ΠΑΝΕΠΙΣΤΗΜΙΟ ΠΕΛΟΠΟΝΝΗΣΟΥ
Σχολη Θετικων Επιστημων
Τμήμα Επιστήμης και Τεχνολογίας Υπολογιστών

Διπλωματική Εργασία

Κώδικας RSA: Υλοποίηση και Επιθέσεις

Σχίζας Αθανάσιος, Α.Μ. 2009030
e-gnosi@hol.gr

Επιβλέπων Καθηγητής

Νικόλαος Κολοκοτρώνης, *Λέκτορας*

Ιανουάριος 2012

Πίνακας Περιεχομένων

Κατάλογος Σχημάτων

Κατάλογος Πινάκων

Κατάλογος Συντομογραφιών

RSA	Rivest, Shamir, Adleman
CRT	Chinese Remainder Theorem Κινέζικο Θεώρημα Υπολοίπων
LLL	Lenstra, Lenstra (Jr) και Lovasz (αναφέρεται στον αλγόριθμο μείωσης διαστάσεων πίνακα)

Κατάλογος Συμβολισμών

p, q	Πρώτοι ακέραιοι
e	Δημόσιος εκθέτης στον RSA
d	Ιδιωτικός εκθέτης στον RSA
N=p*q	Γινόμενο πρώτων
φ(N)	Συνάρτηση του Euler.
$\mathbb{Z}$	Σύνολο ακεραίων αριθμών
n	Πλήθος από bit (κλειδιού)
M	Αρχικό (μη κρυπτογραφημένο) κείμενο
C	Κρυπτογράφημα
P	Παραλήπτης μηνύματος
Μ.Κ.Π	Μέγιστο Κοινό Πολλαπλάσιο
O(k)	Τάξη μεγέθους k
w	Διαστάσεις Πίνακα
LLL	Αλγόριθμος Lovasz, Lenstra & Lenstra
L	Πίνακας διανυσμάτων

1 Πρόλογος

Η πτυχιακή αυτή εργασία αποτελεί μια βασική προσέγγιση στον κόσμο της κρυπτογραφίας και ειδικότερα της ασύμμετρης κρυπτογραφίας. Αποτελείται από δύο μέρη: Το Α' μέρος ξεκινάει με τις βασικές έννοιες και ορολογίες τις κρυπτογραφίας και συνεχίζει με τα είδη κρυπτογραφίας που υπάρχουν. Ακολουθεί αναφορά στον αλγόριθμο RSA που είναι και στο επίκεντρο της παρούσας εργασίας. Στο Β' μέρος αναλύονται συγκεκριμένες επιθέσεις που μπορεί να δεχτεί και παρατίθενται πειραματικά αποτελέσματα εξομοιώσεων για :

- Επιθέσεις σφαλμάτων στα δημόσια κλειδιά RSA
- Επιθέσεις σφαλμάτων κατά υπογραφών EMV

2 Εισαγωγικές Έννοιες

Η λέξη *κρυπτολογία* αποτελείται από την Ελληνική λέξη κρύπτος που σημαίνει κρυφός και την λέξη λόγος. Είναι ο τομέας που ασχολείται με την μελέτη της ασφαλούς επικοινωνίας. Ο κύριος στόχος είναι να επικοινωνούν δυο ή περισσότεροι χρήστες, μέλη, χωρίς κάποιος άλλος μη εξουσιοδοτημένος χρήστης να μπορεί να διαβάσει την πληροφορία που ανταλλάσσουν οι πρώτοι χρήστες.

Η κρυπτολογία χωρίζεται σε δυο επιμέρους ενότητες:

- Την *Κρυπτογραφία*, που ασχολείται με τους μαθηματικούς μετασχηματισμούς προκειμένου να εξασφαλίσει την μυστικότητα της πληροφορίας.
- Την *Κρυπτανάλυση*, που ασχολείται με την ανάλυση και την διάσπαση των κρυπτοσυστημάτων.

Η κρυπτογραφία παρέχει τέσσερις βασικές λειτουργίες:

1. *Εμπιστευτικότητα:* Η πληροφορία προς μετάδοση είναι προσβάσιμη μόνο από τα εξουσιοδοτημένα μέλη και ακατανόητη από οποιονδήποτε άλλο.
2. *Ακεραιότητα:* Η πληροφορία μπορεί να αλλοιωθεί μόνο από τα εξουσιοδοτημένα μέλη και δεν μπορεί να αλλοιώνεται χωρίς αυτό να γίνει αντιληπτό.
3. *Μη απάρνηση:* Ο αποστολέας ή ο παραλήπτης της πληροφορίας δεν μπορεί να αρνηθεί την αυθεντικότητα της μετάδοσης ή της δημιουργίας της.
4. *Πιστοποίηση:* Ο αποστολέας και ο παραλήπτης μπορούν να εξακριβώσουν τις ταυτότητες τους καθώς και την πηγή και τον προορισμό της πληροφορίας με διαβεβαίωση ότι οι ταυτότητές τους δεν είναι πλαστές.

Ιστορικά η κρυπτογραφία χρησιμοποιήθηκε για την κρυπτογράφηση μηνυμάτων, δηλαδή, την μετατροπή της πληροφορίας από μια κανονική κατανοητή μορφή σε ένα γρίφο, που χωρίς τη γνώση του κρυφού μετασχηματισμού παρέμενε ακατανόητος. Το κύριο χαρακτηριστικό αυτών των παλαιότερων μορφών κρυπτογράφησης ήταν ότι η επεξεργασία γινόταν πάνω στη δομή της γλώσσας.

Στις νεότερες μορφές η κρυπτογραφία κάνει χρήση του αριθμητικού ισοδύναμου και δίνεται πλέον έμφαση σε διάφορα πεδία των μαθηματικών όπως: διακριτά μαθηματικά, θεωρία αριθμών, θεωρία πληροφορίας, υπολογιστική πολυπλοκότητα, στατιστική και συνδυαστική ανάλυση.

2.1 Ορισμοί

Ο αντικειμενικός στόχος της κρυπτογραφίας είναι να δώσει την δυνατότητα σε δυο πρόσωπα – χρήστες να επικοινωνήσουν μέσα από ένα μη ασφαλές κανάλι με τέτοιο τρόπο ώστε ένας τρίτος μη εξουσιοδοτημένος χρήστης (ο επιτιθέμενος) να μην μπορεί να παρεμβληθεί στην επικοινωνία και να κατανοήσει το περιεχόμενο των μηνυμάτων.

Κρυπτογράφηση (encryption): ονομάζεται η διαδικασία μετασχηματισμού ενός μηνύματος σε μια ακατανόητη μορφή με τη χρήση κάποιου κρυπτογραφικού αλγόριθμου ούτως ώστε να μην μπορεί να διαβαστεί από κανέναν εκτός του παραλήπτη για τον οποίο έχει δημιουργηθεί.

Αποκρυπτογράφηση (decryption): ονομάζεται η αντίστροφη διαδικασία.

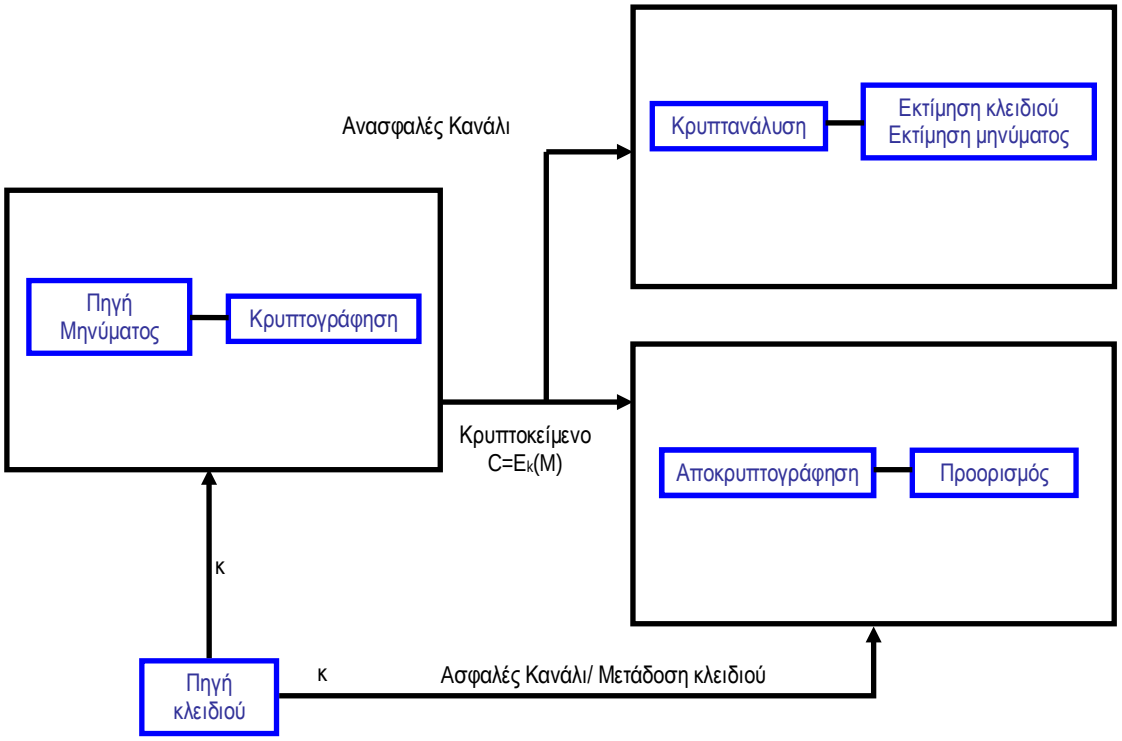
Κρυπτογραφικός αλγόριθμος (cipher): είναι η μέθοδος μετασχηματισμού δεδομένων σε μια μορφή που να μην επιτρέπει την αποκάλυψη των περιεχομένων τους από μη εξουσιοδοτημένα μέρη. Ουσιαστικά ο κρυπτογραφικός αλγόριθμος είναι μια πολύπλοκη μαθηματική συνάρτηση.

Αρχικό κείμενο (plaintext): είναι το μήνυμα το οποίο αποτελεί την είσοδο σε μια διεργασία κρυπτογράφησης.

Κλειδί (Key): είναι ένας αριθμός από bit που χρησιμοποιούνται ως είσοδος στη συνάρτηση κρυπτογράφησης.

Κρυπτογραφημένο κείμενο (ciphertext): είναι το αποτέλεσμα της εφαρμογής ενός κρυπτογραφικού αλγόριθμου πάνω στο αρχικό κείμενο.

Η κρυπτογράφηση και η αποκρυπτογράφηση ενός μηνύματος γίνεται με τη βοήθεια ενός κρυπτοσυστήματος και ενός μυστικού κλειδιού. Ο αλγόριθμος κρυπτογράφησης είναι συνήθως γνωστός, και κατά συνέπεια η εμπιστευτικότητα του κρυπτογραφημένου μηνύματος που μεταδίδεται βασίζεται ως επί το πλείστον στη μυστικότητα του κλειδιού κρυπτογράφησης. Το μέγεθος του κλειδιού μετριέται σε bits. Όσο μεγαλύτερος είναι ο αριθμός των bits του κλειδιού τόσο δυσκολότερα μπορεί να αποκρυπτογραφηθεί το κρυπτοκείμενο από τον επιτιθέμενο, εφόσον υπάρχουν περισσότερα πιθανά κλειδιά που πρέπει να δοκιμαστούν Βέβαια ο κάθε αλγόριθμος κρυπτογράφησης μπορεί να πετύχει ένα ισχυρό επίπεδο κρυπτογράφησης με διαφορετικό μέγεθος κλειδιού.



Σχήμα 1 μοντέλο τυπικού κρυπτοσυστήματος

Το σύνολο των διαδικασιών κρυπτογράφησης και αποκρυπτογράφησης αποτελείται από:

- Το P, που είναι ο χώρος όλων των δυνατών μηνυμάτων, ή αλλιώς ανοικτών κειμένων.
- Το C, που είναι ο χώρος όλων των δυνατών κρυπτογραφημένων μηνυμάτων.

- Το K, που είναι ο χώρος όλων των δυνατών κλειδιών, ή αλλιώς κλειδοχώρος.
- Το E, που είναι η συνάρτηση της κρυπτογράφησης και μετατροπής μηνυμάτων σε κρυπτογραφημένα.
- Το D, που είναι η αντίστροφη συνάρτηση της παραπάνω συνάρτησης που μας δίνει την αποκρυπτογράφηση.

Η συνάρτηση κρυπτογράφησης E δέχεται δυο παραμέτρους, μέσα από το χώρο P και τον χώρο K και παράγει μια ακολουθία που ανήκει στον χώρο C.Ek(P)=C. Η συνάρτηση αποκρυπτογράφησης D δέχεται δυο παραμέτρους, τον χώρο C και τον χώρο K και παράγει μια ακολουθία που ανήκει στο χώρο P. Dk(C)=P.

Το σύστημα στο Σχήμα 1 λειτουργεί με τον ακόλουθο τρόπο:

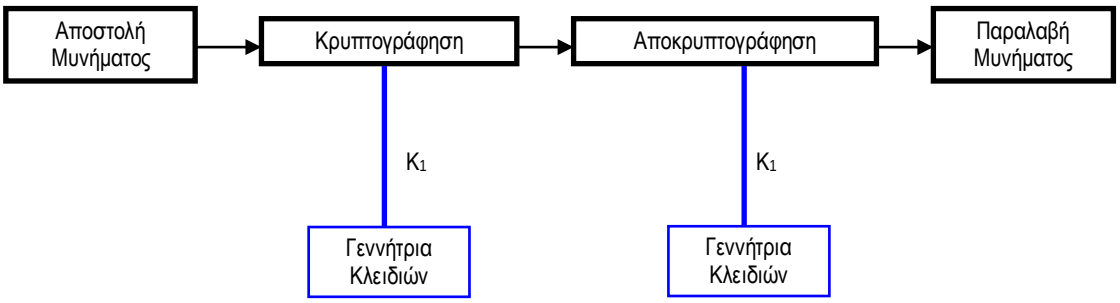
- Ο αποστολέας επιλέγει ένα κλειδί k=[k1,k2,k3,…,kn] μήκους n όπου n>=1, από τον χώρο κλειδιών με τυχαίο τρόπο όπου τα n στοιχεία του K είναι στοιχεία από ένα πεπερασμένο αλφάβητο.
- Αποστέλλει την ακολουθία στον παραλήπτη μέσα από ένα ασφαλές κανάλι.
- Η πηγή δημιουργεί ένα μήνυμα από τον χώρο μηνυμάτων MЄP όπου m=[m1,m2,m3,…,mi] ;και i≥1.
- Η συνάρτηση κρυπτογράφησης παίρνει τις δυο εισόδους και παράγει μια ακολουθία συμβόλων C=[c1,c2,c3,…cj] όπου j≥1 και η ακολουθία αυτή αποστέλλεται διαμέσου ενός μη ασφαλούς καναλιού.
- Η συνάρτηση αποκρυπτογράφησης παίρνει ως όρισμα της 2 τιμές, ακολουθεί την αντίστροφη διαδικασία και αποδίδει το μήνυμα M=[m1,m2,m3,…,mi]

2.2 Είδη Κρυπτοσυστημάτων

Τα κρυπτοσυστήματα χωρίζονται σε δυο μεγάλες κατηγορίες τα *κλασσικά* και τα *μοντέρνα* κρυπτοσυστήματα. Τα κλασσικά περιλαμβάνουν αλγορίθμους, όπως η αναδιάταξη, η μετατόπιση (κρυπταλγόριθμος του Caesar), ο γραμμικός, ο Vigenere, ο Vernam, και ο κρυπταλγόριθμος του Hill. Τα μοντέρνα κρυπτοσυστήματα διακρίνονται με τη σειρά τους σε δυο κατηγορίες: *συμμετρικά* και *ασύμμετρα*.

2.2.1 Συμμετρικά κρυπτοσυστήματα

Το συμμετρικό κρυπτοσύστημα είναι το σύστημα εκείνο το οποίο χρησιμοποιεί για την κρυπτογράφηση και την αποκρυπτογράφηση ένα κοινό κλειδί.

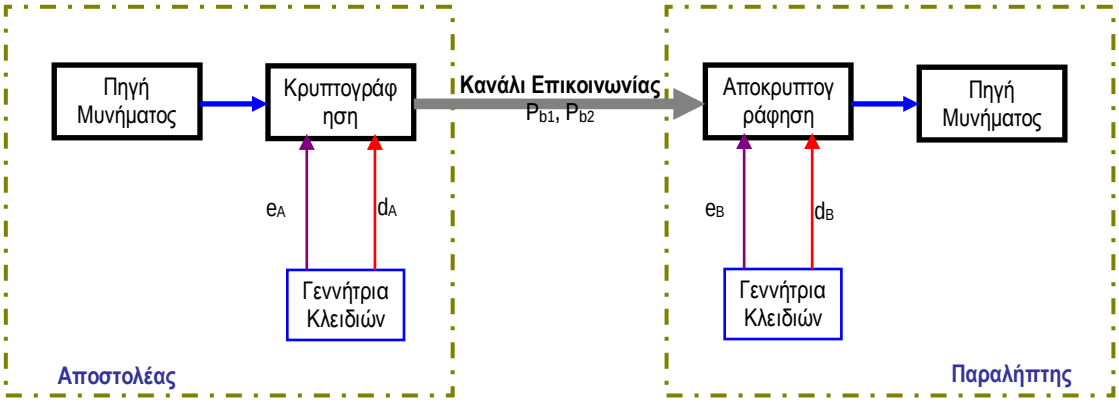


Σχήμα 2 μοντέλο συμμετρικού κρυπτοσυστήματος

Η ασφάλεια του αλγορίθμου βασίζεται μόνο στην μυστικότητα του κλειδιού. Για να το πετύχουν αυτό τα συμμετρικά κρυπτοσυστήματα ανταλλάσσουν το κλειδί μέσα από ένα ασφαλές κανάλι επικοινωνίας, το οποίο καθιστά τελικά δύσκολη την επικοινωνία (ανταλλαγή πληροφορίας) μεταξύ απομακρυσμένων μελών.

2.2.2 Ασύμμετρα κρυπτοσυστήματα

Το ασύμμετρο κρυπτοσύστημα έχει ως χαρακτηριστικό του την ύπαρξη δυο κλειδιών, ένα ιδιωτικό και ένα δημόσιο. Το δημόσιο είναι διαθέσιμο σε όλους ενώ το ιδιωτικό είναι γνωστό μόνο στον κάτοχό του. Για να κρυπτογραφηθεί ένα μήνυμα χρησιμοποιείται το δημόσιο κλειδί, ενώ για να αποκρυπτογραφηθεί χρησιμοποιείται το ιδιωτικό κλειδί.



Σχήμα 3 μοντέλο ασύμμετρου κρυπτοσυστήματος

Τα στάδια της επικοινωνίας είναι τα ακόλουθα

- Η γεννήτρια κλειδιών του Α παράγει δυο ζεύγη κλειδιών eA, dA.
- Η γεννήτρια κλειδιών του Β παράγει δυο ζεύγη κλειδιών eB, dB.
- Ο Α και ο Β ανταλλάσσουν τα δημόσια κλειδιά, Pd1, και Pd2.
- Ο Α δημιουργεί ένα μήνυμα M=[m1,m2,m3,…,mι] όπου τα σύμβολα m ανήκουν στον χώρο των μηνυμάτων.
- Κρυπτογραφεί το μήνυμα με το δημόσιο κλειδί του Β (eB) η παραγόμενη συμβολοσειρά C=[m1,m2,m3,…,mj] αποστέλλεται στο χρήστη Β.
- Ο χρήστης Β λαμβάνει το κρυπτογραφημένο μήνυμα C και στη συνέχεια με το ιδιωτικό του κλειδί αποκρυπτογραφεί το κρυπτογραφημένο μήνυμα και παράγει το αρχικό, M=[m1,m2,m3,…,mi].

2.3 Το σύστημα διαμοίρασης δημοσίου κλειδιού

Το 1976, οι Diffie και Hellman δημοσίευσαν την εργασία που σήμανε την έναρξη της κρυπτογραφίας δημοσίου κλειδιού και ταυτόχρονα οδήγησε σε πληθώρα δημοσίων αναζητήσεων στο πεδίο της κρυπτογραφίας. Η εργασία τους περιείχε ένα καταπληκτικό συμπέρασμα: «είναι δυνατό να σχεδιαστεί ένα ουσιαστικά ασφαλές κρυπτοσύστημα το οποίο δε χρειάζεται την αποστολή μυστικού κλειδιού». Οι Diffie και Hellman εισήγαγαν τη μονόδρομη συνάρτηση –καταπακτή $f(x)$ η οποία είναι εύκολα υπολογίσιμη για κάθε όρισμα x του πεδίου ορισμού, αλλά για κάποιο τυχαία επιλεγμένο y του πεδίου

τιμών είναι υπολογιστικά δύσκολο να βρεθεί x.Η χρήση τέτοιων συναρτήσεων για να προστατευθεί η πρόσβαση σε υπολογιστικά συστήματα μέσω μονόδρομων συνθηματικών είναι πλέον ευρύτατα διαδεδομένη.

Η μονόδρομη συνάρτηση – καταπακτή είναι στη πραγματικότητα μια ολόκληρη οικογένεια αντιστρέψιμων συναρτήσεων f_z όπου για δεδομένο ζ είναι δυνατό να βρεθούν αλγόριθμοι $\mathbf{E_z}$ και $\mathbf{D_z}$ που καθιστούν εύκολο τον υπολογισμό της τιμής της $f_z(x)$ για κάθε x του πεδίου ορισμού καθώς και τον υπολογισμό της τιμής της $f_z(y)$ για κάθε τιμή του y από το πεδίο τιμών. Παρά όλα αυτά είναι υπολογιστικά αδύνατο για όλες τις πιθανές τιμές της παραμέτρου z και όλες τις τιμές της y από το πεδίο τιμών της f_z να βρεθεί η $f_z^{-1}(y)$. Οι Diffie και Hellman πρότειναν τη μονόδρομη συνάρτηση διακριτής ύψωσης σε εκθέτη:

$$f(x) = a^x (\mathrm{mod}\ p)$$

όπου το x είναι ακέραιος αριθμός με $1 \le x \le p-1$ και το p είναι πρώτος αριθμός μήκους k bit. Ο αριθμός $a < p$ επιλέγεται ώστε η υψωμένη δύναμη του modulo p να αποδίδει ένα ταξινομημένο σύνολο αριθμών $\{a^1, a^2, \ldots, a^{p-1}\}$ που είναι μια μετάθεση του αριθμητικού συνόλου $\{1, 2, \ldots, p-1\}$.

Η παραπάνω συνάρτηση είναι πολύ εύκολο να υπολογισθεί ακόμα και για πολύ μεγάλο modulo p (για παράδειγμα όταν k=1024bits) για δοσμένο x. Η διαδικασία υπολογισμού της τιμής της συνάρτησης αυτής ονομάζεται διακριτή ύψωση σε δύναμη και για να επιτελεστεί αρκούν $2\log_2 p$ πολλαπλασιασμοί αριθμών μήκους k bits (ή $\log_2 p$ πολλαπλασιασμοί και $\log_2 p$ διαιρέσεις αριθμών μήκους 2k bits από αριθμούς μήκους k bits). Η διαδικασία της διακριτής ύψωσης σε δύναμη βασίζεται στον αρχικό υπολογισμό των (modulo p) τιμών $a^1, a^2, a^4, a^8, \ldots, a^{2^{k-1}}$.

Η αντίστροφη συνάρτηση είναι της διακριτής ύψωσης σε δύναμη είναι η $f^{-1}(y)$ η οποία θέτει την αντιστοιχία μεταξύ δοσμένης τιμής y και τιμής x για τις οποίες αληθεύει η σχέση : $a^x = y(\mathrm{mod}\ p)$. Το πρόβλημα της εύρεσης του x είναι το περίφημο πρόβλημα διακριτού λογαρίθμου. Οι διακριτοί λογάριθμοι υπολογίζονται δύσκολα όταν ο αριθμός p-1 περιλαμβάνει ένα μεγάλο πρώτο παράγοντα (π.χ. όταν μπορεί να αναπαρασταθεί ως $p-1 = 2p'$ όπου το p' είναι πρώτος αριθμός). Υπό τη συνθήκη αυτή, η πολυπλοκότητα του προβλήματος του διακριτού λογαρίθμου είναι περίπου $p^{1/2} \bmod p$ και η λύση του είναι υπολογιστικά αδύνατη για μεγάλα k (π.χ. $k \ge 512$ bit) και γι αυτό, αν ικανοποιούνται οι συνθήκες επιλογής των p και α η συνάρτηση της διακριτής έκθεσης σε δύναμη είναι μονόδρομη.

Η μέθοδος των Diffie και Hellman για τη διαμοίραση των δημοσίων κλειδιών χρησιμοποιεί ακριβώς τη συνάρτηση διακριτής ύψωσης σε δύναμη ώστε να ανταλλάξει ιδιωτικά κλειδιά μεταξύ δικτυακών χρηστών χρησιμοποιώντας δημόσια μηνύματα. Για

το σκοπό αυτό επιλέγεται ένας μεγάλος πρώτος p καθώς και το αντίστοιχο για αυτόν a ώστε $a < p$. Η ασφάλεια αυτού του δημόσιου συστήματος κρυπτογράφησης επαφίεται πλήρως στον αριθμό p : η ανάπτυξη του αριθμού αυτού σε παράγοντες πρέπει να περιλαμβάνει έναν τουλάχιστον μεγάλο πρώτο αριθμό και το μήκος του σε bit να είναι μεγαλύτερο από 512.

Ο μηχανισμός διαμοιρασμού του δημοσίου κλειδιού μέσα από δημόσιο (μη ασφαλές) κανάλι έχει ως εξής : Κάθε χρήστης επιλέγει ένα τυχαίο ιδιωτικό κλειδί x και υπολογίζει το αντίστοιχο δημόσιο κλειδί y σύμφωνα με τη σχέση : $y = a^x \pmod p$.

Αν και είναι εύκολο να υπολογιστεί το y από οποιοδήποτε x, είναι υπολογιστικά αδύνατο να βρεθεί ο διακριτός λογάριθμος και ως εκ τούτου να βρεθεί ο αριθμός x για τον οποίο το $a^x \pmod p$ ισούται με τη δοσμένη τιμή y. Όλοι οι συνδρομητές της υπηρεσίας τοποθετούν τα δημόσια κλειδιά τους σε φάκελο που είναι ορατός σε όλους, ο οποίος όμως έχει διαπιστευτεί από τρίτη αρχή έκδοσης πιστοποιητικών ώστε να καθίσταται αδύνατη η αλλαγή ενός έγκυρου δημόσιου κλειδιού με κάποιο πλαστό. Αν δυο συνδρομητές **A** και **B**, θέλουν να εγκαταστήσουν μια μυστική συνεδρία ακολουθούν τη παρακάτω διαδικασία: Ο **A** παίρνει το δημόσιο κλειδί του **B** από το κοινό φάκελο και υπολογίζει το από κοινού ιδιωτικό κλειδί χρησιμοποιώντας το δικό του ιδιωτικό κλειδί :

$$Z_{AB} = (y_B)^{x_A} = (a^{x_B})^{x_A} = a^{x_B x_A} \pmod p$$

όπου τα y_A και y_B είναι τα δημόσια κλειδιά των **A** και **B** και x_A, x_B τα αντίστοιχα δημόσια κλειδιά τους. Δεν είναι απαραίτητο να μεταδοθεί το ιδιωτικό κλειδί Z_{AB} μέσω του δικτύου επικοινωνιών γιατί ο συνδρομητής **B** υπολογίζει τη τιμή του με παρόμοιο τρόπο από το δημόσιο κλειδί του **A**, το οποίο βρίσκεται ομοίως στον κοινό και προστατευμένο φάκελο:

$$Z_{AB} = (y_A)^{x_B} = (a^{x_A})^{x_B} = a^{x_A x_B} \pmod p$$

Ένας πιθανός αντίπαλος γνωρίζει τα $y_B = a^{x_B} \pmod p$ και $y_A = a^{x_A} \pmod p$ αλλά για να βρει το ιδιωτικό κλειδί Z_{AB} πρέπει να λύσει ένα πολύπλοκο πρόβλημα διακριτών λογαρίθμων. Το ιδιωτικό κλειδί μπορεί να χρησιμοποιηθεί από τους αντεπιστέλλοντες για την κρυπτογράφηση και ανταλλαγή μυστικών κλειδιών με τα οποία πλέον θα κρυπτογραφούν τα μηνύματα τους χρησιμοποιώντας μάλιστα συμμετρικές μεθόδους κρυπτογράφησης. Υπάρχει λύση στο πρόβλημα των διακριτών λογαρίθμων, αλλά είναι υπολογιστικά ανέφικτη.

Τα δυο εγγενή προβλήματα που εμφανίζονται στα κρυπτοσυστήματα ενός κλειδιού είναι :

- Ο διαμοιρασμός του μυστικού κλειδιού μέσω ασφαλούς καναλιού επικοινωνίας.
- Η αυθεντικοποίηση του μυστικού κλειδιού.

Με τον όρο αυθεντικοποίηση υποδηλώνεται μια διαδικασία που επιτρέπει στον αποδέκτη να βεβαιωθεί ότι το κρυφό κλειδί ανήκει σε έμπιστο αποστολέα (π.χ. τρίτη αρχή διαμοίρασης κλειδιών).

Το σύστημα διαμοίρασης δημοσίων κλειδιών λύνει το πρώτο από τα δυο προβλήματα καθιστώντας δυνατή την αποστολή του κλειδιού χωρίς τη χρήση ασφαλούς καναλιού αλλά και χωρίς να εξαλείφεται και η ανάγκη για μηχανισμό αυθεντικοποίησης. Στην κρυπτογραφία δυο κλειδιών βέβαια, το πρόβλημα της αυθεντικοποίησης δεν είναι τόσο οξύ γιατί η επίλυση του βρίσκεται στην ίδια τη μέθοδο.

2.4 Κρυπτανάλυση (Cryptanalysis)

Ο σκοπός της κρυπτογραφίας είναι να κρατήσει το αρχικό κείμενο, το κλειδί ή και τα δύο, κρυφά από τρίτους, οι οποίοι ονομάζονται και αντίπαλοι, επιτιθέμενοι, παρεμποδιστές, παρεμβαίνοντες, εισβολείς, ανταγωνιστές, ή απλά ο εχθρός. Οι επιτιθέμενοι, θεωρείται ότι έχουν πλήρη πρόσβαση στις επικοινωνίες μεταξύ του αποστολέα και του παραλήπτη.

Κρυπτανάλυση ονομάζεται η επιστήμη ανάκτησης του αρχικού κειμένου ενός κρυπτογραφημένου μηνύματος χωρίς πρόσβαση στο κλειδί. Μια επιτυχής κρυπτανάλυση μπορεί να ανακτήσει το αρχικό κείμενο ή το κλειδί. Μπορεί επιπλέον να εντοπίσει αδυναμίες σε ένα κρυπτοσύστημα, οι οποίες θα οδηγήσουν στα προαναφερόμενα αποτελέσματα. Η απώλεια του κλειδιού από μη κρυπταναλυτικά μέσα ονομάζεται παραβίαση (compromise).

Μια επιχειρούμενη κρυπτανάλυση ονομάζεται επίθεση. Μια θεμελιώδης υπόθεση της κρυπτανάλυσης, που πρωτοδιατυπώθηκε τον 19ο αιώνα από τον Ολλανδό A. Kerckhoffs, είναι πως η μυστικότητα πρέπει να εμπεριέχεται εντελώς στο κλειδί. Ο Kerckhoffs υποθέτει ότι ο κρυπταναλυτής γνωρίζει πλήρως τον αλγόριθμο και την υλοποίηση του. Στην πραγματικότητα, οι κρυπταναλυτές δε διαθέτουν πάντα τόσο λεπτομερείς πληροφορίες. Ωστόσο αυτό μπορεί ευλόγως να υποτεθεί. Εάν κάποιοι δε μπορούν να σπάσουν έναν αλγόριθμο ακόμη και αν γνωρίζουν την λειτουργία του, τότε δε θα μπορούν στα σίγουρα να τον σπάσουν ακόμα και χωρίς αυτή τη γνώση.

Υπάρχουν τέσσερις γενικοί τύποι κρυπταναλυτικών επιθέσεων. Φυσικά ο καθένας από αυτούς υποθέτει πως ο κρυπταναλυτής έχει πλήρη γνώση του κρυπτογραφικού αλγόριθμου που χρησιμοποιείται.

1. Επίθεση στο κρυπτογραφημένο κείμενο μόνο (ciphertext-only attack) Ο κρυπταναλυτής διαθέτει το κρυπτογραφημένο κείμενο διαφόρων μηνυμάτων, το σύνολο των οποίων έχει κρυπτογραφηθεί με χρήση του ίδιου αλγόριθμου. Η δουλειά του κρυπταναλυτή είναι να ανακτήσει το αρχικό κείμενο όσο το δυνατόν

περισσοτέρων μηνυμάτων, ή ακόμη καλύτερα να εκμαιεύσει το κλειδί (ή κλειδιά) που χρησιμοποιήθηκαν για την κρυπτογράφηση των κλειδιών, ώστε να αποκρυπτογραφήσει άλλα μηνύματα που κρυπτογραφήθηκαν με τα ίδια κλειδιά.

2. Επίθεση γνωστού αρχικού κειμένου (Known-plaintext attack). Στην περίπτωση αυτή ο κρυπταναλυτής δεν έχει μόνο πρόσβαση στα κρυπτογραφημένα κείμενα διαφόρων μηνυμάτων αλλά και στα αρχικά κείμενα των μηνυμάτων αυτών. Η δουλειά του είναι να εκμαιεύσει το κλειδί (ή κλειδιά) που χρησιμοποιήθηκε για την κρυπτογράφηση των μηνυμάτων αυτών, ή έναν αλγόριθμο για να αποκρυπτογραφήσει όσο το δυνατόν περισσότερα καινούρια μηνύματα με το ίδιο κλειδί (ή κλειδιά).
3. Επίθεση επιλεγμένου αρχικού κειμένου (Chosen-plaintext attack). Στην περίπτωση αυτή ο κρυπταναλυτής δεν έχει πρόσβαση μόνο στο κρυπτογραφημένο κείμενο και το σχετικό αρχικό κείμενο για αρκετά μηνύματα, αλλά έχει τη δυνατότητα να επιλέξει το μήνυμα, το οποίο θα κρυπτογραφηθεί. Η επίθεση αυτή λοιπόν είναι πιο ισχυρή από την επίθεση γνωστού αρχικού κειμένου γιατί ο κρυπταναλυτής μπορεί να επιλέξει συγκεκριμένα τμήματα κειμένου για να κρυπτογραφηθούν, τα οποία θα αποδώσουν περισσότερες πληροφορίες για το κλειδί. Η δουλειά του είναι να εκμαιεύσει το κλειδί (ή κλειδιά) που χρησιμοποιήθηκε στην κρυπτογράφηση των μηνυμάτων ή έναν αλγόριθμο για να αποκρυπτογραφήσει όσα καινούρια μηνύματα είναι δυνατόν με το ίδιο κλειδί (ή κλειδιά).
4. Προσαρμοζόμενη επίθεση επιλεγμένου αρχικού κειμένου (Adaptive-chosen-plaintext attack). Αυτή είναι μια ειδική περίπτωση επίθεσης επιλεγμένου αρχικού κειμένου. Ο κρυπταναλυτής όχι μόνο μπορεί να επιλέξει το αρχικό κείμενο που θα κρυπτογραφηθεί, αλλά μπορεί να μεταβάλλει την επιλογή του αυτή ανάλογα με τα αποτελέσματα της προηγούμενης κρυπτογράφησης. Σε μια επίθεση επιλεγμένου αρχικού κειμένου, ένας κρυπταναλυτής θα μπορεί να επιλέξει ένα μεγάλο τμήμα αρχικού κειμένου για να κρυπτογραφηθεί. Σε μια προσαρμοζόμενη επίθεση επιλεγμένου αρχικού κειμένου θα μπορεί να διαλέξει ένα μικρότερο τμήμα του αρχικού κειμένου και αναλόγως με τα αποτελέσματα που θα του δώσει αυτό, να επιλέξει το δεύτερο κ.ο.κ.

Υπάρχουν τουλάχιστον τρεις ακόμη τύποι κρυπταναλυτικής επίθεσης.

5. Επίθεση επιλεγμένου κρυπτογραφημένου κειμένου (Chosen-ciphertext attack). Ο κρυπταναλυτής μπορεί να διαλέξει διαφορετικά κρυπτογραφημένα κείμενα για να αποκρυπτογραφηθούν και έχει πρόσβαση στο αποκρυπτογραφημένο κείμενο. Για παράδειγμα, έστω πως ο κρυπταναλυτής έχει πρόσβαση σε ένα ασφαλισμένο κουτί, το οποίο πραγματοποιεί την αποκρυπτογράφηση αυτόματα.

   Η επίθεση αυτή εφαρμόζεται πρωτίστως σε αλγόριθμους γνωστού κλειδιού. Μια επίθεση επιλεγμένου κρυπτογραφημένου κειμένου είναι αποτελεσματική μερικές φορές και εναντίον συμμετρικών αλγόριθμων. Ορισμένες φορές η επίθεση εναντίον επιλεγμένου αρχικού κειμένου και επιλεγμένου κρυπτογραφημένου κειμένου, ονομάζεται επίθεση επιλεγμένου κειμένου.
6. Επίθεση επιλεγμένου κλειδιού (Chosen-key attack). Σε αυτήν την επίθεση, παρά το όνομα της, ο κρυπταναλυτής δεν μπορεί να επιλέξει το κλειδί αλλά έχει κάποια γνώση για τη σχέση μεταξύ των διαφορετικών κλειδιών. Είναι μια περίεργη και δυσνόητη μέθοδος.
7. Εξαναγκαστική κρυπτανάλυση (Rubber-hose Cryptanalysis). Ο κρυπταναλυτής αναλύει, εκβιάζει ή βασανίζει κάποιον μέχρι να του δώσει το κλειδί. Η

δωροδοκία μερικές φορές αναφέρεται και ως επίθεση αγορασμένου κλειδιού (purchase-key attack). Όλες οι επιθέσεις του είδους αυτού είναι πολύ ισχυρές και συχνά ο καλύτερος τρόπος να σπάσει ένας αλγόριθμος.

Οι επιθέσεις γνωστού ή επιλεγμένου αρχικού κειμένου είναι πιο συχνές από ότι πιστεύεται. Δεν είναι σπάνιες οι περιπτώσεις, όπου ένας κρυπταναλυτής έχει αποκτήσει ένα μήνυμα που έχει κρυπτογραφηθεί, ή έχει δωροδοκήσει κάποιον για να κρυπτογραφήσει κάποιο επιλεγμένο μήνυμα. Μερικές φορές μάλιστα η δωροδοκία δεν είναι καν απαραίτητη: αν δοθεί το μήνυμα σε κάποιον πρέσβη, πιθανότατα θα κρυπτογραφηθεί πριν σταλεί πίσω στην χώρα του για μελέτη. Πολλά μηνύματα έχουν τυποποιημένη εισαγωγή και επίλογο, που πιθανότατα είναι γνωστά στον κρυπταναλυτή. Ο κρυπτογραφημένος πηγαίος κωδικός είναι ιδιαιτέρως ευάλωτος εξαιτίας της συχνής εμφάνισης λέξεων-κλειδιών όπως: «define, struct, else, return». Ο κρυπτογραφημένος εκτελέσιμος κωδικός παρουσιάζει επίσης τα ίδια προβλήματα: συναρτήσεις (functions), βρόγχους (loopstructures) κ.λπ. Οι επιθέσεις γνωστού αρχικού κειμένου, ακόμη και επιλεγμένου αρχικού κειμένου, ήταν επιτυχείς κατά τη χρήση τους εναντίον των Γιαπωνέζων και των Γερμανών στο 2ο Παγκόσμιο Πόλεμο. Περισσότερες λεπτομέρειες επί αυτού του θέματος δίνονται στα βιβλία του David Kahn.

