Δημόσια Επαληθεύσιμοι Υπολογισμοί (Publicly Verifiable & Delegatable Computing) + Εισαγωγή στην Pairing-based Κρυπτογραφία

Ανδρέας Στάμος

Υπολογιστική Κρυπτογραφία — Σ.Η.Μ.Μ.Υ. – Ε.Μ.Π.

Φεβρουάριος 2025

Περιεχόμενα

Εισαγωγή

Προβλήματα τύπου Διακριτού Λογάριθμου (DLP):

- Θεμέλιο ασφάλειας των περισσότερων σύγχρονων συστημάτων (Diffie-Hellman, ElGamal κτλ.).
- CDH, DLP, ...

Πρωτόκολλο Ανταλλαγής Κλειδιού Diffie-Hellman:

- Δύο χρήστες (Alice, Bob) υπολογίζουν κοινό μυστικό g^{ab} μέσα από μη ασφαλές κανάλι.
- Επέκταση σε τρεις χρήστες (3-party DH), αλλά τότε απαιτούνται δύο γύροι...

Νέα ιδέα Joux:

- Χρήση *bilinear pairing* για 3-party κοινό μυστικό σε *έναν* γύρο.

Bilinear Pairing - βασικός ορισμός

Ορισμός

Bilinear Pairing $e : G_1 \times G_1 \to G_T$:

1. *Διγραμμικότητα (bilinearity):* $e(P+Q,R) = e(P,R)e(Q,R)$ *και* $e(P,Q+R) = e(P,Q)e(P,R)$.
2. *Μη εκφυλισιμότητα (non-degeneracy): Για* $P \in G_1$, $e(P,P) \neq 1_{G_T}$.
3. *Αποδοτικός υπολογισμός: Υπάρχει αποδοτικός αλγόριθμος (π.χ. Miller) για τον υπολογισμό* $e(P,Q)$.

Συνέπεια: Το *DDH* στην G_1 γίνεται εύκολο: από P, aP, bP, cP, ελέγχουμε αν $cP = abP$ μέσω $e(P,cP) \stackrel{?}{=} e(aP,bP)$.

Πρόβλημα BDHP

Πρόβλημα Υπολογισμού Διγραμμικού Diffie-Hellman (BDHP):

- Είσοδος: $P, aP, bP, cP \in G_1$.
- Σκοπός: Να υπολογιστεί $e(P,P)^{abc} \in G_T$.

Υπολογιστική Δυσκολία:

- $BDHP \leq_P CDH_{G_1}$: Βρίσκω abP και τότε $e(abP, cP) = e(P,P)^{abc}$.
- $BDHP \leq_P CDH_{G_2}$: Λύνω CDH στο $(e(P,P), e(aP,bP), e(P,cP)) = (e(P,P), e(P,P)^{ab}, e(P,P)^c)$

Άρα DDH στην G_1 είναι *εύκολο*, όμως CDH *πρέπει δύσκολο*.
Συνεπως απαιτείται ειδική κατασκευή.

Πρωτόκολλο Joux για 3 χρήστες (1 γύρος)

Κάθε χρήστης επιλέγει a, b, c (μυστικό) και δημοσιεύει aP, bP, cP.

Κοινό κλειδί:

$$K = e(aP, bP)^c = e(P, P)^{abc}$$

Κοινό μυστικό σε *έναν* γύρο.

Ασφάλεια: Απευθείας από υπολογιστική δυσκολία BDHP.

Υπογραφές Boneh-Lynn-Shacham (BLS)

Βασικά στοιχεία:

- Bilinear Pairing $e : G_1 \times G_1 \mapsto G_T$.
- Συνάρτηση κατακερματισμού $H : \{0,1\}^* \to G_1$.

