Εθνικο Μετσοβιο Πολυτεχνειο

Σχολη Ηλεκτρολογων Μηχανικων και Μηχανικων Υπολογιστων

Τομεας Τεχνολογιας Πληροφορικης και Υπολογιστων

Τυφλές Υπογραφές Υπό Συνθήκη και Εφαρμογή τους σε Πρωτόκολλα Ψηφοφοριών Ανθεκτικά σε Επιθέσεις Εξαναγκασμού

Διπλωματικη Εργασια

του

ΑΛΕΞΑΝΔΡΟΥ ΖΑΧΑΡΑΚΗ

Επιβλέπων: Αριστείδης Παγουρτζής
Αν. Καθηγητής Ε.Μ.Π.

Εργαστηριο Λογικης και Επιστημης Υπολογισμων

Αθήνα, Απρίλης 2017

Εθνικό Μετσόβιο Πολυτεχνείο
Σχολή Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών
Τομέας Τεχνολογίας Πληροφορικής και Υπολογιστών
Εργαστήριο Λογικής και Επιστήμης Υπολογισμών

Τυφλές Υπογραφές Υπό Συνθήκη και Εφαρμογή τους σε Πρωτόκολλα Ψηφοφοριών Ανθεκτικά σε Επιθέσεις Εξαναγκασμού

Διπλωματικη Εργασια

του

ΑΛΕΞΑΝΔΡΟΥ ΖΑΧΑΡΑΚΗ

Επιβλέπων: Αριστείδης Παγουρτζής
Καθηγητής Ε.Μ.Π.

Εγκρίθηκε από την τριμελή εξεταστική επιτροπή την 10η Απριλίου 2017.

(Υπογραφή)	*(Υπογραφή)*	*(Υπογραφή)*
.....	.....	.....
Αριστείδης Παγουρτζής	Αντώνιος Συμβώνης	Άγγελος Κιαγιάς
Αν. Καθηγητής Ε.Μ.Π.	Καθηγητής Ε.Μ.Π.	Αν. Καθηγητής Ε.Κ.Π.Α.

Αθήνα, Απρίλης 2017

(Υπογραφή)

.....

Αλεξανδρος Ζαχαρακης
Διπλωματούχος Ηλεκτρολόγος Μηχανικός και Μηχανικός Υπολογιστών Ε.Μ.Π.

Εθνικό Μετσόβιο Πολυτεχνείο
Σχολή Ηλεκτρολόγων Μηχανικών και Μηχανικών Υπολογιστών
Τομέας Τεχνολογίας Πληροφορικής και Υπολογιστών
Εργαστήριο Λογικής και Επιστήμης Υπολογισμών

Ευχαριστίες

Θα ήθελα να ευχαριστήσω τον επιβλέποντα καθηγητή κ. Παγουρτζή για την καθοδήγηση και την επίβλεψη αυτής της εργασίας. Επίσης ευχαριστώ ιδιαίτερα τον διδακτορικό φοιτητή κ. Γροντά για την καθοδήγηση, τις συζητήσεις και την συνεργασία που είχαμε στο διάστημα συγγραφής αυτής της εργασίας. Ευχαριστώ επίσης θερμά τα μέλη της εξεταστικής επιτροπής. Τέλος θα ήθελα να ευχαριστήσω την οικογένειά μου, φίλους μου και κοντινούς μου ανθρώπους για τη διαρκή στήριξη τους.

Περίληψη

Τα τελευταία χρόνια υπάρχει μεγάλη έρευνα γύρω από το ζήτημα των ηλεκτρονικών ψηφοφοριών και ειδικότερα των απομακρυσμένων ηλεκτρονικών ψηφοφοριών, πχ ψηφοφορίες μέσω διαδικτύου. Μία από τις κύριες απαιτήσεις των πρωτοκόλλων ηλεκτρονικής ψηφοφορίας είναι η ανθεκτικότητα σε επιθέσεις εξαναγκασμού, δηλαδή η αδυναμία κάποιας οντότητας να επηρεάσει την συμπεριφορά κάποιου ψηφοφόρου. Η ιδιότητα αυτή γίνεται ακόμα πιο επιτακτική σε συστήματα απομακρυσμένων ηλεκτρονικών ψηφοφοριών, όπου η έλλειψη μίας φυσικής οντότητας υπεύθυνης για την ορθή διεκπεραίωση της ψηφοφορίας κάνει ευκολότερη τη μαζική εκτέλεση τέτοιων επιθέσεων, ακυρώνοντας ουσιαστικά την ψηφοφορία.

Μία επίσης σημαντική ιδιότητα είναι η μυστικότητα της ψήφου. Στα περισσότερα προτεινόμενα στη βιβλιογραφία πρωτόκολλα, αυτή διασφαλίζεται μέσω κρυπτογραφικών μεθόδων που στηρίζονται στη δυσκολία υπολογιστικών προβλημάτων. Ένας εγγενής κίνδυνος για όλα αυτά τα προβλήματα είναι η δυσκολία αυτή να πάψει να ισχύει στο μέλλον αποκαλύπτοντας τις επιλογές των ψηφοφόρων σε παρελθοντικές ψηφοφορίες.

Στην παρούσα διπλωματική εργασία παρουσιάζουμε μία πρόταση για ένα νέο κρυπτογραφικό αρχέτυπο που ονομάζουμε τυφλές υπογραφές υπό συνθήκη και μια υλοποίησή του που στηρίζεται στις τυφλές υπογραφές Okamoto-Schnorr. Το αρχέτυπο αυτό σκοπό έχει να επιτρέπει στον υπογράφων να δίνει έγκυρες ή άκυρες τυφλές υπογραφές κατά βούληση χωρίς ο παραλήπτης να μπορεί να ξεχωρίσει τι είδους υπογραφή πήρε. Η υλοποίηση αυτή παρέχει στατιστική ανωνυμία στον χρήστη που ζητάει μία υπογραφή, δεν στηρίζεται δηλαδή σε κάποιο υπολογιστικό πρόβλημα.

Τέλος δείχνουμε πως το παραπάνω σχήμα μπορεί να εφαρμοστεί σε πρωτόκολλα απομακρυσμένων ηλεκτρονικών ψηφοφοριών για να πάρουμε πρωτόκολλα που αφενός είναι ανθεκτικά σε επιθέσεις εξαναγκασμού και αφετέρου ενισχύουν την ιδιωτικότητα των ψηφοφόρων μέσω των ιδιοτήτων των τυφλών υπογραφών.

Λέξεις Κλειδιά

Ψηφοφορίες, Εκλογές, Κρυπτογραφία, Επιθέσεις εξαναγκασμού, Τυφλές υπογραφές, Τυφλές υπογραφές υπό συνθήκη, Ασφαλής υπολογισμός

Abstract

In the last years there is flourishing research around electroning voting and more specifically around remote electroning voting, for example voting through the internet. One of the most important properties of such protocols is security against coercion attacks, meaning the inability of an entity to influence the behavior of a voter. This property becomes more compelling in remote voting protocols where the lack of a physical entity responsible for the proper execution of the elections makes easier the massive execution of such attacks, thus canceling the purpose of the elections.

Another important property is ballot secrecy. The majority of the proposed electronic election protocols achieve this property through cryptographic means which base their security on computationally hard problems. An inherent disadvantage of such method is that this difficulty might tumble, thus revealing the voters' choices in past elections.

In this diploma thesis we present a new cryptographic primitive which we call Conditional Blind Signatures and one instanciation of it based on the Okamoto-Schnorr blind signatures scheme. The purpose of this primitive is to allow the signer to decide whether to issue a valid or an invalid signature while the recepiant cannot computationally distinguish which is the case. Its implementation provides statistical anonimity to the recepiant, which means that it is not based on any computationally hard problem.

Finally, we show how this new primitive can be applied to remote electronic voting schemes to achieve both coercion resistance and better privacy guarantees to the voters through the properties of blind signatures.

Keywords

Voting, Elections, Cryptography, Coercion, Blind signatures, Conditional blind signatures, Secure Computation

Περιεχόμενα

Κεφάλαιο 1

Εισαγωγή

1.1 Γενικά Περί Ψηφοφοριών

Ένας χαλαρός ορισμός που μπορούμε να δώσουμε για μία ψηφοφορία είναι να την ορίσουμε ως ένα σύνολο σαφώς καθορισμένων κανόνων και μηχανισμών που επιτρέπει σε μία ομάδα να συναποφασίσει με δίκαιο τρόπο για ένα ζήτημα, λαμβάνοντας υπ όψην τις απόψεις όλων των μελών της. Το τί είναι δίκαιο και το πώς οι ατομικές απόψεις συμβάλλουν στην τελική απόφαση δεν μπορεί να οριστεί καθώς είναι μεταβλητό ιστορικά, κοινωνικά αλλά και ανάλογα της ίδιας της φύσης του εκάστοτε ζητήματος.

Χαρακτηριστικότερο παράδειγμα ψηφοφορίας είναι οι βουλευτικές εκλογές. Το πρόβλημα που πρέπει να συναποφασιστεί από τους πολίτες είναι η επιλογή αντιπροσώπων για τα επόμενα χρόνια. Η διαδικασία συνοπτικά είναι η ακόλουθη: Κάθε ψηφοφόρος, αφού πιστοποιήσει την ταυτότητά του και το δικαίωμα συμμετοχής του, επιλέγει από ένα σύνολο δυνατών επιλογών με μυστικό τρόπο ένα ψηφοδέλτιο και το βάζει σε ένα φάκελο, υπογεγραμμένο από μία κατάλληλη αρχή (δικαστικός αντιπρόσωπος). Στη συνέχεια καταθέτει την ψήφο στην κάλπη και με το πέρας του σταδίου κατάθεσης ψήφου μετριούνται οι ψήφοι. Με ένα προκαθορισμένο νομοθετικά πλαίσιο ανακοινώνεται το αποτέλεσμα των εκλογών από τις αρμόδιες αρχές.

Οι κύριες απαιτήσεις που πρέπει να ικανοποιούνται είναι η ορθότητα του αποτελέσματος και η μυστικότητα της ψήφου. Η πρώτη ιδιότητα χωράει μεν πολύ ανάλυση αλλά μπορεί να συνοψιστεί στα ακόλουθα: κάθε ψηφοφόρος πρέπει να μπορεί να εκφράσει την επιλογή του η οποία θα μετρηθεί στο τελικό αποτέλεσμα χωρίς να παραποιηθεί και το αποτέλεσμα βγαίνει μόνο βάση των επιλογών των ψηφοφόρων που έχουν δικαίωμα ψήφου. Η δεύτερη απαίτηση είναι πιο ξεκάθαρη. Κανένας εκτός από τον ίδιο τον ψηφοφόρο δεν πρέπει να μπορεί να μάθει κάτι για την επιλογή του.

Οι μηχανισμοί που απαιτούνται για την πραγματοποίηση των απαιτήσεων είναι κάποιες υλικοτεχνικές υποδομές (εγκαταστάσεις, γραφικές ύλες κλπ), το παραβάν που διασφαλίζει την μυστικότητα της ψήφου και κάποιες αρχές που είναι επιφορτισμένες με την ορθή διεξαγωγή των εκλογών. Οι αρχές αυτές είναι ένα σύνολο δικαστικών αντιπροσώπων και μέλη από τα

διάφορα κόμματα που επιβλέπουν την διαδικασία. Οι δικαστικοί αντιπρόσωποι διασφαλίζουν την ορθότητα του αποτελέσματος λόγω της ιδιότητάς τους ενώ τα κομματικά μέλη μέσω των αντικρουόμενων συμφερόντων τους.

Φυσικά οι παραπάνω μηχανισμοί δεν διασφαλίζουν στο έπακρο τις απαιτήσεις των εκλογών αλλά η κοινωνική νομιμοποίηση που έχουν τους καθιστούν επαρκής για να κάνουν το αποτέλεσμα των εκλογών κοινωνικά αποδεκτό. Μία μελλοντική αμφισβήτησή τους θα οδηγήσει σε αλλαγή του τρόπου διεξαγωγής των εκλογών. Το ίδιο ισχύει και για τις ίδιες τις απαιτήσεις του συστήματος.

Η παρούσα διπλωματική εργασία πραγματεύεται το δεύτερο κομμάτι, τους μηχανισμούς για τη διεξαγωγή εκλογών και συγκεκριμένα της απομακρυσμένες ηλεκτρονικές ψηφοφορίες.

1.1.1 Χρήση Ηλεκτρονικών Μέσων σε Ψηφοφορίες

Η εισαγωγή της τεχνολογίας στις ζωές μας τις τελευταίες δεκαετίες δεν θα μπορούσε να αφήσει ανεπηρέαστο τον τρόπο που διεξάγονται οι ψηφοφορίες. Εδώ και αρκετά χρόνια, πολλές χώρες χρησιμοποιούν σε κάποιο βαθμό ηλεκτρονικά μέσα για τη διεξαγωγή ακόμα και των σημαντικότερων εκλογών (πχ προεδρικές). Πολύ είναι η συζήτηση που γίνεται -δικαίως- και πολλά τα ζητήματα που πρέπει να επιλυθούν.

Ένα από αυτά -και ίσως το σημαντικότερο- είναι η αδυναμία μας να κατασκευάσουμε κατάλληλα μηχανήματα που αποδεδειγμένα δουλεύουν όπως θα έπρεπε. Κάθε είδους λογισμικό είναι επιρρεπές σε επιθέσεις και λάθη, γεγονός απαγορευτικό για ψηφοφορίες. Αλλά ακόμα και να μπορούσαμε να διασφαλίσουμε την σωστή λειτουργία του λογισμικού, η κοινωνία απέχει πολύ από το να εμπιστευτεί ηλεκτρονικά μέσα για ψηφοφορίες μεγάλης κλίμακας. Συνεπώς δεν αρκεί η επιστημονική κοινότητα να δίνει λύσεις, πρέπει αυτές να γίνονται και κοινωνικά αποδεκτές.

Ακόμα χειρότερα είναι τα πράγματα στις απομακρυσμένες ηλεκτρονικές ψηφοφορίες, δηλαδή σε ψηφοφορίες που οι ψηφοφόροι δεν ψηφίζουν σε κάποιο εξουσιοδοτημένο κέντρο, πχ ψηφοφορίες μέσω διαδικτύου. Η έλλειψη κάποιας έμπιστης αρχής, ως φυσική οντότητα, στο στάδιο της ψηφοφορίας κάνει πιο δύσκολη τόσο την ορθή και δίκαιη διεξαγωγή της ψηφοφορίας όσο και την κοινωνική της νομιμοποίηση.

Ένα άλλο σημαντικό ζήτημα, που είναι και το κύριο που θα εξερευνήσουμε στην παρούσα διπλωματική εργασία, είναι η σχεδίαση κατάλληλων πρωτοκόλλων που ικανοποιούν τις δεδομένες απαιτήσεις σε κατάλληλα σχεδιασμένα πλαίσια. Πρέπει δηλαδή για κάθε απαίτηση να σχεδιάζουμε ένα ικανοποιητικό μοντέλο με το οποίο να μπορούμε να ελέγξουμε κατά πόσο ένα δοσμένο πρωτόκολλο την ικανοποιεί. Τα μοντέλα αυτά πρέπει αφενώς να είναι εφικτά και αφετέρου να ανταποκρίνονται στην πραγματικότητα, δηλαδή αν ένα πρωτόκολλο είναι ασφαλές στο μοντέλο μας, να έχουμε κάποια βεβαιότητα ότι η ζητούμενη απαίτηση ικανοποιείται.

Οι μέθοδοι και τα αποτελέσματα του κλάδου της σύγχρονης κρυπτογραφίας έρχονται για να συμβάλλουν στην επίλυση αυτού του ζητήματος. Συγκεκριμένα, το πρόβλημα της ψηφοφορίας,

γενικά, είναι ένα πρόβλημα ασφαλούς υπολογισμού πολλών οντοτήτων (Secure Multi Party Computation [23], [22]). Στο πρόβλημα αυτό έχουμε n οντότητες, κάθε μία από τις οποίες έχει μία μυστική είσοδο x_i και μέσω αλληλεπιδράσεων οι οντότητες πρέπει να παράξουν μία έξοδο από μία συνάρτηση, δηλαδή να υπολογίσουν $f(x_1, \ldots, x_n) = (y_1, \ldots, y_n)$ όπου y_i είναι η έξοδος που παίρνει η i-οστή οντότητα και κάθε οντότητα δεν μαθαίνει τίποτε άλλο εκτός από την έξοδό της. Το πρόβλημα του ασφαλούς υπολογισμού γενικά, όσο κι αν αυτό ακούγεται αντιδιαισθητικό, ξέρουμε ότι έχει λύση [23] αλλά όχι αποδοτική. Αν θεωρήσουμε ένα σύστημα ψηφοφορίας που αποφασίζετε από ένα σύνολο επιλογών $\mathcal{C}$ η επιλογή με τις περισσότερες ψήφους τότε μπορούμε να θεωρήσουμε ότι κάθε ψηφοφόρος έχει μυστική είσοδο $x_i \in \mathcal{C}$ και η έξοδος είναι κοινή για όλους και είναι η νικηφόρα επιλογή.

Είναι φυσικό λοιπόν να χρησιμοποιήσουμε μεθόδους από την ευρύτερη περιοχή της κρυπτογραφίας για να προσεγγίσουμε το πρόβλημα των ηλεκτρονικών ψηφοφοριών. Το μεγάλο στοίχημα είναι, επί της ουσίας, να αντικαταστήσουμε υποδομές ουσιαστικές για την ασφάλεια μιας ψηφοφορίας, όπως για παράδειγμα το παραβάν, με κατάλληλα ηλεκτρονικά ανάλογα που διατηρούν την ασφάλεια αυτή.

1.1.2 Απαιτήσεις Πρωτοκόλλων Ηλεκτρονικών Ψηφοφοριών

Η μελέτη μας δεν μπορεί παρά να ξεκινήσει από τις απαιτήσεις που θέλουμε να έχουμε σε ένα πρωτόκολλο. Πολλές από αυτές, όπως θα δούμε είναι σχεδόν αντιφατικές, γεγονός που κάνει την σχεδίαση τέτοιων πρωτοκόλλων δύσκολη διαδικασία. Σημειώνουμε ότι συζητάμε μεν για ηλεκτρονικές ψηφοφορίες, αλλά οι απαιτήσεις που παρουσιάζουμε παρακάτω ισχύουν γενικά σε ψηφοφορίες.

- **Ακεραιότητα (integrity):** Το πρωτόκολλο πρέπει να βγάζει το αποτέλεσμα σύμφωνα με τον προκαθορισμένο τρόπο. Πιο συγκεκριμένα πρέπει η ψήφος του κάθε ψηφοφόρου να κατατίθεται όπως αυτός επιθυμούσε (cast as intended), να καταγράφεται όπως κατατίθεται (recorded as cast) και το σύστημα να μην αλλοιώνει, εισάγει ή διαγράφει ψήφους.
- **Επαληθευσιμότητα (Verifiability):** Είναι η ικανότητα του συστήματος να δώσει πειστήρια για τον ορθό υπολογισμό του αποτελέσματος, να μπορεί δηλαδή να αποδείξει ότι διατηρείται η ακεραιότητα. Είναι ουσιαστική ιδιότητα, δεδομένου ότι αυτή διασφαλίζει ότι το τελικό αποτέλεσμα εκφράζει την ομάδα. Η επαληθευσιμότητα χωρίζεται σε ατομική, διοικητική και καθολική. Η ατομική σημαίνει ότι ο ψηφοφόρος μπορεί να επαληθεύσει ότι η επιλογή του συμπεριλαμβάνεται στο τελικό αποτέλεσμα, η διοικητική ότι οι αρχές μπορούν να επαληθεύσουν ότι η ψηφοφορία έγινε όπως είχε σχεδιαστεί και η καθολική ότι κάθε ενδιαφερόμενη οντότητα μπορεί να επαληθεύσει την διαδικασία των εκλογών στο σύνολό της.
- **Μυστικότητα Ψήφου (Ballot Secrecy):** Το πρωτόκολλο πρέπει να διασφαλίζει ότι η επιλογή του κάθε ψηφοφόρου είναι μυστική από όλους εκτός τον ίδιο. Φυσικά κάτι τέτοιο δεν μπορεί να επιτευχθεί τέλεια αφού για παράδειγμα σε μία εκλογική διαδικασία

μπορεί όλοι οι συμμετέχοντες να ψηφίσουν την ίδια επιλογή. Τότε το τελικό αποτέλεσμα φανερώνει την επιλογή του κάθε ψηφοφόρου. Αυτό που θέλουμε να πετυχαίνει ένα πρωτόκολλο είναι κάθε ψήφος να παραμένει μυστική εκτός από ότι μπορεί να φανερωθεί από το τελικό αποτέλεσμα.

- **Έλλειψη Απόδειξης (Receipt Freeness):** Είναι η ιδιότητα του πρωτοκόλλου να μην παρέχει κανενός είδους απόδειξη στον ψηφοφόρο για την επιλογή του. Δηλαδή ο ψηφοφόρος δεν θα πρέπει να μπορεί να πείσει έναν τρίτο για την επιλογή του. Η ιδιότητα αυτή είναι μία επέκταση της ιδιωτικότητας.
- **Ανθεκτικότητα σε Επιθέσεις Εξαναγκασμού (Coercion Resistance):** Η ιδιότητα αυτή επεκτείνει την ιδιότητα έλλειψης απόδειξης. Εισήχθηκε από τους Juels, Catalano, Jackobson [31] και ουσιαστικά εκφράζει το γεγονός ότι ο ψηφοφόρος δεν μπορεί να εξαναγκαστεί σε κάποια συμπεριφορά κατά τη διάρκεια του πρωτοκόλλου από κάποιον ενεργό αντίπαλο, έναν αντίπαλο, δηλαδή, που τον προστάζει ενόσω εξελίσσεται η διαδικασία της ψηφοφορίας. Σημειώνουμε ότι ο εξαναγκασμός μπορεί να είναι και εθελοντικός, δηλαδή ο ψηφοφόρος να πουλήσει κάποια συγκεκριμένη συμπεριφορά.
- **Αυθεντικοποίηση:** Σε μία ψηφοφορία μόνο όσοι έχουν δικαίωμα και αφού το επιδείξουν μπορούν να ψηφίσουν.
- **Εξουσιοδότηση (Authorization):** Κάθε ψηφοφόρος πρέπει να υπακούει στο πρωτόκολλο. Χαρακτηριστικό παράδειγμα είναι κανένας ψηφοφόρος να μην μπορεί να διπλοψηφίσει (εφόσον φυσικά δεν το επιτρέπει το πρωτόκολλο).
- **Δικαιοσύνη (Fairness):** Η ιδιότητα αυτή σημαίνει ότι το αποτέλεσμα της διαδικασίας θα πρέπει να εξαρτάται αποκλειστικά και μόνο από τις εισόδους των συμμετεχόντων (την επιλογή τους). Δεν πρέπει δηλαδή να είναι μεταβλητό ως προς τη διαδικασία υπολογισμού. Θα πρέπει λοιπόν το σύστημα να μην διαρρέει ενδιάμεσα αποτελέσματα, αφού αυτά μπορούν να επηρεάσουν τις επιλογές των παικτών. Σημαντική συνέπεια είναι ότι οι εκλογές δεν μπορούν να επαναληφθούν σε περίπτωση που κάτι δεν πάει καλά, αφού είναι βέβαιο ότι οι επιλογές των ψηφοφόρων θα έχουν μεταβληθεί.
- **Σταθερότητα (Robustness):** Τα συστατικά του πρωτοκόλλου πρέπει να είναι διαθέσιμα σε όλη την διάρκεια της ψηφοφορίας και να μην είναι δυνατόν η διαδικασία να σαμποταριστεί ή να αποτύχει λόγω λάθους.
- **Αποδοτικότητα** Το σύστημα πρέπει να είναι αποδοτικό ως προς κάποιους πόρους. Συνήθως αυτοί είναι ο χρόνος και το χρηματικό κόστος.

1.2 Αντικείμενο της Εργασίας

Η παρούσα διπλωματική ως αντικείμενο έχει πρωτόκολλα ηλεκτρονικής ψηφοφορίας με ανθεκτικότητα σε επιθέσεις εξαναγκασμού και έμφαση στην ιδιωτικότητα. Η μελέτη αυτών των συγκεκριμένων απαιτήσεων μόνο τυχαία δεν είναι.

Η ανθεκτικότητα σε επιθέσεις εξαναγκασμού είναι η ελάχιστη απαίτηση που μπορούμε να έχουμε για συστήματα απομακρυσμένων ψηφοφοριών. Αυτό ισχύει λόγω της έλλειψης μιας φυσικής οντότητας που επιβλέπει την διαδικασία στο στάδιο κατάθεσης ψήφου. Συνεπώς τέτοιες επιθέσεις μπορούν εύκολα να προκύψουν σε μεγάλη κλίμακα, κάνοντας έτσι την όλη διαδικασία ανούσια. Το πρόβλημα είναι από τη φύση του δύσκολο και γίνεται ακόμα δυσκολότερο αφού σαν έννοιες η ανθεκτικότητα σε εξαναγκασμούς φαίνεται αντιφατική με την επαληθευσιμότητα.

Η ιδιωτικότητα από την άλλη είναι ίσως η πιο σημαντική ιδιότητα. Διασφαλίζει ότι η επιλογή του ψηφοφόρου είναι η πραγματική και τον προστατεύει από πιθανές μελλοντικές συνέπειες λόγω της επιλογής του. Η κύρια (αλλά όχι η μόνη) κατεύθυνση για τη επίτευξή της βασίζεται σε κρυπτογραφικές μεθόδους όπως θα δούμε στη συνέχεια. Το κακό με τις κρυπτογραφικές μεθόδους είναι ότι έχουν *ημερομηνία λήξης*. Αυτό γιατί οι κρυπτογραφικές μέθοδοι, στην πλειονότητά τους στηρίζονται σε υπολογιστικά δύσκολα προβλήματα. Η πρόοδος όμως της τεχνολογίας μπορεί να κάνει προβλήματα που είναι δύσκολα σήμερα, εύκολα σε 50 -για παράδειγμα- χρόνια.

Συνεπώς, οι ψήφοι μπορεί στο μέλλον να φανερωθούν και οι ψηφοφόροι να υποστούν συνέπειες για την τότε επιλογή τους. Επιπλέον, ο φόβος ενός τέτοιου μελλοντικού συμβάντος μπορεί να τους οδηγήσει στο να μην εκφράσουν την πραγματική τους επιλογή. Θα μπορούσε κάποιος να ισχυριστεί ότι σε κάποιον ικανοποιητικό βαθμό η πρόοδος της τεχνολογίας μπορεί να προβλεφθεί και συνεπώς να επιλέξουμε προβλήματα με κατάλληλη δυσκολία. Αυτό είναι αλήθεια όμως δεν λύνει το πρόβλημα. Αυτό γιατί δεν μπορούμε να προβλέψουμε την πρόοδο στους αλγορίθμους. Μία νέα μέθοδος για την εύρεση του διακριτού λογαρίθμου θα έκανε τις εκτιμήσεις μας να πέσουν πολύ έξω. Πρέπει συνεπώς να κυνηγάμε πληροφοριοθεωρητική μυστικότητα, μυστικότητα δηλαδή που ισχύει a priori και όχι στη βάση ενός υπολογιστικού προβλήματος.

Στόχος, λοιπόν, της παρούσας εργασίας είναι να μελετήσουμε το πρόβλημα αυτό και να προτείνουμε το πρώτο στη βιβλιογραφία πρωτόκολλο που συνδυάζει τις δύο παραπάνω ιδιότητες στηριζόμενο στην υπόθεση ότι έχουμε στη διάθεσή μας ένα τέλεια ανώνυμο κανάλι, ένα κανάλι δηλαδή που δεν αποκαλύπτει καμία πληροφορία για τον αποστολέα.

Μία εκτενέστατη και πλήρης μελέτη συστημάτων ηλεκτρονικών ψηφοφοριών είναι η διπλωματική εργασία [27].

1.3 Συμβολισμοί και Βασικοί Ορισμοί

Στην ενότητα αυτή παρουσιάζουμε τους συμβολισμούς και τους βασικούς ορισμούς που θα χρησιμοποιήσουμε σε αυτήν την εργασία. Στην εργασία αυτή θεωρούμε ότι ο αναγνώστης έχει κάποια βασική εξοικείωση με θεωρία αριθμών και αφηρημένη άλγεβρα. Τα κομμάτια που χρησιμοποιούμε καλύπτονται στο [49] .

1.3.1 Συμβολισμοί

Συμβολίζουμε $\mathbb{Z}$ το σύνολο των ακεραίων $\{\ldots, -2, -1, 0, 1, 2, \ldots\}$. Συμβολίζουμε την προσθετική ομάδα υπολοίπων n ως $\mathbb{Z}_n$. Μία τυχαία ομάδα συμβολίζεται ως $\mathbb{G}$. Αν S είναι ένα σύνολο συμβολίζουμε $a \leftarrow_R S$ την ανάθεση τιμής στο a από το σύνολο S βάση της ομοιόμορφης κατανομής. Η ανάθεση τιμής σε μια μεταβλητή από μία πράξη ή γενικότερα από έναν αλγόριθμο A συμβολίζεται ως $a \leftarrow A(\cdot)$. Σε περίπτωση που ο αλγόριθμος είναι πιθανοτικός ο προηγούμενος συμβολισμός σημαίνει ότι επιλέγουμε την τυχαία ταινία του A ομοιόμορφα από το σύνολο των δυνατών επιλογών, εκτελούμε τον A με αυτή και αναθέτουμε το αποτέλεσμα της εκτέλεσης στην μεταβλητή a. Αν θέλουμε να τονίσουμε την τυχαιότητα που χρησιμοποιήθηκε γράφουμε $a \leftarrow A(\cdot; r)$. Η συμβολοσειρά μήκους 0 συμβολίζεται ως ϵ και η παράθεση δύο συμβολοσειρών s_1, s_2 αναγράφεται ως $s_1||s_2$. Συμβολίζουμε $\{0,1\}^n$ το σύνολο $\{s \mid s = b_1||b_2||\ldots||b_n$ τέτοιω ώστε $b \in \{0,1\}\}$ με $\{0,1\}^0 = \epsilon$ και $\{0,1\}^* = \bigcup_{n\in\mathbb{Z}}\{0,1\}^n$.

1.3.2 Ορισμοί

Ορισμός 1.1. *Μία γλώσσα L με αλφάβητο $\{0,1\}^*$ είναι ένα υποσύνολο του $\{0,1\}^*$.*

Οι γλώσσες μοντελοποιούν τα προβλήματα απόφασης. Για παράδειγμα η γλώσσα $L = \{l|l$ είναι πρώτος αριθμός$\}$ μοντελοποιεί το πρόβλημα αν ένας αριθμός είναι πρώτος. Κατάλληλα υποσύνολα γλωσσών ορίζουν κλάσεις πολυπλοκότητας. Παρακάτω δίνονται (άτυποι) ορισμοί για τις κλάσεις $\mathbf{P}, \mathbf{NP}$.

Ορισμός 1.2. *Μία γλώσσα L ανήκει στο $\boldsymbol{P}$ αν υπάρχει αλγόριθμος που με είσοδο x αποφασίζει αν $x \in L$ σε χρόνο πολυωνιμικό ως προς το μήκος της εισόδου x.*

Ορισμός 1.3. *Μία γλώσσα L ανήκει στο $\boldsymbol{NP}$ αν υπάρχει πολυωνιμικός αλγόριθμος A τέτοιος ώστε για κάθε x ισχύει $x \in L$ ανν υπάρχει w με μήκος πολυωνυμικό ως προς το μήκος της εισόδου x για τα οποία ισχύει $A(x, w) = 1$.*

Η κλάση $\mathbf{P}$ μοντελοποιεί τον αποδοτικό υπολογισμό. Η κλάση $\mathbf{NP}$ μοντελοποιεί το σύνολο των προβλημάτων για τα οποία αν μία είσοδος ανήκει σε μία γλώσσα υπάρχει σύντομο πιστοποιητικό ότι ανήκουν στη γλώσσα. Το w αναφέρεται ως πιστοποιητικό ή μάρτυρας. Το μεγαλύτερο ανοιχτό πρόβλημα της θεωριτικής πληροφορικής είναι αν οι δύο αυτές κλάσεις είναι ίσες ή διαφορετικές.

Στην εργασία αυτή δεν θα μας απασχολήσουν μόνο προβλήματα απόφασης αλλά και προβλήματα αναζήτησης, δηλαδή με είσοδο x να βρεθεί ένα y τέτοιο ώστε $(x, y) \in R$ όπου R μια διμελής σχέση. Για παράδειγμα να βρεθεί ένας πρώτος διαιρέτης του x.

Ορισμός 1.4. *Ένας Probabilistic Polynomial Time (PPT) αλγόριθμος είναι ένας πιθανοτικός αλγόριθμός για ένα πρόβλημα αναζήτησης που με είσοδο x τερματίζει σε πολυωνιμικό*

χρόνο ως προς το μήκος του $|x|$.

Μία συνάρτηση $\mu(n)$ είναι αμελητέα αν είναι ασυμπτωτικά μικρότερη από κάθε αντίστροφο πολυώνυμο. Πιο τυπικά

Ορισμός 1.5. *Μία συνάρτηση* $\mu : \mathbb{N} \to \mathbb{R}$ *είναι αμελητέα αν για κάθε θετικό πολυώνυμο* $p(\cdot)$ *υπάρχει* N *τέτοιο ώστε για κάθε* $n > N$ *ισχύει ότι* $\mu(n) < \frac{1}{p(n)}$.

Άτυπα ένας αλγόριθμος A με μαντείο έναν αλγόριθμο O συμβολίζεται A^O και σημαίνει ότι ο A μπορεί να χρησιμοποιεί τον αλγόριθμο O ως υπορουτίνα σπαταλώντας ένα μόνο υπολογιστικό βήμα. Δηλαδή ο αλγόριθμος A μπορεί σε ένα βήμα εκτέλεσης να πάρει μία απάντηση $O(x)$ για x της επιλογής του.

Για περισσότερα σχετικά με θεωρία πολυπλοκότητας παραπέμπουμε στα [5],[39].

Στη συνέχεια δίνουμε κάποιους ορισμούς για τη στατιστική απόσταση δύο οικογενειών τυχαίων μεταβλητών.

Ορισμός 1.6. *Έστω* I *ένα αριθμήσιμο σύνολο δεικτών. Μία οικογένεια τυχαίων μεταβλητών* $\mathcal{X}$ *είναι μία άπειρη ακολουθία τυχαίων μεταβλητών που παραμετροποιούνται από το* I *δηλαδή* $\mathcal{X} = \{X_i\}_{i \in I}$. *Συνήθως θεωρούμε* $I \equiv \mathbb{N}$.

Ορισμός 1.7. *(Στατιστική απόσταση) Ορίζουμε ως στατιστική απόσταση δύο τυχαίων μεταβλητών* X, Y *την ποσότητα*

$$\Delta = \frac{1}{2} \sum_{u \in V} | \Pr[X = u] - \Pr[Y = u] |$$

όπου V *είναι η ένωση των πεδίων τιμών τους.*

Ορισμός 1.8. *(Τέλεια μη διαχωρισιμότητα) Δύο οικογένειες τυχαίων μεταβλητών* $\mathcal{X}, \mathcal{Y}$ *είναι τέλεια μη διαχωρίσιμες που συμβολίζουμε* $\mathcal{X}_n \equiv \mathcal{Y}_n$ *ανν για κάθε* $n \in \mathbb{N}$ *ισχύει ότι*

$$\Pr[X_n = a] = \Pr[Y_n = a]$$

για κάθε a *στο πεδίο τιμών των* X_n, Y_n.

Ορισμός 1.9. *(Στατιστική μη διαχωρισιμότητα) Δύο οικογένειες τυχαίων μεταβλητών* $\mathcal{X}, \mathcal{Y}$ *είναι στατιστικά μη διαχωρίσιμες που συμβολίζουμε* $\mathcal{X} \equiv^s \mathcal{Y}$ *ανν υπάρχει αμελητέα ως προς το* n *συνάρτηση* μ *τέτοια ώστε για αρκούντως μεγάλα* n *ισχύει* $\Delta_n \leq \mu(n)$ *όπου* Δ_n *η στατιστική απόσταση των* X_n, Y_n.

Ορισμός 1.10. *(Υπολογιστική μη διαχωρισιμότητα) Δύο οικογένειες τυχαίων μεταβλητών* $\mathcal{X}, \mathcal{Y}$ *είναι υπολογιστικά μη διαχωρίσιμες που συμβολίζουμε* $\mathcal{X} \equiv^c \mathcal{Y}$ *ανν για κάθε PPT*

αλγόριθμο D και κάθε πολυώνυμο p υπάρχει αμελητέα συνάρτηση μ τέτοια ώστε για αρκούντως μεγάλα n να ισχύει

$$|\Pr[D(X_n, 1^n) = 1] - |\Pr[D(Y_n, 1^n) = 1]| \leq \mu(n)$$

Παρατηρήστε ότι οι δύο τελευταίοι ορισμοί είναι διαφορετικοί. Δύο οικογένειες τυχαίων μεταβλητών μπορεί να έχουν μη αμελητέα στατιστική απόσταση αλλά κανένας PPT αλγόριθμος να μην μπορεί να τις ξεχωρίσει.

1.4 Δομή της εργασίας

- **Κεφάλαιο 2:** Στο κεφάλαιο αυτό θα κάνουμε μία ανασκόπηση των βασικότερων κρυπτογραφικών εργαλείων και πρωτοκόλλων. Έμφαση δίνεται σε πρωτόκολλα που χρησιμοποιούνται στις ηλεκτρονικές ψηφοφορίες και κυρίως σε αυτά που χρησιμοποιούμε στη συνέχεια της εργασίας.
- **Κεφάλαιο 3:** Στο επόμενο κεφάλαιο μελετάμε πρωτόκολλα ηλεκτρονικών ψηφοφοριών. Εστιάζουμε κυρίως στο πλαίσιο που τέθηκε από τους Juels, Catalano, Jackobson [31] για το coercion resistance και το πρωτόκολλο που παρουσίασαν στην ίδια εργασία, αναλύουμε την πολυπλοκότητά του και στη συνέχεια παρουσιάζουμε μετέπειτα εξελίξεις του που υπάρχουν στη βιβλιογραφία.
- **Κεφάλαιο 4:** Αυτό είναι το κεφάλαιο με την συνεισφορά της εργασία. Ορίζουμε αρχικά ένα νέο κρυπτογραφικό primitive που ονομάζουμε Conditional Blind Signatures, επιχειρηματολογούμε για τις ιδιότητές του και τέλος παρουσιάζουμε μία πρόταση για ένα νέο πρωτόκολλο ηλεκτρονικής ψηφοφορίας που στηρίζεται στα JCJ, FOO με στόχο αφενός να βελτιώσουμε την απόδοση του JCJ και αφετέρου να ενισχύσουμε την ιδιωτικότητα σε Coercion Resistant πρωτόκολλα ηλεκτρονικής ψηφοφορίας.

Κεφάλαιο 2

Βασικά Κρυπτογραφικά Primitives και Πρωτόκολλα

Η ενότητα αυτή σκοπό έχει να κάνει μία σύντομη επισκόπηση στα βασικότερα κρυπτογραφικά primitives και πρωτόκολλα. Έμφαση δίνεται σε αυτά που θα χρειαστούμε στη συνέχεια για να μελετήσουμε πρωτόκολλα ψηφοφοριών. Η ενότητα αυτή δεν προσδοκά να παρουσιάσει τα εξεταζόμενα θέματα στην πληρότητά τους, παρά μόνο να επεξηγήσει βασικές έννοιες με όσο γίνεται πιο εύκολο τρόπο. Για πιο βαθιά μελέτη ο ενδιαφερόμενος αναγνώστης μπορεί να κοιτάξει ένα από τα πολλά σχετικά βιβλία, όπως το [34], στο οποίο βασίζεται σε μεγάλο κομμάτι του η παρούσα ενότητα.

