1 Βασικές Έννοιες Ιδιωτικότητας

Τα κρυπτογραφικά εργαλεία που συζητήσαμε μέχρι στιγμής δεν μπορούν να λύσουν το πρόβλημα της ανάγκης για ιδιωτικότητα των χρηστών ενός συστήματος. Η ιδιωτικότητα με την έννοια της ανωνυμίας είναι μια διαφορετική απαίτηση από κάθε άλλο ζήτημα που μας απασχόλησε. Γι' αυτό το λόγο θα επικεντρώσουμε τις προσπάθειές μας στην απόκρυψη της ταυτότητας των συμμετεχόντων ενός συστήματος. Στην ενότητα αυτή θα εστιάσουμε σε δύο σημαντικά κρυπτογραφικά εργαλεία που σχετίζονται με την ιδιωτικότητα, τις τυφλές υπογραφές και τους mix-servers.

1.1 Τυφλές υπογραφές

Η τυφλή υπογραφή είναι μία μέθοδος που επιτρέπει σε έναν υπογράφοντα να ταυτοποιήσει ένα έγγραφο χωρίς να έχει κάποια πληροφορία για αυτό. Οι δύο βασικοί στόχοι μιας τυφλής υπογραφής είναι η μη πλαστογράφηση και η τυφλότητα (blindness), όπου η τυφλότητα αναφέρεται στην αδυναμία του υπογράφοντα να αντλήσει κάποια πληροφόρηση από το έγγραφο που υπογράφει. Στην ενότητα αυτή εισάγουμε την έννοια της τυφλής υπογραφής και περιγράφουμε τις τυφλές υπογραφές Chaum, που εισήχθησαν από τον David Chaum το 1982.

Ορισμός 1.1.1. Ένα ***σχήμα τυφλών υπογραφών (blind signature scheme)*** είναι μια τριάδα (GGen, Sign, Verify) τέτοια ώστε

1. - Ο GGen είναι ο αλγόριθμος παραγωγής των κλειδιών που έχει ως είσοδο το μήκος του κλειδιού λ και εξάγει το ζεύγος κλειδιών (vk, sk).
   - Το Sign είναι ένα πρωτόκολλο που τρέχει μεταξύ δύο διαδραστικών προγραμμάτων, $\langle \mathcal{U}, \mathcal{S} \rangle$. Ορίζουμε ως $\text{out}^{\mathcal{U}}_{\mathcal{U},\mathcal{S}}(m, sk)$ την έξοδο του $\mathcal{U}$,όταν το Sign εκτελεστεί απο τους $\mathcal{U}$ και $\mathcal{S}$ με είσοδο (m, sk), όπου m το μήνυμα, του οποίου το περιεχόμενο θα αποκρύψει ο $\mathcal{U}$ και θα ζητήσει απο τον $\mathcal{S}$ να υπογράψει.
   - Ο Verify είναι ο αλγόριθμος επαλήθευσης, όπου για $\sigma \leftarrow \text{out}^{\mathcal{U}}_{\mathcal{U},\mathcal{S}}$, με είσοδο τα (vk, m, σ) ελέγχει την εγκυρότητατα της υπογραφής και εξάγει 1 στην περίπτωση που η υπογραφή είναι έγκυρη και 0 αλλιώς.
2. Ισχύουν οι ακόλουθες ιδιότητες:
   - **Ορθότητα (Correctness):**
   $$\underset{(vk,sk)\leftarrow \text{GGen}(1^\lambda)}{\text{Prob}[\text{Verify}(vk, m, \text{out}^{\mathcal{U}}_{\mathcal{U},\mathcal{S}}(m, sk)) = 1]} = 1$$
   - **Μη πλαστογράφηση (Unforgeability):** Για κάθε πολυωνυμικό αλγόριθμο $\mathcal{A}$ που τρέχει το πρωτόκολλο Sign με τον $\mathcal{S}$ και ζητά την υπογραφή q μηνυμάτων $(m_i)_{i=1}^q$ πρέπει να ισχύει ότι:
   $$\text{Prob}\left[[\wedge_{i=1}^q \text{Verify}(m_i, \sigma_i) = 1] \wedge [q > \text{Πλήθος επιτυχημένων εφαρμογών του Sign}]\right] = \text{negl}(\lambda)$$
   - **Τυφλότητα (blindness):** Για κάθε πολυωνυμικό πρόγραμμα $\mathcal{S}^*$ υπάρχει μία μη ντετερμινιστική μηχανή Turing (PTM) Sim που θα την ονομάζουμε Simulator τέτοια ώστε για κάθε μήνυμα m οι τυχαίες μεταβλητές $(\alpha, Sim(vk, sk))$ και $\text{out}^{\mathcal{U},\mathcal{S}^*}_{\mathcal{U},\mathcal{S}^*}(m, sk)$ είναι αδιαχώριστες, όπου α είναι $\perp$ στις περιπτώσεις που ο $\mathcal{U}$ αποτυγχάνει να εξάγει μια υπογραφή από το διαλογικό πρωτόκολλο με τον $\mathcal{S}^*$ ενώ στις άλλες περιπτώσεις ισχύει ότι $\sigma \leftarrow \text{out}^{\mathcal{S}}_{\mathcal{U},\mathcal{S}}$ και $(vk, sk) \leftarrow \text{GGen}(1^\lambda)$. Συνοψίζοντας,
   $$\forall \mathcal{A} \left| \text{Prob}[\mathcal{A}(Sim(vk, sk)) = 1] - \text{Prob}[\mathcal{A}(\text{out}^{\mathcal{U},\mathcal{S}^*}_{\mathcal{U},\mathcal{S}^*}(m, sk) = 1] \right| \leq \epsilon$$
   Όταν το ε είναι αμεληταίο, τότε θα λέμε πως έχουμε ***στατιστική τυφλότητα (statistical blindness)***. Όταν $\varepsilon = 0$, τότε έχουμε ***τέλεια τυφλότητα (perfect blindness)***.