Δημιουργία Κλειδιών:

$$\text{Ιδιωτικό κλειδί: } a \in \mathbb{Z}_p^*, \quad \text{Δημόσιο κλειδί: } A = aP$$

Υπογραφή μηνύματος m:

$$\text{Υπογραφή} : S = a\,H(m)$$

Επαλήθευση:

$$\text{Ελέγχω αν } e(P,S) \stackrel{?}{=} e(H(m),A).$$

Σχόλιο:

- Οι υπογραφές BLS είναι *πολύ σύντομες* (ίδιου μήκους με ένα στοιχείο της G_1).
- Ασφάλεια λόγω CDH στην G_1: Από $(P, aP, M = kP)$ θέλω το $S = aM = akP$.

Identity-Based Cryptography - Εισαγωγή

Κλασικό σχήμα PKI:

- Χρειάζομαι πιστοποιητικά (CA) για επιβεβαίωση δημόσιου κλειδιού.
- Υπάρχει κόστος διαχείρισης/ανανέωσης πιστοποιητικών.

Ιδέα Shamir:

- Ως δημόσιο κλειδί χρησιμοποιείται *αναγνωριστικό* (π.χ. e-mail).
- Μια *Trusted Third Party (TTP)* εκδίδει *ιδιωτικό* κλειδί βάσει ταυτότητας.
- *Χωρίς* πιστοποιητικά. Ο αποστολέας κρυπτογραφεί χρησιμοποιώντας το ID του παραλήπτη.

Υλοποίηση Boneh-Franklin

Συναρτήσεις κατακερματισμού: $H_1 : \{0,1\}^* \mapsto G_1$, $H_2 : G_T \mapsto \{0,1\}^l$.
Αρχικοποίηση:
TTP επιλέγει $t \xleftarrow{R} \mathbb{Z}_n^*$ (ιδιωτικό) και $T = tP \in G_1$ (δημόσιο).
Δημιουργία ιδιωτικού κλειδιού:

$$d_A = t\,Q_A, \quad \text{όπου } Q_A = H_1(\mathsf{ID}_A).$$

Κρυπτογράφηση προς ID_A:

$$\begin{aligned} &r \xleftarrow{\$} \mathbb{Z}_n^*, \\ &R = rP, \\ &c = m \oplus H_2(e(Q_A, T)^r). \end{aligned}$$

Κρυπτοκείμενο (R, c).
Αποκρυπτογράφηση:

$$m = c \oplus H_2(e(d_A, R)).$$

Δεν έχει ασφάλεια CCA. Απαιτείται παραλλαγή τύπου RSA OAEP.

Σχόλια για την κατασκευή pairings

Ορισμός σε *Ελλειπτικές Καμπύλες* σε πεπερασμένο πεδίο $\mathbb{F}_q$.
Ενδεικτικά το *Tate Pairing*.

- G_1 η συνήθης ομάδα, τάξης n, της ελλειπτικής καμπύλης.
- G_T η κυκλική υποομάδα τάξης n του $\mathbb{F}^*_{\mathbb{q}^{\mathbb{k}}}$.
- Αλγόριθμος Miller: $O(logn)$ επαναλήψεις με $O(1)$ πράξεις στο $\mathbb{F}_{\mathbb{q}^{\mathbb{k}}}$
- k: embedding degree, χαρακτηρίζει την καμπύλη
- k μικρό $\implies$ Index-Calculus στην $\mathbb{F}_{\mathbb{q}^{\mathbb{k}}}$.
- ANSI X9.62 δεν επέτρεπε μικρά k.
- Τυχαία καμπύλη $k \approx n$.
- Θέλουμε μικρό k για αποδοτικό pairing.

Καμπύλες **Barreto-Naehrig**. $k = 12 \implies$ Υπογραφές BLS 256 bits για 128 bits security!
$(q^k = 2^{256\cdot 12} = 2^{3072})$

Δημόσια Επαληθεύσιμος Υπολογισμός: Κίνητρο

- Ανάθεση (delegation) υπολογισμών σε τρίτους (cloud, cluster, κ.λπ.).
- Θέλουμε *επαλήθευση* (verification) του αποτελέσματος χωρίς επανεκτέλεση του πλήρους υπολογισμού.
- *Δημόσια* επαλήθευση: Δεν απαιτείται κοινό μυστικό με τον εκτελούντα. Οποιοσδήποτε μπορεί να επαληθεύσει.