2.1 Ανταλλαγή κλειδιού Diffie Hellman

Σε αυτήν την υποενότητα θα εξετάσουμε την ανταλλαγή κλειδιού Diffie Hellman [16], μία επαναστατική δουλειά των Diffie και Hellman, που γέννησε την κρυπτογραφία δημοσίου κλειδιού και άλλαξε ριζικά την αντίληψή μας ως προς την έννοια της ασφαλούς επικοινωνίας. Το σενάριο είναι απλό. Δύο πρόσωπα, η Alice και ο Bob θέλουν να συμφωνήσουν σε ένα μυστικό k μέσω ενός ανασφαλούς καναλιού χωρίς να έχουν ανταλλάξει πιο πριν κάποια μυστική πληροφορία. Ως ανασφαλές κανάλι θεωρούμε έναν οποιονδήποτε δίαυλο επικοινωνίας τα μηνύματα του οποίου διαβάζονται και από άλλους (ας πούμε από την Eve) πέρα από τους Alice και Bob.

Το σενάριο αυτό φαίνεται ότι δεν μπορεί να επιτευχθεί. Είναι τελείως ενάντια στην διαίσθησή μας. Πως μπορούν δύο άνθρωποι που μιλάνε μπροστά σε άλλους χωρίς να ψιθυρίσουν κάτι ο ένας στον άλλον να αρχίσουν να μιλάνε μυστικά και μόνο οι ίδιοι να είναι σε θέση να καταλάβουν τι λένε; Σίγουρα δεν μπορούν να βασιστούν μόνο σε κάτι που λένε αφού οι ωτακουστές θα το ακούσουν. Όπως θα δούμε παρακάτω αυτό που το κάνει δυνατό είναι η κατοχή από τους δύο *μυστικής γνώσης*, δηλαδή καθένας ξέρει κάτι που δεν το ξέρει κανένας άλλος (ούτε ο έτερος παίχτης). Κατάλληλος συνδυασμός αυτών των μυστικών κάνει το

σενάριο εφικτό για τους αγαπητούς Alice και Bob και την προσπάθεια της κακής Eve να κρυφακούσει μάταιη.

2.1.1 Το μοντέλο ασφάλειας για την ανταλλαγή κλειδιού

Πρωτού δούμε το ίδιο το πρωτόκολλο, για να επιχειρηματολογήσουμε για την ασφάλειά του θα πρέπει κάπως να ορίσουμε τι θα πει το πρωτόκολλο να είναι ασφαλές. Θα πρέπει δηλαδή να φτιάξουμε ένα μοντέλο, μία αφαίρεση από την πραγματικότητα, στην οποία θα έχουμε ορίσει με σαφήνεια και ακρίβεια τους στόχους που θέλουμε να πετύχουμε για το σενάριό μας και στα πλαίσια της οποίας θα είναι δυνατό να εξεταστεί ως προς τους στόχους ένα δοθέν πρωτόκολλο. Φυσικά μετά το πρωτόκολλο θα πρέπει να υλοποιηθεί στον πραγματικό κόσμο, πράγμα καθόλου εύκολο. Τέτοια ζητήματα όμως, δεν θα μας απασχολήσουν σε αυτή την εργασία.

Όπως αναφέραμε στην εισαγωγή μελετάμε τα πράγματα από σκοπιά πολυπλοκότητας, θα δούμε δηλαδή την ασυμπτωτική συμπεριφορά του πρωτοκόλλου ως προς μια παράμετρο ασφάλειας. Το σενάριο εδώ είναι απλό. Έχουμε ένα ανασφαλές κανάλι στο οποίο ένας αντίπαλος κρυφακούει τις συνομιλίες δύο παιχτών. Αυτοί θέλουν, ανταλλάσσοντας κάποια μηνύματα, να καταλήξουν σε ένα κοινό μυστικό k χωρίς αυτό να το μάθει ο αντίπαλος. Εδώ ο αντίπαλος είναι *στατικός*, δηλαδή δεν παρεμβαίνει στο κανάλι (δεν τροποποιεί/διαγράφει/προσθέτει μηνύματα). Απλά ακούει την συνομιλία. Για την μελέτη της ασφάλειας του πρωτοκόλλου κατασκευάζουμε ένα παιχνίδι που παίζει ένας διεκδικητής (challenger) $\mathcal{C}$ εναντίον ενός αντιπάλου (adversary) $\mathcal{A}$ που επιτίθεται στο πρωτόκολλο Π. Το παιχνίδι δίνεται παρακάτω.

Algorithm: $\text{KeyExchange}^{\text{eav}}_{\mathcal{A},\Pi}$

Input : security parameter n
Output: $b \in \{0,1\}$

/* τρέχουμε το πρωτόκολλο */
$(k, trans) \leftarrow \Pi(1^n)$
/* στρίβουμε κέρμα */
$b \leftarrow_R \{0,1\}$
if $b = 0$ **then**
  $\hat{k} = k$
else
  $\hat{k} \leftarrow_R \{0,1\}^n$
end
$\mathcal{A}(trans, 1^n, \hat{k}) = b'$
if $b = b'$ **then**
  output 1
else
  output 0
end

Ας δούμε τι σημαίνουν τα παραπάνω. Ο $\mathcal{A}$ είναι ένας αλγόριθμος που παίρνει ένα transcript από τα μηνύματα που ανταλλάχθηκαν από το πρωτόκολλο και ένα κλειδί και προσπαθεί να μαντέψει ένα από δύο πιθανά σενάρια: είτε το κλειδί $\hat{k}$ είναι έξοδος του πρωτοκόλλου είτε είναι ένα τυχαίο στοιχείο από τον ίδιο χώρο. Αυτό εξαρτάται από ένα κέρμα b που ρίχνει ο $\mathcal{C}$. Το αποτέλεσμα του πειράματος καθορίζεται από την επιτυχία του αντιπάλου στο να καταλάβει σε ποιο σενάριο ανήκει η είσοδος που του δόθηκε.

Ας φανταστούμε ένα πρωτόκολλο τώρα που οι παίχτες ανταλλάσσουν τα ίδια μηνύματα αλλά με κάποιον μαγικό, *ιδεατό* τρόπο το k που παράγουν είναι τυχαίο και ανεξάρτητο από τα μηνύματα που αντάλλαξαν. Είναι δηλαδή ένα ομοιόμορφα κατανεμημένο στοιχείο από ένα σύνολο $\mathcal{K}$. Ο αντίπαλος που βλέπει τα μηνύματα δεν κερδίζει τίποτε από αυτά. Το μόνο που ξέρει σε αυτήν την περίπτωση για το k είναι ότι είναι ένα ομοιόμορφα επιλεγμένο στοιχείο του K και δεν έχει κάποιον τρόπο να μάθει κάτι για αυτό. Το καλύτερο που μπορεί να κάνει είναι διαλέξει και αυτός τυχαία. Ένα τέτοιο πρωτόκολλο θα ήταν το τέλειο για την δουλειά που θέλουμε. Το σενάριο αυτό είναι ακριβώς αυτό που συμβαίνει αν $b = 1$.

Στην πραγματικότητα όμως το k εξαρτάται από το transcript και ένας αντίπαλος μπορεί να εκμεταλλευτεί το γεγονός. Αυτό που θα θέλαμε να πετύχουμε είναι να μην μπορεί να κάνει σημαντικά περισσότερα σε αυτήν την περίπτωση. Για να γίνει πιο κατανοητό ας φανταστούμε ότι ο $\mathcal{A}$ δεν μπορεί να ξεχωρίσει σε ποιον κόσμο είναι βάση της συνομιλίας και του $\hat{k}$. Τότε για αυτόν η *όψη* του είναι ίδια και για τα δύο σενάρια. Η μόνη πληροφορία για το κλειδί είναι ότι είναι ομοιόμορφα κατανεμημένο και το transcript του είναι άχρηστο. Οδηγούμαστε στον ακόλουθο ορισμό:

Ορισμός 2.11. *Το πλεονέκτημα ενός αντιπάλου $\mathcal{A}$ ως προς το πρωτόκολλο Π ορίζεται ως*

$$Adv_{\mathcal{A},\Pi}(n) = |\Pr[KeyExchange^{eav}_{\mathcal{A},\Pi}(n) = 1] - \frac{1}{2}|$$

Διαισθητικά το πλεονέκτημα μας λέει πόσο καλά μπορεί ο αντίπαλος στο συγκεκριμένο πρωτόκολλο να ξεχωρίσει τα δύο σενάρια πάντα συναρτήσει της παραμέτρου ασφαλείας n. Όσο μικρότερο είναι το πλεονέκτημα τόσο καλύτερα το πρωτόκολλό μας μιμείται το ιδεατό και συνεπώς τόσο πιο ασφαλές είναι. Με βάση αυτά που είπαμε στην εισαγωγή περί αποδοτικού υπολογισμού θα θέλαμε το πλεονέκτημα του αντιπάλου να είναι αμελητέα συνάρτηση ως προς την παράμετρο ασφαλείας n και αυτό να ισχύει για κάθε πιθανό αντίπαλο. Δηλαδή δεν λαμβάνουμε υπ' όψην μόνο τις στρατηγικές που μπορούμε να σκεφτούμε αλλά ποσοδεικτούμε ως προς κάθε πιθανό PPT αντίπαλο. Οδηγούμαστε λοιπόν στον εξής ορισμό ασφάλειας για πρωτόκολλα ανταλλαγής κλειδιού.

Ορισμός 2.12. *Ένα πρωτόκολλο ανταλλαγής κλειδιού Π με είσοδο την παράμετρο ασφαλείας 1^n είναι ασφαλές ως προς στατικούς αντιπάλους (secure under the precence of eavsdropers) αν για κάθε PPT αντίπαλο $\mathcal{A}$ υπάρχει αμελητέα συνάρτηση μ ως προς το n τέτοια ώστε $Adv_{\mathcal{A},\Pi}(n) \leq \mu(n)$*

2.2 Το πρωτόκολλο ανταλλαγής κλειδιού Diffie Hellman

Στην ενότητα αυτή θα παρουσιάσουμε το πρωτόκολλο DH και θα επιχειρηματολογήσουμε για την ασφάλειά του σχετικά με ένα δύσκολο πρόβλημα. Για το πρωτόκολλο χρειαζόμαστε αρχικά έναν πιθανοτικό αλγόριθμο πολυωνυμικού χρόνου $\mathcal{G}$ που με είσοδο την παράμετρο ασφαλείας n παράγει την περιγραφής μιας κυκλικής ομάδας τάξης q με $\|q\| = n$ πρώτο και έναν γεννήτορά της.

Αλγοριτημ: Diffie Hellman Key Exchange

Input : security parameter n

1 Η Alice παίρνει $(\mathbb{G}, q, g) \leftarrow \mathcal{G}(1^n)$

/* $\mathcal{G}$ η περιγραφή της ομάδας, q η τάξη της και g γεννήτορας */

2 Η Alice διαλέγει τυχαίο $x \leftarrow_R \mathbb{Z}_q$ και στέλνει $h_a = g^x$ στον Bob

3 Ο Bob διαλέγει τυχαίο $y \leftarrow_R \mathbb{Z}_q$ και στέλνει $h_b = g^y$ στην Alice

4 Η Alice υπολογίζει το κοινό μυστικό h_b^x

5 Ο Bob υπολογίζει το κοινό μυστικό h_a^y

Κατ΄ αρχήν ας επαληθεύσουμε ότι το πρωτόκολλο είναι ορθό, δηλαδή ότι οι δύο παίχτες καταλήγουν στο ίδιο μυστικό κλειδί. Αυτό γίνεται χρησιμοποιώντας βασικές ιδιότητες των πράξεων στην ομάδα και στο $\mathbb{Z}_q$. Συγκεκριμένα

$$h_b^x = (g^y)^x = g^{yx} = g^{xy} = (g^x)^y = h_a^y$$

Εισάγουμε στην συνέχεια τρία βασικά προβλήματα που σχετίζονται με το πρωτόκολλο και θα τα συναντάμε συχνά.

Discrete Logarithm Problem (DLOG): Με είσοδο περιγραφή ομάδας G, γεννήτορα g και $h \in H$ να βρεθεί $x \in |G|$ τέτοιο ώστε $h = g^x$.

Computational Diffie Hellman Problem(CDH): Με είσοδο περιγραφή ομάδας G, γεννήτορα g και στοιχεία $h_1, h_2 \in G$ με $h_1 = g^x$, $h_2 = g^y$ να βρεθεί $h \in G$ τέτοιο ώστε $h = g^{xy}$

Decisional Diffie Hellman Problem(DDH): Με είσοδο περιγραφή ομάδας G, γεννήτορα g και στοιχεία $h_1, h_2, h_3 \in G$ αν $h_1 = g^x$, $h_2 = g^y$ να αποφασιστεί αν $h_3 = g^{xy}$.

Εύκολα βλέπει κανείς ότι αν λύναμε το **DLOG** λύναμε το **CDH** και αν λύναμε το **CDH** τότε λύναμε το **DDH** . Δεν ξέρουμε κάτι για την αντίστροφη σειρά. Και τα τρία προβλήματα θεωρούνται δύσκολα προβλήματα για κατάλληλα επιλεγμένες ομάδες και πληθώρα κρυπτογραφικών συστημάτων στηρίζεται στην δυσκολία αυτών των προβλημάτων. Μάλιστα το **DLOG** συνδέεται στενά με το **FACTORING** . Κατάλληλες ομάδες για χρήση σε κρυπτογραφικά πρωτόκολλα είναι υποομάδες τάξης q του $\mathbb{Z}_p$ για p, q πρώτους και διάφορες οικογένειες

ελλειπτικών καμπυλών. Για περισσότερες πληροφορίες σχετικά με αυτά τα προβλήματα παραπέμπουμε στο [34].

Κάπως ταυτολογικά, το πρωτόκολλο Diffie Hellman είναι ασφαλές με βάση αυτά που λέγαμε παραπάνω αν το αποτέλεσμα που δίνει η $\mathcal{G}$ είναι ομάδα για την οποία το **DDH** είναι δύσκολο (δηλαδή αν κάθε αντίπαλος σε έξοδο από τον $\mathcal{G}$ δεν μπορεί να ξεχωρίσει τυχαία τριάδα από τριάδα Diffie Hellman με μη αμελητέα πιθανότητα). Συνεπώς η υπόθεση ότι το **DDH** είναι δύσκολο στις εξόδους του $\mathcal{G}$ μας δίνει ασφαλές πρωτόκολλο.

Τέλος τονίζουμε ότι τα παραπάνω ισχύουν για στατικούς αντιπάλους. Αν ο αντίπαλος μπορούσε να κάνει ενεργές επιθέσεις, να αλλάξει δηλαδή τα δεδομένα που ανταλλάσσονται στο κανάλι, θα μπορούσε να σπάσει τελείως το πρωτόκολλο. Μπορεί να εκτελέσει μια man in the middle επίθεση ως εξής: Στον Bob προσποιείται ότι είναι η Alice και στην Alice ο Bob . Οι παίχτες θα καταλήξουν να έχουν (διαφορετικά) μυστικά κλειδιά με τον αντίπαλο. Δεδομένου ότι η ανταλλαγή DH γίνεται συνήθως για να συμφωνηθεί ένα κοινό κλειδί για συμμετρικά κρυπτογραφημένη επικοινωνία ο αντίπαλος θα προωθεί απλά τα μηνύματα από τον έναν στον άλλο αφού τα διαβάσει ή τα αλλάξει, σενάριο καταστροφικό για την ιδιωτικότητα των Alice και Bob.

2.3 Κρυπτογραφία Δημοσίου Κλειδιού

Στην υποενότητα αυτή θα μελετήσουμε την κρυπτογραφία δημοσίου, δηλαδή κρυπτογραφικά σχήματα στα οποία οι συμμετέχοντες δεν έχουν ανταλλάξει ένα μυστικό κλειδί πριν από την επικοινωνία τους αλλά καθένας έχει ένα ζεύγος κλειδιών, ένα δημόσιο που το έχουν όλοι και ένα ιδιωτικό που είναι γνωστό μόνο σε αυτόν. Με αυτόν τον τρόπο μπορούμε να καταφέρουμε ιδιωτικότητα στην επικοινωνία χωρίς το υψηλό κόστος που χρειάζεται για ασφαλή διαμοιρασμό μυστικών κλειδιών για κάθε δυάδα παιχτών που θέλουν να επικοινωνήσουν. Αρχικά θα ορίσουμε τι είναι σχήματα κρυπτογραφίας δημοσίου κλειδιού και την ασφάλειά τους και θα δούμε δύο τέτοια σχήματα, τα RSA και El Gamal.

2.3.1 Σχήματα Κρυπτογράφησης Δημοσίου Κλειδιού

Ορίζουμε αρχικά τι είναι ένα σχήμα κρυπτογράφησης δημοσίου κλειδιού.

Ορισμός 2.13. *Σχήμα Κρυπτογράφησης Δημοσίου Κλειδιού είναι μία τριάδα πιθανοτικών αλγορίθμων (Gen,Enc,Dec) τέτοιοι ώστε*

- *Ο Gen με είσοδο την παράμετρο ασφαλείας 1^n δίνει έξοδο ένα ζεύγος δημοσίου κλειδιού (sk, pk) από έναν χώρο κλειδιών $\mathcal{K}$ που εξαρτάται από το n. Δηλαδή $(sk, pk) \leftarrow Gen(1^n)$*
- *Ο Enc με είσοδο ένα δημόσιο κλειδί που παράχθηκε από τον Gen και ένα μήνυμα από έναν χώρο μηνυμάτων $\mathcal{M}$ που εξαρτάται από το n δίνει ένα κρυπτοκείμενο c από έναν χώρο $\mathcal{C}$. Δηλαδή $c \leftarrow Enc_{pk}(m)$*

- *Ο Dec με είσοδο ένα ιδιωτικό κλειδί που παράχθηκε από τον Gen και ένα κρυπτοκείμενο* $c \in \mathcal{C}$ *παράγει ένα μήνυμα* $m \in \mathcal{M}$*. Δηλαδή* $m \leftarrow Dec_{sk}(c)$

για τους οποίους ισχύει ότι αν τα sk, pk *είναι έξοδος του Gen και* $m \in \mathcal{M}$ *τότε* $Dec_{sk}(Enc_{pk}(m)) = m$ *εκτός με αμελητέα ως προς το* n *πιθανότητα.*

Όπως και στην περίπτωση της ανταλλαγής κλειδιών θα πρέπει να ορίσουμε την ασφάλεια ενός τέτοιου σχήματος. Θα το κάνουμε και εδώ ως ένα παιχνίδι μεταξύ ενός challenger $\mathcal{C}$ και ενός αντιπάλου $\mathcal{A}$. Για τον ορισμό θα πρέπει να αναρωτηθούμε τι θα θέλαμε αυτός να αντικατοπτρίζει. Όπως και πριν δεν κοιτάμε σε αυτό το σημείο ενεργούς αντιπάλους αλλά περιορίζουμε την ανάλυσή μας σε στατικούς δηλαδή σε αντιπάλους που κρυφακούνε το κανάλι επικοινωνίας. Το δημόσιο κλειδί -όπως φαίνεται άλλωστε και από το όνομα του δίνουμε- το έχουν όλοι, δηλαδή το κατέχει και ο αντίπαλος. Τι σημαίνει να κρυφακούει ο αντίπαλος το κανάλι καθώς επικοινωνούν δύο παίχτες; Σημαίνει ότι παίρνει κρυπτοκείμενα που προορίζονται για τον κάτοχο του ζεύγους κλειδιων. Άρα μπορούμε να θεωρήσουμε ότι ο $\mathcal{A}$ έχει στη διάθεσή του κρυπτοκείμενα κρυπτογραφημένα με το δημόσιο κλειδί. Όμως τελικά αυτό δεν βοηθάει τον αντίπαλο αφού οι κρυπτογραφίσεις γίνονται με το δημόσιο κλειδί και ο αντίπαλος μπορεί να τις κάνει για κάθε μήνυμα της επιλογής του χωρίς να κρυφακούει το κανάλι επικοινωνίας. Συνεπώς αρκεί να επιχειρηματολογήσουμε για αντιπάλους που απλά παίρνουν το δημόσιο κλειδί.

Τέλος, τι θα θέλαμε να μην μπορεί να κάνει ο αντίπαλος; Κάτι πολύ ισχυρό είναι να μην μπορεί να ξεχωρίσει αν ένα κρυπτοκείμενο είναι κρυπτογράφηση ενός m_1 ή ενός m_2 (ίδιου μήκους) ακόμα και αν αυτός διαλέγει τα m_1, m_2. Αν δεν μπορεί να πετύχει αυτό διαισθητικά δεν μπορεί να υπολογίσει καμία συνάρτηση του m από το c που δεν είναι κοινή για όλα τα μηνύματα. Ο ορισμός αυτός λέγεται στην βιβλιογραφία μη διακρισιμότητα (indistinguishable encryption) και έχει αποδειχτεί ισοδύναμος με τον ορισμό της σημασιολογικής ασφάλειας (semantic security). Δίνουμε το παίγνιο ασφάλειας για την κρυπτογραφία δημοσίου κλειδιού.

Algorithm: $\text{PubKey}^{\text{eav}}_{\mathcal{A},\Pi}$

Input : security parameter n

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

/* τρέχουμε τον Gen για να πάρουμε ζεύγος κλειδιών */

$(sk, pk) \leftarrow \text{Gen}(1^n)$

/* δίνουμε στον $\mathcal{A}$ το pk και μας δίνει $m_0, m_1 \in \mathcal{M}$ */

$(m_0, m_1) \leftarrow \mathcal{A}(1^n, pk, \textbf{Set challenge messages})$

/* στρίβουμε κέρμα */

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

$c = \text{Enc}_{sk}(m_b)$

/* δίνουμε στον $\mathcal{A}$ το c και μαντεύει ποιο κρυπτογραφήσαμε */

$b' \leftarrow \mathcal{A}(1^n, pk, m_0, m_1, c, \textbf{Guess})$

if $b = b'$ **then**

  output 1

else

  output 0

end

Στο μοντέλο αυτό ο $\mathcal{A}$ έχει στη διάθεσή του κρυπτοκείμενα από μηνύματα της επιλογής του. Γι αυτό και το μοντέλο αυτό καλείται CPA (chosen plaintext attack). Ένα πιο ισχυρό μοντέλο είναι αυτό που λέγεται CCA (Chosen Ciphertext Attack) που ο αντίπαλος μπορεί να αποκρυπτογραφεί μηνύματα της επιλογής του, έχει δηλαδή στην κατοχή του Decrytion Oracle.

Τέλος, όπως και στην ανταλλαγή κλειδιού Diffie Hellman ορίζουμε το πλεονέκτημα του αντιπάλου στο σχήμα και πότε ένα σχήμα κρυπτογράφησης δημοσίου κλειδιού είναι ασφαλές.

Ορισμός 2.14. *Το πλεονέκτημα ενός αντιπάλου $\mathcal{A}$ ως προς το σχήμα Π = (Gen, Enc, Dec) ορίζεται ως*

$$Adv_{\mathcal{A},\Pi}(n) = |\Pr[PubKeye^{eav}_{\mathcal{A},\Pi}(n) = 1] - \frac{1}{2}|$$

Ορισμός 2.15. *Ένα σχήμα κρυπτογράφησης δημοσίου κλειδιού Π = (Γεν,Ενς,Δες) με παράμετρο ασφαλείας 1^n είναι ασφαλές ως προς στατικούς αντιπάλους (secure under the precence of eavsdropers) αν για κάθε PPT αντίπαλο $\mathcal{A}$ υπάρχει αμελητέα συνάρτηση μ ως προς το n τέτοια ώστε $Adv_{\mathcal{A},\Pi}(n) \leq \mu(n)$*

2.3.2 Το απλό RSA

Θα παρουσιάσουμε σε αυτήν την υποενότητα την απλή μορφή του σχήματος κρυπτογράφησης δημοσίου κλειδιού RSA. Το RSA οφείλεται στους Rivest, Shamir και Adellman και δημοσιε-

ύτηκε το 1978 [42]. Πρόκειται για ιστορικής σημασίας κρυπτοσυστήματος που χρησιμοποιείται ευρύτατα και σήμερα. Το πρόβλημα στο οποίο στηρίζεται η ασφάλειά του είναι αυτό της παραγοντοποίησης ακεραίων με μεγάλους πρώτους παράγοντες, ένα πρόβλημα για το οποίο μέχρι και σήμερα δεν έχουμε αποδοτικό αλγόριθμο που να το λύνει. Η μορφή που θα παρουσιάσουμε δεν είναι καθόλου ασφαλής στην πράξη αλλά παρουσιάζει με απλότητα την βασική ιδέα του συστήματος.

Θεωρούμε κατ αρχήν έναν αλγόρυθμο GenMod που με είσοδο μια παράμετρο ασφαλείας δίνει ως έξοδο ένας αριθμό $N = p \cdot q$ όπου p, q είναι πρώτοι αριθμοί με $\|p\| = \|q\| = n$. Τέτοιος αλγόριθμος είναι εύκολο να κατασκευαστεί (βλ. [49]). Παρακάτω ως ϕ θεωρούμε την συνάρτηση Euler που απεικονίζει $n \mapsto \phi(n)$ όπου $\phi(n) = |\mathbb{Z}_n^*|$, το πλήθος της πολλαπλασιαστικής ομάδας $|\mathbb{Z}_n^*|$.

Το απλό RSA δουλεύει ως εξής.

Algorithm: Gen

Input : security parameter n

Output: (sk, pk)

`/* τρέχουμε τον GenMod για να πάρουμε modulus` N `*/`

1 $N \leftarrow \text{GenMod}(1^n)$

2 $e \leftarrow_R \mathbb{Z}_{\phi(N)}$

`/* υπολογίζουμε τον πολλαπλασιαστικό αντίστροφο του` e `*/`

3 $d \leftarrow e^{-1} \mod \phi(n)$

4 $sk \leftarrow (N, e, d)$

5 $pk \leftarrow (N, e)$

Σημειώνουμε ότι ο ακριβής τρόπος επιλογής του Modulus N και του e είναι πολύ σημαντικός για την ασφάλεια του συστήματος αλλά επειδή σκοπός της υποενότητας είναι να εκθέσει τη βασική ιδέα για το RSA παραβλέπουμε αυτές τις (σημαντικές) λεπτομέρειες.

Algorithm: Enc

Input : N, e, m με $m \in \mathbb{Z}_N$

Output: $c \in \mathbb{Z}_N$

1 $c \leftarrow m^e \mod N$

Algorithm: Dec

Input : N, e, c με $c \in \mathbb{Z}_N$

Output: $m \in \mathbb{Z}_N$

1 $m \leftarrow c^d \mod N$

Ας επαληθεύσουμε αρχικά την ορθότητα του συστήματος μας. Έχουμε

$$\begin{aligned}\mathrm{Dec}_{N,e,d}(\mathrm{Enc}_{N,e}(m)) &= \mathrm{Dec}_{N,e,d}(m^e \mod N)\\ &= m^{ed} \mod N\\ &= m^1 \mod N\\ &= m \mod N\end{aligned}$$

όπου η τρίτη ισότητα ισχύει γιατί $ed = 1 \mod \phi(n)$ (από κατασκευή τους από Gen).

Όπως είπαμε το σύστημα όπως παρουσιάζεται δεν είναι καθόλου ασφαλές. Ο πιο απλός τρόπος να το δει κανείς αυτό είναι ότι το σύστημα δεν είναι πιθανοτικό και κανένο τέτοια σύστημα δεν μπορεί να ελπίζει ότι είναι ασφαλές [25]. Για πληρότητα αναφέρουμε και την υπόθεση RSA στην οποία στηρίζουν την ασφαλειά τους όλα τα σχετικά κρυπτοσυστήματα.

Υπόθεση RSA: Υπάρχει αλγόριθμος Gen για τον RSA τέτοιος ώστε για κάθε PPT $\mathcal{A}$ υπάρχει αμελητέα συνάρτηση μ ως προς την παράμετρο ασφαλείας n τέτοια ώστε η πιθανότητα ο $\mathcal{A}$ με είσοδο N, e που παράγει ο Gen και $y \in \mathbb{Z}_N^*$ να δώσει έξοδο x τέτοια ώστε $x = y^d \mod N$ είναι μικρότερη από $\mu(n)$.

2.3.3 Το σχήμα El Gamal

Θα παρουσιάσουμε το σχήμα κρυπτογράφησης δημοσίου κλειδιού El Gamal. Πρόκειται ουσιαστική για την μεταφορά της ανταλλαγής κλειδιού Diffie Hellman στο πλαίσιο της κρυπτογραφίας δημοσίου κλειδιού. Το σύστημα αυτό χρησιμοποιείται ευρύτατα σε ηλεκτρονικές ψηφοφορίες γεγονός που πηγάζει από την απλότητά του και από τις ομομορφικές ιδιότητες που έχει. Θα αποδείξουμε την ασφάλειά του ως προς την υπόθεση DDH μιας και η απόδειξη αυτή είναι χαρακτηριστικό υπόδειγμα για το πως δουλεύουν τέτοιες αποδείξεις. Παρακάτω βλέπουμε το σχήμα.

Algorithm: Gen

Input : security parameter n

Output: $(G, q, g), x, h$

1 Δίνει μια περιγραφή ομάδας G μεγέθους q και γεννήτορα g με $\|q\| = n$

2 $x \leftarrow_R \mathbb{Z}_q$

3 $h \leftarrow g^x$

Ο χώρος μηνυμάτων είναι $\mathcal{M} = G$ και ο χώρος κρυπτοκειμένων είναι $\mathcal{C} = G^2$. Παρατηρήστε στα επόμενα ότι η κρυπτογράφιση είναι πιθανοτική.

Algorithm: Enc

Input : G, q, g, h, m

Output: c

1 $r \leftarrow_R \mathbb{Z}_q$

2 $c \leftarrow (g^r, m \cdot h^r)$

Algorithm: Dec

Input : $G, q, g, x, h, c = (c_1, c_2)$

Output: c

1 $m = c_2 \cdot (c_1^x)^{-1}$

Παρατηρούμε το εξής: επί της ουσίας ο ένας παίχτης (ο παραλήπτης) κατασκευάζει ένα μερίδιο κλειδιού Diffie Hellman και το δημοσιοποιεί. Όποιος θέλει να επικοινωνήσει μαζί του κατασκευάζει ένα δεύτερο μερίδιο και 'μασκάρει' το μήνυμα πολλαπλασιάζοντας το μήνυμα με το κοινό κλειδί που προκύπτει. Το κρυπτοκείμενο αποτελείται από το μερίδιο Diffie Hellman και από το μασκαρισμένο μήνυμα.

Ας επαληθεύσουμε την ορθότητα του συστήματος μας. Έχουμε

$$\begin{aligned}
\mathrm{Dec}_{sk}(\mathrm{Enc}_{pk}(m)) &= \mathrm{Dec}_{sk}((g^r, m \cdot h^r)) \\
&= m \cdot h^r \cdot ((g^r)^x)^{-1} \\
&= m \cdot h^r \cdot ((g^{rx})^{-1} \\
&= m \cdot h^r \cdot ((g^{xr})^{-1} \\
&= m \cdot h^r \cdot ((g^x)^r)^{-1} \\
&= m \cdot h^r \cdot (h^r)^{-1} \\
&= m
\end{aligned}$$

Πριν κοιτάξουμε την απόδειξη ασφάλειας ας κοιτάξουμε λίγο τις ομομορφικές ιδιότητες του. Έστω δύο κρυπτοκείμενα El Gamal

$$E_{pk}(m_1; r_1) = (g^{r_1}, m_1 \cdot h^{r_1})$$

και

$$E_{pk}(m_2; r_2) = (g^{r_2}, m_2 \cdot h^{r_2})$$

Τότε πολλαπλασιάζοντας τα κρυπτοκείμενα (κατά μέλη φυσικά) παίρνουμε

$$\begin{aligned}
E_{pk}(m_1; r_1) \cdot E_{pk}(m_2; r_2) &= (g^{r_1} g^{r_2}, m_1 \cdot h^{r_1} \cdot m_2 \cdot h^{r_2}) \\
&= (g^{r_1 + r_2}, m_1 \cdot m_2 \cdot h^{r_1 + r_2}) \\
&= E_{pk}(m_1 \cdot m_2; r_1 + r_2)
\end{aligned}$$

Όπως θα δούμε παρακάτω τέτοιες ιδιότητες είναι πολύ επιθυμητές στα συστήματα ηλεκτρονικής ψηφοφορίας. Επίσης παρατηρήστε ότι αν πολλαπλασιάσουμε με $E_{pk}(1; r_2)$ παίρνουμε $E_{pk}(m_1; r_1 + r_2)$ δηλαδή *επανακρυπτογραφούμε* το αρχικό κρυπτοκείμενο.

Διατυπώνουμε κάπως άτυπα πως θεωρούμε ένα πρόβλημα δύσκολο αν για κάθε PPT αλγόριθμο η πιθανότητα να το λύσει φράσσεται από μία αμελητέα συνάρτηση ως προς κάποια παράμετρο.

Θεώρημα 2.1. *Αν το **DDH** είναι δύσκολο για τις ομάδες που παράγει η Gen τότε το σχήμα El Gamal είναι ασφαλές σχήμα κρυπτογραφίας δημοσίου κλειδιού.*

Απόδειξη. Έστω Π το El Gamal και αντίπαλος $\mathcal{A}$. Θεωρούμε $\tilde{\Pi}$ ίδιο με το Π αλλά με τη διαφορά ότι στο κρυπτοκείμενο διαλέγει τυχαίο z και δίνει $(g^r, m \cdot g^z)$. Αφού το z επιλέγεται τυχαία από το $\mathbb{Z}_q$ το δεύτερο συστατικό του κρυπτοκειμένου είναι ένα τυχαίο, ομοιόμορφα κατανεμημένο στοιχείο της ομάδας ανεξάρτητο από το πρώτο συστατικό. Συνεπώς ο $\mathcal{A}$ όταν παίζει το παιχνίδι $\text{PubKey}^{eav}_{\mathcal{A},\tilde{\Pi}}$ δεν μπορεί να ξεχωρίσει αν το κρυπτοκείμενο που λαμβάνει αντιστοιχεί στο m_1 ή m_2 καλύτερο από το να διαλέξει στην τύχη. Συνεπώς

$$\Pr[\text{PubKey}^{eav}_{\mathcal{A},\tilde{\Pi}}(n) = 1] = \frac{1}{2} \tag{2.1}$$

Έστω τώρα ότι ο $\mathcal{A}$ νικάει το $\text{PubKey}^{eav}_{\mathcal{A},\Pi}$ με κάποια πιθανότητα. Θα κατασκευάσουμε αλγόριθμο $\mathcal{S}$ που νικάει το **DDH** . Ο αλγόριθμος δουλεύει ως εξής:

Algorithm: $\mathcal{S}$

Input : G, q, g, g^x, g^y, g^z

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

/* Δίνει 1 αν $z = xy$ και 0 αλλιώς */

Θέτει $pk \leftarrow (G, q, g, g^x)$ και το δίνει στον $\mathcal{A}$

$(m_0, m_1) \leftarrow \mathcal{A}(pk, \textbf{Set challenge Messages})$

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

$c \leftarrow (g^y, g^z \cdot m_b)$

/* παρατηρήστε ότι αν είναι τριάδα Diffie Hellman τότε $z = xy$ και το c είναι El Gamal κρυπτογράφηση. Αλλιώς είναι τυχαίο και αντιστοιχεί στο $\tilde{\Pi}$ */

$b' \leftarrow \mathcal{A}(pk, m_0, m_1, c, \textbf{Answer})$

if $b = b'$ **then**

  output 1

else

  output 0

end

Όπως φαίνεται και στο αντίστοιχο σχόλιο αν g^x, g^y, g^z είναι επιλεγμένα τυχαία και ανεξάρτητα ο $\mathcal{A}$ παίζει το παιχνίδι $\text{PubKey}^{eav}_{\mathcal{A},\tilde{\Pi}}$ και από την 2.1 έχουμε ότι

$$\Pr[\mathcal{S}(G, q, g, g^x, g^y, g^z) = 1] = \Pr[\text{PubKey}^{eav}_{\mathcal{A},\tilde{\Pi}}(n) = 1] = \frac{1}{2}$$

Αντίθετα αν $z = xy$ τότε ο $\mathcal{S}$ δίνει το αποτέλεσμα του πειράματος $\text{PubKey}^{eav}_{\mathcal{A},\Pi}(n)$. Έστω $e(n)$ η πιθανότητα ο $\mathcal{A}$ να νικάει το σύστημα. Τότε

$$\Pr[\mathcal{S}(G, q, g, g^x, g^y, g^{xy}) = 1] = \Pr[\text{PubKey}^{eav}_{\mathcal{A},\Pi}(n) = 1] = e(n)$$

Από την υπόθεση **DDH** έχουμε ότι για κάθε PPT αλγόριθμο, άρα και για τον $\mathcal{S}$ υπάρχει αμελητέα συνάρτηση ως προς n, έστω μ τέτοια ώστε

$$|\Pr[\mathcal{S}(G,q,g,g^x,g^y,g^{xy})=1]-\Pr[\mathcal{S}(G,q,g,g^x,g^y,g^z)=1]|\leq\mu(n)$$

Συνεπώς έχουμε

$$|e(n)-\frac{1}{2}|\leq\mu(n)$$

και παίρνουμε ότι

$$\mathrm{Adv}_{\mathcal{A},\Pi}(n)\leq\mu(n)$$

που δίνει και το ζητούμενο. ■

Τέλος, επειδή χρησιμοποιείται από αρκετά συστήματα, αναφέρουμε και μια παραλλαγή του El Gamal. Μπορούμε αντί να έχουμε $m\in\mathbb{G}$ να έχουμε $m\in\mathbb{Z}_q$ και να κωδικοποιούμε τα μηνύματα ως g^m και χρησιμοποιούμε το κανονικό El Gamal με αυτά. Αποκρυπτογραφούμε κανονικά και παίρνουμε g^m. Για να πάρουμε το m πρέπει να λύσουμε διακριτό λογάριθμο αλλά φροντίζουμε το μέγεθος του $\mathcal{M}$ να είναι μικρό και έτσι μπορούμε γρήγορα να υπολογίσουμε το m με bruteforce, δοκιμάζοντας δηλαδή όλα τα δυνατά μηνύματα .

2.4 Ψηφιακές Υπογραφές

Ένα άλλο πολύ βασικό κρυπτογραφικό primitive είναι οι ψηφιακές υπογραφές. Αυτές αποτελούν το ψηφιακό αντίστοιχο των πραγματικών υπογραφών. Όταν κάποιος υπογράψει ψηφιακά ένα μήνυμα κάθε άλλος μπορεί να ελέγξει ότι το συγκεκριμένο μήνυμα υπογράφηκε από τον κάτοχο της υπογραφής και όχι από κάποιον άλλο. Οι εφαρμογές που έχουν είναι, όπως μπορεί να φανταστεί κανείς, αμέτρητες. Θα ορίσουμε τα σχήματα και το πλαίσιο ασφάλειας και θα δούμε τις υπογραφές RSA-FDH . Στην πορεία θα χρειαστεί και να αναφερθούμε στο μοντέλο του τυχαίου μαντείου, ένα αμφιλεγόμενο και πολυσυζητημένο μοντέλο για σχεδιασμό πρωτοκόλλων. Τα παρακάτω, όπως και το υπόλοιπο κεφάλαιο, βασίζονται στο [34] Μια πιο πλήρης μελέτη για της ψηφιακές υπογραφές βρίσκεται εδώ [33].