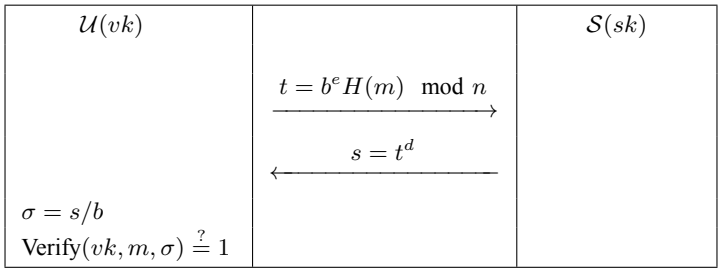
1.2 Σχήμα τυφλών υπογραφών Chaum

Η τυφλή υπογραφή Chaum βασίζεται στο σχήμα υπογραφών RSA με συναρτήσεις κατακερματισμού πλήρους πεδίου ορισμού. Όπως είδαμε στην Ενότητα **??** το μήνυμα M υπογράφεται με το $H(M)^d = \sigma$ και ένα ζευγάρι μυνήματος και υπογραφής (M, σ) επικυρώνεται ελέγχοντας αν $\sigma^e \equiv H(M) \bmod n$. Θα τροποποιήσουμε αυτό το σχήμα για να επιτύχουμε τυφλές υπογραφές.

Ορισμός 1.2.1. Έστω e ένας πρώτος αριθμός, M μια συμβολοακολουθία, n ένα RSA modulus και H μια συνάρτηση κατακερματισμού. Ένα ***σχήμα τυφλών υπογραφών (blind signature scheme)*** είναι μια τριάδα (GGen, Sign, Verify) τέτοια ώστε

- Ο αλγόριθμος GGen είναι ο αλγόριθμος παραγωγής κλειδιού: Επιλέγονται δύο πρώτοι αριθμοί p και q, τέτοιοι ώστε $|p| = |q| = \lambda$. Υπολογίζονται οι $n = pq$ και $\phi(n) = (p-1)(q-1)$. Επιλέγεται ένας πρώτος $e < \phi(n)$ τέτοιος ώστε $\gcd(e, \phi(n)) = 1$ και υπολογίζεται ο $d \equiv e^{-1} \mod \phi(n)$.
- Το Sign είναι ένα πρωτόκολλο υπογραφής, το οποίο λειτουργεί ως εξής:
  1. Ο $\mathcal{U}$ επιλέγει ένα $b \in \mathbb{Z}_n^*$ και θέτει $t = b^e H(m) \mod n$. Στέλνει στον $\mathcal{S}$ το t
  2. Ο $\mathcal{S}$ στέλνει στον $\mathcal{U}$ το $s = t^d \mod n$
  3. Ο $\mathcal{U}$ θέτει $\sigma = s/b \mod n$ και επιστρέφει το σ ως την υπογραφή
- Ο αλγόριθμος επαλήθευσης Verify: Για κάθε (M, σ), ελέγχει αν $\sigma^e = H(M) \mod n$. Σε περίπτωση που ισχύει η ισότητα, ο αλγόριθμος επιστρέφει `True`, αλλιώς επιστρέφει `False`.