Πεδίο εφαρμογής:

- Cloud Computing.
- Blockchain & Smart Contracts (off-chain υπολογισμοί, on-chain επαλήθευση).

Ορισμός Σχήματος Δημόσιας Επαλήθευσης (Publicly Verifiable Computation)

Αποτελείται από **4** αλγόριθμους (PPT):

1. $Setup(1^\kappa, f) \rightarrow (param, PK_f, EK_f)$:
2. $ProbGen(x, PK_f) \rightarrow (\sigma_x, VK_x)$:
3. $Compute(\sigma_x, EK_f) \rightarrow \sigma_y$:
4. $Verify(\sigma_y, VK_x) \rightarrow out_y$:

Όλοι μπορούν να τρέξουν τα πάντα. Δεν υπάρχουν ιδιωτικά κλειδιά ή εκ των προτέρων συνεννόηση.

Ορίζεται για οικογένεια συναρτήσεων $\mathscr{F}$.

Ορθότητα & Αξιοπιστία

Ορθότητα (Correctness):

- Αν όλα γίνουν τίμια, τότε το $Verify$ δίνει πράγματι $f(x)$ με πιθανότητα 1.

Αξιοπιστία (Soundness):

- Αν πετύχει η $Verify$ τότε το αποτέλεσμα είναι σωστό.
- Τυπικά: Πιθανότητα αποδοχής λάθος αποτελέσματος είναι αμελητέα ως προς παράμετρο ασφάλειας κ.
- Τυχαίο Πείραμα Αξιοπιστίας για αντίπαλο $\mathcal{A}$, συνάρτηση $f \in \mathcal{F}$ και παράμετρο ασφάλειας κ:
    1. $(param, PK_f, EK_f) \leftarrow Setup(1^\kappa, f)$
    2. $x \leftarrow \mathcal{A}$
    3. $(\sigma_x, VK_x) \leftarrow ProbGen(x, PK_f)$
    4. $\sigma_y \leftarrow \mathcal{A}$
    5. $out_y \leftarrow Verify(\sigma_y, VK_x)$
- **Αξιοπιστία (Soundness)** $\stackrel{def}{=}$ $\forall$ PPT $\mathcal{A}$, $\forall$ $f \in \mathcal{F}$:

$$\mathbb{P}\left[out_y \neq \perp \wedge out_y \neq f(x)\right] \leq negl(\kappa)$$

Σχήμα Δημόσια Επαλήθευσιμου Υπολογισμού για Αποτίμηση Πολυωνύμου

Στόχος:

- Επαληθεύσιμος υπολογισμός πολυωνύμου σε πεπερασμένο πέδιο $\mathbb{F}_p$.
- Επαλήθευση ταχύτερη από την πλήρη αποτίμηση.

Βασικότερη ιδέα:

- Χρησιμοποιούμε τυχαίο πολυώνυμο $B(x) = x^2 + b_0$ ($b_0 \xleftarrow{R} \mathbb{F}_p^*$).
- Γράφουμε $A(x) = Q(x)B(x) + R(x)$.
- "Καμουφλάρουμε" $\{B, Q, R\}$ ώστε όμως η αποτίμηση να είναι εφικτή (στην "καμουφλαρισμένη" έκδοση).
- Επαλήθευση: $A(x) \stackrel{?}{=} Q(x)B(x) + R(x)$

Στηρίζεται σε $\lfloor \frac{d}{2} \rfloor$-*SDH* υπόθεση ασφαλείας.

Το πρόβλημα υπολογισμού t-SDH

Έστω κυκλικές ομάδες G_1, G_2, G_T τάξης p και bilinear pairing e σε αυτές.

Είσοδος: $(g, g^a, h, h^a, \ldots, h^{a^t})$
Έξοδος: $(\beta, h^{(a+\beta)^{-1} \pmod p})$, $\beta \neq -a$

t-**Strong Diffie-Hellman** (**t-SDH**) **υπόθεση**: Κάθε PPT αλγόριθμος επιτυγχάνει με πιθανότητα αμελητέα ως προς παράμετρο ασφάλειας κ.