2.4.1 Σχήματα Ψηφιακών Υπογραφών

Ορισμός 2.16. *Ένα Σχήμα Ψηφιακών Υπογραφών είναι μία τριάδα πιθανοτικών αλγορίθμων (Gen,Sign,Vrfy) τέτοιοι ώστε*

- *Ο Gen με είσοδο την παράμετρο ασφαλείας 1^n δίνει έξοδο ένα ζεύγος δημοσίου κλειδιού (sk,vk) από έναν χώρο κλειδιών $\mathcal{K}$ που εξαρτάται από το n. Δηλαδή $(sk,vk)\leftarrow Gen(1^n)$. Θα αναφερόμαστε στο sk ως το κλειδί υπογραφής και στο vk ως το κλειδί επαλήθευσης.*

- *Ο Sign με είσοδο ένα δημόσιο κλειδί που παράχθηκε από τον Gen και ένα μήνυμα από έναν χώρο μηνυμάτων $\mathcal{M}$ που εξαρτάται από το n δίνει μια υπογραφή σ από έναν χώρο $\mathcal{S}$. Δηλαδή $\sigma \leftarrow Sign_{sk}(m)$*
- *Ο Vrfy με είσοδο ένα κλειδί επαλήθευσης vk που παράχθηκε από τον Gen , ένα μήνυμα m και μια υπογραφή σ απαντάει με ένα bit 1 ή 0 που εκφράζει την ορθότητα ή μη της υπογραφής*

για τους οποίους ισχύει ότι αν τα sk, vk είναι έξοδος του Gen και $m \in \mathcal{M}$ τότε $Vrfy_{vk}(Sign_{sk}(m)) =$ 1 εκτός ίσως με αμελητέα ως προς το n πιθανότητα.

Ορίζουμε στη συνέχεια το παιχνίδι με το οποίο αξιολογούμε ένα σχήμα ψηφιακών υπογραφών. Ουσιαστικά η απαίτησή μας από ένα τέτοιο σχήμα είναι μια υπογραφή να μην μπορεί να παραχαραχθεί, δηλαδή ένας αντίπαλος να μην μπορεί να υπογράψει ένα μήνυμα της επιλογής του. Όμως οι κρυπτογραφικές απαιτήσεις πολλών πρωτοκόλλων που χρησιμοποιούν ψηφιακές υπογραφές δεν καλύπτονται από το παραπάνω πλαίσιο. Χρειαζόμαστε κάτι πολύ πιο ισχυρό. Θα θέλαμε ένα σχήμα ψηφιακών υπογραφών να χαρακτηρίζεται ασφαλές αν ένας αντίπαλος δεν μπορεί να παράξει υπογραφή για οποιοδήποτε μήνυμα για το οποίο δεν έχει ήδη δει υπογραφεί. Αυτό το ορίζουμε με το παρακάτω παίγνιο.

Algorithm: $\text{SigForge}^{\text{eav}}_{\mathcal{A},\Pi}$

Input : security parameter n

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

`/* τρέχουμε τον Gen για να πάρουμε ζεύγος κλειδιών */`

1 $(sk, vk) \leftarrow \text{Gen}(1^n)$

`/* δίνουμε στον` $\mathcal{A}$ `το` pk `και μαντείο υπογραφών */`

`/* ορίζουμε` Q `το σύνολο των ερωτημάτων που έκανε ο` $\mathcal{A}$ `στο μαντείο */`

2 $(m, \sigma) \leftarrow \mathcal{A}^{\text{Sign}_{sk}(\cdot)}(vk)$

3 **if** $Vrfy_{vk}(m, \sigma) = 1$*και* $m \notin Q$ **then**

4 | output 1

5 **else**

6 | output 0

7 **end**

Το μαντείο ουσιαστικά μοντελοποιεί την δυνατότητα του αντιπάλου να βλέπει υπογραφές που έχουν κατατεθεί με το κλειδί sk. Για την ακρίβεια μοντελοποιεί κάτι πολύ πιο ισχυρό. Ο $\mathcal{A}$ μπορεί να πάρει υπογραφή σε όποιο μήνυμα ζητήσει. Αν καταφέρει να φτιάξει μία μόνο έγκυρη υπογραφή για κάποιο μήνυμα που δεν έχει ήδη μάθει υπογραφή νικάει.

Ορισμός 2.17. *Ένα σχήμα ψηφιακών υπογραφών Π = (Gen,Sign,Vrfy) με παράμετρο ασφαλείας 1^n είναι ασφαλές ως προς επίθεση υπαρξιακής παραχάραξης από προσαρμοστικά επιλεγμένο μήνυμα (existentially unforgeable under an adaptive chosen message attack) αν για κάθε PPT αντίπαλο $\mathcal{A}$ υπάρχει αμελητέα συνάρτηση μ ως προς την παράμετρο ασφαλείας*

n *τέτοια ώστε* $\epsilon \Pr[SigForge^{eav}_{\mathcal{A},\Pi}(n) = 1] \leq \mu(n)$

2.4.2 Το μοντέλο του τυχαίου μαντείου

Πριν δούμε τις υπογραφές RSA-FDH πρέπει να εξηγήσουμε συνοπτικά τι είναι το μοντέλο του τυχαίου μαντείου. Πέρα από την ασφάλεια των ψηφιακών υπογραφών το μοντέλο αυτό θα μας χρειαστεί και αργότερα όταν μιλήσουμε για Non Interactive Zero Knowledge Proofs of Knowledge.

Πολύ συχνά για να κατασκευάσουμε ένα ασφαλές κρυπτογραφικό πρωτόκολλο βοηθάει να έχουμε στη διάθεσή μας μια τυχαία συνάρτηση. Όμως τυχαίες συναρτήσεις δεν μπορούμε να κατασκευάσουμε στην πραγματικότητα. Το μοντέλο του τυχαίου μαντείου που εισήχθηκε από τους Bellare, Rogaway [6] λειτουργεί ως εξής: Κάθε παίχτης που συμμετέχει σε ένα πρωτόκολλο έχει στη διάθεσή του πρόσβαση σε ένα μαντείο (δεν βλέπει δηλαδή τη λειτουργία του αλλά έχει πρόσβαση μαύρου κουτιού σε αυτό) που υπολογίζει μια τυχαία συνάρτηση $f : A \to B$ όπου A, B πεπερασμένα υποσύνολα του $\{0,1\}^*$. Λέγοντας τυχαία εννοούμε ομοιόμορφα κατανεμημένη στο σύνολο των συναρτήσεων $F = \{f | f : A \to B\}$.

Φυσικά δεν μπορούμε να κατασκευάσουμε στην πραγματικότητα τέτοια συνάρτηση. Αυτό που κάνουμε είναι όταν υλοποιούμε το πρωτόκολλο να αντικαθιστούμε κάθε αίτημα για υπολογισμό του $f(x)$ από το μαντείο από μια κλήση σε κάποια κατάλληλα τροποποιημένη κρυπτογραφική συνάρτηση κατακερματισμού. Αυτό γίνεται με το σκεπτικό ότι τέτοιες συναρτήσεις είναι απρόβλεπτες για εμάς και το αποτέλεσμά τους μοιάζει σε εμάς για τυχαίο (όμως δεν είναι).

Για να αποδείξουμε την ασφάλεια στο μοντέλο του τυχαίου μαντείου θεωρούμε τα εξής:

- *συνέπεια:* Αν το μαντείο ρωτηθεί δύο φορές για την απεικόνιση του x θα επιστρέψει την ίδια απάντηση.
- *εκμαιευσιμότητα:* Αν ο αντίπαλος ρωτήσει κάτι το μαντείο ο προσομοιωτής το μαθαίνει
- *προγραμματισιμότητα:* Τις απαντήσεις στις ερωτήσεις του αντιπάλου τις απαντάει ο προσομοιωτής φροντίζοντας οι τιμές που δίνει να είναι σωστά κατανεμημένες

Όταν το μαντείο αρχικοποιηθεί από κάποια κατάλληλη συνάρτηση τα παραπάνω δεν ισχύουν οπότε δεν ισχύουν και τα αποτελέσματα σε μια απόδειξη ασφάλεια που κατασκευάστηκε στο συγκεκριμένο μοντέλο. Για την ακρίβεια έχουν κατασκευαστεί από τους Canetti, Goldreich και Halevi [9] τεχνητά συστήματα που είναι ασφαλή στο μοντέλο τυχαίου μαντείου αλλά ανασφαλή με το που αυτό αρχικοποιηθεί από οποιαδήποτε συνάρτηση.

Αυτό που μπορούμε να ισχυριστούμε με ασφάλεια από μια απόδειξη σε αυτό το μοντέλο είναι ότι το πρωτόκολλό μας δεν έχει κάποιο σημαντικό σχεδιαστικό πρόβλημα και αν σπάσει αυτό θα οφείλεται στην hash function που χρησιμοποιήσαμε. Σημειώνουμε πως μέχρι και σήμερα, σε πραγματικά κρυπτοσυστήματα δεν έχει καταγραφεί επίθεση που να οφείλεται σε αυτό, χωρίς φυσικά αυτό να σημαίνει ότι δεν υπάρχουν και δεν θα βρεθούν.

Γενικά το μοντέλο αυτό χρησιμοποιείται όταν δεν έχουμε απόδειξη στο κανονικό μοντέλο ή όταν μπορούμε με αυτό να κατασκευάσουμε συστήματα με σημαντικά καλύτερη απόδοση.

2.4.3 Σχήμα ψηφιακών υπογραφών RSA-FDH

Το σχήμα υπογραφών RSA-FDH είναι η απλούστερη μορφή των υπογραφών RSA που χρησιμοποιούνται ευρύτατα. Η ασφάλειά του αποδεικνύεται κάτω από την υπόθεση αντιστροφής RSA στο μοντέλο του τυχαίου μαντείου. Παρακάτω παρουσιάζουμε αυτό το σχήμα. Στη συνέχεια θα δώσουμε και μια διαίσθηση για την απόδειξη ασφάλειάς του.

Algorithm: Gen

Input : security parameter n

Output: (sk, pk)

`/* τρέχουμε τον GenMod για να πάρουμε modulus` N `*/`

$N \leftarrow \text{GenMod}(1^n)$

$e \leftarrow_R \mathbb{Z}_{\phi(N)}$

`/* υπολογίζουμε τον πολλαπλασιαστικό αντίστροφο του` e `*/`

$d \leftarrow e^{-1} \mod \phi(n)$

$sk \leftarrow (N, e, d)$

$vk \leftarrow (N, e)$

Όπως και στο σχήμα δημοσίου ιδιωτικού κλειδιού RSA ο ακριβής τρόπος επιλογής του Modulus N και του e είναι πολύ σημαντικός για την ασφάλεια του συστήματος. Εμείς απλά θα θεωρήσουμε ότι το πρόβλημα της αντιστροφής RSA είναι δύσκολο για τα output του Gen.

Algorithm: Sign

Input : N, d, m με $m \in \mathbb{Z}_N$, oracle σε $H : \mathcal{M} \to \mathbb{Z}_N^*$

Output: $\sigma \in \mathbb{Z}_N^*$

$\sigma \leftarrow H(m)^d \mod N$

Algorithm: Vrfy

Input : N, e, m, σ με $c \in \mathbb{Z}_N$

Input : N, e, m, σ με $m, \sigma \in \mathbb{Z}_N$, oracle σε $H : \mathcal{M} \to \mathbb{Z}_N^*$

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

$m' \leftarrow H(m)$

if $\sigma^e \equiv m' \mod N$ **then**

  output 1

else

  output 0

end

Για την ορθότητα του συστήματος έχουμε

$$\mathrm{Vrfy}_{N,e}^{H(\cdot)}(\mathrm{Sign}_{N,d}^{H(\cdot)}(m)) = \mathrm{Vrfy}_{N,e}^{H(\cdot)}(m, H(m)^d \mod N) = 1$$

αφού $m' = H(m)$ και $H(m)^{ed} \equiv H(m) \mod N$.

Κατ΄ αρχήν ας δούμε γιατί χρησιμοποιήσαμε την H. Αν δεν την είχαμε το σύστημα θα ήταν τελείως ανασφαλές. Ένας αντίπαλος θα μπορούσε να διαλέξει τυχαία ένα στοιχείο σ, να υπολογίσει $m = \sigma^e$ και το ζεύγος m, σ θα ήταν παραχάραξη. Επίσης αν ήθελε υπογραφή για ένα συγκεκριμένο μήνυμα m θα μπορούσε να υπολογίζει m_1, m_2 τέτοια ώστε $m \equiv m_1 \cdot m_2 \mod N$ και να ρώταγε το μαντείο για υπογραφές των m_1, m_2. Τότε, αν έπαιρνε σ_1, σ_2 θα μπορούσε να υπολογίσει υπογραφή σ για το m πολλαπλασιάζοντάς τες $\mod N$. Για του λόγου το αληθές

$$\sigma_1 \cdot \sigma_2 \equiv m_1^d \cdot m_2^d \equiv m^d \mod N = \mathrm{Sign}_{N,d}(m)$$

Συνεπώς το σύστημα χωρίς την H θα ήταν άχρηστο.

Ας δούμε τώρα πως μπορούμε να επιχειρηματολογήσουμε για την ασφάλεια του συστήματος στο τυχαίο μαντείο.

Θεώρημα 2.2. *Αν το πρόλημα RSA είναι δύσκολο για την έξοδο του Gen και η H μοντελοποιείται ως τυχαίο μαντείο τότε το σχήμα RSA-FDH είναι ασφαλές (με την έννοια που ορίστηκε παραπάνω) σχήμα ψηφιακών υπογραφών.*

Απόδειξη. (σχέδιο) Θεωρούμε $\mathcal{A}$ που νικάει το SigForge. Θα κατασκευάσουμε $\mathcal{S}$ που αντιστρέφει RSA που παράγεται από τον Gen. Θυμίζουμε ότι ο $\mathcal{S}$ πρέπει να ελέγχει το μαντείο. Μπορούμε να υποθέσουμε ότι για την παραχάραξη ο $\mathcal{A}$ θα ρωτήσει το τυχαίο μαντείο για την εικόνα του m πιο πριν. Δεν θα μπορούσε αλλιώς αφού η υπογραφή εξαρτάται άμεσα από το random oracle. Ο $\mathcal{S}$ παίρνει είσοδο N, e, y και θέλει να υπολογίζει το $y^{e^{-1}} \mod N$. Θα καλέσει τον $\mathcal{A}$ και θα του δώσει ως vk το N, e. Ο $\mathcal{A}$ έχει πρόσβαση σε μαντείο υπογραφών. Στην προσομοίωση είναι δουλειά του προσομοιωτή να απαντάει σε αυτά τα αιτήματα. Πώς όμως θα το κάνει χωρίς τον αντίστροφο του e; Η απάντηση κρύβεται στον έλεγχο του τυχαίου μαντείου. Επίσης ο $\mathcal{S}$ πρέπει να απαντάει στις ερωτήσεις προς το τυχαίο μαντείο. Ας δούμε πως τα κάνει αυτά. Ο $\mathcal{S}$ έχει έναν πίνακα με τριάδες $(\cdot, \cdot, \cdot)$ που τις γεμίζει σταδιακά, αρχικά κενό.

Απαντήσεις σε ερώτημα υπογραφής Όταν ζητείται υπογραφή για το m, αν το m δεν είναι το πρώτο στοιχείο κάποιας τριάδας διαλέγει ένα τυχαίο στοιχείο $\sigma \leftarrow_R \mathbb{Z}_N^*$ και υπολογίζει $y = \sigma^e \mod N$. Βάζει την τριάδα (m, y, σ) στον πίνακα και επιστρέφει σ. Αν υπάρχει δίνει το τρίτο στοιχείο της αντίστοιχης τριάδας (εδώ διατηρείται το consistency).

Απαντήσεις σε ερώτημα τυχαίου μαντείου Δουλεύει ακριβώς όπως στα ερωτήματα υπογραφής αλλά δίνει τα δεύτερα συστατικά των τριάδων

Έστω λοιπόν ότι ο $\mathcal{A}$ θα κάνει q ερωτήσεις. Δεδομένου ότι ο $\mathcal{A}$ είναι πολυωνιμικού χρόνου το q φράσσεται από ένα πολυώνυμο ως προς το m. Αυτό που κάνει ο $\mathcal{S}$ είναι στην αρχή του παιχνιδιού να διαλέγει τυχαία έναν αριθμό $i \in [q]$ και να ελπίζει ότι ο $\mathcal{A}$ θα διαλέξει να πλαστογραφήσει το μήνυμα που ρώτησε την i-οστή φορά. Ο $\mathcal{S}$ αν ρωτηθεί για υπογραφή την i-οστή φορά (η αν ξαναρωτηθεί το ίδιο αργότερα) δηλώνει αποτυχία αλλιώς επιστρέφει στον $\mathcal{A}$ το στοιχείο y που έχει ως είσοδο. Παρατηρούμε ότι η αντίστοιχη υπογραφή θα είναι ο αντίστροφος που ψάχνει ο $\mathcal{S}$.

Αν λοιπόν σταθεί τυχερός και ο $\mathcal{A}$ δώσει έγκυρη υπογραφή για το i-οστό μήνυμα δίνει ως έξοδο την υπογραφή του $\mathcal{A}$ και νικάει με την ίδια πιθανότητα που θα νίκαγε ο $\mathcal{A}$. Αν όμως είναι άτυχος; Η απάντηση είναι ότι δεν μας πειράζει. Αυτό γιατί με πιθανότητα $\frac{1}{q}$ θα σταθεί τυχερός, δηλαδή η πιθανότητα επιτυχίας του μειώνεται πολυωνυμικά ως προς αυτή του $\mathcal{A}$. Όμως η πιθανότητα επιτυχίας του $\mathcal{S}$ από υπόθεση είναι αμελητέα και συνεπώς και αυτή του $\mathcal{A}$ αφού είναι πολυωνυμικά μεγαλύτερη. Συμπεραίνουμε ότι το σχήμα είναι ασφαλές. ■

2.5 Διαμοιρασμός Μυστικού και Κατανεμημένο El Gamal

Ας θεωρήσουμε το εξής σενάριο: Μια ομάδα από n άτομα δεν εμπιστεύονται ο ένας τον άλλον αλλά πρέπει να έχουν ένα κοινό μυστικό (για παράδειγμα ένα μυστικό κλειδί υπογραφής). Κανένας λοιπόν δεν πρέπει να ξέρει το μυστικό αλλά πρέπει αν μαζευτούν όλοι μαζί να μπορούν να ανακατασκευάσουν το μυστικό. Πώς μπορούν να το πετύχουν αυτό; Την απάντηση έδωσε ο Shamir [47] που έλυνε το πρόβλημα στην γενικότερη μορφή που θα δούμε παρακάτω. Αυτό το primitive χρειάζεται σε πολλές κρυπτογραφικές εφαρμογές με πιο σημαντική τον ασφαλή υπολογισμό πολλών μελών (General Secure Multi Palti Computation). Χρησιμοποιείται ευρύτατα και σε ηλεκτρονικές ψηφοφορίες και γενικά σε κάθε σενάριο που θέλουμε να έχουμε *κατανεμημένη εμπιστοσύνη*.

2.5.1 Το σενάριο

Έχουμε ένα μυστικό s. Θέλουμε να το μοιράσουμε σε n πρόσωπα με τρόπο τέτοιο ώστε οποιαδήποτε υποσύνολο από αυτούς μεγέθους μικρότερο από t αν συνεργαστεί να μην μπορεί να πάρει καμία πληροφορία για το s και οποιοδήποτε υποσύνολο μεγέθους τουλάχιστον t αν συνεργαστεί να μπορεί να ανακατασκευάσει το μυστικό s. Αυτό το ονομάζουμε (t, n) διαμοιρασμό μυστικού.

2.5.2 Διαμοιρασμός Μυστικού Shamir

Θα χρησιμοποιήσουμε το γεγονός ότι κάθε πολυώνυμο βαθμού t πάνω από ένα σώμα μπορεί να ανακατασκευαστεί αν ξέρουμε t σημεία του.

Θεώρημα 2.3. *Έστω σώμα $\mathbb{F}$ και σημεία $\{(x_1, y_1)\}_{i=1}^{t}$ με $x_i, y_i \in \mathbb{F}$ και $x_k \neq x_l$ για κάθε $k \neq l$ και $k, l \in \{1, 2, \ldots, t\}$. Τότε υπάρχει μοναδικό πολυώνυμο $p \in \mathbb{F}[x]$ βαθμού $t-1$ τέτοιο ώστε $\forall i \in \{1, 2, \ldots, t\}$ $p(x_i) = y_i$.*

Απόδειξη. Ορίζουμε για κάθε $i \in \{1, \ldots, t\}$ τα πολυώνυμα

$$\lambda_i(x) = \frac{\prod_{j \in \{1,\ldots,t\} \setminus \{i\}} (x - x_j)}{\prod_{j \in \{1,\ldots,t\} \setminus \{i\}} (x_i - x_j)}$$

Οι ποσότητες αυτές ορίζονται αφού τα $\{x_i\}_i \in 1, \ldots, n$ είναι διακεκριμένα. Επίσης ισχύει ότι για κάθε $j \neq i$ $\lambda_i(x_j) = 0$ και $\lambda_i(x_i) = 1$. Κατασκευάζουμε λοιπόν το πολυώνυμο p ως εξής

$$p(x) = \sum_{i=1}^{t} y_i \cdot \lambda_i(x)$$

Το πολυώνυμα αυτό είναι βαθμού $t-1$ και ισχύει ότι $p(x_i) = y_i$. Επίσης είναι το μοναδικό πολυώνυμο βαθμού $t-1$ που έχει αυτές τις τιμές γιατί αν υπήρχε άλλο, έστω p' τότε οι τιμές $x_1, \ldots, x_t$ θα ήταν ρίζες του πολυωνύμου $p - p'$ που έχει βαθμό το πολύ $t-1$. Ένα πολυώνυμο όμως βαθμού το πολύ $t-1$ με t διακεκριμένες ρίζες είναι ταυτοτικά 0 και συνεπώς $p = p'$. ■

Η συγκεκριμένη ιδιότητα μπορεί να χρησιμοποιηθεί ως εξής. Έστω πεπερασμένο σώμα $\mathbb{F}$ και μυστικό $s \in \mathbb{F}$. Ένας dealer θέλει να μοιράσει το μυστικό s στους n παίχτες ως εξής: Αρχικά διαλέγει $t-1$ τυχαία στοιχεία a_i του σώματος και κατασκευάζει το πολυώνυμο $p(x) = s + \sum_{i=1}^{t-1} a_i x^i$. Διαλέγει τυχαία $x_1, \ldots x_n \in \mathbb{F}$ και τα δίνει στους παίχτες. Αυτά δεν είναι μυστικά. Για κάθε x_i υπολογίζει $y_i = p(x_i)$ και δίνει στον i-οστό παίχτη το y_i το οποίο εκείνος κρατάει μυστικό.

Αν t παίχτες θέλουν να ανακατασκευάσουν το μυστικό s δεν έχουν παρά να ανακατασκευάσουν το p όπως περιγράφεται παραπάνω και να υπολογίσουν την τιμή $s = p(0)$.

Μένει να δείξουμε ότι $t-1$ παίχτες δεν μαθαίνουν καμία πληροφορία για το μυστικό s. Χωρίς βλάβη της γενικότητας, έστω ότι οι παίχτες με μερίδια $\{s_i\}_{i \in \{1,\ldots,t-1\}}$ συνεργάζονται για να μάθουν κάτι για το μυστικό s. Υπάρχουν $\mathbb{F}^{t-1}$ πιθανά πολυώνυμα που μπορεί να διαλέξει ο Dealer για κάθε s και επιλέγει ομοιόμορφα. Επίσης υπάρχουν $\mathbb{F}^{t-1}$ πιθανά μέρη που θα μοιράσει στους $t-1$ παίχτες και ισχυριζόμαστε ότι κάθε τέτοια πλειάδα απεικονίζεται μονοσήμαντα σε κάθε πιθανό πολυώνυμο. Πράγματι αν δύο πλειάδες απεικονίζονταν σε διαφορετικά πολυώνυμα p, q θα είχαμε ότι αυτά συμφωνούν σε $t-1$ τιμές και στην τιμή $(0, s)$ και συνεπώς $p - q = 0$.

Γράφοντας με κεφαλαία λοιπόν τις τυχαίες μεταβλητές έχουμε

$$\Pr[S = s|(S_1, \ldots, S_{t-1}) = (s_1, \ldots, s_{t-1})] = \Pr[S = s|(A_1, \ldots, A_{t-1}) = (a_1, \ldots, a_{t-1})] = \Pr[S = s]$$

όπου η τελευταία ισότητα προκύπτει γιατί για για κάθε επιλογή συντελεστών υπάρχουν $|\mathbb{F}|$ πιθανά μυστικά.

Στα παραπάνω θεωρούσαμε έμπιστο Dealer. Με γνωστές τεχνικές Verifiable Secret Sharing [18] μπορούμε να εξαλείψουμε την παραπάνω υπόθεση.

2.5.3 Κατανεμημένο Σχήμα El Gamal

Χρησιμοποιώντας τις τεχνικές της προηγούμενης υποενότητας θα δείξουμε πως μπορούν n παίχτες να μοιράζονται ένα ιδιωτικό κλειδί El Gamal. Το σενάριο που θέλουμε να πετύχουμε είναι για έναν αριθμό $t \leq n$ αν λιγότεροι από t είναι διεφθαρμένοι και συνεργάζονται να μην μπορούν να καταφέρουν καμία επίθεση, δηλαδή αφενώς να μην μπορούν μόνοι τους να αποκρυπτογραφήσουν ένα ciphertext και αφεταίρου να μην μπορούν να εμποδίσουν τους υπόλοιπους από το να ακολουθήσουν το πρωτόκολλο. Θεωρούμε ότι υπάρχει ένας έμπιστος Dealer που κατασκευάζει το ζεύγος κλειδιών και το διαμοιράζει στους παίχτες αλλά αυτή η υπόθεση μπορεί να αποφευχθεί χρησιμοποιώντας τεχνικές Verifiable Secret Sharing.

Η διαδικασία είναι η κάτωθη.

- **Key Distribution**
  1. Ο Dealer δημιουργεί ένα ζεύγος κλειδιών $s, h = g^s$
  2. Χρησιμοποιώντας (t, n) Shamir Secret Sharing δημιουργεί τα shares s_i και τα μοιράζει στους παίχτες.
  3. Δημοσιοποιεί το δημόσιο κλειδί h
- **Decryption**
  1. Οι παίχτες με είσοδο (c_1, c_2) θέλουν να υπολογίσουν $\text{Dec}_s((c_1, c_2)) = \frac{c_2}{c_1^s}$
  2. Κάθε παίχτης υπολογίζει $c_1^{s_i}$ και το δημοσιοποιεί (στο βήμα αυτό μπορεί να χρησιμοποιηθεί κατάλληλα απόδειξη ορθού υπολογισμού)
  3. Οι παίχτες συμφωνούν σε t από τα δημοσιευμένα αποτελέσματα και δίνουν σαν έξοδο $\frac{c_2}{\prod_{i=1}^{t} c_1^{s_i \lambda_i(0)}}$

Ισχύει ότι $p(x) = \sum_{i=1}^{t} s_i \cdot \lambda_i(x)$ και συνεπώς $\sum_{i=1}^{t} s_i \cdot \lambda_i(0) = p(0) = s$ οπότε έχουμε ότι

$$\frac{c_2}{\prod_{i=1}^{t} c_1^{s_i \lambda_i(0)}} = \frac{c_2}{c_1^{\sum_{i=1}^{t} s_i \lambda_i(0)}} = \frac{c_2}{c_1^s}$$

που είναι και το επιθυμητό.

2.6 Αποδείξεις Μηδενικής Γνώσης

Ας σκεφτούμε το εξής σενάριο. Η Alice θέλει να πείσει τον Bob για την αλήθεια μιας μαθηματικής πρότασης. Δεν θέλει όμως να δώσει στον Bob καμία άλλη πληροφορία για την πρόταση αυτή πέρα από ότι είναι αληθής. Θέλει δηλαδή να μεταδώσει ένα μόνο bit πληροφορίας. Αν ο Bob εμπιστεύεται την Alice αυτό μπορεί να γίνει τετριμμένα, του στέλνει απλά 1! Τι γίνεται όμως αν δεν υπάρχει εμπιστοσύνη; Οι Goldwasser, Micali και Rackoff εισήγαγαν τις αποδείξεις μηδενικής γνώσης [26] που λύνουν αυτό το σενάριο. Δεν είναι δύσκολο να φανταστεί κανείς γιατί αυτές είναι εξαιρετικά σημαντικές σε κρυπτογραφικά πρωτόκολλα γενικά και σε ψηφιακές ψηφοφορίες ειδικά. Όταν παίχτες που δεν εμπιστεύονται ο ένας τον άλλον εκτελούν ένα πρωτόκολλο θα χρειαστεί να μεταδώσουν μηνύματα που εξαρτιόνται από μυστικές πληροφορίες του κάθε παίχτη. Θα πρέπει λοιπόν να μπορεί να είναι επαληθεύσιμο ότι ακολουθήθηκε το πρωτόκολλο χωρίς να διαρρέεται καμία μυστική πληροφορία.

Για τις ανάγκες της παρούσας εργασίας θα εξετάσουμε πολύ επιφανειακά τις αποδείξεις μηδενικής γνώσης. Αναλυτική μελέτη τους από θεωρητική σκοπιά μπορεί να βρει ο αναγνώστης στο βιβλίο [21].

2.6.1 Διαλογικά Συστήματα Αποδείξεων

Τα διαλογικά συστήματα αποδείξεων είναι ζεύγη μηχανών Turing που επικοινωνούν και μοντελοποιούν την επαλήθευση μιας απόδειξης. Δεν θα σταθούμε στο πως ορίζουμε διαδραστικές μηχανές Turing αλλά θα παραπέμψουμε εδώ [21] ή εδώ [5]. Ορίζουμε ως $out_V(P,V)(x)$ την έξοδο του V μετά την αλληλεπίδραση. Πρακτικά αυτή θα είναι τυχαία μεταβλητή με πεδίο τιμών 1 ή 0 για αποδοχή ή απόρριψη αντίστοιχα.

Ορισμός 2.18. *Ένα ζεύγος μηχανών Turing (P,V) ονομάζεται διαλογικό σύστημα αποδείξεων για την γλώσσα L αν ο V είναι PPT και αν ισχύουν τα εξής:*

- ***Πληρότητα (Completeness):*** *Για κάθε $x \in L$ ο V πάντα αποδέχεται δηλαδή $\Pr[out_V(P,V)(x) = 1] = 1$*
- ***Ορθότητα (Soundness):*** *Για κάθε $x \notin L$ και για κάθε P^* υπάρχει συνάρτηση αμελητέα ως προς $|x|$ τέτοια ώστε ο V (σχεδόν) πάντα απορρίπτει δηλαδή $\Pr[out_V(P^*,V)(x) = 1] \leq \mu(|x|)$ για κάποια αμελητέα ως προς το $|x|$ συνάρτηση.*

Οι γλώσσες που έχουν διαλογικά συστήματα αποδείξεων ορίζουν την κλάση **IP**. Ο Shamir απέδειξε ότι **IP** = **PSPACE** [48], πρόκειτε δηλαδή για μια τεράστια κλάση. Παρατηρήστε ότι αν βγάζαμε την τυχαιότητα από το ορισμό η κλάση θα γινόταν η κλάση **NP**.

Αν στον ορισμό επιβάλουμε και στον P να είναι PPT αλλά να έχει ένα βοηθητικό μυστικό input τότε το αντίστοιχο σύστημα καλείται διαλογικό επιχείρημα (interactive arguement).

Εμάς όπως είπαμε στην εισαγωγή, μας ενδιαφέρει ένα υποσύνολο των συστημάτων διαλογι-

κών αποδείξεων, αυτά τα οποία δεν διαρρέουν καμία πληροφορία πέρα από την ορθότητα του ισχυρισμού. Αυτά μελετάμε παρακάτω.

2.6.2 Αποδείξεις Μηδενικής Γνώσης

Υπάρχει ένα μη τετριμμένο ζήτημα που πρέπει να λύσουμε για να μπορέσουμε να περάσουμε στον ορισμό. Τι σημαίνει μια μηχανή Turing να ξέρει κάτι ή να μαθαίνει κάτι από επικοινωνία με άλλη μηχανή; Αυτό δεν είναι καθόλου απλό να οριστεί. Η απάντηση που δίνουμε είναι ότι μια μηχανή Turing ξέρει οτιδήποτε μπορεί να υπολογίσει. Έτσι μοντελοποιούμε και το τι μαθαίνει μετά από επικοινωνία με μία άλλη μηχανή. Για παράδειγμα αν η μηχανή είναι PPT δεν μπορεί να υπολογίσει αν ένα μεγάλο γράφημα έχει κύκλο hamilton. Αν όμως 'επικοινωνήσει' με μία άλλη μηχανή και πάρει από αυτήν ένα πιστοποιητικό για έναν κύκλο Hamilton τότε ξέρει να το υπολογίσει.

Με αυτόν τον τρόπο μπορούμε να ορίσουμε και τι θα πει μια μηχανή να μην μαθαίνει τίποτα από ένα διαλογικό σύστημα αποδείξεων εκτός από την ισχύ μιας πρότασης. Σημαίνει πως οτιδήποτε μπορεί να υπολογίσει μετά την εκτέλεση της διαλογικής απόδειξης και δεδομένου ότι αποδέχεται την απόδειξη μπορούσε να το υπολογίσει και μόνη της πριν την απόδειξη, φτάνει να δεχόταν ότι η πρόταση ισχύει. Αυτό μας οδηγεί στον παρακάτω ορισμό

Ορισμός 2.19. *Ένα διαλογικό σύστημα αποδείξεων είναι Σύστημα Απόδειξης Μηδενικής Γνώσης **ZK** για την γλώσσα L αν για κάθε PPT V^* υπάρχει μία άλλη PPT μηχανή M^* τέτοια ώστε οι οικογένειες τυχαίων μεταβλητών*

- $\{out_{V^*}(P, V^*)(x)\}_{x \in L}$
- $\{M^*(x)\}_{x \in L}$

να είναι υπολογιστικά μη διαχωρίσιμες.

Ας ερμηνεύσουμε τον ορισμό. Καταρχήν παρατηρούμε ότι η ιδιότητα της μηδενικής γνώσης είναι ιδιότητα του διαλογικού συστήματος αποδείξεων. Οι δύο οικογένειες τυχαίων μεταβλητών ποσοδεικτούνται βάση των στοιχείων που ανήκουν στην γλώσσα. Δεν μας ενδιαφέρει δηλαδή τι γίνεται αν $x \notin L$. Αυτό είναι λογικό αφού όταν εξετάζουμε αυτή την ιδιότητα θέλουμε να προστατέψουμε έναν τίμιο Prover από έναν κακόβουλο Verifier που θέλει να αποσπάσει πληροφορία από αυτόν. Η πρώτη οικογένεια τυχαίων μεταβλητών είναι η έξοδος του V κατά την εκτέλεση της διαλογικής απόδειξης. Μπορούμε χωρίς βλάβη της γενικότητας να υποθέτουμε ότι η έξοδος αυτή είναι η όψη του V δηλαδή η είσοδός του x, η τυχαιότητα που χρησιμοποιεί και το σύνολο των μηνυμάτων που λαμβάνει. Αυτό γιατί κάθε υπολογισμός που θα ήθελε να κάνει είναι συνάρτηση των παραπάνω.

Ο ορισμός μας λέει ότι για κάθε πιθανό κακόβουλο Verifier υπάρχει μια άλλη μηχανή, ο προσομοιωτής M^* που η έξοδός του είναι ίδια (υπολογιστικά) με την έξοδο του V μετά από την εκτέλεση της διαλογικής απόδειξης που όμως δεν αλληλεπιδρά καθόλου με τον P! Δηλαδή

πρακτικά ο V^* θα μπορούσε πολύ απλά να μην πάει στον P και να υπολόγιζε αυτό που ήθελε καλώντας απλά τον simulator! Συνεπώς δεν μαθαίνει τίποτα από την ίδια την αλληλεπίδραση (που να μπορεί να χρησιμοποιήσει υπολογιστικά).

Υπάρχουν πολλές παραλλαγές του παραπάνω ορισμού. Αν οι δύο οικογένειες τυχαίων μεταβλητών είναι ακριβώς ίδιες τότε έχουμε μια Τέλεια Απόδειξη Μηδενικής Γνώσης **PZK** και αν η στατιστική τους απόσταση είναι αμελητέα έχουμε Στατιστική Απόδειξη Μηδενικής Γνώσης **SZK**. Παρατηρήστε ότι αυτό είναι διαφορετικό με την απλή Υπολογιστική Απόδειξη Μηδενικής Γνώσης **ZK** καθώς μπορεί οι δύο οικογένειες τυχαίων μεταβλητών να έχουν πολύ μεγάλη (μη αμελητέα) στατιστική απόσταση αλλά κανένας PPT αλγόριθμος να μην μπορεί να το ξεχωρίσει αυτό με μη αμελητέα πιθανότητα.

Μία άλλη λιγότερο ισχυρή παραλλαγή είναι η Απόδειξη Μηδενικής Γνώσης Μαύρου Κουτιού. Σε αυτή δεν υπάρχει ένας εξομοιωτής M^* για κάθε V^* άλλα ένας καθολικό εξομοιωτής M που έχει πρόσβαση μαύρου κουτιού σε κάθε επίδοξο V^*. Δηλαδή ρωτάει πως θα απαντούσε ο V^* κατά την εκτέλεση του πρωτοκόλλου σε ένα μαντείο 'επόμενου μηνύματος' και η έξοδος που βγάζει για $x \in L$ είναι ίδια (με μια από τις τρεις παραπάνω έννοιες) με αυτή που θα έβγαζε ο V^*.

Γενικά το θέμα των Αποδείξεων Μηδενικής Γνώσης είναι τεράστιο και έχει εφαρμογές και συνέπειες τόσο στην κρυπτογραφία όσο και στην θεωρία υπολογιστικής πολυπλοκότητας. Όπως είπαμε μια αναλυτική μελέτη του θα βρείτε στο [21].

Τέλος αναφέρουμε την σύνθεση αποδείξεων μηδενικής γνώσης μιας και θα το συναντήσουμε στην συνέχεια. Έχει αποδειχθεί (βλ. [21]) ότι η σειριακή σύνθεση αποδείξεων μηδενικής, δηλαδή η εκτέλεση πολλών αποδείξεων μηδενικής γνώσης στη σειρά, διατηρεί τις ιδιότητες μηδενικής γνώσης. Αντίθετα αυτό δεν ισχύει για την παράλληλη (parallel) ή ταυτόχρονη (concurrent) εκτέλεσή τους. Αυτός είναι και ένας από τους λόγους της δυσκολίας κατασκευής αποδοτικών αποδείξεων μηδενικής γνώσης. Στο επόμενο κεφάλαιο θα δούμε πως να κάνουμε αποδοτικές εφαρμογές σε περιπτώσει που αρκούμαστε σε κάτι λιγότερο αποδοτικό από τις **ZK**. Θα δούμε επίσης πως αυτές οι κατασκευές μπορούν στο μοντέλο του τυχαίου μαντείου να μετατραπούν σε μη διαδραστικές αποδείξεις μηδενικής γνώσης, σε αποδείξεις δηλαδή μηδενικής γνώσης ενός γύρου.