Το σχήμα 1 περιγράφει τον γενικό τύπο των αλγορίθμων για έναν χρήστη $\mathcal{U}$ και έναν υπογράφοντα $\mathcal{S}$ στις τυφλές υπογραφές του Chaum.



Σχήμα 1: Η δημιουργία τυφλής υπογραφής Chaum.

Θεώρημα 1.2.1. *Η τυφλή υπογραφή Chaum ικανοποιεί την στατιστική τυφλότητα.*

Απόδειξη. Κατασκευάζουμε τον $Sim(vk, sk)$ έτσι ώστε να δίνει στον $\mathcal{S}^*$ ένα τυχαίο στοιχείο $t \in \mathbb{Z}_m^*$. Όταν ο $\mathcal{S}^*$ τρέξει με τον $\mathcal{U}$ θα δεχθεί ως είσοδο ένα $H(m)b^e$. Επειδή το $H(m)b^e$ είναι ομοιόμορφα κατανεμημένο στο $\mathbb{Z}_n^*$ θα έχουμε ότι

$$\begin{aligned}\left|\text{Prob}[\mathcal{A}(Sim(vk, sk)) = 1] - \text{Prob}[\mathcal{A}(\text{out}_{\mathcal{U}^*, \mathcal{S}^*}^{\mathcal{U}^*}(m, sk)) = 1]\right| &\leq \Delta[Sim(vk, sk), \text{out}_{\mathcal{U}, \mathcal{S}^*}^{\mathcal{S}^*}(m, sk)] \\ &= \left|\frac{1}{\phi(n)} - \frac{1}{\phi(n)}\right| \\ &= 0\end{aligned}$$

■

Σύστημα Ηλεκτρονικού Χρήματος βασισμένο στις Τυφλές Υπογραφές Chaum

Όταν πραγματοποιούμε μία αγορά με κανονικά χρήματα ή επιταγές, ένας πωλητής μπορεί να ελέγξει την εγκυρότητα τους. Σε μια κοινωνία όμως που πρέπει να έχει τη δυνατότητα απαλλαγής από την ανάγκη για απτό χρήμα χρειάζεται να αναπτυχθούν νέες μέθοδοι για επικυρώση των ηλεκτρονικών συναλλαγών. Συστήματα όπως το ***ηλεκτρονικό χρήμα (e-cash)*** χρησιμοποιούνται για την συμμετοχή μιας τρίτης μεριάς στην συναλλαγή έτσι ώστε να εξασφαλιστεί ότι και ο αγοραστής και ο πωλητής είναι τίμιοι και προστατευμένοι.

Θεωρούμε το εξής σενάριο, στο οποίο συμμετέχουν τρεις πλευρές: η τράπεζα, ο χρήστης της τράπεζας Βαγγέλης και ο πωλητής Μάρκος. Υποθέτουμε πως ο Βαγγέλης αποφασίζει να αγοράσει κάτι από το μαγαζί του Μάρκου. Έχει έναν λογαριασμό στην τράπεζα και αναγνωρίζει τον εαυτό του ως έναν αυθεντικό κάτοχο λογαριασμού. Δίνει στην τράπεζα έναν φάκελο που έχει τον κωδικό αριθμό του και η τράπεζα το υπογράφει χωρίς να ανοίξει τον φάκελο. Η τράπεζα έχει ένα συγκεκριμένη μονάδα ως ***ηλεκτρονικό νόμισμα (e-coin)***, π.χ. $10. Το ποσόν αυτό αφαιρείται από τον λογαριασμό του Βαγγέλη και αντ' αυτού προστίθεται ένα ηλεκτρονικό νόμισμα σε μια περιοχή συλλογής όπου το ηλεκτρονικό νόμισμα του Βαγγέλη ξεχωρίζει από των υπολοίπων πελατών. Ο Βαγγέλης πηγαίνει στο μαγαζί του Μαρκου και καταθέτει τον υπογεγραμμένο φάκελο. Για να ολοκληρωθεί η συναλλαγή ο Μάρκος πηγαίνει στην τράπεζα για να επικυρώσει πως ο κωδικός αριθμός και η υπογραφή είναι έγκυρα. Η τράπεζα παίρνει ένα ηλεκτρινικό νόμισμα από την συλλογή και δίνει στον Μαρκο $10.