Setup και ProbGen

Setup$(1^\kappa, A)$:

1. Επιλέγουμε *φιλικές προς pairing* ομάδες G_1, G_2, G_T τάξης p, bilinear pairing $e(\cdot,\cdot)$.
2. Επιλέγουμε τυχαίο $b_0 \xleftarrow{R} \mathbb{F}_p^*$, ορίζουμε $B(x) = x^2 + b_0$.
3. Βρίσκουμε $Q(x), R(x)$ με $A(x) = Q(x)B(x) + R(x)$.
4. Υπολογίζουμε $\tilde{b_0} = g^{b_0}$, $\tilde{q_i} = h^{q_i}$, $\tilde{r_0} = h^{r_0}$, $\tilde{r_1} = h^{r_1}$.
5. $PK_A = (\tilde{b_0}, \tilde{r_0}, \tilde{r_1}), \quad EK_A = (A, \tilde{q_0}, \dots, \tilde{q_{d-2}})$.

ProbGen(x, PK_A):

1. $\sigma_x = x$.
2. $VK_x = (VK_{x,B}, VK_{x,R}) = (\tilde{b_0} g^{x^2}, \tilde{r_1}^x \tilde{r_0})$.

Compute και Verify

Compute(σ_x, EK_A):

- (με $\sigma_x = x$) $y = A(x)$
- $\pi = \prod_{i=0}^{d-2} \tilde{q}_i^{\,x^i} = h^{Q(x)}$ (πιστοποιητικό).
- Επιστρέφει $\sigma_y = (y, \pi)$.

Verify(σ_y, VK_x):

- Δεδομένου $\sigma_y = (y, \pi)$, και $VK_x = (VK_{x,B}, VK_{x,R})$,
- Ελέγχει αν

$$e(g, h^y) \stackrel{?}{=} e(VK_{x,B}, \pi) \cdot e(g, VK_{x,R}).$$

- Αν ισχύει, τότε αποδέχεται και επιστρέφει y, αλλιώς $\perp$.

Ορθότητα (Correctness)

$$\pi = h^{Q(x)}, \quad y = A(x) = Q(x)\,B(x) + R(x).$$

Ισχύει:

$$e(g, h^y) = e(g,h)^{\,Q(x)\,B(x)+R(x)} = e\big(g^{B(x)},\, h^{Q(x)}\big)\,\cdot\,e\big(g,\, h^{R(x)}\big)$$

$$= e\big(g^{x^2+b_0},\, \pi\big)\,\cdot\,e\big(g,\, h^{r_1 x + r_0}\big) = e\big(\tilde{b_0}\, g^{x^2},\, \pi\big)\,\cdot\,e\big(g,\, \tilde{r_1}^{\,x}\, \tilde{r_0}\big),$$

που ισούται με

$$e(VK_{x,B},\, \pi)\,\cdot\,e(g,\, VK_{x,R}).$$

Άρα το $Verify$ επιστρέφει όντως $y = A(x)$.

Αξιοπιστία (Soundness)

- Αν κάποιος PPT αντίπαλος $\mathcal{A}$ κερδίσει το Τυχαίο Πείραμα Αξιοπιστίας για κάποια συνάρτηση $f \in \mathcal{F}$ τότε μπορούμε να κατασκευάσουμε PPT αλγόριθμο $\mathcal{B}$ που λύνει το $\lfloor \frac{d}{2} \rfloor$-SDH.
- **Κεντρική Ιδέα**:
  - Προσομοιώνουμε την $Setup$ με $b_0 = v$
    (v: ο "εκθέτης" που εμφανίζεται στο στιγμιότυπο $\lfloor \frac{d}{2} \rfloor$-SDH)
  - Εκτελούμε $\mathcal{A}$ και παίρνουμε (y, π) που γίνεται δεκτό από την $Verify$.
  - Εκτελούμε και την $Compute$ και παίρνουμε $(A(x), \pi_*)$ με $y \neq A(x)$.
  - Επεξεργαζόμαστε τις σχέσεις αποδοχής της $Verify$ στα δύο ζεύγη και τελικά βρίσκουμε (κατόπιν πράξεων):

$$h^{(x^2+v)^{-1}} = \left(\pi\pi_*^{-1}\right)^{(y-A(x))^{-1}}$$

  - Άρα λύση: $\left(x^2, \left(\pi\pi_*^{-1}\right)^{(y-A(x))^{-1}}\right)$

Συνολική Πολυπλοκότητα

Setup:

- $O(d)$ υψώσεις/πολλαπλασιασμοί στις αντίστοιχες ομάδες, + επιλογή b_0.

