τυχαίο u_i και απόδειξη PoK_1 . Ο \mathcal{A} ομοίως αιτείται.
8. **Double requests elimination:** Με χρήση του ιδιωτικού κλειδιού x_1, x_2 διπλά αιτήματα που προέρχονται από το ίδιο διαπιστευτήριο διαγράφονται.
9. **Authorization:** Ο \mathcal{S} προσομοιώνει την διαδικασία εξουσιοδότησης με χρήση του κλειδιού που διαθέτει για να υπογράψει. Τα μηνύματα είναι κρυπτογραφημένα ψηφοδέλτια όπως στο w . Ο \mathcal{A} λαμβάνει υπογραφές.
10. **Vote Casting:** Ο \mathcal{S} υποβάλλει ψήφους για τους ελικρινείς ψηφοφόρους και ο \mathcal{A} υποβάλλει για όσους ελέγχει καθώς και για τον ψηφοφόρο που εκδιάζεται.
11. **Tallying:** Ο \mathcal{S} προσομοιώνει την διαδικασία καταμέτρησης.
12. **Guess:** Ο \mathcal{A} μαντεύει το αποτέλεσμα της ρίψης.
13. **Output:** Ο \mathcal{S} επιστρέφει 1 αν $b = b'$.

Ο \mathcal{A} εκτός από τα δεδομένα που παράγονται με τη συμμετοχή του στην φάση της εξουσιοδότησης μπορεί να δει τα κρυπτογραφημένα διαπιστευτήρια με τις αποδείξεις που τα συνοδεύουν καθώς και τις υπογραφές.

Αν στην είσοδο της προσομοίωσης $d = 1$ έχουμε δηλαδή Diffie - Hellman τριάδα τότε η προσομοίωση είναι πρακτικά το πείραμα $Exp_{\mathcal{E}, \mathcal{A}, \mathcal{H}}^{c-resist}$.

Αν υποθέσουμε $g_1 = g, g_2 = g^a, h_1 = g^b, h_2 = g^{ab}$ για κάποιο g κάθε κρυπτοκείμενο θα έχει την ακόλουθη μορφή $(a_{i,1} = h_1^{r_i}, a'_{i,1} = h_2^{r_i}, \beta_{i,1} = h_1^{r_i x_1} h_2^{r_i x_2} m)$.

- $h_1^{r_i} = g^{br_i} = g_1^{br_i}$
- $h_2^{r_i} = g^{abr_i} = g_2^{br_i}$
- $h_1^{r_i x_1} h_2^{r_i x_2} m = g^{br_i x_1} g^{abr_i x_2} m = g_1^{br_i x_1} g_2^{br_i x_2} m = h^{br_i} m.$

Συνεπώς ο αντίπαλος \mathcal{A} θα γνώριζε την μορφή που θα έχουν τα κρυπτοκείμενα. Θα έλεγε κανείς ότι βλέπει τον Bulletin Board.

$$Pr[\mathcal{S} = 1 | d = 1] = Pr[Exp_{\mathcal{E}, \mathcal{A}, \mathcal{H}}^{c-resist}(\mathcal{V}) = 1] = Succ_{\mathcal{E}, \mathcal{A}}^{c-resist}(\mathcal{V})$$

όπου με \mathcal{V} η εικόνα που έχει ο αντίπαλος.

Διαφορετικά αν στην είσοδο της προσομοίωσης η τριάδα δεν είναι Diffie - Hellman (δηλαδή $d = 0$) τότε καμία πληροφορία δεν διαρρέει για της ψήφους των έντιμων ψηφοφόρων. Αν υποθέσουμε ότι $g_1 = g, g_2 = g^a, h_1 = g^b, h_2 = g^c$ για κάποιο τυχαίο $c \in U \mathbb{Z}_q$ το κρυπτοκείμενο $(a_{i,1} = h_1^{r_i}, a'_{i,1} = h_2^{r_i}, \beta_{i,1} = h_1^{r_i x_1} h_2^{r_i x_2} m)$ αποκρύπτει πλήρως το μήνυμα m . Πράγματι,

- $h_1^{r_i} = g^{br_i} = g_1^{br_i}$
- $h_2^{r_i} = g^{cr_i} = g_2^{c' r_i}$

- $h_1^{r_1 x_1} h_2^{r_2 x_2} m = g^{br_1 x_1} g^{cr_2 x_2} m = g_1^{br_1 x_1} g_2^{c' r_2 x_2} m = g_1^{br_1 x_1} g_2^{br_2 x_2} g_2^{c'' r_2 x_2} m.$

Η πιθανότητα η προσομοίωση να επιστρέψει ως αποτέλεσμα 1 είναι ίση με την πιθανότητα ο αντίπαλος να επιστρέψει ως αποτέλεσμα στο πείραμα $Exp_{\mathcal{E}, \mathcal{A}, \mathcal{H}}^{c-resist-ideal}$. Δηλαδή,

$$Pr[\mathcal{S} = 1 | d = 0] = Pr[Exp_{\mathcal{E}, \mathcal{A}, \mathcal{H}}^{c-resist-ideal}(\mathcal{V}) = 1] = Succ_{\mathcal{E}, \mathcal{A}}^{c-resist-ideal}(\mathcal{V})$$

και προκύπτει ότι

$$Adv_{\mathcal{S}}^{ddh} = Pr[\mathcal{S} = 1 | d = 1] - Pr[\mathcal{S} = 1 | d = 0] = Adv_{\mathcal{E}, \mathcal{A}}^{c-resist}$$

Από το decisional Diffie - Hellman η ποσότητα αυτή είναι αμελητέα.

Chapter 7

Adding Everlasting Privacy in JCJ

Στο παρόν κεφάλαιο παρουσιάζεται ένα εκλογικό πρωτόκολλο που παρέχει **Coercion-Resistance** καθώς και **Everlasting Privacy** με την προϋπόθεση ότι πλειοψηφία αρχών δρουν τίμια.