Ας μην ξεχνάμε και την υπόθεση του Kerckhoff: αν η ασφάλεια του νέου κρυπτοσυστήματος εξαρτάται από το γεγονός ότι ο εισβολέας δεν γνωρίζει τις εσωτερικές λειτουργίες του, τότε αυτό έχει ήδη αποτύχει. Είναι λανθασμένη η άποψη, πως κρατώντας τις εσωτερικές λειτουργίες του αλγόριθμου κρυφές αντί να δημοσιεύονται στην ακαδημαϊκή κοινότητα για να αναλυθούν, αυξάνεται η ασφάλεια του αλγόριθμου. Η πεποίθηση επίσης ότι κανείς δεν μπορεί να αποσυναρμολογήσει τον αλγόριθμο και να καταλάβει την λειτουργία του είναι αφελής, όπως συνέβη το 1994 με τον αλγόριθμο RC4.

Οι καλύτεροι αλγόριθμοι είναι αυτοί, οι οποίοι έχουν δημοσιοποιηθεί, έχουν δεχθεί επίθεση από τους καλύτερους κρυπτογράφους του κόσμου και παραμένουν άθικτοι. Η NSA κρατά τους αλγόριθμους της μυστικούς προς τρίτους, ενώ παράλληλα διαθέτει τους καλύτερους κρυπτογράφους. Επιπρόσθετα εξετάζει διεξοδικά τους αλγόριθμους της ώστε να ανακαλύψουν τυχόν αδυναμίες. Οι κρυπταναλυτές δεν έχουν πάντα πρόσβαση στους αλγόριθμους, όπως για παράδειγμα έγινε όταν οι Ηνωμένες Πολιτείες έσπασαν τον Ιαπωνικό διπλωματικό κωδικό κατά το 2ο Παγκόσμιο Πόλεμο, αλλά συνήθως τον διαθέτουν. Αν ο αλγόριθμος χρησιμοποιείται σε κάποιο πρόγραμμα ασφαλείας του εμπορίου, είναι απλώς θέμα χρόνου και χρημάτων να αποσυνθέσουν το πρόγραμμα και να ανακτήσουν τον αλγόριθμο. Αν ο αλγόριθμος χρησιμοποιείται στα συστήματα τηλεπικοινωνιών του στρατού, είναι και πάλι θέμα χρόνου και χρημάτων να αγοραστεί (ή και να κλαπεί) ο απαραίτητος εξοπλισμός και να ανακτηθεί ο αλγόριθμος.

Όσοι πιστεύουν, ότι διαθέτουν έναν αλγόριθμο που δεν μπορεί να παραβιαστεί, επειδή ούτε οι ίδιοι μπορούν να το καταφέρουν είναι είτε διάνοιες, είτε ανόητοι. Δε θα έπρεπε να εμπιστεύονται αλγόριθμους των οποίων οι λειτουργίες δεν δημοσιοποιούνται όσο κι αν αυτοί διαφημίζονται ως ασφαλείς. Οι καλοί κρυπτογράφοι βασίζονται στη διεξοδική επιθεώρηση, για να ξεχωρίσουν τους καλούς αλγόριθμους από τους υπόλοιπους.

2.5 Ασφάλεια των Αλγόριθμων (Security of algorithms)

Διαφορετικοί αλγόριθμοι, προσφέρουν διαφορετικές βαθμίδες ασφάλειας: εξαρτάται από το πόσο δύσκολο είναι να σπάσουν. Εάν το κόστος που απαιτείται για να σπάσει ένας αλγόριθμος ξεπερνά την αξία των δεδομένων που αυτός προστατεύει, τότε αυτός κρίνεται ως πιθανόν ασφαλής. Εάν ο χρόνος που χρειάζεται για να σπάσει ο αλγόριθμος είναι μεγαλύτερος από το χρόνο που πρέπει τα δεδομένα να παραμείνουν

μυστικά, τότε είναι και πάλι πιθανόν ασφαλής. Εάν τέλος, το ποσό των δεδομένων που έχουν κρυπτογραφηθεί με ένα μόνο κλειδί, είναι σημαντικά μικρότερο από τα απαραίτητα για να σπάσουν τον αλγόριθμο δεδομένα, τότε ο αλγόριθμος είναι πιθανόν ασφαλής.

Η λέξη «πιθανόν» χρησιμοποιείται, γιατί υπάρχει πάντα η δυνατότητα να γίνουν πρωτοποριακές ανακαλύψεις στην κρυπτογραφία. Περαιτέρω, η αξία των περισσοτέρων δεδομένων μειώνεται με την πάροδο του χρόνου. Είναι σημαντικό η αξία των δεδομένων να παραμένει πάντα μικρότερη από το κόστος σπασίματος του συστήματος ασφαλείας που τα προστατεύει.

Ο Lars Knudsen κατέταξε τις τρεις αυτές κατηγορίες σπασίματος ενός αλγόριθμου σε φθίνουσα σειρά σοβαρότητας:

- *Ολική Παραβίαση (Total Break).* Ο κρυπταναλυτής βρίσκει το κλειδί, K, από το οποίο DK (C) = P.
- *Πλήρης Ανάκτηση (Global Deduction).* Ο κρυπταναλυτής βρίσκει έναν εναλλακτικό αλγόριθμο, A, ισοδύναμο με τον DK (C), χωρίς όμως να γνωρίζει το K.
- *Μοναδική (η τοπική) Ανάκτηση (Instance -or local- Deduction).* Ο κρυπτ-αναλυτής βρίσκει το αρχικό κείμενο ενός υποκλεμμένου κρυπτογραφημένου μηνύματος.
- *Ανάκτηση Πληροφοριών (Information Deduction).* Ένας κρυπταναλυτής αποκτά μερικές πληροφορίες για το κλειδί ή το αρχικό κείμενο. Αυτές οι πληροφορίες μπορεί να είναι μερικά bits του κλειδιού, λίγες πληροφορίες για την μορφή του αρχικού κειμένου κ.ο.κ.

Ένας αλγόριθμος είναι απόλυτα ασφαλής αν όση ποσότητα κρυπτογραφημένου κειμένου και αν διαθέτει ο κρυπταναλυτής, δεν υπάρχουν αρκετές πληροφορίες για την ανάκτηση του κρυπτογραφημένου κειμένου. Στην πραγματικότητα, μόνο ένα σύστημα σημειωμάτων μιας χρήσης είναι απόλυτα ασφαλές, όσες διευκολύνσεις και αν δίνονται. Όλα τα άλλα κρυπτοσυστήματα μπορούν να παραβιαστούν με επιθέσεις κρυπτογραφημένου κειμένου, απλά και μόνο χρησιμοποιώντας όλα τα πιθανά κλειδιά ένα προς ένα και ελέγχοντας αν το τελικό κείμενο βγάζει νόημα. Μια τέτοια επίθεση ονομάζεται άμεση επίθεση (Brute-force attack).

Η κρυπτογραφία ασχολείται ενδελεχώς με κρυπτοσυστήματα που είναι πρακτικά ακατόρθωτο να σπάσουν υπολογιστικά. Ένας αλγόριθμος καλείται υπολογιστικά ασφαλής (computationally secure) ή ισχυρός (strong) εάν δεν μπορεί να παραβιασθεί με τα δοσμένα μέσα, είτε σύγχρονα ή μελλοντικά. Το τι ακριβώς συνθέτει τα «δοσμένα μέσα» είναι ανοικτό προς ερμηνεία.

Η πολυπλοκότητα μιας επίθεσης υπολογίζεται με διάφορους τρόπους:

- *Πολυπλοκότητα Δεδομένων (Data complexity).* Η ποσότητα δεδομένων που χρειάζονται για την επίθεση.
- *Πολυπλοκότητα Υπολογισμών (Processing Complexity).* Ο χρόνος που χρειάζεται για να γίνει η επίθεση. Ονομάζεται συχνά και παράγοντας εργασίας (work factor).
- *Αποθηκευτικές Απαιτήσεις (Storage Requirements).* Το ποσό της υπολογιστικής μνήμης που χρειάζεται για την επίθεση.

Η πολυπλοκότητα μιας επίθεσης λαμβάνεται ως το ελάχιστο των πιο πάνω παραγόντων. Σε μερικές επιθέσεις, οι ρόλοι των τριών παραγόντων αλληλεπιδρούν,

δηλαδή μια πιο γρήγορη επίθεση μπορεί να υλοποιηθεί εις βάρος μιας πολύ μεγαλύτερης απαίτησης για αποθηκευτικά μέσα.

Οι πολυπλοκότητες εκφράζονται ως τάξεις μεγέθους. Αν ένας αλγόριθμος έχει πολυπλοκότητα υπολογισμών της τάξης του 2128, τότε απαιτούνται 2128 πράξεις για να σπάσει ο αλγόριθμος. Οι λειτουργίες αυτές εκτός από περίπλοκες μπορεί να είναι και χρονοβόρες. Αν υποθέσουμε ότι διαθέτουμε αρκετή υπολογιστική ταχύτητα για να εκτελέσουμε ένα εκατομμύριο πράξεις το δευτερόλεπτο και διαθέσουμε ένα εκατομμύριο παράλληλους επεξεργαστές για το σπάσιμο, θα χρειαστούν πάνω από 1,37E+19 χρόνια για να ανακτηθεί το κλειδί. Δηλαδή, ένα δισεκατομμύριο φορές την ηλικία του σύμπαντος.

Αν και η πολυπλοκότητα μιας επίθεσης μένει σταθερή, έως ότου κάποιος κρυπταναλυτής να βρει μια καλύτερη επίθεση φυσικά, η υπολογιστική ισχύς δεν είναι. Έχουν γίνει εκπληκτικές πρόοδοι στην ισχύ των υπολογιστών τον τελευταίο μισό αιώνα και δεν υπάρχει κανένας λόγος να πιστέψουμε ότι αυτό δε θα συνεχιστεί. Πολλές από τις κρυπταναλυτικές επιθέσεις είναι τέλειες για παράλληλους υπολογιστές. Η όλη εργασία μπορεί να διασπαστεί σε δισεκατομμύρια μικρών κομματιών και κανένας από τους επεξεργαστές δε χρειάζεται να αλληλεπιδρά με τους άλλους. Η ανακήρυξη ενός αλγόριθμου ως εντελώς ασφαλούς επειδή δεν μπορεί να σπάσει με τα τωρινά μέσα, είναι παρακινδυνευμένη στην καλύτερη περίπτωση. Τα καλά κρυπτοσυστήματα σχεδιάζονται, ώστε να είναι αδύνατον να σπάσουν με την υπολογιστική ισχύ που θα εμφανιστεί σε βάθος αρκετών χρόνων στο μέλλον.

2.6 Τι είναι το κρυπτογραφικό πρωτόκολλο.

Οι όροι *αλγόριθμος* και *πρωτόκολλο* χρησιμοποιούνται συχνά στην κρυπτογραφία καθώς και σε άλλα πεδία της επιστήμης και της τεχνολογίας με το νόημα τους να είναι σχετικά ξεκάθαρο. Ο αλγόριθμός είναι μια από τις βασικότερες έννοιες των επιστημών του προγραμματισμού και των εφαρμοσμένων μαθηματικών, όπως και το πρωτόκολλο στο κόσμο των επικοινωνιών. Με άλλα λόγια:

Ως αλγόριθμος ορίζεται ένα σύνολο από εντολές, ενέργειες, οδηγίες ή υπολογισμούς που πρέπει να επιτελεστούν ώστε να ληφθεί κάποιο αποτέλεσμα. Κατά την πιο πάνω διαδικασία μπορεί να δημιουργηθούν νέα δεδομένα ως αποτέλεσμα μετασχηματισμών των δεδομένων της πηγής, να χρειαστεί να ληφθεί τυχαία επιλογή σε κάποιο από τα βήματα του αλγόριθμού, ή ο υπολογιστής να λάβει μετρήσεις περιβαλλοντικών παραμέτρων (δηλαδή παραμέτρων από εξωτερικά αντικείμενα). Ο αλγόριθμος εκτελείται σχεδόν πάντα από υπολογιστή και σπάνια από άνθρωπο.

Ως πρωτόκολλο ορίζεται μια συλλογή από ενέργειες (οδηγίες, εντολές, υπολογισμοί, αλγόριθμοι) που επιτελούνται με προκαθορισμένη σειρά από δυο ή περισσότερα υποκείμενα με σκοπό τη λήψη συγκεκριμένου αποτελέσματος. Η ορθότητα της λειτουργίας ενός πρωτοκόλλου εξαρτάται από τις ενέργειες που επιτελεί το υποκείμενο (χρήστης ή συνδρομητής) του κρυπτοσυστήματος. Ένα υποκείμενο μπορεί να είναι επίσης ένας σταθμός εργασίας, ένα πρόγραμμα υπολογιστή, ένας ραδιομεταδότης, ένας τεχνητός δορυφόρος, ένας διακομιστής, μια αρχή κ.ο.κ. Τα υποκείμενα που συμμετέχουν στο πρωτόκολλο δρουν με βάση καθορισμένους αλγόριθμους · με άλλα λόγια ο αλγόριθμος είναι εσωτερικό δομικό στοιχείο του πρωτοκόλλου. Για να μπορέσει ένα πρωτόκολλο να φτάσει στον τελικό σκοπό του πρέπει :

- Το πρωτόκολλο να είναι ορθό, δηλαδή το σύνολο ενεργειών που καθορίζονται από το πρωτόκολλο επιτρέπουν τη λήψη του επιθυμητού αποτελέσματος υπό όλες τις πιθανές συνθήκες.
- Να είναι πλήρες και σαφές καθορίζοντας ευκρινώς τις ενέργειες του κάθε πιθανού συμμετέχοντα σε κάθε πιθανή κατάσταση που μπορεί να ανακύψει.
- Να παρέχει συνεπή αποτελέσματα τα οποία δεν αντιφάσκουν μεταξύ τους.
- Να προϋποθέτει τη γνώση των λειτουργιών του από όλους τους συμβαλλόμενους αλλά και τη συμφωνία τους να συμμετάσχουν.

Τα κρυπτογραφικά πρωτόκολλα είναι πρωτόκολλα στα οποία λαμβάνουν χώρα μετασχηματισμοί κρυπτογραφικών δεδομένων. Αν και χρησιμοποιούν κάποιου είδους κρυπτογραφικό αλγόριθμο, δεν είναι πάντα η μυστικότητα ο τελικός τους σκοπός. Για παράδειγμα, τα μέλη που συμμετέχουν σε ένα κρυπτογραφικό πρωτόκολλο μπορεί να επιθυμούν την ταυτόχρονη διαδικτυακή υπογραφή μιας σύμβασης, να διεξάγουν μια απομακρυσμένη συνεδρία, να διεξάγουν πείραμα με ρίξιμο κέρματος κλπ.

Η κρυπτογράφηση δεδομένων και ο υπολογισμός μονόδρομων συναρτήσεων είναι στην πραγματικότητα η εφαρμογή των κρυπτογραφικών αλγόριθμων. Εφαρμογές όπως η απομακρυσμένη πρόσβαση σε υπολογιστή και ηλεκτρονική ψηφοφορία είναι παραδείγματα κρυπτογραφικών πρωτοκόλλων. Αν ένα πρωτόκολλο χρησιμοποιεί κάποια κρυπτογραφική συνάρτηση, αυτή πρέπει να είναι ασφαλής. Ακόμα όμως κι όταν οι αλγόριθμοι κρυπτογράφησης είναι ασφαλείς, δεν είναι απαραίτητο ότι και το πρωτόκολλο είναι ασφαλές. Για να συμβαίνει αυτό, πρέπει οι κρυπτογραφικοί αλγόριθμοι να είναι ασφαλείς στο περιβάλλον της εκάστοτε εφαρμογής.

Στα κρυπτοσυστήματα πάντα λαμβάνεται υπόψη η ύπαρξη ενός πιθανού αντιπάλου (στη πραγματικότητα υπάρχει πάντα τέτοια απειλή) και ως εκ τούτου οι δημιουργοί κρυπτογραφικών αλγορίθμων και πρωτοκόλλων λαμβάνουν προφυλάξεις, όσο αυτό είναι δυνατό κατά όλων των πιθανών επιθέσεων ενός αντιπάλου. Η επίθεση σε έναν αλγόριθμο, πρωτόκολλο ή κρυπτοσύστημα περιλαμβάνει τις ενέργειες του αντιπάλου με τις οποίες προσπαθεί να διαβάσει ένα κρυπτογράφημα, να «σπάσει» μια μονόδρομη συνάρτηση (δηλαδή να βρει το όρισμα από τη τιμή της συνάρτησης), να οικειοποιηθεί τη ταυτότητα κάποιου νόμιμου χρήστη, να δημιουργήσει πλαστά μηνύματα, να αυξήσει τα δικαιώματα του και γενικότερα, να δημιουργήσει τις συνθήκες εκείνες υπό τις οποίες θα παραβιάζονται οι κανονισμοί ορθής χρήσης του πρωτοκόλλου και του κρυπτοσυστήματος. Αν τέτοιες δράσεις είναι εφικτές, το κρυπτοσύστημα χαρακτηρίζεται ως «ευάλωτο» στις παραπάνω επιθέσεις. Οι επιτιθέμενοι ανάλογα με τις ενέργειες τους κατηγοριοποιούνται στους εξής δυο τύπους:

[1] Ένας παθητικός αντίπαλος δεν αναλαμβάνει καμία δράση που μπορεί να προκαλέσει την αποδιοργάνωση ενός κρυπτογραφικού πρωτοκόλλου. Ο σκοπός του είναι να υποκλέψει μηνύματα που διεκπεραιώνονται από το κρυπτοσύστημα ώστε να διαβάσει τα περιεχόμενα τους, να υπολογίσει και να χαλκέψει διαμοιραζόμενα κλειδιά και να αλλοιώσει το αποτέλεσμα μιας δικτυακής ψηφοφορίας.

[2] Ένας ενεργητικός αντίπαλος προσπαθεί να δημιουργήσει πλαστά μηνύματα, να αναχαιτίσει και να αλλοιώσει μηνύματα, να αποκτήσει πρόσβαση σε βάσεις δεδομένων, να αποκτήσει περαιτέρω δικαιώματα πρόσβασης, να χαλκεύσει ένα δημόσιο κλειδί ή υπογραφή κλπ. Οι καταλληλότερες συνθήκες για τη δράση ενός τέτοιου αντιπάλου είναι οι ενσύρματες τηλεπικοινωνιακές υποδομές ενώ στο πεδίο των ασύρματων επικοινωνιών η δράση του μπορεί να εντοπιστεί

εύκολα. Οι δυσκολότερες περιπτώσεις αφορούν αντιπάλους οι οποίοι είναι ήδη νόμιμοι χρήστες του συστήματος.

Η σχέση του κάθε επικείμενου εισβολέα (αντιπάλου) με τον οργανισμό που χρησιμοποιεί το κρυπτοσύστημα βοηθά στο διαχωρισμό αυτών στις εξής κατηγορίες:

[1] Ως εσωτερικός αντίπαλος λαμβάνεται ένα άτομο με συγκεκριμένη έγκυρη εξουσιοδότηση από τον ίδιο τον οργανισμό στον οποίο επιτίθεται, ή ένας συμμετέχον σε κρυπτογραφικό πρωτόκολλο επικοινωνίας που προσπαθεί να βλάψει τους λοιπούς συνομιλητές. Τόσο οι εξωτερικοί όσο και οι εσωτερικοί αντίπαλοι μπορούν να είναι ενεργοί ή παθητικοί. Η επίθεση από εσωτερικό αντίπαλο ονομάζεται ε*σωτερική*.

[2] Μια επίθεση στην οποία εμπλέκονται μόνο εξωτερικοί αντίπαλοι ονομάζεται *εξωτερική. Η πλέον επικίνδυνη* επίθεση προέρχεται από πιθανή συνεργασία εσωτερικών και εξωτερικών αντιπάλων: Αν υπάρχει αντίπαλος μεταξύ των δημιουργών – προγραμματιστών είναι δυνατό να χρησιμοποιηθούν «καταπακτές» που έχουν ενσωματωθεί στους αλγόριθμους που υπολογίζουν σημαντικές παραμέτρους ή ακόμα και δύσκολοι στον εντοπισμό ιοί.

3 Ο Αλγόριθμος RSA

3.1 Σύντομη Περιγραφή

Αμέσως μετά την ανάπτυξη του αλγόριθμου Knapsack από τον Merkle, εμφανίστηκε και ο πρώτος πλήρης αλγόριθμος δημοσίου κλειδιού ο οποίος χρησιμοποιείται τόσο για κρυπτογράφηση όσο και για ψηφιακές υπογραφές [1,2]. Ο αλγόριθμος αυτός ήταν ο RSA και ξεχώρισε από όλους τους άλλους που εμφανίστηκαν ταυτόχρονα ή τα αμέσως επόμενα χρόνια γιατί ήταν εύκολος στη κατανόηση και την υλοποίηση. Μια πρώτη περιγραφή του αλγόριθμου δημοσιεύτηκε από το Michael Gardner στη στήλη «Μαθηματικά Παιχνίδια» του περιοδικού Scientific American [3]. Το όνομα του προέρχεται από τους τρεις δημιουργούς του: Ron Rivest, Adi Shamir και Leonard Adleman.

Έχει υπάρξει στόχος επιθέσεων κρυπτανάλυσης από την αρχή σχεδόν της εμφάνισής του και παρά τη παλαιότητα του, χαίρει ακόμα εμπιστοσύνης.

Η ασφάλεια του RSA προέρχεται από τη δυσκολία παραγοντοποίησης μεγάλων αριθμών: Τα δημόσια και ιδιωτικά κλειδιά είναι συναρτήσεις ενός ζεύγους μεγάλων (100 ή και 200 ψηφίων) πρώτων αριθμών. Η ανάκτηση του αρχικού κειμένου από το δημόσιο κλειδί και το κρυπτογράφημα είναι ανάλογη της παραγοντοποίησης του γινομένου των δύο πρώτων αριθμών.

Για να παραχθούν τα δύο κλειδιά (δημόσιο και ιδιωτικό), επιλέγονται τυχαία δύο μεγάλοι πρώτοι αριθμοί *p καιq* ίδιου μήκους και υπολογίζεται το γινόμενο τους :

$$N=pq$$

Εν συνεχεία, επιλέγεται τυχαία επίσης το κλειδί κρυπτογράφησης *e*, ώστε το *e* και ο ακέραιος *(p-1)(q-1)* να είναι μεταξύ τους σχετικά πρώτοι. Τελικά χρησιμοποιείται ο εκτεταμένος Ευκλείδειος Αλγόριθμος για να υπολογιστεί το κλειδί αποκρυπτογράφησης, *d* ως εξής :

$$ed=1 \bmod (p-1)(q-1)$$

Με άλλα λόγια:

$$d=e^{-1} \bmod ((p-1)(q-1))$$

όπου, τόσο ο d, όσο και ο n είναι σχετικά πρώτοι. Οι αριθμοί e και n είναι το δημόσιο κλειδί και το d είναι το ιδιωτικό κλειδί. Οι δύο πρώτοι αριθμοί p και q δεν είναι πλέον απαραίτητοι και απορρίπτονται χωρίς όμως να αποκαλυφθούν ποτέ.

Για να κρυπτογραφηθεί ένα μήνυμα M, πρώτα πρέπει να διασπασθεί σε αριθμητικά μέρη μικρότερα του n (για δυαδικά δεδομένα, πρέπει να επιλεγεί η μεγαλύτερη δύναμη του 2 που είναι αμέσως μικρότερη του n). Αυτό σημαίνει ότι αν το p και το q είναι πρώτοι αριθμοί των 100 ψηφίων, τότε το n θα έχει ελαφρώς λιγότερα από 200 ψηφία και κάθε μέρος του μηνύματος (M_i) πρέπει να είναι επίσης μικρότερο των 200 ψηφίων. Το κρυπτογραφημένο μήνυμα C θα αποτελείται από μέρη σχεδόν παρόμοιου μεγέθους c_i και ο τύπος κρυπτογράφησης θα είναι απλά :

$$C_i=M_i^e \bmod N$$

Για να αποκρυπτογραφηθεί ένα μήνυμα, από κάθε κρυπτογραφημένο μέρος C_i υπολογίζεται:

$$M_i=C_i^d \bmod N$$

Αφού :

$$C_i^d=(M_i^e)^d=M_i^{ed}=M_i^{k(p-1)(q-1)+1}=M_iM_i^{k(p-1)(q-1)+1}Mi*1=Mi;all(\bmod N)$$

ο παραπάνω τύπος ανακτά το μήνυμα όπως φαίνεται και στον πιο κάτω πίνακα.

Πίνακας 1 Κρυπτογράφηση RSA

Παράμετροι:	n γινόμενο δύο πρώτων αριθμών p και q (πρέπει να παραμείνουν κρυφοί)
Δημόσιο Κλειδί:	e σχετικά πρώτος με το (p-1)(q-1)
Ιδιωτικό Κλειδί:	$d=e^{-1} \bmod((p-1)(q-1))$
Κρυπτογράφηση:	$C=M^e \bmod N$
Αποκρυπτογράφηση:	$M=C^d \bmod N$

Το μήνυμα θα μπορούσε να έχει κρυπτογραφηθεί με το d και να αποκρυπτογραφηθεί με το e, η επιλογή είναι αυθαίρετη. Ένα απλό αριθμητικό παράδειγμα έχει ως εξής: Αν p=47 και q=71, τότε :

$$N=pq=3337$$

Το κλειδί κρυπτογράφησης e, δεν πρέπει να έχει κοινούς παράγοντες με

$$(p-1)(q-1)=46*70=3220$$

Διαλέγεται e (τυχαία) το 79. Στη περίπτωση αυτή, :

$$d=79^{-1} \bmod 3220=1019$$

Ο αριθμός αυτός υπολογίστηκε χρησιμοποιώντας τον εκτεταμένο Ευκλείδειο Αλγόριθμο. Εν συνεχεία δημοσιεύονται τα e και N και κρατείται μυστικό το d ενώ απορρίπτονται τα p και q. Για να κρυπτογραφηθεί το μήνυμα :

$$M=6882326879666683$$

διασπάται πρώτα σε έξι μέρη των τριών ψηφίων:

$m_1=688$ $m_2=232$ $m_3=687$

$m_4=966$ $m_5=668$ $m_6=003$

Το πρώτο τμήμα κρυπτογραφείται ως:

$$688^{79} \bmod 3337=1570=C_1$$

Επιτελώντας την ίδια πράξη και στα υπόλοιπα μέρη, παράγεται ένα κρυπτογραφημένο μήνυμα :

$$C=1570\ 2756\ 2091\ 2276\ 2423\ 158$$

Η αποκρυπτογράφηση του μηνύματος απαιτεί την ίδια ύψωση σε δύναμη με τη χρήση του κλειδιού αποκρυπτογράφησης 1019 οπότε:

$$1570^{1019} \bmod 3337 = 688 = M_1$$

Το υπόλοιπο μήνυμα ανακτάται με παρόμοιο τρόπο.

3.2 Ασφάλεια του RSA

Στη βιβλιογραφία αναφέρεται πως η ασφάλεια του RSA βασίζεται εξ' ολοκλήρου στο πρόβλημα παραγοντοποίησης μεγάλων πρώτων αριθμών αλλά αυτό είναι απλώς μια εικασία. Δεν έχει ποτέ αποδειχτεί μαθηματικά ότι πρέπει να παραγοντοποιηθεί το N ώστε να υπολογιστεί το C από τα M και e. Αν και θεωρείται πιθανό να ανακαλυφθεί ένας εντελώς νέος τρόπος κρυπτανάλυσης του RSA υπολογίζοντας το d, η μέθοδος αυτή μπορεί να χρησιμοποιηθεί εξίσου για να παραγοντοποιηθούν μεγάλοι πρώτοι αριθμοί δίνοντας ένα νέο εργαλείο κρυπτογράφησης.

Είναι επίσης δυνατή η επίθεση στον RSA μαντεύοντας το (p-1)(q-1)αλλά αυτού του είδους η επίθεση δεν είναι πιο εύκολη από τη παραγοντοποίηση του N [4].

Κάποιες παραλλαγές του RSA έχουν αποδειχτεί τόσο δύσκολες στη κρυπτανάλυση όσο και το πρόβλημα της παραγοντοποίησης. Για παράδειγμα στην παραπομπή [5] αποδεικνύεται ότι η ανάκτηση μερικών μόνο bit πληροφορίας είναι τόσο δύσκολη όσο και η αποκρυπτογράφηση ολόκληρου του μηνύματος.

Η παραγοντοποίηση του N είναι η πιο προφανής μέθοδος επίθεσης. Ο κάθε επιτιθέμενος θα έχει στη διάθεση του το δημόσιο κλειδί e και το modulus N, οπότε για να βρει το κλειδί αποκρυπτογράφησης d, θα πρέπει να παραγοντοποιήσει το N. Επειδή η τεχνολογία μπορεί ήδη να υπολογίσει ένα 129 δεκαδικών ψηφίων modulus, το N θα πρέπει να είναι μεγαλύτερο από αυτό.

Είναι φυσικά πιθανό για ένα κρυπταναλυτή να δοκιμάσει κάθε πιθανό d ώσπου να βρει το σωστό αλλά αυτή η εξαντλητική (brute-force) επίθεση είναι λιγότερο αποδοτική ακόμα και από τη παραγοντοποίηση του N.

Κατά καιρούς έχουν εμφανιστεί επιστήμονες που υποστηρίζουν ότι έχουν βρει εύκολους τρόπους επίθεσης στον RSA αλλά μέχρι στιγμής δεν έχει αποδειχτεί τίποτα. Για παράδειγμα, σε εργασία του William Payne προτείνεται μια μέθοδος που βασίζεται στο μικρό θεώρημα του Fermat [6] η οποία δυστυχώς αποδεικνύεται πιο αργή και από τη παραγοντοποίηση του modulus.

Ένα άλλο σημείο ανησυχίας είναι πιο κοινοί αλγόριθμοι υπολογισμού πρώτων αριθμών είναι πιθανοτικοί και με τον τρόπο αυτό είναι δυνατό να ελαχιστοποιηθεί η πιθανότητα ένας εκ των p ή q να είναι σύνθετος. Αν αυτό συμβεί, τότε η κρυπτογράφηση ή η αποκρυπτογράφηση δεν θα δουλεύουν σωστά. Υπάρχουν μερικοί αριθμοί που ονομάζονται αριθμοί Carmichael τους οποίους αδυνατούν να υπολογίσουν /εντοπίσουν πιθανοτικές γεννήτριες τυχαίων αριθμών. Αν και αυτοί οι αριθμοί είναι επισφαλείς αν χρησιμοποιηθούν, είναι εξαιρετικά σπάνιοι [7].

3.2.1 Επίθεση επιλεγμένου κρυπτογραφήματος εναντίον του RSA

Κάποιες επιθέσεις εργάζονται κατά της υλοποίησης του RSA. Δεν επιτίθενται στον αλγόριθμο καθαυτό αλλά στο πρωτόκολλο αποδεικνύοντας ότι δεν είναι σημαντικό απλώς να χρησιμοποιείται ο RSA, αλλά να γίνεται και ορθή χρήση του.

1ο Σενάριο: Αν ο κρυπταναλυτής καταφέρει να αποσπάσει μήνυμα c κρυπτογραφημένο κατά RSA με το δημόσιο κλειδί του, d, για να διαβάσει το μήνυμα M:

$$M=c^d$$

Για να ανακτήσει το M πρώτα διαλέγει ένα τυχαίο αριθμό r μικρότερο του N και αφού βρει το δημόσιο κλειδί e του νόμιμου αποστολέα υπολογίζει:

$$x=r^e \bmod N$$

$$y=x^c \bmod N$$

$$t=r^{-1} \bmod N$$

Αν $x=r^e \bmod N$, τότε $r=x^d \bmod N$.

Αν ο κρυπταναλυτής καταφέρει απλώς το νόμιμο αποστολέα να κρυπτογραφήσει ένα μήνυμα με το ιδιωτικό κλειδί του, θα αποκρυπτογραφήσει το y (γιατί πρέπει να κρυπτογραφήσει το μήνυμα και όχι το hash αυτού).

Ο αποστολέας λοιπόν που δεν έχει ξαναδεί το y, στέλνει στο κρυπταναλυτή (προφανώς δεν γνωρίζει την ιδιότητα του):

$$u=y^d \bmod N$$

και ο κρυπταναλυτής υπολογίζει πλέον:

$$tu \bmod N=r^{-1}y^d \bmod N=r^{-1}x^dC^d \bmod N=C^d \bmod N=M$$

και με αυτό τον τρόπο υπολογίζει το αρχικό μήνυμα M.

2ο Σενάριο: Έστω μια εταιρεία εκδόσεως ψηφιακών πιστοποιητικών η οποία «υπογράφει» ψηφιακά έγγραφα νόμιμων χρηστών με RSA ψηφιακή υπογραφή και τα επιστρέφει στους χρήστες. Αν κάποιος χρήστης θέλει μια τέτοια ψηφιακή υπογραφή για ένα αρχείο του M' (το οποίο έχει κάλπικη χρονοσφραγίδα ή δείχνει ότι προέρχεται από άλλον), δρα ως εξής :

Πρώτα επιλέγει μια αυθαίρετη τιμή x και υπολογίζει $y=x^e \bmod N$. Μπορεί επίσης εύκολα να βρει το δημόσιο κλειδί της εταιρείας e αφού χρησιμοποιείται για να επαληθεύσουν οι λοιποί χρήστες τις υπογραφές των αρχείων. Εν συνεχεία υπολογίζει M=yM'mod N και το στέλνει στην εταιρεία για να λάβει υπογραφή. Υπολογίζει κατόπιν το $(M^d \bmod N)x^{-1} \bmod N$ που ισούται με $N'^d \bmod N$ και είναι η υπογραφή του M'.

Ο χρήστης με παράτυπο μήνυμα διαθέτει αρκετές μεθόδους για να επιτύχει τους στόχους του, όλες εκ των οποίων εκμεταλλεύονται το γεγονός ότι η ύψωση σε δύναμη διατηρεί τη πολλαπλασιαστική δομή της εισόδου. Δηλαδή:

$$(xM)^d \bmod N=x^dM^d \bmod N \text{ [8,9,10]}$$

3ο Σενάριο: Αν ο επιτιθέμενος θέλει ένας απλός χρήστης να του υπογράψει μήνυμα M_3, παράγει δυο μηνύματα M_1και M_2 ώστε:

$M_3= M_1M_2 \pmod N$ αν ο επιτιθέμενος καταφέρει το χρήστη να του υπογράψει τα M_1 και M_2 υπολογίζει και το :

$$M_3^d=(M_1^d \bmod N)(M_2^d \bmod N)$$

Με άλλα λόγια πρέπει να γίνεται συνετή χρήση των ψηφιακών υπογραφών και να χρησιμοποιείται πάντα μια μονόδρομη συνάρτηση hash στην αρχή. Το πρότυπο ISO 9796 αποτρέπει τέτοιες επιθέσεις.

3.2.2 Κοινές επιθέσεις Modulus στον RSA

Μια κοινή υλοποίηση του RSA δίνει σε όλους το ίδιο n αλλά διαφορετικές τιμές για τα εκθετικά e και d. Δυστυχώς όμως αυτό συνεπάγεται το πρόβλημα όταν το ίδιο μήνυμα κρυπτογραφείται με δυο διαφορετικά εκθετικά (και τα δυο με το ίδιο modulus) και εάν αυτά είναι σχετικά πρώτοι (όπως και συχνά συμβαίνει), τότε το αρχικό κείμενο μπορεί να ανακτηθεί χωρίς κανένα από τα εκθετικά αποκρυπτογράφησης. [11]

Αν M το μη κρυπτογραφημένο κείμενο, τα δυο κλειδιά είναι e_1 και e_2 με κοινό modulus N. Τα δυο κρυπτογραφήματα θα είναι :

$$C_1=M^{e}{}_1 \bmod N$$

$$C_2=M^{e}{}_2 \bmod N$$

Ο κρυπταναλυτής γνωρίζει τα N, e_1, e_2, C_1 και C_2 και βρίσκει το M ως εξής: Αφού τα e_1 και e_2 είναι σχετικά πρώτοι, μπορεί με το εκτεταμένο Ευκλείδειο Θεώρημα να βρει r και s ώστε:

$$re_1+se_2=1$$

Υποθέτοντας ότι το r είναι αρνητικός, τότε με το εκτεταμένο Ευκλείδειο Θεώρημα υπολογίζει το C_1^{-1}. Τότε:

$$(C_1^{-1})*C_2^{S}=M \bmod N$$

Υπάρχουν δυο ακόμα πιο ήπιες μέθοδοι εναντίον τέτοιων συστημάτων. Η πρώτη μέθοδος χρησιμοποιεί πιθανοτικές μεθόδους υπολογισμού του N. Η άλλη χρησιμοποιεί ντετερμινιστικό αλγόριθμο για να υπολογίσει το κρυφό κλειδί του άλλου χωρίς να παραγοντοποιήσει το modulus.

Η αντιμετώπιση της παραπάνω επίθεσης έγκειται στη μη διαμοίραση κοινού N μεταξύ ομάδων χρηστών [12].

3.3 Ψηφιακές Υπογραφές:

Για να υλοποιήσουν την ιδέα τους οι Diffie και Hillman πρότειναν ένα σύστημα δημοσίου κλειδιού το οποίο δύναται να υποστηρίξει ένα δίκτυο πολλών συνδρομητών και το οποίο βασίζεται στο εξής σχήμα: Ο κάθε συνδρομητής – για παράδειγμα ο i-στος επιλέγει μια τυχαία τιμή για τη z_i παράμετρο χωρίς να τη κοινοποιήσει στους υπόλοιπους. Εν συνεχεία, σχεδιάζει τον αλγόριθμο E_{z_i} και τον δημοσιεύει σε ένα κοινό κατάλογο ενώ σχεδιάζει και τον D_{z_i} , τον οποίο διατηρεί μυστικό. Ο οποιοσδήποτε άλλος συνδρομητής –για παράδειγμα ο j-στος, χρησιμοποιεί το δημόσιο αλγόριθμο

κρυπτογράφησης E_{z_i} και υπολογίζει το κρυπτογράφημα $C = f_{z_i}(M)$ το οποίο στη συνέχεια αποστέλλει στον i-στο συνδρομητή. Αυτός με τη σειρά του, χρησιμοποιώντας τον ιδιωτικό αλγόριθμο D_{z_i}, ανασύρει το αρχικό κείμενο : $f_{z_i}^{-1}(C) = M$.

Οι δημιουργοί του παραπάνω γενικευμένου σχήματος κρυπτογράφησης απέδειξαν ότι μπορεί να χρησιμοποιηθεί και για *ψηφιακές υπογραφές*. Ως *ψηφιακή υπογραφή* ορίζεται ένας αριθμός με συγκεκριμένη δομή που καθιστά δυνατή τη χρήση ενός δημοσίου κλειδιού χάρη στον οποίο είναι δυνατό να επιβεβαιωθεί ότι ο αριθμός αυτός δημιουργήθηκε για κάποιο μήνυμα με τη βοήθεια μυστικού κλειδιού. Για την υλοποίηση του σχήματος αυτού, επιλέγεται μονόδρομη συνάρτηση –καταπακτή f_z τέτοια ώστε για κάθε πιθανή τιμή της παραμέτρου z, το πεδίο ορισμού της f_z να συμπίπτει με το πεδίο τιμών της. Σύμφωνα με αυτή την απαίτηση, για κάθε μήνυμα που μπορεί να αναπαρασταθεί ως αριθμός από το πεδίο ορισμού της συνάρτησης $f_z(x)$, ο i συνδρομητής μπορεί πλέον να χρησιμοποιήσει τον ιδιωτικό αλγόριθμο ώστε να υπολογίσει τον αριθμό $S = f_{z_i}^{-1}(M)$. Αν μάλιστα το μήνυμα έχει πολύ μεγάλο μήκος, μπορεί να διαιρεθεί σε τμήματα του απαραίτητου μεγέθους το καθένα εκ των οποίων θα υπογραφεί ξεχωριστά.