ProbGen:

- 1 ύψωση σε δύναμη και 1 πολλαπλ. σε G_1 και G_2.

Compute:

- $O(d)$ πράξεις στο $\mathbb{F}_p$ (κανόνας Horner για $A(x)$).
- $d-1$ υψώσεις σε δύναμη σε G_2 για το π.

Verify:

- 2 bilinear pairings και λίγες πράξεις σε G_2.
- Συνήθως πολύ πιο γρήγορο από πλήρη επανυπολογισμό του $A(x)$.

Σχήμα Δημόσια Επαληθεύσιμου Υπολογισμού για $\vec{y} = M\vec{x}$

Στόχος:

- Πίνακας $M \in \mathbb{F}_p$ σταθερός για όλες τις αποτιμήσεις.
- Σκοπός ο υπολογισμός του $\vec{y} = M\vec{x}$

Χρήση Διανυσματικού και Μητρωικού Συμβολισμού για κομψότητα και απλότητα

Setup

Είσοδος: παράμετρος ασφαλείας κ, πίνακας $M \in \mathbb{F}_p^{n \times m}$.

1. Επιλέγουμε δύο κυκλικές ομάδες G_1, G_2 τάξης p, και μία τρίτη ομάδα G_T με bilinear pairing
$$e : G_1 \times G_2 \to G_T.$$
2. Υποθέτουμε ότι για αυτές τις ομάδες ισχύει η υπόθεση *co-CDH*.
3. Επιλέγουμε γεννήτορα $h \in G_2$ και $\delta \xleftarrow{R} \mathbb{F}_p^*$. Θέτουμε $\tilde{h} = h^{\delta}$.
4. Επιλέγουμε $\vec{\lambda} \xleftarrow{R} {\mathbb{F}_p^*}^n$ και ορίζουμε $\vec{g} = g^{\vec{\lambda}}$ (όπου g γεννήτορας της G_1).
5. Επιλέγουμε $R \xleftarrow{R} \mathbb{F}_p^{n \times m}$ και υπολογίζουμε
$$N_{ij} \ = \ g_i^{\,\delta\, M_{ij} \,+\, R_{ij}}.$$
6. Υπολογίζουμε $PK_j = e\Big(\prod_{i=1}^{n} g_i^{R_{ij}},\, h\Big)$ (ή μητρωικά $\vec{PK} = e(g^{R^T \vec{\lambda}},\, h)$).

Έξοδος:

$$param \ = \ (p, G_1, G_2, G_T, e, \vec{g}, h, \tilde{h}), \quad EK_M \ = \ (M, N), \quad PK_M \ = \ \{PK_j\}.$$

ProbGen

Είσοδος: $\vec{x} \in \mathbb{F}_p^m$ (η είσοδος που θέλουμε να υπολογιστεί), και το δημόσιο κλειδί PK_M.

Περιγραφή

1. Υπολογίζουμε
$$VK_x = \prod_{j=1}^{m} PK_j^{x_j} = \prod_{j=1}^{m} e\left(g^{(R^T\vec{\lambda})_j}, h\right)^{x_j}.$$
2. Θέτουμε $\sigma_x = \vec{x}$.

Έξοδος:
$$\sigma_x, \quad VK_x.$$

Compute

Είσοδος: $\sigma_x = \vec{x}$ και $EK_M = (M, N)$.