Κατ' αρχάς πρέπει να σημειώνουμε πως ο Βαγγέλης είναι ελεύθερος να κάνει όσα αντίγραφα της υπογραφής του φακέλου επιθυμεί. Η τράπεζα όμως δέχεται την συγκεκριμένη υπογραφή μόνο μια φορά, καθιστώντας έτσι άχρηστο οποιοδήποτε διπλότυπο. Δεύτερον, η τιμή ενός ηλεκτρονικού νομίσματος είναι προκαθορισμένη. Για να μπορέσει να αγοράσει κάτι ο Βαγγέλης ενδεχομένως να χρειαστεί να ζητήσει απο την τράπεζα να υπογράψει διάφορους φακέλους έτσι ώστε να μπορέσει να έχει αρκετά ηλεκτρονικά νομίσματα στην συλλογή του. Συνεπώς η συλλογή εξασφαλίζει πως η τράπεζα δεν γνωρίζει ποιός είναι και πόσο πληρώνει ο Βαγγέλης.

Το παραπάνω σχήμα θα αναπαραστήσουμε με έναν πιο φορμαλιστικό τρόπο. Υποθέτουμε πως η τράπεζα δημοσιεύει τα e, n και H σε όλους του πωλητές. Έστω $\mathcal{U}$ να είναι ένας χρήστης, $\mathcal{M}$ ένας πωλητής, και $\mathcal{B}$ η τράπεζα. Για την ανάληψη ενός ηλεκτρονικού νομίσματος από έναν λογαριασμό ακολουθείται η παρακάτω διαδικασία:

1. Ο $\mathcal{U}$ ταυτοποιείται στην τράπεζα.
2. Ο $\mathcal{U}$ διαλέγει έναν τυχαίο αριθμό $\texttt{rnd}$.
3. Ο $\mathcal{U}$ διαλέγει $r \xleftarrow{\text{r}} \mathbb{Z}_n$.
4. Ο $\mathcal{U}$ υπολογίζει $y = r^e H(\texttt{rnd}) \bmod n$ και στέλνει το y στον $\mathcal{B}$.
5. Η $\mathcal{B}$ μετακινεί την τιμή ενός ηλεκτρονικού νομίσματος από το λογαριασμό του $\mathcal{U}$ στην συλλογή του.
6. Η $\mathcal{B}$ απαντάει στον $\mathcal{U}$ με $\sigma = y^{1/e} \bmod n$.
7. Ο $\mathcal{U}$ υπολογίζει το $\mathsf{coin} = r^{-1}\sigma \bmod n$ και επιστρέφει $\langle \texttt{rnd}, \mathsf{coin} \rangle$ ως το ηλεκτρονικό του νόμισμα.

Στο Βήμα 4, το r χρησιμοποιείται για τύφλωση. Διαιρώντας με το r στο Βήμα 7, η τύφλωση αφαιρείται.

Για να πληρώσει ο $\mathcal{U}$ δίνει τα $\langle \texttt{rnd}, \mathsf{coin} \rangle$ στον $\mathcal{M}$. Πριν του παρέχει το αντίστοιχο αγαθό ο $\mathcal{M}$ ελέγχει αν $(\mathsf{coin})^e \equiv H(\texttt{rnd}) \bmod n$. Αν είναι έγκυρο ο $\mathcal{M}$ καταθέτει $\langle \texttt{rnd}, \mathsf{coin} \rangle$ στην τράπεζα μέσω ενός καναλιού που έχει πιστοποιηθεί ως προς την αυθεντικότητα του.

Όταν η τράπεζα λάβει τα $\langle \texttt{rnd}, \mathsf{coin} \rangle$, ελέγχει αν το ζευγάρι είναι έγκυρο και τότε κοιτάει αν υπάρχει στην βάση δεδομένων των ηλεκτρονικών νομισμάτων. Αν δεν έχει χρησιμοποιηθεί τότε η τράπεζα μεταφέρει την αξία του ηλεκτρονικού στον λογαριασμό του $\mathcal{M}$.

Σε αυτό το σύστημα είναι αδύνατον να συνδέσει η τράπεζα μια πληρωμή στην αντίστοιχη ανάληψη και έτσι διατηρείται η ανωνυμία.

Σχήματα ηλεκτρονικής ψηφοφορίας βασισμένο στην Τυφλή Υπογραφή Chaum

Παρόμοια κατασκευάζουμε ένα σχήμα ηλεκτρονικής ψηφοφορίας (***e-voting***) βασισμένο στις τυφλές υπογραφές Cham. Οι τρεις ομάδες που αναμιγνύονται είναι ο διαχειριστής A, οι ψηφοφόροι V_i και ο μετρητής C.