Κάθε χρήστης του κρυπτοσυστήματος μπορεί να ανακτήσει το μήνυμα *M* από τη τιμή *S*. Αν το *M* είναι ένα κατανοητό μήνυμα ή τουλάχιστον μπορεί να συσχετιστεί με ένα τέτοιο μήνυμα σύμφωνα με προκαθορισμένο κανόνα, τότε η τιμή *S* θεωρείται *η ψηφιακή υπογραφή του M μηνύματος από τον i συνδρομητή.* Αυτό συμβαίνει γιατί μόνο ο κάτοχος του ιδιωτικού αλγόριθμου D_{z_i} μπορεί να δημιουργήσει ένα αρχικό κείμενο *S* το οποίο κρυπτογραφείται στο κρυπτογράφημα *M* με τη βοήθεια του αλγόριθμου E_{z_i}. Με άλλα λόγια, μόνο ο i-στος χρήστης μπορεί να υπολογίσει το $f_{z_i}^{-1}$.

Ο i-στος συνδρομητής μπορεί μάλιστα να στείλει στον j και ψηφιακά υπογεγραμμένο κρυφό μήνυμα κρυπτογραφώντας το *S* χρησιμοποιώντας τον ιδιωτικό αλγόριθμο E_{z_i} λαμβάνοντας έτσι το κρυπτογράφημα $C = E_{z_i}(S)$. Ο αποδέκτης του μηνύματος (δηλαδή ο συνδρομητής j), το αποκρυπτογραφεί με το μυστικό του αλγόριθμο $D_{z_j}(C) = S$ και εν συνεχεία αποκρυπτογραφεί το *S* με το δημόσιο αλγόριθμο του i-χρήστη λαμβάνοντας $E_{z_i}(S) = M$. Ως εκ τούτου, ο j-χρήστης ανακτά τόσο την ψηφιακή υπογραφή του i όσο και το αρχικό κείμενο από το κρυπτογράφημα C.

Η χρήση πρωτοκόλλων που βασίζονται σε μεθόδους συμμετρικής κρυπτογράφησης προϋποθέτει ότι τα δυο αντεπιστέλλοντα μέλη *εμπιστεύονται* το ένα το άλλο. Τα κρυπτοσυστήματα δημοσίου κλειδιού –δηλαδή τα ασύμμετρα κρυπτοσυστήματα- καθιστούν δυνατή την υλοποίηση πρωτοκόλλων επικοινωνίας τα οποία *δε γνωρίζουν ή δεν εμπιστεύονται το ένα το άλλο* και οι ψηφιακές υπογραφές είναι τα πιο σημαντικά υποδείγματα των συστημάτων αυτών. Για να υλοποιηθεί το σχήμα των ψηφιακών υπογραφών σε υπαρκτές εταιρικές δικτυακές συναλλαγές πρέπει πρώτα από όλα να τυποποιηθεί και να γίνει συμβατό με τις κατά τόπους ή διεθνείς νομοθεσίες. Η δε ανταλλαγή των δημοσίων κλειδιών πρέπει να γίνεται με διαφανή και νομότυπο μηχανισμό που θα παρέχει προστασία σε περίπτωση που ένας εκ των συνδρομητών αποπειραθεί να απαρνηθεί το δημόσιο κλειδί του ακυρώνοντας έτσι τη συναλλαγή.

3.4 Είδη επιθέσεων σε ψηφιακές υπογραφές.

Στο σύστημα ψηφιακών υπογραφών, χρησιμοποιούνται τρεις κρυπτογραφικοί αλγόριθμοι : ο αλγόριθμος δημιουργίας ψηφιακής υπογραφής με το ιδιωτικό κλειδί, ο αλγόριθμος επαλήθευσης της υπογραφής με τη χρήση του δημόσιου κλειδιού και ο αλγόριθμος υπολογισμού της συνάρτησης σύνοψης του μηνύματος που υπογράφεται. Οι αλγόριθμοι δημιουργίας των ιδιωτικών και δημοσίων κλειδιών έχουν μαθηματικό υπόβαθρο. Η λειτουργία πραγματικών συστημάτων ψηφιακών υπογραφών απαιτεί υποστήριξη σε νομικό και οργανωτικό επίπεδο πέρα από τις αναγκαίες επενδύσεις σε λογισμικό και υλισμικό. Το νομικό υπόβαθρο περιλαμβάνει την υιοθέτηση νόμων που κατοχυρώνουν τη χρήση τέτοιων συστημάτων. Η οργανωτική βάση προϋποθέτει την καταγραφή των χρηστών σε διαπιστευμένο κέντρο και την υπογραφή εγγράφων μεταξύ του χρήστη και του κέντρου (ή μεταξύ δυο χρηστών), στην οποία δηλώνεται ξεκάθαρα η υπευθυνότητα για την ανταλλαγή κλειδιών. Το υπόβαθρο σε λογισμικό και υλισμικό περιλαμβάνει τα εργαλεία που καθιστούν δυνατή την εκτέλεση περίπλοκων υπολογισμών και παρέχουν ασφάλεια στη βάση δεδομένων που περιέχει υπογεγραμμένα έγγραφα και τα αντίστοιχα δείγματα υπογραφών.

Οι πιθανές επιθέσεις κατά ψηφιακών υπογραφών μπορούν να υποδιαιρεθούν στις παρακάτω κατηγορίες:

- Επιθέσεις κατά κρυπτογραφικών αλγόριθμων
- Επιθέσεις παραβίασης των κανονισμών λειτουργίας των πρωτοκόλλων
- Επιθέσεις παραβίασης της ακεραιότητας του συστήματος ψηφιακών υπογραφών

Ο επιτιθέμενος μπορεί να είναι εξωτερικός ως προς το σύστημα ή να είναι το υπογράφων μέλος (απάρνηση υπογραφής) ή το μέλος που επαληθεύει την υπογραφή (προσπάθεια δημιουργίας ψευδούς υπογραφής).

Οι επιθέσεις στους κρυπτογραφικούς αλγόριθμους περιλαμβάνουν την επίλυση περίπλοκων μαθηματικών προβλημάτων όπως η εύρεση του διακριτού λογαριθμικού modulo ενός μεγάλου πρώτου αριθμού δίνοντας έτσι ελάχιστες πιθανότητες επιτυχίας στον επιτιθέμενο. Τέτοιες επιθέσεις μπορούν να εξαπολυθούν εναντίον αλγόριθμων κρυπτογράφησης δυο κλειδιών ή συναρτήσεων σύνοψης. Στην πρώτη περίπτωση, πλαστογραφείται η υπογραφή ενώ στη δεύτερη το έγγραφο. Οι επιθέσεις που αφορούν τους κανονισμούς λειτουργίας των πρωτοκόλλων περιλαμβάνουν για παράδειγμα την αναπαραγωγή ήδη υπογεγραμμένων μηνυμάτων ή τη χρονική καθυστέρηση μηνυμάτων. Για να αποτραπούν τέτοιες επιθέσεις, το έγγραφο περιλαμβάνει ειδικά πεδία όπου καθορίζονται τα περιεχόμενα και η αρίθμηση του εγγράφου. Κρίνεται επίσης απαραίτητη η χρήση μηχανισμών εναντίον πιθανής απάρνησης του μηνύματος από τον αποστολέα.

Οι επιθέσεις που σχετίζονται με την παραβίαση της ακεραιότητας του συστήματος ψηφιακών υπογραφών είναι περισσότερο ποικίλες· περιλαμβάνουν τη διαγραφή ενός μηνύματος από τη βάση δεδομένων, την υποκλοπή του ιδιωτικού κλειδιού με εργαλεία λογισμικού ή υλισμικού, τη χρήση πλαστού δημόσιου κλειδιού, ακόμα και την αντικατάσταση ενός δημοσίου κλειδιού στη βάση δεδομένων. Τα παραδείγματα αυτά υποδεικνύουν ότι πολλές από τις επιθέσεις σχετίζονται με τη μη εξουσιοδοτημένη πρόσβαση και αλλοίωση των δεδομένων του συστήματος ψηφιακών υπογραφών, γι αυτό κρίνεται απαραίτητη η λειτουργία του σε ασφαλές περιβάλλον.

Οι επιθέσεις σε ένα κρυπτοσύστημα μπορεί να προέρχονται και από την ανάθεση σε χρήστη λανθασμένης ψηφιακής υπογραφής ή από τη μη εγκεκριμένη πρόσβαση στις φυσικές εγκαταστάσεις όπου στεγάζεται το σύστημα αυτό ή από κάποια ατέλεια στο λογισμικό ή άγνωστο ιό. Για να αποφευχθούν όλα αυτά, τα κρυπτογραφικά εργαλεία και ο εξοπλισμός πρέπει να επιθεωρούνται και να πιστοποιούνται από ειδικούς οργανισμούς.

4 Κατηγορίες Επιθέσεων στον RSA

Οι επιθέσεις στον RSA εμπίπτουν σε τέσσερις κατηγορίες: (α) στοιχειώδεις επιθέσεις που εκμεταλλεύονται κατάφωρη κατάχρηση του κρυπτοσυστήματος, (β) επιθέσεις μικρού εκθετικού οι οποίες είναι τόσο σοβαρές που δε θα έπρεπε να χρησιμοποιείται ποτέ εκθετικό με μικρή αριθμητική τιμή, (γ) επιθέσεις μικρού δημόσιου εκθετικού, και (δ) επιθέσεις στην υλοποίηση καθαυτή του αλγόριθμου.

Οι επιθέσεις αυτές υπογραμμίζουν ότι οι έρευνες στο μαθηματικό υπόβαθρο του RSA είναι ακόμα ανεπαρκείς. Οι Desmedt και Odlyzko [22], Joye και Quisquater [21] και τέλος, οι deJonge και Chaum [27], περιγράφουν επιπλέον επιθέσεις. Οι επιθέσεις πάντως που αναφέρονται στα παραπάνω κεφάλαια είναι δυνατό να αχρηστευθούν με το σωστό «παραγέμισμα» του μηνύματος πριν από την κρυπτογράφηση ή την ψηφιακή υπογραφή.

4.1 Θεωρητικό Υπόβαθρο

Ένα μήνυμα είναι ένας ακέραιος $M \in Z_N^*$. Για να κρυπτογραφηθεί το M, πρέπει να υπολογισθεί το $C = M^e \bmod N$. Για να αποκρυπτογραφήσει το κρυπτογράφημα ο νόμιμος αποδέκτης υπολογίζει το $C^d = \bmod N$. Πράγματι $C^d = M^{ed} = M(\bmod N)$ [14] όπου η τελευταία ισότητα υπακούει στο θεώρημα του Euler. Η παραπάνω περιγραφή υπεραπλουστεύει την κρυπτογράφηση κατά RSA. Στην πραγματικότητα το μήνυμα συμπληρώνεται [1] πριν από την κρυπτογράφηση για να επιτευχθεί το επιθυμητό μέγεθος. Για παράδειγμα, ένας απλός αλλά ταυτόχρονα ανεπαρκής αλγόριθμος μπορεί να συμπληρώσει το μη κρυπτογραφημένο κείμενο M, με την παράθεση τυχαίων bit πριν από το τέλος του M, (*πριν την κρυπτογράφηση*). Η προσθήκη τυχαιότητας στην διαδικασία κρυπτογράφησης είναι απαραίτητη για την επίτευξη ασφάλειας.

Η RSA συνάρτηση καθορίζεται ως $x \mapsto x^e \bmod N$. Αν δοθεί το d, η συνάρτηση μπορεί εύκολα να αντιστραφεί δεδομένης της παραπάνω ισότητας. Το d αναφέρεται και ως καταπακτή (trapdoor) η οποία επιτρέπει την αντιστροφή της συνάρτησης.

Προφανής σκοπός της εργασίας αυτής είναι η αντιστροφή της RSA συνάρτησης χωρίς αυτή την καταπακτή (d). Αυτή η τεχνική αναφέρεται και ως σπάσιμο του RSA. Το πρόβλημα για την ακρίβεια τίθεται ως εξής: Δεδομένης της τριάδας $\langle N, e, C \rangle$ αναζητείται πόσο δύσκολο είναι να υπολογισθεί η e-οστή ρίζα του C moduloN $= \mathrm{p} \cdot \mathrm{q}$ όταν η παραγοντοποίηση του N είναι άγνωστη. Αφού το Z_N^* είναι ένα πεπερασμένο σύνολο, είναι δυνατό να απαριθμηθούν όλα τα στοιχεία του Z_N^* μέχρι να βρεθεί το σωστό M.

Δυστυχώς, αυτό καταλήγει[15] σε αλγόριθμο με χρόνο ολοκλήρωσης της τάξης N ή αλλιώς με εκθετική ύψωση του μεγέθους εισόδου το οποίο της τάξης $\log_2 N$. Το ενδιαφέρον φυσικά επικεντρώνεται σε αλγορίθμους με σημαντικά μικρότερο χρόνο ολοκλήρωσης, κυρίως της τάξης n^c *όπου* $n = \log_2 N$ και c μια μικρή σταθερά (για παράδειγμα μικρότερη του 5). Τέτοιοι αλγόριθμοι συνήθως λειτουργούν σωστά, βάση των παραπάνω εισόδων και γι' αυτό αναφέρονται και ως αποτελεσματικοί.

Σε αυτή την έρευνα διερευνάται κατά κύριο λόγο η RSA συνάρτηση σε αντίθεση με το RSA κρυπτοσύστημα.. Με άλλα λόγια, ερευνάται η δυσκολία αντιστροφής της RSA συνάρτησης με τυχαίες εισόδους, αυτό συνεπάγεται ότι δεδομένης της τριάδας $\langle N, e, C \rangle$

ένας επιτιθέμενος δεν μπορεί να ανακτήσει το αρχικό μήνυμα M. Παρόλα αυτά, ένα κρυπτοσύστημα πρέπει να αντιστέκεται και σε πιο «ήπιες» επιθέσεις. Δηλαδή, δεδομένης τριάδας $\langle N,e,C\rangle$, θα πρέπει να είναι αδύνατο να ανακτηθούν οποιεσδήποτε πληροφορίες σχετικά με το M. Αυτό ονομάζεται σημασιολογική ασφάλεια και δεν θα αναφερθούμε σε τέτοιου είδους επιθέσεις, απλώς υπογραμμίζουμε ότι ο RSA δεν είναι σημασιολογικά ασφαλής. Δεδομένης μιας τριάδας $\langle N,e,C\rangle$ κάποιος μπορεί εύκολα να εκμαιεύσει κάποιες πληροφορίες για το M (Για παράδειγμα το Ιακωβιανό σύμβολο M επί του N μπορεί να εκμαιευτεί εύκολα από το C). Ο RSA μπορεί όμως εύκολα να καταστεί σημασιολογικά ασφαλής[16] αν προστεθεί τυχαιότητα στη διαδικασία κρυπτογράφησης.

Η συνάρτηση RSA $x \mapsto x^e \bmod N$ είναι χαρακτηριστικό παράδειγμα μιας μονόδρομης συνάρτησης καταπακτής, Μπορεί εύκολα να υπολογισθεί, αλλά ως σήμερα τουλάχιστον, δεν είναι εύκολο να αντιστραφεί χωρίς την καταπακτή d, με εξαίρεση ιδιάζουσες συνθήκες. Οι μονόδρομες συναρτήσεις «καταπακτές» μπορούν επίσης να χρησιμοποιηθούν για ψηφιακές υπογραφές [19]. Οι ψηφιακές υπογραφές παρέχουν αυθεντικοποίηση και μη απάρνηση ψηφιακών νομικών εγγράφων. Για παράδειγμα, χρησιμοποιούνται για την υπογραφή ψηφιακών επιταγών ή ηλεκτρονικών αγορών.

Για να υπογραφεί ένα μήνυμα $M \in Z_N^*$ με RSA, η Alice εφαρμόζει το ιδιωτικό κλειδί της $\langle N,d\rangle$ επί του M και λαμβάνει την υπογραφή $S = M^d \bmod N$. Δοσμένων των $\langle M,S\rangle$, ο καθένας μπορεί να επαληθεύει την υπογραφή της Alice ελέγχοντας ότι $S^e = M \bmod N$. Εφόσον μόνο η Alice μπορεί να παράγει το S, κάποιος μπορεί να εικάσει ότι ένας αντίπαλος να χαλκεύσει την υπογραφή της Alice. Δυστυχώς όμως τα πράγματα δεν είναι τόσο απλά και χρειάζονται επιπλέον μέτρα για να επιτευχθεί ένα κατάλληλο επίπεδο ασφάλειας. Οι ψηφιακές υπογραφές αποτελούν μια σημαντική εφαρμογή του RSA και κάποιες από τις επιθέσεις που θα αναλυθούν, στοχοποιούν RSA υπογραφές.

Ένα ζεύγος κλειδιών RSA δημιουργείται με την επιλογή δυο τυχαίων πρώτων αριθμών, μήκους n/2 και πολλαπλασιάζοντας τους ώστε να υπολογιστεί το N. Εν συνεχεία για δεδομένο [17] εκθετικό κρυπτογράφησης $e \prec \varphi(N)$, υπολογίζεται το $d = e^{-1} \bmod \varphi(N)$ χρησιμοποιώντας τον εκτεταμένο Ευκλείδειο αλγόριθμο. Εφόσον το σύνολο των πρώτων αριθμών είναι επαρκώς «πυκνό» ένας τυχαίος πρώτος αριθμός μήκους n/2 bit μπορεί να παραχθεί ταχύτατα επιλέγοντας επαναλαμβανόμενα τυχαίους ακεραίους μήκους n/2 bit και εξετάζοντάς τους αν είναι πρώτοι, χρησιμοποιώντας ένα πιθανοτικό έλεγχο (τεστ) πρώτων αριθμών.[31]

4.2 Παραγοντοποίηση μεγάλων ακεραίων

Η πρώτη επίθεση στο δημόσιο κλειδί RSA $\langle N,e\rangle$ είναι η παραγοντοποίηση του modulus N. Δεδομένης της παραγοντοποίησης του N ένας επιτιθέμενος μπορεί εύκολα να κατασκευάσει τη συνάρτηση φ(N) από την οποία μπορεί να προκύψει το εκθετικό αποκρυπτογράφησης $d = e^{-1} \bmod \varphi(N)$. Η παραγοντοποίηση του modulus ονομάζεται επίθεση brute-force (ωμής-δύναμης, βίας) στον RSA.

Παρότι οι αλγόριθμοι παραγοντοποίησης βελτιώνονται συνέχεια οι πλέον εξελιγμένοι δεν απειλούν τον RSA, όταν αυτός εφαρμόζεται σωστά. Η παραγοντοποίηση μεγάλων ακεραίων είναι ένα από τα ωραιότερα προβλήματα των υπολογιστικών μαθηματικών [30, 32] αλλά δε θα αναφερθούμε σε αυτή πιο αναλυτικά. Για πληρότητα σημειώνεται ότι ο πιο σύγχρονος αλγόριθμος παραγοντοποίησης είναι ο General Number Field

Sieve (Κόσκινο Πεδίου Γενικευμένων Αριθμών), του οποίου ο χρόνος ολοκλήρωσης για είσοδο μήκους n-bit είναι

$$\exp((c + o(1))n^{\frac{1}{3}} \log^{\frac{2}{3}} n)$$

για κάποια σταθερά c<2. Οι επιθέσεις[30] κατά του RSA που διαρκούν περισσότερο από το παραπάνω όριο δεν είναι ενδιαφέρουσες. Τέτοιες επιθέσεις είναι η εξαντλητική έρευνα για το M και κάποιες παλαιότερες επιθέσεις που εμφανίσθηκαν αμέσως μετά την πρώτη έκδοση του RSA.

Το αντικείμενο της εργασίας αυτής είναι η διερεύνηση επιθέσεων στον RSA χωρίς την απευθείας παραγοντοποίηση του RSA modulus N. Αξίζει πάντως να αναφερθεί ότι κάποια σύνολα από RSA moduli με μικρό πλήθος στοιχείων, μπορούν να παραγοντοποιηθούν σχετικά εύκολα. Για παράδειγμα αν το (p-1) είναι το γινόμενο πρώτων παραγόντων μικρότερο του B, τότε το N μπορεί να παραγοντοποιηθεί σε χρόνο μικρότερο του B^3**.** Κάποιες υλοποιήσεις του RSA απορρίπτουν σαφώς πρώτους αριθμούς p, στις οποίες το (p-1) είναι γινόμενο μικρών πρώτων αριθμών.

Όπως σημειώθηκε πιο πάνω, αν υπάρχει ένας αποδοτικός αλγόριθμος παραγοντοποίησης, τότε ο RSA είναι μη ασφαλής. Το αντίστροφο είναι ανοιχτό πρόβλημα που προβληματίζει και επιχειρούνται λύσεις εδώ και πολύ καιρό. Πρέπει κάποιος να παραγοντοποιήσει το N ώστε να υπολογίσει την e-οστή ρίζα modulo N; είναι το σπάσιμο του RSA εξίσου δύσκολο πρόβλημα με την παραγοντοποίηση; Το πρόβλημα περιγράφεται σε μαθηματικό φορμαλισμό παρακάτω:

Ανοιχτό πρόβλημα 1. Δοθέντων ακεραίων N και e που ικανοποιούν $\gcd(e,\varphi(N)) = 1$, καθορίστε τη συνάρτηση [19]

$$f_{e,N} : Z_N^* \to Z_N^* \text{ όπου } f_{e,N}(x) = x^{\frac{1}{e}} \bmod N .$$

Υπάρχει πολυωνυμικός αλγόριθμος χρόνου A που υπολογίζει την παραγοντοποίηση του N, δοσμένου του N και πρόσβαση σε ένα «μαντείο» $f_{e,N}(x)$ για κάποιο e;

Ένα «μαντείο» για το $f(x)$ υπολογίζει τη συνάρτηση με οποιαδήποτε είσοδο x σε μοναδιαίο χρόνο. Πρόσφατα ο Boneh και ο Venkatesan [18] παρείχαν στοιχεία ότι για μικρό e η απάντηση στο παραπάνω πρόβλημα μπορεί να είναι 'όχι'. Με άλλα λόγια, με μικρό e μπορεί να μην υπάρχει μια πολυωνυμικού χρόνου μείωση από την παραγοντοποίηση στο σπάσιμο του RSA. Το αποδεικνύουν αυτό με βάση συγκεκριμένο μοντέλο όπου θετική απάντηση στο πρόβλημα των μικρών e παρέχει και έναν αποδοτικό αλγόριθμο παραγοντοποίησης. Σημειώνεται ότι θετική απάντηση στο πρόβλημα 1 δίνει την ευκαιρία για επίθεση επιλεγμένου κρυπτογραφήματος στον RSA. Αυτός είναι και ο λόγος που μια αρνητική απάντηση θα ήταν πιο καλοδεχούμενη.

Στη συνέχεια αποδεικνύεται ότι η ανακάλυψη του δημόσιου κλειδιού d ισοδυναμεί με την παραγοντοποίηση του N για κάθε μέλος που γνωρίζει το d.

Θεώρημα 1. Αν $\langle N, e \rangle$ είναι ένα RSA δημόσιο κλειδί, δοσμένου του ιδιωτικού κλειδιού d, κάποιος μπορεί αποδοτικά να παραγοντοποιήσει το $\text{moduloN} = \text{p} \cdot \text{q}$. Αντίστροφα, δοσμένης της παραγοντοποίησης του N κάποιος μπορεί να βρει αποδοτικά το d.

Απόδειξη: Εφόσον το e είναι γνωστό μπορεί να ανακτηθεί το d. Αυτό αποδεικνύει την αντίστροφη δήλωση. Τώρα θα αποδειχθεί ότι δοσμένου του d, μπορεί να παραγοντοποιηθεί το N. Με δοσμένο το d υπολογίζεται: $k = de - 1$. Από τον ορισμό του d και e είναι γνωστό ότι το k είναι πολλαπλάσιο του φ(N). Άρα εφόσον το φ(N) είναι ζυγός αριθμός, $k = 2^t r$ όπου r περιττός και $t \geq 1$. Έχουμε $g^k = 1$ για κάθε $g \in Z_N^*$ και ως εκ τούτου $g^{\frac{k}{2}}$ είναι τετραγωνική ρίζα του μοναδιαίου moduloN. Με βάση το Κινέζικο θεώρημα Υπολοίπου, το 1 έχει τέσσερις τετραγωνικές ρίζες $\text{moduloN} = \text{p} \cdot \text{q}$. Δύο από αυτές τις ρίζες είναι το ± 1 και οι άλλες δυο είναι το $\pm x$ όπου το x ικανοποιεί το x=1 mod p και το x=-1 mod q. Χρησιμοποιώντας οποιαδήποτε από αυτές τις δυο τελευταίες τετραγωνικές ρίζες, η παραγοντοποίηση του N υπολογίζεται βρίσκοντας το Ε.Κ.Π(x-1, N). Ένα απλό επιχείρημα δείχνει ότι αν το g επιλεχθεί τυχαία στο σύνολο Z_N^*, τότε με πιθανότητα τουλάχιστον ½ (επι της επιλογής του g), ένα από τα στοιχεία στην ακολουθία $g^{\frac{k}{2}}, g^{\frac{k}{4}}, ..., g^{\frac{k}{2^t}} \bmod N$ είναι τετραγωνική ρίζα μονάδος, το οποίο αποδεικνύει την παραγοντοποίηση του N. Όλα τα στοιχεία στην ακολουθία μπορούν να υπολογιστούν αποδοτικά σε χρόνο $O(n^3) \, \text{όπου} \, n = \log_2 N$. ■

4.3 Στοιχειώδεις επιθέσεις

Περιγράφονται οι πιο παλιές στοιχειώδεις επιθέσεις οι οποίες αποδεικνύουν κατάφωρη λάθος χρήση του RSA.[21] Αν και υπάρχουν πολλές τέτοιες επιθέσεις δίνονται μόνο δυο χαρακτηριστικά παραδείγματα.

4.3.1 Κοινό Modulus.

Για να αποφύγει ο αποστολέας να παράγει διαφορετικό $\text{moduloN} = \text{p} \cdot \text{q}$ για κάθε χρήστη, πολύ συχνά παγιώνεται ένα και μόνο N το οποίο χρησιμοποιείται για όλους τους χρήστες. Μια έμπιστη κεντρική αρχή μπορεί να αποδώσει στο χρήστη i το ζεύγος e_i, d_i με τα οποία ο χρήστης i σχηματίζει ένα δημόσιο κλειδί $\langle N, e_i \rangle$ και ένα ιδιωτικό, κρυφό κλειδί $\langle N, d_i \rangle$.

Με μια πρώτη ματιά αυτό το σχήμα δουλεύει: Ένα κρυπτογράφημα $M^{e_a} \bmod N$ που προορίζεται για την Alice δεν μπορεί να αποκρυπτογραφηθεί από τον Bob εφόσον αυτός δεν γνωρίζει το d_a κλειδί της Alice. Παρόλα αυτά, αυτό είναι λάθος και το σύστημα δεν είναι ασφαλές. Ο Bob θα πρέπει να χρησιμοποιήσει τα δικά του εκθετικά e_b, d_b για να παραγοντοποιήσει το moduloN. Όταν παραγοντοποιηθεί το N, ο Bob μπορεί να ανακτήσει το ιδιωτικό κλειδί της Alice d_a από το δημόσιο κλειδί e_a. Αυτή η παρατήρηση του Simmons, δείχνει ότι ένα RSA modulus δεν πρέπει ποτέ να χρησιμοποιείται σε περισσότερες από μια οντότητα.

4.3.2 Τύφλωση

Έστω $\langle N, d \rangle$ το ιδιωτικό κλειδί του Bob και $\langle N, e \rangle$ το [22] αντίστοιχο δημόσιο κλειδί. Έστω επίσης ένας αντίπαλος-επιτιθέμενος Marvin που θέλει την υπογραφή του Bob σε μήνυμα $M \in Z_N^*$ και γνωρίζει ότι ο Bob δε θα υπογράψει το μήνυμα οικειοθελώς, θα δοκιμάσει το εξής: επιλέγει τυχαίο $r \in Z_N^*$ και θέτει $M' = r^e M \bmod N$. Ο Bob Μπορεί να

υπογράψει με S' το μήνυμα M'. Επειδή όμως $S' = (M')^d \bmod N$, ο Marvin υπολογίζει $S = \frac{S'}{r} \bmod N$ και λαμβάνει την υπογραφή του Bob και το αρχικό μήνυμα M. Όντως

$$S^e = \frac{(S')^e}{r^e} = \frac{(M')^{ed}}{r^e} \equiv \frac{M'}{r^e} = M \bmod N .$$

Αυτή η τεχνική που αποκαλείται και τύφλωση επιτρέπει στον Marvin να αποκτήσει έγκυρη υπογραφή σε μήνυμα της επιλογής του ζητώντας από τον Bob να υπογράψει ένα τυχαίο «τυφλό» μήνυμα. Ο Bob στην πραγματικότητα δε γνωρίζει τι μήνυμα υπογράφει. Επειδή τα περισσότερα σχήματα ψηφιακής υπογραφής υλοποιούν μονόδρομη 'hash' συνάρτηση στο μήνυμα M πριν από την υπογραφή [31] η επίθεση δεν αποτελεί μείζον πρόβλημα. Παρότι η «τύφλωση» παρουσιάσθηκε ως κακόβουλη επίθεση, είναι στην πραγματικότητα μια χρήσιμη ιδιότητα του RSA που χρειάζεται για την υλοποίηση ανώνυμου ψηφιακού χρήματος, χρήμα που μπορεί να χρησιμοποιηθεί στην αγορά αγαθών αλλά δεν αποκαλύπτει την ταυτότητα του αγοραστή.

4.4 Μικρός ιδιωτικός εκθέτης, ή μικρό κλειδί

Για να μειωθεί ο χρόνος αποκρυπτογράφησης ή και ο χρόνος παραγωγής υπογραφής, μπορεί να προτιμηθεί μικρό μήκος του d, αντί για τυχαίο d. Εφόσον τμηματικοποιημένη ύψωση σε δύναμη χρειάζεται χρόνο που εξαρτάται γραμμικά από [23] την ποσότητα $\log_2 d$, ένα μικρό d μπορεί να βελτιώσει την επίδοση κατά τουλάχιστον ένα παράγοντα του 10 για modulus 1024bit. Δυστυχώς μια έξυπνη επίθεση που αποδίδεται στον M.Wiener [34] αποδεικνύει ότι μικρό d καταλήγει σε ολική κατάρρευση του κρυπτοσυστήματος.

Θεώρημα 2. (M. Wiener) Έστω N=pq; με q<p<2q, και $d < \frac{1}{3} N^{\frac{1}{4}}$. Δοσμένου ζεύγους $\langle N, e \rangle$ με ed=1modφ(N) ο Marvin μπορεί να ανακτήσει το N επιτυχώς.

Απόδειξη: Η απόδειξη βασίζεται σε προσέγγιση με τη χρήση συνεχόμενων κλασμάτων. Εφόσον ed=1modφ(N) υπάρχει k τέτοιο ώστε ed-kφ(N)=1. Ως εκ τούτου

$$\left| \frac{e}{\varphi(N)} - \frac{k}{d} \right| = \frac{1}{d\varphi(N)}$$

το $\frac{k}{d}$ λοιπόν είναι μια προσέγγιση του $\frac{e}{\varphi(N)}$. Παρότι ο Marvin δεν γνωρίζει το $\varphi(N)$ μπορεί να χρησιμοποιήσει το N για να το προσεγγίσει. Όντως, εφόσον $\varphi(N) = N - p - q + 1 και p + q - 1 < 3\sqrt{N}$ έχουμε $|N - \varphi(N)| \prec 3\sqrt{N}$. Έτσι χρησιμοποιώντας το N αντί του φ(N) λαμβάνουμε: [34]

$$\left| \frac{e}{N} - \frac{k}{d} \right| = \left| \frac{ed - k\varphi(N) - kN + k\varphi(N)}{Nd} \right| = \left| \frac{1 - k(N - \varphi(N))}{Nd} \right| \le \left| \frac{3k\sqrt{N}}{Nd} \right| = \frac{3k}{d\sqrt{N}}$$

Τώρα $k\varphi(\mathrm{N}) = ed - 1 < ed$ και εφόσον e<φ(N) φαίνεται ότι $k < d < \frac{1}{3}N^{\frac{1}{4}}$. Ως εκ τούτου λαμβάνεται : $\left|\frac{e}{N} - \frac{k}{d}\right| \le \frac{1}{dN^{\frac{1}{4}}} < \frac{1}{2d^2}$.

Η παραπάνω είναι μια κλασική σχέση προσέγγισης. Το πλήθος των κλασμάτων k/d με d<N τα οποία προσεγγίζουν το e/N τόσο καλά περιορίζεται από το $\log_2 N$. Στην πραγματικότητα όλα τα παραπάνω κλάσματα λαμβάνονται ως συγκλήσεις της συνεχόμενης ανάπτυξης σε κλάσματα του e/N [24, Th. 177] Το μόνο που χρειάζεται να κάνει κάποιος είναι να υπολογίσει τις $\log_2 N$ συγκλήσεις του συνεχόμενου κλάσματος για e/N. Ένα από αυτά θα ισούται με k/d εφόσον ed-kφ(N)=1, έχουμε lcm(k, d)=1, και το k/d μειωμένο κλάσμα. Ο παραπάνω αλγόριθμος είναι γραμμικού χρόνου για την ανάκτηση του κρυφού κλειδιού d.

Εφόσον το N είναι τυπικά 1024bits, προκύπτει ότι το d θα πρέπει να είναι τουλάχιστον 256bit σε μήκος ώστε να αποφευχθεί η επίθεση αυτή. Αυτό βέβαια δεν είναι καλό για συσκευές χαμηλής κατανάλωσης όπως οι έξυπνες κάρτες, όπου ένα μικρό d θα κατέληγε σε μεγάλη (επεξεργαστική) οικονομία. Πάντως και σε αυτή την περίπτωση υπάρχει λύση, ο Wiener περιγράφει ένα πλήθος από τεχνικές που [24] επιτρέπουν την γρήγορη αποκρυπτογράφηση και δεν είναι ευάλωτες στην συγκεκριμένη επίθεση.

Αν υποθέσουμε ότι αντί να ελαττωθεί το e modulo φ(N), χρησιμοποιείται το $\langle N, e' \rangle$ ως δημόσιο κλειδί όπου $e'e + t \cdot \varphi(\mathrm{N})$ για κάποιο μεγάλο t. Ξεκάθαρα το e' μπορεί να χρησιμοποιηθεί στη θέση του e για κρυπτογράφηση μηνυμάτων. Ωστόσο, όταν χρησιμοποιούμε μια μεγάλη τιμή για το e, το k στην παραπάνω απόδειξη δεν είναι πλέον μικρό. Ένας απλός υπολογισμός δείχνει ότι το αν το $e' > N^{1,5}$, τότε όσο μικρό κι αν είναι το d, η πιο πάνω επίθεση δεν είναι πλέον εφικτή. Δυστυχώς μεγάλες τιμές για το e έχουν σαν αποτέλεσμα μεγάλο χρόνο κρυπτογράφησης.

4.4.1 Χρήση του Κινέζικου Θεωρήματος Υπολοίπου. (CRT).

Έστω ότι επιλέγουμε τέτοιο d ώστε τόσο το $d_p = d \bmod (p-1)$ και $d_p = d \bmod (q-1)$ να είναι μικρά π.χ. 128bit το καθένα. Τότε η γρήγορη αποκρυπτογράφηση ενός κρυπτογραφήματος C εκτελείται ως εξής: Πρώτα υπολογίζεται το $u_1,...,u_w$ και $Mq = Cq \bmod q$. Εν συνεχεία χρησιμοποιείται το CRT ώστε να υπολογίσει τη μοναδική τιμή του $M \in Z_N$ που να ικανοποιεί

$$M = M_p \bmod p$$
$$M = M_q \bmod q .$$

Το λαμβανόμενο M ικανοποιεί το $M = C^d \bmod N$ όπως και αν απαιτηθεί. Το νόημα είναι πως παρότι τα d_p και d_q είναι μικρά, η τιμή του d mod φ(N) μπορεί να είναι μεγάλη. Για παράδειγμα, να είναι της τάξης του φ(N). Ως αποτέλεσμα, η επίθεση στο δεύτερο θεώρημα δεν υφίσταται πλέον. Σημειώνεται πως αν δίνεται το ζεύγος $\langle N, e \rangle$, υπάρχει επίθεση που επιτρέπει στον αντίπαλο-επιτιθέμενο να παραγοντοποιήσει το N σε [26] χρόνο $O(\min(\sqrt{d_p}, \sqrt{d_q}))$ Συνεπώς τα d_p και d_q δεν μπορεί να είναι πολύ μικρά.

Δεν είναι γνωστό αν κάποια από τις δυο προτεινόμενες μεθόδους είναι ασφαλής. Το δεύτερο θεώρημα πολύ πρόσφατα βελτιώθηκε από τους Boneh και Durfee [16] οι οποίοι αποδεικνύουν ότι όσο το $d<N^{0,292}$, ένας επιτιθέμενος μπορεί εύκολα να ανακτήσει το d από το $\langle N,e\rangle$. Αυτά τα αποτελέσματα μας δείχνουν ότι το όριο του Wiener δεν είναι σφιχτό (επαρκώς κλειστό). Είναι πιο πιθανό το σωστό όριο να είναι $d<N^{0,5}$.

Ανοιχτό πρόβλημα 2. Έστω N=pq και $d<N^{0,5}$. Αν στο Marvin δοθεί $\langle N,e\rangle$ με ed=1 mod φ(N) και e<φ(N), μπορεί να ανακτήσει αποδοτικά το d;

4.5 Μικρός δημόσιος εκθέτης ή μικρό δημόσιο εκθετικό κλειδί

Για να μειωθεί ο χρόνος κρυπτογράφησης ή επιβεβαίωσης υπογραφής, είναι σύνηθες να χρησιμοποιείται ένα μικρό δημόσιο εκθετικό e. Η μικρότερη δυνατή τιμή του e είναι το 3, αλλά για να ισχυροποιηθεί απέναντι στις συγκεκριμένες επιθέσεις προτείνεται η τιμή $e = 2^{16} +1= 65537$. Όταν χρησιμοποιείται η τιμή$2^{16} +1$, η επαλήθευση της ψηφιακής υπογραφής απαιτεί 17 πολλαπλασιασμούς σε αντίθεση με τους σχεδόν χίλιους πολλαπλασιασμούς που απαιτούνται με τυχαίο $e \le \varphi(N)$.

Σε αντίθεση με την επίθεση που εξετάσαμε στο προηγούμενο κεφάλαιο, οι επιθέσεις που εφαρμόζονται όταν χρησιμοποιούμε μικρό e απέχουν παρασάγγας από το ολοκληρωτικό σπάσιμο του RSA.

4.5.1 Το θεώρημα του Coppersmith.

Οι πιο ισχυρές επιθέσεις σε μικρό εκθετικό RSA βασίζονται στο θεώρημα του Coppersmith [19] το οποίο έχει πολλές εφαρμογές κατά του RSA. Η απόδειξη του θεωρήματος το Coppersmith χρησιμοποιεί τον αλγόριθμο LLL που ουσιαστικά πετυχαίνει τη μείωση της βάσης πίνακα όπως εξηγείται παρακάτω.

Θεώρημα 3. Έστω N ακέραιος και $f \in Z[x]$ πολυώνυμο βαθμού d. Θέτουμε $X = N^{\frac{1}{d}-\varepsilon}$ για κάποιο $\varepsilon \ge 0$. Τότε δοσμένου $\langle N,f\rangle$ ο Marvin μπορεί να βρει όλους τους ακεραίους $|x_0| < X$ που ικανοποιούν την $f(x_0) = 0 \bmod N$. Ο χρόνος εκτέλεσης καθορίζεται από το χρόνο που χρειάζεται να τρέξει ο LLL αλγόριθμος σε πίνακα / μήτρα διαστάσεων O(w) με $w = \min(\frac{1}{\varepsilon}, \log_2 N)$.

Το θεώρημα παρέχει έναν αλγόριθμο με τον οποίο μπορούν να βρεθούν αποδοτικά όλες οι ρίζες του f modulo N που είναι λιγότερο από $X = N^{1/d}$. Καθώς το X μικραίνει, ο χρόνος εκτέλεσης του αλγόριθμου μικραίνει. Η δύναμη του θεωρήματος είναι η ικανότητα του να εντοπίζει μικρές ρίζες πολυωνύμων με modulo ένα σύνθετο N. Όταν το ενεργό modulo είναι πρώτος αριθμός, δεν υπάρχει λόγος να χρησιμοποιηθεί το θεώρημα του Coppersmith αφού υπάρχουν πολύ καλύτεροι αλγόριθμοι εύρεσης ριζών.