Κορμός του πρωτοκόλλου είναι το γνωστό άρθρο (*JCJ*) [7] των Ari Juels, Dario Catalano και Markus Jakobsson. Αρχική ιδέα ήταν να χρησιμοποιηθεί ένα κρυπτοσύστημα που παρέχει *perfect secrecy* ώστε το εκλογικό πρωτόκολλο να παρέχει *everlasting privacy*. Η αρχική ιδέα αντικαταστάθηκε από μία ίσως ισοδύναμη. Το διαπιστευτήριο κρυπτογραφήθηκε αφού όμως πρώτα πολλαπλασιάστηκε με τυχαίο στοιχείο. Η τυχαιότητα αυτή δεν αποκαλύπτεται ποτέ ώστε τελικά ο έλεγχος γίνεται τυφλά και η ταυτότητα του ψηφοφόρου παραμένει μυστική. Βασική προϋπόθεση πλειοψηφία αρχών να είναι έντιμες και ο αντίπαλος να μην μπορεί να αποθηκεύσει για μελλοντική χρήση πάνω από τη μισή επικοινωνία ψηφοφόρου και αρχών. Το πρωτόκολλο *JCJ* εφοδιάστηκε και με την ιδιότητα *Everlasting Privacy* χωρίς να αυξηθεί η πολυπλοκότητα. Η πολυπλοκότητα του πρωτοκόλλου είναι τετραγωνική στο πλήθος των ψηφοφόρων $\mathcal{O}(n_V^2)$.

Στην παρουσίαση του πρωτοκόλλου θα δώσουμε σημασία στις διαφορές από το *JCJ*. Ο αναγνώστης μπορεί να θυμηθεί τις λεπτομέρειες του *JCJ* στο αντίστοιχο κεφάλαιο.

7.1 Αρχές

Το πρωτόκολλο εκλογών απαρτίζεται από διάφορες οντότητες:

1. **Registrars.** Συμβολίζονται με $\mathcal{R} = \{R_1, R_2, \dots, R_{n_R}\}$ και αποτελούν μία ομάδα αρχών που επωμίζονται την ευθύνη της από κοινού έκδοσης διαπιστευτηρίων στους ψηφοφόρους.
2. **Tallying Authorities.** Συμβολίζονται με $\mathcal{T} = \{T_1, T_2, \dots, T_{n_T}\}$ και αποτελούν μια ομάδα αρχών που επωμίζονται την ευθύνη του ελέγχου των διαπιστευτηρίων και της διαγραφής διπλών καθώς και μη έγκυρων ψηφοδελτίων καθώς και της από κοινού καταμέτρησης των έγκυρων ψηφοδελτίων και έκδοσης του τελικού αποτελέσματος. Το πρωτόκολλο δεν επιτρέπει *write in votes*. Ψηφοδέλτια με μη έγκυρη επιλογή υποψηφίου απορρίπτονται. Διαθέτουν $(PK_{\mathcal{T}}, SK_{\mathcal{T}})$. Το ιδιωτικό κλειδί μοιράζεται από κοινού στους συμμετέχοντες T_i ώστε να απαιτείται πλειοψηφία αρχών για την αποκρυπτογράφηση.

3. **Voters.** Το σύνολο των ψηφοφόρων $\mathcal{V} = \{V_1, V_2, \dots, V_{n_V}\}$ που συμμετέχουν στη συγκεκριμένη εκλογική διαδικασία.

7.1.1 Επιμέρους Στοιχεία.

Ορίζουμε ως **Candidate Slate** μία διατεταγμένη λίστα από n_C δείκτες $\{c_1, c_2, \dots, c_{n_C}\}$ που αντιστοιχούν στους n_C υποψήφιους. Η επιλογή c_j καθορίζεται πλήρως από τον δείκτη j συνεπώς μπορούμε να θεωρήσουμε το Candidate Slate ως $\{1, 2, \dots, n_C\}$ και καθορίζεται πλήρως από τον n_C .

Ορίζουμε ως **Tally** του Candidate Slate \mathcal{C} το διάνυσμα X των n_C θετικών ακέραιων $\{x_1, x_2, \dots, x_{n_C}\}$ ώστε ο x_j να φανερώνει το πλήθος των ψήφων του c_j .

7.2 Δομικοί Λίθοι.

Παρουσιάζουμε συνοπτικά τους δομικούς λίθους που απαρτίζουν το πρωτόκολλο.

7.2.1 Bulletin Board

Όπως συνηθίζεται σε διάφορα εκλογικά πρωτόκολλα υπάρχει πίνακας \mathcal{BB} ο οποίος είναι γνωστός ως Bulletin Board.

7.3 Primitives

Παρατήρηση 1. Έστω $n \in \mathbb{N}$ και $N = \lfloor \frac{n}{2} - \frac{1}{4} \rfloor$. Για $m = \binom{n}{N}$ και σύνολο $\mathcal{A} = \{b_1, b_2, \dots, b_m\}$ υπάρχουν n υποσύνολα A_1, A_2, \dots, A_n του \mathcal{A} ώστε:

- $\cup_{j=1}^{\lfloor \frac{n}{2} - \frac{1}{4} \rfloor} A_{i_j} \subset \mathcal{A}$
- $\cup_{j=1}^k A_{i_j} = \mathcal{A}$ για $k > \lfloor \frac{n}{2} \rfloor$