Ο διαχειριστής A ελέγχει ότι ο V_i έχει το δικαίωμα της ψήφου. Τότε, χρησιμοποιώντας κάποιον παράγοντα τύφλωσης, ο V_i τυφλώνει τον ψήφο του v_i στο v_i' και ζητά από τον A να παράξει την υπογραφή σ_i' για το v_i'. Για να υποβάλλει την ψήφο του, ο V_i λαμβάνει την υπογραφή σ_i του v_i από σ_i' αφού αφαιρεθεί η τύφλωση. Ελέγχει ότι το (v_i, σ_i) είναι ένα έγκυρο ζευγάρι ψήφου-υπογραφής χρησιμοποιώντας το κλειδί επαλήθευσης του διαχειριστή. Αν ο Verify επιστρέψει `True`, τοτε ο V_i στέλνει το (v_i, σ_i) στον μετρητή C μέσω ενός ανώνυμου καναλιού επικοινωνίας.

Ο μετρητής C χρησιμοποιεί το κλειδί επαλήθευσης του διαχειριστή για να ελέγξει την εγκυρότητα του ζευγαριού ψήφου-υπογραφής (v_i, σ_i) και τότε το προσθέτει στη λίστα του. Αφού όλοι οι ψηφοφόροι ψηφίσουν, ο C μετρά τις ψήφους, δημοσιεύει την λίστα και ανακοινώνει τα αποτελέσματα.

Το σχήμα αποτρέπει τον διαχειριστή απο το να δεί ποιος ψήφισε ποιον. Δεν αποκρύπτει όμως αυτή την πληροφορία από τον μετρητή. Για να επιλύσουμε αυτό το πρόβλημα εισάγουμε την έννοια του mix-server.

1.3 Mix-Servers

Ένας ***mix-server*** ή mixer είναι ένα δίκτυο που ανακατεύει μια ομάδα μηνυμάτων και τις περνά στον παραλήπτη σε μια αναδιαταγμένη σειρά. Ο βασικός σκοπός αυτού του μηχανισμού είναι να παρέχει ιδιωτικότητα στον αποτοστολέα, δηλαδή εξασφαλίζει πως η οντότητα που λαμβάνει το αποτέλεσμα του mix-server δεν μπορεί να διακρίνει ποιός έστειλε το μήνυμα. Υπό κάποια έννοια, ο mix-server διαχωρίζει το σύνολο των αποστολέων από έναν απλό παραλήπτη και προσπαθεί να κρύψει την σχέσεις αποστολέα-παραλήπτη.

Τα μηνύματα απο μόνο τους ενδεχομένως να φανερώνουν κάποια πληροφορία για το διάνυσμα εισόδου ή αναδιάταξης. Αν κάθε μήνημα πιστοποιείται για την αυθεντικότητα του, τότε ο στόχος του mix-server γίνεται μάλλον ακατόρθωτος. Γενικότηα αν ο παραλήπτης μπορεί να λάβει κάποια πληροφορία από το διάνυσμα εισόδου τότε ενδεχομένως να μπορέσει να διακρίνει κάποια σχέση αποστολέα-παραλήπτη. Για παράδειγμα, θεωρούμε πως όλα τα μηνύματα περιέχουν την απάντηση σε μια αίτηση και μόνο ένας αποστολέας απαντά `N` , ενώ όλοι οι άλλοι . Τότε με οποιονδήποτε τρόπο και αν ο mixer αναδιατάσσει τα μηνύματα, ο παραλήπτηςς μπορεί επιτυχώς να βρει τον αποστολέα που απάντησε `N` .

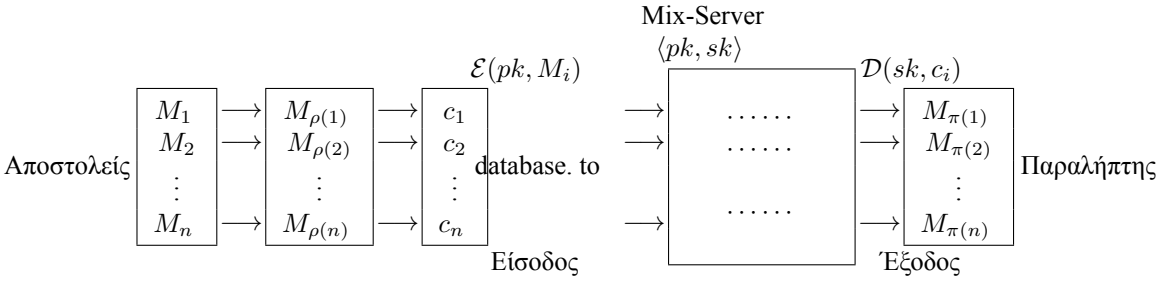
Δεδομένου ότι ο παραλήπτης (ο αντίπαλος σε αυτή την περίπτωση) παίρνει κάθε μήνυμα, ο στόχος ασφάλειας του mix-server δεν συμπεριλαμβάνει την ιδιωτικότητα των δεδομένων. Θέλουμε μεέναν μηχανισμό δημόσιου κλειδιού να καθιστά μη αναγκαία τα ιδιωτικό κανάλι μεταξύ του mix-server και των αποστολέων. Για να το πετύχουμε αυτό, χρησιμοποιούμε το σχήμα κρυπτογράφησης ElGamal.