Σκιαγραφώντας τις βασικές ιδέες πίσω από την απόδειξη του θεωρήματος του Coppersmith, πρέπει πρώτα από όλα να ακολουθηθεί μια απλουστευμένη προσέγγιση λόγω των Howgrave-Graham [26]. Δοσμένου πολυωνύμου

$$h(x) = \sum a_i x^i \in \mathbf{Z}[x]$$

καθόρισε $||h||^2 = \sum_i |a_i|^2$. Η απόδειξη βασίζεται στην ακόλουθη παρατήρηση:

Λήμμα 4. Έστω $h(x) \in \mathbf{Z}[x]$ πολυώνυμο βαθμού d και έστω X ένας θετικός ακέραιος. Έστω $||h(x)|| < n/\sqrt{d}$. Αν $|x_0| < X$ ικανοποιεί τη $h(x_0) < 0 \bmod N$ τότε $h(x_0) = 0$ για όλους τους ακεραίους.

Απόδειξη: Από την ανισότητα Schwarz

$$|h(x_0)| = \left|\sum a_i x_0^i\right| = \left|\sum a_i X^i \left(\frac{x_o}{X}\right)^i\right| \le \left|\sum a_i X^i \left(\frac{x_o}{X}\right)^i\right| \le \sum \left|a_i X^i \left(\frac{x_0}{X}\right)^i\right|$$
$$\le \sum \left|a_i X^i\right| \le \sqrt{d}\,\|h(xX)\| < N$$

Αφού $h(x_0) = 0 \bmod N$ συμπεραίνεται ότι : $h(x_0) = 0$ ■

Το λήμμα δηλώνει ότι αν το h είναι πολυώνυμο με χαμηλή νόρμα, τότε όλες οι μικρές ρίζες του h mod N είναι επίσης ρίζες του η επί των ακεραίων. Το λήμμα προτείνει ότι για να βρεθεί μια μικρή ρίζα x_0 του f(x) mod N, πρέπει να βρεθεί άλλο πολυώνυμο $h(x) \in \mathbf{Z}[x]$ με μικρή νόρμα το οποίο θα έχει τις ίδιες ρίζες με το f modulo N . Τότε το x_0 θα είναι ρίζα του h επί των ακεραίων και μπορεί να βρεθεί εύκολα. Για να γίνει αυτό, αναζητείται πολυώνυμο $g \in \mathbf{Z}[x]$ ώστε το $h = gf$ να έχει μικρή νόρμα, δηλαδή μικρότερη από N. Αυτό καταλήγει στην αναζήτηση ενός ακέραιου γραμμικού συνδυασμού των f, xf, x^2f,…, x^rf με χαμηλή νόρμα. Δυστυχώς τις περισσότερες φορές δεν υπάρχει ένας μη ασήμαντος γραμμικός συνδυασμός με επαρκώς μικρή νόρμα.

Ο Coppersmith επινόησε ένα απλό τρικ για να λύσει το πρόβλημα αυτό : Αν $f(x_0) = 0 \bmod N$, τότε $f(x_0)^k = 0 \bmod N^k$ για κάθε k . Πιο γενικά καθορίζοντας τα ακόλουθα πολυώνυμα :

$$g_{u;v}(x) = N^{m-v} x^u f(x)^v$$

για κάποιο προκαθορισμένο m. Τότε το x_0 είναι ρίζα του $g_{u;v}(x)$ modulo N^m

για κάθε u≥0, 0≤v≤m. Για να χρησιμοποιηθεί το Λήμμα 4 πρέπει να βρεθεί ένας ακέραιος γραμμικός συνδυασμός h(x) των πολυωνύμων $g_{u;v}(x)$ ώστε το h(xX) να έχει νόρμα μεγαλύτερη από N^m (υπενθυμίζεται ότι το X είναι άνω όριο στο x_0 ικανοποιώντας την $X \le N^{1/d}$). Χάρη στο χαλαρό άνω όριο στη νόρμα (N^m αντί N), μπορεί να αποδειχτεί ότι για ικανοποιητικά μεγάλο m, πάντα υπάρχει ένας γραμμικός συνδυασμός h(x) που ικανοποιεί το επιθυμητό όριο. Όταν βρεθεί το h(x), το Λήμμα 4 υπονοεί ότι το x_0 έχει ρίζα επί των ακεραίων και ως εκ τούτου μπορεί να βρεθεί εύκολα.

Απομένει να βρεθεί πώς να υπολογιστεί το h(x) αποδοτικά. Για να γίνει αυτό πρέπει πρώτα να παρατεθούν μερικά γεγονότα σχετικά με τους πίνακες στο χώρο $\mathbb{Z}^w$. Πιο ακριβής εισαγωγή στο αντικείμενο φαίνεται στο [29]. Αν τα $u_1,...,u_w \in \mathbf{Z}^w$ είναι γραμμικά

ανεξάρτητα, ένας πίνακας L ο οποίος αποτελείται από γραμμικά ανεξάρτητα διανύσματα $\langle u_1,...,u_w \rangle$ είναι επί της ουσίας το σύνολο όλων των γραμμικών συνδυασμών των $u_1,...,u_w$. Η ορίζουσα του L καθορίζεται ως η ορίζουσα του τετραγωνικού πίνακα μεγέθους w*w του οποίου οι γραμμές είναι τα διανύσματα $u_1,...,u_w$.

Στη περίπτωση που εξετάζεται, τα πολυώνυμα $g_{u,v}(xX)$ λαμβάνονται ως διανύσματα που συμπληρώνουν το πίνακα L. Έστω v=0,…,m και u=0,…d-1. Ως εκ τούτου, ο πίνακας έχει διαστάσεις w=d(m+1). Για παράδειγμα αν το f είναι πολυώνυμο τετάρτου βαθμού, και το m=3 ο προκύπτον πίνακας έχει τη μορφή :

$$
\begin{array}{c}
 \\
g_{0,0}(xX) \\
g_{1,0}(xX) \\
g_{0,1}(xX) \\
g_{1,1}(xX) \\
g_{0,2}(xX) \\
g_{1,2}(xX) \\
g_{0,3}(xX) \\
g_{1,3}(xX)
\end{array}
\begin{array}{c}
\begin{array}{cccccccc} 1 & x & x^2 & x^3 & x^4 & x^5 & x^6 & x^7 \end{array} \\
\left[\begin{array}{cccccccc}
N^3 & & & & & & & \\
 & XN^3 & & & & & & \\
* & * & X^2N^2 & & & & & \\
 & * & * & X^3N^2 & & & & \\
* & * & * & * & X^4N & & & \\
 & * & * & * & * & X^5N & & \\
* & * & * & * & * & * & X^6 & \\
 & * & * & * & * & * & * & X^7
\end{array}\right]
\end{array}
$$

Οι καταχωρήσεις με * αντιστοιχούν σε συντελεστές των πολυωνύμων των οποίων η τιμή αγνοείται εσκεμμένα. Όλες οι κενές καταχωρήσεις είναι μηδενικά. Αφού ο πίνακας είναι τριγωνικός, η ορίζουσα του ισούται με το γινόμενο των στοιχείων της διαγωνίου (τα οποία φαίνονται καθαρά στη προηγούμενη σελίδα). Ο αντικειμενικός σκοπός είναι να βρεθούν συντετμημένα διανύσματα σε αυτόν το πίνακα.

Ένα κλασσικό αποτέλεσμα της Ερμιτιανής άλγεβρας δηλώνει ότι οποιοσδήποτε πίνακας L, διαστάσεων w περιέχει μη μηδενικό στοιχείο $v \in L$ του οποίου η L_2 ικανοποιεί τη σχέση : $||v|| \leq \gamma_w \det(L)^{1/w}$ όπου το γ_w είναι σταθερά εξαρτώμενη μόνο από το w. Το ερμιτιανό όριο μπορεί να χρησιμοποιηθεί ώστε να αποδειχθεί ότι για αρκετά μεγάλο m, ο πίνακας περιέχει διανύσματα με νόρμα (μέτρο) λιγότερο από N^m που απαιτείται. Η ερώτηση είναι εάν δύναται να κατασκευαστεί ένα συντετμημένο διάνυσμα στο L του οποίου το μήκος να μην είναι πολύ μεγαλύτερο από το ερμιτιανό όριο. Ο αλγόριθμος LLL επιτελεί ακριβώς αυτό.

Ορισμός 5. (LLL) Έστω L ο πίνακας που συμπληρώνεται από τα $\langle u_1,...,u_w \rangle$. Όταν τα $\langle u_1,...,u_w \rangle$ δίνονται ως είσοδοι ο αλγόριθμός LLL δίνει σημείο $v \in L$ που ικανοποιεί: $||v|| \leq 2^{w/4} \det(L)^{1/w}$

Ο χρόνος εκτέλεσης του LLL είναι το μήκος εισόδου στη τετάρτη. Ο αλγόριθμος LLL (που ονομάστηκε από τους δημιουργούς του, L. Lovasz, A. Lenstra και H. Lenstra Jr.) έχει πολλές εφαρμογές τόσο στην υπολογιστική θεωρία αριθμών όσο και στη κρυπτογραφία. Η ανακάλυψη του το 1982 έδωσε έναν αποδοτικό αλγόριθμο για τη παραγοντοποίηση πολυωνύμων επί των ακεραίων και γενικότερα, δακτυλίων αριθμών. Ο LLL χρησιμοποιείται συχνά σε επιθέσεις εναντίων ποικίλων κρυπτοσυστημάτων. Για

παράδειγμα, πολλά κρυπτοσυστήματα που βασίζονται στο πρόβλημα του «σακιδίου» (knapsack), έσπασαν μέσω του LLL.

Χρησιμοποιώντας τον LLL είναι δυνατό να ολοκληρωθεί η απόδειξη του θεωρήματος του Coppersmith. Για να διασφαλιστεί ότι το διάνυσμα που παράγεται από το LLL ικανοποιεί το όριο του λήμματος 4 πρέπει να ισχύει :

$$2^{w/4}\det(L)^{1/w} \le N^m / \sqrt{w}$$

όπου w=d(m+1) είναι η διάσταση του πίνακα L. Ένας συνήθης υπολογισμός δείχνει ότι για αρκετά μεγάλο m, το όριο ικανοποιείται. Πράγματι όταν $X = N^{\frac{1}{d}-\varepsilon}$, αρκεί να ληφθεί m=O(k/d) με k=min(1/ε, logN). Συνεπώς ο χρόνος εκτέλεσης του αλγόριθμου καθορίζεται από την εφαρμογή του LLL σε πίνακα διαστάσεων O(k) όπως απαιτείται.

Μια φυσική ερώτηση είναι αν το θεώρημα του Coppersmith μπορεί να εφαρμοστεί σε πολυώνυμα δύο ή και περισσοτέρων μεταβλητών. Αν δίνεται $f(x,y) \in \mathbf{Z}_N[x,y]$ για την οποία υπάρχει (x_0, y_0) με το $|x_0y_0|$ καταλλήλως φραγμένο, θα μπορέσει ο επιτιθέμενος να βρει αποδοτικά το (x_0, y_0); Παρότι η τεχνική LLL φαίνεται να ισχύει και σε πολυώνυμα δύο μεταβλητών είναι επί του παρόντος η σχετική απόδειξη ένα ανοικτό πρόβλημα. Καθώς ένα αυξημένο πλήθος από αποτελέσματα εξαρτάται από την επέκταση του θεωρήματος του Coppersmith σε δυο μεταβλητές, ένας αυστηρός σχετικός αλγόριθμος θα αποδειχτεί πολύ χρήσιμος.

Ανοικτό Πρόβλημα 3. Η εύρεση γενικών συνθηκών υπό τις οποίες το θεώρημα του Coppersmith μπορεί να επεκταθεί σε πολυώνυμα δυο μεταβλητών.

4.5.2 Επίθεση Ευρυεκπομπής του Hastad

Ως πρώτη εφαρμογή του θεωρήματος Coppersmith, παρουσιάζεται πιο κάτω μια εξέλιξη παλαιότερης επίθεσης από τον Hastad [25]. Έστω ότι ο Bob επιθυμεί να στείλει ένα κρυπτογραφημένο μήνυμα M σε ένα σύνολο από αποδέκτες $P_1,P_2,\ldots,P_k$. Κάθε συμμετέχων διαθέτει το δικό του κλειδί RSA $\langle N_i, e_i \rangle$. Υποθέτουμε ότι το M είναι μικρότερο από όλα τα N_i . Ο Bob, αφελώς για να στείλει το μήνυμα M, το κρυπτογραφεί χρησιμοποιώντας το καθένα από τα δημόσια κλειδιά και στέλνει το i-οστό κρυπτογράφημα στο παραλήπτη P_i .

Ο επιτιθέμενος Marvin μπορεί να κρυφακούσει στη σύνδεση εκτός του βλέμματος του αποστολέα Bob και να συλλέξει τα k μεταδομένα κρυπτογραφήματα. Για απλότητα, έστω ότι όλοι οι δημόσιοι εκθέτες e_i ισούνται με 3. Ένα απλό επιχείρημα δείχνει ότι ο Marvin μπορεί να ανακτήσει το M αν k≥3. Όντως ο Marvin αποσπά τα C_1, C_2, C_3 όπου:

$$C_1 = M^3 \bmod N_1$$
$$C_2 = M^3 \bmod N_2$$
$$C_3 = M^3 \bmod N_3$$

Μπορεί να υποτεθεί ότι ΕΚΠ(N_i,N_j)=1 για κάθε i≠j αφού διαφορετικά ο Marvin μπορεί να παραγοντοποιήσει κάποια από τα N_i . Ως εκ τούτου, εφαρμόζοντας το Κινέζικο Θεώρημα Υπολοίπου (CRT) στα C_1, C_2, C_3 , λαμβάνεται $C' \in \mathbf{Z}_{N_1N_2N_3}$. Τότε $C'=M^3 \bmod N_1N_2N_3$. Εφόσον το M είναι μικρότερο των N_i λαμβάνεται $M^3<N_1N_2N_3$ και το $C'=M^3$ υπερισχύει στους ακεραίους. Ο επιτιθέμενος λοιπόν μπορεί να ανακτήσει το M

υπολογίζοντας τη πραγματική ρίζα τρίτου βαθμού του C΄. Πιο γενικά, αν όλα τα δημόσια εκθετικά (κλειδιά) είναι ίσα με e , ο επιτιθέμενος μπορεί να ανακτήσει το M μόλις k≥e. Η επίθεση αυτή είναι δυνατή μόνο όταν χρησιμοποιείται μικρό e.

Ο Hastad [25] περιγράφει μια πολύ ισχυρότερη επίθεση. Για να φανούν τα αποτελέσματα της επίθεσης αυτής θα ληφθεί υπόψη μια απλοϊκή άμυνα έναντι αυτής. Αντί να ευρυεκπέμψει το κρυπτογράφημα M, ο αποστολέας θα μπορούσε να «παραγεμίσει» το μήνυμα πριν από τη κρυπτογράφηση. Για παράδειγμα, αν το M είναι m bit σε μήκος, ο αποστολέας θα μπορούσε να στείλει $\mathrm{M}_i = i2^m + M$ στο παραλήπτη P_i. Αφού ο επιτιθέμενος αποκτά κρυπτογραφήσεις διαφορετικών μηνυμάτων, δεν μπορεί να ξεκινήσει την επίθεση. Δυστυχώς όμως, ο Hastad απέδειξε ότι η τεχνική αυτή γραμμικού παραγεμίσματος είναι επισφαλής και στην πραγματικότητα, απέδειξε ότι η εφαρμογή οποιουδήποτε σταθερού πολυωνύμου στο μήνυμα πριν από την κρυπτογράφηση δεν αποτρέπει την επίθεση.

Έστω ότι για το καθένα από τους συμμετέχοντες $P_1,\ldots,P_k$, ο αποστολέας έχει ένα σταθερό δημόσιο πολυώνυμο $f_i \in \mathbf{Z}_{N_i}[x]$. Για να ευρυμεταδώσει ένα μήνυμα M ο αποστολέας στέλνει τη κρυπτογράφηση του $f_i(M)$ στο μέλος P_i . Υποκλέπτοντας, ο επιτιθέμενος μαθαίνει το $C_i = f_i(M)^{e_i} \bmod N_i$ για i=1,2,…,k. Ο Hastad έδειξε ότι μπλέκονται πολλοί συμμετέχοντες, ο επιτιθέμενος μπορεί να ανακτήσει το αρχικό κείμενο M από όλα τα κρυπτογραφήματα. Το θεώρημα που ακολουθεί είναι μια ισχυρότερη εκδοχή του αρχικού αποτελέσματος του Hastad:

Θεώρημα 6 (Hastad) Αν $N_1,\ldots N_k$ είναι ταιριασμένοι σχετικά πρώτοι ακέραιοι και τεθεί $N_{\min} = \min_i N_i$ .. Αν $g_i \in \mathbf{Z}_{N_i}[x]$ k το πλήθος πολυώνυμα μέγιστου βαθμού d. Έστω ότι υπάρχει μοναδικό $M < N_{\min}$ που ικανοποιεί την :

$$g_i(M) = 0 \bmod N_i \text{ για όλα τα i=1,…,k}$$

Με την υπόθεση πως k>d, είναι δυνατό να βρεθεί το M δοσμένου του $\langle N_i, g_i \rangle_{i=1}^{k}$.

Απόδειξη: Έστω $\bar{N} = N_1 \ldots N_k$ και γίνεται υπόθεση ότι όλα τα g_i ;έχουν συντελεστή στη μεγαλύτερη δύναμη τους τη μονάδα. Πολλαπλασιάζοντας κάθε g_i με το κατάλληλο x, είναι δυνατό να υποτεθεί πως όλα έχουν βαθμό d. Κατασκευάζεται εν συνεχεία το πολυώνυμο

$$g(x) = \sum_{i=1}^{k} T_i g_i(x) \text{ όπου } T_i = \begin{matrix} 1 \bmod N_j & i = j \\ 0 \bmod N_j & i \neq j \end{matrix}$$

Τα T_i είναι ακέραιοι γνωστοί ως συντελεστές του Κινέζικου Υπόλοιπου. Το g(x) έχει πρώτο συντελεστή στη μεγαλύτερη δύναμη το 1 και ο βαθμός του είναι d. Επιπλέον είναι γνωστό ότι $g(M) = 0 \bmod \bar{N}$. Το Θεώρημα 6 με βάση το Θεώρημα 3 συνεχίζει λαμβάνοντας υπόψη ότι $M < N_{\min} \le \bar{N}^{1/k} < \bar{N}^{1/d}$

Το θεώρημα αποδεικνύει ότι ένα σύστημα από μονοπαραγοντικές εξισώσεις modulo σχετικά πρώτων αριθμών μπορεί να λυθεί αποδοτικά υποθέτοντας ότι είναι διαθέσιμο ένα αρκετά μεγάλο πλήθος εξισώσεων. Θέτοντας $g_i = f_i^{e_i} - C_i \bmod N_i$ ο επιτιθέμενος μπορεί να ανακτήσει το κρυπτογράφημα από τα δοσμένα κρυπτογραφήματα οποτεδήποτε το πλήθος των συμμετεχόντων είναι τουλάχιστον d που είναι το μέγιστο του $e_i \deg(f_i)$ για όλα τα i=1,…,k. Πιο συγκεκριμένα, αν όλα τα e_i είναι ίσα με e και ο

αποστολέας γραμμικά συσχετισμένα μηνύματα, τότε ο επιτιθέμενος μπορεί να εκμαιεύσει το αρχικό μήνυμα μόλις k>e.

Το αρχικό θεώρημα του Hastad είναι ασθενέστερο από το πιο πάνω. Αντί d σε πλήθος πολυώνυμα, ο Hastad απαιτούσε d(d+1)/2 πολυώνυμα. Η απόδειξη του Hastad μοιάζει με αυτή του Coppersmith αλλά επειδή δεν χρησιμοποιεί δυνάμεις του g στο πίνακα, καταλήγει σε πιο ασθενές όριο.

Για να ολοκληρωθεί το κεφάλαιο αυτό, πρέπει να σημειωθεί ότι για να υπάρχει σωστή άμυνα στην παραπάνω επίθεση ευρυμετάδοσης, πρέπει να χρησιμοποιηθεί παραγέμισμα από τυχαία επιλεγμένους αριθμούς αντί σταθερών. [13]

4.5.3 Επίθεση Franklin-Reiter Σχετικού Μηνύματος

Οι Franklin και Reiter [20] βρήκαν μια έξυπνη επίθεση στη περίπτωση που αποστολέας στέλνει στο παραλήπτη μηνύματα με το ίδιο κάθε φορά modulus. Έστω $\langle N,e\rangle$ το δημόσιο κλειδί του παραλήπτη, και $M_1,M_2 \in \mathbf{Z}_N^*$ δυο διακριτά μηνύματα που ικανοποιούν $M_1 = f(M_2) \bmod N$ για κάποιο δημόσια γνωστό πολυώνυμο $f \in \mathbf{Z}_N[x]$. Για να στείλει τα M_1 και M_2 ο αποστολέας στο παραλήπτη, αφελώς κρυπτογραφεί τα μηνύματα και στέλνει τα C_1, C_2. Αποδεικνύεται πιο κάτω ότι με τα δοσμένα C_1, C_2 στη κατοχή του ο επιτιθέμενος μπορεί εύκολα να ανακτήσει τα M_1 και M_2 . Παρότι η επίθεση αυτή λειτουργεί σωστά για οποιοδήποτε μικρό e, παρατίθεται το παρακάτω λήμμα για e=3 ώστε να απλουστευθεί η απόδειξη.

Λήμμα 7. Τίθεται e=3 και έστω ότι το $\langle N,e\rangle$ είναι ένα RSA δημόσιο κλειδί. Έστω ότι τα $M_1 \neq M_2 \in \mathbf{Z}_N^*$ ικανοποιούν τη $M_1 = f(M_2) \bmod N$ για κάποιο γραμμικό πολυώνυμο της μορφής $f = ax + b \in \mathbf{Z}_N[x]$ με $b \neq 0$. Τότε δοσμένων των $\langle N,e,C_1,C_2,f\rangle$, τότε ο επιτιθέμενος μπορεί να ανακτήσει τα M_1,M_2 σε χρόνο ανάλογο της τέταρτης δύναμης του logN.

Απόδειξη: Για να έχει αυτό το κομμάτι της απόδειξης γενικότερη εμβέλεια, χρησιμοποιείται αυθαίρετο e (αντί e=3). Αφού $C_1 = M_1^e \bmod N$, είναι γνωστό ότι το M_2 είναι ρίζα του $g_2(x) = x^e - C_2 \in \mathbf{Z}_N[x]$. Ο γραμμικός παράγοντας x-M_2 διαιρεί και τα δυο πολυώνυμα . Γι αυτό ο επιτιθέμενος μπορεί να χρησιμοποιήσει τον Ευκλείδειο αλγόριθμο για να υπολογίσει το Π.Κ.Π των g_1 και g_2. Αν το Π.Κ.Π βρεθεί γραμμικό, υπολογίζεται το M_2. Το Π.Κ.Π υπολογίζεται σε χρόνο που είναι τέταρτη δύναμη του e και logN.

Αποδεικνύεται πως όταν e=3 το Π.Κ.Π πρέπει να είναι γραμμικό. Το πολυώνυμο x^3-C_2 παραγοντοποιεί κατά modulo τόσο τα p και q κατά γραμμικό παράγοντα και ένα μη απλοποιούμενο παράγοντα τέταρτης δύναμης (υπενθυμίζεται ότι Π.Κ.Π(e,φ(N))=1 και ως εκ τούτου το x^3-C_2 έχει μόνο μια ρίζα στο $\mathbf{Z}_N$). Αφού το g_2 δεν μπορεί να διαιρέσει το g_1, το Π.Κ.Π πρέπει να είναι γραμμικό. Για e>3 το Π.Κ.Π είναι σχεδόν πάντα γραμμικό. Παρόλα αυτά, για κάποια σπάνια M_1,M_2 και f είναι δυνατό να αποκτηθεί μη γραμμικό Π.Κ.Π οπότε και η επίθεση αποτυγχάνει.

Για e>3 η επίθεση χρειάζεται χρόνο που είναι ανάλογος της τέταρτης δύναμης του e. Συνεπώς μπορεί να εφαρμοστεί μόνο όταν χρησιμοποιείται μικρό δημόσιο κλειδί. Για μεγάλο e ο υπολογισμός του Π.Κ.Π είναι απαγορευτικός. Είναι λοιπόν ενδιαφέρον το ερώτημα (αν και δύσκολο να συμβεί) να επινοηθεί τέτοια επίθεση για αυθαίρετο e. Πιο

συγκεκριμένα, είναι δυνατό το Π.Κ.Π των g_1 και g_2 να βρεθεί σε χρόνο που είναι πολυώνυμο του loge;

4.5.4 Επίθεση μικρού παραγεμίσματος του Coppersmith

Η επίθεση Franklin-Reiter μπορεί να φαίνεται λίγο τεχνητή. Εξάλλου, γιατί ο αποστολέας να στέλνει στο παραλήπτη τη κρυπτογράφηση σχετικών μηνυμάτων; Ο Coppersmith ενίσχυσε την επίθεση και ανέδειξε ένα σημαντικό εύρημα που αφορά τη παραγέμιση [19].

Ένας απλοϊκός αλγόριθμος παραγέμισης απλώς θα συμπληρώσει το αρχικό κείμενο με τη προσθήκη μερικών τυχαίων bit στο ένα από τα δυο άκρα. Η ακόλουθη επίθεση υποδεικνύει το κίνδυνο από τη χρήση τέτοιων επικίνδυνων τεχνικών παραγεμίσματος. Έστω ότι ο αποστολέας στέλνει ένα παραγεμισμένο και εν συνεχεία κρυπτογραφημένο αρχικό μήνυμα M στο παραλήπτη. Ο επιτιθέμενος αναχαιτίζει το κρυπτογράφημα και δεν του επιτρέπει να φτάσει στο παραλήπτη. Ο αποστολέας παρατηρεί ότι το μήνυμα του δεν έχει φτάσει και αποφασίζει να το ξαναστείλει. Παραγεμίζει λοιπόν το αρχικό μήνυμα M ξανά, και αφού το κρυπτογραφήσει, το ξαναστέλνει. Ο επιτιθέμενος έχει πλέον στη διάθεση του δυο κρυπτογραφήματα του ίδιου αρχικού μηνύματος τα οποία έχουν δυο διαφορετικά τυχαία παραγεμίσματα. Το παρακάτω θεώρημα δείχνει ότι ο επιτιθέμενος μπορεί υπό αυτές τις συνθήκες να βρει το αρχικό μήνυμα.

Θεώρημα 8. Έστω $\langle N,e\rangle$ το δημόσιο RSA κλειδί όπου το N είναι μήκους n-bit. Τίθεται $m=\left\lfloor \frac{n}{e^2} \right\rfloor$. Έστω $M \in \mathbf{Z}_N^*$ μήνυμα μεγέθους το πολύ n-m bit . Ορίζεται $M_1 = 2^m M + r_1$ και $M_2 = 2^m M + r_2$ όπου r_1 και r_2 είναι διακριτοί ακέραιοι με $0 < r_1, r_2 < 2^m$. Αν στον επιτιθέμενο δοθεί το $\langle N,e\rangle$ και τα κρυπτογραφήματα C_1, C_2 των M_1, M_2 (αλλά όχι τα r_1, r_2), μπορεί να ανακτήσει αποδοτικά το M.

Απόδειξη: Ορίζεται $g_1(x,y) = x^e - C_1$ και $g_2(x,y) = (x+y)^e - C_2$. Είναι γνωστό πως όταν $y = r_2 - r_1$, τα πολυώνυμα αυτά έχουν κοινή ρίζα το M_1. Με άλλα λόγια, το $\Delta = r_2 - r_1$ είναι ρίζα του «καταληκτικού» $h(y) = res_x(g_1, g_2) \in \mathbf{Z}_N[y]$. Ο βαθμός του h είναι το πολύ e^2. Επιπλέον $|\Delta| < 2^m < N^{\frac{1}{e^2}}$ και συνεπώς το Δ είναι μικρή ρίζα του h modulo N και ο επιτιθέμενος μπορεί να το βρει εύκολα χρησιμοποιώντας το θεώρημα του Coppersmith. Όταν το Δ γίνει γνωστό, η επίθεση Franklin-Reiter μπορεί να χρησιμοποιηθεί στην εύρεση του M_2 και εν συνεχεία του M.

Όταν e=3, η επίθεση μπορεί να εκκινήσει όσο το μήκος του παραγεμίσματος είναι μικρότερο από το 1/9 του μηνύματος. Αυτό το αποτέλεσμα είναι σημαντικό – για τη προτεινόμενη τιμή του e=65537, η επίθεση είναι άχρηστη για τα συνήθη μεγέθη modulo.

4.5.5 Επίθεση Αποκάλυψης Μέρος Κλειδιού

Έστω $\langle N,d\rangle$ ένα ιδιωτικό RSA κλειδί. Αν με κάποιο τρόπο υποτεθεί ότι είναι δυνατό ένας επιτιθέμενος να αποκαλύψει κάποια από τα bit του κλειδιού d –για παράδειγμα το ένα τέταρτο του συνολικού πλήθους από bit, είναι δυνατό να εκμαιεύσει όλο το κλειδί d; Η απάντηση είναι θετική όταν το αντίστοιχο δημόσιο κλειδί έχει μικρό μήκος. Πρόσφατα οι Boneh, Durfee και Frankel [17] απέδειξαν ότι όσο ισχύει $e < \sqrt{N}$, είναι δυνατό να

ανακατασκευαστεί ολόκληρο το d με κλάσμα μόνο από τα bit του. Τα αποτελέσματα αυτά αναδεικνύουν τη σημασία της ασφαλούς φύλαξης ολόκληρου του κρυφού κλειδιού RSA.

Θεώρημα 9 (BDF) Έστω $\langle N,d \rangle$ ένα ιδιωτικό RSA κλειδί στο οποίο το N είναι μήκους n bit . Δοσμένων των $\lfloor n/4 \rfloor$ λιγότερο σημαντικών bit του d, ο επιτιθέμενος μπορεί να ανακατασκευάσει ολόκληρο το κλειδί σε χρόνο γραμμικά ανάλογο του $e\log_2 e$.

Η απόδειξη του παραπάνω για μια ακόμη φορά βασίζεται σε ένα όμορφο θεώρημα του Coppersmith [19].

Θεώρημα 10 (Coppersmith) Έστω $N = pq$ ένα RSA modulus μήκους n-bit. Τότε δοσμένων των $n/4$ λιγότερο σημαντικών bit του p ή των $n/4$ περισσότερο σημαντικών bit του p, ένας επιτιθέμενος μπορεί επιτυχώς να παραγοντοποιήσει το N.

Από ορισμό των e και d, υπάρχει ακέραιος που δίνει:

$$ed - k(N - p - q + 1) = 1$$

Αφού $d < \varphi(\mathrm{N})$ πρέπει να ισχύει $0 < k \le e$. Μειώνοντας το modulo της εξίσωσης $2^{n/4}$ και θέτοντας $q = N/p$ λαμβάνεται:

$$(ed)p - kp(N - p + 1) + kN = p(\mathrm{mod}\, 2^{n/4}).$$

Εφόσον ο επιτιθέμενος κατέχει τα n/4 λιγότερο σημαντικά bit του d, γνωρίζει τη τιμή του $ed \bmod 2^{n/4}$ και συνεπώς μπορεί να αποκτήσει μια συνάρτηση με τα k , p. Για κάθε μια από τις e πιθανές τιμές του k, ο επιτιθέμενος λύνει τη τετραβάθμια εξίσωση του p και αποκτά ένα πλήθος από υποψήφιες τιμές για το $p \bmod 2^{n/4}$. Για κάθε μια από αυτές τις υποψήφιες τιμές, τρέχει τον αλγόριθμο του Θεωρήματος 10 ώστε να προσπαθήσει να παραγοντοποιήσει το N. Μπορεί να αποδειχτεί ότι οι υποψήφιες τιμές του $p \bmod 2^{n/4}$ είναι το πολύ $e\log_2 e$ σε πλήθος. Δηλαδή, το αργότερο μετά από $e\log_2 e$, το N θα παραγοντοποιηθεί.

Το Θεώρημα 9 είναι γνωστό και σαν *Επίθεση Μερικού Κλειδιού.* Παρόμοιες επιθέσεις για μεγαλύτερες τιμές το e όσο ισχύει όμως $e < \sqrt{N}$. Παρ' όλα αυτά, οι τεχνικές είναι λίγο περισσότερο πολύπλοκες [17]. Είναι ενδιαφέρον που διακριτά βασισμένα σε λογάριθμο κρυπτοσυστήματα όπως το El Gamal σύστημα δημοσίου κλειδιού εμφανίζονται «αναίσθητα» σε τέτοιου είδους επιθέσεις. Πράγματι, αν δίνεται το $g^x \bmod p$ και ένα σταθερό μέρος των bit του x, δεν υπάρχει γνωστός πολυωνυμικός αλγόριθμος που να υπολογίζει το υπόλοιπο x.

Αν το εκθετικό κρυπτογράφησης είναι μικρό, το σύστημα RSA φανερώνει τα μισά από τα πιο σημαντικά bit του κλειδιού d. Για να γίνει αντιληπτό, λαμβάνεται και πάλι η εξίσωση $ed - k(N - p - q + 1) = 1$ για $0 < k \le e$. Δεδομένου k, ο επιτιθέμενος υπολογίζει:

$$\hat{d} = [(kN + 1)/e]$$

Οπότε : $|\hat{d} - d| \leq k(p+q)/e \leq 3k\sqrt{N}/e < 3\sqrt{N}$

Ως εκ τούτου, το $\hat{d}$ είναι μια καλή προσέγγιση του d. Το όριο δείχνει ότι, για τα περισσότερα d, τα μισά πιο σημαντικά bit του $\hat{d}$ είναι ίσα με αυτά του d. Αφού υπάρχουν e πιθανές τιμές του k, ο επιτιθέμενος μπορεί να κατασκευάσει ένα μικρό σύνολο μεγέθους e ώστε ένα από τα στοιχεία του συνόλου αυτού να είναι ίσο με τα μισά από τα πιο σημαντικά bit του d. Η περίπτωση e=3 είναι ιδιαιτέρως ενδιαφέρουσα αφού μπορεί να αποδειχτεί ότι πάντα k=2 και συνεπώς το σύστημα διαρρέει συνέχεια τα μισά από τα πιο σημαντικά bit.

4.6 Επιθέσεις Κατά Υλοποιήσεων

Οι επιθέσεις που θα αναφερθούν παρακάτω ανήκουν σε ολοκληρωτικά διαφορετική κλάση. Αντί να επιτεθούν στη συνάρτηση του RSA, επιτίθενται στην υλοποίηση αυτής.

4.6.1 Επιθέσεις Χρονισμού

Έστω έξυπνη κάρτα που αποθηκεύει ένα ιδιωτικό RSA κλειδί, αφού ανθίσταται προσπάθειες «διάρρηξης», ο επιτιθέμενος δεν μπορεί να την εξετάσει και να ανακτήσει το κλειδί. Υπάρχει όμως μια έξυπνη επίθεση που αποδίδεται στον Kocher [16] που αποδεικνύει ότι η μέτρηση με ακρίβεια του χρόνου που χρειάζεται η κάρτα για να εκτελέσει μια RSA αποκρυπτογράφηση ή ψηφιακή υπογραφή μπορεί να οδηγήσει στη γρήγορη ανακάλυψη του εκθετικού κρυπτογράφησης d.

Η μέθοδος που θα αναλυθεί είναι ο αλγόριθμος «επαναλαμβανόμενων υψώσεων στο τετράγωνο». Έστω η πρωταρχική αναπαράσταση του d είναι : $d = d_n d_{n-1} ... d_0$ (δηλαδή $d = \sum_{i=0}^{n} 2^i d_i$ με $d_i \in \{0,1\}$). Ο αλγόριθμος υπολογίζει $C = M^d \bmod N$ χρησιμοποιώντας το πολύ 2n τμηματικούς πολλαπλασιασμούς. Βασίζεται στη παρατήρηση ότι $C = \prod_{i=0}^{n} M^{2^i d_i} \bmod N$.

Ο αλγόριθμος εργάζεται ως εξής: Θέσε ζ ίσο με M και C ίσο με 1. Για i=0,…,n ακολούθησε τα βήματα:

- αν di=1θέσε το C=$Cz \bmod N$
- θέσε το z ίσο με $z^2 \bmod N$

Στο τέλος, το C έχει τιμή $C = M^d \bmod N$

Η μεταβλητή z ανατρέχει το σύνολο τιμών $M^{2^i} \bmod N$ i=0,…,n. Η μεταβλητή C «συλλέγει» τις κατάλληλες δυνάμεις στο σύνολο ώστε να βρει το $M^d \bmod N$.

Για να εκκινήσει η επίθεση, ο επιτιθέμενος ζητά από την έξυπνη κάρτα να παράγει μια σειρά από υπογραφές για ένα μεγάλο πλήθος από μηνύματα $M_1,\ldots,M_k \in \mathbf{Z}_N^*$ και μετρά το χρόνο T_i που χρειάζεται για να παράγει τη κάθε υπογραφή.

Η επίθεση αυτή ανακτά ένα bit του κλειδιού κάθε φορά ξεκινώντας από το λιγότερο σημαντικό. Είναι γνωστό ότι το d είναι περιττός και γι αυτό d_0=1. Αρχικά $z = M^2 \bmod N$ και $C = M$. Αν d_1=1, η έξυπνη κάρτα υπολογίζει το $Cz = MM^2 \bmod N$. Αν t_i ο χρόνος για τον υπολογισμό του $M_i M_i^2 \bmod N$. Το κάθε t_i είναι διαφορετικό αφού ο χρόνος υπολογισμού του $M_i M_i^2 \bmod N$ εξαρτάται από το εκάστοτε M_i. Ο επιτιθέμενος

μετρά τα t_i πριν την επίθεση και αφού έχει πρώτα βρει τις φυσικές προδιαγραφές της κάρτας.

Ο Kocher παρατήρησε ότι όταν d_1=1, τα δυο σύνολα $\{t_i\}$ και $\{T_i\}$ συσχετίζονται. Για παράδειγμα αν για κάποιο i το t_i είναι πολύ μεγαλύτερο του αναμενόμενου, τότε πολύ πιθανό το ίδιο να συμβαίνει και με το T_i. Από την άλλη, αν d_1=0 τα δυο σύνολα $\{t_i\}$, $\{T_i\}$ συμπεριφέρονται σαν ανεξάρτητες τυχαίες μεταβλητές. Μετρώντας τη συσχέτιση ο επιτιθέμενος μπορεί να αποφανθεί αν το d_1 είναι 0 ή 1.

Συνεχίζοντας με τη μέθοδο αυτή, ο επιτιθέμενος ανακτά τα d_2, d_3 κ.ο.κ. Αν μάλιστα χρησιμοποιείται μικρό δημόσιο εκθετικό e, η επίθεση αποκάλυψης μέρους του κλειδιού δείχνει ότι η επίθεση χρονισμού του Kochner χρειάζεται μέχρις ότου ανακαλυφθεί το ένα τέταρτο των bit του d.

Υπάρχουν δυο τρόποι άμυνας έναντι της επίθεσης αυτής: Ο πιο απλός είναι η πρόσθεση κατάλληλης υστέρησης ώστε η τμηματικοποιημένη ύψωση σε δύναμη να διαρκεί πάντα τον ίδιο χρόνο. Η δεύτερη προσέγγιση λόγω του Rivest βασίζεται στη τεχνική της τύφλωσης. Πριν από την αποκρυπτογράφηση του M, η έξυπνη κάρτα επιλέγει $r \in \mathbf{Z}_N^*$ και υπολογίζει το $M' = \mathrm{M} r^e \bmod N$. Εφαρμόζει στη συνέχεια το d στο M' και λαμβάνει $C' = (M')^d \bmod N$. Τελικά θέτει $C = C' / r \bmod N$ και η κάρτα εφαρμόζει το d σε άγνωστο μήνυμα M΄ που δε γνωρίζει ο επιτιθέμενος και γι αυτό ακυρώνεται η επίθεση.