Το σημαντικότερο αποτέλεσμα για τις αποδείξεις μηδενικής γνώσης είναι ότι κάθε γλώσσα $L \in \mathbf{NP}$ έχει απόδειξη μηδενικής γνώσης, δηλαδή $\mathbf{NP} \subseteq \mathbf{ZK}$. Το αποτέλεσμα δόθηκε από τους Goldreich, Micali, Wigderson [24].

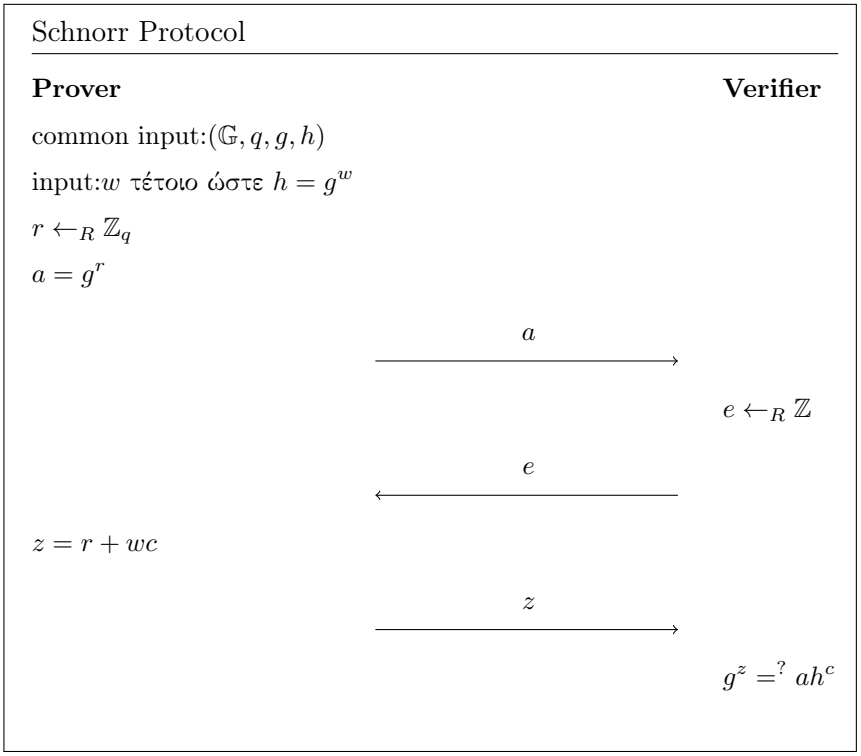
2.7 Σ-Πρωτόκολλα

Τα Σ πρωτόκολλα είναι αποδείξεις γνώσης. Έχουμε δηλαδή έναν Prover P και έναν Verifier V και ο P θέλει να πείσει τον V ότι ξέρει έναν μάρτυρα w μιας **NP** σχέσης για κάποια γλώσσα. Το πρόβλημα αυτό μπορεί να προκύψει για παράδειγμα αν ο P θέλει να πείσει τον V που είναι ένας διακομιστής για την ταυτότητά του χωρίς όμως να στείλει το μυστικό του. Τονίζουμε

ότι τα Σ-πρωτόκολλα δεν είναι αποδείξεις γνώσης μηδενικής γνώσης αλλά λιγότερο ισχυρά. Όπως όμως θα δούμε έχουν πολλές ενδιαφέρουσες και χρήσιμες ιδιότητες. Μεγάλο μέρος αυτής της υποενότητας βασίζεται στις σημειώσεις του Ivan Damagard[15].

2.7.1 Το Πρωτόκολλο του Schnorr

Το πιο χαρακτηριστικό και ιστορικά πρώτο τέτοιο πρωτόκολλο είναι το πρωτόκολλο του Schnorr που δημοσιεύθηκε το [45]. Στο πρωτόκολλο αυτό ο P ξέρει τον διακριτό λογάριθμο ως προς g μιας ομάδας $\langle g \rangle = \mathbb{G}$ τάξης q με q πρώτο. Θέλει να πείσει τον V για αυτό το γεγονός. Το πρωτόκολλο φαίνεται παρακάτω.



Αρχικά κάνουμε κάποιες παρατηρήσεις. Η δουλειά του V είναι να ρίξει δημόσια κάποια κέρματα. Τέτοια συστήματα αποδείξεων ονομάζονται public coin proofs [5]. Το q καθορίζει το πόσο ασφαλές είναι το πρωτόκολλο. Ας δούμε τώρα κάποιες ιδιότητες που έχει το παραπάνω.

- Κατ αρχήν το πρωτόκολλο είναι **πλήρες**. Αν ο P ξέρει τον μάρτυρα w ο V θα αποδεχτεί.
- Αν ο P μπορεί να απαντήσει για το ίδιο commitment a σε δύο διαφορετικά c, c' του V τότε ξέρει (μπορεί να υπολογίσει) τον μάρτυρα w. Πράγματι τότε έχουμε

$$\left.\begin{matrix} g^z = ah^c \\ g^{z'} = ah^{c'} \end{matrix}\right\} \Rightarrow g^{z-z'} = h^{c-c'} \Rightarrow w \equiv (z - z')(c - c')^{-1} \mod q$$

  Η ιδιότητα αυτή καλείται **ειδική ορθότητα (special soundness)**
- Το πρωτόκολλο είναι μηδενικής γνώσης αν ο V είναι τίμιος και ακολουθεί το πρωτόκολλο. Πράγματι για τον V ένας προσομοιωτής αρκεί να διαλέξει τα μηνύματα με

ανάποδη σειρά. Διαλέγει δηλαδή τυχαία z, c και θέτει $I = g^z h^{-c}$. Η έξοδος αυτή είναι ισόνομη με αυτή που θα είχε ο V μετά από αληθινή επικοινωνία. Η ιδιότητα αυτή καλείται **Τίμιου Επαληθευτή Απόδειξη Μηδενικής Γνώσης (Honest Verifier Zero Knowledge HVZK)** . Αν δηλαδή διασφαλίζαμε ότι ο V δεν παρεκλίνει από το πρωτόκολλο τότε δεν θα αποκτούσε καμία πληροφορία για το w.

Είμαστε πλέον σε θέση να ορίσουμε τα Σ-Πρωτόκολλα.

Ορισμός 2.20. *Ένα πρωτόκολλο καλείται Σ-Πρωτόκολλο αν η επικοινωνία είναι της μορφής (I, c, z) όπου c είναι t δημόσια κέρματα του V και το πρωτόκολλο έχει τις τρεις ιδιότητες: πληρότητα, ειδική ορθότητα, και **HVZK** όπως αυτές ορίστηκαν παραπάνω.*

Δίνουμε παρακάτω κάποιες ιδιότητες των Σ-Πρωτοκόλλων. Αρχικά η παράλληλη εκτέλεση 2 ή περισσοτέρων Σ-Πρωτοκόλλων είναι Σ-Πρωτόκολλο. Η πληρότητα και η ειδική ορθότητα προκύπτουν εύκολα. Το ίδιο και η ιδιότητα **HVZK**. Για να το δούμε αυτό αρκεί να πάρουμε τις εξόδους που δίνουν οι δύο προσομοιωτές και να τις ενώσουμε ανά μέλη. Τότε το αποτέλεσμα θα είναι ισόνομο με την κατανομή μιας αληθινής συνομιλίας μεταξύ των P, V.

Επιπλέον μπορούμε να κατασκευάσουμε OR-Proofs. Δηλαδή έστω δύο **NP** σχέσιες R_1, R_2 και δύο στοιχεία x_1, x_2. Έστω ότι ο P ξέρει έναν μάρτυρα w_b τέτοιο ώστε $(x_b, w_b) \in R_b$. Μπορούμε να δείξουμε, αν οι σχέσεις έχουν Σ-Πρωτόκολλα Π_1, Π_2 ότι ξέρει μάρτυρα για μία από τις δύο εισόδους. Μάλιστα η κατασκευή είναι πολύ ενδιαφέρουσα γιατί είναι μία από τις λίγες περιπτώσεις που στην πράξη χρησιμοποιούμε τον προσομοιωτή και δεν είναι απλά ένα θεωρητικό κατασκεύασμα για μια απόδειξη ασφάλειας. Έστω $\mathcal{S}_b$ ο προσομοιωτής για το πρωτόκολλο της σχέσης b. Δίνουμε την κατασκευή:

Algorithm: OR-Proof

Input : x_1, x_2 κοινό και στους δύο. w_b μυστικό για τον P τέτοιο ώστε $(x_b, w_b) \in R_b$

Output: $b \in \{0, 1\}$αποδοχή ή απόρριψη από τον V

1 Ξεκινάει ο P. Τρέχει τον προσομοιωτή $\mathcal{S}_{1-b}$ και παίρνει $(I_{1-b}, c_{1-b}, z_{1-b})$.
2 Διαλέγει I_b και στέλνει (I_1, I_2) στον V
3 Ο V ρίχνει t κέρματα στέλνει c στον P
4 Ο P υπολογίζει $c_b = c \oplus c_{1-b}$ και υπολογίζει για c_b το z_b. Στέλνει c_1, c_2, z_1, z_2 στον V που επαληθεύει ότι τα (I_1, c_1, z_1) είναι αποδεκτά για Π_1, τα (I_2, c_2, z_2) είναι αποδεκτά για Π_2 και ότι $c_1 \oplus c_2 = c$. Δίνει αντίστοιχα 1 ή 0.

Παρατηρήστε ότι επί της ουσίας το c μοιράζεται με ένα πολύ απλό secret sharing και ο V έχει τη δυνατότητα να απαντήσει μόνο ένα challenge για το x_{1-b} και κάθε challenge για x_b. Συνεπώς έχει μόνο μία επιλογή που μπορεί να πάρει για τα challenges. Ο συνδυασμός των δύο $\mathcal{S}_i$ δίνει την **HVZK** ιδιότητα. Επίσης η παράλληλη εκτέλεση εύκολα δίνει και AND-Proofs. Τέλος σημειώνουμε ότι χρησιμοποιώντας Shamir Secret Sharing με τον ίδιο τρόπο μπορούμε να αποδείξουμε ότι ξέρουμε t από n μάρτυρες.

2.7.2 Το Πρωτόκολλο του Chaum-Pedersen

Παρακάτω δίνουμε το Σ-Πρωτόκολλο Chaum-Pedersen. Το πρωτόκολλο αυτό χρησιμοποιείται για να αποδείξουμε ότι αν έχουμε μια ομάδα $\mathbb{G}$ τάξης q για q πρώτο και δύο στοιχεία g_1, g_2 τότε τα h_1, h_2 έχουν ίδιο διακριτό λογάριθμο ως προς τα g_1, g_2 και ξέρουμε w τέτοιο ώστε $h_1 = g_1^w$ και $h_2 = g_2^w$ δηλαδή $\log_{g_1} h_1 = \log_{g_2} h_2 = w$. Ισοδύναμα αυτό σημαίνει ότι η τριάδα g_2, h_1, h_2 είναι τριάδα Diffie Hellman ως προς τον γεννήτορα g_1. Δίνουμε το πρωτόκολλο.

Chaum-Pedersen Protocol

Prover		**Verifier**
common input:$(\mathbb{G}, q, g_1, g_2, h_1, h_2)$		
input:w τέτοιο ώστε $h_1 = g_1^w$ και $h_2 = g_1^w$		
$r \leftarrow_R \mathbb{Z}_q$		
$a = (a_1, a_2) = (g_1^r, g_2^r)$		
	$\xrightarrow{a}$	
		$e \leftarrow_R \mathbb{Z}$
	$\xleftarrow{e}$	
$z = r + wc$		
	$\xrightarrow{z}$	
		$g_1^z =^? a_1 h_1^c$ και $g_2^z =^? a_2 h_2^c$

2.7.3 Η ευρυστική μέθοδος των Fiat-Shamir

Κοιτώντας τα Σ-Πρωτόκολλα παρατηρούμε ότι ο V κατά τη διάρκεια του πρωτοκόλλου δεν κάτι τίποτα πέρα από το να δίνει τυχαιότητα στον P. Μήπως θα μπορούσαμε να αφαιρέσουμε τελείως από το πρωτόκολλο τον V και να παίρναμε από αλλού την τυχαιότητα; Αυτό θα έκανε τα πράγματα πολύ πιο αποδοτικά αφού θα παραγάγαμε μία φορά την απόδειξη και θα μπορούσαν να την ελέγξουν πολλοί διαφορετικοί V χωρίς να αλληλεπιδρούν με εμάς. Τέτοιες αποδείξεις ονομάζονται Non Interactive. Πώς όμως θα διασφαλίζαμε ότι ο P δεν κλέβει; Μια απάντηση σε αυτήν την ερώτηση έδωσαν οι Fiat και Shamir [19].

Ξέρουμε ότι ο P, αν δεν ξέρει τον μάρτυρα, μπορεί να υπολογίσει μόνο ένα challenge c για κάθε I. Αρκεί λοιπόν να διασφαλίσουμε ότι ο P διαλέγει το a πριν επιλεχθεί το c όπως θα γινόταν σε ένα κανονικό πρωτόκολλο. Την λύση σε αυτό μας δίνει το τυχαίο μαντείο. Όπως είπαμε ο P μπορεί για κάθε I να απαντήσει μοναδικό c. Αν λοιπόν ο P διαλέξει a και θέσει ως $c = H(x, a)$, αν το H μοντελοποιείται ως τυχαίο μαντείο τότε το c είναι ομοιόμορφα κατανεμημένο, ακριβώς όπως και στο πραγματικό πρωτόκολλο. Επίσης, αν το t

είναι καταλλήλως μεγάλο τότε όσες φορές κι αν προσπαθήσει (πολυωνιμικές) η πιθανότητα να καταφέρει να πετύχει το c που ξέρει είναι αμελητέα. Συνεπώς το τυχαίο μαντείο αντικαθιστά τον V και με αυτόν τον τρόπο η απόδειξη γίνεται Non Interactive.

Παρατηρούμε επίσης ότι πλέον δεν έχουμε malicious Verifiers αφού ο Prover με τη βοήθεια του μαντείου προσωμοιώνει τον V και πλέον η ιδιότητα **HVZK** γίνεται **ZK**.

2.7.4 Αποδείξης Χρήσιμες για Ηλεκτρονικές Ψηφοφορίες

Μελετάμε στην υποενότητα αυτή αποδείξεις που είναι χρήσιμες σε ηλεκτρονικές ψηφοφορίες και βασίζονται στα παραπάνω. Συγκεκριμένα μελατάμε διάφορες αποδείξεις γύρω από το σχήμα ασύμμετρης κρυπτογράφησης El Gamal.

- **Απόδειξη Γνώσης Plaintext**
  Έστω m ένα μήνυμα και $E(m) = (g^r, mh^r)$. Μπορούμε να δείξουμε ότι ξέρουμε το m με Non Interactiv Zero Knowledge απόδειξη γνώσης της τυχαιότητας r που χρησιμοποιήθηκε, δηλαδή γνώσης του $\log_g g^r$.

- **Απόδειξη Ορθής Κρυπτογράφησης**
  Έστω m και $c = (c_1, c_2) = (g^r, mh^r)$. Μπορούμε να αποδείξουμε ότι κρυπτογραφήσαμε όντως το m ως εξής: Δείχνουμε ότι $\log_g c_1 = \log_h \frac{c_2}{m}$

- **Απόδειξη Ορθής Κρυπτογράφησης Από Σύνολο**
  Έστω σύνολο $V = \{v_1, \ldots, v_k\}$ και m, $c = E_T(m)$. Για να δείξουμε ότι $Dec(c) \in V$ αρκεί να κάνουμε ένα OR-Proof ορθής κρυπτογράφησης για κάθε στοιχείο του V προσομοιώνοντας όλες τις αποδείξεις εκτός από μία.

- **Απόδειξη Ύψωσης Κρυπτοκειμένου σε Δύναμη**
  Έστω $c = (c_1, c_2)$ και θέλουμε να υπολογίσουμε c^k (που αντιστοιχεί σε κρυπτογράφηση του m^k). Υπολογίζουμε c_1^k, c_2^k και δίνουμε μία απόδειξη ότι c_1, c_2, c_1^k, c_2^k είναι τετράδα Diffie Hellman με το πρωτόκολλο των Chaum-Pedersen.

- **Απόδειξη Επανακρυπτογράφησης El Gamal**
  Για να αποδείξει μία αρχή ότι επανακρυπτογράφησε το (g^r, mh^r) σε (g^{r+t}, mh^{r+t}) αρκεί να αποδείξει ότι ξέρει την τυχαιότητα t, δηλαδή αρκεί να δείξει ότι ξέρει t τέτοιο ώστε η τετράδα $(g, h, \frac{g^{r+t}}{g^r}, \frac{mh^{r+t}}{mh^r})$ να είναι Diffie Hellman, αρκεί δηλαδή μία εκτέλεση του πρωτοκόλλου Chaum-Pedersen.

2.7.5 Καθορισμένου Επαληθευτή Αποδείξεις

Ας φανταστούμε ότι δίνουμε μια NIZK απόδειξη στον V. Μπορεί μετά αυτός να ισχυριστεί ότι ξέρει τον μάρτυρα της σχέσης σε κάποιον τρίτο παρουσιάζοντάς του την απόδειξη που πήρε από εμάς, πράγμα που δεν είναι πάντα επιθυμητό. Ένας πολύ απλός τρόπος να το διασφαλίσουμε είναι ο εξής. Ας υποθέσουμε ότι ο V έχει ένα ζεύγος κλειδιών s, g^s. Αντί να του δείξουμε ότι $(x, w) \in R$ μπορούμε να του δώσουμε μια OR απόδειξη που το ένα κομμάτι της είναι για γνώση

w για τη σχέση R με x και το άλλο να είναι απόδειξη ότι ξέρουμε το διακριτό λογάριθμο του g^s. Όπως είδαμε αυτό μπορούμε να το κάνουμε βασιζόμενοι στα Σ-Πρωτόκολλα και μάλιστα με Non Interactive τρόπο.

Ο V ως κάτοχος του ζεύγους κλειδιών μπορεί να υποθέσει με σιγουριά ότι δεν ξέρουμε το s και συνεπώς ξέρουμε το w. Αν όμως πάει να παρουσιάσει αυτή την απόδειξη σε κάποιον τρίτο, αυτός θα γελάσει αφού ως γνώστης του s μπορεί χωρίς να ξέρει το w να την κατασκευάσει και συνεπώς αυτή δεν έχει καμία αξία για αυτόν. Οι αποδείξεις αυτές εισήχθηκαν στο [30].

2.8 Τυφλές Υπογραφές

Πολλές φορές είναι χρήσιμο μία έμπιστη αρχή να μπορεί να υπογράψει μηνύματα που τις στέλνουν οι χρήστες χωρίς να βλέπει το περιεχόμενό τους. Αυτό είναι πολύ επιθυμητή ιδιότητα στην περίπτωση που θέλουμε να προστατέψουμε την ιδιωτικότητα του χρήστη ακόμα και από την ίδια την αρχή, η οποία όμως πιστοποιεί κάτι για τον χρήστη. Παραδείγματα τέτοιων εφαρμογών αποτελούν οι ψηφιακές οικονομικές συναλλαγές στις οποίες θα θέλαμε η τράπεζα να δίνει δικαίωμα στον χρήστη να ξοδέψει χρήματα που έχει χωρίς όμως να μαθαίνει που τα ξόδεψε και οι ψηφιακές ψηφοφορίες στις οποίες θέλουμε μία έμπιστη αρχή να υπογράφει ένα ψηφοδέλτιο χωρίς όμως να βλέπει το περιεχόμενό του. Οι τυφλές υπογραφές ορίστηκαν από τον Chaum [10]. Παρακάτω θα ορίσουμε τι είναι ένα σχήμα τυφλών ψηφιακών υπογραφών και θα παρουσιάσουμε σύντομα το σύστημα τυφλών υπογραφών του Chaum. Στη συνέχεια θα μελετήσουμε το σχήμα τυφλών υπογραφών Okamoto-Schnorr.

Ορισμός 2.21. *Ένα σχήμα τυφλών υπογραφών είναι μια επέκταση ενός συστήματος ψηφιακών υπογραφών και ορίζεται από μια τριάδα αλγορίθμων (Gen,Sign,Vrfy) τέτοια ώστε*

- *Ο Gen με είσοδο την παράμετρο ασφαλείας 1^n δίνει ένα ζεύγος κλειδιών (sk, vk). Το sk ονομάζεται κλειδί υπογραφής και το vk κλειδί επαλήθευσης και είναι δημόσιο. Επίσης παράγονται χώροι $\mathcal{M}, \mathcal{S}$ που εξαρτώνται από το 1^n*
- *Ο Sign είναι πρωτόκολλο που εκτελείται μεταξύ ενός χρήστη U και του signer S (δηλαδή Sign$= (U, S)$) με κοινό input το vk και ιδιωτικό input m, sk για U, S αντίστοιχα. Η έξοδος του πρωτοκόλλου είναι σ*
- *Ο Vrfy με είσοδο $m \in \mathcal{M}, \sigma \in \mathcal{S}$ παράγει $b \in \{0, 1\}$*

για τους οποίους ισχύει ότι αν sk, vk είναι έξοδος του $Gen(1^n)$ τότε $Vrfy(out(U(vk, m), S(sk, vk))) = 1$ εκτός με αμελητέα πιθανότητα ως προς την παράμετρο ασφαλείας n.

2.8.1 Ασφάλεια Τυφλών Υπογραφών

Ένα σχήμα τυφλών υπογραφών πρέπει να ικανοποιεί δύο συνθήκες ασφάλειας.

1. Δεν μπορεί να πλαστογραφηθεί η υπογραφή

2. Ο υπογράφων κοιτώντας κάποιο μήνυμα και μια υπογραφή σε αυτό δεν θα πρέπει να μπορεί να το συσχετίσει με κάποια συγκεκριμένη εκτέλεση του πρωτοκόλλου Sign και θα πρέπει όταν υπογράφει να μην ξέρει ποιο μήνυμα υπογράφει.

Για την πρώτη ιδιότητα παρατηρήστε ότι ο ορισμός τους unforgability για τις απλές υπογραφές δεν μπορεί να εφαρμοστεί εδώ αφού ο Signer δεν ξέρει ποια μηνύματα έχει υπογράψει. Για το λόγο αυτό ο ορισμός του unforgability μας λέει ότι ένας αντίπαλος που λαμβάνει l τυφλές υπογραφές δεν θα πρέπει στο τέλος του πρωτοκόλλου να μπορεί να δώσει $l+1$ ζεύγη έγκυρων μηνυμάτων-υπογραφών, δεν μπορεί δηλαδή να παράξει παραπάνω υπογραφές από όσες έλαβε. Παρακάτω δίνουμε τους ορισμούς που δανειζόμαστε από το [46]. Μία εργασία που μελετάει θεωρητικά τις τυφλές υπογραφές είναι η [32].

Algorithm: $\text{OneMoreForge}_{\mathcal{A},\Pi}$

Input : security parameter n

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

$(sk, vk) \leftarrow \text{Gen}(1^n)$

$((m_1, \sigma_1), \ldots, (m_{k+1}, \sigma_{k+1})) \leftarrow \mathcal{A}^{\langle S(sk),\cdot\rangle}(vk)$

`/* Q ο αριθμός των επιτυχυμένων Interactions στο πρωτόκολλο Sign */`

if $(\forall i, j \mu\epsilon\ i \neq j\ m_i \neq m_j) \wedge (\forall i\ Vrfy(vk, m_i, \sigma_i) = 1) \wedge Q \leq k$ **then**

  return 1

else

  return 0

end

Με λόγια, το παραπάνω παίγνιο μας λέει ότι ένας $\mathcal{A}$ που μπορεί να ζητάει υπογραφές με το πρωτόκολλο Sign προσπαθεί να παράξει υπογραφές για παραπάνω μηνύματα από το πλήθος των επιτυχών εκτελέσεων του πρωτοκόλλου Sign που έκανε στο παίγνιο.

Ορισμός 2.22. *Ένα σχήμα τυφλών υπογραφών* Π *είναι one more unforgeable αν για κάθε PPT* $\mathcal{A}$ *υπάρχει αμελητέα συνάρτηση* μ *ως προς το* n *τέτοια ώστε* $\Pr[OneMoreForge_{\mathcal{A},\Pi}(n) = 1] \leq \mu(n)$.

Ο ορισμός αυτός δόθηκε πρώτη φορά στο [40].

Στη συνέχεια αναλύουμε την ιδιότητα του blindness. Σε αυτήν την εργασία θα ασχοληθούμε με στατιστικά τυφλές υπογραφές. Η ιδιότητα που θέλουμε να ορίσουμε είναι η εξής: ο υπογράφων δεν ξέρει ποιο μήνυμα υπογράφει και δεν μπορεί να συσχετίσει υπογραφές με εκτελέσεις του πρωτοκόλλου. Στο παιχνίδι που θα ορίσουμε ο αντίπαλος διαλέγει δύο μηνύματα και εμείς τα δίνουμε με τυχαία σειρά για υπογραφή από αυτόν. Σκοπός του είναι να μαντέψει την σειρά βλέποντας τις υπογραφές.

Algorithm: $\text{BlindExp}_{\mathcal{A},\Pi}$

Input : security parameter n

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

$(vk, m_0, m_1, st_find) \leftarrow \mathcal{A}(\textbf{find}, 1^n)$
$c \leftarrow_R \{0,1\}$
$st_{issue} \leftarrow \mathcal{A}^{\langle \cdot, U(pk,m_{1-b})\rangle, \langle \cdot, U(pk,m_{1-b})\rangle}(\textbf{issue}, st_find)$
if $\sigma_0 = \perp \vee \sigma_1 = \perp$ **then**
  $(\sigma_0, \sigma_1) = (\perp, \perp)$
else
  $c^* \leftarrow \mathcal{A}(\textbf{guess}, \sigma_0, \sigma_1, st_{issue})$
end
if $c = c^*$ **then**
  return 1
else
  return 0
end

Ορισμός 2.23. *Ένα σχήμα τυφλών υπογραφών* Π *είναι στατιστικά τυφλό αν για κάθε (unbounded)* $\mathcal{A}$ *ισχύει* $Pr[BlindExp_{\mathcal{A},\Pi}(n) = 1] = \frac{1}{2}$.

2.8.2 Τυφλές υπογραφές Chaum

Παρουσιάζουμε παρακάτω το σχήμα τυφλών υπογραφών Chaum. Το σχήμα αυτό είναι μια επέκταση του σχήματος RSA-FDH.

Algorithm: Gen

Input : security parameter n

Output: (sk, pk)

`/* τρέχουμε τον GenMod όπως στο RSA για να πάρουμε modulus` N `*/`
$N \leftarrow \text{GenMod}(1^n)$
$e \leftarrow_R \mathbb{Z}_{\phi(N)}$
`/* υπολογίζουμε τον πολλαπλασιαστικό αντίστροφο του` e `*/`
$d \leftarrow e^{-1} \mod \phi(n)$
$sk \leftarrow (N, e, d)$
$vk \leftarrow (N, e)$

Ο αλγόριθμος αυτός είναι ίδιος με τον αλγόριθμο Gen του σχήματος RSA-FDH.

Algorithm: Sign
Input : vk κοινό, $m \in \mathbb{Z}_N^*$ ιδιωτικό για U, sk ιδιωτικό για S, oracle σε $H : \mathcal{M} \to \mathbb{Z}_N^*$
Output: $\sigma \in \mathbb{Z}_N^*$

1 Ο U διαλέγει $r \leftarrow_R \mathbb{Z}_N^*$ και υπολογίζει $b = H(m)r^e \mod N$. Στέλνει b στον S
2 Ο S υπολογίζει $s \leftarrow b^d \mod N$ και το στέλνει στον U
3 Ο U επαληθεύει ότι $s^e = b \mod N$ και υπολογίζει $\sigma = sr^{-1} \mod N$ που είναι η έξοδος του πρωτοκόλλου.

Παρατηρούμε στο σημείο αυτό ότι το στοιχείο b είναι ομοιόμορφα δεν περιέχει καμία πληροφορία για το m αφού κατανέμεται ομοιόμορφα στο χώρο μηνυμάτων για τυχαίο r. Το ίδιο ισχύει και για την υπογραφή σ. Για να το θέσουμε διαφορετικά έχουμε ότι

$$\Pr[M = m, \Sigma = \sigma | B = b, S = s] \Pr[R = r] = \frac{1}{|\mathbb{Z}_N^*|}$$

απ όπου προκύπτει και η ιδιότητα blindness.

Algorithm: Vrfy
Input : N, e, m, σ με $c \in \mathbb{Z}_N$
Input : N, e, m, σ με $m, \sigma \in \mathbb{Z}_N$, oracle σε $H : \mathcal{M} \to \mathbb{Z}_N^*$
Output: $b \in \{0, 1\}$

1 $m' \leftarrow H(m)$
2 **if** $\sigma^e \equiv m' \mod N$ **then**
3 | output 1
4 **else**
5 | output 0
6 **end**

Και ο Vrfy είναι ίδιος με τον αντίστοιχο του RSA-FDH. Ας επαληθεύσουμε την ορθότητα. Έστω ότι το σ είναι υπογραφή του m που προκύπτει από το παραπάνω σχήμα. Τότε έχουμε

$$\begin{aligned}\sigma^e &\equiv (sr^{-1})^e \equiv (b^d r^{-1})^e \equiv ((H(m)r^e)^d r^{-1})^e \equiv (H(m)^d r^{ed} r^{-1})^e \\ &\equiv (H(m)^d r r^{-1})^e \equiv H(m)^{de} \equiv H(m) \mod N\end{aligned}$$

Ως προς την ασφάλειά τους για το Forgery παρέμενε για πολλά χρόνια ένα ανοιχτό ζήτημα μέχρι που εισήχθηκε μία νέα υπόθεση, η One More Rsa Assumption για να αποδειχτεί ασφαλές το σχήμα [7].

2.8.3 Τυφλές υπογραφές Okamoto-Schnorr

Σε αυτήν την ενότητα θα παρουσιάσουμε τις τυφλές υπογραφές Okamoto-Schnorr που στηρίζονται στο witness indistinguishable [17] πρωτόκολλο [37]. Η ασφάλειά τους ως προς το one more forgery αποδείχθηκε από τους Pointcheval-Stern [40], [41] και οι ίδιες τεχνικές

μπορούν να χρησιμοποιηθούν για την απόδειξη παρόμοιων σχημάτων. Θα παρουσιάσουμε το σχήμα και την κεντρική ιδέα της απόδειξης.

Algorithm: Gen

Input : security parameter n

Output: $sk = (r, s), pk = (\mathbb{G}, q, g, h, y)$ με $\mid \mathbb{G} \mid = q > n$ και q πρώτος

1 $(\mathbb{G}, q) \leftarrow \mathcal{IG}(n)$
2 $(r, s) \leftarrow_R \mathbb{Z}_q$
3 $(g, h) \leftarrow_R \mathbb{Q}$
4 $y \leftarrow g^{-r} h^{-s}$

Στη συνέχεια παρουσιάζουμε το πρωτόκολλο τριών κινήσεων για την υπογραφή μεταξύ Signer και Recepiant. Θεωρούμε μία συνάρτηση $H : \{0,1\}^* \to \mathbb{Z}_q$ την οποία μοντελοποιούμε ως τυχαίο μαντείο.

Sign

Signer		**Recepiant**
input:$(\mathbb{G}, q, g, h, r, s, H)$		input:$(\mathbb{G}, q, g, h, y, m, H)$
$t, u \leftarrow_R \mathbb{Z}$		
$a = g^t h^u$		
	$\xrightarrow{a}$	
		$\beta, \gamma, \delta \leftarrow_R \mathbb{Z}_q$
		$\alpha = a g^\beta h^\gamma y^\delta$
		$\epsilon = H(m, \alpha)$
		$e = \epsilon - \delta$
	$\xleftarrow{e}$	
$R = t + er$		
$S = u + es$		
	$\xrightarrow{R, S}$	
		$a =^? g^R h^S y^e$
		$\rho = R + \beta$
		$\sigma = S + \gamma$
		output $(m, \alpha, \rho, \sigma)$

Τέλος δίνουμε τον αλγόριθμο επαλήθευσης.

Algorithm: Vrfy
Input : $vk, H, m', sig = (m, \alpha, \rho, \sigma)$
Output: $b \in \{0, 1\}$

if $m \neq m'$ **then**
  return 0
end
$\epsilon \leftarrow H(m, a)$ **if** $\alpha = g^\rho h^\sigma y^\epsilon$ **then**
  return 1
else
  return 0
end

Έυκολα μπορούμε να επαληθεύσουμε την ορθότητα. Συγκεκριμένα έχουμε

$$\begin{aligned}
\alpha = g^\rho h^\sigma y^\epsilon &\Leftrightarrow a g^\beta h^\gamma y^\delta = g^{R+\beta} h^{S+\gamma} y^{e+\delta} \\
&\Leftrightarrow a = g^R h^S y^e \\
&\Leftrightarrow g^t h^u = g^{t+er} h^{u+es} (g^{-r} h^{-s})^e \\
&\Leftrightarrow g^t h^u = g^t + g^{er} h^u h^{es} g^{-er} h^{-es} \\
&\Leftrightarrow g^t h^u = g^t h^u
\end{aligned}$$

που ισχύει.

Για το blindness παρατηρούμε ότι οι τιμές των β, γ, δ που διαλέγονται από τον recepiant μασκάρουν τέλεια το μήνυμα και την υπογραφή και συνεπώς πετυχαίνουμε στατιστικό Blindness.

Μας μένει να εξετάσουμε τι συμβαίνει με το one more forgery. Το παρακάτω θεώρημα δόθηκε από τους Pointcheva-Stern.

Θεώρημα 2.4. *Μοντελοποιούμε την H ως Random Oracle. Έστω ένας PPT αλγόριθμος $\mathcal{A}$ που έχει ως είσοδο μόνο το δημόσιο κλειδί ο οποίος μπορεί να ζητάει (ακόμα και παράλληλα) l υπογραφές σε μηνύματα της επιλογής του και κάνει Q ερωτήσεις στο Random Oracle. Αν το l είναι πολυλογαρθμικό ως προς την παράμετρο ασφαλείας, Q πολυωνιμικό ως προς την παράμετρο ασφαλείας και με πιθανότητα μη αμελητέα παράγει $l + 1$ έγκυρες υπογραφές σε διαφορετικά μηνύματα τότε υπάρχει αλγόριθμος που σε πολυωνιμικό χρόνο και με μη αμελητέα πιθανότητα λύνει το DLOG.*

Η κεντρική ιδέα της απόδειξης είναι ότι αν έχουμε δύο διαφορετικές αναπαραστάσεις του α ως προς g, h τότε μπορούμε να βρούμε το $\log_g h$. Συγκεκριμένα θα έχουμε

$$\alpha = g^k h^l = g^{k'} h^{l'} \Rightarrow g^{k-k'} = h^{l'-l} \Rightarrow \log_g h = (k - k')(l' - l)^{-1}$$

Για την αναγωγή κάνουμε το εξής: Τρέχουμε μία φορά επιτυχημένα την επίθεση και παίρνουμε $l+1$ έγκυρες υπογραφές. Ξανατρέχουμε την επίθεση με διαφορετικό Random Oracle από ένα

σημείο και μετά και ελπίζουμε ότι στο σημείο που δώσαμε την πρώτη φορά απάντηση θα έχουμε νέα υπογραφή που θα φανερώνει διαφορετική αναπαράσταση του ίδιου α. Οι Pointcheval και Stern απέδειξαν ότι κάτι τέτοιο θα συμβεί με μη αμελητέα πιθανότητα. Τονίζουμε ότι η απόδειξη ισχύει για αντιπάλους που αιτούνται το πολύ πολυλογαριθμικές τω πλήθος υπογραφές, κάτι που σε πολλές εφαρμογές δεν είναι επιθυμητό.

2.9 Δίκτυα Μίξης

Τα δίκτυα μίξης είναι ένα βασικό κρυπτογραφικό primitive που σκοπό έχει να διασφαλίσει την ανωνυμία των χρηστών που συμμετέχουν σε ένα πρωτόκολλο. Συνοπτικά η δουλειά τους είναι να παίρνουν ένα σύνολο κρυπτοκειμένων, να τα επεξεργάζονται κατάλληλα (αποκρυπτογραφούν ή επανακρυπτογραφούν) και να τα ανακατεύουν. Συνεπώς όταν τα κρυπτοκείμενα αποκρυπτογραφηθούν έχει χαθεί η πληροφορία προέλευσής τους. Στη συνέχεια παρουσιάζουμε δίκτυα μίξης επανακρυπτογράφησης.

2.9.1 Δίκτυα Μίξης Επανακρυπτογράφησης

Θεωρούμε ένα σχήμα κρυπτογραφίας δημοσίου κλειδιού που επιτρέπει επανακρυπτογράφηση (στα επόμενα El Gamal). Θεωρούμε επίσης επίσης n οντότητες $\mathcal{M}_1, \ldots, \mathcal{M}_n$ καθένας από τους οποίους έχει ένα ζεύγος κλειδιών $x_i, y_i = g^{x_i}$. Ορίζουμε το καθολικό κλειδί $y = y_1 \cdot \ldots \cdot y_n$. Παρατηρήστε ότι το ιδιωτικό κλειδί που αντιστοιχεί σε αυτό το δημόσιο κλειδί είναι $x_1 + \ldots + x_n$. Σαν είσοδο στο δίκτυο μίξης θεωρούμε ένα διάνυσμα κρυπτοκειμένων $\{c_j = (g^{r_j}, m_j \cdot h^{r_j})\}_{j=1}^k$. Για να 'ανακατευτούν' τα μηνύματα κάθε μέλος του δικτύου κάνει τα εξής

- Δέχεται είσοδο $\{(c_{j1}, c_{j2})\}_{j=1}^k$
- Με το ιδιωτικό του κλειδί x_i αποκρυπτογραφεί μερικώς τα ciphertexts δηλαδή υπολογίζει

$$\{(c_{j1}, \frac{c_{j2}}{c_{j1}^{x_i}})\}_{j=1}^k$$

- Επανακρυπτογραφεί κάθε ciphertext διαλέγοντας τυχαίο l_j και πολλαπλασιάζοντάς το με $(g^{l_j}, (y_{i+1} \cdot \ldots \cdot y_n)^{l_j})$
- Διαλέγει μία τυχαία μετάθεση του διανύσματος και το δίνει σαν είσοδο στο επόμενο μέλος.

Το τελευταίο μέλος απλά αποκρυπτογραφεί κάθε στοιχείο του διανύσματος.

Ένα απλό επαγωγικό επιχείρημα αρκεί για να μας πείσει ότι στο τέλος η έξοδος είναι μία τυχαία μετάθεση του διανύσματος των αποκρυπτογραφημένων μηνυμάτων δεδομένου ότι κάθε μέλος ακολουθεί το πρωτόκολλο.

Μία εναλλακτική προσέγγιση είναι τα μέλη να μοιράζονται με ένα (t, n) Secret Sharing σχήμα και κάθε μέλος να ανακατεύει απλά επανακρυπτογραφώντας και διαλέγοντας τυχαίο permutation. Στο τέλος συλλογικά υπολογίζεται το τελευταίο αποτέλεσμα.