Απόδειξη. Κατασκευάζουμε m υποσύνολα του \mathcal{A} μεγέθους N , τα $\mathcal{B}_1, \mathcal{B}_2, \dots, \mathcal{B}_m$. Ορίζουμε

$$A_i = \{b_j, j \in \{1, \dots, m\} | b_i \notin \mathcal{B}_j\}$$

- Έστω S οποιαδήποτε συλλογή $A_{i_1}, A_{i_2}, \dots, A_{i_d}$ με $|S| = d \leq N$. Τότε, υπάρχει τουλάχιστον ένα $b_x \in \{b_1, \dots, b_m\}$ τέτοιο ώστε $\{b_{i_1}, b_{i_2}, \dots, b_{i_d}\} \subseteq \mathcal{B}_x$.
 - $b_x \in A_{i_1} \Rightarrow b_{i_1} \notin \mathcal{B}_x$
 - $b_x \in A_{i_2} \Rightarrow b_{i_2} \notin \mathcal{B}_x$
 - \vdots \vdots
 - $b_x \in A_{i_d} \Rightarrow b_{i_d} \notin \mathcal{B}_x$

Συνεπώς, $b_x \notin A_{i_j}$ για $j = 1, \dots, d$ και επομένως η ένωση των A_i που ανήκουν στο S περιέχεται γνήσια στο \mathcal{A} . Επίσης σημαντικό συμπέρασμα το b_x δεν ανήκει σε τουλάχιστον N από τα A_i .

- Έστω S οποιαδήποτε συλλογή A_i με $|S| > N$. Κάθε στοιχείο $b_x \in \{b_1, \dots, b_m\}$ δεν περιέχεται σε ακριβώς N από τα A_i και συνεπώς περιέχεται σε κάποιο από τα $A_i \in S$. Η ένωση των A_i δίνει το \mathcal{A} .

Παρατήρηση 2. Κάθε στοιχείο b_x δεν περιέχεται σε ακριβώς N από τα A_i .

Απόδειξη. Ας υποθέσουμε ότι το b_x δεν περιέχεται στα σύνολα A_{i_1}, \dots, A_{i_N} . Από το ορισμό των συνόλων προκύπτουν τα ακόλουθα:

$$b_x \notin A_{i_1} \iff b_{i_1} \in B_x$$

$$b_x \notin A_{i_2} \iff b_{i_2} \in B_x$$

⋮

$$b_x \notin A_{i_N} \iff b_{i_N} \in B_x$$

Όμως το B_x περιέχει ακριβώς N στοιχεία οπότε το $b_x \in A_{i_{N+1}}, \dots, b_x \in A_{i_n}$ και το b_x ανήκει σε τουλάχιστον $n - N$ σύνολα συνεπώς σε ακριβώς $n - N$.

Παράδειγμα 3. Ας δούμε την περίπτωση $n = 5$. Τότε $m = \binom{5}{2} = 10$ και $\mathcal{A} = \{b_1, b_2, \dots, b_{10}\}$. Κατασκευάζουμε τα σύνολα

$$B_1 = \{b_1, b_2\}, B_2 = \{b_1, b_3\}, B_3 = \{b_1, b_4\}, B_4 = \{b_1, b_5\},$$

$$B_5 = \{b_2, b_3\}, B_6 = \{b_2, b_4\}, B_7 = \{b_2, b_5\},$$

$$B_8 = \{b_3, b_4\}, B_9 = \{b_3, b_5\},$$

$$B_{10} = \{b_4, b_5\}$$

Και ορίζουμε $A_i = \{b_j \in \{b_1, \dots, b_m\} | b_i \notin B_j\}$. Τότε, τα $A_1 = \{b_5, b_6, b_7, b_8, b_9, b_{10}\}$, $A_2 = \{b_2, b_3, b_4, b_8, b_9, b_{10}\}$, $A_3 = \{b_1, b_3, b_4, b_6, b_7, b_{10}\}$, $A_4 = \{b_1, b_2, b_4, b_5, b_7, b_9\}$, $A_5 = \{b_1, b_2, b_3, b_5, b_6, b_8\}$,

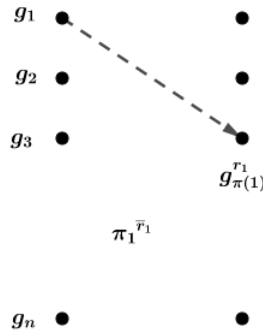
Παρατήρηση 4. Έστω $\beta = \prod_{i=1}^m b_i$ και $\alpha_1, \dots, \alpha_n$ τα γινόμενα των στοιχείων που περιέχονται στα σύνολα A_1, \dots, A_n αντίστοιχα. Τότε, $\prod_{i=1}^n \alpha_i = \beta^{n-N}$.

Απόδειξη. Κάθε στοιχείο $b_i, i = 1, \dots, m$ περιέχεται σε ακριβώς $n - N$ από τα σύνολα A_i . Συνεπώς,

$$\prod_{i=1}^n \alpha_i = \prod_{i=1}^m b_i^{n-m} = (\prod_{i=1}^m b_i)^{n-m} = \beta^{n-N}.$$

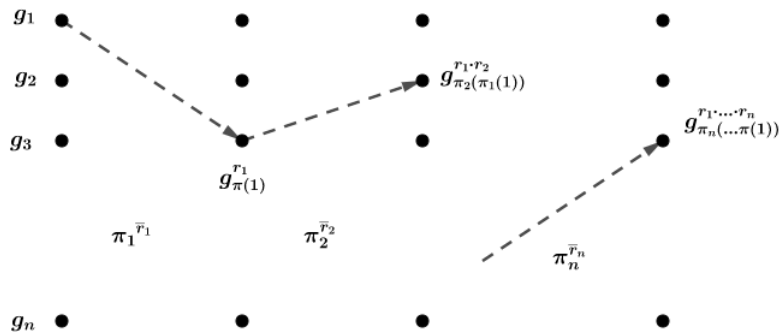
Παράδειγμα 5. Συνεχίζοντας το προηγούμενο παράδειγμα $\alpha_1 = b_5 \cdot b_6 \cdot b_7 \cdot b_8 \cdot b_9 \cdot b_{10}$ κ.ο.κ.