Ο Kocher ανακάλυψε πρόσφατα μια άλλη επίθεση η οποία ονομάζεται κρυπτανάλυση ισχύος. Ο Kocher απέδειξε ότι μετρώντας τη κατανάλωση ισχύος της κάρτας κατά τη δημιουργία ψηφιακών υπογραφών, είναι δυνατό να ανακτηθεί το κλειδί. Όπως αποδείχτηκε, κατά τη διάρκεια πολλαπλασιασμού μεγάλης ακρίβειας, η κάρτα καταναλώνει μεγάλη ισχύ. Μετρώντας τα διαστήματα υψηλής κατανάλωσης, ο Kochner μπόρεσε να καταλάβει αν σε συγκεκριμένη πράξη η κάρτα επιτελεί έναν ή δυο πολλαπλασιασμούς φανερώνοντας έτσι bit του κλειδιού.

4.6.2 Τυχαία Σφάλματα (Επιθέσεις Κολλήματος)

Υλοποιήσεις του RSA για τη λειτουργία της αποκρυπτογράφησης αλλά και για τη παραγωγή ψηφιακών υπογραφών, χρησιμοποιούν το Κινέζικο Θεώρημα Υπολοίπου ώστε να επιταχύνουν τον υπολογισμό της ποσότητας $M^d \bmod N$. Αντί να δουλεύει με το modulo N ο αποστολέας πρώτα υπολογίζει τις υπογραφές modulo p και q, και εν συνεχεία με το Κινέζικο Θεώρημα Υπολοίπου συνδυάζει τα αποτελέσματα. Συγκεκριμένα, υπολογίζει τα

$$C_p = M^{d_p} \bmod p$$
$$C_q = M^{d_q} \bmod q$$

όπου d_p=dmod(p-1) και d_q=dmod(q-1). Εν συνεχεία υπολογίζει το C θέτοντας: $C = T_1 C_p + T_2 C_q \pmod N$ όπου

$$T_1 = \begin{Bmatrix} 1 \bmod p \\ 0 \bmod q \end{Bmatrix} \text{ και } T_2 = \begin{Bmatrix} 0 \bmod p \\ 1 \bmod q \end{Bmatrix}$$

Ο χρόνος εκτέλεσης του τελευταίου βήματος CRT είναι αμελητέος συγκρινόμενος με τις δυο υψώσεις σε δύναμη. Σημειώνεται ότι τα p, q έχουν το μισό μήκος του N. Αφού οι

απλές υλοποιήσεις του πολλαπλασιασμού απαιτούν χρόνο στη τέταρτη δύναμη, ο πολλαπλασιασμός modulo p είναι τέσσερις φορές πιο γρήγορος από αυτόν του modulo N . Επιπλέον, το d_p έχει το μισό μήκος του N και συνεπώς ο υπολογισμό του $M^{d_p} \bmod p$ είναι οκτώ φορές πιο γρήγορος από τον υπολογισμό του $M^d \bmod N$. Ο συνολικός λοιπόν χρόνος για υπογραφή μειώνεται κατά ένα παράγοντα της τάξης του τέσσερα. Πολλές υλοποιήσεις χρησιμοποιούν τη μέθοδο αυτή για να βελτιώσουν τις επιδόσεις.

Οι Boneh, DeMillo και Lipton [15], παρατήρησαν ότι υφίσταται εγγενής κίνδυνος όταν χρησιμοποιείται η μέθοδος CRT. Αν κατά τη παραγωγή της υπογραφής, ο υπολογιστής του αποστολέα λόγω προσωρινού κολλήματος προκαλέσει μια λανθασμένη πράξη και αλλάξει ένα bit (για παράδειγμα κατά τη μεταφορά μιας τιμής από έναν καταχωρητή σε άλλον να συμβεί αλλοίωση λόγω ηλεκτρομαγνητικής παρεμβολής). Με τη λανθασμένη υπογραφή, ένας επιτιθέμενος μπορεί εύκολα να παραγοντοποιήσει το modulus N.

Παρουσιάζεται μια εκδοχή της επίθεσης όπως παρουσιάζεται από τον A. K. Lenstra : Έστω εμφάνιση μοναδικού σφάλματος όταν ο αποστολέας παράγει μια υπογραφή. Ως αποτέλεσμα, ένα εκ των C_p ή C_q θα υπολογιστεί εσφαλμένα. Αν το C_p είναι το σωστό τότε το $\hat{C}_q$ θα είναι το αλλοιωμένο. Η προκύπτουσα υπογραφή θα είναι η

$$\hat{C} = T_1 C_p + T_2 \hat{C}_q .$$

Μόλις ο επιτιθέμενος λάβει το $\hat{C}$ αναγνωρίζει ότι είναι λάθος υπογραφή αφού $\hat{C}^e \neq M \bmod N$. Παρ' όλα αυτά ισχύει

$$\hat{C}^e = M \bmod p ,$$
$$\hat{C}^e \neq M \bmod q .$$

Ως αποτέλεσμα, το $\text{ΜΚΠ}(\text{N}, \hat{C}^e - \text{M})$ φανερώνει ένα σημαντικό παράγοντα του N.

Για να λειτουργήσει η επίθεση αυτή πρέπει ο επιτιθέμενος να ξέρει το αρχικό μήνυμα M. Με άλλα λόγια εικάζεται ότι ο αποστολέας δεν χρησιμοποιεί κάποια διαδικασία παραγεμίσματος. Το παραγέμισμα του M με τυχαία δεδομένα πριν από τη δημιουργία της υπογραφής παρεμποδίζει την επίθεση αυτή. Μια πιο απλή άμυνα είναι ο έλεγχος της υπογραφής από τον ίδιο τον αποστολέα πριν τη μεταδώσει. Ο έλεγχος είναι ιδιαίτερα σημαντικός όταν χρησιμοποιείται η μέθοδος CRT επίσπευσης. Τα τυχαία σφάλματα είναι επικίνδυνα για πολλά κρυπτογραφικά συστήματα. Ακόμα και αυτά που δεν χρησιμοποιούν CRT όπως ορισμένες εκδοχές του RSA δύνανται να δεχτούν επίθεση κατ' αυτό τον τρόπο [15].

5 Επιθέσεις Σφαλμάτων στα Δημόσια Κλειδιά RSA

5.1 Οι Υλοποιήσεις "Αριστερά-προς-Δεξιά" Είναι Επίσης Τρωτές

Η εσκεμμένη εισαγωγή σφαλμάτων κατά τη διάρκεια της εκτέλεσης κρυπτογραφικών αλγορίθμων αποτελεί έναν ισχυρό τρόπο ανάκτησης μυστικών πληροφοριών. Αυτή η ιδέα πρώτο-δημοσιεύτηκε από ερευνητές της Bellcore [38], [39] και αφορά κρυπτοσυστήματα πολλαπλών δημοσίων κλειδιών. Πράγματι, αυτές οι μελέτες παρέχουν επιτυχείς εφαρμογές της παραπάνω μεθόδου περιλαμβάνοντας τον RSA τόσο σε τυποποιημένη (standard), όσο και σε CRT μέθοδο λειτουργίας. Η εργασία αυτή ολοκληρώθηκε και ονομάστηκε **Διαφορική Ανάλυση Σφαλμάτων** (Differential Fault Analysis, **DFA**), από τους E. Biham και A. Shamir με εφαρμογές σε κρυπτοσυστήματα μυστικών κλειδιών [43]. Η αυξανόμενη δημοτικότητα αυτού του είδους προσέγγισης την τελευταία δεκαετία, βασίστηκε στην ευκολία αλλοίωσης μιας εντολής κατά την εκτέλεση της, [40] και στην αρχική δυσκολία της εύρεσης αποτελεσματικών αντίμετρων [42, 53, 48].

Έχουν δημοσιευθεί πολλές εφαρμογές αυτού του είδους επίθεσης, κατά κρυπτοσυστημάτων RSA , που βασίζονται σε εισαγωγή σφαλμάτων. Οι πρώτες είχαν να κάνουν με τη διατάραξη του ιδιωτικού κλειδιού ή των προσωρινών τιμών κατά τον υπολογισμό [38, 37, 39]. Η διατάραξη των δημόσιων στοιχείων του αλγόριθμου θεωρήθηκε πραγματική απειλή όταν ο J-PSeifert δημοσίευσε μια επίθεση στον μηχανισμό ελέγχου της RSA υπογραφής [51, 49]. Στο κεφάλαιο αυτό διερευνάται αρχικά η πιθανότητα τροποποίησης του δημόσιου modulus N έτσι ώστε ο λανθασμένος να είναι πρώτος ή εύκολος να παραγοντοποιηθεί. Στη συνέχεια, η ομάδα του E. Brier επεκτάθηκε στην πλήρη ανάκτηση του ιδιωτικού εκθέτη d σε διάφορες RSA εφαρμογές [36]. Και οι δυο παραπάνω μελέτες βασίζονται στην υπόθεση ότι το σφάλμα συμβαίνει πριν πραγματοποιηθεί η RSA modular ύψωση σε δύναμη. Πρώτη ήταν η ομάδα του A. Berzati που έθεσε το θέμα της τροποποίησης του modulus κατά τη διάρκεια της ύψωσης σε δύναμη [35]. Και πάλι αυτή η μελέτη περιορίστηκε σε εφαρμογή αλγορίθμων «δεξιά-προς-αριστερά» ύψωση σε δύναμη.

Στόχος του παρόντος κεφαλαίου είναι να παρουσιαστεί μια γενίκευση της προηγούμενης επίθεσης σε τύπο «αριστερά –προς -δεξιά» ύψωση σε δύναμη. Υπό την εσφαλμένη παραδοχή ότι το modulus μπορεί να γίνει ένας αριθμός με γνωστή παραγοντοποίηση, αποδεικνύεται ότι είναι δυνατή η ανάκτηση ολόκληρου του ιδιωτικού εκθέτη (άρα και η αποκρυπτογράφηση). Παρέχεται επίσης μια λεπτομερής μελέτη αυτού του μοντέλου σφάλματος, βασιζόμενη στη θεωρία αριθμών, ώστε να αποδειχτεί η συνοχή και η δυνατότητα της πρακτικής εφαρμογής του με διαφορετικά είδη διαταραχής του ιδιωτικού κλειδιού. Τέλος, προτείνεται ένας αλγόριθμος ανάκτησης ολόκληρου του ιδιωτικού εκθέτη, που είναι αποτελεσματικός τόσο όσον αφορά το πλήθος σφαλμάτων, όσο και για τον υπολογιστικό χρόνο.

5.2 Modular Αλγόριθμοι Ύψωσης σε Δύναμη

Οι δυαδικοί αλγόριθμοι ύψωσης σε δύναμη συχνά χρησιμοποιούνται για τον υπολογισμό του RSA modular ύψωσης σε δύναμη m^d όπου ο εκθέτης *d* εκφράζεται σε δυαδική μορφή ως $d = \Sigma_{i=0}^{n-1} 2^i \cdot d_i$ Η πολυωνυμική τους πολυπλοκότητα σε σχέση με το μήκος των εισόδων, τους καθιστούν πολύ ενδιαφέροντες για την εκτέλεση modular

ύψωσης σε δύναμη. Ο αλγόριθμος 1 περιγράφει μια μέθοδο υπολογισμού modular ύψωσης σε δύναμη, μέσω σάρωσης των bits του *d*, από τα λιγότερο σημαντικά bits (LSB) προς τα περισσότερο σημαντικά bits (MSB).

Αλγόριθμος 1. «Δεξιά -προς -Αριστερά» modular ύψωση σε δύναμη

είσοδος: *m, d, N*

```
1:  A:=1, B:=m;
2:  for i = 0, … , n-1 do
3:      if di == 1 then
4:          A:=(A*B) mod N;
5:      end
6:      B:=B^2 mod N;
7:  end
```

έξοδος: *A*

Αλγόριθμος 2. «Αριστερά –προς -Δεξιά» modular ύψωση σε δύναμη

είσοδος: *m, d, N*

```
1:  A:=1;
2:  for i = n-1, … , 0 do
3:      A:=A^2 mod N
4:      if di == 1 then
5:          A:=(A*m) mod N;
6:      end
7:  end
```

έξοδος: *A*

Αυτός είναι ο λόγος που συνήθως αναφέρεται ως ο «*Δεξιά-προς-Αριστερά*» modular ύψωση σε δύναμη αλγόριθμος. Αυτή είναι η ειδική εφαρμογή που έχει δεχθεί επίθεση στο [35] αλλοιώνοντας το δημόσιο συντελεστή του RSA.

Ο δυαδικός αλγόριθμος που υλοποιεί τη δυαδική modular ύψωση σε δύναμη είναι η «*Αριστερά –προς -Δεξιά*» ύψωση σε δύναμη που περιγράφεται στον Αλγόριθμο 2. Αυτός ο αλγόριθμος ανιχνεύει τα bits του εκθέτη από MSB προς LSB και είναι ελαφρύτερος από τον «*Δεξιά-προς-Αριστερά*» σε ότι αφορά την κατανάλωση μνήμης.

5.3 Τροποποίηση του συντελεστή (modulus) και Απόπειρα Επέκτασης

5.3.1 Προηγούμενες Εργασίες

Ο J-PSeifert υπήρξε ο πρώτος που έθεσε το θέμα αλλοίωσης των RSA στοιχείων δημόσιων κλειδιών [51], [49]. Αυτή η επίθεση σφάλματος έχει σκοπό να εξαναγκάσει ένα μηχανισμό επαλήθευσης υπογραφής να αποδέχεται πλαστές υπογραφές τροποποιώντας την τιμή του δημόσιου συντελεστή *N*. Με αυτή την εσφαλμένη επίθεση δεν αποκαλύπτεται καμία πληροφορία σχετικά με τον ιδιωτικό εκθέτη *d*. Η αποτελεσματικότητά της εξαρτάται από την ικανότητα του επιτιθέμενου να αναπαράγει το μοντέλο εισαγωγής σφάλματος που έχει επιλεγεί για την τροποποίηση του συντελεστή (modulus).

Η δουλειά του Seifert ενέπνευσε τους συντάκτες της [36] που πρώτοι χρησιμοποίησαν τη διατάραξη του δημόσιου συντελεστή για την ανάκτηση ολόκληρου του ιδιωτικού κλειδιού d. Ο επιτιθέμενος χρειάζεται να πραγματοποιήσει μια εκστρατεία διατάραξης ώστε να συλλέξει έναν αρκετά μεγάλο αριθμό ζευγών (μηνυμάτων, εσφαλμένων υπογραφών). Όπως και στην επίθεση Seifert, το σφάλμα του συντελεστή εισάγεται πριν την εκτέλεση της ύψωσης σε δύναμη. Τρεις μέθοδοι οι οποίες βασίζονται στη χρήση του Κινέζικου Θεωρήματος Υπολοίπου (Chinese Remainder Theorem) και την επίλυση πολύ μικρών διακριτών λογαρίθμων, προτείνονται στα [36] και [44] για την ανάκτηση του ιδιωτικού εκθέτη από το σύνολο των ζευγών που συγκεντρώθηκαν.

Μια νέα, «επαναστατική», επίθεση σφάλματος τύπου «*Δεξιά –προς -Αριστερά*» ύψωσης σε δύναμη έχει παρουσιαστεί πρόσφατα στο [35], επιτρέποντας στον εισβολέα

να χρησιμοποιεί κατ' επιλογή ποικίλα μοντέλα εισαγωγής σφάλματος για την ανάκτηση του ιδιωτικού εκθέτη. Οι λεπτομέρειες αυτής της επίθεσης παρουσιάζονται παρακάτω.

5.3.2 Διαταραχή Δημόσιου Κλειδιού κατά την Εκτέλεση της RSA: Περίπτωση του «Δεξιά –προς -Αριστερά» Αλγορίθμου

Μοντέλο Σφάλματος. Στις προτάσεις των JP Seifert και E. Brier*etal.* [51, 36] το σφάλμα προκαλείται πριν από την ύψωση σε δύναμη, ώστε η εκτέλεση της εντολής να γίνεται στο σύνολό της με τον λανθασμένο συντελεστή $\hat{N}$.

Η επίθεση που παρουσιάζεται από την ομάδα εργασίας του A. Berzati [35] επεκτείνει το μοντέλο σφάλματος επιτρέποντας στον επιτιθέμενο να εισάγει το σφάλμα κατά την εκτέλεση της «*Δεξιά-προς-Αριστερά*» ύψωσης σε δύναμη. Η αλλοίωση του N υποτίθεται ότι συμβαίνει χάρη σε μια παροδική τυχαία τροποποίηση των bytes. Η τιμή του λανθασμένου συντελεστή $\hat{N}$ δεν είναι γνωστή στον επιτιθέμενο, παρά η χρονική θέση του σφάλματος, την οποία και χρησιμοποιεί για να εκτελέσει την κρυπτανάλυση. Αυτό το μοντέλο σφάλματος έχει επιλεχθεί για την απλότητα και τη δυνατότητα πραγματοποίησης του στα αποσπάσματα [47, 41]. Επιπλέον, μπορεί εύκολα να προσαρμοστεί τόσο σε 16-bit όσο και σε 32-bit αρχιτεκτονικές υπολογιστικών συστημάτων.

Εσφαλμένος Υπολογισμός. Έστω $d = \Sigma_{i=0}^{n-1} 2^i \cdot d_i$ η δυαδική αναπαράσταση του *d*. Το αποτέλεσμα της υπογραφής RSA μπορεί να γραφτεί ως:

$$S \equiv m^{\Sigma_{i=0}^{n-1} 2^i \cdot d_i} \bmod N \tag{1}$$

Θεωρείται ότι ένα σφάλμα συνέβη *j* βήματα πριν το τέλος της ύψωσης σε δύναμη, κατά τον υπολογισμό του τετραγώνου. Σύμφωνα με το μοντέλο σφάλματος που περιγράφεται, όλες οι πράξεις που ακολουθούν εκτελούνται με ελαττωματικό συντελεστή $\hat{N}$. Δεικνύεται με $A \equiv m^{\Sigma_{i=0}^{n-1} 2^i \cdot d_i} \bmod N$ η εσωτερική τιμή καταχωρητή και με $\hat{B}$ το αποτέλεσμα του εσφαλμένου τετραγώνου:

$$\hat{B} \equiv (m^{2^{(n-j-1)}} \bmod N)^2 \bmod \hat{N} \tag{2}$$

Ως εκ τούτου, η εσφαλμένη υπογραφή $\hat{S}$ μπορεί να γραφεί ως:

$$\hat{S} \equiv A \cdot \hat{B}^{\Sigma_{i=(n-j)}^{n-1} 2^{[i-(m-j)]} \cdot d_i} \bmod \hat{N} \tag{3}$$

$$\equiv [(m^{\Sigma_{i=0}^{(n-j-1)} 2^i \cdot d_i} \bmod \hat{N}) \tag{4}$$

$$\cdot (m^{2^{(n-j-1)}} \bmod N)^{\Sigma_{i=(n-j)}^{n-1} 2^{[i-(n-j)+1]} \cdot d_i}] \bmod \hat{N}$$

Από την παραπάνω διατύπωση του $\hat{S}$, μπορεί κανείς να παρατηρήσει ότι η εισαγωγή σφάλματος, χωρίζει τον υπολογισμό σε ένα σωστό (υπολογιζόμενο με *N*) και σε ένα λανθασμένο (υπολογιζόμενο με $\hat{N}$) μέρος. Τμήμα του *d* χρησιμοποιείται κατά το λανθασμένο υπολογισμό. Αυτό ακριβώς είναι το μυστικό τμήμα του εκθέτη που θα ανακτηθεί στην παρακάτω ανάλυση.

Αρχή Εκτέλεσης της Επίθεσης. Τόσο από τη σωστή υπογραφή *S*, όσο και από τη λανθασμένη $\hat{S}$ (που λαμβάνεται από το ίδιο μήνυμα *m*), ο εισβολέας μπορεί να ανακτήσει το απομονωμένο τμήμα του ιδιωτικού κλειδιού $d_{(1)} = \Sigma_{i=n-j}^{n-1} 2^i \cdot d_i$ Μάλιστα, προσπαθεί να βρει ταυτόχρονα υποψήφιες τιμές για τον ελαττωματικό συντελεστή N' (σύμφωνα με την παραδοχή του τυχαίου εσφαλμένου byte) και για το τμήμα του εκθέτη d_1' που ικανοποιεί το παρακάτω:

$$\hat{S} \equiv (S \cdot m^{-d'_{(1)}} \bmod N) \cdot (m^{2^{(n-j-1)}} \bmod N)^{2^{[1-(n-j)]} \cdot d'_{(1)}} \bmod N' \tag{5}$$

Σύμφωνα με τον [35], το ζεύγος $(d'_{(1)}, N)$ που ικανοποιεί την εξίσωση (5) είναι το σωστό με πιθανότητα πολύ κοντά στο 1. Τότε, τα μυστικά bits που ακολουθούν, θα βρεθούν επαναλαμβάνοντας αυτή την επίθεση χρησιμοποιώντας τα ήδη γνωστά περισσότερο σημαντικά bits του *d*, καθώς και μια λανθασμένη υπογραφή από προγενέστερο στάδιο της διαδικασίας. Όσον αφορά τον αριθμό σφαλμάτων, η ανάκτηση ολόκληρου του ιδιωτικού κλειδιού απαιτεί κατά μέσο όρο (*n/l*) εσφαλμένες υπογραφές, όπου *l* ο μέσος αριθμός αυτών που ανακτώνται κάθε φορά. Κατά συνέπεια, αυτός ο περιορισμένος αριθμός απαιτούμενων σφαλμάτων καθιστά την επίθεση τόσο αποτελεσματική, όσο και εφικτή.

5.3.3 Εφαρμογή στην «Αριστερά–προς–Δεξιά» Modular Ύψωση σε Δύναμη

Σε αυτή την ενότητα εφαρμόζεται η επίθεση σφάλματος που ήδη αναλύθηκε, στην «*Αριστερά –προς -Δεξιά*» υλοποίηση του RSA. Σύμφωνα με το ίδιο μοντέλο σφάλματος, διερευνάται πλέον τι εμποδίζει έναν εισβολέα από το να αναπαράγει την επίθεση εναντίον της δυαδικής υλοποίησης. Με συμβολίζεται *A* η εσωτερική τιμή καταχωρητή λίγο πριν την τροποποίηση του συντελεστή *N*:

$$A \equiv m^{\Sigma_{i=j}^{n-1} 2^{i-j} \cdot d_i} \bmod N \tag{6}$$

Ως εκ τούτου, γνωρίζοντας ότι η πρώτη αλλοιωμένη πράξη είναι ύψωση σε τετράγωνο, η εσφαλμένη υπογραφή $\hat{S}$ μπορεί να γραφεί ως:

$$\begin{aligned} \hat{S} &\equiv (((A^2 \cdot m^{d_{j-1}})^2 \cdot m^{d_{j-2}})^2 \ldots)^2 \cdot m^{d_0} \bmod N \\ &\equiv A^{2^j} \cdot m^{\Sigma_{i=0}^{j-1} 2^i \cdot d_i} \bmod N \end{aligned} \tag{7}$$

Παρατηρώντας την (7), μπορεί κάποιος να διακρίνει ότι η διαταραχή έχει δύο επιπτώσεις στην ελαττωματική υπογραφή $\hat{S}$. Πρώτον, χωρίζει τον υπολογισμό σε ένα σωστό μέρος (δηλαδή, στην εσωτερική τιμή καταχωρητή *A*) και σε ένα λάθος, όπως για τη διατάραξη της «Δεξιά –προς -Αριστερά» ύψωσης σε δύναμη [35]. Η δεύτερη είναι η προσθήκη *j* διαδοχικών τετραγώνων της τοπικής μεταβλητής *A*, υπολογισμένης κατά modulo N. Αυτή η προστιθέμενη λειτουργία ανατρέπει την προηγούμενη επίθεση κατά της «Δεξιά –προς -Αριστερά» ύψωσης σε δύναμη [35] λόγω της δυσκολίας υπολογισμού τετραγωνικών ριζών στα RSA δαχτυλίδια (RSA rings).

Η παρούσα προσέγγιση για γενίκευση της προηγούμενης επίθεσης σε «Αριστερά – προς -Δεξιά» ύψωση σε δύναμη εκμεταλλεύεται την τροποποίηση του συντελεστή ώστε να επιτύχει την αλλαγή των αλγεβρικών ιδιοτήτων των δαχτυλιδιών RSA,. Με άλλα λόγια, αν ο N είναι πρώτος αριθμός, τότε είναι δυνατό να υπολογιστούν τετραγωνικές ρίζες σε πολυωνυμικό χρόνο. Επιπροσθέτως, αρκεί ο N να είναι B-ομαλό με B αρκετά μικρό ώστε να επιτρέπει την εύκολη παραγοντοποίηση του N. Στη περίπτωση αυτή, το Κινέζικο Θεώρημα Υπολοίπων (Chinese Remainder Theorem) παρέχει τη δυνατότητα υπολογισμού τετραγωνικών ριζών σε πολυωνυμικό χρόνο. Παρακάτω αποδεικνύεται ότι έτσι κι αλλιώς ο αριθμός των πρώτων αριθμών N είναι επαρκής για να προσφέρει ένα ρεαλιστικό μοντέλο εισαγωγής σφάλματος.

5.4 Μοντέλο Εισαγωγής Σφάλματος

Σύμφωνα με την προηγούμενη ενότητα, το πρόβλημα των τετραγωνικών ριζών μπορεί να ξεπεραστεί διαταράσσοντας το συντελεστή N ώστε ο αλλοιωμένος N να είναι πρώτος. Στην παρούσα παράγραφο μελετάται η συνέπεια και η εφαρμοσιμότητα ενός τέτοιου μοντέλου σφάλματος το οποίο έχει ήδη υιοθετηθεί στην επίθεση Seifert [49, 51]. Παρακάτω προτείνονται περαιτέρω πειραματικά τεκμήρια της πρακτικής εφαρμογής αυτού του μοντέλου.

5.4.1 Θεωρητικές Εκτιμήσεις / Υπολογισμοί

Πρώτα απ' όλα γίνεται εκτίμηση του πλήθους των πρώτων αριθμών με σταθερό αριθμό bits. Από [46, Θεώρημα 1.10], λαμβάνονται τα ακόλουθα όρια για το πλήθος των πρώτων π κάτω από έναν ορισμένο ακέραιο x:

$$\pi(x) \geq \frac{x}{\ln(x)}\left(1+\frac{1}{\ln(x)}+\frac{1.8}{\ln^2(x)}\right),\ for\ x \geq 32299$$
$$\pi(x) \leq \frac{x}{\ln(x)}\left(1+\frac{1}{\ln(x)}+\frac{2.51}{\ln^2(x)}\right),for\ x \geq 355991 \quad (8)$$

Στη συνέχεια, για αριθμούς των ακριβώς t bits, τέτοια ώστε t≥ 19 bits, ο αριθμός των πρώτων αριθμών είναι $\pi_t = \pi(2^t) - \pi(2^{t-1})$. Χρησιμοποιώντας τα προηγούμενα όρια (8), η πιθανότητα ένας αριθμός t-bit να είναι πρώτος, $pr_t = \frac{\pi_t}{2^{t-1}}$, πλήρη την:

$$pr_t \succ Inf(t) = \frac{0.480t^5 - 1.229t^4 + 0.0265t^3 - 7.602t^2 + 9.414t - 3.600}{t^3(t-1)^3\ln^3(2)} \quad (9)$$

$$pr_t \prec Sup(t) = \frac{0.480t^5 - 1.229t^4 + 2.157t^3 - 11.862t^2 + 13.674t - 5.02}{t^3(t-1)^3\ln^3(2)}$$

Για παράδειγμα, αν t = 1024 bits, τότε:

$$Inf(1024) = \frac{1}{709.477} \text{ και } Sup(1024) = \frac{1}{709.477}$$

Άρα, περίπου ένας μήκους 1024-bit αριθμός στους 709 είναι πρώτος, ενώ ανάμεσα στους μήκους 2048 bit αριθμούς πάνω από ένας στους 1419 είναι πρώτος.

Έστω ένα σύνολο από k τυχαία επιλεγμένους αριθμούς των ακριβώς t-bits και έστω *PN* η τυχαία μεταβλητή που εκφράζει τον αναμενόμενο αριθμό των πρώτων αριθμών σε αυτό το σύνολο. Η μεταβλητή αυτή ακολουθεί έναν διωνυμικό νόμο B (k, pr_t). Στη συνέχεια δίνεται το ακόλουθο διάστημα πρώτων αριθμών (με *α* και *b* ακέραια όρια):

$$\Pr[a \le PN \le b] = \sum_{i=a}^{b} \binom{k}{i} pr_t^i (1 - pr_t)^{k-i} \tag{10}$$

Για παράδειγμα, κατασκευάζεται το ακόλουθο σύνολο *N*, σύμφωνα με ένα μοντέλο εισαγωγής σφάλματος σε τυχαίο byte. Με άλλα λόγια, εάν $\oplus$ είναι το bit με bit αποκλειστικό (ή-OR), τότε [1]:

$$N = \{N \oplus R_8 \cdot 2^{8i}, R_8 = 0..255, i = 0..(\frac{n}{8} - 1)\}$$

Τότε ο πληθάριθμος του *N* είναι

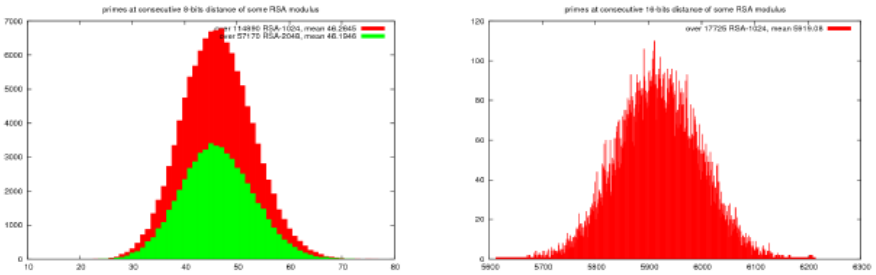
$$|N| = 256 \cdot \frac{n}{8} = 32 \cdot n$$

Αν το σύνολο *N* αποτελείται από τυχαία επιλεγμένες τιμές, τότε η αναλογία των πρώτων αριθμών στο *N*, θα ακολουθεί την (9). Ως εκ τούτου, μπορεί να τεθεί k = |*N* | και να υπολογιστεί ο αντίστοιχος μέσος όρος και τα όρια με μια προσέγγιση του pr_t. Για *n* = 1024, σύμφωνα με την (9), γίνεται εκτίμηση του pr_{1024} και άρα ο μέσος αριθμός των εσφαλμένων πρώτων είναι 32·1024 / 709.47 ≈ 46.186. Η εξίσωση (10) σε συνδυασμό με την εκτίμηση του pr_{1024} δείχνει επίσης ότι ο αριθμός των πρώτων αριθμών σε ένα σύνολο βρίσκεται ανάμεσα στο [18, 80] στο 99.999% των περιπτώσεων. Για *n* = 2048, ο μέσος αριθμός των πρώτων αριθμών είναι 46.176 και βρίσκεται ανάμεσα στο 18 και το 80, στο 99.999% των περιπτώσεων. Προφανώς, το N δεν είναι ένα σύνολο τυχαία επιλεγμένων στοιχείων, εν τούτοις, εμπειρικά δεδομένα αποδεικνύουν πως τέτοια σύνολα συμπεριφέρονται αρκετά παρόμοια με τυχαία σύνολα στοιχείων, όπως φαίνεται και παρακάτω.

5.4.2 Πειραματικά Αποτελέσματα

Έχουν υπολογιστεί σύνολα τυχαία επιλεγμένων RSA moduli και έχει μετρηθεί το πλήθος των πρώτων αριθμών σε αυτά τα σύνολα. Η κατανομή φαίνεται να ακολουθεί έναν διωνυμικό κανόνα (όπως αναμενόταν), και λήφθηκαν τα εξής πειραματικά δεδομένα που υποστηρίξουμε την αναφερθείσα εκτίμηση (βλ. Εικόνα 1).

[1]Για λόγους σαφήνειας θεωρείται ότι ένα σφάλμα μπορεί να πάρει 2^8 τιμές. Στην πραγματικότητα μπορεί να πάρει μόνο 2^8 - 1. Πράγματι, το σφάλμα δεν μπορεί να είναι μηδενικό (null) γιατί αλλιώς η τιμή του N παραμένει αμετάβλητη και το σφάλμα δεν μπορεί να αξιοποιηθεί.



(a) Primes at consecutive 8-bit distance of some RSA modulus

(b) Primes at consecutive 16-bit distance of some RSA modulus

Σχήμα 4 Πειραματική κατανομή των πρώτων αριθμών μεταξύ των ελαττωματικών moduli RSA

Όπως φαίνεται στον Πίνακα 1 που ακολουθεί, έτσι κι αλλιώς *ποτέ* δεν υπήρχε περίπτωση να μη βρεθεί ένας πρώτος αριθμός σε ένα σύνολο *N* (ίσα – ίσα που πάντα βρίσκονται περισσότεροι από 18 πρώτοι αριθμοί σε ένα τέτοιο σύνολο). Αυτό το πειραματικό κατώτατο όριο, ισούται με εκείνο που προκύπτει από την εξέταση ενός τυχαίου συνόλου. Το ίδιο ισχύει και για το ανώτατο όριο. Έτσι, τα αποτελέσματα που λαμβάνονται, επιβεβαιώνουν τη θεωρητική ανάλυση.

Τα αποτελέσματα που παρουσιάζονται μπορούν να επεκταθούν και σε άλλα μοντέλα σφάλματος. Ο Πίνακας 1 παρουσιάζει επίσης θεωρητικά αναμενόμενα αποτελέσματα όταν στοχεύονται αρχιτεκτονικές 16-bit ή 32-bit. Για t = 1024 με αρχιτεκτονική 16-bit ο μέσος αριθμός των πρώτων αριθμών είναι 5911.83 και βρίσκεται στο διάστημα [5520, 6320] στο 99,999% των περιπτώσεων.

Πίνακας 2 Πειραματικές μετρήσεις πρώτων αριθμών στο N

Αρχιτεκτονική	n bits	| N |	| N | prn	# of exp.	# πρώτων αριθμών		
					Ελαχ.	M.O.	Μεγ.
8-bit	1024	215	46.186	114890	18	46.26	79
8-bit	2048	2^{16}	46.176	57170	22	46.19	80
16-bit	1024	2^{22}	5911.83	17725	5621	5919.08	6212
32-bit	1024	2^{37}	≈ $1{,}94 \cdot 10^{8}$				

5.4.3 Συνέπειες

Η μελέτη αυτή ενισχύει την υπόθεση του J-P. Seifert [51, 49] για εξέταση της τροποποίησης μόνο των πρώτων αριθμών του συντελεστή. Έχει αποδειχτεί πως το περιγραφέν μοντέλο σφάλματος μπορεί να θεωρηθεί ως μια τυχαία τροποποίηση του δημόσιου συντελεστή. Στη συνέχεια, ένας μέσος όρος των 709 σφαλμάτων στο *N* θα απαιτηθεί για να ληφθεί ένας πρώτος *N* στην περίπτωση ενός 1024-bit RSA.

Με προσεκτική μελέτη των πειραματικών αποτελεσμάτων, μπορεί κανείς να διαπιστώσει ότι για ένα συγκεκριμένο συντελεστή *N*, η θέση byte του σφάλματος επηρεάζει τον αριθμό των πρώτων αριθμών που βρέθηκαν στο υποσύνολο. Έτσι, αν ο εισβολέας έχει τη δυνατότητα να καθορίσει τη θέση του σφάλματος, μπορεί να αυξήσει τις πιθανότητές του να πάρει έναν πρώτο εσφαλμένο συντελεστή και άρα να μειώσει δραματικά τον αριθμό των εσφαλμένων υπογραφών που απαιτούνται για την πραγματοποίηση της επίθεσης.

5.5 Ο Αλγόριθμος Tonelli και Shanks

Ο αλγόριθμος των Tonelli και Shanks [45] είναι ένας πιθανοθεωρητικός και αρκετά αποτελεσματικός αλγόριθμος που χρησιμοποιείται για να υπολογίζει τετραγωνικές ρίζες modulo P, όπου P πρώτος αριθμός. Η αρχή του αλγορίθμου βασίζεται στον ισομορφισμό μεταξύ της πολλαπλασιαστικής ομάδας $(Z/PZ)^*$ και της προσθετικής ομάδας $Z/(P-1)Z$. Έστωε ότι το P – 1 γράφεται ως:

$$P-1=2^e\cdot r \tag{11}$$

όπου r περιττός. Τότε, η κυκλική ομάδα *G* τάξης2^e, είναι υποομάδα της $Z/(P-1)Z$. Έστω *z* γεννήτορας της G, αν *α* είναι τετραγωνικό υπόλοιπο του modulo N, τότε:

$$a^{(P-1)/2}\equiv (a^r)^{2^{e-1}}\equiv 1 \bmod P \tag{12}$$

Παρατηρώντας ότι α^rmod P είναι ένα τετράγωνο στο *G*, τότε υπάρχει ακέραιος $k\in$ [[0 : 2^e-1]] τέτοιος ώστε

$$a^{(P-1)/2}\equiv (a^r)^{2^{e-1}}\equiv 1 \bmod P \; a^r\cdot z^k=1 in G \tag{13}$$

Και έτσι, $\alpha^{r+1}\cdot z^k = \alpha$ ανήκει στη *G*. Άρα, η τετραγωνική ρίζα του *α*, δίνεται από τη σχέση

$$a^{1/2}\equiv a^{(r+1)/2}\cdot z^{k/2} \bmod P \tag{14}$$

Οι δύο κύριες λειτουργίες αυτού του αλγορίθμου είναι (α) η εύρεση της γεννήτορος *z* της υποομάδας *G*, και (β) ο υπολογισμός του εκθέτη *k*. Η πολυπλοκότητα του αλγορίθμου είναι αυτή της εύρεσης του *k*, $O(\ln^4 P)$ δυαδικές πράξεις, ή $O(\ln P)$ υψώσεις σε δύναμη. Οι λεπτομέρειες του παραπάνω αλγορίθμου περιγράφονται στο [45]. Στην πράξη, σε έναν Pentium IV 3.2GHz, η υλοποίηση GIVARO[2] αυτού του αλγορίθμου παίρνει κατά μέσο όρο 7/1000 του δευτερολέπτου για να βρει μια τετραγωνική ρίζα για έναν πρώτο συντελεστή 1024-bit.

5.6 Ομαλός Συντελεστής (Smooth Modulus)

Όπως και στο [49], αυτό που πραγματικά απαιτείται για τον εσφαλμένο συντελεστή είναι μόνο να είναι εύκολα παραγοντοποιήσιμος. Πράγματι, μπορεί κανείς να υπολογίσει τετραγωνικές ρίζες modulo μη πρώτων συντελεστών, αρκεί η παραγοντοποίηση να είναι γνωστή. Η ιδέα είναι πρώτα να βρεθούν τετραγωνικές ρίζες modulo κάθε πρώτου παράγοντα του N. Στη συνέχεια να υψωθούν ανεξάρτητα σε τετραγωνικές ρίζες modulo κάθε πρώτης δύναμης. Και τέλος να συνδυαστούν χρησιμοποιώντας το Κινέζικο Θεώρημα Υπολοίπου (βλ. π.χ. [52, § 13.3.3] για περισσότερες λεπτομέρειες). Ο αριθμός των τετραγωνικών ριζών αυξάνεται, αλλά, δεδομένου ότι υπολογίζονται με συγκριτικά μικρότερους πρώτους, η συνολική πολυπλοκότητα παραμένει $O(\ln^4 \hat{N})$ δυαδικές πράξεις. Έτσι, στα παρακάτω λαμβάνονται υπόψη μόνο τα εσφαλμένα πρώτα modulus.