Το μεγαλύτερο ζήτημα είναι πως δεν είμαστε βέβαιοι ότι η αρχή ακολουθεί το πρωτόκολλο. Αυτό πετυχαίνεται με αποδείξεις μηδενικής γνώσης ορθού ανακατέματος. Δεν θα επεκταθούμε παραπάνω στην παρούσα εργασία και όπου χρησιμοποιούμε δίκτυα μίξης θα τα αντιμετωπίζουμε σαν μαύρα κουτιά. Για τον ενδιαφερόμενο αναγνώστη παραπέμουμε στο αντίστοιχο κεφάλαιο του [27].

Κεφάλαιο 3

Βασικά Πρωτόκολλα Ψηφοφοριών και Επιθέσεις Εξαναγκασμού

Στην ενότητα αυτή θα μιλήσουμε για κάποια βασικά πρωτόκολλα ηλεκτρονικών ψηφοφοριών με έμφαση σε αυτά που διαθέτουν coercion resistance. Συγκεκριμένα ξεκινάμε με ένα κλασσικό πρωτόκολλο βασισμένο σε ομομορφική κρυπτογραφία, το CGS [14] και στη συνέχεια μελετάμε μία παραλλαγή του πρωτοκόλλου των Fujioka, Okamoto, Ohta [20] που στηρίζεται σε τυφλές υπογραφές [36] και τέλος εστιάζουμε σε πρωτόκολλα που επιδιώκουν να έχουν την ιδιότητα της αντοχής σε επιθέσεις εξαναγκασμού. Συγκεκριμένα ξεκινάμε με το πλαίσιο που έθεσαν οι Juels, Catalano, Jackobson για τον έλεγχο του coercion resistance [31] και το πρωτόκολλό τους που είναι ασφαλές ως προς αυτό το πλαίσιο και εφορμώμενοι από την μη αποδοτικότητά του, συνεχίζουμε εξετάζοντας κάποιες από τις χαρακτηριστικότερες τεχνικές για την βελτίωσή της. Σημειώνουμε για πληρότητα ότι πρόσφατα δημοσιεύτηκε και μία εργασία, το πρωτόκολλο Selene, [43] (που δεν εξετάζουμε σε αυτή την ενότητα) που έχει μία πολύ ενδιαφέρουσα, πιο ανθρωποκεντρική προσέγγιση στο ζήτημα.

3.1 Πρωτόκολλα Ψηφοφοριών Βασισμένα σε Ομομορφική Κρυπτογραφία

Μία πολύ σημαντική και διαδεδομένη μέθοδος πρωτοκόλλων ηλεκτρονικών εκλογών είναι μέσω της χρήσης ενός ομομορφικού σχήματος κρυπτογραφίας δημοσίου κλειδιού. Η γενική μορφή ενός τέτοιου σχήματος είναι η εξής: Μία αρχή κατέχει ένα ζεύγος δημοσίου-ιδιωτικού κλειδιού ενός ομομορφικού κρυπτογραφικού σχήματος και δημοσιοποιεί το δημόσιο κλειδί. Οι ψηφοφόροι καταθέτουν κρυπτογραφημένες ψήφους με αυτό το κλειδί σε ένα Bulletin Board και μετά το πέρας της περιόδου κατάθεσης ψήφων, οι κρυπτογραφημένες ψήφοι 'ενώνονται' μέσω των ομομορφικών ιδιοτήτων του κρυπτογραφικού σχήματος. Τέλος η αρχή καταμέτρησης αποκρυπτογραφεί το 'ενωμένο' αποτέλεσμα που αναπαριστά και το αποτέλεσμα των εκλογών.

Τα πλεονεκτήματα τέτοιων συστημάτων είναι πολλά και συνοψίζονται κυρίως στην απλότητά

τους, στην ανοχή σε σφάλματα ή/και corruptions και στην ικανοποίηση των κύριων ιδιοτήτων που απαιτούνται από ένα πρωτόκολλο ηλεκτρονικής ψηφοφορίας. Τέλος, η υπολογιστική δουλειά του ψηφοφόρου είναι η ελάχιστη δυνατή και το μεγαλύτερο υπολογιστικό βάρος πέφτει στην αρχή.

Στη συνέχεια θα παρουσιάσουμε το αρχέτυπο πρωτόκολλο αυτού του είδους που δημοσιεύτηκε από τους Cramer, Gennaro και Schoenmakers το 1997 [14].

3.1.1 Το Πρωτόκολλο CGS

Όπως αναφέραμε το πρωτόκολλο είναι το αρχέτυπο της κατηγορίας ομομορφικών συστημάτων και πολλά μετέπειτα πρωτόκολλα βασίζονται σε αυτό. Σαν κρυπτοσύστημα χρησιμοποιεί το El Gamal αλλά είναι αρκετά γενικό για να χρησιμοποιεί και άλλα ομομορφικά πρωτόκολλα πχ πρωτόκολλα που στηρίζονται στα τετραγωνικά υπόλοιπα. Ακολουθώντας τους συγγραφείς θα παρουσιάσουμε αρχικά το πρωτόκολλο για ψηφοφορίες με δύο επιλογές NAI και OXI και θα το γενικεύσουμε στη συνέχεια σε k επιλογές. Η μόνη υποδομή που χρειαζόμαστε είναι ένα Bulletin Board. Θεωρούμε ότι η αρχή καταμέτρησης αποτελείται από n μέλη και ότι μπορούμε να ανεχτούμε έως και t από αυτά τα μέλη να είναι διεφθαρμένα. Στη συνέχεια παρουσιάζουμε αναλυτικά το πρωτόκολλο.

Δημιουργία Παραμέτρων Εκλογών
Αρχικά τα n μέλη των καταμετρητών $T_1, \ldots, T_t$ εκτελούν ένα πρωτόκολλο δημιουργίας παραμέτρων και ζεύγους δημοσίου-ιδιωτικού κλειδιού (s, g^s) για το σχήμα Threshold El Gamal όπως περιγράφουμε στην αντίστοιχη ενότητα 2.5. Δημοσιοποιούνται οι παράμετροι p, q, g και το δημόσιο κλειδί g^s. Τέλος δημοσιοποιείται ένας γεννήτορας G της ομάδας. Οι δυνατοί ψήφοι είναι G^1 και G^{-1} δηλαδή επί της ουσίας οι ψήφοι καταγράφονται στον εκθέτη του G.

Κατάθεση Ψήφων

- Ο ψηφοφόρος V_i επιλέγει την ψήφο του G ή G^{-1}.
- Την κρυπτογραφεί με το δημόσιο κλειδί g^s, υπολογίζει δηλαδή $(x_i, y_i) = (g^{r_i}, G^{b_i} g^{sr_i})$
- Δημιουργεί μία μη διαλογική απόδειξη ότι το (x_i, y_i) είναι κρυπτογράφηση του G ή του G^{-1} με τις μεθόδους που εξετάσαμε. Σημειώνουμε εδώ ότι πρέπει να είμαστε πολύ προσεκτικοί στις παραμέτρους που χρησιμοποιούμε στην απόδειξη. Συγκεκριμένα θα πρέπει να συνυπολογίζεται στη χρησιμοποιούμενη τυχαιότητα η ταυτότητα του κάθε χρήστη για την αποφυγή αντιγραφής ψήφου.

Καταμέτρηση Ψήφων
Αφού περάσει η περίοδος κατάθεσης ψήφων, η T βγάζει το αποτέλεσμα των εκλογών ως εξής:

- Ελέγχει όλες τις αποδείξεις και αγνοεί στα επόμενα ψήφους με λαθεμένες αποδείξεις
- Πολλαπλασιάζει όλες τις έγκυρες ψήφους. Δηλαδή αν οι έγκυρες ψήφοι είναι $\{(x_i, y_i)\}_{i=1}^{l}$

τότε υπολογίζει

$$\prod_{i=1}^{l}(x_i, y_i) = \prod_{i=1}^{l}(g^{r_i}, G^{b_i} g^{s r_i}) = (g^{\sum_{i=1}^{l} r_i}, G^{\sum_{i=1}^{l} b_i} g^{s \sum_{i=1}^{l} r_i})$$

- Συνεργατικά αποκρυπτογραφεί το ciphertext αυτό και παίρνει ως αποτέλεσμα το $G^{\sum_{i=1}^{l} b_i}$.
- Δημοσιοποιεί μία απόδειξη ορθής αποκρυπτογράφησης.
- Τέλος υπολογίζει και δημοσιοποιεί το $\sum_{i=1}^{l} b_i$.

Καταρχήν, ερμηνεύοντας το ΝΑΙ ως 1 και το ΌΧΙ ως -1 παρατηρούμε ότι το αποτέλεσμα εκφράζει τη διαφορά των ΝΑΙ και των ΌΧΙ που δεδομένου της γνώσης του πλήθους των έγκυρων ψήφων εκφράζει πλήρως το αποτέλεσμα των εκλογών. Μία σημαντική ερώτηση είναι η εξής: Πώς η T υπολογίζει το αποτέλεσμα των εκλογών δεδομένου ότι αυτό είναι ο διακριτός λογάριθμος του αποκρυπτογραφημένου ciphertext? Η απάντηση είναι με brute force. Συγκεκριμένα έχουμε $2l$ δυνατά αποτελέσματα δεδομένου ότι l ψήφοι είναι έγκυροι, τα $G^{-l}, G^{-l+1}, \ldots, G^{l}$. Η T μπορεί να τα δοκιμάσει όλα δεδομένου ότι είναι λίγα (σίγουρα πολυωνυμικά ως προς το πλήθος των στοιχείων της ομάδας για κάθε ρεαλιστικό σενάριο εκλογών).

3.1.2 Ιδιότητες Ασφάλειας

Το παραπάνω πρωτόκολλο ικανοποιεί τις περισσότερες επιθυμητές ιδιότητες για πρωτόκολλα ηλεκτρονικής ψηφοφορίας. Κατ αρχήν δεν έχουμε διπλοψηφίες λόγω της αυθεντικοποίησης των μηνυμάτων στο Bulletin Board. Οι αποδείξεις καλά σχηματισμένης ψήφους επιβεβαιώνουν ότι κάθε κατατεθιμένη ψήφος αντιστοιχεί σε 1 ή -1 όπως θα έπρεπε. Η χρήση Bulletin Board σε συνδιασμό με το γεγονός ότι εκτός από την αποκρυπτογράφιση όλες οι άλλες πράξεις είναι δυνατόν να εκτελεστούν από τον καθένα μας δίνει individual verifiability και universal verifiability. Η απόδειξη ορθής αποκρυπτογράφισης επιβεβαιώνει την ορθότητα του τελικού αποτελέσματος. Τέλος για την ιδιωτικότητα στηριζόμαστε στην κρυπτογράφηση της κάθε ψήφου. Δηλαδή η δυσκολία του διακριτού λογαρίθμου μας διασφαλίζει ως προς τρίτους. Ως προς την T έχουμε ιδιωτικότητα δεδομένου ότι έχουμε λιγότερα από t διεφθαρμένα μέλη.

Οι ιδιότητες που λείπουν είναι η receipt freeness και κατά συνέπεια η ιδιότητα του coercion resistance. Για να το δούμε αυτό αρκεί να παρατηρήσουμε ότι ο κάθε ψηφοφόρος είναι γνώστης της τυχαιότητας r_i που χρησιμοποίησε για την κρυπτογράφιση και μέσω αυτής μπορεί να αποδείξει σε κάποιον τρίτο την κατατεθειμένη ψήφο. Το πρωτόκολλο αυτό έχει επεκταθεί για να ικανοποιεί την ιδιότητα receipt freeness [28] όχι όμως για την ισχυρότερη ιδιότητα του coercion resistance κατά τη γνώση του συγγραφέα.

3.1.3 Επέκταση για ψηφοφορίες με k επιλογές

Είναι φανερό ότι ο περιορισμός για ψηφοφορίες τύπου ΝΑΙ/ΟΧΙ δεν είναι επιθυμητός και είναι απαγορευτικός για τις περισσότερες ρεαλιστικές μεγάλες ψηφοφορίες. Θέλουμε συνεπώς μια

επέκταση του συστήματος για εκλογές με k επιλογές. Αυτή η επέκταση γίνεται με ευθύ τρόπο.

Οι αλλαγές είναι οι εξής:

- Για κάθε δυνατή από τις k ψήφους επιλέγεται ανεξάρτητα από τις αρχές ένας γεννήτορας G_i. Η ανεξαρτησία μπορεί να επιτευχθεί δεδομένης της threshold φύσης της T.
- Κάθε ψήφος που κατατίθεται είναι της μορφής $(x_j, y_j) = (g^{r_j}, G_i g^{sr_j})$ όπου G_i είναι ένας από τους k γεννήτορες.
- Η απόδειξη ορθά σχηματισμένης ψήφου είναι πλέον ένα OR αποδείξεων της μορφής

$$\log_g x = log_{g^s}(\frac{y}{G_1}) \vee \ldots \vee \log_g x = log_{g^s}(\frac{y}{G_k})$$

  επιβεβαιώνεται δηλαδή ότι το plaintext είναι ένα από τα $G_1, \ldots, G_k$
- Το τελικό κρυπτοκείμενο κατασκευάζεται πολλαπλασιάζοντας τα επιμέρους ciphertexts και συνεπώς έχει τη μορφή

$$G_1^{t_1}, \ldots, G_k^{t_k}$$

- Δοκιμάζοντας τις πιθανές επιλογές η T υπολογίζει τα $t_1, \ldots, t_k$ και τα δημοσιοποιεί

Ο υπολογισμός του τελικού αποτελέσματος είναι σαφώς πολύ πιο απαιτητικός από την εκδοχή με ΝΑΙ/ΟΧΙ αλλά για λογικό πλήθος επιλογών k ο υπολογισμός του με brute force δεν είναι απαγορευτικός. Επιπλέον αξίζει να σημειωθεί ότι αυξάνεται γραμμικά ως προς το k το μέγεθος της μη διαλογικής απόδειξης που χρειάζεται να υπολογίσει ο ψηφοφόρος, και το πλήθος των υπολογισμών που πρέπει να εκτελέσει για τον υπολογισμό της. Τέλος, μία σημαντική έλλειψη του συστήματος και στις δύο του εκδοχές είναι ότι δεν στηρίζει write in ψήφους, δηλαδή ο ψηφοφόρος δεν μπορεί να αποκλείνει από τις δοθείσες επιλογές καταθέτοντας μια ψήφο διαφορετικής από αυτές, χαρακτηριστικό σημαντικό σε πολλές ψηφοφορίες.

3.2 Πρωτόκολλα Ψηφοφοριών Βασισμένα σε Τυφλές Υπογραφές

Η δεύτερη μεγάλη κατηγορία πρωτοκόλλων ηλεκτρονικής ψηφοφορίας που εξετάζουμε στηρίζεται σε σχήματα τυφλών υπογραφών. Το χαρακτηριστικότερο τέτοιο σύστημα είναι το FOO [20].Τέτοια συστήματα είναι διαισθητικά πολύ απλά, πολύ αποδοτικά, και διαθέτουν πολύ δυνατά χαρακτηριστικά ιδιωτικότητας. Σε αυτήν την υποενότητα θα μελετήσουμε μία μικρή παραλλαγή του FOO, το OMAFO [36].

3.2.1 Το Πρωτόκολλο OMAFO

Στο πρωτόκολλο αυτό οι συμμετέχοντες είναι N ψηφοφόροι, ένας administrator A και n μέλη μίας εφορευτικής επιτροπής T. Θεωρούμε ότι το πολύ t από τα μέλη της εφορευτικής επιτροπής μπορεί να είναι διεφθαρμένα και να αποκλίνουν από το πρωτόκολλο. Για την επικοινωνία

θεωρούμε ανώνυμο κανάλι μέσα από το οποίο οι συμμετέχοντες γράφουν σε ένα Bulletin Board. Οι αρχές μοιράζονται ένα ιδιωτικό κλειδί ενός σχήματος τυφλών υπογραφών και ένα ιδιωτικό κλειδί ενός σχήματος κρυπτογραφίας δημοσίου κλειδιού. Θεωρούμε ότι κάθε ψηφοφόρος μπορεί να μιλάει αυθεντικοποιημένα με την αρχή A.

Στάδιο Εγγραφής

- Κάθε ψηφοφόρος επιλέγει μία ψήφο v_i.
- Κρυπτογραφεί την επιλογή του με το κλειδί των T σε $x_i = E_T(v_i)$.
- Εκτελεί το πρωτόκολλο τυφλής υπογραφής Sign με τον A για το μήνυμα x_i. Ο A για να δεχτεί να συμμετάσχει στο πρωτόκολλο κοιτάει κατά πόσο ο ψηφοφόρος έχει δικαίωμα ψήφου και κατά πόσο δεν έχει ψηφίσει. Ο ψηφοφόρος, αν ικανοποιεί τα προηγούμενα, παίρνει την υπογραφή σ_i στο x_i.
- Για επαληθευσιμότητα ο A δημοσιοποιεί τα Transcripts των πρωτοκόλλων που έτρεξε δηλαδή δημοσιοποιεί μία λίστα $(ID_i, trans_i)$

Στάδιο Κατάθεσης Ψήφων

- Ο ψηφοφόρος V_i καταθέτει μέσω ενός ανώνυμου καναλιού την ψήφο του (x_i, σ_i)

Στάδιο Καταμέτρησης Ψήφων

- Κάθε μέλος της επιτροπής T ελέγχει την εγκυρότητα στις υπογραφές που κατατέθηκαν. Κάθε άκυρη ψήφος αγνοείται από το σημείο αυτό και μετά.
- Συνεργατικά αποκρυπτογραφούνται όλες οι κατατεθειμένες ψήφοι και βγαίνει το αποτέλεσμα.

Τα ανώνυμα κανάλια και οι ιδιότητες των τυφλών υπογραφών διασφαλίζουν την ιδιωτικότητα των ψηφοφόρων. Παρατηρήστε ότι εξ ορισμού το transcript του πρωτοκόλλου υπογραφής που δημοσιοποιείται δεν περιέχει καμία πληροφορία που να μπορεί να συνδέσει κάποιον ψηφοφόρο με κάποια ψήφο. Η επαληθευσιμότητα εγγυάται στο ότι το πολύ t μέλη της επιτροπής T μπορούν να είναι διεφθαρμένα. Ως προς τον Administrator, δεν μπορεί να παρεκλείνει από το πρωτόκολλο αφού δημοσιοποιεί τα transcripts και η επικοινωνία είναι αύθεντικοποιημένη. Κάθε τέτοια προσπάθεια (άρνηση υποβολής υπογραφής ή υποβολή ψευδών υπογραφών) μπορεί να γίνει αντιληπτή από έναν τρίτο, δεδομένου ότι όλοι οι ψηφοφόροι που πήραν τυφλή υπογραφή καταθέτουν ψήφο.

Τα συστήματα που βασίζονται σε τυφλές υπογραφές έχουν πολλά θετικά στοιχεία. Ήδη αναφερθήκαμε στην ιδιωτικότητα. Αν θεωρήσουμε ότι τα ανώνυμα κανάλια είναι πληροφοριοθεωριτικά ασφαλή έχουμε παντοτινή ιδιωτικότητα της ψήφου μας λόγω της στατιστικής ασφάλειας που μας προσφέρουν οι τυφλές υπογραφές, κάτι πολύ επιθυμητό αφού η κρυπτογραφική ασφάλεια έχει χρονικό τέλος όπως εξηγούμε στην εισαγωγή. Ένα ακόμη πλεονέκτημα

είναι ότι επιτρέπουν write-in ψήφους, δηλαδή ο ψηφοφόρος δεν είναι περιορισμένος σε ένα προκαθορισμένο σύνολο επιλογών σε αντίθεση με τα περισσότερα άλλα πρωτόκολλα ηλεκτρονικής ψηφοφορίας.

Τέλος θέλουμε να τονίσουμε την απλότητα που διέπουν τέτοια συστήματα. Ουσιαστικά προσομοιώνουν με πολύ καλό τρόπο μία παραδοσιακή ψηφοφορία. Μπορεί κανείς να σκεφτεί το blinded μήνυμα που κατατίθεται ως την υπογραφή το φάκελο. Ο δικαστικός αντιπρόσωπος (η A) μας υπογράφει έναν φάκελο μετά από έλεγχο και εμείς κρυφά συμπληρώνουμε κάτι μέσα σε αυτόν (το ψηφοδέλτιο). Η καταμέτρηση στη συνέχεια γίνεται από ένα σύνολο οντοτήτων με αντικρουόμενα συμφέροντα (κομματικοί αντιπρόσωποι) και βγαίνει το τελικό αποτέλεσμα.

Το πρωτόκολλο αυτό έχει παραλλαχθεί ώστε να έχει την ιδιότητα receipt freeness στο [38]. Η κύρια φιλοδοξία αυτής της εργασίας είναι να ενισχύσει το πρωτόκολλο αυτό (ή καλύτερα μία μίξη αυτού με το JCJ που παρουσιάζεται στη συνέχεια) ώστε να έχει την ιδιότητα της ανθεκτικότητας σε επιθέσεις εξαναγκασμού. Το εν λόγω αποτέλεσμα παρουσιάζεται στο κεφάλαιο 4.

3.3 Το πρωτόκολλο JCJ

Το 2002 οι Aris Juels, Dario Catalano και Markus Jackobson δημοσίευσαν μια εργασία στην οποία ορίζανε την έννοια του coercion resistance για συστήματα ηλεκτρονικής ψηφοφορίας, ένα πλάισιο για τον έλεγχό της και ένα πρωτόκολλο που ικανοποιεί την συγκεκριμένη ιδιότητα [31]. Στο παρόν κεφάλαιο θα παρουσιάσουμε το πλαίσιο το οποίο έθεσαν για τον έλεγχο του coercion resistance και το πρωτόκολλο τους, το οποίο στάθηκε ως βάση για πληθώρα άλλων πρωτοκόλλων ηλεκτρονικής ψηφοφορίας τα επόμενα χρόνια [52],[51],[11],[44],[1], [4], [2],[50],[3].

3.3.1 Το πλαίσιο για έλεγχο Coercion Resistance

Όπως έχουμε δει μια σημαντική ιδιότητα για τα συστήματα ηλεκτρονικής ψηφοφορίας είναι να είναι ανθεκτικά σε επιθέσεις εξαναγκασμού. Θα θέλαμε δηλαδή το πρωτόκολλο να λειτουργεί με τρόπο που να μην επιτρέπει σε κάποιον κακόβουλο αντίπαλο αν προστάξει ένα μέλος (ψηφοφόρο) να δράσει με συγκεκριμένο τρόπο (εν προκειμένω να ψηφίσει τον αγαπημένο υποψήφιο του αντιπάλου). Η συμπεριφορά των χρηστών δεν μπορεί να καθοριστεί από το ίδιο το πρωτόκολλο, αλλά το πρωτόκολλο μπορεί να αποτρέψει αυτήν τη συμπεριφορά με το να μην δίνει τη δυνατότητα στον αντίπαλο να επιβεβαιώσει αν το μέλος που πρόσταξε υπάκουσε ή όχι. Έτσι δεν έχει νόημα μια τέτοια προσταγή αφού ο ψηφοφόρος μπορεί απλά να ισχυριστεί ότι υπέκυψε στη θέληση του αντιπάλου ενώ στην πραγματικότητα να έδρασε όπως επιθυμούσε. Έτσι φαινόμενα όπως στιγνή προσταγή ή πώληση ψήφου αποτρέπονται από το πρωτόκολλο. Όπως είδαμε η ιδιότητα του receipt freeness δεν επαρκεί αφού για παράδειγμα ο αντίπαλος μπορεί να απαιτήσει από τον ψηφοφόρο να απέχει ή να ρίξει άκυρη ψήφο. Το JCJ έχει σκοπό

να είναι ανθεκτικό σε κάθε τέτοιου είδους επίθεση. Παρακάτω παρουσιάζουμε το πλαίσιο για τον έλεγχο του coercion resistance όπως τέθηκε στη συγκεκριμένη εργασία.

3.3.1.1 Το μοντέλο

Στο πρωτόκολλο ψηφοφορίας συμμετέχουν η Registration Authority $\mathcal{R}$ που αποτελείται από r μέρη που διαμοιράζονται το αντίστοιχο μυστικό κλειδί, την Tally Authority $\mathcal{T}$ που αποτελείται από t μέλη που επίσης διαμοιράζονται ένα μυστικό κλειδί και το σύνολο των n ψηφοφόρων $\mathcal{V}$.

Θεωρούμε ότι οι ψηφοφόροι έχουν να διαλέξουν από ένα σύνολο υποψηφίων $\mathcal{C} = \{c_1, \ldots, c_m\}$. Όλοι οι συμμετέχοντες έχουν πρόσβαση σε ένα Bulletin Board $\mathcal{BB}$. Το αποτέλεσμα της καταμέτρησης των ψήφος είναι ένα διάνυσμα $\mathbf{X} = \{x_1, \ldots, x_m\}$ όπου x_j είναι ο αριθμός των ψήφων που έλαβε ο ψηφοφόρος j.

Έχουμε τα παρακάτω υπόπρωτόκολλα που αποτελούν το πρωτόκολλο ηλεκτρονικής ψηφοφορίας. Τα θεωρούμε ως συναρτήσεις

- **Register:** $register(sk_{\mathcal{R}}, i, 1^k) = (sk_i, pk_i)$
  Η συνάρτηση αυτή παίρνει ως είσοδο το μυστικό κλειδί της $\mathcal{T}$ και τον αναγνωριστικό αριθμό του ψηφοφόρου i και του αποδίδει ένα ζεύγος δημοσίου-ιδιωτικού κλειδιού.
- **Vote:** $vote(sk_i, pk_{\mathcal{T}}, \mathcal{C}, \beta, 1^k) = b$
  Η συνάρτηση αυτή παίρνει ως είσοδο το μυστικό κλειδί ενός ψηφοφόρου, το δημόσιο κλειδί των καταμετρητών, την κωδικοποίηση του συνόλου των υποψηφίων και την επιλογή του ψηφοφόρου β και φτιάχνει μία ψήφο προς καταμέτρηση.
- **tally:** $tally(sk_{\mathcal{T}}, \mathcal{BB}, C, \{pk_i\}_{\text{ß}\in\mathcal{V}}, 1^k) = (\mathbf{X}, P)$
  Η συνάρτηση αυτή παίρνει ως είσοδο το μυστικό κλειδί των καταμετρητών, τα περιεχόμενα του bulletin board, τα δημόσια κλειδία των ψηφοφόρων και την κωδικοποίηση των υποψηφίων και δίνει το αποτέλεσμα της καταμέτρησης και μια μη διαδραστική απόδειξη ορθότητας.
- **verify:** $verify(pk_{\mathcal{T}}, \mathcal{BB}, C, \mathbf{X}, P) = b$
  Η συνάρτηση αυτή παίρνει ως είσοδο το δημόσιο κλειδί των καταμετρητών, τα περιεχόμενα του bulletin board, την κωδικοποίηση των υποψηφίων, το αποτέλεσμα της καταμέτρησης και την αντίστοιχη απόδειξη και δίνει $b \in \{0, 1\}$ με το b να εκφράζει την ορθότητα της καταμέτρησης (0 λάθος, 1 σωστή).

3.3.1.2 Υποθέσεις Ασφάλειας

Θεωρούμε ότι ο αντίπαλος μπορεί να διαφθείρει μία μειονότητα των $\mathcal{T}$ και $\mathcal{R}$ αλλά αφού έχουν ήδη διαμοιραστεί τα μυστικά κλειδιά τους με ασφαλή τρόπο. Επιπλέον υποθέτουμε ότι ο αντίπαλος μπορεί να διαφθείρει μια μειοψηφία των ψηφοφόρων αλλά μετά το στάδιο εγγραφής τους. Όλα τα corruptions του αντιπάλου είναι στατικά δηλαδή δεν αλλάζουν κατά τη διάρκεια

εκτέλεσης του πρωτοκόλλου με δυναμικό τρόπο. Τέλος θεωρούμε ότι οι ψηφοφόροι γράφουν στο bulletin board μέσω ανώνυμων καναλιών.

3.3.1.3 Το πλαίσιο για έλεγχο coercion resistance

Συνοπτικά, ορίζουμε ένα παιχνίδι στο οποίο εκτελείται μια ψηφοφορία και ο ψηφοφόρος που εξαναγκάζεται από τον αντίπαλο να έχει μία συγκεκριμένη συμπεριφορά είτε υποκύπτει είτε όχι. Σκοπός του αντιπάλου είναι να μαντέψει την συμπεριφορά του ψηφοφόρου. Στη συνέχεια ορίζουμε το αντίστοιχο παιχνίδι σε έναν ιδεατό κόσμο που το πρωτόκολλο είναι εξ ορισμού ασφαλές. Αν κάθε αντίπαλος που νικάει το πραγματικό πρωτόκολλο μπορεί να προσομοιωθεί και να χρησιμοποιηθεί για μια επίθεση στο ιδεατό πρωτόκολλο που νικάει σχεδόν με ίδια πιθανότητα και δεδομένου ότι το ιδεατό είναι ασφαλές τότε μπορούμε να συμπεράνουμε ότι και το πραγματικό είναι ασφαλές. Αυτή η μεθοδολογία είναι ευρύτατα χρησιμοποιημένη και αποδεκτή για την μοντελοποίηση ασφαλείας multi party υπολογισμού και προτείνουμε στον ενδιαφερόμενο αναγνώστη να ανατρέξει στα [21], [22],[35] για την μελέτη της.

Παρατηρούμε ότι δεν είναι δυνατόν να πετύχουμε πλήρες coercion resistance. Αρκεί να φανταστούμε το σενάριο όπου ο αντίπαλος ξέρει τις προθέσεις όλων των ψηφοφόρων. Το αποτέλεσμα τότε καθορίζει για τον αντίπαλο ακριβώς το πως έδρασε ο υπό εξαναγκασμό ψηφοφόρος. Όμως σε αυτήν την περίπτωση το ίδιο θα γινόταν και στο ιδεατό μοντέλο όπως θα δούμε. Αυτό που θέλουμε να αντικατοπτρίσουμε είναι ότι ο αντίπαλος δεν μαθαίνει πράγματα από την εκτέλεση του πρωτοκόλλου. Δηλαδή η γνώση που έχει μετά την εκτέλεση είναι (υπολογιστικά) ίδια με αυτή που θα είχε αν απλά μάθαινε το αποτέλεσμα της ψηφοφορίας.

Ορίζουμε λοιπόν μια κατανομή D_{n,n_c} που είναι γνωστή στον αντίπαλο και αποτελείται από στοιχεία $(\beta_1, \ldots, \beta_n) \in (C \cup \phi \cup \lambda)^n$. Αυτή εκφράζει την αβεβαιότητα του αντιπάλου ως προς τις επιλογές των ψηφοφόρων που δεν ελέγχει, με ϕ να είναι η αποχή και λ η ψήφος με λάθος credential.

Θεωρούμε ότι οι ψηφοφόροι έχουν στη διάθεσή τους μια συνάρτηση

$$fakekey(pk_{\mathcal{T}}, sk_i, pk_i) = \tilde{sk_i}$$

Η συνάρτηση αυτή μέσω του δημόσιου κλειδιού της επιτροπής καταμέτρησης και του ζεύγους κλειδιών του ψηφοφόρου παράγει ένα ψεύτικο ιδιωτικό κλειδί του ψηφοφόρου που θα δώσει στον αντίπαλο και θα ισχυριστεί ότι είναι το αληθινό.

Με βάση αυτά ορίζουμε τα δύο παιχνίδια c-resist και c-resist-ideal. Θεωρούμε ότι ο αντίπαλος είναι ένας statefull αλγόριθμος με την έννοια ότι σε κάθε φάση του παιχνιδιού εκτελεί άλλες εντολές (η κατάσταση δίνεται ως input.Στα παρακάτω $\mathcal{A}$ είναι ο αντίπαλος και $\Leftarrow$ δηλώνει append στον bulletin board.

Algorithm: c-resist-game

Input : $1^k, n, t, C$

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

$V^* \leftarrow \mathcal{A}(V, \textbf{corrupt voters})$
$\{(sk_i, pk_i) \leftarrow register(sk_{\mathcal{R}}, i, 1^k)\}_{i=1}^{n}$
$(voter, vote) \leftarrow \mathcal{A}(\{sk_i\}_{i \in V^*} \textbf{coerce voter})$
if $voter \notin V \backslash V^*$ *or* $vote \notin C \cup \{\phi\}$ **then**
    output 0
end
$b \leftarrow^R \{0,1\}$
if $b = 0$ **then**
    $\tilde{sk} \leftarrow fakekey(pk_{\mathcal{T}}, sk, pk)$
    $BB \Leftarrow vote(sk, PK_T, C, vote, 1^k)$
else
    $\tilde{sk} \leftarrow sk$
end
$\mathcal{BB} \Leftarrow vote(\{sk_i\}_{i \in V \backslash V^* \cup voter}, pk_{\mathcal{T}}, C, D_{V \backslash V^* \cup voter}, 1^k)$
$\mathcal{BB} \Leftarrow \mathcal{A}(sk, \{sk_i\}_{i \in voter \cup V^*}, \mathcal{BB})$
$(X, P) \leftarrow tally(sk_{\mathcal{T}}, \mathcal{BB}, C, \{pk_i\}_{i \in V}, 1^k)$
$b' \leftarrow \mathcal{A}(\mathbf{X}, P, \mathcal{BB}, \textbf{Guess})$
output $b' == b$

Algorithm: c-resist-ideal-game

Input : $1^k, n, t, C$

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

$V^* \leftarrow \mathcal{A}'(V, \textbf{corrupt voters})$
$\{(sk_i, pk_i) \leftarrow register(sk_{\mathcal{R}}, i, 1^k)\}_{i=1}^{n}$
$(voter, vote) \leftarrow \mathcal{A}'(\textbf{coerce voter})$
if $voter \notin V \backslash V^*$ *or* $vote \notin C \cup \{\phi\}$ **then**
  output 0
end
$b \leftarrow^R \{0,1\}$
if $b = 0$ **then**
  $BB \Leftarrow vote(sk, PK_T, C, vote, 1^k)$
end
$\tilde{sk} \leftarrow sk$
$BB \Leftarrow vote(\{sk_i\}_{i \in V \backslash V^* \cup voter}, PK_{\mathcal{T}}, C, D_{V \backslash V^* \cup voter}, 1^k)$
$\mathcal{BB} \Leftarrow \mathcal{A}'(sk, \{sk_i\}_{i \in voter \cup V^*}, \mathcal{BB})$
$(X, P) \leftarrow idealtally(sk_{\mathcal{T}}, \mathcal{BB}, C, \{pk_i\}_{i \in V}, 1^k)$
$b' \leftarrow \mathcal{A}(\mathbf{X}, P, \Gamma, \textbf{Guess})$
output $b' == b$

Με λίγα λόγια στο πραγματικό πείραμα ο $\mathcal{A}$ διαφθείρει κάποιους ψηφοφόρους και επιλέγει στόχο για εκβιασμό. Αυτός δεν θα πρέπει να είναι στους corrupted . Ρίχνεται ένα κέρμα και αν βγει 0 ο ψηφοφόρος δεν υποκύπτει στον εκβιασμό και δίνει ψεύτικο κλειδί στον $\mathcal{A}$. Αλλιώς δίνει το πραγματικό του ιδιωτικό κλειδί (αυτή είναι η πιο γενική περίπτωση). Αφού βάλουν στον $\mathcal{BB}$ τις μυστικές ψήφους τους οι τίμιοι ψηφοφόροι και ο $\mathcal{A}$ για τους διεφθαρμένους, δίνεται το αποτέλεσμα της ψηφοφορίας. Ο $\mathcal{A}$ καλείται να απαντήσει στο τι τελικά έκανε ο coerced ψηφοφόρος και αν το βρει νικάει.

Στο ιδεατό πείραμα, υπάρχουν οι εξής διαφορές: Ο $\mathcal{A}$ δεν χρησιμοποιεί τα μυστικά κλειδιά των χρηστών για να κάνει coerce αφού θα θέλαμε αυτά του να μην του δίνουν καμία πληροφορία. Επιπλέον ο voter πάντα δίνει το μυστικό κλειδί στον αντίπαλο αλλά ψηφίζει ανάλογα με το κέρμα. Αυτό εκφράζει την επιθυμία μας το κλειδί να μην περιέχει καμία πληροφορία για το τι έκανε ο ψηφοφόρος. Τέλος το αποτέλεσμα δίνεται από μια ιδεατή συνάρτηση idealtally η οποία κάνει ότι θα έπρεπε να γινόταν, δηλαδή υπολογίζει τις ψήφους μόνο με σωστά credentials (χωρίς να διπλομετράει) και ανάλογα με το κέρμα είτε μετράει την ψήφο του ψηφοφόρου είτε του $\mathcal{A}$. Ο αντίπαλος εδώ, για να μαντέψει δεν βλέπει τα περιεχόμενα του $\mathcal{BB}$, δηλαδή δεν παίρνει καμία πληροφορία για τις κατατεθειμένες ψήφους. Το μόνο που βλέπει είναι το τελικό αποτέλεσμα και το πλήθος των άκυρων ψήφων που κατατέθηκαν.

3.3.1.4 Τα απαιτούμενα primitives

- **Threshold cryptosystem with reencryption**
  Οι αρχές μοιράζονται το κλειδί ενός συστήματος κρυπτογράφησης δημοσίου κλειδιού με δυνατότητα reencryption (πχ El Gamal)
- **PETS (Plaintext Equivalece Test)** Με είσοδο δύο κρυπτοκείμενα δίνει 1 αν και μόνο αν είναι κρυπτογραφήσεις του ίδιου μηνύματος. Τα PETS δημοσιεύτηκαν στο [29] μαζί με ένα ασφαλές κατανεμημένο πρωτόκολλο για τον υπολογισμό τους.
- **Mix Networks** Με είσοδο ένα διάνυσμα κρυπυοκειμένων $C = (C_1, \ldots, C_n)$ δίνει μια τυχαία αντιμετάθεση των κρυπτοκειμένων αφού πρώτα τα έχει επανακρυπτογραφήσει
- **Proofs of Knowledge** Χρησιμοποιούμε σε διάφορα σημεία του πρωτοκόλου Non Interactive Zero Knowledge Proofs of Knowledge (NIZKPoK) για να διασφαλίσουμε ότι οι παίχτες ακολουθούν το πρωτόκολλο

Τα διάφορα primitives τα χρησιμοποιούμε στο πρωτόκολλο ως ideal functionalities ($D\tilde{E}C$, $P\tilde{E}T$, $\tilde{MN}$) κλπ. Δεδομένου ότι υπάρχει secure MPC πρωτόκολλο γι αυτά μπορούμε να το κάνουμε αυτό και το τελικό μας πρωτόκολλο παραμένει ασφαλές.

3.3.1.5 Το πρωτόκολλο JCJ

Στην παρούσα υποενότητα περιγράφουμε τις φάσεις του πρωτοκόλλου.