Άσκηση 6. Έστω ομάδα \mathbb{G} με τάξη πρώτο αριθμό p και λίστα $g = \{g_1, g_2, \dots, g_n\}$, $g_i \in \mathbb{G} \wedge g_i \neq 1$ για $i = 1, \dots, n$. Για μετάθεση π και τυχαίο διάνυσμα $\bar{r} \in \mathbb{Z}_p^n$, $r_i \neq p, i = 1, \dots, n$ ορίζουμε ως $\pi_{\bar{r}}(g)$ την λίστα που προκύπτει από την μετάθεση των στοιχείων της g υψωμένων στους τυχαίους εκθέτες που καθορίζονται από το \bar{r} .



Παρατήρηση 7. Με την προϋπόθεση ότι η μετάθεση π και το τυχαίο διάνυσμα \bar{r} είναι μυστικά, input και output είναι ασυσχέτιστα ακόμη κι από computationally unbounded αντίπαλο. Κάθε μη μοναδιαίο στοιχείο της ομάδας έχει τάξη p και είναι γεννήτορας αυτής. Συνεπώς η τυχαιότητα στην οποία υψώνεται αποκρύπτει πλήρως την ταυτότητα του στοιχείου καθώς αν η μετάθεση είναι μυστική όλες οι πιθανές μεταθέσεις και όλοι οι πιθανοί εκθέτες είναι ισοπίθανοι.

Παρατήρηση 8. Θεωρούμε ότι η λίστα δίνεται διαδοχικά σε mix net αυτού του είδους ώστε η τελική μετάθεση να προκύπτει από την σύνθεση των επιμέρους μεταθέσεων και ο τελικός εκθέτης ως γινόμενο των επιμέρους εκθετών. Η συσχέτιση μεταξύ input και output χάνεται με την προϋπόθεση ότι ένα από αυτά τα mix είναι ειλικρινές και δεν διαρρέει την τυχαιότητα και την μετάθεση που χρησιμοποιεί.



Παρατήρηση 9. Για τις ανάγκες του πρωτοκόλλου που θα παρουσιάσουμε θα χρειαστούμε n το πλήθος mix-nets αυτού του είδους που θα λειτουργούν παράλληλα χρησιμοποιώντας κάθε φορά την ίδια μετάθεση π και το ίδιο διάνυσμα τυχαιότητας \bar{r} . Σε κάθε επιμέρους βήμα παρέχουν απόδειξη μηδενικής γνώσης ότι χρησιμοποιήθηκε η ίδια μετάθεση και το ίδιο διάνυσμα.

Παρατήρηση 10. Μαζί με το ειδικό mix net που παρουσιάσαμε σε διάφορα σημεία γίνεται χρήση mix nets που παραδοσιακά μεταδέτουν και κρυπτογραφούν. Η χρήση τους αποσκοπεί κυρίως στην εξασφάλιση του Coercion Resistance καθώς απέναντι σε computationally unbounded αντίπαλο δεν προσφέρουν καμία ασφάλεια.

Παρατήρηση 11. Υποθέτουμε ότι είναι αδύνατη η συγκέντρωση και αποθήκευση για μεμβλοντική χρήση όβων των μηνυμάτων που θα σταθούν από τον ψηφοφόρο.

7.4 Voting

Σε ότι ακολουθεί με x συμβολίζουμε την κρυπτογράφηση του στοιχείου με το κρυπτοσύστημα δημοσίου κλειδιού, $x = E_h(x)$.

1. Διαμοιρασμός β - επιλογή τυχειότητας.

1. Ο ψηφοφόρος v_i επιλέγει τυχειά m το πλήθος στοιχεία της \mathbb{G}_1 και κατασκευάζει το σύνολο $\mathcal{A}_i = \{b_{i1}, b_{i2}, \dots, b_{im}\}$. Θέτει $\beta_i = \prod_{j=1}^m b_{ij}$ και κατασκευάζει n υποσύνολα $A_{i1}, A_{i2}, \dots, A_{in}$ του \mathcal{A}_i τέτοια ώστε:

- Η ένωση οποιασδήποτε μειοψηφίας συνόλων A_{ij} να περιέχεται γνήσια στο \mathcal{A} .

$$\cup_{j=1}^{\lfloor \frac{n-1}{4} \rfloor} A_{il_j} \subset \mathcal{A}_i$$

- Η ένωση οποιασδήποτε πλειοψηφίας $k > \lfloor \frac{n}{2} \rfloor$ συνόλων A_{ij} να είναι ίση με το \mathcal{A}_i

$$\cup_{j=1}^k A_{il_j} = \mathcal{A}_i$$

2. Στην επιλογή τυχειότητας δεν επιτρέπεται η αντίστροφη τιμή του διαπιστευτηρίου. Επισυνάπτουν απόδειξη μηδενικής γνώσης ότι $\sigma_i \cdot \beta_i \neq 1$.
3. Κρυπτογραφεί τα περιεχόμενα κάθε $A_{ij}, j = 1, \dots, n$, κατασκευάζοντας σύνολα $H_j^i = E_h(A_{ij})$ για $j = 1, \dots, n$.

2. Υποβολή ψηφοδελτίου

Το ψηφοδέλτιο αποτελείται από τα:

1. Κατάλληλα επιλεγμένη ταυτότητα μηνύματος r_i ώστε να αποφεύγονται συμπτώσεις και απόδειξη μηδενικής γνώσης PoK του r_i .
2. $E_1^{(i)} = (g_1^r, g_2^r, \sigma_i \beta_i h^r)$ όπου σ_i το διαπιστευτήριο του ψηφοφόρου και β_i η τυχειότητα που το αποκρύπτει.
3. n -άδα $(H_j^i, E_h(r_i))$ για $j = 1, \dots, n$. Κάθε αρχή έχει προσωπικό χώρο κομμάτι του οποίου είναι δημόσιο.

$$T_j = \left[\begin{array}{ccc} \vdots & & \vdots \\ E_h(r_i), PoK(r_i) & j_{i1}, \dots, j_{i\lambda} & b_{j_{i\lambda}} \in H_j^i \\ \vdots & & \vdots \end{array} \right] \parallel \left[\begin{array}{c} \vdots \\ H_j^i \\ \vdots \end{array} \right]$$

Οι δύο πρώτες στήλες του πίνακα είναι δημόσιες ενώ η τρίτη ιδιωτική. Στην δεύτερη στήλη αναγράφονται οι δείκτες των στοιχείων που συνθέτουν την τυχαιότητα για τον κάθε υποψήφιο.