[2]Το GIVARO είναι μια open source C++ βιβλιοθήκη στην GNU Multi-Precision Library.Είναι διαθέσιμη στην ιστοσελίδα http://packages.debian.org/fr/sid/libgivaro-dev

5.7 Κρυπτανάλυση

Ο σκοπός επίθεσης εισαγωγής σφάλματος εναντίον της «*Αριστερά–προς–Δεξιά*» ύψωσης σε δύναμη είναι παρόμοιος με την επίθεση εναντίον του «*Δεξιά–προς–Αριστερά*» [35]. Ο συντελεστής N παροδικά τροποποιείται σε πρώτη τιμή κατά τη διάρκεια του τετραγωνισμού, j_k βήματα πριν από το τέλος της ύψωσης σε δύναμη. Στη συνέχεια, από ένα σωστή / εσφαλμένη ζεύγος υπογραφών $(S,\hat{S}_k)$, η επίθεση έχει ως στόχο να ανακτήσει το μέρος του ιδιωτικού εκθέτη

$$d_{(k)} = \sum_{i=0}^{jk-1} 2^i \cdot d_i$$

το οποίο είχε απομονωθεί από το σφάλμα. Ανατρέχοντας στην παραπομπή [35], η ανάλυση που ακολουθεί μπορεί εύκολα να προσαρμοστεί για σφάλματα που συμβαίνουν πρώτη φορά κατά τη διάρκεια ενός πολλαπλασιασμού.

Λεξικό των Πρώτων Συντελεστών. Το πρώτο βήμα συνίσταται στον υπολογισμό ενός λεξικού από υποψήφιους πρώτους εσφαλμένους συντελεστές $(\hat{N}_i)$. Ο επιτιθέμενος δοκιμάζει όλες τις πιθανές τιμές που λαμβάνονται τροποποιώντας τον *N* σύμφωνα με κάποιο μοντέλο σφάλματος που έχει επιλεγεί. Στη συνέχεια δοκιμάζονται οι υποψήφιες τιμές του $\hat{N}$ χρησιμοποιώντας τον πιθανοθεωρητικό Miller-Rabin αλγόριθμο [50]. Σύμφωνα με τη μελέτη που παρουσιάστηκε παραπάνω, για την παραδοχή σφάλματος τυχαίων byte, το λεξικό εσφαλμένου συντελεστή θα περιέχει 46 καταχωρήσεις κατά μέσο όρο, είτε πρόκειται για 1024-bit είτε για 2048-bit υλοποίηση RSA. Το μέγεθος του λεξικού εξαρτάται από το μοντέλο σφάλματος (βλέπε Πίνακα 1).

Υπολογισμός Τετραγωνικών Ριζών. Για κάθε καταχώρηση $(\hat{N}_i)$ του λεξικού συντελεστών, ο επιτιθέμενος επιλέγει μια υποψήφια τιμή του ανιχνευμένου μέρους του ιδιωτικού εκθέτη $d'_{(k)}$. Τώρα μπορεί να υπολογιστεί[3]:

$$R_{(d'_{(k)},\hat{N}_i)} \equiv \hat{S}_k \cdot m^{-d'_{(k)}} \bmod \hat{N}_i \qquad (15)$$

Για το σωστό ζεύγος $(d_{(k)},\hat{N}),R_{(d_{(k)},\hat{N})}$, το $R_{(d_{(k)},\hat{N})}$ αναμένεται να είναι πολλαπλάσιο τετραγωνικού υπολοίπου (δηλαδή: ένα j_k–οστο τετραγωνικό υπόλοιπο, βλ. παράγραφο 3.3). Συνεπώς, αν $R_{(d'_{(k)},\hat{N}_i)}$ δεν είναι τετραγωνικό υπόλοιπο, ο εισβολέας μπορεί άμεσα να συμπεράνει ότι το υποψήφιο ζεύγος $(d'_{(k)},\hat{N}_i)$ είναι εσφαλμένο. Η δοκιμή τετραγωνικού υπολοίπου μπορεί να γίνει στη παρούσα περίπτωσή, διότι όλες οι προ-υπολογισμένες υποψήφιες τιμές για τον εσφαλμένο συντελεστή είναι πρώτοι αριθμοί. Η δοκιμή βασίζεται στο θεώρημα του Fermat:

[3]Αυτός ο υπολογισμός είναι δυνατός μόνο όταν ο d'k είναι αντιστρέψιμος εντός του $\mathbb{Z}/\mathbb{Z}\hat{N}_i$. Στη παρούσα περίπτωσή το σύνολο των N_i είναι πρώτοι και ο αλγόριθμος του Ευκλείδη πάντα υπολογίζει το αντίστροφο.

Αν

$$\left(R_{(d'_{(k)},\hat{N}_i)}\right)^{(\hat{N}_i-1)/2} \equiv 1 \bmod \hat{N}_i \tag{16}$$

Τότε $R_{(d'_{(k)},\hat{N}_i)}$ είναι τετραγωνικό υπόλοιπο modulo $\hat{N}_i$. Εάν ικανοποιείται η δοκιμή, τότε ο εισβολέας μπορεί να χρησιμοποιήσει τον αλγόριθμο Tonelli και Shanks (βλ. παράγραφο 4.4) για να υπολογίσει τις τετραγωνικές ρίζες του $R_{(d'_{(k)},\hat{N}_i)}$. Προκειμένου να υπολογιστεί η j_k –στη τετραγωνική ρίζα του $R_{(d'_{(k)},\hat{N}_i)}$ το βήμα αυτό αναμένεται να επαναληφθεί j_k –φορές. Εάν μια από τις δοκιμές τετραγωνικού υπολοίπου αποτύχει, τότε το παρόν ζεύγος $(d'_{(k)},\hat{N}_i)$ απορρίπτεται άμεσα και ο υπολογισμός της τετραγωνικής ρίζας ματαιώνεται. Ο επιτιθέμενος πρέπει να επιλέξει άλλο υποψήφιο ζεύγος.

Τελικός Έλεγχος Modular. Ο σκοπός των δύο πρώτων βημάτων είναι να ακυρωθούν οι επιπτώσεις στην εσφαλμένη υπογραφή λόγω της διαταραχής. Τώρα, από την j_k –στη τετραγωνική ρίζα του $R_{(d'_{(k)},\hat{N}_i)}$ ο εισβολέας θα προσομοιώσει μια χωρίς λάθη εκτέλεση, υπολογίζοντας:

$$S' \equiv \left((R_{(d'_{(k)},\hat{N}_i)})^{1/2^{jk}} \bmod \hat{N}_i \right)^{2^{jk}} \cdot m^{d'_{(k)}} \bmod N \tag{17}$$

Τέλος, ελέγχει αν ικανοποιείται η παρακάτω εξίσωση:

$$S' \equiv S \bmod N \tag{18}$$

Όπως και στη «*Δεξιά –προς -Αριστερά*» επίθεση [35], όταν αυτή η τελευταία συνθήκη ικανοποιείται, σημαίνει ότι το υποψήφιο ζεύγος είναι πιθανότατα το προς αναζήτηση ζεύγος (βλ. παράγραφο 5.8.3). Επιπλέον, η γνώση των λιγότερο σημαντικών bits του d που έχουν ήδη βρεθεί, χρησιμοποιείται για αναπαραγωγή της επίθεσης στα επόμενα μυστικά bits. Συνεπώς, ο επιτιθέμενος πρέπει να συλλέξει μια σειρά από εσφαλμένες υπογραφές $\hat{S}_k$ εισάγοντας το σφάλμα σε διαφορετικά j_k βήματα πριν το τέλος της ύψωσης σε δύναμη. Επιπλέον, πολλαπλές εσφαλμένες υπογραφές $\hat{S}_{k,f}$ πρέπει να συγκεντρωθούν για ένα δεδομένο βήμα j_k, για να ληφθεί υπόψη η πιθανότητα να παραχθεί μια εσφαλμένη υπογραφή $\hat{S}_k$ που να υπολογίζεται σε ένα πρώτο $\hat{N}$, δηλαδή εκμεταλλεύσιμη από την κρυπτανάλυση.

Το σύνολο $(\hat{S}_{k,f}, j_k)_{k,f}$ ταξινομείται σε φθίνουσα σειρά τοποθέτησης σφάλματος. Αν εισάγονται σφάλματα τακτικά, κάθε ταξινομημένο ζεύγος χρησιμοποιείται για να

ανακτήσει ένα *l*-bit μέρος του εκθέτη, έτσι ώστε για το *k*–στο ζεύγος $(\hat{S}_{k,f}\ j_k)$, το ανακτηθέν μέρος του d να είναι

$$d_k = \sum_{i=0}^{j_k-1} 2^i \cdot d_i = \sum_{i=0}^{k\cdot l-1} 2^i \cdot d_i \ .$$

Αυτά τα αποτελέσματα μπορούν να εφαρμοστούν σε σφάλματα που δεν εισάγονται τακτικά (δηλαδή: $j_k - j_{k-1} = l_k < l_{max}$). Παρακάτω δίνεται ο αλγόριθμος επίθεσης με περισσότερες λεπτομέρειες.

Αλγόριθμος 3. DFA

είσοδος: το *N*, *m*, η σωστή υπογραφή *S*, το μέγεθος του λεξικού D_{length}, το σύνολο από τα ζεύγη $(\hat{S}_{k,f}, j_k)_{0\le k<n/1, 1\le f\le \mu(Fn)}$

1: Dict=Build_Prime_Dict(N, D_{length});
2: d:=0;
3: **for** k = 0 , … , [*n*/*l*] **do**
4: **for** f = 1 , … , μ(Fn) **do**
5: **for** $d_{(k)}$ = 0, … , 2^l -1 **do**
6: $d' := d_{(k)} \cdot 2^{jk} + d$;
7: **for** i = 1, … , D_{length} **do**
8: R:= $\hat{S}_{k,f} \cdot m^{-d'} \bmod Dict[i]$;
9: R:=Test_And_Tonelli(R, j_k, $\text{Dict}_{[i]}$);
10: **if** R==0 **then**
11: **break;**
12: **else**
13: $S' := R^{2^j k} \cdot m^{d'} \bmod N$
14: **if** S'==S mod N **then**
15: d:=d';
16: **goto** line_3;
17: **end**
18: **end**
19: **end**
20: **end**
21: **end**
22: **end**

έξοδος: ο ιδιωτικός εκθέτης *d*

5.8 Επιδόσεις

5.8.1 Εσφαλμένος Αριθμός

Το παρακάτω μοντέλο σφάλματος βασίζεται στη μετατροπή του συντελεστή *N* ώστε η αντίστοιχη εσφαλμένη τιμή του να είναι πρώτος αριθμός. Στην §5. 4.1, αποδείχτηκε ότι η πιθανότητα να είναι ένας αριθμός των *t*-bit πρώτος, pr_t, μπορεί να οριοθετηθεί. Έστω τώρα ότι, ο αριθμός των λαθών που κάνουν το $\hat{N}$ πρώτο, είναι η τυχαία μεταβλητή F_t. Αυτή η τυχαία μεταβλητή ακολουθεί ένα γεωμετρικό νόμο των πιθανοτήτων. Έτσι, ο μέσος αριθμός των σφαλμάτων για να γίνει ο $\hat{N}$ πρώτος είναι:

$$\frac{1}{Sup(t)} < \mu(Ft) = \frac{1}{pr_t} < \frac{1}{Inf(t)} \tag{19}$$

Για μεγάλες τιμές του *t* (δηλ. τουλάχιστον 1024 ή 2048-bit RSA), μπορεί να χρησιμοποιηθεί το θεώρημα σύνθλιψης (ή σάντουιτς – pinching ή sandwich theorem) για να προσεγγισθεί αυτή η τιμή ασυμπτωτικά:

$$\mu(Ft) \approx \frac{t \cdot \ln^3(2)}{0.480} \approx \frac{1}{Inf(t)} \tag{20}$$

Από μια δοθείσα εσφαλμένη υπογραφή, ο επιτιθέμενος μπορεί να ανακτήσει ένα *l* -bit μέρος του *d*. Υπάρχουν το πολύ *n / l*, τέτοια μέρη για την RSA μεγέθους *n*. Αυτό δείχνει ότι ο μέσος αριθμός των σφαλμάτων που απαιτούνται για ένα ολόκληρο ιδιωτικό κλειδί πλήρη τη σχέση:

$$\text{Αριθμός σφαλμάτων} = O\left(\frac{n^2}{1.441 \cdot l}\right) \text{δοκιμές.} \tag{21}$$

Ο αριθμός αυτός μπορεί να μειωθεί δραματικά εάν ο εισβολέας έχει τη δυνατότητα να επιλέξει τη θέση του byte του σφάλματος (βλ. § 5.4.1) ή αν το μοντέλο σφάλματος είναι μεγαλύτερο (ομαλός συντελεστής / smooth modulus, διαφορετικές στοχευόμενες αρχιτεκτονικές)

5.8.2 Υπολογιστική Πολυπλοκότητα

Δίνουμε τώρα τη συνολική πολυπλοκότητα της επίθεσης. Το μέγεθος του λεξικού, D_{length}, αφήνεται ως μια παράμετρος της επίθεσης αφού ο εισβολέας μπορεί να καθορίσει ένα όριο εφόσον το επιλεγμένο μοντέλο σφάλματος απαιτεί περισσότερους πόρους από όσους μπορεί να πάρει. Σύμφωνα με την προηγούμενη ανάλυση, επιλέγεται (βλ. § 4.1), D_{length} = 46 για ένα τυχαίο byte παραδοχής σφάλματος.

Θεώρημα 1.*Ο Αλγόριθμος 3 είναι σωστός και η μέση πολυπλοκότητα του για μια εσφαλμένη διατάραξη τυχαίου byte του συντελεστή πληροί την:*

$$\text{Cattack} = O\left(\frac{2^{8+l} \cdot n^3 \cdot (n+1)}{16 \cdot l}\right) \text{ υψώσεις σε δύναμη}$$

Απόδειξη. Ορθότητα όπως αποδείχθηκε στην παράγραφο 5. Σε ότι αφορά την πολυπλοκότητα, ο επιτιθέμενος πρέπει να δοκιμάσει όλα τα πιθανά ζεύγη υποψηφίων $(d'_{(k)}, \hat{N}_i)$. Ο αριθμός των ζευγών εξαρτάται από το μέγεθος του λεξικού των πρώτων συντελεστών που σημειώνεται με D_{length} και το παράθυρο ανάκτησης μήκους:

$$|(d'_{(k)}, \hat{N}_i)| = 2^l \cdot D_{length} \tag{22}$$

Για κάθε ζεύγος ο εισβολέας υπολογίζει πρώτα το $R_{(d'_{(k)}, \hat{N}_i)}$ (βλ. (15)), εκτελώντας μια modular ύψωση σε δύναμη του μηνύματος και ένα πολλαπλασιασμό.

Στη συνέχεια, εκτελεί μια σειρά από το πολύ j_k δοκιμές τετραγωνικού υπολοίπου και, για κάθε επιτυχημένη, υπολογίζεται μια τετραγωνική ρίζα. Παρατηρώντας ότι η πιθανότητα να αποτύχει στη δοκιμή ακολουθεί ένα νόμο γεωμετρικών πιθανοτήτων, ο μέσος αριθμός εκτελούμενων δοκιμών[4] είναι $\frac{1}{\Pr[Testfails]} = 2$ Κατά συνέπεια, η μέση πολυπλοκότητα του βήματος αυτού είναι:

$$C_{Square\ roots}(k) = O(2 \cdot C_{Test} + C_{Tonelli\&Shanks}) = O(j_k \cdot n)\exp onentiations \quad (23)$$

Το τελευταίο βήμα της επίθεσης είναι ο τελικός έλεγχος (βλ. (17)). Απαιτεί να υπολογιστούν j_k modular τετράγωνα και μια modular ύψωση σε δύναμη του μηνύματος ακολουθούμενη από έναν πολλαπλασιασμό. Ο τελευταίος υπολογισμός οριοθετείται επίσης από $O(j_k \cdot n)$ υψώσεις σε δύναμη.

Στην περίπτωση λεξικού σταθερού μεγέθους, ο μέσος αριθμός των πρώτων αριθμών αυτού του λεξικού για μια byte τροποποίηση του συντελεστή είναι $N_{\sigma\phi\acute{\alpha}\lambda\mu\alpha\tau\alpha\ \alpha\nu\acute{\alpha}\ \mu\pi\lambda o\kappa} = \frac{2^8 n / 8}{D_{length}}$.

Στη συνέχεια, η επίθεση πρέπει να δοκιμάσει όλες τις συγκεντρωμένες εσφαλμένες υπογραφές προκειμένου να ανακτήσει ολόκληρο τον εκθέτη. Άρα, καθώς το j_k οριοθετείται από $k \cdot l$, η συνολική υπολογιστική πολυπλοκότητα οριοθετείται από:

$$C_{attack} = \sum_{k=0}^{n/l} N_{faults\ per\ blocs} \cdot C_{Square\ roots}(k) \cdot 2^l \cdot D_{length} = O\left(\frac{2^{8+l} \cdot n^3 \cdot (n+1)}{16 \cdot l}\right) \quad (24)$$

Η επίθεση που παρουσιάζεται είναι επομένως περισσότερο εκτενής από τη «*Δεξιά-προς-Αριστερά*» [35], με κύρια αιτία τον επιπλέον αριθμό των εσφαλμένων ζευγών που πρέπει να αναλυθούν, ώστε να πάρουμε έναν πρώτο συντελεστή.

5.8.3 Πιθανότητα Αποδοχής-Σφάλματος

Όπως ορίζεται στο [35], η πιθανότητα αποδοχής-σφάλματος είναι η πιθανότητα ένα λανθασμένο ζεύγος $(d'_{(k)}, \hat{N}_i)$ να ικανοποιεί την (18). Στην περίπτωσή αυτή, ο υπολογισμός του τελικού ελέγχου γίνεται στο Z / NZ και απαιτεί επιπλέον τετράγωνα. Κατά συνέπεια η πιθανότητα αποδοχής-σφάλματος που δίνεται στο [35] θα πρέπει να προσαρμοστεί αντικαθιστώντας το διάστημα έρευνας του $\hat{N}$ από το μήκος του λεξικού D_{length}:

[4] Η δοκιμή αποτυγχάνει όταν η δοκιμαζόμενη τιμή δεν είναι τετραγωνικό υπόλοιπο. Αλλά όλα τα $\hat{N}_i$ είναι πρώτοι αριθμοί. Έστω z_i γεννήτρια στο $\square / \hat{N}_i \square$, όλα τα στοιχεία του συνόλου μπορούν να εκφράζονται σαν δύναμη του z_i. Συνεπώς, το ένα στα δύο στοιχεία είναι δύναμη του z_i^2 και τετραγωνικό υπόλοιπο.

$$0 < \Pr[F.A] < \min\left(\frac{(N-1)\cdot 2^{l}\cdot D_{length}}{N\cdot(2^{l}\cdot D_{length}-1}, \frac{2^{l}\cdot D_{length}}{N}\right) \tag{25}$$

Επιπλέον, λόγω των δοκιμών τετραγωνικού υπολοίπου (βλ. § 5), οι εσφαλμένοι υποψήφιοι μπορεί να απορριφθούν πριν από τον υπολογισμό του τελικού ελέγχου. Άρα, ο τελικός έλεγχος δε θα γίνεται πάντα. Η πιθανότητα ένα λανθασμένο ζεύγος να περάσει όλα τα j_k τεστ δίνεται από την:

$$\Pr[R_{(d'_{(k)},\hat{N}_i)} \text{ is a } j_k - \text{times quadratic residue}] \tag{26}$$

$$\prod_{i=0}^{j_k-1}\Pr\left[\left(R_{(d'_{(k)},\hat{N}_i)}\right)^{1/2^{i}} \text{ is a quadratic residue}\right]$$

$$= \frac{1}{2^{j_k}}$$

Αυτή η πιθανότητα δείχνει ότι, για την ανάκτηση του *k*–στου μέρους του *d*, μόνο ένα στα 2^{jk} λανθασμένα ζεύγη θα περάσει όλες τις δοκιμές τετραγωνικού υπολοίπου. Τελικά η πιθανότητα αποδοχής-σφάλματος μπορεί να έχει άνω όριο:

$$\Pr[F.A] < \min\left(\frac{1}{2^{j_k}}, \frac{(N-1)\cdot 2^{l}\cdot D_{length}}{N\cdot(2^{l}\cdot D_{length}-1)}, \frac{2^{l}\cdot D_{length}}{N}\right) \tag{27}$$

Αυτή η αναπαράσταση δείχνει, καταρχάς, ότι λόγω του τελευταίου όρου $\frac{2^{l}\cdot D_{length}}{N}$, η πιθανότητα αποδοχής-σφάλματος είναι άκρως αμελητέα για μήκος RSA που συνήθως χρησιμοποιείται. Επιπλέον, μπορεί κανείς επωφελώς να παρατηρήσει ότι ο τελικός έλεγχος μπορεί να αποφευχθεί όταν ο αριθμός των διαδοχικών δοκιμών τετραγωνικού υπολοίπου που θα περάσουν είναι αρκετά μεγάλος (δηλ.: $2^{jk} > D_{length}\cdot 2^{l}$).

5.9 Συμπεράσματα

Σε αυτό το κεφάλαιο, έγινε μια γενικευμένη αναφορά επίθεση σφάλματος που παρουσιάστηκε στο [35] σε «*Αριστερά -προς-Δεξιά*» εφαρμογή της RSA, με την παραδοχή ότι ο εσφαλμένος συντελεστής μπορεί να είναι πρώτος αριθμός. Αν και αυτό το μοντέλο έχει ήδη χρησιμοποιηθεί [51], στο παρόν κεφάλαιο προσφέρθηκε μια λεπτομερής θεωρητική ανάλυση που αποδεικνύει ότι ένα τέτοιο μοντέλο σφάλματος δεν είναι μόνο εφικτό, αλλά και με δυνατότητα επέκτασης σε διαφορετικές αρχιτεκτονικές υπολογιστικών συστημάτων. Αυτό το γεγονός δίνει έμφαση στην ανάγκη προστασίας των RSA δημόσιων στοιχείων από αλλοίωση κατά τη διάρκεια της εκτέλεσης υπολογισμών.

Γενικότερα η χρήση ενός εσφαλμένου πρώτου συντελεστή για τον υπολογισμό των τετραγωνικών ριζών σε πολυωνυμικό χρόνο θέτει το ζήτημα της χρήσης σφαλμάτων για την αλλαγή αλγεβρικών ιδιοτήτων της υποκείμενης πεπερασμένης περιοχής. Στο παρόν κεφάλαιο παρουσιάστηκε στη πραγματικότητα μια από τις λίγες εφικτές επιθέσεις κατά του RSA η οποία μάλιστα δύναται να εξελιχθεί καθώς θα ανακαλύπτονται νέες τεχνικές εισαγωγής σφαλμάτων.

6 Επιθέσεις Σφαλμάτων Κατά Υπογραφών EMV

6.1 Σχετικά με τις RSA υπογραφές

Στο Συνέδριο CHES (Cryptographic Hardware & Embedded Systems) του 2009, οι Coron, Joux, Kizhvatov, Naccache και Paillier (CJKNP) παρουσίασαν μια επίθεση σφάλματος κατά υπογραφών RSA με μερικώς γνωστά μηνύματα. Αυτή η επίθεση σφάλματος επιτρέπει την παραγοντοποίηση του δημόσιου συντελεστή *N*. Ενώ το μέγεθος του άγνωστου μέρους του μηνύματος (UMP – Unknown Message Part) αυξάνεται με τον αριθμό των διαθέσιμων εσφαλμένων υπογραφών, η πολυπλοκότητα της επίθεσης CJKNP αυξάνεται εκθετικά με τον αριθμό των εσφαλμένων υπογραφών.

Το κεφάλαιο αυτό περιγράφει μια απλούστερη επίθεση, η πολυπλοκότητα της οποίας παραμένει πολυώνυμο του αριθμού των σφαλμάτων, και άρα, η νέα επίθεση μπορεί να χειριστεί πολύ μεγαλύτερα UMPs. Η νέα τεχνική μπορεί να παραγοντοποιήσει τον *N* σε ένα κλάσμα του δευτερολέπτου χρησιμοποιώντας δέκα εσφαλμένες υπογραφές EMV – στόχος πέρα από την έκταση της CJKNP. Επιδεικνύεται επίσης πώς μπορεί να εφαρμοστεί η επίθεση ακόμα και όταν ο *N* είναι άγνωστος το οποίο είναι συχνό φαινόμενο στις πραγματικές επιθέσεις.

Το σύστημα υπογραφών RSA [73] είναι σίγουρα το πιο διαδεδομένο σε χρήση. Για να υπογράψει ένα μήνυμα *m* με την RSA, ο υπογράφων εφαρμόζει πρώτα μια λειτουργία κωδικοποίησης *μ* στο *m* και στη συνέχεια υπολογίζει την υπογραφή $\sigma = \mu(m)^d \bmod N$. Για να επιβεβαιώσει την υπογραφή, ο αποδέκτης ελέγχει ότι

$$\sigma^e = \mu(m) \bmod N.$$

Το Κινέζικο Θεώρημα Υπολοίπου (Chinese Remainder Theorem – CRT) χρησιμοποιείται συχνά για τη μείωση του υπολογιστικού φορτίου του υπογράφοντος. Αυτό γίνεται υπολογίζοντας:

$$\sigma_p = \mu(m)^d \bmod p \text{ και } \sigma_q = \mu(m)^q \bmod q$$

και η υπογραφή *σ* υπολογίζεται από σ_p και σ_q από το Κινέζικο Θεώρημα Υπολοίπων.

Στο [55], οι Boneh, DeMillo και Lipton έδειξαν ότι οι υλοποιήσεις RSA μπορεί να είναι ευάλωτες σε επιθέσεις σφαλμάτων (δες επίσης [68]). Αν υποτεθεί ότι ο εισβολέας μπορεί να προκαλέσει σφάλμα όταν υπολογίζεται το σ_q διατηρώντας παράλληλα τον υπολογισμό του σ_p σωστό, τότε

$$\sigma_p = \mu(m)^d \bmod p \text{ και } \sigma_q \neq \mu(m)^q \bmod q$$

και η συνακόλουθη (εσφαλμένη) σ υπογραφή ικανοποιεί την

$$\sigma^e = \mu(m) \bmod p \text{ και } \sigma^e \neq \mu(m) \bmod q.$$

Με την οποία ο εισβολέας μπορεί στη συνέχεια να παραγοντοποιήσει τον *N* με

$$\gcd(\sigma^e - \mu(m) \bmod N, N) = p. \quad (1)$$

Είναι εύκολο να διαπιστωθεί ότι επίθεση σφάλματος κατά Bonehetal εφαρμόζεται σε κάθε ντετερμινιστική κωδικοποίηση RSA, π.χ. την Full Domain Hash (FDH) [54] κωδικοποίηση όπου $\sigma = H(m)^d \bmod N$ και $H : \{0,1\}^* \mapsto Z_N$ είναι μια συνάρτηση κατακερματισμού. Η επίθεση αυτή εφαρμόζεται επίσης σε πιθανοθεωρητικά συστήματα υπογραφών, όπου ο τυχαίος παράγοντας που χρησιμοποιείται για τη δημιουργία της υπογραφής αποστέλλεται μαζί με την υπογραφή, όπως και στο σύστημα υπογραφών PFDH [60]. Ωστόσο, αν ο τυχαίος παράγοντας ανακτάται μόνο κατά τον έλεγχο επαλήθευσης της υπογραφής, ή εάν κάποιο μέρος του μηνύματος είναι άγνωστο, η επίθεση αποτυγχάνει. Για παράδειγμα, θεωρήστε μια υπογραφή $\sigma = (m \| r)^d \bmod N$. Η τυχαία r ανακτάται μόνο όταν επαληθεύεται μια σωστή υπογραφή. Δοθείσης μιας εσφαλμένης υπογραφής ο επιτιθέμενος δεν μπορεί να ανακτήσει την r , ούτε να συναγάγει το $(m \| r)$, το οποίο θα ήταν αναγκαίο για να υπολογιστεί το $\gcd(\sigma^e - (m \| r) \bmod N, N) = p$.

Στο Συνέδριο CHES του 2009, οι Coron, Joux, Kizhvatov, Naccache και Paillier (CJKNP) έδειξαν πώς μπορεί να επεκταθεί η επίθεση σε RSA υπογραφές κατά Bonehetal. με μερικώς άγνωστα μηνύματα (ή άγνωστους nonces) [57]. Η CJKNP επίθεση παρουσιάστηκε και επεξηγήθηκε με μια πιθανοτική παραλλαγή του προτύπου ISO / IEC 9796-2 [66], όπως χρησιμοποιείται στις προδιαγραφές EMV [63]. Στο ISO / IEC 9796-2 το κωδικοποιημένο μήνυμα έχει τη μορφή:

$$\mu(m) = \mathbf{6A_{16}} \| m[1] \| H(m) \| \mathbf{BC_{16}}$$

όπου το $m = m[1] \| m[2]$, χωρίζεται σε δύο μέρη. Οι CJKNP δείχνουν ότι αν το άγνωστο μέρος του $m[1]$ δεν είναι πολύ μεγάλο (π.χ. λιγότερα από 160 bits για έναν RSA συντελεστή 2048-bit), τότε μια και μόνη εσφαλμένη υπογραφή επιτρέπει την παραγοντοποίηση του N όπως στην κατά Bonehetal. επίθεση. Η CJKNP επίθεση βασίζεται σε μια τεχνική η οποία αποδίδεται στους Herrmann και May [62] για την εύρεση μικρών ριζών γραμμικών εξισώσεων modulo έναν άγνωστο παράγοντα p του N. Η ίδια η [62] βασίζεται στην τεχνική Coppersmith [56] για την εύρεση μικρών ριζών πολυωνυμικών εξισώσεων χρησιμοποιώντας τον αλγόριθμο LLL [71]. Επιπλέον, η [57] εισήγαγε μια πολλαπλών - σφαλμάτων επίθεση χρησιμοποιώντας μια επέκταση της επίθεσης Coppersmith. Τα πολλαπλά σφάλματα καταστούν δυνατή την επίθεση σε μεγαλύτερα άγνωστα μέρη μηνύματος (UMPs). Ωστόσο, αυτό έρχεται σε βάρος μιας εκθετικής πολυπλοκότητας του αριθμού των σφαλμάτων.

Σε αυτό το κεφάλαιο περιγράφεται μια απλούστερη επίθεση πολλαπλών σφαλμάτων. Η πολυπλοκότητα της νέας επίθεσης είναι πολυώνυμο του αριθμού των εσφαλμένων υπογραφών. Αυτό επιτρέπει τη διαχείριση μεγαλύτερων UMPs τα οποία ήταν πέραν των δυνατοτήτων της [57]. Για παράδειγμα, σε μια τυπική περίπτωση χρήσης των EMV, δέκα εσφαλμένες υπογραφές είναι αρκετές για να παραγοντοποιήσουν τον N σε ένα κλάσμα του δευτερολέπτου με τη νέα επίθεση, ενώ η επίθεση της [57] ήταν εντελώς αδύνατη σε μια τέτοια κατάσταση.

Επιδεικνύεται τέλος, ότι μια παρόμοια τεχνική θα μπορούσε ακόμη και να ανακτήσει τον N από μια συλλογή έγκυρων υπογραφών, έτσι ώστε η επίθεση να μπορεί να εφαρμοστεί σε πρωτόκολλα των οποίων οι δημόσιες παράμετροι RSA δεν είναι διαθέσιμες σε τρίτους, κάτι που προκύπτει στην πράξη (π.χ. ιδιόκτητες τραπεζικές κάρτες ή e-διαβατήρια).

6.2 Επίθεση Coron – Joux – Kizhvatov – Naccache – Paillier

Στην [57] μελετάται μια τυχαιοποιημένη (randomized) έκδοση του προτύπου ISO / IEC 9796-2. Το ISO / IEC 9796 – 2 είναι ένα πρότυπο κωδικοποίησης που επιτρέπει μερική ή ολική ανάκτηση μηνυμάτων [66, 67]. Η κωδικοποίηση μπορεί να χρησιμοποιηθεί με συναρτήσεις σύνοψης (hash functions) $H:\{0,1\}^{*}\to\{0,1\}^{k_h}$ διαφόρων συλλεγόμενων μεγεθών k_h. Όταν το k_h, το μέγεθος του m και το μέγεθος του N (φερόμενο ως k) είναι όλα πολλαπλάσια του 8, τότε η κατά ISO / IEC 9796-2 κωδικοποίηση ενός μηνύματος $m = m[1] \,||\, m[2]$ είναι

$$\mu(m) = 6A_{16} \,||\, m[1] \,||\, H(m) \,||\, BC_{16}$$

όπου $m[1]$ αποτελείται από τα $k - k_h - 16$ αριστερότερα bits του m και $m[2]$ αντιπροσωπεύει τα εναπομείναντα bits του m. Στο [67] απαιτείται ότι $k_h \geq 160$. Το ίδιο ισχύει και στις προδιαγραφές EMV [63]. Σημειώνεται ότι μια λειτουργική επίθεση πλαστογραφίας (χωρίς σφάλματα) έναντι του ISO / IEC 9796-2 περιγράφτηκε πρόσφατα στο [59], επεκτείνοντας την επίθεση του [58]. Ωστόσο η επίθεση έχει πρακτική αξία μόνο όταν το $m[1]$, μπορεί να επιλέγεται πλήρως από τον αντίπαλο, πράγμα που δε συμβαίνει στην τεχνολογία EMV και στην τυχαιοποιημένη εκδοχή του προτύπου ISO / IEC 9796-2 όπως εξετάζεται στο παρόν κεφάλαιο. Πιο συγκεκριμένα, η [4] θεωρεί ένα μήνυμα $m = m[1] \,||\, m[2]$ του τύπου

$$m[1] = a \,||\, r \,||\, a',\ m[2] = DATA$$

όπου το r είναι ένα μέρος του μηνύματος άγνωστο στον αντίπαλο (UMP), τα α και α' strings γνωστά στον αντίπαλο και το DATA κάποιο γνωστό ή άγνωστο string. Το μέγεθος του r συμβολίζεται με k_r και το μέγεθος του $m[1]$ είναι $k - k_h - 16$, όπως απαιτείται στο ISO / IEC9796-2. Τότε το κωδικοποιημένο μήνυμα είναι

$$\mu(m) = \mathbf{6A_{16}} \,\|\alpha\|\, r \,\|\alpha'\|\, H(\alpha \,||\, r \,\|\alpha'\|\, \mathrm{DATA}) \,||\, \mathbf{BC_{16}} \qquad (2)$$

Επομένως, τόσο το r όσο και το $H(\alpha \,||\, r \,||\, \alpha' \,||\, \mathrm{DATA})$ είναι άγνωστα και έτσι ο συνολικός αριθμός των άγνωστων bits μέσα στο $\mu(m)$ είναι $k_r + k_h$.

6.3 Χωριστή Επίθεση Σφάλματος (Single Fault Attack)

Η [574] περιγράφει μια επίθεση σφάλματος κατά του προηγούμενου συστήματος υπογραφών. Πιο συγκεκριμένα, κάποιος μπορεί να υποθέσει ότι μετά την εισαγωγή σφάλματος ο αντίπαλος έχει μια εσφαλμένη υπογραφή σ τέτοια ώστε:

$$\sigma^e = \mu(m) \bmod p,\ \sigma^e \neq \mu(m) \bmod q \qquad (3)$$

Από τη (2) μπορεί κανείς να γράψει

$$\mu(m) = t + r \cdot 2^{n_r} + H(m) \cdot 2^8 \qquad (4)$$

όπου t είναι μια γνωστή τιμή. Από την (3) μπορεί κανείς να πάρει:

$$\sigma^e = t + r \cdot 2^{n_r} + H(m) \cdot 2^8 \bmod p\,.$$

Αυτό δείχνει ότι το (*r*, *H* (*m*)) θα πρέπει να είναι μια λύση της εξίσωσης

$$a + b \cdot x + c \cdot y = 0 \bmod p \quad (5)$$

όπου $a := t - \sigma^e \bmod N$, $b := 2^{n_r}$ και $c := 2^8$ γνωστοί. Αυτή η εξίσωση δυο μεταβλητών (x,y) έχει μια μικρή ρίζα (x_0, y_0) = (*r*, *H* (*m*)). Για την επίλυση αυτής της εξίσωσης, μπορεί κανείς να χρησιμοποιήσει ένα αποτέλεσμα από τους Herrmann και May [62] βασισμένο στην τεχνική Coppersmith για εξεύρεση μικρών ριζών των πολυωνυμικών εξισώσεων [56].

Η τεχνική Coppersmith χρησιμοποιεί την LLL για τη λήψη δύο πολυωνύμων h_1 (*x*,*y*) και h_2 (*x*,*y*) τέτοιων ώστε

$$h_1 (x_0, y_0) = h_2 (x_0, y_0) = 0$$

αναβάλλει τους ακέραιους. Στη συνέχεια μπορεί κανείς να υπολογίσει τη συνισταμένη μεταξύ των h_1 και h_2 για να ανακτήσει την κοινή ρίζα (x_0, y_0). Για να καταλήξει κανείς εκεί, πρέπει να υποθέσει ότι h_1 και h_2 είναι αλγεβρικά ανεξάρτητα. Αυτή η adhoc υπόθεση, κάνει τη μέθοδο επαγωγικών συλλογισμών σε μη αλγοριθμικές διαδικασίες, η οποία ωστόσο καταλήγει να λειτουργεί αρκετά καλά στην πράξη. Στη συνέχεια, δεδομένης της ρίζας (x_0,y_0) μπορεί κανείς να ανακτήσει το τυχαιοποιημένο κωδικοποιημένο μήνυμα *μ* (*m*) και να παραγοντοποιήσει το *N* με GCD.

Υποθέτοντας ότι $r<N^{\gamma}$ και $H(m) \le N^{\delta}$, για ένα ισορροπημένο RSA συντελεστή κανείς παίρνει τη συνθήκη:

$$\gamma + \delta \le \frac{\sqrt{2} - 1}{2} \cong 0.207 \quad (6)$$

Αυτό σημαίνει ότι για ένα 1024-bit συντελεστή *N*, το συνολικό μέγεθος των αγνώστων x_0 και y_0 μπορεί να είναι το πολύ 212 bits. Για ISO / IEC 9796-2 υπογραφές με k_h = 160, το άγνωστο *r* μπορεί συνεπώς να είναι το πολύ 52 bits μακρύ.

6.4 Επέκταση σε Πολλά Σφάλματα Modulo του Ίδιου Συντελεστή

Η [57] δείχνει τον τρόπο να επεκταθεί η επίθεση σε πολλαπλά σφάλματα, προκειμένου να βελτιωθεί το όριο στο μέγεθος του UMP. Πιο συγκεκριμένα, δεδομένων *l* λαθών, παίρνει κανείς ένα σύστημα εξισώσεων:

$$\alpha_i + b \cdot x_i + c \cdot y_i = 0 \mod p$$

για $1 \le i \le l$, όπου α_i, *b* και *c* είναι γνωστές και x_i και y_i είναι άγνωστες και μικρές. Ο στόχος εξακολουθεί να είναι η ανάκτηση του *p*. Σημειώνεται ότι μπορεί να υποτεθεί ότι *b* = 1 πολλαπλασιάζοντας τις εξισώσεις με $b^{-1} \bmod N$.

Η [4] θεωρεί ένα πιο γενικό σύστημα όπου αντί των γνωστών σταθερών *b* και *c*, κανείς θεωρεί τις γνωστές b_i και c_i. Πιο συγκεκριμένα, δίνονται *l* διαφορετικά πολυώνυμα

$$f_u (x_u, y_u) = \alpha_u + x_u + c_u y_u \quad (7)$$

όπου κάθε πολυώνυμο f_u έχει μια μικρή ρίζα (ξ_u, v_u) modulo*p* με $|\xi_u| \le X$ και $|v_u| \le Y$. Σημειώνεται ότι, όπως και στη βασική περίπτωση, ομαλοποιείται εκ νέου κάθε πολυώνυμο f_u για να εξισωθεί ο συντελεστής x_u του f_u με ένα.