Περιγραφή

1. Υπολογίζουμε την κανονική έξοδο $\vec{y} = M\vec{x}$.
2. Υπολογίζουμε το πιστοποιητικό

$$\Pi = \prod_{i=1}^{n} \prod_{j=1}^{m} N_{ij}^{x_j}.$$

3. Θέτουμε $\sigma_y = (\vec{y}, \Pi)$ και το επιστρέφουμε.

Έξοδος: $\sigma_y = (\vec{y}, \Pi)$.

Verify

Είσοδος: $\sigma_y = (\vec{y}, \Pi)$ και VK_x.

Περιγραφή

Ελέγχουμε αν $e(\Pi,\ h) \stackrel{?}{=} e\Big(\prod_{i=1}^{n} g_i^{y_i},\ \tilde{h}\Big) \cdot VK_x.$

Αν ισχύει, επιστρέφουμε $\vec{y}$, αλλιώς $\perp$.

Ορθότητα (Correctness)

Βασική Ιδέα (πράξεις...)

$$\Pi \;=\; \prod_{i=1}^{n}\prod_{j=1}^{m} N_{ij}^{x_j} \;=\; \cdots \;=\; g^{((\delta M+R)\,\vec{x})\,\vec{\lambda}} \quad\Longrightarrow\quad e(\Pi,\; h) \;=\; e\Big(g^{((\delta M+R)\,\vec{x})\,\vec{\lambda}},\; h\Big).$$

Επίσης:

$$VK_x \;=\; \cdots \;=\; e(g^{(R\vec{x})\vec{\lambda}},\, h), \quad e\Big(\prod_{i=1}^{n} g_i^{y_i},\, \tilde{h}\Big) \;=\; e(g^{(\delta\,\vec{\lambda})\vec{y}},\, h)$$

Οπότε:

$$e(\Pi, h) \;=\; e\Big(g^{(\delta\,\vec{y}+R\,\vec{x})\,\vec{\lambda}},\, h\Big) \;=\; e\Big(\prod_{i=1}^{n} g_i^{y_i},\, \tilde{h}\Big)\cdot VK_x.$$

Άρα η $Verify$ επιστρέφει $\vec{y} = M\vec{x}$.

Το πρόβλημα υπολογισμού co-CDH

Έστω κυκλικές ομάδες G_1, G_2, G_T τάξης p και bilinear pairing e σε αυτές.

Είσοδος: $(g', {g'}^{a}) \in G_1$, $(h', {h'}^{b}) \in G_2$
Έξοδος: ${g'}^{ab}$

co-**Computational Diffie-Hellman (**co-**SDH) υπόθεση**: Κάθε PPT αλγόριθμος επιτυγχάνει με πιθανότητα αμελητέα ως προς παράμετρο ασφάλειας κ.

Αξιοπιστία (Soundness)

- Υποθέτουμε ότι υπάρχει PPT αντίπαλος $\mathcal{A}$ που κερδίζει Τυχαίο Πείραμα Αξιοπιστίας για πίνακα M.
- Φτιάχνουμε PPT $\mathcal{B}$ που λύνει το *co-CDH*.
- Δίνονται: $(g', {g'}^a)$, $(h, {h'}^b)$. Ζητείται το ${g'}^{ab}$.
- Προσομοιώνω την $Setup$ με $g = {g'}^a$, $h = h'$, $\delta = \delta'\beta$ όπου $\delta' \xleftarrow{R} \mathbb{F}_p^*$.
- Τυχαίος πίνακας $N \xleftarrow{R} G_1^{n\times m}$ (έστω $N_{ij} = g^{n_{ij}}$).
- Υπολογίζουμε:

$$PK_j = \frac{e\left(\prod_{i=1}^{n} N_{ij}, h\right)}{e\left(\prod_{i=1}^{n} g_i^{M_{ij}}, \tilde{h}\right), \tilde{h}}$$