$\mathcal{G}(1^\lambda)$:
$\langle pk, sk\rangle \leftarrow \mathcal{G}(1^\lambda)$
$x \xleftarrow{\text{r}} \mathbb{Z}_m, h = g^x \bmod p$
$pk = \langle\langle p, m, g\rangle, h\rangle$
$sk = x$

$\mathcal{E}(pk, M)$:
$M \in \langle g\rangle$
$r \xleftarrow{\text{r}} \mathbb{Z}_m$
υπολογίζουμε τα $G = g^r \bmod p$, $H = h^r M \bmod p$
επιστρέφουμε το $\langle G, H\rangle$

$\mathcal{D}(sk, g, H)$:
υπολογίζουμε τα $M = H/G^x \bmod p$
υπολογίζουμε τα M

Η Εικόνα 2 παρουσιάζει πώς ένας mix-server αλληλεπιδρά με τους αποστολείς και τους παραλήπτες. Ένας αντίπαλος μπορεί να δει τα αυθεντικά μηνύματα $\{M_i\}$, τα καλώδια "εισόδου" $\{c_i\}$ και τα καλώδια



Σχήμα 2: Ένας mix-server με τυχαίες αναδιατάξεις ρ, π στο $\{1, \ldots, n\}$.

"εξόδου" $\left\{M_{\pi(i)}\right\}$. Ο σκοπός του είναι να βρει τη σχέση μεταξύ των καλωδίων εξόδου και εισόδου, δηλαδή την αναδιάταξη ρ που αναθέτει τα μηνύματα στους χρήστες.

Εκτός από την κατασκευή ενός σχήματος δημοσίου κλειδιού έχουμε διάφορους σκοπούς όταν κατασκευάζουμε έναν αποτελεσματικό mix-server:

1. Θέλουμε να χαλαρώσουμε την υπόθεση εμπιστοσύνης μας στον mixer για να προστατευτούμε από τις περιπτώσεις που είναι κακόβουλος. Συγκεκριμένα, μπορούμε να εξασφαλίσουμε ότι ο mixer επιστρέφει πιστοποιημένα μηνύματα και όχι κάποια που έχει δημιουργήσει ο ίδιος.

2. Ο παραλήπτης θα πρέπει να μην μπορεί να κάνει συσχετίσεις μεταξύ μηνυμάτων και αποστολέων.

3. Θέλουμε να αποφύγουμε τον mix-server και τον αποστολέα από το να διαμορφώσουν μια συνομιλία αλλιώς χάνεται όλη η ιδιωτικότητα. Μια μέθοδος είναι να χρησιμοποιήσουμε πολλαπλούς mix-servers. Η ιδιωτικότητα είναι πιο πειστική όταν υπάρχουν στοιχεία ενός μη κακόβουλου mixer.

4. Τέλος, θέλουμε να έχουμε την δυνατότητα να μετατρέπουμε έναν παραλήπτη σε mix-server, δημιουργώντας έναν δεύτερο mix-server. Αν το επαναλάβουμε αυτό συνθέτουμε μια ακολουθία από servers και βελτιώνουμε την ιδιωτικότητα.

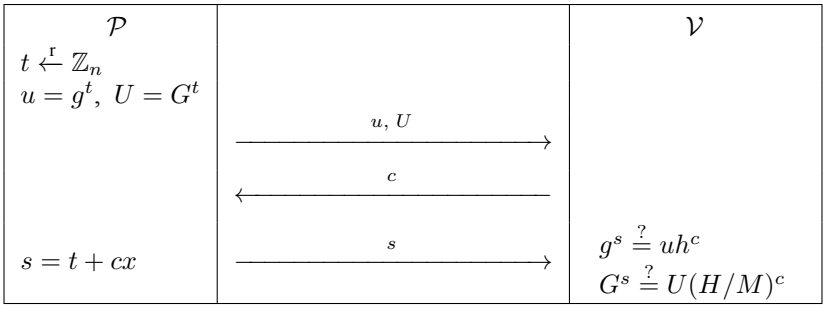
Αποδείξεις Μηδενικής Γνώσης της Σωστής Απόκρυπτογράφησης