Η [57] δείχνει πώς μπορεί να επεκταθεί η επίθεση Coppersmith σε αυτές τις πολλαπλές πολυωνυμικές εξισώσεις, αποκτώντας έτσι ένα καλύτερο όριο για το UMP μέγεθος. Θεωρητικά, δεδομένου ενός επαρκώς μεγάλου αριθμού σφαλμάτων, η εκτεταμένη επίθεση θα μπορούσε να αντιμετωπίσει περιπτώσεις όπου $\gamma + \delta$ βρίσκονται ασυμπτωτικά κοντά στο ½. Ωστόσο, η πολυπλοκότητα της επίθεσης αυξάνεται εκθετικά με τον αριθμό των σφαλμάτων l, αποσκοπώντας έτσι σε τιμές του $\gamma + \delta$ σημαντικά μεγαλύτερες από το ενιαίο μέγιστο σφάλμα του 0,207 που είναι εντελώς ανέφικτο. Στον Πίνακα 2 απεικονίζεται πόσο δυσεπίλυτο γίνεται το πρόβλημα καθώς το $\gamma + \delta$ προσεγγίζει το ½.

6.5 Μια Νέα Επίθεση Πολλαπλών Σφαλμάτων

Η προηγούμενη επίθεση έχει εφαρμογή μόνο για ένα μικρό αριθμό σφαλμάτων, διότι η διάσταση του πίνακα αυξάνεται εκθετικά με τον αριθμό των σφαλμάτων. Αυτή η ενότητα περιγράφει μια διαφορετική επίθεση που μπορεί να επωφεληθεί από ένα μεγάλο αριθμό σφαλμάτων και επομένως να χειριστεί πολύ μεγαλύτερα UMP. Πράγματι, στη νέα επίθεση, η διάσταση του πίνακα παραμένει ίση με τον αριθμό των σφαλμάτων, συν ένα.

Όπως και προηγουμένως, θεωρείται μια λειτουργία κωδικοποίησης που δίνεται από την εξίσωση (4)

$$\mu(m) = t + r \cdot 2^{n_r} + H(m) \cdot 2^8$$

Δοσμένου ενός συνόλου από ελαττωματικές υπογραφές σ_i τέτοιες ώστε:

$$\sigma_i^e = \mu(m_i) = t + r_i \cdot 2^{n_r} + H(m) \cdot 2^8 \bmod p$$

παίρνουμε ένα σύνολο εξισώσεων της μορφής

$$A_i + B \cdot x_i + D \cdot y_i = 0 \bmod p$$

όπου $A_i := t - \sigma^e_i \bmod N$, $B := 2^{nr}$ και $D := 2^8$ είναι γνωστές. Όπως και σε προηγούμενη ενότητα, μπορούμε να υποθέσουμε ότι $B = 1$, πολλαπλασιάζοντας τις εξισώσεις με $B^{-1} \bmod N$. Αυτό έχει ως αποτέλεσμα τις ακόλουθες εξισώσεις

$$a_i + x_i + c \cdot y_i = 0 \bmod p \qquad (8)$$

για $1 \le i \le l$, όπου l είναι ο αριθμός των ελαττωματικών υπογραφών. Σημειώνεται ότι σε αντίθεση με τις εξισώσεις (7) της προηγούμενης ενότητας, εδώ υπάρχει σταθερός συντελεστής c[5].

Η νέα επίθεση είναι παρόμοια με εκείνη στο [16]. Εφαρμόζοντας LLL [71] στη διάσταση που εκτείνεται από τις στήλες του ακόλουθου πίνακα

[5]Η επίθεση σε αυτή την ενότητα δε θα μπορούσε να λειτουργήσει με διαφορετικά c_i .

$$\begin{pmatrix} \kappa\alpha_1 & \kappa\alpha_2 & \cdots & \kappa\alpha_l & \kappa N \\ 1 & 0 & \cdots & 0 & 0 \\ & 1 & \cdots & . & . \\ & & . & . & . \\ & & & 0 & 0 \\ & & & 1 & 0 \end{pmatrix} \tag{9}$$

για μια επαρκώς μεγάλη σταθερά *κ* (όπως περιγράφεται στο [70]), ο εισβολέας υπολογίζει ένα σύντομο διάνυσμα $(u_1,..., u_l)$ έτσι ώστε

$$\sum_{i=1}^{l} u_i \cdot a_i = 0 \bmod N$$

Αυτό υπονοεί από την (8)

$$\sum_{i=1}^{l} u_i \cdot x_i + c \cdot \left(\sum_{i=1}^{l} u_i \cdot y_i \right) = 0 \bmod p$$

επιτρέποντας

$$a_0 := \sum_{i=1}^{l} u_i \cdot x_i \ \ \kappa\alpha\iota \ \ \beta_0 := \sum_{i=1}^{l} u_i \cdot y_i \tag{10}$$

το οποίο δίνει

$$a_0 + c \cdot \beta_0 = 0 \bmod p$$

Συνεπώς το διάνυσμα (α_0, β_0) ανήκει στη συσχέτιση.

$$L\ (c, p) = \{(\alpha, \beta) \quad \in Z^2 \mid \alpha + c \cdot \beta = 0 \bmod p \} \tag{11}$$

Δεδομένου ότι τα x_i και y_i είναι μικρά, φαίνεται ότι, αν τα u_i είναι μικρά, τότε το (α_0, β_0) είναι ένα σύντομο διάνυσμα στη συσχέτιση *L* (*c*, *p*). Πιο συγκεκριμένα, έστω *u* το μικρότερο μη μηδενικό διάνυσμα του *L* (*c*, *p*). Εάν τα u_i είναι αρκετά μικρά ώστε $||(\alpha_0, \beta_0)|| < ||u||$, τότε εξ ορισμού του *u*, πρέπει $\alpha_0 = \beta_0 = 0$. Σε αυτή την περίπτωση λαμβάνεται:

$$\sum_{i=1}^{l} u_i \cdot x_i = \sum_{i=1}^{l} u_i \cdot y_i = 0$$

πράγμα που σημαίνει ότι το γνωστό διάνυσμα $(u_1,..., u_l)$ είναι ορθογώνιο (στο Z) στα δύο άγνωστα διανύσματα $(x_1,...,x_l)$ και $(y_1,...,y_l)$.

Στην πραγματικότητα, η LLL μείωση της συσχέτισης (9) αποδίδει πολλά άλλα διανύσματα (u_i), τα οποία είναι ορθογώνια στο Z τόσο στα (x_i) όσο και στα (y_i). Αν υποτεθεί ότι μπορούν να κατασκευαστούν $l - 2$ τέτοια διανύσματα, γίνεται στη συνέχεια εφικτή μια δισδιάστατη συσχέτιση, η οποία να περιέχει και τα δύο διανύσματα $x = (x_i)$ και $y = (y_i)$. Έστω ότι τα ***x'*** και ***y'*** αποτελούν τη βάση αυτή τη συσχέτιση. Μια τέτοια

βάση μπορεί να προκύπτει εφαρμόζοντας την για δεύτερη φορά την LLL στη συσχέτιση που εκτείνεται από τις στήλες του:

$$\begin{pmatrix} \kappa' u_{1,1} & \cdots & \kappa' u_{1,l} \\ . & & . \\ . & & . \\ \kappa' u_{l-2,1} & \cdots & \kappa' u_{l-2,l} \\ 1 & & \\ & & 1 \end{pmatrix}$$

για μια επαρκώς μεγάλη σταθερά ***κ '*** .

Έστω τώρα ένα διάνυσμα $u = (u_i)$ που είναι ορθογώνιο modulo*N* τόσο στο ***x '*** όσο και στο ***y '*** , έτσι ώστε:

$$\sum_{i=1}^{l} \upsilon_i \cdot x'_i = 0 \bmod N, \quad \sum_{i=1}^{l} \upsilon_i \cdot y'_i = 0 \bmod N$$

Τότε, εφόσον τα ***x*** και ***y*** ανήκουν στο συσχέτιση που εκτείνεται από τα ***x '*** και ***y '*** , θα πρέπει να ισχύει

$$\begin{cases} x = a \cdot x' + \beta \cdot y' \\ y = a' \cdot x' + \beta' \cdot y' \end{cases}$$

για κάποια ***α***, ***α '***, ***β***, ***β '*** ∈ Z. Από αυτό συνεπάγεται ότι:

$$\sum_{i=1}^{l} \upsilon_i \cdot x_i = 0 \bmod N, \quad \sum_{i=1}^{l} \upsilon_i \cdot y_i = 0 \bmod N$$

Τότε από την εξίσωση (8) αυτό δίνεται:

$$\sum_{i=1}^{l} \upsilon_i \cdot a_i = 0 \bmod p \qquad (12)$$

Ως εκ τούτου $\gcd\left(\sum_i \upsilon_i \cdot a_i, N\right) = p$. Σημειώνεται ότι ο προηγούμενος υπολογισμός μπορεί να απλοποιηθεί αν ληφθούν υπόψη τα τρία μόνο πρώτα στοιχεία των ***x '*** και ***y '***. Σε αυτή την περίπτωση, προκύπτει ένα μοναδικό (μέχρι κάποια πολλαπλασιαστική σταθερά) τρισδιάστατο διάνυσμα *u* ορθογώνιο modulo*N* τόσο για τα τρία πρώτα στοιχεία του ***x '*** όσο και για τα τρία πρώτα στοιχεία του ***y '*** . Τότε, η εξίσωση (12) εξακολουθεί ισχύει και, όπως και πριν $\gcd\left(\sum_i \upsilon_i \cdot a_i, N\right) = p$

Απομένει να αιτιολογηθεί το γιατί δύναται $||(\alpha_0, \beta_0)|| < ||u||$, όπου *u* το συντομότερο μη μηδενικό διάνυσμα του *L* (*c*, *p*). Παρέχεται μια επιχειρηματολογία ανάλογη της [72] (βλ. επίσης [61] για υψηλότερες διαστάσεις συσχέτισης). Ορίζεται μια συσχέτιση *L* (*c*, *p*) να είναι *B*-καλή, αν κάποιο μη μηδενικό διάνυσμα έχει νόρμα>*B*, διαφορετικά λέγεται ότι η συσχέτιση είναι *B*-κακή. Θεωρούμε σταθερό πρώτο αριθμό *p*. Εξ ορισμού της συσχέτισης *L* (*c*, *p*) στην (11), η τιμή του *c*modulo*p* καθορίζεται από κάθε μη μηδενικό διάνυσμα στο*L* (*c*, *p*). Δεδομένου ότι υπάρχουν το πολύ $4B^2$ διανύσματα στο δίσκο της ακτίνας *B* , υπάρχουν το πολύ $4B^2$ συσχετίσεις *L* (*c*, *p*) οι οποίες είναι *B*-κακές. Επομένως, για ένα τυχαίο *c* modulo *p*, η πιθανότητα να είναι μια συσχέτιση Β-κακή

είναι το πολύ $4B^2 / p$. Λαμβάνοντας $B := \sqrt{p/3}$, η πιθανότητα να είναι μια συσχέτιση B-κακή γίνεται το πολύ 1/2.

Επομένως μια συσχέτιση είναι *B*-καλή με πιθανότητα τουλάχιστον ½ . Από αυτό συνεπάγεται ότι αν $||(a_0, \beta_0)|| \leq \sqrt{p/3}$, τότε με πιθανότητα τουλάχιστον ½ το διάνυσμα (α_0, β_0) είναι μικρότερο από το μικρότερο μη μηδενικό διάνυσμα του *L* (*c*, *p*), το οποίο συνεπάγεται ότι $\alpha_0 = \beta_0 = 0$, όπως απαιτείται.

Εδώ έχει θεωρηθεί ένα σταθερό *p* και ένα τυχαίο *c* modulo *p*. Στη παρούσα επίθεσή όμως το *c* είναι σταθερός ακέραιος και το *p* τυχαίο, ως εκ τούτου η παραπάνω ανάλυση είναι μόνο heuristic. Γενικότερα, αν οι ακέραιοι α_0 και β_0 έχουν διαφορετικά μεγέθη, προκύπτει ότι $\alpha_0 = \beta_0 = 0$ υπό τη συνθήκη ότι:

$$|a_0| \cdot |\beta_0| \prec \frac{p}{3} \tag{13}$$

Χρησιμοποιώντας την LLL αναμένεται να ληφθούν διανύσματα (u_i) με νόρμα περίπου $N^{1/l}$, όπου *l* ο αριθμός των εσφαλμένων υπογραφών (δες [69]). Έστω *X* και *Y* τα άνω όρια για τους αγνώστους x_i και y_i. Έτσι λαμβάνεται από τη (10)

$$|a_0| \leq N^{1/l} \cdot X, \; |\beta_0| \leq N^{1/l} \cdot Y$$

Από τη (13) λαμβάνεται το ακόλουθο όριο

$$N^{2/l} \cdot X \cdot Y \prec \frac{p}{3}$$

Με $X = N^{\gamma}$ και $Y = N^{\delta}$ αυτό αποδίδει περίπου

$$\frac{2}{l} + \gamma + \delta \prec \frac{1}{2} \tag{14}$$

Άρα, για ένα μεγάλο αριθμό σφαλμάτων *l* υπολογίζεται το ίδιο ασυμπτωτικό όριο $\gamma + \delta <$ ½ όπως και στην [57]. Ωστόσο η διάσταση πίνακα στην (9) είναι μόνο *l* + 1 αντί να είναι εκθετική στο *l* όπως στην [57]. Στη παράγραφο 6.7.1 παρατίθεται το αποτέλεσμα των πρακτικών προσομοιώσεων επικυρώνοντας την επίθεση. Στη συνέχεια εφαρμόζεται η νέα τεχνική στις προδιαγραφές EMV στη παράγραφο 6.7.2.

6.6 Ανακτώντας Άγνωστα Moduli

Σε πολλές πρακτικές καταστάσεις, ο συντελεστής *N* ενδέχεται να μην είναι στη διάθεση του εισβολέα. Ακριβώς το αντίθετο συμβαίνει συχνά στις ιδιόκτητες (μη διαθέσιμες προς πώληση) υλοποιήσεις του RSA που προορίζονται για αποκλειστική χρήση. Η τεχνική που περιγράφεται στην προηγούμενη ενότητα μπορεί να προσαρμοστεί για την ανάκτηση *άγνωστων* moduli *N* από *ορθές* υπογραφές όταν ο δημόσιος εκθέτης *e* είναι μικρός. Όταν το *N* έχει ανακτηθεί, μπορεί κανείς να εφαρμόσει στη συνέχεια την επίθεση σφάλματος που περιγράφεται στην προηγούμενη ενότητα.

Όπως και προηγουμένως, θεωρείται μια λειτουργία κωδικοποίησης που δίνεται από την εξίσωση (4)

$$\mu(m) = t + r \cdot 2^{n_r} + H(m) \cdot 2^8$$

Δεδομένου ενός συνόλου *l* από ορθές υπογραφές σ_i τέτοιες ώστε

$$\sigma_i^e = \mu(m_i) = t + r_i \cdot 2^{n_r} + H(m_i) \cdot 2^8 \bmod N$$

λαμβάνεται ένα σύνολο από *l* εξισώσεις της μορφής

$$A_i + B \cdot x_i + D \cdot y_i = 0 \bmod N \qquad (15)$$

όπου $A_i := t - \sigma_i^e$,$B := 2^{n_r}$ και $D := 2^8$ είναι γνωστοί, αλλά x_i , y_i και *N* άγνωστοι. Σημειώνεται ότι σε αντίθεση με την προηγούμενη ενότητα, το A_i δε μειώνεται modulo *N*, οπότε το bit μέγεθος του A_i είναι περίπου $e \cdot \log_2 N$.

Όπως και στην προηγούμενη ενότητα, χρησιμοποιώντας την LLL μπορεί να βρεθεί ένα συντομευμένο διάνυσμα (u_i) τέτοιο ώστε

$$\sum_{i=1}^{l} u_i \cdot A_i = 0$$

στο Z. Αυτό συνεπάγεται από την (15)

$$B \cdot \left(\sum_{i=1}^{l} u_i \cdot x_i \right) + D \cdot \left(\sum_{i=1}^{l} u_i \cdot y_i \right) = 0 \bmod N$$

Όπως και προηγουμένως, αν τα (u_i) είναι αρκετά μικρά, τότε λαμβάνεται $\sum_i u_i \cdot x_i = \sum_i u_i \cdot y_i = 0$ στο Z. Τότε, πάλι από *l* - 2 γραμμικά ανεξάρτητα διανύσματα (u_i) μπορεί κανείς να ανακτήσει ένα δισδιάστατο πίνακα που περιέχει τα δύο διανύσματα (x_i) και (y_i).

Στη συνέχεια υπολογίζονται δύο διανύσματα $\boldsymbol{u_1}$ και $\boldsymbol{u_2}$ τα οποία είναι και τα δύο ορθογώνια στο Z σε κάθε διάνυσμα σε αυτό το δισδιάστατο πίνακα. Αυτό συνεπάγεται ότι αμφότερα τα διανύσματα είναι ορθογώνια στο Z στα δύο διανύσματα (x_i) και (y_i). Η εξίσωση (15) συνεπάγεται ότι $\boldsymbol{u_1}$ και $\boldsymbol{u_2}$ είναι και τα δύο ορθογώνια modulo *N* προς το διάνυσμα (A_i), άρα για να ανακτηθεί το *N,* υπολογίζεται απλώς το GCD των αντίστοιχων τους μονόμετρων προϊόντων με το διάνυσμα (A_i).

Δεδομένου ότι η νόρμα του διανύσματος (A_i) είναι περίπου N^e, αναμένεται (βλ. [69]) να βρεθεί ένα διάνυσμα (u_i) νόρμας $\cong N^{e/(l-1)}$. Επιπλέον, αφήνοντας $a_0 = \sum_i u_i \cdot x_i$ και $\beta_0 = \sum_i u_i \cdot y_i$, όπως και στην προηγούμενη ενότητα, πρέπει να ληφθεί $|a_0| \cdot |\beta_0| \prec \frac{N}{3}$ έτσι ώστε $a_0 = \beta_0 = 0$ ισχύει με αρκετή (heuristic) πιθανότητα. Αυτό δίνει το ακόλουθο όριο κατά προσέγγιση

$$N^{2e/(l-1)} \cdot X \cdot Y \prec \frac{N}{3}$$

δηλαδή χρησιμοποιώντας τις προηγούμενες παραστάσεις

$$\frac{2e}{l-1}+\gamma+\delta \prec 1 \tag{16}$$

και τον απαιτούμενο αριθμό ορθών υπογραφών:

$$l \succ \frac{2e}{1-\gamma-\delta}+1$$

Με άλλα λόγια, ο αριθμός των απαιτούμενων υπογραφών είναι ανάλογος με το δημόσιο εκθέτη *e*, πράγμα που σημαίνει ότι η τεχνική ανάκτησης του συντελεστή είναι πρακτική μόνο για μικρούς δημόσιους εκθέτες. Τότε λειτουργεί καλά στην πράξη, όπως αποδεικνύεται στη παράγραφο 6.7.2 παρακάτω.

6.7 Αποτελέσματα Εξομοίωσης

6.7.1 Επίθεση Πολλαπλών Σφαλμάτων

Η επίθεση σφάλματος που περιγράφεται στη παράγραφο 6.5 προσομοιώνεται ως εξής: πρώτα δημιουργείται μια σωστή $\sigma_p = \mu(m)^d \bmod p$ και ένα τυχαίο $\sigma_q \in Z_q$ και μετά υπολογίζεται η ελαττωματική υπογραφή σ χρησιμοποιώντας το CRT. Αυτή μιμείται τη διαδικασία που περιγράφεται στο [4] όπου τα συγκεκριμένα λάθη εισάγονται σε συσκευές που γεννούν τυχαιοποιημένες ISO/IEC9796-2 υπογραφές.

Τα αποτελέσματα της προσομοίωσης συνοψίζονται στον Πίνακα 1. Υπολογίζεται το ποσοστό επιτυχίας της επίθεσης, για $\gamma + \delta = 1/3$ για 12, 13 και 14 λάθη. Η θεωρία προβλέπει επιτυχία με καλές πιθανότητες όταν $l > 12$. Ο Πίνακας 1 επιβεβαιώνει αυτή την πρόβλεψη τόσο για ισόρροπες όσο και για ασύμμετρες γ και δ διαμορφώσεις.

Ο Πίνακας 2 παρέχει μια σύγκριση με επίθεση πολλαπλών σφαλμάτων της [4]. Για μεγάλα *l*, η νέα επίθεση έχει την ασυμπτωτική προϋπόθεση $\gamma + \delta < ½$, που ταυτίζεται με το θεωρητικό ασυμπτωτικό όριο της παραλλαγής πολλαπλών σφαλμάτων της [57]. Ωστόσο, είναι πολύ πιο εύκολο να αντιμετωπιστούν περιπτώσεις όπου το γ + δ προσεγγίζει το ½ με τη νέα επίθεση απ' ότι είναι στην [57].

Πίνακας 3 Αποτελέσματα Προσομοίωσης Επίθεσης με Χρήση SAGE. Τυχαία 1024-bit moduli. 2,5 GHz Intel CPU core.

Αριθμός Σφαλμάτων *l*	12	13	14
Ποσοστό επιτυχίας με $\gamma = \delta = 1/6$	13%	100%	100%
Ποσοστό επιτυχίας με $\gamma = ¼$, $\delta = 1/12$	0%	100%	100%
Μέσος CPU χρόνος (δευτερόλεπτα)	0.19	0.14	0.17

Δηλαδή, όπως φαίνεται στον Πίνακα 2, όταν $\gamma + \delta$ προσεγγίζει το μισό της διάστασης του πίνακα της [57], η επίθεση καθίσταται εντελώς ανέφικτη. Συγκεκριμένα, αποδεικνύεται στο κεφάλαιο 6 ότι η νέα επίθεση δίνει τη δυνατότητα για επίθεση σε EMV μορφές υπογραφής, κάτι που ήταν πέρα από ότι η [57] μπορούσε να κάνει.

Ωστόσο, πρέπει να υπογραμμιστεί ότι για μικρότερες $\gamma + \delta$ τιμές, η [57] μπορεί να είναι πιο πρακτική, διότι απαιτεί λιγότερες εσφαλμένες υπογραφές. Για παράδειγμα για $\gamma + \delta = 0.214$ απαιτούνται μόνο 2 ελαττωματικές υπογραφές αντί για οκτώ στη νέα επίθεση.

Με άλλα λόγια, οι δύο τεχνικές συμπληρώνουν με πολύ ωραίο τρόπο η μια την άλλη και παρέχουν στον επιτιθέμενο μια εργαλειοθήκη που του επιτρέπει να προσαρμόσει την τεχνική του στις στοχευόμενες διαμορφώσεις γ και δ.

Πίνακας 4 Σύγκριση της νέας επίθεσης με την [4], για τυχαίο 1024-bit συντελεστή

$\gamma + \delta$	l_{new}	ω_{new}	Χρόνος CPU (νέος)	l_{old}	ω_{old}	Χρόνος CPU (παλιός)
0.204	7	8	0.03 s	3	84	49 s
0.214	8	9	0.04 s	2	126	22 min
0.230	8	9	0.04 s	2	462	-
0.280	10	11	0.07 s	6	6188	-
0.330	14	15	0.17 s	8	2^{21}	-
0.400	25	26	1.44 s	-	-	-
0.450	70	71	36.94 s	-	-	-

Επεξηγηματικές σημειώσεις σχετικά με τον Πίνακα 2: Ο Πίνακας 2 παρέχει, για διάφορες τιμές του $\gamma + \delta$, τις ακόλουθες πληροφορίες: τον αριθμό των ελαττωματικών υπογραφώνl_{new} που χρησιμοποιούνται στην προσομοίωσή, την αντίστοιχη διάσταση πίνακα ω_{new}, και το χρόνο εκτέλεσης της νέας επίθεσης. Για την επίθεση που περιγράφεται στο [57], ο πίνακας παραθέτει την ελάχιστη διάσταση πλέγμα ω_{old} που απαιτείται για την επίλυση των $\gamma + \delta$ τιμών που έχουν θεωρηθεί και τον αντίστοιχο αριθμό ελαττωματικών υπογραφών l_{old}. Βρίσκεται το ω_{old} μέσω εξαντλητικής αναζήτησης στις παραμέτρους (l, t, m) με l<50, m< 80. Για $\gamma + \delta$ = 0.214 και $\gamma + \delta$ = 0.23, μπορεί κανείς πραγματικά να ξεφύγει με ελαφρώς μικρότερες διαστάσεις πίνακα από τις αναγραφόμενες στον πίνακα (120 και 378 αντί για 126 και 462) με το τίμημα των περισσότερων σφαλμάτων (7 και 13 αντίστοιχα).

6.7.2 Ανάκτηση Άγνωστων Moduli

Η τεχνική που περιγράφεται στην παράγραφο 6.6 εφαρμόζεται και για την ανάκτηση του N από ορθές υπογραφές (όταν N είναι άγνωστο στον επιτιθέμενο). Όπως φαίνεται στον Πίνακα 3η επίθεση είναι αρκετά πρακτική για δημόσιο εκθέτη (e) με μικρές τιμές. Πιο συγκεκριμένα, δίνονται τα ποσοστά επιτυχίας για $\gamma + \delta$ = 1/3 με 10 έως 13 έγκυρες υπογραφές για e = 3. Εν προκειμένω, το θεωρητικό όριο (16) προβλέπει ότι η τεχνική θα πρέπει να πετύχει με καλή πιθανότητα όταν l> 10. Αυτό επαληθεύεται πολύ καλά τόσο για ισόρροπες όσο και για ασύμμετρες γ και δ διαμορφώσεις.

Πίνακας 5 Προσομοίωση Ανάκτησης Συντελεστή στη SAGE. Τυχαία 1024-bit moduli και e = 3. 2,5 GHz Intel CPU core

Αριθμός υπογραφών l	10	11	12	13
Ποσοστό επιτυχίας με $\gamma = \delta$ = 1/6	2%	59%	61%	61%
Ποσοστό επιτυχίας με γ = ¼, δ = 1/12	2%	62%	64%	64%
Μέσος CPU χρόνος (δευτερόλεπτα)	0.20	0.21	0.25	0.31

6.8 Εφαρμογή σε Υπογραφές EMV

6.8.1 Οι προδιαγραφές EMV

Η EMV είναι μια συλλογή από τις προδιαγραφές της βιομηχανίας για τη διαλειτουργικότητα των καρτών πληρωμής, των τερματικών POS και των ATM. Οι προδιαγραφές EMV [63] χρησιμοποιούν ISO/IEC 9796-2 υπογραφές για την πιστοποίηση δημόσιων-κλειδιών και για την επικύρωση των δεδομένων. Για παράδειγμα, για να επικυρώσει αυτή καθαυτή, η κάρτα πληρωμών πρέπει να εκδώσει μια υπογραφή σε δεδομένα που παρέχονται από το τερματικό. Ο αλγόριθμος υπογραφής είναι RSA με ISO/IEC 9796-2 που χρησιμοποιεί $e = 3$ ή $e = 2^{16} + 1$. Το μήκος των bit όλων των moduli είναι πάντα πολλαπλάσιο του 8. Η EMV χρησιμοποιεί ειδικά format μηνυμάτων. Χρησιμοποιούνται 7 διαφορετικά format, ανάλογα με τον τύπο του μηνύματος.

Στη συνέχεια, για λόγους σαφήνειας, αναλύουμε ένα μόνο από αυτά τα format: το *Offline Dynamic Data Authentication, Dynamic Application Data* format, που περιγράφεται στο Βιβλίο 2, Κεφάλαιο 6.5, Πίνακας 15, σελίδα 67 των EMV προδιαγραφών [10]. Το προς υπογραφή αντικείμενο είναι η κάρτα. Το μήνυμα m που θα υπογραφεί έχει το format$m = m\,[1] \,||\, m\,[2]$ με:

$$m\,[1] = 0501_{16} || \, L_{DD} \,||\, \text{ICCDD} \,||\, BB_{16} \cdots BB_{16}$$

$$m\,[2] = \text{DATA}$$

όπου L_{DD} είναι το byte που προσδιορίζει το μήκος (σε bytes) του ICC Dynamic Datastring ICCDD, και DATA είναι κάποια στοιχεία / δεδομένα που παρέχονται από το τερματικό. Σε γενικές γραμμές, το ICC Dynamic Datastring έχει την ακόλουθη μορφή:

$$\text{ICCDD} = L_{ICCDN} \,||\, \text{ICCDN} \,||\, \text{ADD}$$

όπου L_{ICCDN} είναι ένα byte που προσδιορίζει το μήκος (σε bytes) της μεταβλητής χρόνου ICC Dynamic Number ICCDN, και ADD αποτελείται από L_{DD} - L_{ICCDN} - 1 bytes Additional Dynamic Data που θα υπογραφούν. Διευκρινίζεται ότι κάποιος πρέπει να έχει $2 \le L_{ICCDN} \le 8$.

Όπως αναφέρθηκε στις προδιαγραφές EMV, ο ICC Dynamic Number μπορεί να είναι ένας απρόβλεπτος αριθμός ή ένα μετρητής που προσαυξάνεται για κάθε νέα υπογραφή. Σε μια τυπική περίπτωση χρήσης (όπως περιγράφεται, για παράδειγμα, ως μέρος του EMV Test 2CC.086.1 Case 07 [11]), το ICCDN είναι ένα τυχαίο string 8-byte που παράγεται από την κάρτα, και ADD είναι μεταβλητή 8-byte string, που κωδικοποιείται σύμφωνα με το [12]. Σε αυτή την περίπτωση, έχουμε:

$$m\,[1] = 0501_{16} \,||\, 11_{16} \,||\, 08_{16} \,||\, \text{ICCDN} \,||\, \text{ADD} \,||\, BB_{16} \cdots BB_{16}$$

η οποία μπορεί να ξαναγραφτεί ως:

$$m\,[1] = \mathbf{X} \,||\, r \,||\, BB_{16} \cdots BB_{16}$$

όπου **X** είναι μια γνωστή τιμή και r είναι μια μεταβλητή byte string με bit-size$k_r = 128$. Αυτό δίνει:

$$\mu(m) = 6A_{16} \,||\, \mathbf{X} \,||\, r \,||\, BB_{16} \cdots BB_{16} \,||\, H(m) \,||\, BC_{16} \qquad (17)$$

όπου H (m) είναι μια 160-bit σύνοψη του κωδικοποιημένου μηνύματος m. Σημειώνεται ότι η άνευ λάθους επίθεση πλαστογραφίας από την [59] δεν εφαρμόζεται διότι εδώ το m [1] δεν μπορεί να ελεγχθεί από τον αντίπαλο.

6.8.2 Επίθεση Σφαλμάτων

Το EMV format για το μ (m) που δίνεται στη (17) είναι το ίδιο με αυτό που εξετάζεται στην [57] και υπενθυμίζεται επίσης στο κεφάλαιο 2, και το ίδιο με αυτό που εξετάζεται στη νέα μας επίθεση στο κεφάλαιο 3. Στη συγκεκριμένη περίπτωση χρήσης (use case) που περιγράφεται παραπάνω, το string **X** είναι γνωστό αλλά οι μεταβλητές r και H (m) είναι άγνωστες στον εισβολέα. Συνεπώς, ο συνολικός αριθμός των άγνωστων bits είναι:

$$k_r + k_h = 128 + 160 = 288$$

Ως εκ τούτου, για ένα 1024-bit συντελεστή N, έχουμε:

$$\gamma + \delta = \frac{288}{1024} \approx 0.28$$

η οποία είναι πολύ πέρα από το φάσμα της πρακτικής εφαρμογής της [57], όπως φαίνεται στον Πίνακα 2. Ωστόσο, όπως φαίνεται στον Πίνακα 2 η νέα επίθεση θα παραγοντοποιήσει τον N σε ένα κλάσμα του δευτερόλεπτου χρησιμοποιώντας περίπου δέκα εσφαλμένες υπογραφές.

7 Επίλογος

Παρά τις πάνω από τρεις δεκαετίες που έχουν περάσει από την πρώτη εμφάνιση του RSA, έχει αποδειχτεί απίστευτα σθεναρός απέναντι σε επιθέσεις και εξονυχιστικούς ελέγχους από θεωρητικούς της κρυπτογραφίας. Οι επιθέσεις χρονισμού μάλιστα, που είναι το ένα από τα ελάχιστα είδη υλοποιήσιμων επιθέσεων εναντίον κρυπτοσυστημάτων όπου γίνεται ορθή χρήση του RSA, δεν προκύπτουν από κάποια εγγενή αδυναμία των αλγορίθμων. Πέρα από αυτό, υπάρχουν πλέον ευρέως γνωστές και άκρως αποδοτικές άμυνες εναντίων τέτοιου είδους επιθέσεων.

Το δεύτερο είδος υλοποιήσιμων επιθέσεων, οι επιθέσεις «κολλήματος» -glitching attacks- οδηγούν σε παραβίαση της ασφάλειας του RSA αφού καταλήγουν σε παραγοντοποίηση του modulus. Ειδικότερα, όλες οι συσκευές που εφαρμόζουν τον RSA με χρήση του Κινέζικου Θεωρήματος Υπολοίπων (CRT), είναι εκτεθειμένες σε τέτοιου είδους επιθέσεις.

Και οι δυο παραπάνω επιθέσεις έχουν κοινό σημείο το γεγονός ότι έστω και προσωρινά πρέπει να παραβιαστεί η ακεραιότητα του συστήματος ψηφιακών υπογραφών στο φυσικό επίπεδο. Για παράδειγμα πρέπει η έξυπνη κάρτα (smartcard) που υλοποιεί τον RSA με σκοπό την αυθεντικοποίηση του κατόχου, να πέσει στα χέρια του εισβολέα/κλέφτη. Αν αυτός διαθέτει εξοπλισμό αρκετών εκατοντάδων χιλιάδων ευρώ, και γνώσεις επιπέδου διδακτορικού, μπορεί να τρέφει ελπίδες, αν ο κάτοχος της κάρτας δεν δηλώσει την απώλεια. Επιθέσεις τέτοιου τύπου έχουν ερευνηθεί μόνο στο θεωρητικό επίπεδο και πραγματικά κρούσματα δεν έχουν αναφερθεί.

Ο αλγόριθμος RSA είναι το πρότυπο για τη κρυπτογραφία δημοσίου κλειδιού και σε αντίθεση με πιο εξελιγμένα σχήματα, παρέχει υπηρεσίες κρυπτογράφησης αλλά και ψηφιακών υπογραφών (αυθεντικοποίησης). Επίσης είναι διαθέσιμος στο κοινό (τα πνευματικά δικαιώματα έχουν πλέον λήξει) με την προϋπόθεση ότι τηρούνται οι κατά τόπους νομοθεσίες για το επιτρεπόμενο μήκος κλειδιού (μέγιστο 256 bit στην Αμερική) όσον αφορά εμπορικά προϊόντα. Η ευρύτατη διάδοση φαίνεται και από το δικτυακό πρωτόκολλο SSL χάρη στο οποίο διεξάγονται καθημερινά δεκάδες εκατομμύρια ασφαλείς συνεδρίες μέσω του διαδικτύου και πάνω από το δημόσιο δίκτυο τηλεπικοινωνιών. Αν και το πρωτόκολλο καθαυτό στη παλαιότερη (2.0) έκδοση του έχει παραβιαστεί, αυτό οφείλεται στην αδυναμία άλλων απαρχαιωμένων πρωτοκόλλων όπως το DNS. Άρα η επίθεση αυτή δεν εμπίπτει καν στη κατηγορία παραβίασης των κανονισμών λειτουργίας των πρωτοκόλλων γιατί απλά δεν αφορά τον RSA αλλά υποστηρικτικά πρωτόκολλα που μπορούν να αντικατασταθούν.

Αν δεν υπάρξει μια συγκλονιστική ανακάλυψη όσον αφορά τη παραγοντοποίηση αριθμών ή σε κάποιο άλλο πεδίο της θεωρίας αριθμών που θα φέρει την επανάσταση στη κρυπτανάλυση, ο RSA θα παραμείνει στη πρωτοκαθεδρία της ασύμμετρης κρυπτογραφίας για πολλά χρόνια ακόμα.

Αυτό συμβαίνει γιατί οι δημιουργοί του σοφά ταύτισαν τον αλγόριθμο με ένα πολύ δύσκολο μαθηματικό πρόβλημα : την αντιστροφή μονόδρομων συναρτήσεων. Για να αντιστραφεί μια τέτοια συνάρτηση με σύγχρονα υπολογιστικά μέσα με τη μέθοδο της εξαντλητικής αναζήτησης (brute force) και για το ελάχιστο προτεινόμενο μήκος κλειδιού των 512 bit, χρειάζονται δεκαετίες, λαμβάνοντας υπόψη και την αύξηση της υπολογιστικής ισχύος (αξίζει σε αυτό το σημείο να αναφερθεί πως ούτε οι κβαντικοί υπολογιστές με τη προβλεπόμενη επεξεργαστική ισχύ τους αναμένεται να αποτελέσουν σημαντική απειλή).

Τελικά, δεν έχει προκύψει σε καμία περίπτωση αδυναμία του ίδιου του αλγόριθμου όταν αυτός εφαρμόζεται σωστά. Αυτό σημαίνει ότι αν το κλειδί ικανοποιεί τη προϋπόθεση του ελάχιστου μήκους και δεν ανήκει στο σύνολο των λεγόμενων «αδύναμων κλειδιών» που επιτρέπουν την εύκολη αποκρυπτογράφηση (εδώ σημειώνεται ότι τέτοια κατηγορία κλειδιών υπάρχει και στους συμμετρικούς αλγόριθμους όπως ο DES), σε καμία περίπτωση δεν πρόκειται να υπάρξει επιτυχής επίθεση κατά του κρυπτογραφικού αλγόριθμου.

Συνεπώς όλο το βάρος περνά στις επιλογές που κάνουν οι οργανισμοί που υλοποιούν τον RSA. Σε αρκετές εφαρμογές όπως είναι οι έξυπνες κάρτες και οι ασύρματες ταυτότητες (RFID), το κλειδί δεν έχει το προβλεπόμενο μήκος γιατί κάτι τέτοιο θα αύξανε τη πολυπλοκότητα των υπολογισμών αρά και το κόστος αλλά και τη χρονική καθυστέρηση. Οι οργανισμοί που προχωρούν σε τέτοιου είδους «εκπτώσεις» γνωρίζουν πολύ καλά ότι εκτίθενται σε επιθέσεις μικρού εκθετικού αλλά προφανώς έχουν κάνει ανάλυση του κόστους, φτάνοντας στο συμπέρασμα ότι τα επίπεδο ασφάλειας είναι επαρκές και η απαραίτητη τεχνογνωσία δεν είναι για την ώρα ευπρόσιτη στους εισβολείς. Εξάλλου οι έξυπνες κάρτες και οι ασύρματες ταυτότητες είναι αναλώσιμα ηλεκτρονικά με ελεγχόμενο κόστος.

Κλείνοντας, αξίζει να σημειωθεί ότι ο RSA είναι στο επίκεντρο του ενδιαφέροντος λόγω της συνεχιζόμενης έρευνας που γίνεται για τη δημιουργία ασφαλών συστημάτων ηλεκτρονικής ψηφοφορίας. Η σημασία των συστημάτων αυτών είναι μεγαλύτερη και από αυτή των κρυπτοσυστημάτων που χρησιμοποιούν τα οικονομικά ιδρύματα αφού τυχόν παραβίαση τους μπορεί να αλλάξει τη μοίρα ενός κράτους ή και ολόκληρου του πλανήτη αν το κράτος αυτό είναι η Αμερική ή η Κίνα. Το σχετικό πλεονέκτημα του RSA είναι ότι σαν μαθηματική συνάρτηση διαθέτει την πολλαπλασιαστική ιδιότητα επιτρέποντας σε μια ψήφο να λάβει την υπογραφή όχι μόνο του ψηφοφόρου αλλά και της εφορευτικής επιτροπής και του εκλογικού κέντρου. Οι αντίπαλοι του RSA από την άλλη υπερτονίζουν τις ανάγκες τους σε επεξεργαστική ισχύ εφόσον σε αυτή την εφαρμογή δεν υπάρχουν περιθώρια για εκπτώσεις στο μήκος του κλειδιού.