- **Setup:** Παράγονται τα ζεύγη κλειδιών $(sk_{\mathcal{T}}, pk_{\mathcal{T}})$ και $(sk_{\mathcal{R}}, pk_{\mathcal{R}})$. Αυτά μαζί με τις άλλες δημόσιες παραμέτρους του συστήματος (ομάδα, γεννήτορα κλπ) δημοσιοποιούνται.
- **Registration:** Κάθε ψηφοφόρος V_i πηγαίνει για εγγραφή στην $\mathcal{R}$. Αφού επιβεβαιωθεί ότι έχει δικαίωμα ψήφου λαμβάνει $\sigma_i \in_R \mathcal{G}$ με distributed τρόπο από τα μέλη της επιτροπής. Η επιτροπή δημοσιοποιεί το κρυπτογραφημένο credential $S_i = E_{\mathcal{T}}(\sigma_i)$ στο voter roll $\mathbf{L}$ υπογεγραμένο.
- **Candidate Slate Publication:** Η $\mathcal{R}$ δημοσιοποιεί τα ονόματα των υποψηφίων και τα στοιχεία του χώρου μηνυμάτων $\mathbf{C}$ που αντιστοιχούν σε αυτά.
- **Voting:** Κάθε ψηφοφόρος V_i επιλέγει έναν υποψήφιο $c_j \in \mathcal{C}$ και παράγει τα εξής δύο κρυπτοκείμενα

$$E^1 = E_{\mathcal{T}}(c_j)$$

$$E^2 = E_{\mathcal{T}}(\sigma_i)$$

  Στη συνέχεια κατασκευάζει ένα σύνολο αποδείξεων Pf που περιλαμβάνουν

  1. Απόδειξη ότι ξέρει αποκρυπτογράφηση του E^1
  2. Απόδειξη ότι ξέρει αποκρυπτογράφηση του E^2
  3. Απόδειξη ότι η αποκρυπτογράφηση του E_1 ανήκει στο $\mathbf{C}$
  4. Μέσω του ανώνυμου καναλιού τυπώνει στον $\mathcal{BB}$ το ballot του $B = (E^1, E^2, Pf)$
- **Tally:** Η επιτροπή καταμέτρησης $\mathcal{T}$ ακολουθεί τα εξής βήματα
  1. **Έλεγχος Αποδείξεων:** Για κάθε ballot επαληθεύει τις αποδείξεις και στα επόμενα αγνοεί τα ψηφοδέλτια με άκυρες αποδείξεις.
  2. **Απαλοιφή Διπλών ψήφων:** Σε όσα ψηφοδέλτια μείναν παίρνει ανά δύο όλα τα δεύτερα συστατικά τους (τα E_2) και κάνει $P\tilde{E}T$ για να διαγράψει διπλές ψήφους.
  3. **Mixing:** Περνάει το σύνολο των ψηφοδελτίων που έμεινε από το $\tilde{MN}$.
  4. **Έλεγχος πιστοποιητικών:** Αφού περάσει το voter roll $\mathbf{L}$ από το mix network κάνει $P\tilde{E}T$ όλων των δεύτερων συστατικών των ψηφοδελτίων με τα στοιχεία του voter roll $\mathbf{L}$ για να κρατήσει μόνο τα ψηφοδέλτια με έγκυρα credentials.
  5. **Άνοιγμα Ψήφων:** Ανοίγει τα έγκυρα ballots που περάσαν το $P\tilde{E}T$ και βλέπουν όλοι το αποτέλεσμα.

3.3.1.6 Ασφάλεια και αποδοτικότητα

Οι Juels, Catalano και Jackobson παρουσιάζουν στην εργασία τους μια απόδειξη ασφάλειας για το πρωτόκολλό τους ως προς το coercion resistance στο πλαίσιο το οποίο έθεσαν και είδαμε. Συγκεκριμένα αποδεικνύουν ότι χρησιμοποιώντας μια παραλλαγή του κρυπτοσυστήματος El Gamal αν κάποιος αλγόριθμος μπορούσε να ξεχωρίσει αν παίζει το παιχνίδι c-resist ή το

c-resist-ideal με μη αμελητέα πιθανότητα τότε θα μπορούσε να χρησιμοποιηθεί για να λύσει με μη αμελητέα πιθανότητα το Decisional Diffie-Hellman πρόβλημα.

Το μεγάλο μειονέκτημα του πρωτοκόλλου είναι η αποδοτικότητά του. Αν ορίσουμε ως v το σύνολο των ψήφων με σωστές αποδείξεις και ως v' το σύνολο των ψήφων χωρίς τις διπλές, το πλήθος των ελέγχων $P\tilde{E}T$ που πρέπει να γίνουν είναι $\mathcal{O}(|v|^2 + |v'||L|)$. Αυτό είναι απαγορευτικό ακόμα και για εκλογές με σχετικά μικρό αριθμό συμμετεχόντων. Πολλές εργασίες που ακολούθησαν [52],[51],[11],[44],[1], [4], [2],[50],[3] προσπάθησαν να κάνουν τον παραπάνω αριθμό ελέγχων γραμμικό ως προς το πλήθος των ψηφοφόρων και των ψήφων. Στη συνέχεια θα παρουσιάσουμε τις βασικές ιδέες μερικών εξ αυτών.

3.4 Οι βελτιώσεις των Smith και Weber et al

Όπως είδαμε το JCJ είναι ένα ασφαλές ως προς το coersion resistance πρωτόκολλο ηλεκτρονικής ψηφοφορίας, αλλά μη πρακτικό για μεγάλης κλίμακας εκλογές λόγω της υψηλής υπολογιστικής του πολυπλοκότητας. Αυτή πηγάζει από τον τετραγωνικό ως προς τις κατατεθιμένες ψήφους και ψηφοφόρους αριθμό plaintext equivalence text που χρείαζεται να γίνουν για να εντοπιστούν διπλές ψήφοι και ψηφοδέλτια που αντιστοιχούν σε αποδεκτά credentials. Μια από τις πρώτες βελτιώσεις στον χρόνο εκτέλεσης του JCJ δόθηκε από τον Smith [50] και βελτιώθηκε σημαντικά από τους Weber, Araujo και Buchmann [52]. Σε αυτό το υποκεφάλαιο θα εξετάσουμε τη δεύτερη. Θα δούμε ότι ο χρόνος που χρειάζεται πέφτει από τετραγωνικό σε γραμμικό αλλά έχει ένα τίμημα: το νέο πρωτόκολλο παύει να έχει την ιδιότητα του Coercion Resistance. Παρόλα αυτά η ιδέα που εισάγεται είναι πολύ χρήσιμη και χρησιμοποιήθηκε από πολλά άλλα αντίστοιχα πρωτόκολλα στο μέλλον για τη διαγραφή των διπλών ψήφων.

3.4.1 Το πρωτόκολλο των WAB

Το πρόβλημα των διπλών ψήφων και της σύγκρισης των credentials με τα αυθεντικά έχουν ίδια χαρακτηριστικά. Στη μη κρυπτογραφική τους έκδοση είναι και τα δύο της μορφής: Δοθέντος μιας (μεγάλης) λίστας με αντικείμενα να ελεγχεί ποια από αυτά ταυτίζονται. Αυτό είναι ένα αρκετά μελετημένο αλγοριθμικό πρόβλημα και η λύση γίνεται χρησιμοποιόντας lookup hashtables. Πολύ συνοπτικά αυτά είναι μια δομή δεδομένων που αντιστοιχίζουν κλειδιά σε αντικειμενα με τα αντικείμενα που έχουν μία ιδιότητα (πχ είναι ίδια) να αντιστοιχίζονται στο ίδιο κλειδί. Αν λοιπόν δεν είχαμε τις ιδιότητες 'κρυμμένες' από τα κρυπτογραφικά πρωτόκολλα με μια τέτοια δομή θα υπολογίζαμε τα κλειδιά και θα βρίσκαμε τις διπλές ψήφους και τις ψήφους που αντιστοιχούν σε πραγματικά credentials σε αναμενόμενο γραμμικό χρόνο χρησιμοποιόντας κατάλληλη δομή. Αυτή ακριβώς είναι η ιδέα του που εισήγαγε ο Smith. Το ζήτημα είναι με ποιον τρόπο θα βγάλουμε από το παιχνίδι την κρυπτογραφία χωρίς να επηρεάζεται η ασφάλεια του συστήματος. Για μια αναλυτική περιγραφή των hashtables παραπέμπουμε τον αναγνώστη στο [13].

3.4.1.1 Κατανεμημένος υπολογισμός Hashkeys

Θέλουμε οι αρχές καταμέτρησης με ασφαλή τρόπο να υπολογίζουν ένα αποτύπωμα για κάθε κατατεθιμένο ψηφοδέλτιο. Δηλαδή μια συνάρτηση h με πεδίο ορισμού των χώρο των κρυπτοκειμένων. Αυτή με είσοδο ένα ψηφοδέλτιο θα υπολογίζει το αποτύπωμα (το κλειδί του lookup table). Δηλαδή $h(E_T(g^r, mh^r)) = key(m)$. Στη συνέχεια κοιτώντας τα κλειδιά μπορούμε να βρούμε collisions στα credentials. Στην περίπτωση εξέτασης για διπλές ψήφους, collision σημαίνει διπλή ψήφος ενώ στην εξέταση εγκυρότητας των credentials σημαίνει έγκυρη ψήφος αφού το credential της ψήφου υπάρχει και στο Voter roll. Η συνάρτηση αυτή θα είναι παραμετροποιημένη από κάποιον $z \in \mathbb{Z}_q$ και θα δίνει $h_z(E_T(g^r, mh^r) = m^z$ όπου το πεδίο τιμών είναι τα στοιχεία της ομάδας $\mathcal{G}$ που χρησιμοποιήται από το El Gamal. Μια σημαντική παρατήρηση είναι ότι αν το z είναι άγνωστο τότε το m^z είναι ομοιόμορφα κατανεμημένο στην $\mathcal{G}$ και συνεπώς δεν περιέχει πληροφορία για το m. Επειδή η ασφάλεια θέλουμε να ισχύει ακόμα και αν μια μειοψηφία της $\mathcal{T}$ είναι διεφθαρμένη θα δείξούμε πως μπορούμε να υπολογίσουμε κατανεμημένα την συνάρτηση κατακερματισμού h_z.

- Μέσω ενός πρωτοκόλου διαμοιρασμού μυστικού τα μέλη της $\mathcal{T}$ διαλέγουν ένα z. Κάθε μέλος παίρνει το μερίδιό του, έστω z_i. Αν έχουμε n μέλη υποθέτουμε ότι k από τα n μπορούν να κατασκευάσουν το z αλλά όχι $k-1$. Επίσης δεσμεύεται δημόσια για το z_i δημοσιοποιώντας το g^{z_i} και απόδειξη γνώσης διακριτού λογαρίθμου. Υπενθυμίζουμε η ανακατασκευή μπορεί να γίνει από ένα υποσύνολο K μεγέθους k ως εξής

$$z = \sum_{i \in K} z_i \lambda_i(0)$$

- Με είσοδο μια κρυπτογράφηση El Gamal $c = (g^r, mh^r)$ κάθε μέλος υπολογίζει $c^{z_i\lambda_i(0)} = (g^{rz\lambda_i(0)}, m^{z\lambda_i(0)}h^{rz\lambda_i(0)}) = E_T(m^{z\lambda_i(0)}; z\lambda_i(0)r)$ συνοδευόμενο από ZKPoK για ορθότητα υπολογισμού.

- Πολλαπλασιάζοντας αυτές τις ποσότητες και δεδομένων των ομομορφικών ιδιοτήτων του El Gamal παίρνουμε

$$\prod_{i \in K} E_T(m^{z_i\lambda_i(0)}; z_i\lambda_i(0)r) = E_T(m^{(\sum_{i \in K} z_i\lambda_i(0))}; \sum_{i \in K} z_i\lambda_i(0)r) = E_T(m^z; zr)$$

- Το κρυπτοκείμενο που προκύπτει αποκρυπτογραφείται κατανεμημένα και προκύπτει το ζητούμενο fingerprint m^z.

Παρατηρούμε ότι η όλη διαδικασία γίνεται για να διώξουμε από τις κρπτογραφήσεις την τυχαιότητα και να μπορούμε να έχουμε αιτιοκρατικό fingerprint χωρίς να αποκαλύψουμε το ciphertext. Φυσικά αυτό δεν θα μπορούσαμε να το κάνουμε κατ ευθείαν στα κρυπτοκείμενα αφού αυτό θα υπονοούσε ότι μπορούμε να ξεχωρίσουμε δύο κρυπτογραφήσεις κάτι που δεν γίνεται λόγω της ασφάλειας του κρυπτοσυστήματος. Επίσης τα μυστικά δεν θα μπορούσαν να κατατίθενται με αιτιοκρατικό σύστημα κρυπτογράφησης καθώς όπως έχουμε αναφέρει αυτό δεν θα μας παρείχε ασφάλεια. Τέλος, το ίδιο το κρυπτοκείμενο στο παραπάνω πρωτόκολλο δεν

αποκρυπτογραφείται ποτέ κατά τη διάρκεια του πρωτοκόλλου και συνεπώς δεν αποκαλύπτεται πληροφορία για το credential σε πιθανώς corrupted μέλη της $\mathcal{T}$.

3.4.2 Το πρωτόκολλο

- **Setup:** Γίνεται όπως στο JCJ. Παράγονται τα ζεύγη κλειδιών $(sk_{\mathcal{T}}, pk_{\mathcal{T}})$ και $(sk_{\mathcal{R}}, pk_{\mathcal{R}})$. Αυτά μαζί με τις άλλες δημόσιες παραμέτρους του συστήματος (ομάδα, γεννήτορα κλπ) δημοσιοποιούνται.
- **Registration:** Γίνεται όπως και στο JCJ . Μετά από την φάση αυτή κάθε ψηφοφόρος έχει ένα μυστικό credential η κρυπτογράφηση του οποίου είναι υπογεγραμμένη σε έναν $\mathcal{BB}$ και αναφερόμαστε σε αυτόν ως το Voter Roll
- **Election Setup:** Η $\mathcal{R}$ δημοσιοποιεί τα ονόματα των υποψηφίων και τα στοιχεία του χώρου μηνυμάτων $\mathbf{C}$ που αντιστοιχούν σε αυτά όπως στο JCJ. Στη συνέχεια οι $\mathcal{T}$ δημιουργούν κατανεμημένα δύο hashkeys j, k με secret sharing και δεσμεύονται στα μερίδιά τους με τον τρόπο που περιγράφηκε παραπάνω.
- **Voting:** Η ψηφοφορία γίνεται ακριβώς όπως στο JCJ. Υπενθυμίζουμε ότι κάθε ψηφοδέλτιο είναι της μορφής (C, S, Pf) όπου $C = E_T(c_i)$ η επιλογή του ψηφοφόρου κρυπτογραφημένη, $S = E_T(\sigma_i)$ η κρυπτογράφηση του credential του ψηφοφόρου (φυσικά διαφορετική από την αντίστοιχη του Voter Roll) και Pf ένα κατάλληλο σύνολο αποδείξων που επιβεβαιώνει ότι ο ψηφοφόρος λειτουργεί βάσει του πρωτοκόλλου.
- **Tally:**
  1. Οι talliers ελέγχουν όλες τις αποδείξεις Pf και μαρκάρουν αυτά με τις έγκυρες. Στα επόμενα βήματα μόνο αυτά λαμβάνονται υπ΄ όψην στη συνέχεια. Αντιγράφει τα έγκυρα ως προς τις αποδείξεις ψηφοδέλτια χωρίς τις αποδείξεις. Έστω (C_2, S_2) αυτό το σύνολο.
  2. Σε αυτό το στάδιο διαγράφονται τα διπλά ψηφοδέλτια ως εξής: για κάθε στοιχείο του πίνακα οι Talliers υπολογίζουν για κάθε κρυπτοκείμενο του S_2 το fingerprint του με το πρωτόκολλο που περιγράψαμε νωρίτερα. Υπολογίζουν δηλαδή με το κλειδί k τα $h_k(E_T(\sigma_i))$ και μέσω του hashtable ψάχνουν για collisions. Σε όσα βρεθούν προσμετρείται ένα με βάση κάποια πολιτική (πχ το τελευταίο που κατατέθηκε). Έστω C_3, S_3 το αντίστοιχο σύνολο. Τα hashkeys δημοσιοποιούνται για επαλήθευση. Με hashtables ο αναμενόμενος χρόνος για αυτό το βήμα είναι $\mathcal{O}(|S_3|)$. Το σύνολο (C_3, S_3) περνιέται από ένα mix network και το αποτέλεσμα είναι το C_4, S_4.
  3. Με το κλειδί j οι talliers υπολογίζουν και δημοσιοποιούν τα hashkeys του voter roll. Με το ίδιο κλειδί υπολογίζουν και δημοσιοποιούν τα hashkeys των στοιχείων του S_4. Σε αυτό το στάδιο κάθε στοιχείο του S_4 έχει διαφορετικό credential και συνπώς διαφορετικό fingerprint. Επίσης όλα τα στοιχεία του voter roll έχουν διαφορετικό fingerprint. Μια ψήφος λοιπόν είναι έγκυρη αν και μόνο αν υπάρχει σε

ένα collision. Όσες είναι έγκυρες καταγράφονται στο C_5, S_5. Εδώ ο (αναμενόμενος) χρόνος που χρειάζεται είναι $\mathcal{O}(|V|)$.

4. Τα στοιχεία του C_5 περνιούνται από ένα mix network και στη συνέχεια αποκρυπτογραφούνται.

Όπως και στο JCJ ένας υπό εκβιασμό ψηφοφόρος μπορεί να δώσει ένα οποιοδήποτε κλειδί στον εκβιαστή και να ισχυριστεί ότι είναι το δικό του. Στη συνέχεια μπορεί να ψηφίσει με το αληθινό.

3.4.2.1 (Αν)ασφάλεια του συστήματος

Το παραπάνω σύστημα δεν είναι ασφαλές ως προς επιθέσεις εξαναγκασμού όπως αναφέραμε στην εισαγωγή. Θα δείξουμε μια στρατηγική που μπορεί να ακολουθήσει ο coercer για να επαληθεύει αν ο ψηφοφόρος υπέκυψε ή όχι στον εκβιασμό. Η επίθεση αυτή οφείλεται στους Araujo, Foulle, Traor [1].

Αυτό που θα εκμεταλλευτεί ο coercer είναι οι ομομορφικές ιδιότητες του El Gamal (ή οποιουδήποτε άλλου ομομορφικού συστήματος χρησιμοποιούνταν στο πρωτόκολλο). Έστω ότι ο υπό εκβιασμό ψηφοφόρος ισχυρίζεται ότι το κλειδί του είναι το $\tilde{\sigma}$. Ο coercer κάνει το εξής. Ετοιμάζει δύο ψήφους με κλειδιά $\tilde{\sigma}$ και $\tilde{\sigma^2}$. Στο τρίτο στάδιο της καταμέτρησης οι Talliers δημοσιοποιούν τα fingerprints όλων των κλειδιών. Σε αυτή τη λίστα θα υπάρχουν τα στοιχεία $\tilde{\sigma}^j$ και $\tilde{\sigma}^{2j}$. Ο coercer ψάχνει όλη αυτή τη λίστα για να βρει δύο στοιχεία με σχέση τετραγώνου. Όταν τα βρει και δεδομένου ότι η πιθανότητα να υπάρχουν άλλα στοιχεία με αυτή τη σχέση είναι αμελητέα μαθαίνει την τιμή $\tilde{\sigma}^z$. Στη συνέχεια ανατρέχει στη λίστα με τα fingerprints των στοιχείων του voter roll και ψάχνει αν υπάρχει στοιχείο με fingerprint $\tilde{\sigma}^j$. Αν υπάρχει ο εκβιαζόμενος υπέκυψε και έδωσε το αληθινό κλειδί και αν δεν υπάρχει προσπάθησε να κοροϊδέψει τον coercer. Η πρόθεσή του λοιπόν αποκαλύπτεται στον coercer με την παραπάνω στρατηγική και το σύστημα είναι ανασφαλές.

Μπορεί το σύστημα να μην πετυχαίνει τη ζητούμενη ιδιότητα όμως η ιδέα είναι πολύ χρήσιμη για να γλιτώσουμε την πρώτη αδυναμία στην αποδοτικότητα του JCJ: την διαγραφή διπλών ψήφων. Αυτή η τεχνική έχει υιοθετηθεί σε πολλές παραλλαγές του JCJ από όταν προτάθηκε.

3.5 Credentials με μαθηματική δομή

Μια κατεύθυνση στην βιβλιογραφία για βελτίωση της αποδοτικότητας του JCJ είναι να γίνεται κατάλληλη επιλογή των credential.. Δηλαδή αντί τα credentials να είναι τυχαία στοιχεία να διατηρούν κάποιες μαθηματικές ιδιότητες που αφενός δεν θα είναι αρκετές για να απειλήσουν την ασφάλεια του συστήματος και αφετέρου θα αξιοποιούνται από τους καταμετρητές για να μην χρειάζεται ο τετραγωνικός αριθμός Plaintext Equivalence Tests. Σε αυτή την υποενότητα παρουσιάζουμε ένα παράδειγμα τέτοιου συστήματος ηλεκτρονικής ψηφοφορίας που δημοσιεύθηκε από τους Araujo, Foulle, Traor [1]. Σημειώνουμε ότι στην ίδια εργασία παρουσιάζεται

και η επίθεση στο πρωτόκολλο ψηφοφορίας των Weber et al που είδαμε στο προηγούμενο κεφάλαιο.

3.5.1 Η δομή των credentials στο AFT

Σε αντίθεση με το JCJ και πολλές παραλλαγές του, το πρωτόκολλο ηλεκτρονικής ψηφοφορίας AFT ορίζει τα credential με τρόπο που να ικανοποιούν κάποιες μαθηματικές ιδιότητες. Όπως αναφέρεται και στο [1] η δομή αυτή στηρίζεται στις group signatures των Camenisch, Lysyanskaya [8]. Τα credentials έχουν δύο μέρη. Συγκεκριμένα κάθε credential είναι της μορφής $\sigma = (r, (a, b, c))$ όπου το r είναι το μυστικό. Αν η ομάδα που χρησιμοποιείται στο σύστημα είναι η $\mathcal{G}$ τότε για τα (a, b, c) ισχύουν τα εξής

- $a \in_R \mathcal{G}$
- $b = a^y$
- $c = a^{x+rxy}$

όπου τα $x, y \in \mathbb{Z}_{|\mathcal{G}|}$.

Μια χρήσιμη ιδιότητα είναι ότι για κάθε l αν το $\sigma = (r, (a, b, c))$ είναι έγκυρο το ίδιο ισχύει και για το $\sigma' = (r, (a^l, b^l, c^l)$ αφού τότε έχουμε $b = a^{yl} = (a^l)^y$ και $c = a^{(x+rxy)l} = (a^l)^{x+rxy}$. Βάση μιας υπόθεσης, της $LRSW$ [8] ισχύει ότι ακόμα και αν ένας (PPT) αντίπαλος έχει ένα μαντείο που με είσοδο r δίνει ένα έγκυρο credential για το r δεν μπορεί να φτιάξει μόνος του ένα credential για κάποιο r που δεν ρωτήσει το μαντείο με μη αμελητέα πιθανότητα.

Επίσης παρατηρούμε το εξής. Αν κάποιος PPT αντίπαλος ξεχώριζε μια ένα έγκυρο μυστικό $(r, (a, b, c))$ από μια τυχαία τριάδα τότε θα μπορούσε να χρησιμοποιηθεί για να ξεχωρίσει μια τριάδα Diffie Hellman από μια τυχαία τριάδα με μη αμελητέα πιθανότητα, γεγονός που αντιβαίνει την υπόθεση DDH . Συγκεκριμένα έστω $\mathcal{A}$ που ξεχωρίζει με μη αμελητέα πιθανότητα ψεύτικα από αληθινά κλειδιά. Θα φτιάξουμε D που επιτίθεται στο DDH.

- Ο D παίρνει είσοδο (g, g^x, g^y, g^z) και θέλει να απαντήσει αν $z = xy$ ή $z \neq xy$.
- Διαλέγει ένα r και θέτει $a = g$, $b = g^x$ και $c = (g^z)^r g^x$
- Δίνει στον $\mathcal{A}$ το credential $(r, (a, b, c))$ και απαντάει ότι απαντήσει ο $\mathcal{A}$.

Έχουμε τις εξής περιπτώσεις.

1. $z = xy$
   Τότε $b = g^x = a^x$ και $c = g^{xyr} g^x = g^{x+xyr} = a^{x+xyr}$ δηλαδή το credential είναι έγκυρο
2. $z \neq xy$
   Τότε και $c = g^{zr} g^x = g^{x+zr} = a^{x+zr} = a^x (a^r)^z$ δηλαδή το credential είναι τυχαίο στοιχείο της ομάδας.

Μια εύκολη ανάλυση δείχνει ότι η πιθανότητα να απαντήσει σωστά ο D είναι ίση με την πιθανότητα να απαντήσει σωστά ο $\mathcal{A}$ το οποίο όμως γίνεται με αμελητέα πιθανότητα βάση

της υπόθεσης DDH. Συνεπώς ο υπό εκβιασμό ψηφοφόρος μπορεί να κατασκευάσει ψεύτικα credentials και να τα δώσει στον εκβιαστή, τα οποία δεν ξεχωρίζουν από τα κανονικά.

3.5.2 Το πρωτόκολλο AFT

- **Setup:** Παράγονται τα ζεύγη κλειδιών $(sk_{\mathcal{T}}, pk_{\mathcal{T}})$. Αυτά μαζί με τις άλλες δημόσιες παραμέτρους του συστήματος (ομάδα, γεννήτορα κλπ) δημοσιοποιούνται. Επίσης οι καταμετρητές $\mathcal{T}$ διαμοιράζονται δύο μυστικά x, y που καθορίζουν τη δομή των credentials
- **Registration:** Κάθε ψηφοφόρος με δικαίωμα ψήφου ζητάει και παίρνει ένα μυστικό $(r, (a, b, c))$ που έχει την δομή που περιγράφηκε παραπάνω. Αυτό υπολογίζεται κατανεμημένα από τα μέλη της επιτροπής $\mathcal{R}$.
- **Election Setup:** Η $\mathcal{R}$ δημοσιοποιεί τα ονόματα των υποψηφίων και τα στοιχεία του χώρου μηνυμάτων $\mathbf{C}$ που αντιστοιχούν σε αυτά όπως στο JCJ. Επίσης δημοσιοποιείται ένα id m για τις εκλογές.
- **Voting:** Ο ψηφοφόρος (αφού πιθανώς έχει αλλάξει το credential υψώνοντας στην l) με credential $\sigma = (r, (a, b, c))$ και ψήφο $c \in C \cup \{\phi\}$ υπολογίζει το ψηφοδέλτιο του $B = (E_{\mathcal{T}}(c), a, E_{\mathcal{T}}(a^r), E_{\mathcal{T}}(a^{x+rxy}), m^r, Pf)$. Pf είναι οι κατάλληλες αποδείξεις σωστής δομής του B. Το m^r χρησιμοποιείται για να διαγραφούν αργότερα οι διπλές ψήφοι στο πνεύμα των Smith, Weber et al.
- **Tally:**

  1. Κατά τα γνωστά, οι Talliers ελέγχουν τις αποδείξεις και αγνοούν τα ψηφοδέλτια με μη έγκυρες και μέσω lookup hashtables διαγράφουν τις διπλές ψήφους και κρατάνε μόνο μία με βάση μια πολιτική. Τα περιττά πλέον κομμάτια m^r, Pf σβήνονται και το δεύτερο στοιχείο του ψηφοδελτίου αλλάζει από a σε $E_{\mathcal{T}}(a)$.
  2. Το αποτέλεσμα περνιέται από ένα mix network και η έξοδος που είναι ένα διάνυσμα με στοιχεία της μορφής

  $$(t, u, v, w) = (E_{\mathcal{T}}(c), E_{\mathcal{T}}(S), E_{\mathcal{T}}(B), E_{\mathcal{T}}(C))$$

  δημοσιοποιείται.

  3. Για κάθε στοιχείο οι Talliers διαλέγουν κατανεμημένα ένα τυχαίο α και υπολογίζουν $(wu^{-x}v^{-xy})^{\alpha}$ και προσμετρούν το αποτέλεσμα αν και μόνο αν η ποσότητα αυτή αποκρυπτογραφηθεί σε 1.
  4. Τέλος οι Talliers περνάνε τα πρώτα συστατικά των έγκυρων ψηφοδελτίων από ένα mix network και τα αποκρυπτογραφούν δημοσιοποιώντας το αποτέλεσμα.

  Παρατηρούμε ότι αν το credential είναι τυχαίο τυχαία θα είναι και η τιμή του αποτελέσματος του τελευταίου βήματος. Αντίθετα σε έγκυρο credential λαμβάνοντας υπόψην τις ομομορφικές ιδιότητες του El Gamal η αποκρυπτογράφιση δίνει

$$D_{sk_{\mathcal{T}}}((wu^{-x}v^{-xy})^{\alpha}) = (a^{x+rxy}a^{-x}(a^r)^{-xy})^{\alpha} = (a^{x+rxy-x-rxy})^{\alpha} = 1$$

Σημειώνουμε ότι η ύψωση στην α γίνεται για να αποτρέψει πιθανή διαρροή πληροφοριών σε περίπτωση λάθους credential ακριβώς όπως και στα PETs.

Το σύστημα δεν συνοδεύεται από απόδειξη ασφάλειας. Κατά τη γνώση του συγγραφέα δεν έχει δημοσιοποιηθεί κάποια επίθεση στο σύστημα χωρίς φυσικά αυτό να σημαίνει κάτι για την ασφάλειά του. Ο χρόνος που χρειάζεται για την καταμέτρηση εύκολα βλέπουμε ότι είναι γραμμικός ως προς ψηφοφόρους και ψήφους. Το σημαντικότερο αυτής της εργασίας δεν είναι το ίδιο το πρωτόκολλο που παρουσιάζεται αλλά το πως θα μπορούσαμε να εκμεταλλευτούμε μια μαθηματική δομή και να ορίσουμε credentials με βάση αυτή για να πετύχουμε τη ζητούμενη ιδιότητα σε ένα σύστημα ηλεκτρονικής ψηφοφορίας. Το παρουσιαζόμενο πρωτόκολλο είναι απλά ένα τέτοιο παράδειγμα. Υπάρχει πληθώρα τέτοιων πρωτοκόλλων στη βιβλιογραφία όπως έχουμε ήδη αναφέρει.

3.6 Το πρωτόκολλο Selections

Μία ιδέα κάπως διαφορετική από τις προηγούμενες αποτελεί το πρωτόκολλο selections των Clark, Hengartner [11]. Η δομή του πρωτοκόλλου πατάει όπως και οι προηγούμενες που είδαμε στο JCJ. Η κεντρική ιδέα είναι κάθε ψήφος να συνοδεύεται από ένα anonimity set , υποσύνολο των ψηφοφόρων και με αυτόν τον τρόπο να κρύβεται ο πραγματικός καταθέτης της. Δεδομένου λοιπόν της ύπαρξης ανώνυμου καναλιού ο coercer δεν μπορεί να επαληθεύσει αν κάποιος υπό εκβιασμό ψηφοφόρος επιχείρησε να ψηφίσει γλιτώνοντας τις επιθέσεις forced absention, που είναι και η κύρια αιτία της μη αποδοτηκότητας του JCJ . Στη συνέχεια παρουσιάζουμε το βασικό κορμό του συστήματος αυτού αμελόντας κάποιες λεπτομέριες από την παρουσίαση μας.

3.6.1 Το πρωτόκολλο Selections

Παρακάτω παρουσιάζεται η δομή του πρωτοκόλλου. Οι συμμετέχοντες είναι όπως και στο JCJ.

- **Setup:** Παράγονται τα ζεύγη κλειδιών $(sk_{\mathcal{T}}, pk_{\mathcal{T}})$ και $(sk_{\mathcal{R}}, pk_{\mathcal{R}})$. Αυτά μαζί με τις άλλες δημόσιες παραμέτρους του συστήματος (ομάδα, γεννήτορα κλπ) δημοσιοποιούνται.
- **Registration:** Γίνεται όπως και στο JCJ . Οι ψηφοφόροι, αφού έπιβεβαιωθεί από την $\mathcal{R}$ ότι έχουν δικαίωμα ψήφου παίρνουν ένα μυστικό credential . Συγκεκριμένα επιλέγουν ένα στοιχείο $\sigma \in \mathbb{Z}_q$, όπου q το πλήθος των στοιχείων της ομάδας, και υπολογίζουν το g^σ (δηλαδή το μήνυμα κρυπτογραφείται στον εκθέτη). Υπολογίζει την κρυπτογράφησή του με το κλειδί της $\mathcal{T}$ και NIZKPoK γνώσης του plaintext και το κρυπτοκείμενο $(c_1, c_2) = (g^r, g^\sigma \cdot h)$ δημοσιοποιείται στο Voter Roll με υπογραφή της $\mathcal{R}$.
- **Election Setup:** Η $\mathcal{R}$ δημοσιοποιεί τα ονόματα των υποψηφίων και τα στοιχεία του χώρου μηνυμάτων $\mathbf{C}$ που αντιστοιχούν σε αυτά όπως στο JCJ. Επιπλέον, για να χαθεί κάθε σύνδεση με προηγούμενες εκλογές και να πετύχουμε η φάση registration να

γίνεται μόνο μία φορά, σε κάθε εκλογή αλλάζει ο γεννήτορας που χρησιμοποιήται στην κρυπτογράφηση. Πιο συγκεκριμένα, κάθε μέλος της $\mathcal{T}$ διαλέγει τυχαία ένα $b_i \in \mathbb{Z}_q$ και υπολογίζει $g_i = g^{b_i}$. Στη συνέχεια υψώνει κάθε κρυπτογραφημένο credential στην b_i δηλαδή το $(g^r, g^\sigma \cdot h)$ γίνεται

$$(g^{r \cdot b_i}, (g^\sigma \cdot h)^{b_i}) = ((g^{b_i})^r, ((g^{b_i})^\sigma \cdot h^{b_i})) = (g_i^r, g_i^\sigma \cdot h_i))$$

και αποδεικνύει με NIZKP ότι το $g, c_1, c_2, g^{b_i}, c_1^{b_i}, c_2^{b_i}$ είναι τριπλή Diffie Hellman πλειάδα. Αφού γίνει αυτό από όλα τα μέλη αν θέσουμε $b = \prod_{i=1}^{|\mathcal{T}|} b_i$ τότε τα κρυπτοκείμενα είναι πλέον κρυπτογραφημένα με το ίδιο κλειδί και γενήτορα $g_0 = g^b$ που δημοσιοποιήται και είναι αυτός που χρησιμοποιήται σε αυτό το στιγμιότυπο εκλογών.

- **Voting:** Μέσω ενός ανώνυμου καναλιού ο ψηφοφόρος καταθέτει την ψήφο του b_i ως εξής:

  1. Διαλέγει ένα υποσύνολο β των στοιχείων του Voter Roll . Θα αναφερθούμε αργότερα στο μέγεθος του β.
  2. Υπολογίζει $g_0^{\sigma_i}$ όπου σ_i το μυστικό του. Επιλέγει ένα στοιχείο του Voter Roll και του αλλάζει την τυχαιότητα. Έστω ότι επέλεξε c και το άλλαξε σε c'. Η ψήφος θα μετρήσει μόνο αν $Dec_{sk_T}(c') = g_0^{\sigma_i}$.
  3. Υπολογίζει μία NIZKPoK του διακριτού λογαρίθμου σ_i
  4. Κατασκευάζει NIZKP ότι c' είναι επανακρυπτογράφηση ενός από τα στοιχεία του β. Το β είναι το anonimity set.
  5. Υπολογίζει το ψηφοδέλτιο προς καταμέτρηση $\mathbf{B}$ που το selections για συμβατότητα με διάφορα πρωτόκολλα το αφήνει απροσδιόριστο. Θα πρέπει το $\mathbf{B}$ να μπορεί να εισαχθεί σε mix network και να περιέχει απόδειξη ορθότητας, δηλαδή να αποτρέπεται να μετρηθούν μη έκγυρες ψήφοι για να αποφευχθεί η randomization attack που περιγράφεται στο JCJ (ένα τυχαίο στοιχείο είναι απόδειξη ότι ακυρώθηκε μία ψήφος).
  6. Γράφει στον $\mathcal{BB}$ $(g_0^{\sigma_i}, c', \beta, \mathbf{B}, Proofs)$.

  Οι διπλές ψήφοι αναγνωρίζονται άμεσα από το $g_0^{\sigma_i}$. Σημειώνουμε ότι για να μην μπορεί κάποιος τρίτος να αντιγράψει τις αποδείξεις μιας ψήφου με σκοπό να την ακυρώσει πρέπει η τυχαιότητα στην απόδειξη γνώσης του διακριτού λογαρίθμου να πηγάζει από το ψηφοδέλτιο που κατατίθεται. Σε περίπτωση coercion αρκεί ο ψηφοφόρος να δώσει ένα ψεύτικο credential. Στη συνέχεια ψηφίζει με το κανονικό καταθέτοντας stealth vote δηλαδή χρησιμοποιόντας ως β όλο το Voter Roll.

- **Tally:** Η επιτροπή καταμέτρησης $\mathcal{T}$ παίρνει τις ψήφους με σωστές αποδείξεις, κρατάει τις τελευταίες που κατατέθικαν και διαπιστώνει την εγκυρότητα κάθε ψήφου κάνοντας β $PETs$. Στη συνέχεια περνάει τα έγκυρα $\mathbf{B}$ σε mix network και τα αποκρυπτογραφεί ανάλογα με το εκάστοτε πρωτόκολλο.

3.6.1.1 Ασφάλεια και αποδοτικότητα

Όπως έχουμε δει το πρόβλημα του JCJ είναι η αποδοτικότητά του και συγκεκριμένα το τετραγωνικό ως προς ψηφοφόρους και ψήφους κόστος της καταμέτρησης της ψηφοφορίας. Στην περίπτωση που το μέγεθος του anonimity set β είναι $|\beta| = |Voters|$ τότε δεν κερδίζουμε κάτι σε απόδοση συγκριτικά με το JCJ . Για την ακρίβεια κάθε ψηφοφόρος θα πρέπει να προσωμοιώσει $|Voters| - 1$ μη διαδραστικές αποδείξεις μηδενικές αποδείξεις γνώσης τυχαιότητας για επανακρυπτογράφηση του credential που καταθέτει για την απόδειξη ότι είναι επανακρυπτογράφηση ενός από τα credential του voter roll, κάτι πολύ μη αποδοτικό, και οι καταμετρητές θα πρέπει για κάθε ψηφοδέλτιο να εκτελέσουν $|Voters|$ PETs με συνέπεια συνολικά να χρειάζονται $\mathcal{O}(|Voters| \cdot |Votes|)$, δηλαδή ασυμπτωτικά ίδιο με το χρόνο του JCJ.

Στην περίπτωση που το μέγεθος του anonimity set είναι $|\beta| = k$ για κάποιο k σταθερό οι αντίστοιχοι χρόνοι για την ψηφοφορία και την καταμέτρηση γίνονται αντίστοιχα $\mathcal{O}(1)$ και $\mathcal{O}(|Votes|)$.

Στην πρώτη περίπτωση, δηλαδή για $|\beta| = |Voters|$ οι δημιουργοί του πρωτοκόλλου αποδυκνύουν ότι το Selections είναι ασφαλές ως προς το coercion resistance μέσα στο πλαίσιο που ορίζεται από το JCJ. Στην δεύτερη περίπτωση, για σταθερό $|\beta|$ η ιδιότητα αυτή δεν παραμένει. Αυτό που προτείνεται στο πρωτόκολλο λοιπόν είναι σταθερό μέγεθος anonimity set με δυνατότητα κατάθεσης αόρατης ψήφου (stealth vote) δηλαδή ψήφο που χρησιμοποιεί ως anonimity set όλο το voter roll. Στην περίπτωση αυτή το Selections συνεχίζει να είναι coercion resistant και η απόδοσή του πρακτικά είναι ίδια με όταν $|\beta| = k$.