4. $E_2^{(i)} = (g_1^{r'}, g_2^{r'}, c_j h^{r'})$ όπου c_j όπως και στο JCJ η επιλογή του υποψηφίου.

Ο ψηφοφόρος περιλαμβάνει επίσης στο ηλεκτρονικό ψηφοδέλτιο κάθε απόδειξη μηδενικής γνώσης που υπάρχει και στο JCJ με εξαίρεση μία, αντί να αποδείξει με μηδενική γνώση ότι γνωρίζει το σ_i αποδεικνύει με μηδενική γνώση ότι γνωρίζει το $\beta\sigma_i$. Οι αποδείξεις αυτές συνθέτουν το σύνολο Pf .

Ο ψηφοφόρος δημοσιεύει στον \mathcal{BB} το

$$B_i = (E_1^{(i)}, E_2^{(i)}, Pf, E_h(r_i))$$

μέσω ανώνυμου καναλιού.

$$\mathcal{BB} = \left[\begin{array}{ccc|c} \vdots & \vdots & \vdots & \vdots \\ E_1^{(i)} & E_2^{(i)} & Pf & E_h(r_i) \\ \vdots & \vdots & \vdots & \vdots \end{array} \right]$$

Χρησιμοποιώντας διαφορετικά κανάλια στέλνει στην αρχή T_1 το $(H_1^i, E_h(r_i))$, στην αρχή T_2 το $(H_2^i, E_h(r_i))$ κ.ο.κ. Από την κατασκευή των συνόλων απαιτείται πλειοψηφία αρχών για απόκτηση της τυχαιότητας β . Ακόμη κι αν μειοψηφία αρχών δημοσιεύσουν τμήμα της πληροφορίας που διαθέτουν, η τυχαιότητα δεν διαρρέει.

7.5 Credential Checking

1. Οι αρχές \mathcal{T} ελέγχουν τις αποδείξεις μηδενικής γνώσης (3η στήλη του πίνακα) και διαγράφουν όσες ψήφους δεν συνοδεύονται από έγκυρες αποδείξεις.

2. Η σύνθεση της τυχαιότητας β_i από τις αρχές γίνεται με ελέγχους PET στην ταυτότητα μηνύματος.

3. Διαγραφή πολλαπλών ψηφοδελτίων.

Αν $L = \{\sigma_1\beta_1, \sigma_2\beta_2, \dots, \sigma_n\beta_n\}$ η λίστα με τα κρυπτογραφημένα διαπιστευτήρια και τις αντίστοιχες τυχαιότητες των ψηφοφόρων που παράγεται από την πρώτη στήλη του Bulletin Board. Οι αρχές για να επιτύχουν διαγραφή πολλαπλών ψηφοδελτίων:

- Επιλέγουν το πρώτο στη σειρά διαπιστευτήριο $\sigma_1\beta_1$ και κατασκευάζουν λίστα $M_1 = \{\sigma_2\beta_2, \dots, \sigma_n\beta_n\}$.
- Πολλαπλασιάζουν τα στοιχεία της λίστας M_1 με το $(\sigma_1\beta_1)^{-1}$ ώστε $M_2 = \{\sigma_2\beta_2 \cdot \sigma_1^{-1}\beta_1^{-1}, \dots, \sigma_n\beta_n \cdot \sigma_1^{-1}\beta_1^{-1}\} = \{(\sigma_2 \cdot \sigma_1^{-1})(\beta_2 \cdot \beta_1^{-1}), \dots, (\sigma_n \cdot \sigma_1^{-1})(\beta_n \cdot \beta_1^{-1})\}$.
- Έστω $\alpha_{11}, \dots, \alpha_{1n}$ τα γινόμενα των στοιχείων που έχουν οι αρχές T_1, \dots, T_n αντίστοιχα και αφορούν τον ψηφοφόρο με διαπιστευτήριο σ_1 . Ομοίως, $\alpha_{21}, \dots, \alpha_{2n}$ τα γινόμενα των στοιχείων που έχουν οι αρχές T_1, \dots, T_n αντίστοιχα και αφορούν τον ψηφοφόρο με διαπιστευτήριο σ_2 .

- Κάθε αρχή T_i ιδιωτικά υπολογίζει

$$M_{2i} = M_2 \cdot (\alpha_{1i} \cdot \alpha_{2i}^{-1})^{\frac{n}{n-N}}$$

- Οι λίστες δίνονται σε n Mix-Nets που λειτουργούν παράλληλα χρησιμοποιώντας σε κάθε βήμα την ίδια μετάθεση π και το ίδιο διάνυσμα τυχαιότητας \bar{r} παρέχοντας αποδείξεις μηδενικής γνώσης για την ορθή λειτουργία τους. Στην έξοδο κάθε αρχή λαμβάνει,

$$\bar{\pi}^{\bar{r}}(M_{2i}) = \bar{\pi}(\{(\sigma_2 \cdot \sigma_1^{-1})^{r_1} (\beta_2 \cdot \beta_1^{-1})^{r_1} (\alpha_{1i} \cdot \alpha_{2i}^{-1})^{\frac{nr_1}{n-N}}, \dots, (\sigma_n \cdot \sigma_1^{-1})^{r_n} (\beta_n \cdot \beta_1^{-1})^{r_n} (\alpha_{1i} \cdot \alpha_{2i}^{-1})^{\frac{nr_n}{n-N}}\}).$$