8 ΠΑΡΑΡΤΗΜΑ

ΜΑΘΗΜΑΤΙΚΑ ΑΠΟΤΕΛΕΣΜΑΤΑ ΠΟΥ ΧΡΗΣΙΜΟΠΟΙΟΥΝΤΑΙ ΣΤΗ ΠΑΡΟΥΣΑ ΕΡΓΑΣΙΑ

Τα αποτελέσματα που παρατίθενται παρακάτω μπορούν μόνο συμπληρωματικά να δράσουν για τα κυριότερα μαθηματικά εργαλεία που αναφέρονται στην εργασία. Η κρυπτογραφία ως πεδίο γνώσης είναι άμεσα συνδεδεμένη με τις πλέον σύγχρονες θεωρίες των μαθηματικών και ειδικότερα με τη Θεωρία Αριθμών. Έχουν παραλειφθεί αρκετά σημαντικά αποτελέσματα της σύγχρονης Άλγεβρας με κριτήριο το γεγονός ότι ο αναγνώστης μπορεί να τα εντοπίσει ευκολότερα στη βιβλιογραφία.

8.1 Κινέζικο Θεώρημα Υπολοίπων

Το θεώρημα αυτό επιτρέπει την επίλυση συστημάτων γραμμικών ισοδυναμιών. Πιο συγκεκριμένα: Έστω $m_1,\ldots,m_n$ θετικοί ακέραιοι αριθμοί οι οποίοι είναι ανά ζεύγη σχετικά πρώτοι και έστω οι ακέραιοι $\alpha_1,\ldots,\alpha_n$. *Το σύστημα γραμμικών ισοδυναμιών :*

$$x \equiv \alpha_1 \ (\mathrm{mod} m_1)$$

.

.

.

$$x \equiv \alpha_n \ (\mathrm{mod} m_n),$$

έχει μοναδική λύση modulo $M = \prod_{i=1}^{n} m_i$ *. Επιπλέον αν* $x = \sum_{i=1}^{n} a_i M_i (M_i^{-1} \bmod m_i) \bmod M$ $M_i = M / m_i$ *για κάθε i=1,…,n, η λύση δίνεται από :*

$$x = \sum_{i=1}^{n} a_i M_i (M_i^{-1} \bmod m_i) \bmod M$$

Σε ολόκληρη την εργασία το παραπάνω αποτέλεσμα θα αναφέρεται ως το ***Κινέζικο Θεώρημα Υπολοίπων (Chinese Remainder Theorem – CRT)*** και θα χρησιμοποιείται η έκφραση ***Κινέζικο Υπόλοιπο*** για να υπογραμμιστεί η εφαρμογή αυτού. Συνήθως χρησιμοποιείται ο αλγόριθμος του Garner (βλ. Menezes, van Oorschot και Vanstone) για να βρεθεί τελικό αποτέλεσμα, το οποίο έχει ιδιαίτερη σημασία για τον RSA μιας και οι υπολογισμοί στο πεδίο Z_N μπορούν πρώτα να γίνουν στα Z_P και Z_Q και εν συνεχεία τα αποτελέσματα, (μέσω του **Κινέζικου Θεωρήματος**) να συνδυαστούν ώστε να βρεθεί η λύση στο Z_N.

8.2 Απλοποίηση Πινάκων κατά LLL

Κάθε πίνακας Λ με διαστάσεις dim(Λ)≥2 έχει ένα άπειρο πλήθος από διανύσματα βάσης. Κάποιες όμως από αυτές κρίνονται «καλύτερες» από άλλες (ανάλογα πάντα με την εφαρμογή) αλλά συνήθως προτιμώνται οι ελαχιστοποιημένες. Οι βάσεις αυτές αποτελούνται από κανονικοποιημένα διανύσματα τα οποία κατατάσσονται κατά αύξουσα σειρά (π.χ. $||b_1|| < \ldots < ||b_m||$). Η ελαχιστοποίηση της βάσης πίνακα ή απλά η *ελαχιστοποίηση βάσης* είναι μια διαδικασία με την οποία παράγεται μια ελαχιστοποιημένη βάση από μια ήδη δοσμένη.
Ο πρώτος αλγόριθμος ελαχιστοποίησης αποδίδεται στο Gauss και αφορά πίνακες δυο διαστάσεων. Ο αλγόριθμος αυτός μετατρέπει κάθε βάση του Λ σε βάση διανυσμάτων b_1, b_2, ώστε το b_1 να είναι το πιο μικρό διάνυσμα και η συνιστώσα του b_2 που είναι παράλληλη στο b_1 να έχει μήκος το πολύ 1/2. Η νέα βάση ονομάζεται πλέον ελαχιστοποιημένη κατά Gauss. Ο αλγόριθμος του Gauss έχει χρόνο εκτέλεσης ανάλογο της τετάρτης δύναμης της εισόδου και δίνεται παρακάτω:

Είσοδος : b_1 , b_2

1. **repeat**
2. **if** $||b_1|| > ||b_2||$ **then**
3. swap b_1 and b_2
4. **end if**
5. $b_2 \leftarrow b_2 - \lceil \mu \rfloor b_1 \;\; \mu \leftarrow \dfrac{\langle b_1, b_2 \rangle}{\|b_1\|^2}$
6. $b_2 \leftarrow b_2 - \lceil \mu \rfloor b_1$
7. **until** $||b_1|| < ||b_2||$

Έξοδος : b_1 , b_2 κατά Gauss ελαχιστοποιημένα

Μια σημαντική κλάση από ελαχιστοποιημένες βάσεις είναι οι ***ελαχιστοποιημένες κατά Lovasz*** *ή πιο απλά* ***LLL-ελαχιστοποιημένες*** *βάσεις.* Αν $b_1,\ldots,b_m$ τα διανύσματα βάσης του Λ και τα $b^*_1,\ldots,b^*_m$ τα ορθογωνοποιημένα κατά Gram-Schmidt. Η βάση $b_1,\ldots,b_m$ λέγεται LLL ελαχιστοποιημένη αν οι παράγοντες Gram-Schmidt ικανοποιούν την $|\mu_{i,j}| \le 1/2$ για $1 \le j < i \le n$ και

$$\left\| b_i^* + \mu_{i,i-1} b_{i-1}^* \right\|^2 \ge \frac{3}{4} \left\| b_{i-1}^* \right\|^2 \quad \text{για } 1 < i \le n,$$

ή ισοδύναμα

$$\left\| b_i^* \right\|^2 \ge \left(\frac{3}{4} - \mu_{i,i-1}^2 \right) \left\| b_{i-1}^* \right\|^2 \quad \text{για } 1 < i \le n,$$

Τα διανύσματα $b_i^* + \mu_{i,i-1} b_{i-1}^*$ και b_{i-1}^* είναι οι προβολές αντίστοιχα των b_i και b_{i-1} στο ορθογώνιο συμπλήρωμα του $\{b_1,\ldots,b_{i-2}\}$.

Μια πολύ επωφελής ιδιότητα μιας κατά LLL ελαχιστοποιημένης βάσης είναι (όπως θα φανεί και από το παρακάτω θεώρημα) ότι προσδιορίζει ότι υπάρχουν όρια για κάθε ένα από τα διανύσματα βάσης και αυτά τα όρια εξαρτώνται μόνο από τις διαστάσεις του

πίνακα. Το αποτέλεσμα ισχύει για πίνακες ακεραίων $\Lambda \subseteq Z^n$ το οποίο απλά μεταφράζεται στο ότι το κάθε διάνυσμα του πίνακα έχει μόνο ακέραιες συνιστώσες.

Θεώρημα : Έστω $b_1, ..., b_m$ μια κατά LLL ελαχιστοποιημένη βάση του πίνακα $\Lambda \subseteq Z^n$, τότε:

$$\|b_1\| \leq 2^{(m-1)/4} vol(L)^{1/m}$$

και όταν, $\|b_1\| \geq 2^{(l-2)/2}$ τα λοιπά διανύσματα βάσης θα ικανοποιούν:

$$\|b_l\| \leq 2^{(m+l-2)/4} vol(L)^{1/(m-l+1)}$$

Το όριο για το μικρότερο διάνυσμα βάσης βρέθηκε από τους Lenstra, Lenstra και Lovasz [145] ενώ τα όρια για τα άλλα διανύσματα βάσης βρέθηκαν από τον Proos [196].
Σε περίπτωση που ισχύει $\|b_1\| < 2^{(l-2)/2}$, το όριο που δύναται να χρησιμοποιηθεί είναι το :

$$\|b_l\| \leq 2^{\frac{m(m-1)}{4(m-l+1)}} vol(L)^{1/(m-l+1)}$$

όπως έχει αποδειχτεί από τους Blömer και May [21].

Θεώρημα : Έστω $b_1, ..., b_m$ μια κατά LLL ελαχιστοποιημένη βάση του πίνακα $\Lambda \subseteq Z^n$. Τότε για κάθε $x \in L, x \neq 0$,

$$\|b_1\| \leq 2^{(m-1)/2} \|x\|$$

Επιπρόσθετα σε αυτές τις ιδιότητες πρέπει να τονιστεί ότι οι κατά LLL ελαχιστοποιημένες βάσεις μπορούν να υπολογιστούν εύκολα χάρη στον αλγόριθμο του Lovasz (αλγόριθμος ελαχιστοποίησης του Lovasz – αλγόριθμος LLL) [145]. Για πίνακα m διαστάσεων που αποτελείται από n-διαστατα διανύσματα, (π.χ. $\Lambda \in Z^n$), ο LLL αλγόριθμος εμφανίζει χρόνο εκτέλεσης $O(nm^5B^3)$ όπου το B περιορίζεται από το μέγεθος (δηλαδή το μήκος σε bit) των διανυσμάτων βάσης εισόδου. Αν και ο χρόνος εκτέλεσης είναι πολυώνυμο του μήκους της εισόδου (της αρχικής βάσης του πίνακα), ο αλγόριθμος αποδεικνύεται ανεπαρκής για πολύ μεγάλους πίνακες ή όταν τα συνιστάμενα διανύσματα είναι πολύ μεγάλα. Ο πλέον γνωστός και συνάμα γρήγορος αλγόριθμος που υπολογίζει κατά LLL ελαχιστοποιημένες βάσεις είναι ο L^2 αλγόριθμος των Stehle και Nguyen [182] με χρόνο εκτέλεσης ανάλογο με $O(nm^4(m+B)B)$ και ο οποίος αποδίδει πολύ καλά όταν τα διανύσματα βάσης είναι πολύ μεγάλα.

8.3 Μέθοδοι του Coppersmith

Για την εύρεση μικρών λύσεων σε συγκεκριμένες μη γραμμικές εξισώσεις μπορούν να χρησιμοποιηθούν τα αποτελέσματα του Coppersmith [50,51,52]. Πιο συγκεκριμένα, χάρη στο Coppersmith μπορούν πλέον να υπολογιστούν οι μικρές ρίζες πολυωνύμων δύο μεταβλητών στο Z και οι μικρές ρίζες πολυωνύμων μιας μεταβλητής επί του Z_N για N με άγνωστη παραγοντοποίηση.

Και οι δυο παραπάνω μέθοδοι έχουν επεκταθεί στην επίλυση γενικευμένων πολυωνύμων πολλών μεταβλητών τόσο στο Z όσο και στο Z_N αποδίδοντας καλά αποτελέσματα χωρίς όμως να έχει αποδειχτεί μαθηματικά η ορθότητα της πρακτικής αυτής.

Στη περίπτωση πολυωνύμου βαθμού d μιας μεταβλητής όπου έστω N ακέραιος με άγνωστη παραγοντοποίηση :

$$f_N(x) = x^d + a_{d-1}x^{d-1} + \ldots + a_2x^2 + a_1x + a_0 \in \mathbb{Z}[x]$$

Ο σκοπός είναι να βρεθούν σε επαρκή βαθμό όλα τα $|x_0| < X$ που ικανοποιούν :

$$f_N(x_0) \equiv 0 \pmod{N}$$

για όσο μεγαλύτερο όριο του X είναι δυνατό.
Σε ορισμένες περιπτώσεις είναι δυνατό να υπολογιστούν μικρές λύσεις της τμηματικοποιημένης εξίσωσης λύνοντας απλά την ακέραια εξίσωση $f_N(x) = 0$. Για παράδειγμα, όταν $f_N(x) = x^d - a_0$, είναι δυνατό δυνατόν να βρεθούν όλες οι λύσεις ως το όριο $X = N^{1/d}$. Αυτό συμβαίνει επειδή μπορεί να βρεθεί $|x_0| < X = N^{1/d}$ υπολογίζοντας απλά τις d-οστες ρίζες του α_0 επί των ακεραίων. Γενικότερα, αν κάθε συντελεστής του $f_N(x)$ ικανοποιεί :

$$|a_i| < \frac{N^{(1-\frac{i}{d})}}{(d+1)},$$

τότε όλες οι λύσεις $|x_0| < X = N^{1/d}$ μπορούν να βρεθούν επιλύοντας την $f_N(x) = 0$ επί των ακεραίων, αφού το N διαιρεί το $f_N(x_0)$ και :

$$|f_N(x_0)| \le \sum_{i=0}^{d} |a_i| |x_0^i| < \sum_{i=0}^{d} \frac{N^{(1-i/d)}}{d+1} N^{i/d} = N.$$

Επειδή η παραπάνω σχέση είναι περιοριστική χρησιμοποιείται μια πιο επιτυχής συνθήκη η οποία προέρχεται από το θεώρημα των Howgrave-Graham και η οποία επιτρέπει στις λύσεις της $f_N(x) \equiv 0 (\mathrm{mod}\, N)$ να είναι επίσης λύσεις της $f_N(x) = 0$ [111].

Θεώρημα : Έστω $h(x) \in \mathbf{Z}[x]$ το άθροισμα το πολύ ω σε πλήθος πολυωνύμων. Για οποιοδήποτε $X > 0$, αν υπάρχει $|x_0| < X$ ώστε $h(x_0) \equiv 0 (\mathrm{mod}\, N)$ και $\|h(xX)\| < \frac{1}{\sqrt{\omega}} \mathrm{N}$, τότε το x_0 είναι ρίζα του $h(x)$ στο σύνολο Z. Δηλαδή $h(x_0) = 0$.

Γενικά, τα περισσότερα πολυώνυμα δεν θα έχουν τόσο μικρούς συντελεστές ώστε να ικανοποιείται η συνθήκη του παραπάνω θεωρήματος. Είναι παρ' όλα αυτά δυνατό μερικές φορές να κατασκευαστεί ένα νέο πολυώνυμο το οποίο θα έχει τις ίδιες ρίζες με το αρχικό αλλά με επαρκώς μικρούς συντελεστές που θα ικανοποιούν τη παραπάνω συνθήκη. Χρησιμοποιώντας δηλαδή το πολυώνυμο $f_N(x)$ είναι δυνατόν να

κατασκευαστεί πίνακας του οποίου τα συνιστούντα διανύσματα αντιστοιχούν στο σύνολο τους σε διάνυσμα συντελεστών ενός πολυωνύμου που έχει ρίζα $x_0 \bmod uloN$. Υπολογίζοντας μια κατά LLL ελαχιστοποιημένη βάση του πίνακα αυτού, το μικρότερο διάνυσμα αυτής θα αντιστοιχεί στο διάνυσμα συντελεστών κάποιου πολυωνύμου με ρίζα το $x_0 \bmod uloN$. Όλα τα διανύσματα στο πίνακα αυτό θα έχουν την ίδια ιδιότητα και θα έχουν επίσης μικρούς συντελεστές επειδή τα διανύσματα καθαυτά έχουν μικρό μέτρο. Αν οι συντελεστές αυτού του μικρού και κανονικοποιημένου πολυωνύμου είναι αρκετά μικροί για να ικανοποιούν τη συνθήκη του παραπάνω θεωρήματος, είναι δυνατό πλέον να υπολογιστεί το x_0 λύνοντας στο πεδίο Z την $f(x)=0$ χρησιμοποιώντας μάλιστα κάποιο από τους ήδη γνωστούς αλγόριθμους εύρεσης ριζών πολυωνύμων μιας μεταβλητής επί του πεδίου Z.
Τα παραπάνω είναι στην ουσία το πλαίσιο που ακολουθείται για την εύρεση μικρών ριζών πολυωνύμων όταν χρησιμοποιείται η μέθοδος των πινάκων.

8.4 Μικρές Λύσεις Τμηματικών Πολυωνύμων

Αν N είναι ένας θετικός ακέραιος με άγνωστη παραγοντοποίηση και $f_N(x) \in \square[x]$ πολυώνυμο βαθμού d. Ο σκοπός είναι να βρεθούν επαρκώς όλα τα $|x_0| < X$ που ικανοποιούν τη $f_N(x) \equiv (0 \bmod N)$, για όσο το μεγαλύτερο X είναι δυνατόν.

Η εργασίες των Hastad [97, 98] and Vallee, Girault και Toffin [242, 87] απέδειξαν ότι οι βασιζόμενες σε μήτρες μέθοδοι μπορούν να βρουν επαρκείς λύσεις για $|x_0| < X$ και για όρια όσο :

$$X = N^{\frac{2}{d(d+1)} - \varepsilon}$$

όπου το ε>0 είναι συνάρτηση του βαθμού d. Ουσιαστικά, η μέθοδος αυτή χρησιμοποιεί ελαχιστοποίηση βάσης μήτρας ώστε να βρει πολυώνυμο $h(x)$ με μικρούς συντελεστές, το οποίο είναι ταυτόχρονα και πολλαπλάσιο του $f_N(x) \bmod uloN$. Το διάνυσμα βάσης που χρησιμοποιείται για να κατασκευαστεί η μήτρα είναι τα διανύσματα συντελεστών των d+1 πολυωνύμων.

$$f_i(xX) = \begin{matrix} N(xX)^i & 0 \le i \le d-1 \\ f_N(xX) & i = d \end{matrix}$$

Για το πολυώνυμο βαθμού d της μορφής : $f_N(x) = a_0 + a_1 x + \ldots + a_{d-1} x^{d-1} + x^d$

χρησιμοποιείται η μήτρα βάσης :

$$\mathcal{B} = \begin{bmatrix} N & & & & & \\ & NX & & & & \\ & & NX^2 & & & \\ & & & \ddots & & \\ & & & & NX^{d-1} & \\ a_0 & a_1X & a_2X^2 & \cdots & a_{d-1}X^{d-1} & X^d \end{bmatrix}.$$

Κάθε στοιχείο του πίνακα Λ μπορεί να παραχθεί ως : $(-c_0, -c_1, \ldots, -c_{d-1}, c)B$,
το οποίο δίνεται από : $((ca_0 - c_0N), (ca_1 - c_1N)X, \ldots, (ca_{d-1} - c_{d-1}N)X^{d-1}, cX^d)$.

Αυτό το διάνυσμα είναι το διάνυσμα συντελεστών ενός πολυωνύμου $h(x)$ το οποίο δίνεται από : $h(x) = (ca_0 - c_0N) + (ca_1 - c_1N)x + \ldots + (ca_{d-1} - c_{d-1}N)x^{d-1} + cx^d$, όταν αυτό υπολογίζεται στο xX.
Πιο συγκεκριμένα, $h(x) \equiv cf_N(x) (\mathrm{mod}\, N)$ για οποιαδήποτε επιλογή c_i και c. Ως εκ τούτου, όλες οι ρίζες του $f_N(x) \bmod uloN$ είναι επίσης ρίζες του $h(x) \bmod uloN$. Ο υπολογισμός ενός μικρού διανύσματος της μήτρας αυτής αντιστοιχεί στη ταυτόχρονη ελαχιστοποίηση καθενός από τους συντελεστές του $h(x)$. Ο υπολογισμός μιας κατά LLL ελαχιστοποιημένης βάσης για τη μήτρα αυτή επιτρέπει το καθορισμό ενός ορίου όσον αφορά το μέγεθος του μικρότερου διανύσματος αυτής. Αυτό το αποτέλεσμα αντιστοιχίζεται με το όριο ενός πολυωνύμου της μορφής $h(xX)$.
Το όριο : $X = N^{2/(d(d+1))-\varepsilon}$, ονομάζεται *επιτρέπουσα συνθήκη* και για τη παραπάνω μέθοδο λαμβάνεται όταν ικανοποιείται η : $\|h(xX)\| \le 2^{d/4} vol(L)^{1/(d+1)}$, με $vol(L) = |\det(B)| = N^d X^{(d^2+d)/2}$.
Τελικά, για να ικανοποιείται και το όριο κατά Howgrave-Graham :

$$2^{d/4} N^{d/(d+1)} X^{(d^2+d)/(2(d+1))} \le (d+1)^{-1/2} N,$$

ή πιο απλά : $X^{(d^2+d)/2} \le \gamma N$, με $\gamma = 2^{d(d+1)/4}(d+1)^{-(d_1)/2}$.
Λύνοντας ως προς X : $X \le N^{2/(d(d+1)-\varepsilon)}$ με ε>0 να αντιστοιχεί στο γ.
Το παραπάνω όριο βελτιώθηκε από τον Coppersmith [51][52] ο οποίος απέδειξε ότι ρίζες μικρότερες από $N^{1/d-\varepsilon}$ μπορούσαν να υπολογιστούν αν ληφθούν υπόψη πολυωνυμικοί συνδυασμοί της $f_N(x) \bmod uloN^m$ όπου m τυχαίος ακέραιος και όχι πολλαπλάσια της $f_N(x) \bmod uloN$.

8.5 Ορισμός Μονόδρομης Συνάρτησης.

Είναι μια μαθηματική συνάρτηση η οποία είναι ευκολότερο να υπολογιστεί μόνο από μια κατεύθυνση (την κανονική) και είναι δύσκολο να αντιστραφεί. Μπορεί να είναι δυνατό για παράδειγμα, να υπολογιστεί η συνάρτηση στην κανονική της μορφή σε

δευτερόλεπτα, αλλά το αντίστροφό της θα χρειαζόταν μήνες ή χρόνια αν αυτή αντιστρέφεται. Η μονόδρομη συνάρτηση καταπακτή έχει το χαρακτηριστικό πως υπολογίζεται εύκολα από την ανάστροφη φορά αλλά όχι από την ορθή.
Τα κρυπτοσυστήματα δημοσίου κλειδιού βασίζονται σε (υποτιθέμενες) μονόδρομες συναρτήσεις –καταπακτής. Το δημόσιο κλειδί παρέχει πληροφορίες για τη συγκεκριμένη εφαρμογή της συνάρτησης ενώ το ιδιωτικό δίνει πληροφορίες για την «καταπακτή». Όποιος γνωρίζει την καταπακτή μπορεί να υπολογίσει τη συνάρτηση και από τις δυο κατευθύνσεις, αλλά όποιος δε διαθέτει την πληροφορία αυτή, μπορεί να υπολογίσει τη συνάρτηση μόνο κατά την ορθή φορά της η οποία χρησιμοποιείται για κρυπτογράφηση και επιβεβαίωση ψηφιακής υπογραφής. Η αντίστροφη κατεύθυνση –η πιο σημαντική για τους επιτιθέμενους– χρησιμοποιείται για αποκρυπτογράφηση και δημιουργία ψηφιακών υπογραφών.
Σχεδόν σε όλα τα κρυπτοσυστήματα δημοσίου κλειδιού, το μέγεθος του κλειδιού αντιστοιχεί στο μέγεθος της εισόδου της μονόδρομης συνάρτησης. Όσο μεγαλύτερο το κλειδί, τόσο μεγαλύτερη η διαφορά στον επεξεργαστικό χρόνο που απαιτείται για να υπολογιστεί συνάρτηση στην ορθή και στην αντίστροφη φορά για κάποιον που δεν έχει την «καταπακτή». Για παράδειγμα, για να παραμείνει ασφαλής για πολλά χρόνια μια ψηφιακή υπογραφή, είναι απαραίτητη η χρήση συνάρτησης –καταπακτής με εισόδους αρκετά μεγάλες ώστε κάποιος χωρίς την καταπακτή να χρειαστεί πολλά χρόνια για να υπολογίσει την αντίστροφη συνάρτηση και να φτιάξει μια νομότυπη υπογραφή.
Όλα τα πραγματικά κρυπτοσυστήματα δημοσίου κλειδιού βασίζονται σε συναρτήσεις που πιστεύεται ότι είναι μονόδρομες, αλλά καμία από αυτές δεν έχει αποδειχτεί επί του παρόντος ότι είναι πραγματική. Αυτό σημαίνει ότι είναι θεωρητικά πιθανό να ανακαλυφθούν αλγόριθμοι που θα μπορούν να υπολογίσουν την αντίστροφη συνάρτηση γρήγορα, χωρίς την ανάγκη καταπακτής. Μια τέτοια εξέλιξη θα καθιστούσε όλα τα κρυπτοσυστήματα που χρησιμοποιούν τέτοια μαθηματικά εργαλεία, άχρηστα. Από την άλλη, αναμένεται ότι περαιτέρω έρευνα στη θεωρία της επιστήμης υπολογιστών δύναται να καταλήξει σε πιο στέρεα κάτω όρια όσον αφορά τη δυσκολία αντιστροφής ορισμένων συναρτήσεων. Κάτι τέτοιο θα αποτελούσε γεγονός ορόσημο με σημαντική συνεισφορά στο πεδίο της κρυπτογραφίας.

Πηγή: http://www.rsa.com/rsalabs/node.asp?id=2188

9 Βιβλιογραφία

[1] R.L. Rivest, A. Shamir, and L.M. Adleman, "A Method for Obtaining Digital Signatures and Public-Key Cryptosystems," Communications of the ACM, v. 21, n. 2, Feb 1978, pp. 120-126.

[2] R.L. Rivest, A. Shamir, and L.M. Adleman, "On Digital Signatures and Public Key Cryptosystems," MIT Laboratory for Computer Science, Technical Report, MIT/LCS/TR-212, Jan 1979.

[3] M. Gardner, "A New Kind of Cipher That Would Take Millions of Years to Break," Scientific American, v. 237, n. 8, Aug 1977, pp. 120-124.

[4] C.K. Wu and X.M. Wang, "Determination of the True Value of the Euler Totient Function in the RSA Cryptosystem from a Set of Possibilities," Electronics Letters, v. 29, n. 1, 7 Jan 1993, pp. 84-85.

[5] W. Alexi, B.-Z. Chor, O. Goldreich, and C.P. Schnorr, "RSA and Rabin Functions: Certain Parts are as Hard as the Whole," SIAM Journal on Computing, v. 17, n. 2, Apr 1988, pp. 194-209.

[6] W.H. Payne, "Public Key Cryptography Is Easy to Break," William H. Payne, unpublished manuscript, 16 Oct 90.

[7] H. Hule and W.B. Müller, "On the RSA-Cryptosystem with Wrong Keys," Contributions to General Algebra 6, Vienna: Verlag Hölder-Pichler-Tempsky, 1988, pp. 103-109.

[8] G.I. Davida, "Chosen Signature Cryptanalysis of the RSA (MIT) Public Key Cryptosystem," Technical Report TR-CS-82-2, Department of EECS, University of Wisconsin, 1982.

[9] D.E. Denning, "Digital Signatures with RSA and Other Public-Key Cryptosystems," Communications of the ACM, v. 27, n. 4, Apr 1984, pp. 388-392.

[10] Y. Desmedt and A.M. Odlykzo, "A Chosen Text Attack on the RSA Cryptosystem and Some Discrete Logarithm Problems," Advances in Cryptology-CRYPTO '85 Proceedings, Springer-Verlag, 1986, pp. 516-522.

[11] G.J. Simmons, "A `Weak' Privacy Protocol Using the RSA Cryptosystem," Cryptologia, v. 7, n. 2, Apr 1983, pp. 180-182.

[12] J.M. DeLaurentis, "A Further Weakness in the Common Modulus Protocol for the RSA Cryptosystem," Cryptologia, v. 8, n. 3, Jul 1984, pp. 253-259.

[13] M. Bellare and P. Rogaway. Optimal asymmetric encryption. In EUROCRYPT '94, volume 950 of Lecture Notes in Computer Science, pages 92{111. Springer-Verlag, 1994.

[14] D. Bleichenbacher. Chosen ciphertext attacks against protocols based on the RSA encryption standard PKCS #1. In CRYPTO '98, volume 1462 of Lecture Notes in Computer Science, pages 1{12. Springer-Verlag, 1998.

[15] D. Boneh, R. DeMillo, and R. Lipton. On the importance of checking cryptographic protocols for faults. In EUROCRYPT '97, volume 1233 of Lecture Notes in Computer Science, pages 37{51. Springer-Verlag, 1997.

[16] D. Boneh and G. Durfee. New results on cryptanalysis of low private exponent RSA. Preprint, 1998.

[17] D. Boneh, G. Durfee, and Y. Frankel. An attack on RSA given a fraction of the private key bits. In AsiaCrypt '98, volume 1514 of Lecture Notes in Computer Science, pages 25{34. Springer-Verlag, 1998.

[18] D. Boneh and R. Venkatesan. Breaking RSA may not be equivalent to factoring. In EURO-CRYPT '98, volume 1403 of Lecture Notes in Computer Science, pages 59{71. Springer-Verlag, 1998.

[19] D. Coppersmith. Small solutions to polynomial equations, and low exponent RSA vulnerabilities. Journal of Cryptology, 10:233{260, 1997.

[20] D. Coppersmith, M. Franklin, J. Patarin, and M. Reiter. Low-exponent RSA with related messages. In EUROCRYPT '96, volume 1070 of Lecture Notes in Computer Science, pages 1{9. Springer-Verlag, 1996.

[21] W. de Jonge and D. Chaum. Attacks on some RSA signatures. In Crypto '85, volume 218 of Lecture Notes in Computer Science, pages 18{27. Springer-Verlag, 1986.

[22] Y. Desmedt and A. Odlyzko. A chosen text attack on the rsa cryptosystem and some discrete logarithm schemes. In CRYPTO '85, Lecture Notes in Computer Science, pages 516{522. Springer-Verlag, 1985.

[23] S. Goldwasser. The search for provably secure cryptosystems. In Cryptology and computational number theory, volume 42 of Proceedings of the 42nd Symposium in Applied Mathematics. American Mathematical Society, 1990.

[24] G. H. Hardy and E. M. Wright. An Introduction to the Theory of Numbers. Oxford Clarendon Press, 1975. fourth edition.

[25] J. Hastad. Solving simultaneous modular equations of low degree. SIAM J. of Computing, 336{341, 1988.

[26] N. Howgrave-Graham. Finding small roots of univariate modular equations revisited. In Cryptography and Coding, volume 1355 of Lecture Notes in Computer Science, pages 131{142. Springer-Verlag, 1997.

[27] M. Joye and J.-J. Quisquater. On the importance of securing your bins: The garbage-man-in-the-middle attack. In 4th ACM Conference on Computer and Communications Security, pages 135{141. ACM Press, 1997.

[28] P. Kocher. Timing attacks on implementations of Die-Hellman, RSA, DSS, and other systems. In CRYPTO '96, volume 1109 of Lecture Notes in Computer Science, pages 104{113. Springer-Verlag, 1996.

[29] L. Lovasz. An Algorithmic Theory of Number, Graphs and Convexity. SIAM Publications, 1986.

[30] K. Lenstra and H. W. Lenstra, Jr. Algorithms in number theory. In Handbook of Theoretical Computer Science (Volume A: Algorithms and Complexity), chapter 12, pages 673{715. Elsevier and MIT Press, 1990.

[31] Menezes, P. van Oorschot, and S. Vanstone. Handbook of Applied Cryptography. CRC, 1996.

[32] Pomerance. A tale of two sieves. Notices Amer. Math. Soc., 43:1473{1485, 1996.

[33] R. L. Rivest, A. Shamir, and L. Adleman. A method for obtaining digital signatures and public key cryptosystems. Commun. of the ACM, 21:120{126, 1978.

[34] M. Wiener. Cryptanalysis of short RSA secret exponents. IEEE Transactions on InformationTheory, 36:553{558, 1990.

[35] Berzati, A., Canovas, C., Goubin, L.: Perturbating RSA Public Keys:An Improved Attack. In: Oswald, E., Rohatgi, P. (eds.) CHES 2008.LNCS, vol. 5154, pp. 380–395. Springer, Heidelberg (2008)

[36] Brier, ´E., Chevallier-Mames, B., Ciet, M., Clavier, C.: Why One Should Also Secure RSA Public Key Elements. In: Goubin, L., Matsui, M. (eds.) CHES 2006. LNCS, vol. 4249, pp. 324–338. Springer, Heidelberg (2006)

[37] Bao, F., Deng, R.H., Jeng, A., Narasimhalu, A.D., Ngair, T.: BreakingPublic Key Cryptosystems on Tamper Resistant Devices in the Presence of Transient Faults. In: Christianson, B., Lomas, M. (eds.) Security Protocols 1997. LNCS, vol. 1361, pp. 115–124. Springer, Heidelberg (1998)

[38] Boneh, D., DeMillo, R.A., Lipton, R.J.: On the Importance of CheckingCryptographic Protocols for Faults. In: Fumy, W. (ed.) EUROCRYPT 1997. LNCS, vol. 1233, pp. 37–51. Springer, Heidelberg (1997)

[39] Boneh, D., DeMillo, R.A., Lipton, R.J.: On the Importance of Eliminating Errors in Cryptographic Computations. Journal of Cryptology 14(2), 101–119 (2001)

[40] Bar-El, H., Choukri, H., Naccache, D., Tunstall, M., Whelan, C.: The Sorcerer's Apprentice Guide to Fault Attacks. Cryptology ePrint Archive, Report 2004/100 (2004)

[41] Bl¨omer, J., Otto, M.: Wagner's Attack on a secure CRT-RSA Algorithm Reconsidered. In: Breveglieri, L., Koren, I., Naccache, D., Seifert, J.-P.(eds.) FDTC 2006. LNCS, vol. 4236, pp. 13–23. Springer, Heidelberg(2006)

[42] Bl¨omer, J., Otto, M., Seifert, J.-P.: A New CRT-RSA Algorithm Secure Against Bellcore Attack. In: ACM Conference on Computer and Communication Security (CCS 2003), pp. 311–320. ACM Press, New York (2003)

[43] Biham, E., Shamir, A.: Differential Fault Analysis of Secret Key Cryptosystems.In: Kaliski Jr., B.S. (ed.) CRYPTO 1997. LNCS, vol. 1294, pp. 513–525. Springer, Heidelberg (1997)

[44] Clavier, C.: De la s´ecurit´e physique des crypto-syst`emes embarqu´es. PhD thesis, Universit´e de Versailles Saint-Quentin (2007)

[45] Cohen, H.: A Course in Computational Algebraic Number Theory. Springer, New York (1993)

[46] Dusart, P.: Autour de la fonction qui compte le nombre de nombres premiers. PhD thesis, Universit´e de Limoges (1998)

[47] Giraud, C.: DFA on AES. In: Dobbertin, H., Rijmen, V., Sowa, A. (eds.) AES 2005. LNCS, vol. 3373, pp. 27–41. Springer, Heidelberg (2005)

[48] Giraud, C.: Fault-Resistant RSA Implementation. In: Breveglieri, L., Koren, I. (eds.) Fault Diagnosis and Tolerance in Cryptography, pp. 142–151 (2005)

[49] Muir, J.A.: Seifert's RSA Fault Attack: Simplified Analysis and Generalizations. Cryptology ePrint Archive, Report 2005/458 (2006)

[50] Rabin, M.O.: Probabilistic algorithm for testing primality. Journal ofNumber Thoery 12(1), 128–138 (1980)

[51] Seifert, J.-P.: On Authenticated Computing and RSA-Based Authentication.In: ACM Conference on Computer and Communications Security (CCS 2005), pp. 122–127. ACM Press, New York (2005)

[52] Shoup, V.: A Computational Introduction to Number Theory and Algebra. Cambridge University Press, Cambridge (2005)

[53] Wagner, D.: Cryptanalysis of a provably secure CRT-RSA algorithm. In: Proceedings of the 11th ACM Conference on Computer Security (CCS 2004), pp. 92–97. ACM Press, New York (2004)

[54] Bellare, M., Rogaway, P.: The Exact security of digital signatures: How to sign with rsa and Rabin. In: Maurer, U.M. (ed.) EUROCRYPT 1996. LNCS, vol. 1070, pp.399–416. Springer, Heidelberg (1996)

[55] Boneh, D., DeMillo, R.A., Lipton, R.J.: On the importance of checking cryptographic protocols for faults. Journal of Cryptology 14(2), 101–119 (2001)

[56] Coppersmith, D.: Small solutions to polynomial equations, and low exponent vulnerabilities. Journal of Cryptology 10(4), 233–260 (1997)

[57] Coron, J.-S., Joux, A., Kizhvatov, I., Naccache, D., Paillier, P.: Fault attacks on rsa signatures with partially unknown messages. In: Clavier, C., Gaj, K. (eds.) CHES 2009. LNCS, vol. 5747, pp. 444–456. Springer, Heidelberg (2009), eprint.iacr.org/2009/309

[58] Coron, J.-S., Naccache, D., Stern, J.P.: On the security of RSA padding. In: Wiener, M. (ed.) CRYPTO 1999. LNCS, vol. 1666, pp. 1–18. Springer, Heidelberg (1999)

[59] Coron, J.-S., Naccache, D., Tibouchi, M., Weinmann, R.P.: Practical cryptanalysis of iso/iec 9796-2 and emv signatures. In: Halevi, S. (ed.) Advances in Cryptology- CRYPTO 2009. LNCS, vol. 5677, pp. 428–444. Springer, Heidelberg (2009), eprint.iacr.org/2009/203

[60] Coron, J.-S.: Optimal security proofs for pss and other signature schemes. In: Knudsen, L.R. (ed.) EUROCRYPT 2002. LNCS, vol. 2332, pp. 272–287. Springer, Heidelberg (2002)

[61] Coron, J.-S., Joye, M., Naccache, D., Paillier, P.: Universal padding schemes for RSA. In: Yung, M. (ed.) CRYPTO 2002. LNCS, vol. 2442, pp. 226–241. Springer, Heidelberg (2002)

[62] Herrmann, M., May, A.: Solving linear equations modulo divisors: On factoring given any bits. In: Pieprzyk, J. (ed.) ASIACRYPT 2008. LNCS, vol. 5350, pp.406–424. Springer, Heidelberg (2008)

[63] EMV, Integrated circuit card specifications for payment systems, Book 2. Security and Key Management. Version 4.2 (June 2008), http://www.emvco.com

[64] emv, EMVCo type approval terminal level 2 test cases. Version 4.2a (April 2009), http://www.emvco.com

[65] iso/iec 8825-1:2002, Information technology – ASN.1 encoding rules: Specification of Basic Encoding Rules (BER), Canonical Encoding Rules (CER) and Distinguished Encoding Rules (DER) (2002)

[66] iso/iec 9796-2, Information technology – Security techniques – Digital signature schemes giving message recovery – Part 2: Mechanisms using a hash-funcion (1997)

[67] iso/iec 9796-2:2002 Information technology – Security techniques – Digital signature schemes giving message recovery– Part 2: Integer factorization based mechanisms (2002)

[68] Joye, M., Lenstra, A., Quisquater, J.-J.: Chinese remaindering cryptosystems in the presence of faults. Journal of Cryptology 21(1), 27–51 (1999)

[69] Nguyen, P., Stern, J.: Cryptanalysis of a fast public key cryptosystem presented at sac 1997. In: Tavares, S., Meijer, H. (eds.) SAC 1998. LNCS, vol. 1556, pp. 213–218. Springer, Heidelberg (1999)

[70] Nguyen, P., Stern, J.: Merkle-Hellman revisited: a cryptanalysis of the Qu-Vanstone cryptosystem based on group factorization. In: Kaliski Jr., B.S. (ed.) CRYPTO 1997. LNCS, vol. 1294, pp. 198–212. Springer, Heidelberg (1997)

[71] Lenstra, A., Lenstra Jr., H., Lov´asz, L.: Factoring polynomials with rational coefficients. In: Mathematische Annalen, vol. 261, pp. 513–534. Springer, Heidelberg (1982)

[72] Fujisaki, E., Okamoto, T., Pointcheval, D., Stern, J.: rsa-oaep is secure under the rsa assumption. Journal of Cryptology 17(2), 81–104 (2004)

[73] Rivest, R., Shamir, A., Adleman, L.: A method for obtaining digital signatures and public key cryptosystems. Communications of the acm, 120–126 (1978)