Τέλος αναφέρουμε μια ιδέα για βελτίωση της απόδοσης που προτάθηκε σε ένα άλλο πρωτόκολλο [3]. Επειδή η προσωμοίωση αποδείξεων μηδενικής γνώσης είναι αρκετά απαιτιτική υπολογιστικά προτείνανε κάθε ψήφος που κατατίθεται να σπάει σε $|\beta|$ διαφορετικές ψήφους μία για κάθε επιλογή credential από το υποσύνολο του voter roll που καθορίζει το β. Έτσι φεύγει ένα μεγάλο υπολογιστικό κόστος από τους ψηφοφόρους και επιβαρύνονται περισσότερο οι αρχές, οι οποίες όμως μπορούν να κάνουν τους υπολογισμούς κατά τη διάρκεια της ψηφοφορίας, δηλαδή κάθε ballot που εμφανίζεται να αρχίσουν οι Talliers να το επεξεργάζονται (φυσικά χωρίς να διαρέεται πληροφορία η οποία θα απειλούσε το fairness) ενόσω οι ψηφοφόροι συνεχίζουν να καταθέτουν ψηφοδέλτια.

Κεφάλαιο 4

Τυφλές Υπογραφές υπό Συνθήκη και Εφαρμογή τους σε Απομακρυσμένες Ψηφοφορίες

Στην ενότητα αυτή θα εισαγάγουμε ένα νέο κρυπτογραφικό primitive που ονομάζουμε Conditional Blind Signatures και στη συνέχεια το χρησιμοποιούμε για να κατασκευάσουμε ένα πρωτόκολλο ηλεκτρονικής ψηφοφορίας που είναι ανθεκτικό ως προς το coercion resistance και ενισχύει το Privacy εκμεταλλευόμενο τις ιδιότητες των τυφλών υπογραφών.

Συνοπτικά το primitive αυτό επιτρέπει στον υποφγράφων να δώσει μία έγκυρη ή άκυρη τυφλή υπογραφή, κάτι που καθορίζεται από την μυστική είσοδο του Signer. Ο Recepiant δεν μπορεί να επαληθεύσει την εγκυρότητα της υπογραφής και μόνο ο Signer που κατέχει ένα μυστικό μπορεί. Για να εφαρμοστεί σε voting ο αιτών πάει με είσοδο που εξαρτάται από το credential του να ζητήσει μία υπογραφή σε μία ψήφο. Παίρνει απάντηση που δεν ξέρει αν είναι σωστή (αν έκανε coercion συνεπώς δεν μπορεί να ξεχωρίσει αν πήρε αληθινό ή όχι credential) και ανώνυμα καταθέτει την ψήφο του με την υπογραφή.

Στην υποενότητα 4.1 παρουσιάζουμε το νέο αυτό primitive και τις ιδιότητές του. Ξεκινάμε ορίζοντας τι είναι ένα σχήμα Conditional Blind Signatures, στη συνέχεια κατασκευάζουμε ένα τέτοιο σχήμα που στηρίζεται στις υπογραφές Okamoto-Schnoor και αποδεικνύουμε την ασφάλειά του και τέλος το βελτιώνουμε ως προς τον απαιτούμενο αριθμό γύρων. Στην ενότητα 4.2 παρουσιάζουμε το νέο πρωτόκολλο ηλεκτρονικής ψηφοφορίας που βασίζεται σε αυτό και εξετάζουμε την ασφάλειά του. Τέλος παραθέτουμε ένα κατανεμημένο πρωτόκολλο για την έκδοση των υπογραφών στο προτεινόμενο πρωτόκολλο.

4.1 Τυφλές Υπογραφές Υπό Συνθήκη

Στην ενότητα αυτή ορίζουμε το νέο primitive. Οι ιδιότητες που θέλουμε να έχει είναι

- Να δίνονται τυφλές υπογραφές με στατιστικό blindness.
- Ο υπογράφων, με επιπλέον μυστική είσοδο εκτός από το κλειδί του ένα bit να δίνει έγκυρη ή άκυρη υπογραφή με τέτοιο τρόπο ώστε κανένας PPT αντίπαλος να μην μπορεί να πάρει επιπλέον πληροφορία για το bit εκτός με αμελητέα πιθανότητα ως προς την παράμετρο ασφάλειας.

Δίνουμε στη συνέχεια τον ορισμό για τις τυφλές υπογραφές υπό συνθήκη.

Ορισμός 4.24. *Ένα σχήμα τυφλών υπογραφών υπό συνθήκη (Conditional Blind Signatures) είναι μία τριάδα (Gen, Sign, Vrfy) για την οποία ισχύουν*

- *Ο Gen με είσοδο την παράμετρο ασφαλείας 1^k δίνει ένα ζεύγος κλειδιών (sk, vk). Το vk είναι το δημόσιο κλειδί και το sk είναι το κλειδί υπογραφής και επαλήθευσης. Επίσης παράγονται χώροι μηνυμάτων $\mathcal{M}, \mathcal{S}$ για τα μηνύματα και τις υπογραφές που εξαρτώνται από την παράμετρο ασφαλείας 1^k. Συμβολίζουμε το σύνολο των απαιτούμενων παραμέτρων $params$.*
- *Το Sign είναι ένα πρωτόκολλο μεταξύ του υπογράφωντος S και του χρήστη U δηλαδή $Sign = \langle S, U\rangle$ με κοινή είσοδο $params, vk$, μυστική είσοδο για τον S $sk, b \in \{0,1\}$ και μυστική είσοδο για τον U ένα μήνυμα m. Η έξοδος του πρωτοκόλλου είναι (ϵ, sig) όπου ϵ η κενή συμβολοσειρά.*
- *Ο Vrfy είναι αλγόριθμός που με είσοδο sk, m, sig δίνει 1 αν το sig είναι έγκυρη υπογραφή για το m*

τέτοια ώστε να ισχύει η ορθότητα, δηλαδή αν το sig είναι η έξοδος του πρωτοκόλλου για τον χρήστη με μυστική είσοδο του S $b = 1$ τότε και μόνο τότε $Vrfy(sk, m, sig) = 1$ εκτός ίσως με αμελητέα πιθανότητα.

Οι Conditional Blind Signatures κληρονομούν από τις Blind Signatures τις ιδιότητες Blindness και One More Forgery. Πρέπει βέβαια να κάνουμε μικρές αλλαγές για να λάβουμε υπ όψην το μυστικό input b. Συγκεκριμένα, στο Blindness Game, δεδομένου ότι το b τετριμένα ξεχωρίζει δύο μηνύματα, απαιτούμε στα δύο μηνύματα να χρησιμοποιείται το ίδιο b. Για το Ονε Μορε Φοργερψ παρατηρούμε ότι άκυρες υπογραφές πιθανώς να ευνοούν τον αντίπαλο. Συνεπώς δίνουμε τη δυνατότητα στον αντίπαλο να ζητάει υπογραφές διαλέγοντας αυτός το μυστικό bit b. Τέλος, οι τυφλές υπογραφές υπό συνθήκη έχουν μία επιπλέον ιδιότητα την Conditional Verifiability που ορίζεται από παίγνιο Αλγόριθμος ;;.

Ορισμός 4.25. *Ένα σχήμα Conditional Blind Signatures Π είναι Conditioanlly Verifiable αν για κάθε PPT $\mathcal{A}$ υπάρχει αμελητέα συνάρτηση μ ως προς την παράμετρο ασφαλείας k τέτοιο ώστε* $\Pr[\boldsymbol{CondVerExp}_{\mathcal{A},\Pi} = 1] \leq \frac{1}{2} + \mu(k)$.

Ορισμός 4.26. *Ένα σχήμα Conditional Blind Signatures Π είναι ασφαλές αν έχει τις τρεις ιδιότητες Statistical Blindness, One More Forgery και Conditional Verifiability.*

Μία σημαντική παρατήρηση είναι ότι επειδή ο χρήστης δεν μπορεί να επαληθεύσει την υπογραφή

Algorithm: $\text{CondVerExp}_{\mathcal{A},\Pi}$

Input : security parameter λ

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

$c \leftarrow_R \{0,1\}$
$(sk, pk, params) \leftarrow \text{Gen}(1^\lambda)$
$\{(m_i, sig_i) \leftarrow \text{Sign}\langle S(params, sk_{\mathcal{S}}, b_i), \mathcal{A}(params, pk, \{m_j, sig_j\}_{j=1}^{i-1}, b_i)\rangle\}_{i=1}^{l_1}$
$m_c \leftarrow \mathcal{A}(pk, params, \{(m_i, sig_i)\}_{i=1}^{l_1}, \textbf{Challenge})$
$(\perp, sig_c) \leftarrow \text{Sign}\langle S(params, sk, b), \mathcal{A}(params, pk, m_c)\rangle$
$\{(m_i, sig_i) \leftarrow \text{Sign}\langle S(params, sk_{\mathcal{S}}, b_i), \mathcal{A}(params, pk, \{m_j, sig_j\}_{j=1}^{i-1}, b_i)\rangle\}_{i=l_1+1}^{l_2}$
$c' \leftarrow \mathcal{A}(\{m_i, sig_i\}_{i=1}^{l_1+l_2}, m_c, sig_c, \textbf{Guess})$
if $c = c'$ **then**
  return 1
else
  return 0
end

που λαμβάνει και να γνωρίζει το μυστικό bit b δεν μπορεί να ξέρει αν ο υπογράφων ακολουθεί ή όχι το πρωτόκολλο. Για την εφαρμογή που το χρειαζόμαστε, τις incoercible ηλεκτρονικές ψηφοφορίες, θα θεωρούμε ότι ο υπογράφων είναι πολλές οντότητες n εκ των οποίων το πολύ t αποκλείνουν από την επιβαλλόμενη συμπεριφορά σύμφωνα με το πρωτόκολλο.

4.1.1 Okamoto Schnorr Conditional Blind Signatures

Το νέο σχήμα Conditional Blind Signatures που προτείνουμε στηρίζεται στις τυφλές υπογραφές Okamoto-Schnorr. Η κεντρική ιδέα είναι αντί να δίνουμε την υπογραφή R, S να μασκάρουμε το ένα στοιχείο R ως k^R για κάποιο k για το οποίο μόνο ο Signer ξέρει τον λογάριθμό του v ως προς g. Για την επαλήθευση χρειάζεται το v. Παρακάτω παρουσιάζεται το νέο σχήμα.

Algorithm: Gen

Input : security parameter n

Output: $sk, pk, params$

/* Επιλέγεται ομάδα τάξης q πρώτου με $q > n$ όπου το DDH είναι δύσκολο */

1 $(q, \mathbb{G}) \leftarrow GenDDHGroup(1^k)$

/* Επιλέγονται γεννήτορες */

2 $(g, h) \leftarrow_R \mathbb{G}$

/* Επιλέγονται κλειδιά για υπογραφές και επαλήθευση */

3 $r, s, v \leftarrow_R \mathbb{Z}_q$

4 $y \leftarrow g^{-r} h^{-s}$

5 $k \leftarrow g^v$

6 $(sk_S, vk_S) \leftarrow ((r, s), y)$

7 $(sk_V, vk_V) \leftarrow (v, k)$

8 $sk \leftarrow (sk_S, sk_V)$

9 $pk \leftarrow (pk_S, vk_V)$

10 $params \leftarrow (q, \mathbb{G}, g, h)$

Στη συνέχεια παρουσιάζουμε το πρωτόκολλο τριών κινήσεων για την υπογραφή μεταξύ Signer και Recepiant. Θεωρούμε μία συνάρτηση $H : \{0,1\}^* \rightarrow \mathbb{Z}_q$ την οποία μοντελοποιούμε ως τυχαίο μαντείο.

Sign

Signer		**Recepiant**
input:$(params, sk, pk, b)$		input:$(params, pk, m)$
$t, u \leftarrow_R \mathbb{Z}$		
$a = g^t h^u$		
	$a \longrightarrow$	
		$\beta, \gamma, \delta \leftarrow_R \mathbb{Z}_q$
		$\alpha = ag^\beta h^\gamma y^\delta$
		$\epsilon = H(m, \alpha)$
		$e = \epsilon - \delta$
	$\longleftarrow e$	
$R = t + er$		
$S = u + es$		
if $b = 1$ **then** $bsig_1 = k^R$		
else $bsig_1 \leftarrow_R \mathbb{G}$		
	$bsig_1, bsig_2 = S \longrightarrow$	
		$sig_1 = bsig_1 \cdot k^\beta$
		$sig_2 = \sigma = bsig_2 + \gamma$
		output $(m, \alpha, sig_1, sig_2)$

Τέλος δίνουμε τον αλγόριθμο επαλήθευσης.

Algorithm: Vrfy

Input : $sk, pk, params, H, m, sig = (m, \alpha, sig_1, sig_2)$

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

if $m \neq m'$ **then**
  return 0
end
$\epsilon \leftarrow H(m, a)$
$\rho' = sig_1$
$\sigma' = sig_2$
if $\alpha^v = \rho' h^{\sigma' \cdot v} y^{\epsilon \cdot v}$ **then**
  return 1
else
  return 0
end

Έυκολα μπορούμε να επαληθεύσουμε την ορθότητα. Συγκεκριμένα για $b = 1$ έχουμε

$$\begin{aligned}
\alpha^v = \rho' h^{\sigma' \cdot v} y^{\epsilon \cdot v} &\Leftrightarrow (a g^{\beta} h^{\gamma} y^{\delta})^v = k^{R+\beta} h^{(S+\gamma)\cdot v} y^{(e+\delta)\cdot v} \\
&\Leftrightarrow a^v k^{\beta} h^{v\cdot\gamma} y^{v\cdot\delta} = k^{R+\beta} h^{(S+\gamma)\cdot v} y^{(e+\delta)\cdot v} \\
&\Leftrightarrow a^v = k^R h^{S\cdot v} y^{e\cdot v} \\
&\Leftrightarrow a = g^R h^S y^e \\
&\Leftrightarrow g^t h^u = g^{t+er} h^{u+es} (g^{-r} h^{-s})^e \\
&\Leftrightarrow g^t h^u = g^t + g^{er} h^u h^{es} g^{-er} h^{-es} \\
&\Leftrightarrow g^t h^u = g^t h^u
\end{aligned}$$

που ισχύει.

Για το blindness παρατηρούμε ότι δεν αλλάζει κάτι ως προς το απλό Okamoto-Schnorr. Συγκεκριμένα τόσο το μήνυμα όσο και η τελική υπογραφή μασκάρονται πληροφοριοθεωριτικά από τα β, γ, δ.

Δείχνουμε στη συνέχεια ότι το σύστημα είναι είναι ασφαλές ως προς One-More-Forgery. Σημειώνουμε ότι ένας αντίπαλος μπορεί να παράγει ψεύτικες υπογραφές μόνος του ισόνομες με αυτές που θα έπαιρνε από τον Signer επιλέγοντας τυχαίο S και ένα τυχαίο ciphertext οπότε δεν τον βοηθάει να παίρνει άκυρες υπογραφές από τον Signer.

Θεώρημα 4.5. *Μοντελοποιούμε την H ως Random Oracle. Έστω ένας PPT αλγόριθμος $\mathcal{A}$ που έχει ως είσοδο μόνο το δημόσιο κλειδί ο οποίος μπορεί να ζητάει (ακόμα και παράλληλα) l υπογραφές σε μηνύματα της επιλογής του και κάνει Q ερωτήσεις στο Random Oracle. Αν το l είναι πολυλογαρθμικό ως προς την παράμετρο ασφαλείας, Q πολυωνιμικό ως προς την παράμετρο ασφαλείας και με πιθανότητα μη αμελητέα παράγει $l + 1$ έγκυρες υπογραφές σε διαφορετικά μηνύματα τότε υπάρχει αλγόριθμος που σε πολυωνιμικό χρόνο και με μη αμελητέα πιθανότητα λύνει το CDH.*

Απόδειξη. Έστω τέτοιος $\mathcal{A}$. Δείχνουμε το εξής: Αν πάρουμε δύο υπογραφές στο ίδιο μήνυμα με ίδιο αρχικό commitment $(m, \alpha, k^{\rho_1}, \sigma_1), (m, \alpha, k^{\rho_2}, \sigma_2)$ με $\sigma_1 - s\epsilon_1 \neq \sigma_2 - s\epsilon_2$ τότε μπορούμε να υπολογίσουμε με είσοδο g, g^a, g^v το g^{av}. Για να πάρουμε τις δύο αυτές υπογραφές εφαρμόζουμε ένα Replay Attack δηλαδή εκτελούμε τον αλγόριθμο με ένα Random Oracle H_1 και στη συνέχεια επαναλαμβάνουμε με ίδια είσοδο και Random Oracle H_2 τέτοιο ώστε να δίνει ίδιες απαντήσεις στις πρώτες $i - 1$ ερωτήσεις. Ελπίζουμε να πάρουμε τις δύο υπογραφές από την i-οστή ερώτηση. Δείχνουμε παρακάτω την αναγωγή.

- Έχουμε είσοδο g, g^a, g^v και θέλουμε να υπολογίσουμε g^{av}
- Θέτουμε $g = g, h = g^a, k = g^v$. Διαλέγουμε r, s και υπολογίζουμε δημόσιο κλειδί y.
- Με αυτά μπορούμε να υπογράφουμε.
- Δίνουμε $(params, y, k)$ σαν είσοδο στον $\mathcal{A}$.
- Εκτελούμε το παίγνιο για το Forgery με τον αντίπαλο $\mathcal{A}$ και H_1.

- Επαναλάβουμε την επίθεση με random oracle H_2.
- Αν πάρουμε τις ζητούμενες υπογραφές έχουμε

$$k^{\rho_1} = \alpha^v h^{-\sigma_1 v} y^{-\epsilon_1 v}$$

$$k^{\rho_2} = \alpha^v h^{-\sigma_2 v} y^{-\epsilon_2 v}$$

  Δηλαδή θα ισχύει ότι

$$k^{\rho_1} k^{-\rho_2} = h^{(-\sigma_1+\sigma_2)v} y^{(-\epsilon_1+\epsilon_2)v}$$

- Έχουμε γνωστά τα
  - $t = k^{\rho_1} k^{-\rho_2}$
  - $\sigma = (-\sigma_1 + \sigma_2)$
  - $\epsilon = (-\epsilon_1 + \epsilon_2)$
  - τα r, s του κλειδιού

Με αυτά υπολογίζουμε g^{av} ως εξής:

$$\begin{aligned} t = h^{\sigma v} y^{cv} &\Rightarrow t = g^{a\sigma v}(g^{-r} h^{-s})^{\epsilon v} \\ &\Rightarrow t = g^{a\sigma v}(g^{-r} g^{-sa})^{\epsilon v} \\ &\Rightarrow t = g^{a\sigma v} g^{-r\epsilon v} g^{-sa\epsilon v} \\ &\Rightarrow t g^{r\epsilon v} = g^{av(\sigma - s\epsilon)} \\ &\Rightarrow g^{av} = (t \cdot (g^v)^{r\epsilon})^{(\sigma - s\epsilon)^{-1}} \end{aligned}$$

Χρησιμοποιώντας τις ίδιες τεχνικές όπως παρουσιάζονται στο [40] αποδεικνύεται ότι η πιθανότητα να πετύχει η επίθεση είναι μη αμελητέα. ■

Τέλος δείχνουμε ότι το σύστημα είναι Conditionally Verifiable αν το DDH είναι δύσκολο.

Τονίζουμε ότι μία υπογραφή είναι έγκυρη αν και μόνο αν

$$k^R = a^v h^{-Sv} y^{-ev} \tag{4.1}$$

Θεώρημα 4.6. *Έστω ότι υπάρχει PPT αλγόριθος $\mathcal{A}$ που νικάει το CondVerExp με μη αμελητέα πιθανότητα. Τότε υπάρχει ένας PPT αλγόριθμος $\mathcal{B}$ που με μη αμελητέα πιθανότητα ξεχωρίζει μία τυχαία τριάδα από μία τριάδα Diffie Hellman.*

Απόδειξη. Χρησιμοποιώντας τον $\mathcal{A}$ θα κατασκευάσουμε τον $\mathcal{B}$.

- Ο $\mathcal{B}$ παίρνει ως είσοδο g, g^a, g^v, g^c. Προσπαθεί να απαντήσει αν $c = av$ ή αν το c είναι ομοιόμορφα κατανεμημένα στο $\mathbb{Z}_q$.

- Ο $\mathcal{B}$ θέτει $g = g$, $h = g^a$ και $k = g^v$. Διαλέγει τυχαία $r, s \leftarrow_R \mathbb{Z}_q$ και θέτει $y = g^{-r}h^{-s}$. Δίνει στον $\mathcal{A}$ τα g, h, k, y.
- Μέσω του μυστικού του κλειδιού (r, s) μπορεί να απαντάει ορθά στα αιτήματα του $\mathcal{A}$ για έγκυρες υπογραφές.
- Όταν ο $\mathcal{B}$ ζητήσει να ξεκινήσει το Challenge, ο $\mathcal{B}$ απαντάει ως εξής:
  - Διαλέγει τυχαία $t, u \in \mathbb{Z}_q$ και θέτει $a \leftarrow g^t h^u$. Στέλνει το a στον $\mathcal{A}$.
  - Ο $\mathcal{A}$ του απαντάει με ένα $e \in \mathbb{Z}_q$.
  - Ο $\mathcal{B}$ διαλέγει τυχαίο $S \in \mathbb{Z}_q$ και θέτει $k^R = g^{vt}g^{cu}g^{-cS}g^{vre}g^{cse}$.
  - Στέλνει στον $\mathcal{A}$ την υπογραφή k^R, S.
- Όπως πριν ο $\mathcal{B}$ με το μυστικό του κλειδί απαντάει τα αιτήματα του $\mathcal{A}$ για υπογραφές.
- Ο $\mathcal{B}$ δίνει έξοδο 1 (η είσοδος είναι τετράδα Diffie Hellman) αν και μόνο αν ο $\mathcal{A}$ δώσει έξοδο 1 (η υπογραφή είναι έγκυρη).

Από την 4.1 ξέρουμε ότι η υπογραφή είναι έγκυρη αν και μόνο αν $k^R = a^v h^{-Sv} y^{-ev}$. Έχουμε λοιπόν ότι

$$\begin{aligned} k^R = a^v h^{-Sv} y^{-ev} &\Leftrightarrow g^{vt}g^{cu}g^{-cS}g^{vre}g^{cse} = a^v h^{-Sv} y^{-ev} \Leftrightarrow \\ &\Leftrightarrow g^{vt}g^{cu}g^{-cS}g^{vre}g^{cse} = g^{tv}h^{uv}h^{-Sv}g^{rev}h^{sev} \Leftrightarrow \\ &\Leftrightarrow g^{cu}g^{-cS}g^{cse} = h^{uv}h^{-Sv}h^{sev} \Leftrightarrow \\ &\Leftrightarrow g^{c(u-S+se)} = g^{av(u-S+se)} \end{aligned}$$

Αυτό σημαίνει ότι αν $u - S + se \neq 0$ τότε η υπογραφή είναι έγκυρη αν και μόνο αν $g^c = g^{av}$ δηλαδή αν και μόνο αν η είσοδος είναι τετράδα Diffie Hellman. Αφού το S διαλέγεται τυχαία και ομοιόμορφα, η πιθανότητα να ισχύει $u - S + se = 0$ είναι αμελητέα.

■

Τελικά το σύστημα είναι ασφαλές με την σημαντική λεπτομέρια ότι η ασφάλεια για το One More Forgery ισχύει ανν οι έγκυρες υπογραφές που παίρνει ο αντίπαλος είναι πολυλογαριθμικές ως προς την παράμετρο ασφάλειας, κάτι που φυσικά δεν είναι επιθυμητό. Αφήνουμε ως μελλοντική δουλειά να βρούμε αντίστοιχο σχήμα ανθεκτικό σε επιθέσεις με πολυωνιμικό αριθμό υπογραφών.

4.1.2 Round Optimal Conditional Blind Signatures

Θα δείξουμε σε αυτήν την υποενότητα πως να μειώσουμε την αλληλεπίδραση για το πρωτόκολλο sign. Ο τρόπος είναι απλός. Ο υπογράφων μπορεί να υπογράφει μόνο με το κλειδί επαλήθευσης. Αν λοιπόν έχουμε προκαθορισμένη τυχαιότητα για το a τότε μπορούν και οι δύο παίχτες να το υπολογίσουν χωρίς να αλληλεπιδράσουν. Παραλείπουμε σε αυτήν την υποενότητα τη μελέτη για το πως προκύπτει η τυχαιότητα αυτή, κάτι που θα μας απασχολήσει στην

επόμενη ενότητα που εφαρμόζουμε το νέο σχήμα σε ένα πρωτόκολλο ηλεκτρονικής ψηφοφορίας. Δείχνουμε μόνο πως μπορεί να υπολογιστεί η υπογραφή μόνο με το κλειδί επαλήθευσης. Θεωρούμε το a λοιπόν ως κοινό input.

Sign

Signer		**Recepiant**
input:$(params, sk, pk, b, a)$		input:$(params, pk, m, a)$
		$\beta, \gamma, \delta \leftarrow_R \mathbb{Z}_q$
		$\alpha = ag^{\beta}h^{\gamma}y^{\delta}$
		$\epsilon = H(m, \alpha)$
		$e = \epsilon - \delta$
	$\xleftarrow{e}$	
$S \leftarrow_R \mathbb{Z}_q$		
$bsig_2 = S$		
if $b = 1$ **then**		
$bsig_1 = (ah^{-S}y^{e\cdot})^v$		
else $bsig_1 \leftarrow_R \mathbb{G}^2$		
	$\xrightarrow{bsig_1, bsig_2 = S}$	
		$sig_1 = bsig_1 \cdot k^{\beta}$
		$sig_2 = \sigma = bsig_2 + \gamma$
		output $(m, \alpha, sig_1, sig_2)$

Εύκολα βλέπουμε ότι επαληθεύεται η εξίσωση επαλήθευσης. Επίσης η όψη του U είναι ίδια με αυτή στο προηγούμενο πρωτόκολλο αν θεωρήσουμε το a τυχαίο. Συνεπώς η ασφάλεια του νέου συστήματος είναι ίδια με την ασφάλεια του αρχικού.

4.2 Ένα Everlasting Private Coercion Restintant Πρωτόκολλο για Ηλεκτρονικές Ψηφοφορίες

Σε αυτήν την ενότητα προτείνουμε μια παραλλαγή του JCJ και δείχνουμε ότι είναι ασφαλές ως προς το coercion resistance (τροποποιώντας ελαφρώς το πλαίσιο που τέθηκε από τους JCJ) και ότι δεν δίνει καμία πληροφορία για την ψήφο που κατατέθηκε από τους ψηφοφόρους. Για να πετύχουμε τη δεύτερη ιδιότητα θεωρούμε πρόσβαση σε ένα τέλεια ανώνυμο κανάλι, θεωρούμε δηλαδή ότι κανένας αντίπαλος δεν μπορεί να συσχετίσει ένα μήνυμα που κατατέθηκε με έναν συγκεκριμένο αποστολέα. Αν και φαίνεται πολύ ισχυρή ιδιότητα μπορούμε να την δικαιολογήσουμε ως εξής: Θεωρούμε ότι η επικοινωνία γίνεται από οπουδήποτε στο διαδίκτυο

(συμπεριλαμβανομένων δημόσιων σημείων) και ότι κανένας δεν μπορεί να ελέγχει συνεχώς κατά τη διάρκεια των εκλογών όλα αυτά τα σημεία.

Η γενική ιδέα του προτεινόμενου συστήματος είναι η εξής: θα 'ενώσουμε' τα συστήματα FOO και JCJ σε ένα χρησιμοποιώντας το νέο primitive που παρουσιάσαμε. Συνοπτικά, ο κάθε ψηφοφόρος θα καταθέτει στην αρχή μία κρυπτογράφηση του credential του μαζί με το ID του και θα ζητάει μία τυφλή υπογραφή σε μία ψήφο της επιλογής του. Αν το credential είναι σωστό η αρχή θα τον προμηθεύει με έγκυρη υπογραφή αλλιώς με άκυρη. Ο ίδιος ο ψηφοφόρος δεν μπορεί να είναι σε θέση να ξεχωρίσει μία έγκυρη από μία άκυρη υπογραφή. Καταθέτει λοιπόν την ψήφο του υπογεγραμμένη μέσα από το ανώνυμο κανάλι, χωρίς να δίνει καμία πληροφορία για την ταυτότητά του, την ψήφο του μαζί με την υπογραφή. Μόλις κατατεθούν όλες οι ψήφοι η αρχή, αφού τις ανακατέψει, θα μετράει μόνο τις ψήφους για τις οποίες εξέδωσε έγκυρη υπογραφή.

Επειδή οι ψήφοι που κατατίθενται δεν έχουν πληροφορία για την ταυτότητα του χρήστη θα πρέπει η αρχή να φροντίσει πριν δώσει τις υπογραφές να έχει επιληφθεί για το θέμα των διπλών ψήφων. Επίσης ο ψηφοφόρος θα πρέπει να λέει στην αρχή ποιο είναι το ID του για να ξέρει η αρχή ως προς ποιο στοιχείο θα συγκρίνει το credential που πήρε. Για να μην χάνουμε την ιδιότητα του coercion resistance θεωρούμε ότι αυτή η πληροφορία δεν δίνει πληροφορία στον αντίπαλο. Συγκεκριμένα θεωρούμε ότι οι τίμιοι παίχτες θα ρίχνουν και άκυρες ψήφους με άλλα ID και συνεπώς ο αντίπαλος δεν μπορεί να ξέρει (λόγω του ανώνυμου καναλιού) αν ένα αίτημα για υπογραφή ήρθε από τον παίχτη που εκβιάζει ή όχι. Στην πράξη θεωρούμε ότι θα υπάρχουν φιλοδημοκρατικές αρχές και ψηφοφόροι που θα κάνουν ακριβώς αυτό. Το γεγονός αυτό το εκφράζουμε στην κατανομή αβεβαιότητας του αντιπάλου. Φυσικά την ιδιότητα αυτή την μεταφέρουμε και στο ιδεατό μας παίγνιο.

4.2.1 Το Πρωτόκολλο Αναλυτικά

Σε αυτήν την ενότητα παρουσιάζουμε το νέο πρωτόκολλο. Θεωρούμε μία ideal functionlity auth που ορίζεται ως εξής

$$auth((E_T(\sigma), E_T^{VR}(\sigma'), sk_T, pk_T), (E_T(vote; r))) = sign(b, sk_T, pk_T, (E_T(vote; r)))$$

με $b = 0$ αν $\sigma = \sigma'$ και $b = 1$ αλλιώς. Δηλαδή με κοινό input $(E_T(\sigma), E_T^{VR}(\sigma'), pk_T)$, ιδιωτικό input της αρχής το μυστικό κλειδί της sk_T και ιδιωτικό input του ψηφοφόρου την κρυπτογραφημένη ψήφο του $(E_T(vote; r))$ (και φυσικά με ιδιωτικές τυχαιότητες των παιχτών) δίνει μία έγκυρη υπογραφή αν τα credentials είναι ίδια και άκυρη διαφορετικά.

- **Registration**
  - Δημιουργούνται παράμετροι ομάδας $\mathcal{G}$ για στην οποία το DDH είναι δύσκολο.
  - Η T δημιουργεί κλειδιά κρυπτογράφησης (t, T) ως εξής: επιλέγει g_1 γεννήτορα και μυστικό κλειδί $t \in \mathbb{Z}_q$ και δίνει ως δημόσιο $h_1 = g_1^t$.

- Η T δημιουργεί κλειδί υπογραφής και επαλήθευσης για το σύστημα υπογραφών: επιλέγει $v \leftarrow_R \mathbb{Z}_q$ και δίνει ως δημόσιο κλειδί υπογραφής $y \leftarrow_R \mathbb{G}$ και κλειδί επαλήθευσης $k = g^v$.
- Κάθε παίχτης παίρνει (μέσα από τέλειο κανάλι) ένα ID_i και ένα μυστικό σ_i.
- Τέλος δημοσιποιεί δύο συναρτήσεις $H_1 : \{0,1\}^* \leftarrow \mathbb{G}$ και $H_2 : \{0,1\}^* \leftarrow \mathbb{Z}_q$. Αυτές τις μοντελοποιούμε ως Random Oracles.

- **Preelection**
  - Η αρχή αποφασίζει ποιοι ψηφοφόροι έχουν δικαίωμα ψήφου και δημοσιοποιεί το Voter Roll με στοιχεία $\{(ID_i, E_T^{VR}(\sigma_i))\}_{i=1}^{n_v}$.
  - Για κάθε υποψήφιο διαλέγει ένα τυχαίο στοιχείο v_j και δημοσιοποιεί το candidate slate C

- **Blind Authorization**
  - Κάθε παίχτης δημοσιοποιεί $E_T(\sigma_i), ID_i, pf_1$ όπου pf_1 είναι μία απόδειξη μηδενικής γνώσης γνώσης του σ_i.
  - Αφού περάσει το απαραίτητο διάστημα λήγει αυτή η φάση.
  - Η αρχή ελέγχει τις αποδείξεις και απορρίπτει όσα αιτήματα έχουν λάθος αποδείξεις.
  - Γίνεται έλεγχος για διπλοψηφίες. Συγκεκριμένα για κάθε ID_i γίνεται έλεγχος αν κατατέθηκαν δύο αιτήματα με ίδιο $E_T(\sigma_i)$ κάνοντας ανά δύο PETS σε αυτά με το ίδιο ID_i. Με κάποια πολιτική απορρίπτονται τα διπλά.
  - Για κάθε αίτημα σειριακά δίνονται υπογραφές ως εξής:
    1. Ο ψηφοφόρος διαλέγει ψήφο $vote$ και τυχαιότητα r και υπολογίζει $E_T(vote; r)$.
    2. παίρνει $(sig_1, sig_2) = auth((E_T(\sigma_i), E_T^{VR}(\sigma_i'), y), sk_T, (E_T(vote; r)))$. Ως a θεωρείται το $H_1(ID_i, j)$ με j τη γραμμή της αίτησης στο Bulletin Board. Με την H_2 υπολογίζεται το Challenge.

- **Vote**
  - Κάθε ψηφοφόρος στέλνει $E_T(vote; r), \alpha, E_T(sig_1), sig_2, pf_2$ όπου pf_2 είναι απόδειξη ορθής επιλογής ψήφου, δηλαδή ότι $v \in C$.

- **Tally**
  - Ελέγχει τις αποδείξεις και αγνοεί ψήφους με λανθασμένες αποδείξεις.
  - Για κάθε ψήφο η T υπολογίζει $\alpha h^{-sig_2} y^{-\epsilon}$ όπου $\epsilon = H_2(E_T(vote; r), \alpha)$ και το κρυπτογραφεί με κατάλληλη απόδειξη σε $E_T(validity)$.
  - Περνάει τα στοιχεία $E_T(vote), E_T(sig_1), E_T(validity)$ από ένα mix net.
  - Υψώνει τα $E_T(validity)$ στην v όπου v το ιδιωτικό κλειδί επαλήθευσης. Λόγω των ομομορφικών ιδιοτήτων του El Gamal έχουμε ότι αυτό ισούται με $E_T(validity^v)$.

– Κάνει PET στα $E_T(validity^v)$ και $E_T(sig_1)$ και αποκρυπτογραφεί τις ψήφους που δίνουν σε αυτό το στάδιο 1.

4.2.2 Ανάλυση του πρωτοκόλλου

Σε αυτήν την υποενότητα θα αναλύσουμε τις ιδιότητες του πρωτοκόλλου, εστιάζουντας πιο αναλυτικά στο Coercion Resistance. Στην παρούσα μορφή οι ιδιότητες αυτές στηρίζονται στο γεγονός ότι η αρχή T είναι έμπιστη. Αυτό δεν είναι φυσικά επιθυμητό. Παρατηρούμε όμως ότι αν αλλάξουμε την functionality $auth$ με ένα κατανεμημένο (t, n) πρωτόκολλο για τον υπολογισμό της μπορούμε να έχουμε τις ιδιότητες δεδομένου ότι έχουμε το πολύ t διεφθαρμένους παίχτες. Θα παρουσιάσουμε μία πρόταση για ένα τέτοιο πρωτόκολλο στη συνέχεια.

Για την επαλήθευσιμότητα ισχύουν ακριβώς τα παραπάνω, δηλαδή στην παρούσα μορφή βασιζόμαστε στην T. Το eligibility στηρίζεται τόσο στην T όσο και στην ιδιότητα $(l, l+1)$-unforgability του συστήματος υπογραφών. Κανένας αντίπαλος δεν μπορεί να ψηφίσει χωρίς υπογραφή. Συνεπώς κανένας αντίπαλος που κάνει πολυλογαριθμικό πλήθος corruptions δεν μπορεί να δώσει μία επιπλέον ψήφο. Φυσικά αυτό δεν είναι επιθυμητό. Θα θέλαμε λοιπόν για καλύτερες εγγυήσεις ασφάλειας να το αντικαθιστούσαμε με σύστημα υπογραφών που να αντέχει πολυωνυμικό αριθμό αιτημάτων.

Για το everlasting privacy επιχειρηματολογούμε όπως και στο FOO. Στο στάδιο Authorization δεν αποκαλύπτεται καμία πληροφορία για την επιλογή του ψηφοφόρου. Αυτό διασφαλίζεται από την ιδιότητα blindess του σχήματος υπογραφών. Στις κατατεθιμένες ψήφους παρατηρούμε ότι δεν υπάρχει κανένα στοιχείο για την ταυτότητα του χρήστη. Κατατίθεται μόνο μία κρυπτογραφημένη ψήφος, μία υπογραφή η οποία πάλι λόγω του blindness δεν μπορεί να συσχετιστεί με τις εκδόσεις υπογραφών που κατατέθηκαν και μια απόδειξη μηδενικής γνώσης. Θεωρώντας λοιπόν τέλεια ανώνυμο κανάλι κανένας semi honest unbounded αντίπαλος παρακολουθώντας τα δημόσια δεδομένα δεν μπορεί να συσχετίσει ψηφοφόρο με ψήφο.

4.2.2.1 Ασφάλεια ως προς coercion resistance

Εδώ θα προσαρμόσουμε το πλαίσιο των JCJ στο πρωτοκολλό μας και θα δείξουμε ότι έχει την ιδιότητα ακολουθώντας ουσιαστικά την απόδειξη που παρουσιάζεται στο JCJ.

Καταρχήν ας δούμε τι μπορεί να κάνει ένας υπό εκβιασμό ψηφοφόρος. Αρκεί να δώσει ένα ψεύτικο credential και να ισχυριστεί ότι είναι το δικό του. Δεδομένου ότι δεν μπορεί να αποκρυπτογραφήσει το αντίστοιχο από το Voter Roll, ο αντίπαλος δεν μπορεί αρχικά να ξέρει αν πήρε το αληθινό ή το ψεύτικο. Αν δοκιμάσει να ψηφίσει θα πάρει μία υπογραφή που από τις ιδιότητές της δεν θα μπορεί ο ίδιος να επαληθεύσει. Τέλος το mixnet τον κάνει να χάνει την πληροφορία για την ψήφο του. Θα δείξουμε πιο τυπικά τα παραπάνω ακολουθώντας το JCJ. Όπως και εκεί θα χρησιμοποιήσυμε ως κρυπτοσύστημα το Modified ElGamal.