- Από κοινού υπολογίζουν το γινόμενο:

$$\prod_{i=1}^n \bar{\pi}^{\bar{r}}(M_{2i}) = \bar{\pi}(\{(\sigma_2 \cdot \sigma_1^{-1})^{nr_1} (\beta_2 \cdot \beta_1^{-1})^{nr_1} \prod_{i=1}^n (\alpha_{1i} \cdot \alpha_{2i}^{-1})^{\frac{nr_1}{n-N}}, \dots, (\sigma_n \cdot \sigma_1^{-1})^{nr_n} (\beta_n \cdot \beta_1^{-1})^{nr_n} \prod_{i=1}^n (\alpha_{1i} \cdot \alpha_{2i}^{-1})^{\frac{nr_n}{n-N}}\}).$$

$$\prod_{i=1}^n \bar{\pi}^{\bar{r}}(M_{2i}) = \bar{\pi}(\{(\sigma_2 \cdot \sigma_1^{-1})^{nr_1} (\beta_2 \cdot \beta_1^{-1})^{nr_1} (\beta_1^{n-N} \cdot \beta_2^{-1(n-N)})^{\frac{nr_1}{n-N}}, \dots, (\sigma_n \cdot \sigma_1^{-1})^{nr_n} (\beta_n \cdot \beta_1^{-1})^{nr_n} (\beta_1^{n-N} \cdot \beta_2^{-1(n-N)})^{\frac{nr_n}{n-N}}\}).$$

$$\prod_{i=1}^n \bar{\pi}^{\bar{r}}(M_{2i}) = \bar{\pi}(\{(\sigma_2 \cdot \sigma_1^{-1})^{nr_1}, \dots, (\sigma_n \cdot \sigma_1^{-1})^{nr_n} (\beta_n)^{nr_n} (\beta_2)^{-nr_n}\}).$$

Αν $\sigma_1 = \sigma_2$ υπάρχει μονάδα σε κάποια συντεταγμένη συνεπώς διπλοσηφία και το ψηφοδέλτιο διαγράφεται. Η πιθανότητα να διαγραφεί ψηφοδέλτιο που αντιστοιχεί σε διαφορετικό διαπιστευτήριο θεωρούμε ότι είναι μικρή για κατάλληλη επιλογή της τάξης της ομάδας.

$$(\sigma_n \cdot \beta_n)^{nr_n} (\beta_2 \cdot \sigma_1^{-1})^{nr_n} = 1.$$

$$(\sigma_n \cdot \beta_n \cdot \beta_2 \cdot \sigma_1^{-1})^{nr_n} = 1.$$

$$\sigma_n \cdot \beta_n \cdot \beta_2 \cdot \sigma_1^{-1} = 1.$$

Δυστυχώς ο έλεγχος πρέπει να επαναληφθεί για τα υπόλοιπα στοιχεία της λίστας ώστε τελικά η πολυπλοκότητα να γίνει κυβική.

Παρατήρηση 12. Δεδομένου ότι ελέγχουμε μόνο κατά ζεύγη τα στοιχεία και επιθυμούμε να εξασφαλίσουμε ότι τα επόμενα είναι διαφορετικά από τη μονάδα η λίστα M_1 μπορεί να αντικατασταθεί από την $M'_1 = \{\sigma_2 \beta_2, l_1, l_2, \dots, l_c\}$ για κατάλληλο c (μικρό;) και τυχαία μη μοναδιαία στοιχεία l_1, l_2, \dots, l_c . Επαναλαμβάνοντας δε την διαδικασία για διαφορετικά τυχαία στοιχεία που συμπληρώνουν τη λίστα η πιθανότητα να διαγραφεί ψηφοδέλτιο χωρίς να έχει υποβληθεί άλλο με το ίδιο διαπιστευτήριο είναι αμελητέα. Σε κάθε περίπτωση θεωρώντας το μήκος της λίστα σταθερό η πολυπλοκότητα είναι τετραγωνική και δεν αυξάνει την πολυπλοκότητα του JCS.

4. Έστω M η λίστα που απέμεινε μετά τις διαγραφές.

5. Προς αποφυγή επιθέσεων βασιζόμενων στην χρονική στιγμή υποβολής του ψηφοδελτίου ο πίνακας \mathcal{BB} δίνεται ως είσοδο σε `reencryption mixnet`. (Διεφθαρμένη αρχή ενημερώνει τον αντίπαλο ότι στην π.χ. 345η εγγραφή ο έλεγχος ταυτότητας αποτυγχάνει.) Έστω M_1 η λίστα μετά το `mix - net`. Δεδομένου ότι ο αντίπαλος διαθέτει απεριόριστη υπολογιστική ισχύ το βήμα αυτό είναι αναγκαίο μόνο για το `Coercion Resistance`.

6. Έλεγχος διαπιστευτηρίου.

Για τον έλεγχο της εγκυρότητας των διαπιστευτηρίων οι αρχές λαμβάνουν το πρώτο διαπιστευτήριο από τη λίστα M_1 έστω το $\sigma_1\beta_1$ και

- Πολλαπλασιάζουν τη λίστα $L = \{\sigma_1, \dots, \sigma_n\}$ με τα κρυπτογραφημένα διαπιστευτήρια με το $(\sigma_1 \cdot \beta_1)^{-1}$. Προκύπτει η λίστα

$$L_1 = \{\sigma_1 \cdot (\sigma_1 \cdot \beta_1)^{-1}, \dots, \sigma_n \cdot (\sigma_1 \cdot \beta_1)^{-1}\} = \{(\beta_1)^{-1}, \dots, \sigma_n \cdot (\sigma_1 \cdot \beta_1)^{-1}\}$$

$$\{(\beta_1)^{-1}, \dots, \sigma_n \cdot (\sigma_1)^{-1} \cdot (\beta_1)^{-1}\}$$

- Κάθε αρχή συνεισφέρει το κομμάτι της πληροφορίας που διαθέτει κατασκευάζοντας λίστα

$$L_{1i} = L_1 \cdot \alpha_{1i}^{\frac{n}{n-N}} = \{(\beta_1)^{-1}, \dots, \sigma_n \cdot (\sigma_1)^{-1} \cdot (\beta_1)^{-1}\} \cdot \alpha_{1i}^{\frac{n}{n-N}} =$$

$$\{(\beta_1)^{-1} \alpha_{1i}^{\frac{n}{n-N}}, \dots, \sigma_n \cdot (\sigma_1)^{-1} \cdot (\beta_1)^{-1} \alpha_{1i}^{\frac{n}{n-N}}\}$$

όπου α_{1i} το γινόμενο των στοιχείων που διαθέτει η αρχή T_i .