Υποθέτουμε ότι ο παραλήπτης ζητά απο τον mixer να αποδείξει ότι αποκρυπτογραφεί σωστά το κρυπτογράφημα $c = \langle G, H\rangle$. Ο mixer θα μπορούσε να δημοσιεύσει το M και να δείξει πως γνωρίζει το μυστικό κλειδί x τέτοιο ώστε $G^x = H/M$. Αυτό ανάγεται στο να αποδείξει πως γνωρίζει τον διακριτό λογάριθμο του $\log_G(H/M)$, που δεν είναι πειστικό επιχείρημα. Όντως ένας κακόβουλος mixer θα μπορούσε να φτιάξει ένα μήνυμα διαλέγοντας ένα τυχαίο x' και θέτοντας το $M = G^{x'}/H$. Ένα ισχυρότερο επιχείρημα απαιτεί πως η ανάμειξη είναι καλη και στη συνέχεια κοιτάμε αν υπάρχει το ηλεκτρονικό νόμισμα στη βάση. Αν δεν έχει χρησιμοποιηθεί τότε η τράπεζα μετακινεί την αξία του ηλεκτρονικού νομίσματος από τη συλλογή για να αποδείξει την ισότητα των δύο λογαρίθμων $\log_G(H/M) \stackrel{?}{=} \log_g h$. Η αντίστοιχη απόδειξη μηδενικής γνώσης φαίνεται στην Εικόνα 3.

Όταν επαληθεύσουμε πως ο mix-server αποκρυπτογράφησε σωστά το κρυπτογράφημα, θέλουμε να αποδείξουμε πως κάθε μήνυμα M_i' είναι η σωστή αποκρυπτογράφηση μιας από τις εισοδους του. Δηλαδή να υπάρχει μια αναδιάταξη π τέτοια ώστε για κάθε κρυπτογράφημα $c_i = \langle G_i, H_i\rangle$ with $\mathcal{D}(sk, c_i) = M_i'$, να υπάρχει κάποιο j για το οποίο να ισχύει $M_i' = M_{\pi(j)}$. Ορίζουμε τη γλώσσα

$$\mathsf{L} = \left\{\langle c_1, \ldots, c_n, M_1', \ldots, M_n'\rangle : \text{ υπάρχει } (\pi, sk) \text{ τέτοιο ώστε } \mathcal{D}(sk, c_i) = M_{\pi(i)}' \text{ για κάθε } 1 \le i \le n\right\}.$$

Η L ανήκει στο NP. Δεδομένων π και sk, μπορούμε να επαληθεύσουμε αν μια συμβολοακολουθία είναι έγκυρη σε πολυωνυμικό χρόνο. Το πρόβλημα ανάγεται σε έναν κύκλο Hamilton. Παρότι αυτό δουλεύει, ο



Σχήμα 3: Μια απόδειξη μηδενικής γνώσης που αποδεικνύει ότι $x = \log_g h = \log_G H/M$.

γράφος που καταλήγουμε μπορεί να ειναι αρκετά μεγάλος. Μια πιο αποτελεσματική μέθοδος είαι να ορίσουμε την γλώσσα

$$\mathsf{L}' = \left\{\langle c_1, \ldots, c_n, M'_1, \ldots, M'_n\rangle : \text{ υπάρχει ένα} sk \text{ τέτοιο ώστε για κάθε } i \text{ υπάρχει ένα } j \text{ με } \mathcal{D}(sk, c_i) = M'_j\right\}$$

Η γλώσσα αυτή γράφεται επίσης ως

$$\mathsf{L}' = \bigwedge_{i=1}^{n} \left(\bigvee_{j=1}^{n} \mathcal{D}(sk, c_i) = m_j \right).$$

Σημειώνουμε ότι $\mathsf{L} \subsetneq \mathsf{L}'$ αφού η L' δεν απαιτεί ότι όλα τα πρωτότυπα μηνύματα εμφανίζονται, δηλαδή στην L', ένας mixer θα μπορούσε να θέσει $\mathcal{D}(sk, c_i) = M'_j$ για κάθε i και φιξαρισμένο j. Η L απαιτεί όλα τα κρυπτογραφημένα μηνήματα να ειναι διακριτά και συνεπώς όλα τα απλά μηνύματα να ειναι διακριτά (κανένα κρυπτογράφημα δεν μπορεί να κρυπτογραφηθεί με δύο τρόπους). Παρότι οι δύο γλώσσες είναι διαφορετικές μπορούμε να χρησιμοποιήσουμε την διάζευξη των αποδείξεων μηδενικής γνώσης στην L' για να επαληθεύσουμε μια λύση.