- Ισοδύναμα πίνακας $R_{ij} = n_{ij}\lambda_i^{-1} - \delta M_{ij} \implies$ ίδια κατανομή πιθανότητας με $Setup$.
- Λαμβάνουμε $(\vec{y}, \Pi)$, $\vec{y} \neq \vec{y_*} = M\vec{x}$.
- Βρίσκουμε και Π_* μέσω Compute.
- Κατόπιν πράξεων:

$$\left(\Pi\Pi_*^{-1}\right)^{\delta'^{-1}\left((\vec{y}-\vec{y_*})\vec{\lambda}\right)^{-1}} = {g'}^{ab}$$

Βοηθητικό Θεώρημα

Χρειάζεται να ισχύει $(\vec{y} - \vec{y}_*)\vec{\lambda} \neq 0$. Ισχύει με αμελητέα πιθανότητα, αλλιώς λύνω $DLog$ στην G_1: Δίνονται g, g^a, να βρεθεί το a.

- $k \xleftarrow{R} \mathbb{Z}_n$.
- $\lambda_k = a$, λ_i τυχαία ως πριν.
- Στο παίγνιο DLog το a τυχαίο $\implies \vec{\lambda}$ ανεξάρτητο k.
- $y \neq y_* \implies y_{k'} \neq y_{*k'}$.
- $k = k'$ με $\frac{1}{n}$ πιθανότητα – μη αμελητέα!
- Υπολογίζω:

$$a = (y_k - y_{*k})^{-1} \sum_{i=1, i\neq k}^{n} (y_{*i} - y_i)\lambda_i$$

Βελτίωση της επίδοσης – *δική μου συνεισφορά*

Χρήση διανύσματος $\vec{N}$ αντί πίνακα N:

$$\vec{N} = g^{(\delta M+R)^T\vec{\lambda}}$$

Υπολογισμός Π ως:

$$\Pi = \prod_{i=1}^{n} \left(g^{(\delta \mathrm{M}+R)^T\vec{\lambda}}\right)_i^{x_i}$$

Ισχύει: $N_j = \prod_{i=1}^{n} N_{ij}$. Συνεπώς ισχύει η απόδειξη αξιοπιστίας.
Μείωση επιβάρυνσης από $\Theta(nm)$ σε $\Theta(m)$.

Βιβλιογραφία I

[1] Kaoutar Elkhiyaoui et al. "Efficient Techniques for Publicly Verifiable Delegation of Computation". In: *Proceedings of the 11th ACM on Asia Conference on Computer and Communications Security*. ASIA CCS '16. ACM, May 2016. DOI: 10.1145/2897845.2897910. URL: http://dx.doi.org/10.1145/2897845.2897910.

[2] Alfred Menezes. *An introduction to pairing-based cryptography*. 2009. DOI: 10.1090/conm/477/09303. URL: http://dx.doi.org/10.1090/conm/477/09303.

[3] Antoine Joux. "A One Round Protocol for Tripartite Diffie–Hellman". In: *Journal of Cryptology* 17.4 (June 2004), pp. 263–276. ISSN: 1432-1378. DOI: 10.1007/s00145-004-0312-y. URL: http://dx.doi.org/10.1007/s00145-004-0312-y.

[4] Dan Boneh, Ben Lynn, and Hovav Shacham. "Short Signatures from the Weil Pairing". In: *Advances in Cryptology — ASIACRYPT 2001*. Springer Berlin Heidelberg, 2001, pp. 514–532. ISBN: 9783540456827. DOI: 10.1007/3-540-45682-1_30. URL: http://dx.doi.org/10.1007/3-540-45682-1_30.

Βιβλιογραφία II

[5] Dan Boneh and Matthew Franklin. "Identity-Based Encryption from the Weil Pairing". In: *SIAM Journal on Computing* 32.3 (Jan. 2003), pp. 586–615. ISSN: 1095-7111. DOI: 10.1137/s0097539701398521. URL: http://dx.doi.org/10.1137/S0097539701398521.

[6] Paulo S. L. M. Barreto and Michael Naehrig. "Pairing-Friendly Elliptic Curves of Prime Order". In: *Selected Areas in Cryptography*. Springer Berlin Heidelberg, 2006, pp. 319–331. ISBN: 9783540331094. DOI: 10.1007/11693383_22. URL: http://dx.doi.org/10.1007/11693383_22.