- Κάθε λίστα δίνεται είσοδος στα n `Mix nets` που λειτουργούν παράλληλα. Όστε οποιαδήποτε πληροφορία για τα αρχικά κρυπτοκείμενα να χαθεί. Στην έξοδο κάθε αρχή λαμβάνει την λίστα

$$\bar{\pi}^r(L_{1i}) = \bar{\pi}(\{(\beta_1)^{-r_1} \alpha_{1i}^{\frac{nr_1}{n-N}}, \dots, \sigma_n^{r_n} \cdot (\sigma_1)^{-r_n} \cdot (\beta_1)^{-r_n} \alpha_{1i}^{\frac{nr_n}{n-N}}\})$$

- Οι αρχές υπολογίζουν από κοινού το γινόμενο των επιμέρους λιστών

$$\prod_{i=1}^n \bar{\pi}^r(L_{1i}) = \bar{\pi}(\{(\beta_1)^{-nr_1} \prod_{i=1}^n \alpha_{1i}^{\frac{nr_1}{n-N}}, \dots, \sigma_n^{nr_n} \cdot (\sigma_1)^{-nr_n} \cdot (\beta_1)^{-nr_n} \prod_{i=1}^n \alpha_{1i}^{\frac{nr_n}{n-N}}\})$$

$$= \bar{\pi}(\{(\beta_1)^{-nr_1} (\beta_1^{n-N})^{\frac{nr_1}{n-N}}, \dots, \sigma_n^{nr_n} \cdot (\sigma_1)^{-nr_n} \cdot (\beta_1)^{-nr_n} (\beta_1^{n-N})^{\frac{nr_n}{n-N}}\})$$

$$= \bar{\pi}(\{1, \dots, (\sigma_n \cdot \sigma_1^{-1})^{nr_n}\})$$

- Οι αρχές αποδέχονται αν και μόνο αν υπάρχει 1 σε κάποια θέση στην τελική λίστα. Παρατηρείστε ότι μονάδα υπάρχει αν και μόνο αν το διαπιστευτήριο αντιστοιχεί σε έγκυρο διαπιστευτήριο της λίστας L .
- Συνεχίζουν μέχρι να εξαντληθεί η λίστα.

Παρατήρηση 13. Ο έλεγχος για διπλοψηφίες καθώς και ο έλεγχος για την εγκυρότητα του διαπιστευτηρίου μπορεί να πραγματοποιηθεί από πλειοψηφία αρχών. Στην περίπτωση αυτή οι αρχές συνεισφέρουν μόνο μέρος της πληροφορίας που διαδέτουν ανάλογα με τους δημόσιους δείκτες της δεύτερης στήλης και βάσει προκαθορισμένης πολιτικής. Για παράδειγμα αν T_1, T_2, \dots, T_{N+1} οι $N + 1$ αρχές που θα συμμετέχουν στον έλεγχο η πρώτη αρχή συνεισφέρει όλα τα στοιχεία, η δεύτερη αρχή από τα στοιχεία που διαδέτει μόνο όσα δεν έχει συνεισφέρει η αρχή T_1 κ.ο.κ.

Παρατήρηση 14. Ένα ακόμα σημείο το οποίο χρειάζεται ιδιαίτερη προσοχή είναι η συμπεριφορά διεφθαρμένης αρχής. Μία διεφθαρμένη αρχή μπορεί να συμμετέχει προσφέροντας στην εκτέλεση του πρωτοκόλλου διαφορετικά στοιχεία από αυτά που παρέλαβε. Σε αυτήν την περίπτωση κάθε έλεγχος θα αποτύχει. Αν και η αρχή δεν μπορεί να λειτουργήσει υπέρ κάποιας παράταξης καθώς δεν γνωρίζει την επιλογή του υποψηφίου (η οποία είναι κρυπτογραφημένη και απαιτείται πλειοψηφία αρχών για αποκρυπτογράφηση) είναι μια πιθανή επίθεση στο πρωτόκολλο. Δεν ξέρω αν είναι δυνατή μία απόδειξη μηδενικής γνώσης ότι συμμετέχει με το τμήμα της πληροφορίας που παρέλαβε ή κάποιου είδους *integrity protection* των στοιχείων που παρέλαβε ώστε να μην μπορεί να τα παραποιήσει ή να τα αντικαταστήσει.

Διαφορετικά, υπολογίζοντας για όλους τους δυνατούς συνδυασμούς αρχών ανά $N + 1$ του ίδιο έλεγχου, εξακριβώνουμε αν τα αποτελέσματα συμπίπτουν. Δεδομένου ότι πλειοψηφία αρχών είναι ειλικρινείς ένα τουλάχιστον αποτέλεσμα είναι έγκυρο. Αν ταυτίζονται αποδεχόμαστε. Ακόμη κι με αυτόν τον μη αποδοτικό τρόπο δεν αθλιάζει η πολυπλοκότητα του πρωτοκόλλου και παραμένει τετραγωνική.

7.6 Tallying

Όπως και στο *JCJ*.

7.7 Coercion Resistance

Όπως και στο *JCJ*.