Θα απαιτήσουμε όλα τα απλά κείμενα να είναι διακριτά. Ανάλογα με το σύστημα, αυτό μπορεί να μην είναι κάτι φυσικό. Μπορούμε όμως να πετύχουμε την διακριτότητα εισάγοντας τυχαιότητα σε κάθε είσοδο. Για παράδειγμα, θεωρούμε την τροποποιημένη κρυπτογράφηση $\mathcal{E}(pk, M) = (g^r, h^r(M\|s))$ για κάποια τυχαία συμβολοακολουθία s. Ο παραλήπτης τώρα πρέπει να ελέγξει ότι όλα τα ανοιγμένα απλά μηνύματα έχουν s-συνιστώσες και αποδέχεται μόνο σε αυτή τη περιπτωση.Θεωρώντας πως το s είναι αρκετά μεγάλο (τουλάχιστον 128 bits), μια σύγκρουση συμβαίνει με πιθανότητα 2^{-64} βάσει του παραδούξου των γενεθλίων. Έπεται πως μια συγκρουση είναι πιθανόν να αποκαλύψει έναν κακόβουλο mixer.

Ακολουθιακή Σύνθεση n Mixers

Για επιπρόσθετη ασφάλεια, μπορούμε να χρησιμοποιήσουμε ένα σύνολο από n mixer για να κρυπτογραφήσουμε μηνύματα από επίπεδο σε επίπεδο. Αυτό εισάγει το πρόβλημα της κρυπτογράφησης ενός κρυπτογραφήματος, που είναι αναποτελεσματικό στην περίπτωση του δημοσίου κλειδιού επειδή προκαλεί την αύξηση του μεγέθους του μηνύματος. Θα παρουσιάσουμε κάποιες αποδοτικές λύσεις σε αυτό το δίλημμα.

Υποθέτουμε πως υπάρχει ένα σύνολο από n mixer. Ο mixer i έχει το δημόσιο κλειδί $pk_i = \langle\langle p, m, g\rangle, h_i\rangle$ και το μυστικό κλειδί $sk_i = x_i$. Ένα μήνυμα M κρυπτογραφείται ως $\langle g^r, (h_1 h_2 \cdots h_n)^r (M\|s)\rangle = \langle G, H\rangle$. Αν ο mixer 1 αποκρυπτογραφήσει το κρυπτογράφημα $\langle G, H\rangle$ ως $\langle G, H/G^{x_1}\rangle$, τότε το αποτέλεσμα είναι ένας έγκυρος συντελεστής του κρυπτογραφήματος για τους miaxer $2, 3, \ldots, n$. Το πρόβλημα είναι πώς φιξάραμε το πρώτο κομμάτι του κρυπτογραφήματος, κάνοντας ενδεχομένως την σχέση εισόδου-εξόδου παρατηρήσιμη

αφού ο mixer δεν αναδιατάσσει ουσιαστικά. Θα πρέπει να επιτρέψουμε στους mixer να επανακρυπτογραφήσουν κάθε μήνυμα. Η αποκρυπτογράφηση του mixer i γίνεται τότε,

$$\mathcal{D}_i(x_i, G, H) = \left\langle g^{r'} G, \left(\frac{H}{G^{x_i}} \prod_{j=i+1}^{n} h_j^{r'} \right) \right\rangle .$$

Για να αποδείξουμε πως ο mix-server ακολουθεί πιστά το πρωτόκολλο, θεωρούμε την περίπτωση των 2 mixer. Υποθέτουμε πως ο πρώτος αποκρυπτογραφεί το $\langle G, H \rangle = \langle g^r, (h_1 h_2)^r M \rangle$ και επιστρέφει το $\langle G', H' \rangle = \langle G g^{r'}, h_2^{r'} H / G^{x_1} \rangle$. Αν ο mixer μπορεί ταυτόχρονα να αποδείξει τα εξής$g^{x_1} = h_1$, $g^{r'} = G'/G$ και $h_2^{r'} / G^{x_1} = H'/H$, μπορεί επιτυχώς να αποδείξει ότι είναι πιστός.

Notes by S. Pehlivanoglu, J. Todd, & H.S. Zhou