4.2.2.1.1

Παρουσιάζουμε εδώ τα τροποποιημένα παίγνια για το coercion resistance. Η κύρια διαφορά με το πλαίσιο του JCJ είναι ότι εδώ έχουμε μία παραπάνω λειτουργία, την authorization. Πρακτικά αυτό είναι ένα πρωτόκολλο μεταξύ ενός ψηφοφόρου και της αρχής που καταλήγει στο να πάρει ο ψηφοφόρος ένα ψηφοδέλτιο, έγκυρο ή άκυρο. Σε αυτό το πρωτόκολλο ανταλλάσσονται δεδομένα που θεωρούμε ότι είναι γνωστά στον αντίπαλο $\mathcal{A}$.

Για το ιδεατό παιχνίδι ορίζουμε μία ιδεατή functionality, την $ideal - auth$. Αυτή σε κάθε αίτημα για δημιουργία ballot κάνει το εξής: για τους τίμιους ψηφοφόρους δίνει κανονικά τα ψηφοδέλτια ανάλογα με τον τρόπο που τα ζητάνε (έγκυρα ή άκυρα). Για τον $\mathcal{A}$ κοιτάει το κλειδί που δόθηκε και αν είναι κάποιο από τα κλειδιά των corrupted δίνει έγκυρο ψηφοδέλτιο και αν είναι από το κλειδί του coerced απαντάει ανάλογα με το κέρμα. Αν $b = 0$ δίνει άκυρη ψήφο αλλιώς έγκυρη. Φυσικά για άλλα κλειδιά δίνει άκυρη ψήφο. Η $ideal - auth$ (όπως και η $auth$) φροντίζει να δώσει μία έγκυρη ψήφο για κάθε κλειδί και συνεπώς λύνουμε σε αυτό το στάδιο το θέμα των διπλών ψήφων.

Όπως και στο JCJ στο ιδεατό πείραμα δίνουμε το αληθινό κλειδί στον $\mathcal{A}$. Επίσης πρέπει τα μόνα δεδομένα που έχει ο αντίπαλος στη διάθεσή του να είναι το τελικό tally μόνο δηλαδή δεν θα πρέπει να μαθαίνει τίποτα για κάθε κατατεθειμένη ψήφο και στο τέλος να αποφασίζει μόνο με βάση το τελικό αποτέλεσμα και τον αριθμό των άκυρων ψήφων. Στο ιδεατό πείραμα αυτό το πετυχαίνουμε μέσω της ideal tally που παίρνει ψηφοδέλτια και δίνει το αποτέλεσμα μόνο λαμβάνοντας υπ όψην την εγκυρότητά τους. Στην απόδειξη αυτό το πετυχαίνουμε φροντίζοντας να έχουμε τυχαία δεδομένα που μπορεί να παράξει και μόνος του.

Σημαντική λεπτομέρεια είναι ο ορισμός της auth. Για τις ανάγκες της απόδειξης θεωρούμε ότι είναι μια συνάρτηση που δίνει μία έγκυρη ή άκυρη ψήφο σε κάθε ψηφοφόρο βάσει του ιδιωτικού του input. Για να μπορούμε να το επικαλεστούμε αυτό θα πρέπει να είναι προσωμοιώσιμη από ένα ασφαλές πρωτόκολλο.

Algorithm: game c-resist

Input : k παράμετρος ασφάλειας

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

/* ο $\mathcal{A}$ διαμερίζει τους ψηφοφόρους σε corrupted (V)και μη */

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

/* Γίνεται register. Ο $\mathcal{A}$ παίρνει κλειδιά των V */

2 $\{(sk_i, pk_i) \leftarrow register(sk_R, ID_i, k)\}_{i\in\{V\cup U\}}$

/* Ο $\mathcal{A}$ διαλέγει ψηφοφόρο και ψήφο j, β για coerce */

3 $(j,\beta) \leftarrow \mathcal{A}(\{\sigma_i\}_{i\in V}, \textbf{Coerce})$

/* Γίνεται έλεγχος ότι διάλεξε σωστά (από το σύνολο U) */

4 **if** $\beta \notin CandSlate$ *or* $j \notin U$ **then**

5 ABORT

6 **end**

7 $b \leftarrow_R \{0,1\}$

8 **if** $b = 0$ **then**

9 $sk^* \leftarrow fakekey(PK_T, sk_j, pk_j\mathbb{G})$

10 $ballot_j \leftarrow auth(sk_j, pk_j, sk_T, pk_T, \beta, k)$

11 **else**

12 $sk^* = sk_j$

13 **end**

/* Παίρνουν Ballots οι honest */

14 $\{ballot_i \leftarrow auth(sk_i, pk_i, sk_T, pk_T, CandSlate, D, k)\}_{i\in HonVotes}$

/* Παίρνει Ballots ο $\mathcal{A}$ */

15 $ballots \leftarrow \mathcal{A}(\{sk_i\}_{i\in V}, sk^*, pk_T, CandSlate, BB)$

/* Ψηφίζουν οι honest */

16 $\{BB \Leftarrow vote(ballot_i)\}_{i\in HonVotes}$

17 **if** $b = 0$ **then**

18 $BB \Leftarrow vote(ballot_j)$

19 **end**

/* Ψηφίζει ο $\mathcal{A}$ */

20 $BB \Leftarrow \mathcal{A}(ballots, \textbf{Vote})$

/* Βγαίνει το αποτέλεσμα */

21 $(\mathbf{X}, P) \leftarrow tally(sk_T, BB, CandSlate, \{pk_i\}_{i\in V\cup U}, k)$

/* Ο $\mathcal{A}$ μαντεύει το αποτέλεσμα της ρίψης */

22 $b' = \mathcal{A}(\mathbf{X}, P, BB, \textbf{Guess})$

23 output $b == b'$

Algorithm: game c-resist-ideal

Input : k παράμετρος ασφάλειας

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

/* ο $\mathcal{A}$ διαμερίζει τους ψηφοφόρους σε corrupted (V)και μη */

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

/* Γίνεται register. Ο $\mathcal{A}$ παίρνει κλειδιά των V */

2 $\{(sk_i, pk_i) \leftarrow register(sk_R, ID_i, k)\}_{i \in \{V \cup U\}}$

/* Ο $\mathcal{A}$ διαλέγει ψηφοφόρο και ψήφο j, β για coerce */

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

/* Γίνεται έλεγχος ότι διάλεξε σωστά (από το σύνολο U) */

4 **if** $\beta \notin CandSlate$ *or* $j \notin U$ **then**

5 ABORT

6 **end**

7 $b \leftarrow_R \{0,1\}$

8 **if** $b = 0$ **then**

9 $ballot_j \leftarrow idealauth(sk_j, pk_j, sk_T, pk_T, \beta, k)$

10 **end**

11 $sk^* \leftarrow sk_j$ /* Παίρνουν Ballots οι honest */

12 $\{ballot_i \leftarrow auth(sk_i, pk_i, sk_T, pk_T, CandSlate, D, k)\}_{i \in HonVotes}$

/* Παίρνει Ballots ο $\mathcal{A}$ */

13 $ballots \leftarrow \mathcal{A}(\{sk_i\}_{i \in V}, sk^*, pk_T, CandSlate)$

/* Ψηφίζουν οι honest */

14 $\{BB \Leftarrow vote(ballot_i)\}_{i \in HonVotes}$

15 **if** $b = 0$ **then**

16 $BB \Leftarrow vote(ballot_j)$

17 **end**

/* Ψηφίζει ο $\mathcal{A}$ */

18 $BB \Leftarrow \mathcal{A}(ballots, \textbf{Vote})$

/* Βγαίνει το αποτέλεσμα */

19 $(\mathbf{X}, P) \leftarrow ideal - tally(sk_T, BB, CandSlate, \{pk_i\}_{i \in V \cup U}, k)$

/* Ο $\mathcal{A}$ μαντεύει το αποτέλεσμα της ρίψης */

20 $b' = \mathcal{A}(\mathbf{X}, \textbf{Guess})$

21 output $b == b'$

4.2.2.1.2

Θα δείξουμε τώρα ότι το προτεινόμενο πρωτόκολλο είναι coercion resistant. Ουσιαστικά μιμούμαστε την απόδειξη που παρουσιάζεται στο JCJ στο δικό μας σύστημα. Αυτό που δείχνουμε είναι ότι η πιθανότητα κάποιος PPT αντίπαλος $\mathcal{A}$ να νικήσει το παίγνιο $c - resist$ είναι αμελητέα μεγαλύτερη από την πιθανότητα να νικήσει το ιδεατό παιχνίδι που είναι ασφαλές

εξ ορισμού, δεδομένου ότι το DDH είναι δύσκολο. Παρακάτω είναι η προσομοίωση.

1. **Είσοδος:** Ο $\mathcal{S}$ δέχεται ως είσοδο g_1, g_2, h_1, h_2 και ένα στοιχείο w από μια κατανομή D. Η κατανομή αυτή καθορίζεται από την αβεβαιότητα του $\mathcal{A}$ και το w περιλαμβάνει για κάθε ψηφοφόρο την επιλογή του καθώς και ένα πλήθος άκυρων (για κάθε λόγο) ψήφους. Ο σκοπός του $\mathcal{S}$ είναι να αποφανθεί αν η τετράδα που πήρε είναι DH δηλαδή αν $\log_{g_1} h_1 = \log_{g_2} h_2$.

2. **Δημιουργία Παραμέτρων:** Αρχικά ο $\mathcal{S}$ δημιουργεί ένα κλειδί κρυπτογράφησης ως εξής: διαλέγει $x_1, x_2 \in \mathbb{Z}_q$ και υπολογίζει $h = g_1^{x_1} g_2^{x_2}$ Δίνει ως δημόσιο κλειδί g_1, g_2, h. Στη συνέχεια υπολογίζει κανονικά ζεύγος κλειδιών υπογραφής και επαλήθευσης (r, s, v) και $(g_3, g_4, y = g_3^{-r} g_4^{-s}, k = g_3^v)$. Θεωρούμε $d = 1$ αν είναι τετράδα DH και $d = 0$ διαφορετικά.

3. **Registration:** Στη συνέχεια προσομοιώνει την εγγραφή με ευθύ τρόπο. Διαλέγει σ_i για κάθε παίχτη τυχαία και δημοσιοποιεί το Voter Roll κρυπτογραφώντας τα με το δημόσιο κλειδί του.

4. **Corruption:** Ο $\mathcal{A}$ επιλέγει ποιους παίχτες να διαφθείρει. Έστω V το σύνολο αυτών και U οι υπόλοιποι.

5. **Coercion:** Ο $\mathcal{A}$ επιλέγει ποιον παίχτη θέλει να εξαναγκάσει σε μια συμπεριφορά. Επιλέγει (j, β) όπου j ο υπό εξαναγκασμό παίχτης και β η πρόθεσή του. Σε περίπτωση που δεν διαλέξει σωστά χάνει.

6. **Στρίψιμο Κέρματος:** Ο $\mathcal{S}$ στρίβει ένα κέρμα $b \leftarrow_R \{0, 1\}$. Ο $\mathcal{S}$ στρίβει ένα κέρμα $b \leftarrow_R \{0, 1\}$. Αν έρθει 0 δίνει $\sigma^* \leftarrow_R \mathbb{Z}_q$ στον αντίπαλο αλλιώς του δίνει $\sigma^* = \sigma_j$.

7. **Κατάθεση Credentials:** Ο $\mathcal{A}$ με βάση το w καταθέτει αιτήματα για υπογραφές για τους τίμιους παίχτες. Για κάθε στοιχείο δίνει $(E_T^{VR}(\sigma_i), E_T(\sigma_i'), pf_1)$ ως εξής: κρυπτογραφεί το σ_i' ως $(h_1^{u_i}, h_2^{u_i}, h^{u_i}\sigma_i')$ διαλέγοντας τυχαίο r_i. Τις αποδείξεις μπορεί να τις προσομοιώσει μέσω του ελέγχου του στο τυχαίο μαντείο. Καταθέτει και ο αντίπαλος αιτήματα της επιλογής του.

8. **Σβήσιμο διπλών ψήφων** Έχοντας το μυστικό κλειδί x_1, x_2 και ξέροντας τις δικές του επιλογές μπορεί να μαρκάρει τις ψήφους με ίδιο credential σύμφωνα με το πρωτόκολλο.

9. **Authorization για τίμιους παίχτες:** Ο $\mathcal{S}$ προσομοιώνει το Authorization για τους τίμιους παίχτες και για το σύνολο άκυρο ψήφων που έλαβε. Υπολογίζει τις υπογραφές για τα ψηφοδέλτια κανονικά και το πρώτο κομμάτι τους το κρυπτογραφεί με τον ίδιο τρόπο όπως πριν, δηλαδή επιλέγει u_i και υπολογίζει $(h_1^{r_i}, h_2^{r_i}, h^{r_i} sig_1)$. Κατασκευάζει λοιπόν με αυτόν τον τρόπο ένα σύνολο από ψηφοδέλτια.

10. **Authorization για τον $\mathcal{A}$:** Γίνεται με ευθύ τρόπο, σύμφωνα με το πρωτόκολλο.

11. **Vote Casting:** Ο $\mathcal{S}$ καταθέτει τα ballots για τους τίμιους και ο $\mathcal{A}$ καταθέτει τα ψηφοδέλτια της επιλογής του.

12. **Καταμέτρηση Ψήφων:** Ο $\mathcal{S}$ προσομοιώνει την καταμέτρηση με ευθύ τρόπο με το κλειδί του και τη γνώση των δικών του επιλογών.
13. **Απόφαση $\mathcal{A}$:** Ο $\mathcal{A}$ αποφασίζει το αποτέλεσμα του νομίσματος b'
14. **Έξοδος $\mathcal{S}$:** Ο $\mathcal{S}$ δίνει ως έξοδο 1 ανν $b = b'$.

Αν η είσοδος του $\mathcal{S}$ είναι DH δηλαδή αν $(g_1, g_2, h_1, h_2) = (g, g^a, g^b, g^{ab})$. Τότε όλες οι κρυπτογραφήσεις που έκανε ο $\mathcal{S}$ είναι έγκυρες αφού θα έχουμε

$$(h_1^{r_i}, h_2^{r_i}, mh^{r_i}) = (g^{ar_i}, g^{b(ar_i)}, m(g^{x_1}g^{bx_2})^{ar_i}) = (g_1^{ar_i}, g_2^{ar_i}, m(g_1^{x_1}g_2^{x_2})^{ar_i})$$

και συνεπώς η όψη του $\mathcal{A}$ είναι ακριβώς ίδια με την όψη του στο παιχνίδι $c-resist$. Συνεπώς έχουμε ότι

$$\Pr[S = 1|d = 1] = \Pr[\mathbf{Exp}^{\text{c-resist}}_{ES,\mathcal{A}}(\mathcal{V})] = 1 = \mathbf{Succ}^{\text{c-resits}}_{ES,\mathcal{A}}(\mathcal{V}) \tag{4.2}$$

όπου $\mathcal{V}$ η όψη του $\mathcal{A}$ στο πείραμα.

Αντίθετα αν η είσοδος του $\mathcal{S}$ δεν είναι DH δηλαδή αν $(g_1, g_2, h_1, h_2) = (g, g^a, g^b, g^c)$ με c τυχαίο τότε όλες οι κρυπτογραφήσεις που έκανε ο $\mathcal{S}$ κρύβουν πληροφοριοθεωριτικά τα κρυπτοκείμενα των honest παιχτών. Συνεπώς η μόνη πληροφορία που έχει ο αντίπαλος είναι αυτή που αποφασίζει μόνος του και τίποτε άλλο αφού τόσο τα κατατεθειμένα credentials όσο και οι υπογραφές των τίμιων παιχτών δεν του δίνουν καμία πληροφορία. Συνεπώς η όψη του είναι ίδια με το παιχνίδι c-resist-ideal. Έχουμε λοιπόν

$$(h_1^{r_i}, h_2^{r_i}, mh^{r_i}) = (g^{ar_i}, g^{b(ar_i)}, m(g^{x_1}g^{bx_2})^{ar_i}) = (g_1^{ar_i}, g_2^{ar_i}, m(g_1^{x_1}g_2^{x_2})^{ar_i})$$

και συνεπώς η όψη του $\mathcal{A}$ είναι ακριβώς ίδια με την όψη του στο παιχνίδι $c-resist$. Συνεπώς έχουμε ότι

$$\Pr[S = 1|d = 0] = \Pr[\mathbf{Exp}^{\text{c-resist-ideal}}_{ES,\mathcal{A}}(\mathcal{V})] = 1 = \mathbf{Succ}^{\text{c-resits-ideal}}_{ES,\mathcal{A}}(\mathcal{V}) \tag{4.3}$$

και συνδυάζοντας τις δύο αυτές σχέσεις έχουμε ότι

$$\mathbf{Adv}^{ddh}_{\mathcal{S}} = |\Pr[S = 1|d = 1] - \Pr[S = 1|d = 0]| = \mathbf{Adv}^{\text{c-resist}}_{ES,\mathcal{S}}$$

που μας δίνει και το ζητούμενο.

4.2.2.2 Threshold Protocol για Authentication

Σε αυτήν την υποενότητα δείχνουμε πως μπορούν να εκδοθούν υπογραφές από n οντότητες με το πολύ t από αυτές να είναι διεφθαρμένες. Θεωρούμε ότι οι οντότητες μοιράζονται ένα κλειδί s με ένα (t, n) Secret Sharing σχήμα. Κάθε οντότητα έχει ένα μυστικό Share v_i και δημοσιοποιεί ένα Commitment στο share της $\{g^{v_i}\}_{i=1}^n$. Παρακάτω φαίνονται τα βήματα για τον υπολογισμό του auth.

- Οι παίχτες δέχονται είσοδο $(E_T(\sigma), E_T^{VR}(\sigma'), a)$ με το a να προκύπτει όπως εξηγείται παραπάνω.

- Οι παίχτες κατανεμημένα συμφωνούν σε ένα τυχαίο S.
- Κάθε παίχτης υπολογίζει $E_T(ah^{-S}y^e)$ με απόδειξη ορθού υπολογισμού.
- Κάθε παίχτης υπολογίζει $(E_T(\sigma)/E_T(\sigma'))^{z_i}$ για τυχαίο z_i με κατάλληλη απόδειξη.
- Πολλαπλασιάζει τα δύο Ciphertexts και παίρνει $E_T(m) = E_T(ah^{-s}y^e)\cdot(E_T(\sigma)/E_T(\sigma'))^{z_i}$
- Κάθε παίχτης υπολογίζει $E_T(m)^{v_i}$ με κατάλληλη απόδειξη.
- Από υποσύνολο T με $|T| = t$ ορθά υπολογισμένα $E_T(m)$ υπολογίζεται $E_T(validity) = \prod_{i\in T} E_T(m)^{v_i\lambda_i(0)}$.

Για την ορθότητα βλέπουμε το εξής: Αν τα αρχικά ciphertexts είναι διαφορετικά, το $(E_T(\sigma)/E_T(\sigma'))^{z_i}$ είναι ομοιόμορφα κατανεμημένο για τυχαίο z_i και το ίδιο ισχύει για το τελικό αποτέλεσμα. Αν είναι ίσα τότε $E_T(\sigma)/E_T(\sigma') = E_T(1)$ και σαν τελικό αποτέλεσμα έχουμε

$$\begin{aligned} E_T(validity) &= \prod_{i\in T} E_T(m)^{v_i\lambda_i(0)} \\ &= \prod_{i\in T} E_T(m^{v_i\lambda_i(0)}) \\ &= E_T(\prod_{i\in T} m^{v_i\lambda_i(0)}) \\ &= E_T(m^{\sum_{i\in T} v_i\lambda_i(0))} \\ &= E_T(m^v) \\ &= E_T(ah^{-S}y^{-e})^v) \end{aligned}$$

που είναι και το επιθυμητό.

4.3 Συμπεράσματα και Μελλοντική Έρευνα

Στην προηγούμενη ενότητα ορίσαμε ένα νέο primitive που το ονομάσαμε Conditional Blind Signatures που επιτρέπει στον υπογράφων να δίνει υπογραφές, διαλέγοντας ποιες θα είναι έγκυρες χωρίς να παραβιάζει την ιδιωτηκότητα των παιχτών. Ορίσαμε τις επιθυμητές, για αυτό το Primitive, ιδιότητες και το πλαίσιο για τον έλεγχό τους επεκτείνοντας το αντίστοιχο των τυφλών υπογραφών. Στη συνέχεια ορίσαμε ένα τέτοιο σχήμα που βασίζεται στο Okamoto-Schnorr, αποδείξαμε τις ιδιότητες του και εξετάσαμε πώς, χρησιμοποιώντας μόνο το κλειδί επαλήθευσης, να κάνουμε το πρωτόκολλο βέλτιστο ως προς τον αριθμό των γύρων που απαιτεί.

Στη συνέχεια, χρησιμοποιώντας αυτό το Primitive, ορίσαμε ένα νέο σχήμα απομακρυσμένων ηλεκτρονικών ψηφοφοριών που είναι ανθεκτικό σε επιθέσεις εξαναγκασμού, έχει καλή αποδοτικότητα και ενισχύει την ιδιωτικότητα των ψηφοφόρων. Συγκεκριμένα αν υποθέσουμε ότι η επικοινωνία γίνεται από τέλεια ανώνυμο κανάλι, το νέο σχήμα παρέχει πληροφοριοθεωριτική ιδιωτικότητα στους ψηφοφόρους ακόμη και ως προς τις αρχές.

Η μελλοντική έρευνα θα έχει δύο κατευθύνσεις: Αρχικά, το ίδιο το Primitive είναι από μόνο του ενδιαφέρον και σκοπεύουμε να μελετήσουμε καλύτερα τις ιδιότητες του. Το μεγαλύτερο μειονέκτημα στο σχήμα που στηρίζεται στις υπογραφές Okamoto Schnorr είναι το πολυλογαριθμικό πλήθος αιτήσεων που επιτρέπουμε στον αντίπαλο. Θα θέλαμε να δημιουργήσουμε νέα σχήματα που να αντέχουν πολυωνιμικό αριθμό αιτήσεων. Επίσης θα θέλαμε να μελετήσουμε και άλλες εφαρμογές του συγκεκριμένου primitive, κυρίως γύρω από την περιοχή των ανώνυμων πιστοποιητικών.

Η δεύτερη κατεύθυνση είναι στις ηλεκτρονικές ψηφοφορίες. Συγκεκριμένα, σκοπεύουμε να αναλύσουμε καλύτερα το παρουσιαζόμενο πρωτόκολλο εστιάζοντας κυρίως στα μεγέθη των παραμέτρων ασφάλειας που χρειαζόμαστε για να γίνει πρακτικό. Θα θέλαμε επίσης να εξετάσουμε πως μπορούμε να το μορφοποιήσουμε με άλλα πιθανά σχήματα Conditional Blind Signatures, πιο ασφαλή και αποδοτικά. Το πιο φιλόδοξο βήμα είναι να εξετάσουμε γενικά το Verifiability σε coercion resistant πρωτόκολλα ψηφοφορίας και συγκεκριμένα να δούμε αν μπορούμε να έχουμε ταυτόχρονα και Coercion Resistance και Univesal Verifiability.

Bibliography

[1] Roberto Araújo, Sébastien Foulle, and Jacques Traoré. "A practical and secure coercion-resistant scheme for remote elections". In: *Frontiers of Electronic Voting*. 2007.

[2] Roberto Araújo and Jacques Traoré. "A Practical Coercion Resistant Voting Scheme Revisited". In: *VOTE-ID*. 2013, pp. 193–209.

[3] Roberto Araújo et al. "Remote Electronic Voting Can Be Efficient, Verifiable and Coercion-Resistant". In: *Financial Cryptography and Data Security - FC 2016 International Workshops, BITCOIN, VOTING, and WAHC, Christ Church, Barbados, February 26, 2016, Revised Selected Papers*. Ed. by Jeremy Clark et al. Vol. 9604. Lecture Notes in Computer Science. Springer, 2016, pp. 224–232. ISBN: 978-3-662-53356-7. DOI: 10.1007/978-3-662-53357-4_15. URL: http://dx.doi.org/10.1007/978-3-662-53357-4_15.

[4] Roberto Araújo et al. "Towards Practical and Secure Coercion-Resistant Electronic Elections". In: *CANS*. 2010, pp. 278–297.

[5] Sanjeev Arora and Boaz Barak. *Computational Complexity - A Modern Approach*. Cambridge University Press, 2009. ISBN: 978-0-521-42426-4. URL: http://www.cambridge.org/catalogue/catalogue.asp?isbn=9780521424264.

[6] Mihir Bellare and Phillip Rogaway. "Random Oracles are Practical: A Paradigm for Designing Efficient Protocols". In: *CCS '93, Proceedings of the 1st ACM Conference on Computer and Communications Security, Fairfax, Virginia, USA, November 3-5, 1993*. Ed. by Dorothy E. Denning et al. ACM, 1993, pp. 62–73. ISBN: 0-89791-629-8. DOI: 10.1145/168588.168596. URL: http://doi.acm.org/10.1145/168588.168596.

[7] Mihir Bellare et al. "The One-More-RSA-Inversion Problems and the Security of Chaum's Blind Signature Scheme". In: *J. Cryptology* 16.3 (2003), pp. 185–215. DOI: 10.1007/s00145-002-0120-1. URL: http://dx.doi.org/10.1007/s00145-002-0120-1.

[8] Jan Camenisch and Anna Lysyanskaya. "Signature Schemes and Anonymous Credentials from Bilinear Maps". In: *Advances in Cryptology - CRYPTO 2004, 24th Annual International CryptologyConference, Santa Barbara, California, USA, August 15-19,*

2004, Proceedings. 2004, pp. 56–72. DOI: 10.1007/978-3-540-28628-8_4. URL: http://dx.doi.org/10.1007/978-3-540-28628-8_4.

[9] Ran Canetti, Oded Goldreich, and Shai Halevi. "The random oracle methodology, revisited". In: *J. ACM* 51.4 (2004), pp. 557–594. DOI: 10.1145/1008731.1008734. URL: http://doi.acm.org/10.1145/1008731.1008734.

[10] David Chaum. "Blind Signatures for Untraceable Payments". In: *Advances in Cryptology: Proceedings of CRYPTO '82, Santa Barbara, California, USA, August 23-25, 1982*. 1982, pp. 199–203.

[11] Jeremy Clark and Urs Hengartner. "Selections: Internet Voting with Over-the-Shoulder Coercion-Resistance". In: *Financial Cryptography and Data Security: 15th International Conference, FC 2011, Gros Islet, St. Lucia, February 28 - March 4, 2011, Revised Selected Papers*. Ed. by George Danezis. Berlin, Heidelberg: Springer Berlin Heidelberg, 2012, pp. 47–61. ISBN: 978-3-642-27576-0. DOI: 10.1007/978-3-642-27576-0_4. URL: http://dx.doi.org/10.1007/978-3-642-27576-0_4.

[12] Jeremy Clark et al., eds. *Financial Cryptography and Data Security - FC 2016 International Workshops, BITCOIN, VOTING, and WAHC, Christ Church, Barbados, February 26, 2016, Revised Selected Papers*. Vol. 9604. Lecture Notes in Computer Science. Springer, 2016. ISBN: 978-3-662-53356-7. DOI: 10.1007/978-3-662-53357-4. URL: http://dx.doi.org/10.1007/978-3-662-53357-4.

[13] Thomas H. Cormen et al. *Introduction to Algorithms (3. ed.)* MIT Press, 2009. ISBN: 978-0-262-03384-8. URL: http://mitpress.mit.edu/books/introduction-algorithms.

[14] Ronald Cramer, Rosario Gennaro, and Berry Schoenmakers. "A Secure and Optimally Efficient Multi-Authority Election Scheme". In: Springer-Verlag, 1997, pp. 103–118.

[15] Ivan Damgård. *On Sigma Protocols*. 2010. URL: www.cs.au.dk/~ivan/Sigma.pdf.

[16] W. Diffie and M. Hellman. "New Directions in Cryptography". In: *IEEE Trans. Inf. Theor.* 22.6 (Sept. 2006), pp. 644–654. ISSN: 0018-9448. DOI: 10.1109/TIT.1976.1055638. URL: http://dx.doi.org/10.1109/TIT.1976.1055638.

[17] Uriel Feige and Adi Shamir. "Witness Indistinguishable and Witness Hiding Protocols". In: *Proceedings of the 22nd Annual ACM Symposium on Theory of Computing, May 13-17, 1990, Baltimore, Maryland, USA*. 1990, pp. 416–426. DOI: 10.1145/100216.100272. URL: http://doi.acm.org/10.1145/100216.100272.

[18] Paul Feldman. "A Practical Scheme for Non-interactive Verifiable Secret Sharing". In: *28th Annual Symposium on Foundations of Computer Science, Los Angeles, California, USA, 27-29 October 1987*. IEEE Computer Society, 1987, pp. 427–437. ISBN: 0-8186-0807-2. DOI: 10.1109/SFCS.1987.4. URL: http://dx.doi.org/10.1109/SFCS.1987.4.

[19] Amos Fiat and Adi Shamir. "How to Prove Yourself: Practical Solutions to Identification and Signature Problems". In: *Advances in Cryptology - CRYPTO '86, Santa Barbara, California, USA, 1986, Proceedings*. 1986, pp. 186–194. DOI: 10.1007/3-540-47721-7_12. URL: http://dx.doi.org/10.1007/3-540-47721-7_12.

[20] Atsushi Fujioka, Tatsuaki Okamoto, and Kazuo Ohta. "A Practical Secret Voting Scheme for Large Scale Elections". In: *Advances in Cryptology - AUSCRYPT '92, Workshop on the Theory and Application of Cryptographic Techniques, Gold Coast, Queensland, Australia, December 13-16, 1992, Proceedings*. 1992, pp. 244–251. DOI: 10.1007/3-540-57220-1_66. URL: http://dx.doi.org/10.1007/3-540-57220-1_66.

[21] Oded Goldreich. *The Foundations of Cryptography - Volume 1, Basic Techniques*. Cambridge University Press, 2001. ISBN: 0-521-79172-3.

[22] Oded Goldreich. *The Foundations of Cryptography - Volume 2, Basic Applications*. Cambridge University Press, 2004. ISBN: 0-521-83084-2.

[23] Oded Goldreich, Silvio Micali, and Avi Wigderson. "How to Play any Mental Game or A Completeness Theorem for Protocols with Honest Majority". In: *Proceedings of the 19th Annual ACM Symposium on Theory of Computing, 1987, New York, New York, USA*. Ed. by Alfred V. Aho. ACM, 1987, pp. 218–229. ISBN: 0-89791-221-7. DOI: 10.1145/28395.28420. URL: http://doi.acm.org/10.1145/28395.28420.

[24] Oded Goldreich, Silvio Micali, and Avi Wigderson. "Proofs that Yield Nothing But Their Validity for All Languages in NP Have Zero-Knowledge Proof Systems". In: *J. ACM* 38.3 (1991), pp. 691–729. DOI: 10.1145/116825.116852. URL: http://doi.acm.org/10.1145/116825.116852.

[25] Shafi Goldwasser and Silvio Micali. "Probabilistic Encryption". In: *J. Comput. Syst. Sci.* 28.2 (1984), pp. 270–299. DOI: 10.1016/0022-0000(84)90070-9. URL: http://dx.doi.org/10.1016/0022-0000(84)90070-9.

[26] Shafi Goldwasser, Silvio Micali, and Charles Rackoff. "The Knowledge Complexity of Interactive Proof Systems". In: *SIAM J. Comput.* 18.1 (1989), pp. 186–208. DOI: 10.1137/0218012. URL: http://dx.doi.org/10.1137/0218012.

[27] Panagiotis Grontas. "Secure Multi Party Computations for Electronic Voting". MA thesis. Athens, Greece: MPLA Graduate Program, University of Athens, 2014. URL: mpla.math.uoa.gr/media/theses/msc/P.%20Grontas.pdf.

[28] Martin Hirt and Kazue Sako. "Efficient receipt-free voting based on homomorphic encryption". In: *Proceedings of the 19th international conference on Theory and application of cryptographic techniques*. EUROCRYPT'00. Bruges, Belgium: Springer-Verlag, 2000, pp. 539–556. ISBN: 3-540-67517-5. URL: http://dl.acm.org/citation.cfm?id=1756169.1756222.

[29] Markus Jakobsson and Ari Juels. "Mix and Match: Secure Function Evaluation via Ciphertexts". In: *ASIACRYPT '00: Proceedings of the 6th International Conference on the Theory and Application of Cryptology and Information Security*. London, UK: Springer-Verlag, 2000, pp. 162–177. URL: http://markus-jakobsson.com/papers/jakobsson-asiacrypt00-mixmatch.pdf.

[30] Markus Jakobsson, Kazue Sako, and Russell Impagliazzo. "Designated Verifier Proofs and Their Applications". In: *Advances in Cryptology - EUROCRYPT '96, International Conference on the Theory and Application of Cryptographic Techniques, Saragossa, Spain, May 12-16, 1996, Proceeding*. 1996, pp. 143–154. DOI: 10.1007/3-540-68339-9_13. URL: http://dx.doi.org/10.1007/3-540-68339-9_13.

[31] Ari Juels, Dario Catalano, and Markus Jakobsson. "Coercion-resistant electronic elections". In: *Proceedings of the 2005 ACM Workshop on Privacy in the Electronic Society, WPES 2005, Alexandria, VA, USA, November 7, 2005*. 2005, pp. 61–70. DOI: 10.1145/1102199.1102213. URL: http://doi.acm.org/10.1145/1102199.1102213.

[32] Ari Juels, Michael Luby, and Rafail Ostrovsky. "Security of Blind Digital Signatures (Extended Abstract)". In: *Advances in Cryptology - CRYPTO '97, 17th Annual International Cryptology Conference, Santa Barbara, California, USA, August 17-21, 1997, Proceedings*. 1997, pp. 150–164. DOI: 10.1007/BFb0052233. URL: http://dx.doi.org/10.1007/BFb0052233.

[33] Jonathan Katz. *Digital Signatures*. Springer, 2010. ISBN: 978-0-387-27711-0. DOI: 10.1007/978-0-387-27712-7. URL: http://dx.doi.org/10.1007/978-0-387-27712-7.

[34] Jonathan Katz and Yehuda Lindell. *Introduction to Modern Cryptography*. Chapman and Hall/CRC Press, 2007. ISBN: 978-1-58488-551-1.

[35] Yehuda Lindell. "How To Simulate It - A Tutorial on the Simulation Proof Technique". In: *IACR Cryptology ePrint Archive* 2016 (2016), p. 46. URL: http://eprint.iacr.org/2016/046.

[36] Miyako Ohkubo et al. "An Improvement on a Practical Secret Voting Scheme". English. In: *Information Security*. Vol. 1729. Lecture Notes in Computer Science. Springer Berlin Heidelberg, 1999, pp. 225–234. ISBN: 978-3-540-66695-0. DOI: 10.1007/3-540-47790-X_19. URL: http://dx.doi.org/10.1007/3-540-47790-X_19.

[37] Tatsuaki Okamoto. "Provably Secure and Practical Identification Schemes and Corresponding Signature Schemes". In: *Advances in Cryptology - CRYPTO '92, 12th Annual International Cryptology Conference, Santa Barbara, California, USA, August 16-20, 1992, Proceedings*. 1992, pp. 31–53. DOI: 10.1007/3-540-48071-4_3. URL: http://dx.doi.org/10.1007/3-540-48071-4_3.

[38] Tatsuaki Okamoto. "Receipt-free electronic voting schemes for large scale elections". In: *Security Protocols*. Springer. 1998, pp. 25–35.

[39] Christos H. Papadimitriou. *Computational complexity*. Addison-Wesley, 1994. ISBN: 978-0-201-53082-7.

[40] David Pointcheval and Jacques Stern. "Provably Secure Blind Signature Schemes". In: *Advances in Cryptology - ASIACRYPT '96, International Conference on the Theory and Applications of Cryptology and Information Security, Kyongju, Korea, November 3-7, 1996, Proceedings*. 1996, pp. 252–265. DOI: 10.1007/BFb0034852. URL: http://dx.doi.org/10.1007/BFb0034852.

[41] David Pointcheval and Jacques Stern. "Security Arguments for Digital Signatures and Blind Signatures". In: *J. Cryptology* 13.3 (2000), pp. 361–396. DOI: 10.1007/s001450010003. URL: http://dx.doi.org/10.1007/s001450010003.

[42] Ronald L. Rivest, Adi Shamir, and Leonard M. Adleman. "A Method for Obtaining Digital Signatures and Public-Key Cryptosystems". In: *Commun. ACM* 21.2 (1978), pp. 120–126. DOI: 10.1145/359340.359342. URL: http://doi.acm.org/10.1145/359340.359342.

[43] Peter Y. A. Ryan, Peter B. Rønne, and Vincenzo Iovino. "Selene: Voting with Transparent Verifiability and Coercion-Mitigation". In: *Financial Cryptography and Data Security - FC 2016 International Workshops, BITCOIN, VOTING, and WAHC, Christ Church, Barbados, February 26, 2016, Revised Selected Papers*. Ed. by Jeremy Clark et al. Vol. 9604. Lecture Notes in Computer Science. Springer, 2016, pp. 176–192. ISBN: 978-3-662-53356-7. DOI: 10.1007/978-3-662-53357-4_12. URL: http://dx.doi.org/10.1007/978-3-662-53357-4_12.

[44] Michael Schlapfer et al. "Efficient Vote Authorization in Coercion-Resistant Internet Voting." In: *VOTE-ID*. Ed. by Aggelos Kiayias and Helger Lipmaa. Vol. 7187. Lecture Notes in Computer Science. Springer, 2011, pp. 71–88. ISBN: 978-3-642-32746-9. URL: http://dblp.uni-trier.de/db/conf/voteid/voteid2011.html#SchlapferHKS11.

[45] Claus-Peter Schnorr. "Efficient Identification and Signatures for Smart Cards". In: *Advances in Cryptology - CRYPTO '89, 9th Annual International Cryptology Conference, Santa Barbara, California, USA, August 20-24, 1989, Proceedings*. 1989, pp. 239–252. DOI: 10.1007/0-387-34805-0_22. URL: http://dx.doi.org/10.1007/0-387-34805-0_22.

[46] Dominique Schröder and Dominique Unruh. "Security of Blind Signatures Revisited". In: *IACR Cryptology ePrint Archive* 2011 (2011), p. 316. URL: http://eprint.iacr.org/2011/316.

[47] Adi Shamir. "How to Share a Secret". In: *Commun. ACM* 22.11 (1979), pp. 612–613. DOI: 10.1145/359168.359176. URL: http://doi.acm.org/10.1145/359168.359176.

[48] Adi Shamir. "IP = PSPACE". In: *J. ACM* 39.4 (1992), pp. 869–877. DOI: 10.1145/146585.146609. URL: http://doi.acm.org/10.1145/146585.146609.

[49] Victor Shoup. *A computational introduction to number theory and algebra*. Cambridge University Press, 2006. ISBN: 978-0-521-85154-1.

[50] Warren D. Smith. *New cryptographic voting scheme with best-known theoretical properties*. June 2005.

[51] Oliver Spycher et al. "A new approach towards coercion-resistant remote e-voting in linear time". In: *Proceedings of the 15th international conference on Financial Cryptography and Data Security*. FC'11. Gros Islet, St. Lucia: Springer-Verlag, 2012, pp. 182–189. ISBN: 978-3-642-27575-3. DOI: 10.1007/978-3-642-27576-0_15. URL: http://dx.doi.org/10.1007/978-3-642-27576-0_15.

[52] Stefan G. Weber, Roberto Araujo, and Johannes Buchmann. "On Coercion-Resistant Electronic Elections with Linear Work." In: *ARES*. IEEE Computer Society, Apr. 25, 2008, pp. 908–916. URL: http://dblp.uni-trier.de/db/conf/IEEEares/ares2007.html#WeberAB07.