7.8 Everlasting privacy

Η ταυτότητα του διαπιστευτηρίου δεν αποκαλύπτεται. Η τυχαιότητα που χρησιμοποιεί ο ψηφοφόρος αποκρύπτει την ταυτότητα του ακόμη κι από τις αρχές. Πλειοψηφία αρχών πρέπει να συνεργαστεί ώστε να αποκαλυφθεί το πραγματικό διαπιστευτήριο. Η ύψωση σε δύναμη δε αποσυνδέει κάθε στοιχείο από την αρχική του τιμή ώστε ο έλεγχος να γίνεται τυφλά μεταξύ στοιχείων αναζητώντας κάπου μία μονάδα.

Bibliography

- [1] Moran, T., Naor, M. *Split-Ballot voting: Everlasting Privacy with distributed trust*. ACM Trans. Inf. Syst. Secur. 13(2) (2010).
- [2] Pereira, O., Cuvelier, E., Peters, T. *Election verifiability or vote privacy: Do we need to choose?*. SecVote 2012(2012), <http://secvote.uni.lu/>
- [3] Groth, J. *Short pairing-based non-interactive zero-knowledge arguments* ASI-ACRYPT. pp.321-340 (2010)
- [4] Lipmaa, H., Zhang, B. *A more efficient computationally sound non-interactive zero-knowledge shuffle argument*. In: SCN. pp. 477-502 (2012)
- [5] T. Okamoto *Receipt-free electronic voting schemes for large scale elections*. In B. Christianson et al., editor, Security Protocols Workshop, pages 25-35. Springer-Verlag, 1997. LNCS no. 921
- [6] B. Schoenmakers *Personal Communication*. 2000
- [7] Ari Juels, Dario Catalano, Markus Jakobsson *Coercion-Resistant Electronic Elections*.
- [8] Philipp Locher, Rolf Haenni, Reto E. Koenig *Coercion-Resistant Internet Voting with Everlasting Privacy* International Conference on Financial Cryptography and Data Security FC 2016:Financial Cryptography and Data Security, pp 161-175
- [9] Ueli Maurer *Zero-knowledge Proofs of Knowledge for Group Homomorphisms*. Africacrypt 2009.
- [10] Au, M.H., Susilo, W., Mu, Y.: *Proof of knowledge of representation of committed value and its applications*. ACISP'10, 15th Australian Conference on Information Security and Privacy. pp. 352-369 (2010)
- [11] Panagiotis Grontas, Ariw Pagourtzis, Alexandros Zacharakis. *Efficient coercion resistant and everlasting privacy in remote electronic elections*
- [12] David Chaum. *Blind Signatures for untraceable payments*. In D. Chaum, R.L. Rivest and A.T. Sherman, editors, Advances in Cryptology Proceedings of Crypto 82, pages 199-203, 1983.
- [13] Tatsuaki Okamoto. *Provably secure and practical identification schemes and corresponding signatures schemes*. In Advances in Cryptology - CRYPTO '92, 12th Annual International

Cryptology Conference, Santa Barbara, California, USA, August 16-20, 1992 Proceedings, pages 31-53, 1992.

- [14] Miyako Ohkubo, Fumiaki Miura, Masayuki Abe, Atsusi Fujioka, and Tatsuaki Okamoto. *An improvement on a practical secret voting scheme*. In information Security, volume 1729 of Lecture Notes in Computer Science, pg 225-234. Springer Berlin Heidelberg, 1999.
- [15] Adi Shamir *How to share a secret*. Communications of the ACM,22(11):612-613,1979.
- [16] Stefan G. Weber, Roberto Araujo, and Johannes Buchmann. *On coercion-resistant electronic elections with linear work*. In ARES, pages 908-916. IEEE Computer Society, 2007.
- [17] Ronald Cramer, Rosario Gennaro, and Berry Schoenmakers. *A secure and optimal efficient multi-authority election scheme*. Transactions on Emerging Telecommunications Technologies, 8(5):481-490,1997.
- [18] Ben Adida. *Helios: web-based open audit voting*. In Proceedings of the 17th conference on Security symposium, SS'08, pages 335-348, Berkeley, CA USA, 2008. USENIX Association.
- [19] Atsushi Fujioka, Tatsuaki Okamoto, and Kazuo Ohta. *A practical secret voting scheme for large elections*. In proceedings of the Workshop on the Theory and Application of Cryptographic Techniques: Advances In Cryptology, ASIACRYPT '92,pages 244-251, London, UK, 1992. Springer-Verlag.
- [20] Ronald Cramer, Ivan Damgard, and Berry Schoenmakers. *Proofs of partial knowledge and simplified design of witness hiding protocols*. In CRYPTO 'A94, volume 839 of LNCS, pages 174-187. Springer, 1994.
- [21] Ronald Cramer, Rosario Gennaro, and Berry Schoenmakers. *A secure and optimally efficient multi-authority election scheme*. In Walter Fumy, editor, EUROCRYPT 'A97, volume 1233 of LNCS, pages 103-118, Konstanz, Germany, May 1997. Springer.
- [22] Johannes Buchmann, Denise Demirel, Jeroen van de Graal. *Towards a Publicly-Verifiable Mix-Net Providing Everlasting Privacy*.
- [23] W. Diffie and M. E. Hellman. *New Directions In Cryptography*. IEEE Transactions on Information Theory, Vol. IT-22, NO 6, pp. 644-654 (Nov., 1976)
- [24] Atsushi Fujioka, Tatsuaki Okamoto, Kazuo Ohta. *A Practical Secret Voting Scheme for Large Scale Elections*.
- [25] Olivier Blazy, David Derler, Daniel Slamanig, Raphael Spreitzer *Non-Interactive Plaintext (In-)equality Proofs and Group Signatures with Verifiable Controllable Linkability*
- [26] Jens Groth and Amit Sahai. *Efficient Non-Interactive Proof Systems for Bilinear Groups*. In Eurocrypt, 2008.
- [27] Torben Pryds Pedersen. *A Threshold Cryptosystem without a Trusted Party*.
- [28] Brands, S. : *Rethinking Public Key Infrastructures and Digital Certificates: Building in Privacy*, MIT Press (2000)